# Ozeki 10 SMS Gateway

Product Guide

sms.ozeki.hu

# Table of Contents

# Quick Start Guide - SMS Gateway

***This Quick start guide is about to explain that how to send your first SMS in just a few steps with Ozeki 10 SMS Gateway. This guide provides you the capability to configure your SMS modem to send or receive an SMS message successfully. The following steps will guide you through the process of login, creating users or connections and routing your text messages.***

## Video tutorial

## Requirements

**SIM card:** To use all the functionality of Ozeki 10 SMS Gateway, first you need a mobile subscription that comes with a SIM card. This SIM card will provide you a telephone number, that is crucial to send and receive SMS messages with Ozeki 10 SMS Gateway.

**GSM modem:** A special type of modem, that accepts SIM cards. That modem can be connected to your PC via a data cable and use it as a broadcaster for your messages. The GSM modem is avaiable directly from Ozeki Ltd.

**Data cable:** To connect your GSM modem to the PC, you just need a simple USB data cable that is provided for each modems that has been ordered from Ozeki Ltd.

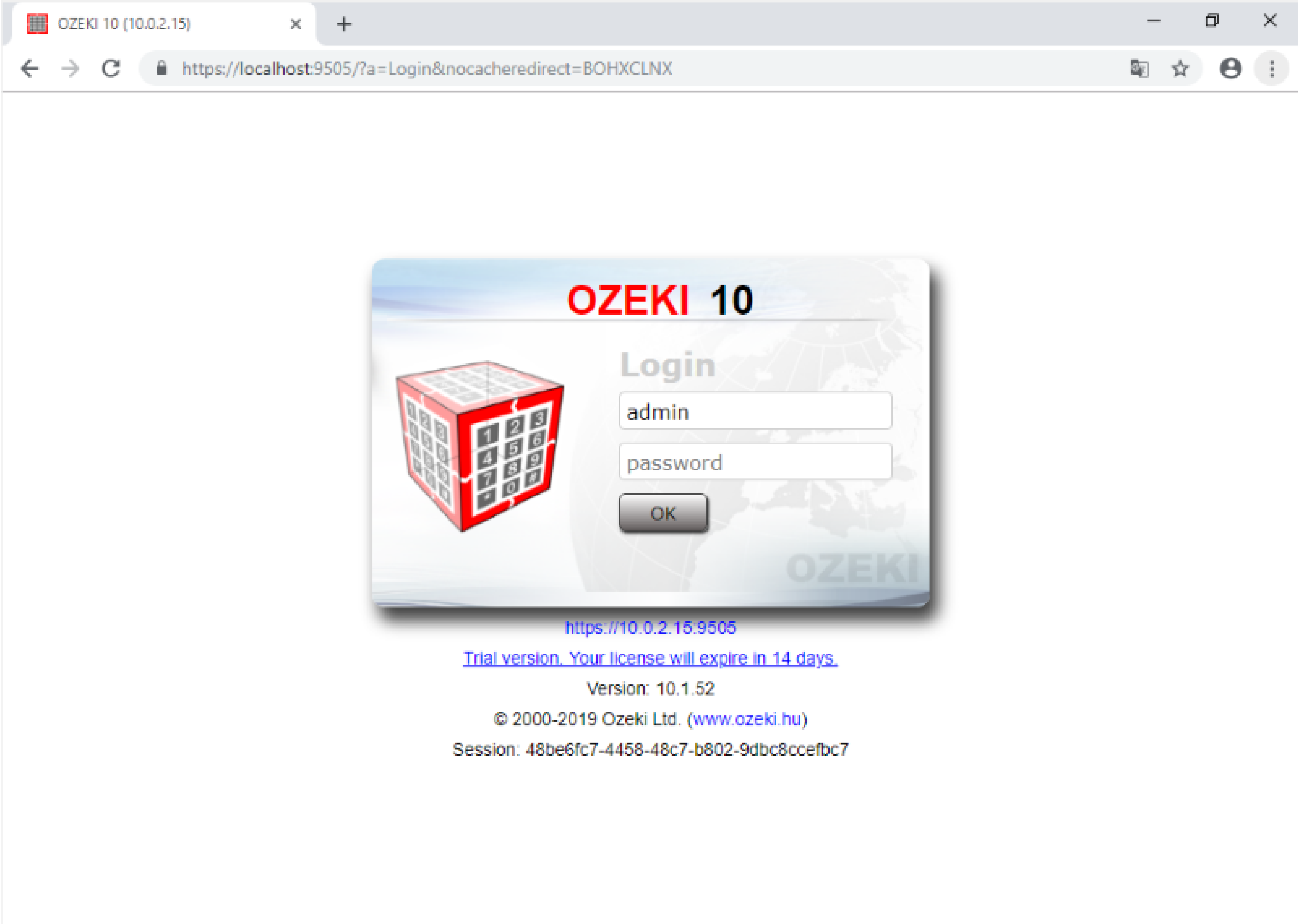
**Ozeki 10 SMS Gateway:** You need to download Ozeki 10 SMS Gateway from http://www.sms.ozeki.hu/index.php?owpn=727 and install it on your computer.

Ozeki 10 SMS Gateway can be obtained by from the download page:
**Download Ozeki 10 SMS Gateway!**

Setup instructions for the modem are available at:
**Installation of an SMS modem**

## Step 1: Log in to your account

The first thing that you need to do after the installation to log in to your account that you created during the installation. For that just click on the Ozeki Desktop icon and your Ozeki 10 SMS Gateway will open up in your web browser (**Figure 1**).



OZEKI 10 (10.0.2.15)

https://localhost:9505/?a=Login&nocacheredirect=BOHXCLNX

OZEKI 10

Login

admin

password

OK

https://10.0.2.15:9505

Trial version. Your license will expire in 14 days.

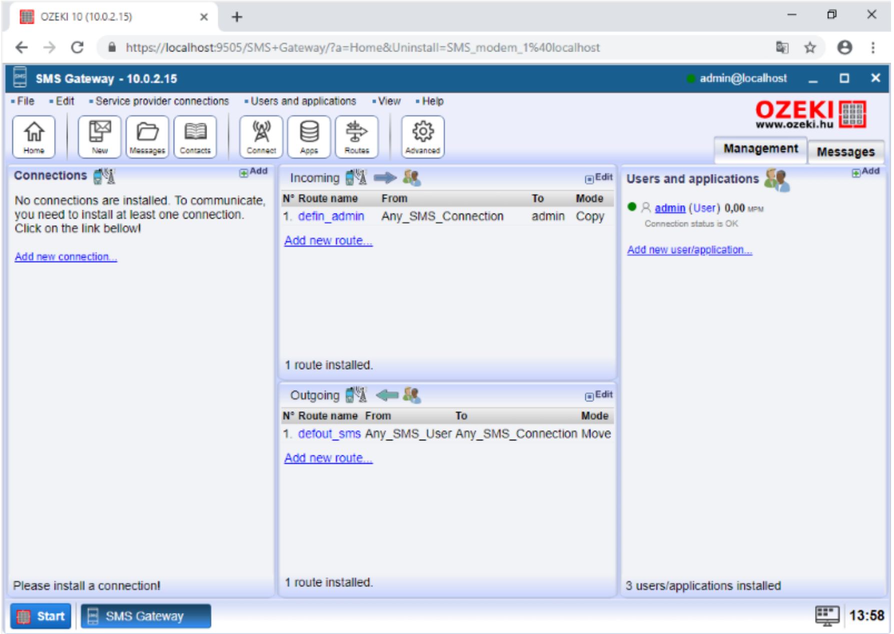
Version: 10.1.52

© 2000-2019 Ozeki Ltd. (www.ozeki.hu)

Session: 48be6fc7-4458-48c7-b802-9dbc8ccefbc7

Figure 1 - Log in for the first time

In the login window just enter the username which is 'admin' by default and your password that you specified during the installation. After you pressed **OK**, the SMS Gateway starts automatically with an opening window as **Figure 2** demonstrates it. Here there are four main panels that collect all the main features of the SMS Gateway you need to handle SMS messages. If you want to change the password of the default 'admin' user, you can easily do that by following the quick guide on the link.



OZEKI 10 (10.0.2.15)

https://localhost:9505/SMS+Gateway/?a=Home&Uninstall=SMS_modem_1%40localhost

SMS Gateway - 10.0.2.15 admin@localhost

Connections

No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

Please install a connection!

Incoming

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defin_admin | Any_SMS_Connection | admin | Copy |

Add new route...

1 route installed.

Outgoing

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defaut_sms | Any_SMS_User | Any_SMS_Connection | Move |

Add new route...

1 route installed.

Users and applications

admin (User) 0,00 MPM

Connection status is OK

Add new user/application...

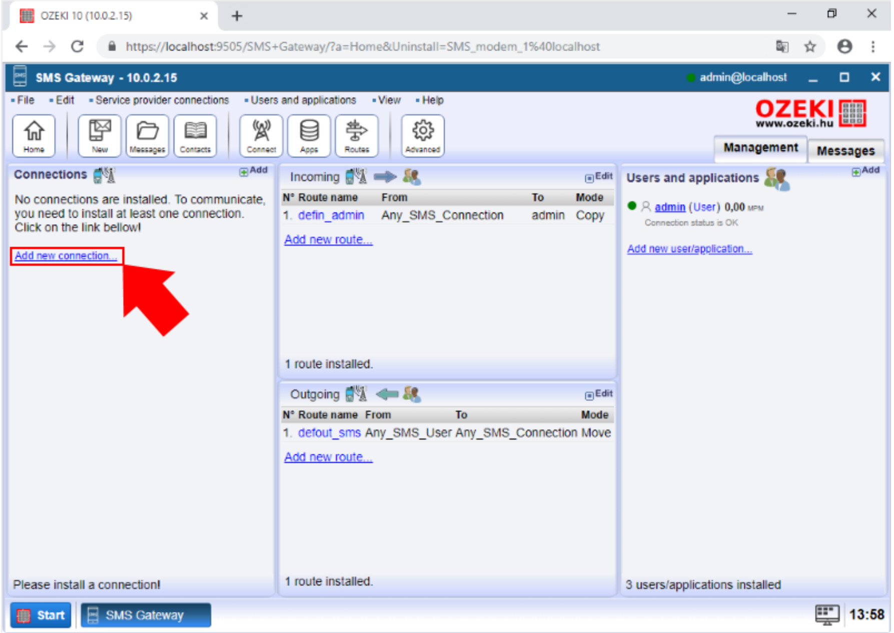
3 users/applications installed

Figure 2 - Opening window of the SMS Gateway

## Step 2: Set up your mobile network connection

The next step is to connect your system to the mobile network. To do this there are two options. You can connect to the mobile network like any other mobile phone does, in this case you need to connect a GSM modem or you need to connect through an Androi mobile phone. The other option is to connect to the Short Message Service Center (SMSC) of a mobile network operator over the Internet. In this case you would setup an SMPP connection, an UCP connection a CIMD2 connection or an HTTP SMS connection.

The most simple option is to connect your computer to the mobile network using an GSM/SMS modem. An SMS modem is s simplified mobile phone. You can put a SIM card in it and it has an antenna. In this case Ozeki 10 SMS Gateway will communicate with you modem over a data cable and will receive incoming SMS and will send outgoing messages through the wireless link provided by the modem. The way you will do that is very simple, you only need to create a connection that can manage the GSM modem and use to send and receive SMS messages. To create that connection just select **Add new connection...** like in **Figure 3**.



Connections — Add

No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

Please install a connection!

Incoming — Edit

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defin_admin | Any_SMS_Connection | admin | Copy |

Add new route...

1 route installed.

Outgoing — Edit

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move |

Add new route...

1 route installed.

Users and applications — Add

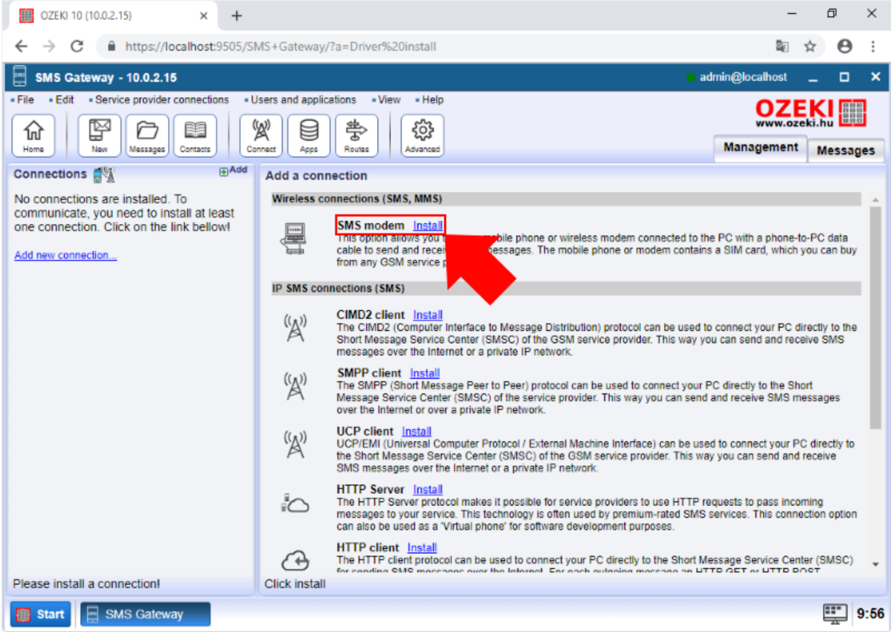
● admin (User) 0,00 MPM
Connection status is OK

Add new user/application...

3 users/applications installed

Figure 3 - Choose **Add new connection** to create one

Next, as **Figure 4** shows that, you can choose from numerous connections since Ozeki 10 SMS Gateway supports many protocol types. But now, to follow this guide you need to create a wireless connection, more specifically an SMS modem connection that can be performed by clicking on the **Install** button.



Connections — Add

No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

Please install a connection!

Add a connection

Wireless connections (SMS, MMS)

SMS modem Install
This option allows you t[...]bile phone or wireless modem connected to the PC with a phone-to-PC data cable to send and recei[...]essages. The mobile phone or modem contains a SIM card, which you can buy from any GSM service p

IP SMS connections (SMS)

CIMD2 client Install
The CIMD2 (Computer Interface to Message Distribution) protocol can be used to connect your PC directly to the Short Message Service Center (SMSC) of the GSM service provider. This way you can send and receive SMS messages over the Internet or a private IP network.

SMPP client Install
The SMPP (Short Message Peer to Peer) protocol can be used to connect your PC directly to the Short Message Service Center (SMSC) of the service provider. This way you can send and receive SMS messages over the Internet or over a private IP network.

UCP client Install
UCP/EMI (Universal Computer Protocol / External Machine Interface) can be used to connect your PC directly to the Short Message Service Center (SMSC) of the GSM service provider. This way you can send and receive SMS messages over the Internet or a private IP network.

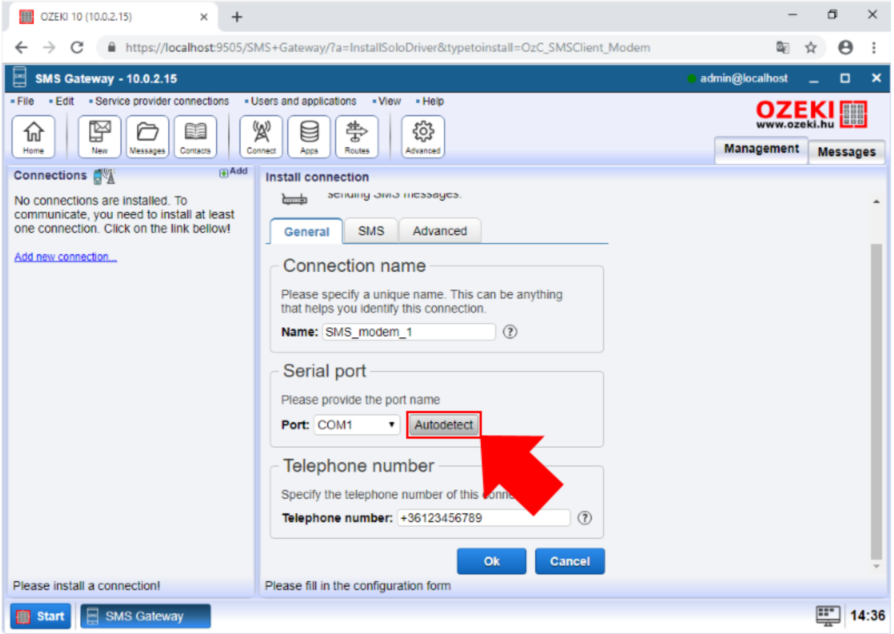
HTTP Server Install
The HTTP Server protocol makes it possible for service providers to use HTTP requests to pass incoming messages to your service. This technology is often used by premium-rated SMS services. This connection option can also be used as a 'Virtual phone' for software development purposes.

HTTP client Install
The HTTP client protocol can be used to connect your PC directly to the Short Message Service Center (SMSC) for sending SMS messages over the Internet. For each outgoing message an HTTP GET or HTTP POST

Click install

Figure 4 - Avaiable connections in Ozeki 10 SMS Gateway

Before finishing the creation of the connection, you can give a name to the connection and select the port, where you connected the SMS modem. If you are not sure about specific number of the port, just click on the **Autodetect** button (**Figure 5**), and a few moments later, the system finds that for you. Lastly you have to enter the telephone number of the SIM card that you plugged into the SMS modem. To finish the creation just click on OK.



No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

Connection name

Please specify a unique name. This can be anything that helps you identify this connection.

Name: SMS_modem_1

Serial port

Please provide the port name

Port: COM1 Autodetect

Telephone number

Specify the telephone number of this conne...

Telephone number: +36123456789

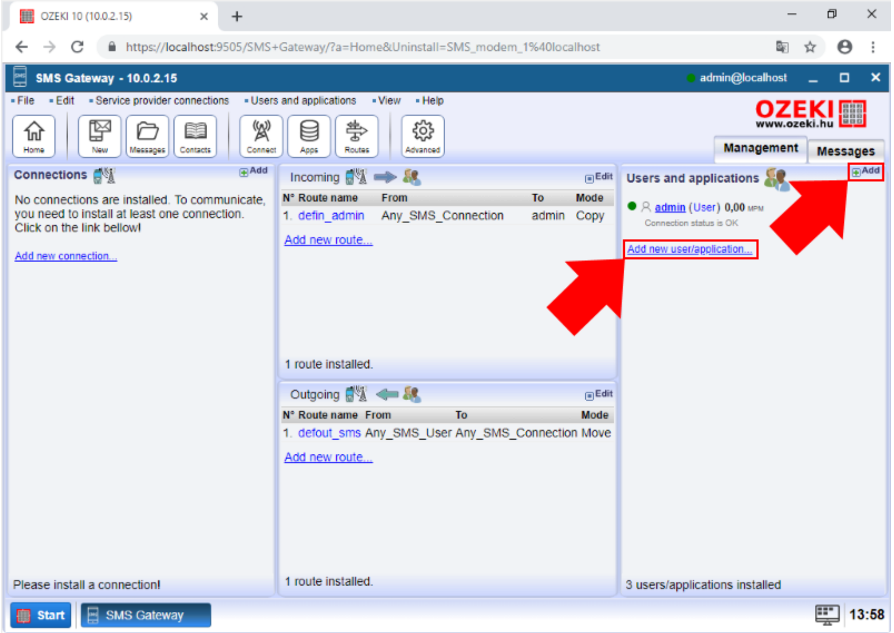
Please install a connection!

Please fill in the configuration form

Figure 5 - Autodetect the port where you plugged the GSM modem

# Step 3: Create new user or application

Ozeki 10 SMS Gateway provides you the functionality of handling multiple users, so the messages can be seperated by the different users. To add a new user just select the **Add new user/application...** or click on the **Add** button on the top of the Users and applications panel as you can see in **Figure 6**.



No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

Please install a connection!

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defin_admin | Any_SMS_Connection | admin | Copy |

Add new route...

1 route installed.

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move |

Add new route...

1 route installed.

admin (User) 0,00 MPM

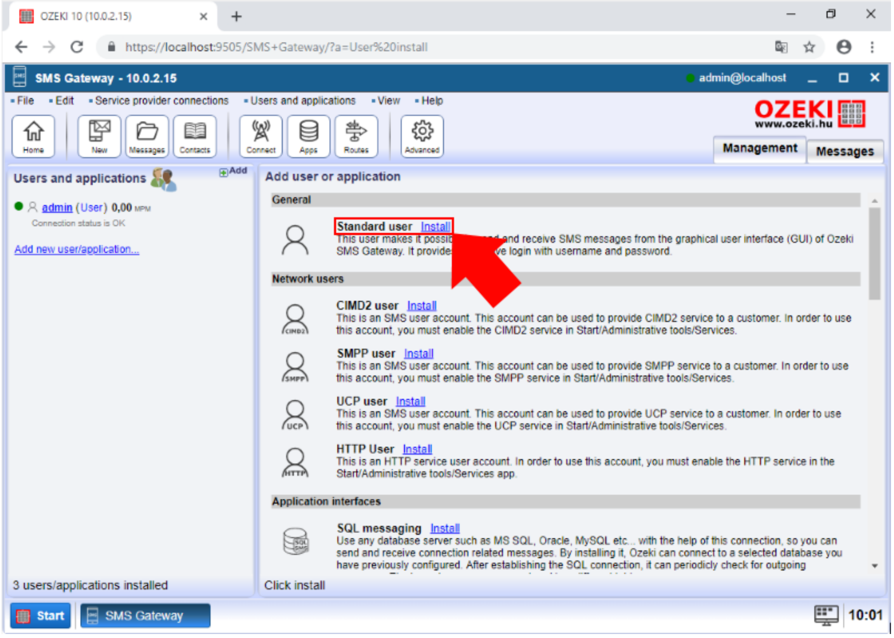
Connection status is OK

Add new user/application...

3 users/applications installed

Figure 6 - Two ways of entering to the creation of a new user or application

Next, you can choose between various types of users and applications. Here you can create a genaral or a network user, select from numerous application interfaces and also you can create services that can handle the incoming messages. If you follow this guide you will create a standard user by selecting the **Standard user** like in **Figure 7**.



admin (User) 0,00 MPM
Connection status is OK

Add new user/application...

3 users/applications installed

Add user or application

General

Standard user Install
This user makes it possib... and receive SMS messages from the graphical user interface (GUI) of Ozeki SMS Gateway. It provide... ve login with username and password.

Network users

CIMD2 user Install
This is an SMS user account. This account can be used to provide CIMD2 service to a customer. In order to use this account, you must enable the CIMD2 service in Start/Administrative tools/Services.

SMPP user Install
This is an SMS user account. This account can be used to provide SMPP service to a customer. In order to use this account, you must enable the SMPP service in Start/Administrative tools/Services.

UCP user Install
This is an SMS user account. This account can be used to provide UCP service to a customer. In order to use this account, you must enable the UCP service in Start/Administrative tools/Services.

HTTP User Install
This is an HTTP service user account. In order to use this account, you must enable the HTTP service in the Start/Administrative tools/Services app.

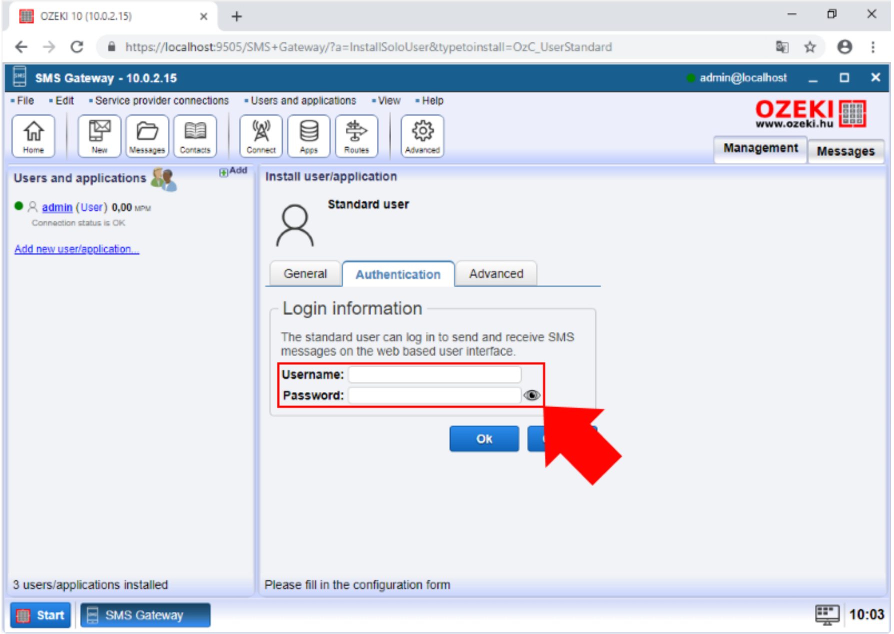
Application interfaces

SQL messaging Install
Use any database server such as MS SQL, Oracle, MySQL etc... with the help of this connection, so you can send and receive connection related messages. By installing it, Ozeki can connect to a selected database you have previously configured. After establishing the SQL connection, it can periodicly check for outgoing

Click install

Figure 7 - List of creatable users and applications

In the installation menu you can name the user at the **Genaral** panel. This name will show up in the list of users. In the **Authentication** tab (**Figure 8**) you can secure the account by entering a unique username and a password. The rules of the password are the same as it was at the creation of the 'admin' account. To finish the creation just click on **OK**.



admin (User) 0,00 MPM
Connection status is OK

Add new user/application...

3 users/applications installed

Install user/application

Standard user

Login information

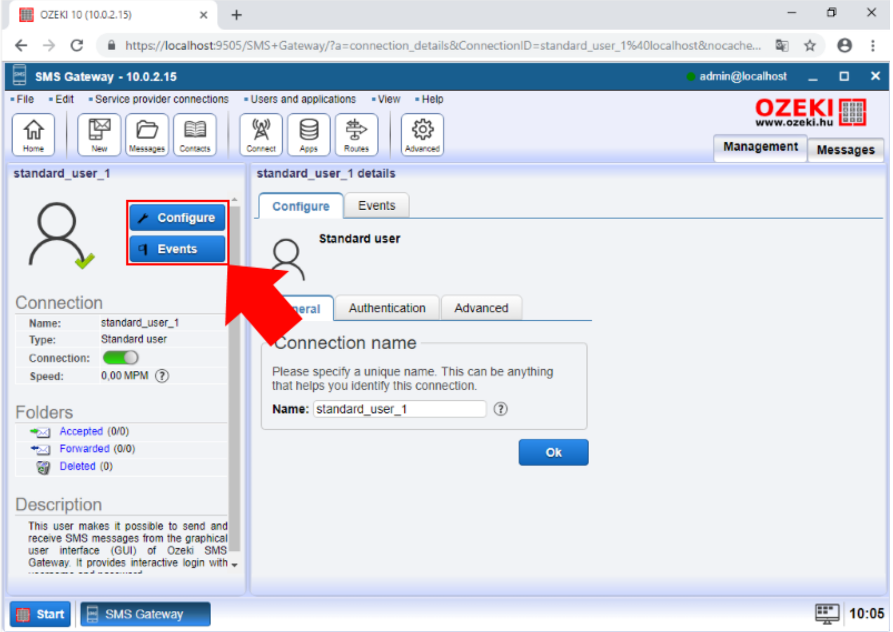
The standard user can log in to send and receive SMS messages on the web based user interface.

Username:
Password:

Please fill in the configuration form

Figure 8 - Select a username and password for the account

Now, you can see the created user in the Users and applications panel. To modify the parameters of the account, just select the user, and you will be able to do that in **Configure** tab (**Figure 9**). To view the events related to the user, select the **Events** tab.


OZEKI 10 (10.0.2.15)
https://localhost:9505/SMS+Gateway/?a=connection_details&ConnectionID=standard_user_1%40localhost&nocache...
SMS Gateway - 10.0.2.15
admin@localhost
File Edit Service provider connections Users and applications View Help
Home New Messages Contacts Connect Apps Routes Advanced
Management Messages
Configure
Events
Standard user
General Authentication Advanced
Ok
Start SMS Gateway 10:05

standard_user_1

Connection

| | |
|---|---|
| Name: | standard_user_1 |
| Type: | Standard user |
| Connection: | |
| Speed: | 0.00 MPM |

Folders

Accepted (0/0)
Forwarded (0/0)
Deleted (0)

Description

This user makes it possible to send and receive SMS messages from the graphical user interface (GUI) of Ozeki SMS Gateway. It provides interactive login with

standard_user_1 details

Connection name

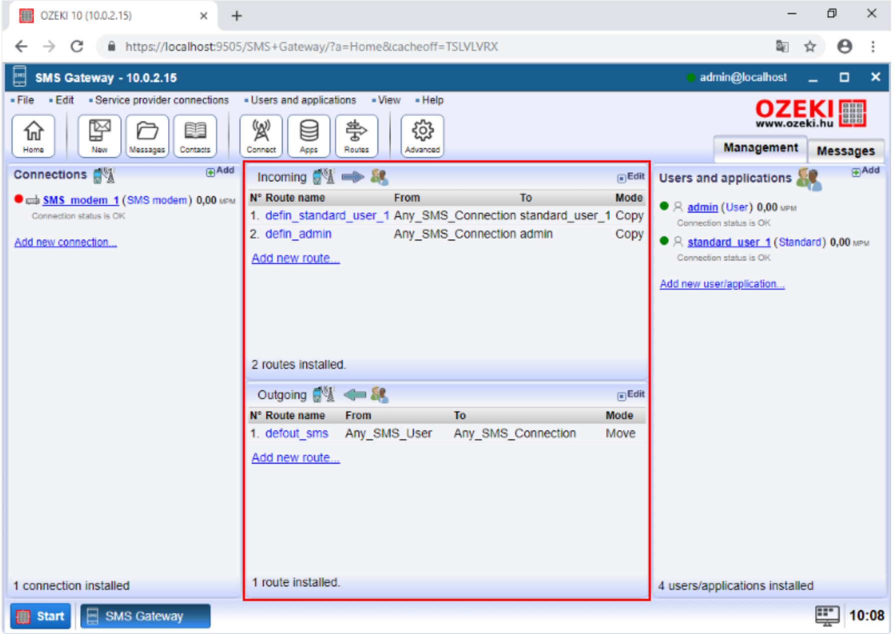
Please specify a unique name. This can be anything that helps you identify this connection.

Name: standard_user_1

Figure 9 - Configuration menu of the Standard user

# Step 4: Routing your messages

With the installation and the creation of the default 'admin' account, Ozeki 10 SMS Gateway also created a default routing rules for the incoming and outgoing messages that can be seen in the **Incoming** and the **Outgoing** panels like in **Figure 10**. But of course, you have the opportunity to define your own routing protocols.


OZEKI 10 (10.0.2.15)
https://localhost:9505/SMS+Gateway/?a=Home&cacheoff=TSLVLVRX
SMS Gateway - 10.0.2.15
admin@localhost
File Edit Service provider connections Users and applications View Help
Home New Messages Contacts Connect Apps Routes Advanced
Management Messages
Add
Edit
Start SMS Gateway 10:08

Connections

SMS_modem_1 (SMS modem) 0,00 MPM
Connection status is OK

Add new connection...

1 connection installed

Incoming

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defin_standard_user_1 | Any_SMS_Connection | standard_user_1 | Copy |
| 2. | defin_admin | Any_SMS_Connection | admin | Copy |

Add new route...

2 routes installed.

Outgoing

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move |

Add new route...

1 route installed.

Users and applications

admin (User) 0,00 MPM
Connection status is OK

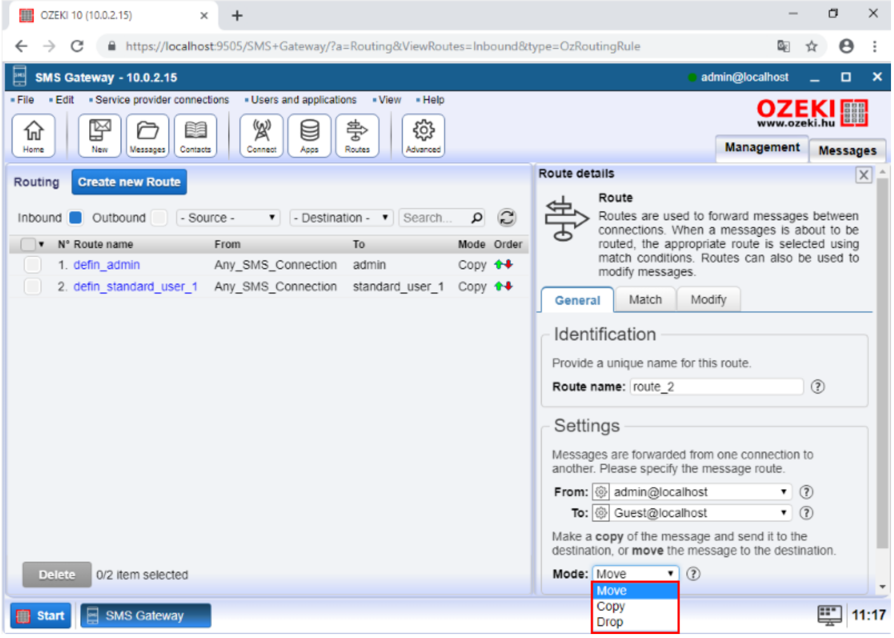
standard_user_1 (Standard) 0,00 MPM
Connection status is OK

Add new user/application...

4 users/applications installed

Figure 10 - Routing tables in Ozeki 10 SMS Gateway

**Inbound routing:** The incoming messages can be destined to one or more users. There is a default route that was created at the installation process, but you can easily set your own routing rules. For that, just select the **Add new route...** option and now you will be able to add a new route or configure the existing routes as well. If you would like to create a new route, first of all you have to define which user or connection send the message to which user or connection. The other main feature that you need to setup, is the mode of the route. There are three options to choose from: Move, Copy or Drop (**Figure 11**). Now, if you click on **OK**, the freshly created route will appear in the list of the routes.



| N° Route name | From | To | Mode | Order |
|---|---|---|---|---|
| 1. defin_admin | Any_SMS_Connection | admin | Copy | |
| 2. defin_standard_user_1 | Any_SMS_Connection | standard_user_1 | Copy | |

Route

Routes are used to forward messages between connections. When a messages is about to be routed, the appropriate route is selected using match conditions. Routes can also be used to modify messages.

Identification

Provide a unique name for this route.

Route name: route_2

Settings

Messages are forwarded from one connection to another. Please specify the message route.

From: admin@localhost

To: Guest@localhost

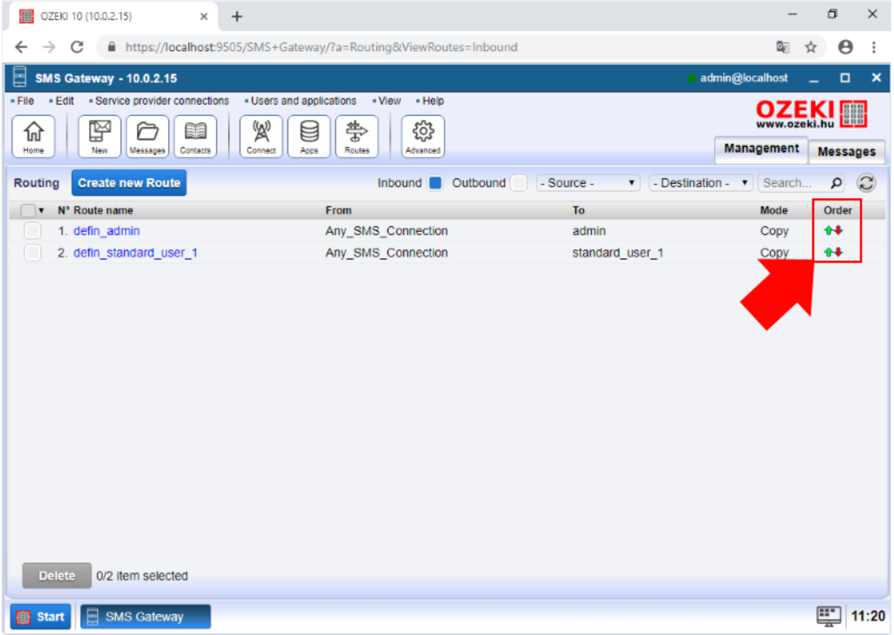
Make a **copy** of the message and send it to the destination, or **move** the message to the destination.

Mode: Move

Move
Copy
Drop

Figure 11 - Select a mode for the routing rule

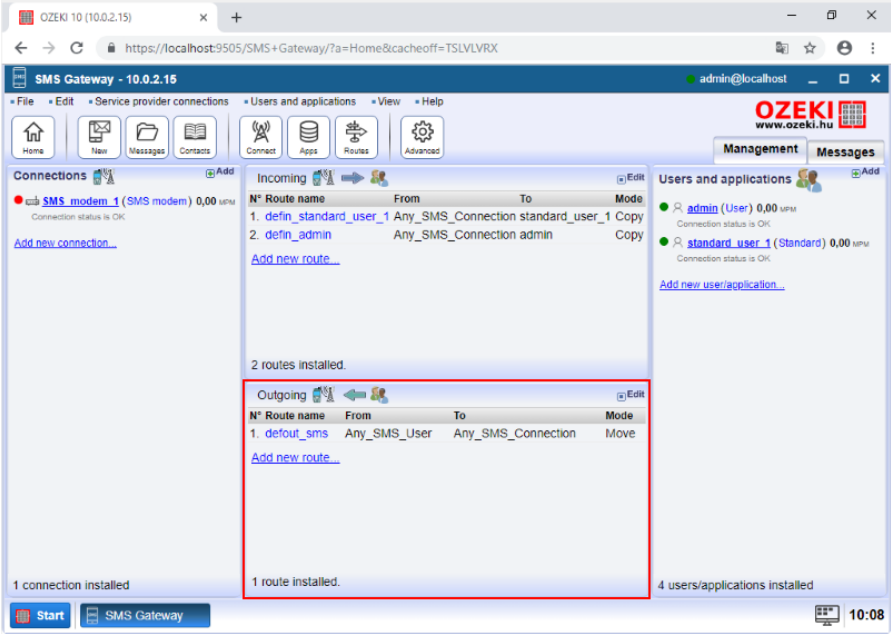
Now, if you take a look at your routing table you will be able to see that your routes list up in order. So, that means in case of an incoming messages, the program tries to activate a first route of the table. If there is a match of the rules, the messages will be delivered to the user specified in the rule. But if the parameters do not meet the criteria the program will go on to the next rule. If a rule matched with the criteria, there are two ways the routing can continue. If the matched rule has the mode **Copy**, the program will continue going through the subsequent routing rules. The other option is that the rule is in **Move** mode. In that case the routing will finish and the subsequent rules will be ignored. The order of the rules can be easily changed as **Figure 12** shows by clicking on the icons in the **Order** column.



| N° Route name | From | To | Mode | Order |
|---|---|---|---|---|
| 1. defin_admin | Any_SMS_Connection | admin | Copy | |
| 2. defin_standard_user_1 | Any_SMS_Connection | standard_user_1 | Copy | |

Figure 12 - Change the order of the routing rules

**Outbound routing:** An outgoing message can be sent using one of the service providers you are connected to. With the outbound routing you can configure the rules for the program to decide which service provider connection to use to send out a message. The routing rules are the same as it was at the inbound routing, so the program goes through the routing table from top to bottom. To create a new route, just select the **Add new route...** on the **Outgoing** panel (**Figure 13**).



Connections

SMS_modem_1 (SMS modem) 0,00 MPM
Connection status is OK

Add new connection...

1 connection installed

Incoming

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defin_standard_user_1 | Any_SMS_Connection | standard_user_1 | Copy |
| 2. | defin_admin | Any_SMS_Connection | admin | Copy |

Add new route...

2 routes installed.

Outgoing

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move |

Add new route...

1 route installed.

Users and applications

admin (User) 0,00 MPM
Connection status is OK

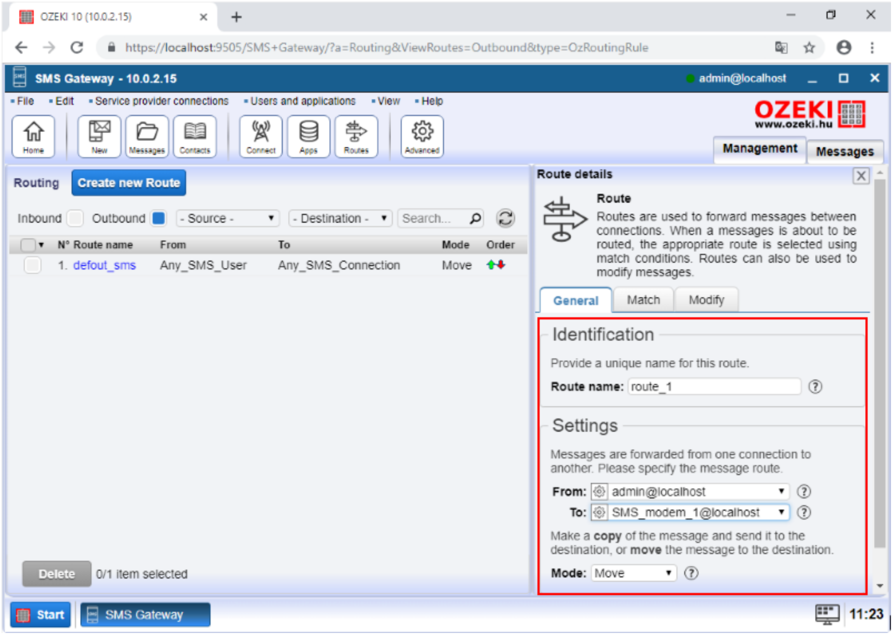
standard_user_1 (Standard) 0,00 MPM
Connection status is OK

Add new user/application...

4 users/applications installed

Figure 13 - Routing table of the outgoing messages

Here in the creation panel (**Figure 14**), the setting up of the route is pretty much the same as at the creation of the inbound route. That means the main parameter that you have to define is the two endpoints of the route. With that you can set the connection or user where the messages will be sent from, and also the recipient user or connection that will recieve the message. The other main thing that can be modified is the mode which can be Move, Copy or Drop. When you finished with the setting up, just click on **OK** to create your own route for the outgoing messages.



| N° | Route name | From | To | Mode | Order |
|---|---|---|---|---|---|
| 1. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move | |

0/1 item selected

Route details

Route

Routes are used to forward messages between connections. When a messages is about to be routed, the appropriate route is selected using match conditions. Routes can also be used to modify messages.

Identification

Provide a unique name for this route.

Route name: route_1

Settings

Messages are forwarded from one connection to another. Please specify the message route.

From: admin@localhost

To: SMS_modem_1@localhost

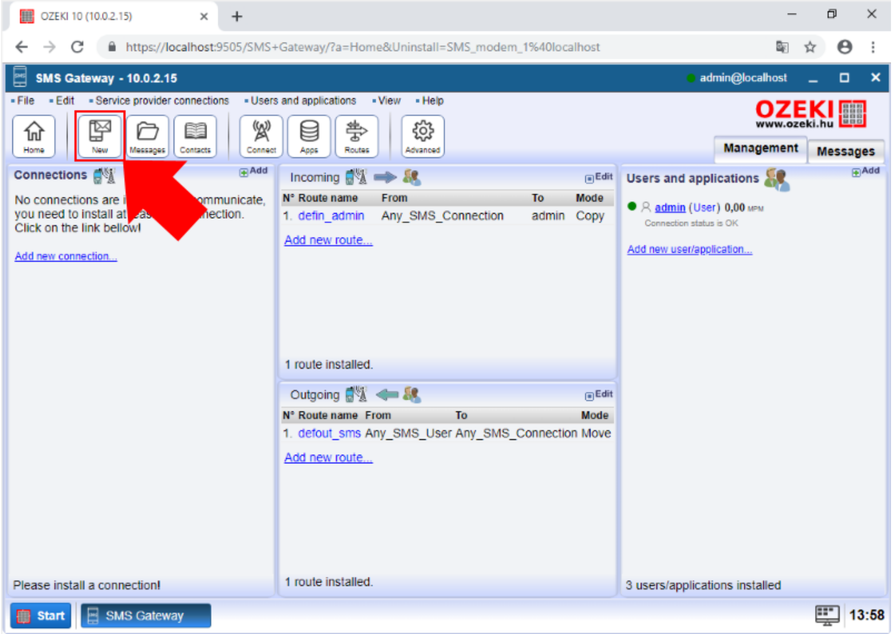
Make a **copy** of the message and send it to the destination, or **move** the message to the destination.

Mode: Move

Figure 14 - Creation of route for outgoing messages

# Step 5: Send your first SMS using the connection

Now with the previous steps you prepared to send and receive SMS messages with Ozeki 10 SMS Gateway and in that last step you can write and send your first SMS using the created connection. So, to enter the **Message composer** just click on the **New** icon as **Figure 15** shows, or select **File** and next click on **Compose.**



Connections

No connections are i... communicate, you need to install at leas... nection. Click on the link bellow!

Add new connection...

Please install a connection!

Incoming

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defin_admin | Any_SMS_Connection | admin | Copy |

Add new route...

1 route installed.

Outgoing

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defaut_sms | Any_SMS_User | Any_SMS_Connection | Move |

Add new route...

1 route installed.

Users and applications

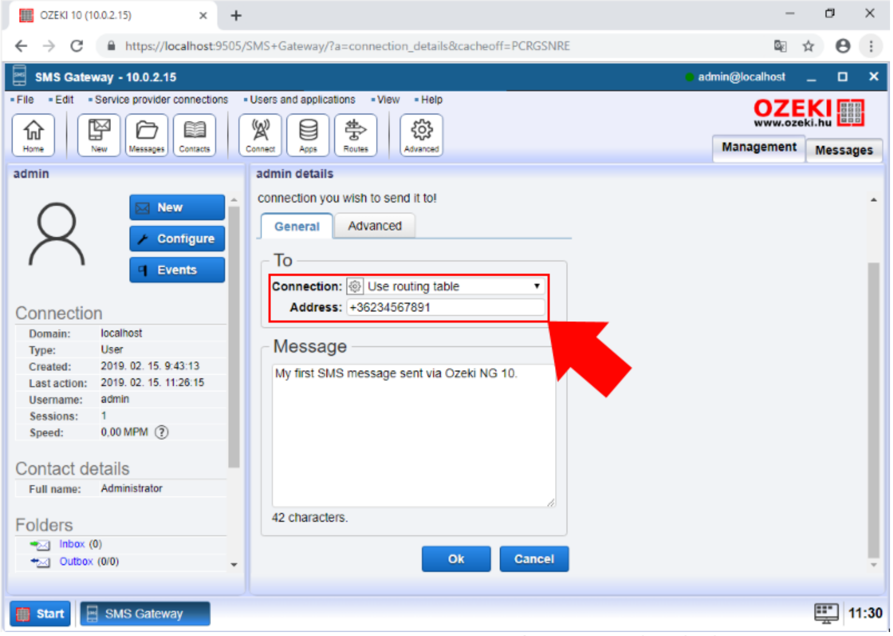
● admin (User) 0,00 MPM
Connection status is OK

Add new user/application...

3 users/applications installed

Figure 15 - Write a new message by Clicking on the **New** icon

Here in **Message composer**, you have to specify that who is going to be the recipient of your message (**Figure 16**). For that you can select the option **Use routing table** or select a connection that will send your message to the specified address that you have to enter in the **Address** field. In the **Message** field just type the message and if you click on **OK** you just sent your first SMS message using Ozeki 10 SMS Gateway.



admin

Connection

| | |
|---|---|
| Domain: | localhost |
| Type: | User |
| Created: | 2019. 02. 15. 9:43:13 |
| Last action: | 2019. 02. 15. 11:26:15 |
| Username: | admin |
| Sessions: | 1 |
| Speed: | 0,00 MPM |

Contact details

| | |
|---|---|
| Full name: | Administrator |

Folders

Inbox (0)
Outbox (0/0)

admin details

connection you wish to send it to!

To

Connection: Use routing table
Address: +36234567891

Message

My first SMS message sent via Ozeki NG 10.

42 characters.

Figure 16 - Message composer in Ozeki 10 SMS Gateway

# SMS Gateway Hardware Requirements

***This guide outlines the hardware requirements of the Ozeki software. You will first find the basic definitions and the recommended hardware configurations. You will find information about the supported operating systems, and the recommended hardware setup for you system.***

## Terms and definitions

**Connection:** A connection is typically defined as a link to the outside world. For example a link to a mobile network or a link to a database server would be a connection. Connections are able to receive and send messages. Hardware requirements depend on the number of connections you wish to setup.

**Message throughput:** The number of messages that travel through the system. It is measured in Messages per Second (MPS) or for lower capacity systems in terms of Messages per Minute (MPM). The message engine in Ozeki was designed to handle a maximum of 1000 messages per second on a single computer. Higher throughput can be achieved using multiple computers.

**CPU:** Bursts of high message throughput require higher single core performance. A high number of connections require higher number of thread processing capabilities. When choosing a CPU hierarchy for your system consider how you wish to use Ozeki. For best performance we recommend top of the line Multi core Intel CPU's. AMD CPUs are supported based on their equivalence to the suggested Intel CPUs.

**Memory:** For increased preformance, Ozeki caches message queues in memory. Make sure to put as much RAM into your system as possible, since this will greatly improve your system's performance.

**Network:** Ozeki can operate on lower network speeds, but a 1Gb LAN network connectivity is recommended. Depending on the number of simultaneous connections and the message throughput the network bandwidth can be a limiting factor.

## Supported operating systems

Ozeki 10 SMS Gateway supports the following operating systems. All operating systems must be installed in their x64 version. 32-bit versions are not compatible with Ozeki. Full Windows and Linux updates are also mandatory to be applied to ensure compatibility with Ozeki.

Ubuntu LTS 20.04
Raspbian Stretch
Windows 10 (Home, Pro, Enterprise)
Windows Server 2016 (Standard, Datacenter)
Windows Server 2019 (Standard, Datacenter)

## System sizes

**Small system:**
The number of connections is up to 10.
The message throughput is up to 1 MPS (message per second).

**Medium system:**
The number of connections is up to 20.
The message throughput is up to 25 MPS (message per second).

**Large system:**
The number of connections is up to 50.
The message throughput is up to 100 MPS (message per second).

**Enterprise system:**
The number of connections is up to 100.
The message throughput is up to 500 MPS (message per second).

**Enterprise+ system:**
The number of connections is up to 200.
The message throughput is up to 1000 MPS (message per second).

# Hardware variants

Ozeki supportes and was tested on both **bare metal** systems and **virtual machines**. All systems (hardware CPUs and virtual CPUs) must offer the 64bit instruction set.

**Supported virtual machine platforms:**
VMware vSphere Hypervisor (ESXi) 6.X and above
Microsoft Hyper-V Server 2012 and above
Hyper-V on Windows Server 2012 and above
Citrix XenServer 8.0 and above
KVM 2.6 and up.

# Ozeki system size

Small

| Ozeki 10 | Linux Debian-based | Windows-based |
|---|---|---|
| CPU Family | Intel i3 (Gen.8) or equivalent | |
| vCPUs | 1 | 2 |
| Memory | 1 | 2 |
| Storage | 30 GB SSD based storage | |

Medium

| Ozeki 10 | Linux Debian-based | Windows-based |
|---|---|---|
| CPU Family | Intel i5 (Gen.8) or equivalent | |
| vCPUs | 4 | 6 |
| Memory | 4 | 6 |
| Storage | 100 GB SSD based storage | |

Large

| Ozeki 10 | Linux Debian-based | Windows-based |
|---|---|---|
| CPU Family | Intel i7 (Gen.8) or equivalent | |
| vCPUs | 6 | 8 |
| Memory | 8 | 10 |
| Storage | 300 GB SSD based storage | |

Enterprise

| Ozeki 10 | Linux Debian-based | Windows-based |
|---|---|---|
| CPU Family | Intel Xeon E5 v4 or equivalent | |
| vCPUs | 8 | 10 |
| Memory | 16 | 18 |

| | | |
|---|---|---|
| Storage | 500 GB SSD based storage | |

Enterprise +

| Ozeki 10 | Linux Debian-based | Windows-based |
|---|---|---|
| CPU Family | Intel Xeon E7 v4 or equivalent | |
| vCPUs | 8+ | 10+ |
| Memory | 32+ | 34+ |
| Storage | 500+ GB SSD based storage | |

ARM-Based

| Raspberry Pi 3B+ (Debian9) | Ozeki 10 |
|---|---|
| Extensions | up to 16 |
| SIM Calls (SC) | up to 8 |
| SD Card | minimum 32GB Class 10 Micro SD |
| Required | Heat sink housing and Raspberry Pi-compatible 2.5 Amp Micro USB power supply. |

Cloud Provider

| Cloud Provider | Google (GCP) | Microsoft (Azure) | Amazon EC2 | Amazon Lightsail |
|---|---|---|---|---|
| Small (up to 10 ext) | G1 Small | B1ms | a1.medium | t2.small |
| Medium (up to 50 ext) | n1-standard-4 | D4 v3 | m5ad.xlarge | t2.xlarge |
| Large (up to 250 ext) | n1-highmem-4 | D12 v2 | r5ad.xlarge | t2.2xlarge |

# Ozeki SMS gateway features

## Mobile network connectivity features

☑ **SMS modem**: You can attach an SMS modem to your computer with a USB cable to send and receive wirelessly over the mobile network. In this case you don't need to sign up for an Internet SMS service. All you need to do is purchase a SIM card and put it into your SMS modem.

☑ **SMS modem pools**: If you wish to operate several modems to increase capacity, or you wish to have several phone numbers, you can connect an SMS modem pool to Ozeki SMS gateway. Ozeki will handle each modem in the pool as an independent SMS connection, and you can use Ozeki's advanced routing capabilities to decide which modem to use for sending. The SMS modem pool is usually connected to your computer with a USB cable. If you use an SMS modem pool, you will have to purchase a SIM card for every modem in the pool. For example if you use a modem pool with 8 modems, you need 8 SIM cards and you will have 8 phone numbers to use.

☑ **Android SMS connection**: You can connect to an Android mobile phone over Wifi and install the Ozeki Android SMPP gateway app to send and receive SMS messages from your computer.

☑ **SMPP SMS connection**: You can connect to an IP SMS service using the Short Message Peer to Peer (SMPP) protocol. Ozeki SMS gateway supports SMPP version 3.3, 3.4, 5.0. This protocol is used by Mobile network operators to exchange SMS messages. You can connect directly to an SMS Service Center (SMSC) using SMPP.

☑ **UCP SMS connection**: You can connect to an IP SMS service using the Universal Computer Protocol / External Machine Interface (UCP/EMI) protocol, version 3.5 or higher, to connect to an SMSC over the internet or other private TCP/IP network. Some mobile operators provide UCP connections for higher volumes of SMS traffic.

☑ **CIMD2 SMS connection**: You can connect to an IP SMS service using the Computer Interface to Machine Distribution (CIMD2) protocol, version 2 or higher to connect to an SMSC over the internet or other private TCP/IP network. This protocol is implemented by Nokia SMSCs.

☑ **HTTP client SMS connection**: You can connect to an IP SMS services services using the HyperText Transfer Protocol (HTTP). This is a customizable connection type, that allows you to use HTTP or HTTPS. For HTTP SMS you can set client certificate and validate server certificates. By creating an appropriate HTML template you can connect to virtually an HTTP SMS service provider.

## SMS routing features

☑ **Least cost SMS routing**: Rre-route SMS messages based on phone number prefix or route costs to ensure optimum delivery costs.

☑ **Backup SMS routing**: You can route a message to different connection in case the primary connection is not functioning. For example if the Internet fails, or your primary Internet based SMS service provider fails, you can automatically route the messages to a wireless connection (e.g. an SMS modem connected to your server with a data cable). Of course you can route the messages to other Internet based SMS service providers.

☑ **Copy messages**: You can copy each messge going through your system and forward it to a database for logging or to any SMS connection. You can make one or more copies and route them to your prefered destination, which can be a phone number, an email address a database or any application.

## Phone number manipulation features

☑ **Sender number modification**: You can change the sender ID as an SMS message goes through your system

☑ **Sender number pools**: You can pick a number randomly from a list and use it as the sender ID in your SMS message. This is great if you wish to operate a modem pool.

☑ **Receipient number modification**: You can change the recipient phone number. You can use this to reroute a message if somebody's phone number changes, or you can correct invalid recipient phone numbers by fixing the phone number prefix

☑ **Regular expression based phone number modification**: You can write a regular expression to detect and change phone numbers for both sender and recipient numbers.

## Message text manipulation features

☑ **Message text replacement**: You can change the SMS message text on the fly by simply rewriting it.

☑ **Append text to the message**: You can change the SMS message text by appendin an advertisment to the end of each SMS message, e.g. "Hello world - Sent by mycompan.

☑ **Word filtering**: You can filter out unwanted words and replace their content to stars (****) or you can drop messages containing such words

☑ **Message encryption and decryption**: You can encrypt SMS messages and decrypt SMS messages in the routing table.

☑ **Regular expression based modifications**: You can write a regular expression to modify the SMS message text.

☑ **C# algorithm based modifications**: You can write code in C# to change the SMS message text on the fly.

# Message encoding features

☑ **GSM 7 bit message encoding**: By default SMS mesages are encoded using the 7 bit SMS alphabet.

☑ **Multipart SMS encoding**: Messages longer than 160 characters (if the 7 bit SMS alphabet is used) or if they are longer than 70 characters (if the unicode character set is used) are split into multiple segments and are sent through the network according to the multipart SMS specification standards. The Ozeki SMS gateway does the multipart segmentation and reassembly.

☑ **Unicode character encoding**: You can send chinese, arabic, hebrew and any other character supported by the unicode alphabet. Ozeki implements the unicode character encoding.

☑ **Binary message encoding**: You can send 140 bytes of binary data in binary SMS messages. With Ozeki you can set all SMS attributes, so you can specify the binary message flag to set such data.

☑ **Application port numbers**: Application port numbers can be used if SMS messages are sent through the mobile network. Both sender and recipient port numbers are supported. Port numbers are sent in the User Data Header (UDH) according to the SMS standard. To send a messages to a port number specify the phone number in the following format: +36201234657:7777 where 7777 is the port number (you can change 7777 to any port number you like). This format applies to both the sender and recipient numbers. Port numbers are often used if mobile applications runing on Android or iOS mobile phones communicate with PC based services using Ozeki SMS Gatway.

☑ **Special message types**: Ozeki nativly supports WAP Push, SyncML, Voicemail notification, vCard, vCalendar and other native SMS message formats used in mobil networks.

☑ **Message encoding policies**: You can set "don't transform", "best match" and "enforce charset" replacement policies, to do character set transformation. These are very useful because it gives you total control over which character sets are used when your SMS messages are encoded.

# Application connectivity features

☑ **Excel SMS**: You can use Microsoft Excel or a compatible word processor to send messages to a list of phone numbers

☑ **SQL messaging**: Send and receive SMS mesages from a database.

☑ **Text file messaging**: Send and receive SMS messages by creating text files in a directory

☑ **Http SMS messaging**: You can use the HTTP API of Ozeki SMS gateway to send and receive SMS messagse

# High performance features

☑ **Load balancing**: You can setup SMS connection groups for load balancing. This means you can create multiple SMS network connections to IP SMS services or through wireless connections. If you include these connections in a group, and you route messages to this group, sending will be done using load balancing. In load balancing configuration messages are distributed among sending links according to the link's capacity.

☑ **Fail safe messaging**: By creating SMS connection groups your messages will be sent even if a one or more members in the connection group fails. For example if you have 4 mobile network connections to 4 different SMS service centers (SMSCs), and any 3 of them fails, all your messages will still be delivered through the remaining last one.

☑ **Memory queues**: The system stores message queues in memory for performance and keeps these queues in sync with the hard drive. Batches of 1000 messages are loaded into memory for sending cycles. This makes high performance messaging possible.

# Reporting features

☑ **Reporting database**: You can save all SMS messaging going through your system into a reporting database. In your database a database table will be created and updated. You have will have detailed information about all event related to each SMS that goes through the system. You can use Microsoft SQL Server, Oracle, MySQL, Postgres and many other database server systems.

☑ **No loss reporting**: This feature makes it possible to not loose a single record. If your reporting database goes offline for whatever reason, the Ozeki SMS software will store messages in a text file on the local hard drive until the reporting database comes back online. When the reporting database becomes available again it will push these messages to the database.

☑ **Low level logging**: All connections provide detailed logging down to the protocol layer. You can check the exact protocol data units (PDUs) sent through the connections. This is very helpful in finding problems or figuring out why a certain message was sent in a certain way. It is also great in disupte resolution with customers and service providers.

# Addressbook features

☑ **Addressbook builder**: You can setup routing rules to automatically register sender and/or recipient phone numbers into an addressbook automatically. You can also define conditions, such as keyword filter to decide which phone number to record. This addressbook can later be used to send group messages

☑ **Addressbook export**: You can export addressbook in various formats, such as Microsoft Excel xlsx, csv, text files, json files, etc.

☑ **Addressbook import**: You can import your exported address books

☑ **Database addressbook**: You can setup database tables to use as addressbooks. Ozeki can SELECT and INSERT addressbook records into these tables, and you can use them as you would use any other othressbook on the user interface. Microsoft SQL Server, Oracle, MySQL and serveral other database servers are supported.

☑ **LDAP addressbook**: You can use LDAP directories, such as Microsoft Active Directory to lookup contacts

# Authentication features

☑ **Standard system users**: You can create user accounts on the Graphical user interface (GUI) of Ozeki

☑ **LDAP authentication**: You can authenticate users using an LDAP server, such as Microsoft Active Directory.

☑ **RADIUS authentication**: You can authenticate users using RADIUS

☑ **HTTP/HTTPS authentication**: You can authenticate users using HTTP or HTTPS request. This allows you to authenticate your users using web services.

☑ **SQL authentication**: You can connect Ozeki to a user database, and ozeki can lookup user accounts in the database using customizable SQL queries. Microsoft SQL Server, Oracle, MySQL and serveral other database servers are supported.

# Security features

☑ **HTTPS**: Https is supported for the Graphical User Interface (GUI), for the HTTP API, and for sending SMS messages to HTTP sms service providers

☑ **SMPP client over TLS/SSL**: When you connect to SMPP SMS service providers, you can use SSL.

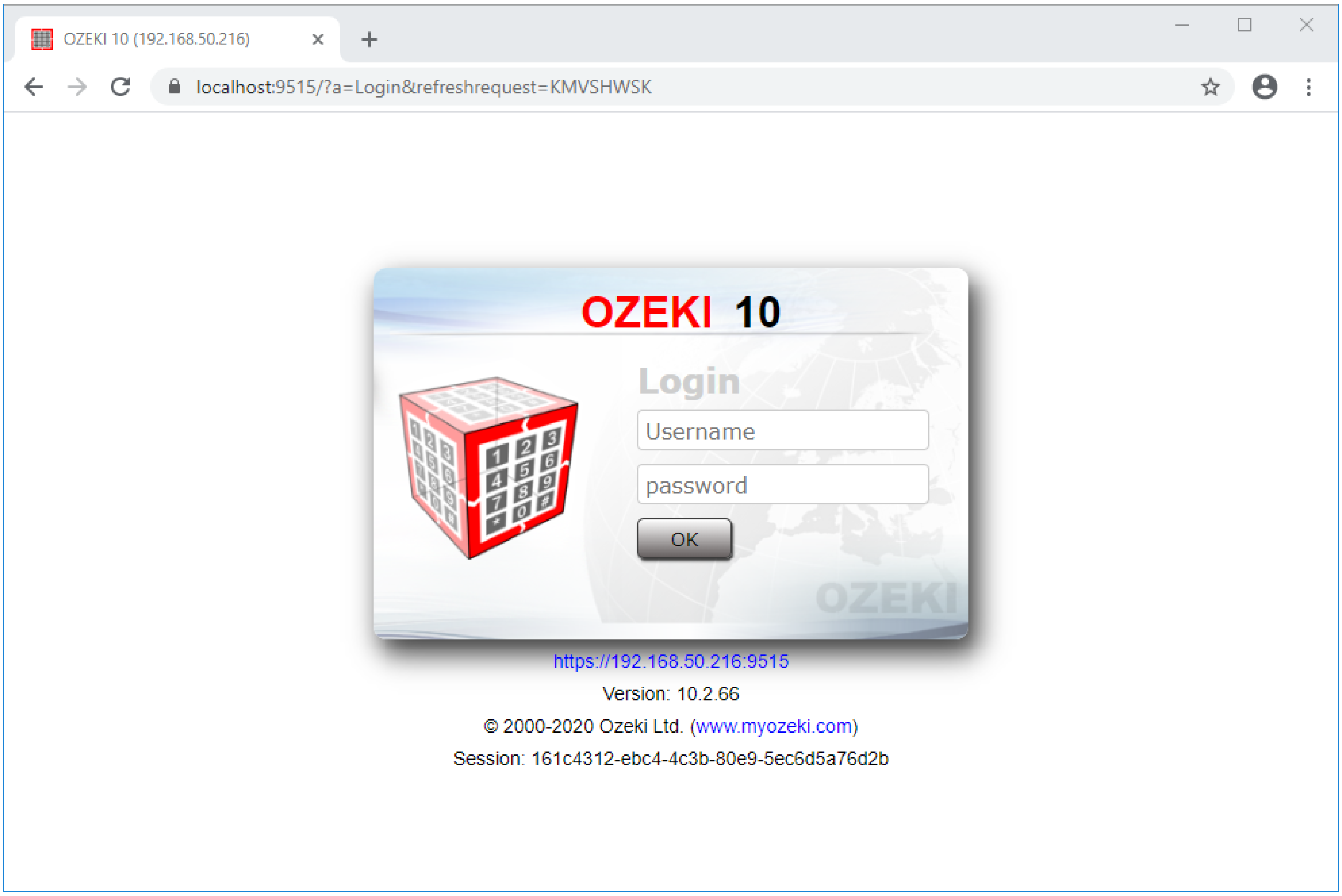
☑ **SMPP server using SSL**: You can offer SMPP SMS service to your customers using SSL links

☑ **UCP client over TLS/SSL**: You can connect to UCP SMS service providers over an SSL channel

☑ **CIMD2 client over TLS/SSL**: You can connect to CIMD2 SMS service providers over an SSL channel

☑ **Secure password storage**: Passwords are store in an encrypted way in the system

# Ozeki SMS gateway screenshots



OZEKI 10 (192.168.50.216)
localhost:9515/?a=Login&refreshrequest=KMVSHWSK
OZEKI 10
Login
Username
password
OK
OZEKI

https://192.168.50.216:9515

Version: 10.2.66

© 2000-2020 Ozeki Ltd. (www.myozeki.com)

Session: 161c4312-ebc4-4c3b-80e9-5ec6d5a76d2b

Figure 1 - Login form of Ozeki SMS Gateway

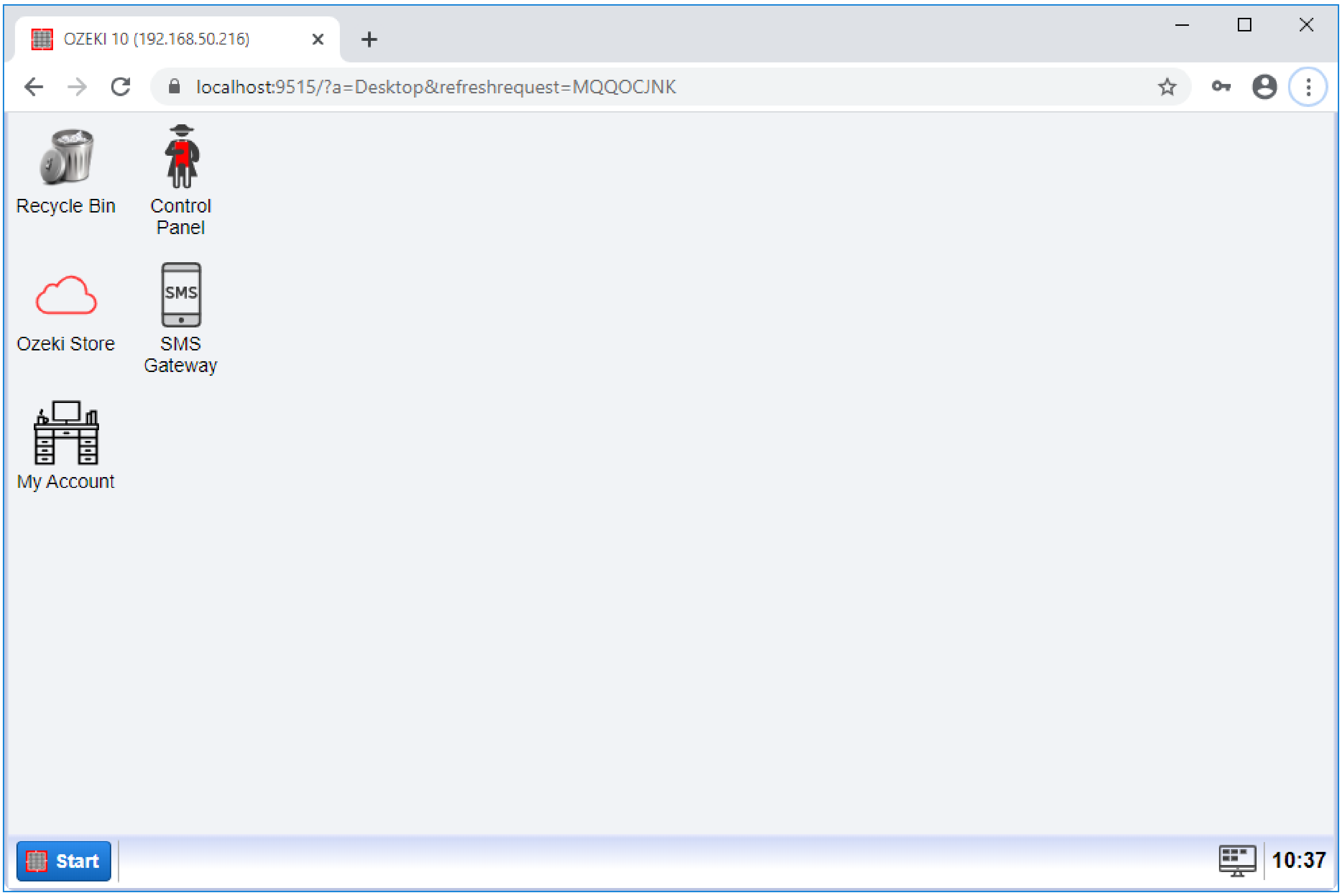
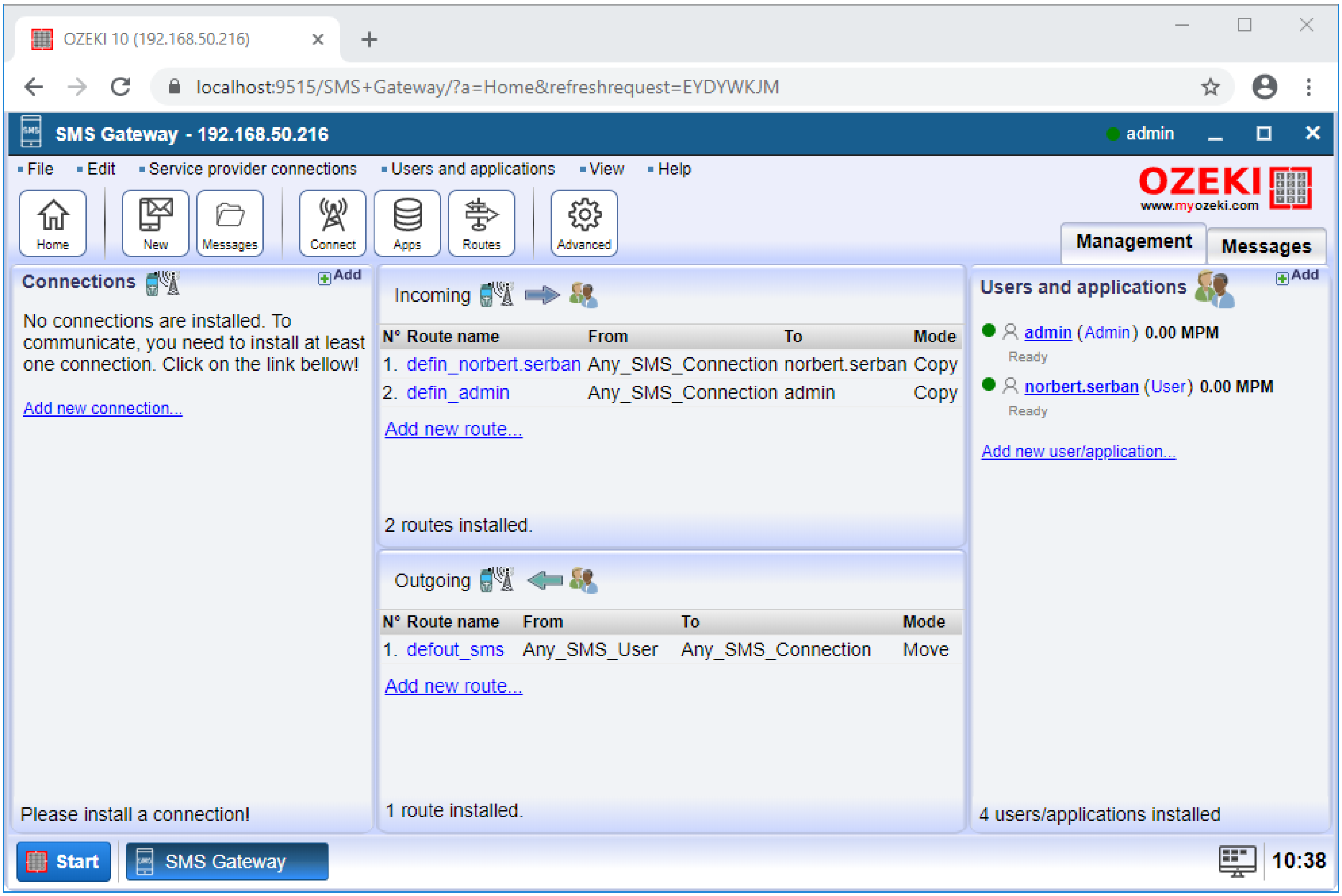

OZEKI 10 (192.168.50.216)
localhost:9515/?a=Desktop&refreshrequest=MQQOCJNK
Recycle Bin
Control Panel
Ozeki Store
SMS Gateway
My Account
Start
10:37

Figure 2 - Desktop of Ozeki SMS Gateway



Connections

No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

Please install a connection!

Incoming

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defin_norbert.serban | Any_SMS_Connection | norbert.serban | Copy |
| 2. | defin_admin | Any_SMS_Connection | admin | Copy |

Add new route...

2 routes installed.

Outgoing

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move |

Add new route...

1 route installed.

Users and applications

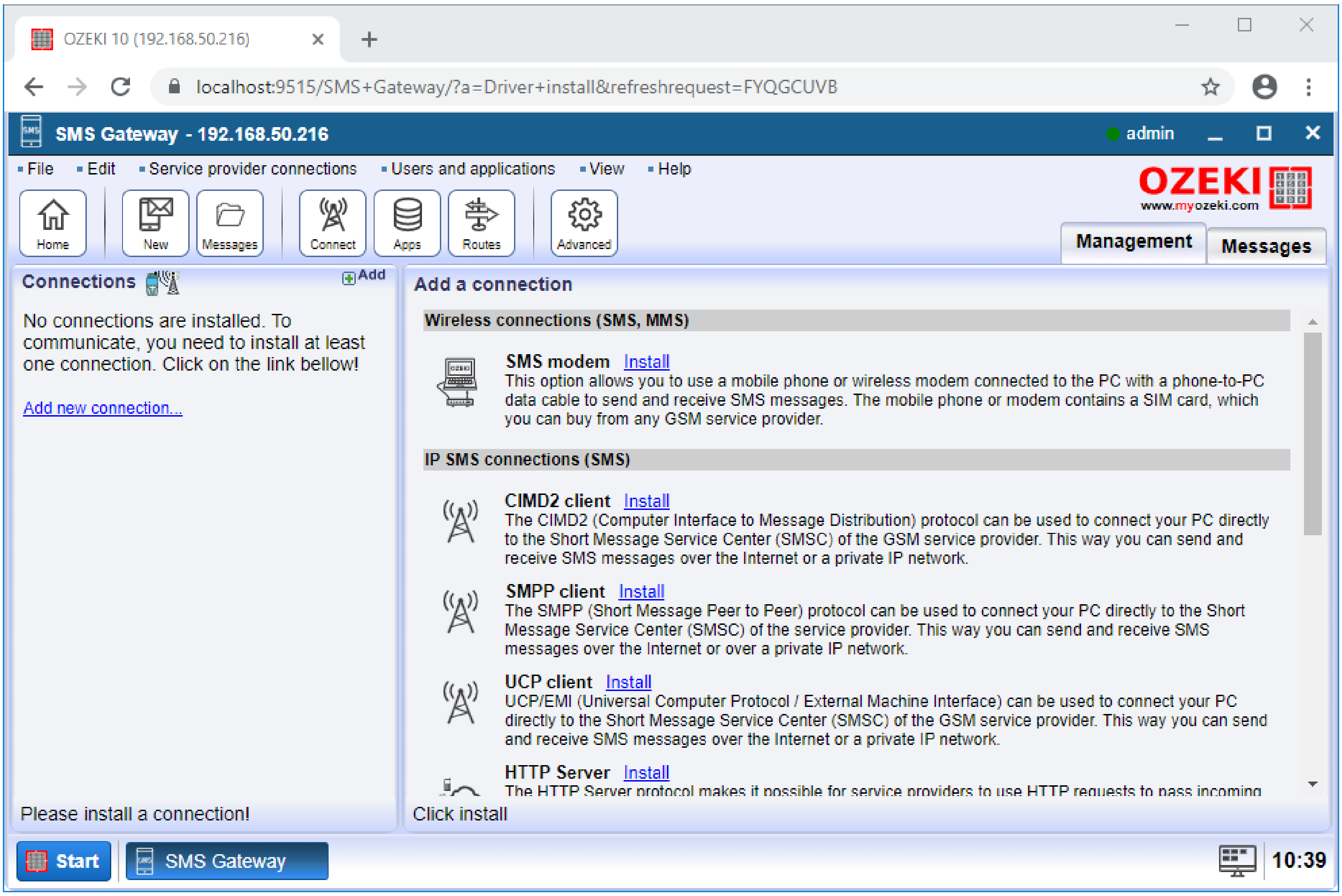
● admin (Admin) 0.00 MPM
Ready

● norbert.serban (User) 0.00 MPM
Ready

Add new user/application...

4 users/applications installed

Figure 3 - The main menu of SMS Gateway application


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=Driver+install&refreshrequest=FYQGCUVB
SMS Gateway - 192.168.50.216
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
OZEKI
www.myozeki.com
Management
Messages
Start
SMS Gateway
10:39

Connections

Add

No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

Please install a connection!

Add a connection

Wireless connections (SMS, MMS)

SMS modem Install
This option allows you to use a mobile phone or wireless modem connected to the PC with a phone-to-PC data cable to send and receive SMS messages. The mobile phone or modem contains a SIM card, which you can buy from any GSM service provider.

IP SMS connections (SMS)

CIMD2 client Install
The CIMD2 (Computer Interface to Message Distribution) protocol can be used to connect your PC directly to the Short Message Service Center (SMSC) of the GSM service provider. This way you can send and receive SMS messages over the Internet or a private IP network.

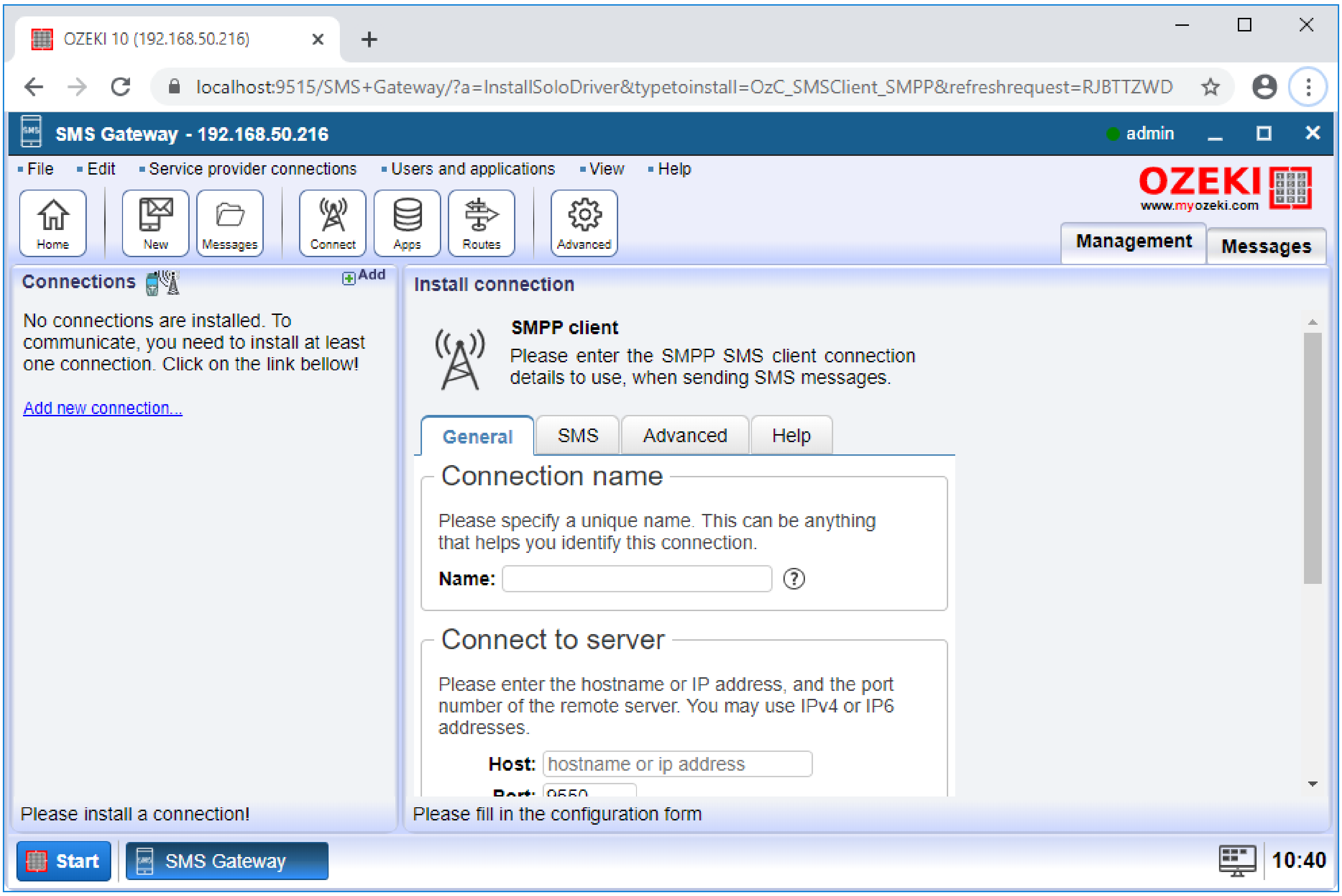
SMPP client Install
The SMPP (Short Message Peer to Peer) protocol can be used to connect your PC directly to the Short Message Service Center (SMSC) of the service provider. This way you can send and receive SMS messages over the Internet or over a private IP network.

UCP client Install
UCP/EMI (Universal Computer Protocol / External Machine Interface) can be used to connect your PC directly to the Short Message Service Center (SMSC) of the GSM service provider. This way you can send and receive SMS messages over the Internet or a private IP network.

HTTP Server Install
The HTTP Server protocol makes it possible for service providers to use HTTP requests to pass incoming

Click install

Figure 4 - Add a new connection


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=InstallSoloDriver&typetoinstall=OzC_SMSClient_SMPP&refreshrequest=RJBTTZWD
SMS Gateway - 192.168.50.216
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
OZEKI
www.myozeki.com
Management
Messages
General
SMS
Advanced
Help
Start
SMS Gateway
10:40

Connections

Add

No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

Please install a connection!

Install connection

SMPP client
Please enter the SMPP SMS client connection details to use, when sending SMS messages.

Connection name

Please specify a unique name. This can be anything that helps you identify this connection.

Name:

Connect to server

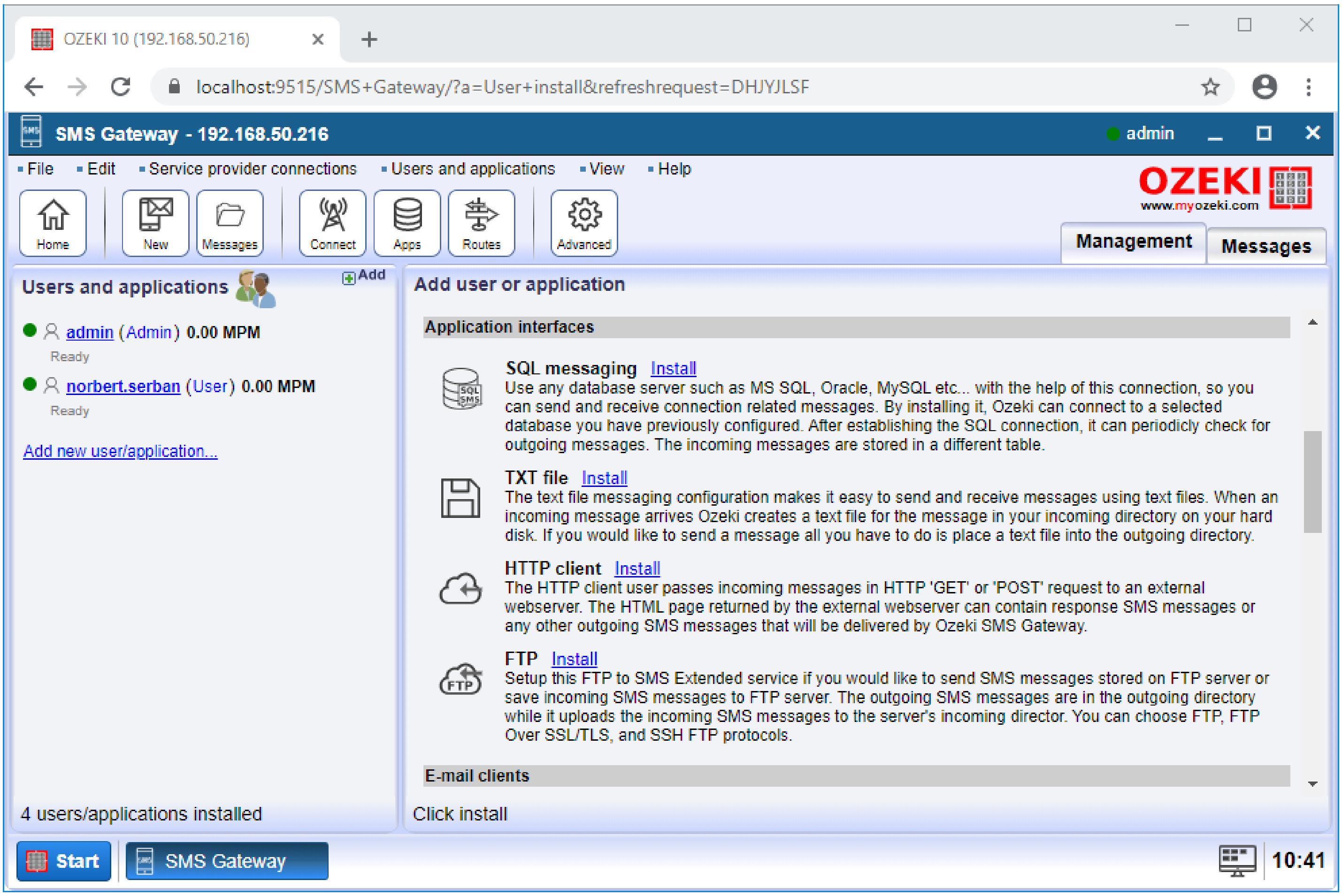
Please enter the hostname or IP address, and the port number of the remote server. You may use IPv4 or IP6 addresses.

Host: hostname or ip address

Port: 9550

Please fill in the configuration form

Figure 5 - Create a new SMPP connection



OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=User+install&refreshrequest=DHJYJLSF
SMS Gateway - 192.168.50.216
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Users and applications
Add
admin (Admin) 0.00 MPM
Ready
norbert.serban (User) 0.00 MPM
Ready
Add new user/application...
4 users/applications installed
Start
SMS Gateway
10:41

Add user or application

Application interfaces

**SQL messaging** Install
Use any database server such as MS SQL, Oracle, MySQL etc... with the help of this connection, so you can send and receive connection related messages. By installing it, Ozeki can connect to a selected database you have previously configured. After establishing the SQL connection, it can periodicly check for outgoing messages. The incoming messages are stored in a different table.

**TXT file** Install
The text file messaging configuration makes it easy to send and receive messages using text files. When an incoming message arrives Ozeki creates a text file for the message in your incoming directory on your hard disk. If you would like to send a message all you have to do is place a text file into the outgoing directory.

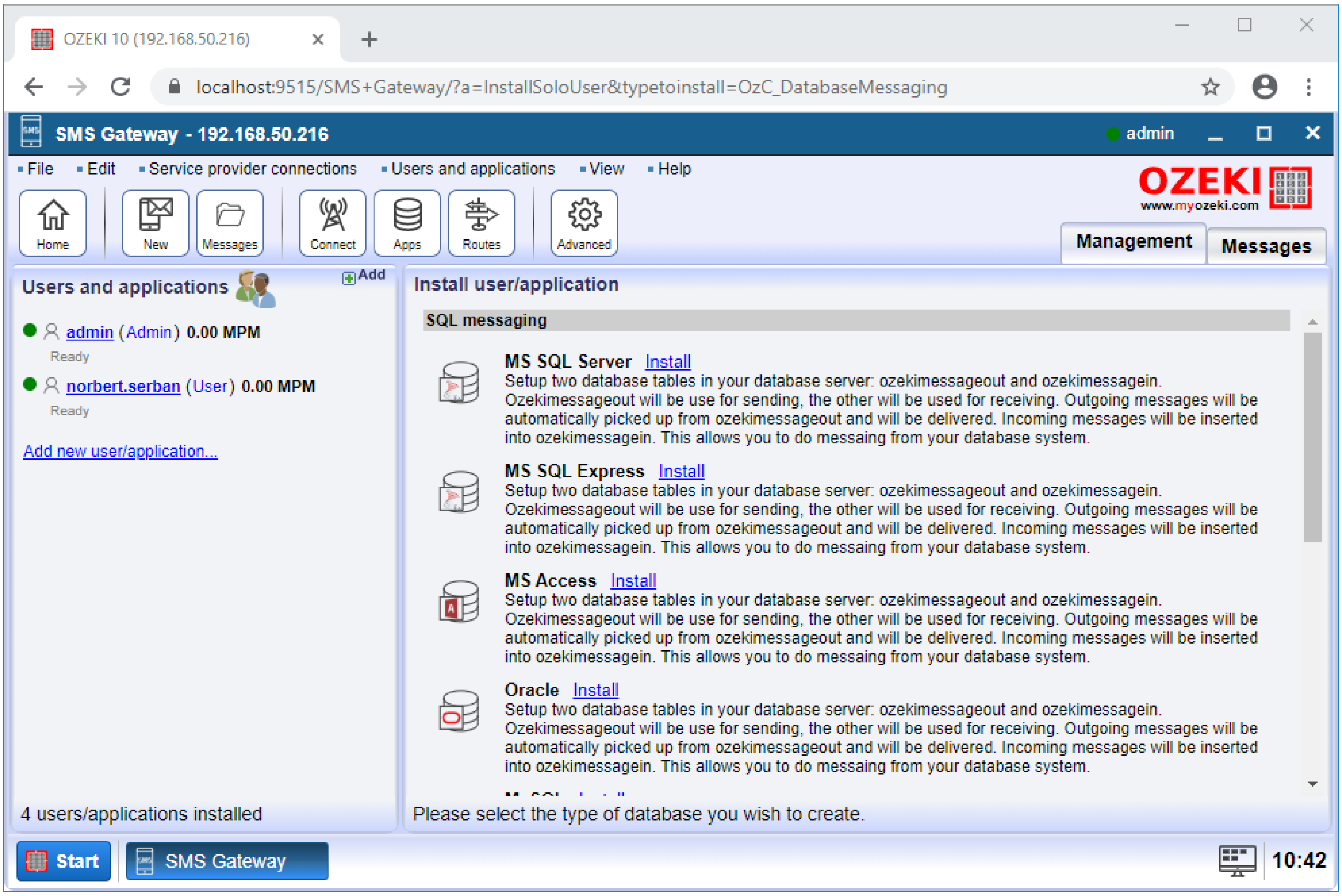
**HTTP client** Install
The HTTP client user passes incoming messages in HTTP 'GET' or 'POST' request to an external webserver. The HTML page returned by the external webserver can contain response SMS messages or any other outgoing SMS messages that will be delivered by Ozeki SMS Gateway.

**FTP** Install
Setup this FTP to SMS Extended service if you would like to send SMS messages stored on FTP server or save incoming SMS messages to FTP server. The outgoing SMS messages are in the outgoing directory while it uploads the incoming SMS messages to the server's incoming director. You can choose FTP, FTP Over SSL/TLS, and SSH FTP protocols.

E-mail clients

Click install

Figure 6 - Add a new user or application



OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=InstallSoloUser&typetoinstall=OzC_DatabaseMessaging
SMS Gateway - 192.168.50.216
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Users and applications
Add
admin (Admin) 0.00 MPM
Ready
norbert.serban (User) 0.00 MPM
Ready
Add new user/application...
4 users/applications installed
Start
SMS Gateway
10:42

Install user/application

SQL messaging

**MS SQL Server** Install
Setup two database tables in your database server: ozekimessageout and ozekimessagein. Ozekimessageout will be use for sending, the other will be used for receiving. Outgoing messages will be automatically picked up from ozekimessageout and will be delivered. Incoming messages will be inserted into ozekimessagein. This allows you to do messaging from your database system.

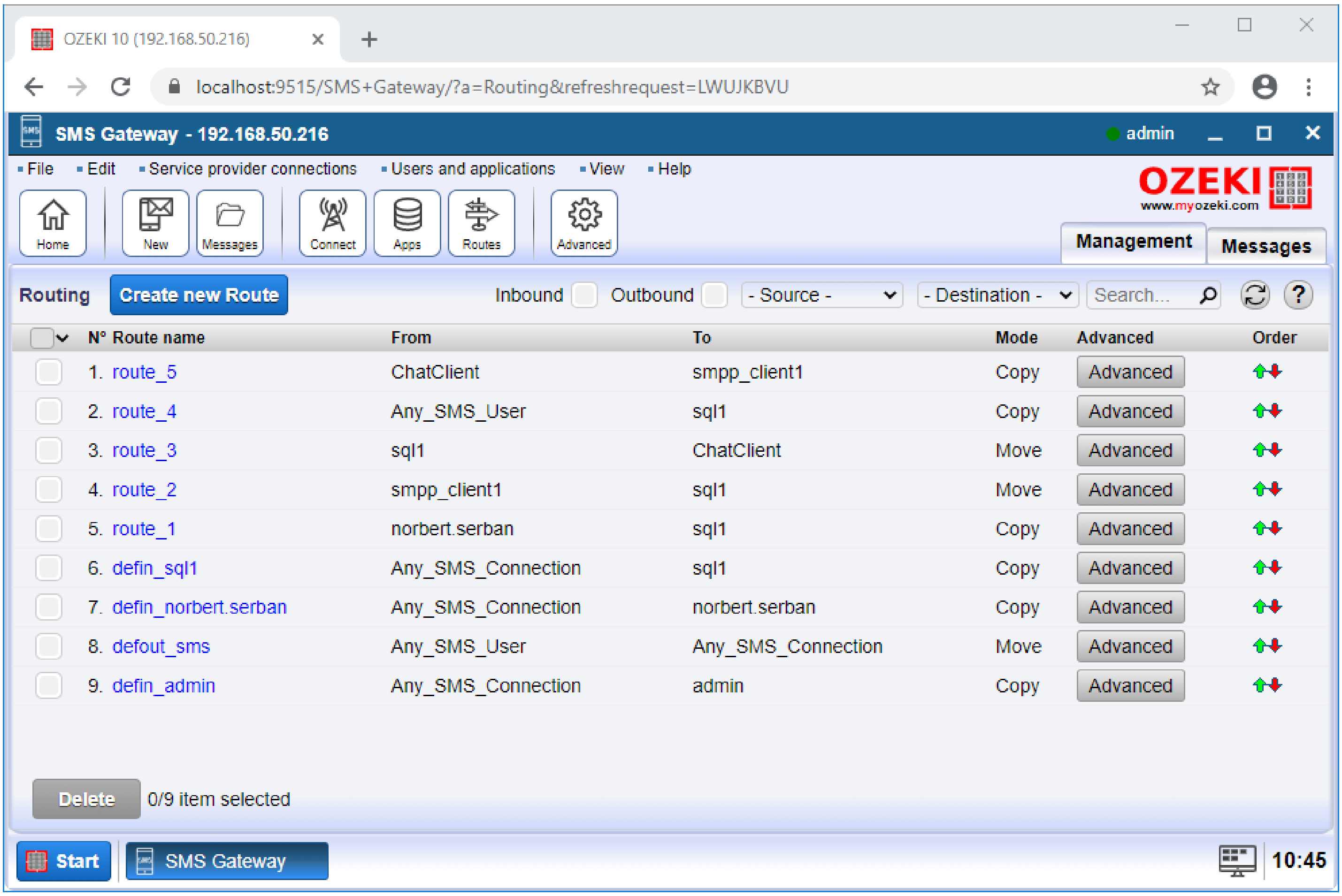
**MS SQL Express** Install
Setup two database tables in your database server: ozekimessageout and ozekimessagein. Ozekimessageout will be use for sending, the other will be used for receiving. Outgoing messages will be automatically picked up from ozekimessageout and will be delivered. Incoming messages will be inserted into ozekimessagein. This allows you to do messaging from your database system.

**MS Access** Install
Setup two database tables in your database server: ozekimessageout and ozekimessagein. Ozekimessageout will be use for sending, the other will be used for receiving. Outgoing messages will be automatically picked up from ozekimessageout and will be delivered. Incoming messages will be inserted into ozekimessagein. This allows you to do messaging from your database system.

**Oracle** Install
Setup two database tables in your database server: ozekimessageout and ozekimessagein. Ozekimessageout will be use for sending, the other will be used for receiving. Outgoing messages will be automatically picked up from ozekimessageout and will be delivered. Incoming messages will be inserted into ozekimessagein. This allows you to do messaging from your database system.

Please select the type of database you wish to create.

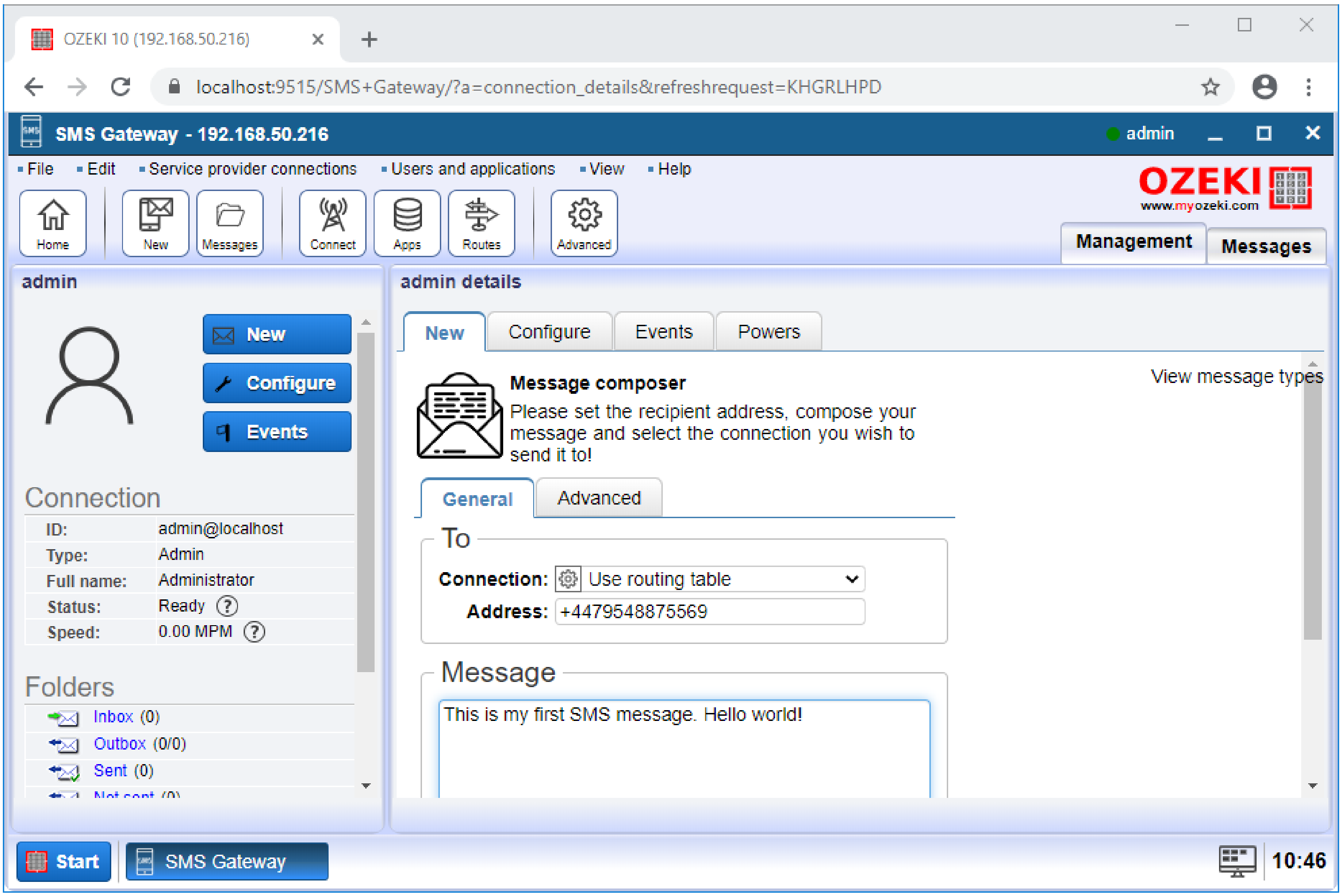
Figure 7 - List of the available database applications



| N° | Route name | From | To | Mode | Advanced | Order |
|---|---|---|---|---|---|---|
| 1. | route_5 | ChatClient | smpp_client1 | Copy | Advanced | ↑↓ |
| 2. | route_4 | Any_SMS_User | sql1 | Copy | Advanced | ↑↓ |
| 3. | route_3 | sql1 | ChatClient | Move | Advanced | ↑↓ |
| 4. | route_2 | smpp_client1 | sql1 | Move | Advanced | ↑↓ |
| 5. | route_1 | norbert.serban | sql1 | Copy | Advanced | ↑↓ |
| 6. | defin_sql1 | Any_SMS_Connection | sql1 | Copy | Advanced | ↑↓ |
| 7. | defin_norbert.serban | Any_SMS_Connection | norbert.serban | Copy | Advanced | ↑↓ |
| 8. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move | Advanced | ↑↓ |
| 9. | defin_admin | Any_SMS_Connection | admin | Copy | Advanced | ↑↓ |

0/9 item selected

Figure 8 - Routing table for the messages



Connection

| | |
|---|---|
| ID: | admin@localhost |
| Type: | Admin |
| Full name: | Administrator |
| Status: | Ready |
| Speed: | 0.00 MPM |

Folders

Inbox (0)
Outbox (0/0)
Sent (0)
Not sent (0)

Message composer

Please set the recipient address, compose your message and select the connection you wish to send it to!

To

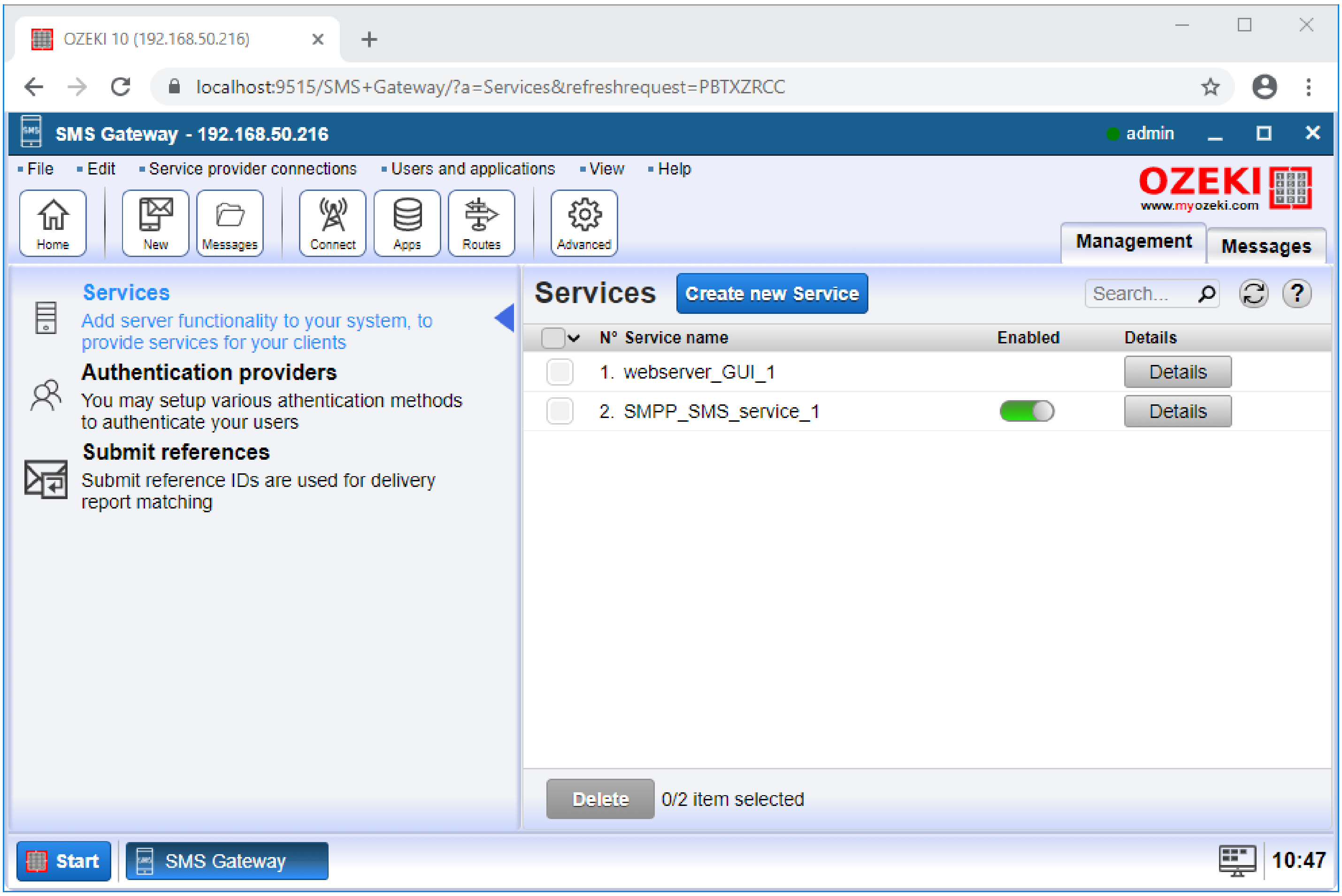
Connection: Use routing table

Address: +4479548875569

Message

This is my first SMS message. Hello world!

Figure 9 - Compose a new message


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=Services&refreshrequest=PBTXZRCC
SMS Gateway - 192.168.50.216
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Start
SMS Gateway
10:47

**Services**
Add server functionality to your system, to provide services for your clients

**Authentication providers**
You may setup various athentication methods to authenticate your users

**Submit references**
Submit reference IDs are used for delivery report matching

**Services** Create new Service

| N° | Service name | Enabled | Details |
|---|---|---|---|
| 1. | webserver_GUI_1 | | Details |
| 2. | SMPP_SMS_service_1 | ● | Details |

Delete 0/2 item selected

Figure 10 - The Advanced menu of SMS Gateway

# Application connectivity features

☑ **Standard User**: With Ozeki SMS Gateway Standard user you can log in to the grafical user interface and you are able to send SMS. And you can investigate the details of the delivery of the message.

☑ **Autoreply SMS**: Ozeki SMS Gateway has 3 type of Autoreply Users which are used for automatically replying SMS messages or forwarding them to any recipient. These users can work in 3 simple ways. You will see how to install and configure these users on Ozeki SMS Gateway to serve your needs.

☑ **E-mail to SMS**: Ozeki SMS Gateway offers various methods to setup E-mail to SMS functionality. You can use IMAP, POP3 or SMTP to send and receive E-mails and convert them to and from SMS messages. You can put the phone number(s) into the subject line of the e-mails or you may send an e-mail to an address containing the phone nubmer, such as +441234657@smsgw.yourcompany.com

☑ **SMS from/to Email through your Email Account**: Ozeki SMS Gateway's E-mail User can be used for sending or downloading emails from a mailbox. For sending emails it should connect to your email account's SMTP server. For downloading emails from a mailbox it should connect to the POP3 server through your email account.

☑ **SMS from/to File**: Ozeki SMS Gateway's File User is capable to send and receive SMS messages in different file formats. Simple, Text, Verbose, List, CSV, XML, SAP, ATF and KAL file formats are supported by the Ozeki SMS Gateway's File User. Your application can place these files in proper directories to send SMS messages.

☑ **SMS from/to FTP**: Ozeki SMS Gateway's FTP to SMS Extended User can synchronize directories with SMS Gateway through standard FTP, FTPS or SFTP protocols. Incoming SMS messages will be uploaded and outgoing SMS messages will be downloaded from the FTP server. You can also look at the accepted file formats.

☑ **Start your Applications with SMS**: Ozeki SMS Gateway's Application Starter User can run any process or application in case an SMS message arrives. You simply need to provide the file path. You can also fetch parameters from received SMS messages to use them as process parameters or command line arguments.

# Installation Guide

***The following list shows the operating systems on which Ozeki 10 can be installed.***

## Supported operating systems:

### How to install Ozeki 10 on Windows

Ozeki 10 is compatible with Windows 10, Windows 8, Windows 7 and Windows Server 20xx. Our step-by-step guide will show you exactly how to install it on any of them. It does not require any specific knowledge. It will take you about 5-10 minutes to complete.

Download and install Ozeki 10 sms gateway for Windows

### How to install Ozeki 10 on Linux

If you use a Ubuntu based Linux distribution then Ozeki 10 can be installed on your computer as well. You don't need specific knowledge of ubuntu software to install our product, because it is presented by a step-by-step guide. You will need 10-20 minutes to install the software. You need to install the Mono Framework in order to use our product.

Download and install Ozeki 10 sms gateway for Linux

### How to install Ozeki 10 on Raspberry PI

If you have an ARM based PC, such as Raspberry PI, the Ozeki 10 can also be installed on it. You don't need any specific knowledge of ARM based PCs to install our product, because it is done in a graphical user interface and it is presented by a step-by-step guide. You will need the Mono Framework in order to use our product.

Download and install Ozeki 10 sms gateway for Rasberry Pi

# How to install Ozeki on Windows 10

This guide provides clear instructions about how you can perform the set up of Ozeki Installer. This software is an app store that allows you to get any Ozeki product that you want. With this product, you will be able to install ozeki apps with one click. The whole procedure will take just about 5 minutes. The document is organized into 7 simple steps. The installation process does not need any specific knowledge since it is just a straightforward 'next-next' installation. Setting up of Ozeki Installer software on your Windows computer can be done in a pretty easy way. So, let's get started!

## Check how easy it is to set up Ozeki Installer!

## Prerequisites

In order to use Ozeki Installer to setup Ozeki apps, you need a suitable computer, with a network connection (Internet or LAN). On this computer you should have Windows 10 (or Windows Server 2016 or newer) installed. For more information, please check out the detailed list of system requirements in our System requirements page.

**Ozeki Prerequisites:**

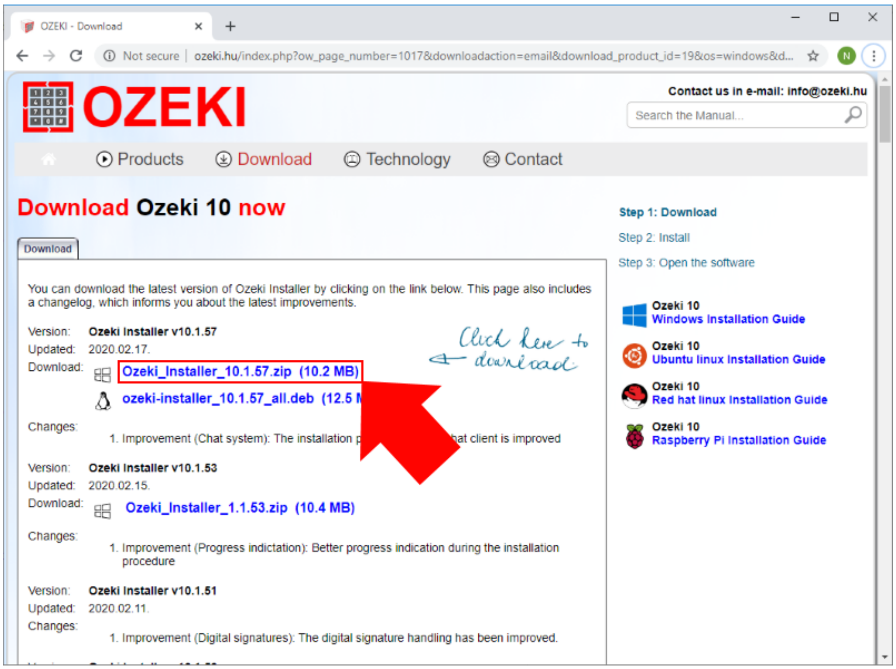
☑ Personal computer (PC)
☑ 8 GB RAM
☑ Quad Core CPU
☑ SSD drive (for better performance) with 100 MB of free disk space
☑ Windows 10 operating system
☑ Internet connection
☑ Web browser: Google Chrome or Microsoft Edge (Chrome based)
☑ Ozeki Installer

## Step 1 - Download Ozeki Installer

The latest release of Ozeki Installer can be downloaded from the following Downloads page: http://www.ozeki.hu/index.php?owpn=1017&dpid=19 To download the Installer in compressed format, you need to click on the first Ozeki_Installer.x.xx.zip link as **Figure 1** shows below. We recommend you to always download the latest version. This version always contains all the currently available features and improvements.

You may download Ozeki Installer from the link below.
**http://www.ozeki.hu/index.php?owpn=1017&dpid=19**
**Download Ozeki Installer**

Click here to
← download.



OZEKI - Download
Not secure | ozeki.hu/index.php?ow_page_number=1017&downloadaction=email&download_product_id=19&os=windows&d...
OZEKI
Contact us in e-mail: info@ozeki.hu
Search the Manual...
Products
Download
Technology
Contact
Download

Download Ozeki 10 now

You can download the latest version of Ozeki Installer by clicking on the link below. This page also includes a changelog, which informs you about the latest improvements.

Version: Ozeki Installer v10.1.57
Updated: 2020.02.17.
Download: Ozeki_Installer_10.1.57.zip (10.2 MB)
ozeki-installer_10.1.57_all.deb (12.5 M

Changes:
1. Improvement (Chat system): The installation p ... at client is improved

Version: Ozeki Installer v10.1.53
Updated: 2020.02.15.
Download: Ozeki_Installer_1.1.53.zip (10.4 MB)

Changes:
1. Improvement (Progress indictation): Better progress indication during the installation procedure

Version: Ozeki Installer v10.1.51
Updated: 2020.02.11.
Changes:
1. Improvement (Digital signatures): The digital signature handling has been improved.

Click here to download

Step 1: Download
Step 2: Install
Step 3: Open the software

Ozeki 10
Windows Installation Guide

Ozeki 10
Ubuntu linux Installation Guide

Ozeki 10
Red hat linux Installation Guide

Ozeki 10
Raspberry Pi Installation Guide

Figure 1 - Downloading Ozeki Installer

## Step 2 - Extract the downloaded file

When the download finished, you have to extract the compressed file to be able to run the installer of the software. For that, please open the Downloads folder on your computer. Then, as you can see it on **Figure 2**, right click on the file and select **Extract All...**. By doing this, you will get the executable installer of Ozeki Installer.

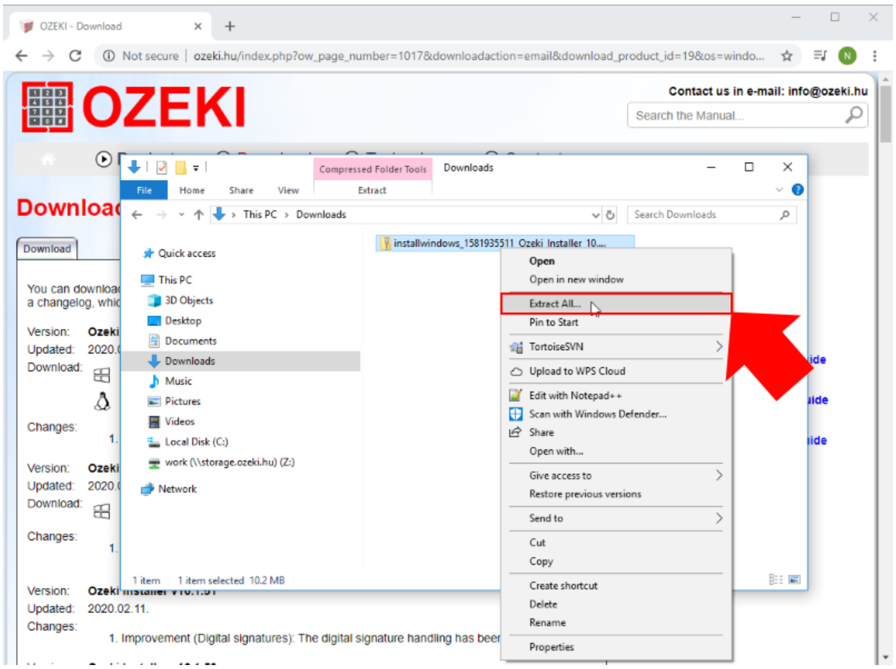
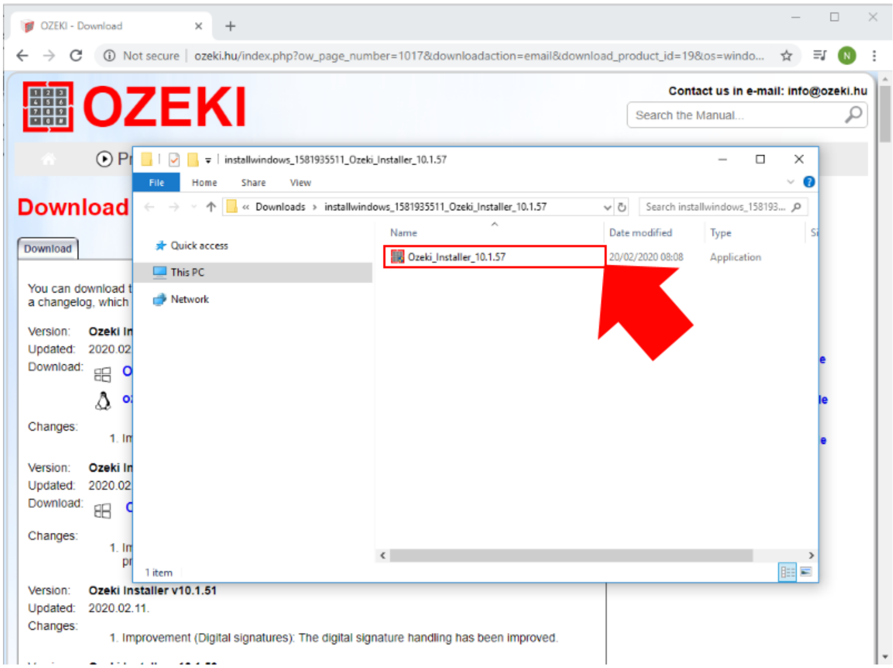


Figure 2 - Extract installer

## Step 3 - Run the installer

After you extracted the .zip file, please open the newly created folder, and just like in **Figure 3**, just click on the executable file to start the installation process.



OZEKI - Download
Not secure | ozeki.hu/index.php?ow_page_number=1017&downloadaction=email&download_product_id=19&os=windo...
OZEKI
Contact us in e-mail: info@ozeki.hu
Search the Manual...
Download
installwindows_1581935511_Ozeki_Installer_10.1.57
File
Home
Share
View
Downloads › installwindows_1581935511_Ozeki_Installer_10.1.57
Search installwindows_158193...
Quick access
This PC
Network
1 item

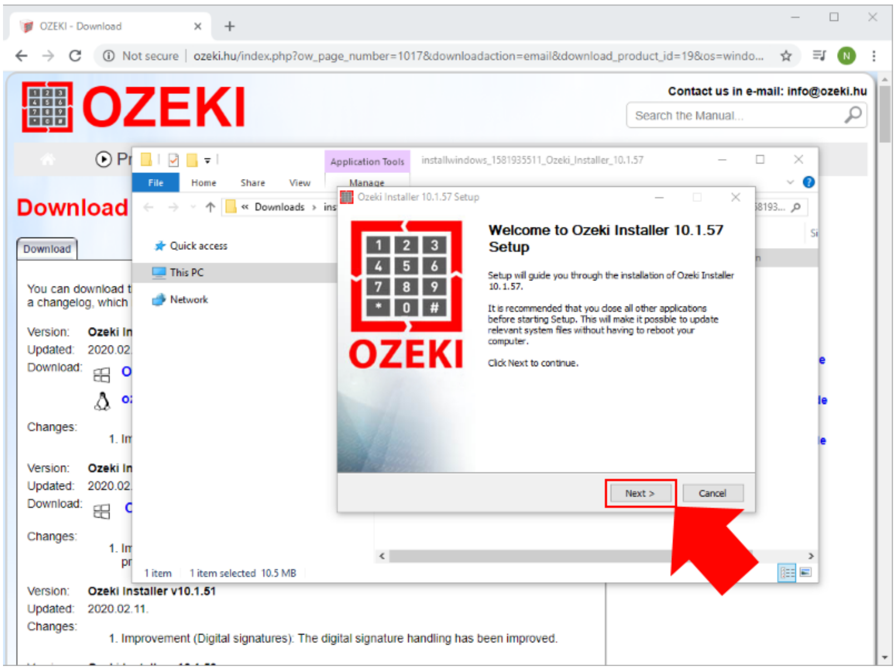
| Name | Date modified | Type |
|---|---|---|
| Ozeki_Installer_10.1.57 | 20/02/2020 08:08 | Application |

Version: Ozeki Installer v10.1.51
Updated: 2020.02.11.
Changes:
1. Improvement (Digital signatures): The digital signature handling has been improved.

Figure 3 - Start installation

# Step 4 - Install Ozeki Installer

As soon as you clicked on the .exe file, the installer starts with a welcome screen (**Figure 4**). All you have to do here is to click **Next** to continue the installation.



OZEKI - Download
Not secure | ozeki.hu/index.php?ow_page_number=1017&downloadaction=email&download_product_id=19&os=windo...
OZEKI
Contact us in e-mail: info@ozeki.hu
Search the Manual...
Download
Application Tools
installwindows_1581935511_Ozeki_Installer_10.1.57
Manage
Quick access
This PC
Network
1 item    1 item selected  10.5 MB
Ozeki Installer 10.1.57 Setup
OZEKI
Next >
Cancel

Welcome to Ozeki Installer 10.1.57 Setup

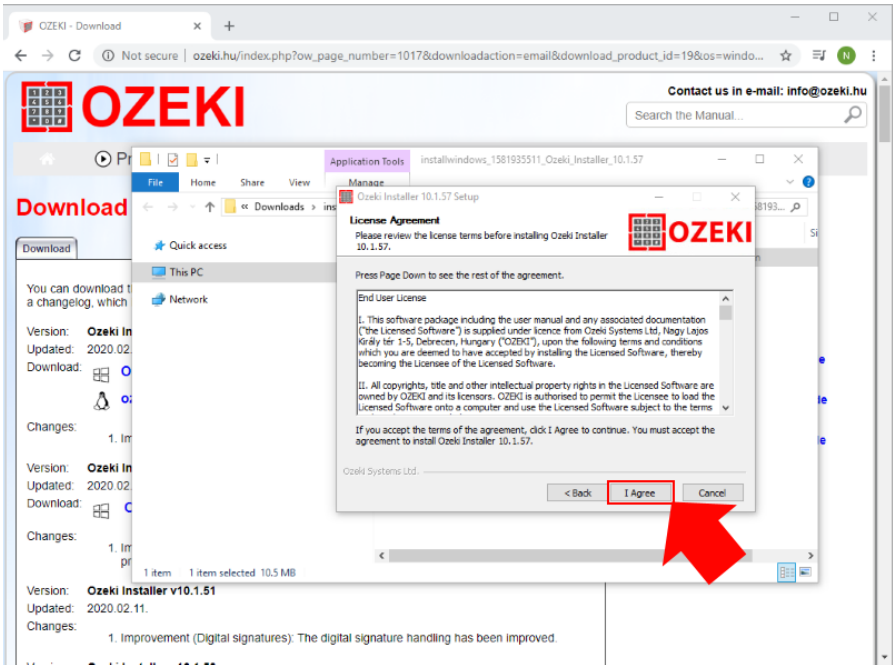
Setup will guide you through the installation of Ozeki Installer 10.1.57.

It is recommended that you close all other applications before starting Setup. This will make it possible to update relevant system files without having to reboot your computer.

Click Next to continue.

Figure 4 - Welcome to Ozeki Installer

To install Ozeki Installer on your computer, you must accept the terms of the License Agreement. Please review them then as **Figure 5** demonstrates, click '**I Agree**' button to continue.



Ozeki Installer 10.1.57 Setup

License Agreement

Please review the license terms before installing Ozeki Installer 10.1.57.

Press Page Down to see the rest of the agreement.

End User License

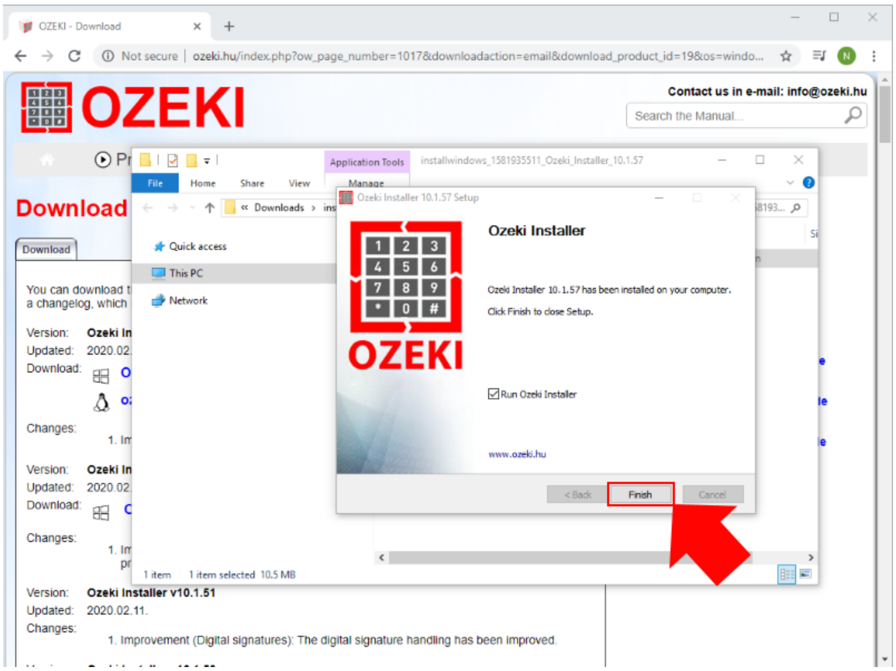
I. This software package including the user manual and any associated documentation ("the Licensed Software") is supplied under licence from Ozeki Systems Ltd, Nagy Lajos Király tér 1-5, Debrecen, Hungary ("OZEKI"), upon the following terms and conditions which you are deemed to have accepted by installing the Licensed Software, thereby becoming the Licensee of the Licensed Software.

II. All copyrights, title and other intellectual property rights in the Licensed Software are owned by OZEKI and its licensors. OZEKI is authorised to permit the Licensee to load the Licensed Software onto a computer and use the Licensed Software subject to the terms

If you accept the terms of the agreement, click I Agree to continue. You must accept the agreement to install Ozeki Installer 10.1.57.

Figure 5 - Accept license agreement

After the installation process is finished the Ozeki Installer service will start automatically in the background. To run Ozeki Installer on your web browser, just check **Run Ozeki Installer** as you can see it on **Figure 6** and click **Finish**.



Ozeki Installer 10.1.57 Setup

Ozeki Installer

Ozeki Installer 10.1.57 has been installed on your computer.

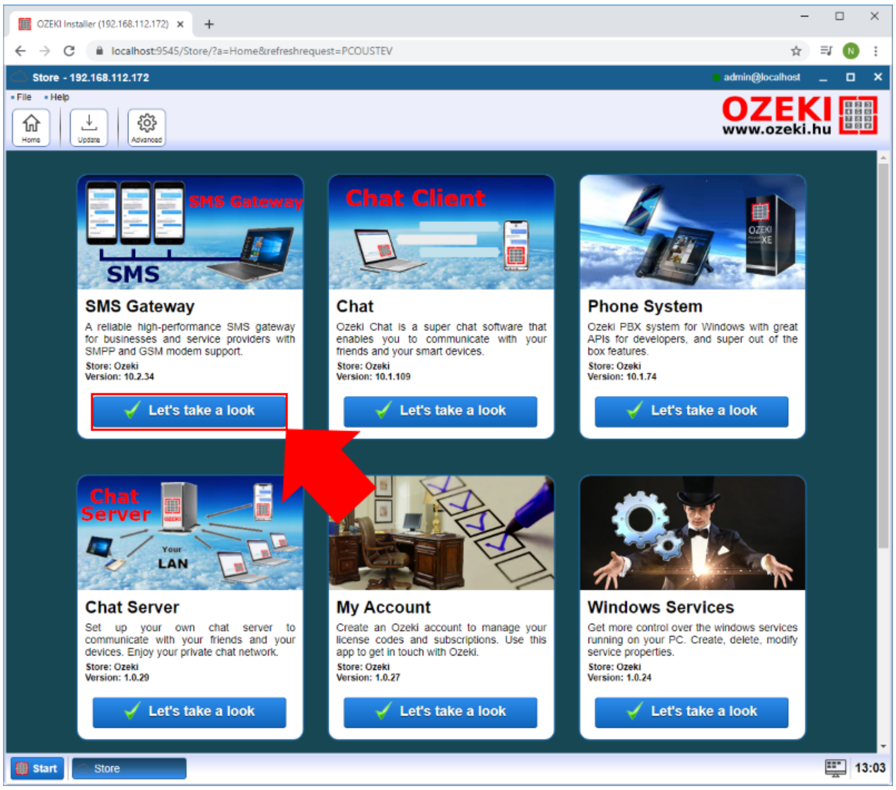
Click Finish to close Setup.

☑ Run Ozeki Installer

Figure 6 - Installation successful

# Step 5 - Select a product to install

The home page of Ozeki Installer is a great app store, where you can choose from many Ozeki applications as **Figure 7** shows it. By following this guide, you can install the SMS Gateway on your computer.



SMS Gateway
A reliable high-performance SMS gateway for businesses and service providers with SMPP and GSM modem support.
Store: Ozeki
Version: 10.2.34
Let's take a look

Chat
Ozeki Chat is a super chat software that enables you to communicate with your friends and your smart devices.
Store: Ozeki
Version: 10.1.109
Let's take a look

Phone System
Ozeki PBX system for Windows with great APIs for developers, and super out of the box features.
Store: Ozeki
Version: 10.1.74
Let's take a look

Chat Server
Set up your own chat server to communicate with your friends and your devices. Enjoy your private chat network.
Store: Ozeki
Version: 1.0.29
Let's take a look

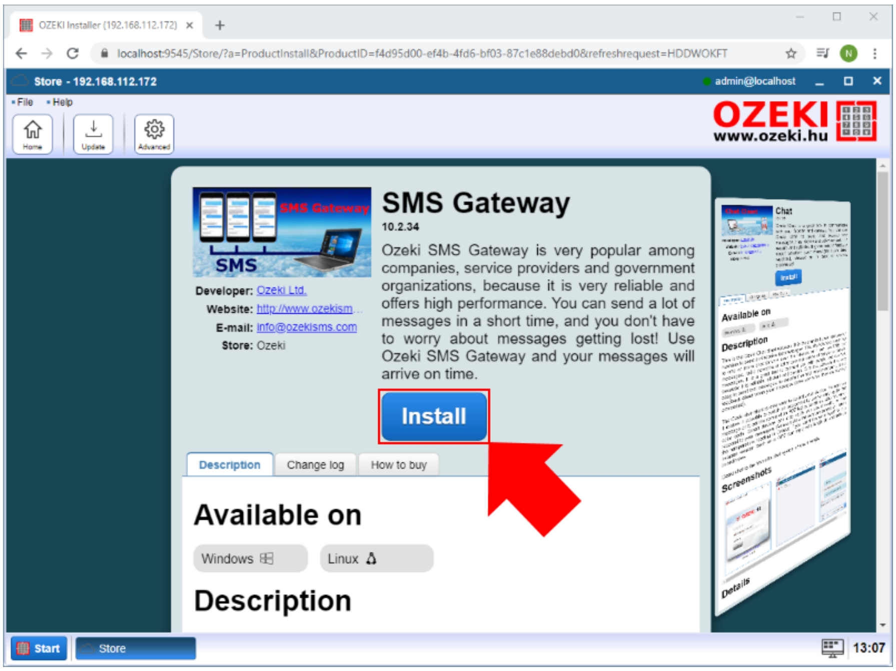
My Account
Create an Ozeki account to manage your license codes and subscriptions. Use this app to get in touch with Ozeki.
Store: Ozeki
Version: 1.0.27
Let's take a look

Windows Services
Get more control over the windows services running on your PC. Create, delete, modify service properties.
Store: Ozeki
Version: 1.0.24
Let's take a look

Figure 7 - Select SMS Gateway

# Step 6 - Install the SMS Gateway product

When you select a product, the next page that shows up is the details page of the selected app. Here you can see every bit of detail that gives you a brief description of the purpose and advantage of that application. If you would like to install the product on your computer, just click on **Install** as you can see it on **Figure 8**.



SMS Gateway
10.2.34
Ozeki SMS Gateway is very popular among companies, service providers and government organizations, because it is very reliable and offers high performance. You can send a lot of messages in a short time, and you don't have to worry about messages getting lost! Use Ozeki SMS Gateway and your messages will arrive on time.

Install

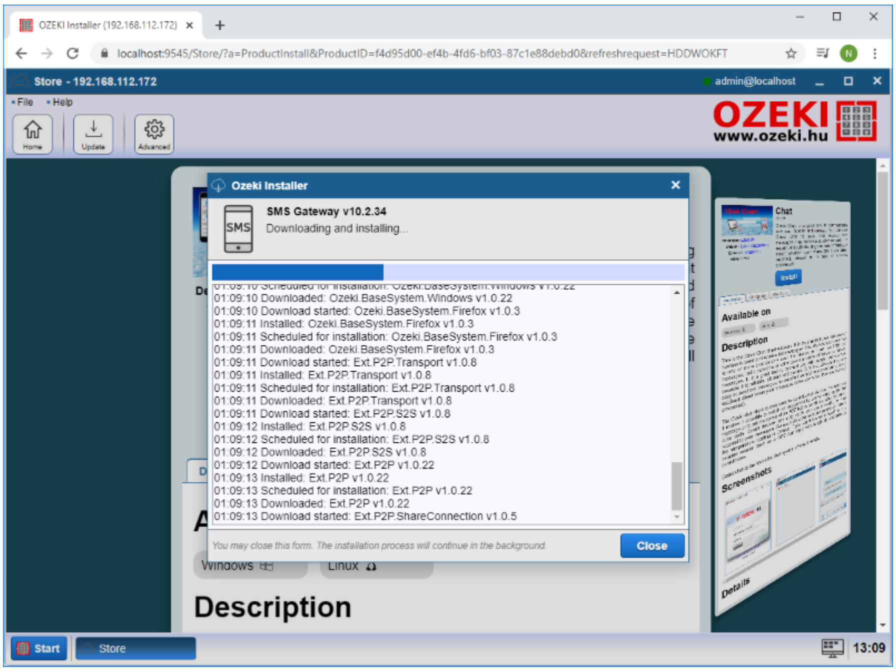
Developer: Ozeki Ltd.
Website: http://www.ozekism...
E-mail: info@ozekisms.com
Store: Ozeki

Available on

Description

Figure 8 - Install SMS Gateway

During the installation of the product, a dialog window shows up to inform you about the progress (**Figure 9**). This window logs you all events that occurred during the installation process.



OZEKI Installer (192.168.112.172)
localhost:9545/Store/?a=ProductInstall&ProductID=f4d95d00-ef4b-4fd6-bf03-87c1e88debd0&refreshrequest=HDDWOKFT
Store - 192.168.112.172
admin@localhost
File
Help
Home
Update
Advanced
OZEKI
www.ozeki.hu
Ozeki Installer
SMS
Close
Start
Store
13:09

SMS Gateway v10.2.34
Downloading and installing...

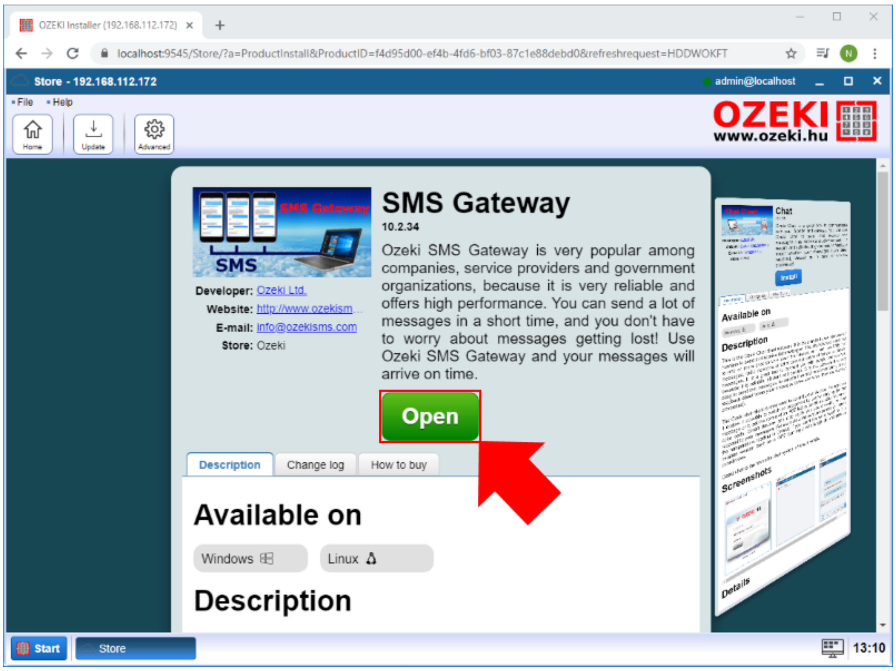
```
01:09:10 Scheduled for installation: Ozeki.BaseSystem.Windows v1.0.22
01:09:10 Downloaded: Ozeki.BaseSystem.Windows v1.0.22
01:09:10 Download started: Ozeki.BaseSystem.Firefox v1.0.3
01:09:11 Installed: Ozeki.BaseSystem.Firefox v1.0.3
01:09:11 Scheduled for installation: Ozeki.BaseSystem.Firefox v1.0.3
01:09:11 Downloaded: Ozeki.BaseSystem.Firefox v1.0.3
01:09:11 Download started: Ext.P2P.Transport v1.0.8
01:09:11 Installed: Ext.P2P.Transport v1.0.8
01:09:11 Scheduled for installation: Ext.P2P.Transport v1.0.8
01:09:11 Downloaded: Ext.P2P.Transport v1.0.8
01:09:11 Download started: Ext.P2P.S2S v1.0.8
01:09:12 Installed: Ext.P2P.S2S v1.0.8
01:09:12 Scheduled for installation: Ext.P2P.S2S v1.0.8
01:09:12 Downloaded: Ext.P2P.S2S v1.0.8
01:09:12 Download started: Ext.P2P v1.0.22
01:09:13 Installed: Ext.P2P v1.0.22
01:09:13 Scheduled for installation: Ext.P2P v1.0.22
01:09:13 Downloaded: Ext.P2P v1.0.22
01:09:13 Download started: Ext.P2P.ShareConnection v1.0.5
```

You may close this form. The installation process will continue in the background.

Figure 9 - Installation procedure

# Step 7 - Open SMS Gateway

After the installation is finished you can directly open the application with the '**Open**' button. Click on it like in **Figure 10** to open the SMS Gateway product in your web browser. The installed products runs on the port 9515.



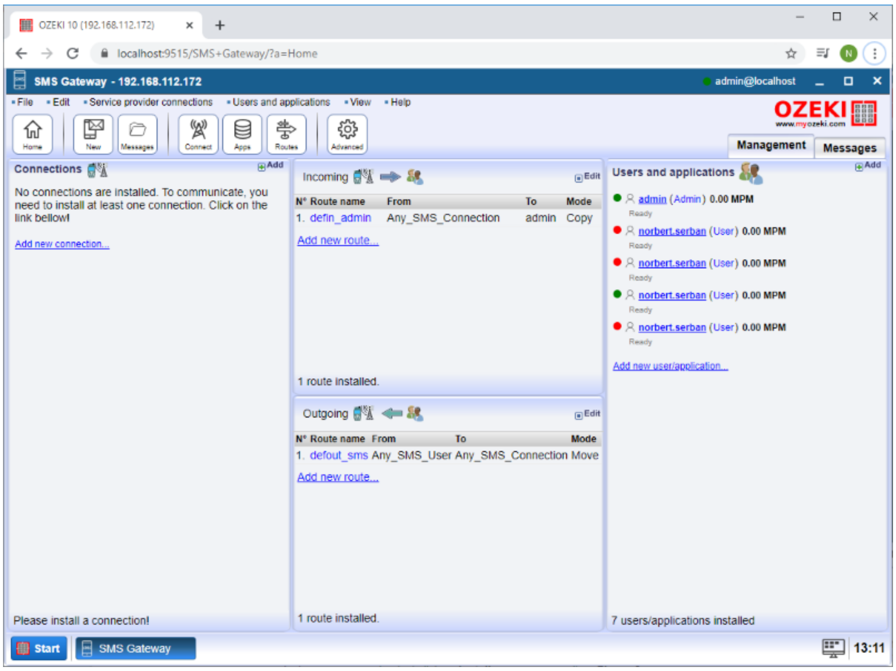
SMS Gateway
10.2.34

Developer: Ozeki Ltd.
Website: http://www.ozekism...
E-mail: info@ozekisms.com
Store: Ozeki

Ozeki SMS Gateway is very popular among companies, service providers and government organizations, because it is very reliable and offers high performance. You can send a lot of messages in a short time, and you don't have to worry about messages getting lost! Use Ozeki SMS Gateway and your messages will arrive on time.

Figure 10 - Open SMS Gateway

After clicking on the **Open** button, SMS Gateway start running on the port 9515 and opens in your web browser. If you can see the similar window as **Figure 11** that means you successfully set up Ozeki Installer and installed an Ozeki product from the app store.



Connections ⊞Add

No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

Please install a connection!

Incoming ⊞Edit

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defin_admin | Any_SMS_Connection | admin | Copy |

Add new route...

1 route installed.

Outgoing ⊞Edit

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move |

Add new route...

1 route installed.

Users and applications ⊞Add

- ● admin (Admin) 0.00 MPM
  Ready
- ● norbert.serban (User) 0.00 MPM
  Ready
- ● norbert.serban (User) 0.00 MPM
  Ready
- ● norbert.serban (User) 0.00 MPM
  Ready
- ● norbert.serban (User) 0.00 MPM
  Ready

Add new user/application...

7 users/applications installed

Figure 11 - SMS Gateway successfully installed

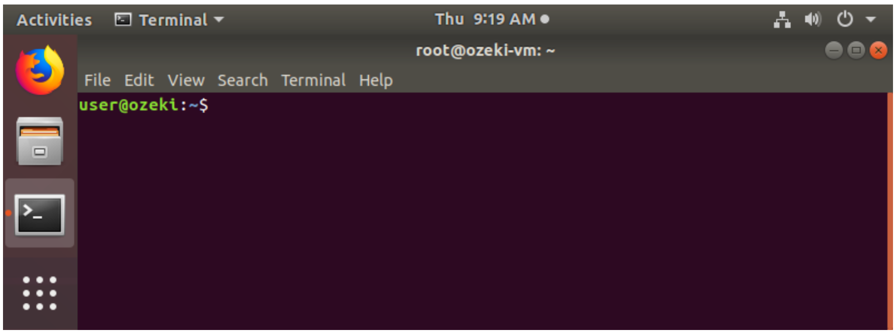
To be able to use an Ozeki 10 product, you need to purchase and activate it. To learn how you can activate an Ozeki 10 product please click on the following link: http://www.ozeki.hu/index.php?owpn=7689.

# How to install Ozeki on Linux

The next guide is about to demonstrate how you can set up Ozeki Installer on your Linux operating system. This product is a great app store that allows you to get any Ozeki product that you want by just one click. The installation process will take about 10 minutes. The guide contains four steps with clean instructions. The setup procedure does not require any further Linux knowledge. This document contains each command that you have to use during the installation. You just need to copy-paste them. Let's get started!

## Step 1 - Open a terminal

To perform the following installation steps, you need to open a terminal on your Linux host. To open a terminal windows, you can use the **Ctrl + Alt + T** shortkey (demonstrated in **Figure 1**).



```
user@ozeki:~$
```

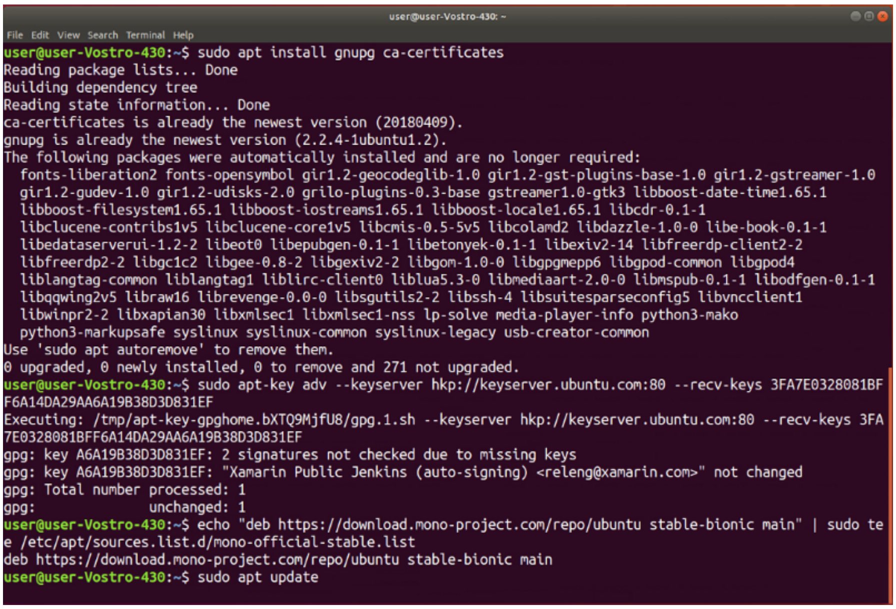
Figure 1 - Open a terminal

## Step 2 - Install all prerequisites

### Add the Mono repository to your system

To install Ozeki Installer first, you need to get some dependencies to perform the installation. The first one is the Mono, which is an open-source implementation of Microsoft's .NET Framework. This product is crucial for running Ozeki Installer. To get the latest version of Mono, you have to add the repository to your system as you can see it on **Figure 2** below. These commands can be found on the official Mono website: https://www.mono-project.com/download/stable/#download-lin-ubuntu

```
sudo apt install gnupg ca-certificates
sudo apt-key adv --keyserver hkp://keyserver.ubuntu.com:80 --recv-keys 3FA7E0328081BFF6A14DA29AA6A19B38
echo "deb https://download.mono-project.com/repo/ubuntu stable-bionic main" | sudo tee /etc/apt/sources
sudo apt update
```



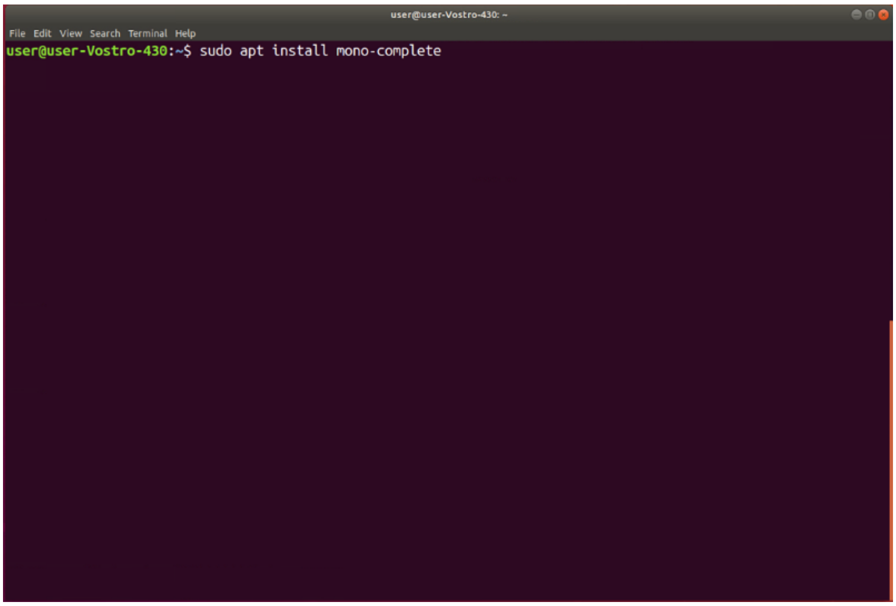
```
user@user-Vostro-430:~$ sudo apt install gnupg ca-certificates
Reading package lists... Done
Building dependency tree
Reading state information... Done
ca-certificates is already the newest version (20180409).
gnupg is already the newest version (2.2.4-1ubuntu1.2).
The following packages were automatically installed and are no longer required:
  fonts-liberation2 fonts-opensymbol gir1.2-geocodeglib-1.0 gir1.2-gst-plugins-base-1.0 gir1.2-gstreamer-1.0
  gir1.2-gudev-1.0 gir1.2-udisks-2.0 grilo-plugins-0.3-base gstreamer1.0-gtk3 libboost-date-time1.65.1
  libboost-filesystem1.65.1 libboost-iostreams1.65.1 libboost-locale1.65.1 libcdr-0.1-1
  libclucene-contribs1v5 libclucene-core1v5 libcmis-0.5-5v5 libcolamd2 libdazzle-1.0-0 libe-book-0.1-1
  libedataserverui-1.2-2 libeot0 libepubgen-0.1-1 libetonyek-0.1-1 libexiv2-14 libfreerdp-client2-2
  libfreerdp2-2 libgc1c2 libgee-0.8-2 libgexiv2-2 libgom-1.0-0 libgpgmepp6 libgpod-common libgpod4
  liblangtag-common liblangtag1 liblirc-client0 liblua5.3-0 libmediaart-2.0-0 libmspub-0.1-1 libodfgen-0.1-1
  libqqwing2v5 libraw16 librevenge-0.0-0 libsgutils2-2 libssh-4 libsuitesparseconfig5 libvncclient1
  libwinpr2-2 libxapian30 libxmlsec1 libxmlsec1-nss lp-solve media-player-info python3-mako
  python3-markupsafe syslinux syslinux-common syslinux-legacy usb-creator-common
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 271 not upgraded.
user@user-Vostro-430:~$ sudo apt-key adv --keyserver hkp://keyserver.ubuntu.com:80 --recv-keys 3FA7E0328081BF
F6A14DA29AA6A19B38D3D831EF
Executing: /tmp/apt-key-gpghome.bXTQ9MjfU8/gpg.1.sh --keyserver hkp://keyserver.ubuntu.com:80 --recv-keys 3FA
7E0328081BFF6A14DA29AA6A19B38D3D831EF
gpg: key A6A19B38D3D831EF: 2 signatures not checked due to missing keys
gpg: key A6A19B38D3D831EF: "Xamarin Public Jenkins (auto-signing) <releng@xamarin.com>" not changed
gpg: Total number processed: 1
gpg:              unchanged: 1
user@user-Vostro-430:~$ echo "deb https://download.mono-project.com/repo/ubuntu stable-bionic main" | sudo te
e /etc/apt/sources.list.d/mono-official-stable.list
deb https://download.mono-project.com/repo/ubuntu stable-bionic main
user@user-Vostro-430:~$ sudo apt update
```

Figure 2 - Adding the Mono repository to your system

### Install mono

The next step is to install Mono on your computer. This operation can be done by a single command as **Figure 3** demonstrates it.

```
sudo apt install mono-complete
```



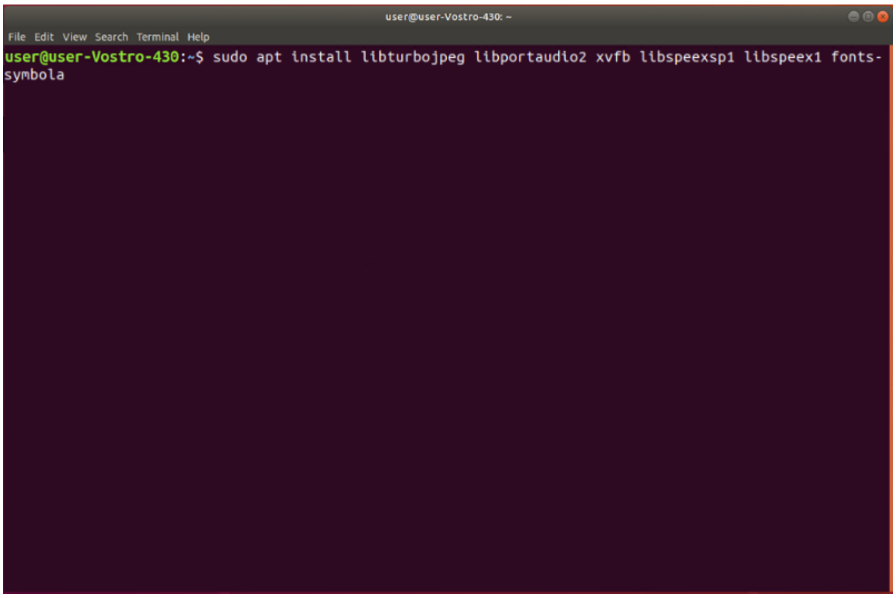
```
user@user-Vostro-430:~$ sudo apt install mono-complete
```

Figure 3 - Installing of Mono

### Install other required dependencies

The last step of the preparation is to install the other dependencies that needed for running Ozeki Installer properly. **Figure 4** shows the command that you have to type to install these prerequisites.

```
sudo apt install libturbojpeg libportaudio2 xvfb libspeexdsp1 libspeex1 fonts-symbola
```



```
user@user-Vostro-430:~$ sudo apt install libturbojpeg libportaudio2 xvfb libspeexsp1 libspeex1 fonts-
symbola
```

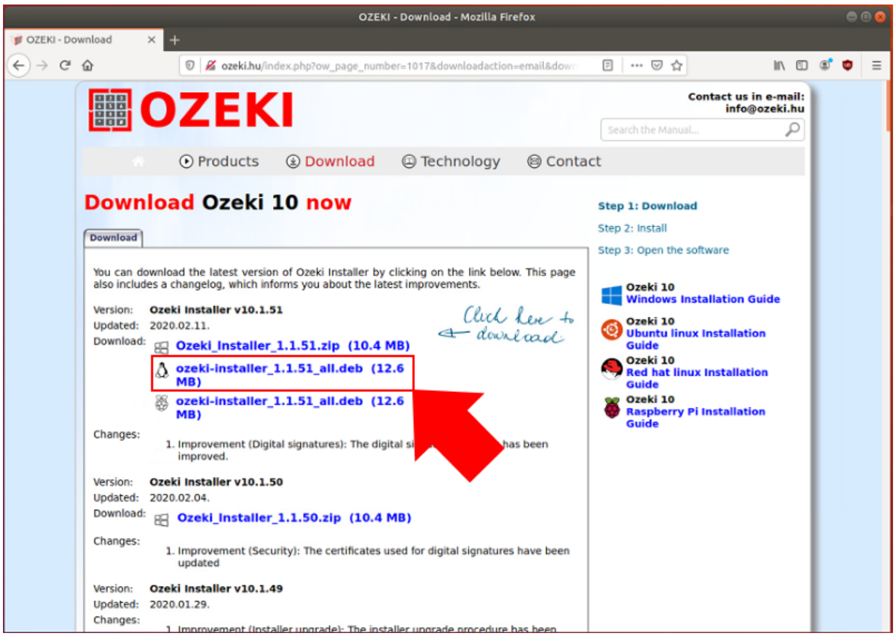
Figure 4 - Installing other dependencies

## Step 2 - Download Ozeki Installer

The very first step of getting Ozeki installer on your Linux operating system is to download is from www.ozeki.hu. The latest Ozeki products can be downloaded from the following Downloads page: http://www.ozeki.hu/index.php?owpn=1017&dpid=19 page. Now, on this page, as **Figure 5** shows, click on the .deb file of the Ozeki Installer to start downloading it.

You may download Ozeki Installer from the link below.
**http://www.ozeki.hu/index.php?owpn=1017&dpid=19**
**Download Ozeki Installer**

Click here to download


OZEKI - Download - Mozilla Firefox
OZEKI - Download
ozeki.hu/index.php?ow_page_number=1017&downloadaction=email&down
OZEKI
Contact us in e-mail:
info@ozeki.hu
Search the Manual...
Products
Download
Technology
Contact
Download
Step 1: Download
Step 2: Install
Step 3: Open the software
Ozeki 10
Windows Installation Guide
Ozeki 10
Ubuntu linux Installation Guide
Ozeki 10
Red hat linux Installation Guide
Ozeki 10
Raspberry Pi Installation Guide
Click here to download

Download Ozeki 10 now

You can download the latest version of Ozeki Installer by clicking on the link below. This page also includes a changelog, which informs you about the latest improvements.

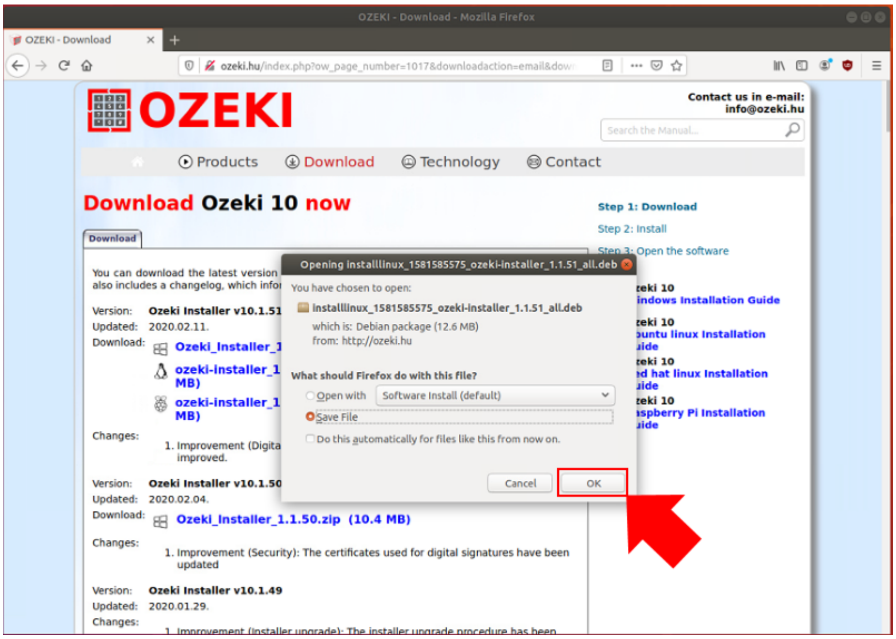
Version: Ozeki Installer v10.1.51
Updated: 2020.02.11.
Download: Ozeki_Installer_1.1.51.zip (10.4 MB)
ozeki-installer_1.1.51_all.deb (12.6 MB)
ozeki-installer_1.1.51_all.deb (12.6 MB)
Changes:
1. Improvement (Digital signatures): The digital si has been improved.

Version: Ozeki Installer v10.1.50
Updated: 2020.02.04.
Download: Ozeki_Installer_1.1.50.zip (10.4 MB)
Changes:
1. Improvement (Security): The certificates used for digital signatures have been updated

Version: Ozeki Installer v10.1.49
Updated: 2020.01.29.
Changes:
1. Improvement (Installer upgrade): The installer upgrade procedure has been

Figure 5 - Download Ozeki Installer

The browser usually asks for how to download the file. It can be opened by a selected software or just downloaded to the designated download folder. In the case of Ozeki Installer just check Save file and click on OK just as **Figure 6** demonstrates it.



Download Ozeki 10 now

You can download the latest version
also includes a changelog, which infor

Version: Ozeki Installer v10.1.51
Updated: 2020.02.11.
Download: Ozeki_Installer_1
ozeki-installer_1
MB)
ozeki-installer_1
MB)
Changes:
1. Improvement (Digita
improved.

Version: Ozeki Installer v10.1.50
Updated: 2020.02.04.
Download: Ozeki_Installer_1.1.50.zip (10.4 MB)
Changes:
1. Improvement (Security): The certificates used for digital signatures have been updated

Version: Ozeki Installer v10.1.49
Updated: 2020.01.29.
Changes:
1. Improvement (Installer upgrade): The installer upgrade procedure has been

zeki 10
indows Installation Guide
zeki 10
buntu linux Installation
uide
zeki 10
ed hat linux Installation
uide
zeki 10
aspberry Pi Installation
uide

Opening installlinux_1581585575_ozeki-installer_1.1.51_all.deb

You have chosen to open:

installlinux_1581585575_ozeki-installer_1.1.51_all.deb

which is: Debian package (12.6 MB)
from: http://ozeki.hu

What should Firefox do with this file?

○ Open with Software Install (default)

● Save File

☐ Do this automatically for files like this from now on.

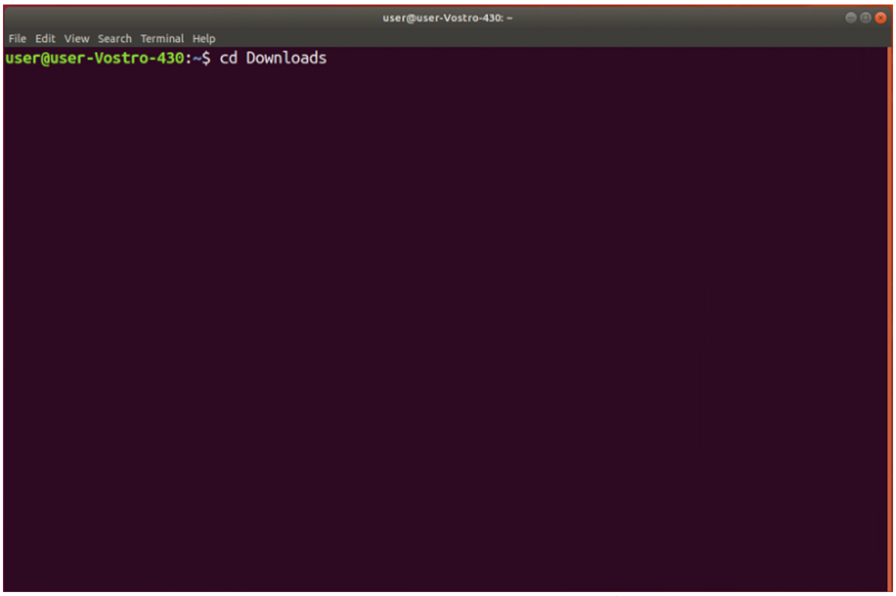
Cancel OK

Figure 6 - Save Ozeki Installer .deb file

# Step 3 - Install Ozeki Installer

The next step is the main part of this guide by showing the procedure of installing Ozeki Installer to Linux operating system. To do that, first open **Terminal**. The first thing that you have to perform in Terminal, is to move to the folder where the .deb file was downloaded. In that case, it is the Downloads folder, so as you can see it on **Figure 7**, with the **cd** command followed by the name of the directory navigate the control to that folder.

```
cd Downloads
```

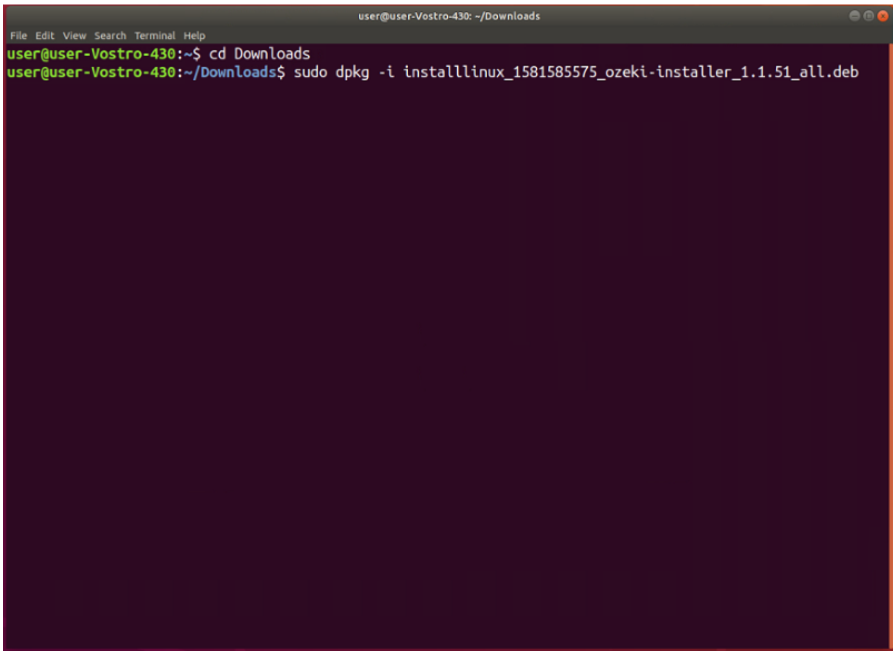


```
user@user-Vostro-430:~$ cd Downloads
```

Figure 7 - Move to Downloads folder

On Linux operating system, .deb files can be installed by using the **dpkg** command. To perform an installation, add **-i** option after the command and type the name of the downloaded .deb file. This process requires administrator permission, so you have to type **sudo** at the beginning of the command (**Figure 8**).

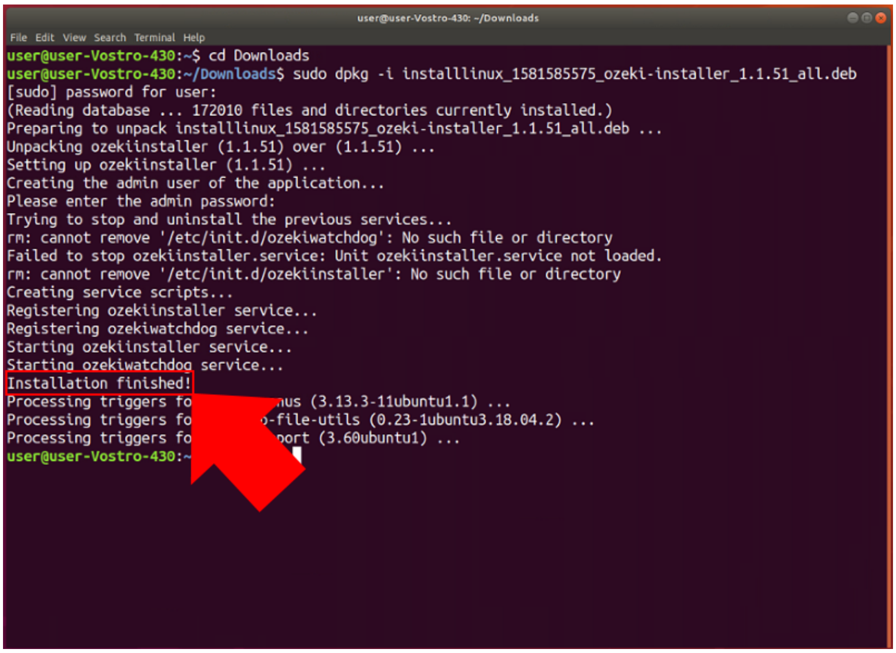
```
sudo dpkg -i installinux_1581585575_ozeki-installer_1.1.51_all.deb
```


user@user-Vostro-430: ~/Downloads
File Edit View Search Terminal Help

```
user@user-Vostro-430:~$ cd Downloads
user@user-Vostro-430:~/Downloads$ sudo dpkg -i installlinux_1581585575_ozeki-installer_1.1.51_all.deb
```

Figure 8 - Install Ozeki Installer

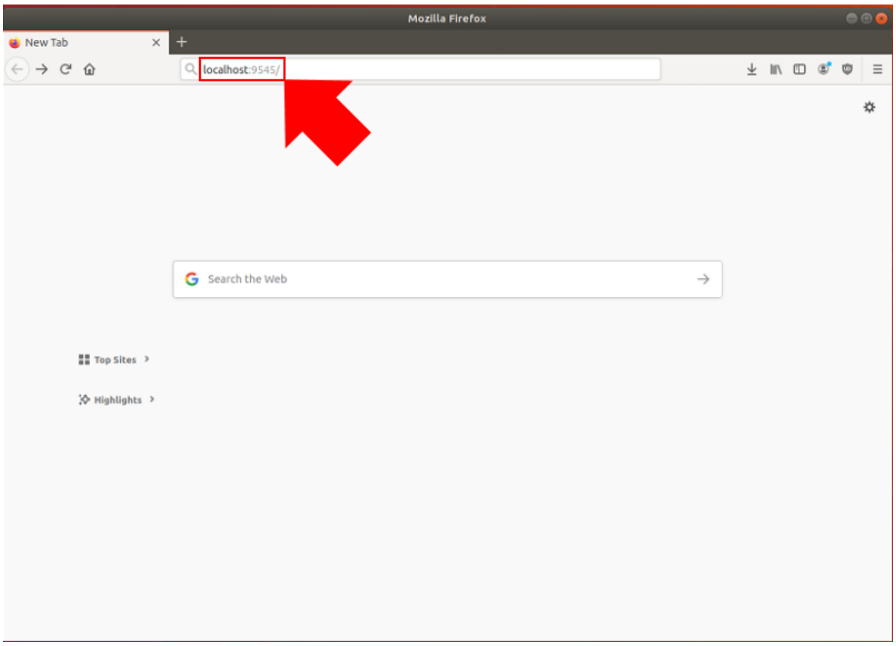
When you hit Enter, the installation process begins. If you have done everything right, Ozeki Installer is going to be installed on your computer just after a few seconds. To make sure about the success of the installation, just search for the **Installation finished!** line in the logs of Terminal as on **Figure 9**.


user@user-Vostro-430: ~/Downloads
File Edit View Search Terminal Help

```
user@user-Vostro-430:~$ cd Downloads
user@user-Vostro-430:~/Downloads$ sudo dpkg -i installlinux_1581585575_ozeki-installer_1.1.51_all.deb
[sudo] password for user:
(Reading database ... 172010 files and directories currently installed.)
Preparing to unpack installlinux_1581585575_ozeki-installer_1.1.51_all.deb ...
Unpacking ozekiinstaller (1.1.51) over (1.1.51) ...
Setting up ozekiinstaller (1.1.51) ...
Creating the admin user of the application...
Please enter the admin password:
Trying to stop and uninstall the previous services...
rm: cannot remove '/etc/init.d/ozekiwatchdog': No such file or directory
Failed to stop ozekiinstaller.service: Unit ozekiinstaller.service not loaded.
rm: cannot remove '/etc/init.d/ozekiinstaller': No such file or directory
Creating service scripts...
Registering ozekiinstaller service...
Registering ozekiwatchdog service...
Starting ozekiinstaller service...
Starting ozekiwatchdog service...
Installation finished!
Processing triggers fo          nus (3.13.3-11ubuntu1.1) ...
Processing triggers fo         -file-utils (0.23-1ubuntu3.18.04.2) ...
Processing triggers fo          port (3.60ubuntu1) ...
user@user-Vostro-430:~
```

Figure 9 - Successful installation

After the successful installation, now you should be able to use Ozeki Installer. For that, just open up your web browser and type **localhost:9545** since Ozeki Installer runs on the port 9545 on your computer. To open Ozeki Installer just hit Enter.

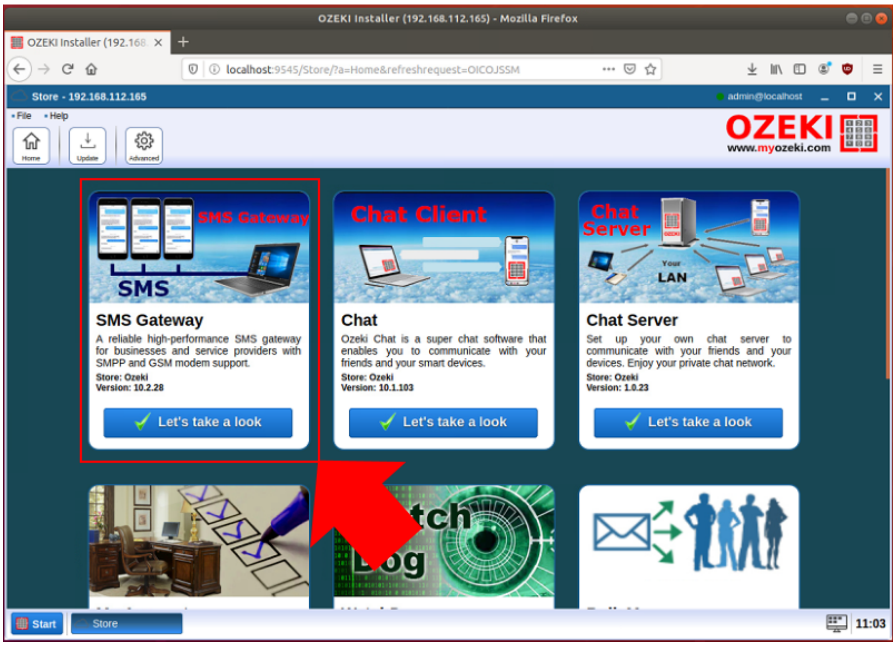


localhost:9545/

Figure 10 - Open Ozeki Installer

# Step 4 - Install SMS Gateway using Ozeki Installer

The main menu of Ozeki Installer is an app store, where you can choose from many Ozeki products. By following this guide, you are going to install the SMS Gateway application. To open its details page, just click on the tile of the app.



localhost:9545/Store/?a=Home&refreshrequest=OICOJSSM

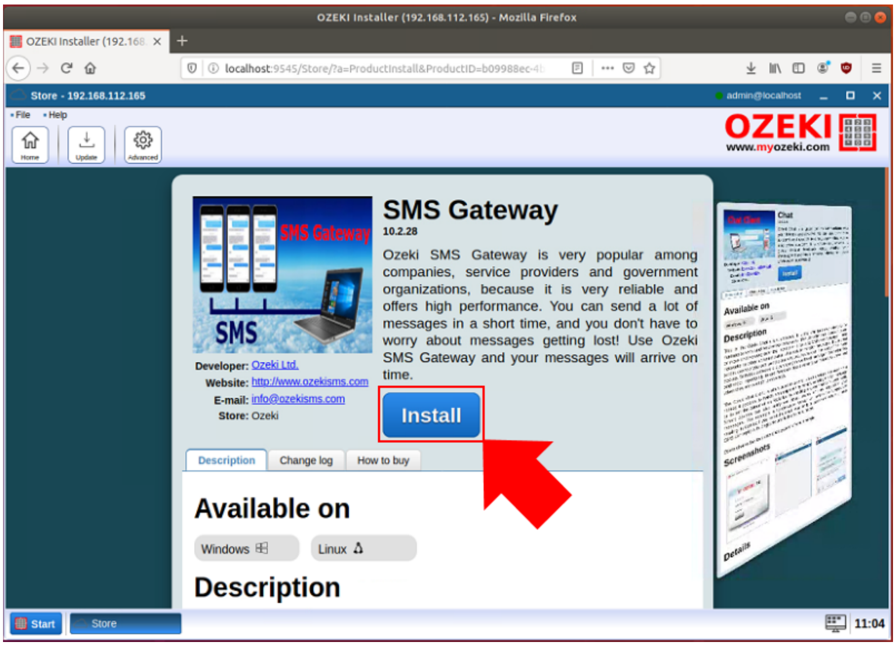
**SMS Gateway**
A reliable high-performance SMS gateway for businesses and service providers with SMPP and GSM modem support.
Store: Ozeki
Version: 10.2.28

**Chat**
Ozeki Chat is a super chat software that enables you to communicate with your friends and your smart devices.
Store: Ozeki
Version: 10.1.103

**Chat Server**
Set up your own chat server to communicate with your friends and your devices. Enjoy your private chat network.
Store: Ozeki
Version: 1.0.23

Figure 11 - Select SMS Gateway

The details page of each product contains all important information about the selected application such as version number, descriptions and screenshots. To start the installation of the product just click on the **Install** as you can see it on **Figure 12**.



SMS Gateway

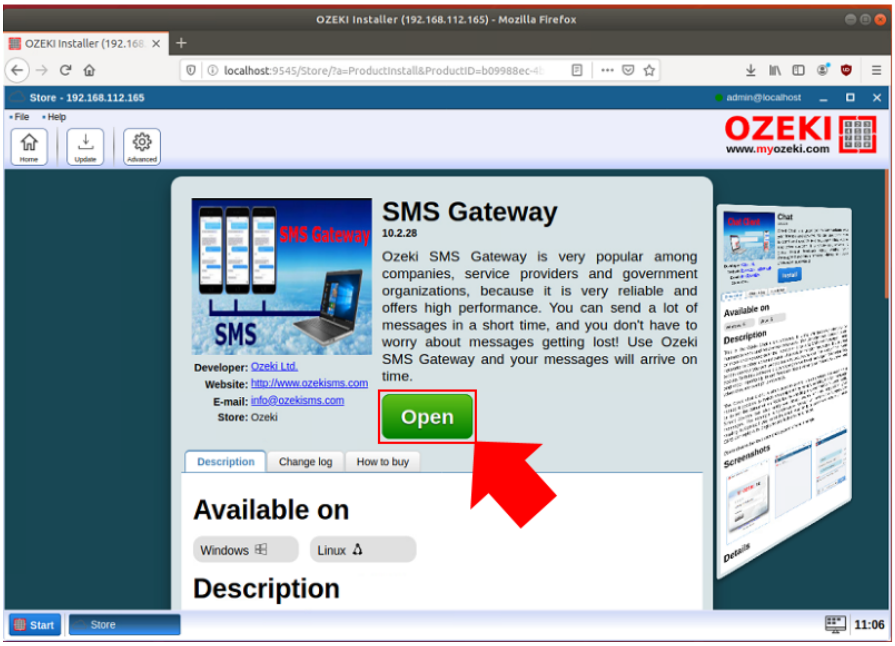
10.2.28

Ozeki SMS Gateway is very popular among companies, service providers and government organizations, because it is very reliable and offers high performance. You can send a lot of messages in a short time, and you don't have to worry about messages getting lost! Use Ozeki SMS Gateway and your messages will arrive on time.

Developer: Ozeki Ltd.
Website: http://www.ozekisms.com
E-mail: info@ozekisms.com
Store: Ozeki

Figure 12 - Install SMS Gateway

After clicking on the Install button, a progress bar shows the status of the installation process. When it finished successfully, the blue button switches to a green **Open** button. By clicking that, you can open the installed SMS Gateway.



SMS Gateway

10.2.28

Ozeki SMS Gateway is very popular among companies, service providers and government organizations, because it is very reliable and offers high performance. You can send a lot of messages in a short time, and you don't have to worry about messages getting lost! Use Ozeki SMS Gateway and your messages will arrive on time.

Developer: Ozeki Ltd.
Website: http://www.ozekisms.com
E-mail: info@ozekisms.com
Store: Ozeki

Figure 13 - Open SMS Gateway

The installed applications can be reached at port 9515. So, the installed SMS Gateway opens up on that port as well. This desktop shows the products that were installed previously on your computer. To open SMS Gateway, click on its icon (**Figure 14**)

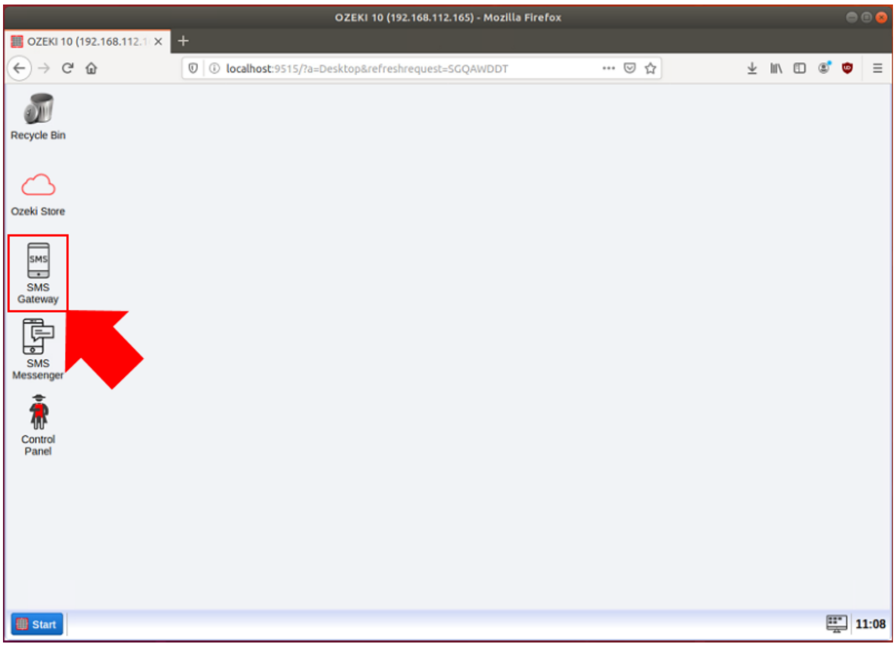
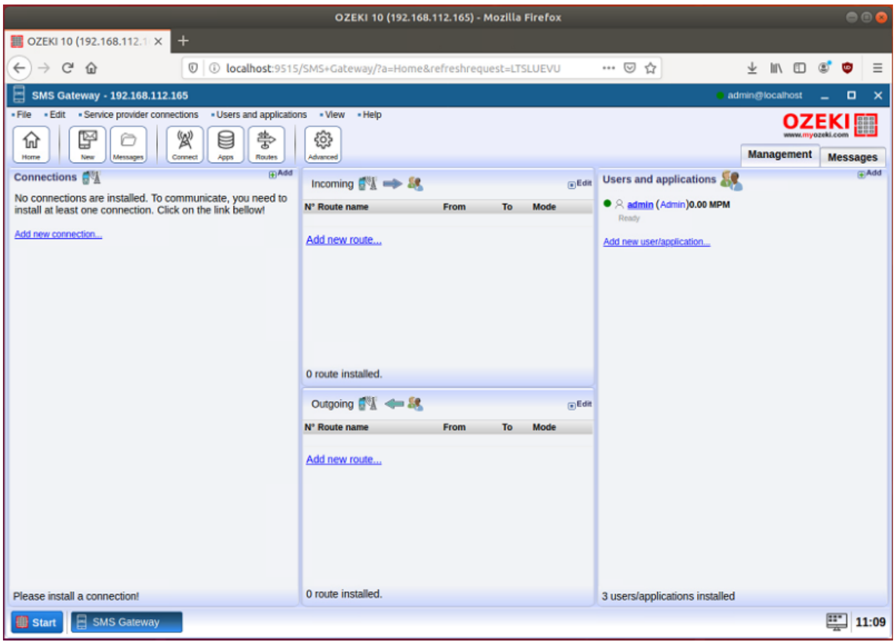

OZEKI 10 (192.168.112.165) - Mozilla Firefox
OZEKI 10 (192.168.112.1
localhost:9515/?a=Desktop&refreshrequest=SGQAWDDT
Recycle Bin
Ozeki Store
SMS Gateway
SMS Messenger
Control Panel
Start
11:08

Figure 14 - Desktop of SMS Gateway

And finally, now you will be able to see the main menu of the installed SMS Gateway, which is ready to use.



No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

0 route installed.

Please install a connection!

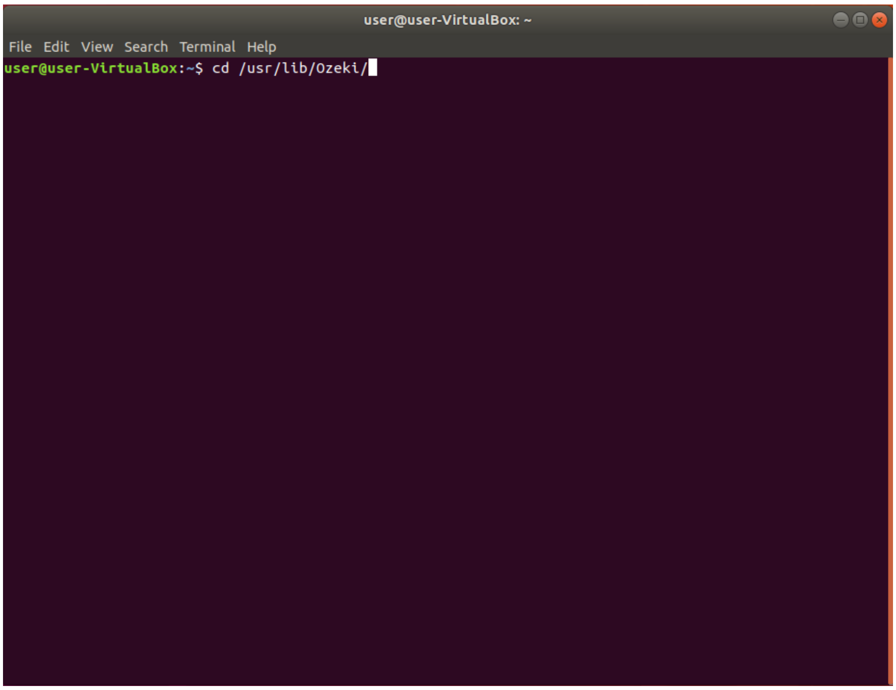
3 users/applications installed

Figure 15 - Main menu of SMS Gateway

# Step 5 - Set a root password for remote access

For security reasons, it is not possible to log in remotely in Ozeki 10 by default. For remote access to work, a secure password must be set for admin users. You can do this by running a script called install-password.sh and it is located in the Ozeki folder. To do this, go to the Ozeki folder with the following command:

```
cd /usr/lib/Ozeki
```

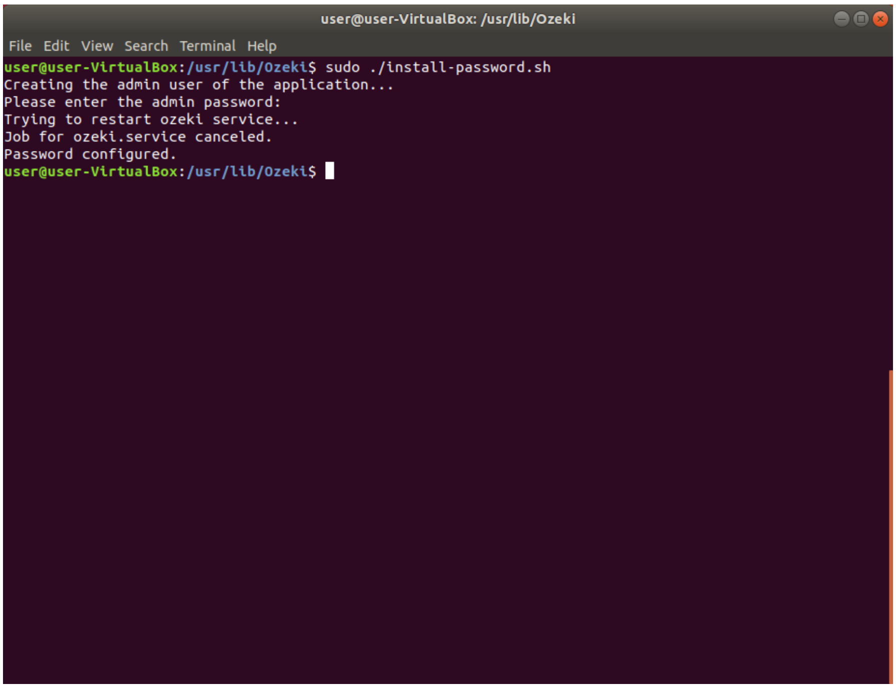


```
user@user-VirtualBox:~$ cd /usr/lib/Ozeki/
```

Figure 16 - Change to Ozeki directory

Then run the install-password.sh script with the following command. The script will ask for the new password and then set it for the admin user.

```
sudo ./install-password.sh
```

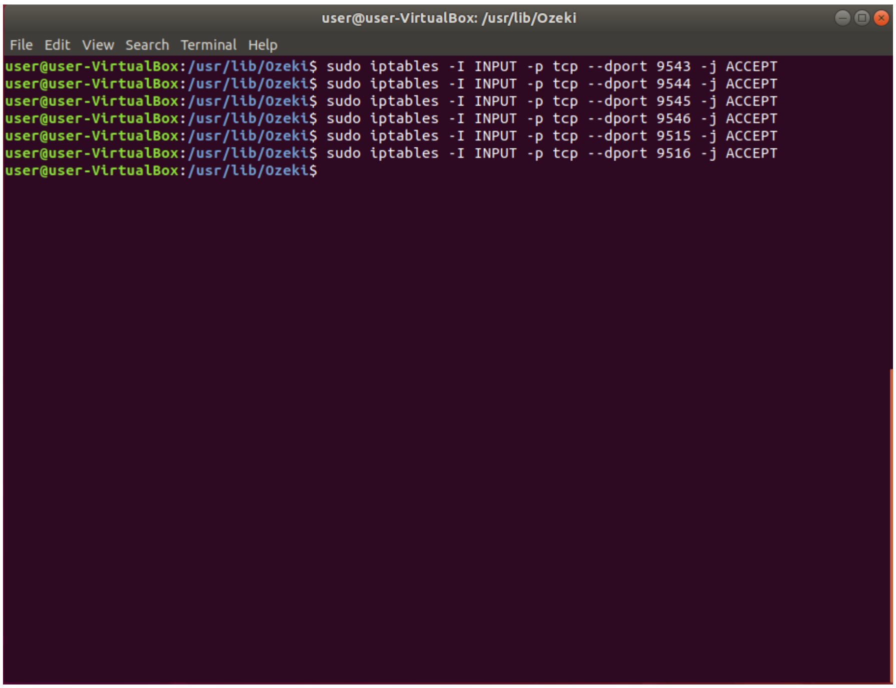

user@user-VirtualBox: /usr/lib/Ozeki
File Edit View Search Terminal Help

```
user@user-VirtualBox:/usr/lib/Ozeki$ sudo ./install-password.sh
Creating the admin user of the application...
Please enter the admin password:
Trying to restart ozeki service...
Job for ozeki.service canceled.
Password configured.
user@user-VirtualBox:/usr/lib/Ozeki$
```

Figure 17 - Run install-password.sh

Finally, you need to enable the appropriate ports on your linux firewall in order to access Ozeki 10 from the outside. To do this, you need to run the following commands. 9543 is the http port and 9544 is the websocket port, 9545 is the https port and 9546 is the secure websocket port.

```
sudo iptables -I INPUT -p tcp --dport 9543 -j ACCEPT
sudo iptables -I INPUT -p tcp --dport 9544 -j ACCEPT
sudo iptables -I INPUT -p tcp --dport 9545 -j ACCEPT
sudo iptables -I INPUT -p tcp --dport 9546 -j ACCEPT
sudo iptables -I INPUT -p tcp --dport 9515 -j ACCEPT
sudo iptables -I INPUT -p tcp --dport 9516 -j ACCEPT
```



user@user-VirtualBox: /usr/lib/Ozeki
File Edit View Search Terminal Help

```
user@user-VirtualBox:/usr/lib/Ozeki$ sudo iptables -I INPUT -p tcp --dport 9543 -j ACCEPT
user@user-VirtualBox:/usr/lib/Ozeki$ sudo iptables -I INPUT -p tcp --dport 9544 -j ACCEPT
user@user-VirtualBox:/usr/lib/Ozeki$ sudo iptables -I INPUT -p tcp --dport 9545 -j ACCEPT
user@user-VirtualBox:/usr/lib/Ozeki$ sudo iptables -I INPUT -p tcp --dport 9546 -j ACCEPT
user@user-VirtualBox:/usr/lib/Ozeki$ sudo iptables -I INPUT -p tcp --dport 9515 -j ACCEPT
user@user-VirtualBox:/usr/lib/Ozeki$ sudo iptables -I INPUT -p tcp --dport 9516 -j ACCEPT
user@user-VirtualBox:/usr/lib/Ozeki$
```

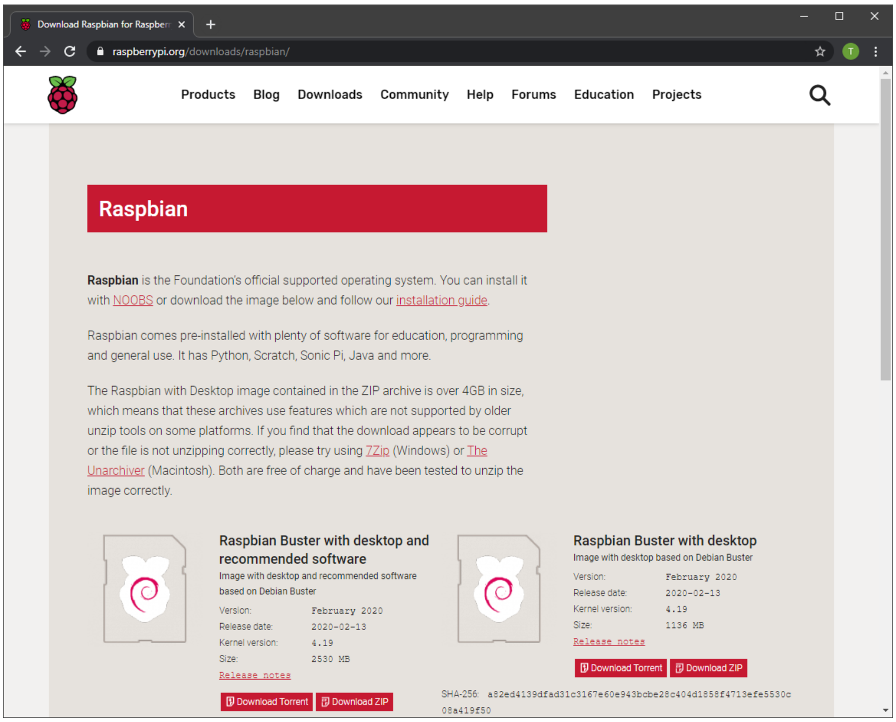
Figure 18 - Configure IP tables

# How to install Ozeki on Raspberry Pi

**On this page, you will find a detailed guide about how you can set up Ozeki Installer on Raspberry Pi. By installing this software, you will have access to every Ozeki product from Ozeki Installer's app store. The installation process takes about 10 to 15 minutes. This guide does not require any specific knowledge since each command listed in the document, so you need to just copy-paste them. So, do not waste any time, and let's get started now!**

## Step 1 - Download Raspbian

First step is to download Raspbian operation system. It can be downloaded from the Raspberry Pi offical page. You can reach that page by clicking on the following link: https://www.raspberrypi.org/downloads/raspbian/. **Figure 1** shows the official Raspberry Pi page.


Download Raspbian for Raspberr
raspberrypi.org/downloads/raspbian/
Products
Blog
Downloads
Community
Help
Forums
Education
Projects

Raspbian

**Raspbian** is the Foundation's official supported operating system. You can install it with NOOBS or download the image below and follow our installation guide.

Raspbian comes pre-installed with plenty of software for education, programming and general use. It has Python, Scratch, Sonic Pi, Java and more.

The Raspbian with Desktop image contained in the ZIP archive is over 4GB in size, which means that these archives use features which are not supported by older unzip tools on some platforms. If you find that the download appears to be corrupt or the file is not unzipping correctly, please try using 7Zip (Windows) or The Unarchiver (Macintosh). Both are free of charge and have been tested to unzip the image correctly.

Raspbian Buster with desktop and recommended software

Image with desktop and recommended software based on Debian Buster

| | |
|---|---|
| Version: | February 2020 |
| Release date: | 2020-02-13 |
| Kernel version: | 4.19 |
| Size: | 2530 MB |

Release notes

Download Torrent Download ZIP

Raspbian Buster with desktop

Image with desktop based on Debian Buster

| | |
|---|---|
| Version: | February 2020 |
| Release date: | 2020-02-13 |
| Kernel version: | 4.19 |
| Size: | 1136 MB |

Release notes

Download Torrent Download ZIP

SHA-256: a82ed4139dfad31c3167e60e943bcbe28c404d1858f4713efe5530c08a419f50

Figure 1 - Download Raspbian

## Step 2 - Write Raspbian to SD Card

After you downloaded the Raspbian operation system, you have to write it to an SD Card. You can perform it with the win32diskimager software which can be downloaded here: https://sourceforge.net/projects/win32diskimager/. In the win32diskimager browse the downloaded Raspbian image file and select the device then click on Write as you can see it on **Figure 2**.

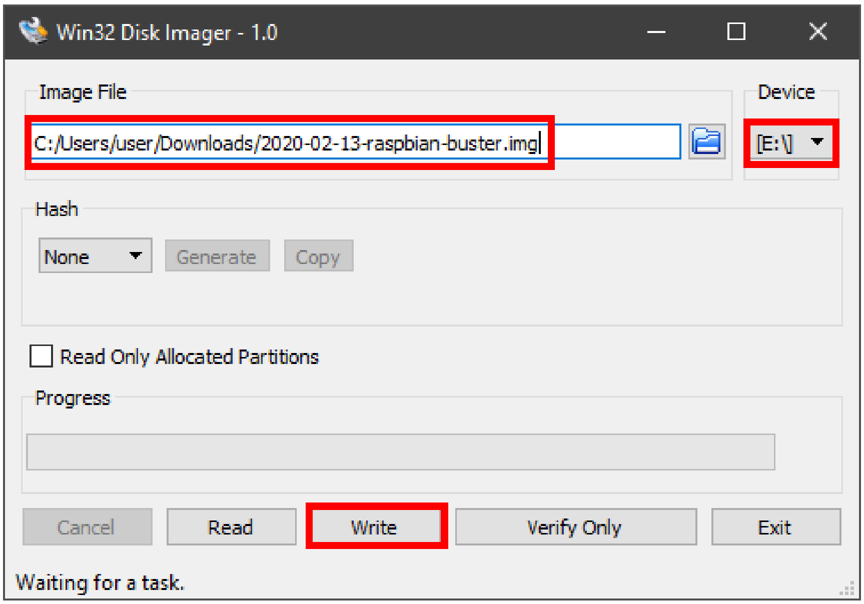

Win32 Disk Imager - 1.0

Image File: C:/Users/user/Downloads/2020-02-13-raspbian-buster.img

Device: [E:\]

Hash: None

☐ Read Only Allocated Partitions

Waiting for a task.

Figure 2 - Write Raspbian to SD Card

The writing process could take a bit of time, but as soon as it finished, a little dialog window shows up as **Figure 3** demonstrates. This window informs you about the success of writing.

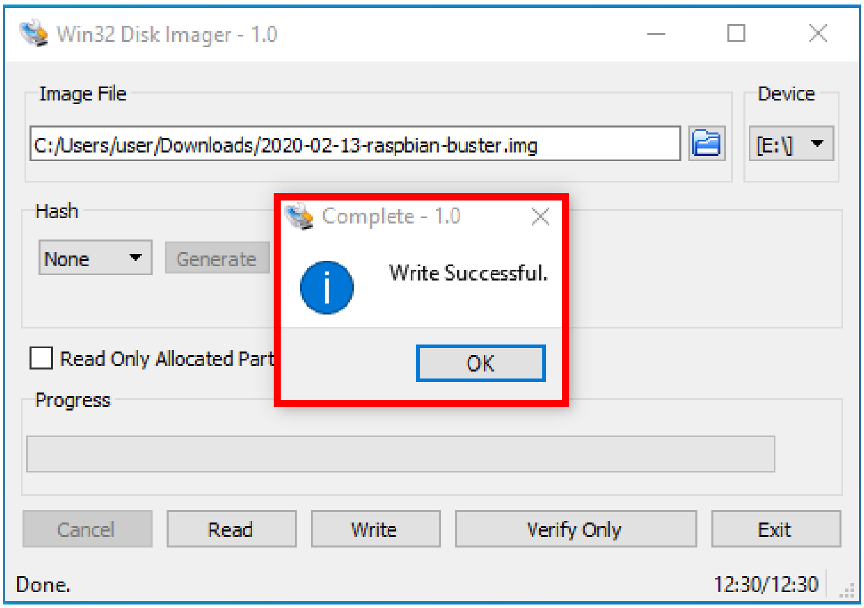

Win32 Disk Imager - 1.0

Image File: C:/Users/user/Downloads/2020-02-13-raspbian-buster.img

Device: [E:\]

Hash: None

☐ Read Only Allocated Part

Complete - 1.0

Write Successful.

Done. 12:30/12:30

Figure 3 - Write Successful

## Step 3 - Open terminal

After you inserted the card to the raspberry pie, boot the system. It will take a while but after it is done, you will see the graphical user interface of the Raspbian operating system. Please click to the terminal button (**Figure 4**)

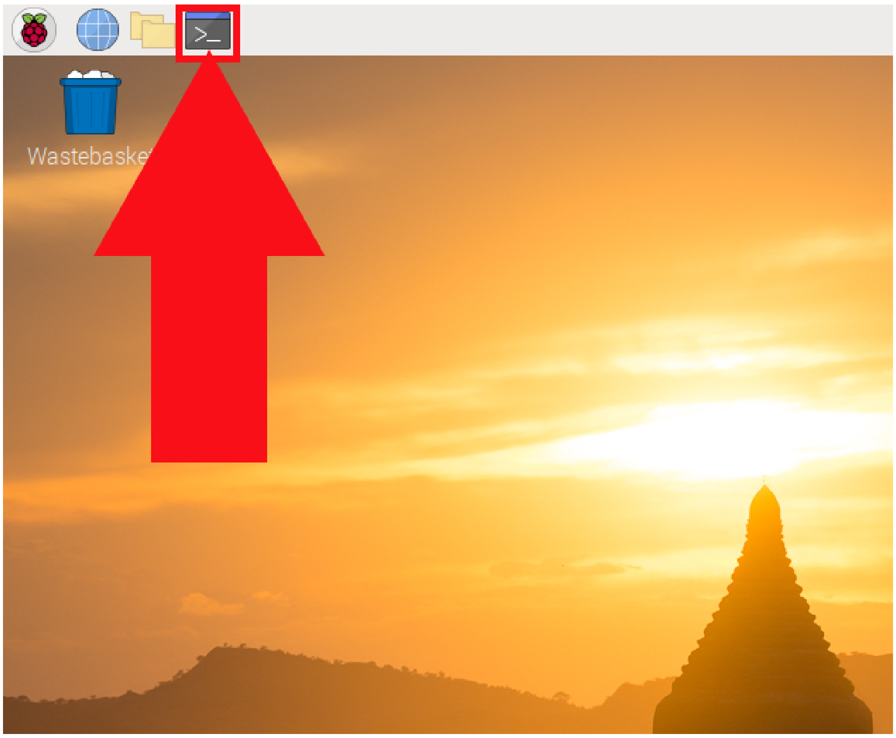
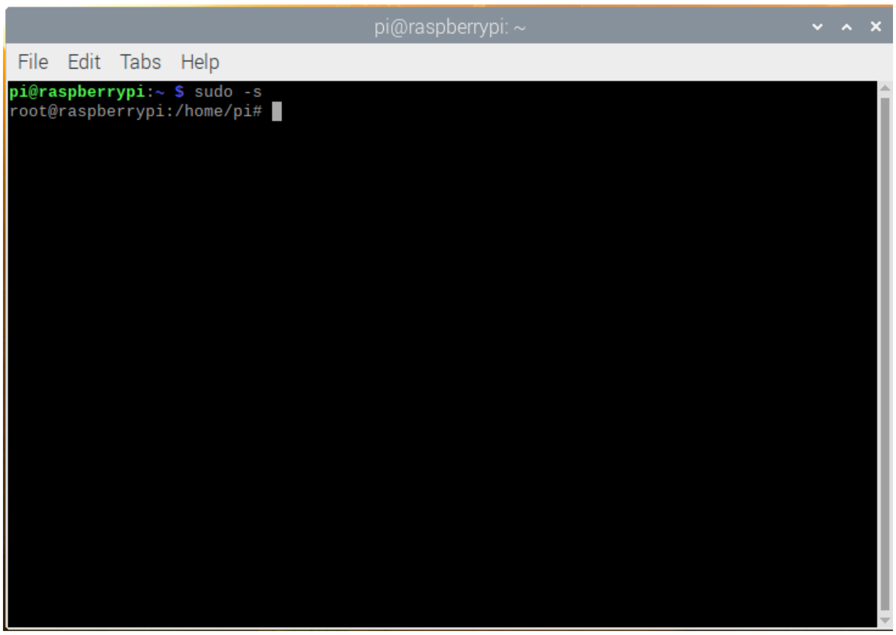


Figure 4 - See how you can open terminal

## Step 3 - Get root privileges

You need to have root privileges to install the Ozeki software and its dependencies. To perform this, issue **sudo -s** command at the beginning of the installation. This command delegates the authority to you to run all forthcoming commands as root. Due to this, your Linux operating system will not require you to use 'sudo' command before each command that need root privileges to perform. Open Terminal using **Ctrl + Alt + T** shortkey (by default) then issue the following command (demonstrated in **Figure 5**).

```
sudo -s
```



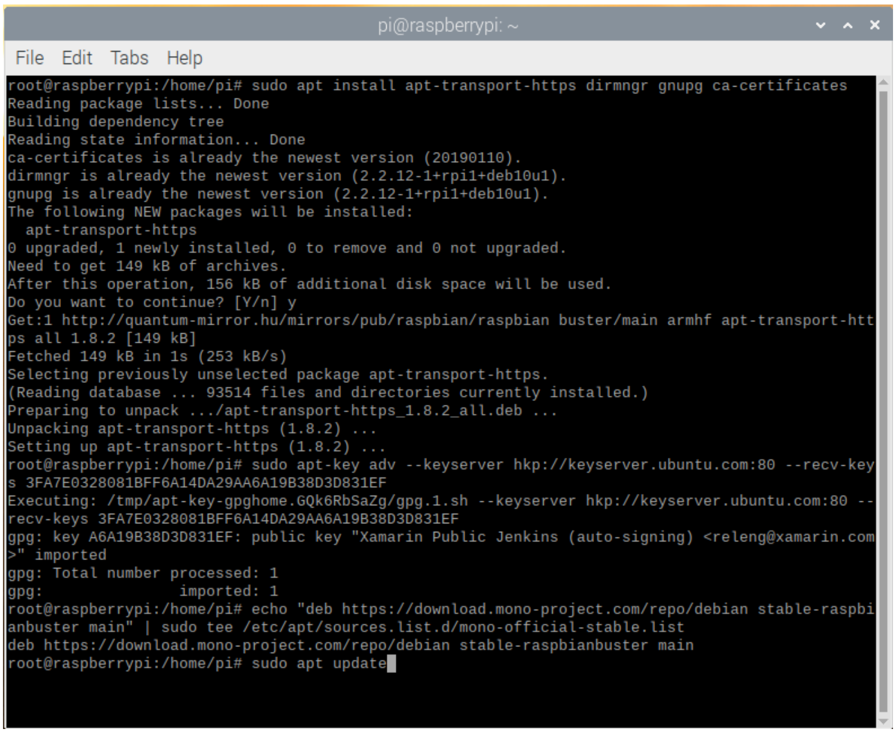
```
pi@raspberrypi:~ $ sudo -s
root@raspberrypi:/home/pi#
```

Figure 5 - Get root privileges

## Step 4 - Add the Mono repository to your system

Mono is an open source implementation of Microsoft's .NET Framework. Ozeki Installer needs it to operate. To be able to install the latest release of Mono, you need to add the official Mono repository to your system. If you are running **Raspbian 10** on Raspberry Pi, you can use the commands below to perform this like in **Figure 6**. If you are running another version of Raspbian, please follow the instructions on the official Mono site at https://www.mono-project.com/download/stable/#download-lin-raspbian.

```
sudo apt install apt-transport-https dirmngr gnupg ca-certificates
sudo apt-key adv --keyserver hkp://keyserver.ubuntu.com:80 --recv-keys 3FA7E0328081BFF6A14DA29AA6A19B38
echo "deb https://download.mono-project.com/repo/debian stable-raspbianbuster main" | sudo tee /etc/apt
sudo apt update
```



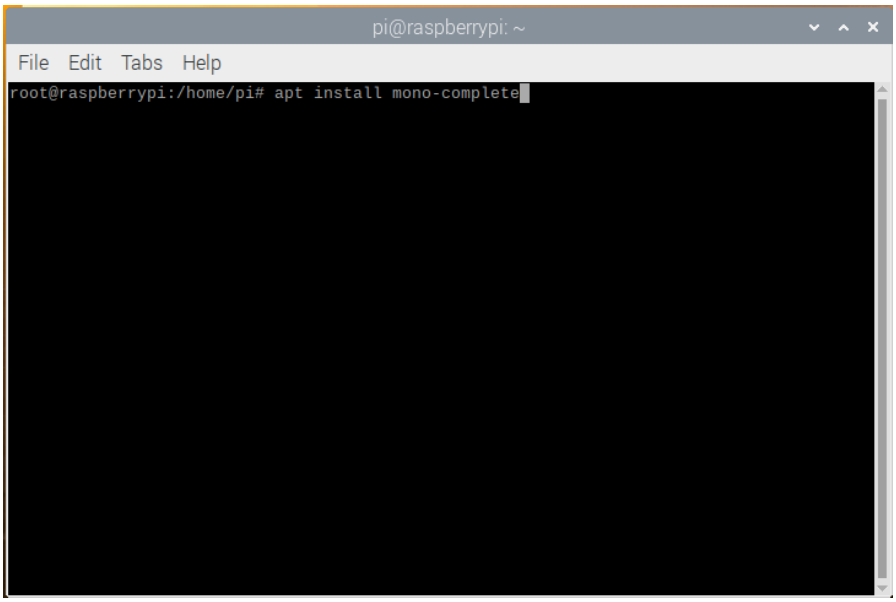
```
root@raspberrypi:/home/pi# sudo apt install apt-transport-https dirmngr gnupg ca-certificates
Reading package lists... Done
Building dependency tree
Reading state information... Done
ca-certificates is already the newest version (20190110).
dirmngr is already the newest version (2.2.12-1+rpi1+deb10u1).
gnupg is already the newest version (2.2.12-1+rpi1+deb10u1).
The following NEW packages will be installed:
  apt-transport-https
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 149 kB of archives.
After this operation, 156 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://quantum-mirror.hu/mirrors/pub/raspbian/raspbian buster/main armhf apt-transport-htt
ps all 1.8.2 [149 kB]
Fetched 149 kB in 1s (253 kB/s)
Selecting previously unselected package apt-transport-https.
(Reading database ... 93514 files and directories currently installed.)
Preparing to unpack .../apt-transport-https_1.8.2_all.deb ...
Unpacking apt-transport-https (1.8.2) ...
Setting up apt-transport-https (1.8.2) ...
root@raspberrypi:/home/pi# sudo apt-key adv --keyserver hkp://keyserver.ubuntu.com:80 --recv-key
s 3FA7E0328081BFF6A14DA29AA6A19B38D3D831EF
Executing: /tmp/apt-key-gpghome.GQk6RbSaZg/gpg.1.sh --keyserver hkp://keyserver.ubuntu.com:80 --
recv-keys 3FA7E0328081BFF6A14DA29AA6A19B38D3D831EF
gpg: key A6A19B38D3D831EF: public key "Xamarin Public Jenkins (auto-signing) <releng@xamarin.com
>" imported
gpg: Total number processed: 1
gpg:               imported: 1
root@raspberrypi:/home/pi# echo "deb https://download.mono-project.com/repo/debian stable-raspbi
anbuster main" | sudo tee /etc/apt/sources.list.d/mono-official-stable.list
deb https://download.mono-project.com/repo/debian stable-raspbianbuster main
root@raspberrypi:/home/pi# sudo apt update
```

Figure 6 - Add mono repository

# Step 5 - Install Mono

During the previous step, you added the official Mono repository to your system. Due to this, you will be able to install the latest Mono package and all its dependencies by issuing **apt install mono-complete** command as on **Figure 7**. This will install all of the components of Mono. The installation process may take a few minutes.

```
apt install mono-complete
```



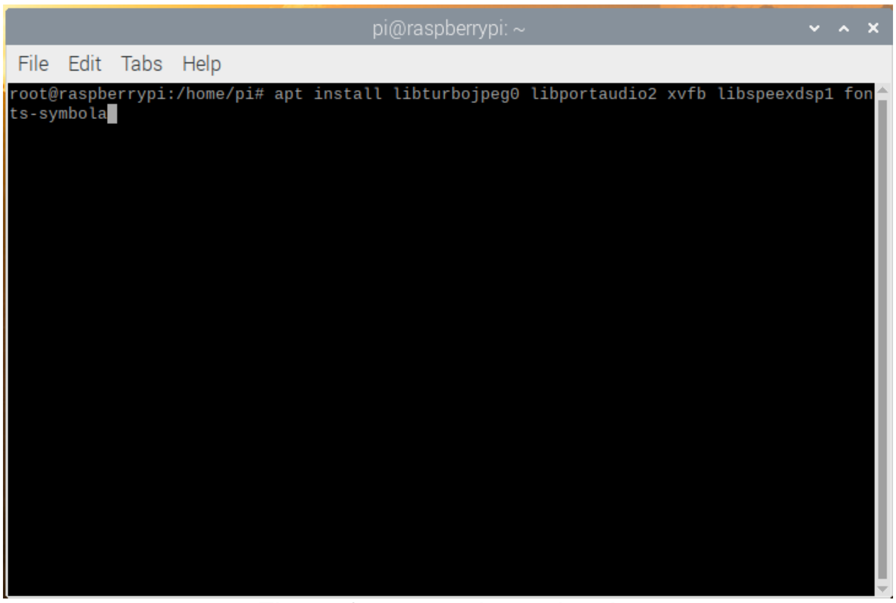
```
root@raspberrypi:/home/pi# apt install mono-complete
```

Figure 7 - Install mono

## Step 6 - Install other required dependencies

There are some other dependencies, that need to be installed for the clean installation of Ozeki Installer without any error. These dependencies are exactly the following packages: **libturbojpeg0**, **libportaudio2**, **xvfb**, **libspeexdsp1** and **fonts-symbola**. You can install them by issuing the command below. The **Figure 8** demonstrates this step.

```
apt install libturbojpeg0 libportaudio2 xvfb libspeexdsp1 fonts-symbola
```



```
root@raspberrypi:/home/pi# apt install libturbojpeg0 libportaudio2 xvfb libspeexdsp1 fon
ts-symbola
```

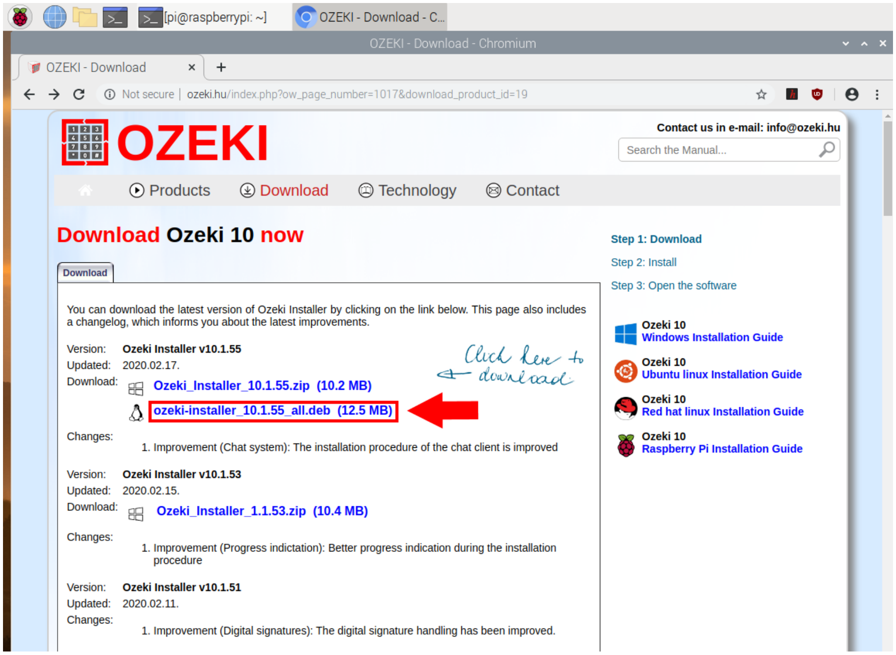
Figure 8 - Install dependencies

## Step 7 - Download the latest installer

The latest release of Ozeki Installer can be downloaded from http://www.ozeki.hu/index.php?owpn=1017&download_product_id=19. To download the latest package for Raspbian, you need to click on the **ozeki_Installer.x.x.x_all.deb** link as **Figure 9** shows below. We recommend you to always download the latest software package. This one always includes the current features and improvements.

You may download Ozeki Installer from the link below.
**http://www.ozeki.hu/index.php?owpn=1017&dpid=19**
**Download Ozeki Installer**

← Click here to download



[pi@raspberrypi: ~]
OZEKI - Download - C...
OZEKI - Download - Chromium
OZEKI - Download
Not secure | ozeki.hu/index.php?ow_page_number=1017&download_product_id=19
OZEKI
Contact us in e-mail: info@ozeki.hu
Search the Manual...
Products
Download
Technology
Contact

Download Ozeki 10 now

Download

You can download the latest version of Ozeki Installer by clicking on the link below. This page also includes a changelog, which informs you about the latest improvements.

Version: **Ozeki Installer v10.1.55**
Updated: 2020.02.17.
Download: Ozeki_Installer_10.1.55.zip (10.2 MB)
ozeki-installer_10.1.55_all.deb (12.5 MB)
Changes:
1. Improvement (Chat system): The installation procedure of the chat client is improved

Version: **Ozeki Installer v10.1.53**
Updated: 2020.02.15.
Download: Ozeki_Installer_1.1.53.zip (10.4 MB)
Changes:
1. Improvement (Progress indictation): Better progress indication during the installation procedure

Version: **Ozeki Installer v10.1.51**
Updated: 2020.02.11.
Changes:
1. Improvement (Digital signatures): The digital signature handling has been improved.

Click here to download

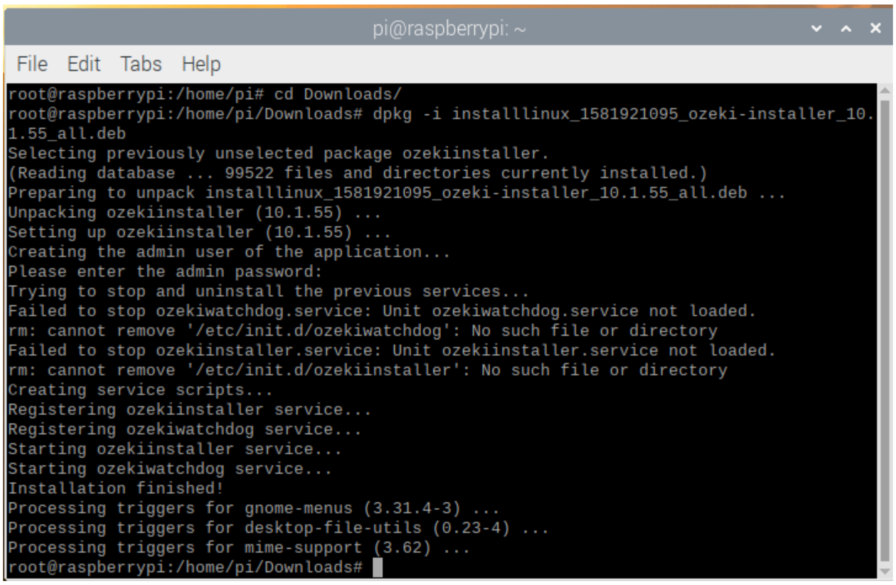
Step 1: Download
Step 2: Install
Step 3: Open the software
Ozeki 10
Windows Installation Guide
Ozeki 10
Ubuntu linux Installation Guide
Ozeki 10
Red hat linux Installation Guide
Ozeki 10
Raspberry Pi Installation Guide

Figure 9 - Download Ozeki Installer

# Step 8 - Install Ozeki Installer

During the previous step, you downloaded the Ozeki Installer package to the 'Downloads' folder of your system (by default). Please navigate here by issuing **cd Downloads/** command. To install Ozeki Installer, issue **dpkg -i package_name.db** (replace the package name with the name of the downloaded file) command. During the installation, the installer asks you to **provide a password** for the admin user. This will be the first user you can use to log in to Ozeki Installer. You can see a successful installation in **Figure 10**.

```
cd Downloads/
dpkg -i installlinux_xxxxxxxxxx_ozeki_xx.x.xx_all.deb
```



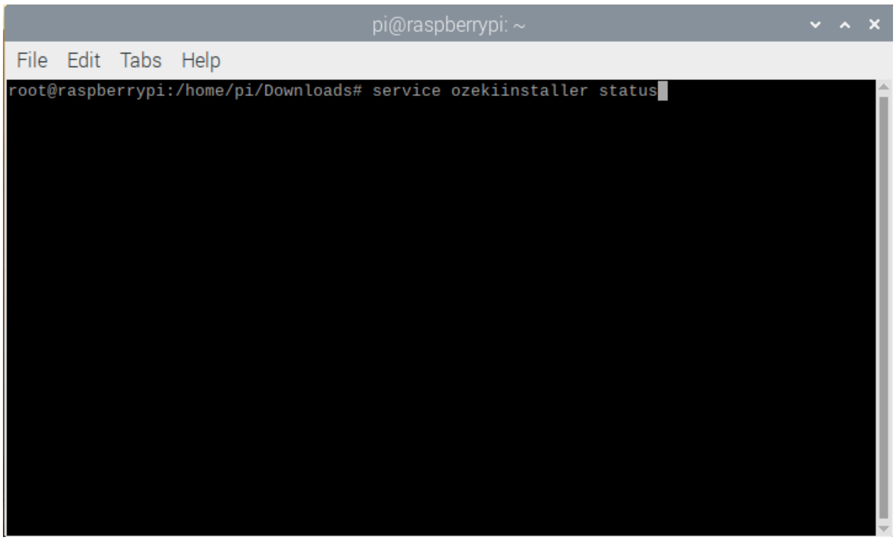
```
root@raspberrypi:/home/pi# cd Downloads/
root@raspberrypi:/home/pi/Downloads# dpkg -i installlinux_1581921095_ozeki-installer_10.
1.55_all.deb
Selecting previously unselected package ozekiinstaller.
(Reading database ... 99522 files and directories currently installed.)
Preparing to unpack installlinux_1581921095_ozeki-installer_10.1.55_all.deb ...
Unpacking ozekiinstaller (10.1.55) ...
Setting up ozekiinstaller (10.1.55) ...
Creating the admin user of the application...
Please enter the admin password:
Trying to stop and uninstall the previous services...
Failed to stop ozekiwatchdog.service: Unit ozekiwatchdog.service not loaded.
rm: cannot remove '/etc/init.d/ozekiwatchdog': No such file or directory
Failed to stop ozekiinstaller.service: Unit ozekiinstaller.service not loaded.
rm: cannot remove '/etc/init.d/ozekiinstaller': No such file or directory
Creating service scripts...
Registering ozekiinstaller service...
Registering ozekiwatchdog service...
Starting ozekiinstaller service...
Starting ozekiwatchdog service...
Installation finished!
Processing triggers for gnome-menus (3.31.4-3) ...
Processing triggers for desktop-file-utils (0.23-4) ...
Processing triggers for mime-support (3.62) ...
root@raspberrypi:/home/pi/Downloads# 
```

Figure 10 - Install Ozeki Installer

# Step 9 - Check Ozeki service

Services are programs that operate in the background. They can be automatically started when the computer boots, can be paused and restarted. Ozeki Installer runs in the background as a service. As you can see it on **Figure 11**, you can check the status of Ozeki Installer service by using the **service ozekiinstaller status** command.

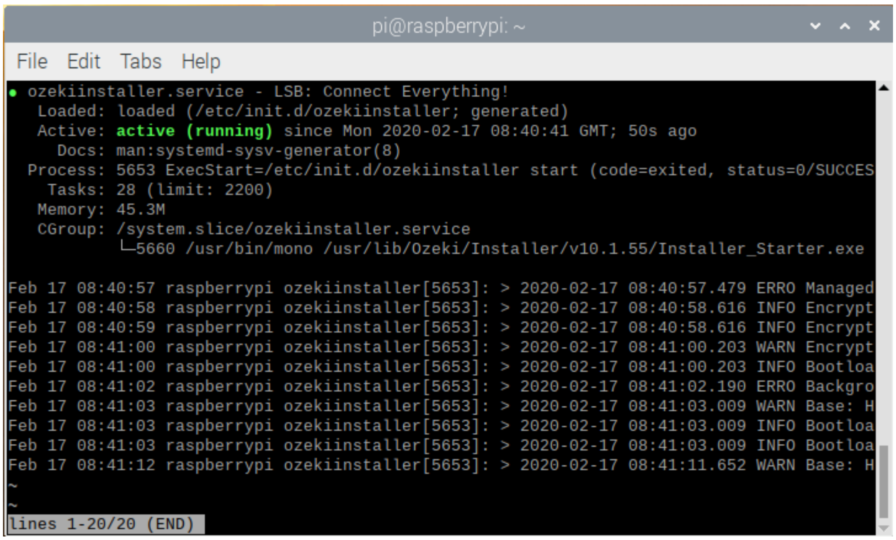
```
service ozekiinstaller status
```



```
root@raspberrypi:/home/pi/Downloads# service ozekiinstaller status
```

Figure 11 - Ozeki status

By issuing **service ozekiinstaller status** command, you can check its status. If it is active (**Figure 12**), it means that it is running in the background.

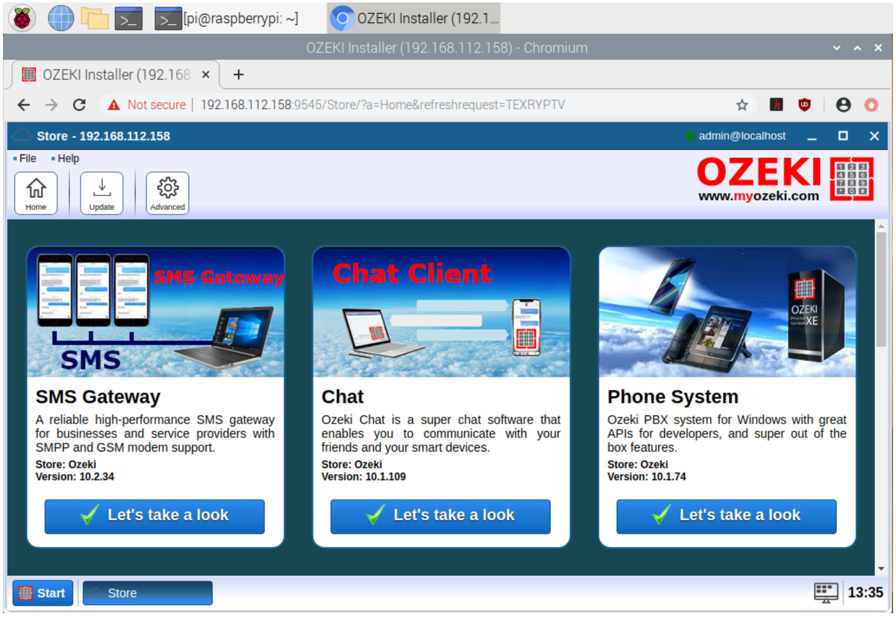


```
● ozekiinstaller.service - LSB: Connect Everything!
   Loaded: loaded (/etc/init.d/ozekiinstaller; generated)
   Active: active (running) since Mon 2020-02-17 08:40:41 GMT; 50s ago
     Docs: man:systemd-sysv-generator(8)
  Process: 5653 ExecStart=/etc/init.d/ozekiinstaller start (code=exited, status=0/SUCCES
    Tasks: 28 (limit: 2200)
   Memory: 45.3M
   CGroup: /system.slice/ozekiinstaller.service
           └─5660 /usr/bin/mono /usr/lib/Ozeki/Installer/v10.1.55/Installer_Starter.exe

Feb 17 08:40:57 raspberrypi ozekiinstaller[5653]: > 2020-02-17 08:40:57.479 ERRO Managed
Feb 17 08:40:58 raspberrypi ozekiinstaller[5653]: > 2020-02-17 08:40:58.616 INFO Encrypt
Feb 17 08:40:59 raspberrypi ozekiinstaller[5653]: > 2020-02-17 08:40:58.616 INFO Encrypt
Feb 17 08:41:00 raspberrypi ozekiinstaller[5653]: > 2020-02-17 08:41:00.203 WARN Encrypt
Feb 17 08:41:00 raspberrypi ozekiinstaller[5653]: > 2020-02-17 08:41:00.203 INFO Bootloa
Feb 17 08:41:02 raspberrypi ozekiinstaller[5653]: > 2020-02-17 08:41:02.190 ERRO Backgro
Feb 17 08:41:03 raspberrypi ozekiinstaller[5653]: > 2020-02-17 08:41:03.009 WARN Base: H
Feb 17 08:41:03 raspberrypi ozekiinstaller[5653]: > 2020-02-17 08:41:03.009 INFO Bootloa
Feb 17 08:41:03 raspberrypi ozekiinstaller[5653]: > 2020-02-17 08:41:03.009 INFO Bootloa
Feb 17 08:41:12 raspberrypi ozekiinstaller[5653]: > 2020-02-17 08:41:11.652 WARN Base: H
~
~
lines 1-20/20 (END)
```

Figure 12 - Ozeki service is running

# Step 10 - Check programs in Ozeki Installer

Ozeki Installer can be opened through your web browser. For that just open it, type localhost:9545 and hit Enter. On the main page, you will be able to see the Ozeki app store (**Figure 13**). On this page, you can choose from many Ozeki products such as SMS Gateway, Chat, Phone System etc.



**SMS Gateway**

A reliable high-performance SMS gateway for businesses and service providers with SMPP and GSM modem support.

**Store: Ozeki**
**Version: 10.2.34**

Let's take a look

**Chat**

Ozeki Chat is a super chat software that enables you to communicate with your friends and your smart devices.

**Store: Ozeki**
**Version: 10.1.109**

Let's take a look

**Phone System**

Ozeki PBX system for Windows with great APIs for developers, and super out of the box features.

**Store: Ozeki**
**Version: 10.1.74**

Let's take a look

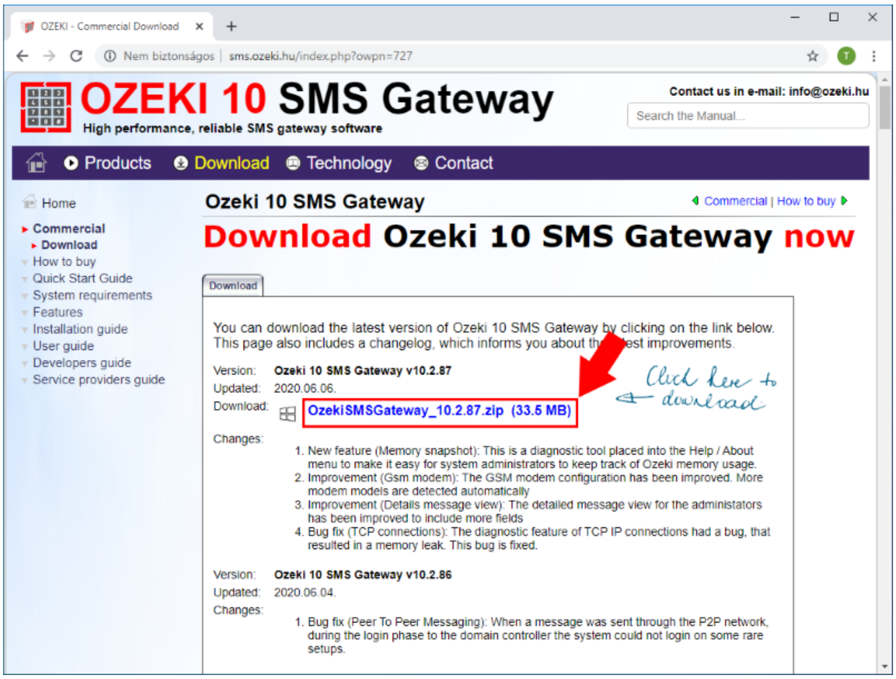
Figure 13 - Open Ozeki store

# Ozeki SMS Gateway Offline Installation

***This guide gives you the steps to do an Offline installation of Ozeki 10 SMS Gateway. This information is useful if you wish to install the software into an environment where you don't have Internet access, or your corporate firewall policy prevents software to communicate with the Internet. The offline installation is quite simple: you need to download the installation package and follow the instrations on the screen. It is similar to a classic Next...Next...Next... Windows installation procedure.***

## Step 1 - Download Ozeki SMS Gateway

The latest release of Ozeki SMS Gateway can be downloaded from the Ozeki 10 SMS Gateway download page. To download the installer in compressed format, you need to click on the first OzekiSMSGateway_x.x.xx.zip link as Figure 1 shows below. We recommend you to always download the latest version. This version always contains all the currently available features and improvements.



Ozeki 10 SMS Gateway

Download Ozeki 10 SMS Gateway now

You can download the latest version of Ozeki 10 SMS Gateway by clicking on the link below. This page also includes a changelog, which informs you about the latest improvements.

Version: **Ozeki 10 SMS Gateway v10.2.87**
Updated: 2020.06.06.
Download: OzekiSMSGateway_10.2.87.zip (33.5 MB)
Changes:

1. New feature (Memory snapshot): This is a diagnostic tool placed into the Help / About menu to make it easy for system administrators to keep track of Ozeki memory usage.
2. Improvement (Gsm modem): The GSM modem configuration has been improved. More modem models are detected automatically
3. Improvement (Details message view): The detailed message view for the administators has been improved to include more fields
4. Bug fix (TCP connections): The diagnostic feature of TCP IP connections had a bug, that resulted in a memory leak. This bug is fixed.

Version: **Ozeki 10 SMS Gateway v10.2.86**
Updated: 2020.06.04.
Changes:

1. Bug fix (Peer To Peer Messaging): When a message was sent through the P2P network, during the login phase to the domain controller the system could not login on some rare setups.

Figure 1 - Downloading Ozeki SMS Gateway

## Step 2 - Extract the downloaded file

When the download finished, you have to extract the compressed file to be able to run the installer of the software. For that, please open the Downloads folder on your computer. Then, as you can see it on Figure 2, right click on the file and select Extract All.... By doing this, you will get the executable installer of Ozeki Installer.

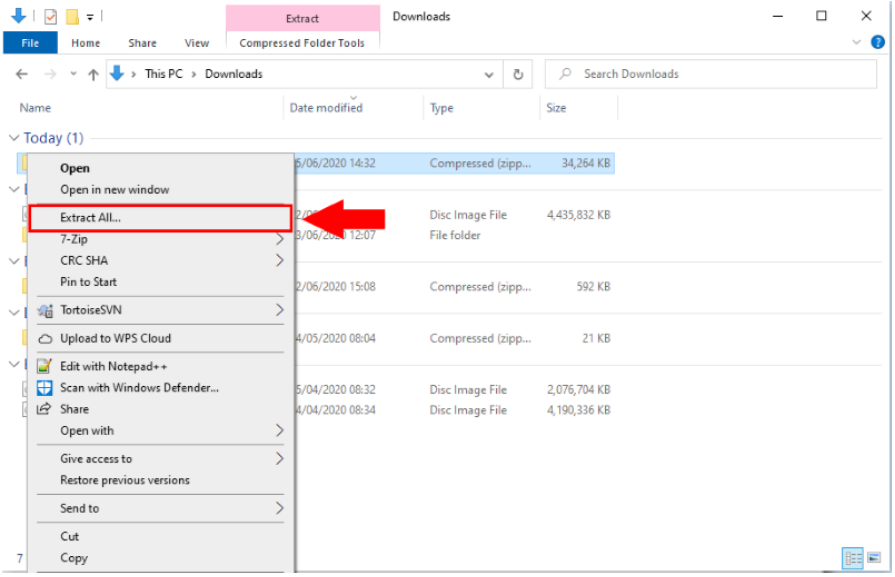

Extract
Downloads
File
Home
Share
View
Compressed Folder Tools
This PC › Downloads
Search Downloads
Name
Date modified
Type
Size
Today (1)
Open
Open in new window
Extract All...
7-Zip
CRC SHA
Pin to Start
TortoiseSVN
Upload to WPS Cloud
Edit with Notepad++
Scan with Windows Defender...
Share
Open with
Give access to
Restore previous versions
Send to
Cut
Copy
5/06/2020 14:32 Compressed (zipp... 34,264 KB
2/0... Disc Image File 4,435,832 KB
3/06/2020 12:07 File folder
2/06/2020 15:08 Compressed (zipp... 592 KB
4/05/2020 08:04 Compressed (zipp... 21 KB
5/04/2020 08:32 Disc Image File 2,076,704 KB
4/04/2020 08:34 Disc Image File 4,190,336 KB
7

Figure 2 - Extract installer

## Step 3 - Run the installer

After you extracted the .zip file, please open the newly created folder, and just like in Figure 3, just click on the executable file to start the installation process.

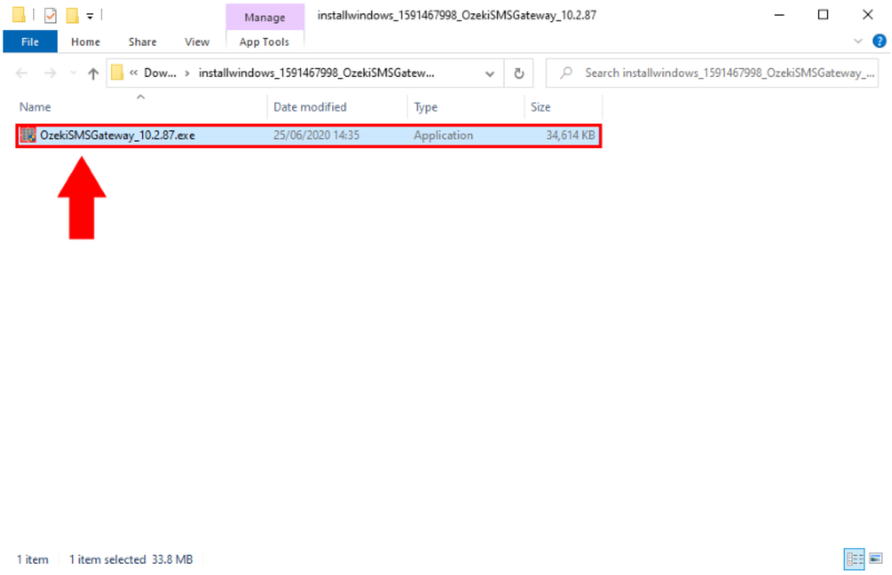
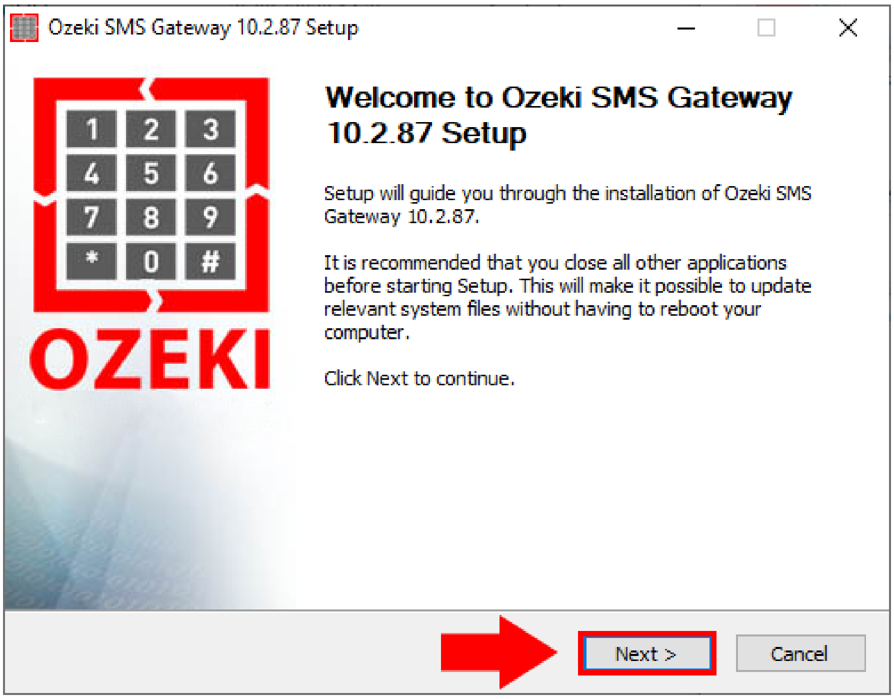

Manage
installwindows_1591467998_OzekiSMSGateway_10.2.87
File
Home
Share
View
App Tools
« Dow... › installwindows_1591467998_OzekiSMSGatew...
Search installwindows_1591467998_OzekiSMSGateway_...
Name
Date modified
Type
Size
OzekiSMSGateway_10.2.87.exe 25/06/2020 14:35 Application 34,614 KB
1 item
1 item selected 33.8 MB

Figure 3 - Start installation

## Step 4 - Install Ozeki SMS Gateway

As soon as you clicked on the .exe file, the installer starts with a welcome screen (Figure 4). All you have to do here is to click Next to continue the installation.



Ozeki SMS Gateway 10.2.87 Setup

**Welcome to Ozeki SMS Gateway 10.2.87 Setup**

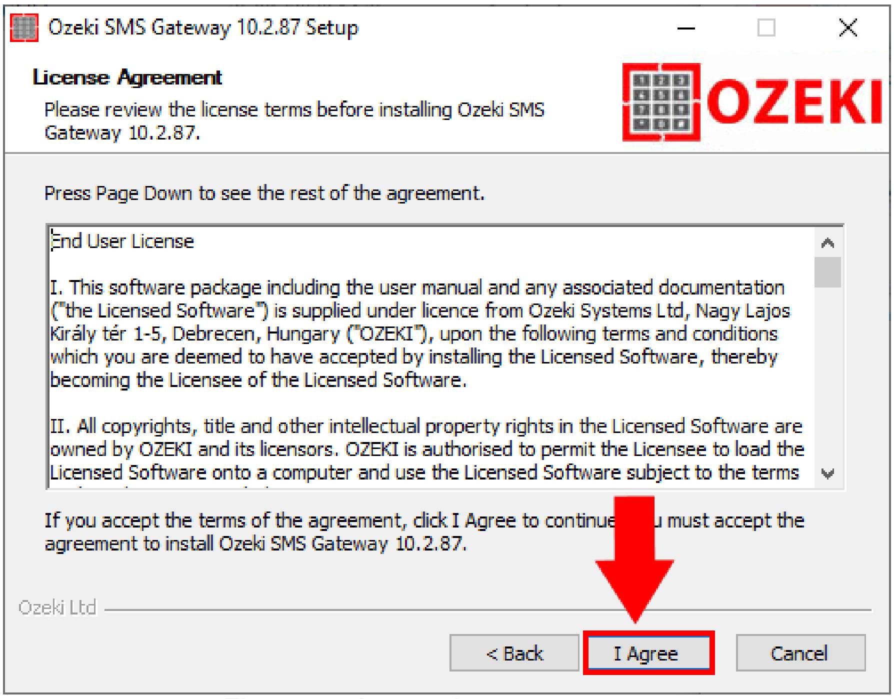
Setup will guide you through the installation of Ozeki SMS Gateway 10.2.87.

It is recommended that you close all other applications before starting Setup. This will make it possible to update relevant system files without having to reboot your computer.

Click Next to continue.

Figure 4 - Welcome to SMS Gateway

To install Ozeki SMS Gateway on your computer, you must accept the terms of the License Agreement. Please review them then as Figure 5 demonstrates, click 'I Agree' button to continue.



Ozeki SMS Gateway 10.2.87 Setup

**License Agreement**

Please review the license terms before installing Ozeki SMS Gateway 10.2.87.

Press Page Down to see the rest of the agreement.

End User License

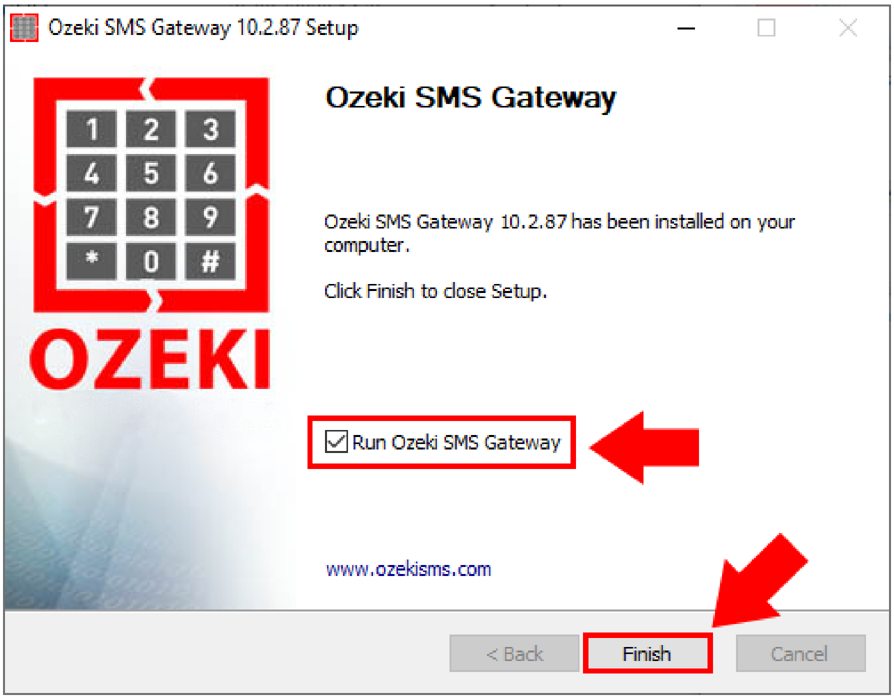
I. This software package including the user manual and any associated documentation ("the Licensed Software") is supplied under licence from Ozeki Systems Ltd, Nagy Lajos Király tér 1-5, Debrecen, Hungary ("OZEKI"), upon the following terms and conditions which you are deemed to have accepted by installing the Licensed Software, thereby becoming the Licensee of the Licensed Software.

II. All copyrights, title and other intellectual property rights in the Licensed Software are owned by OZEKI and its licensors. OZEKI is authorised to permit the Licensee to load the Licensed Software onto a computer and use the Licensed Software subject to the terms

If you accept the terms of the agreement, click I Agree to continue. You must accept the agreement to install Ozeki SMS Gateway 10.2.87.

Figure 5 - Accept license agreement

After the installation process is finished the Ozeki SMS Gateway service will start automatically in the background. To run Ozeki SMS Gateway on your web browser, just check Run Ozeki SMS Gateway as you can see it on Figure 6 and click Finish.


Ozeki SMS Gateway 10.2.87 Setup

Ozeki SMS Gateway

Ozeki SMS Gateway 10.2.87 has been installed on your computer.

Click Finish to close Setup.

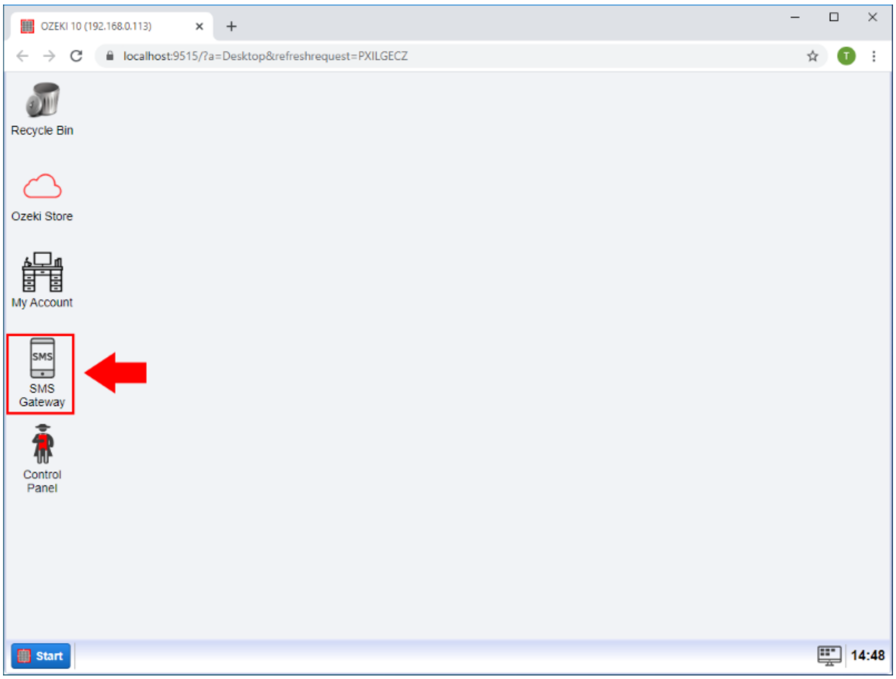
☑ Run Ozeki SMS Gateway

www.ozekisms.com

Figure 6 - Installation successful

## Step 5 - Ozeki SMS Gateway desktop

The home page of Ozeki SMS Gateway is a desktop, where you can find the SMS Gateway application's icon as Figure 7 shows it. Click on the Icon to open the application.


OZEKI 10 (192.168.0.113)

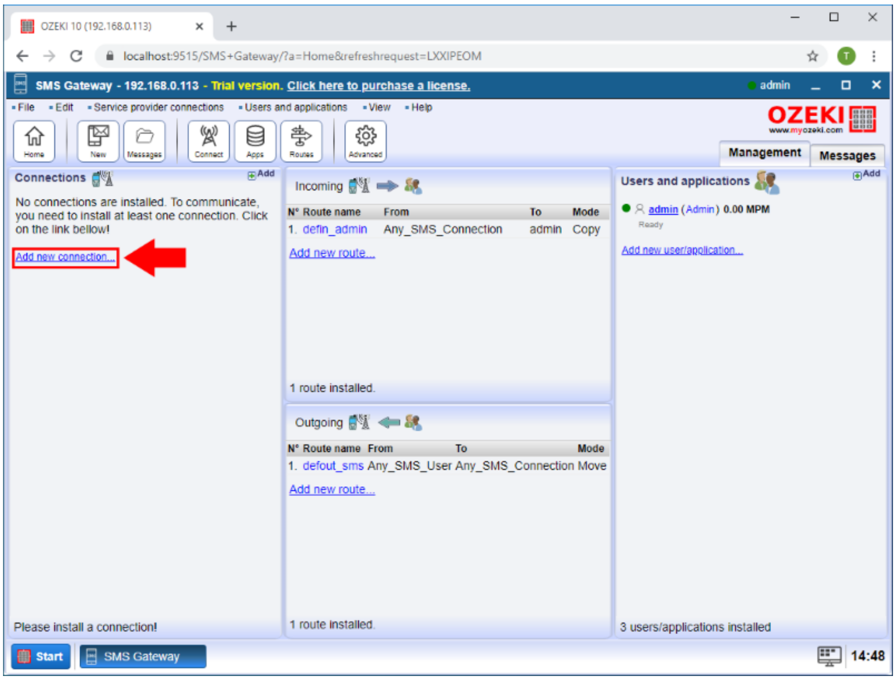
localhost:9515/?a=Desktop&refreshrequest=PXILGECZ

Figure 7 - Ozeki SMS Gateway desktop

## Step 6 - Connect to the mobile network

SMS messages are sent through the mobile network, so the next step in the procedure, is to select a method to connect your computer to the mobile network. There are various options to setup this connection. You can connect using a GSM modem, for higher capacity you can connect an SMS modem pool. You can also connect through an Androi mobile phone. If you operate an Internet SMS service, you would likele want to connect directly to the Short Message Service Center (SMSC) of a mobile network operator over the Internet. In this case you would setup an SMPP connection, an UCP connection a CIMD2 connection or an HTTP SMS connection. The most popular choice is to setup an SMPP connection over the Internet.

To install and configure an SMPP connection, please click on the Add new connection in the SMS Gateway's Home page (Figure 8).



Connections

No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

Please install a connection!

Incoming

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defin_admin | Any_SMS_Connection | admin | Copy |

Add new route...

1 route installed.

Outgoing

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move |

Add new route...

1 route installed.

Users and applications

admin (Admin) 0.00 MPM
Ready

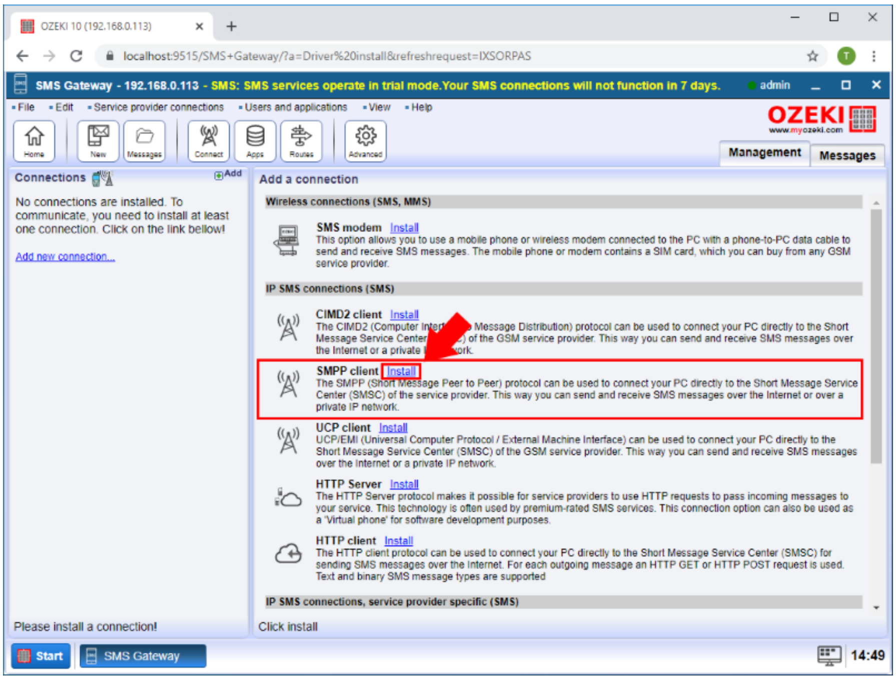
Add new user/application...

3 users/applications installed

Figure 8 - Add new connection

# Step 7 - Install SMPP Client

Then click the Install button next to the SMPP client in the list. (Figure 9).



OZEKI 10 (192.168.0.113)
localhost:9515/SMS+Gateway/?a=Driver%20install&refreshrequest=IXSORPAS
SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.Your SMS connections will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
OZEKI
www.myozeki.com
Management
Messages
Start
SMS Gateway
14:49

Connections

No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

Please install a connection!

Add a connection

Wireless connections (SMS, MMS)

SMS modem Install
This option allows you to use a mobile phone or wireless modem connected to the PC with a phone-to-PC data cable to send and receive SMS messages. The mobile phone or modem contains a SIM card, which you can buy from any GSM service provider.

IP SMS connections (SMS)

CIMD2 client Install
The CIMD2 (Computer Inter[illegible] Message Distribution) protocol can be used to connect your PC directly to the Short Message Service Center[illegible]) of the GSM service provider. This way you can send and receive SMS messages over the Internet or a private [illegible]ork.

SMPP client Install
The SMPP (Short Message Peer to Peer) protocol can be used to connect your PC directly to the Short Message Service Center (SMSC) of the service provider. This way you can send and receive SMS messages over the Internet or over a private IP network.

UCP client Install
UCP/EMI (Universal Computer Protocol / External Machine Interface) can be used to connect your PC directly to the Short Message Service Center (SMSC) of the GSM service provider. This way you can send and receive SMS messages over the Internet or a private IP network.

HTTP Server Install
The HTTP Server protocol makes it possible for service providers to use HTTP requests to pass incoming messages to your service. This technology is often used by premium-rated SMS services. This connection option can also be used as a 'Virtual phone' for software development purposes.

HTTP client Install
The HTTP client protocol can be used to connect your PC directly to the Short Message Service Center (SMSC) for sending SMS messages over the Internet. For each outgoing message an HTTP GET or HTTP POST request is used. Text and binary SMS message types are supported

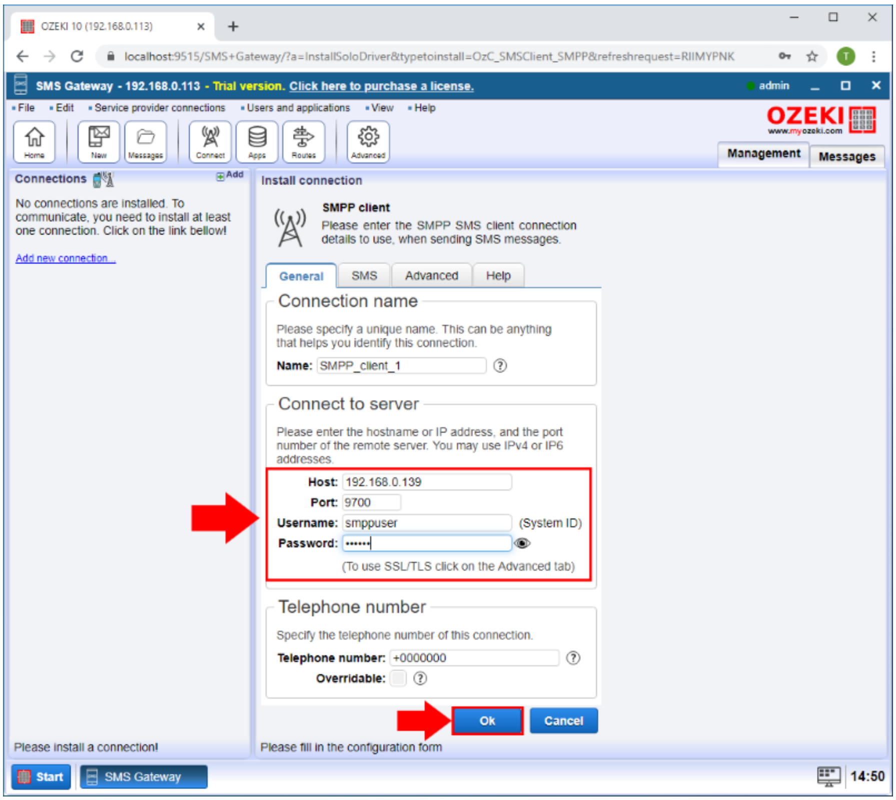
IP SMS connections, service provider specific (SMS)

Click install

Figure 9 - Install SMPP Client

# Step 8 - Provide SMPP client informations

After it please enter the host name, the port number, a username and a password. This information is provided by the service provider whose service you have subscribed to (Figure 10).


OZEKI 10 (192.168.0.113)
localhost:9515/SMS+Gateway/?a=InstallSoloDriver&typetoinstall=OzC_SMSClient_SMPP&refreshrequest=RIIMYPNK
SMS Gateway - 192.168.0.113 - Trial version. Click here to purchase a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
OZEKI
www.myozeki.com
Management
Messages
Connections
Add
General
SMS
Advanced
Help
Ok
Cancel
Start
SMS Gateway
14:50

No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

Install connection

SMPP client

Please enter the SMPP SMS client connection details to use, when sending SMS messages.

Connection name

Please specify a unique name. This can be anything that helps you identify this connection.

Name: SMPP_client_1

Connect to server

Please enter the hostname or IP address, and the port number of the remote server. You may use IPv4 or IP6 addresses.

Host: 192.168.0.139

Port: 9700

Username: smppuser (System ID)

Password: ••••••

(To use SSL/TLS click on the Advanced tab)

Telephone number

Specify the telephone number of this connection.

Telephone number: +0000000

Overridable: □

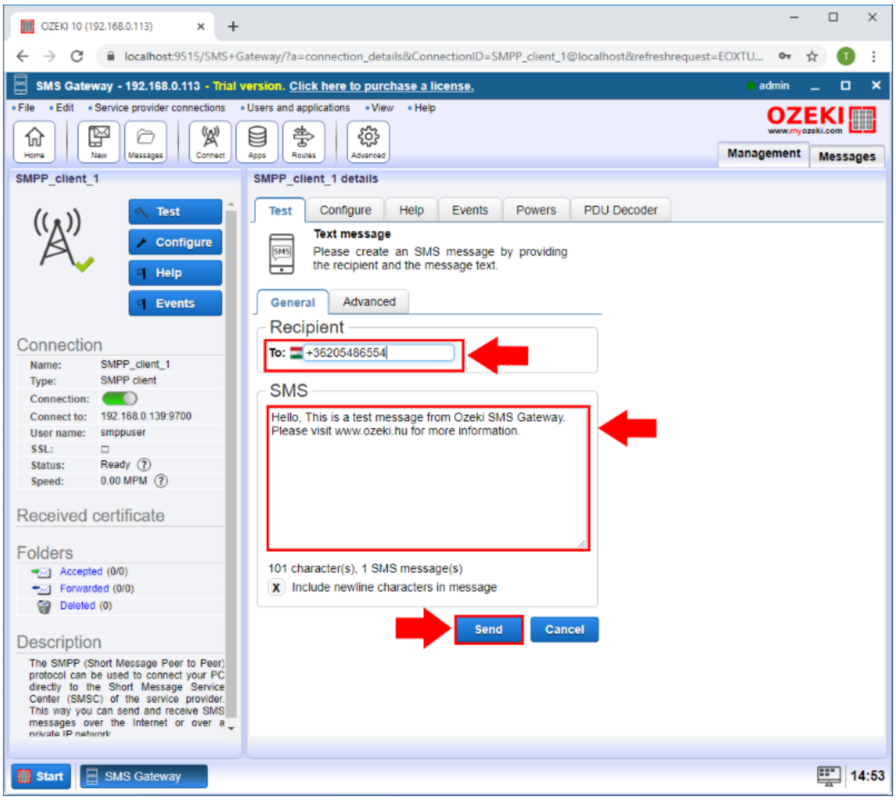
Please install a connection!

Please fill in the configuration form

Figure 10 - Provide SMPP client informations

## Step 9 - Send test message

In the Test tab you can send a test SMS message. Please provide the phone number and the message text, then click on the Send button (Figure 11).



SMPP_client_1

SMPP_client_1 details

Text message

Please create an SMS message by providing the recipient and the message text.

Recipient

To: +36205486554

SMS

Hello, This is a test message from Ozeki SMS Gateway. Please visit www.ozeki.hu for more information.

101 character(s), 1 SMS message(s)

☒ Include newline characters in message

Connection

Name: SMPP_client_1

Type: SMPP client

Connection:

Connect to: 192.168.0.139:9700

User name: smppuser

SSL: □

Status: Ready

Speed: 0.00 MPM

Received certificate

Folders

Accepted (0/0)

Forwarded (0/0)

Deleted (0)

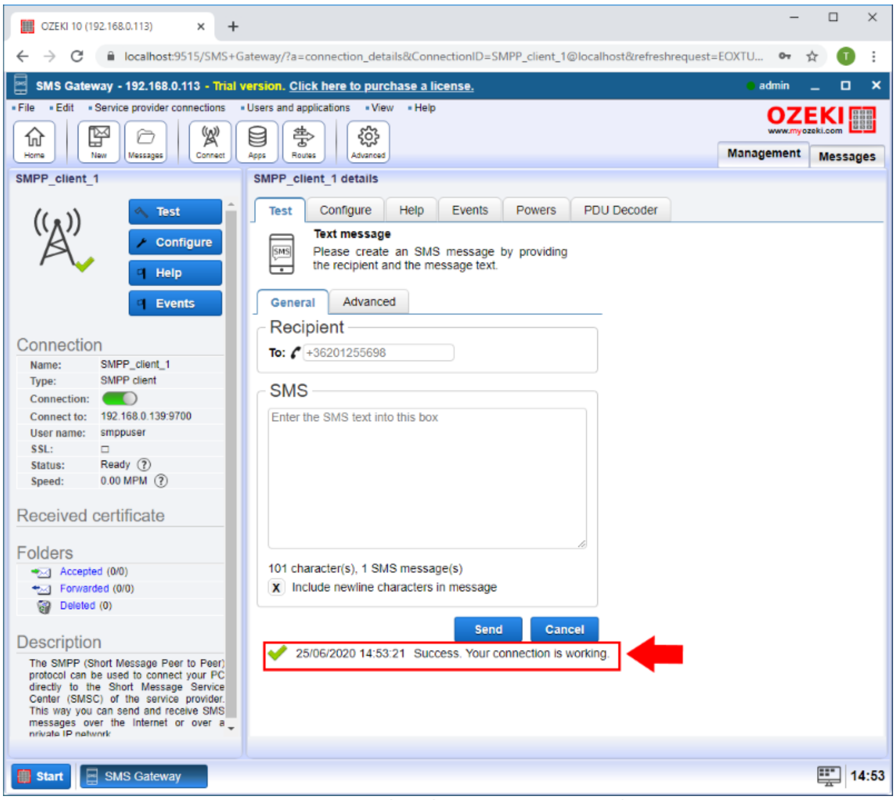
Description

The SMPP (Short Message Peer to Peer) protocol can be used to connect your PC directly to the Short Message Service Center (SMSC) of the service provider. This way you can send and receive SMS messages over the Internet or over a private IP network

Figure 11 - Send test message

## Step 10 - SMS sent successfully

You can see that the SMS was sent successfully (Figure 12).



Connection

Name: SMPP_client_1
Type: SMPP client
Connection:
Connect to: 192.168.0.139:9700
User name: smppuser
SSL: □
Status: Ready
Speed: 0.00 MPM

Received certificate

Folders

Accepted (0/0)
Forwarded (0/0)
Deleted (0)

Description

The SMPP (Short Message Peer to Peer) protocol can be used to connect your PC directly to the Short Message Service Center (SMSC) of the service provider. This way you can send and receive SMS messages over the Internet or over a private IP network

Text message

Please create an SMS message by providing the recipient and the message text.

Recipient

To: +36201255698

SMS

Enter the SMS text into this box

101 character(s), 1 SMS message(s)

☒ Include newline characters in message

✓ 25/06/2020 14:53:21 Success. Your connection is working.

Figure 12 - SMS sent successfully

# Product Activation

***This page explains how you can activate your pruchased license. To be able to use Ozeki with full funcationality, you need to purchase a license, and you need to and activate it. During the purchase procedure you will receive a serial number from Ozeki in e-mail. Without activation, you can only use the software in trial mode for 7 days. (The trial mode does not require activation.)***

To start the activation procedure, click on the "**My Account**" icon

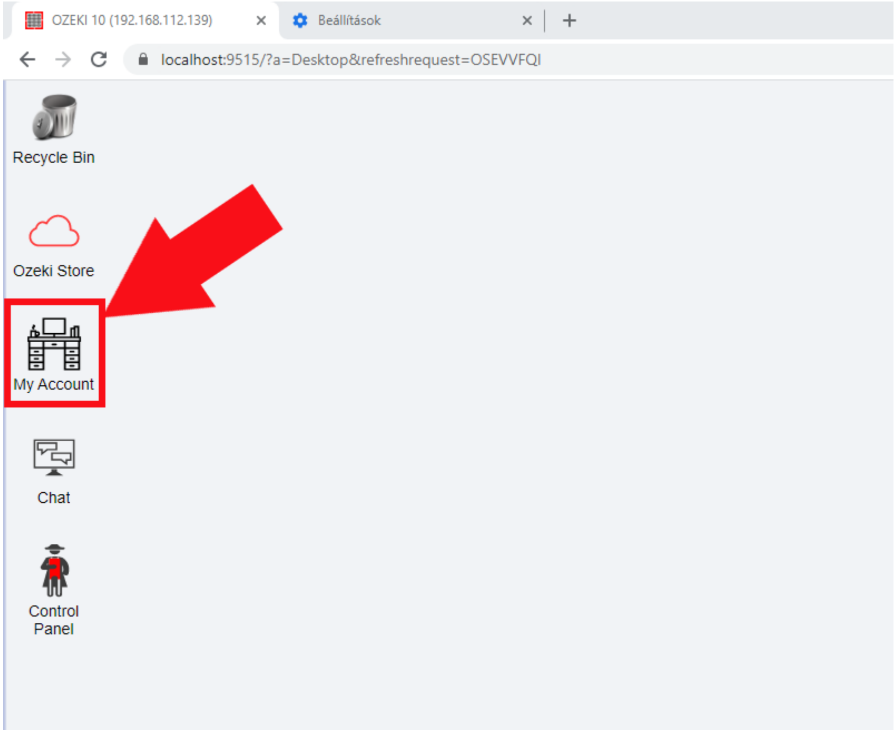

OZEKI 10 (192.168.112.139)
Beállítások
localhost:9515/?a=Desktop&refreshrequest=OSEVVFQI
Recycle Bin
Ozeki Store
My Account
Chat
Control Panel

Figure 1 - My account

Then click the green" **Add license code**" button

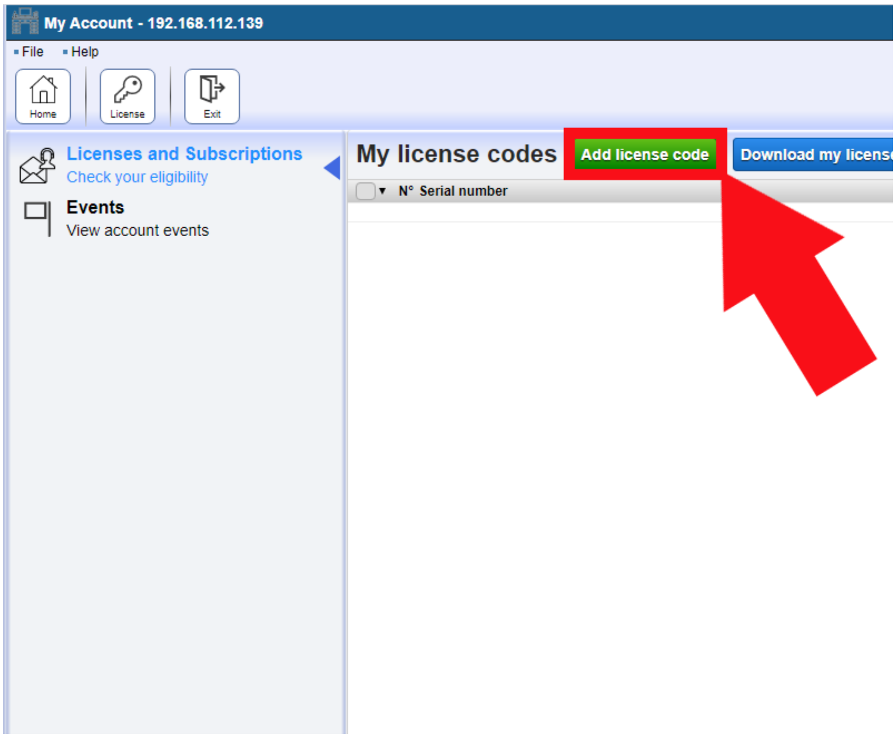
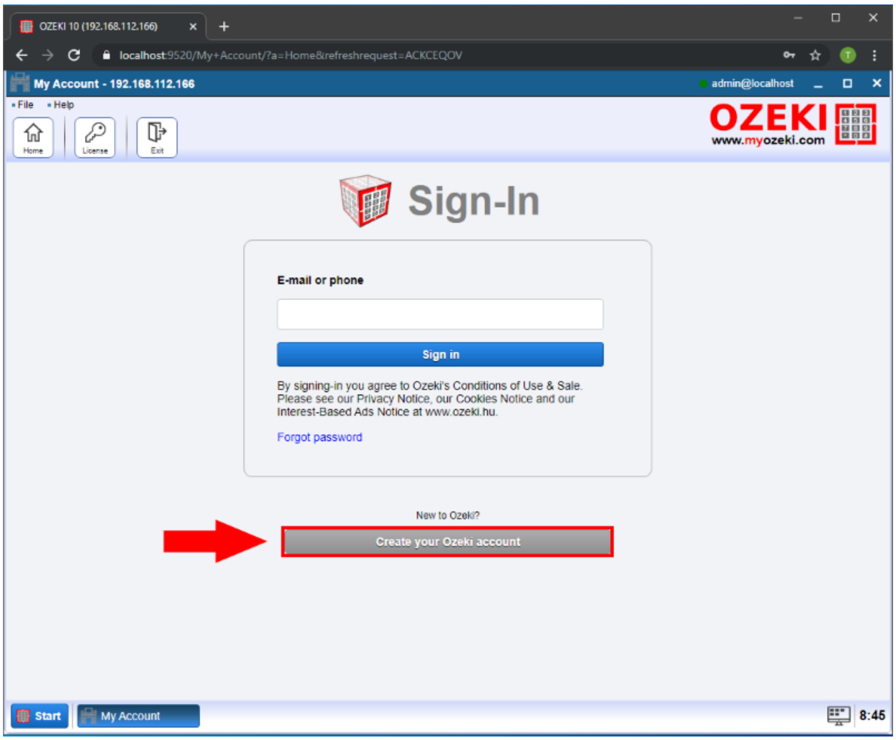


Figure 2 - Add license code

If you do not have an Ozeki account yet please click on "**Create your Ozeki account**" button.


OZEKI 10 (192.168.112.166)
localhost:9520/My+Account/?a=Home&refreshrequest=ACKCEQOV
My Account - 192.168.112.166
admin@localhost
File
Help
Home
License
Exit
OZEKI
www.myozeki.com
Start
My Account
8:45

Sign-In

E-mail or phone

Sign in

By signing-in you agree to Ozeki's Conditions of Use & Sale. Please see our Privacy Notice, our Cookies Notice and our Interest-Based Ads Notice at www.ozeki.hu.

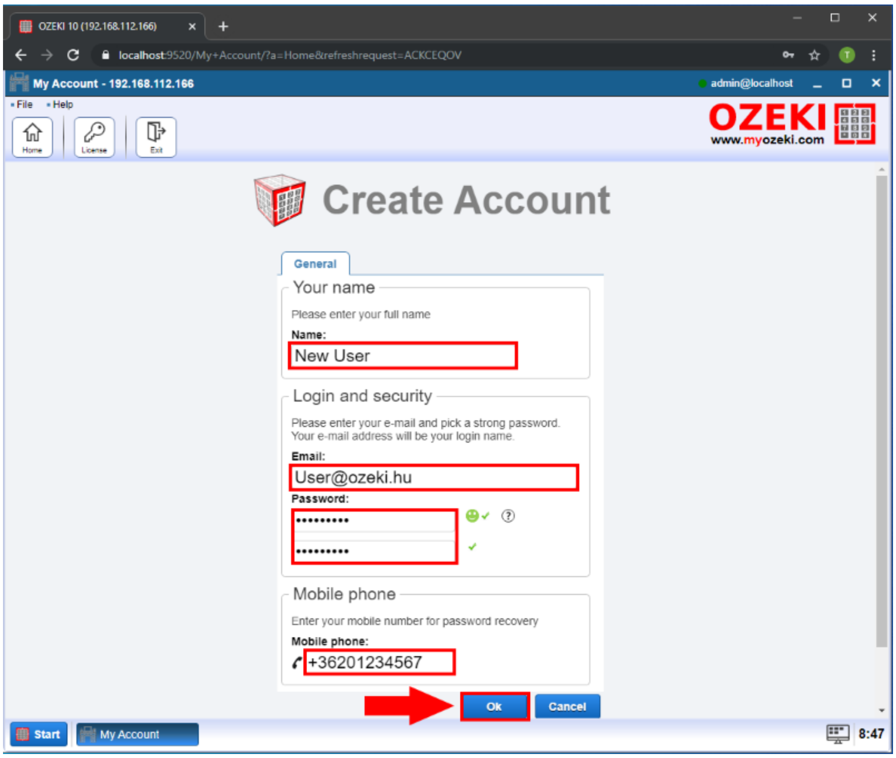
Forgot password

New to Ozeki?

Create your Ozeki account

Figure 3 - Create new Account

Please enter your Name, Email address, Password and Mobile number, then click on "**OK**".


OZEKI 10 (192.168.112.166)
localhost:9520/My+Account/?a=Home&refreshrequest=ACKCEQOV
My Account - 192.168.112.166
admin@localhost
File
Help
Home
License
Exit
OZEKI
www.myozeki.com
General
Start
My Account
8:47

Create Account

Your name

Please enter your full name

Name: New User

Login and security

Please enter your e-mail and pick a strong password. Your e-mail address will be your login name.

Email: User@ozeki.hu

Password: •••••••••

•••••••••

Mobile phone

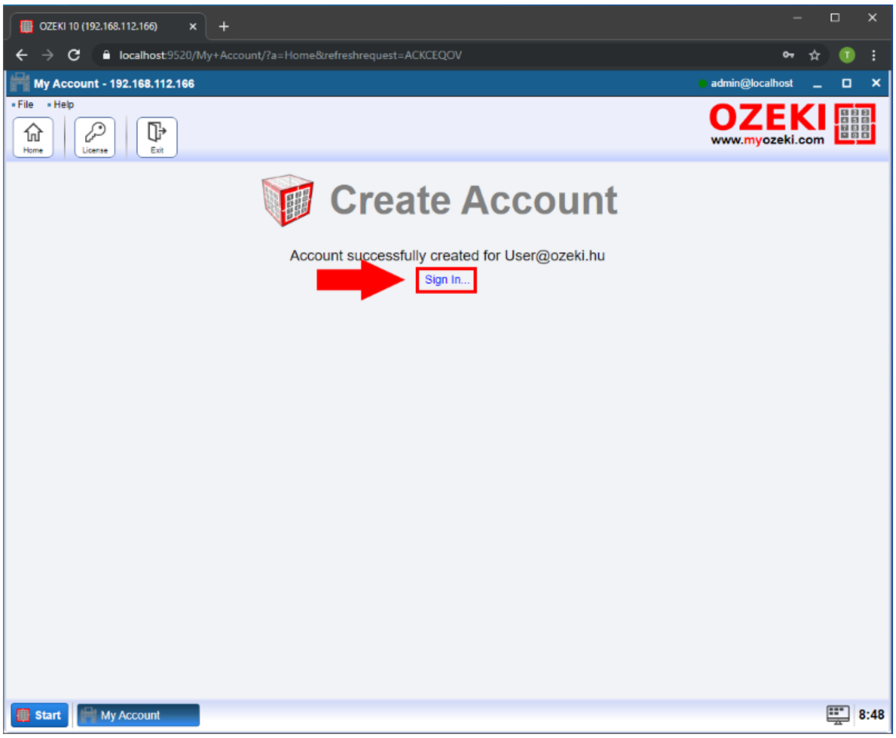
Enter your mobile number for password recovery

Mobile phone: +36201234567

Ok Cancel

Figure 4 - New User

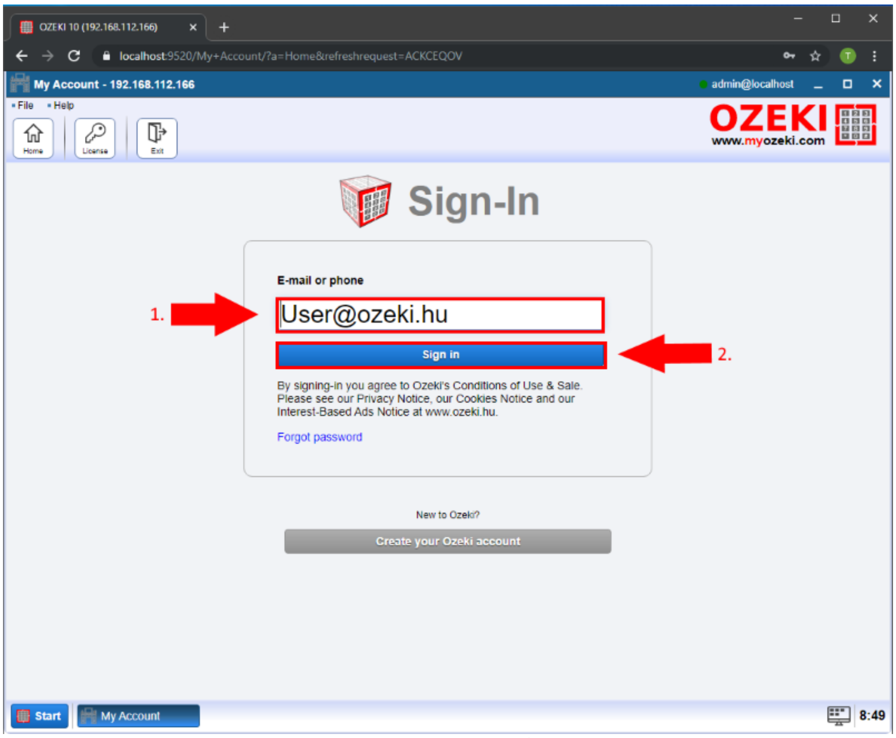
New Ozeki Account cerated.


OZEKI 10 (192.168.112.166)
localhost:9520/My+Account/?a=Home&refreshrequest=ACKCEQOV
My Account - 192.168.112.166
admin@localhost
File
Help
Home
License
Exit
OZEKI
www.myozeki.com
Create Account
Sign In...
Start
My Account
8:48

Account successfully created for User@ozeki.hu

Figure 5 - Account Created

After you click on "**Sign in**", You can login with your Ozeki account.


OZEKI 10 (192.168.112.166)
localhost:9520/My+Account/?a=Home&refreshrequest=ACKCEQOV
My Account - 192.168.112.166
admin@localhost
File
Help
Home
License
Exit
OZEKI
www.myozeki.com
Sign-In
1.
2.
Sign in
Forgot password
Create your Ozeki account
Start
My Account
8:49

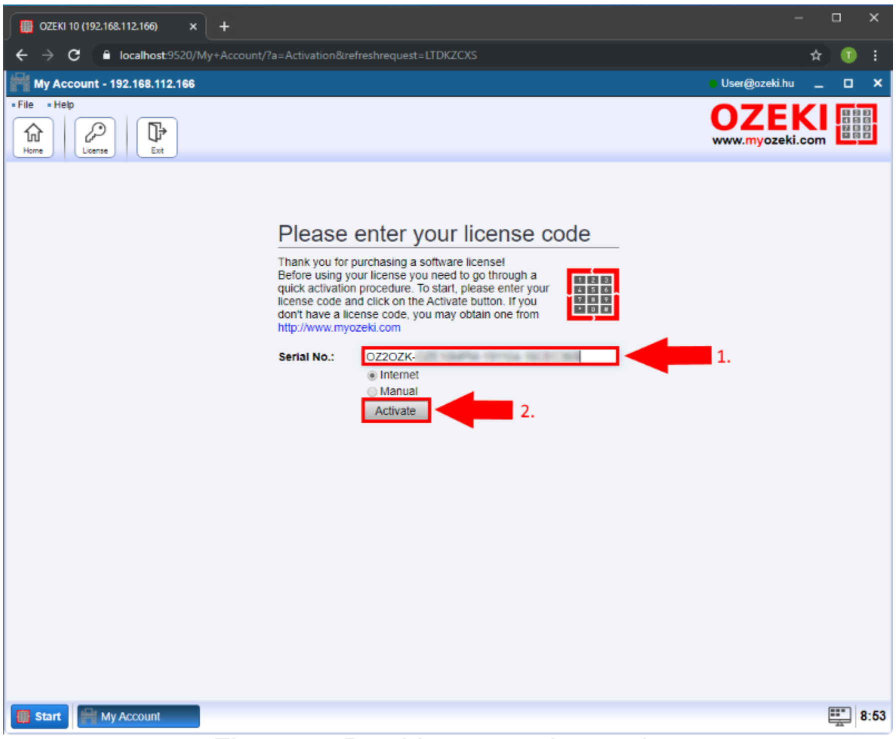
E-mail or phone: User@ozeki.hu

By signing-in you agree to Ozeki's Conditions of Use & Sale. Please see our Privacy Notice, our Cookies Notice and our Interest-Based Ads Notice at www.ozeki.hu.

New to Ozeki?

Figure 6 - Login with your Ozeki account.

On the next page you need to provide your serial number into the Serial No. textbox then click on "**Activate**" button.



OZEKI 10 (192.168.112.166)
localhost:9520/My+Account/?a=Activation&refreshrequest=LTDKZCXS
My Account - 192.168.112.166
User@ozeki.hu
File
Help
Home
License
Exit
OZEKI
www.myozeki.com
1.
2.
Activate
Start
My Account
8:53

Please enter your license code

Thank you for purchasing a software license!
Before using your license you need to go through a quick activation procedure. To start, please enter your license code and click on the Activate button. If you don't have a license code, you may obtain one from http://www.myozeki.com

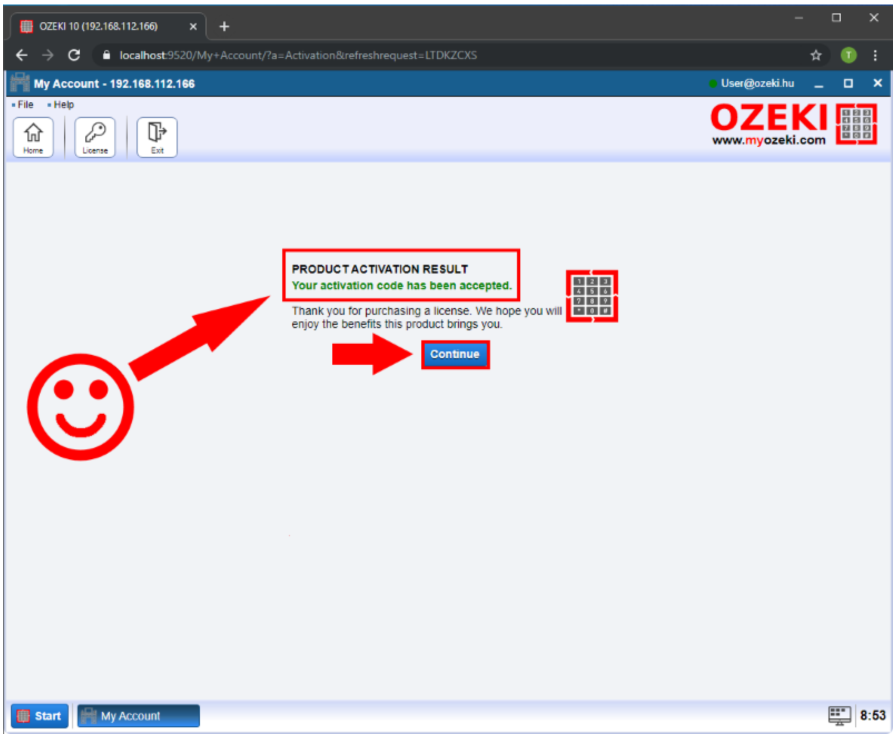
Serial No.: OZ2OZK-[redacted]

● Internet

○ Manual

Figure 7 - Provide your serial number

If you have provided your serial number correctly you will see that your activation code has accepted and you can use the full version of the product without limitations.



OZEKI 10 (192.168.112.166)
localhost:9520/My+Account/?a=Activation&refreshrequest=LTDKZCXS
My Account - 192.168.112.166
User@ozeki.hu
File
Help
Home
License
Exit
OZEKI
www.myozeki.com
Continue
Start
My Account
8:53

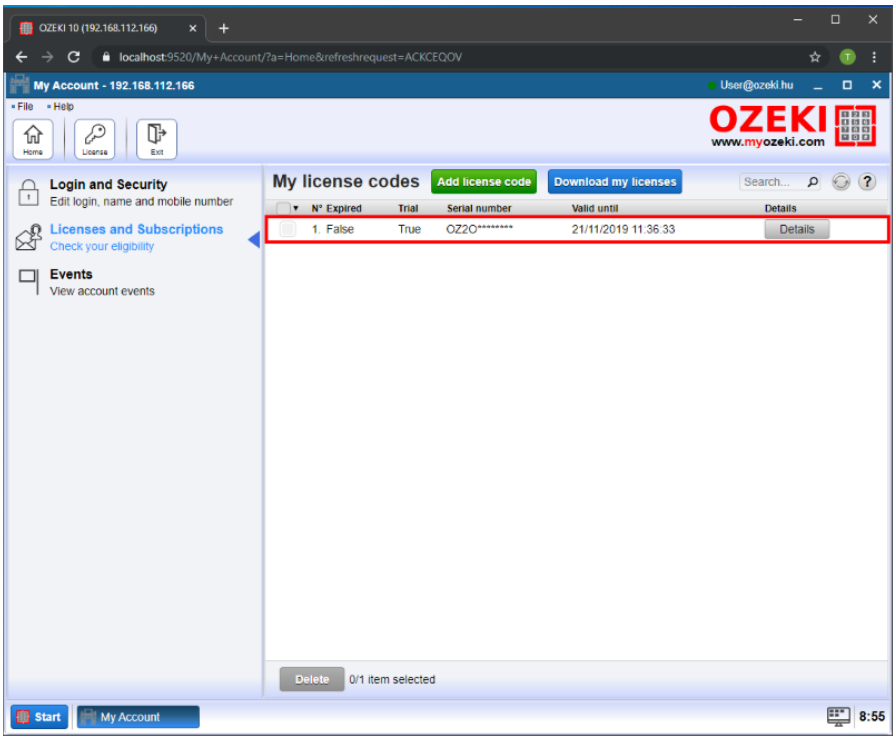
PRODUCT ACTIVATION RESULT

Your activation code has been accepted.

Thank you for purchasing a license. We hope you will enjoy the benefits this product brings you.

Figure 8 - Activation code accepted

In the Home page you can see your License codes.



**Login and Security**
Edit login, name and mobile number

**Licenses and Subscriptions**
Check your eligibility

**Events**
View account events

**My license codes**

| N° | Expired | Trial | Serial number | Valid until | Details |
|---|---|---|---|---|---|
| 1. | False | True | OZ2O******** | 21/11/2019 11:36:33 | Details |

0/1 item selected

Figure 9 - My License codes

# How to change the admin password

***The documentation on this page is about to give you a brief description of how easily you can change the password of your Ozeki account. This operation is really crucial to keep your account safe from unwanted actions. Changing your password takes just about 2 minutes and does not require any further technological knowledge. You just need to follow the simple instructions on this page. So, let's get started!***

The changing of your password can be easily done in its dedicated menu. You can reach this 'Change Password' form simply from the desktop. As you can see it in Figure 1, just click on the Start menu, here select the 'Help' menu, and finally, just click on the 'Change password' option.

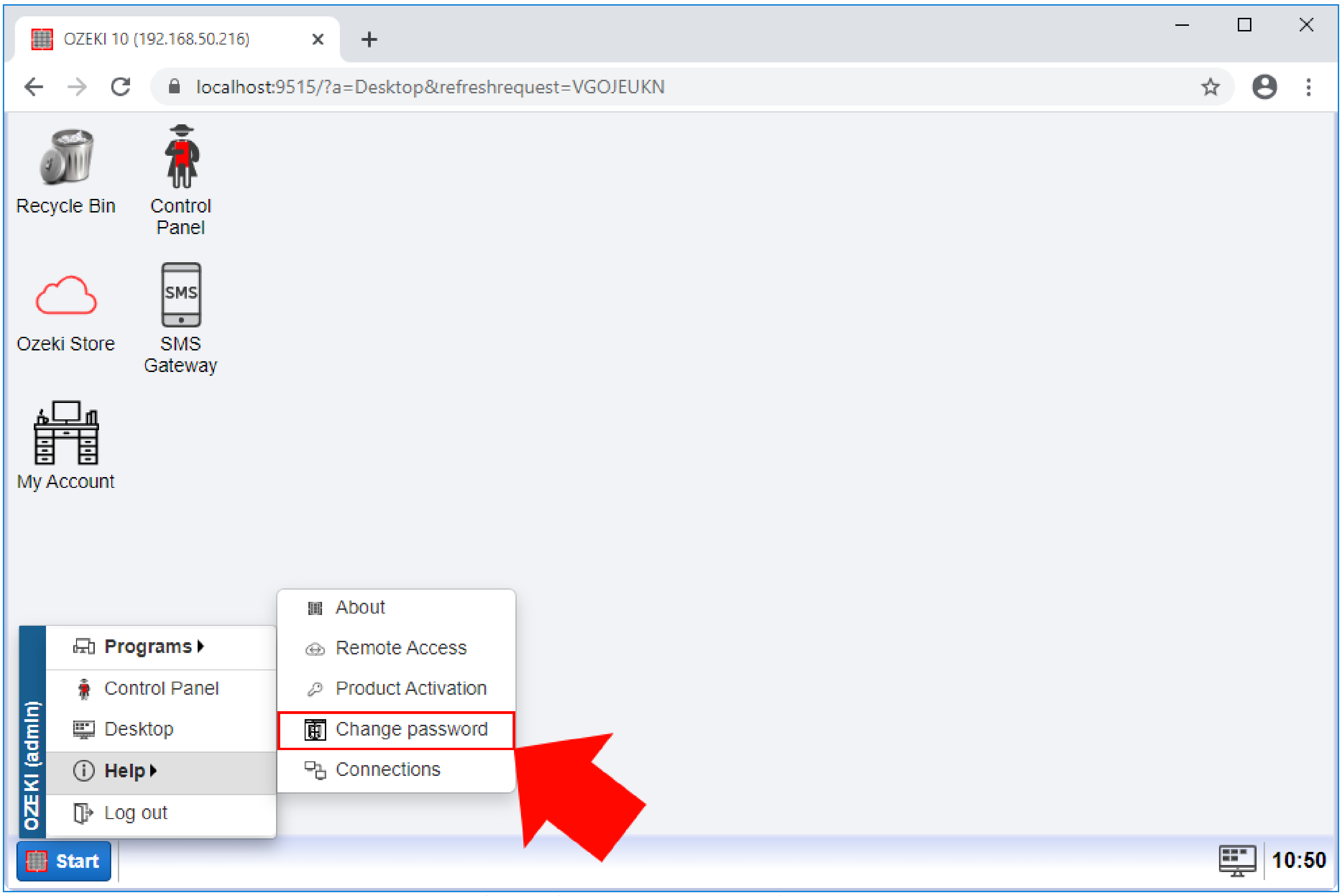
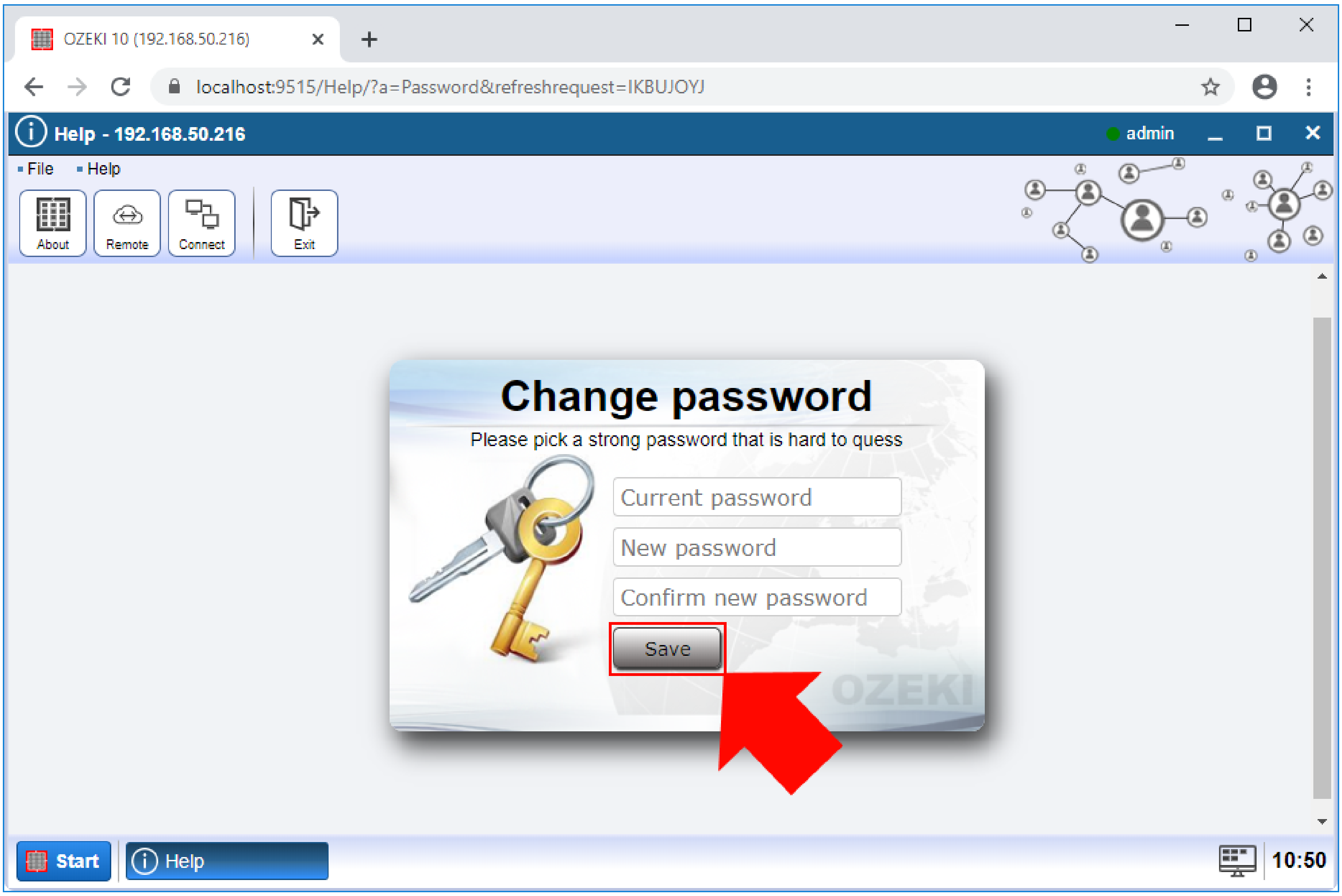


Figure 1 - Go to Change Password from the Desktop

After you selected the 'Change password' option, the following form shows up (Figure 2). This is the menu, where you can change and store a new password. For that, just type your current password in the first field, then type the new password in the second field and retype it in the third field as well. To save the modifications, just click on 'Save'.


OZEKI 10 (192.168.50.216)
localhost:9515/Help/?a=Password&refreshrequest=IKBUJOYJ
Help - 192.168.50.216
admin
File
Help
About
Remote
Connect
Exit
Start
Help
10:50

Change password

Please pick a strong password that is hard to quess

Current password

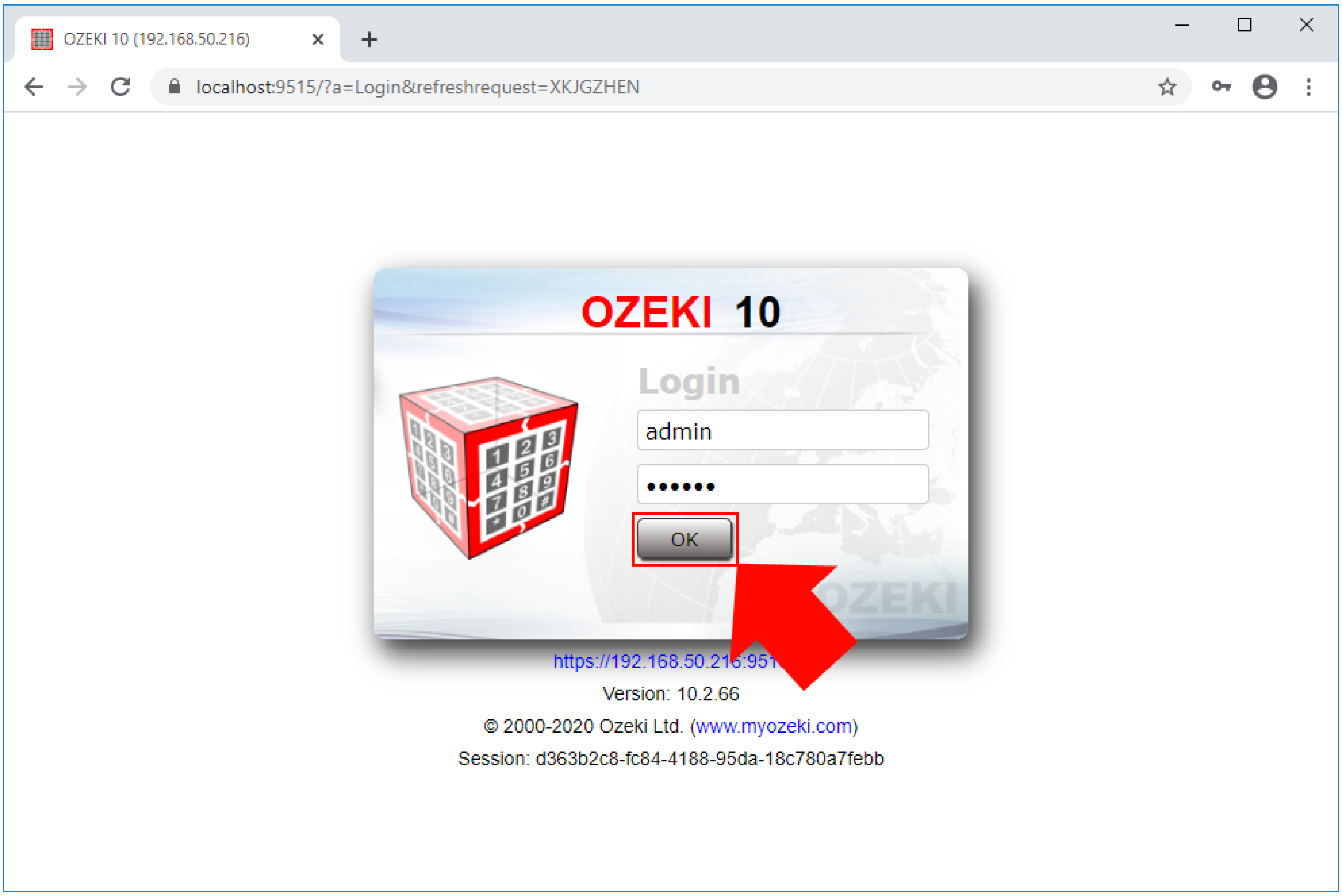
New password

Confirm new password

Save

Figure 2 - Change the password of your account

After saving a new password for your account, the application logs you out to log in again but now using your new password. All you have to do here is to enter your username and the changed password in the fields of the login form (Figure 3) and just click on 'OK'.



OZEKI 10 (192.168.50.216)
localhost:9515/?a=Login&refreshrequest=XKJGZHEN
OZEKI 10
Login
OK
OZEKI

admin

••••••

https://192.168.50.216:951

Version: 10.2.66

© 2000-2020 Ozeki Ltd. (www.myozeki.com)

Session: d363b2c8-fc84-4188-95da-18c780a7febb

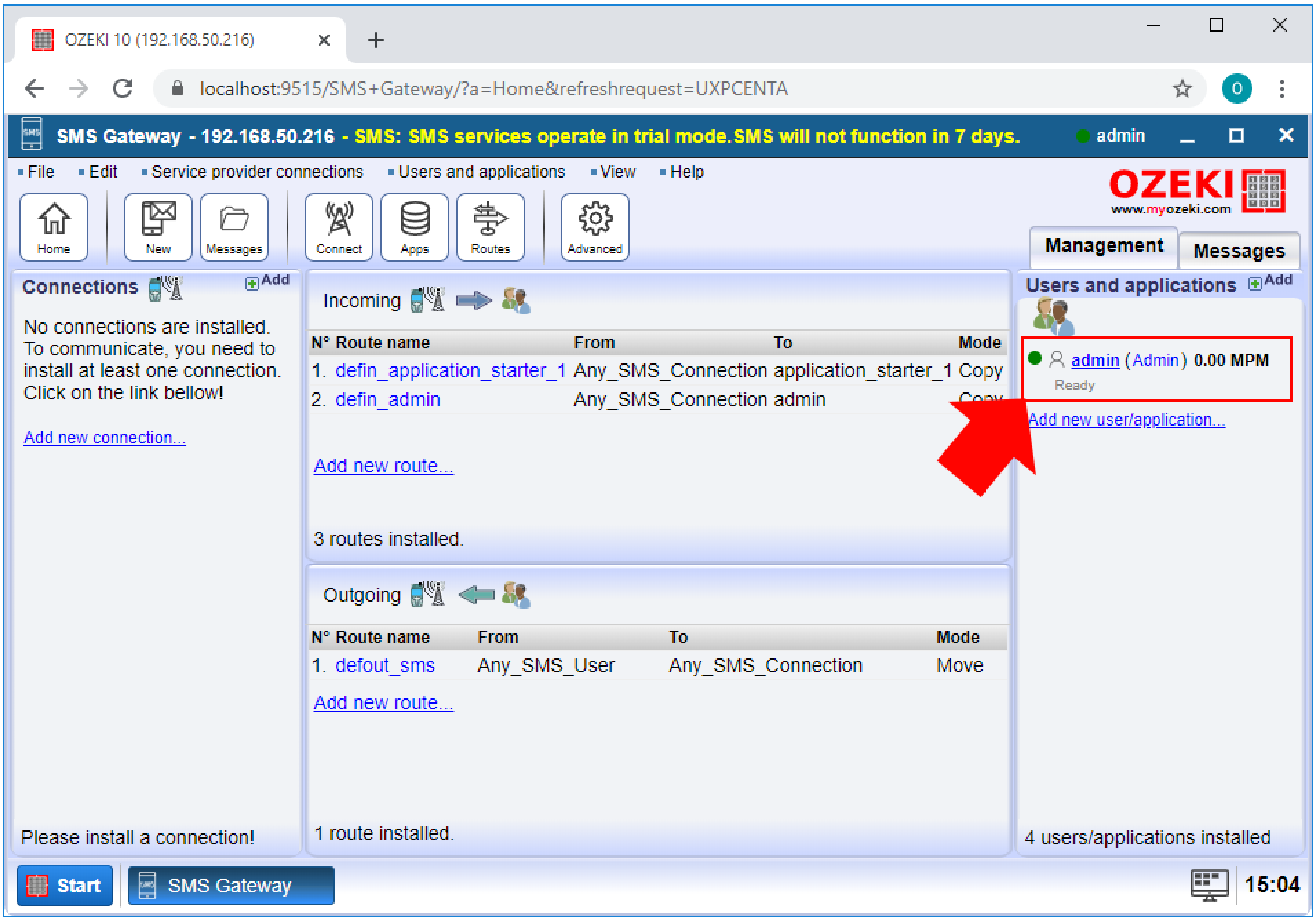
Figure 3 - Log in using the new password

# How to change a user password

***After you install Ozeki 10 SMS Gateway, you will create user accounts. These user accounts will be listed in the right hand panel of the SMS gateway management form. This guide explains how you can change the user password. The steps will be shown for the admin user. The steps are similar to other user accounts.***

## Step 1 - Open the details page of the user account

To reach the configuration setting of the user account, first you have to select it in the SMS Gateway Management page. On this page, simply click on the name of the user (Figure 1) in the Users and application panel.



SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode.SMS will not function in 7 days. admin

Connections

No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

Please install a connection!

Incoming

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defin_application_starter_1 | Any_SMS_Connection | application_starter_1 | Copy |
| 2. | defin_admin | Any_SMS_Connection | admin | Copy |

Add new route...

3 routes installed.

Outgoing

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move |

Add new route...

1 route installed.

Users and applications

● admin (Admin) 0.00 MPM
Ready

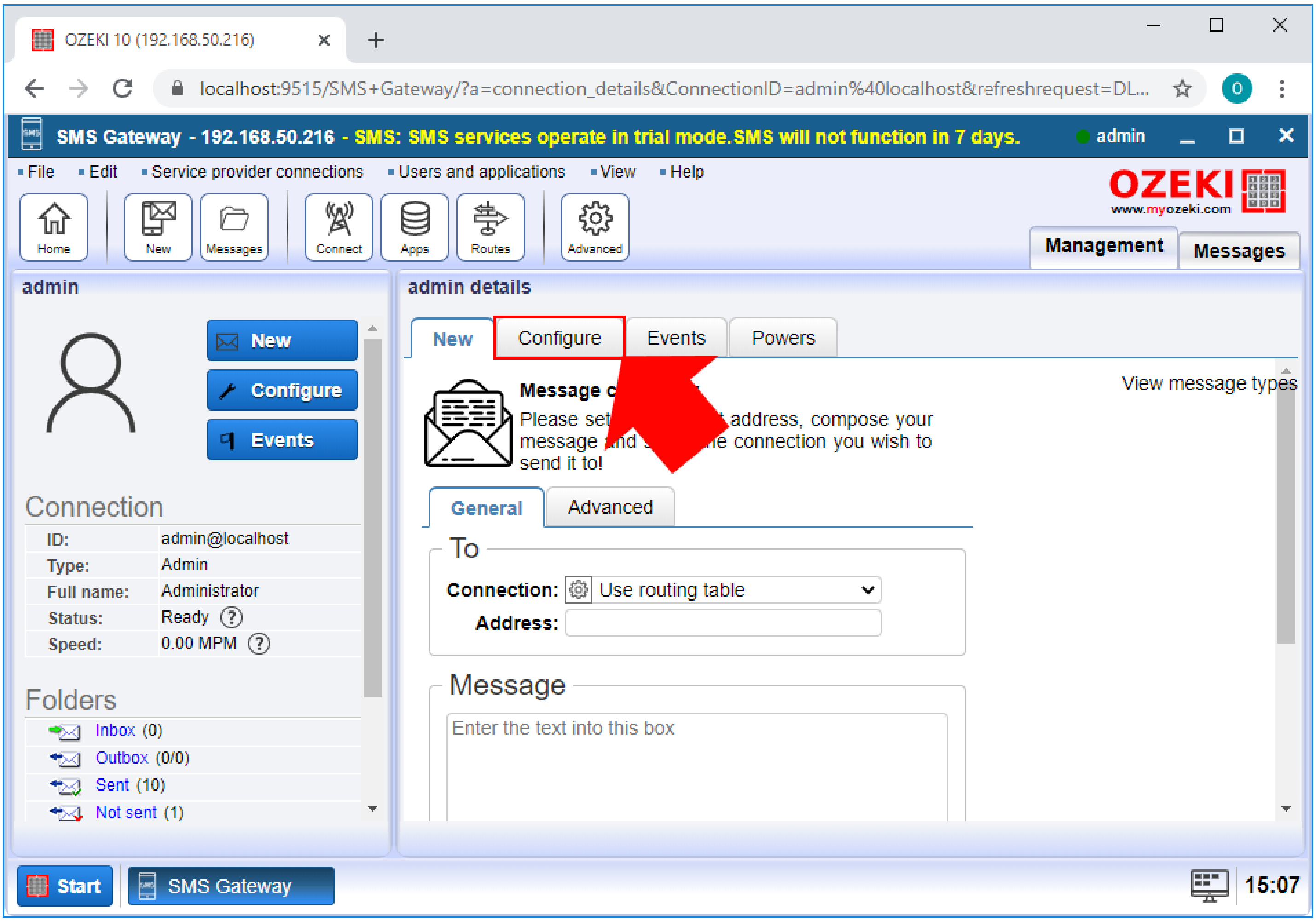
Add new user/application...

4 users/applications installed

Figure 1 - Select the user

## Step 2 - Open the Configure tab

This will bring up the details page of the selected user, you will notice that the main panel of the user account is the Message composer, so later you can use it to send SMS messages. To change the password, click on the Configure tab (Figure 2). This will bring you to the user's configuration form.



OZEKI 10 (192.168.50.216)

localhost:9515/SMS+Gateway/?a=connection_details&ConnectionID=admin%40localhost&refreshrequest=DL...

SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.

| | |
|---|---|
| ID: | admin@localhost |
| Type: | Admin |
| Full name: | Administrator |
| Status: | Ready |
| Speed: | 0.00 MPM |

Message c... Please set ... address, compose your message and s... he connection you wish to send it to!

Connection: Use routing table

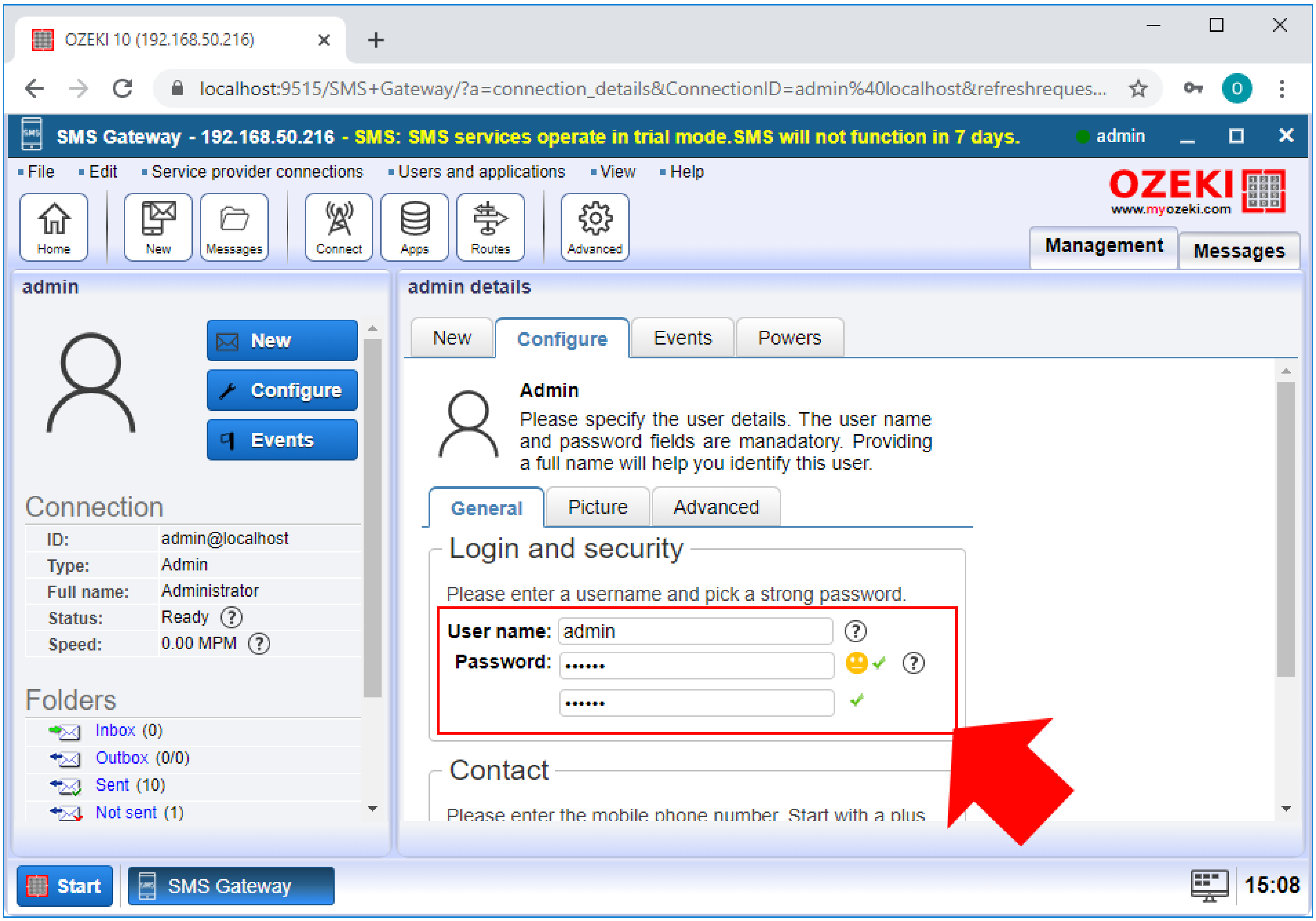
Address:

Enter the text into this box

Figure 2 - Message composer form

## Step 3 - Change the password

In the Configuration form, you can update the account details of the user (Figure 3). You can change the username and you can change the password. The password needs to be the combination of letters and digits and it should contains at least six characters to improve security.


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=connection_details&ConnectionID=admin%40localhost&refreshreques...
SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
OZEKI
www.myozeki.com
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
New
Configure
Events
Powers
General
Picture
Advanced
Start
SMS Gateway
15:08

admin

Connection

| | |
|---|---|
| ID: | admin@localhost |
| Type: | Admin |
| Full name: | Administrator |
| Status: | Ready |
| Speed: | 0.00 MPM |

Folders

- Inbox (0)
- Outbox (0/0)
- Sent (10)
- Not sent (1)

admin details

**Admin**

Please specify the user details. The user name and password fields are manadatory. Providing a full name will help you identify this user.

Login and security

Please enter a username and pick a strong password.

User name: admin

Password: ••••••

••••••

Contact

Please enter the mobile phone number. Start with a plus

Figure 3 - Configure the account details

# Ozeki 10 Administrators Guide

**How to backup your Ozeki 10 system**
This page explains how you can back up the configuration and other data of the Ozeki system. It is essential to back up your configuration and other data related to Ozeki 10 system. Your computer may break down, or there may be other problems causing data loss.
Learn how to backup you Ozeki 10 system

**How to restore the Ozeki 10 configuration**
From this guide you can learn how you can restore your Ozeki 10 configuration from the backup file after a system reinstallation, computer break down or any other data loss situation.
Learn to restore the Ozeki 10 configuration

**Firewall configuration of Ozeki 10**
Ozeki 10 uses different ports and protocols. It provides various services, so the ports in use may differ from system to system. The following table lists the most common ports used by Ozeki to help you with firewall setup.
Firewall configuration of Ozeki 10

**How to setup your SSL cert in Ozeki 10**
From this guide you can learn how you can setup your SSL certificate in Ozeki 10 to open it safely from anywhere.
Learn how to setup your SSL cert in Ozeki 10

**Trouble shooting HTTPS**
In case you experience problems with the HTTPS access to the GUI, you may use the following steps to figure out what causes the problem.
Learn how to trouble shoot HTTPS

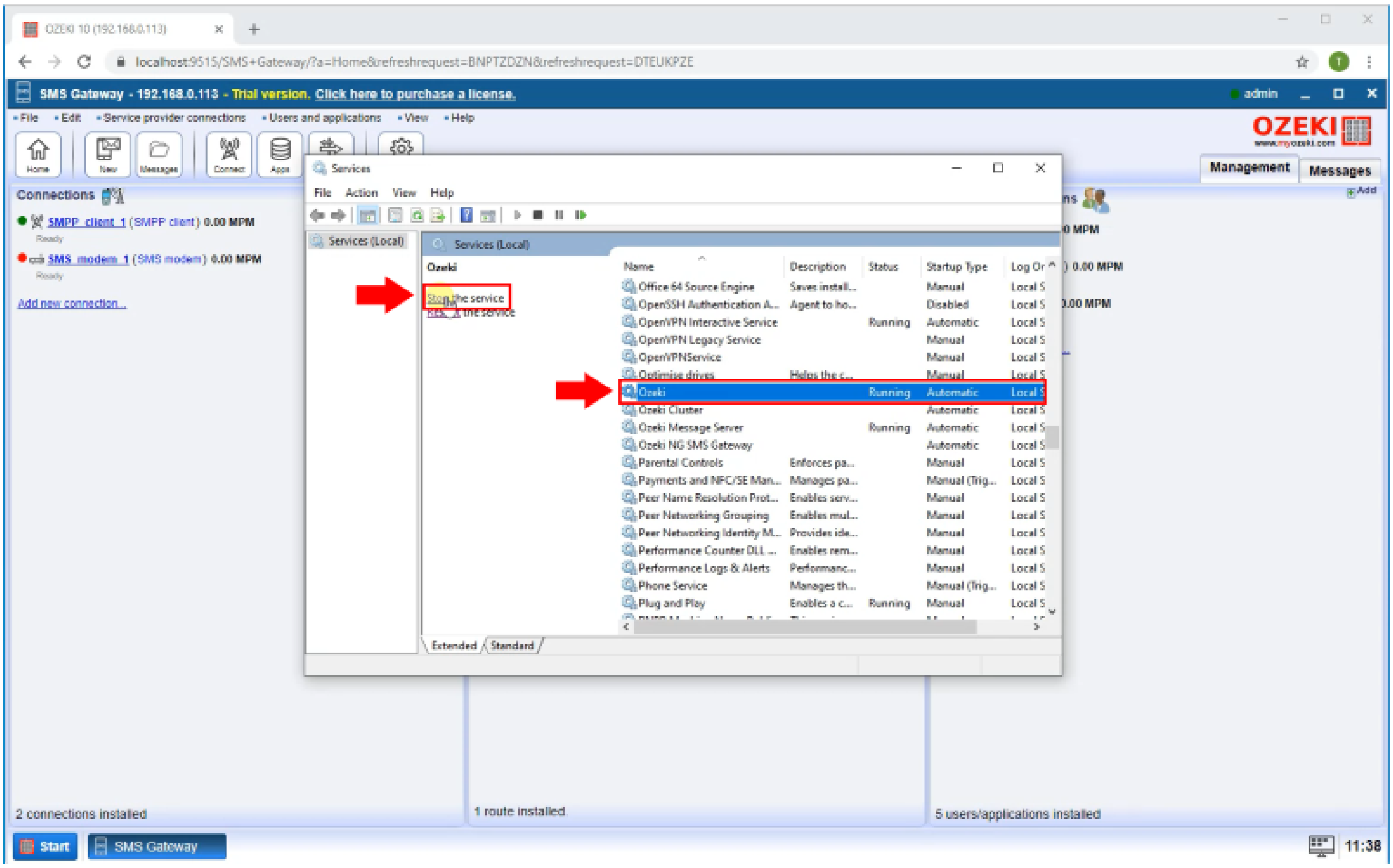
# How to backup your Ozeki 10 system

***This guide gives you information on how to backup the configuration and the data of your Ozeki 10 system. The procedure is relatively easy. You need to stop the Ozeki service and backup the Data directory. This directory contains all the configuration and data files. To restore your system using this data the steps can be found in the How to restore your Ozeki 10 system guide.***

## Video tutorial: Ozeki backup

## Step 1 - Stop the Ozeki service

First step is to stop the Ozeki windows service. You can do this by launching the services.msc command in windows. In the service manager select the Ozeki service from the list. After this click Stop on the left.



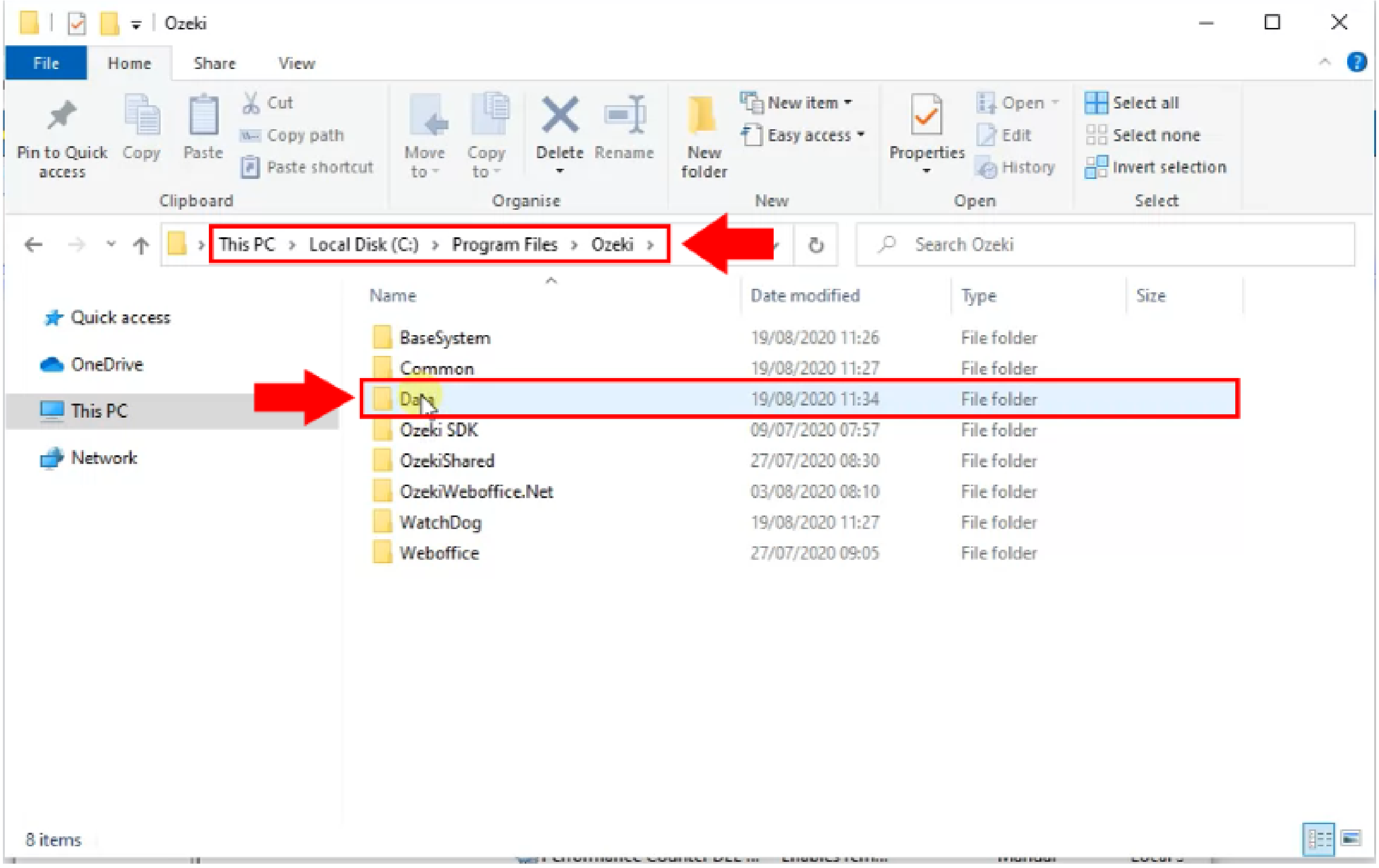
OZEKI 10 (192.168.0.113)
localhost:9515/SMS+Gateway/?a=Home&refreshrequest=BNPTZDZN&refreshrequest=DTEUKPZE
SMS Gateway - 192.168.0.113 - Trial version. Click here to purchase a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
OZEKI
Management
Messages
Connections
SMPP_client_1 (SMPP client) 0.00 MPM
SMS_modem_1 (SMS modem) 0.00 MPM
Add new connection...
Services
File
Action
View
Help
Services (Local)
Ozeki
Stop the service
Restart the service
Extended
Standard
2 connections installed
1 route installed
5 users/applications installed
Start
SMS Gateway
11:38

| Name | Description | Status | Startup Type | Log On |
|---|---|---|---|---|
| Office 64 Source Engine | Saves install... | | Manual | Local S |
| OpenSSH Authentication A... | Agent to ho... | | Disabled | Local S |
| OpenVPN Interactive Service | | Running | Automatic | Local S |
| OpenVPN Legacy Service | | | Manual | Local S |
| OpenVPNService | | | Manual | Local S |
| Optimise drives | Helps the c... | | Manual | Local S |
| Ozeki | | Running | Automatic | Local S |
| Ozeki Cluster | | | Automatic | Local S |
| Ozeki Message Server | | Running | Automatic | Local S |
| Ozeki NG SMS Gateway | | | Automatic | Local S |
| Parental Controls | Enforces pa... | | Manual | Local S |
| Payments and NFC/SE Man... | Manages pa... | | Manual (Trig... | Local S |
| Peer Name Resolution Prot... | Enables serv... | | Manual | Local S |
| Peer Networking Grouping | Enables mul... | | Manual | Local S |
| Peer Networking Identity M... | Provides ide... | | Manual | Local S |
| Performance Counter DLL ... | Enables rem... | | Manual | Local S |
| Performance Logs & Alerts | Performanc... | | Manual | Local S |
| Phone Service | Manages th... | | Manual (Trig... | Local S |
| Plug and Play | Enables a c... | Running | Manual | Local S |

Figure 1 - Stop Ozeki Service

## Step 2 - Locate the Ozeki data folder

Once the Ozeki service is stopped, you need to locate the Ozeki Data folder. You can find it in the C:\Program Files\Ozeki\. The name of the folder is C:\Program Files\Ozeki\Data.



| Name | Date modified | Type | Size |
|---|---|---|---|
| BaseSystem | 19/08/2020 11:26 | File folder | |
| Common | 19/08/2020 11:27 | File folder | |
| Data | 19/08/2020 11:34 | File folder | |
| Ozeki SDK | 09/07/2020 07:57 | File folder | |
| OzekiShared | 27/07/2020 08:30 | File folder | |
| OzekiWeboffice.Net | 03/08/2020 08:10 | File folder | |
| WatchDog | 19/08/2020 11:27 | File folder | |
| Weboffice | 27/07/2020 09:05 | File folder | |

Figure 2 - Locate data folder

## Step 3 - Compress the Ozeki data folder into a ZIP file

Once you have located the data folder, compress it into a Zip file. Right click on the folder then select "Send to" and "Compressed (zipped) folder".

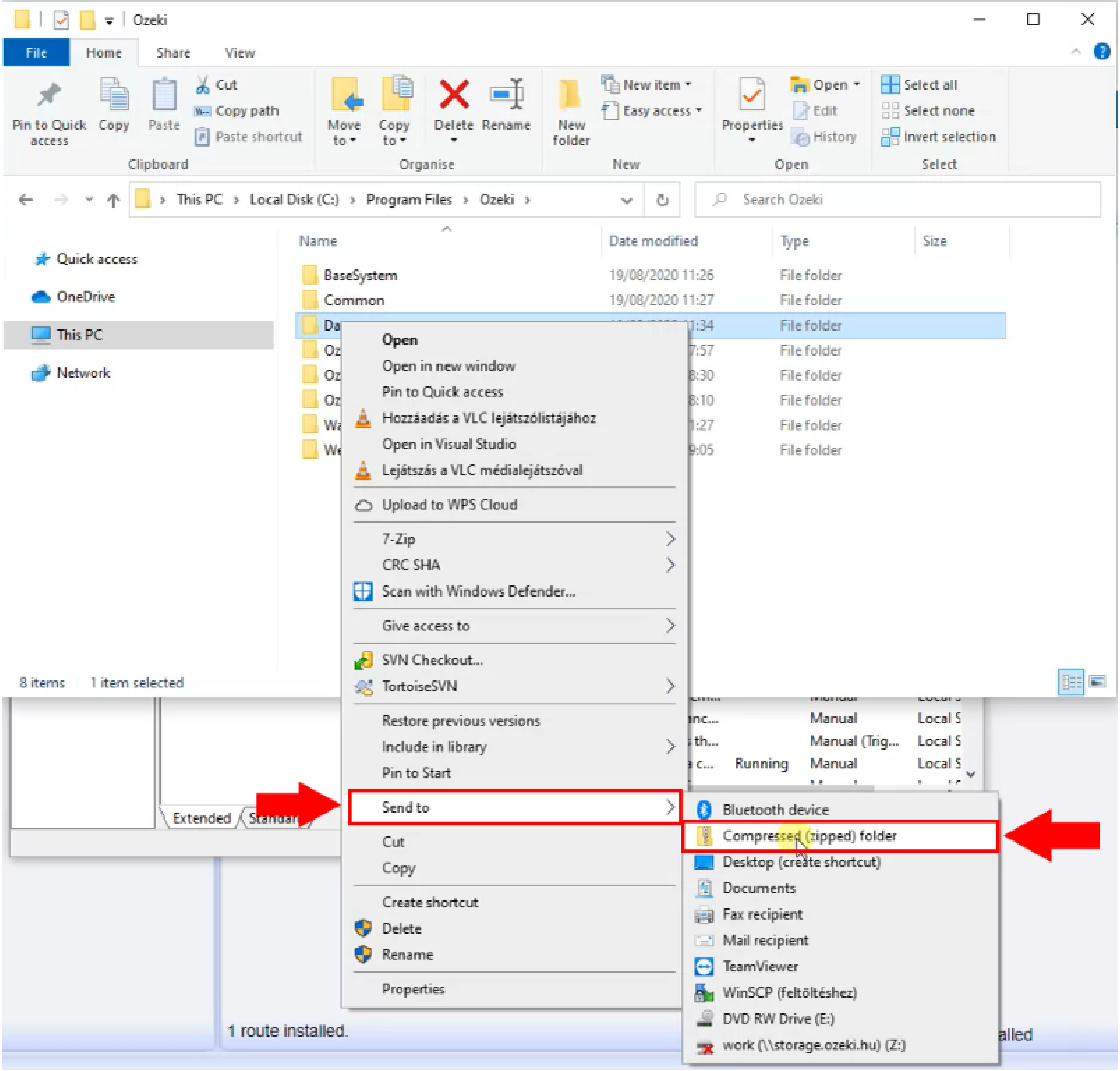


Figure 3 - Compress Ozeki data folder

## Step 4 - Locate the compressed backup file

The previous step created the Data.zip file on your desktop. Look through your desktop icons and you will find the file.

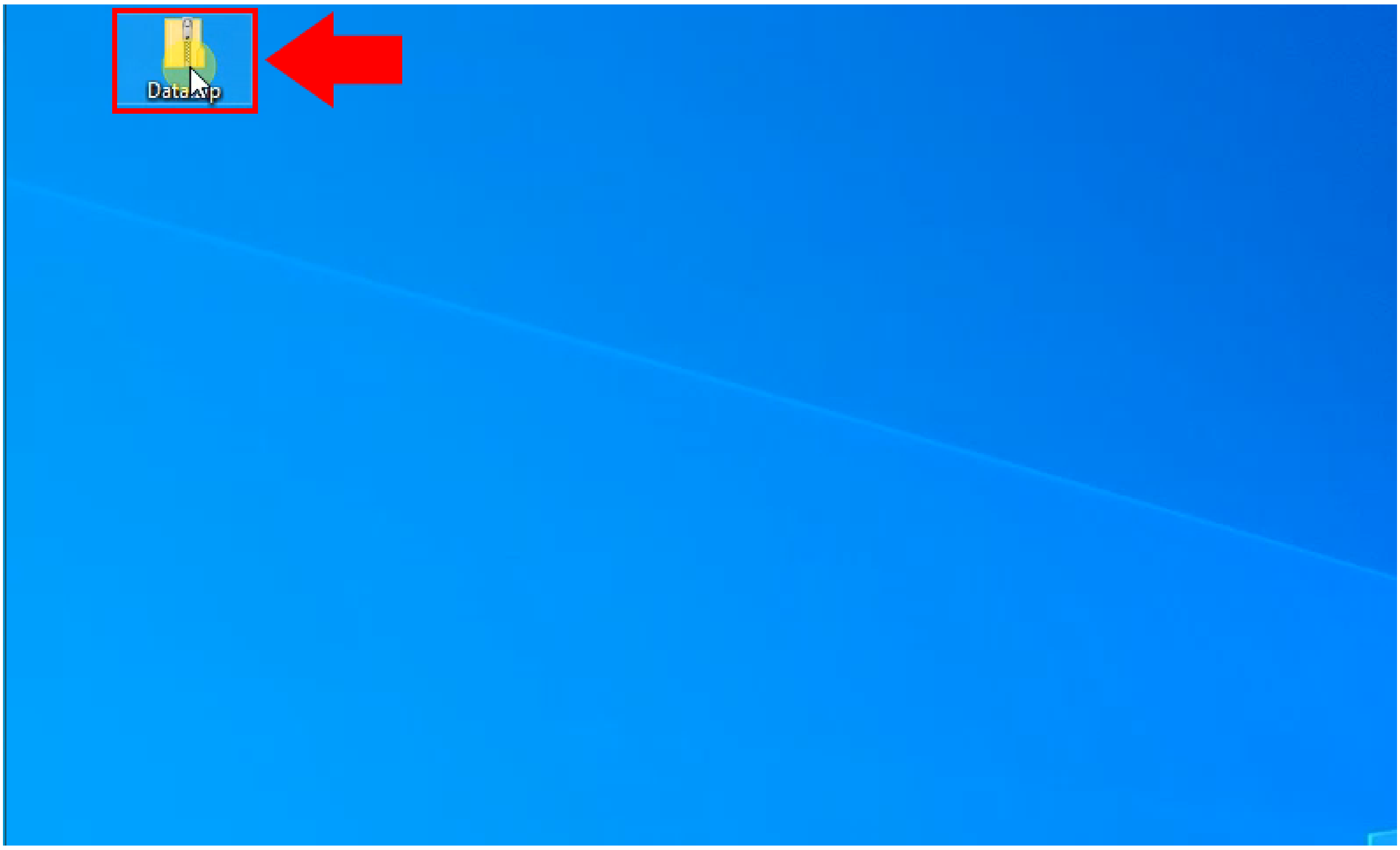


Figure 4 - Backup data folder

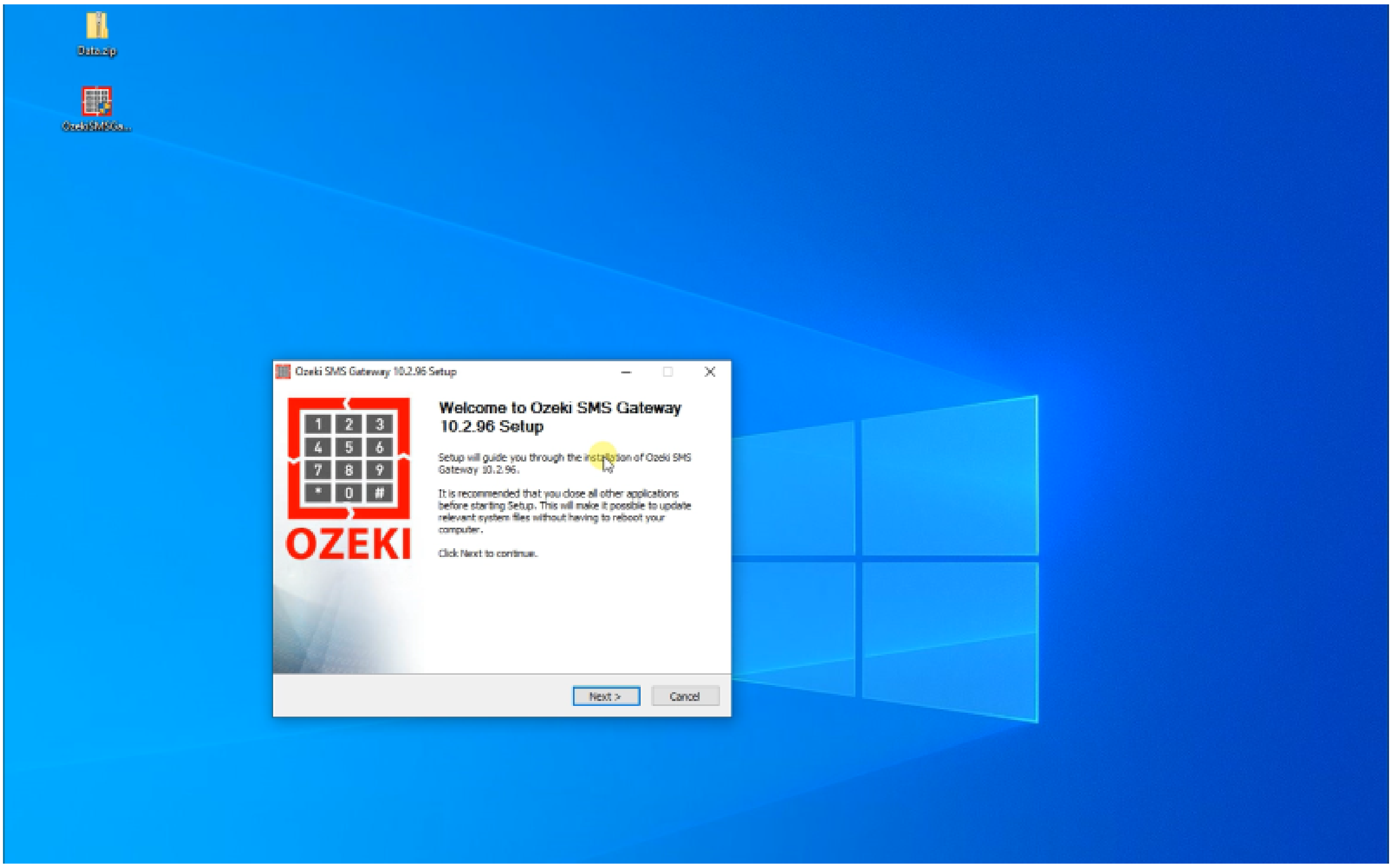
# How to restore the Ozeki 10 configuration

***This guide explains how you can restore your data and your configuration files. The guide requires you to have a backup file called Data.zip. It was saved when you have created a backup of your Ozeki 10 system. This procedure can be used if you move Ozeki 10 from one computer to another or you when you reinstall Windows and you need to install Ozeki again.***

## Video tutorial: Restore Ozeki 10 from backup

## Step 1 - Install Ozeki 10

The first step is to install the Ozeki software on your PC. You can find a detaild installation guide in the Ozeki SMS Gateway Offline Installation page.



Welcome to Ozeki SMS Gateway 10.2.96 Setup

Setup will guide you through the installation of Ozeki SMS Gateway 10.2.96.

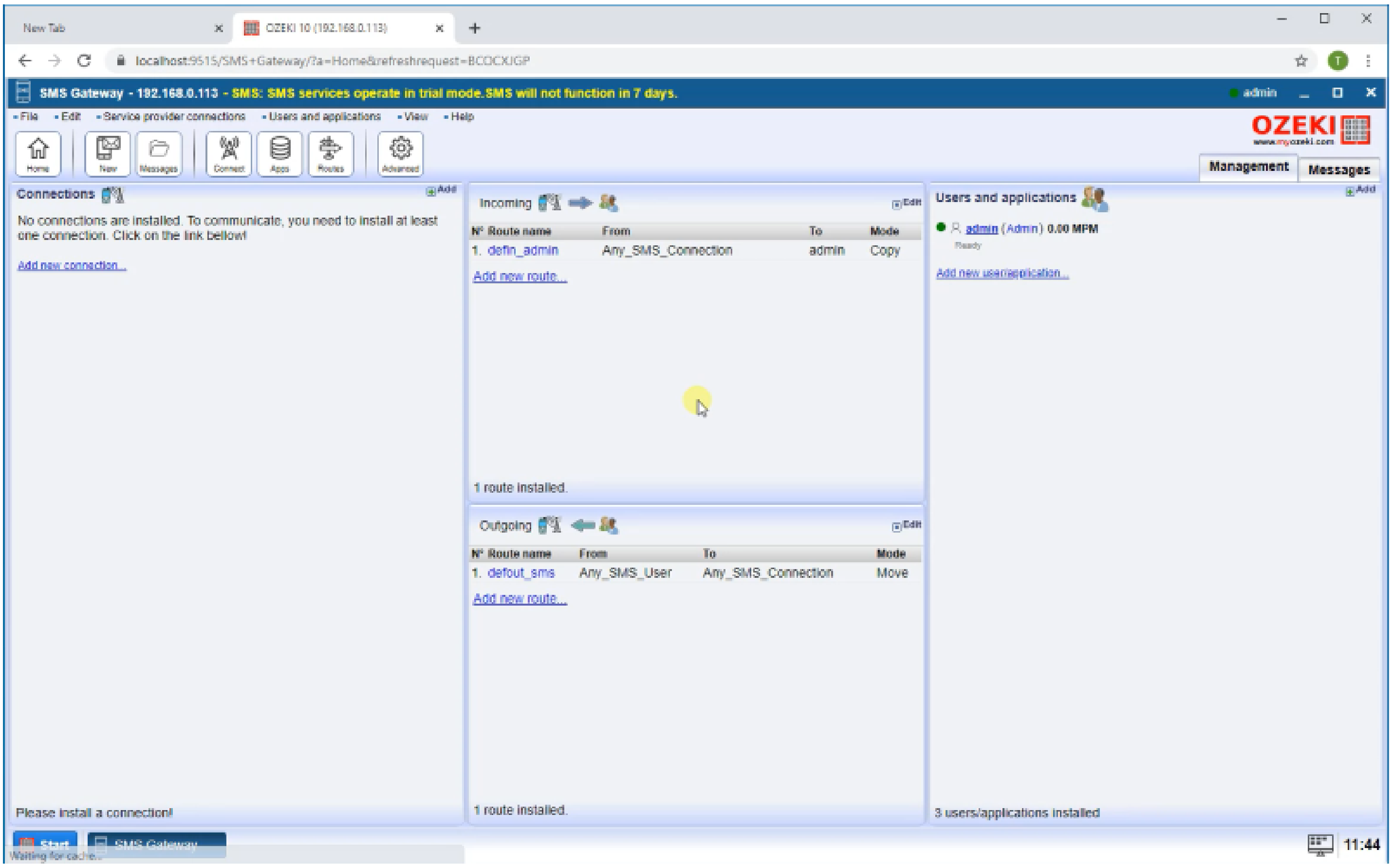
It is recommended that you close all other applications before starting Setup. This will make it possible to update relevant system files without having to reboot your computer.

Click Next to continue.

Figure 1 - Install Ozeki SMS Gateway

## Step 2 - Stop the Ozeki service

After the installation you can see that the Ozeki SMS Gateway has no configuration yet.



SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.

Connections

No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

Please install a connection!

Incoming

| Nº | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defin_admin | Any_SMS_Connection | admin | Copy |

Add new route...

1 route installed.

Outgoing

| Nº | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move |

Add new route...

1 route installed.

Users and applications

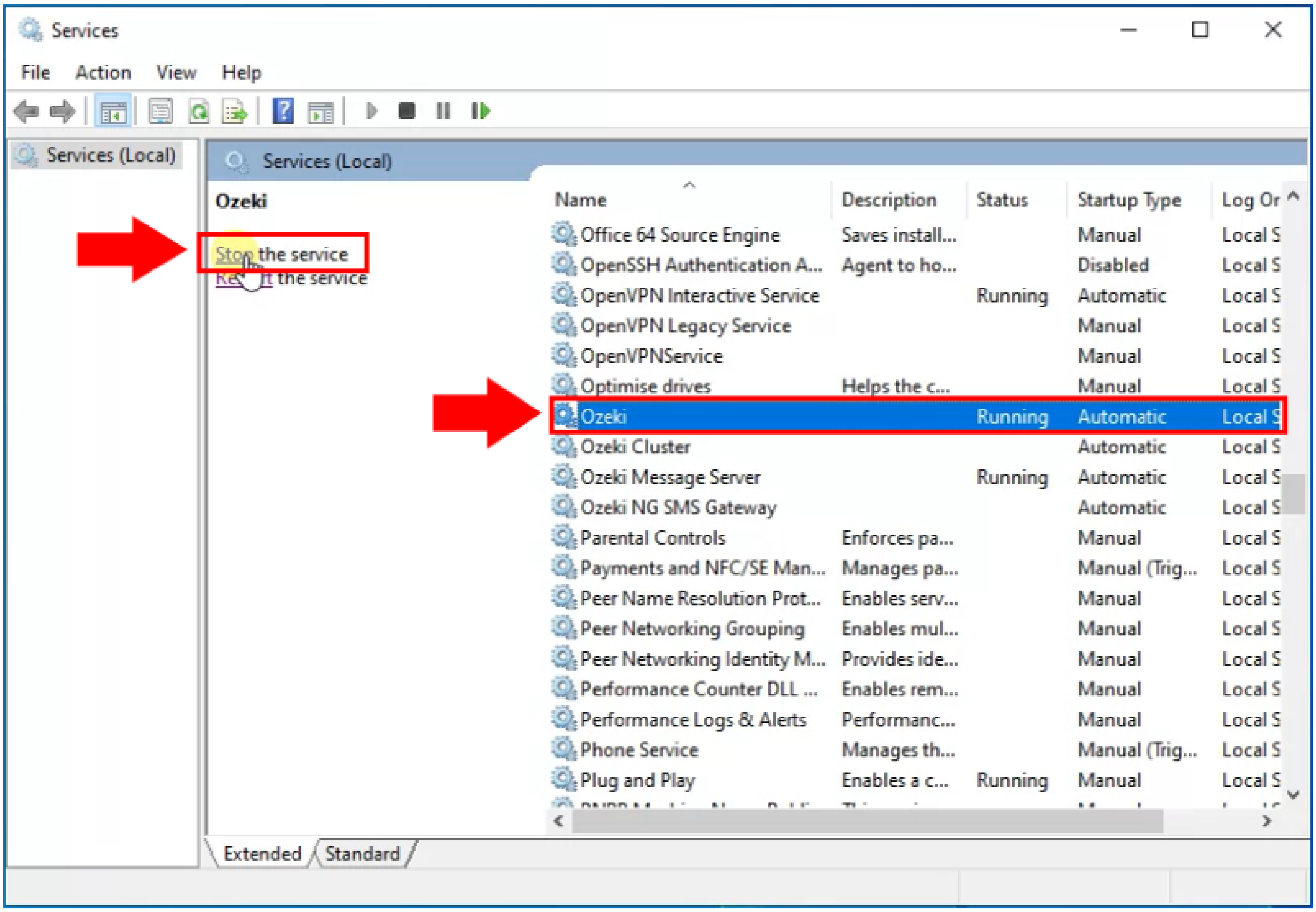
● admin (Admin) 0.00 MPM
Ready

Add new user/application...

3 users/applications installed

Figure 2 - SMS Gateway has no configuration

To restore the configuration you need to stop the Ozeki service under the services.msc. Select the Ozeki service from the list and click Stop on the left side.



Services

Ozeki

Stop the service
Restart the service

| Name | Description | Status | Startup Type | Log On |
|---|---|---|---|---|
| Office 64 Source Engine | Saves install... | | Manual | Local S |
| OpenSSH Authentication A... | Agent to ho... | | Disabled | Local S |
| OpenVPN Interactive Service | | Running | Automatic | Local S |
| OpenVPN Legacy Service | | | Manual | Local S |
| OpenVPNService | | | Manual | Local S |
| Optimise drives | Helps the c... | | Manual | Local S |
| Ozeki | | Running | Automatic | Local S |
| Ozeki Cluster | | | Automatic | Local S |
| Ozeki Message Server | | Running | Automatic | Local S |
| Ozeki NG SMS Gateway | | | Automatic | Local S |
| Parental Controls | Enforces pa... | | Manual | Local S |
| Payments and NFC/SE Man... | Manages pa... | | Manual (Trig... | Local S |
| Peer Name Resolution Prot... | Enables serv... | | Manual | Local S |
| Peer Networking Grouping | Enables mul... | | Manual | Local S |
| Peer Networking Identity M... | Provides ide... | | Manual | Local S |
| Performance Counter DLL ... | Enables rem... | | Manual | Local S |
| Performance Logs & Alerts | Performanc... | | Manual | Local S |
| Phone Service | Manages th... | | Manual (Trig... | Local S |
| Plug and Play | Enables a c... | Running | Manual | Local S |

Figure 3 - Stop Ozeki service

# Step 3 - Delete the default data folder

Now delete the existing data folder. This folder can be found at C:\Program Files\Ozeki\Data. This is the default config that came with the Ozeki installation package.

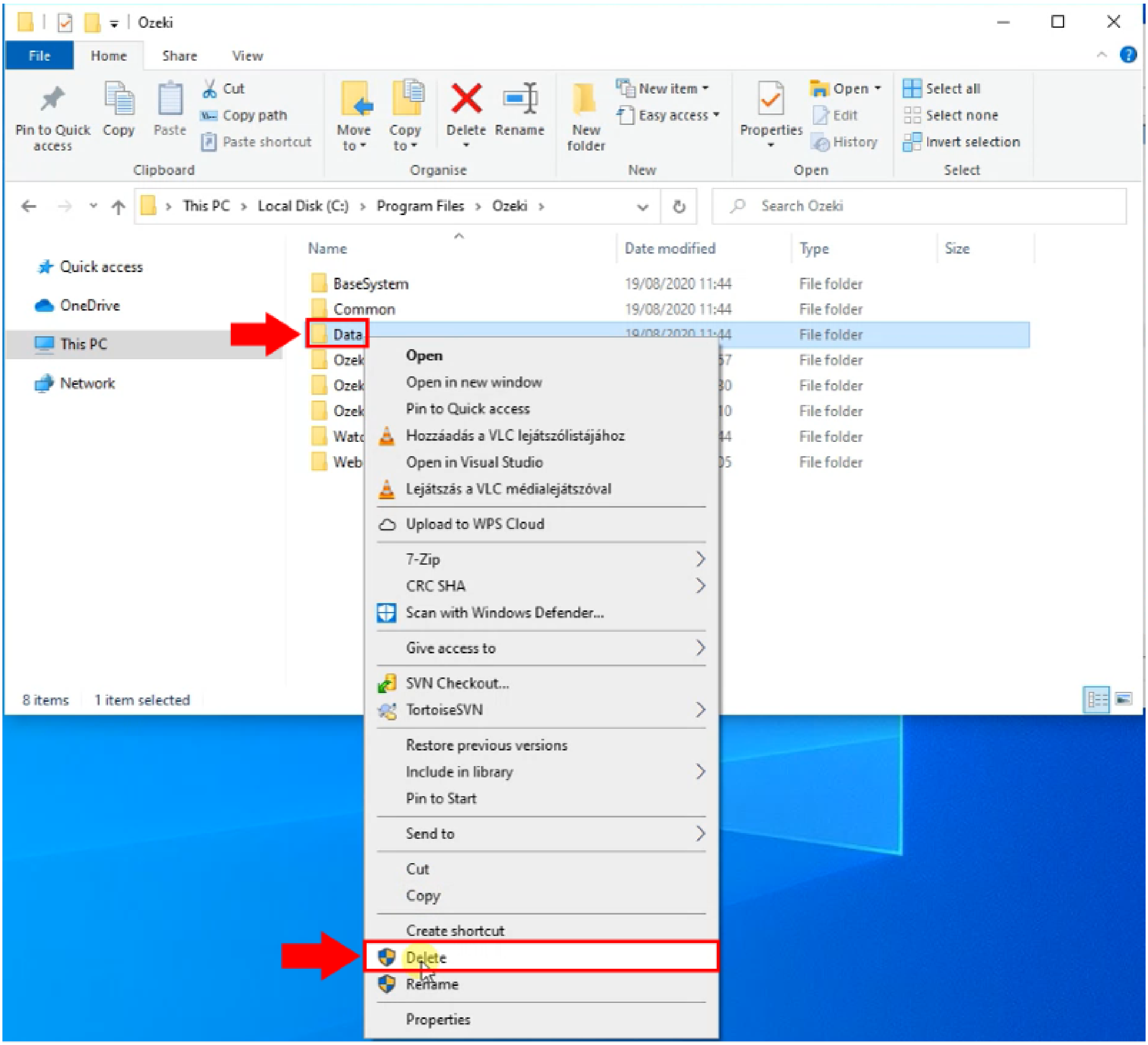


Figure 4 - Delete data folder

# Step 4 - Extract the backup config

Next extract the backup data zip by right clicking on the Data.zip and by selecting "Extract All...".

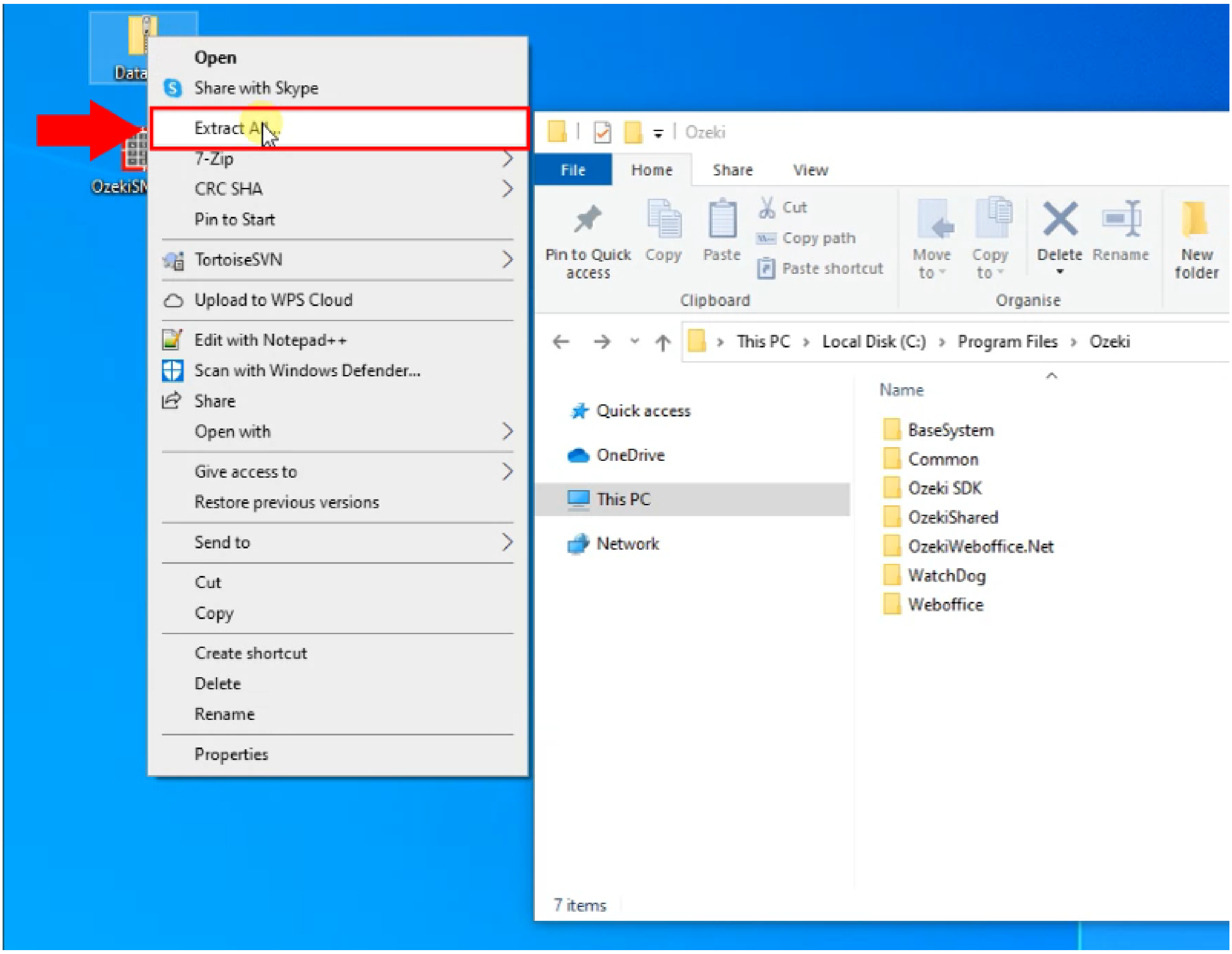


Figure 5 - Extract backup data file

## Step 5 - Copy the data folder to the appropriate location

After the zip file was successfully extracted, move the extracted Data folder to the C:\Program Files\Ozeki folder.

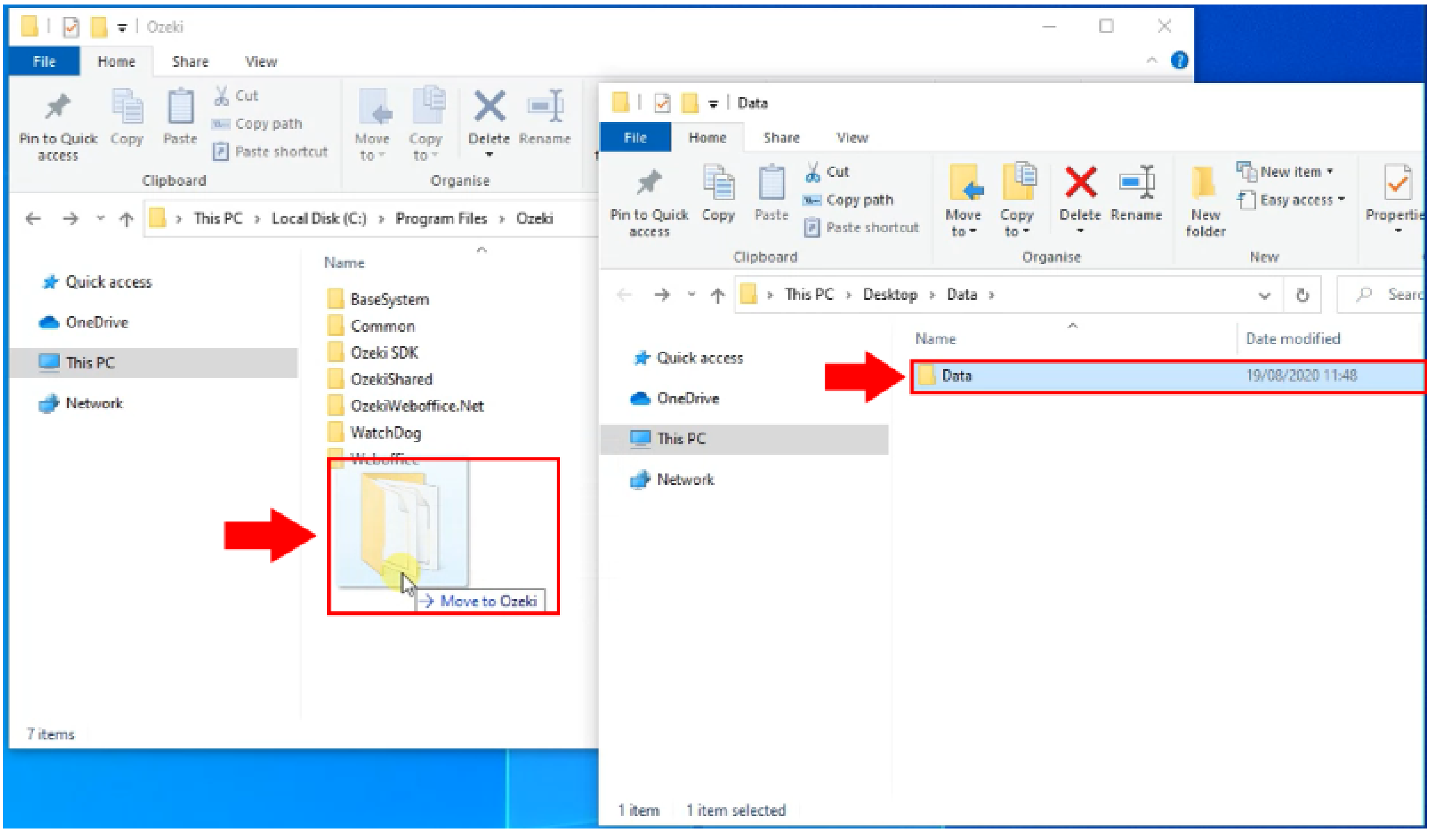
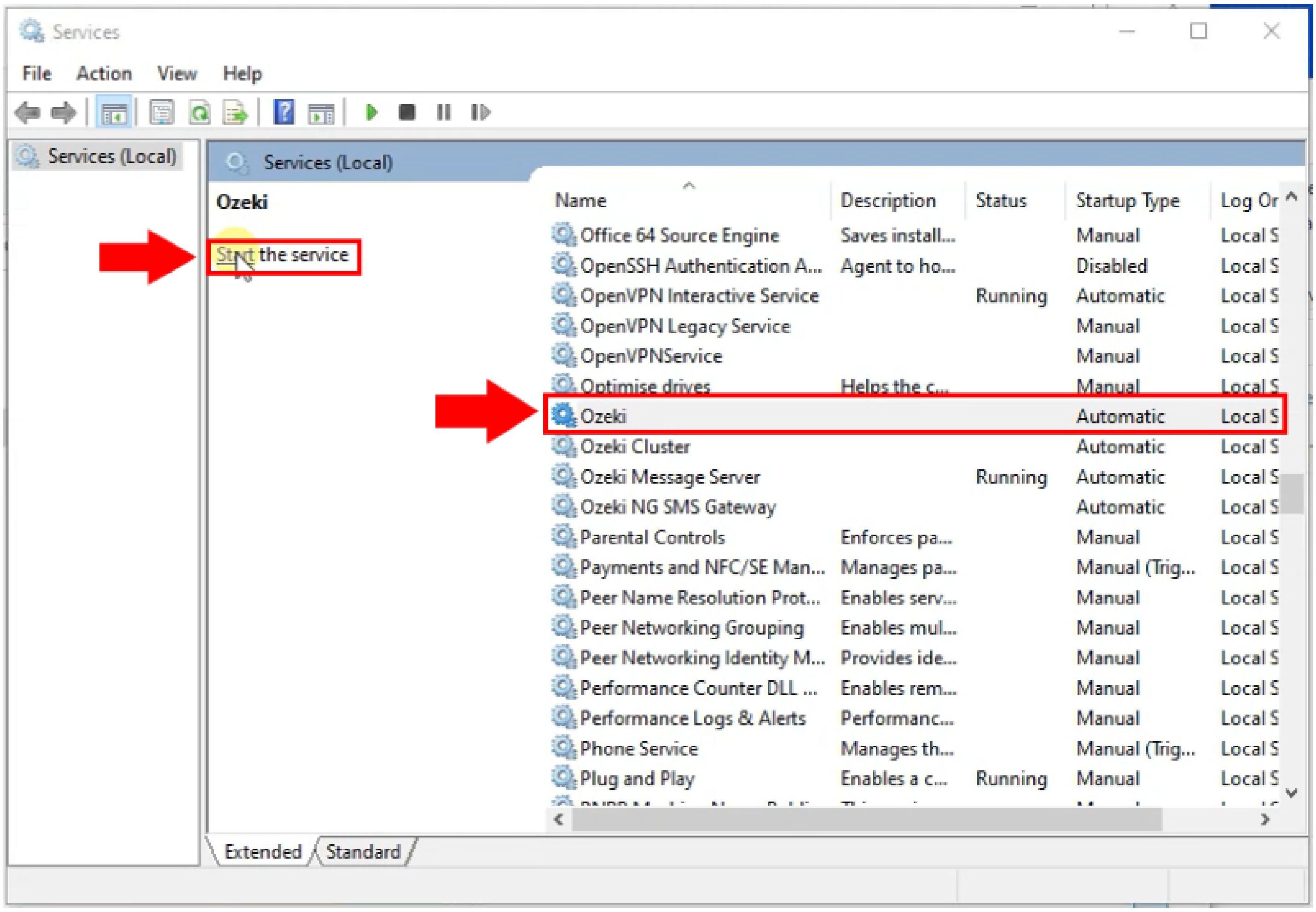


Figure 6 - Move new data folder to the Ozeki folder

# Step 5 - Start the Ozeki service

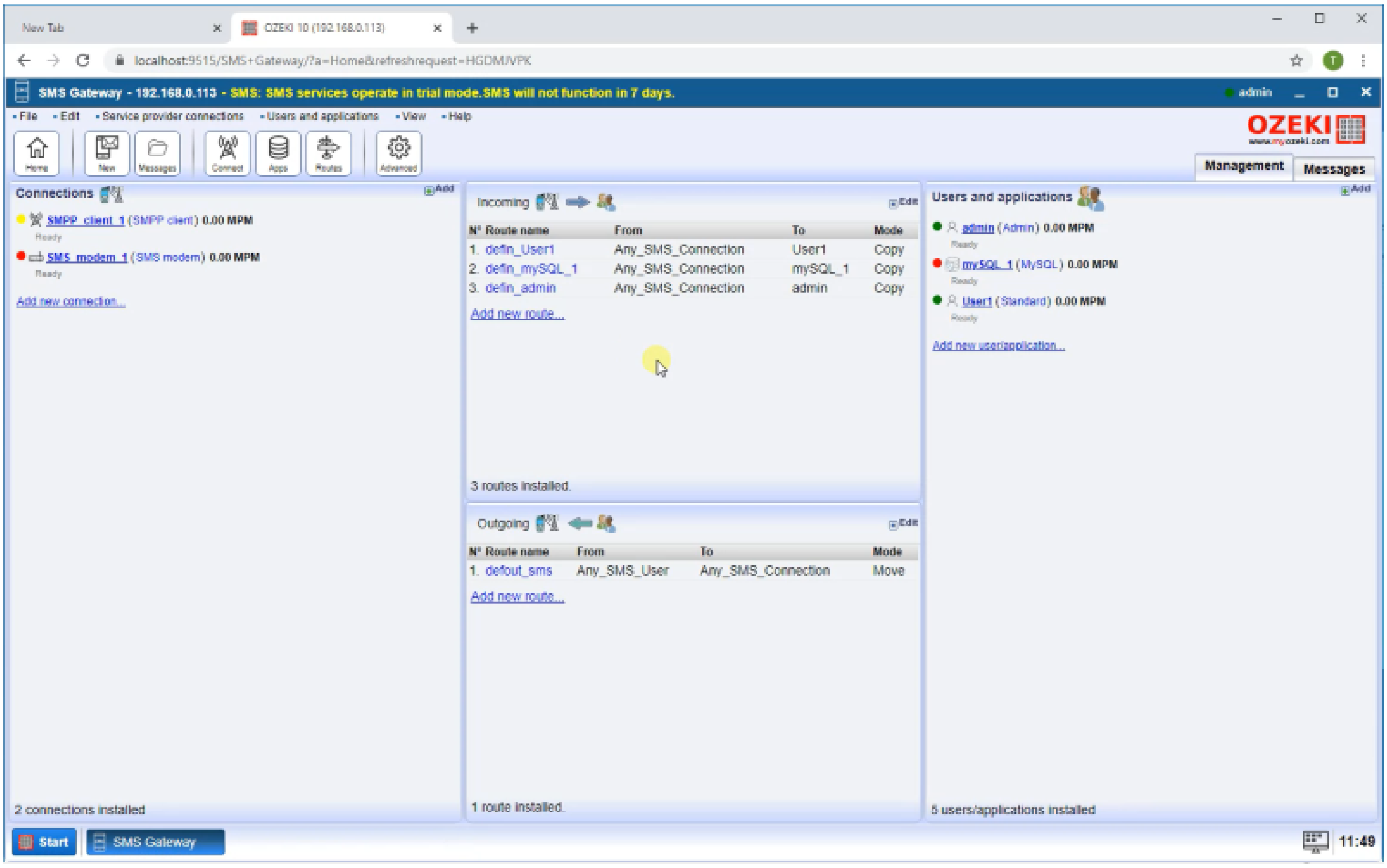
Finally you need to start the Ozeki servce again under the services.msc. Select the Ozeki service from the list and click Start on the left side.



| Name | Description | Status | Startup Type | Log On |
|---|---|---|---|---|
| Office 64 Source Engine | Saves install... | | Manual | Local S |
| OpenSSH Authentication A... | Agent to ho... | | Disabled | Local S |
| OpenVPN Interactive Service | | Running | Automatic | Local S |
| OpenVPN Legacy Service | | | Manual | Local S |
| OpenVPNService | | | Manual | Local S |
| Optimise drives | Helps the c... | | Manual | Local S |
| Ozeki | | | Automatic | Local S |
| Ozeki Cluster | | | Automatic | Local S |
| Ozeki Message Server | | Running | Automatic | Local S |
| Ozeki NG SMS Gateway | | | Automatic | Local S |
| Parental Controls | Enforces pa... | | Manual | Local S |
| Payments and NFC/SE Man... | Manages pa... | | Manual (Trig... | Local S |
| Peer Name Resolution Prot... | Enables serv... | | Manual | Local S |
| Peer Networking Grouping | Enables mul... | | Manual | Local S |
| Peer Networking Identity M... | Provides ide... | | Manual | Local S |
| Performance Counter DLL ... | Enables rem... | | Manual | Local S |
| Performance Logs & Alerts | Performanc... | | Manual | Local S |
| Phone Service | Manages th... | | Manual (Trig... | Local S |
| Plug and Play | Enables a c... | Running | Manual | Local S |

Figure 7 - Start Ozeki service

If the service is started you can see that the configuration is restored successfully.



SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.

Connections

- ● SMPP_client_1 (SMPP client) 0.00 MPM Ready
- ● SMS_modem_1 (SMS modem) 0.00 MPM Ready

Add new connection...

2 connections installed

Incoming

| Nº | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defin_User1 | Any_SMS_Connection | User1 | Copy |
| 2. | defin_mySQL_1 | Any_SMS_Connection | mySQL_1 | Copy |
| 3. | defin_admin | Any_SMS_Connection | admin | Copy |

Add new route...

3 routes installed.

Outgoing

| Nº | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move |

Add new route...

1 route installed.

Users and applications

- ● admin (Admin) 0.00 MPM Ready
- ● mySQL_1 (MySQL) 0.00 MPM Ready
- ● User1 (Standard) 0.00 MPM Ready

Add new user/application...

5 users/applications installed

Figure 8 - Ozeki configuration restored

# How to configure logging

***The low level communication of connections, users or application can be really useful when you would like to see every event that occurred during the sending process. The low level communication helps you to understand how a message is delivered to the recipient and in case of a failure, you can understand the reason by analyzing the low level communication of connections. Here, you can check what types of low level communication can be reached in Ozeki SMS Gateway.***

**Maintaining a log for each message**
Ozeki SMS Gateway allows you to follow the route of the messages. That means you can check what connections are involved in the message sending process. Every message has got a delivery history where you can view the route of the message, and check if the message was delivered successfully to the recipient.
Learn more about how you can trace your messages

**Logging message events**
Every messaging connection ensure logging of the message events by default. This logging means that only the protocol communication is displayed as events. You can activate further logging of the messaging events, which allows you to see every event of the messaging delivery with information about the event, the connections involved and the time when the event occurred.
Check how you can log events of messages

**Logging low level communication**
The low level communication can be activated for every Ozeki connection. This type of communication stores the events of the connections, which makes it possible to see how the connection communicate with each other. These logged communication also stored in a text file that can be opened with a simple notepad.
See how to log low level communication

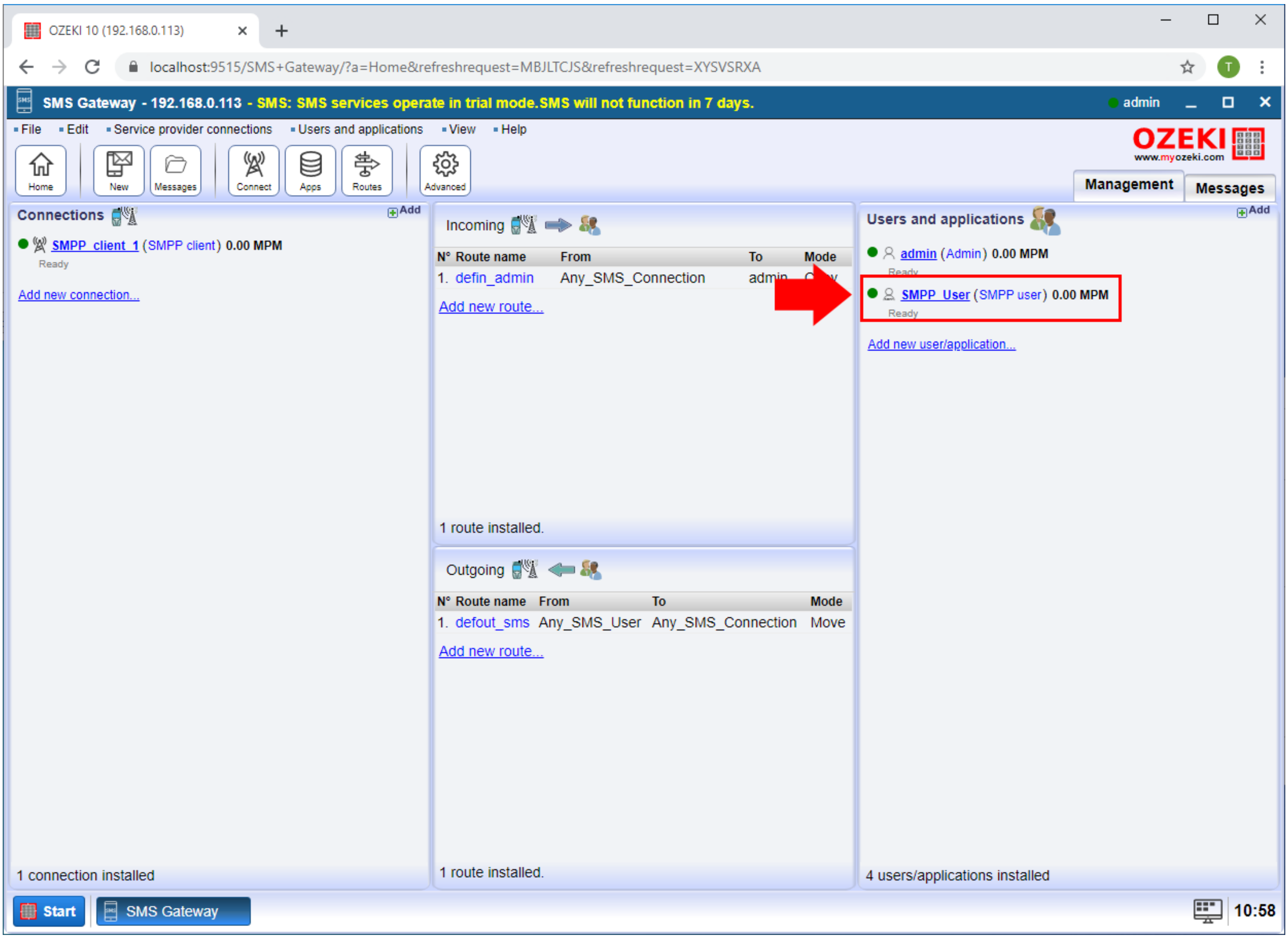
# How to log low level communication

Itt egy SMPP usernél azt kell bemtuatni, hogy a low level communication log bekapcsolásával megjelennek az SMPP PDU-k a logban. Meg kell mutatni bekapcsolás előtt és után az event tabpage-et. Meg kell azt is mutatni, hogy az event log fejlécéből, hogyan lehet kinézni a log fájl helyét, és, hogy a C drive-on hol lehet megtalálni a logokat, hogyan lehet megnyitni notepad-ban.

***This document is about to provide some information about how you can enable and use low level communication for any of your connections in Ozeki SMS Gateway. The step by step guide below explains how you can enable low level communication in the settings of a connection and what difference the low level communication makes in loggins of the events. You will be able to see also, where the SMS Gateway saves the log file and how you can open it on your computer. It does not take more than 10 minutes to complete the guide, so let's start right now!***

# How to trace what happened to a message

The first step is to open the source user connection on the right side of the SMS Gateway Manager console page as you can see it on the Figure 1.



localhost:9515/SMS+Gateway/?a=Home&refreshrequest=MBJLTCJS&refreshrequest=XYSVSRXA

SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.

Connections

SMPP_client_1 (SMPP client) 0.00 MPM
Ready

Add new connection...

1 connection installed

Incoming

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defin_admin | Any_SMS_Connection | admin | Copy |

Add new route...

1 route installed.

Outgoing

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move |

Add new route...

1 route installed.

Users and applications

admin (Admin) 0.00 MPM
Ready

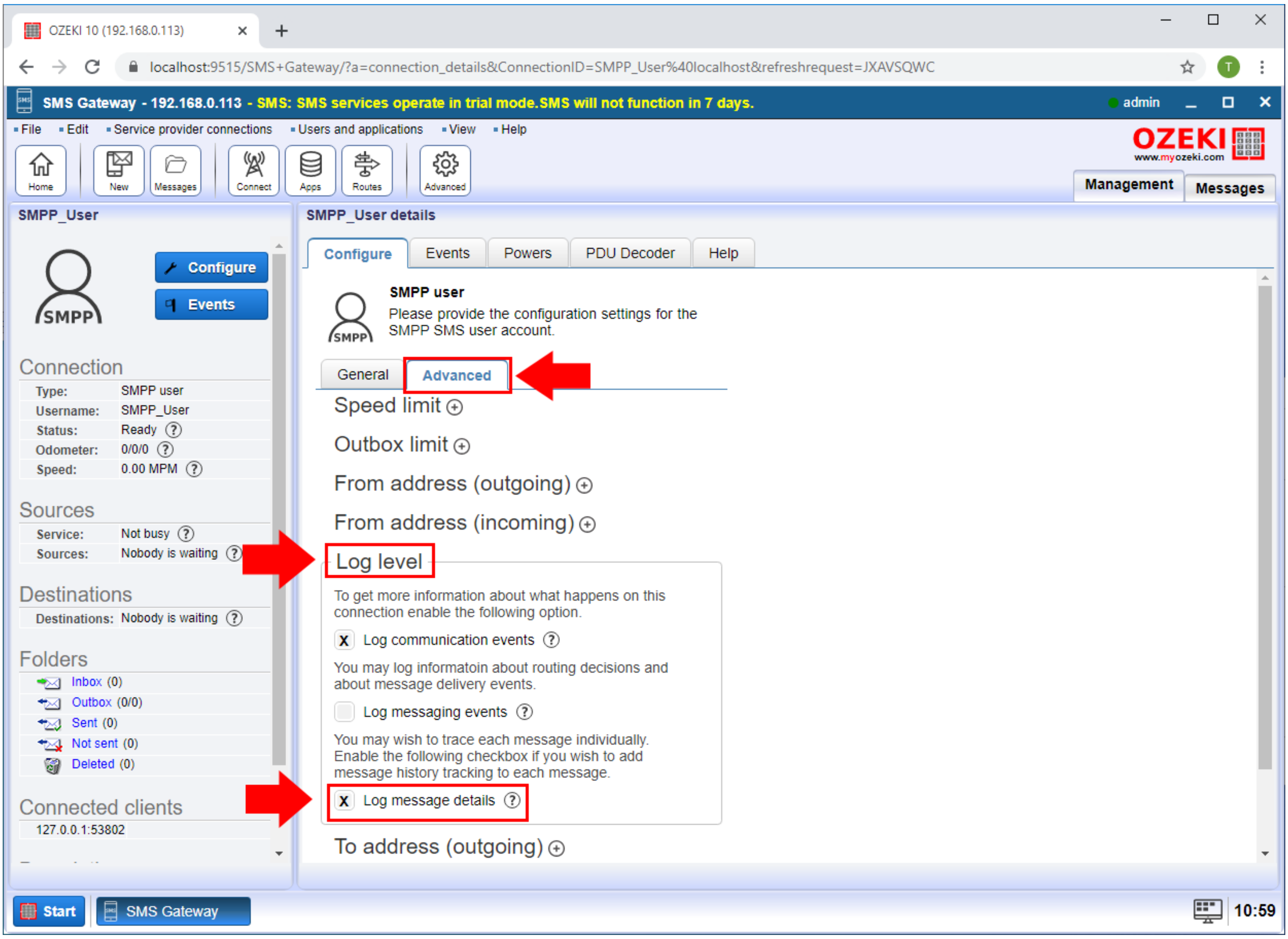
SMPP_User (SMPP user) 0.00 MPM
Ready

Add new user/application...

4 users/applications installed

Figure 1 - Open SMS user

Now in the Log level section of the user's Advanced tab enable the 'Log message details' checkbox if you wish to add message history tracking to each message (Figure 2).



SMPP user

Please provide the configuration settings for the SMPP SMS user account.

Connection

| | |
|---|---|
| Type: | SMPP user |
| Username: | SMPP_User |
| Status: | Ready |
| Odometer: | 0/0/0 |
| Speed: | 0.00 MPM |

Sources

| | |
|---|---|
| Service: | Not busy |
| Sources: | Nobody is waiting |

Destinations

| | |
|---|---|
| Destinations: | Nobody is waiting |

Folders

- Inbox (0)
- Outbox (0/0)
- Sent (0)
- Not sent (0)
- Deleted (0)

Connected clients

127.0.0.1:53802

Log level

To get more information about what happens on this connection enable the following option.

☒ Log communication events

You may log informatoin about routing decisions and about message delivery events.

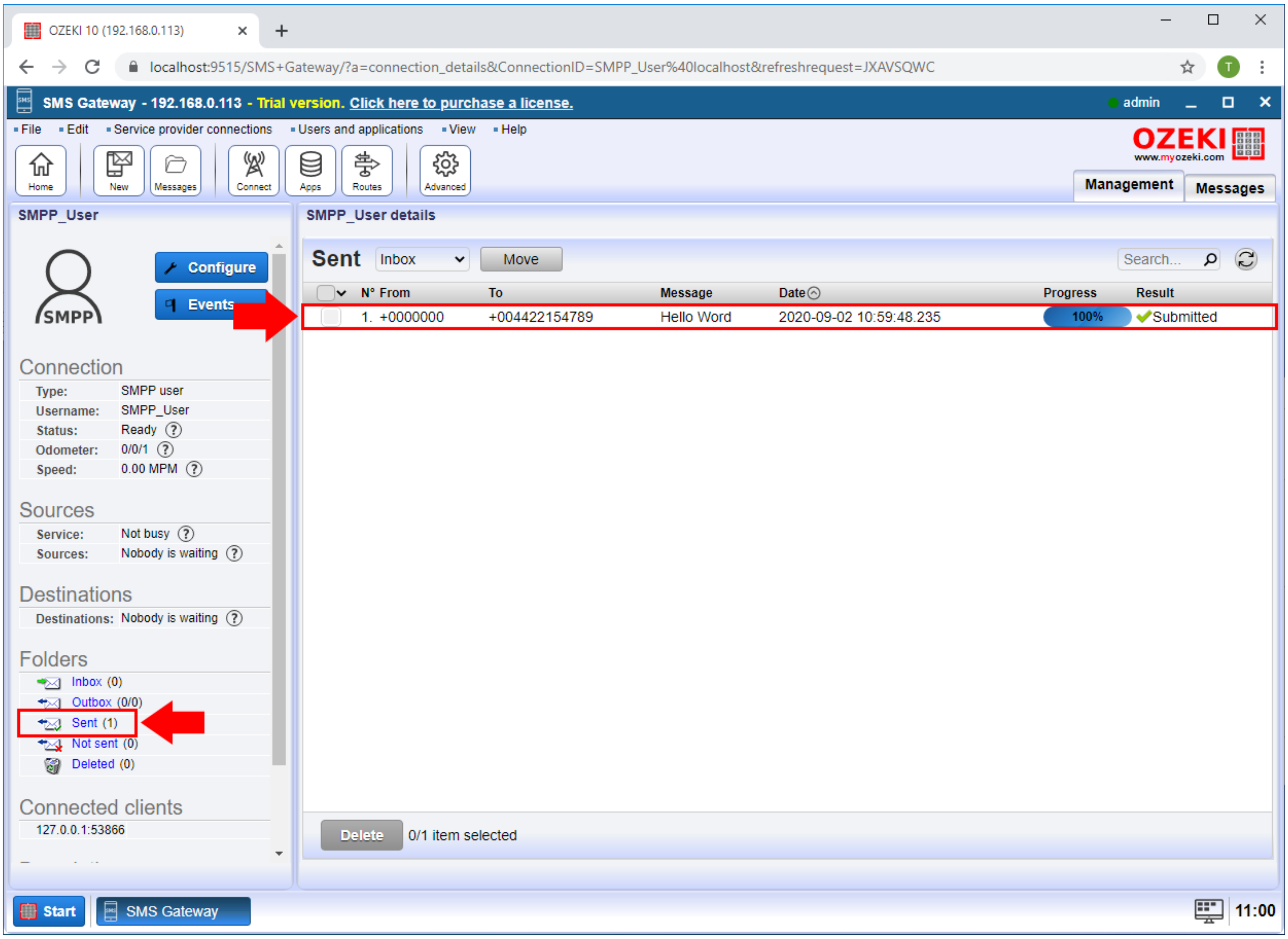
☐ Log messaging events

You may wish to trace each message individually. Enable the following checkbox if you wish to add message history tracking to each message.

☒ Log message details

Figure 2 - Enable Log Message Details

You can find the sent message in the sent folder. The Ozeki SMS Gateway stores here all the message which have been sent out via one of the connections as the Figure 3 shows.



localhost:9515/SMS+Gateway/?a=connection_details&ConnectionID=SMPP_User%40localhost&refreshrequest=JXAVSQWC

SMS Gateway - 192.168.0.113 - Trial version. Click here to purchase a license.

SMPP_User

Connection

Type: SMPP user
Username: SMPP_User
Status: Ready
Odometer: 0/0/1
Speed: 0.00 MPM

Sources

Service: Not busy
Sources: Nobody is waiting

Destinations

Destinations: Nobody is waiting

Folders

Inbox (0)
Outbox (0/0)
Sent (1)
Not sent (0)
Deleted (0)

Connected clients

127.0.0.1:53866

SMPP_User details

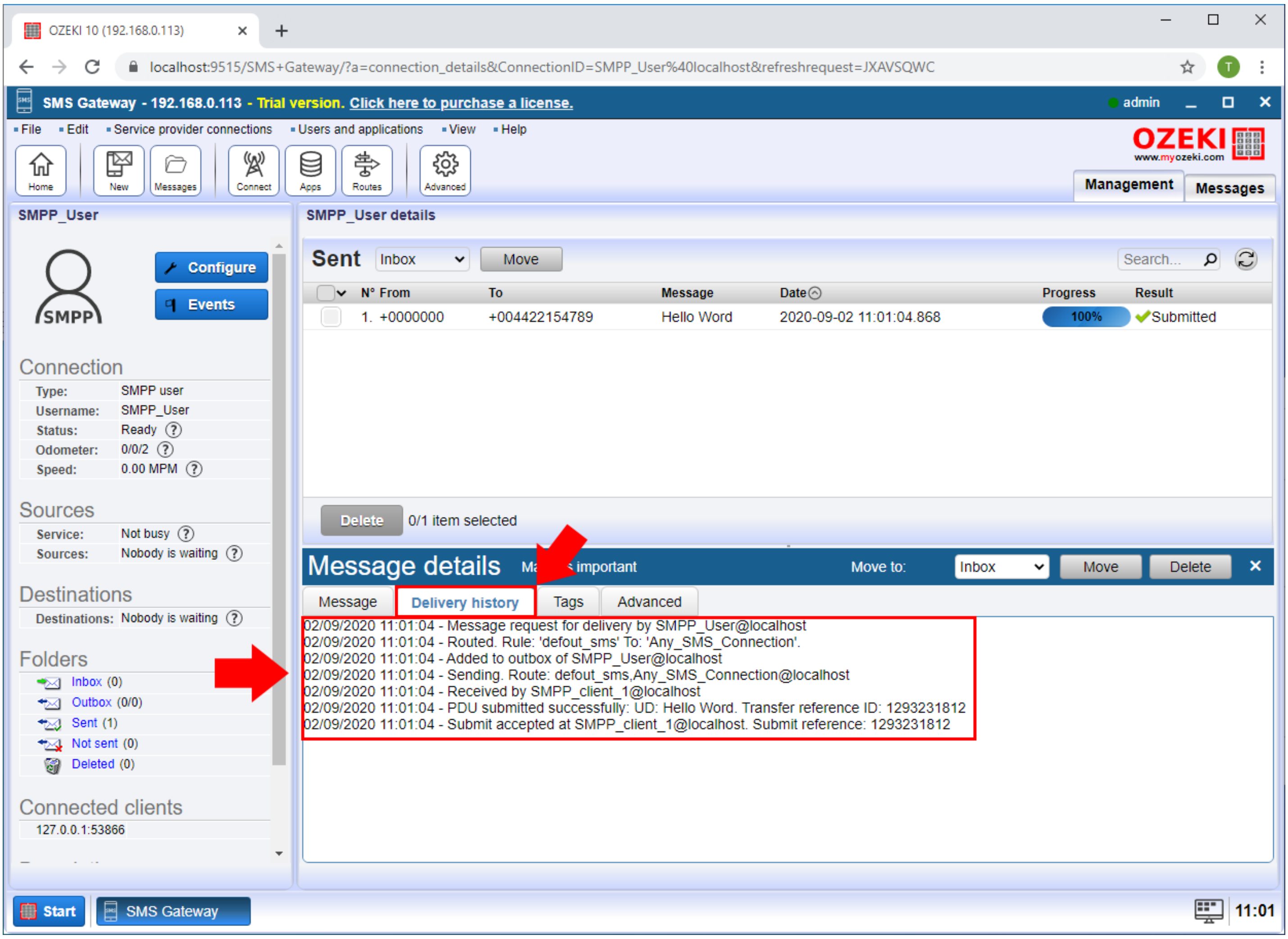
Sent

| N° | From | To | Message | Date | Progress | Result |
|---|---|---|---|---|---|---|
| 1. | +0000000 | +004422154789 | Hello Word | 2020-09-02 10:59:48.235 | 100% | ✓Submitted |

0/1 item selected

Figure 3 - Message sent

If you click on the message in the sent folder in the Delivery history tab you can see the event log for the specific message sent by the Ozeki SMS Gateway (Figure 4).



Connection

| | |
|---|---|
| Type: | SMPP user |
| Username: | SMPP_User |
| Status: | Ready |
| Odometer: | 0/0/2 |
| Speed: | 0.00 MPM |

Sources

| | |
|---|---|
| Service: | Not busy |
| Sources: | Nobody is waiting |

Destinations

| | |
|---|---|
| Destinations: | Nobody is waiting |

Folders

- Inbox (0)
- Outbox (0/0)
- Sent (1)
- Not sent (0)
- Deleted (0)

Connected clients

127.0.0.1:53866

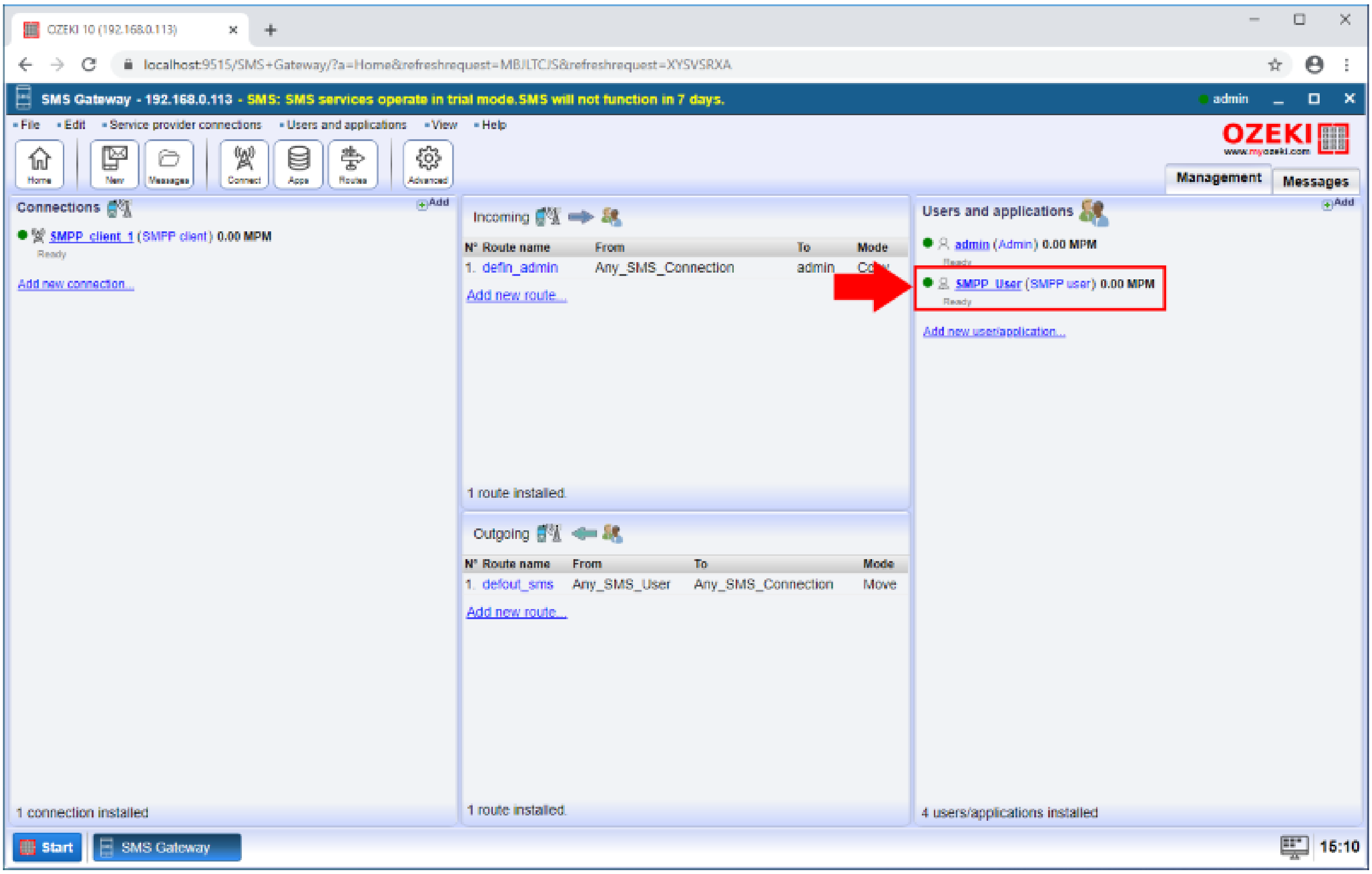
| N° | From | To | Message | Date | Progress | Result |
|---|---|---|---|---|---|---|
| 1. | +0000000 | +004422154789 | Hello Word | 2020-09-02 11:01:04.868 | 100% | ✓Submitted |

```
02/09/2020 11:01:04 - Message request for delivery by SMPP_User@localhost
02/09/2020 11:01:04 - Routed. Rule: 'defout_sms' To: 'Any_SMS_Connection'.
02/09/2020 11:01:04 - Added to outbox of SMPP_User@localhost
02/09/2020 11:01:04 - Sending. Route: defout_sms,Any_SMS_Connection@localhost
02/09/2020 11:01:04 - Received by SMPP_client_1@localhost
02/09/2020 11:01:04 - PDU submitted successfully: UD: Hello Word. Transfer reference ID: 1293231812
02/09/2020 11:01:04 - Submit accepted at SMPP_client_1@localhost. Submit reference: 1293231812
```

Figure 4 - Message Delivery history

# How to log message evengs

The first step is to open the source user connection on the right side of the SMS Gateway Manager console page as you can see it on the Figure 1.

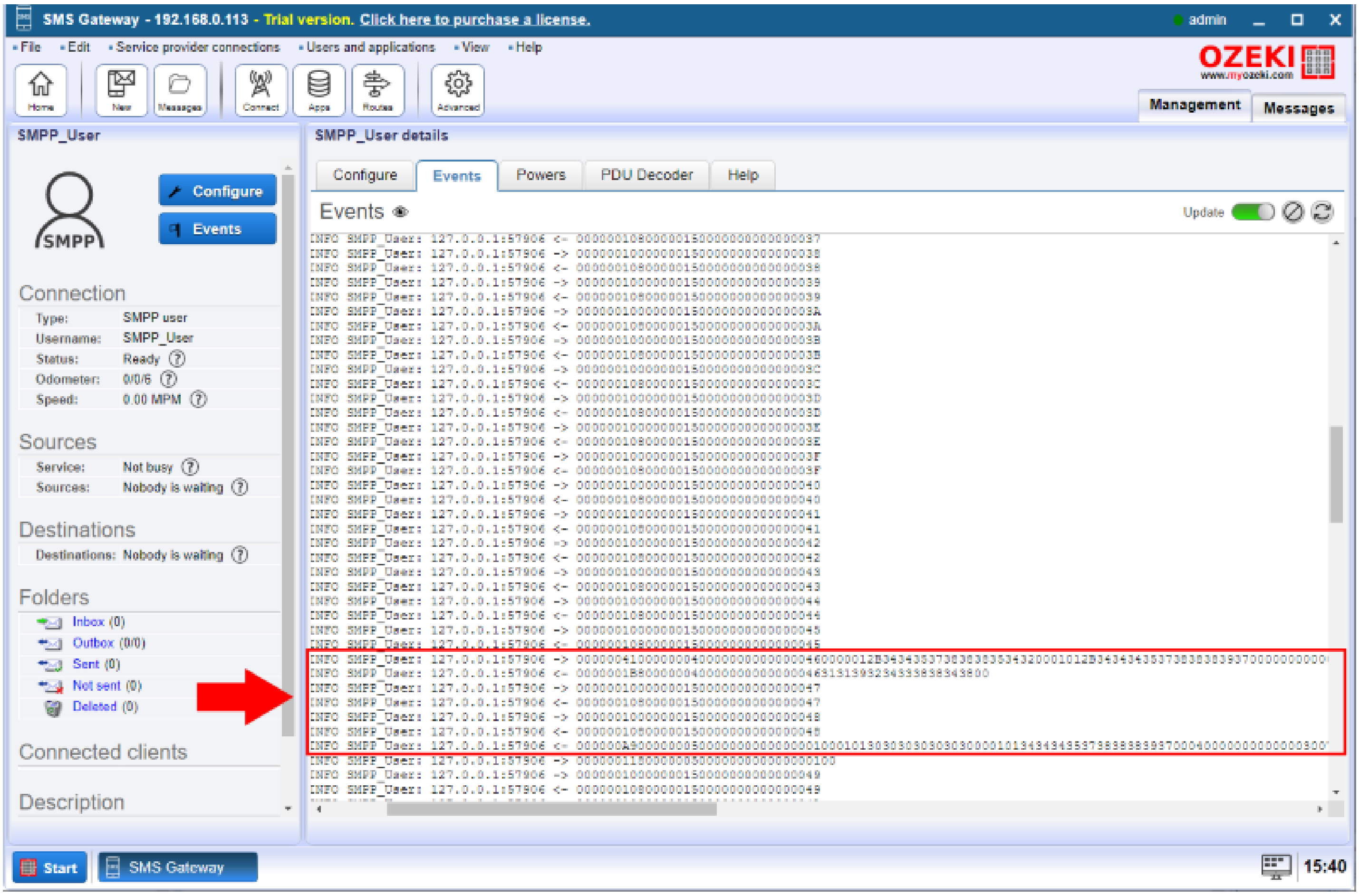


| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defin_admin | Any_SMS_Connection | admin | Copy |

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defaut_sms | Any_SMS_User | Any_SMS_Connection | Move |

Figure 1 - Open source user connection

By default, when a message arrives, only protocol communication is displayed in the event menu as the Figure 2 shows. To see more detailed information about the message, turn on the Log messaging events option.



Connection

Type: SMPP user
Username: SMPP_User
Status: Ready
Odometer: 0/0/6
Speed: 0.00 MPM

Sources

Service: Not busy
Sources: Nobody is waiting

Destinations

Destinations: Nobody is waiting

Folders

Inbox (0)
Outbox (0/0)
Sent (0)
Not sent (0)
Deleted (0)

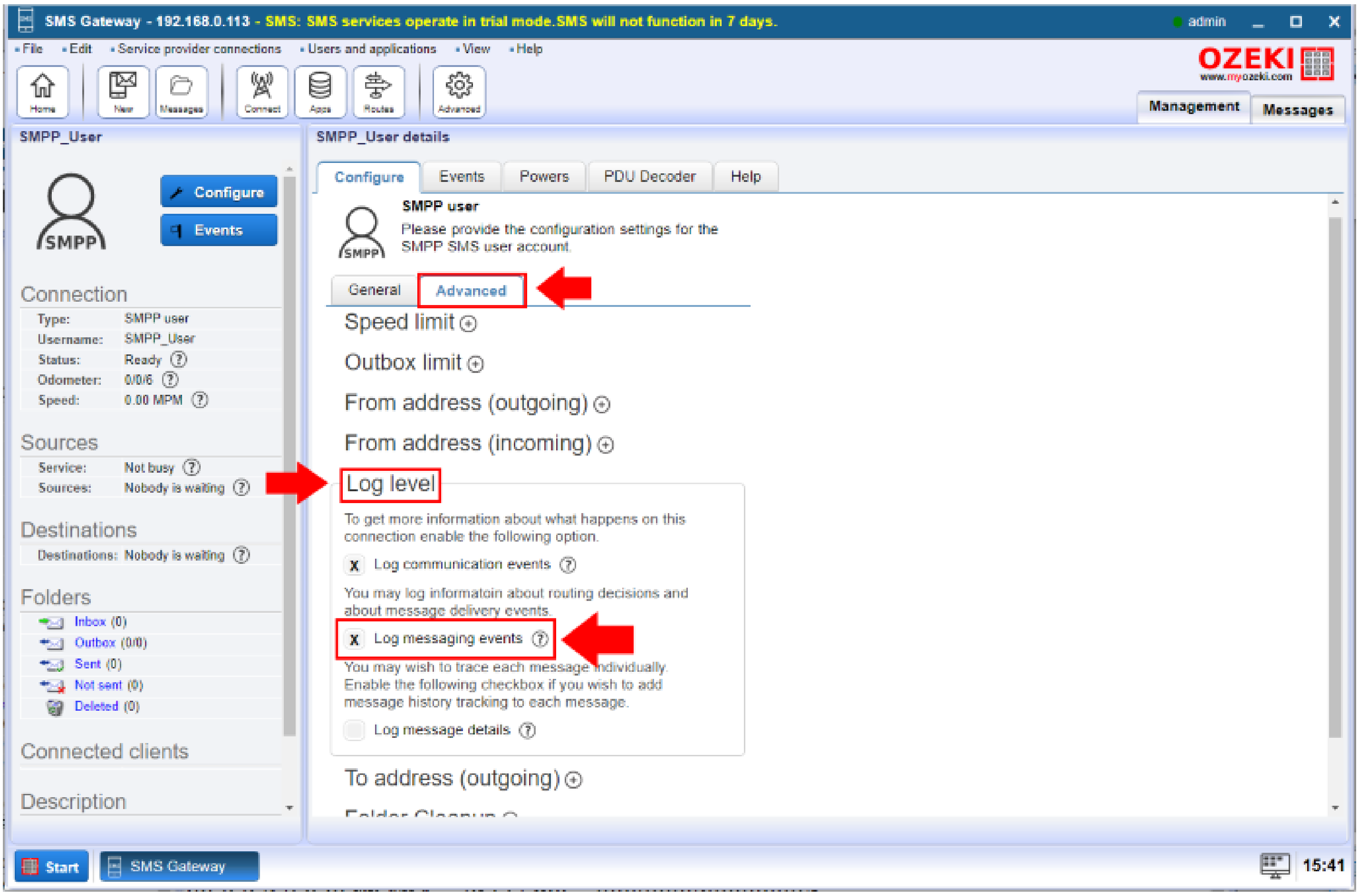
Connected clients

Description

```
INFO SMPP_User: 127.0.0.1:57906 <- 00000010800000150000000000000037
INFO SMPP_User: 127.0.0.1:57906 -> 00000010000000150000000000000038
INFO SMPP_User: 127.0.0.1:57906 <- 00000010800000150000000000000038
INFO SMPP_User: 127.0.0.1:57906 -> 00000010000000150000000000000039
INFO SMPP_User: 127.0.0.1:57906 <- 00000010800000150000000000000039
INFO SMPP_User: 127.0.0.1:57906 -> 0000001000000015000000000000003A
INFO SMPP_User: 127.0.0.1:57906 <- 0000001080000015000000000000003A
INFO SMPP_User: 127.0.0.1:57906 -> 0000001000000015000000000000003B
INFO SMPP_User: 127.0.0.1:57906 <- 0000001080000015000000000000003B
INFO SMPP_User: 127.0.0.1:57906 -> 0000001000000015000000000000003C
INFO SMPP_User: 127.0.0.1:57906 <- 0000001080000015000000000000003C
INFO SMPP_User: 127.0.0.1:57906 -> 0000001000000015000000000000003D
INFO SMPP_User: 127.0.0.1:57906 <- 0000001080000015000000000000003D
INFO SMPP_User: 127.0.0.1:57906 -> 0000001000000015000000000000003E
INFO SMPP_User: 127.0.0.1:57906 <- 0000001080000015000000000000003E
INFO SMPP_User: 127.0.0.1:57906 -> 0000001000000015000000000000003F
INFO SMPP_User: 127.0.0.1:57906 <- 0000001080000015000000000000003F
INFO SMPP_User: 127.0.0.1:57906 -> 00000010000000150000000000000040
INFO SMPP_User: 127.0.0.1:57906 <- 00000010800000150000000000000040
INFO SMPP_User: 127.0.0.1:57906 -> 00000010000000150000000000000041
INFO SMPP_User: 127.0.0.1:57906 <- 00000010800000150000000000000041
INFO SMPP_User: 127.0.0.1:57906 -> 00000010000000150000000000000042
INFO SMPP_User: 127.0.0.1:57906 <- 00000010800000150000000000000042
INFO SMPP_User: 127.0.0.1:57906 -> 00000010000000150000000000000043
INFO SMPP_User: 127.0.0.1:57906 <- 00000010800000150000000000000043
INFO SMPP_User: 127.0.0.1:57906 -> 00000010000000150000000000000044
INFO SMPP_User: 127.0.0.1:57906 <- 00000010800000150000000000000044
INFO SMPP_User: 127.0.0.1:57906 -> 00000010000000150000000000000045
INFO SMPP_User: 127.0.0.1:57906 <- 00000010800000150000000000000045
INFO SMPP_User: 127.0.0.1:57906 -> 000000410000000400000000000000460000012B343435373838383534320001012B34343435373838383937000000000000
INFO SMPP_User: 127.0.0.1:57906 <- 0000001B8000000400000000000000463131393234333838343800
INFO SMPP_User: 127.0.0.1:57906 -> 00000010000000150000000000000047
INFO SMPP_User: 127.0.0.1:57906 <- 00000010800000150000000000000047
INFO SMPP_User: 127.0.0.1:57906 -> 00000010000000150000000000000048
INFO SMPP_User: 127.0.0.1:57906 <- 00000010800000150000000000000048
INFO SMPP_User: 127.0.0.1:57906 <- 000000A90000000500000000000000010001013030303030303000010134343435373838383937000400000000000000000300
INFO SMPP_User: 127.0.0.1:57906 -> 0000001180000005000000000000000100
INFO SMPP_User: 127.0.0.1:57906 -> 00000010000000150000000000000049
INFO SMPP_User: 127.0.0.1:57906 <- 00000010800000150000000000000049
```

Figure 2 - Default message events

Now in the Log level section of the user's Advanced tab enable the 'Log messaging events' checkbox if you wish to add log informatoin about routing decisions and about message delivery events (Figure 3).



Connection

Type: SMPP user
Username: SMPP_User
Status: Ready
Odometer: 0/0/6
Speed: 0.00 MPM

Sources

Service: Not busy
Sources: Nobody is waiting

Destinations

Destinations: Nobody is waiting

Folders

Inbox (0)
Outbox (0/0)
Sent (0)
Not sent (0)
Deleted (0)

Connected clients

Description

SMPP user

Please provide the configuration settings for the SMPP SMS user account.

Speed limit ⊕

Outbox limit ⊕

From address (outgoing) ⊕

From address (incoming) ⊕

Log level

To get more information about what happens on this connection enable the following option.

☒ Log communication events

You may log informatoin about routing decisions and about message delivery events.

☒ Log messaging events

You may wish to trace each message individually. Enable the following checkbox if you wish to add message history tracking to each message.

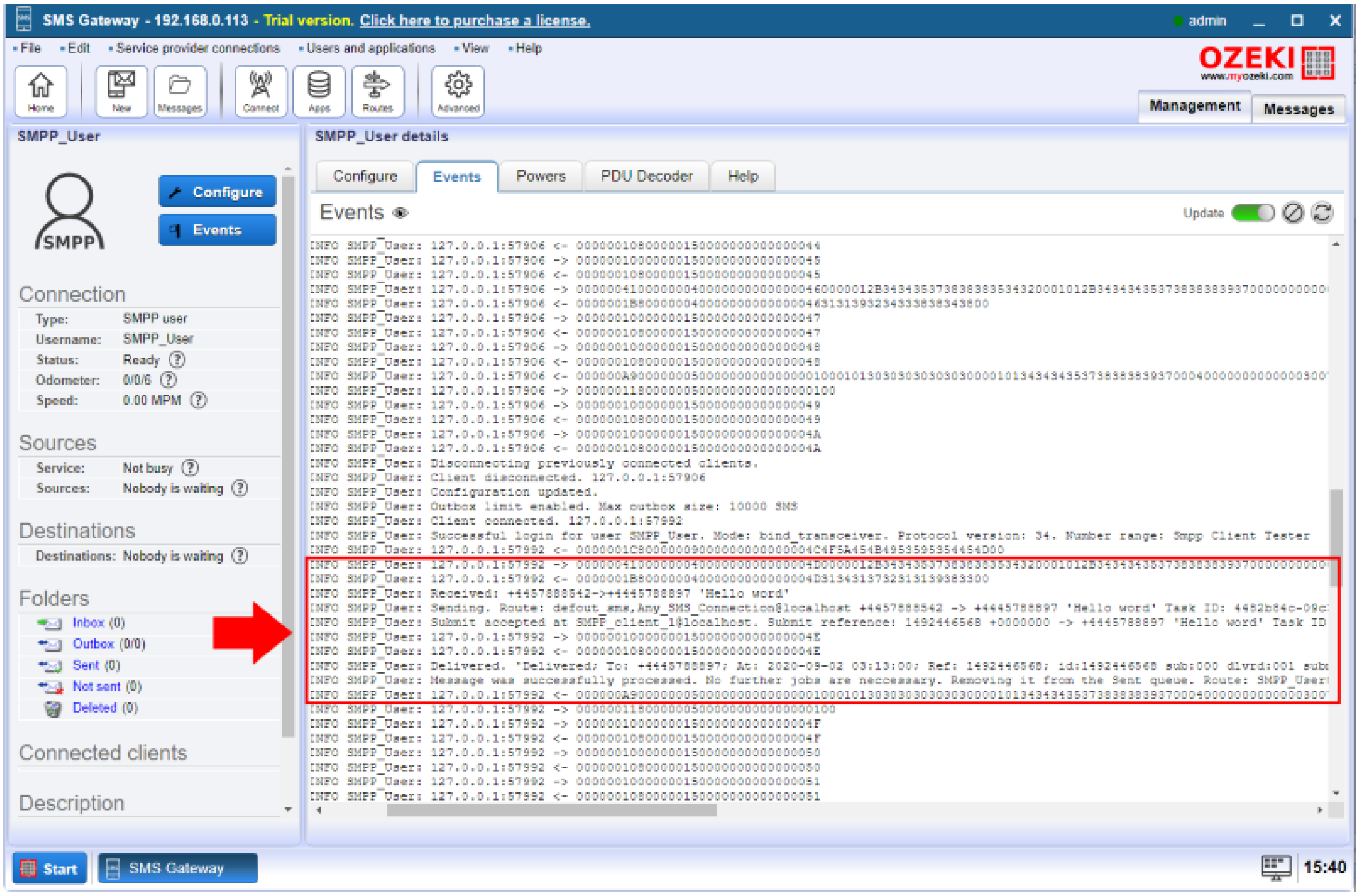
□ Log message details

To address (outgoing) ⊕

Folder Cleanup ⊕

Figure 3 - Enable Log messaging events

Once you have enabled the Log messaging events option, you will see informatoin about routing decisions and about message delivery events in the Events menu when you receive an SMS.



Connection

Type: SMPP user
Username: SMPP_User
Status: Ready
Odometer: 0/0/6
Speed: 0.00 MPM

Sources

Service: Not busy
Sources: Nobody is waiting

Destinations

Destinations: Nobody is waiting

Folders

Inbox (0)
Outbox (0/0)
Sent (0)
Not sent (0)
Deleted (0)

Connected clients

Description

```
INFO SMPP_User: 127.0.0.1:57906 <- 00000010800000150000000000000044
INFO SMPP_User: 127.0.0.1:57906 -> 00000010000000150000000000000045
INFO SMPP_User: 127.0.0.1:57906 <- 00000010800000150000000000000045
INFO SMPP_User: 127.0.0.1:57906 -> 00000041000000040000000000000046000000012B343435373838383534320001012B3434343537383838393700000000000
INFO SMPP_User: 127.0.0.1:57906 <- 0000001B8000000400000000000000046313139323433383638343800
INFO SMPP_User: 127.0.0.1:57906 -> 00000010000000150000000000000047
INFO SMPP_User: 127.0.0.1:57906 <- 00000010800000150000000000000047
INFO SMPP_User: 127.0.0.1:57906 -> 00000010000000150000000000000048
INFO SMPP_User: 127.0.0.1:57906 <- 00000010800000150000000000000048
INFO SMPP_User: 127.0.0.1:57906 <- 000000A900000005000000000000000010001013030303030303000010134343435373838383937000400000000000000300
INFO SMPP_User: 127.0.0.1:57906 -> 000000118000000500000000000000000100
INFO SMPP_User: 127.0.0.1:57906 -> 00000010000000150000000000000049
INFO SMPP_User: 127.0.0.1:57906 <- 00000010800000150000000000000049
INFO SMPP_User: 127.0.0.1:57906 -> 0000001000000015000000000000004A
INFO SMPP_User: 127.0.0.1:57906 <- 0000001080000015000000000000004A
INFO SMPP_User: Disconnecting previously connected clients.
INFO SMPP_User: Client disconnected. 127.0.0.1:57906
INFO SMPP_User: Configuration updated.
INFO SMPP_User: Outbox limit enabled. Max outbox size: 10000 SMS
INFO SMPP_User: Client connected. 127.0.0.1:57992
INFO SMPP_User: Successful login for user SMPP_User. Mode: bind_transceiver. Protocol version: 34. Number range: Smpp Client Tester
INFO SMPP_User: 127.0.0.1:57992 <- 0000001C8000000900000000000000004C4F5A454B4953595354454D00
INFO SMPP_User: 127.0.0.1:57992 -> 0000004100000004000000000000004D000000012B343435373838383534320001012B3434343537383838393700000000000
INFO SMPP_User: 127.0.0.1:57992 <- 0000001B8000000400000000000000004D3134313732313139383300
INFO SMPP_User: Received: +4457888542->+4445788897 'Hello word'
INFO SMPP_User: Sending. Route: defout_sms,Any_SMS_Connection@localhost +4457888542 -> +4445788897 'Hello word' Task ID: 4482b84c-09c
INFO SMPP_User: Submit accepted at SMPP_client_1@localhost. Submit reference: 1492446568 +0000000 -> +4445788897 'Hello word' Task ID
INFO SMPP_User: 127.0.0.1:57992 -> 0000001000000015000000000000004E
INFO SMPP_User: 127.0.0.1:57992 <- 0000001080000015000000000000004E
INFO SMPP_User: Delivered. 'Delivered; To: +4445788897; At: 2020-09-02 03:13:00; Ref: 1492446568; id:1492446568 sub:000 dlvrd:001 sub
INFO SMPP_User: Message was successfully processed. No further jobs are neccessary. Removing it from the Sent queue. Route: SMPP_User
INFO SMPP_User: 127.0.0.1:57992 <- 000000A900000005000000000000000010001013030303030303000010134343435373838383937000400000000000000300
INFO SMPP_User: 127.0.0.1:57992 -> 000000118000000500000000000000000100
INFO SMPP_User: 127.0.0.1:57992 -> 0000001000000015000000000000004F
INFO SMPP_User: 127.0.0.1:57992 <- 0000001080000015000000000000004F
INFO SMPP_User: 127.0.0.1:57992 -> 00000010000000150000000000000050
INFO SMPP_User: 127.0.0.1:57992 <- 00000010800000150000000000000050
INFO SMPP_User: 127.0.0.1:57992 -> 00000010000000150000000000000051
INFO SMPP_User: 127.0.0.1:57992 <- 00000010800000150000000000000051
```

Figure 4 - Detailed message events

# Firewall configuration of Ozeki 10

***Ozeki 10 uses different ports and protocols. It provides various services, so the ports in use may differ from system to system. The following table lists the most common ports used by Ozeki to help you with firewall setup. Firewall setup is needed if you wish to access the Ozeki GUI or some services remotely.***

## Graphical user interface (HTTPS)

Firewall information: Ports 9515 and 9516 should be enabled on your firewall if you want to access the Ozeki GUI remotely. If you enable port 9515, but you don't enable port 9516, the GUI response times and GUI updates will be significantly slower.

**Service: GUI (Graphical User Interface)**
Port: 9515
Firewall: TCP
Protocol: HTTPS
Description: This port is used to access the graphical user interface in a webbrowser. You may change the HTTPS port of Ozeki in the control panel.

**Service: GUI websocket (Graphical User Interface Websocket**
Port: 9516
Firewall: TCP
Protocol: WSS
Description: This port is used to speed up screen updates for the graphical user interface in a webbrowser. This port is always the HTTPS port +1. When you change the HTTPS port of Ozeki, this port is automatically assigned.

## Graphical user interface (HTTP)

Firewall information: Ports 9513 and 9514 should not be enabled on your firewall by default. These ports are open for compatibility reasons to provided access to older SMS clients.

**Service: GUI (Graphical User Interface)**
Port: 9513
Firewall: TCP
Protocol: HTTS
Description: This port is used to access the graphical user interface in a webbrowser. You may change the HTTPS port of Ozeki in the control panel.

**Service: GUI websocket (Graphical User Interface Websocket**
Port: 9514
Firewall: TCP
Protocol: WS
Description: This port is used to speed up screen updates for the graphical user interface in a webbrowser. This port is always the HTTPS port +1. When you change the HTTPS port of Ozeki, this port is automatically assigned.

## HTTP API (HTTPS)

Firewall information: Ports 9508 should not be enabled on your firewall by default. These ports are open for compatibility reasons to provided access to older SMS clients.

**Service: HTTP API**
Port: 9508
Firewall: TCP
Protocol: HTTPS
Description: This port is used to access the HTTP API service. You may change the HTTPS port of Ozeki in the control panel.

# SMS services

Firewall information: Ports 9550, 9560 and 9570 should not be enabled on your firewall by default. These ports are required only if you wish to provide SMS service to SMS clients.

**Service: SMPP (Short Message Peer to Peer)**
Port: 9550
Firewall: TCP
Protocol: SMPP
Description: This port is used to provide SMPP (Short Message Peer to Peer) service. You may change the SMPP port of Ozeki in the SMS Gateway Advanced menu.

**Service: CIMD2 (Computer Interface to Message Distribution)**
Port: 9560
Firewall: TCP
Protocol: CIMD2
Description: This port is used to provide the CIMD2 (Computer Interface to Message Distribution) service. You may change the CIMD2 port of Ozeki in the SMS Gateway Advanced menu.

**Service: UCP (Universal Computer Protocol)**
Port: 9570
Firewall: TCP
Protocol: UCP
Description: This port is used to provide the UCP (Universal Computer Protocol) service. You may change the UCP port of Ozeki in SMS Gateway Advanced menu.

# PBX service

Firewall information: Ports 5060 and 5000-10000 should not be enabled on your firewall by default. These ports are only needed if you with provide VoIP phone system services.

**Service: SIP (Session Initiation Protocol)**
Port: 5060
Firewall: TCP
Protocol: SIP
Description: This port is used if you provide VoIP phone system service. This port is used for call setup according to the SIP 2.0 protocol, when a “session” needs to be created between 2 or more participants. You may change the SIP port of Ozeki in the Phone System Advanced menu.

**Service: RTP (Real-time Transport Protocol)**
Port: 5000-10000
Firewall: UTP
Protocol: RTP
Description: This port is used for delivering audio and video over IP networks. It is used if the call audio/video is routed to or through the phone system. You may change the RTP port of Ozeki in the Phone System Advanced menu.

# Chat service

Firewall information: Ports 9507 should not be enabled on your firewall by default. These ports are only needed if you wish to provide Ozeki Chat connection setup service.

**Service: Chat**
Port: 9507
Firewall: TCP
Protocol: Chat
Description: This port is used by the Ozeki Chat System. It is used for sending and receiving chat messages between the server and the chat clients. You may change the this port of Ozeki in the Chat server Advanced menu.

# How to setup your SSL cert in Ozeki 10

***The following document provides some useful information about how you can setup your SSL certificate in Ozeki 10. By setting the certificate, you will be able to open your Ozeki 10 safely from anywhere. The document contains a step by step guide that describes every action that you have to perform to successfully set up the solution. Each step is visually demonstrated by a screenshot for the better understanding. It does not take more than ten minutes to complete the guide, so let's start right now!***

## Step 1 - Open Security app

Since Ozeki uses 10 PFX format certificates, the first step is to convert the previously generated certificate to PFX. But don't worry in Ozeki 10 this can be easily done. Just open the Security app under the Start menu, Porgrams, Adminstrative Tools as Figure 1 demonstrates it below.

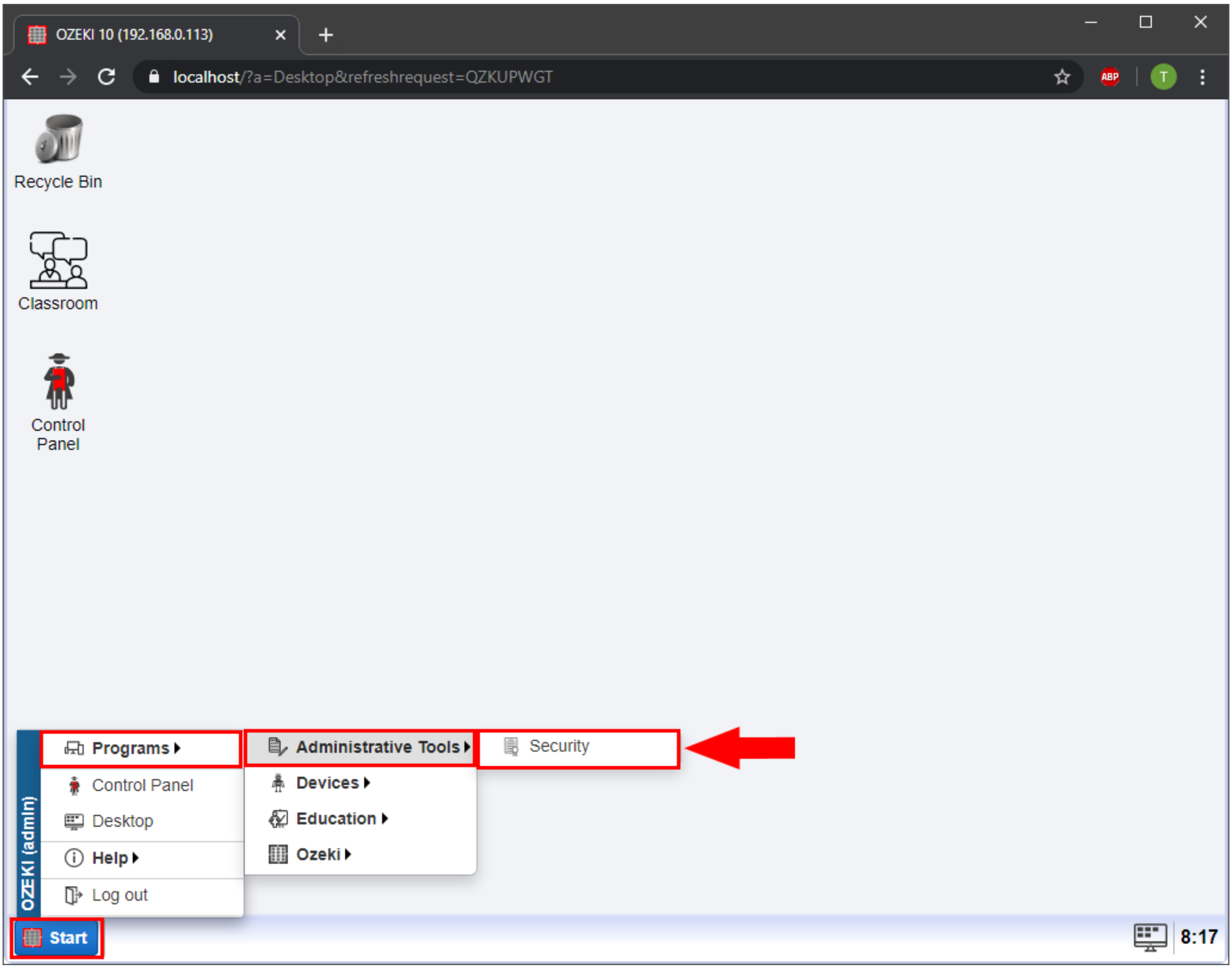
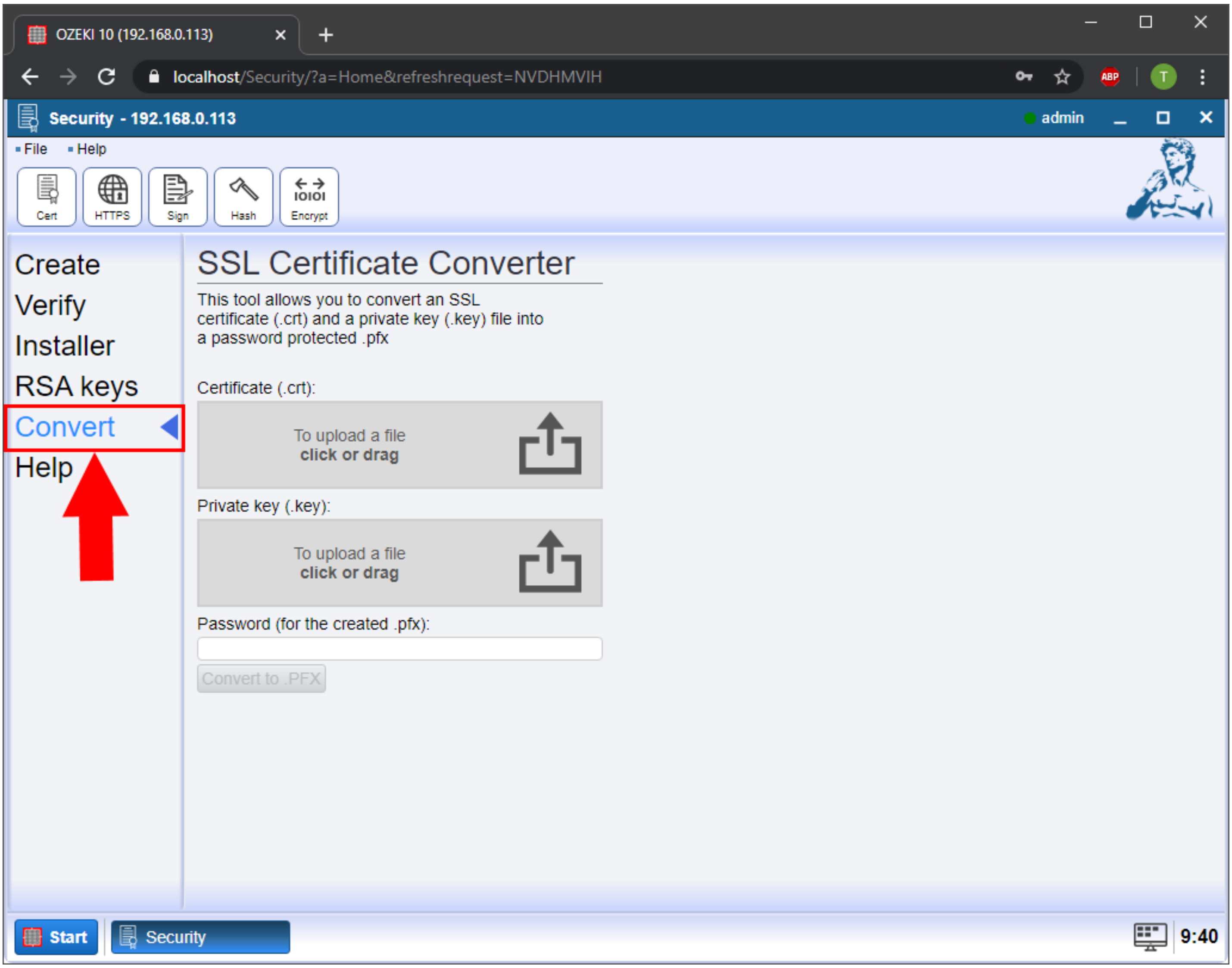


Figure 1 - Security app

## Step 2 - Select Convert tab

Next, the main menu of the Security application shows up. This application ensures that you can create and manage certifications or sign them as well. At this point, you need to select the Convert submenu, so as you can see it in Figure 2, just click on the name of this menu.



SSL Certificate Converter

This tool allows you to convert an SSL certificate (.crt) and a private key (.key) file into a password protected .pfx

Certificate (.crt):

To upload a file
click or drag

Private key (.key):

To upload a file
click or drag

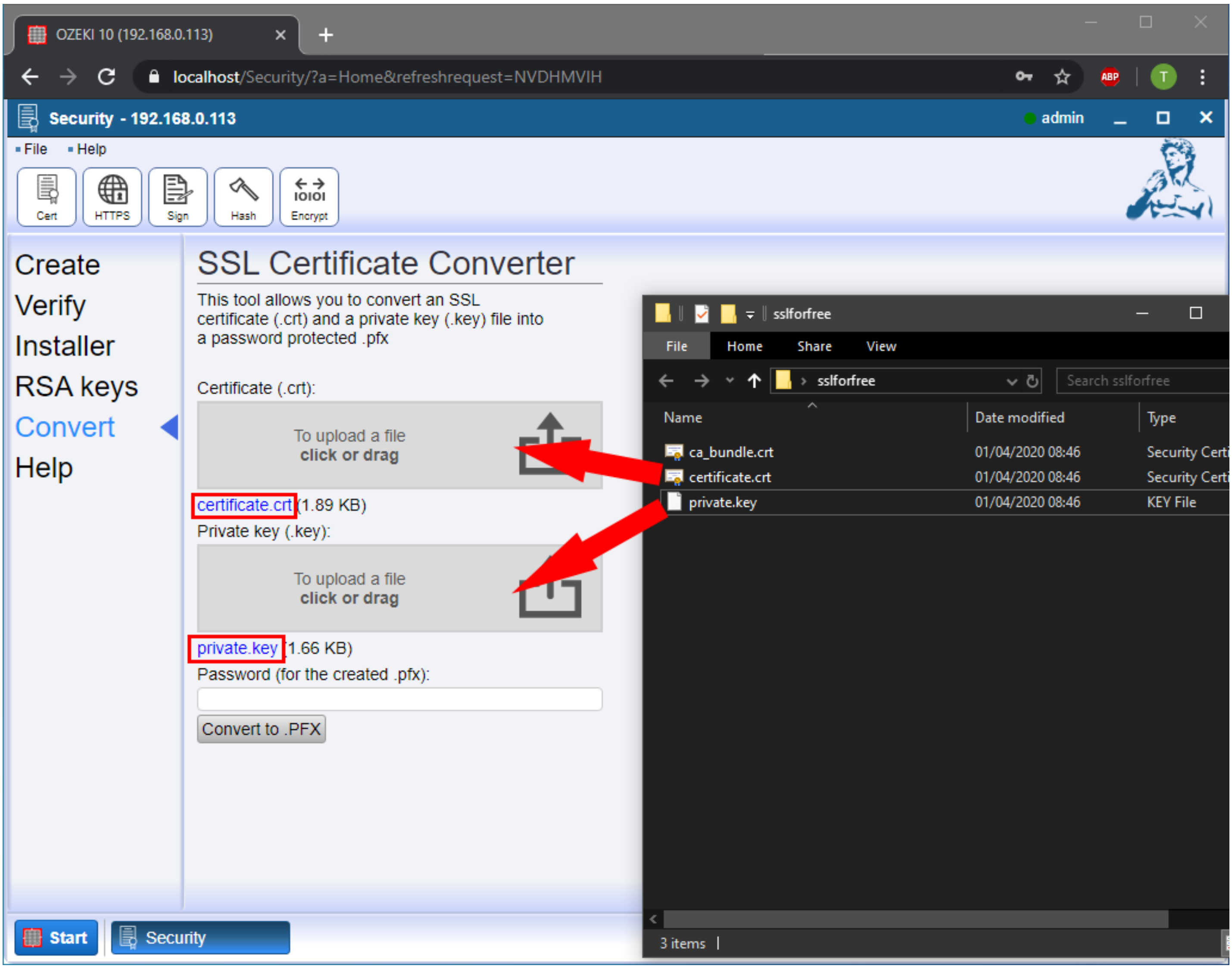
Password (for the created .pfx):

Convert to .PFX

Figure 2 - Convert tab

## Step 3 - Upload certificate files

This Convert menu is the place where you can convert your certificate to a password protected .pfx file. This operation is needed to use your certificate in Ozeki 10. The conversion of the files is quite simple, first, you need to drag and drop the certificate and the private key files as Figure 3 shows that.



SSL Certificate Converter

This tool allows you to convert an SSL certificate (.crt) and a private key (.key) file into a password protected .pfx

Certificate (.crt):

To upload a file
click or drag

certificate.crt (1.89 KB)

Private key (.key):

To upload a file
click or drag

private.key (1.66 KB)

Password (for the created .pfx):

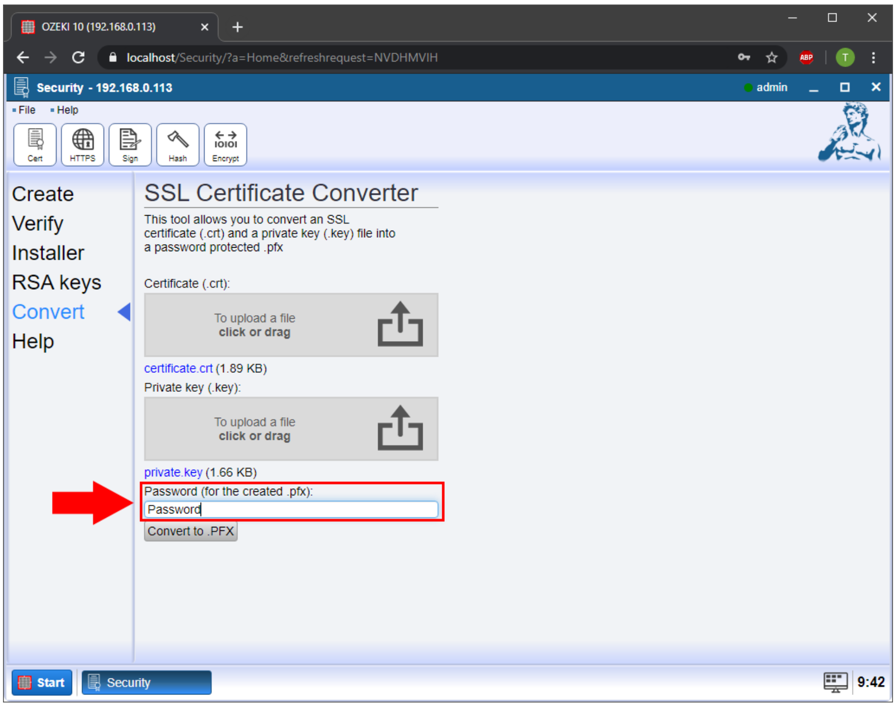
Convert to .PFX

| Name | Date modified | Type |
|---|---|---|
| ca_bundle.crt | 01/04/2020 08:46 | Security Certi |
| certificate.crt | 01/04/2020 08:46 | Security Certi |
| private.key | 01/04/2020 08:46 | KEY File |

Figure 3 - Upload certificates

## Step 4 - Add Password

Next, you need to provide a password for the certificate. This password will protect the private key that you uploaded with the certificate as well. After that, you will be able to use the converted .pfx file only in case you give the password as well. So, just enter a password as Figure 4 demonstrates that.



SSL Certificate Converter

This tool allows you to convert an SSL certificate (.crt) and a private key (.key) file into a password protected .pfx

Certificate (.crt):

certificate.crt (1.89 KB)

Private key (.key):

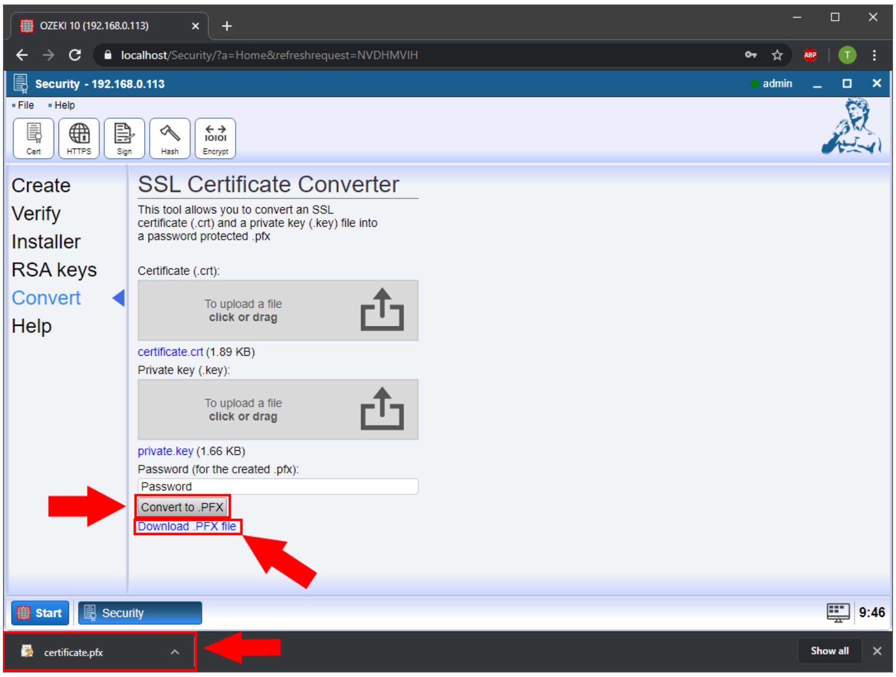
private.key (1.66 KB)

Password (for the created .pfx): Password

Figure 4 - Add Password

# Step 5 - Convert and download the PFX file

After you uploaded your certificate and private key files and then, you provided a password as well, now, you can simply convert them to a .pfx file. For that, you just need to click on the 'Convert to .PFX' button like in Figure 5. As soon as the conversion finishes, you will be able to see a new button below, which is the 'Download .PFX file' button. This button makes it possible to download the converted file to your computer. So, just go ahead, and click on this download button.



SSL Certificate Converter

This tool allows you to convert an SSL certificate (.crt) and a private key (.key) file into a password protected .pfx

Certificate (.crt):

certificate.crt (1.89 KB)

Private key (.key):

private.key (1.66 KB)

Password (for the created .pfx): Password

Figure 5 - Convert to PFX

# Step 6 - Open Control Panel

The next main operation of this guide is to activate the converted .pfx certificate in Ozeki 10. To do that, first, you need to open the Control Panel application in Ozeki 10. So, just navigate back to the desktop of Ozeki 10, and here, as you can see it in Figure 6, just open Control Panel by clicking on its icon.

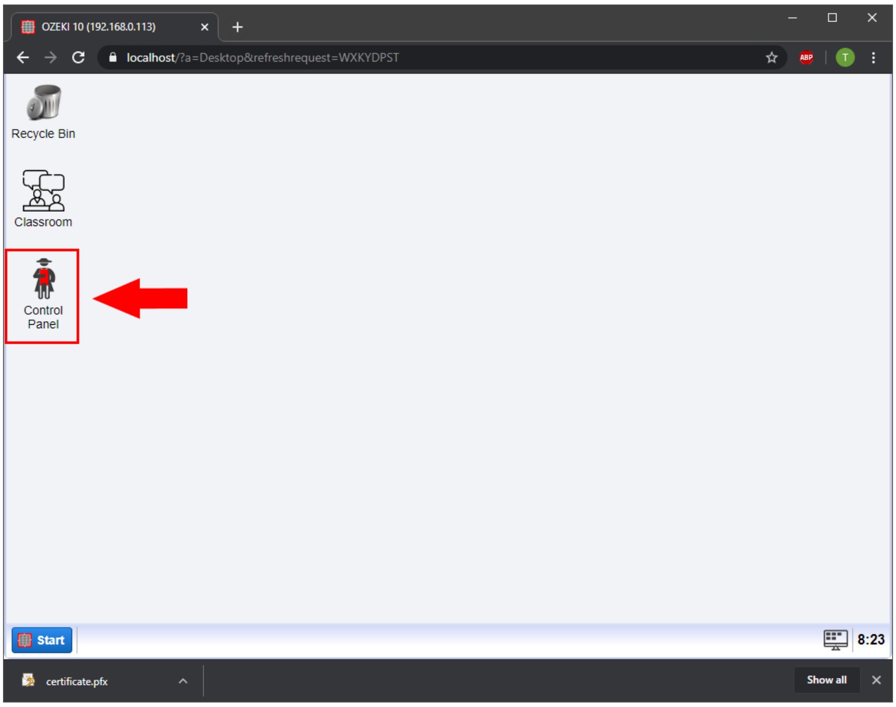
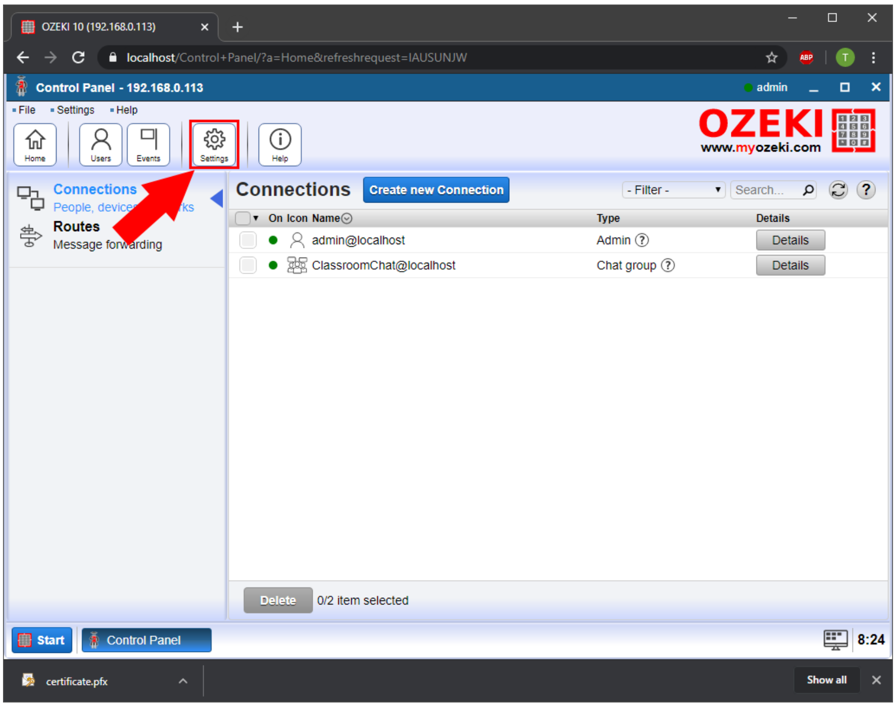


Figure 6 - Control Panel

# Step 7 - Select the Setting menu in Control Panel

The Control Panel is the application, where you can configure your connections and the main settings of your Ozeki 10. At this point, to activate the converted .pfx certificate, you need to go to the Settings menu. So, to do that, please click on the icon of the Settings menu on the toolbar like in Figure 7.

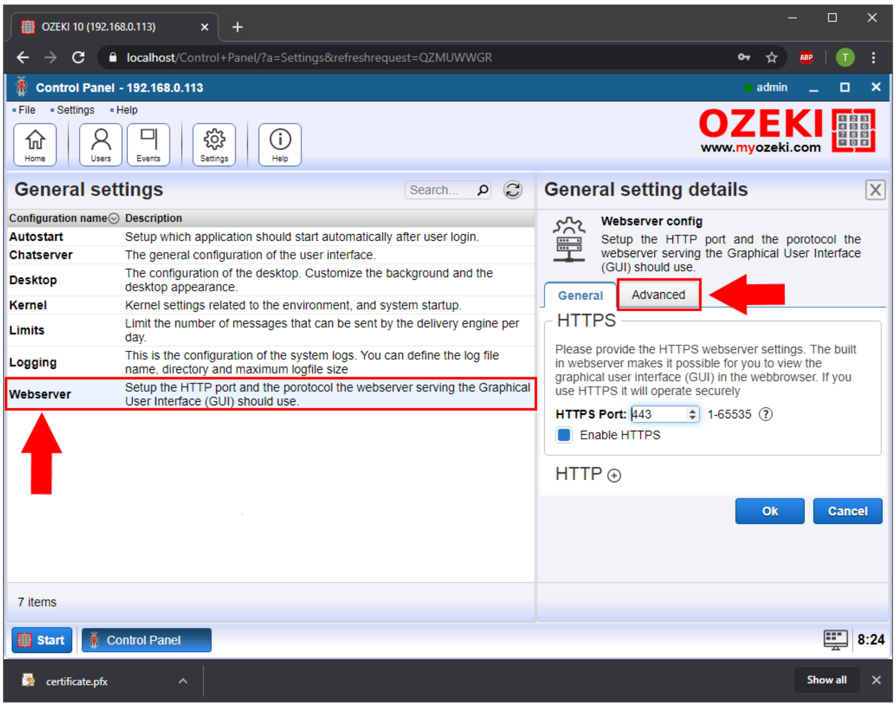


| On | Icon | Name | Type | Details |
|---|---|---|---|---|
| ● | | admin@localhost | Admin ? | Details |
| ● | | ClassroomChat@localhost | Chat group ? | Details |

Figure 7 - Select Settings menu

# Step 8 - Open Webserver Advanced menu

Here, in the Settings menu, you can see all the configuration submenus, where you can make modifications. Now, you need to select the 'Webserver' menu from that list. After you selected the 'Webserver' menu, a new windows pops up, where you can perform the configuration. To be able to upload the converted certificate, here, select the Advanced tab as Figure 8 shows that.



General settings

| Configuration name | Description |
|---|---|
| Autostart | Setup which application should start automatically after user login. |
| Chatserver | The general configuration of the user interface. |
| Desktop | The configuration of the desktop. Customize the background and the desktop appearance. |
| Kernel | Kernel settings related to the environment, and system startup. |
| Limits | Limit the number of messages that can be sent by the delivery engine per day. |
| Logging | This is the configuration of the system logs. You can define the log file name, directory and maximum logfile size |
| Webserver | Setup the HTTP port and the protocol the webserver serving the Graphical User Interface (GUI) should use. |

7 items

General setting details

Webserver config

Setup the HTTP port and the porotocol the webserver serving the Graphical User Interface (GUI) should use.

General | Advanced

HTTPS

Please provide the HTTPS webserver settings. The built in webserver makes it possible for you to view the graphical user interface (GUI) in the webbrowser. If you use HTTPS it will operate securely

HTTPS Port: 443 1-65535

☐ Enable HTTPS

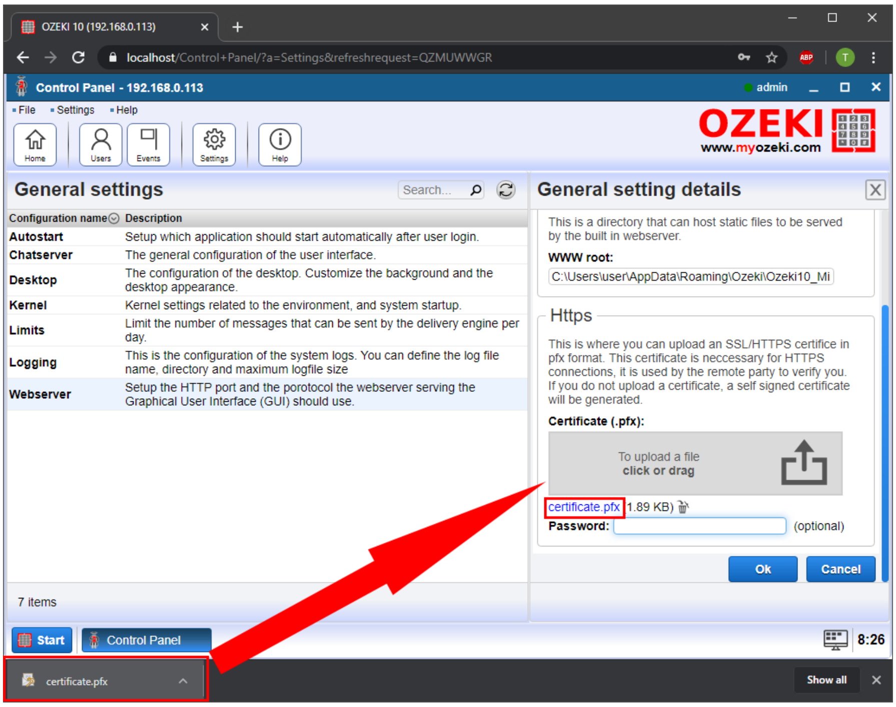
HTTP

Ok Cancel

Figure 8 - Advanced menu

# Step 9 - Upload PFX for the Webserver

In the Advanced menu, you can upload the .pfx file that you converted in the previous steps. It is quite easy to upload the certificate, since you can browse the file after clicking on the upload field or simply drag and drop the certificate file as you can see it in Figure 9.



General settings

| Configuration name | Description |
|---|---|
| Autostart | Setup which application should start automatically after user login. |
| Chatserver | The general configuration of the user interface. |
| Desktop | The configuration of the desktop. Customize the background and the desktop appearance. |
| Kernel | Kernel settings related to the environment, and system startup. |
| Limits | Limit the number of messages that can be sent by the delivery engine per day. |
| Logging | This is the configuration of the system logs. You can define the log file name, directory and maximum logfile size |
| Webserver | Setup the HTTP port and the porotocol the webserver serving the Graphical User Interface (GUI) should use. |

7 items

General setting details

This is a directory that can host static files to be served by the built in webserver.

WWW root: C:\Users\user\AppData\Roaming\Ozeki\Ozeki10_Mi

Https

This is where you can upload an SSL/HTTPS certifice in pfx format. This certificate is neccessary for HTTPS connections, it is used by the remote party to verify you. If you do not upload a certificate, a self signed certificate will be generated.

Certificate (.pfx):

To upload a file
click or drag

certificate.pfx (1.89 KB)

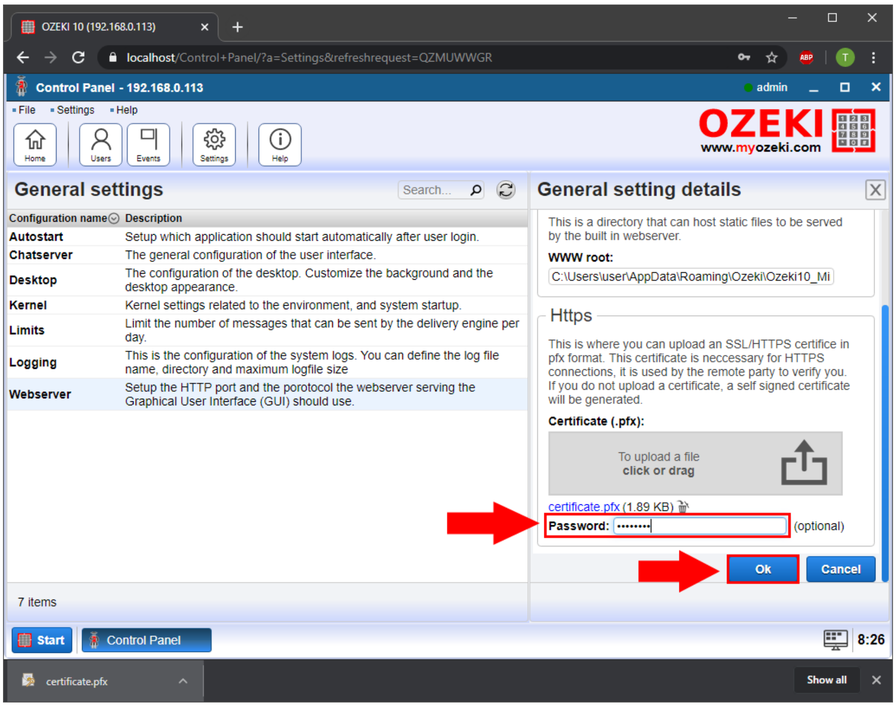
Password: (optional)

Ok Cancel

Figure 9 - Upload PFX certificate

# Step 10 - Type Password

Before finalizing the upload of the converted PFX certificate, you need to type the password of the certificate as well. This is the password that you had to provide when you converted the certificate. So, just type it in the field like in Figure 10, and finally, just click on OK.



General settings

| Configuration name | Description |
|---|---|
| Autostart | Setup which application should start automatically after user login. |
| Chatserver | The general configuration of the user interface. |
| Desktop | The configuration of the desktop. Customize the background and the desktop appearance. |
| Kernel | Kernel settings related to the environment, and system startup. |
| Limits | Limit the number of messages that can be sent by the delivery engine per day. |
| Logging | This is the configuration of the system logs. You can define the log file name, directory and maximum logfile size |
| Webserver | Setup the HTTP port and the porotocol the webserver serving the Graphical User Interface (GUI) should use. |

7 items

General setting details

This is a directory that can host static files to be served by the built in webserver.

WWW root: C:\Users\user\AppData\Roaming\Ozeki\Ozeki10_Mi

Https

This is where you can upload an SSL/HTTPS certifice in pfx format. This certificate is neccessary for HTTPS connections, it is used by the remote party to verify you. If you do not upload a certificate, a self signed certificate will be generated.

Certificate (.pfx):

To upload a file
click or drag

certificate.pfx (1.89 KB)

Password: •••••••• (optional)

Figure 10 - Type the password of the certificate

## Step 11 - Open Windows Services

At this point, you uploaded the PFX certificate to your Ozeki 10, but to be able to use it, you need to restart the Ozeki service. You can restart the service quite easily. For that, first, you need to open the Services application on your computer. To do this, type 'services' to the search field on your taskbar as Figure 11 shows that, and when the Services application appears, just click on it to start.

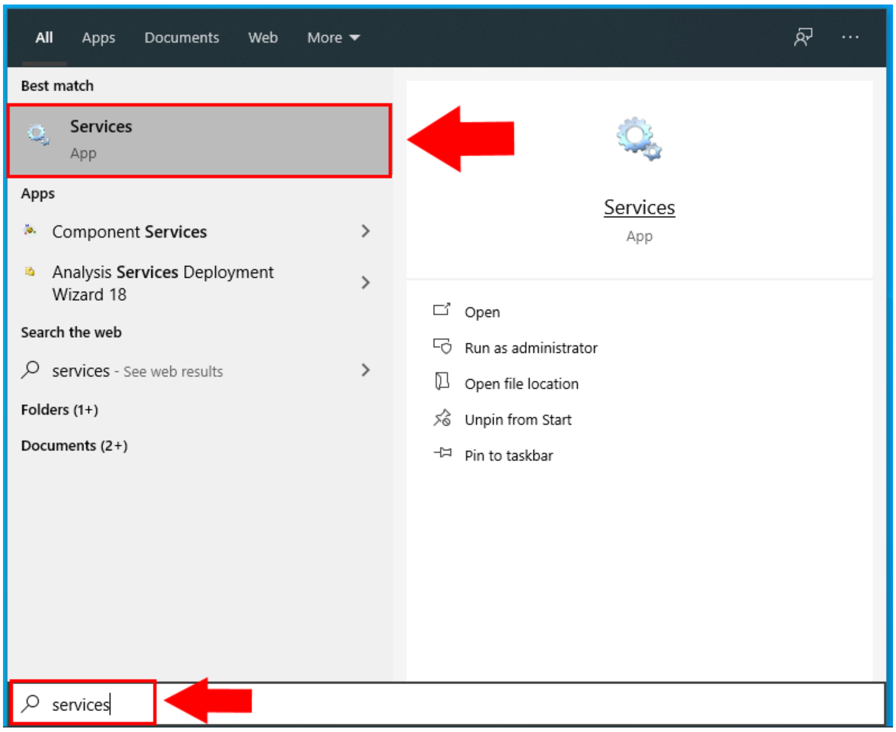
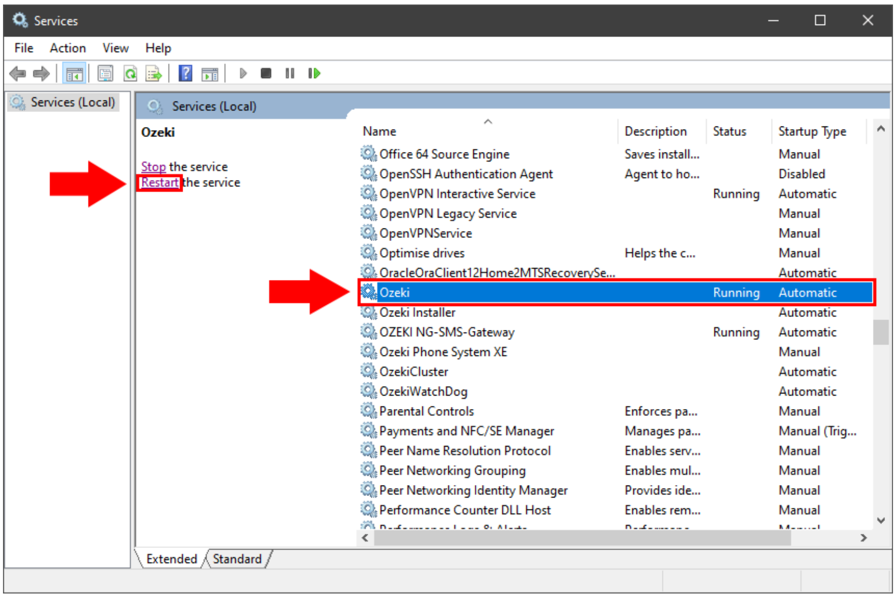


Figure 11 - Open Windows Services

## Step 12 - Restart Ozeki Service

In the Services application you can see every service that runs on your computer. Here, in that list, you need to find the Ozeki service. When you found it, select it by clicking on it like in Figure 12. Then, you need to click on the 'Restart' button to restart the service. This operation takes a few seconds, but after that, it restart with the installed certificate.

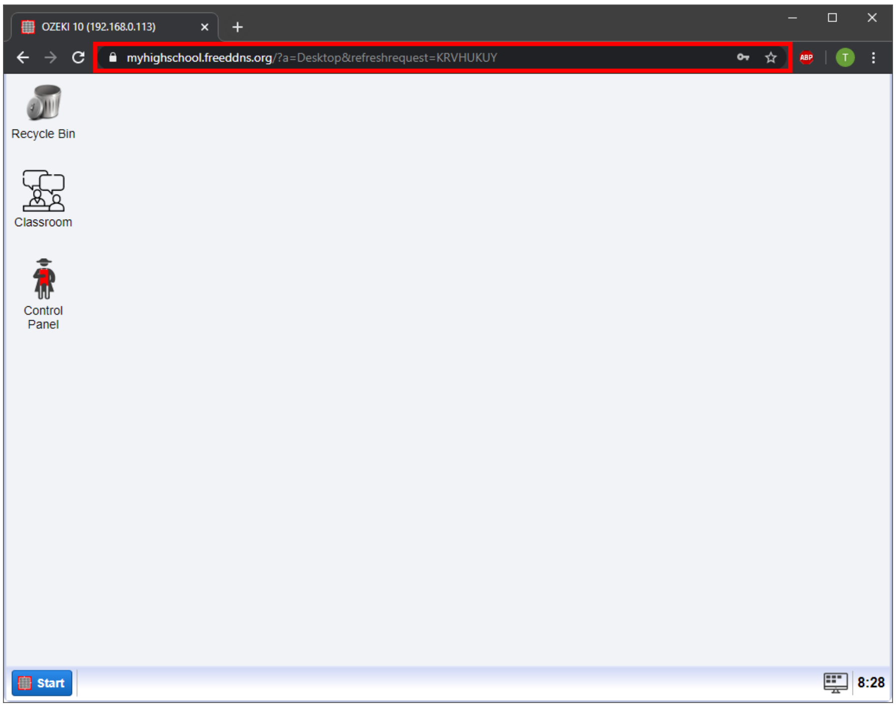


| Name | Description | Status | Startup Type |
|---|---|---|---|
| Office 64 Source Engine | Saves install... | | Manual |
| OpenSSH Authentication Agent | Agent to ho... | | Disabled |
| OpenVPN Interactive Service | | Running | Automatic |
| OpenVPN Legacy Service | | | Manual |
| OpenVPNService | | | Manual |
| Optimise drives | Helps the c... | | Manual |
| OracleOraClient12Home2MTSRecoverySe... | | | Automatic |
| Ozeki | | Running | Automatic |
| Ozeki Installer | | | Automatic |
| OZEKI NG-SMS-Gateway | | Running | Automatic |
| Ozeki Phone System XE | | | Manual |
| OzekiCluster | | | Automatic |
| OzekiWatchDog | | | Automatic |
| Parental Controls | Enforces pa... | | Manual |
| Payments and NFC/SE Manager | Manages pa... | | Manual (Trig... |
| Peer Name Resolution Protocol | Enables serv... | | Manual |
| Peer Networking Grouping | Enables mul... | | Manual |
| Peer Networking Identity Manager | Provides ide... | | Manual |
| Performance Counter DLL Host | Enables rem... | | Manual |
| Performance Logs & Alerts | Performance... | | Manual |

Figure 12- Restart Ozeki Service

# Step 13 - Open the Ozeki 10 safely

After you restarted the Ozeki service, now it is ready to use Ozeki 10 from anywhere with the connected domain name. Figure 13 demonstrates that the Ozeki 10 started not from the localhost but using the connected domain name.

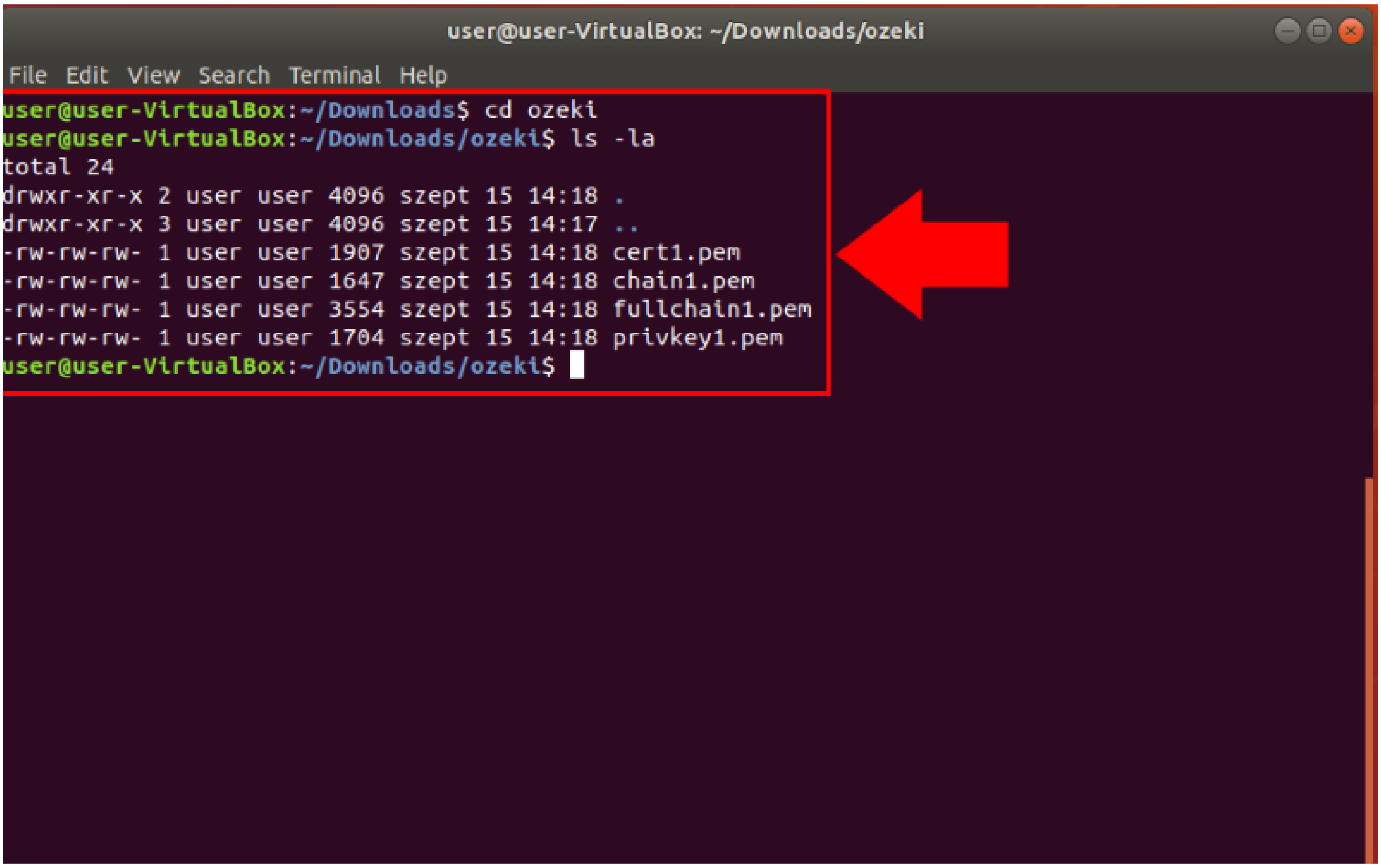


myhighschool.freeddns.org/?a=Desktop&refreshrequest=KRVHUKUY

Figure 13 - Open the Ozeki 10 safely

# How to convert an Apache certificate into pfx

Since Ozeki SMS Gateway uses PFX format certificates, to use Apache certificate you need to convert it to PFX. Just open the folder where the Apache certificates are stored (Figure 1).



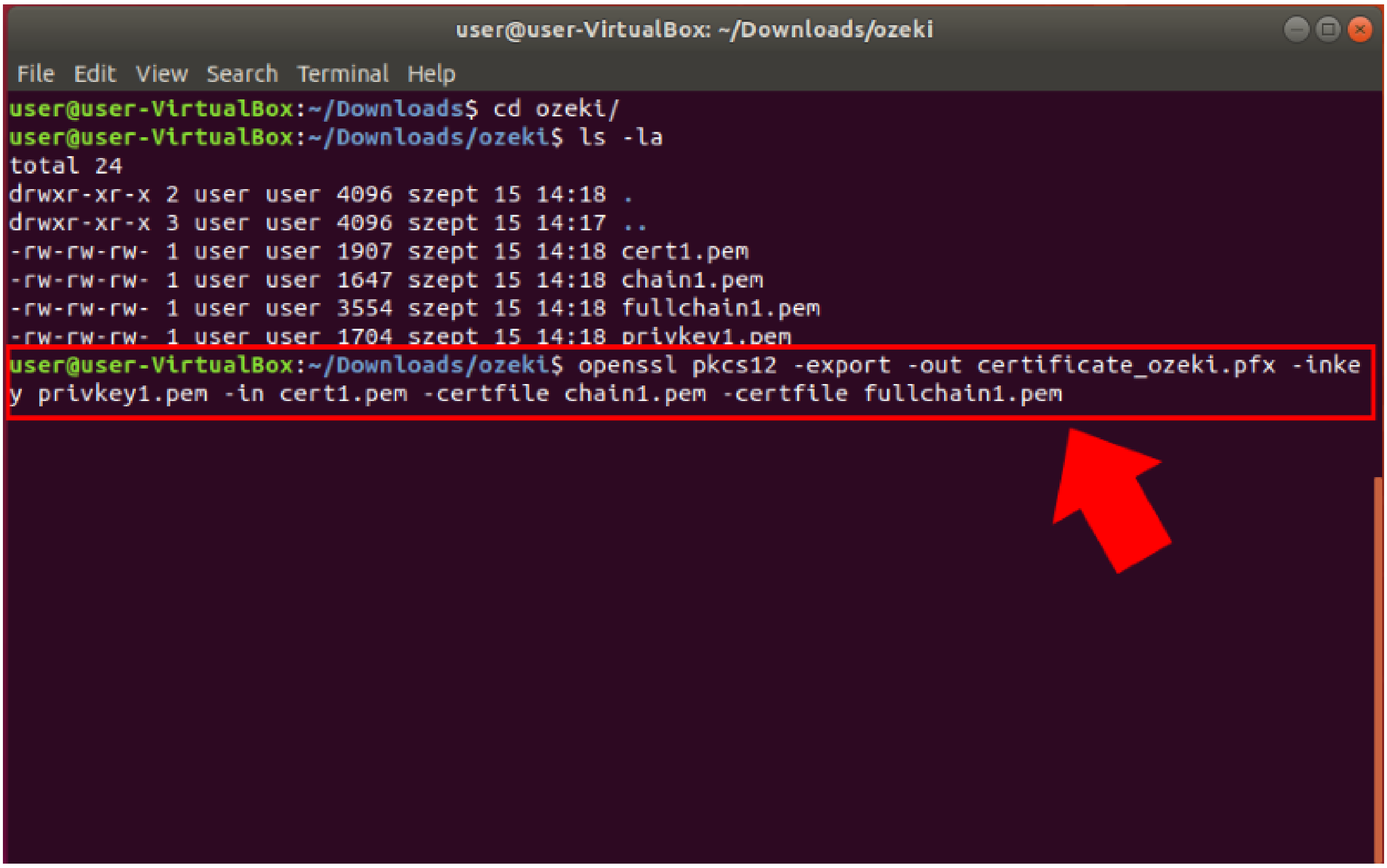
user@user-VirtualBox: ~/Downloads/ozeki
File Edit View Search Terminal Help

```
user@user-VirtualBox:~/Downloads$ cd ozeki
user@user-VirtualBox:~/Downloads/ozeki$ ls -la
total 24
drwxr-xr-x 2 user user 4096 szept 15 14:18 .
drwxr-xr-x 3 user user 4096 szept 15 14:17 ..
-rw-rw-rw- 1 user user 1907 szept 15 14:18 cert1.pem
-rw-rw-rw- 1 user user 1647 szept 15 14:18 chain1.pem
-rw-rw-rw- 1 user user 3554 szept 15 14:18 fullchain1.pem
-rw-rw-rw- 1 user user 1704 szept 15 14:18 privkey1.pem
user@user-VirtualBox:~/Downloads/ozeki$
```

Figure 1 - Open the certificate folder

Now run the following OpenSSL command (Figure 2).
openssl pkcs12 -export -out certificate_ozeki.pfx -inkey privkey1.pem -in cert1.pem -certfile chain1.pem -certfile fullchain1.pem

- -export -out certificate.pfx means export and save the PFX file as certificate_ozeki.pfx.
- -inkey privkey1.pem – Use the private key file privkey1.pem as the private key to combine with the certificate.
- -in cert1.pem – use cert1.pem as the certificate the private key will be combined with.
- -certfile chain1.pem – This is add additional certificates in the PFX file.

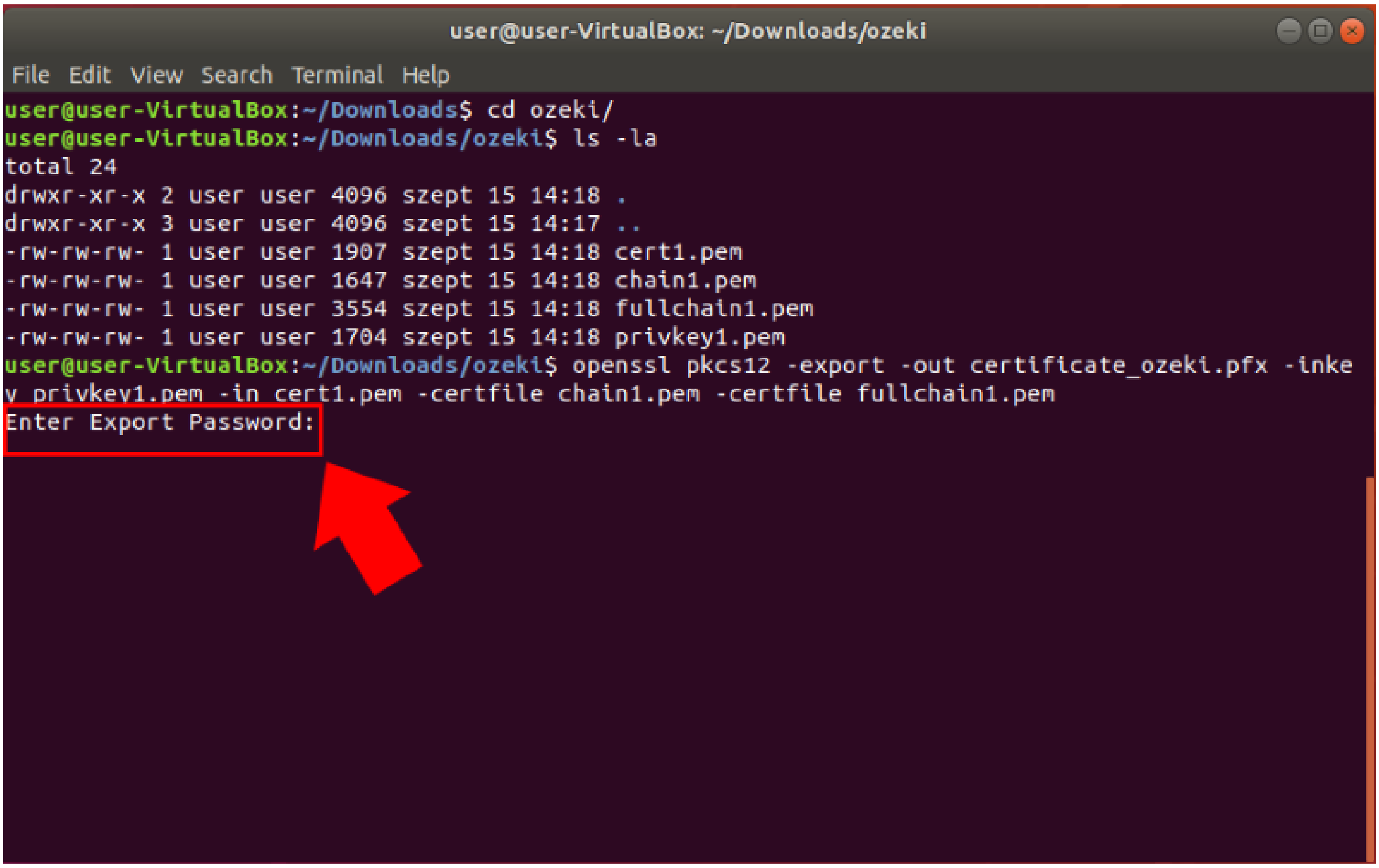


user@user-VirtualBox: ~/Downloads/ozeki
File Edit View Search Terminal Help

```
user@user-VirtualBox:~/Downloads$ cd ozeki/
user@user-VirtualBox:~/Downloads/ozeki$ ls -la
total 24
drwxr-xr-x 2 user user 4096 szept 15 14:18 .
drwxr-xr-x 3 user user 4096 szept 15 14:17 ..
-rw-rw-rw- 1 user user 1907 szept 15 14:18 cert1.pem
-rw-rw-rw- 1 user user 1647 szept 15 14:18 chain1.pem
-rw-rw-rw- 1 user user 3554 szept 15 14:18 fullchain1.pem
-rw-rw-rw- 1 user user 1704 szept 15 14:18 privkey1.pem
user@user-VirtualBox:~/Downloads/ozeki$ openssl pkcs12 -export -out certificate_ozeki.pfx -inke
y privkey1.pem -in cert1.pem -certfile chain1.pem -certfile fullchain1.pem
```

Figure 2 - Convert the certificate with Open SSL

After entering the command, you will be prompted to enter and verify an export password to protect the PFX file (Figure 3).

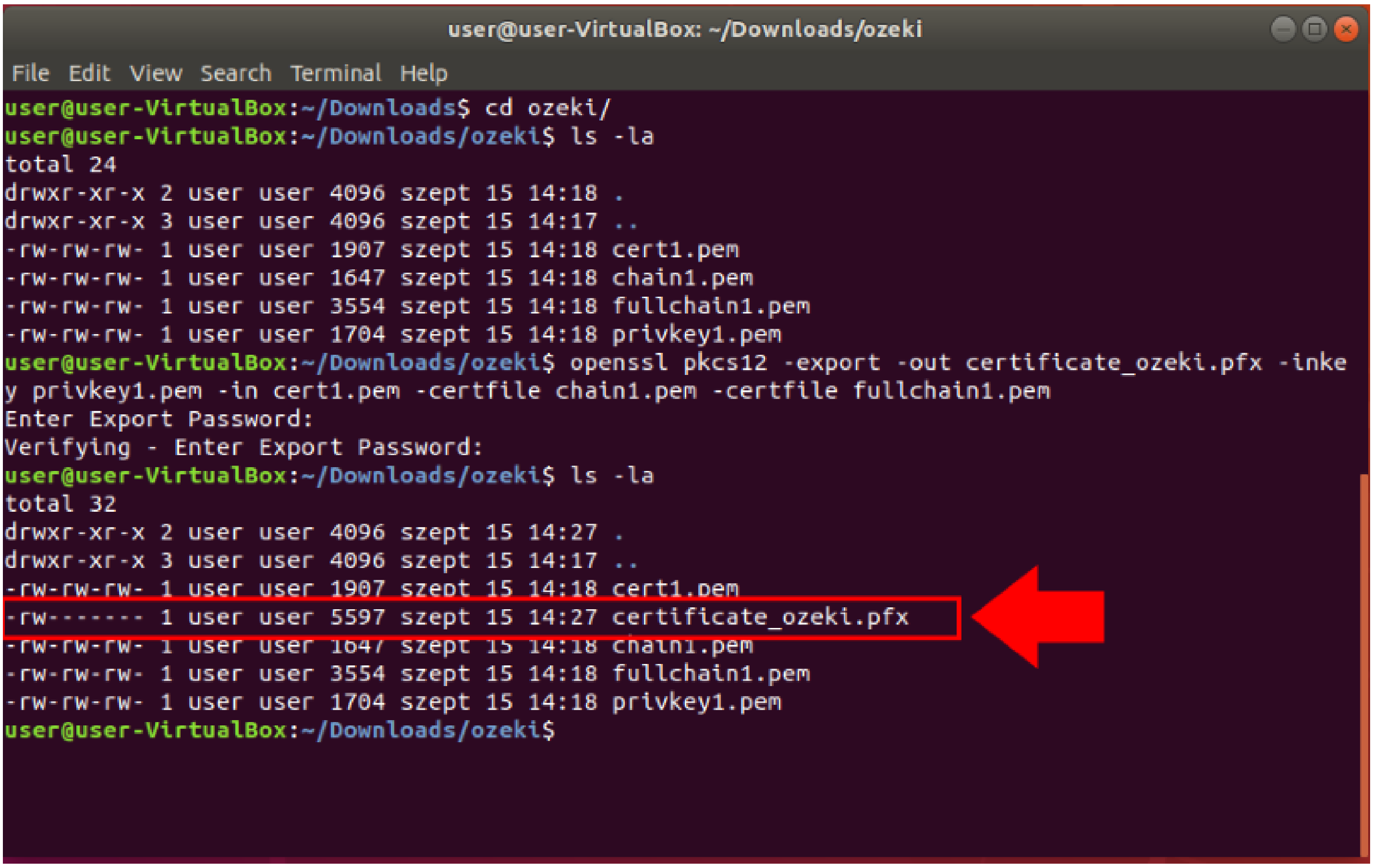


user@user-VirtualBox: ~/Downloads/ozeki
File Edit View Search Terminal Help

```
user@user-VirtualBox:~/Downloads$ cd ozeki/
user@user-VirtualBox:~/Downloads/ozeki$ ls -la
total 24
drwxr-xr-x 2 user user 4096 szept 15 14:18 .
drwxr-xr-x 3 user user 4096 szept 15 14:17 ..
-rw-rw-rw- 1 user user 1907 szept 15 14:18 cert1.pem
-rw-rw-rw- 1 user user 1647 szept 15 14:18 chain1.pem
-rw-rw-rw- 1 user user 3554 szept 15 14:18 fullchain1.pem
-rw-rw-rw- 1 user user 1704 szept 15 14:18 privkey1.pem
user@user-VirtualBox:~/Downloads/ozeki$ openssl pkcs12 -export -out certificate_ozeki.pfx -inke
y privkey1.pem -in cert1.pem -certfile chain1.pem -certfile fullchain1.pem
Enter Export Password:
```

Figure 3 - Enter Password

Then if you entered the password you will see that the new PFX certificate is created in the folder as you can see in the Figure 4.



user@user-VirtualBox: ~/Downloads/ozeki
File Edit View Search Terminal Help

```
user@user-VirtualBox:~/Downloads$ cd ozeki/
user@user-VirtualBox:~/Downloads/ozeki$ ls -la
total 24
drwxr-xr-x 2 user user 4096 szept 15 14:18 .
drwxr-xr-x 3 user user 4096 szept 15 14:17 ..
-rw-rw-rw- 1 user user 1907 szept 15 14:18 cert1.pem
-rw-rw-rw- 1 user user 1647 szept 15 14:18 chain1.pem
-rw-rw-rw- 1 user user 3554 szept 15 14:18 fullchain1.pem
-rw-rw-rw- 1 user user 1704 szept 15 14:18 privkey1.pem
user@user-VirtualBox:~/Downloads/ozeki$ openssl pkcs12 -export -out certificate_ozeki.pfx -inke
y privkey1.pem -in cert1.pem -certfile chain1.pem -certfile fullchain1.pem
Enter Export Password:
Verifying - Enter Export Password:
user@user-VirtualBox:~/Downloads/ozeki$ ls -la
total 32
drwxr-xr-x 2 user user 4096 szept 15 14:27 .
drwxr-xr-x 3 user user 4096 szept 15 14:17 ..
-rw-rw-rw- 1 user user 1907 szept 15 14:18 cert1.pem
-rw------- 1 user user 5597 szept 15 14:27 certificate_ozeki.pfx
-rw-rw-rw- 1 user user 1647 szept 15 14:18 chain1.pem
-rw-rw-rw- 1 user user 3554 szept 15 14:18 fullchain1.pem
-rw-rw-rw- 1 user user 1704 szept 15 14:18 privkey1.pem
user@user-VirtualBox:~/Downloads/ozeki$
```

Figure 4 - PFX cerated

The next main operation of this guide is to activate the converted .pfx certificate in Ozeki SMS Gateway. To do that, first, you need to open the Control Panel application in Ozeki SMS Gateway. So, just navigate to the desktop of Ozeki SMS Gateway, and here, as you can see it in Figure 5, just open Control Panel by clicking on its icon.

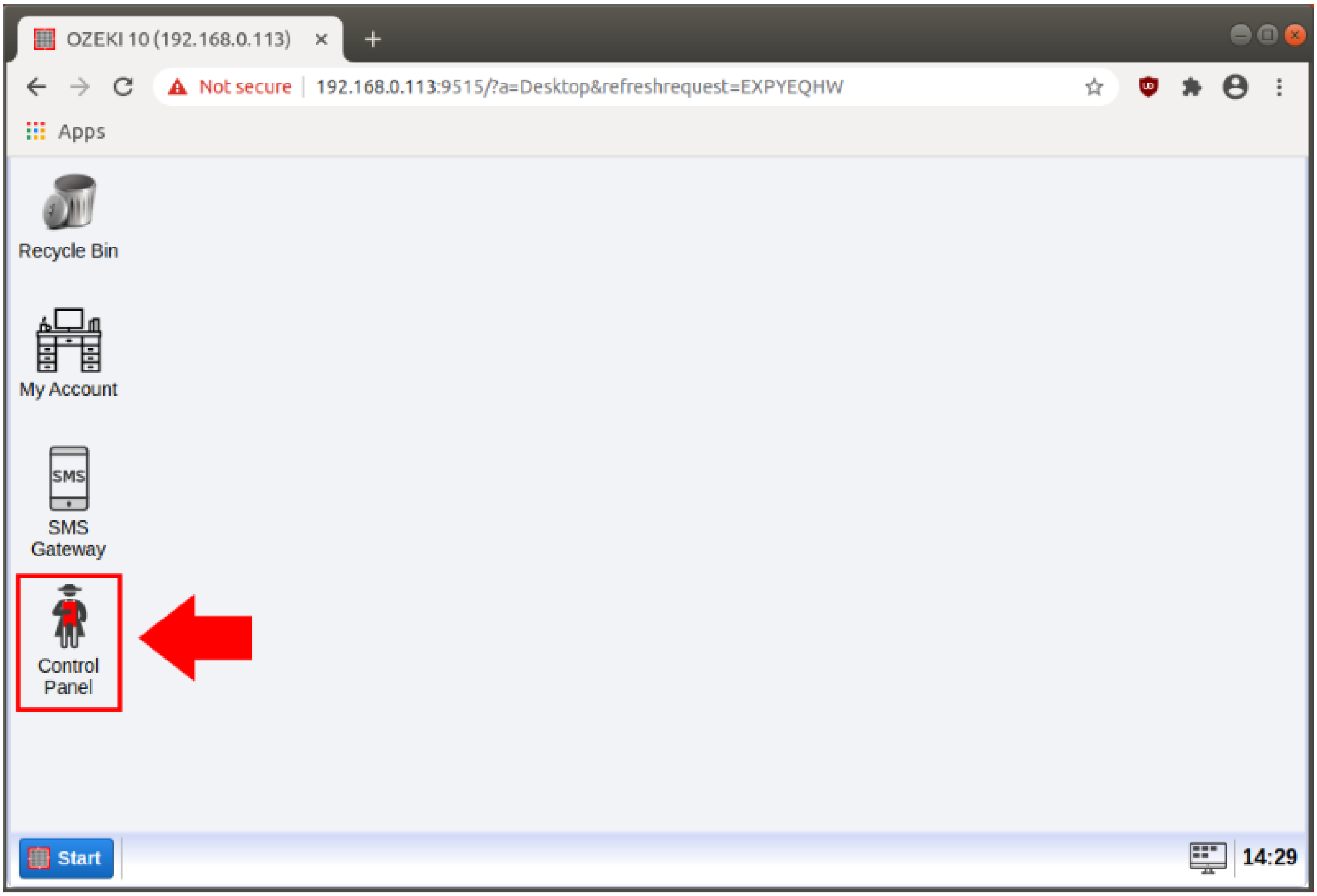
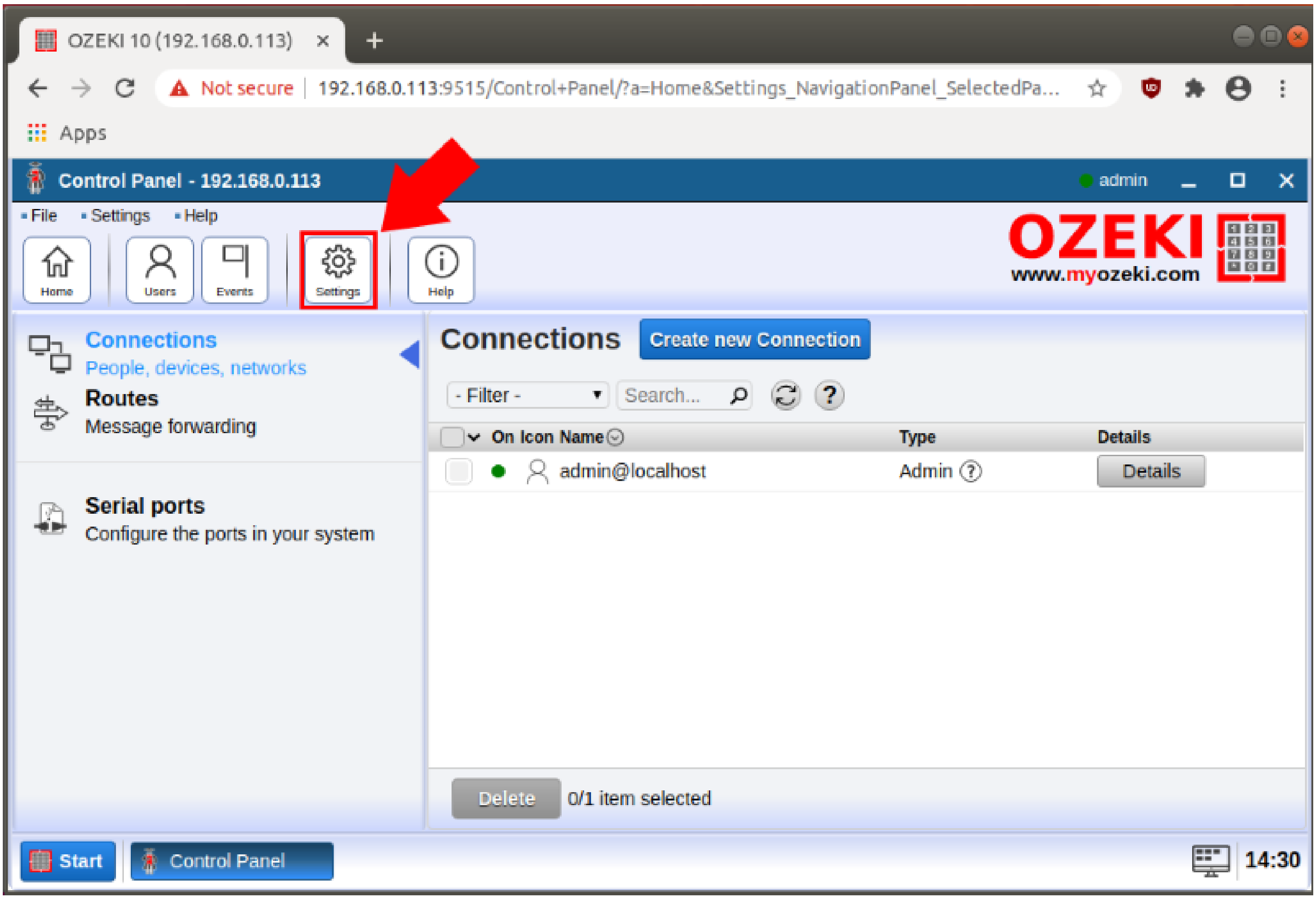


Figure 5 - Open Control panel

The Control Panel is the application, where you can configure your connections and the main settings of your Ozeki SMS Gateway. At this point, to activate the converted .pfx certificate, you need to go to the Settings menu. So, to do that, please click on the icon of the Settings menu on the toolbar like in Figure 6.

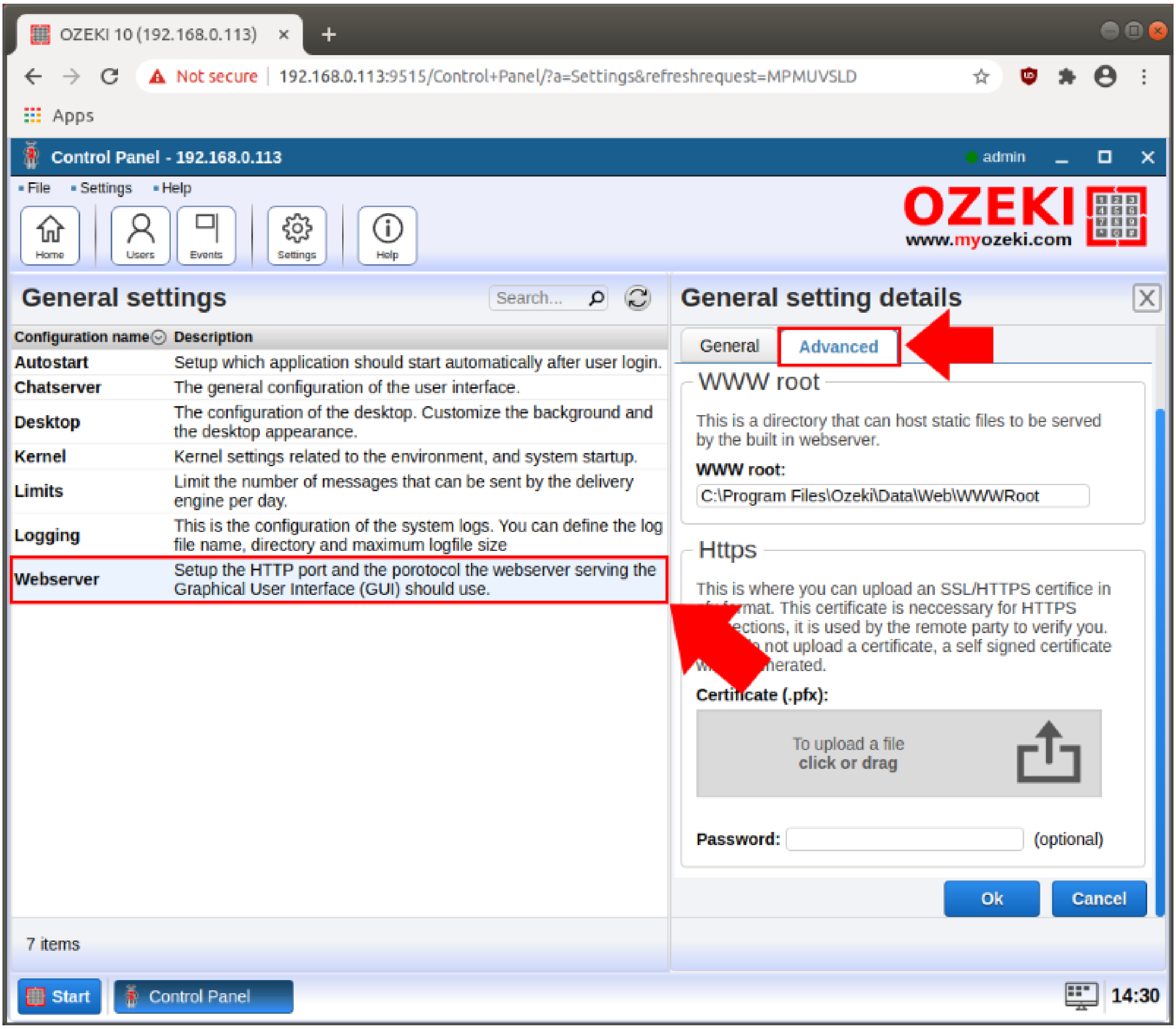


| On | Icon | Name | Type | Details |
|---|---|---|---|---|
| ● | | admin@localhost | Admin | Details |

0/1 item selected

Figure 6 - Contorl Panel Settings menu

Here, in the Settings menu, you can see all the configuration submenus, where you can make modifications. Now, you need to select the 'Webserver' menu from that list. After you selected the 'Webserver' menu, a new windows pops up, where you can perform the configuration. To be able to upload the converted certificate, here, select the Advanced tab as Figure 7 shows that.



OZEKI 10 (192.168.0.113)

Not secure | 192.168.0.113:9515/Control+Panel/?a=Settings&refreshrequest=MPMUVSLD

Control Panel - 192.168.0.113

## General settings

| Configuration name | Description |
|---|---|
| Autostart | Setup which application should start automatically after user login. |
| Chatserver | The general configuration of the user interface. |
| Desktop | The configuration of the desktop. Customize the background and the desktop appearance. |
| Kernel | Kernel settings related to the environment, and system startup. |
| Limits | Limit the number of messages that can be sent by the delivery engine per day. |
| Logging | This is the configuration of the system logs. You can define the log file name, directory and maximum logfile size |
| Webserver | Setup the HTTP port and the porotocol the webserver serving the Graphical User Interface (GUI) should use. |

7 items

## General setting details

General | Advanced

WWW root

This is a directory that can host static files to be served by the built in webserver.

WWW root: C:\Program Files\Ozeki\Data\Web\WWWRoot

Https

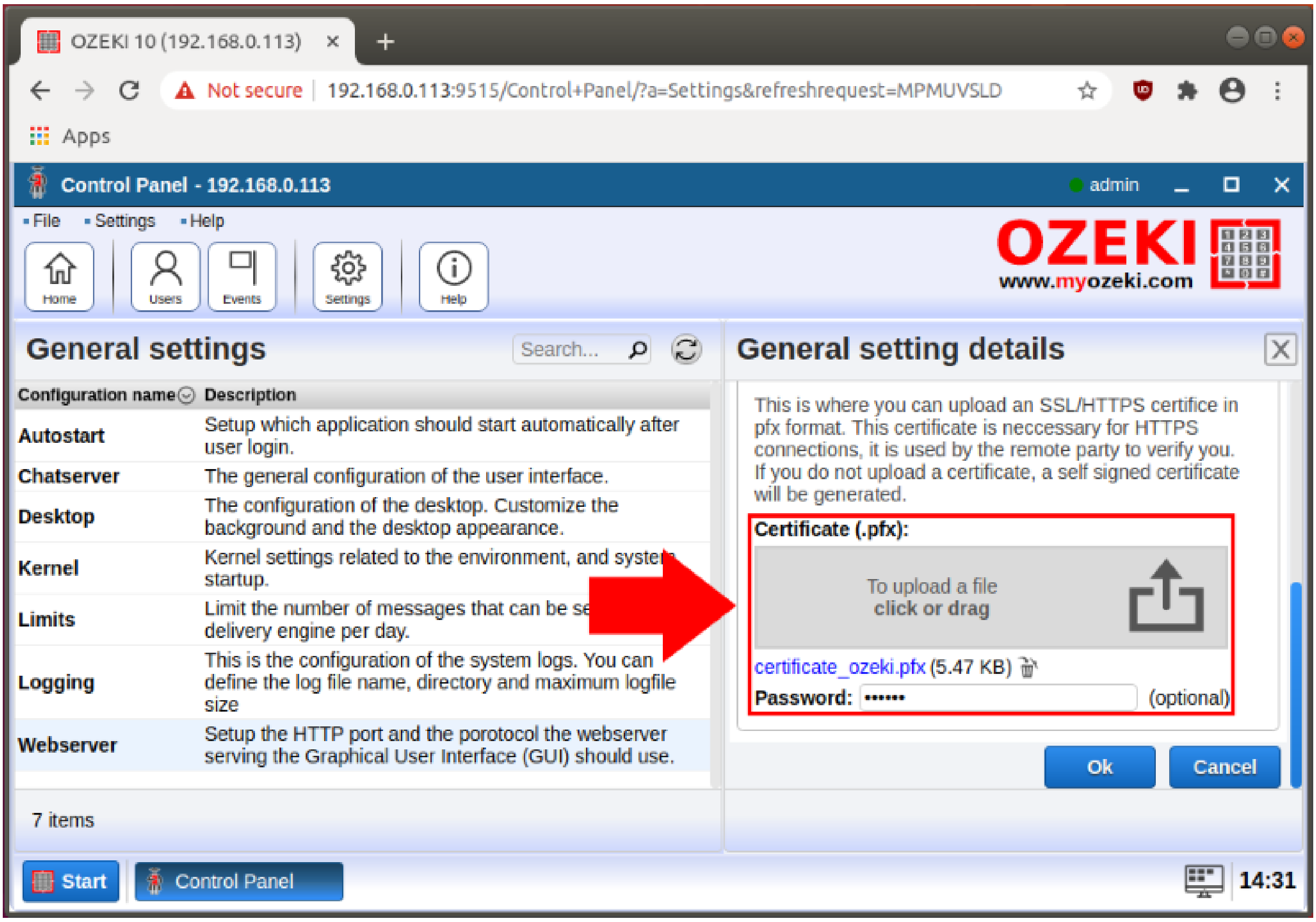
This is where you can upload an SSL/HTTPS certifice in ... format. This certificate is neccessary for HTTPS ...ections, it is used by the remote party to verify you. ... not upload a certificate, a self signed certificate w... nerated.

Certificate (.pfx): To upload a file click or drag

Password: (optional)

Figure 7 - Open Webserver Advanced menu

In the Advanced menu, you can upload the .pfx file that you converted in the previous steps. It is quite easy to upload the certificate, since you can browse the file after clicking on the upload field or simply drag and drop the certificate file as you can see it in Figure 8. Before finalizing the upload of the converted PFX certificate, you need to type the password of the certificate as well. This is the password that you had to provide when you converted the certificate.



**General settings**

| Configuration name | Description |
|---|---|
| Autostart | Setup which application should start automatically after user login. |
| Chatserver | The general configuration of the user interface. |
| Desktop | The configuration of the desktop. Customize the background and the desktop appearance. |
| Kernel | Kernel settings related to the environment, and syste startup. |
| Limits | Limit the number of messages that can be se delivery engine per day. |
| Logging | This is the configuration of the system logs. You can define the log file name, directory and maximum logfile size |
| Webserver | Setup the HTTP port and the porotocol the webserver serving the Graphical User Interface (GUI) should use. |

7 items

**General setting details**

This is where you can upload an SSL/HTTPS certifice in pfx format. This certificate is neccessary for HTTPS connections, it is used by the remote party to verify you. If you do not upload a certificate, a self signed certificate will be generated.

Certificate (.pfx):

To upload a file
click or drag

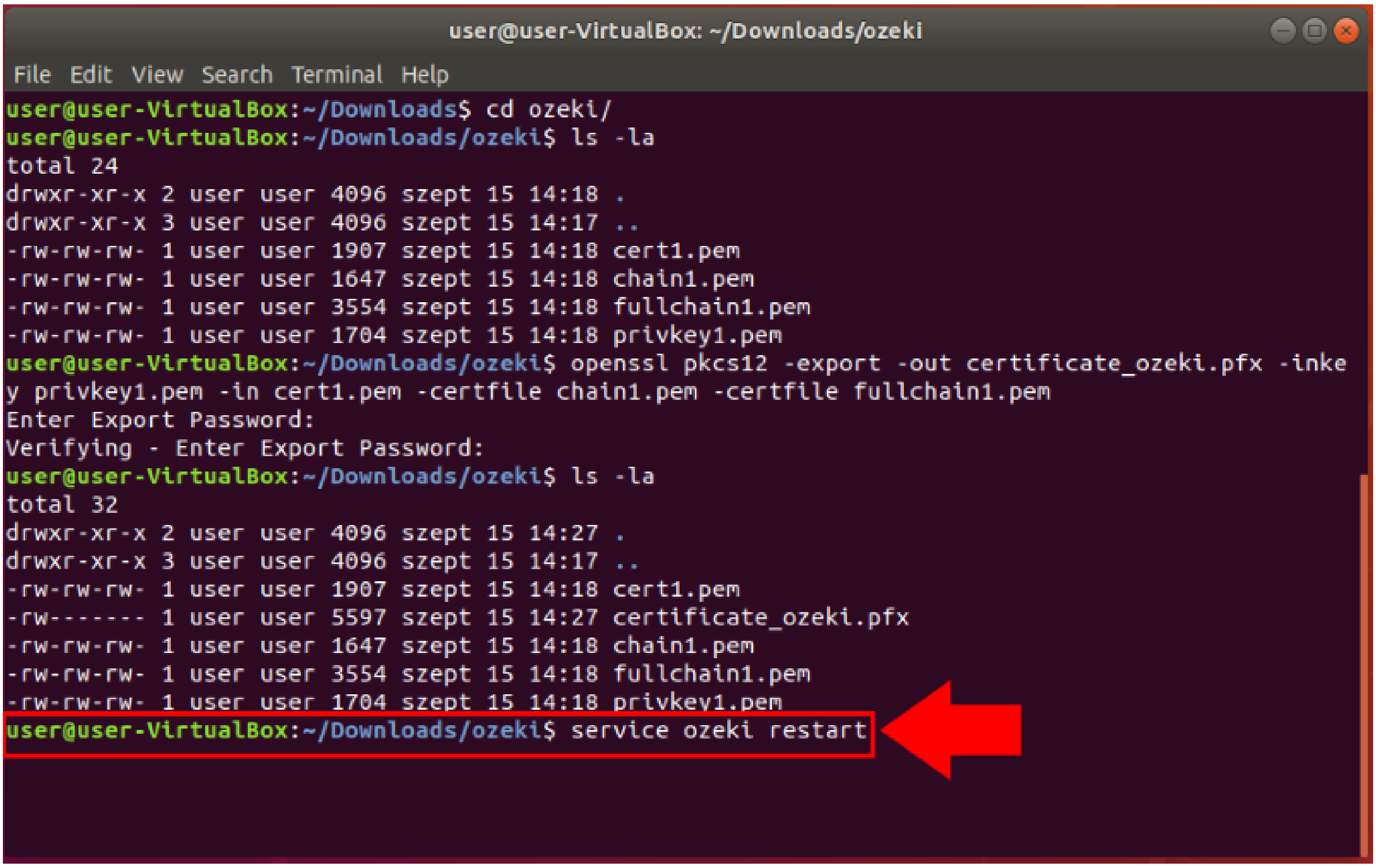
certificate_ozeki.pfx (5.47 KB)

Password: •••••• (optional)

Ok Cancel

Figure 8 - Upload PFX certificate

At this point, you uploaded the PFX certificate to your Ozeki SMS Gateway, but to be able to use it, you need to restart the Ozeki service. You can restart the service quite easily. For that, run the Service Ozeki restart command in the Command line as Figure 9 shows that.


user@user-VirtualBox: ~/Downloads/ozeki
File Edit View Search Terminal Help

```
user@user-VirtualBox:~/Downloads$ cd ozeki/
user@user-VirtualBox:~/Downloads/ozeki$ ls -la
total 24
drwxr-xr-x 2 user user 4096 szept 15 14:18 .
drwxr-xr-x 3 user user 4096 szept 15 14:17 ..
-rw-rw-rw- 1 user user 1907 szept 15 14:18 cert1.pem
-rw-rw-rw- 1 user user 1647 szept 15 14:18 chain1.pem
-rw-rw-rw- 1 user user 3554 szept 15 14:18 fullchain1.pem
-rw-rw-rw- 1 user user 1704 szept 15 14:18 privkey1.pem
user@user-VirtualBox:~/Downloads/ozeki$ openssl pkcs12 -export -out certificate_ozeki.pfx -inke
y privkey1.pem -in cert1.pem -certfile chain1.pem -certfile fullchain1.pem
Enter Export Password:
Verifying - Enter Export Password:
user@user-VirtualBox:~/Downloads/ozeki$ ls -la
total 32
drwxr-xr-x 2 user user 4096 szept 15 14:27 .
drwxr-xr-x 3 user user 4096 szept 15 14:17 ..
-rw-rw-rw- 1 user user 1907 szept 15 14:18 cert1.pem
-rw------- 1 user user 5597 szept 15 14:27 certificate_ozeki.pfx
-rw-rw-rw- 1 user user 1647 szept 15 14:18 chain1.pem
-rw-rw-rw- 1 user user 3554 szept 15 14:18 fullchain1.pem
-rw-rw-rw- 1 user user 1704 szept 15 14:18 privkey1.pem
user@user-VirtualBox:~/Downloads/ozeki$ service ozeki restart
```

Figure 9 - Restart Ozeki Service

# Trouble shooting HTTPS

***In case you experience problems with the HTTPS access to the GUI, you may use the following steps to figure out what causes the problem. The good news is you can still use the software on the HTTP port, while the HTTPS connection issue is not resolved.***

## Check certificate binding in the scurity appkication

Open security application form the Start menu as you can see in the Figure 1.

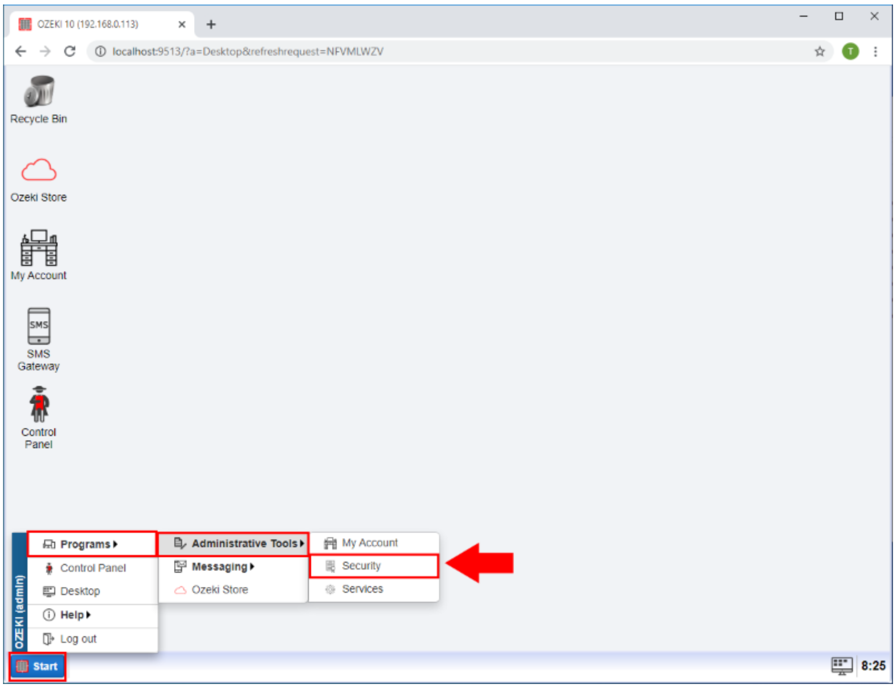
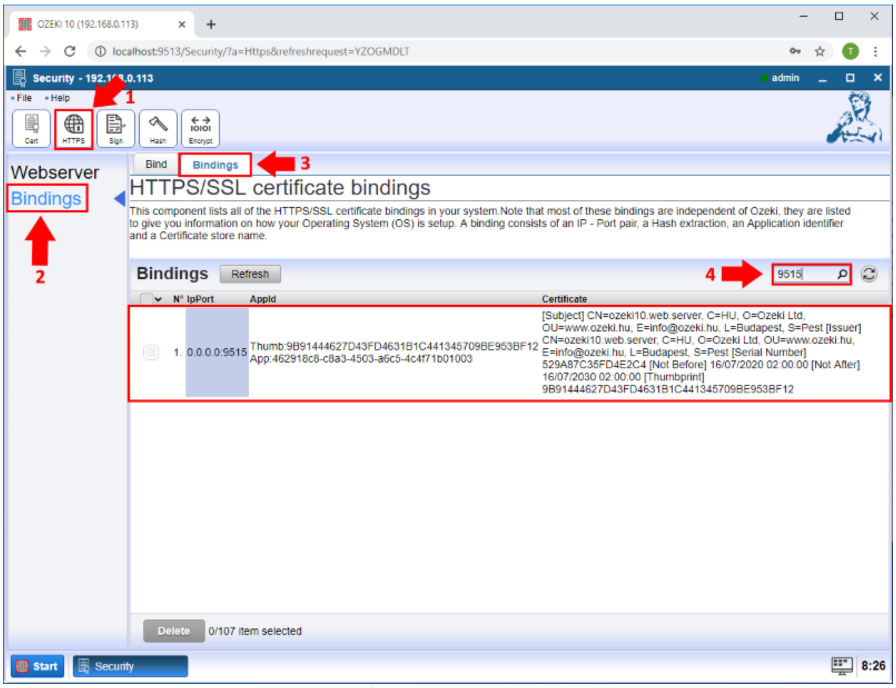


**Figure 1 -** Open Scurity application

After the Security app is opened go to the HTTPS menu. In the HTTPS menu select the Bindings section and open the Bindings tab. Here you can find all the certificate bindings on your computer. In the search field type the 9515 because by default the Ozeki use this port for HTTPS. Now you have to see that the ozeki certficate is binded to the 9515 port as the Figure 2 shows.



HTTPS/SSL certificate bindings

This component lists all of the HTTPS/SSL certificate bindings in your system.Note that most of these bindings are independent of Ozeki, they are listed to give you information on how your Operating System (OS) is setup. A binding consists of an IP - Port pair, a Hash extraction, an Application identifier and a Certificate store name.

Bindings

| N° | IpPort | AppId | Certificate |
|---|---|---|---|
| 1. | 0.0.0.0:9515 | Thumb:9B91444627D43FD4631B1C441345709BE953BF12 App:462918c8-c8a3-4503-a6c5-4c4f71b01003 | [Subject] CN=ozeki10.web.server, C=HU, O=Ozeki Ltd, OU=www.ozeki.hu, E=info@ozeki.hu, L=Budapest, S=Pest [Issuer] CN=ozeki10.web.server, C=HU, O=Ozeki Ltd, OU=www.ozeki.hu, E=info@ozeki.hu, L=Budapest, S=Pest [Serial Number] 529A87C35FD4E2C4 [Not Before] 16/07/2020 02:00:00 [Not After] 16/07/2030 02:00:00 [Thumbprint] 9B91444627D43FD4631B1C441345709BE953BF12 |

**Figure 2 -** Check certificate binding

# Check that the https listener has started on the correct port

Open the Control Panel application from the Ozeki desktop (Figure 3).

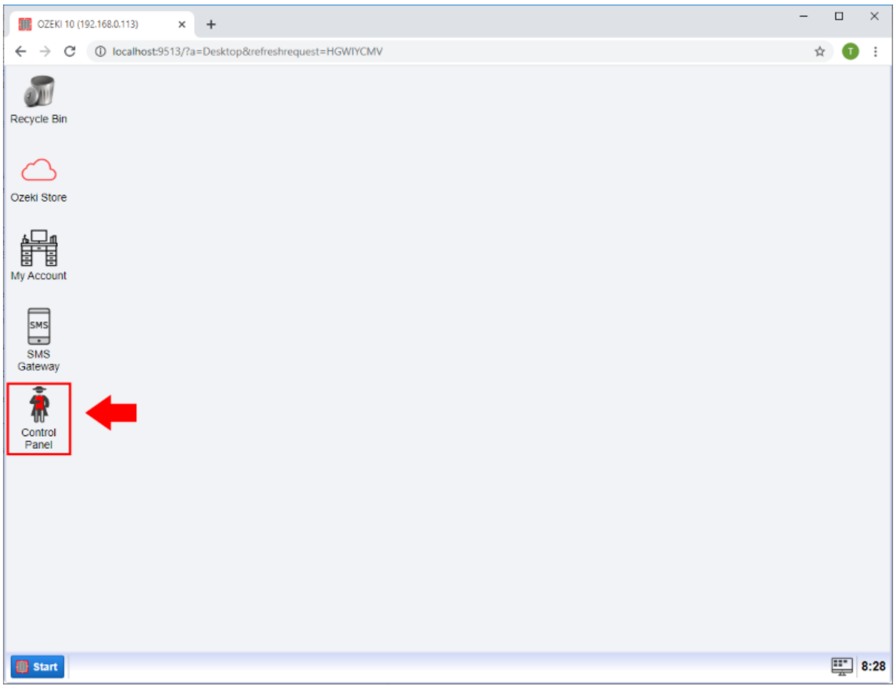
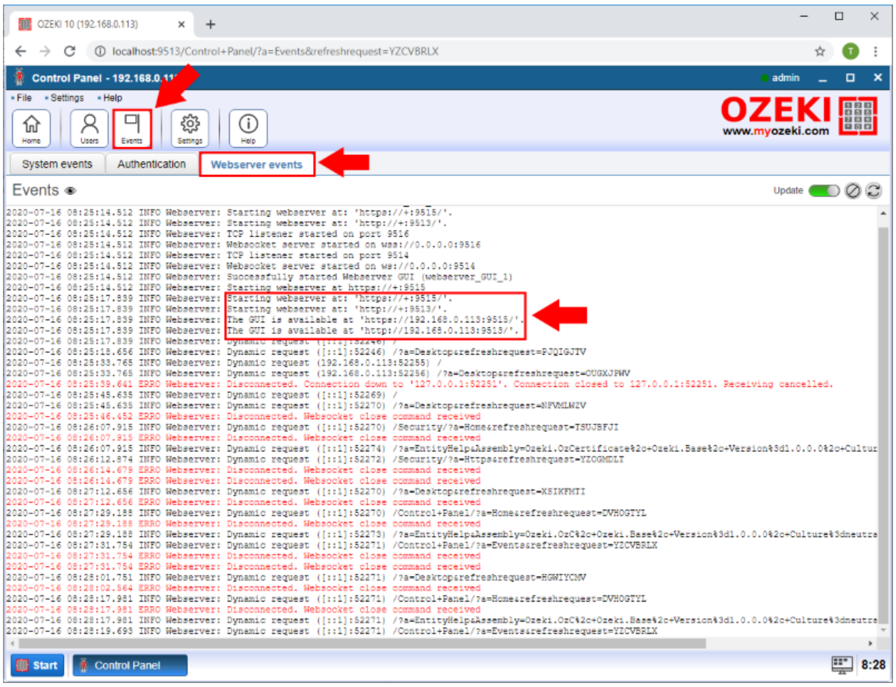


**Figure 3 -** Open Control Panel application

In the Control Panel Events menu open the Webserver events tab. Here you can see the logs for the Ozeki's webserver. From the logs you can see if the webserver has started properly in the 9515 port (Figure 4).

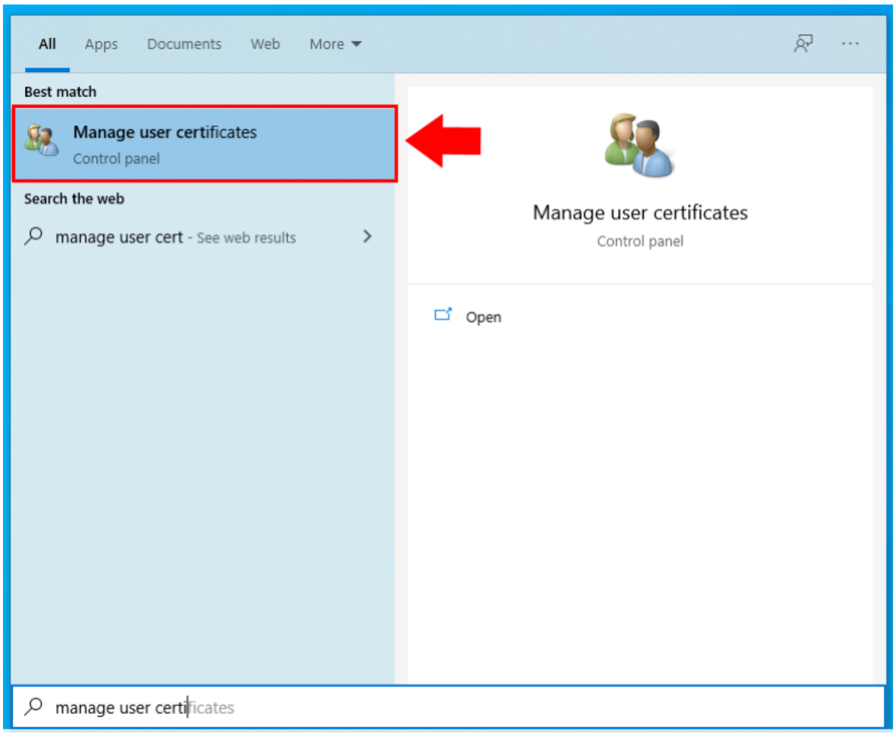


```
2020-07-16 08:25:14.512 INFO Webserver: Starting webserver at: 'https://+:9515/'.
2020-07-16 08:25:14.512 INFO Webserver: Starting webserver at: 'http://+:9513/'.
2020-07-16 08:25:14.512 INFO Webserver: TCP listener started on port 9516
2020-07-16 08:25:14.512 INFO Webserver: Websocket server started on wss://0.0.0.0:9516
2020-07-16 08:25:14.512 INFO Webserver: TCP listener started on port 9514
2020-07-16 08:25:14.512 INFO Webserver: Websocket server started on ws://0.0.0.0:9514
2020-07-16 08:25:14.512 INFO Webserver: Successfully started Webserver GUI (webserver_GUI_1)
2020-07-16 08:25:14.512 INFO Webserver: Starting webserver at https://+:9515
2020-07-16 08:25:17.839 INFO Webserver: Starting webserver at: 'https://+:9515/'.
2020-07-16 08:25:17.839 INFO Webserver: Starting webserver at: 'http://+:9513/'.
2020-07-16 08:25:17.839 INFO Webserver: The GUI is available at 'https://192.168.0.113:9515/'.
2020-07-16 08:25:17.839 INFO Webserver: The GUI is available at 'http://192.168.0.113:9513/'.
2020-07-16 08:25:17.839 INFO Webserver: Dynamic request ([::1]:52246) /
2020-07-16 08:25:18.656 INFO Webserver: Dynamic request ([::1]:52246) /?a=Desktop&refreshrequest=PJQIGJTV
2020-07-16 08:25:33.765 INFO Webserver: Dynamic request (192.168.0.113:52255) /
2020-07-16 08:25:33.765 INFO Webserver: Dynamic request (192.168.0.113:52256) /?a=Desktop&refreshrequest=OUGXJPWV
2020-07-16 08:25:39.641 ERRO Webserver: Disconnected. Connection down to '127.0.0.1:52251'. Connection closed to 127.0.0.1:52251. Receiving cancelled.
2020-07-16 08:25:45.635 INFO Webserver: Dynamic request ([::1]:52269) /
2020-07-16 08:25:45.635 INFO Webserver: Dynamic request ([::1]:52270) /?a=Desktop&refreshrequest=NFVMLWZV
2020-07-16 08:25:46.452 ERRO Webserver: Disconnected. Websocket close command received
2020-07-16 08:26:07.915 INFO Webserver: Dynamic request ([::1]:52270) /Security/?a=Home&refreshrequest=TSUJBFJI
2020-07-16 08:26:07.915 ERRO Webserver: Disconnected. Websocket close command received
2020-07-16 08:26:07.915 INFO Webserver: Dynamic request ([::1]:52274) /?a=EntityHelp&Assembly=Ozeki.OzCertificate%2c+Ozeki.Base%2c+Version%3d1.0.0.0%2c+Cultur
2020-07-16 08:26:12.874 INFO Webserver: Dynamic request ([::1]:52272) /Security/?a=Https&refreshrequest=YZOGMDLT
2020-07-16 08:26:14.679 ERRO Webserver: Disconnected. Websocket close command received
2020-07-16 08:26:14.679 ERRO Webserver: Disconnected. Websocket close command received
2020-07-16 08:27:12.656 INFO Webserver: Dynamic request ([::1]:52270) /?a=Desktop&refreshrequest=XSIKFMTI
2020-07-16 08:27:12.656 ERRO Webserver: Disconnected. Websocket close command received
2020-07-16 08:27:29.188 INFO Webserver: Dynamic request ([::1]:52270) /Control+Panel/?a=Home&refreshrequest=DVHOGTYL
2020-07-16 08:27:29.188 ERRO Webserver: Disconnected. Websocket close command received
2020-07-16 08:27:29.188 INFO Webserver: Dynamic request ([::1]:52273) /?a=EntityHelp&Assembly=Ozeki.OzC%2c+Ozeki.Base%2c+Version%3d1.0.0.0%2c+Culture%3dneutra
2020-07-16 08:27:31.754 INFO Webserver: Dynamic request ([::1]:52271) /Control+Panel/?a=Events&refreshrequest=YZCVBRLX
2020-07-16 08:27:31.754 ERRO Webserver: Disconnected. Websocket close command received
2020-07-16 08:27:31.754 ERRO Webserver: Disconnected. Websocket close command received
2020-07-16 08:28:01.751 INFO Webserver: Dynamic request ([::1]:52271) /?a=Desktop&refreshrequest=HGWIYCMV
2020-07-16 08:28:02.564 ERRO Webserver: Disconnected. Websocket close command received
2020-07-16 08:28:17.981 INFO Webserver: Dynamic request ([::1]:52271) /Control+Panel/?a=Home&refreshrequest=DVHOGTYL
2020-07-16 08:28:17.981 ERRO Webserver: Disconnected. Websocket close command received
2020-07-16 08:28:17.981 INFO Webserver: Dynamic request ([::1]:52271) /?a=EntityHelp&Assembly=Ozeki.OzC%2c+Ozeki.Base%2c+Version%3d1.0.0.0%2c+Culture%3dneutra
2020-07-16 08:28:19.693 INFO Webserver: Dynamic request ([::1]:52271) /Control+Panel/?a=Events&refreshrequest=YZCVBRLX
```

**Figure 4 -** Webserver events

# Check if the Ozeki certificate is installed into the cert store

In the certificate manager you can find all the installed certificates in your Windows system. Open it form the Windows start menu. Search for the Manage user certificates as you can see in the Figure 5.



All Apps Documents Web More

Best match

Manage user certificates
Control panel

Search the web

manage user cert - See web results

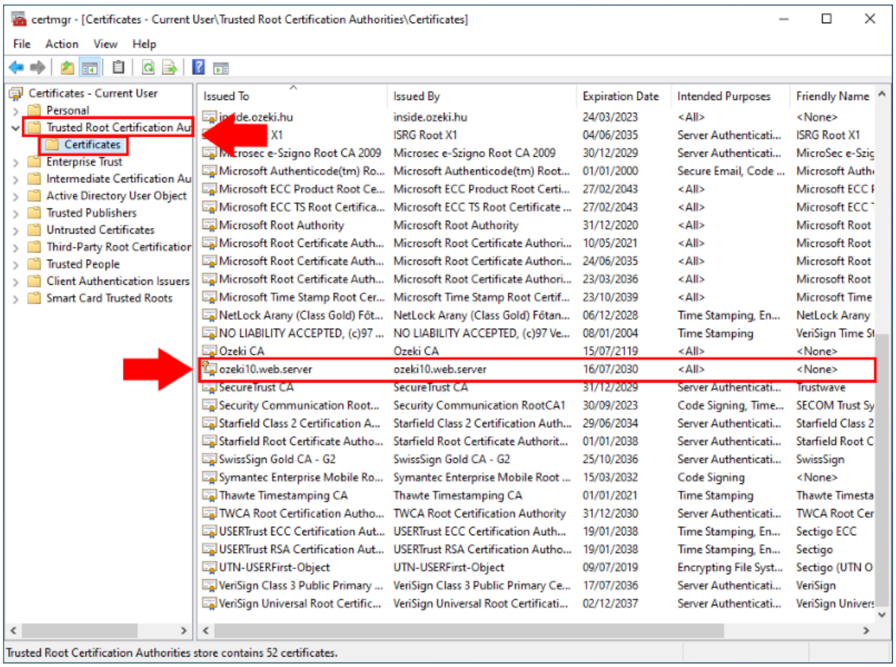
Manage user certificates
Control panel

Open

manage user certificates

Figure 5 - Manage user certificates

In the certificate manager open the certificates under the Trusted Root Authorites folder. Here you can find the ozeki10.web.server certificate as the Figure 6 shows.



certmgr - [Certificates - Current User\Trusted Root Certification Authorities\Certificates]

File Action View Help

- Certificates - Current User
  - Personal
  - Trusted Root Certification Au
    - Certificates
  - Enterprise Trust
  - Intermediate Certification Au
  - Active Directory User Object
  - Trusted Publishers
  - Untrusted Certificates
  - Third-Party Root Certificatior
  - Trusted People
  - Client Authentication Issuers
  - Smart Card Trusted Roots

| Issued To | Issued By | Expiration Date | Intended Purposes | Friendly Name |
|---|---|---|---|---|
| in ide.ozeki.hu | inside.ozeki.hu | 24/03/2023 | <All> | <None> |
| X1 | ISRG Root X1 | 04/06/2035 | Server Authenticati... | ISRG Root X1 |
| M crosec e-Szigno Root CA 2009 | Microsec e-Szigno Root CA 2009 | 30/12/2029 | Server Authenticati... | MicroSec e-Szig |
| Microsoft Authenticode(tm) Ro... | Microsoft Authenticode(tm) Root... | 01/01/2000 | Secure Email, Code ... | Microsoft Auth |
| Microsoft ECC Product Root Ce... | Microsoft ECC Product Root Certi... | 27/02/2043 | <All> | Microsoft ECC I |
| Microsoft ECC TS Root Certifica... | Microsoft ECC TS Root Certificate ... | 27/02/2043 | <All> | Microsoft ECC T |
| Microsoft Root Authority | Microsoft Root Authority | 31/12/2020 | <All> | Microsoft Root |
| Microsoft Root Certificate Auth... | Microsoft Root Certificate Authori... | 10/05/2021 | <All> | Microsoft Root |
| Microsoft Root Certificate Auth... | Microsoft Root Certificate Authori... | 24/06/2035 | <All> | Microsoft Root |
| Microsoft Root Certificate Auth... | Microsoft Root Certificate Authori... | 23/03/2036 | <All> | Microsoft Root |
| Microsoft Time Stamp Root Cer... | Microsoft Time Stamp Root Certif... | 23/10/2039 | <All> | Microsoft Time |
| NetLock Arany (Class Gold) Főt... | NetLock Arany (Class Gold) Főtan... | 06/12/2028 | Time Stamping, En... | NetLock Arany |
| NO LIABILITY ACCEPTED, (c)97 ... | NO LIABILITY ACCEPTED, (c)97 Ve... | 08/01/2004 | Time Stamping | VeriSign Time St |
| Ozeki CA | Ozeki CA | 15/07/2119 | <All> | <None> |
| ozeki10.web.server | ozeki10.web.server | 16/07/2030 | <All> | <None> |
| SecureTrust CA | SecureTrust CA | 31/12/2029 | Server Authenticati... | Trustwave |
| Security Communication Root... | Security Communication RootCA1 | 30/09/2023 | Code Signing, Time... | SECOM Trust Sy |
| Starfield Class 2 Certification A... | Starfield Class 2 Certification Auth... | 29/06/2034 | Server Authenticati... | Starfield Class 2 |
| Starfield Root Certificate Autho... | Starfield Root Certificate Authorit... | 01/01/2038 | Server Authenticati... | Starfield Root C |
| SwissSign Gold CA - G2 | SwissSign Gold CA - G2 | 25/10/2036 | Server Authenticati... | SwissSign |
| Symantec Enterprise Mobile Ro... | Symantec Enterprise Mobile Root ... | 15/03/2032 | Code Signing | <None> |
| Thawte Timestamping CA | Thawte Timestamping CA | 01/01/2021 | Time Stamping | Thawte Timesta |
| TWCA Root Certification Autho... | TWCA Root Certification Authority | 31/12/2030 | Server Authenticati... | TWCA Root Cer |
| USERTrust ECC Certification Aut... | USERTrust ECC Certification Auth... | 19/01/2038 | Time Stamping, En... | Sectigo ECC |
| USERTrust RSA Certification Aut... | USERTrust RSA Certification Autho... | 19/01/2038 | Time Stamping, En... | Sectigo |
| UTN-USERFirst-Object | UTN-USERFirst-Object | 09/07/2019 | Encrypting File Syst... | Sectigo (UTN O |
| VeriSign Class 3 Public Primary ... | VeriSign Class 3 Public Primary Ce... | 17/07/2036 | Server Authenticati... | VeriSign |
| VeriSign Universal Root Certific... | VeriSign Universal Root Certificati... | 02/12/2037 | Server Authenticati... | VeriSign Univers |

Trusted Root Certification Authorities store contains 52 certificates.

Figure 6 - Ozeki certificate in the cert store

# Try to open the GUI in another Browser

If you use Google Chrome please try Mozzilla Firefox. The Microsoft Edge is not good because it is also Chrome based.

# Check webserver configuration in the Control Panel

Open the Control Panel application from the Ozeki desktop (Figure 7).

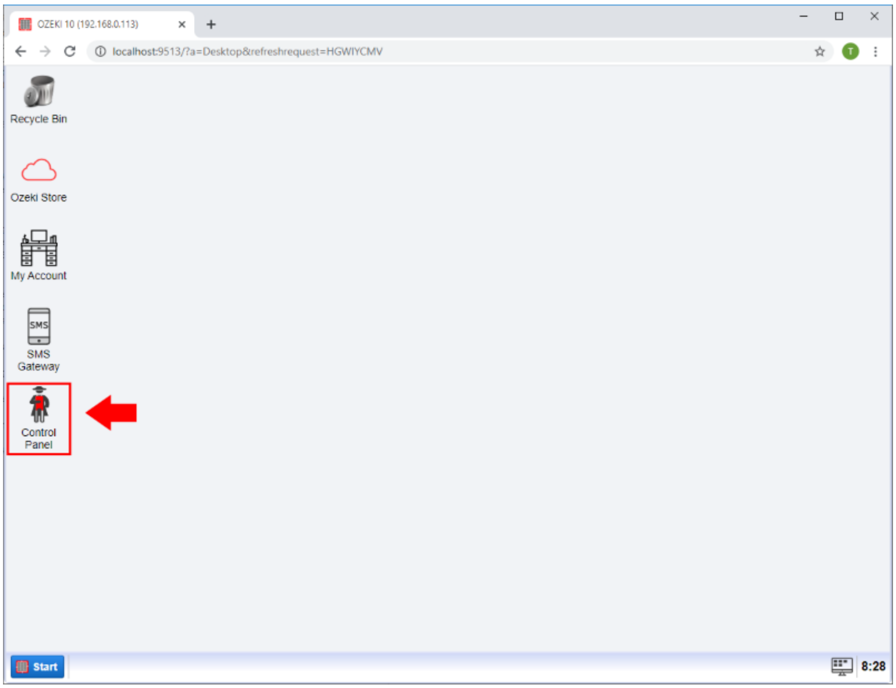
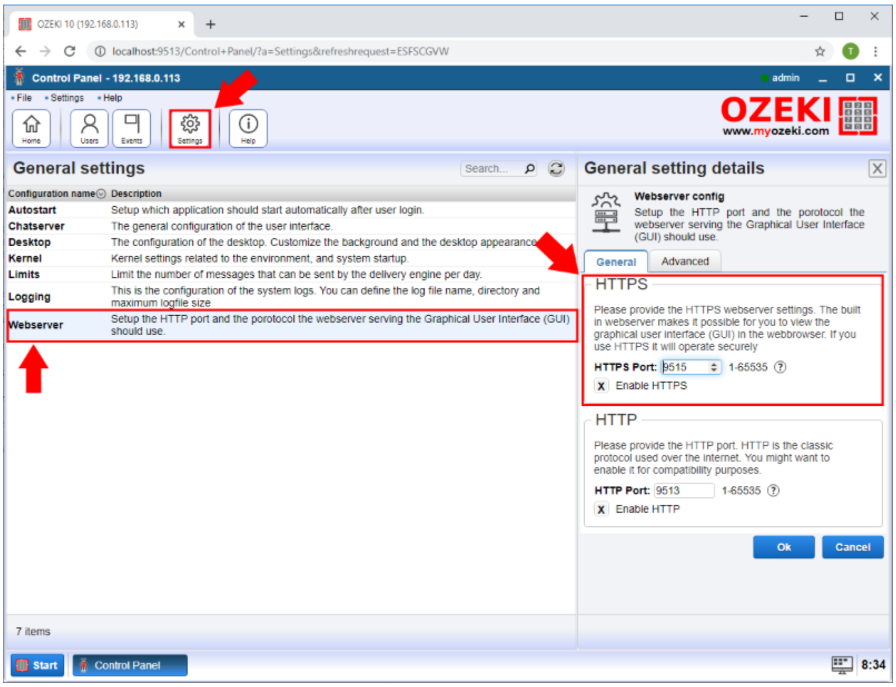


**Figure 7** - Open Control Panel application

In the Control Panel Settings menu open the Webserver config. Here make sure that the HTTPS porotocoll is enabled it runs on the proper port as you can see on the Figure 8.



General settings

| Configuration name | Description |
|---|---|
| Autostart | Setup which application should start automatically after user login. |
| Chatserver | The general configuration of the user interface. |
| Desktop | The configuration of the desktop. Customize the background and the desktop appearance |
| Kernel | Kernel settings related to the environment, and system startup. |
| Limits | Limit the number of messages that can be sent by the delivery engine per day. |
| Logging | This is the configuration of the system logs. You can define the log file name, directory and maximum logfile size |
| Webserver | Setup the HTTP port and the porotocol the webserver serving the Graphical User Interface (GUI) should use. |

General setting details

**Webserver config**

Setup the HTTP port and the porotocol the webserver serving the Graphical User Interface (GUI) should use.

HTTPS

Please provide the HTTPS webserver settings. The built in webserver makes it possible for you to view the graphical user interface (GUI) in the webbrowser. If you use HTTPS it will operate securely

**HTTPS Port:** 9515 1-65535

☒ Enable HTTPS

HTTP

Please provide the HTTP port. HTTP is the classic protocol used over the internet. You might want to enable it for compatibility purposes.

**HTTP Port:** 9513 1-65535

☒ Enable HTTP

**Figure 8** - Webserver settings

# Check Windows proxy settings

Open the Settings in the Chrome browser (Figure 9).

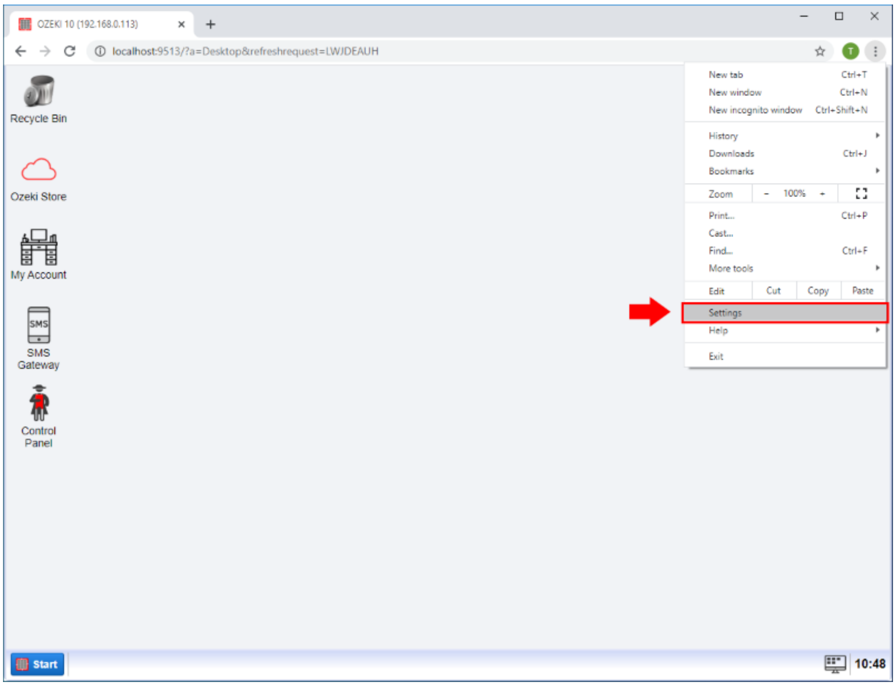
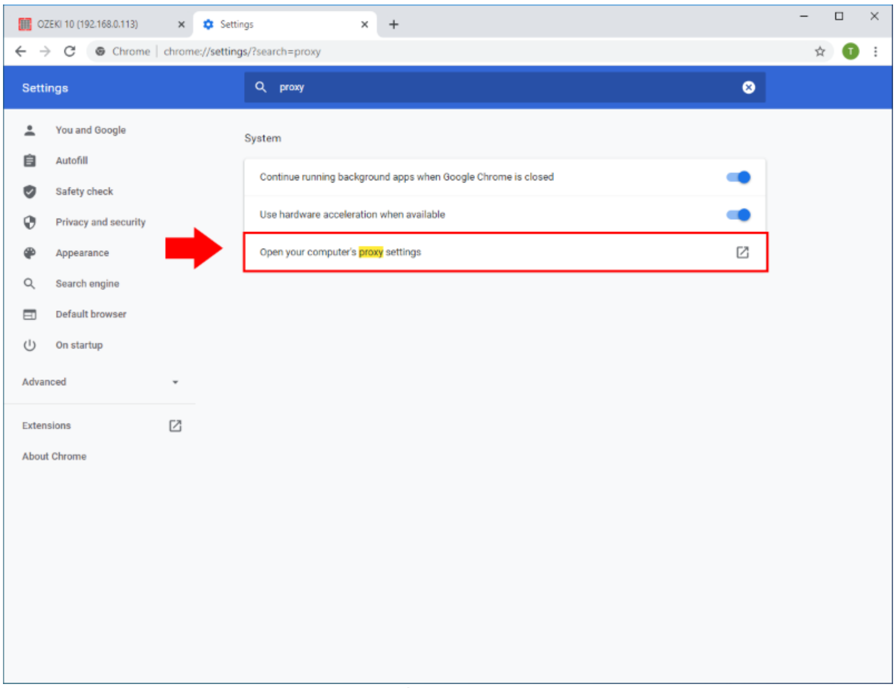


**Figure 9** - Chrome Settings

In the Chrome Settings menu select the Open your computer's proxy settings option (Figure 10).



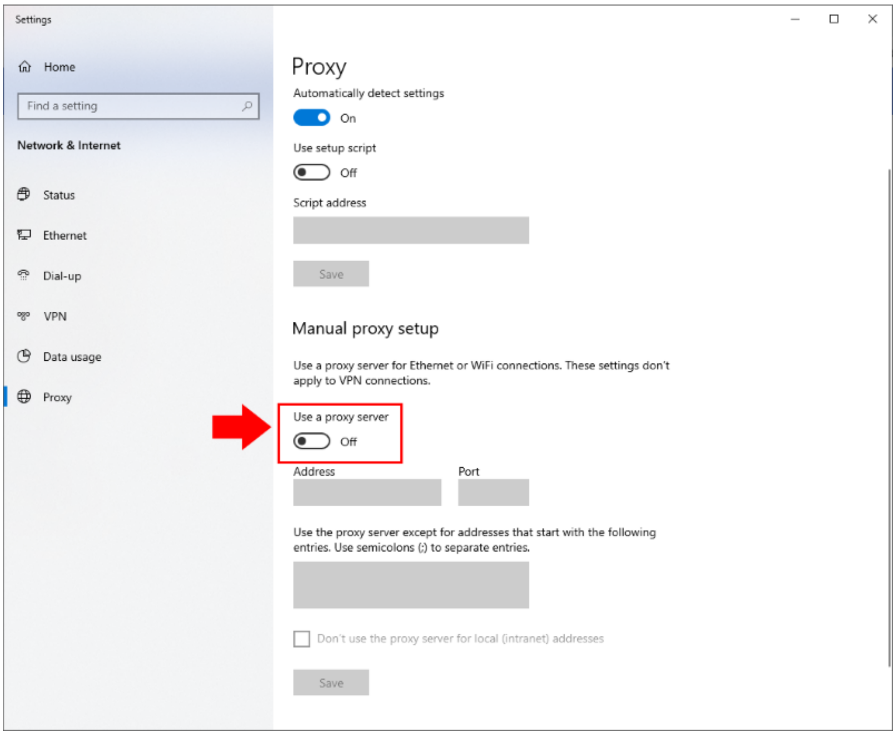
Continue running background apps when Google Chrome is closed

Use hardware acceleration when available

Open your computer's proxy settings

**Figure 10** - Open proxy settings

Finally in the pop up windows Proxy settings menu make sure the Use proxy server option is turned off (Figure 11).



Proxy

Automatically detect settings

On

Use setup script

Off

Script address

Save

Manual proxy setup

Use a proxy server for Ethernet or WiFi connections. These settings don't apply to VPN connections.

Use a proxy server

Off

Address

Port

Use the proxy server except for addresses that start with the following entries. Use semicolons (;) to separate entries.

□ Don't use the proxy server for local (intranet) addresses

Save

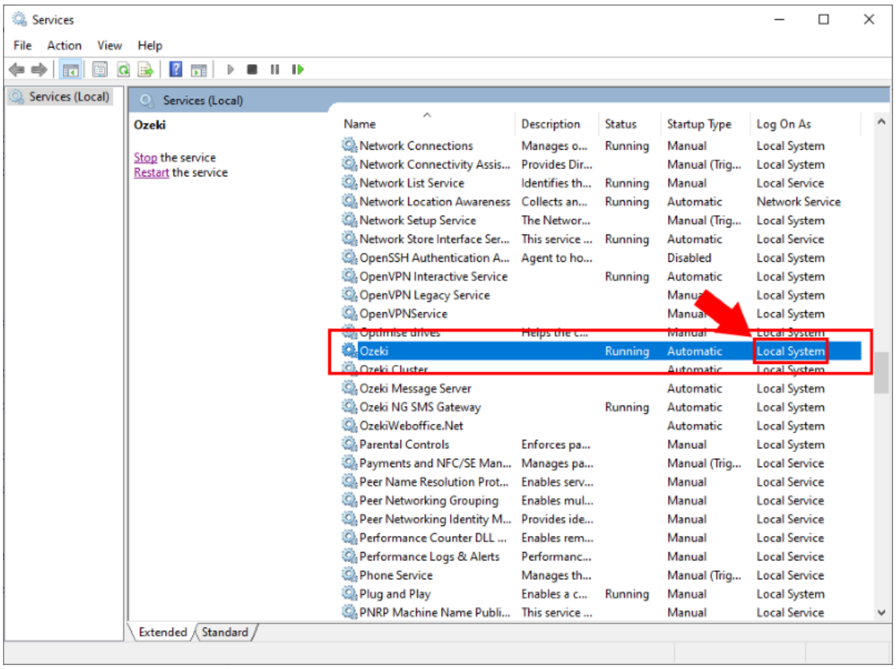
**Figure 11** - Windows proxy settings

## Disable your anti-virus software

Disable your anti-virus software to make sure it doesn't block HTTPS access.

## Check Windows services to see if Ozeki runs as Administrator

Open the Services menu from the Windows and make sure that the Ozeki service is Running as the Local System user as the Figure 12 shows.



Ozeki

Stop the service
Restart the service

| Name | Description | Status | Startup Type | Log On As |
|---|---|---|---|---|
| Network Connections | Manages o... | Running | Manual | Local System |
| Network Connectivity Assis... | Provides Dir... | | Manual (Trig... | Local System |
| Network List Service | Identifies th... | Running | Manual | Local Service |
| Network Location Awareness | Collects an... | Running | Automatic | Network Service |
| Network Setup Service | The Networ... | | Manual (Trig... | Local System |
| Network Store Interface Ser... | This service ... | Running | Automatic | Local Service |
| OpenSSH Authentication A... | Agent to ho... | | Disabled | Local System |
| OpenVPN Interactive Service | | Running | Automatic | Local System |
| OpenVPN Legacy Service | | | Manual | Local System |
| OpenVPNService | | | Manual | Local System |
| Optimise drives | Helps the c... | | Manual | Local System |
| Ozeki | | Running | Automatic | Local System |
| Ozeki Cluster | | | Automatic | Local System |
| Ozeki Message Server | | | Automatic | Local System |
| Ozeki NG SMS Gateway | | Running | Automatic | Local System |
| OzekiWeboffice.Net | | | Automatic | Local System |
| Parental Controls | Enforces pa... | | Manual | Local System |
| Payments and NFC/SE Man... | Manages pa... | | Manual (Trig... | Local Service |
| Peer Name Resolution Prot... | Enables serv... | | Manual | Local Service |
| Peer Networking Grouping | Enables mul... | | Manual | Local Service |
| Peer Networking Identity M... | Provides ide... | | Manual | Local Service |
| Performance Counter DLL ... | Enables rem... | | Manual | Local Service |
| Performance Logs & Alerts | Performanc... | | Manual | Local Service |
| Phone Service | Manages th... | | Manual (Trig... | Local Service |
| Plug and Play | Enables a c... | Running | Manual | Local System |
| PNRP Machine Name Publi... | This service ... | | Manual | Local Service |

**Figure 12** - Windows Services

# How to uninstall Ozeki 10 SMS Gateway

***This simple guide is going to demonstrate what actions that you need to perform to uninstall Ozeki SMS Gateway from your computer. To uninstall the SMS Gateway from your computer, you need to uninstall the software from the application list in Windows Settings, and then, you also need to remove the remaining files from the installation folder of the SMS Gateway. The document contains a step by step guide with each step demonstrated with a short description and a screenshot as well.***

## Step 1 - Open the Windows Settings menu

If you would like to uninstall Ozeki SMS Gateway from your computer, the first thing that you need to do is to open the Settings menu of your Windows computer. This Settings menu can be easily reached using the Start menu. So, just click on the icon of the Start menu in the bottom left corner, and as you can see it in Figure 1, select the Settings menu from the list.

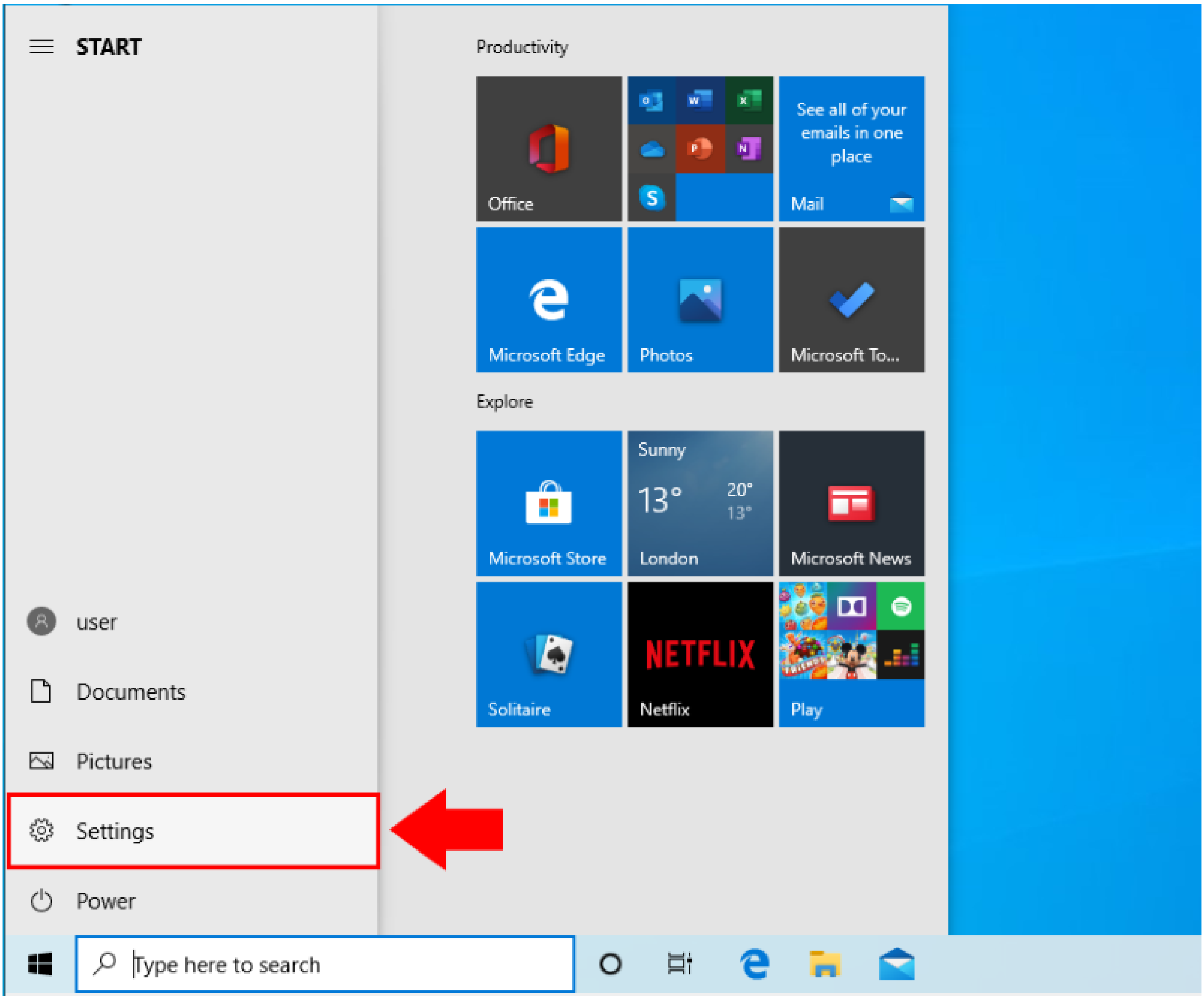


Figure 1 - Open Settings from Windows start menu

## Step 2 - Select Apps in the Settings menu

This Settings menu is the place where you can configure your computer, how it should behave and you can manage the applications installed on your computer from here as well. To reach the list of all installed applications, you need to select the Apps menu. To open the application list, just click on the icon of the Apps menu as Figure 2 shows that.

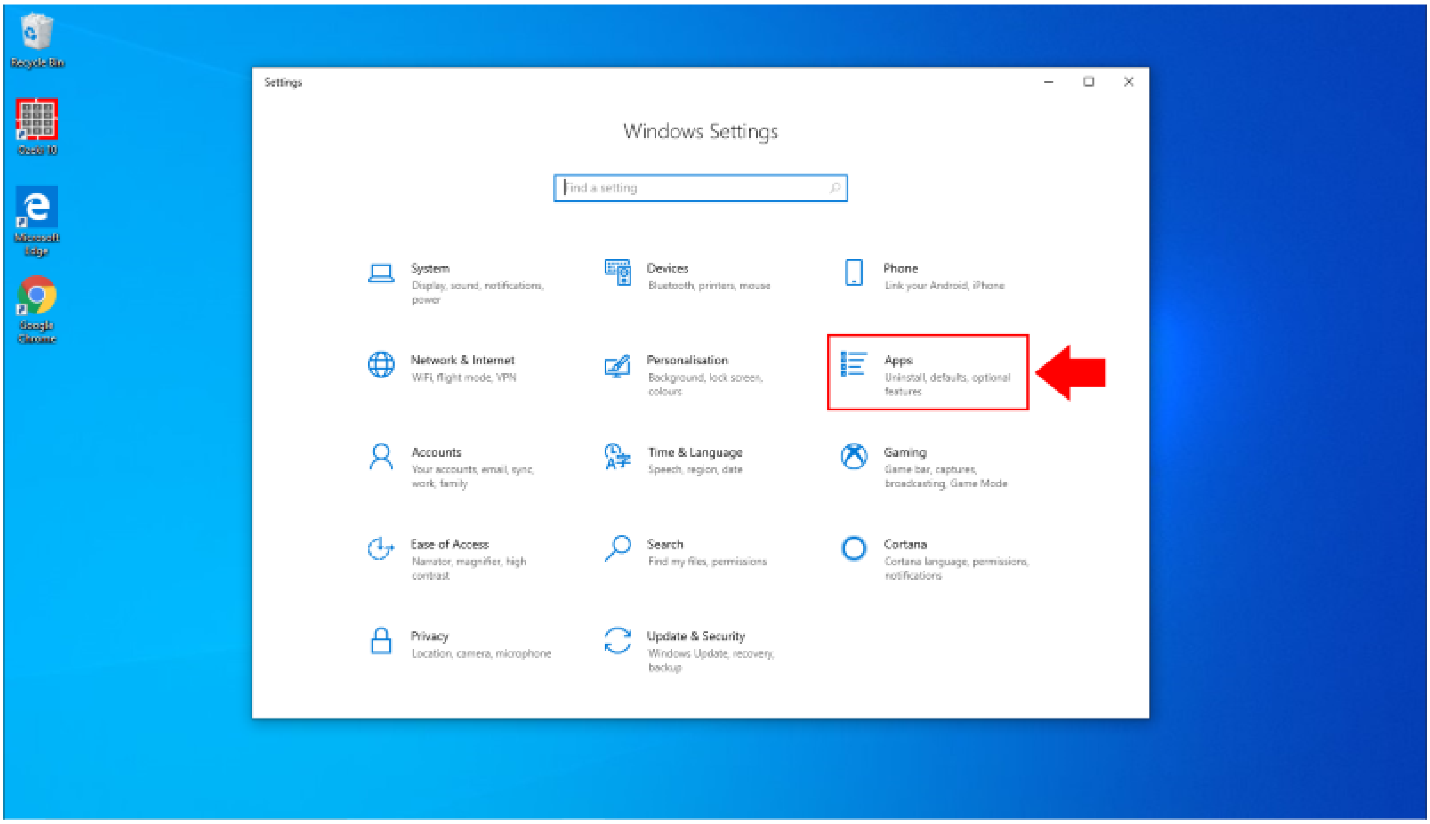
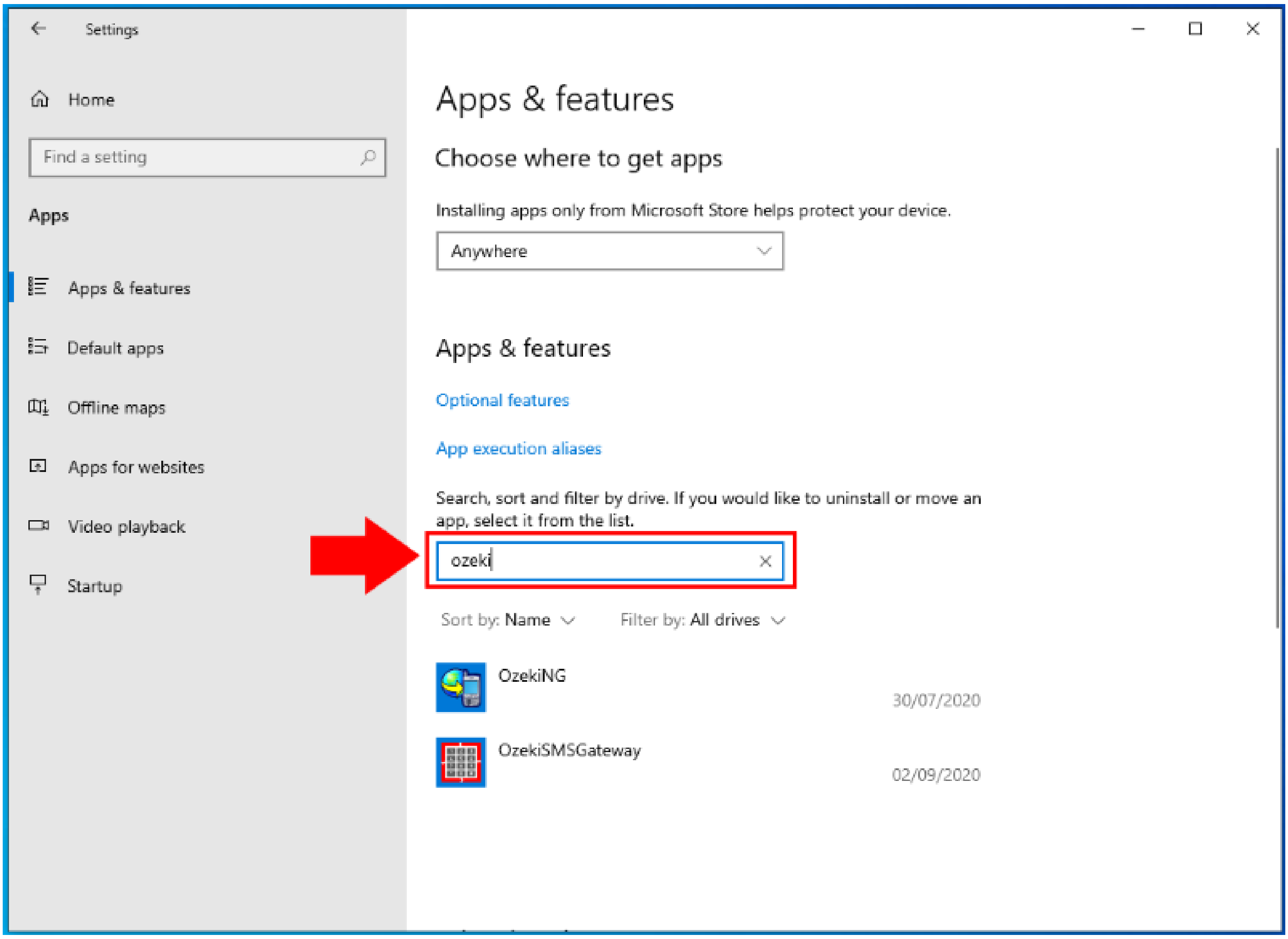

Recycle Bin
Ozeki 10
Microsoft Edge
Google Chrome
Settings
Windows Settings
Find a setting
System
Display, sound, notifications, power
Devices
Bluetooth, printers, mouse
Phone
Link your Android, iPhone
Network & Internet
Wi-Fi, flight mode, VPN
Personalisation
Background, lock screen, colours
Apps
Uninstall, defaults, optional features
Accounts
Your accounts, email, sync, work, family
Time & Language
Speech, region, date
Gaming
Game bar, captures, broadcasting, Game Mode
Ease of Access
Narrator, magnifier, high contrast
Search
Find my files, permissions
Cortana
Cortana language, permissions, notifications
Privacy
Location, camera, microphone
Update & Security
Windows Update, recovery, backup

Figure 2 - Select Apps in the Settings menu

## Step 3 - Search for Ozeki SMS Gateway from the list

In the Apps menu, you can uninstall any of your applications. If you have got many applications installed on your computer, it can take a long time to find the exact application that you would like to uninstall. So, to solve that, you can simply search for applications, like in Figure 3, just type 'ozeki' in the search textbox, and the installed Ozeki applications show up as the result of the search.



Apps & features

Choose where to get apps

Installing apps only from Microsoft Store helps protect your device.

Anywhere

Apps & features

Optional features

App execution aliases

Search, sort and filter by drive. If you would like to uninstall or move an app, select it from the list.

ozeki

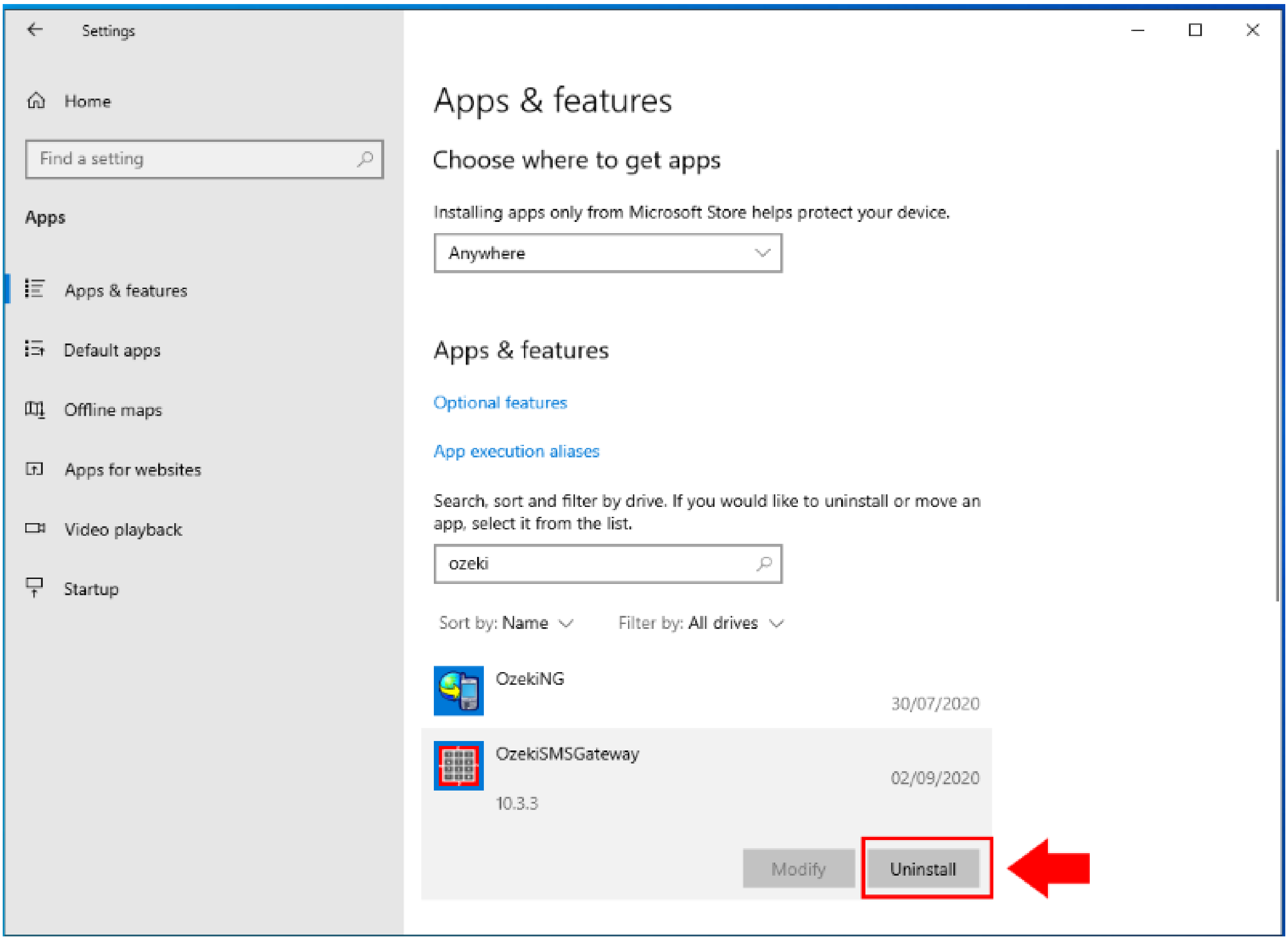
Sort by: Name  Filter by: All drives

| | |
|---|---|
| OzekiNG | 30/07/2020 |
| OzekiSMSGateway | 02/09/2020 |

Figure 3 - Search Ozeki in the installed apps

# Step 4 - Start the uninstallation of Ozeki SMS Gateway

After you found the installed Ozeki sorfware from your list of installed applications, now you can start the uninstallation process. For that, just click on the Ozeki SMS Gateway application to exapand its menu with the available options. Here, as Figure 4 demonstrates it, you just have to click on the Uninstall button to start the uninstallation process of the SMS Gateway.



Settings

Home

Find a setting

Apps

Apps & features

Default apps

Offline maps

Apps for websites

Video playback

Startup

Apps & features

Choose where to get apps

Installing apps only from Microsoft Store helps protect your device.

Anywhere

Apps & features

Optional features

App execution aliases

Search, sort and filter by drive. If you would like to uninstall or move an app, select it from the list.

ozeki

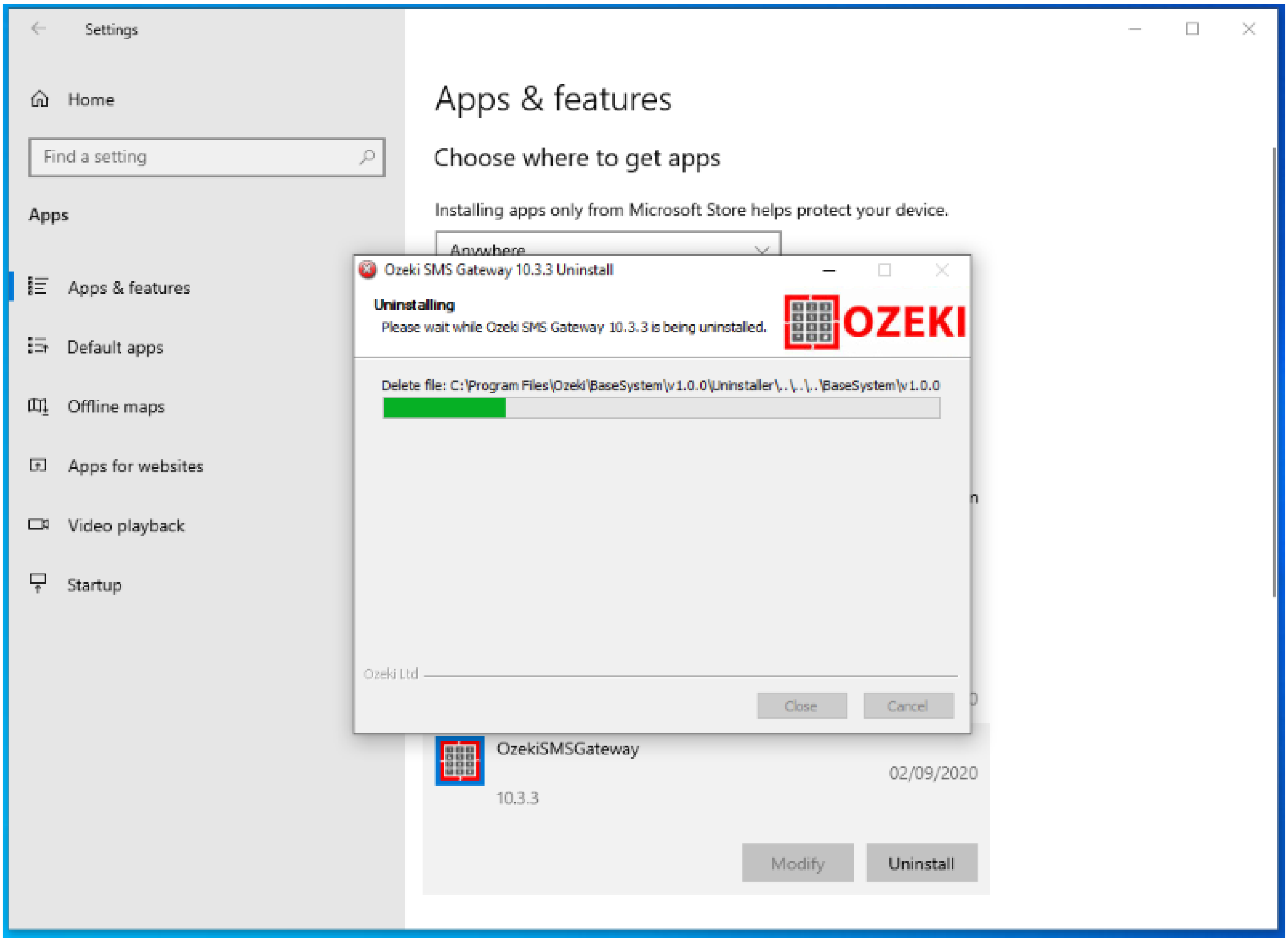
Sort by: Name  Filter by: All drives

| | |
|---|---|
| OzekiNG | 30/07/2020 |
| OzekiSMSGateway<br>10.3.3 | 02/09/2020 |

Figure 4 - Uninstall Ozeki software

# Step 5 - Uninstall Ozeki SMS Gateway

As soon as you clicked on the Uninstall button, a new window shows up that starts the uninstallation process as you can see it in Figure 5. At this point, you don't need to do anything, you can just wait until the uninstallation process finishes. When the SMS Gateway is uninstalled, you just have to click on the Finish button in the uninstall window to close it.



Apps & features

Choose where to get apps

Installing apps only from Microsoft Store helps protect your device.

Anywhere

Uninstalling

Please wait while Ozeki SMS Gateway 10.3.3 is being uninstalled.

Delete file: C:\Program Files\Ozeki\BaseSystem\v1.0.0\Uninstaller\..\..\..\BaseSystem\v1.0.0

OzekiSMSGateway 10.3.3 02/09/2020

Figure 5 - Uninstallation process

# Step 6 - Remove all files from Ozeki folder

The uninstallation process didn't remove every Ozeki file from your computer. Some files are left in the installation folder of Ozeki SMS Gateway. These files need to be deleted manually. So, to do that, open the File Explorer and navigate to the folder, where you installed the SMS Gateway. This folder is 'C:\Program Files\Ozeki' by default. Here, you will be able to see two folders which are Common and Data. As you can see it in Figure 6, you need to mark out both folders, and with a right-click, select the Delete option. You might need admin permissions to perform this action.

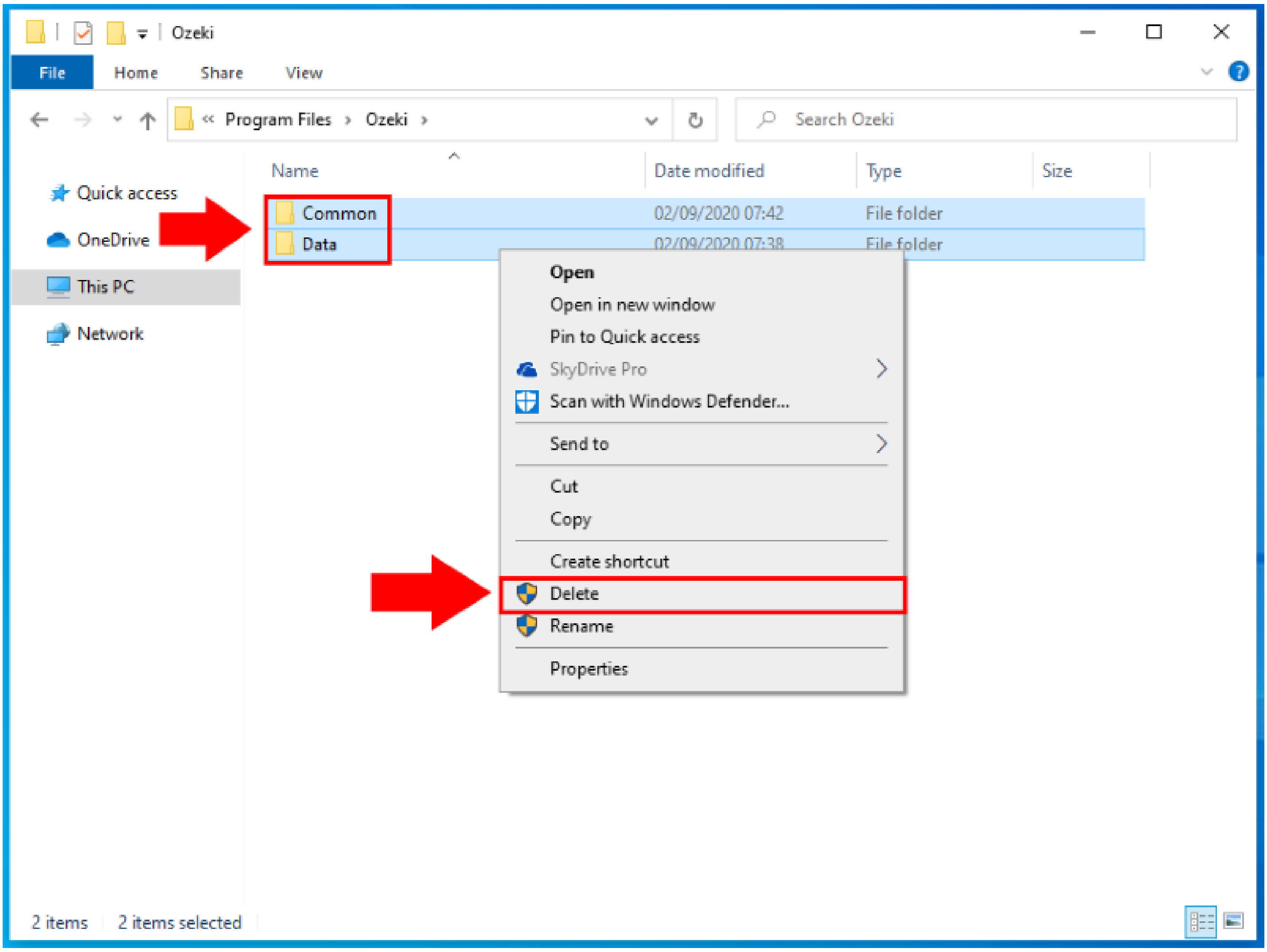


Figure 6 - Delete folders from the Ozeki folder

# Mobile network connections

***The Ozeki SMS Gateway enables you to connect to SMS service providers over the Internet or through wireless connections. The following guides give you information on how to setup connections to various mobile networks. These mobile networks offer IP SMS connections based on SMS protocols. The most popular SMS protocol used by the SMS service providers is the Short Message Peer to Peer (SMPP) protocol. You might also find SMS providers who use UCP or CIMD2, and some will offer HTTP, which is not very efficient when it comes to high volume messaging.***

**SMPP Mobile Network Connection**
This guide gives you information about how to setup an SMPP SMS client connection. SMPP stands for Short Message Peer to Peer protocol and is a very popular choice among SMS service providers. The Ozeki system offers an excellent, high performance implementation of this great protocol.
Learn More

**UCP/EMI SMS Client Connection**
By following the guide on this page, you will get a brief introduction to UCP/EMI SMS client connection. The Universal Computer Protocol and its extension, External Machine Interface is a greate choice to connect to short message service centers for mobile phones.
Learn More

**CIMD2 SMS Client Connection**
Learn about how you can set up a CIMD2 SMS client connection in Ozeki SMS Gateway. The Computer Interface Message Disctribution protocol was introduced by Nokia. The CIMD2 interface transfers messages between applications and the Nokia Short Message Service Center.
Learn More

**HTTP SMS Client Connection**
You can get useful information about how you can set up a HTTP SMS client connection in Ozeki SMS Gateway by following the guide here. By using this solution, you will be able to communicate with web applications by using the HTTP protocol.
Learn More

**Android SMS Client Connection**
This guide gives you information about how you can set up an Android SMS client connection in Ozeki SMS Gateway. By downloading the Android SMPP Gateway to your smartphone, you can use it as an SMPP server, and connect to that smartphone with an SMPP client connection in SMS Gateway.
Learn More

**GSM Modem Link**
The following guide is about to demonstrate how you can connect your GSM modem to your PC and configure it with Ozeki SMS Gateway. The high speed, professional cellular modem works well with Ozeki SMS Gateway.
Learn More

**Nexmo SMS Service Provider**
Learn about how you can establish a Nexmo service provider connection in Ozeki SMS Gateway. The Nexmo service provide allows you to send and receive text and binary SMS messages. Ozeki provides the implementation of the Nexmo API.
Learn More

**MessageBird SMS Service Provider**
By following this guide, you will be able to learn about how you can set up a MessageBird service provider connection in Ozeki SMS Gateway. MessageBird is a great tool for sending or receiving SMS messages. You can easily connect your MessageBird account to SMS Gateway and use for managing your SMS messages.
Learn More

# How to connect to an OZX service

***This guide is about OZX client connection setup. It defines what an OZX connection is, what does OZX stand for and how you can use an OZX client connection to connect your Ozeki SMS gateway system directly to the Short Message Service Center (SMSC) of a mobile network operator over the Internet.***

## What is an OZX connection?

An OZX client is a software that allows you to connect to an SMS service provider on the Internet. An OZX client, such as Ozeki SMS Gateway, uses the OZX protocol to send and receive SMS text messages.

## How to connect an OZX connection

The following short video shows the steps you need to create to setup an OZX client connection in Ozeki 10 SMS Gateway. The video starts with the login form, and takes you all the way to sending your first SMS test message over the newly created SMPP connection.

## Connection steps

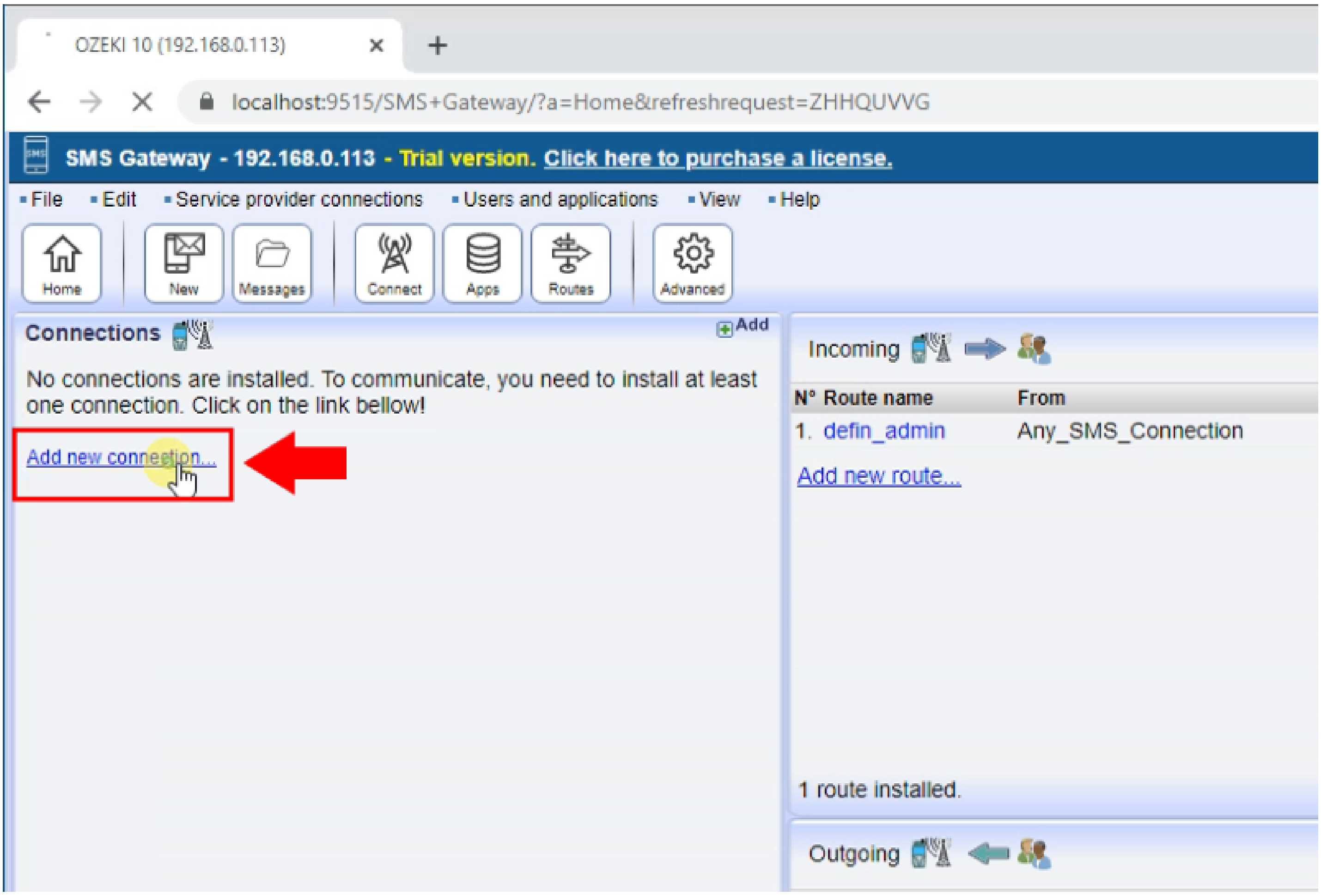
- Open https://localhost:9515 in your browser
- Login using your username and password
- Click on "Add new connection"
- Select "OZX client"
- Enter the OZX host name and port
- Enter the telephone number
- Click ok and send a test message

## Detailed setup instructions

Setting up an SMS connection in Ozeki SMS Gateway is a relatively simple procedure. You need to login to the SMS gateway using a web browser as administrator, and you need to perform a few simple steps. For configuration we recommend to login using the administrator account. The administrator account username is "admin", and the password is the one you provided during install.

## How to create a new OZX connection

To create a new OZX connection after login, you need to click on the '**Add new connection**' link in the management console of the Ozeki 10 SMS Gateway app. This will bring up a list of available protocols. You will have to select OZX client from the list. Note, that an **OZX client** connection is used if you wish to connect your SMS gateway to an SMS service provider over the Internet. (If you wish to provide an SMS service, and you want your customers to connect to your SMS gateway over OZX, you need to setup an OZX user account and you need to configure an OZX service.)


OZEKI 10 (192.168.0.113)
localhost:9515/SMS+Gateway/?a=Home&refreshrequest=ZHHQUVVG
SMS Gateway - 192.168.0.113 - Trial version. Click here to purchase a license.
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Connections
Add
Incoming
Outgoing

No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

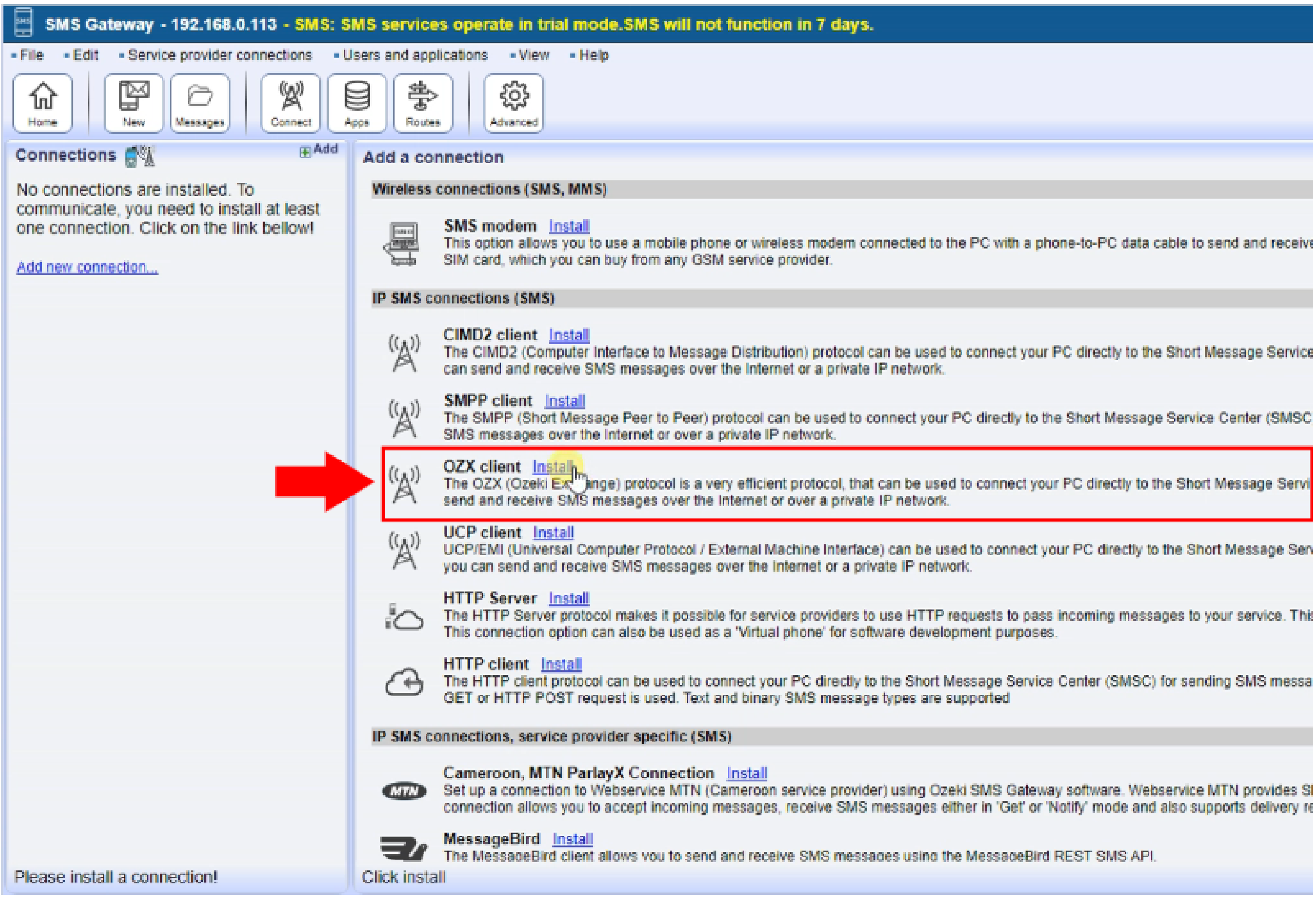
| N° | Route name | From |
|---|---|---|
| 1. | defin_admin | Any_SMS_Connection |

Add new route...

1 route installed.

Figure 1 - Add new connection

Select the '**OZX Client Connection**' from the list and click on install next to it.


SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Connections
Add

No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

Add a connection

Wireless connections (SMS, MMS)

SMS modem Install
This option allows you to use a mobile phone or wireless modem connected to the PC with a phone-to-PC data cable to send and receive
SIM card, which you can buy from any GSM service provider.

IP SMS connections (SMS)

CIMD2 client Install
The CIMD2 (Computer Interface to Message Distribution) protocol can be used to connect your PC directly to the Short Message Service
can send and receive SMS messages over the Internet or a private IP network.

SMPP client Install
The SMPP (Short Message Peer to Peer) protocol can be used to connect your PC directly to the Short Message Service Center (SMSC
SMS messages over the Internet or over a private IP network.

OZX client Install
The OZX (Ozeki Exchange) protocol is a very efficient protocol, that can be used to connect your PC directly to the Short Message Servi
send and receive SMS messages over the Internet or over a private IP network.

UCP client Install
UCP/EMI (Universal Computer Protocol / External Machine Interface) can be used to connect your PC directly to the Short Message Ser
you can send and receive SMS messages over the Internet or a private IP network.

HTTP Server Install
The HTTP Server protocol makes it possible for service providers to use HTTP requests to pass incoming messages to your service. This
This connection option can also be used as a 'Virtual phone' for software development purposes.

HTTP client Install
The HTTP client protocol can be used to connect your PC directly to the Short Message Service Center (SMSC) for sending SMS messa
GET or HTTP POST request is used. Text and binary SMS message types are supported

IP SMS connections, service provider specific (SMS)

Cameroon, MTN ParlayX Connection Install
Set up a connection to Webservice MTN (Cameroon service provider) using Ozeki SMS Gateway software. Webservice MTN provides SI
connection allows you to accept incoming messages, receive SMS messages either in 'Get' or 'Notify' mode and also supports delivery re

MessageBird Install
The MessageBird client allows you to send and receive SMS messages using the MessageBird REST SMS API.

Please install a connection!

Click install

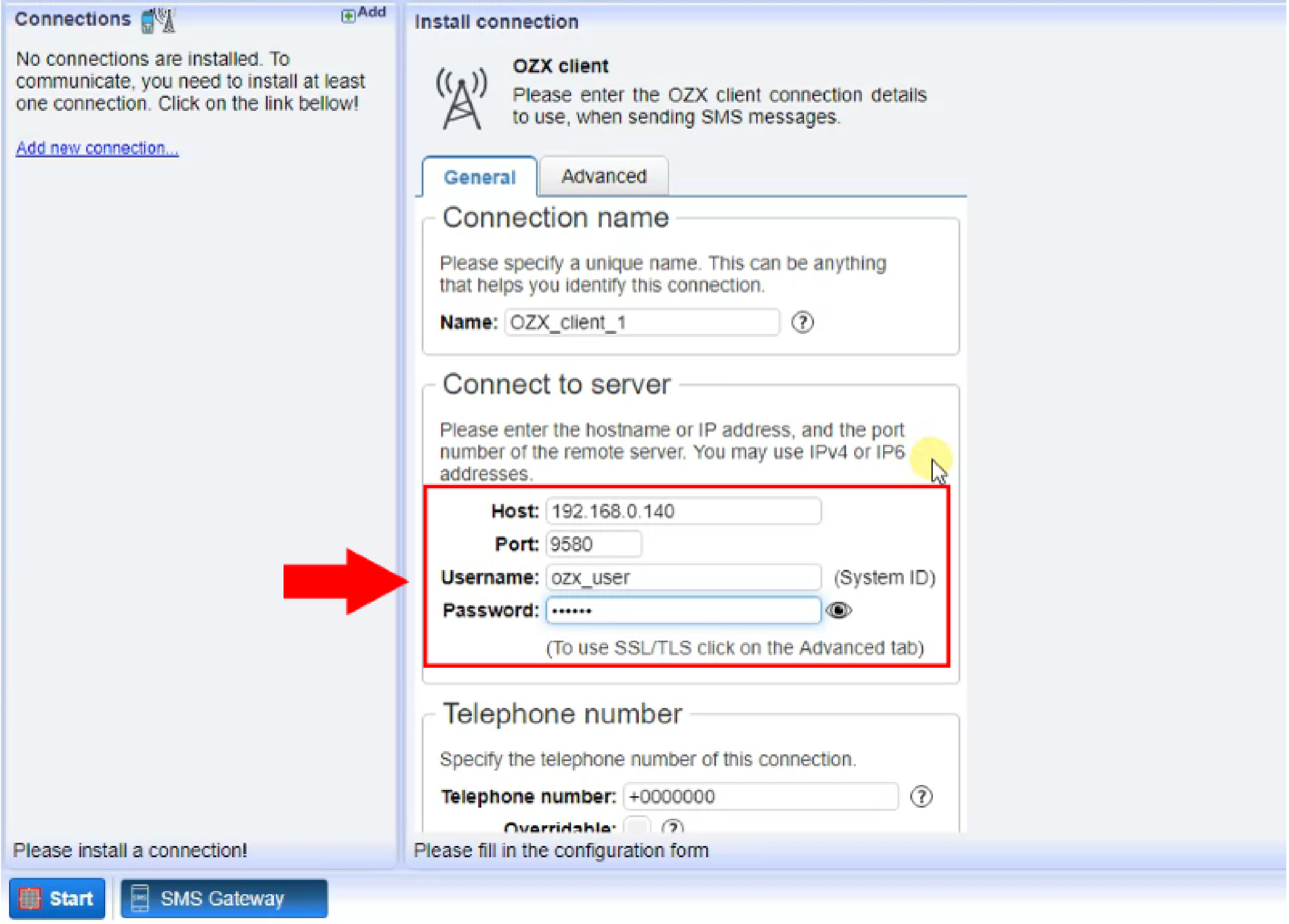
Figure 2 - Install OZX client

# Configure your OZX connection

In order to configure your OZX connection, you need to provide the host name and port number of the OZX service, your OZX credentials and your must specify telephone number associated with this connection. There could be more than one phone numbers associated with this connection. In this case provide the first one and check the overridable checkbox. If this checkbox is checked, you will be able to use all phone numbers as sender IDs.

**OZX connection configuration steps:**

- Select the General tab on the OZX connection form
- Give a name to this OZX connection
- Enter the OZX hostname
- Enter the OZX port number
- Provide your OZX username
- Enter your password
- Assign a telephone number to this OZX connection
- Click OK



Connections

No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

Please install a connection!

Install connection

OZX client

Please enter the OZX client connection details to use, when sending SMS messages.

Connection name

Please specify a unique name. This can be anything that helps you identify this connection.

Name: OZX_client_1

Connect to server

Please enter the hostname or IP address, and the port number of the remote server. You may use IPv4 or IP6 addresses.

Host: 192.168.0.140

Port: 9580

Username: ozx_user (System ID)

Password: ••••••

(To use SSL/TLS click on the Advanced tab)

Telephone number

Specify the telephone number of this connection.

Telephone number: +0000000

Overridable: □

Please fill in the configuration form

Figure 3 - OZX connection details

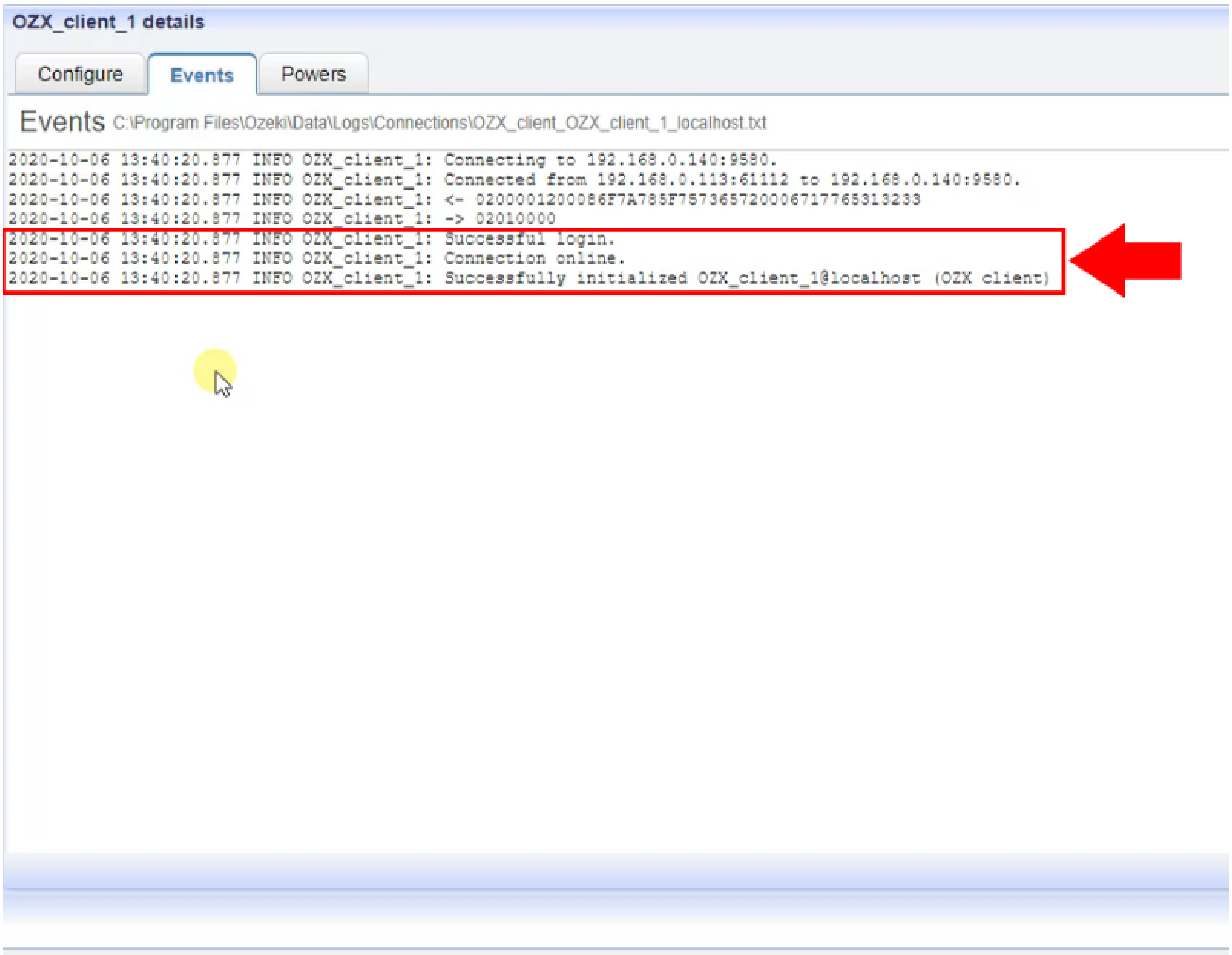
It is important to mention that the OZX hostname and port plus the username and password are provided by your SMS service provider. For example if you contact Vodafone, and ask for an OZX SMS service, you will sign a contract with them, and often an attachment of this contract will contain the connection following information. If you subscribe to an SMS service on-line, you will likely find this information in the on-line control panel of your SMS service provider. Of course you can always ask your SMS service provider in e-mail and ask what are the OZX server connection details.

# Check the OZX log

Once the OZX connection is configured, you should check the OZX connection log to see if the system connected properly to your SMS service provider. If the system connected properly you will see the "Connetion online" log entry. The SMS connection logs can be found in the following directory in Windows: C:\Program Files\Ozeki\Data\Logs\Connections\. In this Event tab of the OZX connection's detalis page, you can see a preview of the log. This page contains the last 100 log entries. Note that the OZX log files are rotated to save disk space.

**How to view the OZX connection log:**

- Open the OZX connection's details page
- Click on the Events tab
- Click on the eye icon next to the Events title
- Copy the file name next to the Events title
- Open notepad
- Paste the filename into the File/Open dialog in notepad
- Click OK to open the OZX log file
- Use F3 to find the date you are interested in
- Search for "ERRO" to find errors in the log



Events C:\Program Files\Ozeki\Data\Logs\Connections\OZX_client_OZX_client_1_localhost.txt

```
2020-10-06 13:40:20.877 INFO OZX_client_1: Connecting to 192.168.0.140:9580.
2020-10-06 13:40:20.877 INFO OZX_client_1: Connected from 192.168.0.113:61112 to 192.168.0.140:9580.
2020-10-06 13:40:20.877 INFO OZX_client_1: <- 0200001200086F7A785F7573657200067177656313233
2020-10-06 13:40:20.877 INFO OZX_client_1: -> 02010000
2020-10-06 13:40:20.877 INFO OZX_client_1: Successful login.
2020-10-06 13:40:20.877 INFO OZX_client_1: Connection online.
2020-10-06 13:40:20.877 INFO OZX_client_1: Successfully initialized OZX_client_1@localhost (OZX client)
```

Figure 4 - OZX client connected

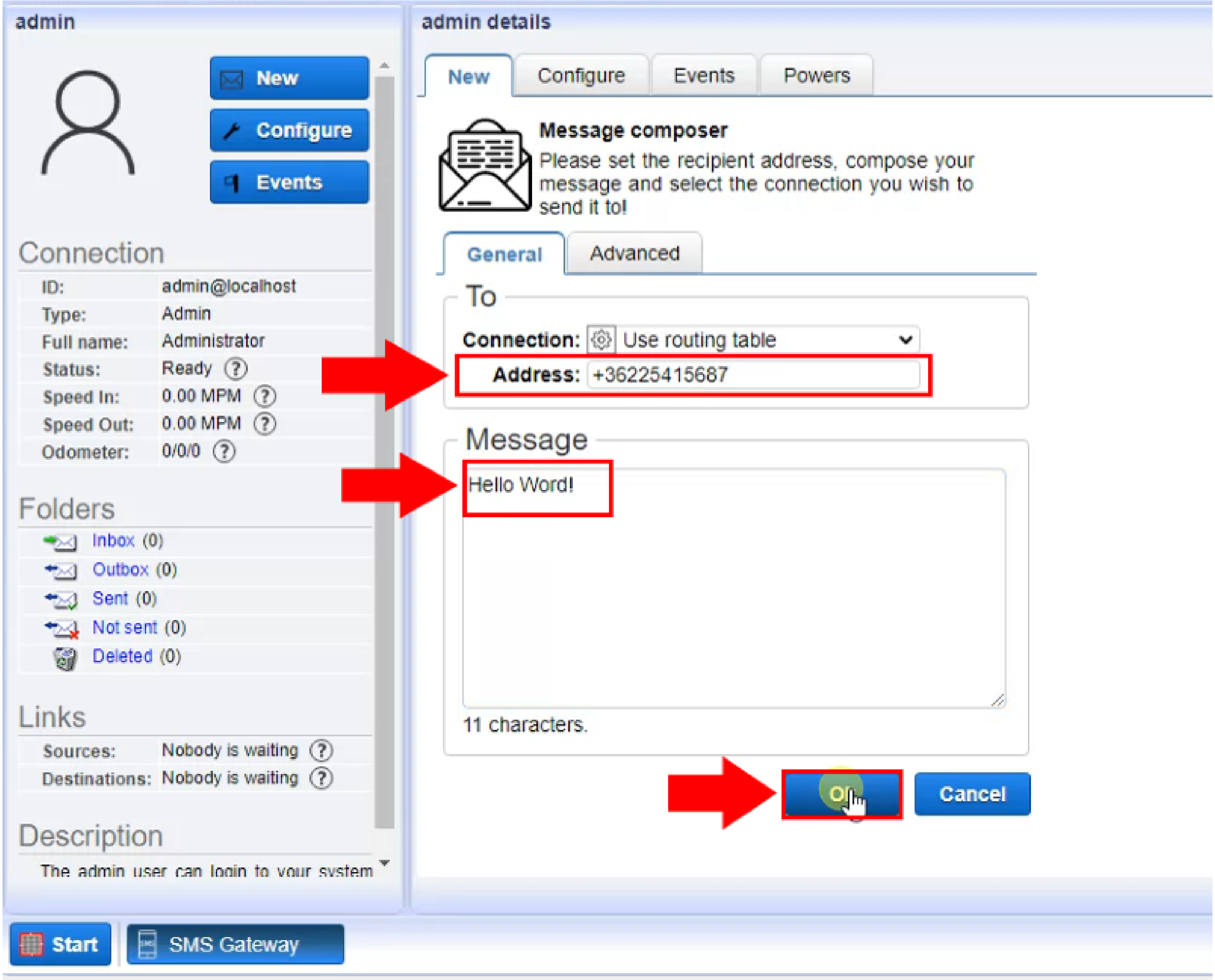
# Send a test SMS message

Once your connection is connected, you can check to see if it is ready to deliver SMS messages by sending a test SMS. To send a test SMS message, you need to open the Admin user, and you need to enter the phone number and message text. It is recommended to provide the phone number in international format. This means the phone number should start with a plus sign followed by a country code. If your local phone number starts with a 0, it is likely that you will need to drop the 0 prefix. For example if your UK phone number 07958663698, you would send the test SMS to +447958663698.

**How to send a test OZX message:**

- Navigate to the Admin connection's details page
- Select the New tab
- Enter the recipient phone number
- Make sure the phone number is in international format
- Enter the message text
- Make sure the message text is less than 160 characters
- Click on the Send button
- Check the OZX logs

You might ask why should the message text be less than 160 characters. This is because GSM system was designed to send 160 character long text messages. If a message is longer, it will be split into multiple message segments, and will be delivered in more than one SMS message.



Connection

| | |
|---|---|
| ID: | admin@localhost |
| Type: | Admin |
| Full name: | Administrator |
| Status: | Ready |
| Speed In: | 0.00 MPM |
| Speed Out: | 0.00 MPM |
| Odometer: | 0/0/0 |

Folders

Inbox (0)
Outbox (0)
Sent (0)
Not sent (0)
Deleted (0)

Links

| | |
|---|---|
| Sources: | Nobody is waiting |
| Destinations: | Nobody is waiting |

Description

The admin user can login to your system

Message composer

Please set the recipient address, compose your message and select the connection you wish to send it to!

To

Connection: Use routing table

Address: +36225415687

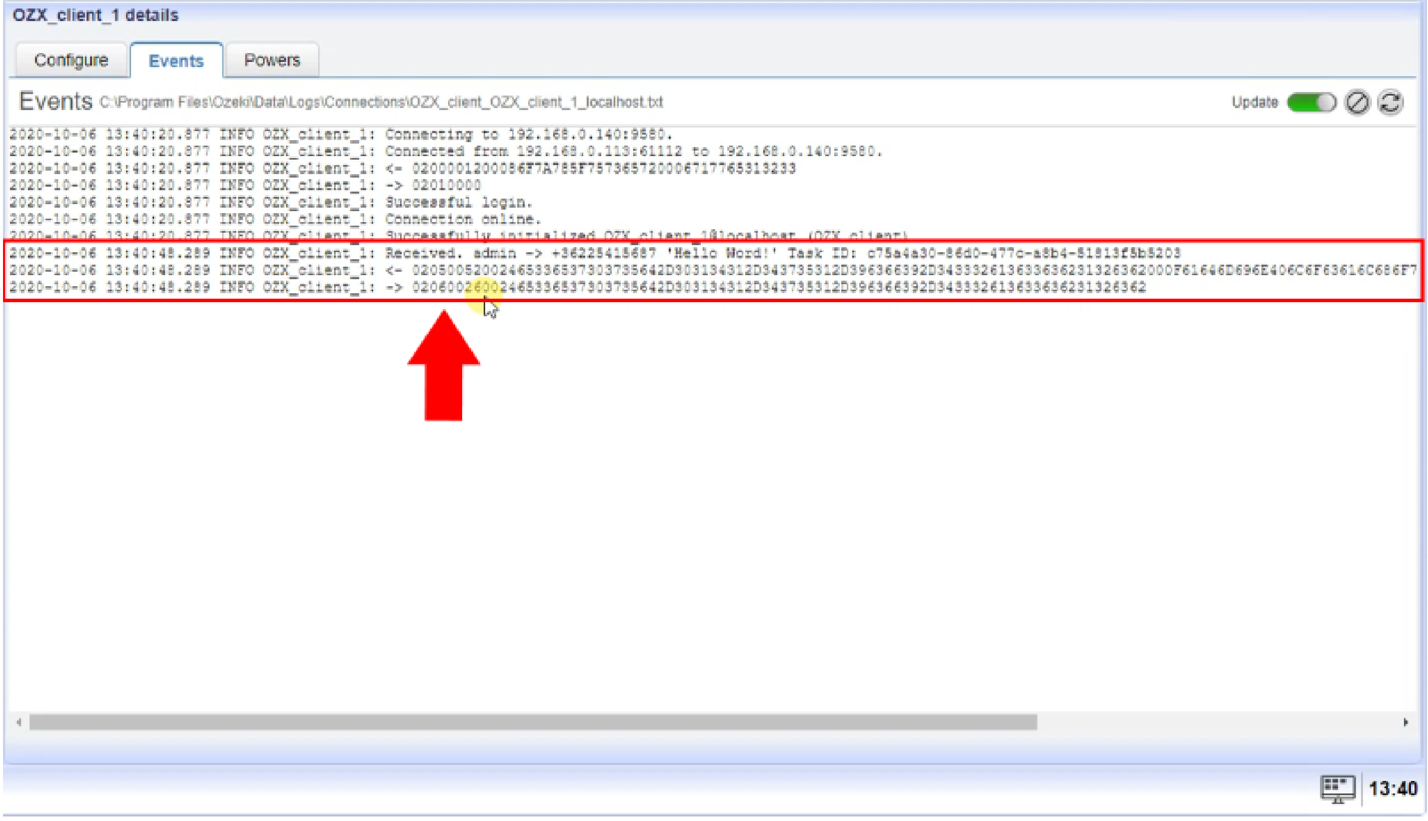
Message

Hello Word!

11 characters.

Figure 5 - Send test message

After the message is successfully sent, you should check the OZX logs. The OZX logs will reveal the low level OZX messages, that are used to pass the message content to the SMS service provider. These messages are also called an OZX PDU (OZX protocol data unit). If there is a problem with message submission, your SMS service provider will ask for the OZX logs. In this case, you need to send the OZX PDUs to them. For every SMS submission, two OZX PDUs will appear in the log.



Events C:\Program Files\Ozeki\Data\Logs\Connections\OZX_client_OZX_client_1_localhost.txt

```
2020-10-06 13:40:20.877 INFO OZX_client_1: Connecting to 192.168.0.140:9580.
2020-10-06 13:40:20.877 INFO OZX_client_1: Connected from 192.168.0.113:61112 to 192.168.0.140:9580.
2020-10-06 13:40:20.877 INFO OZX_client_1: <- 0200001200086F7A785F757365720006717765313233
2020-10-06 13:40:20.877 INFO OZX_client_1: -> 02010000
2020-10-06 13:40:20.877 INFO OZX_client_1: Successful login.
2020-10-06 13:40:20.877 INFO OZX_client_1: Connection online.
2020-10-06 13:40:20.877 INFO OZX_client_1: Successfully initialized OZX_client_1@localhost (OZX client)
2020-10-06 13:40:48.289 INFO OZX_client_1: Received. admin -> +36225415687 'Hello Word!' Task ID: c75a4a30-86d0-477c-a8b4-51813f5b5203
2020-10-06 13:40:48.289 INFO OZX_client_1: <- 020500520024653365373037356 42D303134312D343735312D396366392D343332613633636231326362000F61646D696E406C6F63616C686F7
2020-10-06 13:40:48.289 INFO OZX_client_1: -> 0206002600246533653730373564 2D303134312D343735312D396366392D343332613633636231326362
```

Figure 6 - Message log

# SMPP client connection

***This guide is about SMPP client connection setup. It defines what an SMPP connection is, what does SMPP stand for and how you can use an SMPP client connection to connect your Ozeki SMS gateway system directly to the Short Message Service Center (SMSC) of a mobile network operator over the Internet.***

## What is an SMPP connection?

An SMPP client is a software that allows you to connect to an SMS service provider on the Internet. An SMPP client, such as Ozeki SMS Gateway, uses the SMPP protocol to send and receive SMS text messages.

## What does SMPP stand for?

SMPP stands for Short Message Peer-to-Peer Protocol. This is an industry standard protocol designed to deliver SMS messages over TCP/IP connections through the Internet. This protocol is implemented by Ozeki SMS Gateway.

## How to connect an SMPP connection

The following short video shows the steps you need to create to setup an SMPP client connection in Ozeki 10 SMS Gateway. The video starts with the login form, and takes you all the way to sending your first SMS test message over the newly created SMPP connection.

## Connection steps

- Open https://localhost:9515 in your browser
- Login using your username and password
- Click on "Add new connection"
- Select "SMPP client"
- Enter the smpp host name and port
- Enter the telephone number
- Click ok and send a test message

## Detailed setup instructions

Setting up an SMS connection in Ozeki SMS Gateway is a relatively simple procedure. You need to login to the SMS gateway using a web browser as administrator, and you need to perform a few simple steps. For configuration we recommend to login using the administrator account. The administrator account username is "admin", and the password is the one you provided during install.

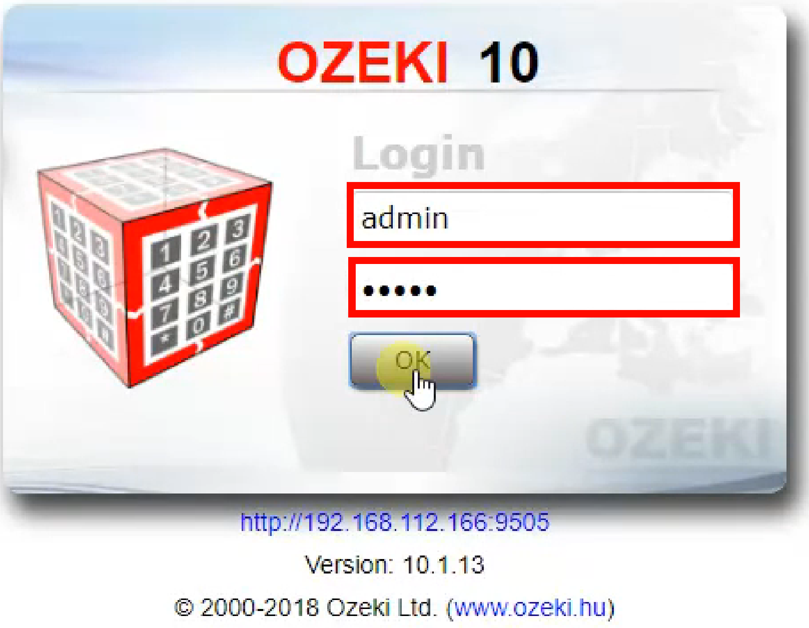
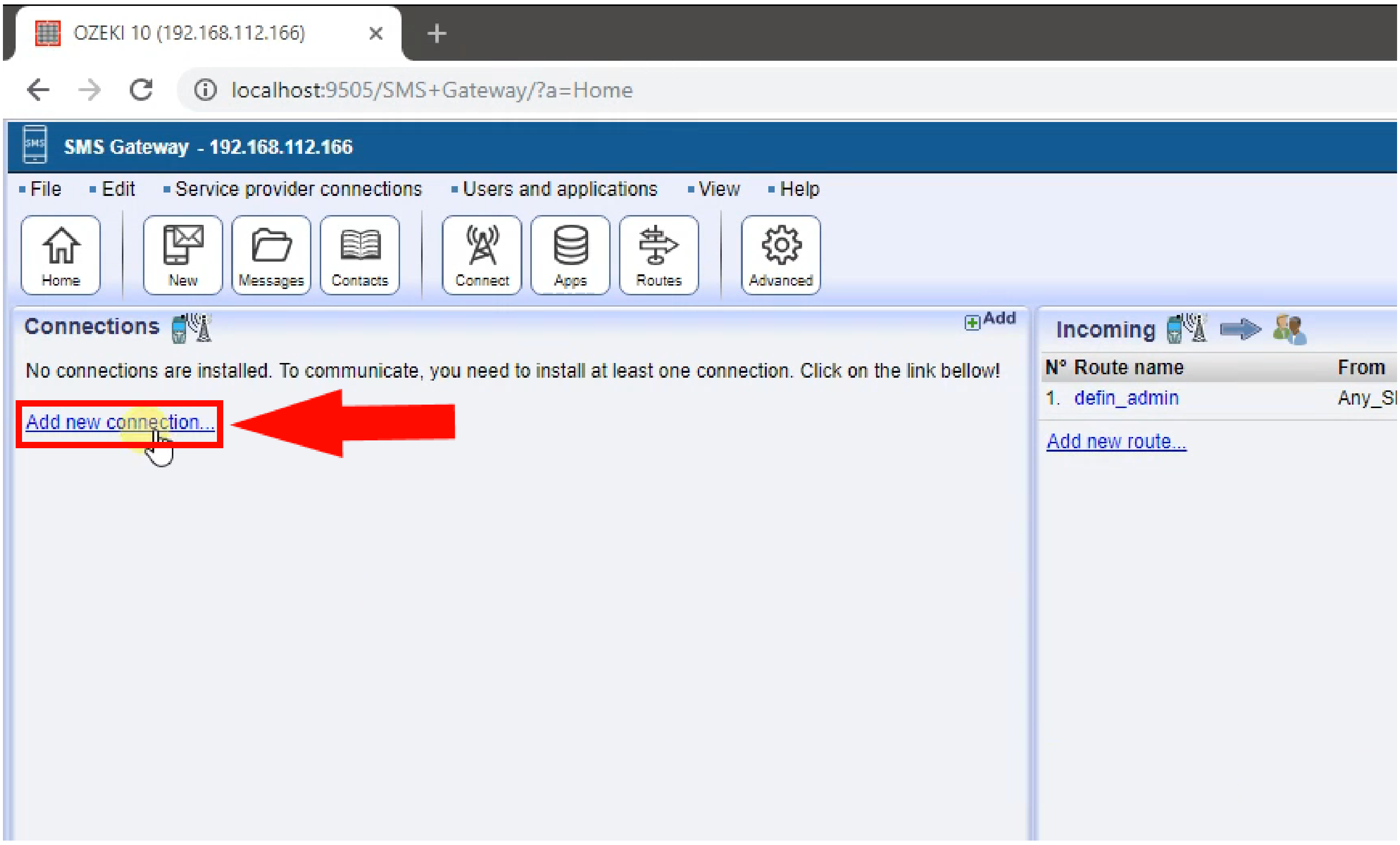

OZEKI 10
Login
admin
•••••
OK
http://192.168.112.166:9505
Version: 10.1.13
© 2000-2018 Ozeki Ltd. (www.ozeki.hu)

Figure 1 - Logging in to Ozeki 10 for SMPP client connection setup

# How to create a new SMPP connection

To create a new SMPP connection after login, you need to click on the '**Add new connection**' link in the management console of the Ozeki 10 SMS Gateway app. This will bring up a list of available protocols. You will have to select SMPP client from the list. Note, that an **SMPP client** connection is used if you wish to connect your SMS gateway to an SMS service provider over the Internet. (If you wish to provide an SMS service, and you want your customers to connect to your SMS gateway over SMPP, you need to setup an SMPP user account and you need to configure an SMPP service.)


OZEKI 10 (192.168.112.166)
localhost:9505/SMS+Gateway/?a=Home
SMS Gateway - 192.168.112.166
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Contacts
Connect
Apps
Routes
Advanced
Connections
Add
Incoming

No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

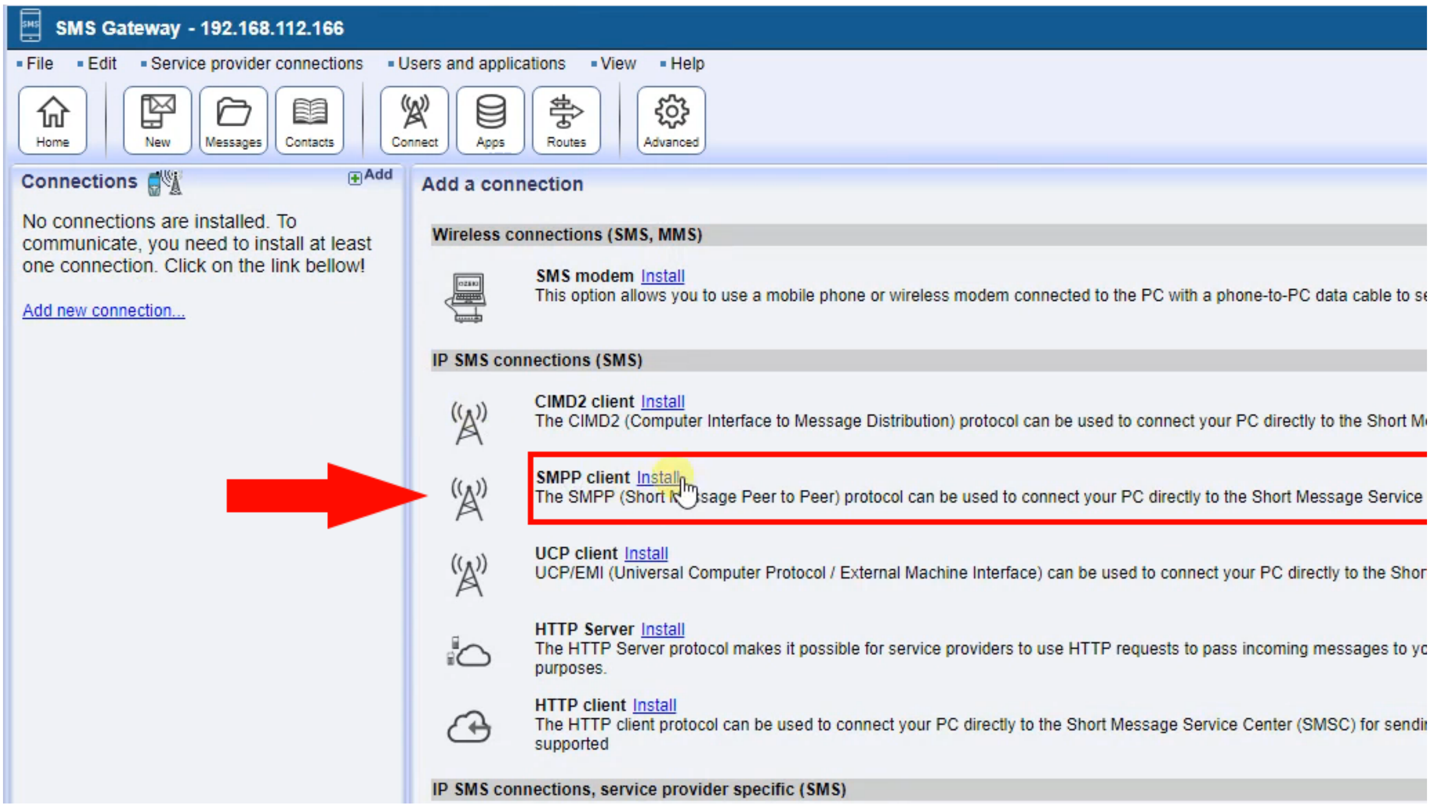
Add new connection...

| N° | Route name | From |
|---|---|---|
| 1. | defin_admin | Any_SI |

Add new route...

Figure 2 - Create a new SMPP connection

Select the '**SMPP Client Connection**' from the list and click on install next to it.



No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

**Wireless connections (SMS, MMS)**

**SMS modem** Install
This option allows you to use a mobile phone or wireless modem connected to the PC with a phone-to-PC data cable to se

**IP SMS connections (SMS)**

**CIMD2 client** Install
The CIMD2 (Computer Interface to Message Distribution) protocol can be used to connect your PC directly to the Short M

**SMPP client** Install
The SMPP (Short M sage Peer to Peer) protocol can be used to connect your PC directly to the Short Message Service

**UCP client** Install
UCP/EMI (Universal Computer Protocol / External Machine Interface) can be used to connect your PC directly to the Shor

**HTTP Server** Install
The HTTP Server protocol makes it possible for service providers to use HTTP requests to pass incoming messages to yo purposes.

**HTTP client** Install
The HTTP client protocol can be used to connect your PC directly to the Short Message Service Center (SMSC) for sendi supported

**IP SMS connections, service provider specific (SMS)**

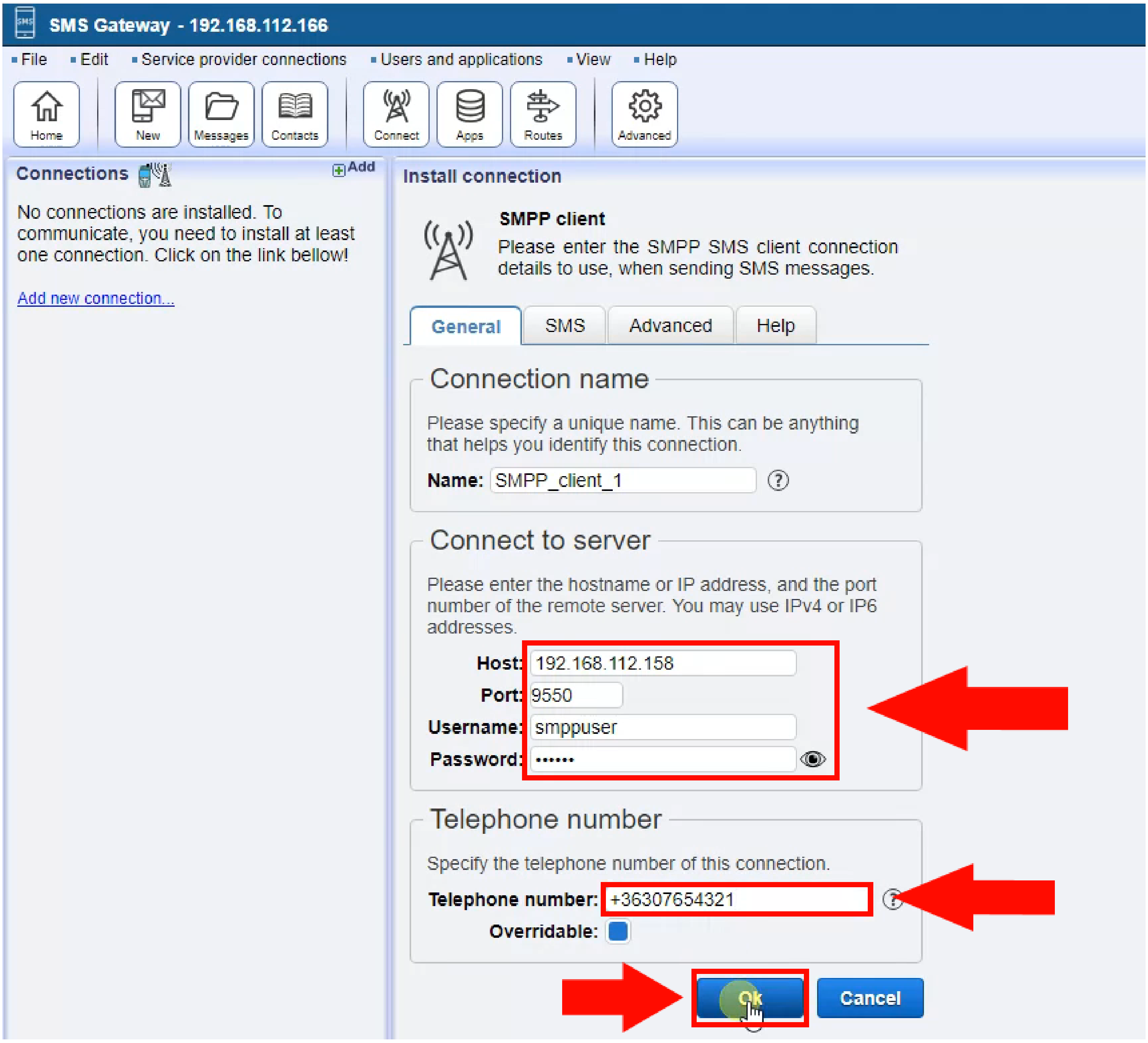
Figure 3 - Installing the SMPP connection

# Configure your SMPP connection

In order to configure your SMPP connection, you need to provide the host name and port number of the SMPP service, your SMPP credentials and your must specify telephone number associated with this connection. There could be more than one phone numbers associated with this connection. In this case provide the first one and check the overridable checkbox. If this checkbox is checked, you will be able to use all phone numbers as sender IDs.

**SMPP connection configuration steps:**

- Select the General tab on the SMPP connection form
- Give a name to this SMPP connection
- Enter the SMPP hostname
- Enter the SMPP port number
- Provide your SMPP username
- Enter your password
- Assign a telephone number to this SMPP connection
- Click OK


SMS Gateway - 192.168.112.166

Connections

No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

Install connection

SMPP client

Please enter the SMPP SMS client connection details to use, when sending SMS messages.

Connection name

Please specify a unique name. This can be anything that helps you identify this connection.

Name: SMPP_client_1

Connect to server

Please enter the hostname or IP address, and the port number of the remote server. You may use IPv4 or IP6 addresses.

Host: 192.168.112.158
Port: 9550
Username: smppuser
Password: ••••••

Telephone number

Specify the telephone number of this connection.

Telephone number: +36307654321
Overridable: ☑

Figure 4 - Providing SMPP Client credentials

It is important to mention that the SMPP hostname and port plus the username and password are provided by your SMS service provider. For example if you contact Vodafone, and ask for an SMPP SMS service, you will sign a contract with them, and often an attachment of this contract will contain the connection following information. If you subscribe to an SMS service on-line, you will likely find this information in the on-line control panel of your SMS service provider. Of course you can always ask your SMS service provider in e-mail and ask what are the SMPP server connection details.

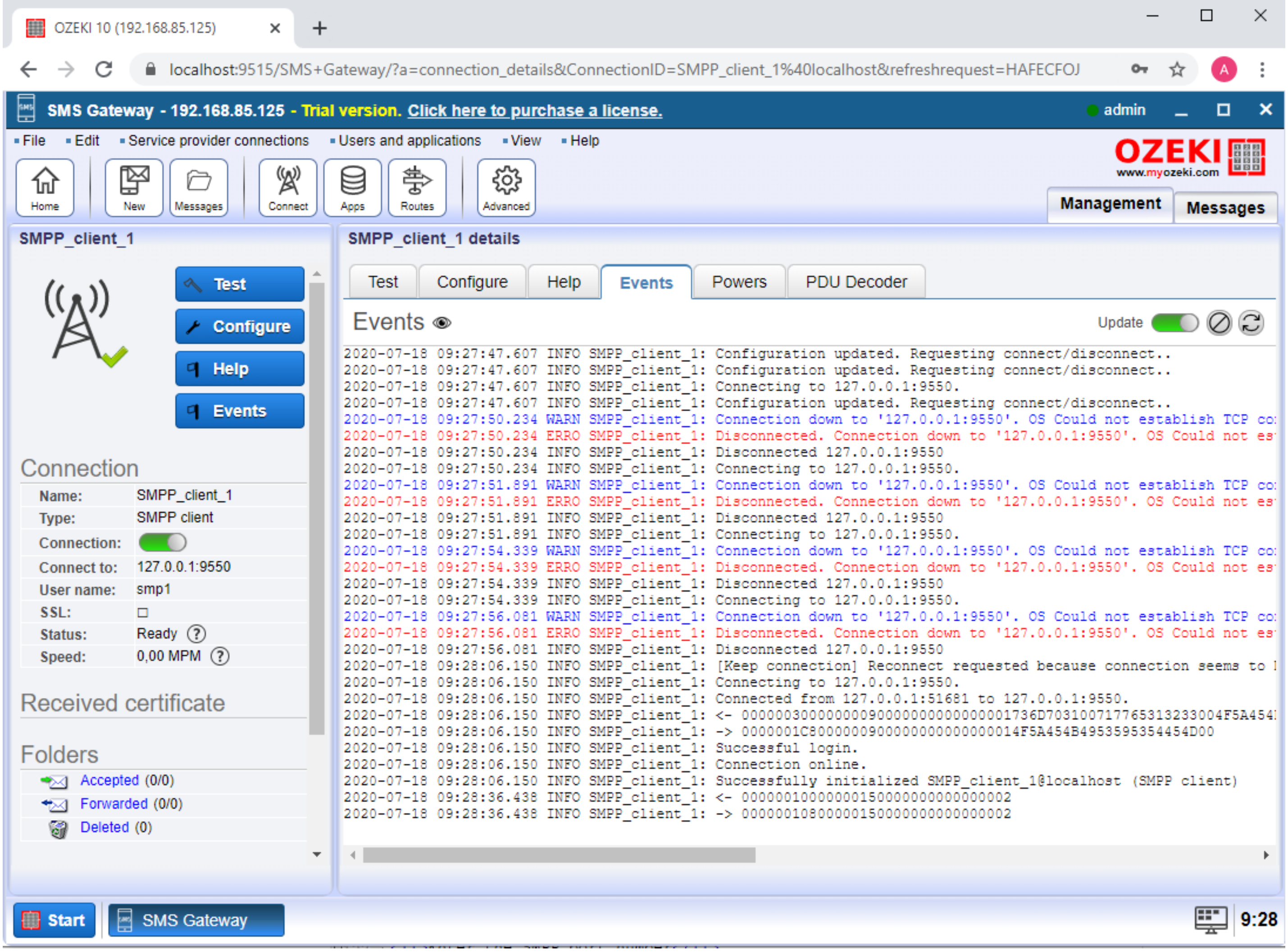
# Check the SMPP log

Once the SMPP connection is configured, you should check the SMPP connection log to see if the system connected properly to your SMS service provider. If the system connected properly you will see the "Connetion online" log entry. The SMS connection logs can be found in the following directory in Windows: C:\Program Files\Ozeki\Data\Logs\Connections\. In this Event tab of the SMPP connection's detalis page, you can see a preview of the log. This page contains the last 100 log entries. Note that the SMPP log files are rotated to save disk space.

**How to view the SMPP connection log:**

- Open the SMPP connection's details page
- Click on the Events tab
- Click on the eye icon next to the Events title
- Copy the file name next to the Events title
- Open notepad

- Paste the filename into the File/Open dialog in notepad
- Click OK to open the SMPP log file
- Use F3 to find the date you are interested in
- Search for "ERRO" to find errors in the log



Connection

Name: SMPP_client_1
Type: SMPP client
Connection:
Connect to: 127.0.0.1:9550
User name: smp1
SSL: □
Status: Ready
Speed: 0,00 MPM

Received certificate

Folders

Accepted (0/0)
Forwarded (0/0)
Deleted (0)

```
2020-07-18 09:27:47.607 INFO SMPP_client_1: Configuration updated. Requesting connect/disconnect..
2020-07-18 09:27:47.607 INFO SMPP_client_1: Configuration updated. Requesting connect/disconnect..
2020-07-18 09:27:47.607 INFO SMPP_client_1: Connecting to 127.0.0.1:9550.
2020-07-18 09:27:47.607 INFO SMPP_client_1: Configuration updated. Requesting connect/disconnect..
2020-07-18 09:27:50.234 WARN SMPP_client_1: Connection down to '127.0.0.1:9550'. OS Could not establish TCP co
2020-07-18 09:27:50.234 ERRO SMPP_client_1: Disconnected. Connection down to '127.0.0.1:9550'. OS Could not es
2020-07-18 09:27:50.234 INFO SMPP_client_1: Disconnected 127.0.0.1:9550
2020-07-18 09:27:50.234 INFO SMPP_client_1: Connecting to 127.0.0.1:9550.
2020-07-18 09:27:51.891 WARN SMPP_client_1: Connection down to '127.0.0.1:9550'. OS Could not establish TCP co
2020-07-18 09:27:51.891 ERRO SMPP_client_1: Disconnected. Connection down to '127.0.0.1:9550'. OS Could not es
2020-07-18 09:27:51.891 INFO SMPP_client_1: Disconnected 127.0.0.1:9550
2020-07-18 09:27:51.891 INFO SMPP_client_1: Connecting to 127.0.0.1:9550.
2020-07-18 09:27:54.339 WARN SMPP_client_1: Connection down to '127.0.0.1:9550'. OS Could not establish TCP co
2020-07-18 09:27:54.339 ERRO SMPP_client_1: Disconnected. Connection down to '127.0.0.1:9550'. OS Could not es
2020-07-18 09:27:54.339 INFO SMPP_client_1: Disconnected 127.0.0.1:9550
2020-07-18 09:27:54.339 INFO SMPP_client_1: Connecting to 127.0.0.1:9550.
2020-07-18 09:27:56.081 WARN SMPP_client_1: Connection down to '127.0.0.1:9550'. OS Could not establish TCP co
2020-07-18 09:27:56.081 ERRO SMPP_client_1: Disconnected. Connection down to '127.0.0.1:9550'. OS Could not es
2020-07-18 09:27:56.081 INFO SMPP_client_1: Disconnected 127.0.0.1:9550
2020-07-18 09:28:06.150 INFO SMPP_client_1: [Keep connection] Reconnect requested because connection seems to
2020-07-18 09:28:06.150 INFO SMPP_client_1: Connecting to 127.0.0.1:9550.
2020-07-18 09:28:06.150 INFO SMPP_client_1: Connected from 127.0.0.1:51681 to 127.0.0.1:9550.
2020-07-18 09:28:06.150 INFO SMPP_client_1: <- 00000030000000090000000000000001736D703100717765313233004F5A454
2020-07-18 09:28:06.150 INFO SMPP_client_1: -> 0000001C800000090000000000000014F5A454B4953595354454D00
2020-07-18 09:28:06.150 INFO SMPP_client_1: Successful login.
2020-07-18 09:28:06.150 INFO SMPP_client_1: Connection online.
2020-07-18 09:28:06.150 INFO SMPP_client_1: Successfully initialized SMPP_client_1@localhost (SMPP client)
2020-07-18 09:28:36.438 INFO SMPP_client_1: <- 00000010000000150000000000000002
2020-07-18 09:28:36.438 INFO SMPP_client_1: -> 00000010800000150000000000000002
```

Figure 5 - View the SMPP connection log

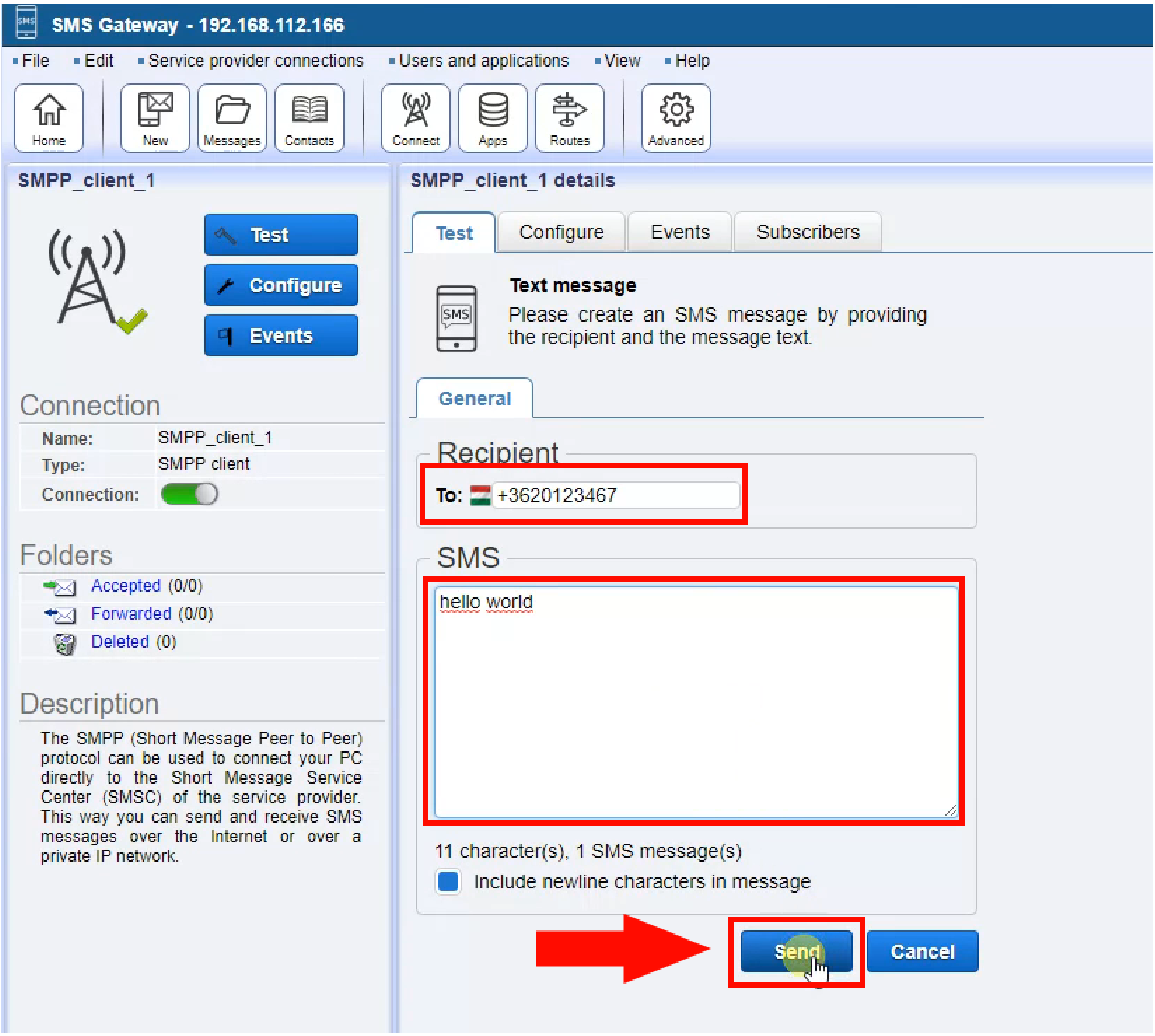
# Send a test SMS message

Once your connection is connected, you can check to see if it is ready to deliver SMS messages by sending a test SMS. To send a test SMS message, you need to open the Test tab, and you need to enter the phone number and message text. It is recommended to provide the phone number in international format. This means the phone number should start with a plus sign followed by a country code. If your local phone number starts with a 0, it is likely that you will need to drop the 0 prefix. For example if your UK phone number 07958663698, you would send the test SMS to +447958663698.

**How to send a test SMPP message:**

- Navigate to the SMPP connection's details page
- Select the Test tab
- Enter the recipient phone number
- Make sure the phone number is in international format
- Enter the message text
- Make sure the message text is less than 160 characters
- Click on the Send button
- Check the SMPP logs

You might ask why should the message text be less than 160 characters. This is because GSM system was designed to send 160 character long text messages. If a message is longer, it will be split into multiple message segments, and will be delivered in more than one SMS message.



SMPP_client_1

Connection

| | |
|---|---|
| Name: | SMPP_client_1 |
| Type: | SMPP client |
| Connection: | |

Folders

Accepted (0/0)
Forwarded (0/0)
Deleted (0)

Description

The SMPP (Short Message Peer to Peer) protocol can be used to connect your PC directly to the Short Message Service Center (SMSC) of the service provider. This way you can send and receive SMS messages over the Internet or over a private IP network.

SMPP_client_1 details

Text message

Please create an SMS message by providing the recipient and the message text.

Recipient

To: +3620123467

SMS

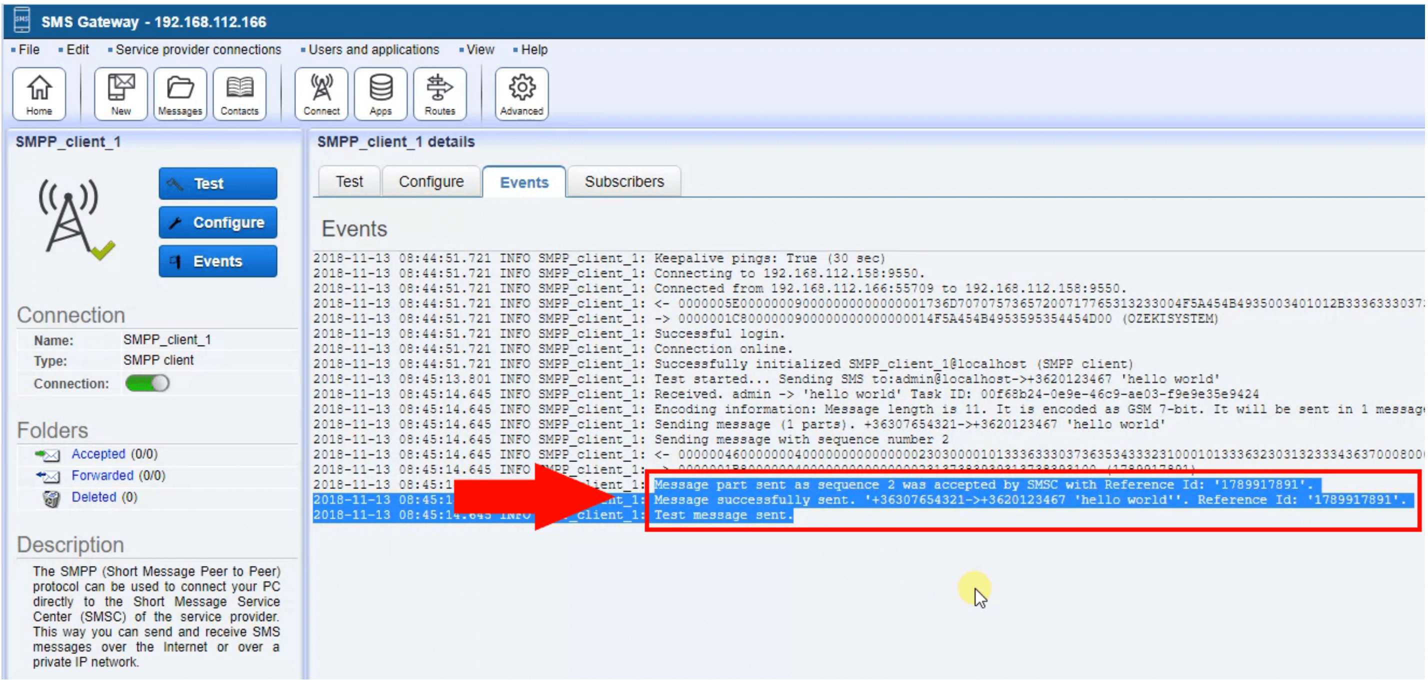
hello world

11 character(s), 1 SMS message(s)

☑ Include newline characters in message

Figure 6 - Sending a test message

After the message is successfully sent, you should check the SMPP logs. The SMPP logs will reveal the low level SMPP messages, that are used to pass the message content to the SMS service provider. These messages are also called an SMPP PDU (SMPP protocol data unit). If there is a problem with message submission, your SMS service provider will ask for the SMPP logs. In this case, you need to send the SMPP PDUs to them. For every SMS submission, two SMPP PDUs will appear in the log. One is the SMPP SUBMIT_SM pdu, which passes the message to the SMS service provider's over, and the SMPP SSUBMIT_SM_RESP, which returns a reference ID, that can ce used for tracking a message, and for finding the delivery reports returned to you when the message is delivered to the recipient handset.



Name: SMPP_client_1
Type: SMPP client
Connection:

The SMPP (Short Message Peer to Peer) protocol can be used to connect your PC directly to the Short Message Service Center (SMSC) of the service provider. This way you can send and receive SMS messages over the Internet or over a private IP network.

```
2018-11-13 08:44:51.721 INFO SMPP_client_1: Keepalive pings: True (30 sec)
2018-11-13 08:44:51.721 INFO SMPP_client_1: Connecting to 192.168.112.158:9550.
2018-11-13 08:44:51.721 INFO SMPP_client_1: Connected from 192.168.112.166:55709 to 192.168.112.158:9550.
2018-11-13 08:44:51.721 INFO SMPP_client_1: <- 0000005E000000090000000000000001736D707075736572007177653132330004F5A454B4935003401012B3336333037
2018-11-13 08:44:51.721 INFO SMPP_client_1: -> 0000001C80000009000000000000000014F5A454B4953595354454D00 (OZEKISYSTEM)
2018-11-13 08:44:51.721 INFO SMPP_client_1: Successful login.
2018-11-13 08:44:51.721 INFO SMPP_client_1: Connection online.
2018-11-13 08:44:51.721 INFO SMPP_client_1: Successfully initialized SMPP_client_1@localhost (SMPP client)
2018-11-13 08:45:13.801 INFO SMPP_client_1: Test started... Sending SMS to:admin@localhost->+3620123467 'hello world'
2018-11-13 08:45:14.645 INFO SMPP_client_1: Received. admin -> 'hello world' Task ID: 00f68b24-0e9e-46c9-ae03-f9e9e35e9424
2018-11-13 08:45:14.645 INFO SMPP_client_1: Encoding information: Message length is 11. It is encoded as GSM 7-bit. It will be sent in 1 messag
2018-11-13 08:45:14.645 INFO SMPP_client_1: Sending message (1 parts). +36307654321->+3620123467 'hello world'
2018-11-13 08:45:14.645 INFO SMPP_client_1: Sending message with sequence number 2
2018-11-13 08:45:14.645 INFO SMPP_client_1: <- 00000046000000040000000000000002303000010133363330373635343332310001013336323031323334363700080
2018-11-13 08:45:14.645 INFO SMPP_client_1: -> 0000001B8000000400000000000000023137383939313738393100 (1789917891)
2018-11-13 08:45:1             ient_1: Message part sent as sequence 2 was accepted by SMSC with Reference Id: '1789917891'.
2018-11-13 08:45:1              nt_1: Message successfully sent. '+36307654321->+3620123467 'hello world''. Reference Id: '1789917891'.
2018-11-13 08:45:14.645 INFO    _client_1: Test message sent.
```

Figure 7 - SMPP SMS submit result in the logs

# SMPP protocol specification

The SMPP protocol specification can be used to understand each SMPP PDU you see in the logs. You will see, that there are several operations in this protocol specifications. For example there is an operation for keeping the SMPP connection alive using keepalive messages, and there are multiple operations for submitting and receiving SMS messages and SMS delivery reports.

**SMPP protocol specification:**

Download: smpp-protocol-specification-v3.4.pdf

# SMPP protocol versions

The SMPP standard is an evolving protocol. The first widely adopted version was v3.3. Currently the most common version you will find is v3.4, but there is also a newer version v5.0 which is rarely used in SMS services.

SMPP 3.3 the oldest version supports GSM SMS messages only. It generates an immediate response for each message sent. In most cases this version is not supported over SSL connections. The problem with SMPP 3.3 is that it requires two SMPP links: an SMPP transmitter and an SMPP receiver link to the SMS service provider. When you setup an SMPP v3.3 link in Ozeki SMS Gateway, you will have to create two SMPP connections, and configure on as SMPP transmitter and the other as SMPP receiver.

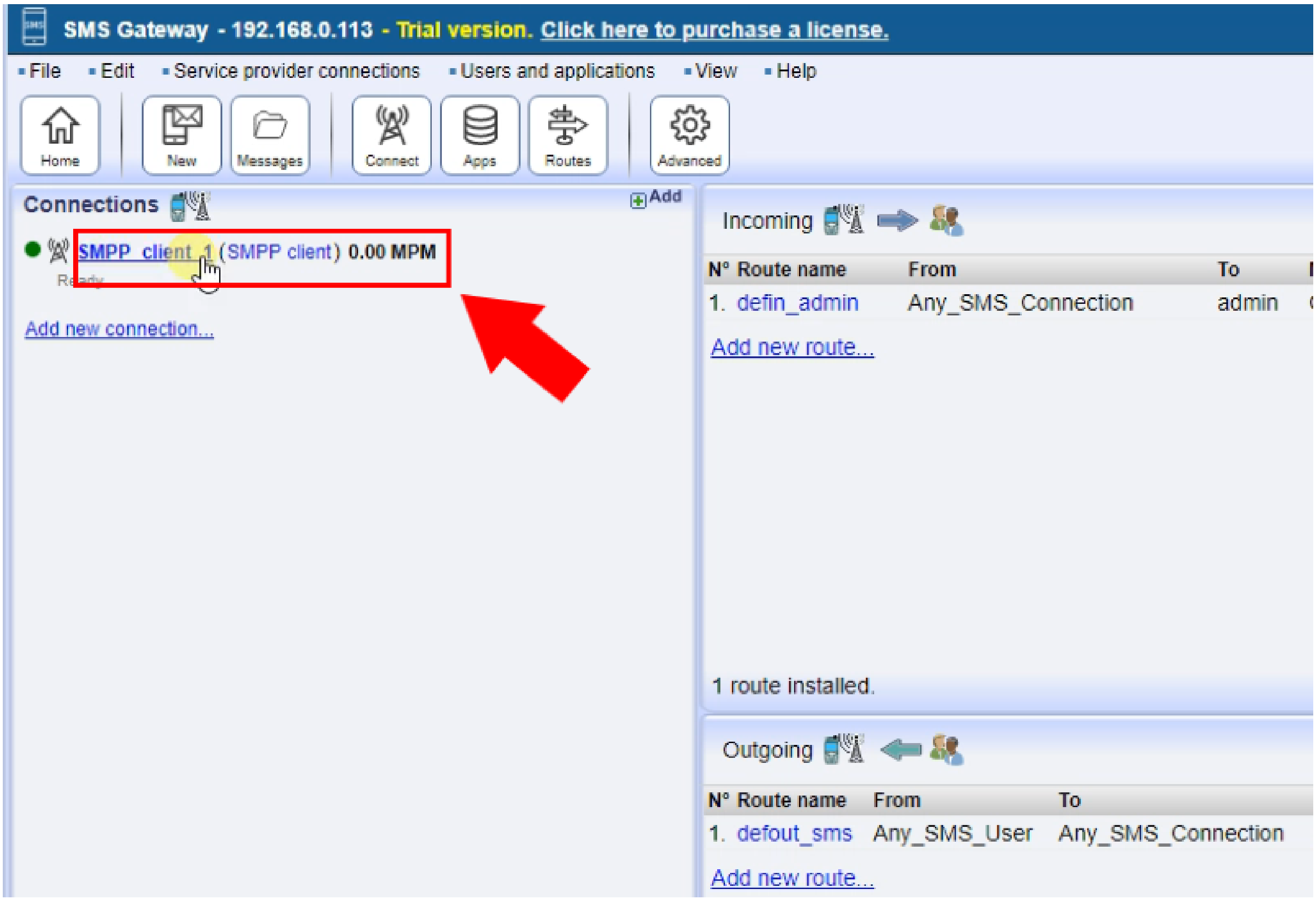
The SMPP 3.4 protocol is similar to SMPP 3.3, but it has a strong advantage: it allows you to send and receive SMS messages over a single TCP/IP link. In Ozeki SMS gateway you will only have to setup on SMPP client connection if you wish to use this protocol. SMPP v3.4 also adds optional Tag-Length-Value (TLV) parameters, to the SMS message, which allows the user to work with non-GSM SMS technologies, such as SMS messaging in CDMA networks.

SMPP 5.0 is the latest version of SMPP. It extends v3.4 by adding support for cell broadcasting, smart flow control. Not many SMS service providers use this protocol. We recommend you to setup SMPP v3.4 connections in Ozeki SMS gateway.

# SMPP PDU logging

***Find out how to view the SMPP PDUs (Protocol Data Units) in your connection's log. This guide gives you the steps to turn on SMPP low level logging for an SMPP client connection. It also shows you where you can find the log file in the file system.***

The first step is to open the SMPP client connection on the left side of the SMS Gateway Manager console page as you can see it on the Figure 1.

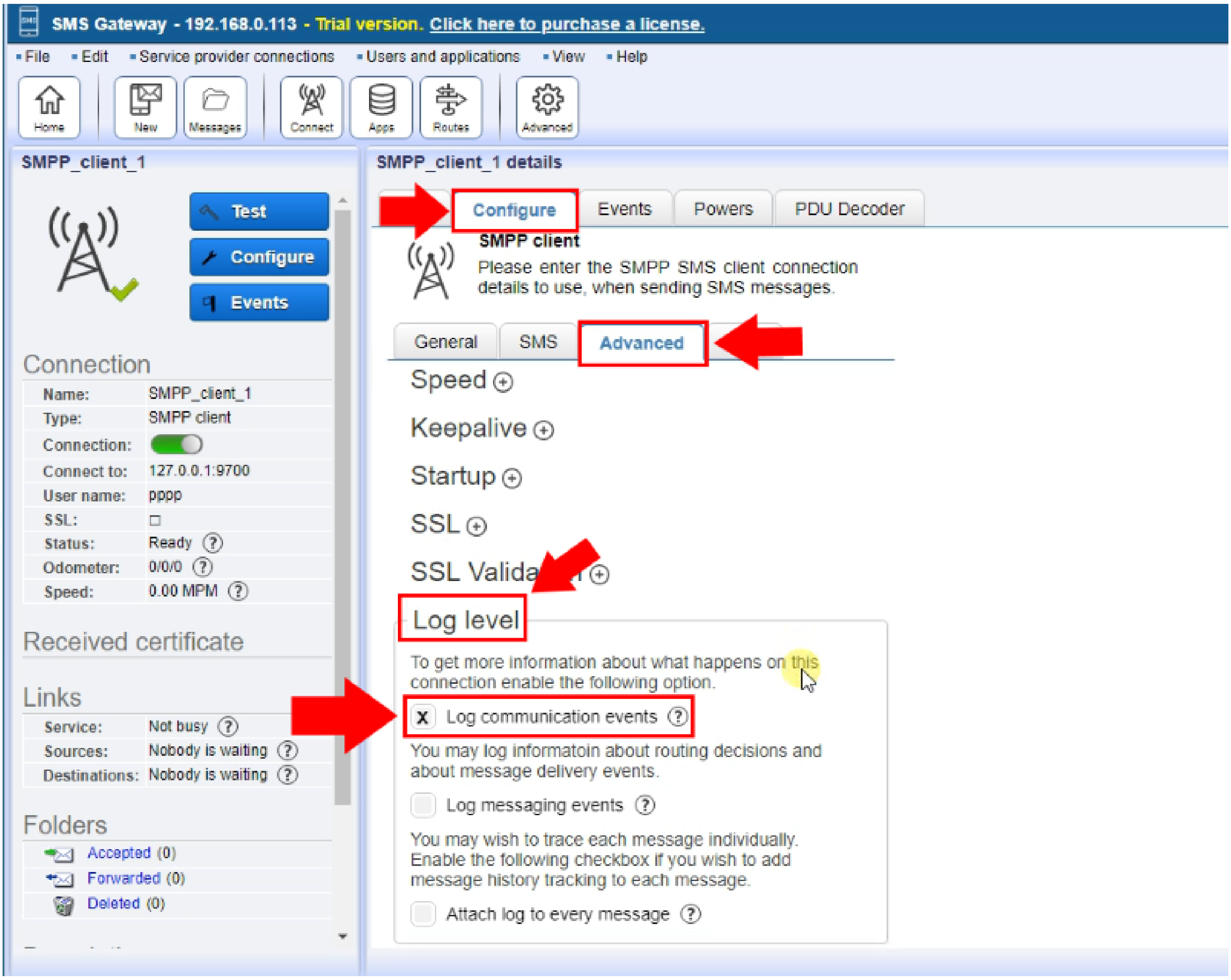


| N° | Route name | From | To |
|---|---|---|---|
| 1. | defin_admin | Any_SMS_Connection | admin |

| N° | Route name | From | To |
|---|---|---|---|
| 1. | defout_sms | Any_SMS_User | Any_SMS_Connection |

Figure 1 - Open SMPP Client

Now in the Log level section of the user's Advanced tab enable the 'Log Communication Events' checkbox if you wish to get Low Level log information about what happens on this connection (Figure 2).



SMPP_client_1

Connection

| | |
|---|---|
| Name: | SMPP_client_1 |
| Type: | SMPP client |
| Connection: | |
| Connect to: | 127.0.0.1:9700 |
| User name: | pppp |
| SSL: | □ |
| Status: | Ready ? |
| Odometer: | 0/0/0 ? |
| Speed: | 0.00 MPM ? |

Received certificate

Links

| | |
|---|---|
| Service: | Not busy ? |
| Sources: | Nobody is waiting ? |
| Destinations: | Nobody is waiting ? |

Folders

Accepted (0)
Forwarded (0)
Deleted (0)

SMPP_client_1 details

SMPP client

Please enter the SMPP SMS client connection details to use, when sending SMS messages.

Log level

To get more information about what happens on this connection enable the following option.

☒ Log communication events ?

You may log informatoin about routing decisions and about message delivery events.

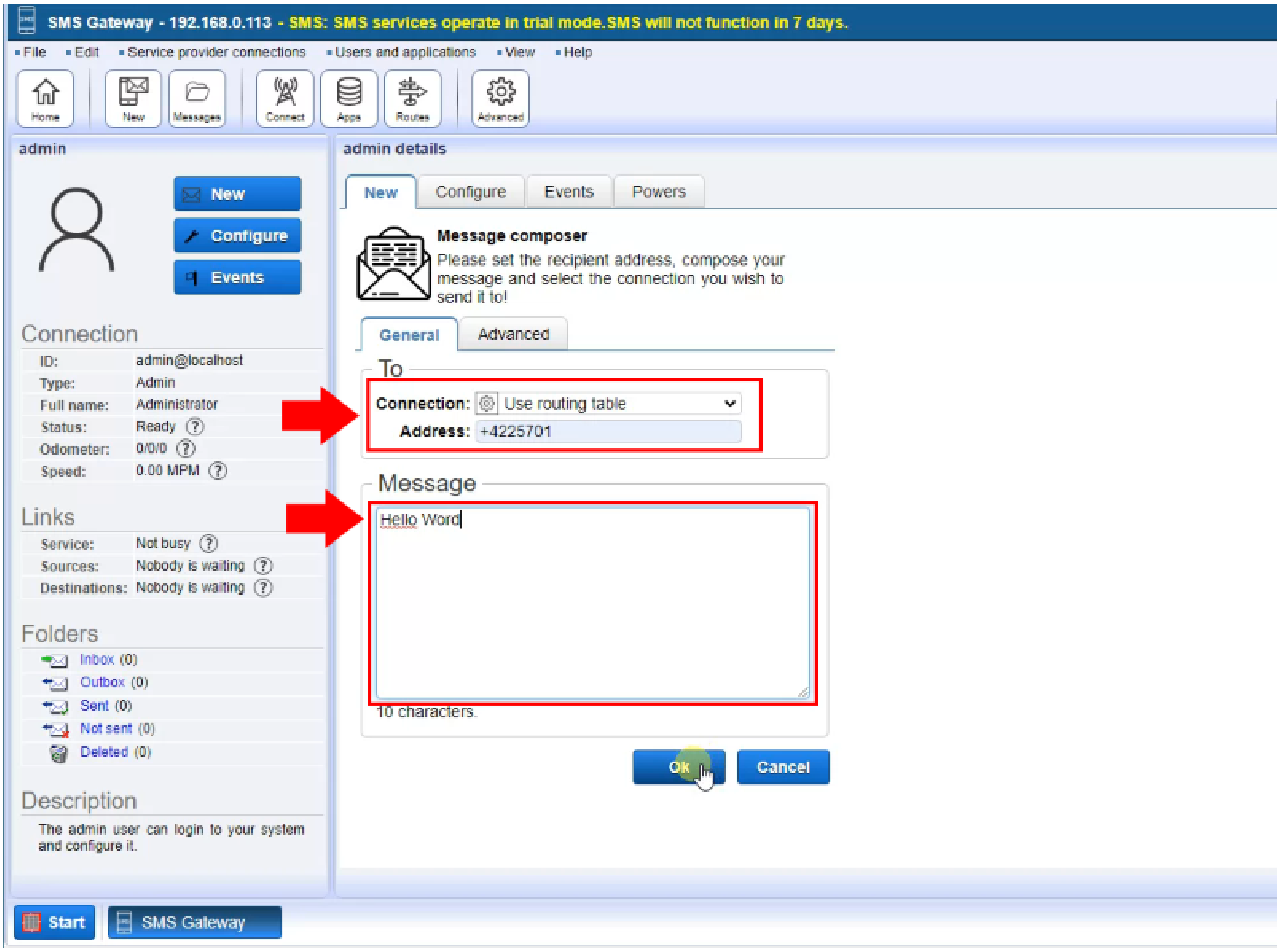
□ Log messaging events ?

You may wish to trace each message individually. Enable the following checkbox if you wish to add message history tracking to each message.

□ Attach log to every message ?

Figure 2 - Enable Log Communication Events

To send a test SMS message, you need to open the Admin user, and you need to enter the phone number and message text. It is recommended to provide the phone number in international format. This means the phone number should start with a plus sign followed by a country code (Figure 3).



SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.

Connection

| | |
|---|---|
| ID: | admin@localhost |
| Type: | Admin |
| Full name: | Administrator |
| Status: | Ready |
| Odometer: | 0/0/0 |
| Speed: | 0.00 MPM |

Links

| | |
|---|---|
| Service: | Not busy |
| Sources: | Nobody is waiting |
| Destinations: | Nobody is waiting |

Folders

- Inbox (0)
- Outbox (0)
- Sent (0)
- Not sent (0)
- Deleted (0)

Description

The admin user can login to your system and configure it.

**Message composer**

Please set the recipient address, compose your message and select the connection you wish to send it to!

Connection: Use routing table

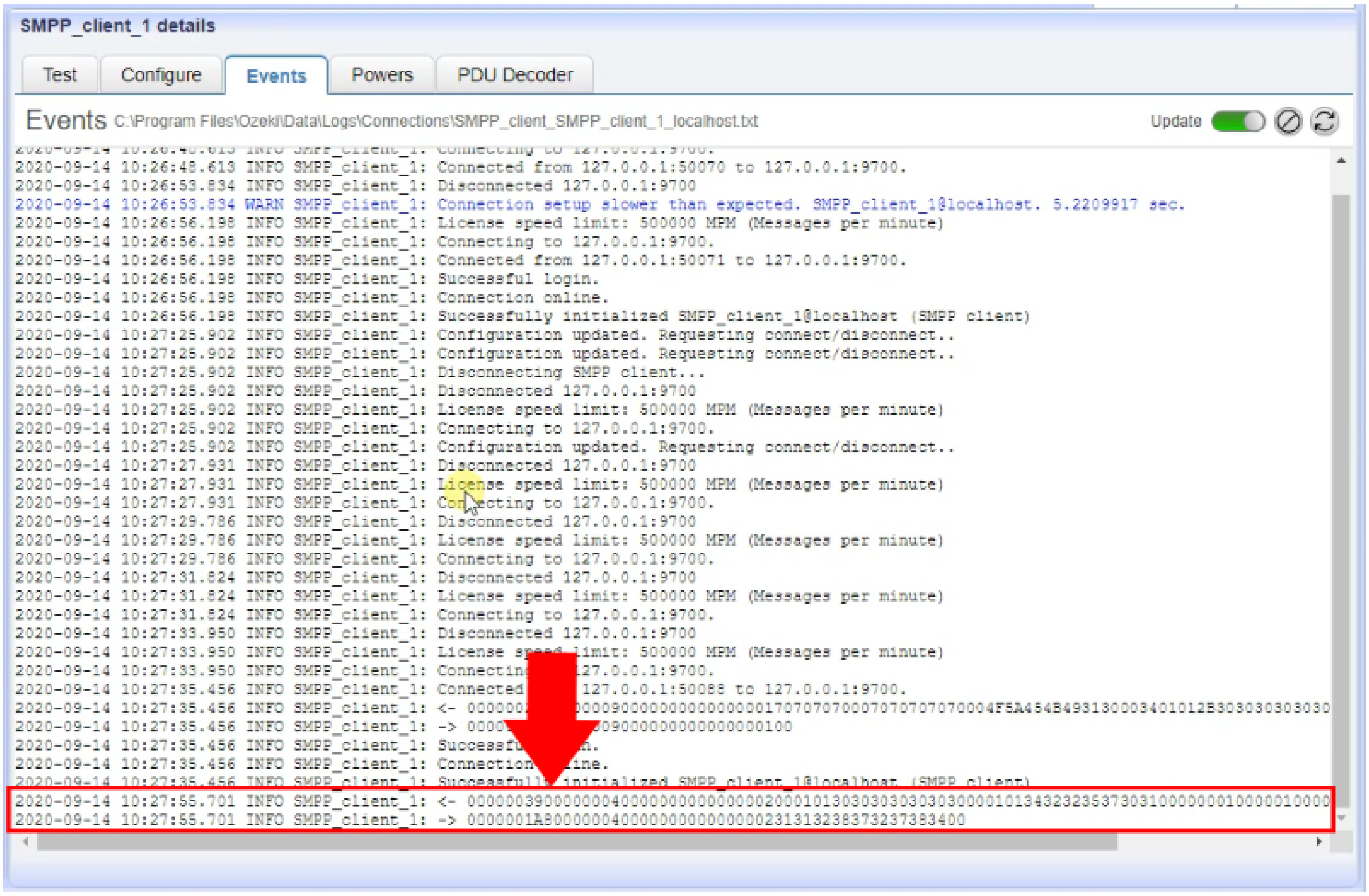
Address: +4225701

Message: Hello Word

10 characters.

Figure 3 - Send test message

After the mesage is successfully sent, you should check the SMPP logs. The SMPP logs will reveal the low level SMPP messages, that are used to pass the message content to the SMS service provider. These messages are also called an SMPP PDU (SMPP protocol data unit). If there is a problem with message submission, your SMS service provider will ask for the SMPP logs. In this case, you need to send the SMPP PDUs to them. For every SMS submission, two SMPP PDUs will appear in the log (Figure 4).



C:\Program Files\Ozeki\Data\Logs\Connections\SMPP_client_SMPP_client_1_localhost.txt

```
2020-09-14 10:26:48.613 INFO SMPP_client_1: Connecting to 127.0.0.1:9700.
2020-09-14 10:26:48.613 INFO SMPP_client_1: Connected from 127.0.0.1:50070 to 127.0.0.1:9700.
2020-09-14 10:26:53.834 INFO SMPP_client_1: Disconnected 127.0.0.1:9700
2020-09-14 10:26:53.834 WARN SMPP_client_1: Connection setup slower than expected. SMPP_client_1@localhost. 5.2209917 sec.
2020-09-14 10:26:56.198 INFO SMPP_client_1: License speed limit: 500000 MPM (Messages per minute)
2020-09-14 10:26:56.198 INFO SMPP_client_1: Connecting to 127.0.0.1:9700.
2020-09-14 10:26:56.198 INFO SMPP_client_1: Connected from 127.0.0.1:50071 to 127.0.0.1:9700.
2020-09-14 10:26:56.198 INFO SMPP_client_1: Successful login.
2020-09-14 10:26:56.198 INFO SMPP_client_1: Connection online.
2020-09-14 10:26:56.198 INFO SMPP_client_1: Successfully initialized SMPP_client_1@localhost (SMPP client)
2020-09-14 10:27:25.902 INFO SMPP_client_1: Configuration updated. Requesting connect/disconnect..
2020-09-14 10:27:25.902 INFO SMPP_client_1: Configuration updated. Requesting connect/disconnect..
2020-09-14 10:27:25.902 INFO SMPP_client_1: Disconnecting SMPP client...
2020-09-14 10:27:25.902 INFO SMPP_client_1: Disconnected 127.0.0.1:9700
2020-09-14 10:27:25.902 INFO SMPP_client_1: License speed limit: 500000 MPM (Messages per minute)
2020-09-14 10:27:25.902 INFO SMPP_client_1: Connecting to 127.0.0.1:9700.
2020-09-14 10:27:25.902 INFO SMPP_client_1: Configuration updated. Requesting connect/disconnect..
2020-09-14 10:27:27.931 INFO SMPP_client_1: Disconnected 127.0.0.1:9700
2020-09-14 10:27:27.931 INFO SMPP_client_1: License speed limit: 500000 MPM (Messages per minute)
2020-09-14 10:27:27.931 INFO SMPP_client_1: Connecting to 127.0.0.1:9700.
2020-09-14 10:27:29.786 INFO SMPP_client_1: Disconnected 127.0.0.1:9700
2020-09-14 10:27:29.786 INFO SMPP_client_1: License speed limit: 500000 MPM (Messages per minute)
2020-09-14 10:27:29.786 INFO SMPP_client_1: Connecting to 127.0.0.1:9700.
2020-09-14 10:27:31.824 INFO SMPP_client_1: Disconnected 127.0.0.1:9700
2020-09-14 10:27:31.824 INFO SMPP_client_1: License speed limit: 500000 MPM (Messages per minute)
2020-09-14 10:27:31.824 INFO SMPP_client_1: Connecting to 127.0.0.1:9700.
2020-09-14 10:27:33.950 INFO SMPP_client_1: Disconnected 127.0.0.1:9700
2020-09-14 10:27:33.950 INFO SMPP_client_1: License sp    limit: 500000 MPM (Messages per minute)
2020-09-14 10:27:33.950 INFO SMPP_client_1: Connectin     127.0.0.1:9700.
2020-09-14 10:27:35.456 INFO SMPP_client_1: Connected      127.0.0.1:50088 to 127.0.0.1:9700.
2020-09-14 10:27:35.456 INFO SMPP_client_1: <- 000000     0000900000000000000000170707070007070707070004F5A454B4931300034010128303030303030
2020-09-14 10:27:35.456 INFO SMPP_client_1: -> 0000      0090000000000000000100
2020-09-14 10:27:35.456 INFO SMPP_client_1: Successfu     n.
2020-09-14 10:27:35.456 INFO SMPP_client_1: Connection    ine.
2020-09-14 10:27:35.456 INFO SMPP_client_1: Successfull  initialized SMPP_client_1@localhost (SMPP client)
2020-09-14 10:27:55.701 INFO SMPP_client_1: <- 00000039000000040000000000000000200010130303030303030000101343232353730310000000010000010000
2020-09-14 10:27:55.701 INFO SMPP_client_1: -> 0000001A80000004000000000000000231313238373237383400
```

Figure 4 - Message PDU Log

# SMPP Error Codes

| 0 | 0x0 | ESME_ROK | Ok - Message Acceptable |
|---|---|---|---|
| 1 | 0x1 | ESME_RINVMSGLEN | Invalid Message Length |
| 2 | 0x2 | ESME_RINVCMDLEN | Invalid Command Length |
| 3 | 0x3 | ESME_RINVCMDID | Invalid Command ID |
| 4 | 0x4 | ESME_RINVBNDSTS | Invalid bind status |
| 5 | 0x5 | ESME_RALYBND | Bind attempted when already bound |
| 6 | 0x6 | ESME_RINVPRTFLG | Invalid priority flag |
| 7 | 0x7 | ESME_RINVREGDLVFLG | Invalid registered-delivery flag |
| 8 | 0x8 | ESME_RSYSERR | SMSC system error |
| 10 | 0xa | ESME_RINVSRCADR | Invalid source address |
| 11 | 0xb | ESME_RINVDSTADR | Invalid destination address |
| 12 | 0xc | ESME_RINVMSGID | Invalid message-id |
| 13 | 0xd | ESME_RBINDFAIL | Generic bind failure |
| 14 | 0xe | ESME_RINVPASWD | Invalid password |
| 15 | 0xf | ESME_RINVSYSID | Invalid System-ID |
| 17 | 0x11 | ESME_RCANCELFAIL | Cancel failure |
| 19 | 0x13 | ESME_RREPLACEFAIL | Replace failure |
| 20 | 0x14 | ESME_RMSGQFUL | Too many messages in queue, at present |
| 22 | 0x16 | ESME_RINVSERTYP | Invalid services type |
| 51 | 0x33 | ESME_RINVNUMDESTS | Invalid number of destination addresses |
| 52 | 0x34 | ESME_RINVDLNAME | Invalid name |
| 64 | 0x40 | ESME_RINVDESTFLAG | Invalid Destination Flag Option |
| 66 | 0x42 | ESME_RINVSUBREP | Invalid value for submit with replace option |
| 67 | 0x43 | ESME_RINVESMCLASS | Invalid value for esm_class field |
| 68 | 0x44 | ESME_RCNTSUBDL | Cannot submit to a distribution list |
| 69 | 0x45 | ESME_RSUBMITFAIL | Generic submission failure |
| 72 | 0x48 | ESME_RINVSRCTON | Invalid type of number for source |
| 73 | 0x49 | ESME_RINVSRCNPI | Invalid numbering plan indicator for source |
| 74 | 0x4a | ESME_RINVDSTTON | Invalid type of number for destination |
| 75 | 0x4b | ESME_RINVDSTNPI | Invalid numbering plan indicator for destination |
| 77 | 0x4d | ESME_RINVSYSTYP | Invalid esm type |
| 78 | 0x4e | ESME_RINVREPFLAG | Invalid submit with replace flag option |
| 85 | 0x55 | ESME_RINVNUMMSGS | Invalid number of messages specified for query_last_msgs primitive |
| 88 | 0x58 | ESME_RTHROTTLED | SMSC is throttling inbound messages |
| 98 | 0x62 | ESME_RINVEXPIRY | Invalid Validity Date |
| 103 | 0x67 | ESME_RQUERYFAIL | Quota violation, add credit to account |
| 194 | 0xc2 | ESME_RINVPARLEN | Invalid optional parameter length |
| 195 | 0xc3 | ESME_RMISSINGOPTPARAM | Missing optional parameter |

| 0 | 0x0 | ESME_ROK | Ok - Message Acceptable |
|---|---|---|---|
| 196 | 0xc4 | ESME_RINVOPTPARAMVAL | Invalid optional parameter value |
| 254 | 0xfe | ESME_RDELIVERYFAILURE | Generic delivery failure |
| 255 | 0xff | ESME_RUNKNOWNERR | Unknown Error |

# SMPP over SSL/TLS

***This guide gives you the instructions you should follow if you wish to use SMPP over a secure SSL/TLS connection. SMPP over SSL/TLS has 3 main advantages: The network traffic between your system and the system of the SMS service provider will be encrypted. You will be able to validate the identity of the SMS service provider, which is great, because you can detect a Man in the Middle attack this way, and finally the service provider will also be able to verify your system if you setup client SSL certificates. Ozeki 10 SMS gateway supports all of these options.***

Open Security app from the **Start** menu.

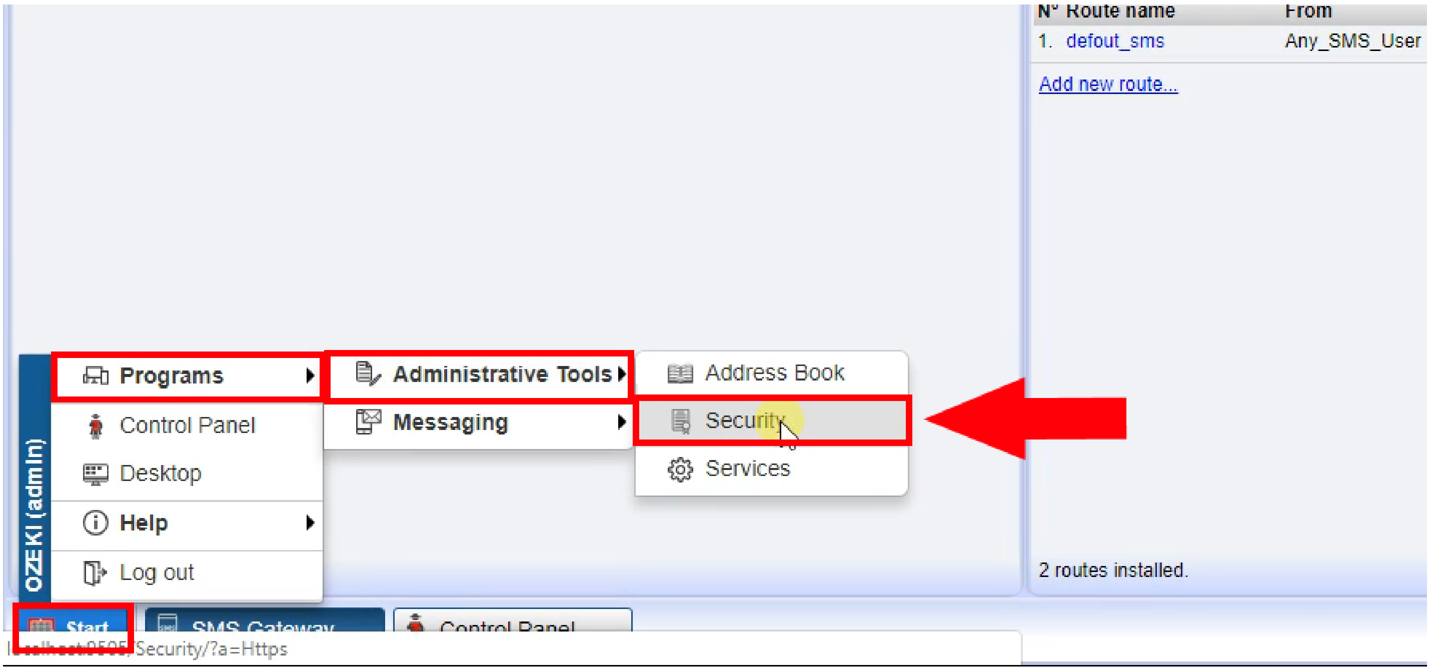
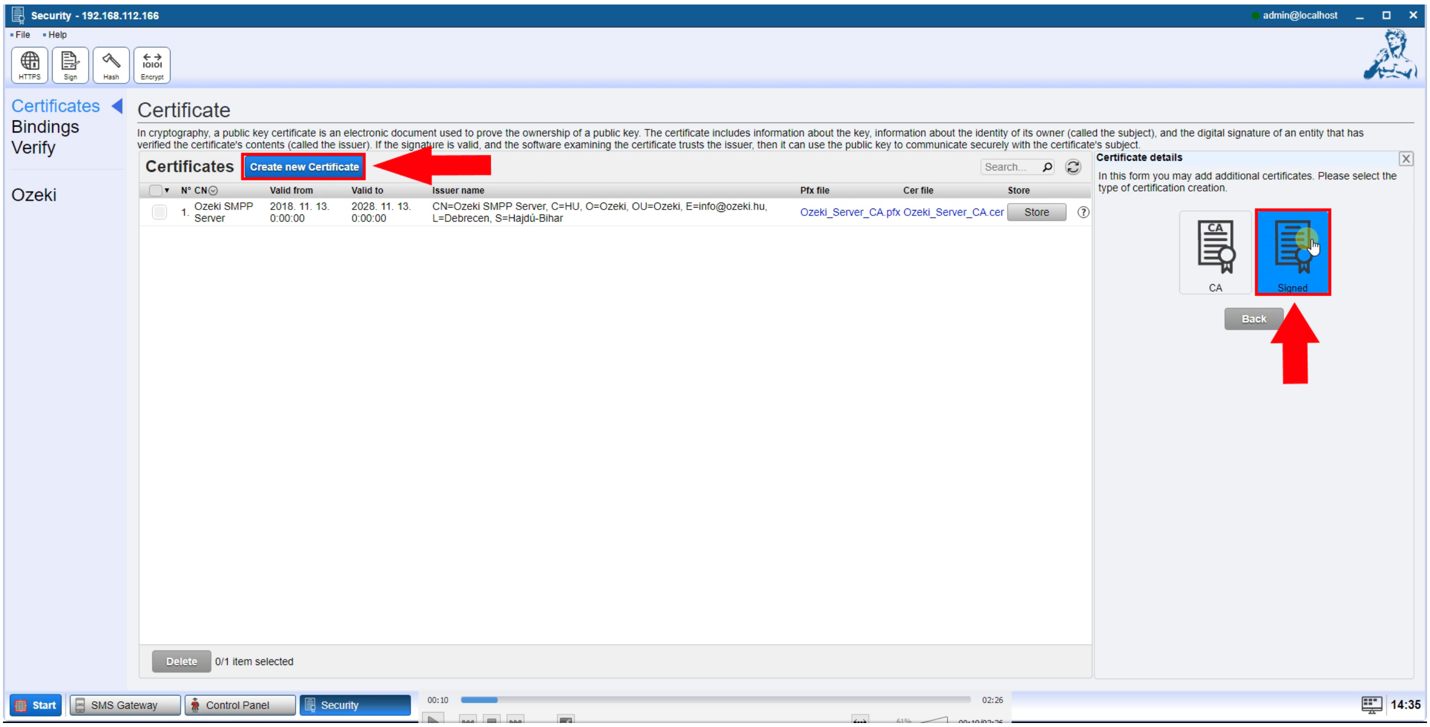


Figure 1 - Open Security app

Create a Signed certificate.



Certificate

In cryptography, a public key certificate is an electronic document used to prove the ownership of a public key. The certificate includes information about the key, information about the identity of its owner (called the subject), and the digital signature of an entity that has verified the certificate's contents (called the issuer). If the signature is valid, and the software examining the certificate trusts the issuer, then it can use the public key to communicate securely with the certificate's subject.

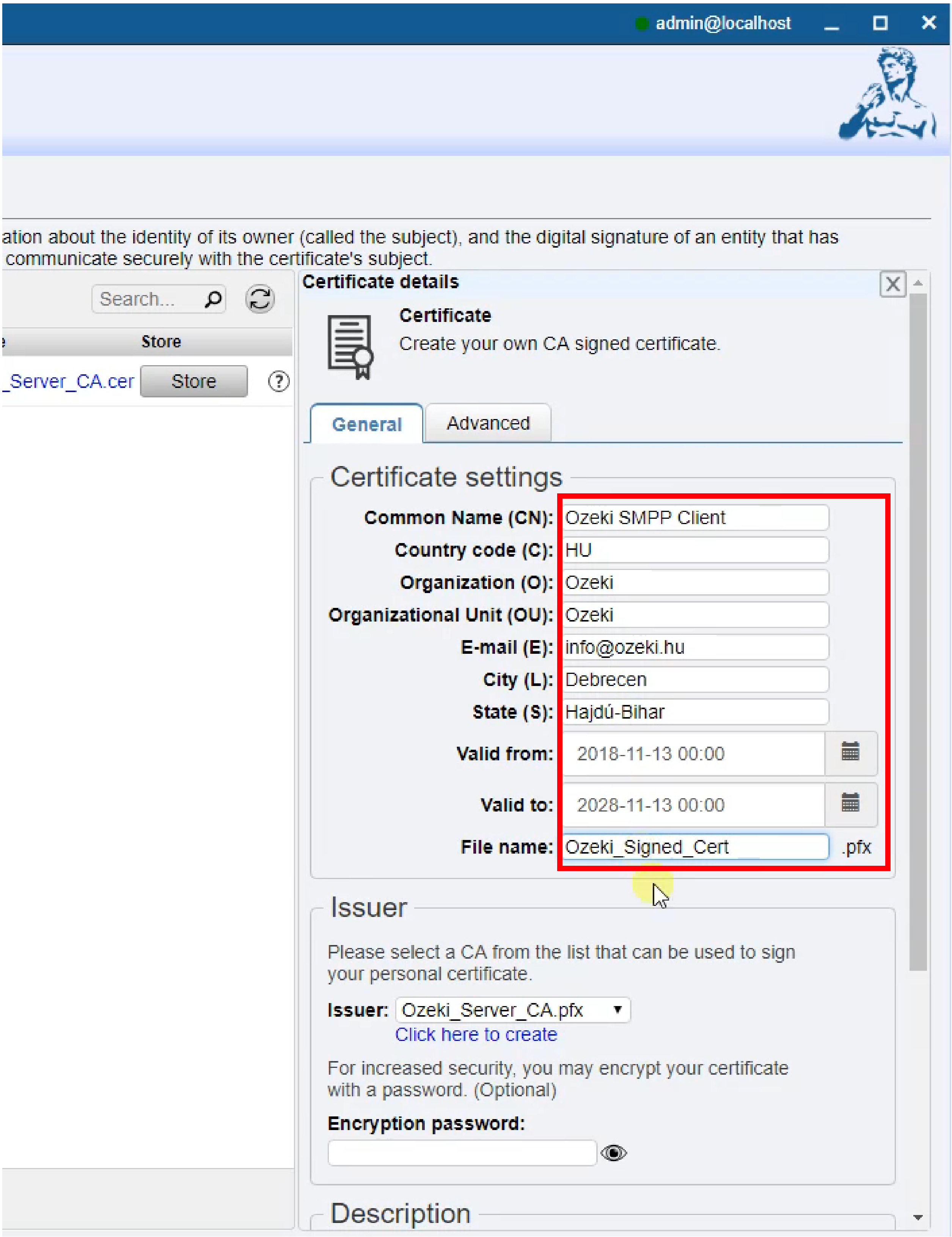
Certificates

| N° | CN | Valid from | Valid to | Issuer name | Pfx file | Cer file | Store |
|---|---|---|---|---|---|---|---|
| 1. | Ozeki SMPP Server | 2018. 11. 13. 0:00:00 | 2028. 11. 13. 0:00:00 | CN=Ozeki SMPP Server, C=HU, O=Ozeki, OU=Ozeki, E=info@ozeki.hu, L=Debrecen, S=Hajdú-Bihar | Ozeki_Server_CA.pfx | Ozeki_Server_CA.cer | Store |

In this form you may add additional certificates. Please select the type of certification creation.

Figure 2 - Create Signed certificate

Specify the certificate details.



ation about the identity of its owner (called the subject), and the digital signature of an entity that has communicate securely with the certificate's subject.

Certificate details

Certificate

Create your own CA signed certificate.

Certificate settings

Common Name (CN): Ozeki SMPP Client

Country code (C): HU

Organization (O): Ozeki

Organizational Unit (OU): Ozeki

E-mail (E): info@ozeki.hu

City (L): Debrecen

State (S): Hajdú-Bihar

Valid from: 2018-11-13 00:00

Valid to: 2028-11-13 00:00

File name: Ozeki_Signed_Cert .pfx

Issuer

Please select a CA from the list that can be used to sign your personal certificate.

Issuer: Ozeki_Server_CA.pfx

Click here to create

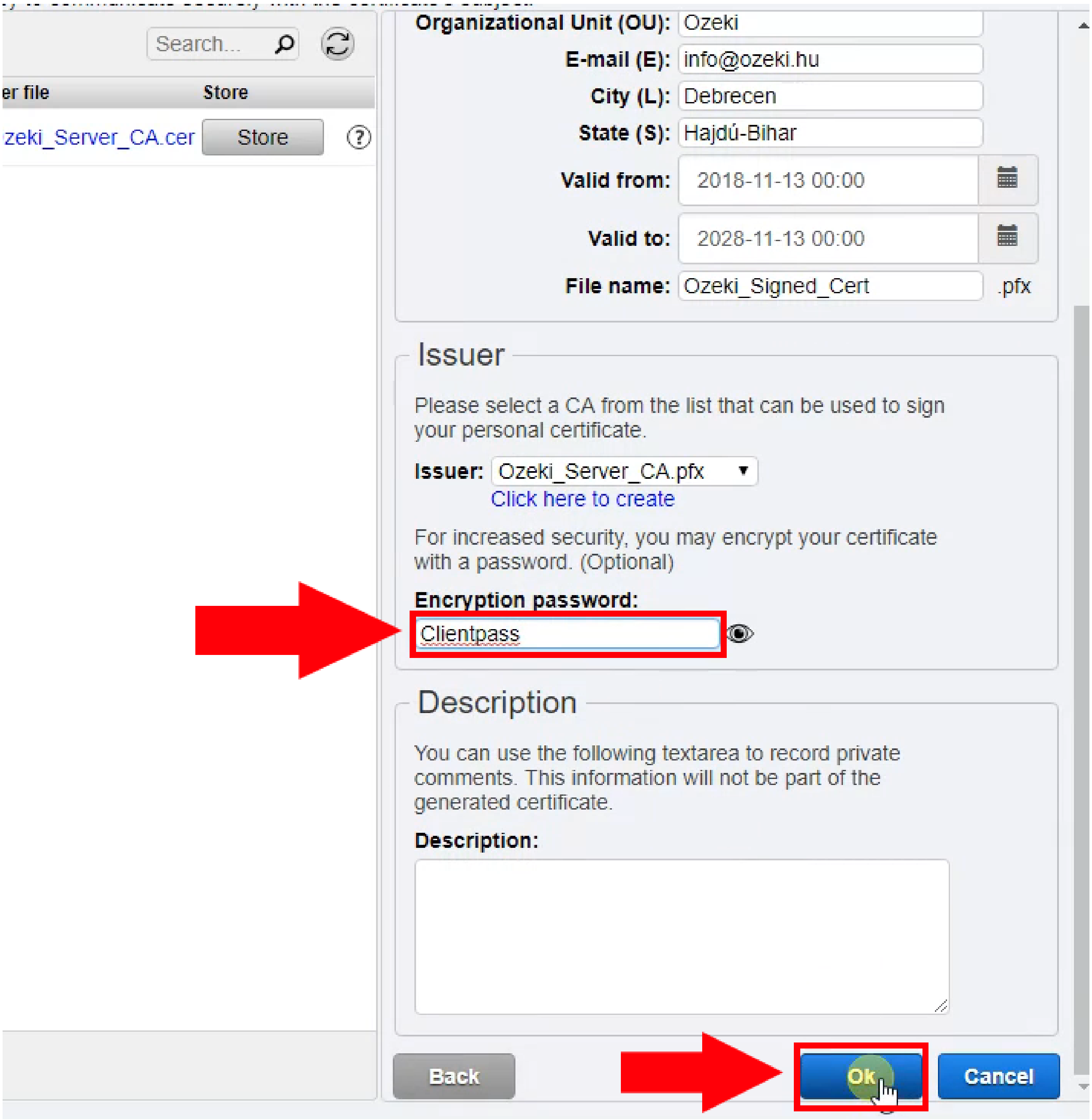
For increased security, you may encrypt your certificate with a password. (Optional)

Encryption password:

Description

Figure 3 - Provide cerificate details

Provide the password for the created certificate. Then click OK.



Organizational Unit (OU): Ozeki
E-mail (E): info@ozeki.hu
City (L): Debrecen
State (S): Hajdú-Bihar
Valid from: 2018-11-13 00:00
Valid to: 2028-11-13 00:00
File name: Ozeki_Signed_Cert .pfx

Issuer

Please select a CA from the list that can be used to sign your personal certificate.

Issuer: Ozeki_Server_CA.pfx

For increased security, you may encrypt your certificate with a password. (Optional)

Encryption password: Clientpass

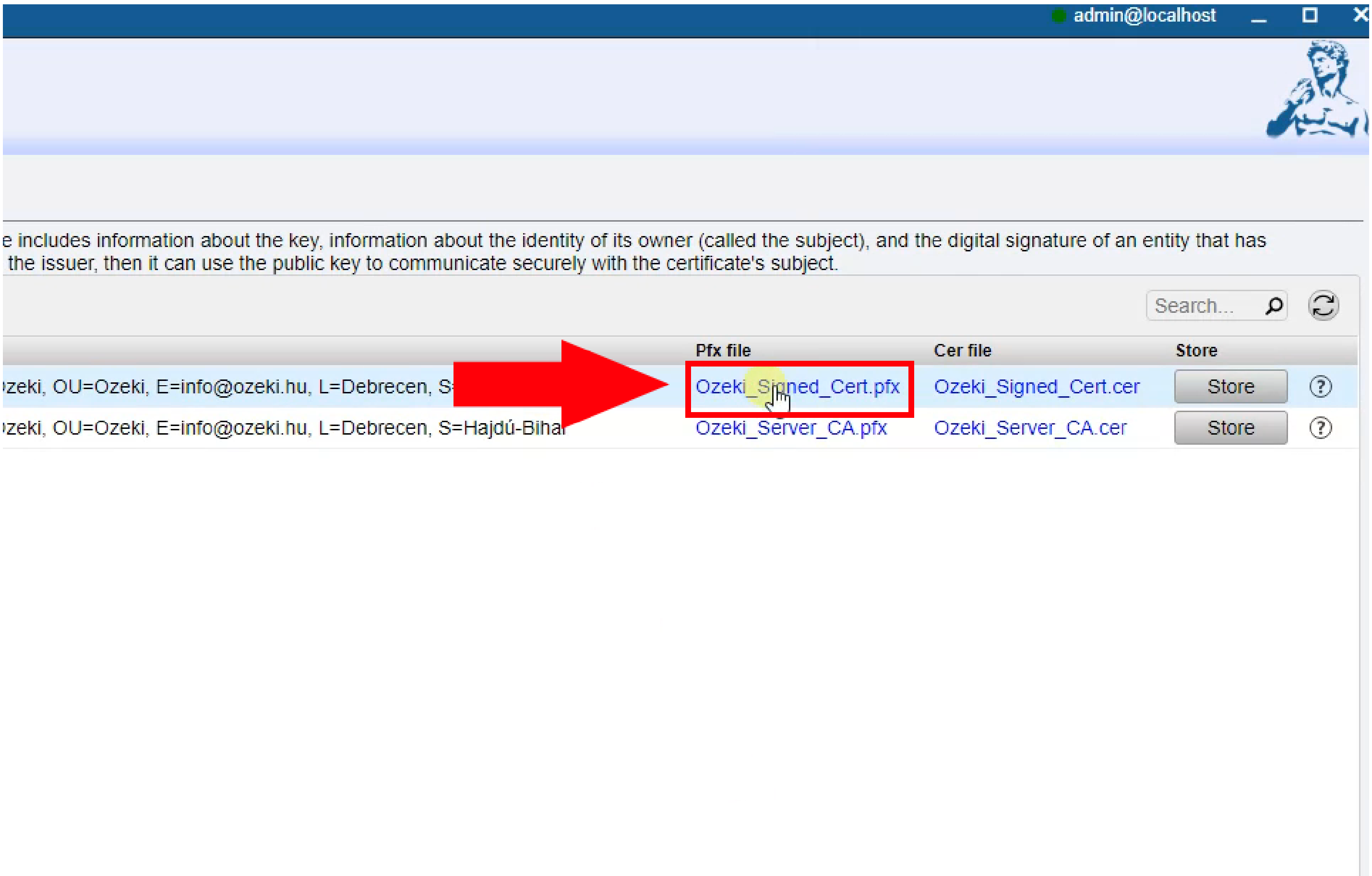
Description

You can use the following textarea to record private comments. This information will not be part of the generated certificate.

Description:

Figure 4 - Provide certificate password

Download the cerificate and send it to the client.

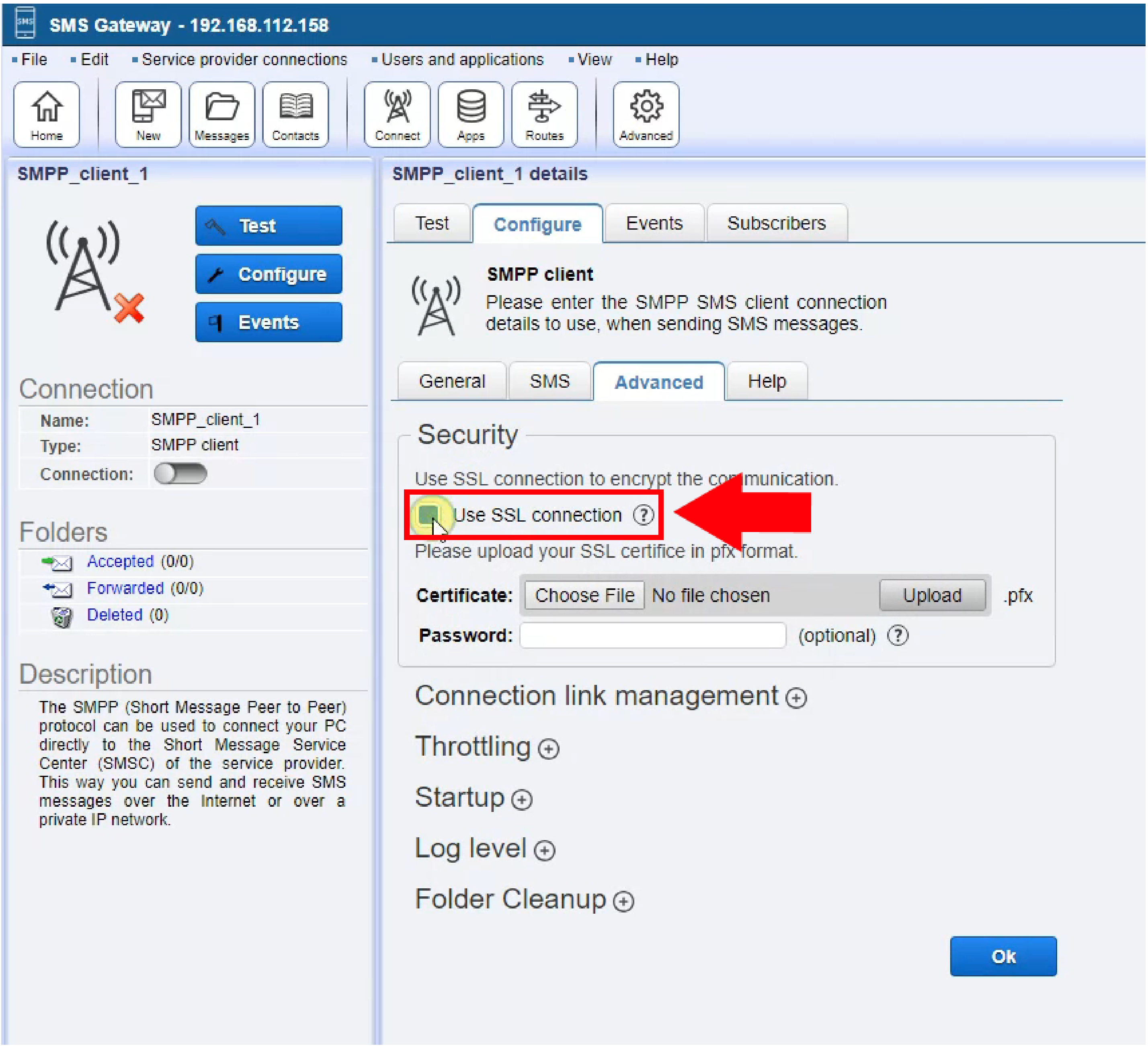


e includes information about the key, information about the identity of its owner (called the subject), and the digital signature of an entity that has the issuer, then it can use the public key to communicate securely with the certificate's subject.

|  | Pfx file | Cer file | Store |  |
|---|---|---|---|---|
| ›zeki, OU=Ozeki, E=info@ozeki.hu, L=Debrecen, S- | Ozeki_Signed_Cert.pfx | Ozeki_Signed_Cert.cer | Store | ? |
| ›zeki, OU=Ozeki, E=info@ozeki.hu, L=Debrecen, S=Hajdú-Bihar | Ozeki_Server_CA.pfx | Ozeki_Server_CA.cer | Store | ? |

Figure 5 - Download certificate

Enable SSL connection on the SMPP client.



SMPP_client_1

Connection

| | |
|---|---|
| Name: | SMPP_client_1 |
| Type: | SMPP client |
| Connection: | ○ |

Folders

- Accepted (0/0)
- Forwarded (0/0)
- Deleted (0)

Description

The SMPP (Short Message Peer to Peer) protocol can be used to connect your PC directly to the Short Message Service Center (SMSC) of the service provider. This way you can send and receive SMS messages over the Internet or over a private IP network.

SMPP_client_1 details

**SMPP client**
Please enter the SMPP SMS client connection details to use, when sending SMS messages.

Security

Use SSL connection to encrypt the communication.

☑ Use SSL connection ?

Please upload your SSL certifice in pfx format.

**Certificate:** No file chosen .pfx

**Password:** (optional) ?

Connection link management ⊕

Throttling ⊕

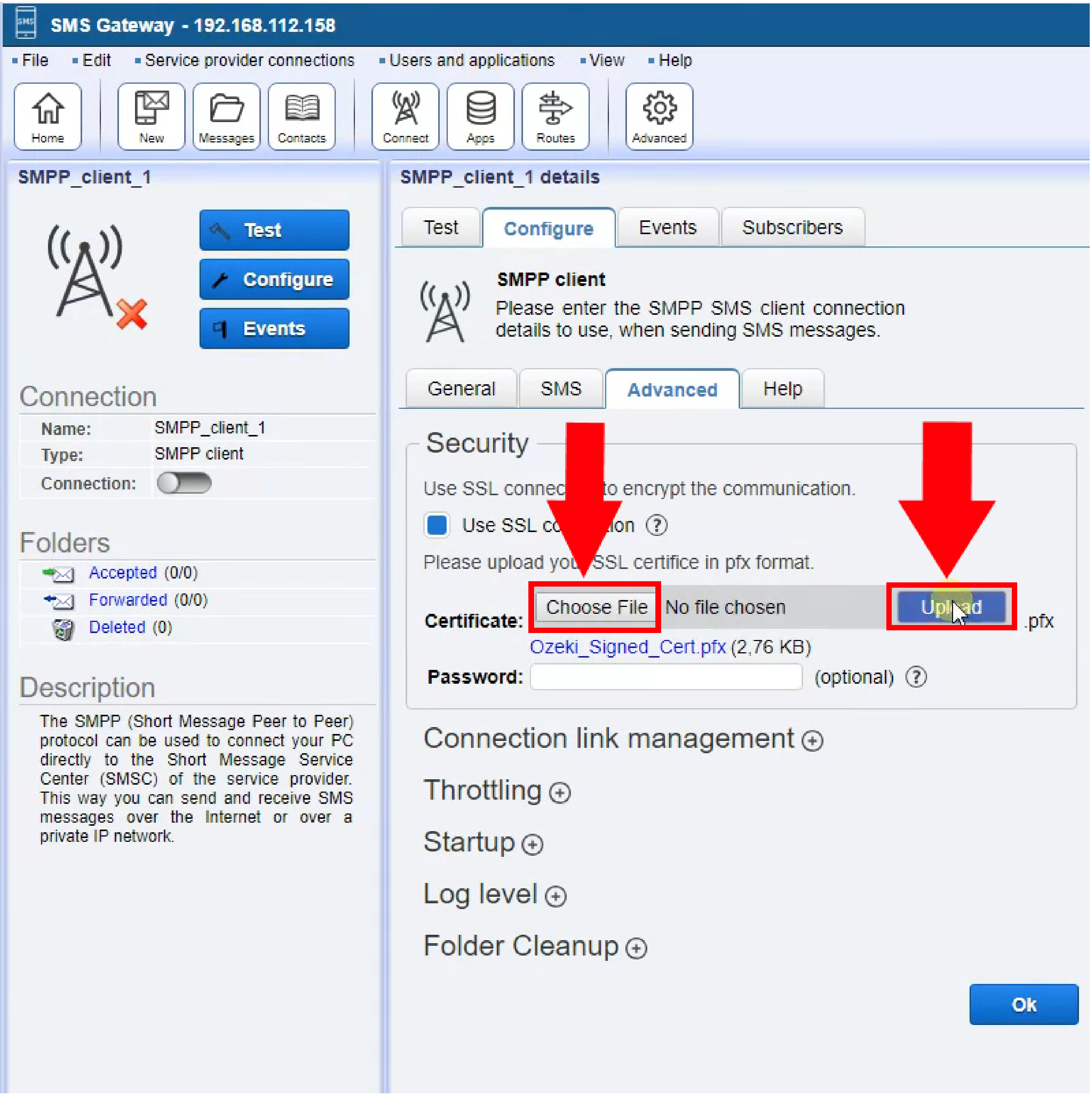
Startup ⊕

Log level ⊕

Folder Cleanup ⊕

Figure 6 - Use SSL connection

Upload the Signed certificate.



SMPP_client_1

Connection

Name: SMPP_client_1
Type: SMPP client
Connection:

Folders

Accepted (0/0)
Forwarded (0/0)
Deleted (0)

Description

The SMPP (Short Message Peer to Peer) protocol can be used to connect your PC directly to the Short Message Service Center (SMSC) of the service provider. This way you can send and receive SMS messages over the Internet or over a private IP network.

SMPP_client_1 details

**SMPP client**
Please enter the SMPP SMS client connection details to use, when sending SMS messages.

Security

Use SSL connec to encrypt the communication.

☑ Use SSL co ion ?

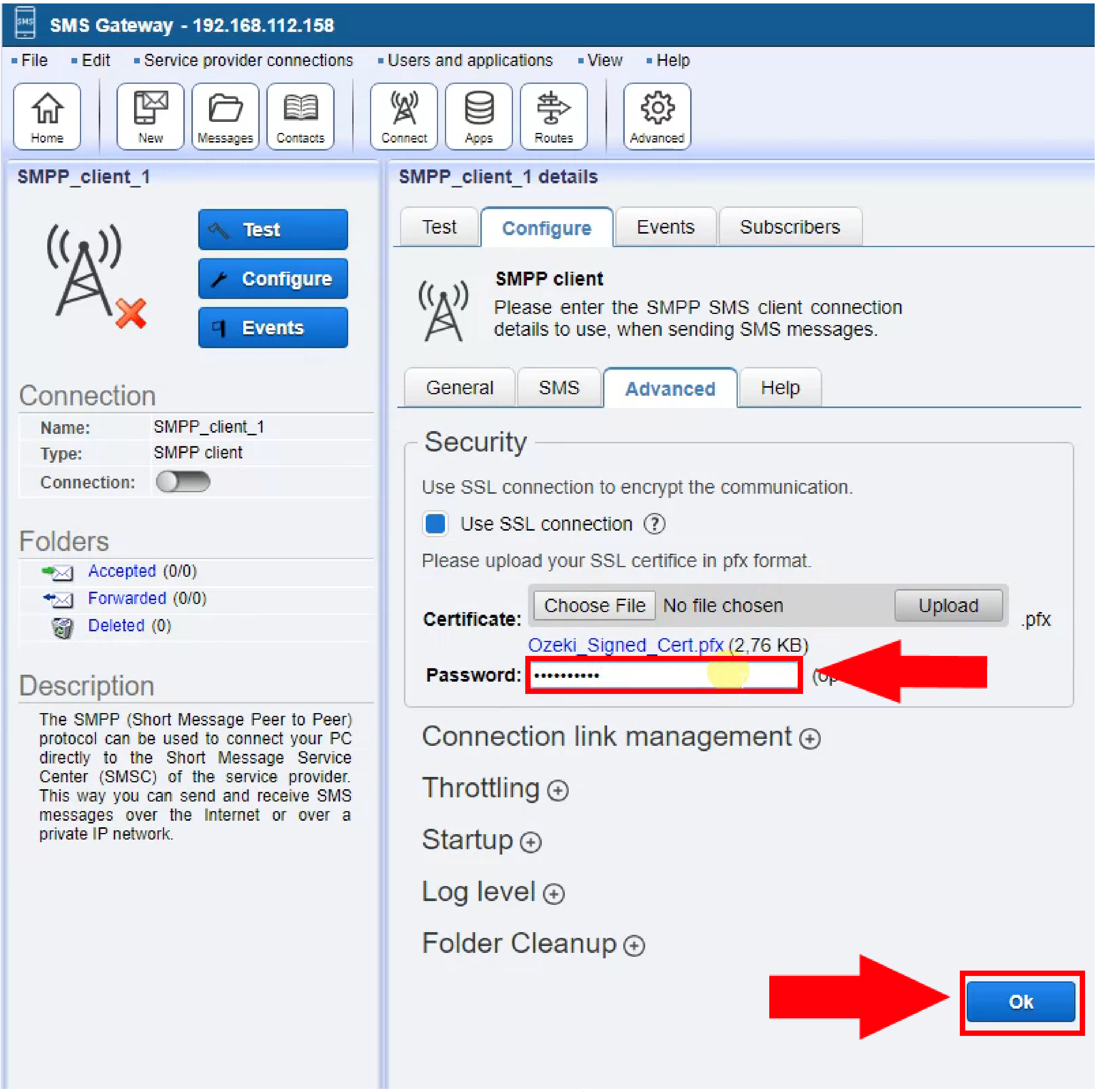
Please upload you SSL certifice in pfx format.

**Certificate:** No file chosen .pfx
Ozeki_Signed_Cert.pfx (2,76 KB)

**Password:** (optional) ?

Figure 7 - Upload cerificate

Provide the certificate password and then click OK.



**SMPP client**

Please enter the SMPP SMS client connection details to use, when sending SMS messages.

Connection

| | |
|---|---|
| Name: | SMPP_client_1 |
| Type: | SMPP client |
| Connection: | |

Folders

- Accepted (0/0)
- Forwarded (0/0)
- Deleted (0)

Description

The SMPP (Short Message Peer to Peer) protocol can be used to connect your PC directly to the Short Message Service Center (SMSC) of the service provider. This way you can send and receive SMS messages over the Internet or over a private IP network.

Security

Use SSL connection to encrypt the communication.

☑ Use SSL connection ?

Please upload your SSL certifice in pfx format.

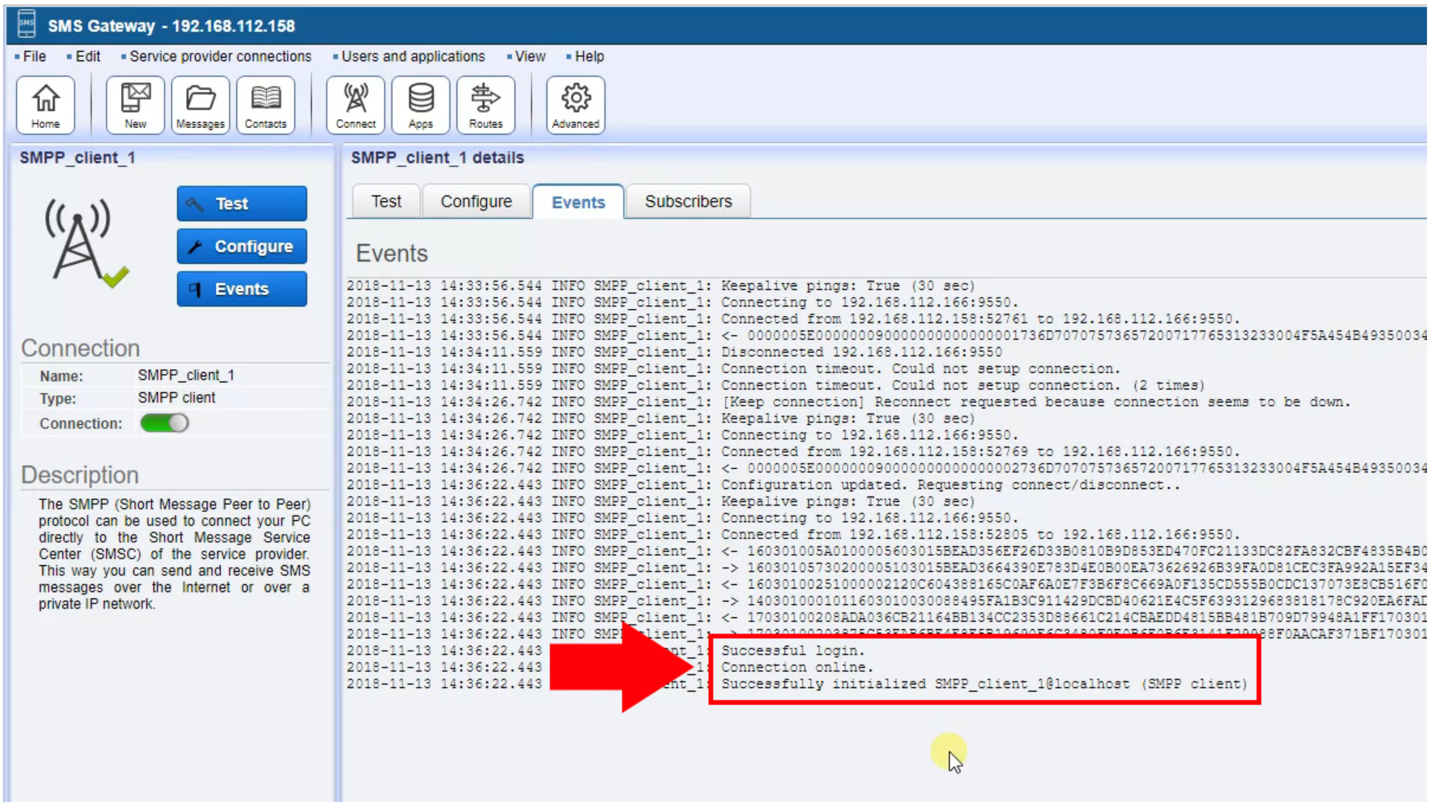
Certificate: Choose File No file chosen Upload .pfx

Ozeki_Signed_Cert.pfx (2,76 KB)

Password: •••••••••• (Op

Figure 8 - Provide certificate password

The secure connection successfully initialized.



Connection

Name: SMPP_client_1
Type: SMPP client
Connection:

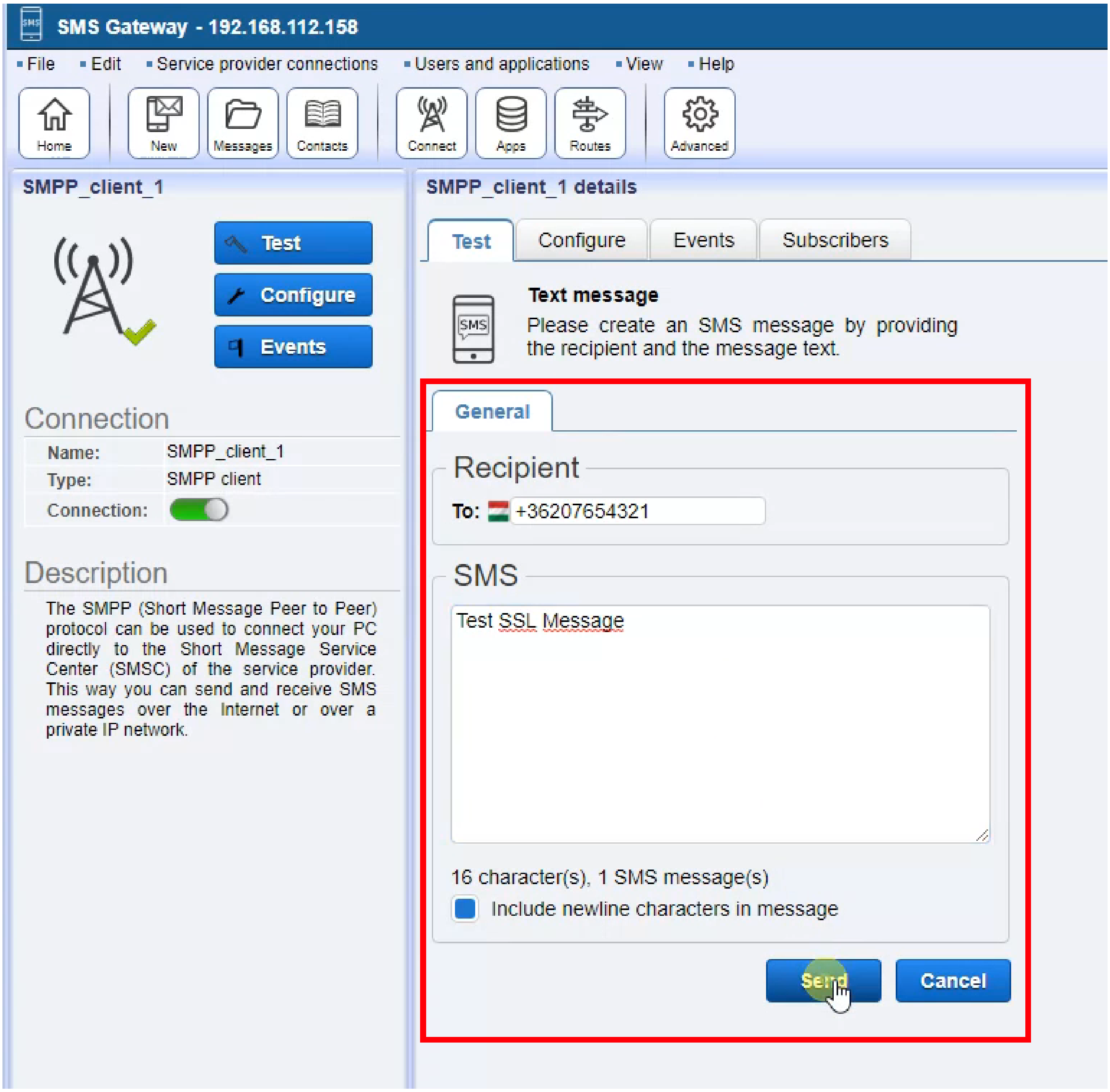
Description

The SMPP (Short Message Peer to Peer) protocol can be used to connect your PC directly to the Short Message Service Center (SMSC) of the service provider. This way you can send and receive SMS messages over the Internet or over a private IP network.

```
2018-11-13 14:33:56.544 INFO SMPP_client_1: Keepalive pings: True (30 sec)
2018-11-13 14:33:56.544 INFO SMPP_client_1: Connecting to 192.168.112.166:9550.
2018-11-13 14:33:56.544 INFO SMPP_client_1: Connected from 192.168.112.158:52761 to 192.168.112.166:9550.
2018-11-13 14:33:56.544 INFO SMPP_client_1: <- 0000005E000000090000000000000001736D707075736572007177653132330O4F5A454B49350034
2018-11-13 14:34:11.559 INFO SMPP_client_1: Disconnected 192.168.112.166:9550
2018-11-13 14:34:11.559 INFO SMPP_client_1: Connection timeout. Could not setup connection.
2018-11-13 14:34:11.559 INFO SMPP_client_1: Connection timeout. Could not setup connection. (2 times)
2018-11-13 14:34:26.742 INFO SMPP_client_1: [Keep connection] Reconnect requested because connection seems to be down.
2018-11-13 14:34:26.742 INFO SMPP_client_1: Keepalive pings: True (30 sec)
2018-11-13 14:34:26.742 INFO SMPP_client_1: Connecting to 192.168.112.166:9550.
2018-11-13 14:34:26.742 INFO SMPP_client_1: Connected from 192.168.112.158:52769 to 192.168.112.166:9550.
2018-11-13 14:34:26.742 INFO SMPP_client_1: <- 0000005E000000090000000000000002736D7070757365720071776531323300 4F5A454B49350034
2018-11-13 14:36:22.443 INFO SMPP_client_1: Configuration updated. Requesting connect/disconnect..
2018-11-13 14:36:22.443 INFO SMPP_client_1: Keepalive pings: True (30 sec)
2018-11-13 14:36:22.443 INFO SMPP_client_1: Connecting to 192.168.112.166:9550.
2018-11-13 14:36:22.443 INFO SMPP_client_1: Connected from 192.168.112.158:52805 to 192.168.112.166:9550.
2018-11-13 14:36:22.443 INFO SMPP_client_1: <- 160301005A0100005603015BEAD356EF26D33B0810B9D853ED470FC21133DC82FA832CBF4835B4B0
2018-11-13 14:36:22.443 INFO SMPP_client_1: -> 1603010573020000510301 5BEAD3664390E783D4E0B00EA73626926B39FA0D81CEC3FA992A15EF34
2018-11-13 14:36:22.443 INFO SMPP_client_1: <- 16030100251000002120C604388165C0AF6A0E7F3B6F8C669A0F135CD555B0CDC137073E8CB516F0
2018-11-13 14:36:22.443 INFO SMPP_client_1: -> 1403010001011603010030088495FA1B3C911429DCBD40621E4C5F63931296838181 78C920EA6FAD
2018-11-13 14:36:22.443 INFO SMPP_client_1: <- 17030100208ADA036CB21164BB134CC2353D88661C214CBAEDD4815BB481B709D79948A1FF170301
2018-11-13 14:36:22.443 INFO SMP  lient_1: -> 17030100203875C53FAB6BF4F8F5B10600F6C3480F0F0B6F0B6F3141F20088F0AACAF371BF170301
2018-11-13 14:36:22.443               nt_1: Successful login.
2018-11-13 14:36:22.443                  1: Connection online.
2018-11-13 14:36:22.443               nt_1: Successfully initialized SMPP_client_1@localhost (SMPP client)
```

Figure 9 - Successful secure connection

Now try to send a test message.



SMPP_client_1

Connection

Name: SMPP_client_1
Type: SMPP client
Connection:

Description

The SMPP (Short Message Peer to Peer) protocol can be used to connect your PC directly to the Short Message Service Center (SMSC) of the service provider. This way you can send and receive SMS messages over the Internet or over a private IP network.

SMPP_client_1 details

**Text message**
Please create an SMS message by providing the recipient and the message text.

Recipient

To: +36207654321

SMS

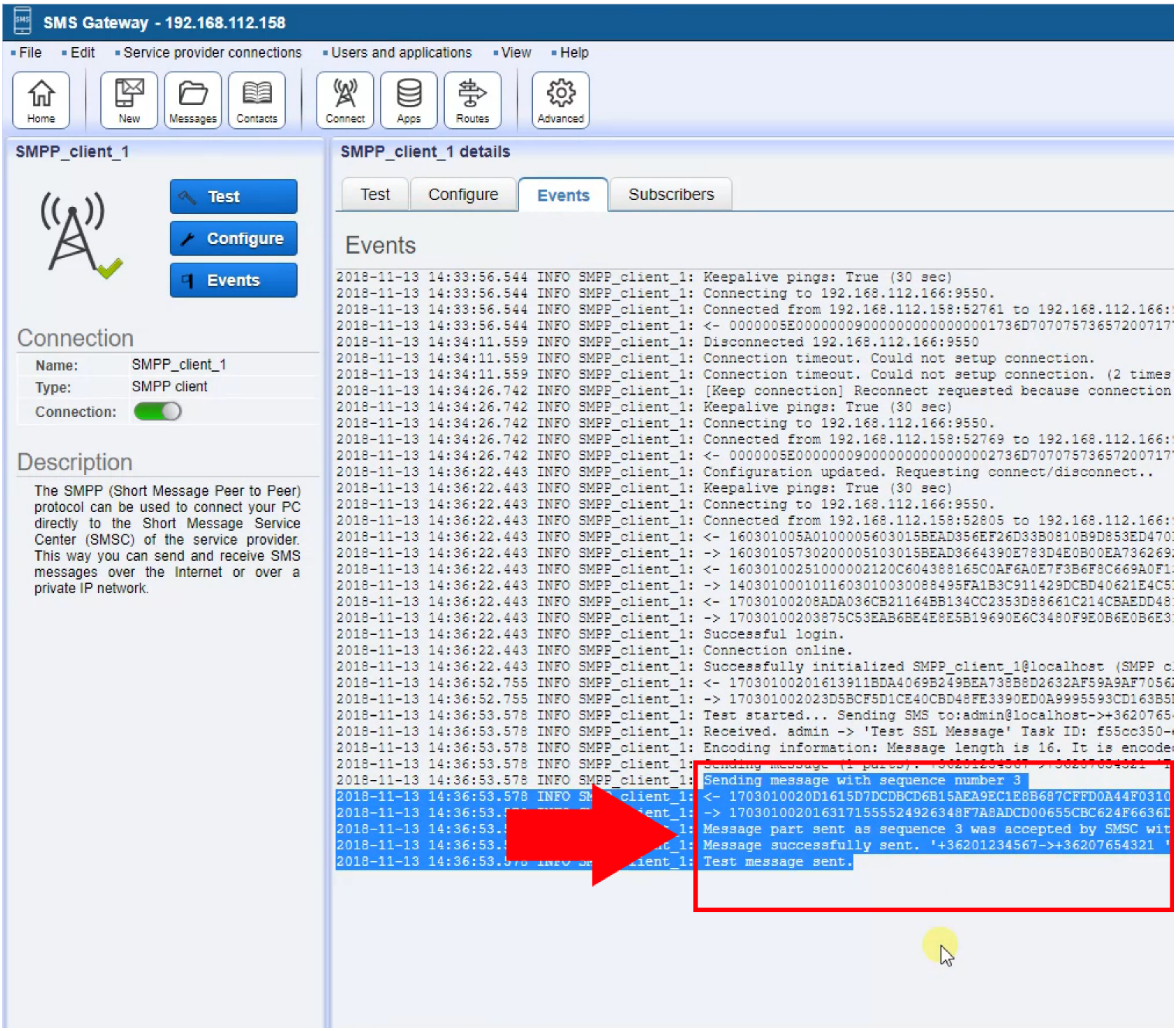
Test SSL Message

16 character(s), 1 SMS message(s)

☑ Include newline characters in message

Figure 10 - Send test message

The test message successfully sent.



Connection

Name: SMPP_client_1
Type: SMPP client
Connection:

Description

The SMPP (Short Message Peer to Peer) protocol can be used to connect your PC directly to the Short Message Service Center (SMSC) of the service provider. This way you can send and receive SMS messages over the Internet or over a private IP network.

Events

```
2018-11-13 14:33:56.544 INFO SMPP_client_1: Keepalive pings: True (30 sec)
2018-11-13 14:33:56.544 INFO SMPP_client_1: Connecting to 192.168.112.166:9550.
2018-11-13 14:33:56.544 INFO SMPP_client_1: Connected from 192.168.112.158:52761 to 192.168.112.166:
2018-11-13 14:33:56.544 INFO SMPP_client_1: <- 0000005E000000090000000000000001736D70707573657200717
2018-11-13 14:34:11.559 INFO SMPP_client_1: Disconnected 192.168.112.166:9550
2018-11-13 14:34:11.559 INFO SMPP_client_1: Connection timeout. Could not setup connection.
2018-11-13 14:34:11.559 INFO SMPP_client_1: Connection timeout. Could not setup connection. (2 times
2018-11-13 14:34:26.742 INFO SMPP_client_1: [Keep connection] Reconnect requested because connection
2018-11-13 14:34:26.742 INFO SMPP_client_1: Keepalive pings: True (30 sec)
2018-11-13 14:34:26.742 INFO SMPP_client_1: Connecting to 192.168.112.166:9550.
2018-11-13 14:34:26.742 INFO SMPP_client_1: Connected from 192.168.112.158:52769 to 192.168.112.166:
2018-11-13 14:34:26.742 INFO SMPP_client_1: <- 0000005E000000090000000000000002736D70707573657200717
2018-11-13 14:36:22.443 INFO SMPP_client_1: Configuration updated. Requesting connect/disconnect..
2018-11-13 14:36:22.443 INFO SMPP_client_1: Keepalive pings: True (30 sec)
2018-11-13 14:36:22.443 INFO SMPP_client_1: Connecting to 192.168.112.166:9550.
2018-11-13 14:36:22.443 INFO SMPP_client_1: Connected from 192.168.112.158:52805 to 192.168.112.166:
2018-11-13 14:36:22.443 INFO SMPP_client_1: <- 160301005A0100005603015BEAD356EF26D33B0810B9D853ED470
2018-11-13 14:36:22.443 INFO SMPP_client_1: -> 16030105730200005103015BEAD3664390E783D4E0B00EA736269
2018-11-13 14:36:22.443 INFO SMPP_client_1: <- 16030100251000002120C604388165C0AF6A0E7F3B6F8C669A0F1
2018-11-13 14:36:22.443 INFO SMPP_client_1: -> 1403010001011603010030088495FA1B3C911429DCBD40621E4C5
2018-11-13 14:36:22.443 INFO SMPP_client_1: <- 17030100208ADA036CB21164BB134CC2353D88661C214CBAEDD48
2018-11-13 14:36:22.443 INFO SMPP_client_1: -> 1703010020387 5C53EAB6BE4E8E5B19690E6C3480F9E0B6E0B6E3
2018-11-13 14:36:22.443 INFO SMPP_client_1: Successful login.
2018-11-13 14:36:22.443 INFO SMPP_client_1: Connection online.
2018-11-13 14:36:22.443 INFO SMPP_client_1: Successfully initialized SMPP_client_1@localhost (SMPP c
2018-11-13 14:36:52.755 INFO SMPP_client_1: <- 170301002016139 11BDA4069B249BEA738B8D2632AF59A9AF7056
2018-11-13 14:36:52.755 INFO SMPP_client_1: -> 17030100202 3D5BCF5D1CE40CBD48FE3390ED0A9995593CD163B5
2018-11-13 14:36:53.578 INFO SMPP_client_1: Test started... Sending SMS to:admin@localhost->+3620765
2018-11-13 14:36:53.578 INFO SMPP_client_1: Received. admin -> 'Test SSL Message' Task ID: f55cc350-
2018-11-13 14:36:53.578 INFO SMPP_client_1: Encoding information: Message length is 16. It is encode
2018-11-13 14:36:53.578 INFO SMPP_client_1: Sending message (1 parts): +36201234567->+36207654321 'T
2018-11-13 14:36:53.578 INFO SMPP_client_1: Sending message with sequence number 3
2018-11-13 14:36:53.578 INFO SMPP_client_1: <- 1703010020D1615D7DCDBCD6B15AEA9EC1E8B687CFFD0A44F0310
2018-11-13 14:36:53.578 INFO SMPP_client_1: -> 17030100201631715555249 26348F7A8ADCD00655CBC624F6636D
2018-11-13 14:36:53.578 INFO SMPP_client_1: Message part sent as sequence 3 was accepted by SMSC wit
2018-11-13 14:36:53.578 INFO SMPP_client_1: Message successfully sent. '+36201234567->+36207654321 '
2018-11-13 14:36:53.578 INFO SMPP_client_1: Test message sent.
```

Figure 11 - Message successfully sent

# SMPP Delivery Reports

# 1. Introduction

***SMPP supports delivery receipts / reports (DLRs) for SMS messages so that your application can determine delivery outcomes.***

The returning of a message delivery receipt / report (DLR) is dependent on the value set in the registered_delivery field of the message originally sent from the ESME to the MC in an submit_sm operation. This can be configured for non-delivery and delivery-only scenarios that can result in circumstances where the receipt will not be returned. Message delivery receipts are returned in the deliver_sm and data_sm operations.

The following fields are relevant in the deliver_sm and data_sm operations when used for transmitting delivery receipts.

- source address (i.e. source_addr_ton, source_addr_npi, source_addr) - The source address will be taken from the destination address of the original short message, which generated the delivery receipt. The receipt appears as if it emanated from the recipient of the original registered message.
- destination address (i.e. dest_addr_ton, dest_addr_npi, destination_addr) - The destination address will be taken from the source address of the original short message, which generated the delivery receipt. The receipt is addressed to the SME that originally sent the registered message.
- esm_class - Bit 2 of the esm_class is set to 1 to indicate that the message is an MC Delivery Receipt. If bit 5 is set then the message is an Intermediate Notification.
- message_state TLV - Indicates the final state of the original message. See **Message states** below.
- network_error_code TLV - See **Error codes** below.
- receipted_message_id TLV - Message ID that was returned to the ESME by the MC in the submit_sm_resp PDU.

## 2. MC Delivery Receipt

This message type is used to carry a MC delivery receipt. The MC, on detecting the final state of a registered message, would normally generate a new receipt message addressed to the originator of the first message. The MC Delivery Receipt is then delivered to the ESME in a deliver_sm or data_sm operation.

**ESME-to-MC**: *Set bits 0 and 1 in a submit_sm operation registered_delivery field to request an MC Delivery Receipt.*

| Bit 1 | Bit 0 | Meaning |
|---|---|---|
| 0 | 0 | no receipt |
| 0 | 1 | receipt requested when final outcome is delivery success or failure |
| 1 | 0 | receipt requested when final outcome is delivery failure |
| 1 | 1 | receipt requested when final outcome is delivery success (SMPP v5 only) |

**MC-to-ESME**: *Bit 2 in the esm_class field of a deliver_sm indicates the receipt is an MC Delivery Receipt.*

## 3. Intermediate Notification

An intermediate notification is a special form of message that the MC may send to an ESME for a mobile terminated message delivery. It provides an intermediate status of a message delivery attempt.

Typical uses are to report the outcome of delivery attempts made during the message’s retry lifetime within the MC. This could be used to track the various reasons why a message is not delivered to its destination and use this to profile the subscriber’s availability.

Support for Intermediate Notification functionality is specific to the MC implementation and the MC Service Provider and is beyond the scope of this specification.

**ESME-to-MC**: *Set bit 4 in a submit_sm PDU registered_delivery field to request an Intermediate Notification.*

**MC-to-ESME**: *Bit 5 in the esm_class field of a deliver_sm indicates the receipt is an Intermediate Notification.*

# 4. Receipt in short_message field

Many pre-v3.4 APIs and Message Centers supporting v3.3 are likely to have a means of passing receipt information within the short_message field. This applies to MC Delivery Receipts and Intermediate Notifications. The format specifics of this information are SMS gateway and SMSC platform specific and beyond the scope of the specification. However, the following shows the approach typically taken:

id:123A456B sub:1 dlvrd:1 submit date:1702281424 done date:1702281424 stat:DELIVRD err:0 text:

The fields are specified as follows:

| Field | Size (octets) | Description |
|---|---|---|
| id | Variable | The message ID allocated to the message by the SMSC when originally submitted. |
| sub | 3 | Number of short messages originally submitted. The value may be padded with leading zeros. |
| dlvrd | 3 | Number of short messages delivered. The value may be padded with leading zeros. |
| `submit date` | 10 | The time and date at which the short message was submitted. In the case of a message which has been replaced, this is the date that the original message was replaced. The format is as follows:<br>**YYMMDDhhmm** where:<br>**YY** last two digits of the year (00-99) MM = month (01-12)<br>**DD** day (01-31)<br>**hh** hour (00-23)<br>**mm** minute (00-59 |
| `done date` | 10 | The time and date at which the short message reached it's final state. The format is the same as for the submit date. |
| stat | 7 | The final status of the message. See **Message states** below. State text may be abbreviated. |
| err | 3 | A network or SMSC error code for the message. See **Error codes** below. |
| text | 20 | Unused field, result will be blank. |

# 5. Ozeki SMPP improvements

As we have implemented a very large number of SMPP connectoins we have found the following issues in various implementations:

**Finding 1:**

The value of the ID field in the delivery report (which we call Submit Reference in Ozeki) is often different from the ID we receive from the SMS service provider. The most common difference is that the original ID is returned as a standard integer number and the ID in the delivery report is returned as a hexadecimal number. This can also happen vice versa. The good thing, is that in this sutation, when converted back the two numbers match, so the delivery reprots can match. The Ozeki SMS implementations perform various checks, and can handle the described situation properly.

**Finding 2:**

The value of the date fields often come in in non standard format. Ozeki currently parses the date fields using the following patterns. You may also define a custom date field pattern on the configuration form of the software.

- "yyMMddHHmm",
- "yyMMddHHmmss",
- "dd-MMM-yyHH:mm",
- "dd-MMM-yyHH:mm:ss",
- "dd-MMM-yy HH:mm",
- "dd-MMM-yy HH:mm:ss",
- "yyyyMMddHHmmss",
- "yyyyMMddHHmm",
- custom

# 6. Message states

The following is a list of allowable states for a short message. The MC returns the message_state value to the ESME as part of the query_sm_resp, query_broadcast_sm_resp or deliver_sm delivery receipt PDU.

Intermediate states are states that can change. Final states are states that represent an end of life state for a message.

For example, a message in retry may return an ENROUTE state. At some point in the future, this message will either expire or be delivered. The state will then progress to EXPIRED or DELIVERED. Thus a message in ENROUTE state is said to be in an intermediate state.

A message in DELIVERED or EXPIRED state cannot progress to another state. These states are therefore final states.

| Message State | Value | Type |
|---|---|---|
| SCHEDULED | 0 | Intermediate |
| The message is scheduled. Delivery has not yet been initiated. A message submitted with a scheduled delivery time may return this state when queried. This value was added for SMPP v5.0. MCs supporting earlier version of SMPP v3.3 and SMPP v3.4 are likely to return ENROUTE for scheduled messages. | | |
| ENROUTE<br>or EN_ROUTE | 1 | Intermediate |
| The message is in enroute state. This is a general state used to describe a message as being active within the MC. The message may be in retry or dispatched to a mobile network for delivery to the mobile. | | |
| DELIVERED | 2 | Final |
| Message is delivered to destination. The message has been delivered to the destination. No further deliveries will occur. | | |
| EXPIRED | 3 | Final |
| Message validity period has expired. The message has failed to be delivered within its validity period and/or retry period. No further delivery attempts will be made. | | |
| DELETED | 4 | Final |
| Message has been deleted. The message has been cancelled or deleted from the MC. No further delivery attempts will take place. | | |
| UNDELIVERABLE | 5 | Final |
| Message is undeliverable. The message has encountered a delivery error and is deemed permanently undeliverable. No further delivery attempts will be made. Certain network or MC internal errors result in the permanent non-delivery of a message. Examples of such errors would be an unknown subscriber or network error that indicated that the given destination mobile was denied SMS service or could not support SMS. | | |

# 7. SMPP Delivery Receipt Error Codes

Error codes returned in delivery receipts are used to indicate any error situation encountered when attempting to deliver a message. Error codes are SMS gateway and SMSC platform specific. However, the following shows an approach often taken:

| Code | Meaning |
|---|---|
| 1 | MT number is unknown in the MT network’s HLR |
| 2 | MT number is unknown in the MT network’s HLR |
| 5 | MT number is unknown in the MT network’s MSC |
| 9 | MT number is classed as an illegal subscriber in the MT network’s MSC |
| 11 | MT HLR sends back a “Teleservice not provisioned” error in responseto the SRI |
| 12 | MT handset is listed as an Illegal device on the MSC. |
| 13 | Customer is barred according to the MT HLR from receiving SMS |
| 15 | MT customer is part of a CUG that is not allowed to receive SMS |
| 21 | SMS not supported in the MT network. |
| 22 | SMS not supported in the MT MSC |
| 31 | MT handset is busy. The signalling control channel is in use.<br>(Probably receiving another SMS at the same time) |
| 32 | GPRS – As above |
| 34 | System failure in the MT network. |
| 35 | Data Missing in either the MT HLR or MSC |
| 36 | Unexpected data value received in response to a FSM or SRI |
| 40 | Memory capacity exceeded on the MT handset |
| 41 | MT handset protocol error |
| 42 | MT handset is not equipped to support SMS |
| 43 | Short message type “0” not supported by the MT handset. |
| 44 | MT network unable to replace the SMS on the MT customers’ handset |
| 45 | Unspecified protocol error on the MT handset |

```
46      Message class not supported on the MT handset
47      Unspecified DCS (Data coding scheme) error on the MT handset
48      Transfer layer PDU not supported by MT handset
49      SIM card full on MT handset
50      MT handset’s SIM is unable to store the message
51      Error in MT handset
52      Memory capacity exceeded on the MT handset
53      SIM application toolkit busy on the MS handset
54      SIM data download error on the MT customer’s handset
55      Unspecified MS handset error
60      Absent subscriber. No reason known
61      Absent subscriber due to phone being switched off
62      Absent subscriber due to phone out of coverage/flat battery
63      Absent subscriber due to roaming restriction/restricted area
64      Absent subscriber due to being deregistered in the HLR
65      Absent subscriber due to being purged in the VLR (off for 24+ hours)
66      Absent subscriber (GPRS) cannot be paged by the SGSN
67      Absent subscriber due to GPRS detached
68      Absent subscriber due to deregistration in the HLR (GPRS)
69      Absent subscriber due to GPRS MS purged in VLR
70      Absent subscriber due to unidentified subscriber on the MSC that the FSM was sent to.
71      Absent subscriber due to unidentified subscriber on the SGSN
```

**Source: https://smpp.org/smpp-delivery-receipt.html** - The original article can be found at smpp.org, at the above URL. This article has been improved and extended to support more information, that we thought was improtant to add.

# UCP SMS client

***In this chapter, we will connect an UCP sms client (UCP sms link) to an SMS service provider and show you all steps neccessary to setup this mobile network connection. There are two UCP sms connection modes: one uses a simple TCP/IP link, the other uses SSL/TLS over TCP/IP. The connection steps presented in this chapter are for a simple TCP/IP link.***

## What is an UCP/EMI sms connection?

An UCP/EMI sms connection is mobile network link that you can setup to connect your SMS gateway over the Internet to the Short Message Service Center (SMSC) of a mobile network operator. UCP/EMI is the communication protocol used over this link.

## What does UCP/EMI stand for?

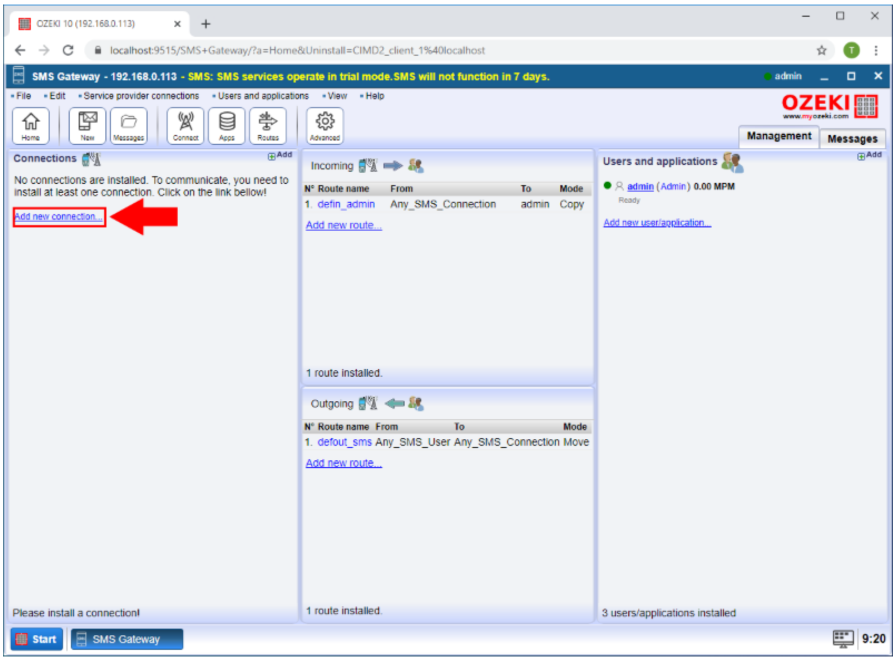
UCP stands for Universal Computer Protocol. EMI stands for External Machine Interface. This is the name of the protocol developed by CMG Wirless Solutions Inc.

## Connection steps

- Open the https://localhost:9515 URL
- Login with your username and password
- Open "Add new connection"
- Click on "UCP client"
- Type in the Hotname and port
- Enter the UCP OAdC and password
- Enter the telephone number
- Send a test message

## Detailed setup

The following screenshots show you the detailed cofiguration steps to take to setup an UCP SMS connection in Ozeki SMS gateway. These steps are easy to follow. The setup procedure usually takes around 2-3 minutes.



SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.

No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

Please install a connection!

Incoming

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defin_admin | Any_SMS_Connection | admin | Copy |

Add new route...

1 route installed.

Outgoing

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move |

Add new route...

1 route installed.

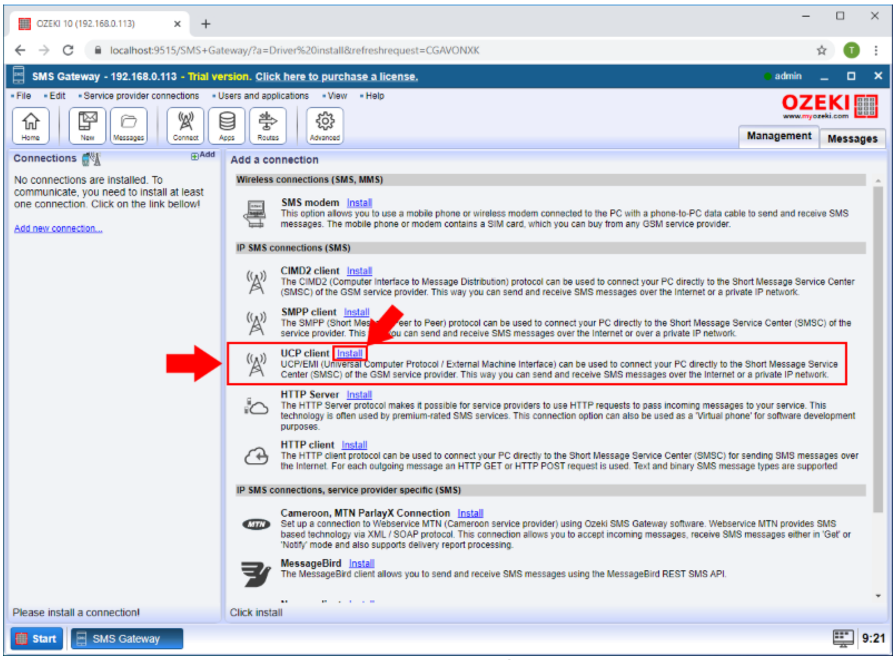
admin (Admin) 0.00 MPM
Ready

Add new user/application...

3 users/applications installed

**Figure 1 -** Creating a new connection

Select the UCP Client connection from the list and click Install (Figure 2).



No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

Please install a connection!

Add a connection

Wireless connections (SMS, MMS)

SMS modem Install
This option allows you to use a mobile phone or wireless modem connected to the PC with a phone-to-PC data cable to send and receive SMS messages. The mobile phone or modem contains a SIM card, which you can buy from any GSM service provider.

IP SMS connections (SMS)

CIMD2 client Install
The CIMD2 (Computer Interface to Message Distribution) protocol can be used to connect your PC directly to the Short Message Service Center (SMSC) of the GSM service provider. This way you can send and receive SMS messages over the Internet or a private IP network.

SMPP client Install
The SMPP (Short Mes eer to Peer) protocol can be used to connect your PC directly to the Short Message Service Center (SMSC) of the service provider. This ou can send and receive SMS messages over the Internet or over a private IP network.

UCP client Install
UCP/EMI (Universal Computer Protocol / External Machine Interface) can be used to connect your PC directly to the Short Message Service Center (SMSC) of the GSM service provider. This way you can send and receive SMS messages over the Internet or a private IP network.

HTTP Server Install
The HTTP Server protocol makes it possible for service providers to use HTTP requests to pass incoming messages to your service. This technology is often used by premium-rated SMS services. This connection option can also be used as a 'Virtual phone' for software development purposes.

HTTP client Install
The HTTP client protocol can be used to connect your PC directly to the Short Message Service Center (SMSC) for sending SMS messages over the Internet. For each outgoing message an HTTP GET or HTTP POST request is used. Text and binary SMS message types are supported

IP SMS connections, service provider specific (SMS)

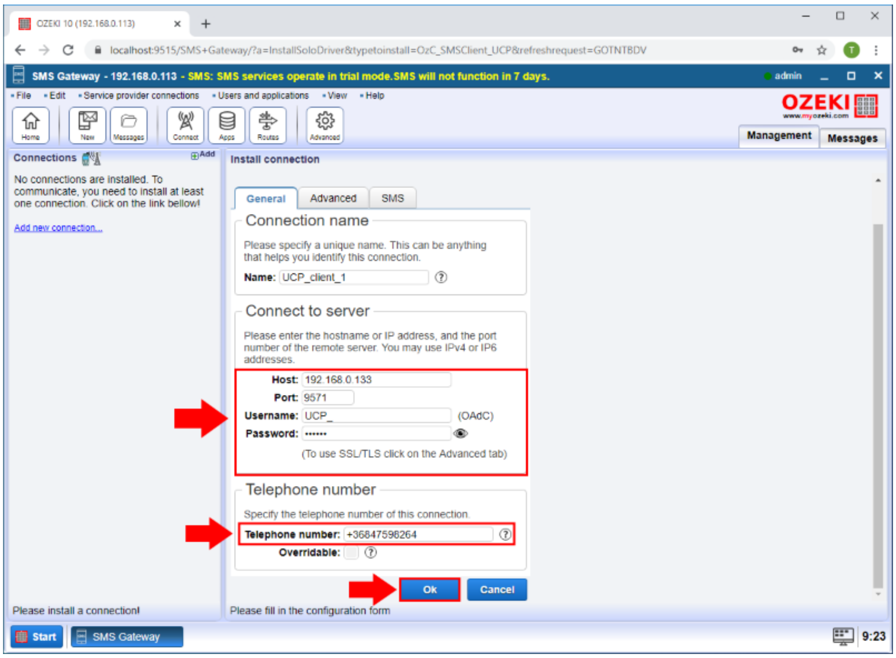
Cameroon, MTN ParlayX Connection Install
Set up a connection to Webservice MTN (Cameroon service provider) using Ozeki SMS Gateway software. Webservice MTN provides SMS based technology via XML / SOAP protocol. This connection allows you to accept incoming messages, receive SMS messages either in 'Get' or 'Notify' mode and also supports delivery report processing.

MessageBird Install
The MessageBird client allows you to send and receive SMS messages using the MessageBird REST SMS API.

Click install

**Figure 2 -** Install the UCP connection

Now enter the contact details like Host, port, username and password. And enter the phone number that the SMS Gateway uses to send the message. Finally click on the OK button (Figure 3).


OZEKI 10 (192.168.0.113)
localhost:9515/SMS+Gateway/?a=InstallSoloDriver&typetoinstall=OzC_SMSClient_UCP&refreshrequest=GOTNTBDV
SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
OZEKI
www.myozeki.com
Management
Messages
Connections
Add
General
Advanced
SMS
Ok
Cancel
Start
SMS Gateway
9:23

No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

Please install a connection!

Install connection

Connection name

Please specify a unique name. This can be anything that helps you identify this connection.

Name: UCP_client_1

Connect to server

Please enter the hostname or IP address, and the port number of the remote server. You may use IPv4 or IP6 addresses.

Host: 192.168.0.133
Port: 9571
Username: UCP_ (OAdC)
Password: ••••••
(To use SSL/TLS click on the Advanced tab)

Telephone number

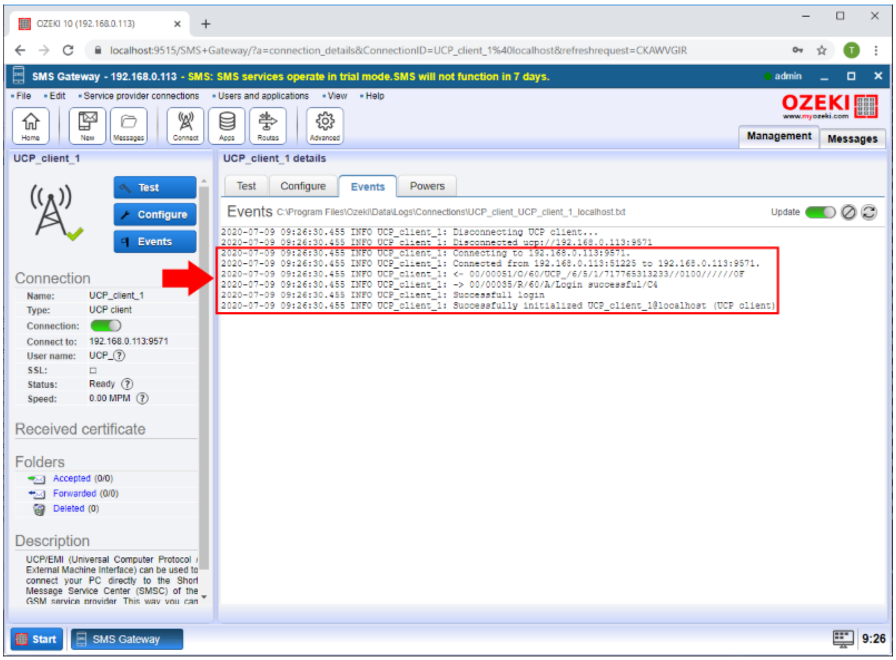
Specify the telephone number of this connection.

Telephone number: +36847598264
Overridable: □

Please fill in the configuration form

**Figure 3 -** Providing UCP Client credentials

In the Events tab you can see the logging of the latest server events. As you can see on the Figure 4 the login is succesful.



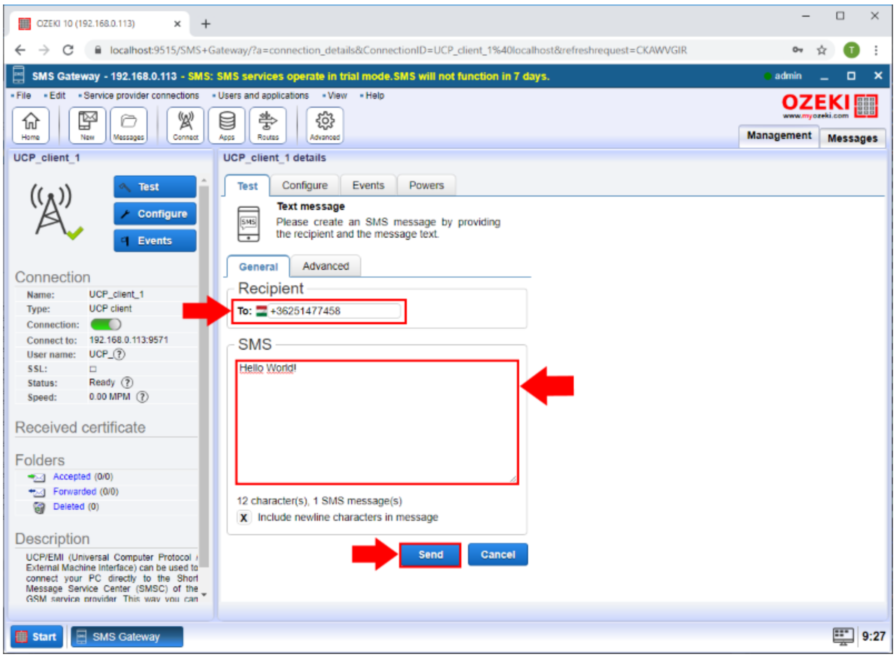
Events C:\Program Files\Ozeki\Data\Logs\Connections\UCP_client_UCP_client_1_localhost.txt

```
2020-07-09 09:26:30.455 INFO UCP_client_1: Disconnecting UCP client...
2020-07-09 09:26:30.455 INFO UCP_client_1: Disconnected ucp://192.168.0.113:9571
2020-07-09 09:26:30.455 INFO UCP_client_1: Connecting to 192.168.0.113:9571.
2020-07-09 09:26:30.455 INFO UCP_client_1: Connected from 192.168.0.113:51225 to 192.168.0.113:9571.
2020-07-09 09:26:30.455 INFO UCP_client_1: <- 00/00051/O/60/UCP_/6/5/1/717765313233//0100//////0F
2020-07-09 09:26:30.455 INFO UCP_client_1: -> 00/00035/R/60/A/Login successful/C4
2020-07-09 09:26:30.455 INFO UCP_client_1: Successfull login
2020-07-09 09:26:30.455 INFO UCP_client_1: Successfully initialized UCP_client_1@localhost (UCP client)
```

Connection
Name: UCP_client_1
Type: UCP client
Connect to: 192.168.0.113:9571
User name: UCP_
Status: Ready
Speed: 0.00 MPM

**Figure 4** - Successful login

From the Test tab you can send test SMS message. Provide the recipient address, the message and click on the Send button (Figure 5).



Text message
Please create an SMS message by providing the recipient and the message text.

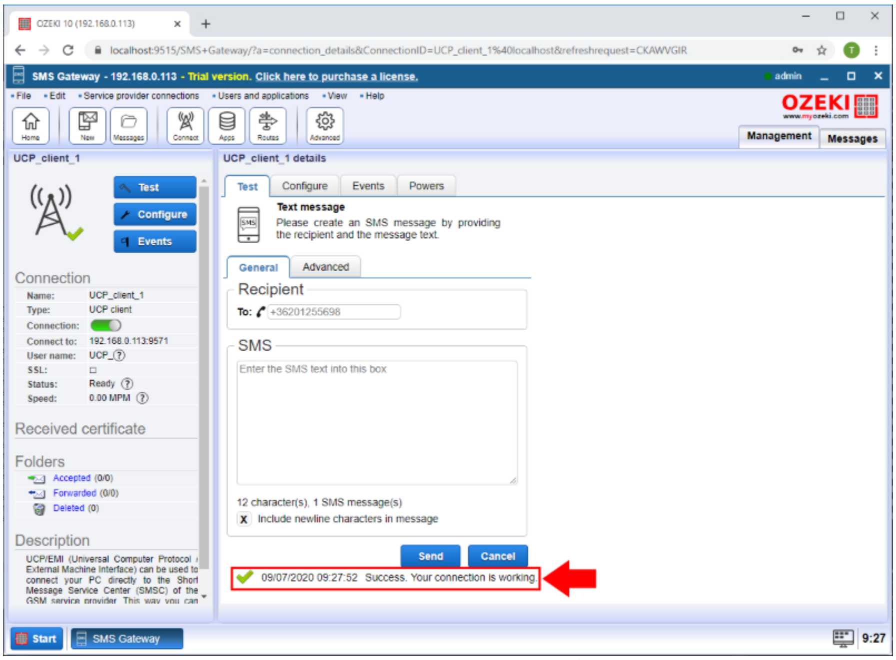
Recipient
To: +36251477458

SMS
Hello World!

12 character(s), 1 SMS message(s)
☒ Include newline characters in message

**Figure 5** - Sending a test message

Finally you can see if the message is sent successfully (Figure 6).



Connection

| | |
|---|---|
| Name: | UCP_client_1 |
| Type: | UCP client |
| Connection: | |
| Connect to: | 192.168.0.113:9571 |
| User name: | UCP_ |
| SSL: | □ |
| Status: | Ready |
| Speed: | 0.00 MPM |

Received certificate

Folders

Accepted (0/0)
Forwarded (0/0)
Deleted (0)

Description

UCP/EMI (Universal Computer Protocol / External Machine Interface) can be used to connect your PC directly to the Short Message Service Center (SMSC) of the GSM service provider. This way you can

Text message

Please create an SMS message by providing the recipient and the message text.

Recipient

To: +36201255698

SMS

Enter the SMS text into this box

12 character(s), 1 SMS message(s)

☒ Include newline characters in message

✓ 09/07/2020 09:27:52 Success. Your connection is working.

**Figure 6** - Message successfully sent

# Example UCP PDU

The following example UCP PDU is a **login PDU** sent by the SMS gateway to the SMSC

<- 00/00051/O/60/ucp1/6/5/1/717765313233//0100//////41
-> 00/00035/R/60/A/Login successful/C4

The next is an **UCP submit message PDU**, used to send an SMS to the SMSC.

Sending message (1 parts). +3620123546->+3620123456 'Hello world'
<-
01/00106/O/51/003620123456/3620123546//1//7/////////////3//48656C6C6F20776F726C64////////1139//020100///A1
-> 01/00045/R/51/A//003620123456:200718075707/A2
Message part sent as sequence 01 was accepted by remote server. with Reference Id: '200718075707'.

The above PDUs were extracted from the Ozeki SMS Gateway logs.

# Where can I find the UCP protocol specification?

- ucp-emi-specification-4.7.pdf
- ucp-emi-specification-5.3.pdf

# CIMD2 connection

***The CIMD2 protocol is and SMS protocol that enables you to connect to Nokia SMSCs. In Europe many mobile networks use this protocol to offer IP SMS services to customers. This guide explains, how you can connect to the mobile network that offers IPS SMS services through the CIMD2 protocol. You will also find direct download links to the protocol specifications.***

## What is a CIMD2 connection?

A CIMD2 connection is TCP/IP connection to the short message service centre (SMSC) of a mobile network operator. It allows your Ozeki SMS Gateway system to connect directly your mobile network operator's SMSC.

## What does CIMD2 stand for?

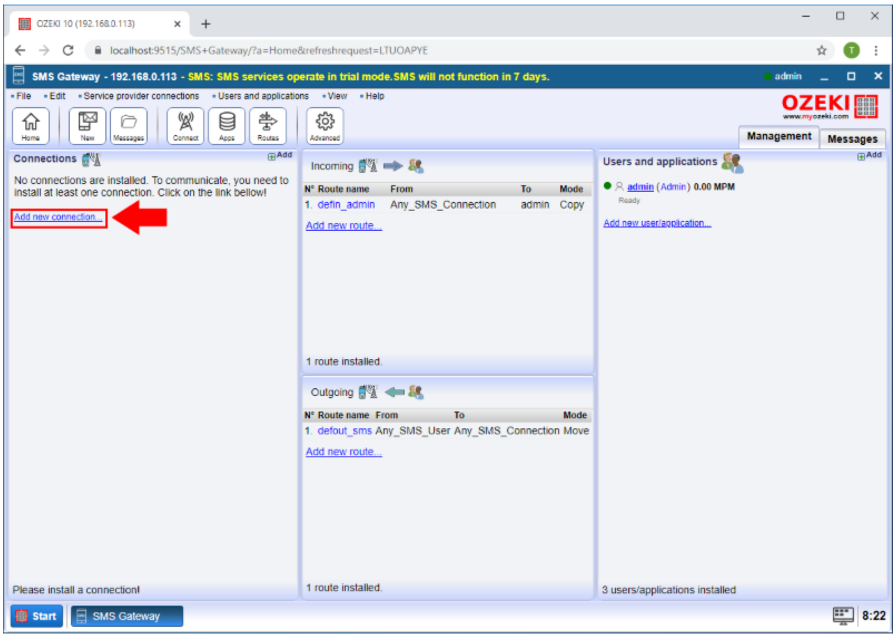
CIMD stands for Computer Interface to Message Distribution. CIMD2 stands for version two of this protocol. This is protocol is implemented by Ozeki SMS Gateway.

## Connection steps

- Navigate to https://localhost:9515 in your browser
- Login using your username and password
- In the connections panel click "Add new connection"
- Select "CIMD2 client"
- Type in the cimd2 server details
- Sepcify the telephone number
- Click ok to save your entries
- Send a test SMS message

## Screenshots of the cimd2 connection setup procedure

The first screenshot shows you how to start the connection procedure to connect your SMS gateway system to ae CIMD2 service provider. To do this, you need to create an CIMD2 client connection. Click on the "Add new connection ..." as you can see on the Figure 1.



No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

Please install a connection!

Incoming

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defin_admin | Any_SMS_Connection | admin | Copy |

Add new route...

1 route installed.

Outgoing

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move |

Add new route...

1 route installed.

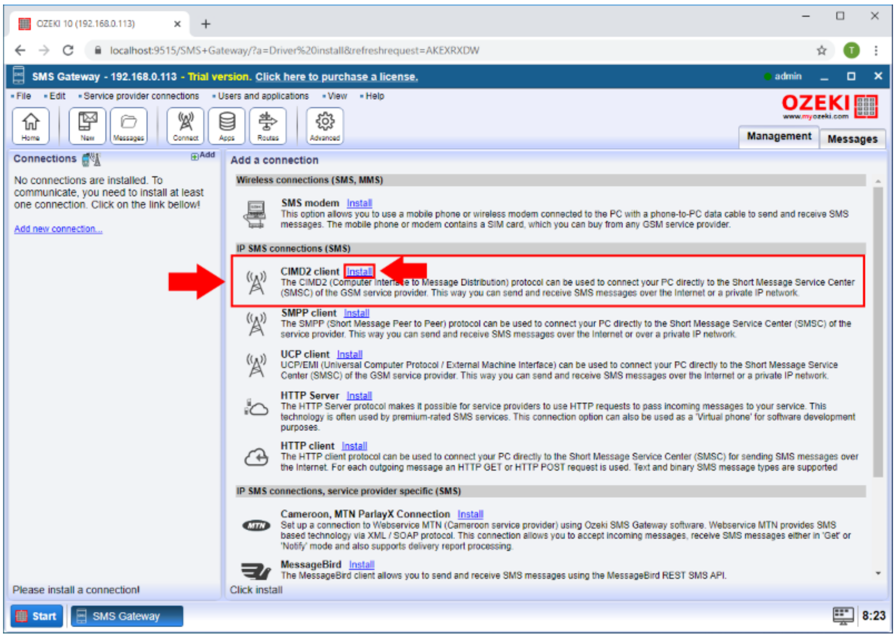
admin (Admin) 0.00 MPM
Ready

Add new user/application...

3 users/applications installed

**Figure 1** - Creating a new connection

Select the CIMD2 Client connection from the list and click Install (Figure 2).



SMS Gateway - 192.168.0.113 - Trial version. Click here to purchase a license.

Connections

No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

Please install a connection!

Add a connection

Wireless connections (SMS, MMS)

SMS modem Install
This option allows you to use a mobile phone or wireless modem connected to the PC with a phone-to-PC data cable to send and receive SMS messages. The mobile phone or modem contains a SIM card, which you can buy from any GSM service provider.

IP SMS connections (SMS)

CIMD2 client Install
The CIMD2 (Computer Interface to Message Distribution) protocol can be used to connect your PC directly to the Short Message Service Center (SMSC) of the GSM service provider. This way you can send and receive SMS messages over the Internet or a private IP network.

SMPP client Install
The SMPP (Short Message Peer to Peer) protocol can be used to connect your PC directly to the Short Message Service Center (SMSC) of the service provider. This way you can send and receive SMS messages over the Internet or over a private IP network.

UCP client Install
UCP/EMI (Universal Computer Protocol / External Machine Interface) can be used to connect your PC directly to the Short Message Service Center (SMSC) of the GSM service provider. This way you can send and receive SMS messages over the Internet or a private IP network.

HTTP Server Install
The HTTP Server protocol makes it possible for service providers to use HTTP requests to pass incoming messages to your service. This technology is often used by premium-rated SMS services. This connection option can also be used as a 'Virtual phone' for software development purposes.

HTTP client Install
The HTTP client protocol can be used to connect your PC directly to the Short Message Service Center (SMSC) for sending SMS messages over the Internet. For each outgoing message an HTTP GET or HTTP POST request is used. Text and binary SMS message types are supported

IP SMS connections, service provider specific (SMS)

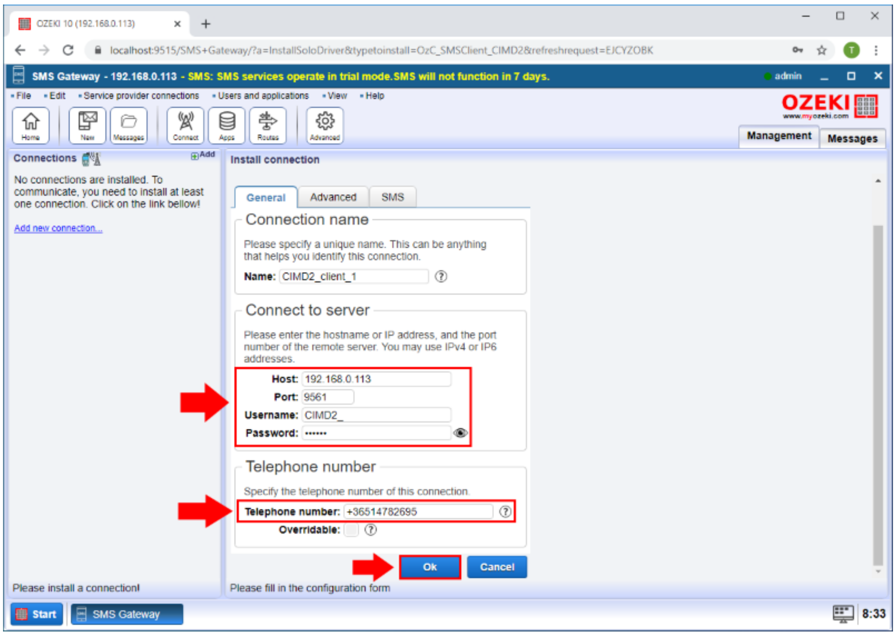
Cameroon, MTN ParlayX Connection Install
Set up a connection to Webservice MTN (Cameroon service provider) using Ozeki SMS Gateway software. Webservice MTN provides SMS based technology via XML / SOAP protocol. This connection allows you to accept incoming messages, receive SMS messages either in 'Get' or 'Notify' mode and also supports delivery report processing.

MessageBird Install
The MessageBird client allows you to send and receive SMS messages using the MessageBird REST SMS API.

Click install

**Figure 2** - Install the CIMD2 connection

Now enter the contact details like Host, port, username and password. And enter the phone number that the SMS Gateway uses to send the message. Finally click on the OK button (Figure 3).



SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.

Connections

No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

Please install a connection!

Install connection

Connection name

Please specify a unique name. This can be anything that helps you identify this connection.

Name: CIMD2_client_1

Connect to server

Please enter the hostname or IP address, and the port number of the remote server. You may use IPv4 or IP6 addresses.

Host: 192.168.0.113
Port: 9561
Username: CIMD2_
Password: ••••••

Telephone number

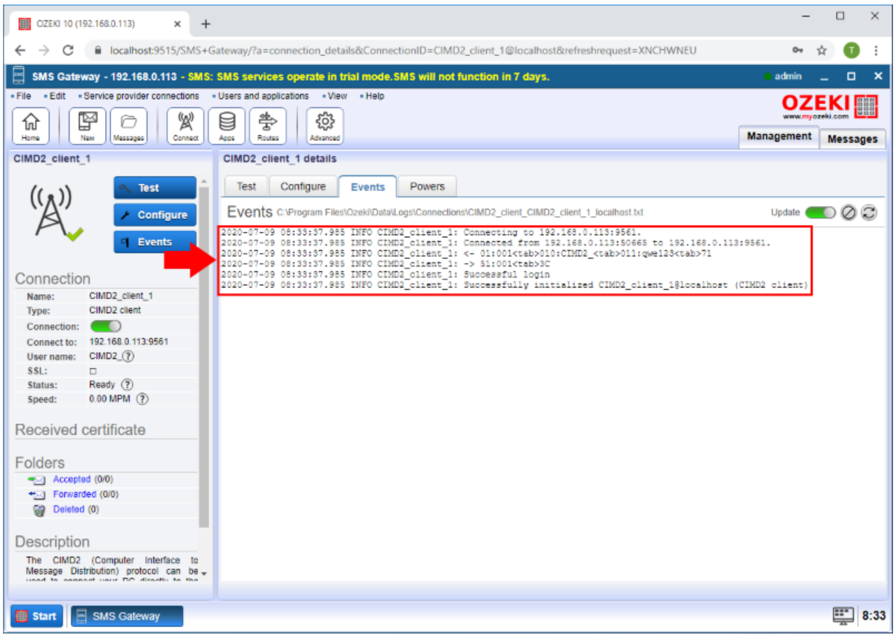
Specify the telephone number of this connection.

Telephone number: +36514782695
Overridable: □

Please fill in the configuration form

**Figure 3** - Providing CIMD2 Client credentials

In the Events tab you can see the logging of the latest server events. As you can see on the Figure 4 the login is succesful.



CIMD2_client_1 details

Events C:\Program Files\Ozeki\Data\Logs\Connections\CIMD2_client_CIMD2_client_1_localhost.txt

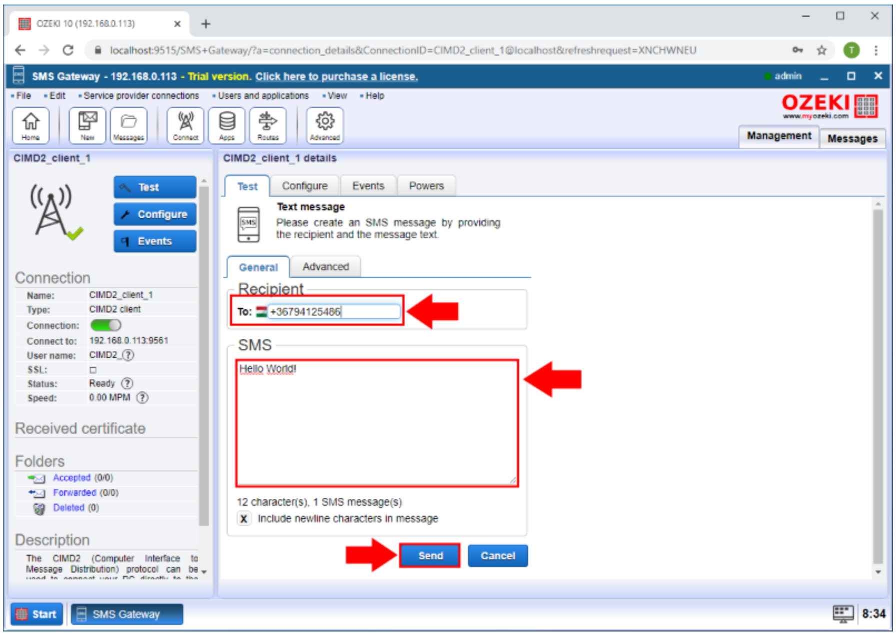
```
2020-07-09 08:33:37.985 INFO CIMD2_client_1: Connecting to 192.168.0.113:9561.
2020-07-09 08:33:37.985 INFO CIMD2_client_1: Connected from 192.168.0.113:50665 to 192.168.0.113:9561.
2020-07-09 08:33:37.985 INFO CIMD2_client_1: <- 01:001<tab>010:CIMD2_<tab>011:qwe123<tab>71
2020-07-09 08:33:37.985 INFO CIMD2_client_1: -> 51:001<tab>3C
2020-07-09 08:33:37.985 INFO CIMD2_client_1: Successful login
2020-07-09 08:33:37.985 INFO CIMD2_client_1: Successfully initialized CIMD2_client_1@localhost (CIMD2 client)
```

Connection

| | |
|---|---|
| Name: | CIMD2_client_1 |
| Type: | CIMD2 client |
| Connect to: | 192.168.0.113:9561 |
| User name: | CIMD2_ |
| Status: | Ready |
| Speed: | 0.00 MPM |

**Figure 4** - Successful login

From the Test tab you can send test SMS message. Provide the recipient address, the message and click on the Send button (Figure 5).



Text message

Please create an SMS message by providing the recipient and the message text.

Recipient

To: +36794125486

SMS

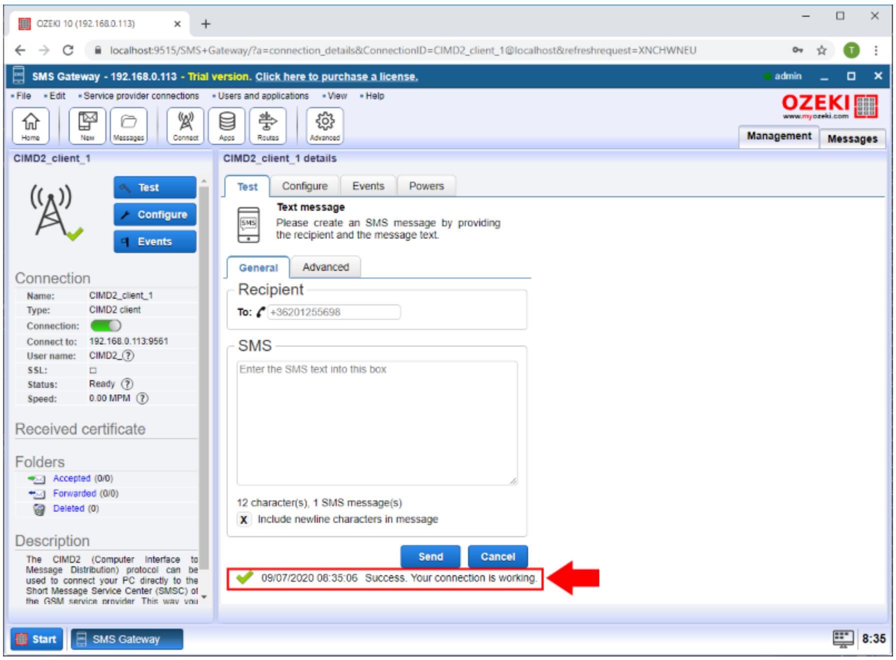
Hello World!

12 character(s), 1 SMS message(s)

☒ Include newline characters in message

**Figure 5** - Sending a test message

Finally you can see if the message is sent successfully. (Figure 6)



Name: CIMD2_client_1
Type: CIMD2 client
Connection:
Connect to: 192.168.0.113:9561
User name: CIMD2_
SSL: □
Status: Ready
Speed: 0.00 MPM

The CIMD2 (Computer Interface to Message Distribution) protocol can be used to connect your PC directly to the Short Message Service Center (SMSC) of the GSM service provider. This way you

Please create an SMS message by providing the recipient and the message text.

To: +36201255698

Enter the SMS text into this box

12 character(s), 1 SMS message(s)

☒ Include newline characters in message

09/07/2020 08:35:06 Success. Your connection is working.

**Figure 6 -** Message successfully sent

# Example CIMD2 PDU

An example CIMD PDUs look like the following:

```
03:007021:06201234657033:hello world
53:007021:06201234657060:971107130808
```

Each packet starts with STX (hex 02) and ends with ETX (hex 03). The content of the packet consists of fields separated by TAB (hex 09). Each field, in turn, consists of a parameter type, a colon (:), and the parameter value. Note that the last field must also be terminated with a TAB before the ETX.

In the CIMD2 PDUs two-digit parameter types are operation codes and each message must have exactly one. The number after the operation code is the sequence number used to match an operation to its response. The response code often referred to as acknowledgement of the message is equal to the operation code plus 50.

In the example above, the operation code 03 means submit message. Field 021 defines the destination address (telephone number), with field 033 is the user data (content) of the message. Response code 53 with a field 060 time stamp indicates that the message was accepted; if the message failed, the SMSC would reply with field 900 (error code) instead.

# CIMD2 protocol specification

CMID2-protocol-specification.pdf

# HTTP SMS client connection

***This page is about to demonstrate how to set up an HTTP SMS Client connection in Ozeki 10 SMS Gateway. The guide shows all steps that you have to follow to create the connection, how to configure it, and most importantly, how you can send SMS messages using this connection. The guide itself does not require any specific knowledge, you can easily complete it by just following the steps and it does not take more than ten minutes to successfully complete the guide. So, let's begin right now!***

## What is HTTP?

HTTP is a protocol, the foundation of any data exchange on the global network. It uses client-server protocol which the client and the server communicate with each other by exchanging individual messages.

## What is an SMS?

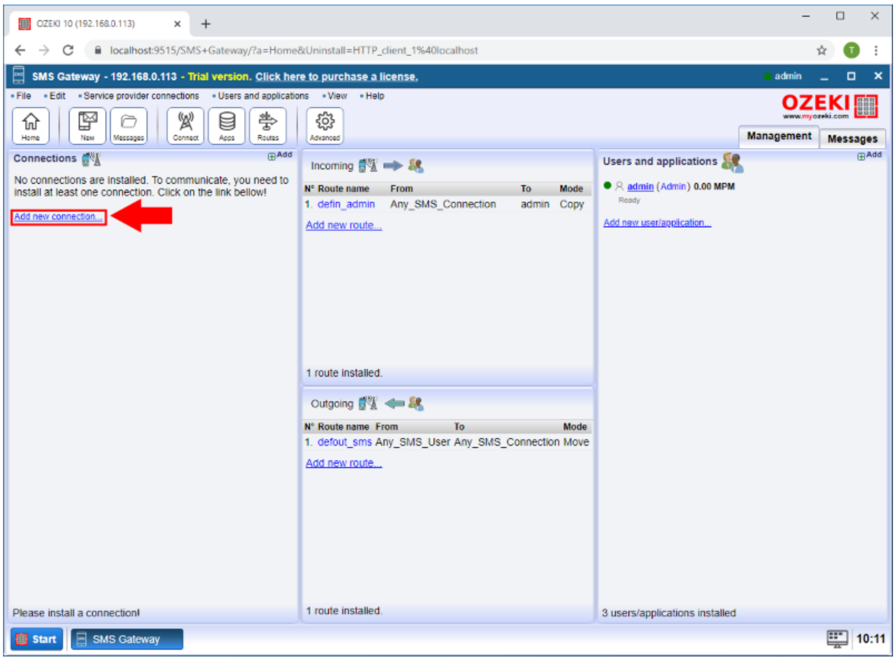
SMS is stands for 'Short Message Service'. It is used to send text messages to mobile phones. It was originally created for phones that use GSM communication, but now all the major cell phone systems support it.

## Connection steps

1. Type 'https://localhost:9515' into your browser
2. Select the SMS Gateway application
3. Click on 'Add new connection...'
4. Select the HTTP client connection
5. Configure the HTTP client connection
6. Enable SMS forwarding for the connection
7. Send test message
8. Check the event to see if the connection is working

## Step 1 - Create a new connection

The first step that you have to fulfill is to connect to the HTTP service provider. For that, you need to create the HTTP client connection. So, open the SMS Gateway in your browser by typing 'https://localhost:9515' in your browser and hit Enter. Then, open the SMS Gateway application from the Ozeki desktop. By doing this, the main menu of the SMS Gateway shows up as you can see it in Figure 1. Here at this point, click on the 'Add new connection...' to create a new connection.



No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

Please install a connection!

Incoming

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defin_admin | Any_SMS_Connection | admin | Copy |

Add new route...

1 route installed.

Outgoing

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move |

Add new route...

1 route installed.

Users and applications

● admin (Admin) 0.00 MPM
Ready

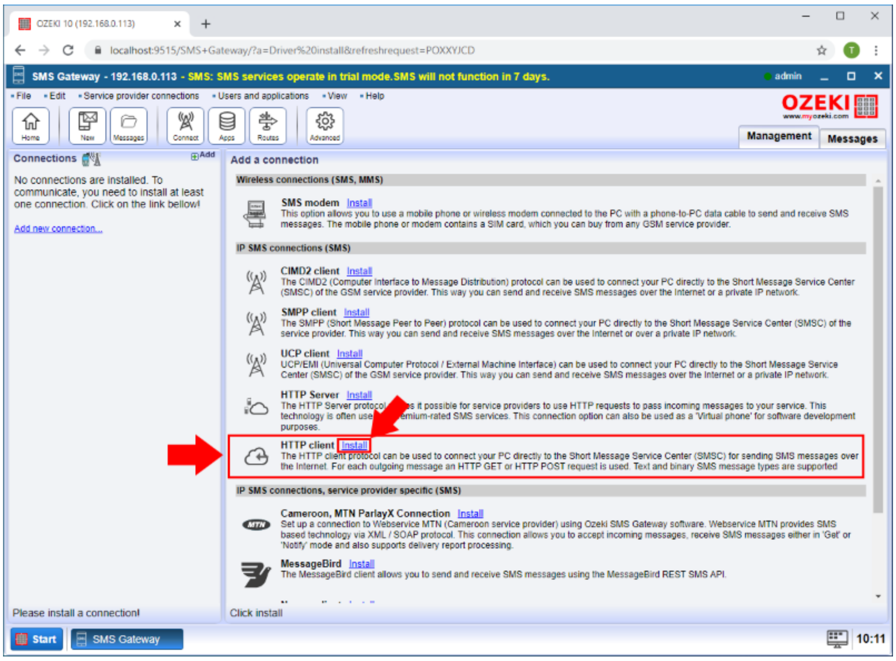
Add new user/application...

3 users/applications installed

Figure 1 - Creating a new connection

# Step 2 - Select HTTP client connection

The next menu contains every connection that can be created in the SMS Gateway. Each of these connections has got a short description to inform you which case you need to use them. In this list, you need to find the HTTP client connection, and as Figure 2 shows that, click on its 'Install' button to create it.



localhost:9515/SMS+Gateway/?a=Driver%20install&refreshrequest=POXXYJCD

SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.

No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

Add a connection

Wireless connections (SMS, MMS)

SMS modem Install
This option allows you to use a mobile phone or wireless modem connected to the PC with a phone-to-PC data cable to send and receive SMS messages. The mobile phone or modem contains a SIM card, which you can buy from any GSM service provider.

IP SMS connections (SMS)

CIMD2 client Install
The CIMD2 (Computer Interface to Message Distribution) protocol can be used to connect your PC directly to the Short Message Service Center (SMSC) of the GSM service provider. This way you can send and receive SMS messages over the Internet or a private IP network.

SMPP client Install
The SMPP (Short Message Peer to Peer) protocol can be used to connect your PC directly to the Short Message Service Center (SMSC) of the service provider. This way you can send and receive SMS messages over the Internet or over a private IP network.

UCP client Install
UCP/EMI (Universal Computer Protocol / External Machine Interface) can be used to connect your PC directly to the Short Message Service Center (SMSC) of the GSM service provider. This way you can send and receive SMS messages over the Internet or a private IP network.

HTTP Server Install
The HTTP Server protocol [illegible]s it possible for service providers to use HTTP requests to pass incoming messages to your service. This technology is often use[illegible]emium-rated SMS services. This connection option can also be used as a 'Virtual phone' for software development purposes.

HTTP client Install
The HTTP client protocol can be used to connect your PC directly to the Short Message Service Center (SMSC) for sending SMS messages over the Internet. For each outgoing message an HTTP GET or HTTP POST request is used. Text and binary SMS message types are supported

IP SMS connections, service provider specific (SMS)

Cameroon, MTN ParlayX Connection Install
Set up a connection to Webservice MTN (Cameroon service provider) using Ozeki SMS Gateway software. Webservice MTN provides SMS based technology via XML / SOAP protocol. This connection allows you to accept incoming messages, receive SMS messages either in 'Get' or 'Notify' mode and also supports delivery report processing.

MessageBird Install
The MessageBird client allows you to send and receive SMS messages using the MessageBird REST SMS API.

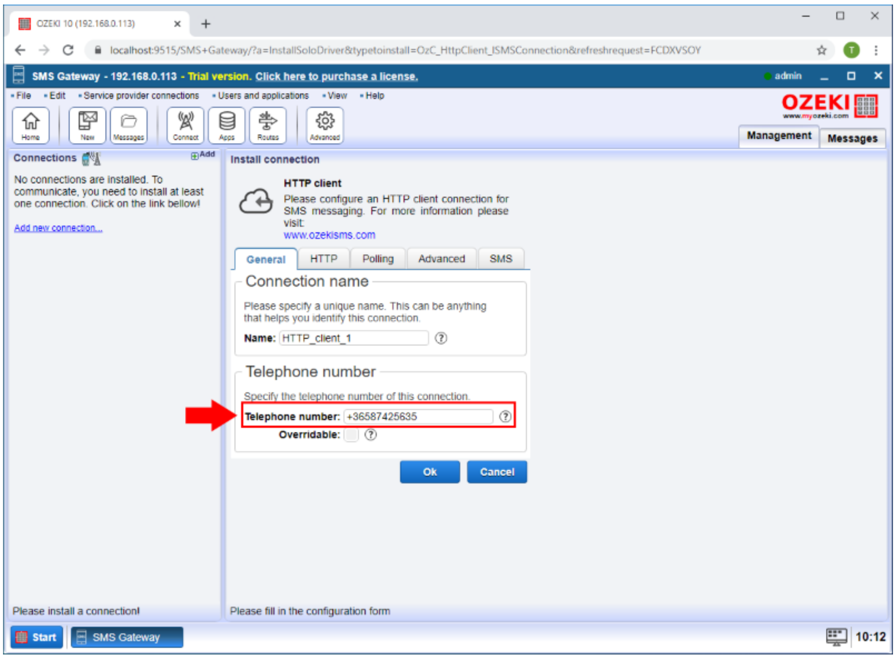
Please install a connection!

Click install

Figure 2 - Install the HTTP Client connection

# Step 3 - Configure the HTTP client connection

In the configuration menu, you need to specify some details for the connection. First of all, just give a name for the connection or you can leave the default name if you wish. Next, as Figure 3 demonstrates that, specify a phone number for the HTTP client connection. This phone number will be used to identify the connection if it sends or receives an SMS message.



localhost:9515/SMS+Gateway/?a=InstallSoloDriver&typetoinstall=OzC_HttpClient_ISMSConnection&refreshrequest=FCDXVSOY

SMS Gateway - 192.168.0.113 - Trial version. Click here to purchase a license.

No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

Install connection

HTTP client
Please configure an HTTP client connection for SMS messaging. For more information please visit:
www.ozekisms.com

Connection name

Please specify a unique name. This can be anything that helps you identify this connection.

Name: HTTP_client_1

Telephone number

Specify the telephone number of this connection.

Telephone number: +36587425635

Overridable: □

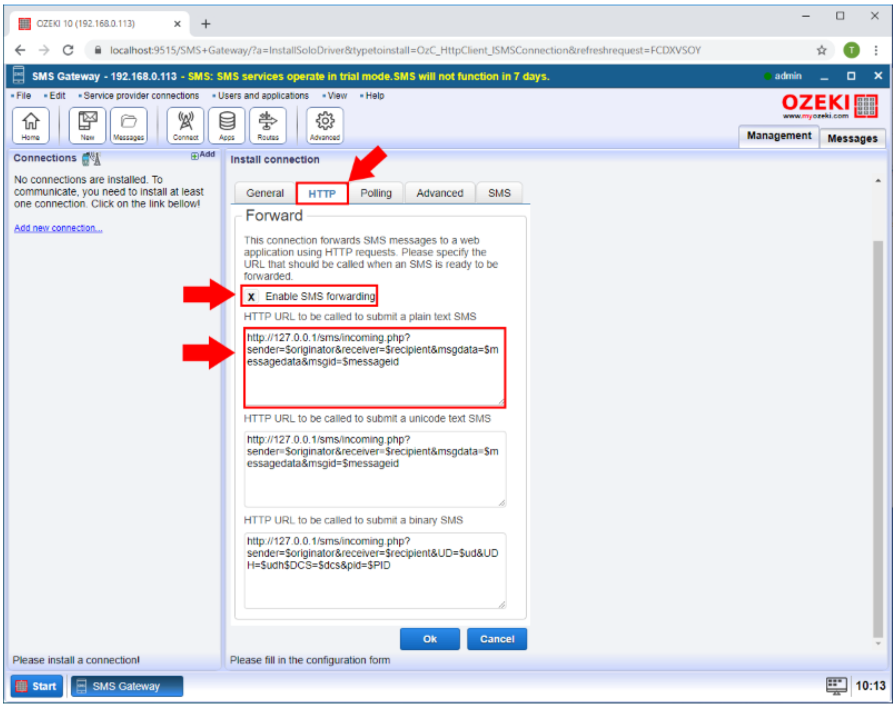
Please install a connection!

Please fill in the configuration form

Figure 3 - Providing HTTP Client phone number

## Step 4 - Enable SMS forwarding

At last configuration step, that you need to perform is to enable the SMS forwarding for that connection. To do this, select the HTTP tab in the configuration menu. Here, in the first step, check the 'Enable SMS forwarding' option as Figure 4 shows that. Then, you can configure the HTTP URL that will be called to submit the SMS messages. If you finished with all the configuration, just click OK to create the HTTP client connection.



localhost:9515/SMS+Gateway/?a=InstallSoloDriver&typetoinstall=OzC_HttpClient_ISMSConnection&refreshrequest=FCDXVSOY

SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.

Connections

No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

Install connection

General | HTTP | Polling | Advanced | SMS

Forward

This connection forwards SMS messages to a web application using HTTP requests. Please specify the URL that should be called when an SMS is ready to be forwarded.

☒ Enable SMS forwarding

HTTP URL to be called to submit a plain text SMS

```
http://127.0.0.1/sms/incoming.php?sender=$originator&receiver=$recipient&msgdata=$messagedata&msgid=$messageid
```

HTTP URL to be called to submit a unicode text SMS

```
http://127.0.0.1/sms/incoming.php?sender=$originator&receiver=$recipient&msgdata=$messagedata&msgid=$messageid
```

HTTP URL to be called to submit a binary SMS

```
http://127.0.0.1/sms/incoming.php?sender=$originator&receiver=$recipient&UD=$ud&UDH=$udh$DCS=$dcs&pid=$PID
```

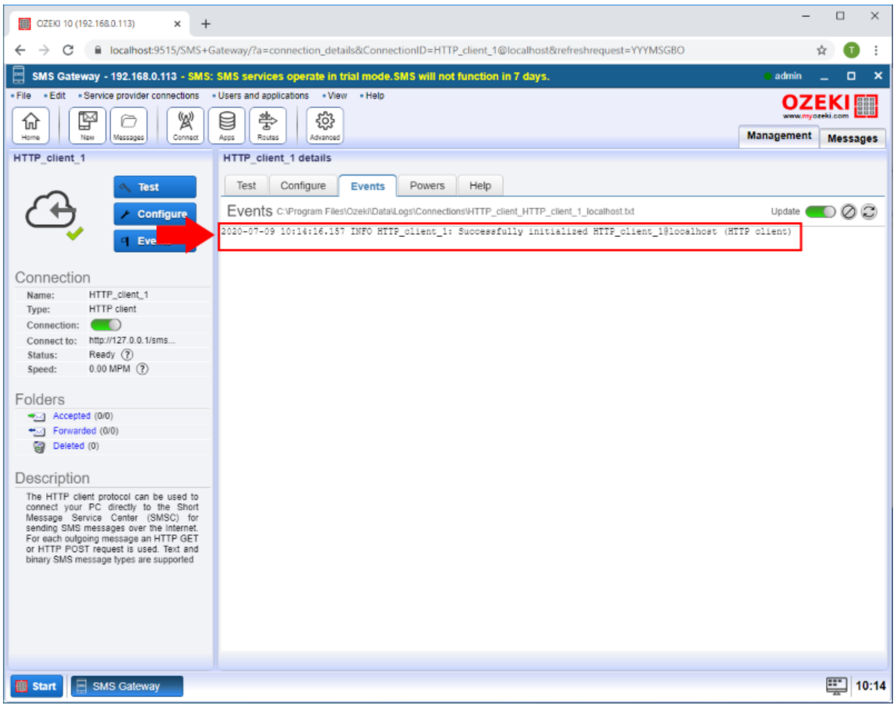
Please install a connection!

Please fill in the configuration form

Figure 4 - Specify the URL

## Step 5 - Check the Events menu of the connection

Now, your HTTP client connection is created and ready to use. To check that, you can go to the Event menu of the connection. In this menu, you will be able to see every event that occurred with the connection. Figure 5 shows the event that informs you about that connection is successfully initialized and ready to use.



localhost:9515/SMS+Gateway/?a=connection_details&ConnectionID=HTTP_client_1@localhost&refreshrequest=YYYMSGBO

SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.

HTTP_client_1 details

Events C:\Program Files\Ozeki\Data\Logs\Connections\HTTP_client_HTTP_client_1_localhost.txt

```
2020-07-09 10:14:16.157 INFO HTTP_client_1: Successfully initialized HTTP_client_1@localhost (HTTP client)
```

Connection

| | |
|---|---|
| Name: | HTTP_client_1 |
| Type: | HTTP client |
| Connection: | |
| Connect to: | http://127.0.0.1/sms... |
| Status: | Ready |
| Speed: | 0.00 MPM |

Folders

Accepted (0/0)
Forwarded (0/0)
Deleted (0)

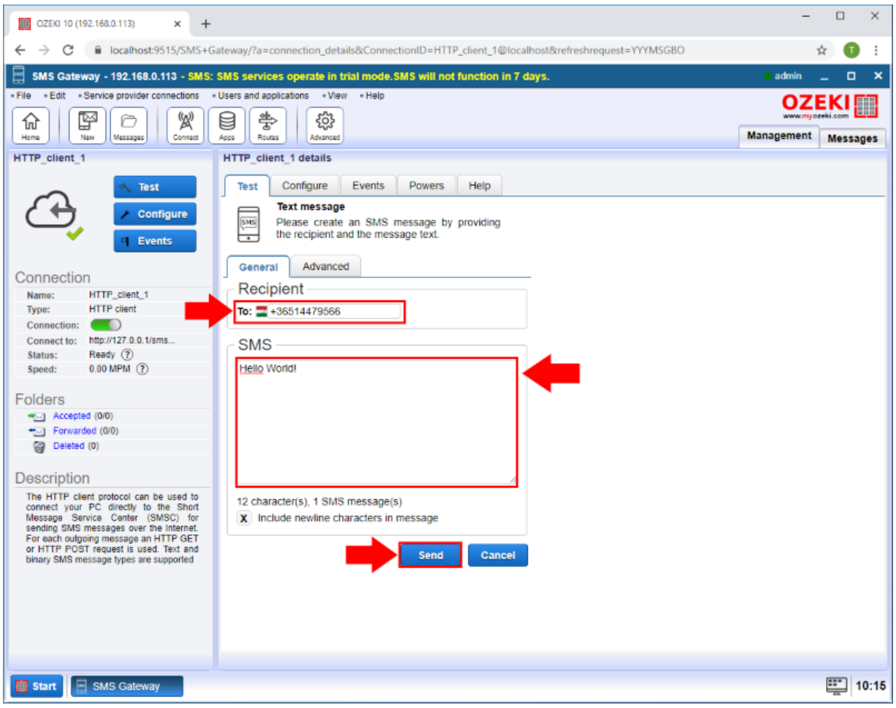
Description

The HTTP client protocol can be used to connect your PC directly to the Short Message Service Center (SMSC) for sending SMS messages over the Internet. For each outgoing message an HTTP GET or HTTP POST request is used. Text and binary SMS message types are supported

Figure 5 - Successful login

# Step 6 - Send a test message

The HTTP client connection is capable of sending and receiving SMS messages. To test the connection, you can simply send a message using the connection. For that, go to the menu of the connection, and select the 'Test' tab. Here you can compose a new message. So, as you can see it in Figure 6, set a recipient and the text of the message. If you have done it, just click on 'Send' to send the message.



OZEKI 10 (192.168.0.113)

localhost:9515/SMS+Gateway/?a=connection_details&ConnectionID=HTTP_client_1@localhost&refreshrequest=YYYMSGBO

SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.

HTTP_client_1

Connection

Name: HTTP_client_1
Type: HTTP client
Connection:
Connect to: http://127.0.0.1/sms...
Status: Ready
Speed: 0.00 MPM

Folders

Accepted (0/0)
Forwarded (0/0)
Deleted (0)

Description

The HTTP client protocol can be used to connect your PC directly to the Short Message Service Center (SMSC) for sending SMS messages over the Internet. For each outgoing message an HTTP GET or HTTP POST request is used. Text and binary SMS message types are supported

HTTP_client_1 details

Text message
Please create an SMS message by providing the recipient and the message text.

Recipient
To: +36514479566

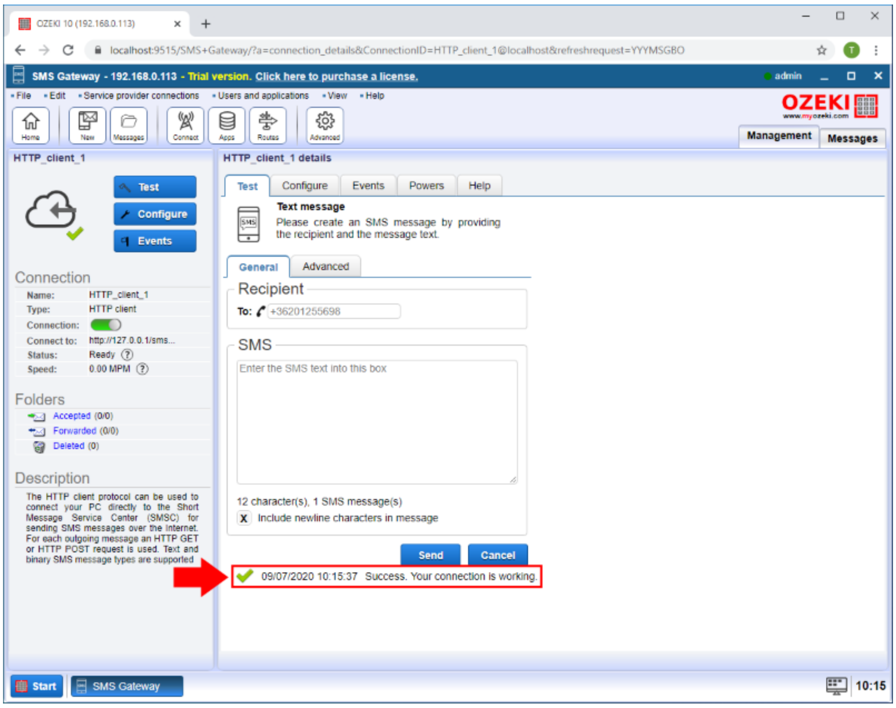
SMS
Hello World!

12 character(s), 1 SMS message(s)
☒ Include newline characters in message

Figure 6 - Sending a test message

# Step 7 - Check if the connection is working

As soon as you sent the message, the connection is going to give you a feedback about the delivery of the message was successful or not. Figure 7 demonstrates it below, that if the connection is working and able to send messages, it displays it with an event message.



OZEKI 10 (192.168.0.113)

localhost:9515/SMS+Gateway/?a=connection_details&ConnectionID=HTTP_client_1@localhost&refreshrequest=YYYMSGBO

SMS Gateway - 192.168.0.113 - Trial version. Click here to purchase a license.

HTTP_client_1

Connection

Name: HTTP_client_1
Type: HTTP client
Connection:
Connect to: http://127.0.0.1/sms...
Status: Ready
Speed: 0.00 MPM

Folders

Accepted (0/0)
Forwarded (0/0)
Deleted (0)

Description

The HTTP client protocol can be used to connect your PC directly to the Short Message Service Center (SMSC) for sending SMS messages over the Internet. For each outgoing message an HTTP GET or HTTP POST request is used. Text and binary SMS message types are supported

HTTP_client_1 details

Text message
Please create an SMS message by providing the recipient and the message text.

Recipient
To: +36201255698

SMS
Enter the SMS text into this box

12 character(s), 1 SMS message(s)
☒ Include newline characters in message

✓ 09/07/2020 10:15:37 Success. Your connection is working.

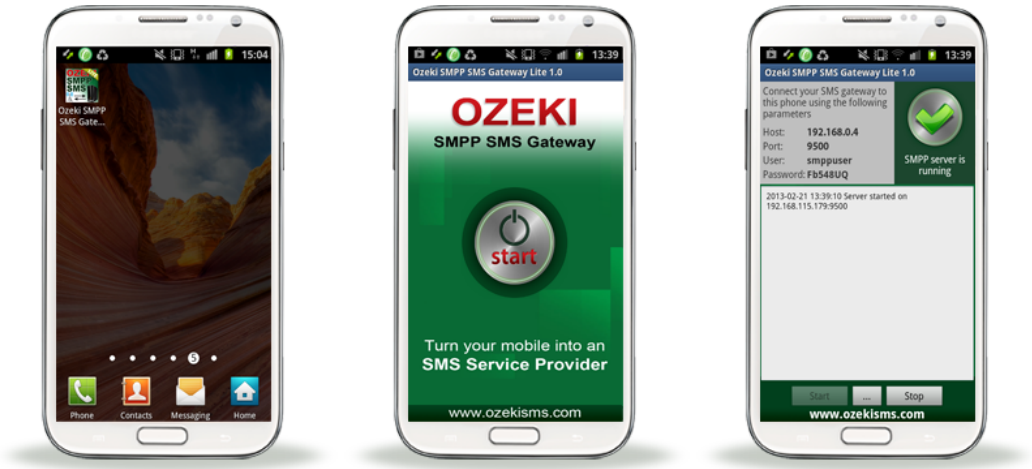
Figure 7 - Message successfully sent

# Android SMS client connection

## Step 1. - Install the Android SMPP App on your smartphone

Download, install and configure the application form Ozeki website: Download Android SMPP gateway

Start the application by clicking on its icon then "Start" button. You will see the followings on the top left-hand side of the screen: Host address, Port number, User name, Password (Figure 1). You need these parameters later.



Connect your SMS gateway to this phone using the following parameters

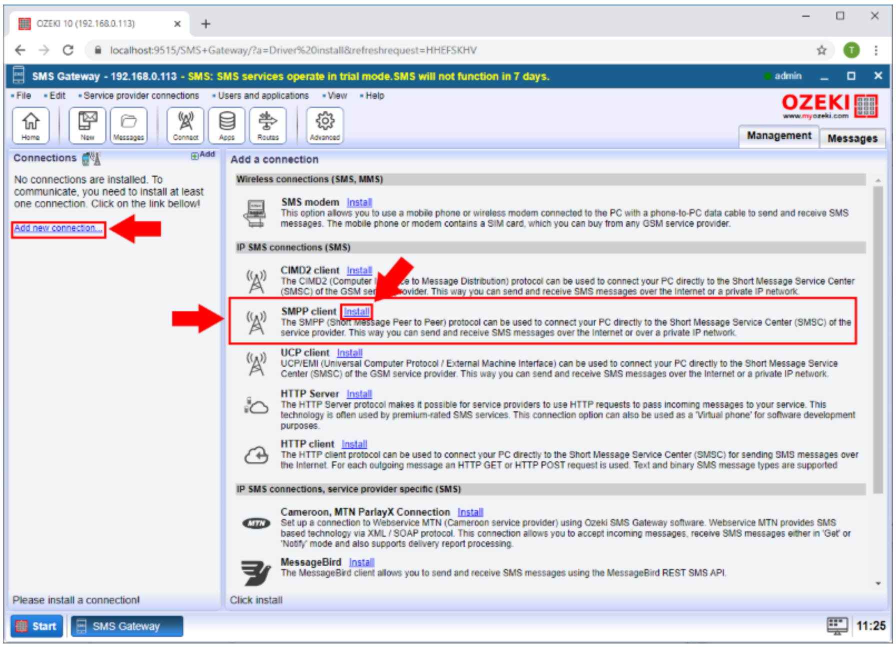
Host: 192.168.0.4
Port: 9500
User: smppuser
Password: Fb548UQ

2013-02-21 13:39:10 Server started on 192.168.115.179:9500

Figure 1 - Ozeki Android SMPP SMS Gateway has been installed and launched succesfully

## Step 2. - Setup SMS Gateway to use your Android phone

Now you need to install and configure SMPP (Short Message Peer-to-Peer) connection. Firstly, add new service provider connection by selecting and clicking on "Add new connection" textlink. After this the "Add connection" will appear on the right side of the GUI. Look for "SMPP Connection", then click on "Install" to select the SMPP protocol (Figure 2).


OZEKI 10 (192.168.0.113)
localhost:9515/SMS+Gateway/?a=Driver%20install&refreshrequest=HHEFSKHV
SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
OZEKI
www.myozeki.com
Management
Messages
Connections
Add
Start
SMS Gateway
11:25

No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

Add a connection

Wireless connections (SMS, MMS)

SMS modem Install
This option allows you to use a mobile phone or wireless modem connected to the PC with a phone-to-PC data cable to send and receive SMS messages. The mobile phone or modem contains a SIM card, which you can buy from any GSM service provider.

IP SMS connections (SMS)

CIMD2 client Install
The CIMD2 (Computer I ce to Message Distribution) protocol can be used to connect your PC directly to the Short Message Service Center (SMSC) of the GSM ser ovider. This way you can send and receive SMS messages over the Internet or a private IP network.

SMPP client Install
The SMPP (Short Message Peer to Peer) protocol can be used to connect your PC directly to the Short Message Service Center (SMSC) of the service provider. This way you can send and receive SMS messages over the Internet or over a private IP network.

UCP client Install
UCP/EMI (Universal Computer Protocol / External Machine Interface) can be used to connect your PC directly to the Short Message Service Center (SMSC) of the GSM service provider. This way you can send and receive SMS messages over the Internet or a private IP network.

HTTP Server Install
The HTTP Server protocol makes it possible for service providers to use HTTP requests to pass incoming messages to your service. This technology is often used by premium-rated SMS services. This connection option can also be used as a 'Virtual phone' for software development purposes.

HTTP client Install
The HTTP client protocol can be used to connect your PC directly to the Short Message Service Center (SMSC) for sending SMS messages over the Internet. For each outgoing message an HTTP GET or HTTP POST request is used. Text and binary SMS message types are supported

IP SMS connections, service provider specific (SMS)

Cameroon, MTN ParlayX Connection Install
Set up a connection to Webservice MTN (Cameroon service provider) using Ozeki SMS Gateway software. Webservice MTN provides SMS based technology via XML / SOAP protocol. This connection allows you to accept incoming messages, receive SMS messages either in 'Get' or 'Notify' mode and also supports delivery report processing.

MessageBird Install
The MessageBird client allows you to send and receive SMS messages using the MessageBird REST SMS API.

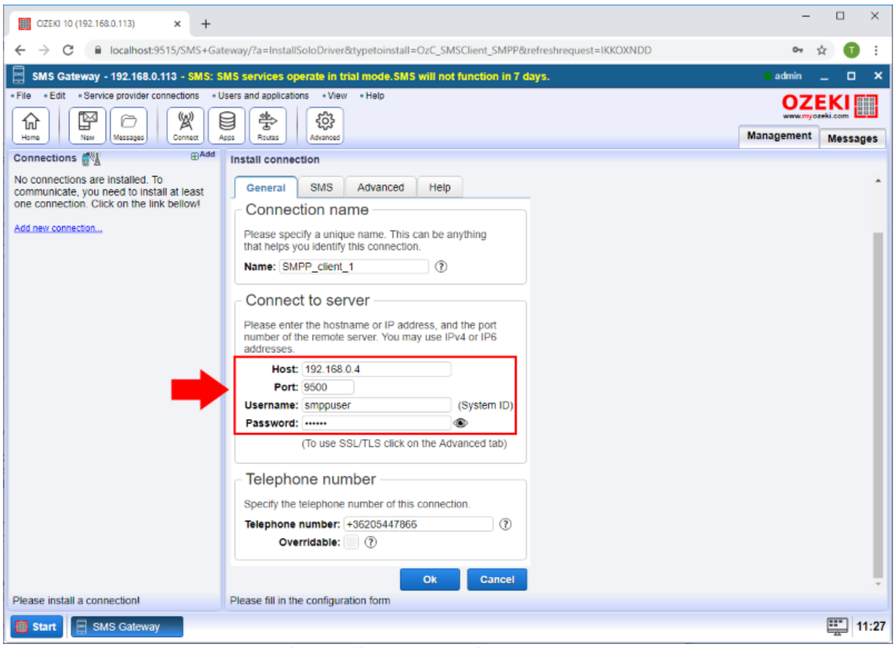
Please install a connection!

Click install

Figure 2 - Adding a new SMPP connection

Now provide your SMPP credentials on the Install connection page. You need to enter the following parameters: Host, Port number, Username, Password. You can find this information in your Ozeki Android SMPP SMS

Gateway application (see Figure 3).



localhost:9515/SMS+Gateway/?a=InstallSoloDriver&typetoinstall=OzC_SMSClient_SMPP&refreshrequest=IKKOXNDD

SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.

Connections

No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

Install connection

Connection name

Please specify a unique name. This can be anything that helps you identify this connection.

Name: SMPP_client_1

Connect to server

Please enter the hostname or IP address, and the port number of the remote server. You may use IPv4 or IP6 addresses.

Host: 192.168.0.4
Port: 9500
Username: smppuser (System ID)
Password: ••••••

(To use SSL/TLS click on the Advanced tab)

Telephone number

Specify the telephone number of this connection.

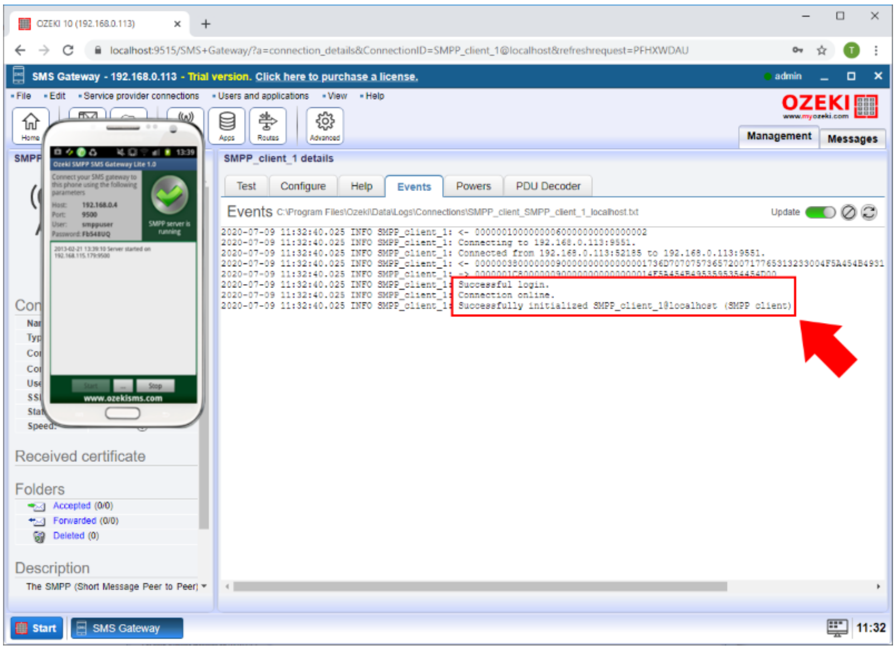
Telephone number: +36205447866
Overridable: □

Please install a connection!

Please fill in the configuration form

Figure 3 - Specifying the SMPP server settings

In the Events tab the "Successful login" and "Connection online" messages indicate that the SMPP connection has been installed successfully (Figure 4).



localhost:9515/SMS+Gateway/?a=connection_details&ConnectionID=SMPP_client_1@localhost&refreshrequest=PFHXWDAU

SMS Gateway - 192.168.0.113 - Trial version. Click here to purchase a license.

SMPP_client_1 details

Events C:\Program Files\Ozeki\Data\Logs\Connections\SMPP_client_SMPP_client_1_localhost.txt

```
2020-07-09 11:32:40.025 INFO SMPP_client_1: <- 00000010000000060000000000000002
2020-07-09 11:32:40.025 INFO SMPP_client_1: Connecting to 192.168.0.113:9551.
2020-07-09 11:32:40.025 INFO SMPP_client_1: Connected from 192.168.0.113:52185 to 192.168.0.113:9551.
2020-07-09 11:32:40.025 INFO SMPP_client_1: <- 0000003800000009000000000000000173 6D707075736572007177653132330 04F5A454B4931
2020-07-09 11:32:40.025 INFO SMPP_client_1: -> 0000001C800000090000000000000000014F5A454B4953595354454D00
2020-07-09 11:32:40.025 INFO SMPP_client_1: Successful login.
2020-07-09 11:32:40.025 INFO SMPP_client_1: Connection online.
2020-07-09 11:32:40.025 INFO SMPP_client_1: Successfully initialized SMPP_client_1@localhost (SMPP client)
```

Received certificate

Folders

Accepted (0/0)
Forwarded (0/0)
Deleted (0)

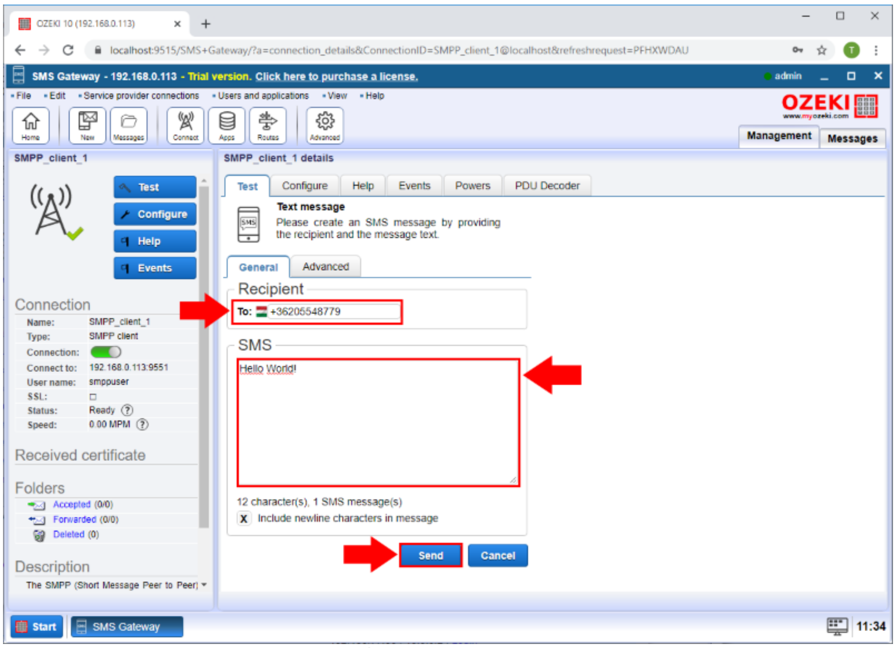
Description

The SMPP (Short Message Peer to Peer)

Figure 4 - SMPP connection has been installed successfully

# Step 3. - Send your first SMS

From the Test tab you can send test SMS message. Provide the recipient address, the message and click on the Send button (Figure 4).



OZEKI 10 (192.168.0.113)

localhost:9515/SMS+Gateway/?a=connection_details&ConnectionID=SMPP_client_1@localhost&refreshrequest=PFHXWDAU

SMS Gateway - 192.168.0.113 - Trial version. Click here to purchase a license.

SMPP_client_1

Connection

Name: SMPP_client_1
Type: SMPP client
Connection:
Connect to: 192.168.0.113:9551
User name: smppuser
SSL: □
Status: Ready
Speed: 0.00 MPM

Received certificate

Folders

Accepted (0/0)
Forwarded (0/0)
Deleted (0)

Description

The SMPP (Short Message Peer to Peer)

SMPP_client_1 details

Text message

Please create an SMS message by providing the recipient and the message text.

Recipient

To: +36205548779

SMS

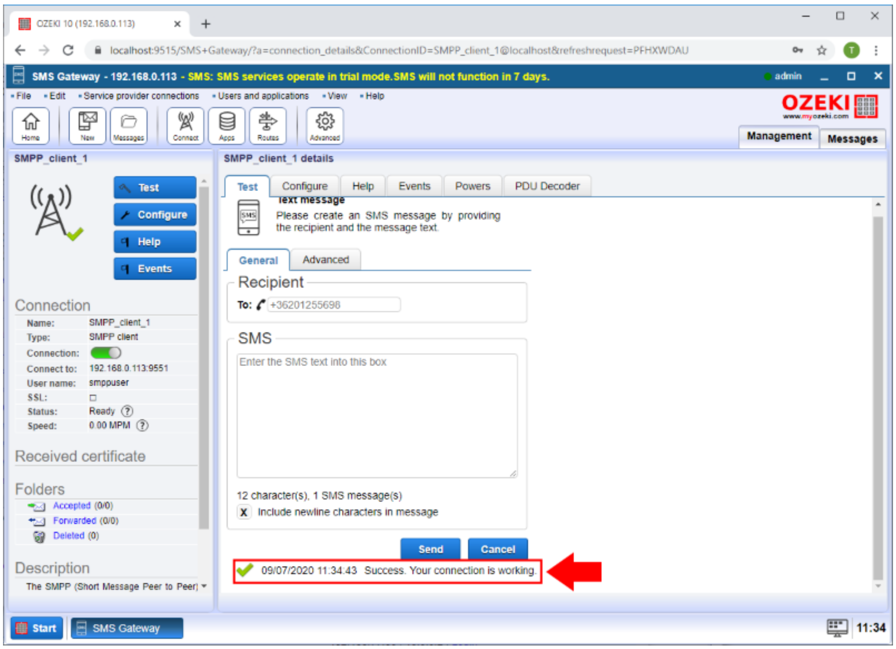
Hello World!

12 character(s), 1 SMS message(s)

X Include newline characters in message

Figure 4 - Sending a test message

Finally you can see if the message is sent successfully (Figure 5).



OZEKI 10 (192.168.0.113)

localhost:9515/SMS+Gateway/?a=connection_details&ConnectionID=SMPP_client_1@localhost&refreshrequest=PFHXWDAU

SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.

SMPP_client_1

Connection

Name: SMPP_client_1
Type: SMPP client
Connection:
Connect to: 192.168.0.113:9551
User name: smppuser
SSL: □
Status: Ready
Speed: 0.00 MPM

Received certificate

Folders

Accepted (0/0)
Forwarded (0/0)
Deleted (0)

Description

The SMPP (Short Message Peer to Peer)

SMPP_client_1 details

Text message

Please create an SMS message by providing the recipient and the message text.

Recipient

To: +36201255698

SMS

Enter the SMS text into this box

12 character(s), 1 SMS message(s)

X Include newline characters in message

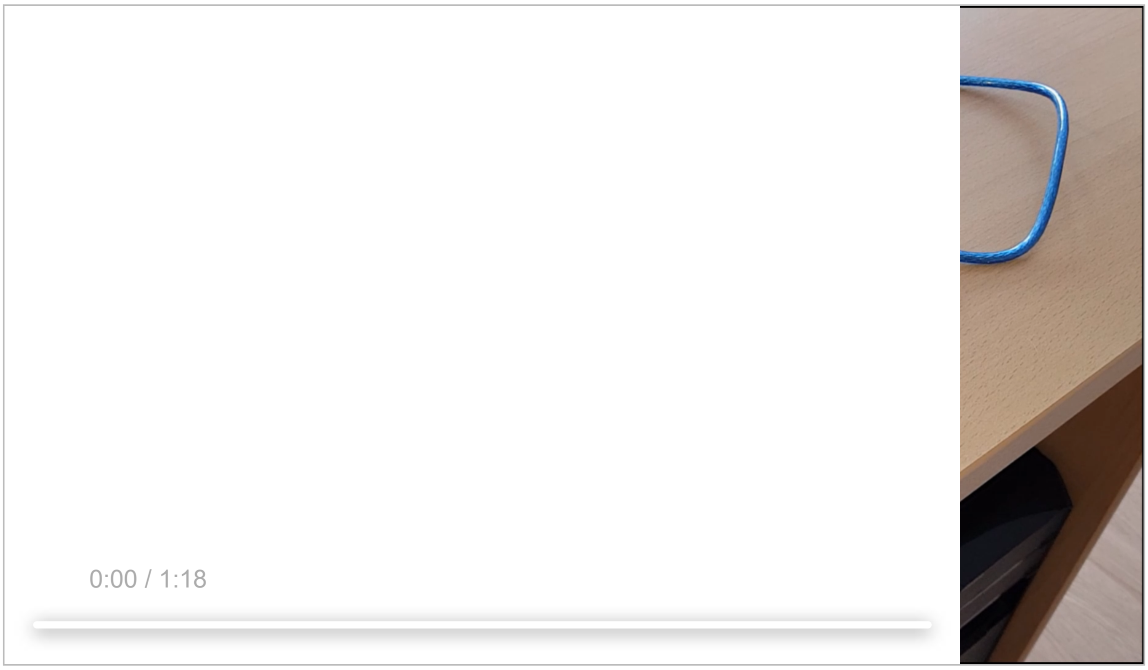
✓ 09/07/2020 11:34:43 Success. Your connection is working.

Figure 5 - Message successfully sent

# Multitech SMS modem

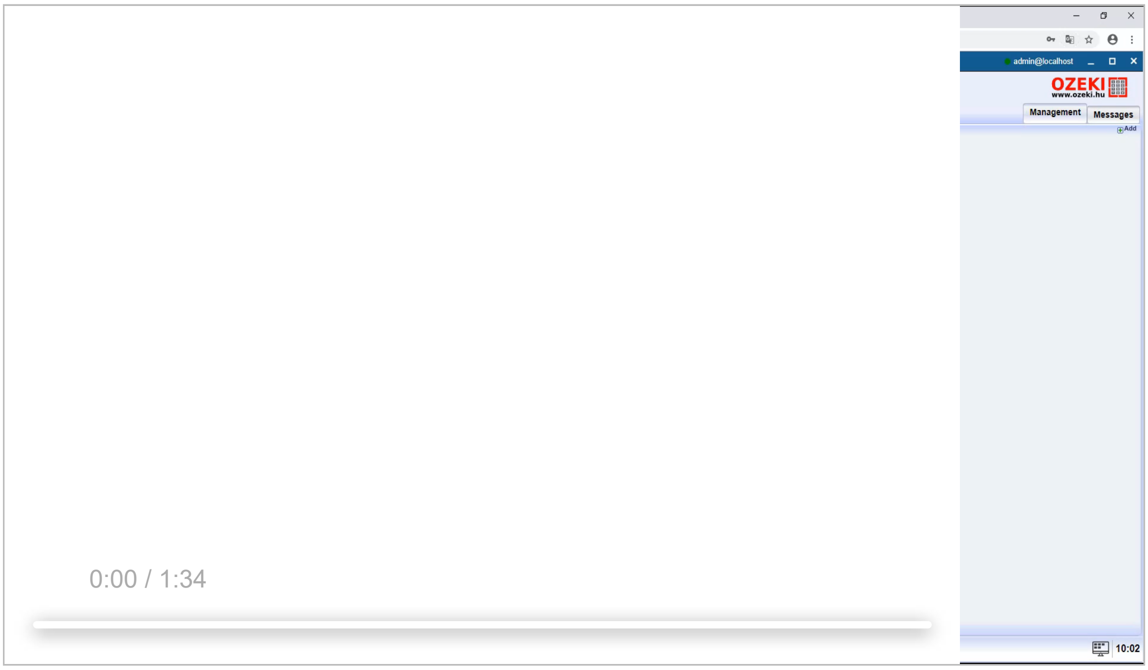
The MultiTech MultiModem Quad band EDGE modem w/USB MTCBA-E-U is an excellent choice for SMS messaging. It works well with Ozeki NG 10 and Ozeki 10. This is a high speed, professional wireless cellular modem. It operates at 230400 baud, 8N1, which is one of the highest speeds we have found in USB modems. It is very reliable. It does not require an external power supply, it powers itself over the USB port.

# Multitech SMS modem setup



Start the Video

# Ozeki NG 10 configuration



Start the Video

# Drivers for Windows 10

| Data Sheet | Pages | Size (kB) |
|---|---|---|
| Multitech SMS Modem Driver for Windows 10 (zip) | 64bit | 1464 Kb |
| Multitech SMS Modem Driver for Windows Server 2008 (zip) | 64bit | 1464 Kb |

# MultiTech SMS modem documents

| Data Sheet | Pages | Size (kB) |
|---|---|---|
| MultiModem Quad band EDGE SMS modem w/USB User Gude (PDF) | 29 | 960 |
| MTCBA-E-U MultiModem Quad band EDGE SMS modem w/USB Data Sheet (PDF) | 4 | 922 |
| MTCBA SMS Modem User Manual (PDF) | 42 | 923 |

## Connect MultiTech MultiModem Quad band EDGE modem to your PC

As **Figure 1** shows that you need a SIM card, an antenna, a USB cable and the modem itself to set up your Multitech SMS modem to your PC.

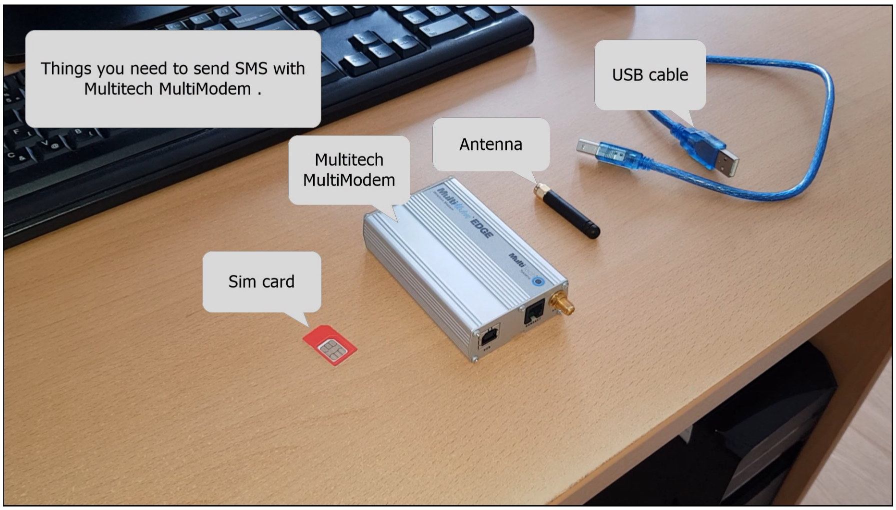


Figure 1 - The parts you need to connect the modem succesfully

The first step that you have to perform is to plug the SIM card into the modem as **Figure 2** demonstrates it.

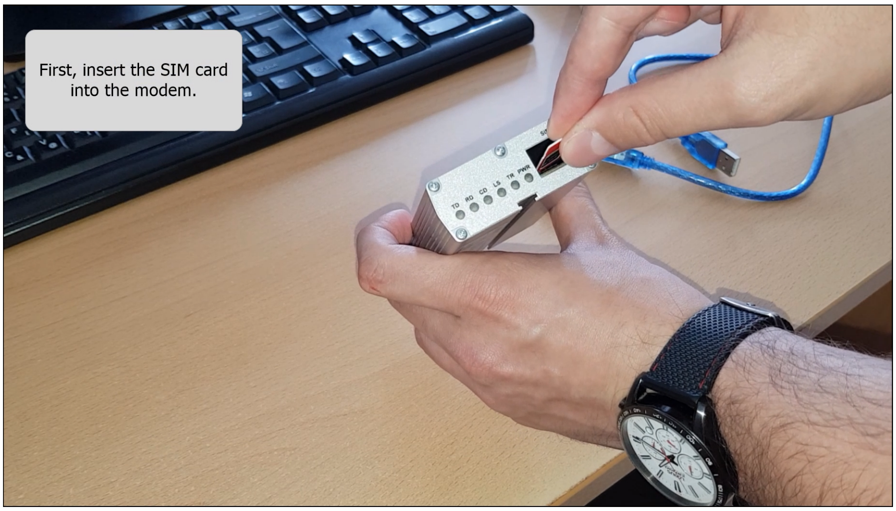


Figure 2 - The SIM card inserted into the modem

After that, you have to connect the modem to its antenna like in **Figure 3**.

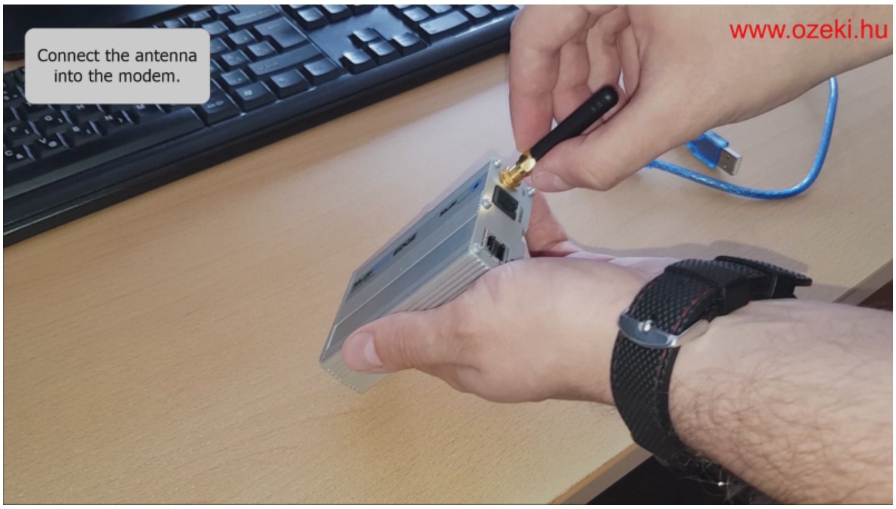


Figure 3 - The antenna connected to the modem

The next step as you can see on **Figure 4** is to insert the USB cable into the modem.

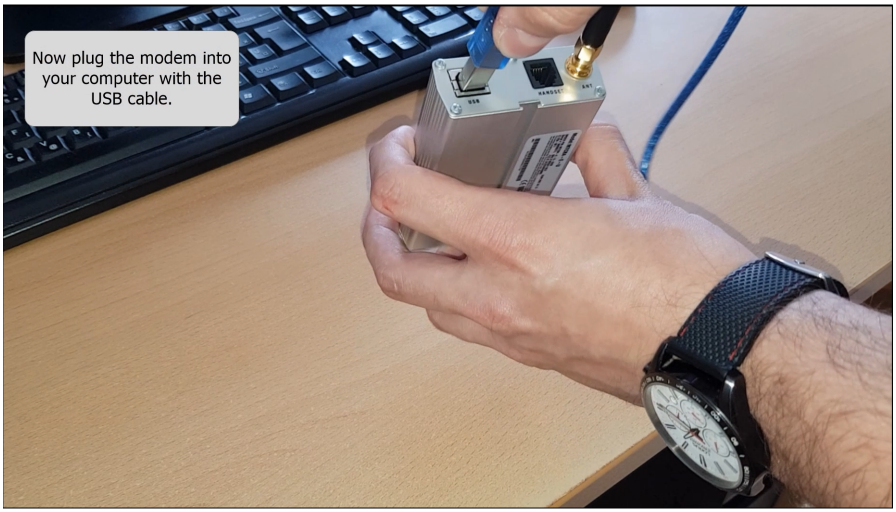


Figure 4 - The USB cable have to be inserted

Lastly, as you can see on **Figure 5**, the other end of the USB cable have to be connected to your PC.

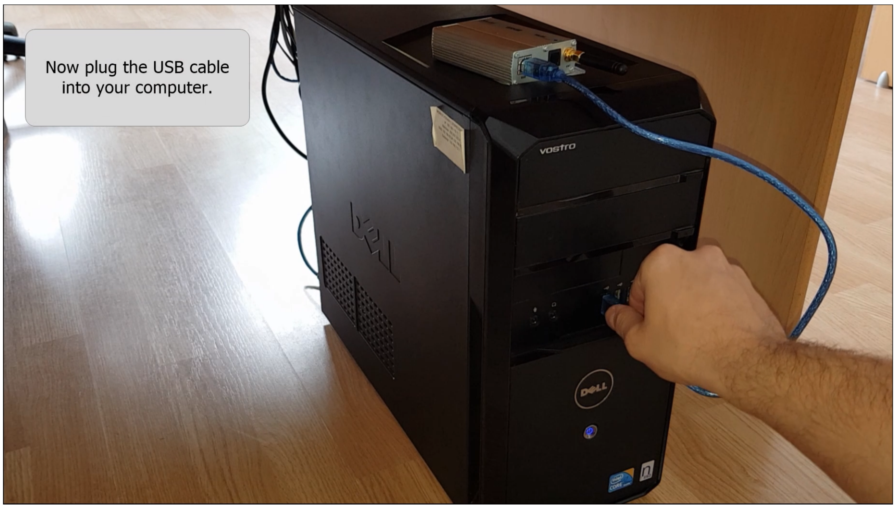


Figure 5 - The USB cable connected to the PC

If you have done everything right until this point the power led on the modem should be on as **Figure 6** shows.

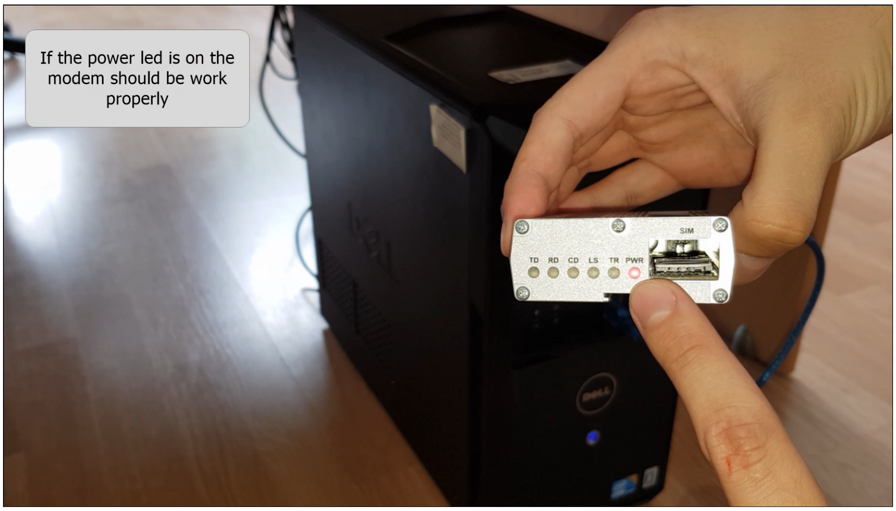


Figure 6 - The modem is turned on now

# Configure the modem in Ozeki NG 10

Now your modem is ready to be configured. The first step you have to fulfill is to open Ozeki NG 10 on your desktop (**Figure 7**).

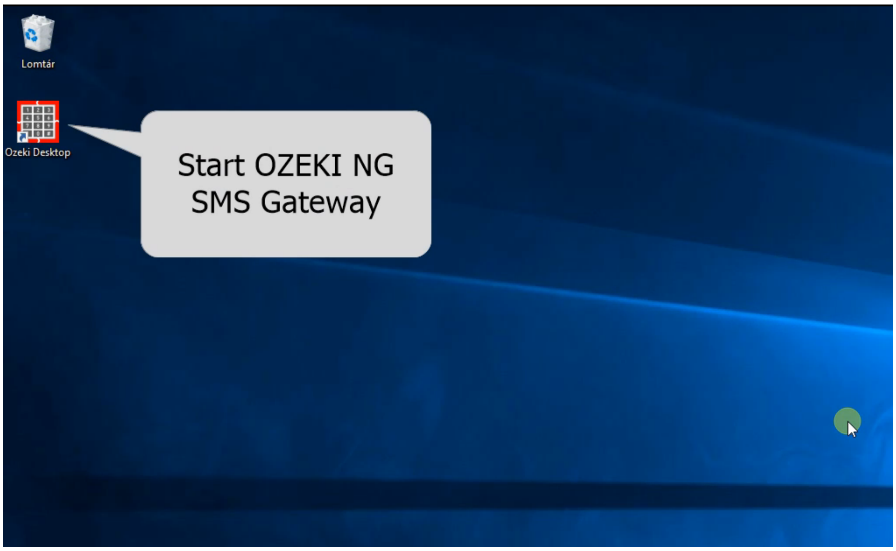


Figure 7 - Open Ozeki NG 10

Next, log in with your Ozeki NG 10 account by typing your username and password as you can see on **Figure 8** This account has to be created during the installation process.

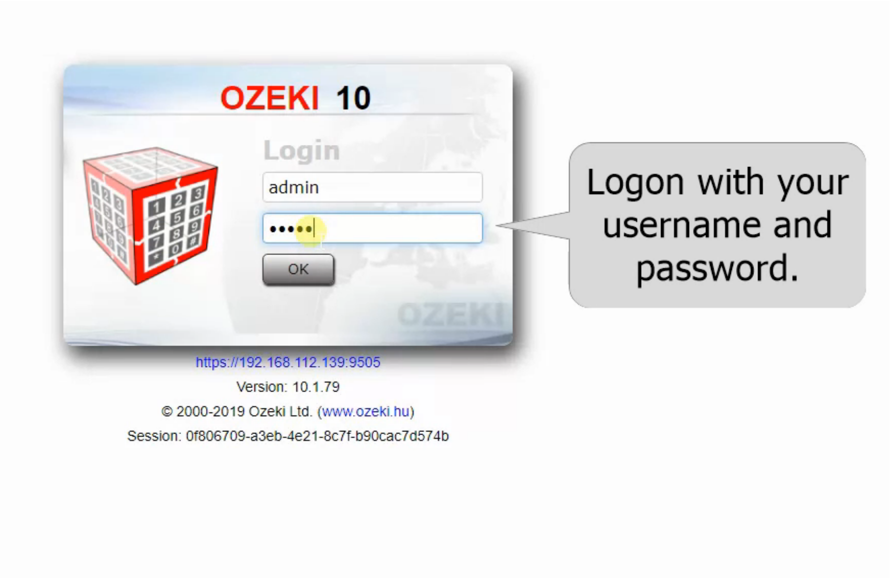

OZEKI 10
Login
admin
OK
Logon with your username and password.
https://192.168.112.139:9505
Version: 10.1.79
© 2000-2019 Ozeki Ltd. (www.ozeki.hu)
Session: 0f806709-a3eb-4e21-8c7f-b90cac7d574b

Figure 8 - Login with your Ozeki NG 10 account

Here, in the SMS Gateway application, the first thing that you need to do is to create a new connection. This can be done by clicking on **Add new connection** as you can see on **Figure 9**.

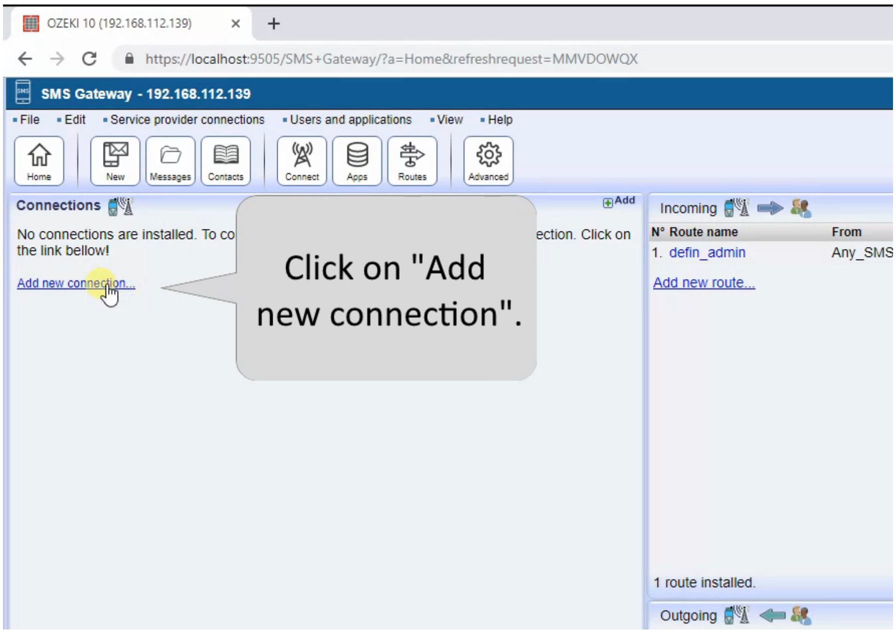
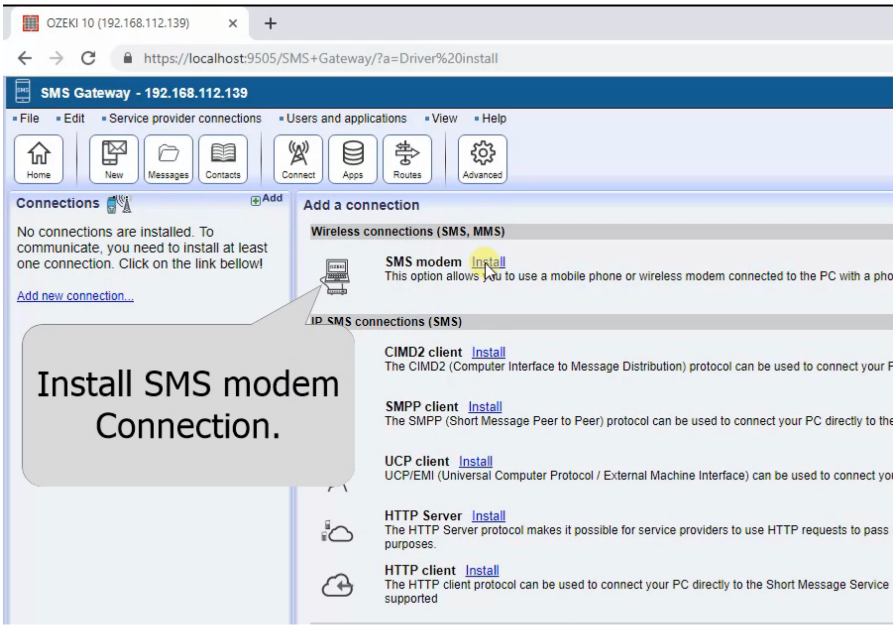


Figure 9 - Add new connection in the SMS Gateway

The next window lists you all the connections that you can create in the SMS Gateway application. Now, as **Figure 10** demonstrates, you have to select the **SMS modem** connection by clicking on **Install**.



OZEKI 10 (192.168.112.139)
https://localhost:9505/SMS+Gateway/?a=Driver%20install
SMS Gateway - 192.168.112.139
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Contacts
Connect
Apps
Routes
Advanced
Connections
Add
Add a connection
Wireless connections (SMS, MMS)
IP SMS connections (SMS)

No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

SMS modem Install
This option allows you to use a mobile phone or wireless modem connected to the PC with a pho

CIMD2 client Install
The CIMD2 (Computer Interface to Message Distribution) protocol can be used to connect your P

SMPP client Install
The SMPP (Short Message Peer to Peer) protocol can be used to connect your PC directly to the

UCP client Install
UCP/EMI (Universal Computer Protocol / External Machine Interface) can be used to connect yo

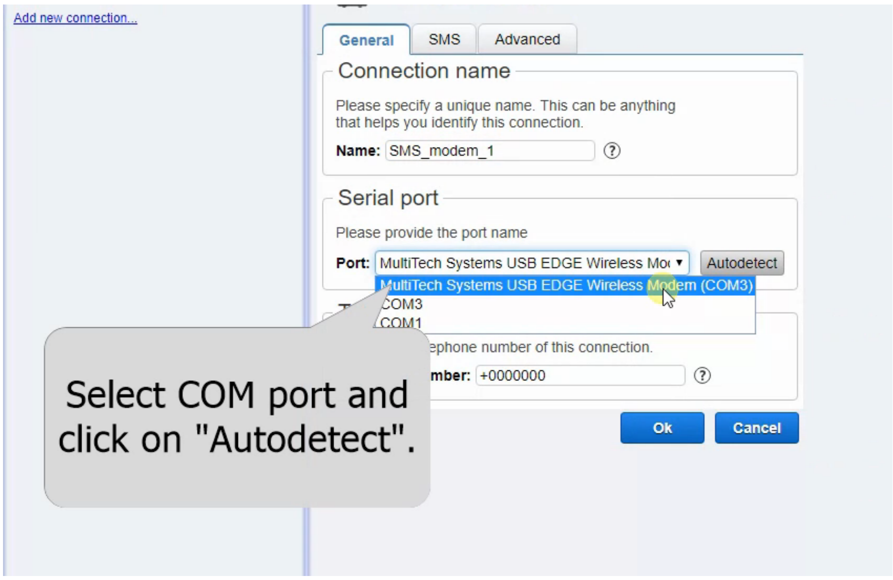
HTTP Server Install
The HTTP Server protocol makes it possible for service providers to use HTTP requests to pass purposes.

HTTP client Install
The HTTP client protocol can be used to connect your PC directly to the Short Message Service supported

Install SMS modem Connection.

Figure 10 - Select the SMS modem connection to install

In the installation menu, you need to select the port, where you connected the modem. For that, just click on the **Autodetect** button as you can see on **Figure 11**.



Connection name

Please specify a unique name. This can be anything that helps you identify this connection.

Name: SMS_modem_1

Serial port

Please provide the port name

Port: MultiTech Systems USB EDGE Wireless Mo

MultiTech Systems USB EDGE Wireless Modem (COM3)
COM3
COM1

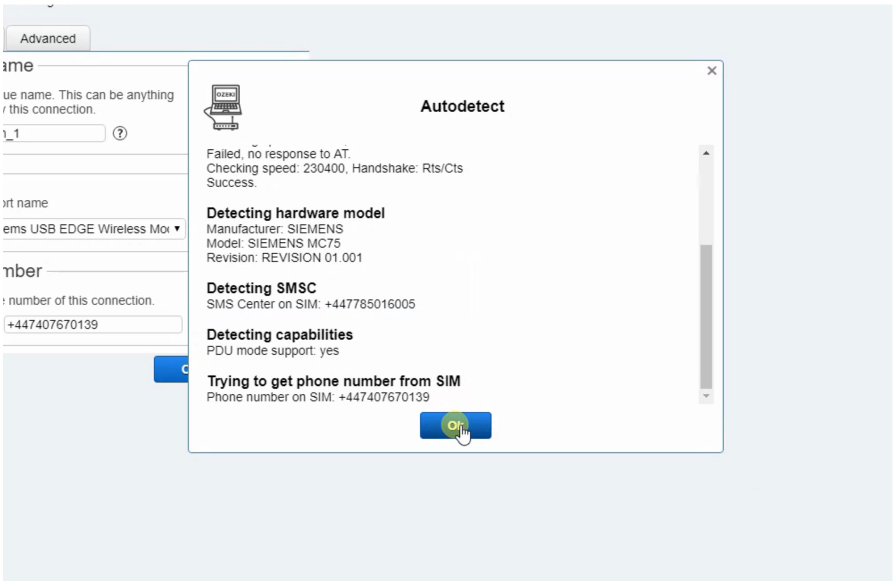
ephone number of this connection.

mber: +0000000

Select COM port and click on "Autodetect".

Figure 11 - Autodetect the port where you connected the modem

This Autodetect option will detect your modem, and if it is connected right the following window should show up like in **Figure 12**.



Failed, no response to AT.
Checking speed: 230400, Handshake: Rts/Cts
Success.

**Detecting hardware model**
Manufacturer: SIEMENS
Model: SIEMENS MC75
Revision: REVISION 01.001

**Detecting SMSC**
SMS Center on SIM: +447785016005

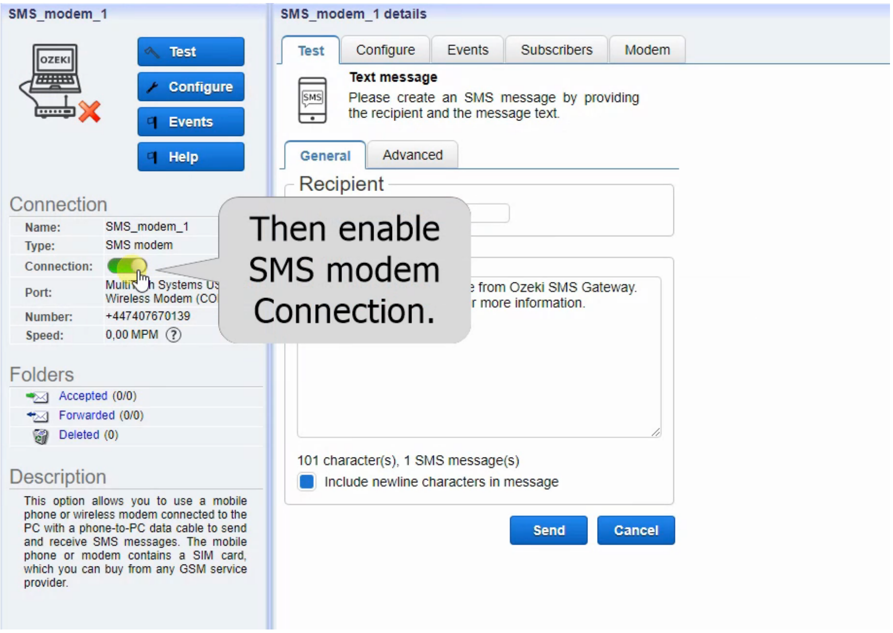
**Detecting capabilities**
PDU mode support: yes

**Trying to get phone number from SIM**
Phone number on SIM: +447407670139

Figure 12 - Details of the detected modem

# Send an SMS message using the modem connection

At this point, you have an SMS modem connection in the SMS Gateway application. To handle SMS messages with this connection, first, you need to enable the connection (**Figure 13**) by turning on the switch button in the connection details window.



**Text message**
Please create an SMS message by providing the recipient and the message text.

Connection
Name: SMS_modem_1
Type: SMS modem
Connection:
Port: Multi...h Systems US Wireless Modem (CO
Number: +447407670139
Speed: 0,00 MPM

Folders
Accepted (0/0)
Forwarded (0/0)
Deleted (0)

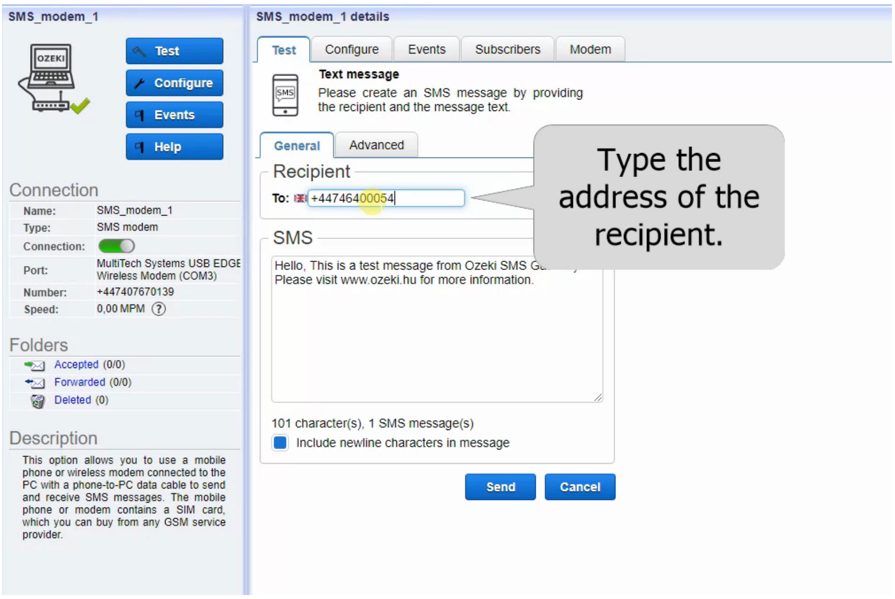
Description
This option allows you to use a mobile phone or wireless modem connected to the PC with a phone-to-PC data cable to send and receive SMS messages. The mobile phone or modem contains a SIM card, which you can buy from any GSM service provider.

...e from Ozeki SMS Gateway.
...r more information.

101 character(s), 1 SMS message(s)
☑ Include newline characters in message

Figure 13 - Turn on the SMS modem connection

To send the SMS message, you need to type the phone number of the recipient (**Figure 14**) in the **Recipient** field.



Connection

Name: SMS_modem_1
Type: SMS modem
Connection:
Port: MultiTech Systems USB EDGE Wireless Modem (COM3)
Number: +447407670139
Speed: 0,00 MPM

Folders

Accepted (0/0)
Forwarded (0/0)
Deleted (0)

Description

This option allows you to use a mobile phone or wireless modem connected to the PC with a phone-to-PC data cable to send and receive SMS messages. The mobile phone or modem contains a SIM card, which you can buy from any GSM service provider.

Text message

Please create an SMS message by providing the recipient and the message text.

Recipient

To: +44746400054

SMS

Hello, This is a test message from Ozeki SMS G... Please visit www.ozeki.hu for more information.

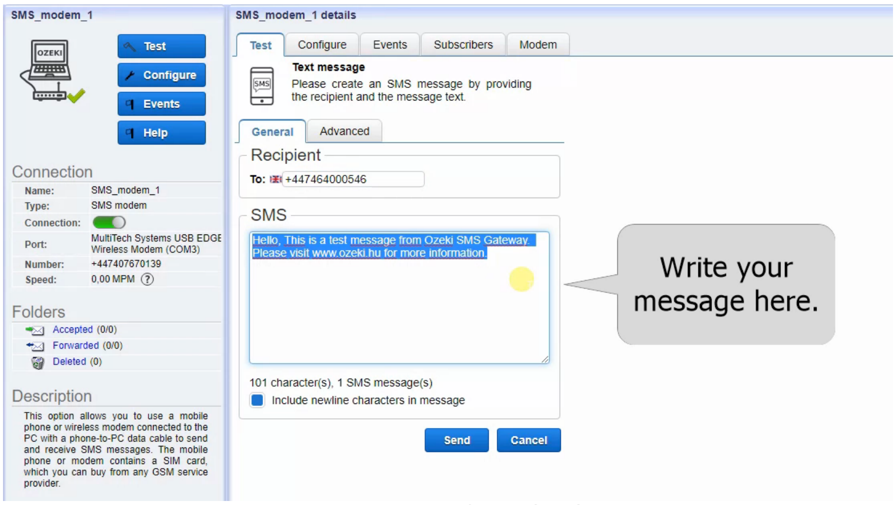
101 character(s), 1 SMS message(s)

■ Include newline characters in message

Type the address of the recipient.

Figure 14 - Type the address where you want to send the SMS message

Next, just type the text of the SMS message to the **SMS** field like in **Figure 15**.



Connection

Name: SMS_modem_1
Type: SMS modem
Connection:
Port: MultiTech Systems USB EDGE Wireless Modem (COM3)
Number: +447407670139
Speed: 0,00 MPM

Folders

Accepted (0/0)
Forwarded (0/0)
Deleted (0)

Description

This option allows you to use a mobile phone or wireless modem connected to the PC with a phone-to-PC data cable to send and receive SMS messages. The mobile phone or modem contains a SIM card, which you can buy from any GSM service provider.

Text message

Please create an SMS message by providing the recipient and the message text.

Recipient

To: +447464000546

SMS

Hello, This is a test message from Ozeki SMS Gateway. Please visit www.ozeki.hu for more information.

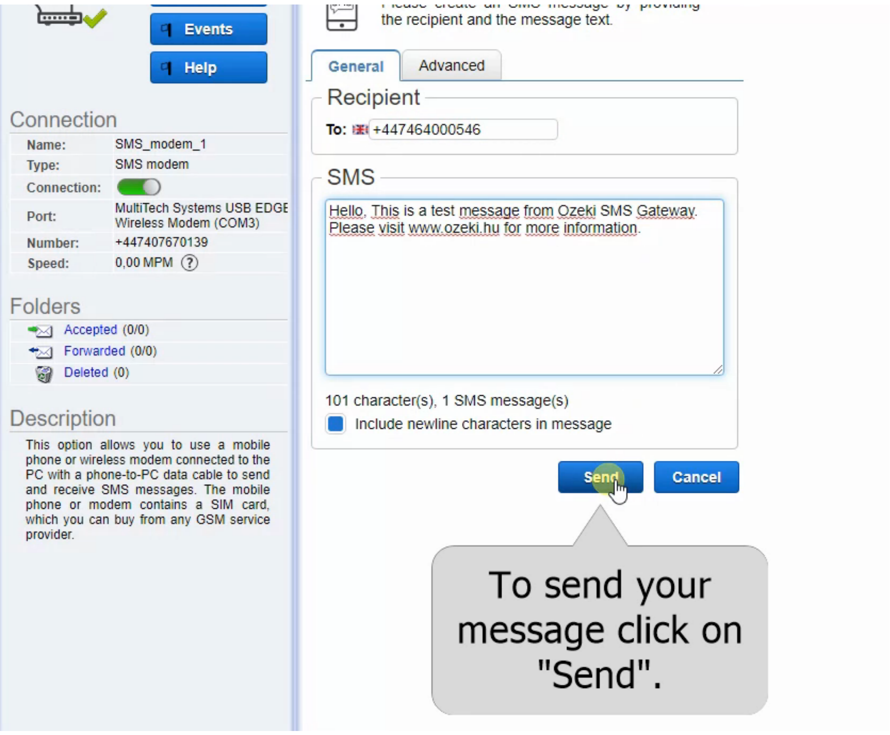
101 character(s), 1 SMS message(s)

■ Include newline characters in message

Write your message here.

Figure 15 - The text of the SMS message

If you are ready to send the SMS message, just click on the **Send** button (**Figure 16**).



Connection

Name: SMS_modem_1
Type: SMS modem
Connection:
Port: MultiTech Systems USB EDGE Wireless Modem (COM3)
Number: +447407670139
Speed: 0,00 MPM

Folders

- Accepted (0/0)
- Forwarded (0/0)
- Deleted (0)

Description

This option allows you to use a mobile phone or wireless modem connected to the PC with a phone-to-PC data cable to send and receive SMS messages. The mobile phone or modem contains a SIM card, which you can buy from any GSM service provider.

the recipient and the message text.

Recipient

To: +447464000546

SMS

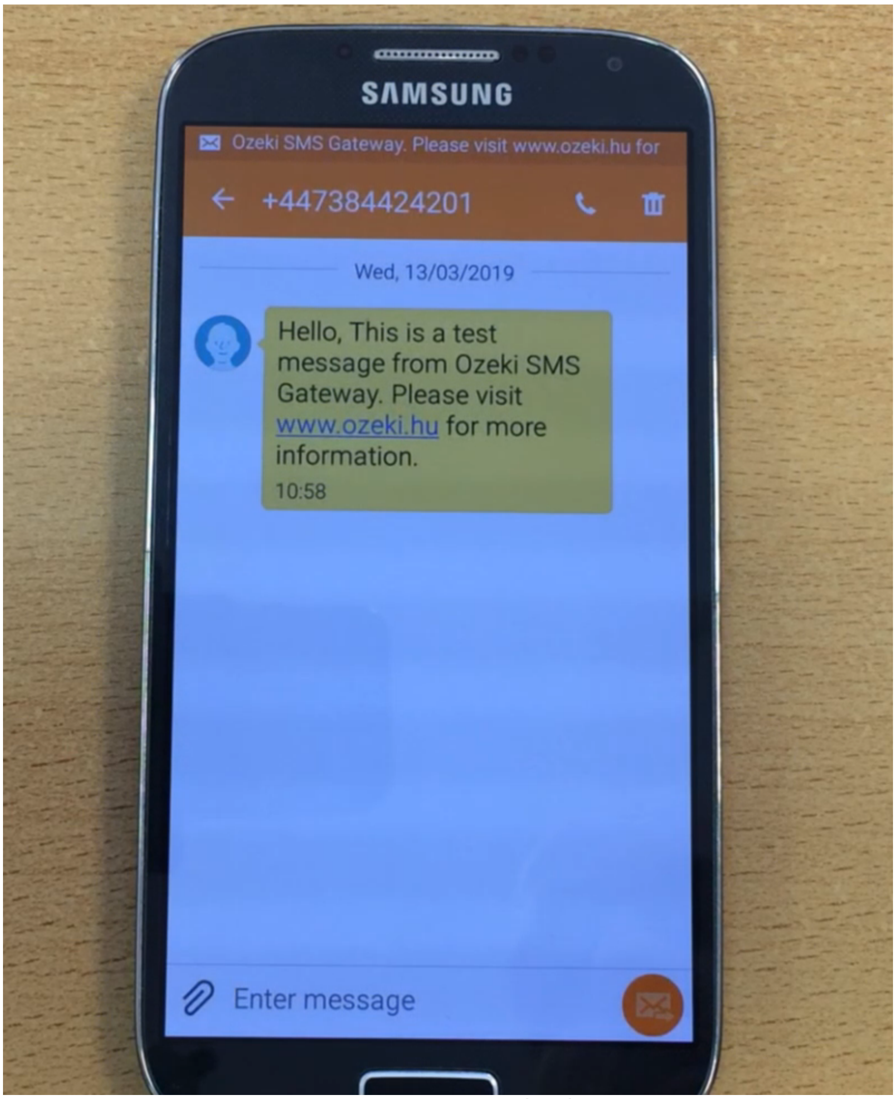
Hello, This is a test message from Ozeki SMS Gateway. Please visit www.ozeki.hu for more information.

101 character(s), 1 SMS message(s)

☑ Include newline characters in message

Figure 16 - The SMS can be sent by clicking on the Send button

If you did everything right, the message will be received by the recipient as you can see on **Figure 17**.



SAMSUNG
Ozeki SMS Gateway. Please visit www.ozeki.hu for
+447384424201
Wed, 13/03/2019
Hello, This is a test message from Ozeki SMS Gateway. Please visit www.ozeki.hu for more information.
10:58
Enter message

Figure 17 - The received SMS message

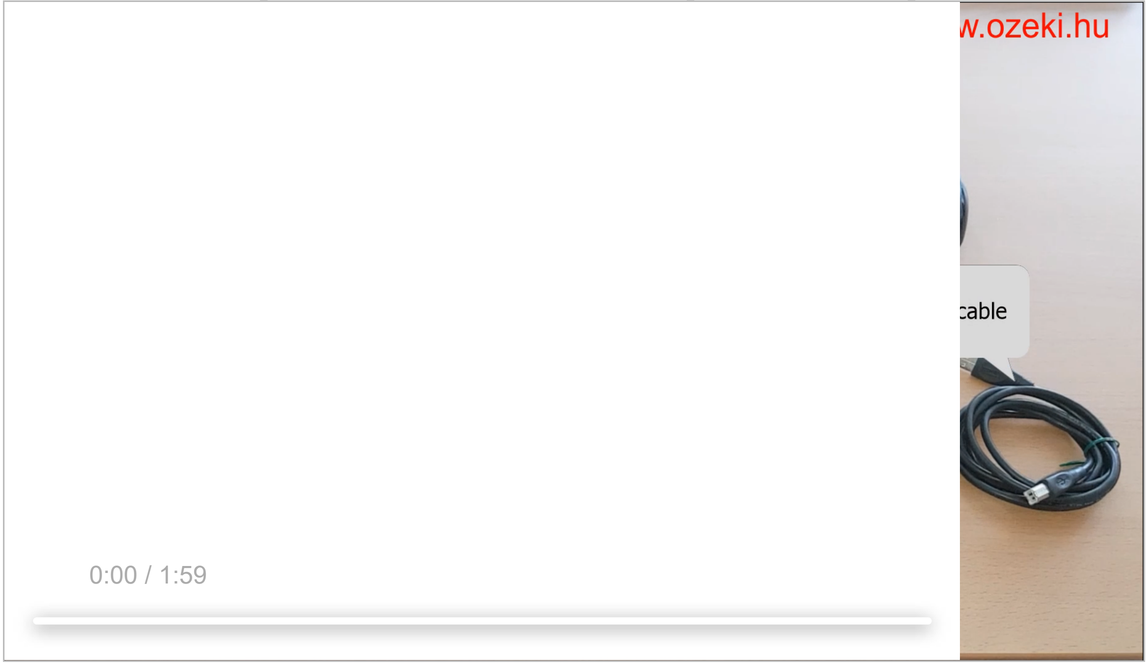
# Quick Data sheet

| General | |
|---|---|
| Form Factor | External |
| Manufacturer | Multi-Tech Systems Inc. |
| Interface Type | USB |
| Connectivity Technology | wireless |
| Max Transfer Rate | 240 Kbps |
| Fax Compatibility | G3 |
| Cellular Protocol | EDGE, GPRS, GSM |
| Antenna | External detachable |
| Antenna Qty | 1 |
| Package Type | OEM |
| Connector Type | RJ-9 |
| Min Operating Temperature | -30°C |
| Max Operating Temperature | +65°C |
| Brand | Multi-Tech |
| Product-Line | Multi-Tech MultiModem EDGE |
| Model | MTCBA-E-U |
| Packaged Quantity | 1 |
| Compatibility | PC |

# SMS modem pool

*This guide gives information on how to connect SMS modem pools to Ozeki 10 SMS Gateway. Setting up a modem pool is usually very easy. Modems in the modem pool will be handled as independent modems in your system. In this guide you can read the detailed steps for configuring an 8 port GSM modem pool. The steps are similar for larger 16, 32, 64, 128 port pools. Modem pools are a good choice if you want to operate many phone numbers. To get the phone numbers, you need to purchase SIM cards for each modem slot in the modem pool. The SIM cards will have their mobile phone nubmers assigned to them, and you can use these phone numbers in the SMS routing in Ozeki 10 SMS gateway.*

## 8 port GSM modem pool setup



**Start the Video**

## Ozeki NG 10 configuration

**Start the Video**

# Set up your 8 port GSM modem pool

The first step of setting up your modem pool is to insert a SIM card into one or more ports. The SIM card can be plugged into the modem as **Figure 1** shows.

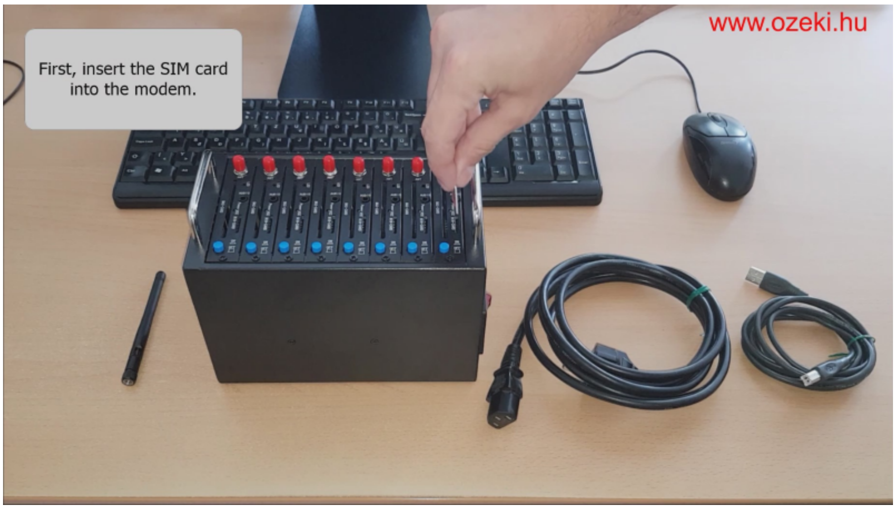


**Figure 1 -** Insert the SIM card

Next, you need to connect the antenna to the ports that have a SIM card inserted into. You can connect the antenna right after you removed the cover like in **Figure 2**.

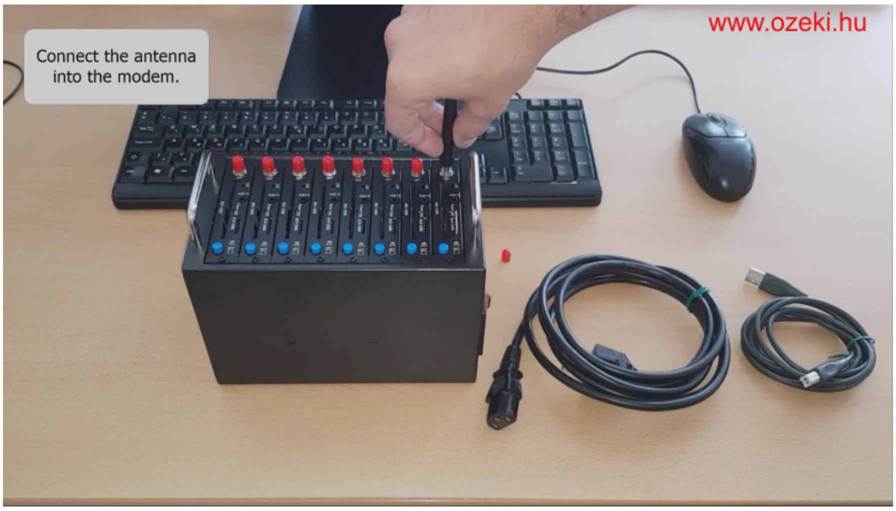


**Figure 2 -** Connect the antenna to the selected ports

To use the modem pool, next you have to connect it to a light socket with a power cable. As you can see on **Figure 3**, the power cable can be connected to the modem pool easily.

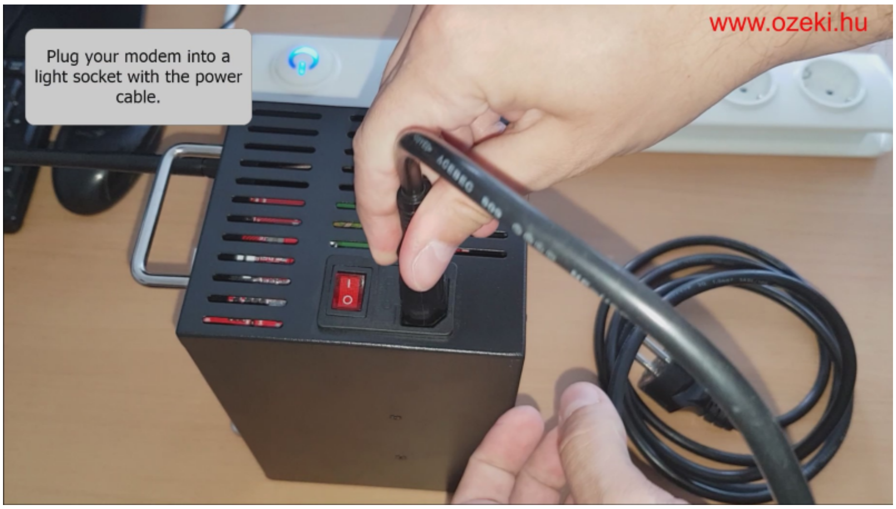


**Figure 3 -** The power cable plugged into the modem pool

After you connected the modem pool to the power supply, you just need to turn on the modem pool by switching on the button as you can see it on **Figure 4**. If your modem is under power, the button lights up.

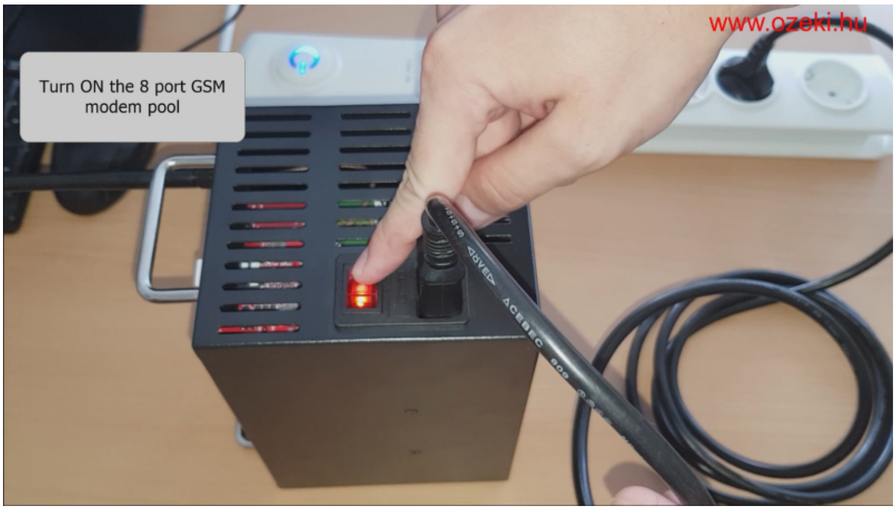


**Figure 4 -** The modem pool is now turned on

Next, you need to turn on the selected GSM port as well like in **Figure 5**.

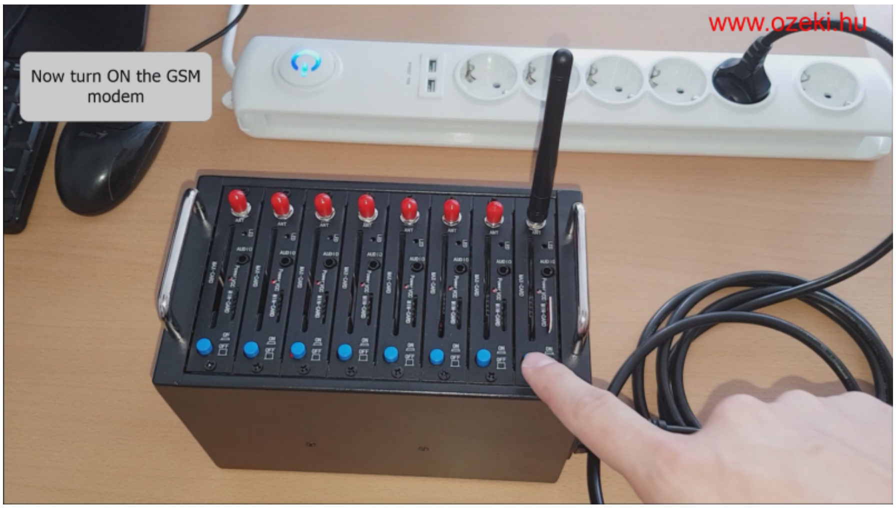


**Figure 5 -** The ports can be turned on or off separately

The connectivity of the ports can be checked easily by the blue light next to the antenna (**Figure 6**). If it starts to blink, the SIM card connected right.

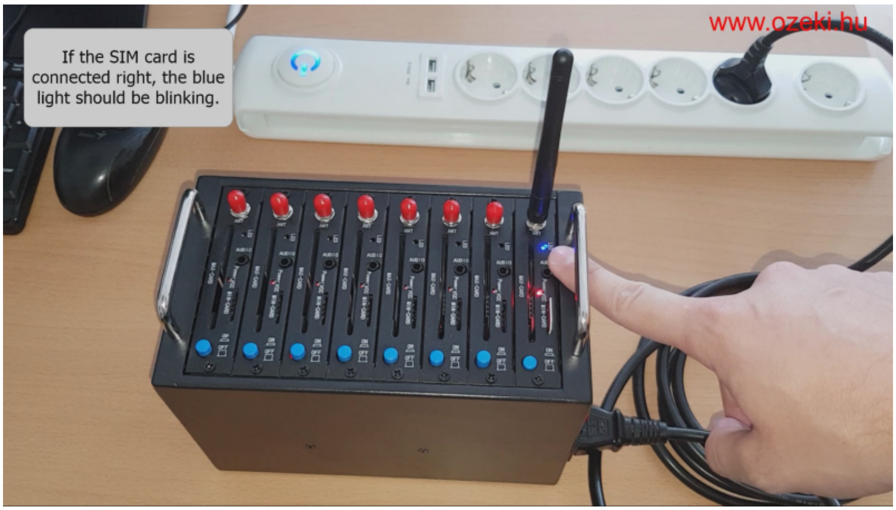

www.ozeki.hu
If the SIM card is connected right, the blue light should be blinking.

**Figure 6 -** The blue light indicates the status of connectivity

The last step of the setup process is to connect the modem pool to the PC. For that, as you can see on **Figure 7**, plug the USB cable into your modem pool.

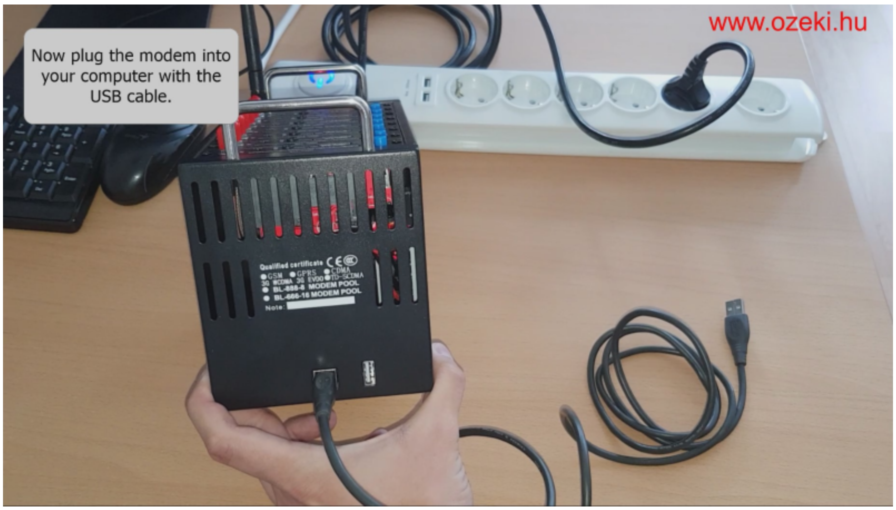


**Figure 7 -** Plug the USB cable into the modem pool

To finish the setup, just insert the other end of the cable into your computer (**Figure 8**).

**Figure 8 -** The modem connected to the PC via the USB cable

# Configure the modem pool with Ozeki NG 10

Now your modem is ready to be configured. The first step you have to fulfill is to open Ozeki NG 10 on your desktop (**Figure 9**).

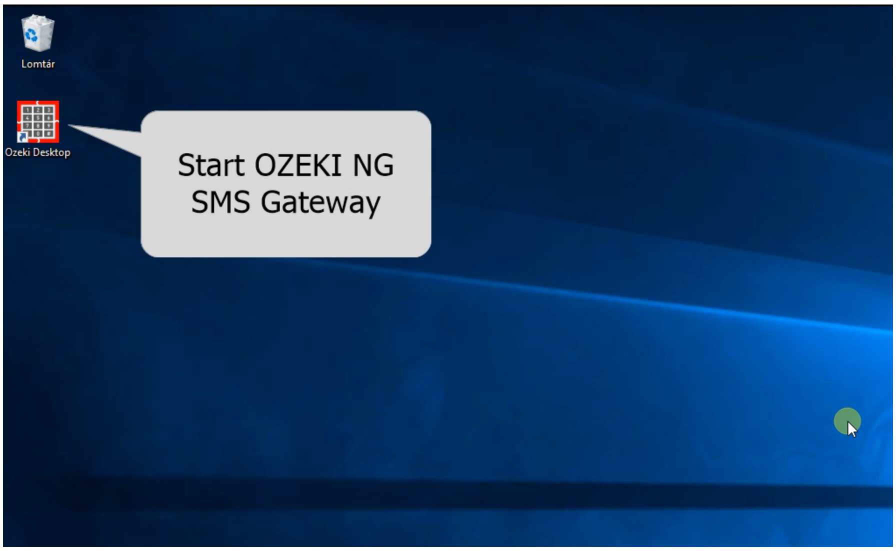
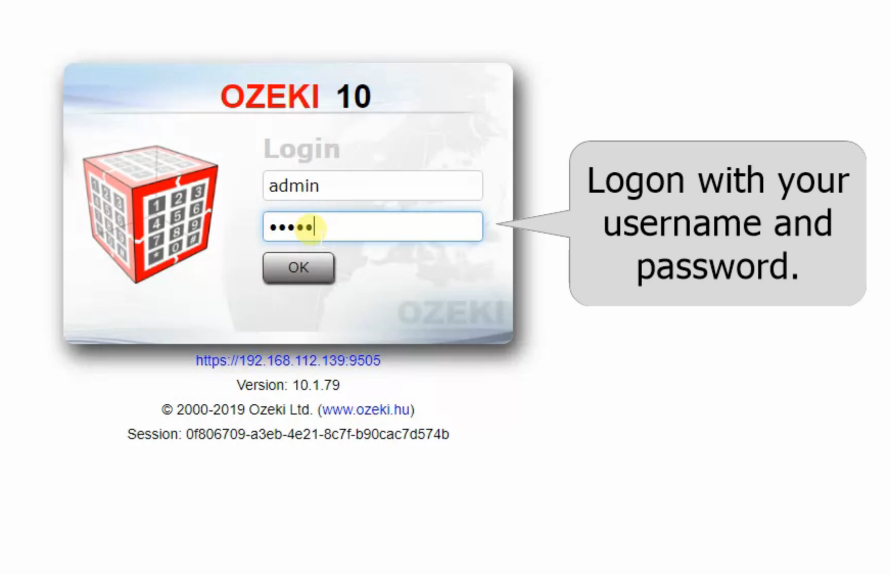


**Figure 9 -** Open Ozeki NG 10

Next, log in with your Ozeki NG 10 account by typing your username and password as you can see on **Figure 10** This account has to be created during the installation process.



admin

https://192.168.112.139:9505

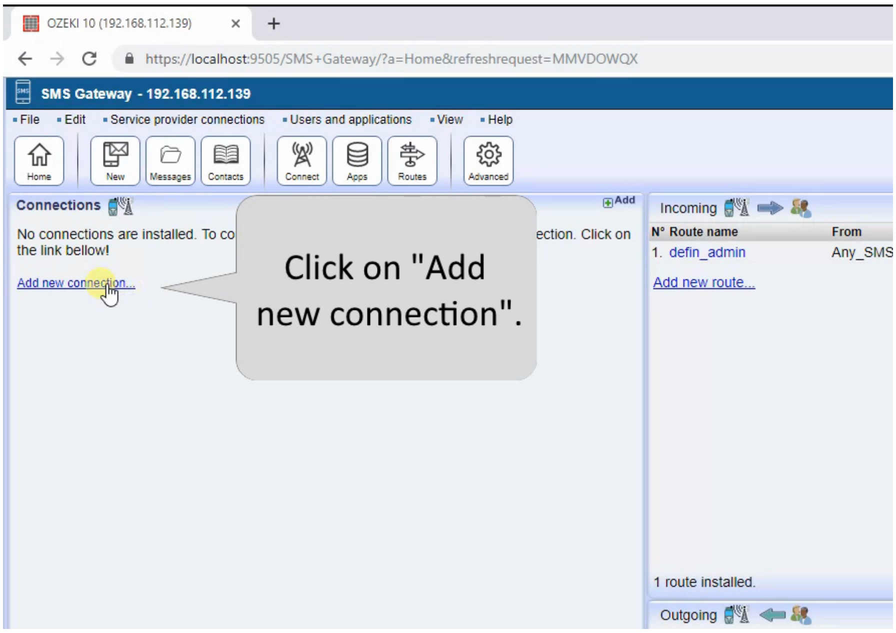
Version: 10.1.79

© 2000-2019 Ozeki Ltd. (www.ozeki.hu)

Session: 0f806709-a3eb-4e21-8c7f-b90cac7d574b

**Figure 10 -** Login with your Ozeki NG 10 account

Here, in the SMS Gateway application, the first thing that you need to do is to create a new connection. This can be done by clicking on **Add new connection** as you can see on **Figure 11**.


OZEKI 10 (192.168.112.139)
https://localhost:9505/SMS+Gateway/?a=Home&refreshrequest=MMVDOWQX
SMS Gateway - 192.168.112.139
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Contacts
Connect
Apps
Routes
Advanced
Connections
Add
Incoming
Outgoing

No connections are installed. To co ection. Click on the link bellow!

Add new connection...

Click on "Add new connection".

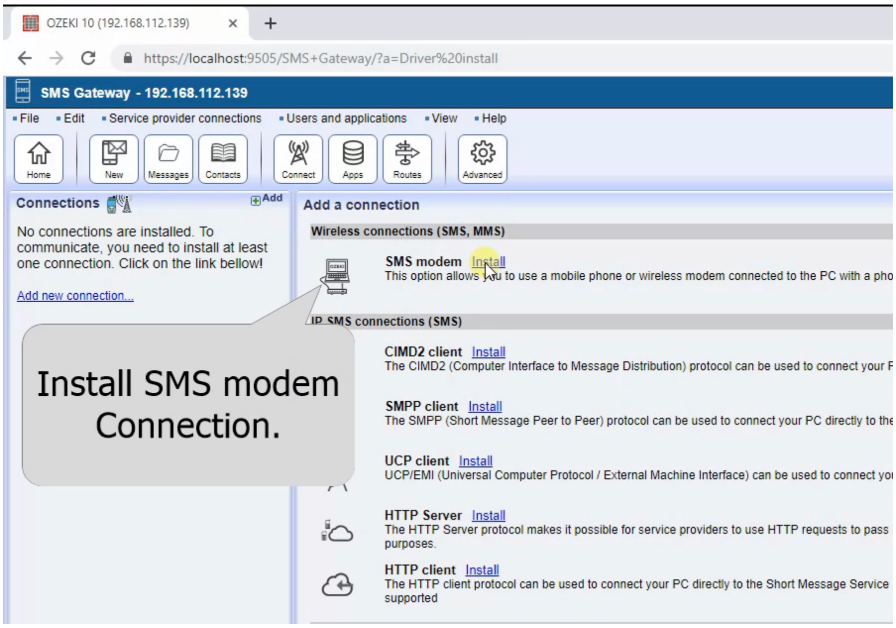
| N° | Route name | From |
|---|---|---|
| 1. | defin_admin | Any_SMS |

Add new route...

1 route installed.

**Figure 11** - Add new connection in the SMS Gateway

The next window lists you all the connections that you can create in the SMS Gateway application. Now, as **Figure 12** demonstrates, you have to select the **SMS modem** connection by clicking on **Install**.


OZEKI 10 (192.168.112.139)
https://localhost:9505/SMS+Gateway/?a=Driver%20install
SMS Gateway - 192.168.112.139
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Contacts
Connect
Apps
Routes
Advanced
Connections
Add

No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

Install SMS modem Connection.

Add a connection

Wireless connections (SMS, MMS)

SMS modem Install
This option allows you to use a mobile phone or wireless modem connected to the PC with a pho

IP SMS connections (SMS)

CIMD2 client Install
The CIMD2 (Computer Interface to Message Distribution) protocol can be used to connect your P

SMPP client Install
The SMPP (Short Message Peer to Peer) protocol can be used to connect your PC directly to the

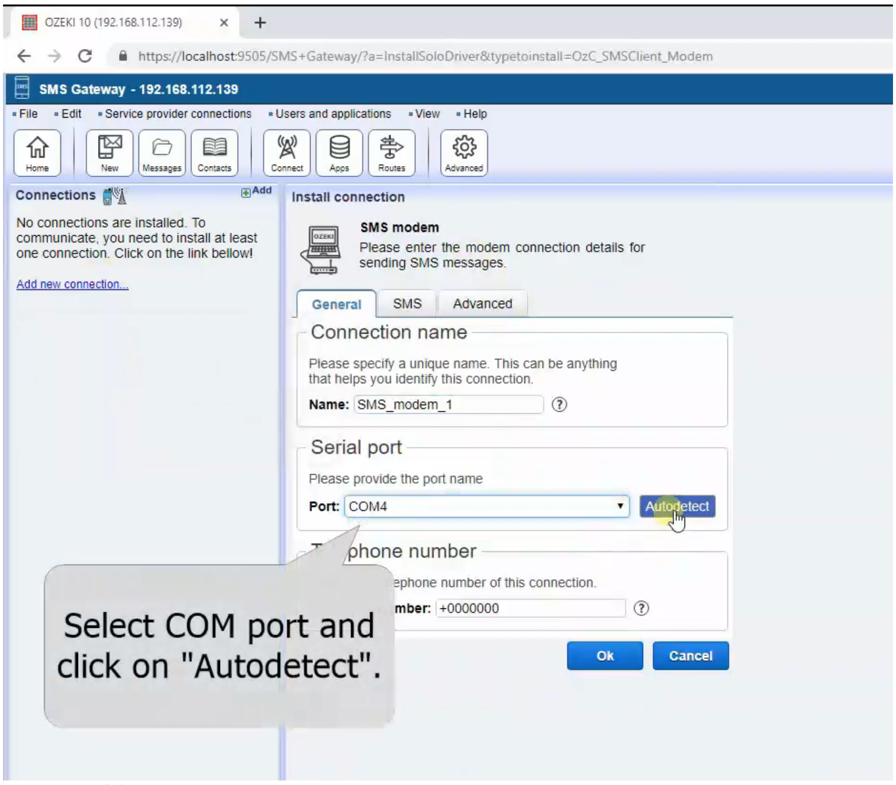
UCP client Install
UCP/EMI (Universal Computer Protocol / External Machine Interface) can be used to connect yo

HTTP Server Install
The HTTP Server protocol makes it possible for service providers to use HTTP requests to pass purposes.

HTTP client Install
The HTTP client protocol can be used to connect your PC directly to the Short Message Service supported

**Figure 12** - Select the SMS modem connection to install

In the installation menu, you need to select the port, where you connected the modem. For that, just click on the **Autodetect** button as you can see on **Figure 13**.



No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

SMS modem

Please enter the modem connection details for sending SMS messages.

Connection name

Please specify a unique name. This can be anything that helps you identify this connection.

Name: SMS_modem_1

Serial port

Please provide the port name

Port: COM4

T phone number

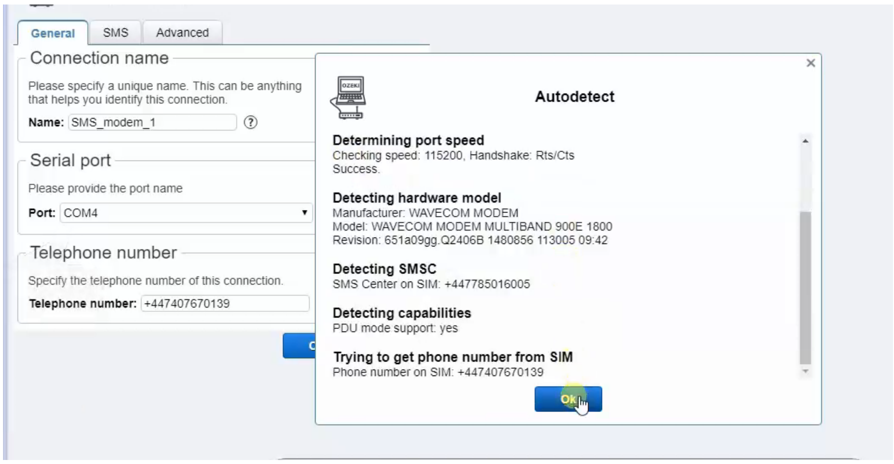
ephone number of this connection.

mber: +0000000

Select COM port and click on "Autodetect".

**Figure 13 -** Autodetect the port where you connected the modem

This Autodetect option will detect your modem, and if it is connected right the following window should show up like in **Figure 14**.



Connection name

Please specify a unique name. This can be anything that helps you identify this connection.

Name: SMS_modem_1

Serial port

Please provide the port name

Port: COM4

Telephone number

Specify the telephone number of this connection.

Telephone number: +447407670139

Determining port speed
Checking speed: 115200, Handshake: Rts/Cts
Success.

Detecting hardware model
Manufacturer: WAVECOM MODEM
Model: WAVECOM MODEM MULTIBAND 900E 1800
Revision: 651a09gg.Q2406B 1480856 113005 09:42

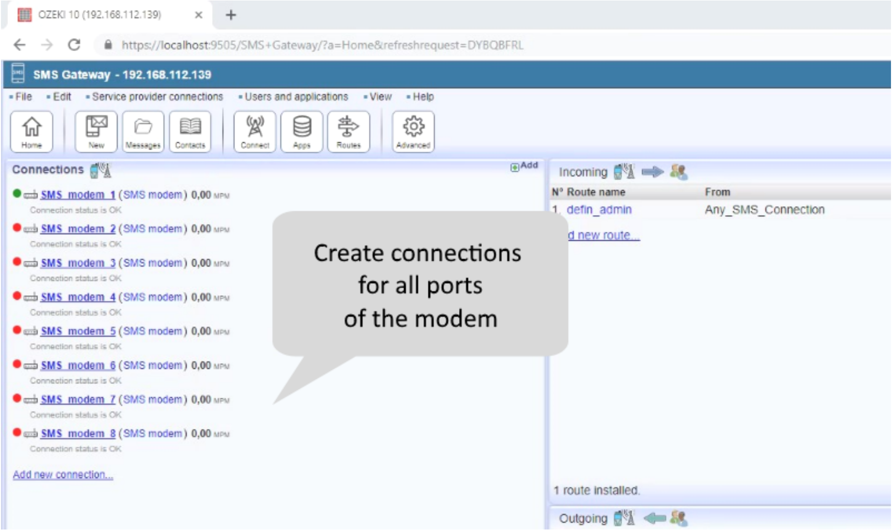
Detecting SMSC
SMS Center on SIM: +447785016005

Detecting capabilities
PDU mode support: yes

Trying to get phone number from SIM
Phone number on SIM: +447407670139

**Figure 14 -** Details of the detected modem

You can do this process with all the other ports, so like in **Figure 15**, you can manage maximum eight SMS modem connection at the same time.

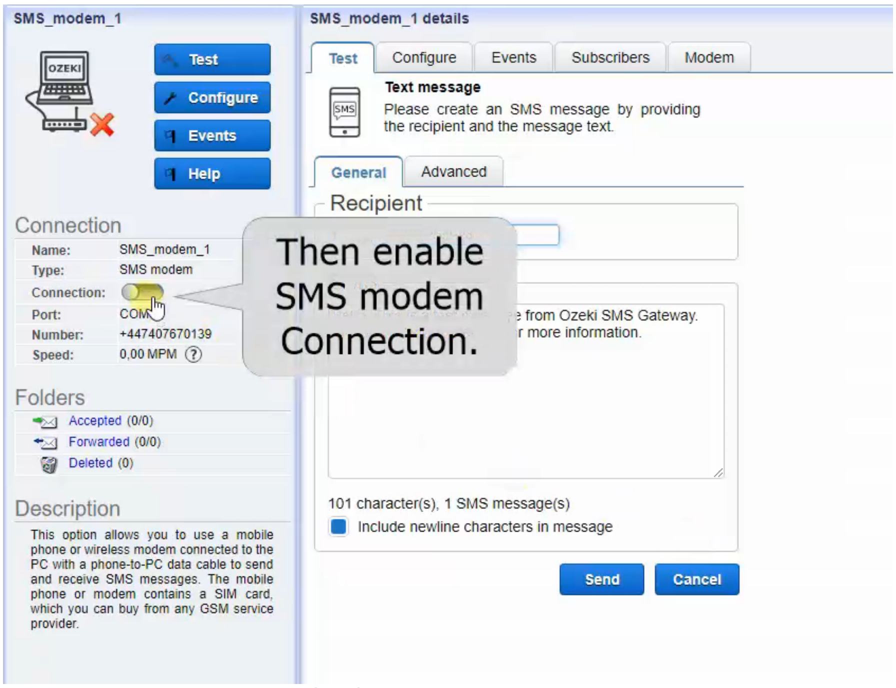

OZEKI 10 (192.168.112.139)
https://localhost:9505/SMS+Gateway/?a=Home&refreshrequest=DYBQBFRL
SMS Gateway - 192.168.112.139
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Contacts
Connect
Apps
Routes
Advanced
Connections
Add
SMS_modem_1 (SMS modem) 0,00 MPM
Connection status is OK
SMS_modem_2 (SMS modem) 0,00 MPM
Connection status is OK
SMS_modem_3 (SMS modem) 0,00 MPM
Connection status is OK
SMS_modem_4 (SMS modem) 0,00 MPM
Connection status is OK
SMS_modem_5 (SMS modem) 0,00 MPM
Connection status is OK
SMS_modem_6 (SMS modem) 0,00 MPM
Connection status is OK
SMS_modem_7 (SMS modem) 0,00 MPM
Connection status is OK
SMS_modem_8 (SMS modem) 0,00 MPM
Connection status is OK
Add new connection...
Incoming
N° Route name
From
1 defin_admin
Any_SMS_Connection
d new route...
1 route installed.
Outgoing

Create connections for all ports of the modem

**Figure 15 -** Connection for each port

# Send an SMS message using the modem connection

At this point, you have an SMS modem connection in the SMS Gateway application. To handle SMS messages with this connection, first, you need to enable the connection (**Figure 16**) by turning on the switch button in the connection details window.

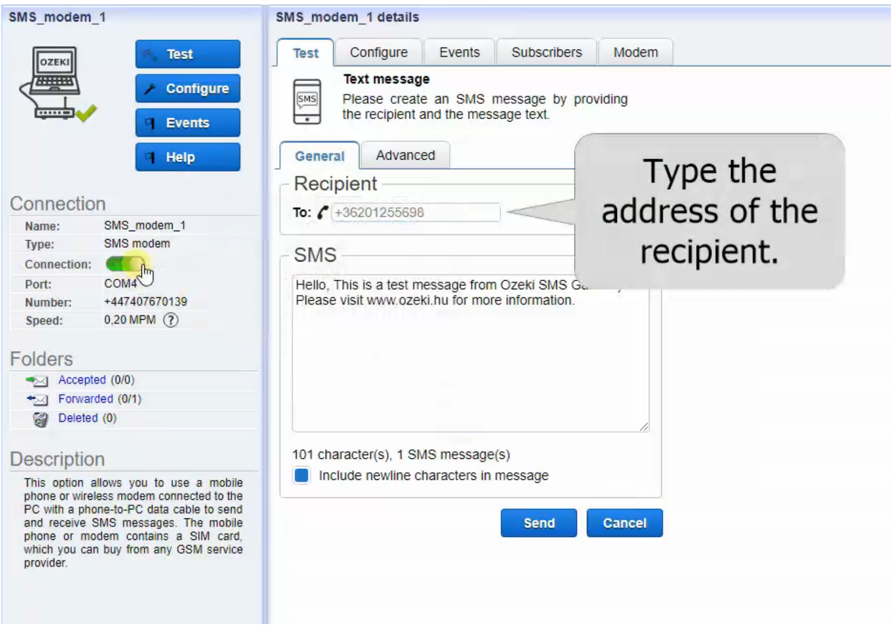


This option allows you to use a mobile phone or wireless modem connected to the PC with a phone-to-PC data cable to send and receive SMS messages. The mobile phone or modem contains a SIM card, which you can buy from any GSM service provider.

Then enable SMS modem Connection.

**Figure 16 -** Turn on the SMS modem connection

To send the SMS message, you need to type the phone number of the recipient (**Figure 17**) in the **Recipient** field.



Text message
Please create an SMS message by providing the recipient and the message text.

Recipient
To: +36201255698

SMS
Hello, This is a test message from Ozeki SMS Ga...
Please visit www.ozeki.hu for more information.

101 character(s), 1 SMS message(s)
Include newline characters in message

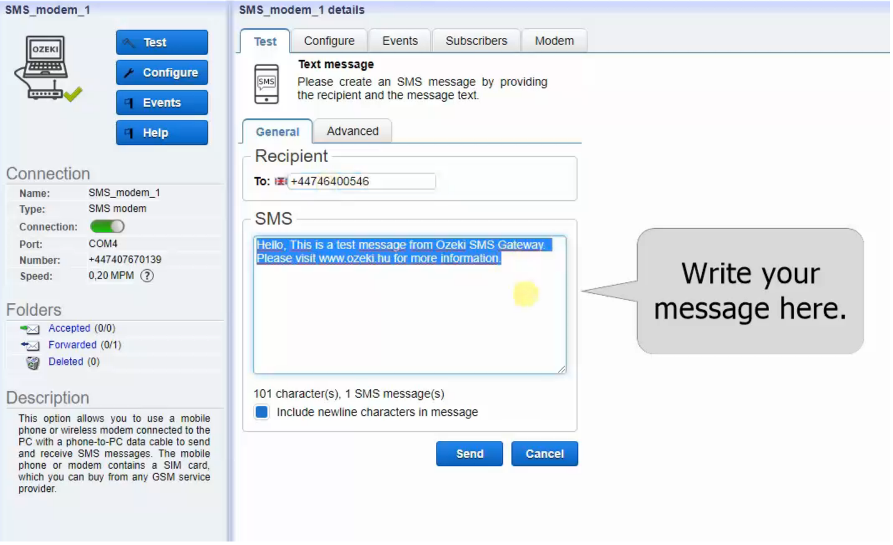
Connection
Name: SMS_modem_1
Type: SMS modem
Connection:
Port: COM4
Number: +447407670139
Speed: 0,20 MPM

Folders
Accepted (0/0)
Forwarded (0/1)
Deleted (0)

Description
This option allows you to use a mobile phone or wireless modem connected to the PC with a phone-to-PC data cable to send and receive SMS messages. The mobile phone or modem contains a SIM card, which you can buy from any GSM service provider.

**Figure 17 -** Type the address where you want to send the SMS message

Next, just type the text of the SMS message to the **SMS** field like in **Figure 18**.



Text message
Please create an SMS message by providing the recipient and the message text.

Recipient
To: +44746400546

SMS
Hello, This is a test message from Ozeki SMS Gateway.
Please visit www.ozeki.hu for more information.

101 character(s), 1 SMS message(s)
Include newline characters in message

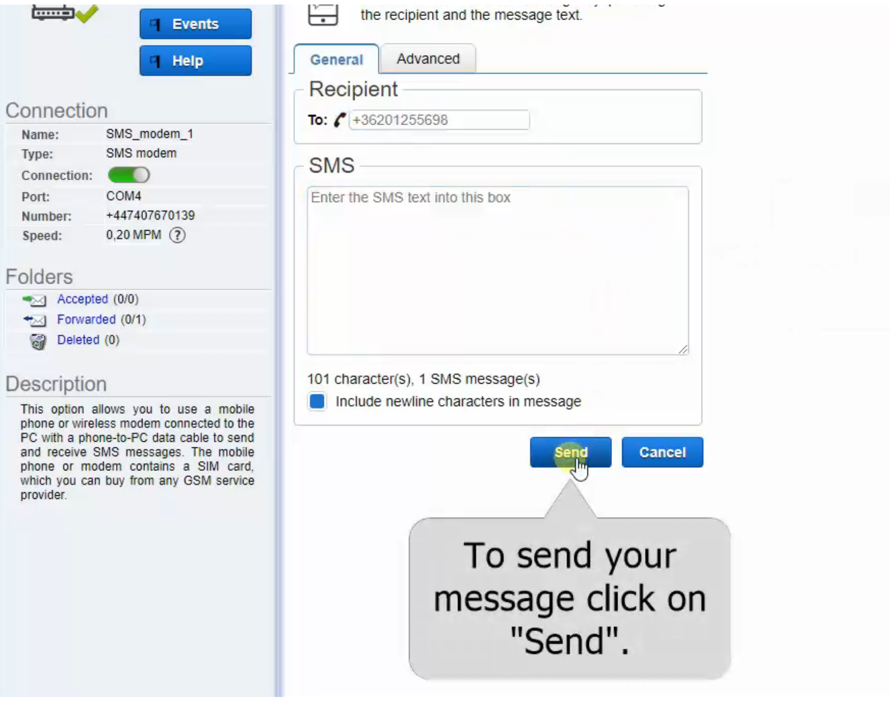
Connection
Name: SMS_modem_1
Type: SMS modem
Connection:
Port: COM4
Number: +447407670139
Speed: 0,20 MPM

Folders
Accepted (0/0)
Forwarded (0/1)
Deleted (0)

Description
This option allows you to use a mobile phone or wireless modem connected to the PC with a phone-to-PC data cable to send and receive SMS messages. The mobile phone or modem contains a SIM card, which you can buy from any GSM service provider.

**Figure 18 -** The text of the SMS message

If you are ready to send the SMS message, just click on the **Send** button (**Figure 19**).



the recipient and the message text.

Connection

Name: SMS_modem_1
Type: SMS modem
Connection:
Port: COM4
Number: +447407670139
Speed: 0,20 MPM

Folders

Accepted (0/0)
Forwarded (0/1)
Deleted (0)

Description

This option allows you to use a mobile phone or wireless modem connected to the PC with a phone-to-PC data cable to send and receive SMS messages. The mobile phone or modem contains a SIM card, which you can buy from any GSM service provider.

Recipient

To: +36201255698

SMS

Enter the SMS text into this box

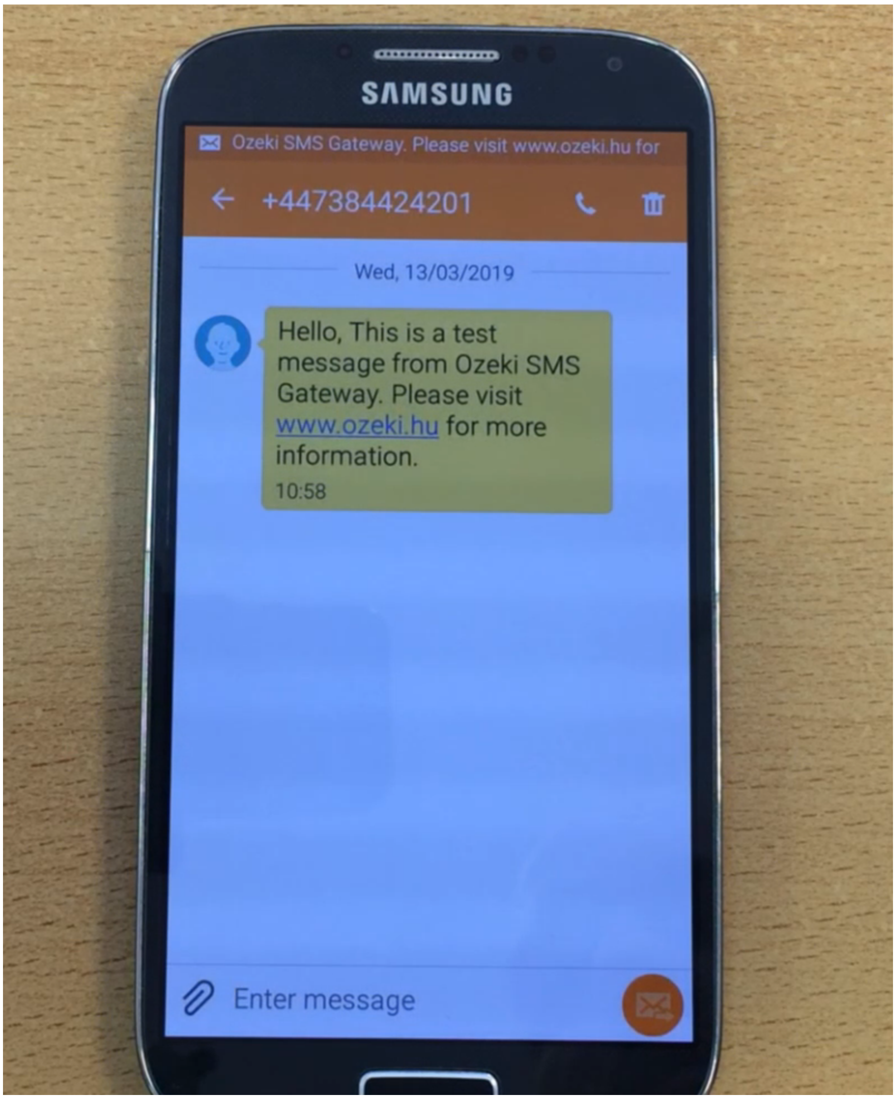
101 character(s), 1 SMS message(s)

☑ Include newline characters in message

To send your message click on "Send".

**Figure 19 -** The SMS can be sent by clicking on the Send button

If you did everything right, the message will be received by the recipient as you can see on **Figure 20**.



Hello, This is a test message from Ozeki SMS Gateway. Please visit www.ozeki.hu for more information.
10:58

**Figure 20 -** The received SMS message

# Quick Data sheet

| | |
|---|---|
| Support | Windows, Linux, Unix OS |
| Messaging capacity | 6800 SMS or 3200 MMS per hour |
| Size | 440x220x50mm |
| TCP/IP Stack | yes |
| Internet | Wireless Access to internet |
| Internet services | PPP, TCP, UDP, HTTP, SMTP, POP3 |
| SIM card slots | 8 |
| Dual band | 900/1800 Mhz |
| Tri band | 850/1800/1900 Mhz 900/1800/1900 Mhz |
| Quad band | 850/900/1800/1900 Mhz |
| Transmatting Speed | Maximum Transmatting Speed 115.2 Kbps |
| Input voltage | 100V - 240V |
| Working temperature | -20°C to +60°C |
| Storage temperature | -25°C to +70°C |

# Nexmo SMS service provider

***The Nexmo SMS service provider allows you to send and receive text and binary SMS messages. You may purchase phone numbers for different countries. Their API is HTTP/HTTPS based. For incoming delivery reports and incoming messages you will need to open a port in your firewall. Ozeki provides the implementation of their API.***

## How to create a Nexmo account

To use the Nexmo service first of all you need to create a Nexmo account. In the below section we guide you how to do it.

On the main page of Nexmo please click on the **'Sign up'** button.

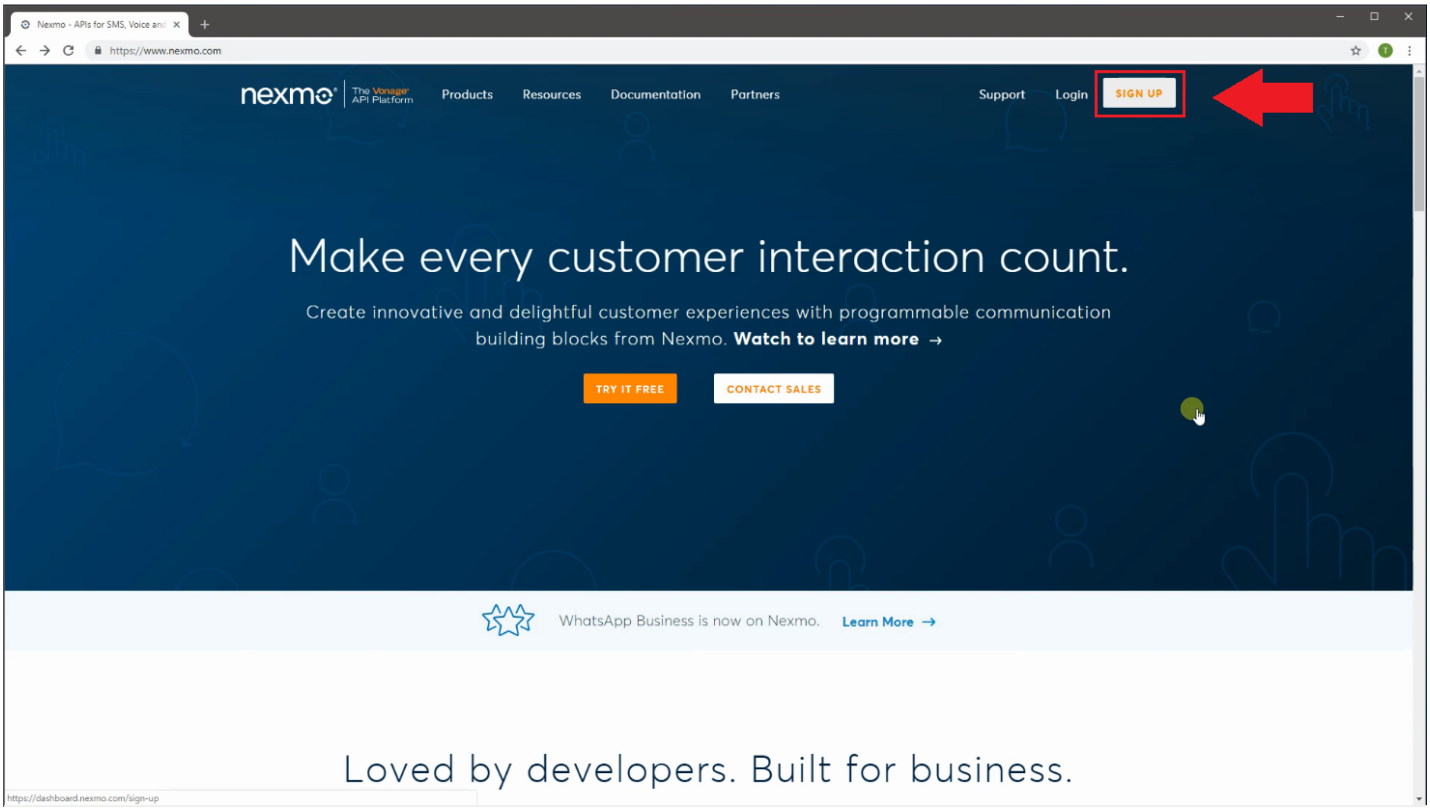
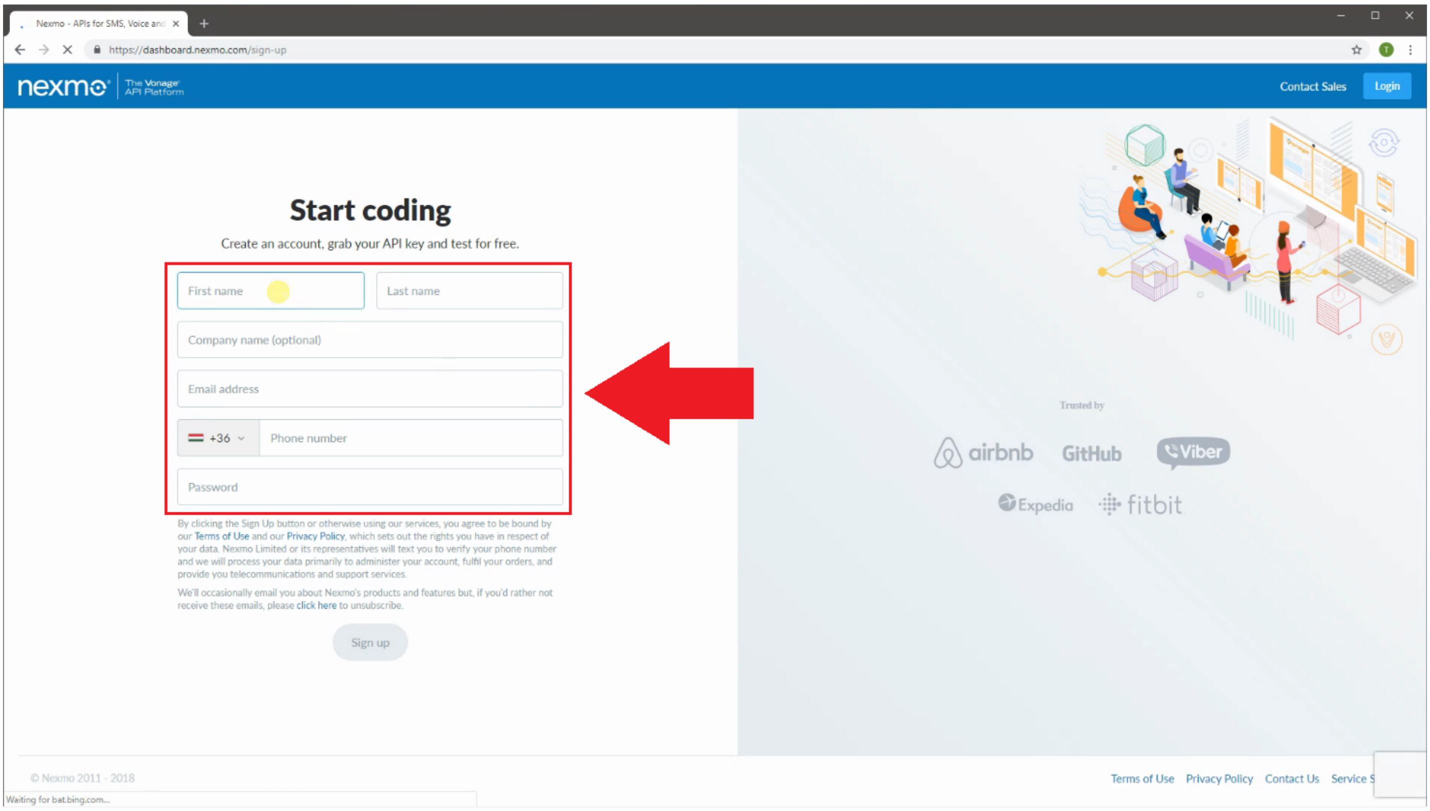

Nexmo - APIs for SMS, Voice and
https://www.nexmo.com
nexmo The Vonage API Platform
Products
Resources
Documentation
Partners
Support
Login
SIGN UP
Make every customer interaction count.
Create innovative and delightful customer experiences with programmable communication building blocks from Nexmo. Watch to learn more →
TRY IT FREE
CONTACT SALES
WhatsApp Business is now on Nexmo.
Learn More →
Loved by developers. Built for business.
https://dashboard.nexmo.com/sign-up

Figure 1 - Clicking **'Sign up'**

Then you need to provide your account details.



Nexmo - APIs for SMS, Voice and
https://dashboard.nexmo.com/sign-up
nexmo The Vonage API Platform
Contact Sales
Login

## Start coding

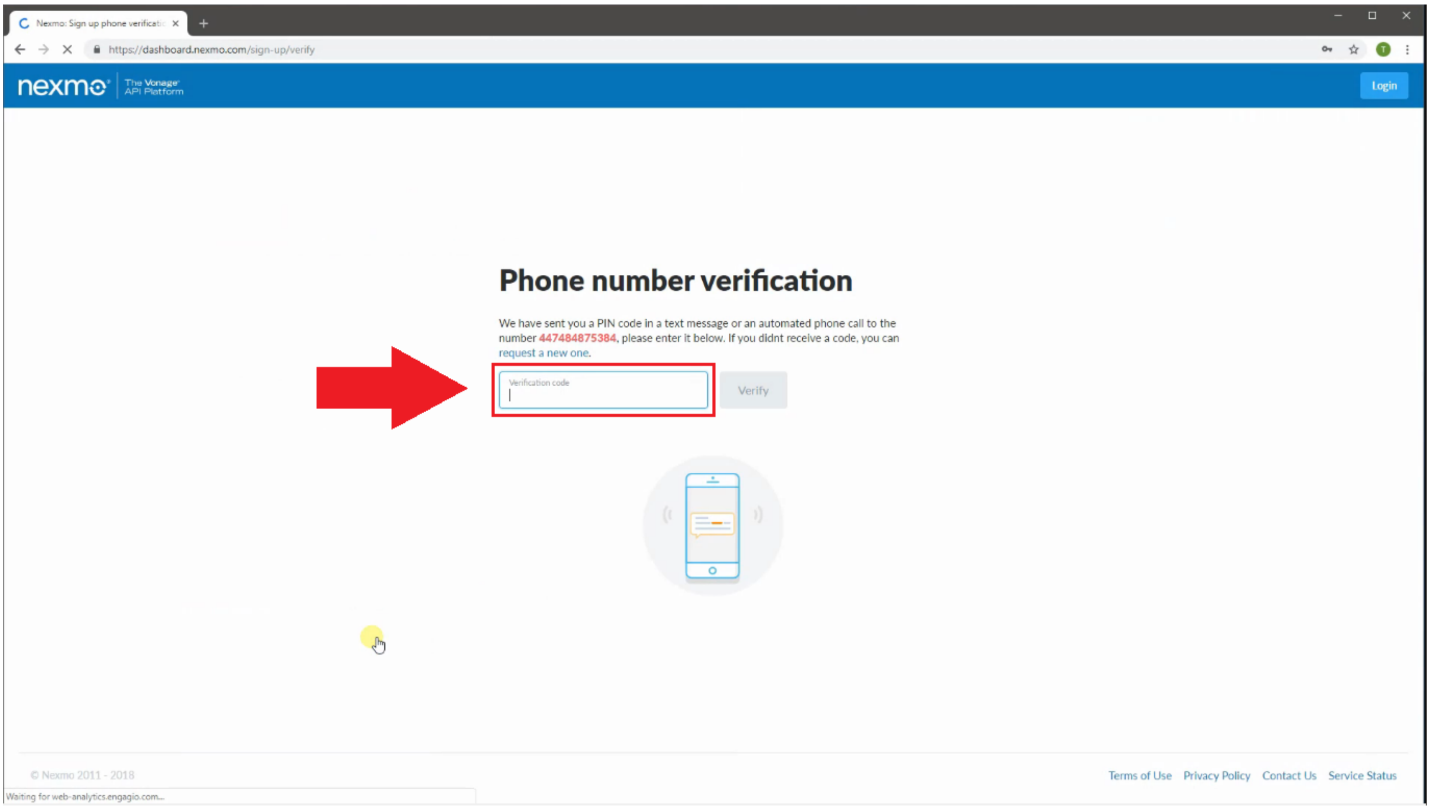
Create an account, grab your API key and test for free.

By clicking the Sign Up button or otherwise using our services, you agree to be bound by our Terms of Use and our Privacy Policy, which sets out the rights you have in respect of your data. Nexmo Limited or its representatives will text you to verify your phone number and we will process your data primarily to administer your account, fulfil your orders, and provide you telecommunications and support services.

We'll occasionally email you about Nexmo's products and features but, if you'd rather not receive these emails, please click here to unsubscribe.

Figure 2 - Providing account details

If you are finished then click on **'Sign Up'** and wait until Nexmo sends the verification code to your phone.



Nexmo: Sign up phone verificati
https://dashboard.nexmo.com/sign-up/verify
nexmo The Vonage API Platform
Login

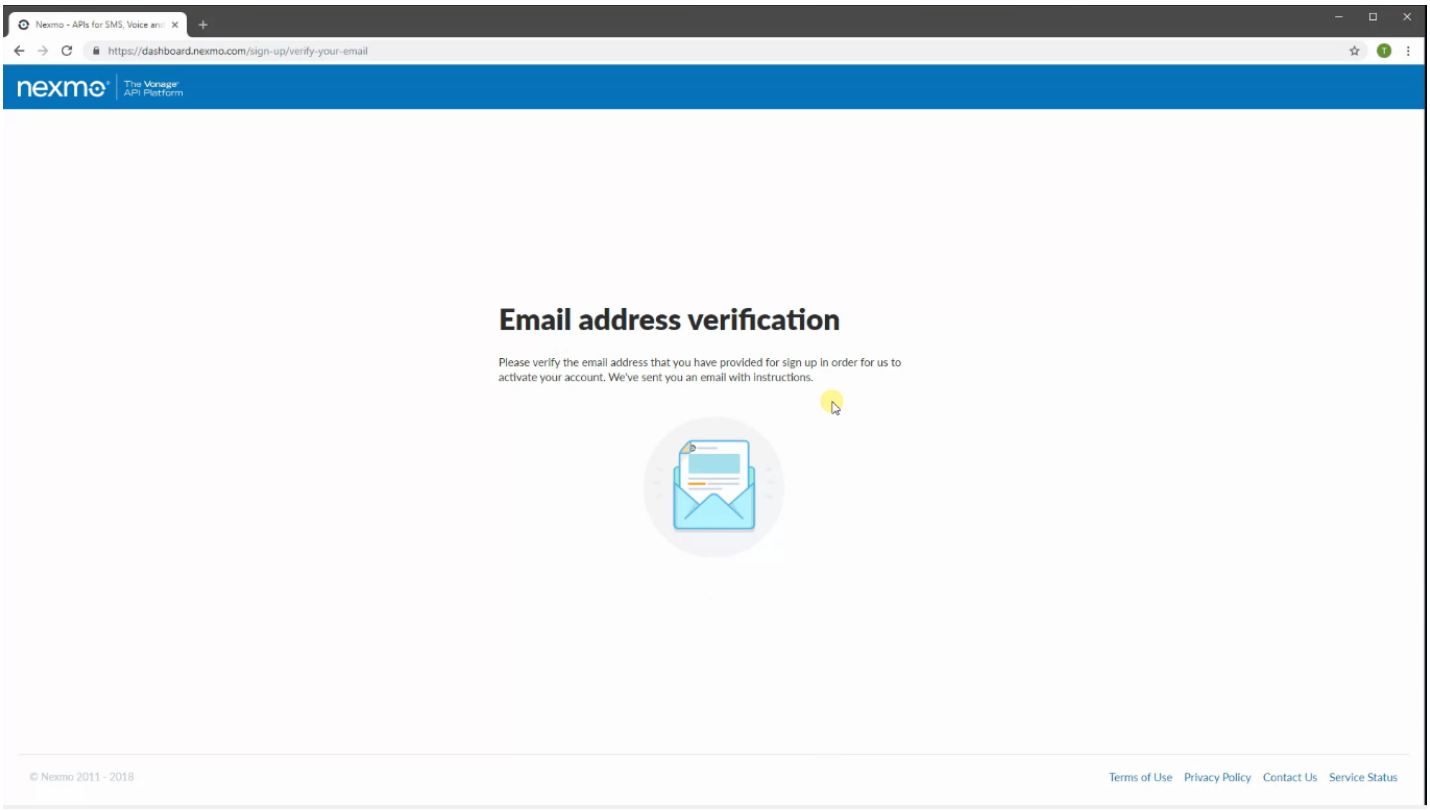
## Phone number verification

We have sent you a PIN code in a text message or an automated phone call to the number 447484875384, please enter it below. If you didnt receive a code, you can request a new one.

Verification code
Verify
© Nexmo 2011 - 2018
Terms of Use
Privacy Policy
Contact Us
Service Status
Waiting for web-analytics.engagio.com...

Figure 3 - Phone number verification

Now check your mail box and verify your E-mail address.


Nexmo - APIs for SMS, Voice and
https://dashboard.nexmo.com/sign-up/verify-your-email
nexmo The Vonage API Platform

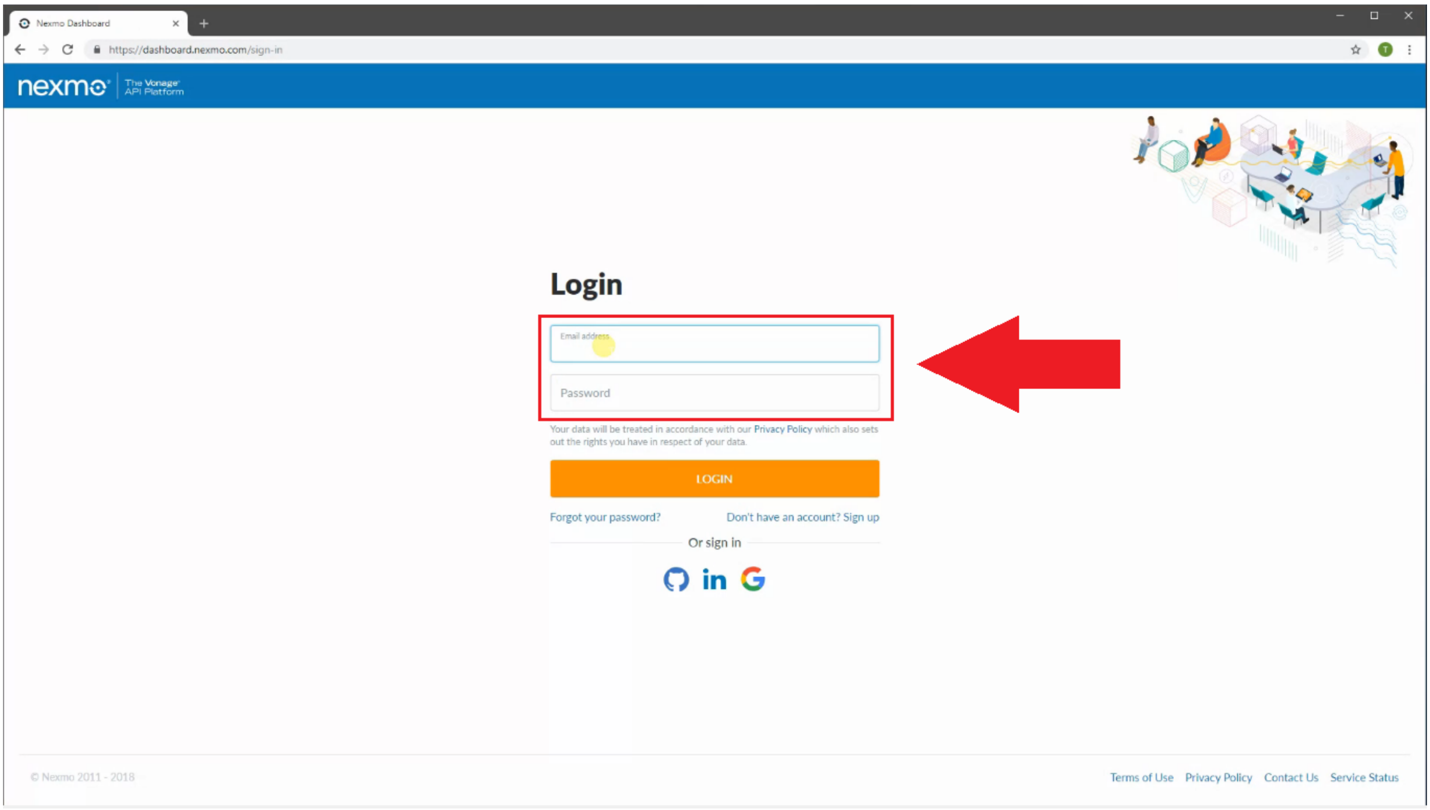
Email address verification

Please verify the email address that you have provided for sign up in order for us to activate your account. We've sent you an email with instructions.

© Nexmo 2011 - 2018
Terms of Use
Privacy Policy
Contact Us
Service Status

Figure 4 - E-mail verification

Finally Login to your new Nexmo account.


Nexmo Dashboard
https://dashboard.nexmo.com/sign-in
nexmo The Vonage API Platform

Login

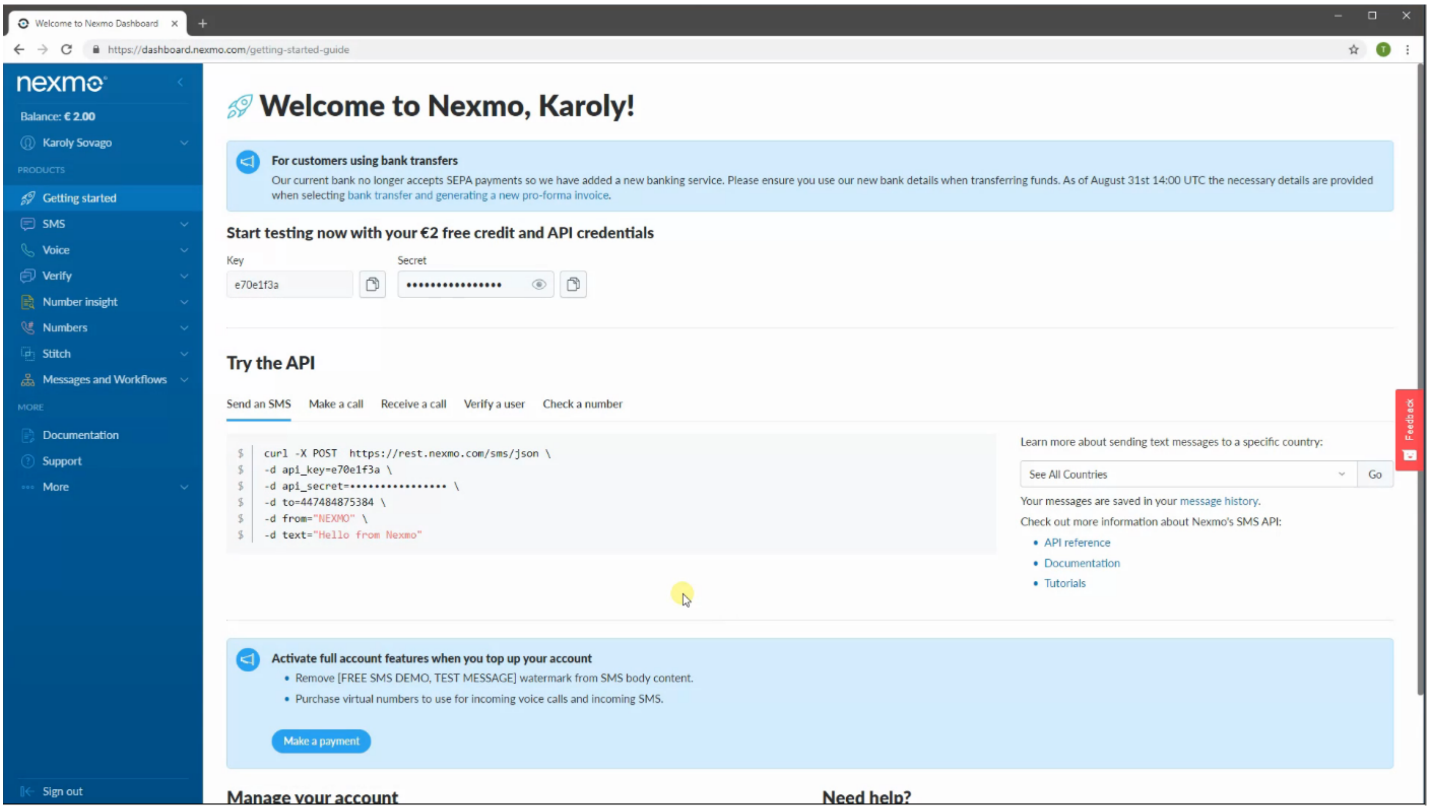
Email address

Password

Your data will be treated in accordance with our Privacy Policy which also sets out the rights you have in respect of your data.

Figure 5 - Logging in

If everything is fine you will see the **'Getting started'** page.



Welcome to Nexmo, Karoly!

**For customers using bank transfers**

Our current bank no longer accepts SEPA payments so we have added a new banking service. Please ensure you use our new bank details when transferring funds. As of August 31st 14:00 UTC the necessary details are provided when selecting bank transfer and generating a new pro-forma invoice.

**Start testing now with your €2 free credit and API credentials**

Key: e70e1f3a

Secret: ••••••••••••••••

Try the API

Send an SMS | Make a call | Receive a call | Verify a user | Check a number

```
$ curl -X POST  https://rest.nexmo.com/sms/json \
$ -d api_key=e70e1f3a \
$ -d api_secret=•••••••••••••••• \
$ -d to=447484875384 \
$ -d from="NEXMO" \
$ -d text="Hello from Nexmo"
```

Learn more about sending text messages to a specific country:

See All Countries | Go

Your messages are saved in your message history.

Check out more information about Nexmo's SMS API:

- API reference
- Documentation
- Tutorials

**Activate full account features when you top up your account**

- Remove [FREE SMS DEMO, TEST MESSAGE] watermark from SMS body content.
- Purchase virtual numbers to use for incoming voice calls and incoming SMS.

Make a payment

Manage your account

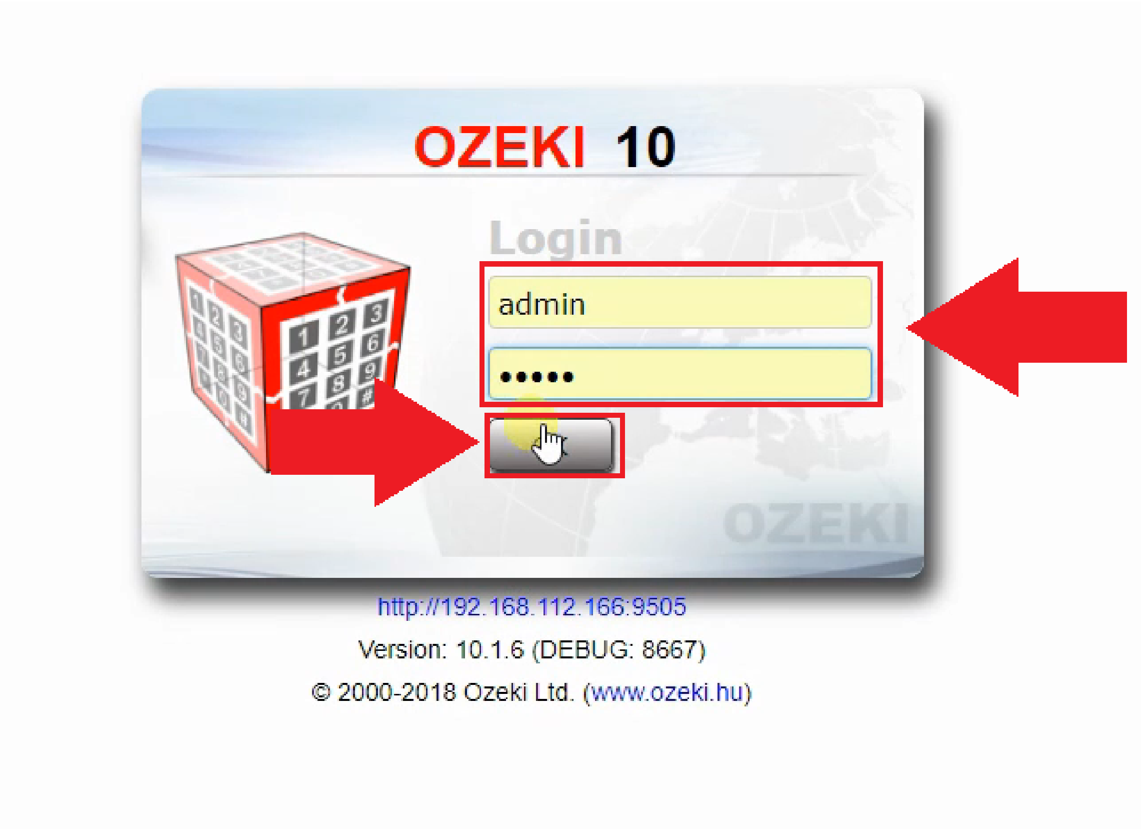
Need help?

Figure 6 - Getting started page

# How to setup Ozeki 10 and Nexmo for outbound SMS messages

This chapter provides you detailed information on how to install and configure Nexmo Connection for SMS sending in the Ozeki 10 SMS Gateway software.

At first, please Log in to Ozeki 10 with your username and password.



admin

•••••

http://192.168.112.166:9505

Version: 10.1.6 (DEBUG: 8667)

© 2000-2018 Ozeki Ltd. (www.ozeki.hu)

Figure 7 - Logging in to Ozeki 10

Then open the **'SMS Gateway'** App

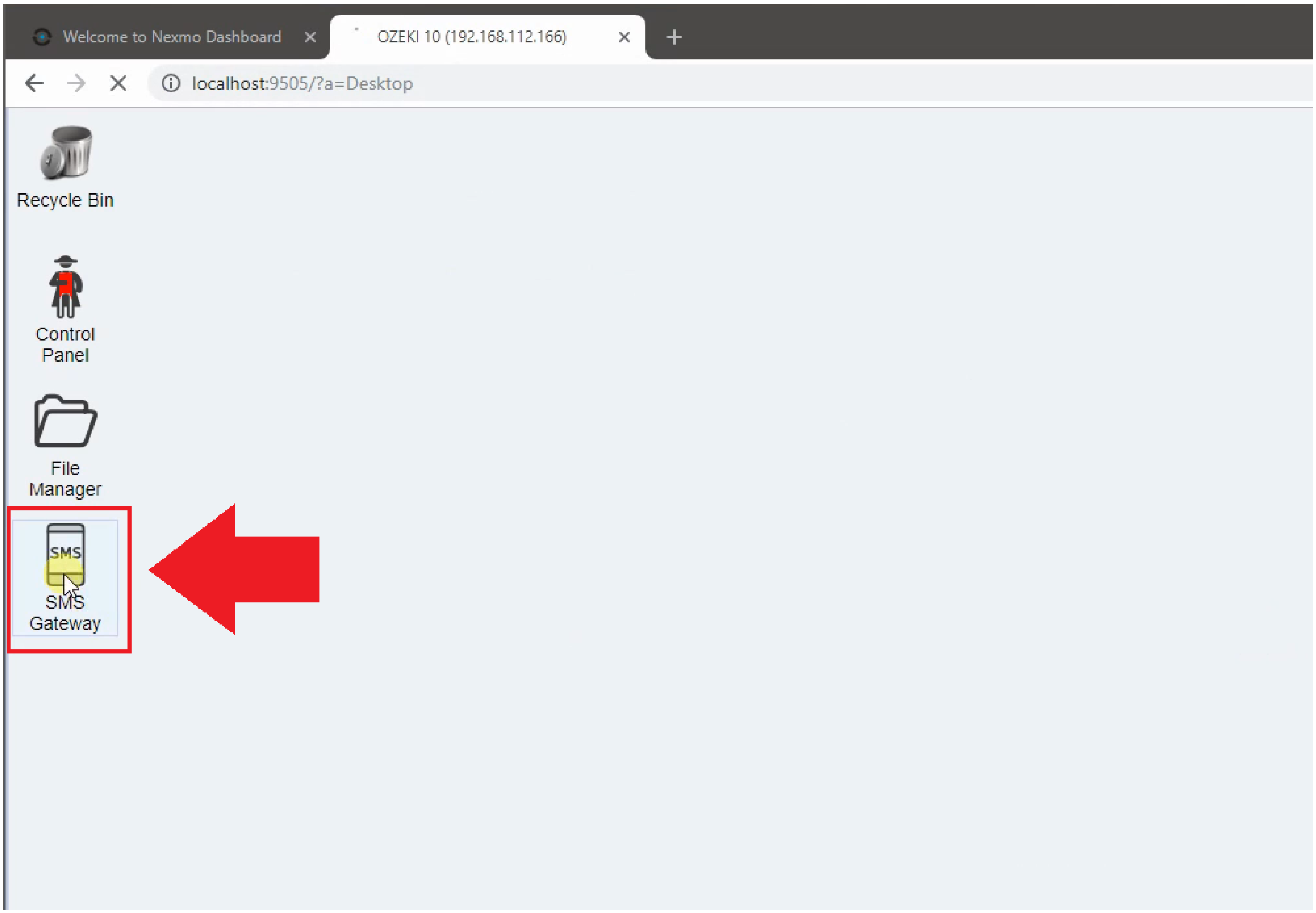
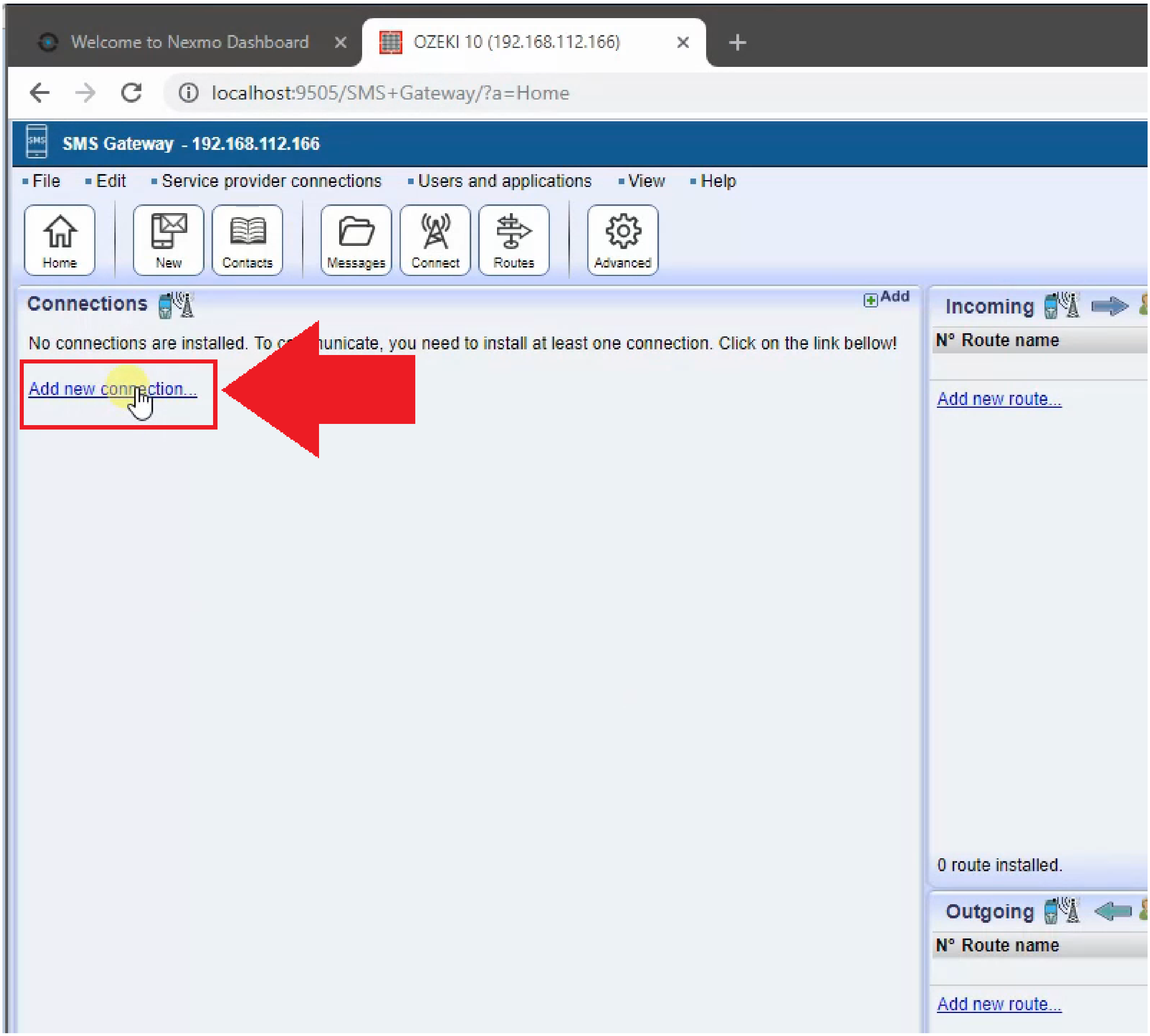

Welcome to Nexmo Dashboard
OZEKI 10 (192.168.112.166)
localhost:9505/?a=Desktop
Recycle Bin
Control Panel
File Manager
SMS
SMS Gateway

Figure 8 - Opening the SMS Gateway App

Click on **'Add new connection'** on the left.

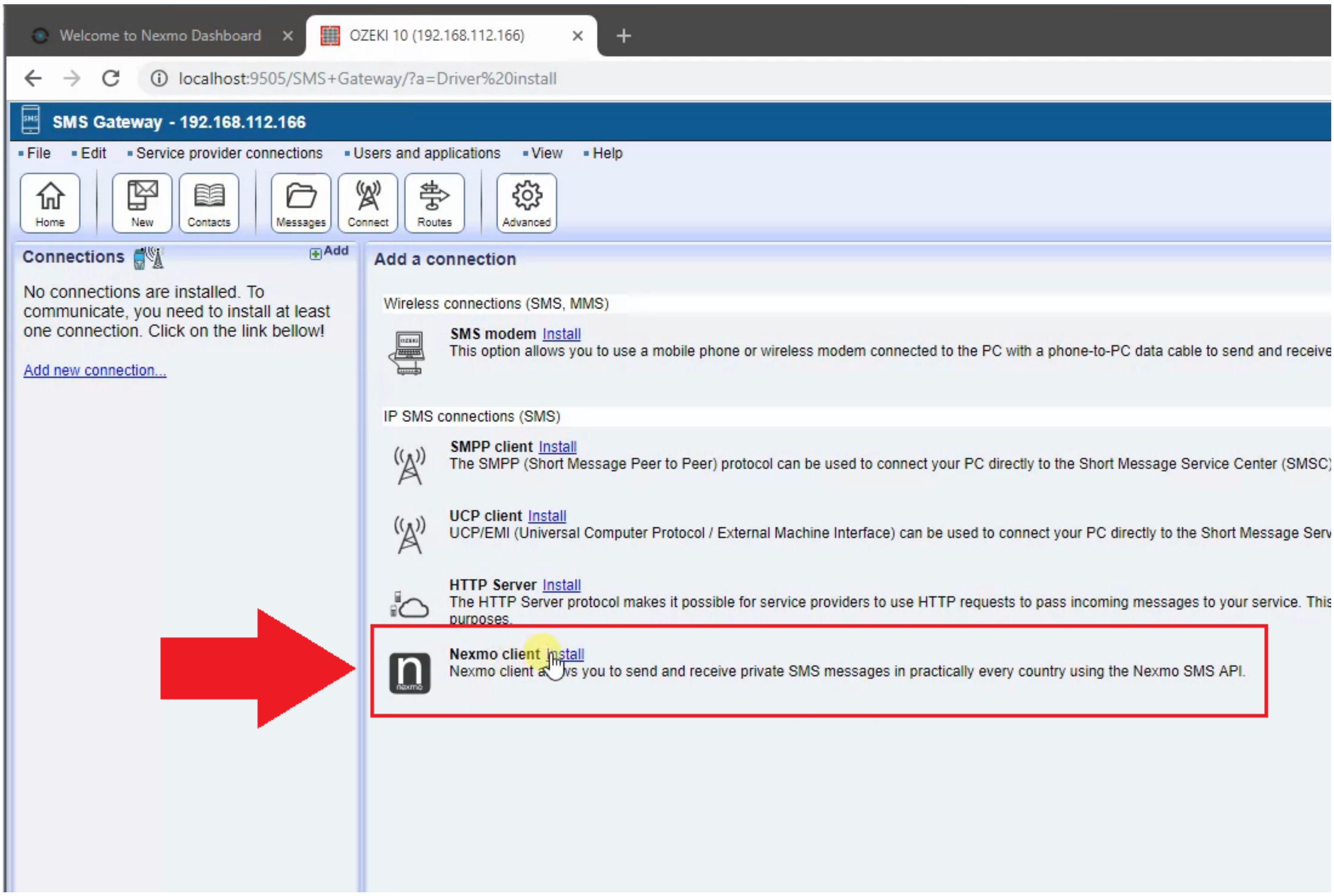


No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

Figure 9 - Creating a new connection

Select the **'Nexmo Connection'** from the list and click on install next to it.


Welcome to Nexmo Dashboard
OZEKI 10 (192.168.112.166)
localhost:9505/SMS+Gateway/?a=Driver%20install
SMS Gateway - 192.168.112.166
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Contacts
Messages
Connect
Routes
Advanced
Add

Connections

No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

Add a connection

Wireless connections (SMS, MMS)

**SMS modem** Install
This option allows you to use a mobile phone or wireless modem connected to the PC with a phone-to-PC data cable to send and receive

IP SMS connections (SMS)

**SMPP client** Install
The SMPP (Short Message Peer to Peer) protocol can be used to connect your PC directly to the Short Message Service Center (SMSC)

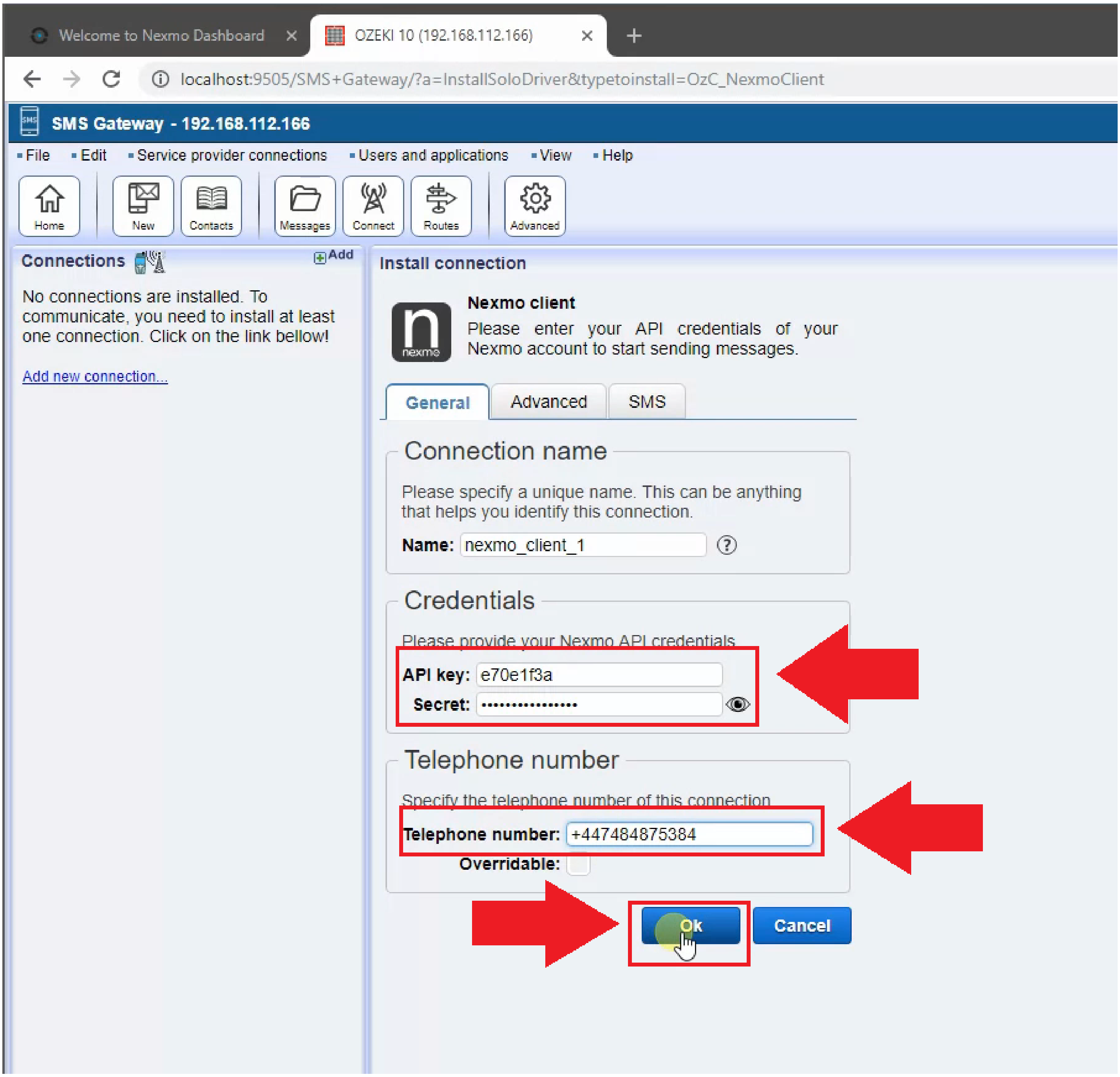
**UCP client** Install
UCP/EMI (Universal Computer Protocol / External Machine Interface) can be used to connect your PC directly to the Short Message Serv

**HTTP Server** Install
The HTTP Server protocol makes it possible for service providers to use HTTP requests to pass incoming messages to your service. This purposes.

**Nexmo client** Install
Nexmo client allows you to send and receive private SMS messages in practically every country using the Nexmo SMS API.

Figure 10 - Installing the Nexmo connection

Provide your Nexmo API credentials and specify telephone number on the Install connection page.


Welcome to Nexmo Dashboard
OZEKI 10 (192.168.112.166)
localhost:9505/SMS+Gateway/?a=InstallSoloDriver&typetoinstall=OzC_NexmoClient
SMS Gateway - 192.168.112.166
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Contacts
Messages
Connect
Routes
Advanced
Connections
Add
Install connection
General
Advanced
SMS
Ok
Cancel

No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

Nexmo client

Please enter your API credentials of your Nexmo account to start sending messages.

Connection name

Please specify a unique name. This can be anything that helps you identify this connection.

Name: nexmo_client_1

Credentials

Please provide your Nexmo API credentials

API key: e70e1f3a

Secret: ••••••••••••••••

Telephone number

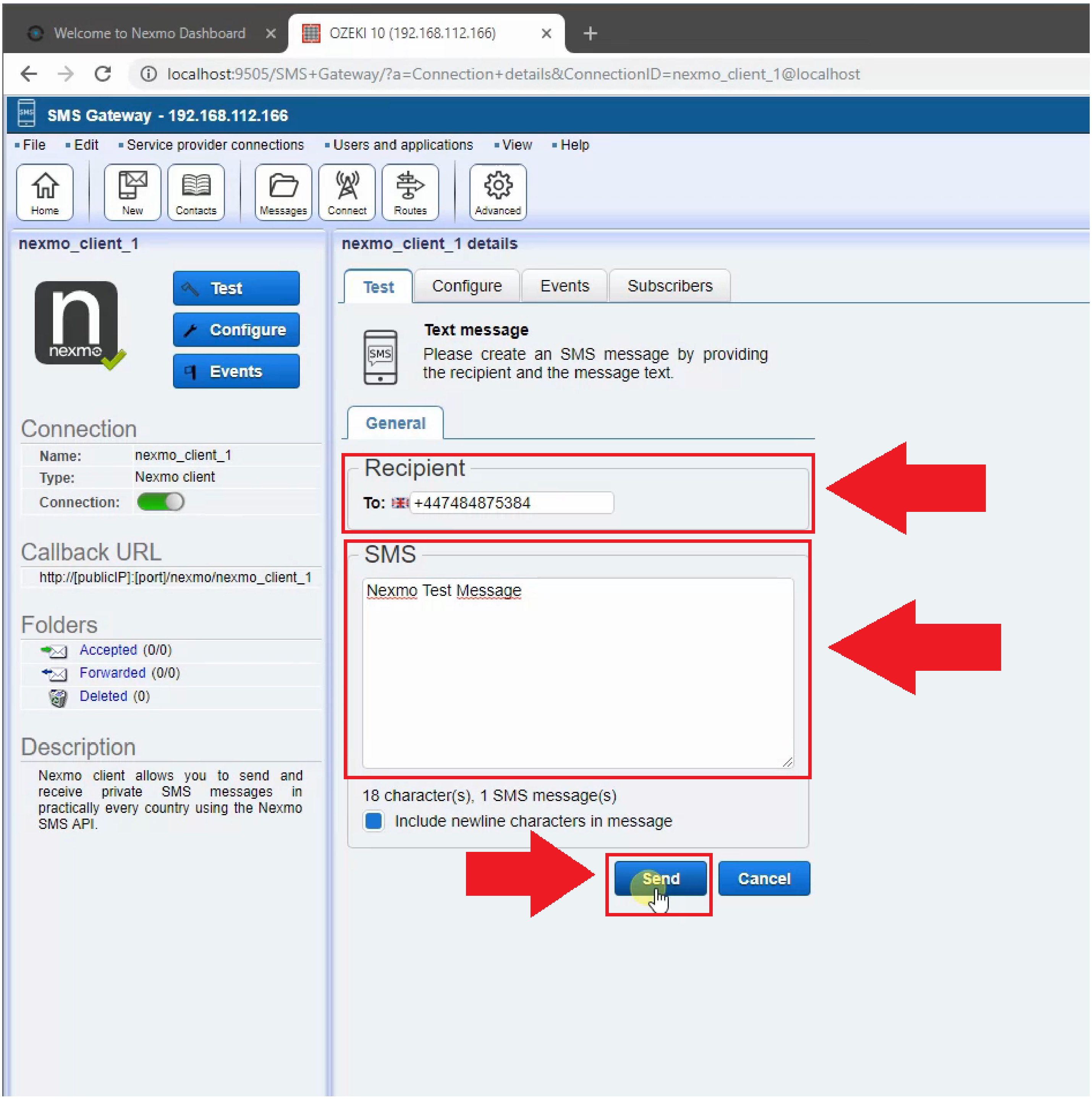
Specify the telephone number of this connection

Telephone number: +447484875384

Overridable: □

Figure 11 - Providing Nexmo credentials

Now it is time to send a test message (see **Figure 12**).



Text message

Please create an SMS message by providing the recipient and the message text.

To: +447484875384

Nexmo Test Message

18 character(s), 1 SMS message(s)

☑ Include newline characters in message

Name: nexmo_client_1

Type: Nexmo client

Connection: ●

http://[publicIP]:[port]/nexmo/nexmo_client_1

Nexmo client allows you to send and receive private SMS messages in practically every country using the Nexmo SMS API.

Figure 12 - Sending a test message

# How to setup Ozeki 10 and Nexmo for incoming delivery reports

Before you can receive incoming delivery reports and inbound SMS, you need to forward ports in order to route the external traffic to your network.

1. Open a Web browser and type your Default Gateway number into the address bar then press Enter.

2. Enter your username and password to access the interface of your router. The default username and password should be listed in the documentation of your router or on a sticker on the side of your router. If the default username and password have been changed and you do not remember them, you will need to reset your router.

3. To forward ports on your router, look for a tab or menu named 'Port Forwarding/Port Triggering' or something similar.

4. No matter what type of router or interface you have, you will need to enter the same basic information. Enter the port you want to open under External (Port 1) and Internal (Port 2), or enter a range of ports to open under Start and End.
In this case Ozeki 10 uses port 9505 by default so you may also provide that in the 'Internal' section.

5. Select the Protocol (TCP, UDP, or both).

6. Enter the Private IP address of your PC where your Ozeki 10 installed.

7. Be sure you saved the changes.

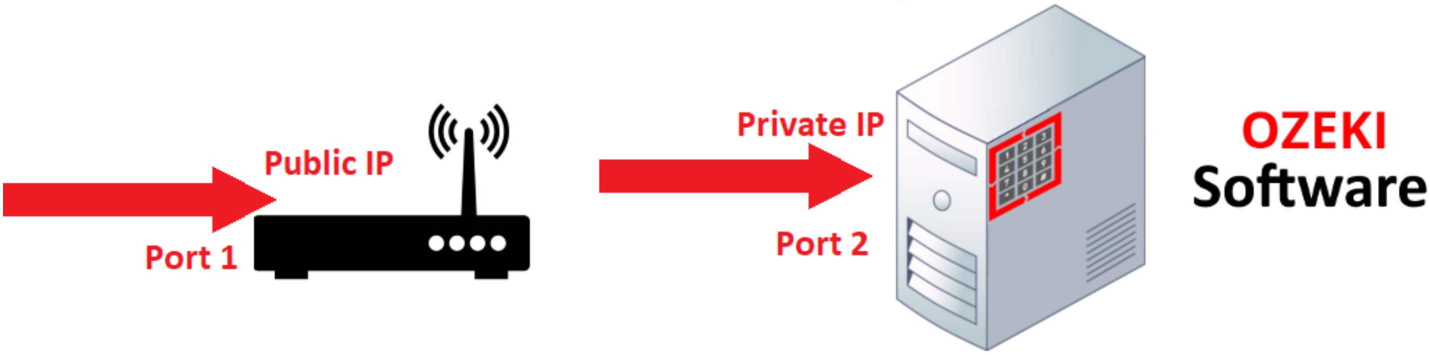
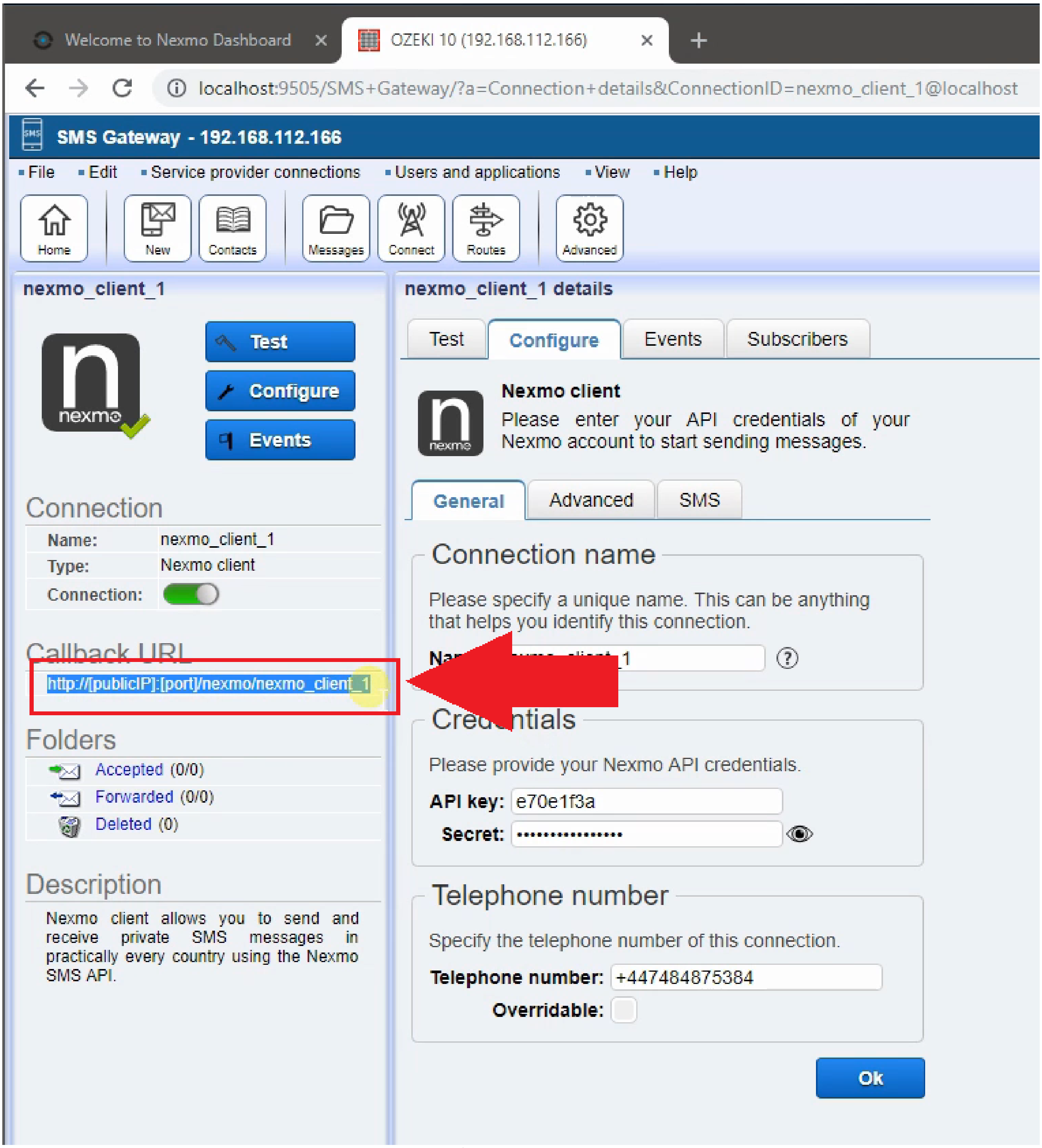


Figure 13 - Port Forwarding

After having the changes saved, please copy the callback URL of the Nexmo connection which you will find on the left.


Welcome to Nexmo Dashboard
OZEKI 10 (192.168.112.166)
localhost:9505/SMS+Gateway/?a=Connection+details&ConnectionID=nexmo_client_1@localhost
SMS Gateway - 192.168.112.166
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Contacts
Messages
Connect
Routes
Advanced
nexmo_client_1
Test
Configure
Events
nexmo_client_1 details
Test
Configure
Events
Subscribers
General
Advanced
SMS
Ok

Nexmo client

Please enter your API credentials of your Nexmo account to start sending messages.

Connection

Name: nexmo_client_1

Type: Nexmo client

Connection:

Callback URL

http://[publicIP]:[port]/nexmo/nexmo_client_1

Folders

Accepted (0/0)

Forwarded (0/0)

Deleted (0)

Description

Nexmo client allows you to send and receive private SMS messages in practically every country using the Nexmo SMS API.

Connection name

Please specify a unique name. This can be anything that helps you identify this connection.

Name: nexmo_client_1

Credentials

Please provide your Nexmo API credentials.

API key: e70e1f3a

Secret: ••••••••••••••••

Telephone number

Specify the telephone number of this connection.

Telephone number: +447484875384

Overridable: □

Figure 14 - Copying the callback URL

Now we will enter the above in the Nexmo settings.

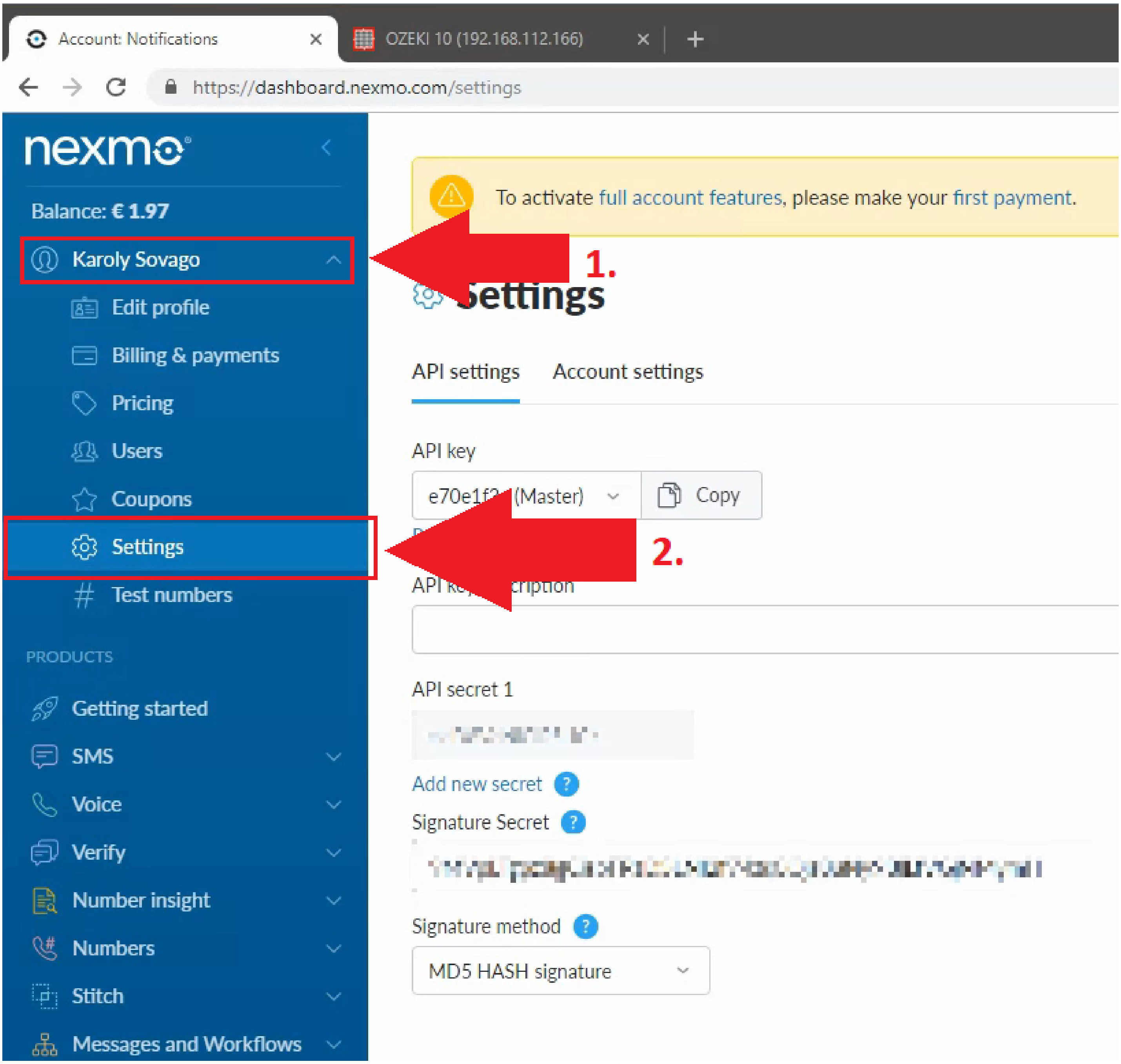
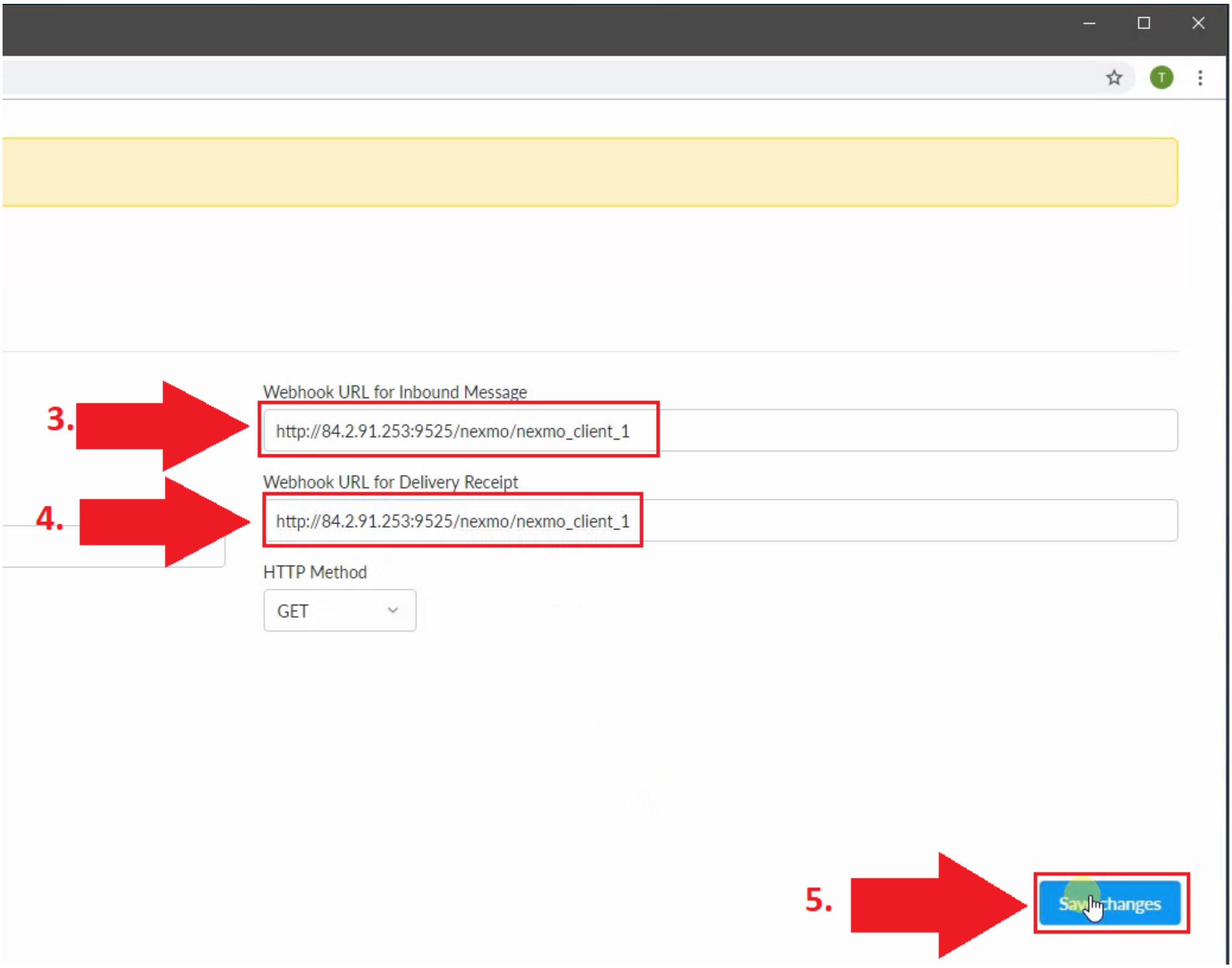


Figure 15 - Nexmo Settings

Enter your Pubblic IP and the port you forwarded to the Ozeki 10 PC then save them.



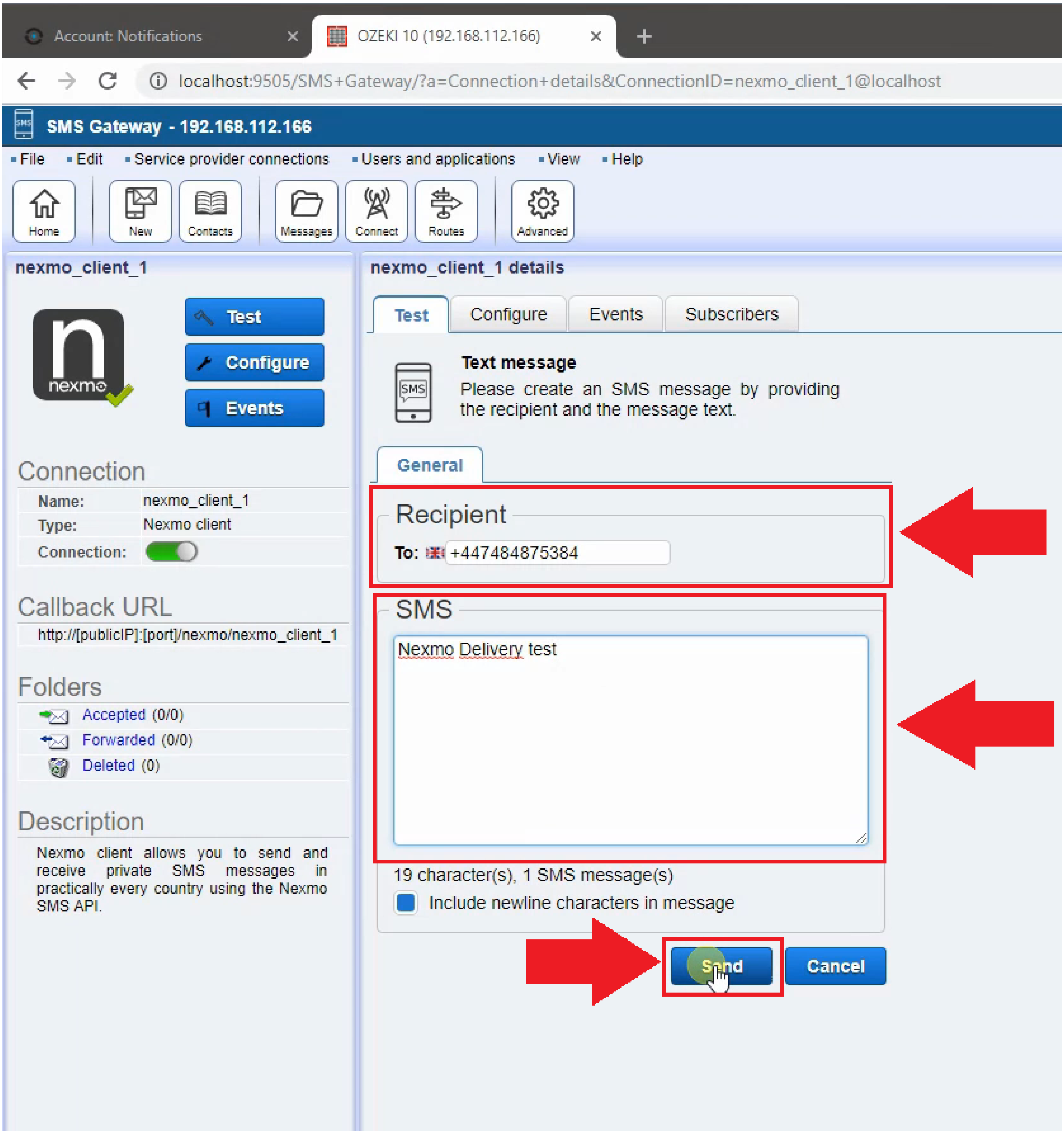
3. Webhook URL for Inbound Message: http://84.2.91.253:9525/nexmo/nexmo_client_1

4. Webhook URL for Delivery Receipt: http://84.2.91.253:9525/nexmo/nexmo_client_1

HTTP Method: GET

Figure 16 - Nexmo Settings

We are ready to send a Test Message as seen in **Figure 17**.


Account: Notifications
OZEKI 10 (192.168.112.166)
localhost:9505/SMS+Gateway/?a=Connection+details&ConnectionID=nexmo_client_1@localhost
SMS Gateway - 192.168.112.166
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Contacts
Messages
Connect
Routes
Advanced
Test
Configure
Events
Subscribers
General
Send
Cancel

nexmo_client_1

Connection

Name: nexmo_client_1

Type: Nexmo client

Connection:

Callback URL

http://[publicIP]:[port]/nexmo/nexmo_client_1

Folders

Accepted (0/0)

Forwarded (0/0)

Deleted (0)

Description

Nexmo client allows you to send and receive private SMS messages in practically every country using the Nexmo SMS API.

nexmo_client_1 details

Text message

Please create an SMS message by providing the recipient and the message text.

Recipient

To: +447484875384

SMS

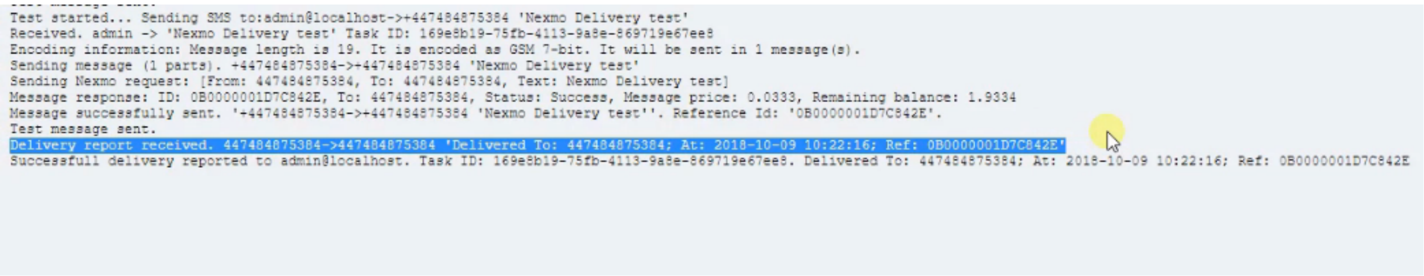
Nexmo Delivery test

19 character(s), 1 SMS message(s)

☐ Include newline characters in message

Figure 17 - Sending a test SMS

If you have forwarded the port properly you will see the delivery report received.



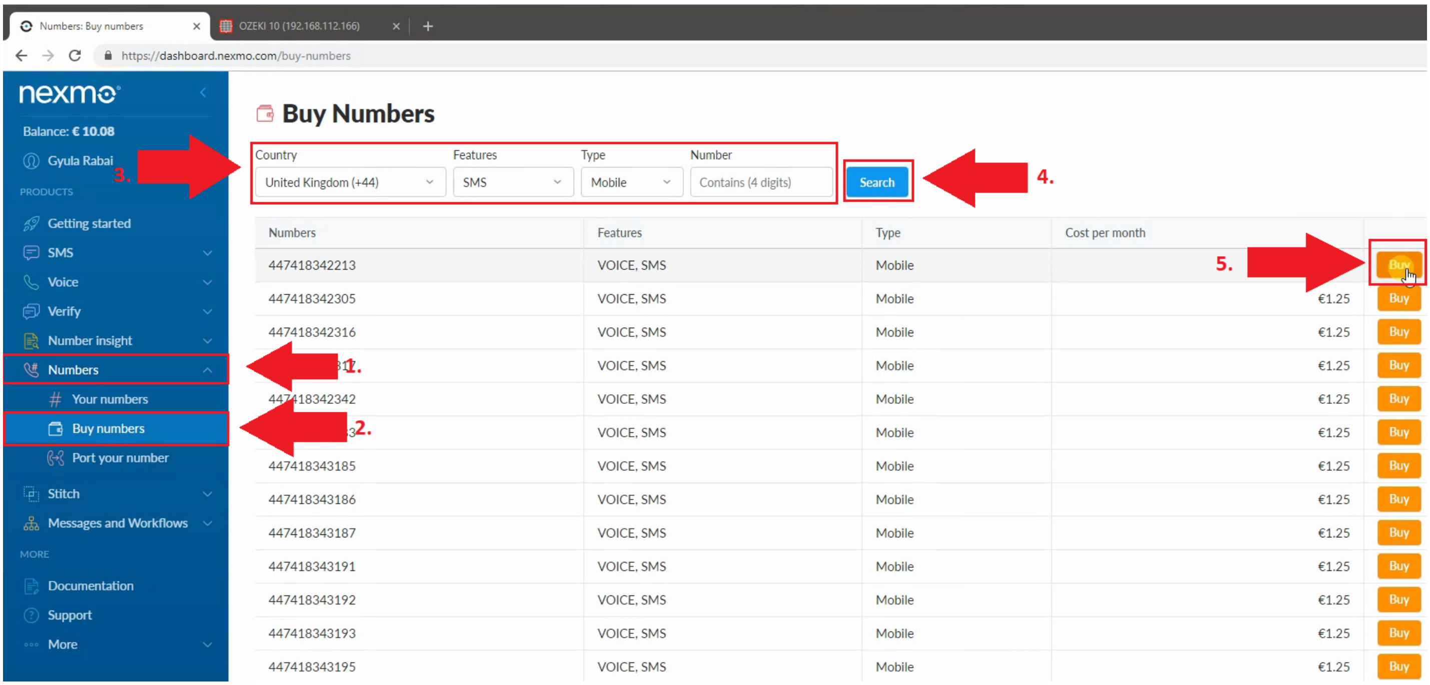
```
Test started... Sending SMS to:admin@localhost->+447484875384 'Nexmo Delivery test'
Received. admin -> 'Nexmo Delivery test' Task ID: 169e8b19-75fb-4113-9a8e-869719e67ee8
Encoding information: Message length is 19. It is encoded as GSM 7-bit. It will be sent in 1 message(s).
Sending message (1 parts). +447484875384->+447484875384 'Nexmo Delivery test'
Sending Nexmo request: [From: 447484875384, To: 447484875384, Text: Nexmo Delivery test]
Message response: ID: 0B0000001D7C842E, To: 447484875384, Status: Success, Message price: 0.0333, Remaining balance: 1.9334
Message successfully sent. '+447484875384->+447484875384 'Nexmo Delivery test''. Reference Id: '0B0000001D7C842E'.
Test message sent.
Delivery report received. 447484875384->447484875384 'Delivered To: 447484875384; At: 2018-10-09 10:22:16; Ref: 0B0000001D7C842E'
Successfull delivery reported to admin@localhost. Task ID: 169e8b19-75fb-4113-9a8e-869719e67ee8. Delivered To: 447484875384; At: 2018-10-09 10:22:16; Ref: 0B0000001D7C842E
```

Figure 18 - Delivery report received

## How to buy a Nexmo SMS number

In the below section we will demonstrate how to buy Nexmo number to receive SMS messages.

Please navigate to the **'Buy numbers'** menu, select a number and click on the buy button next to it.



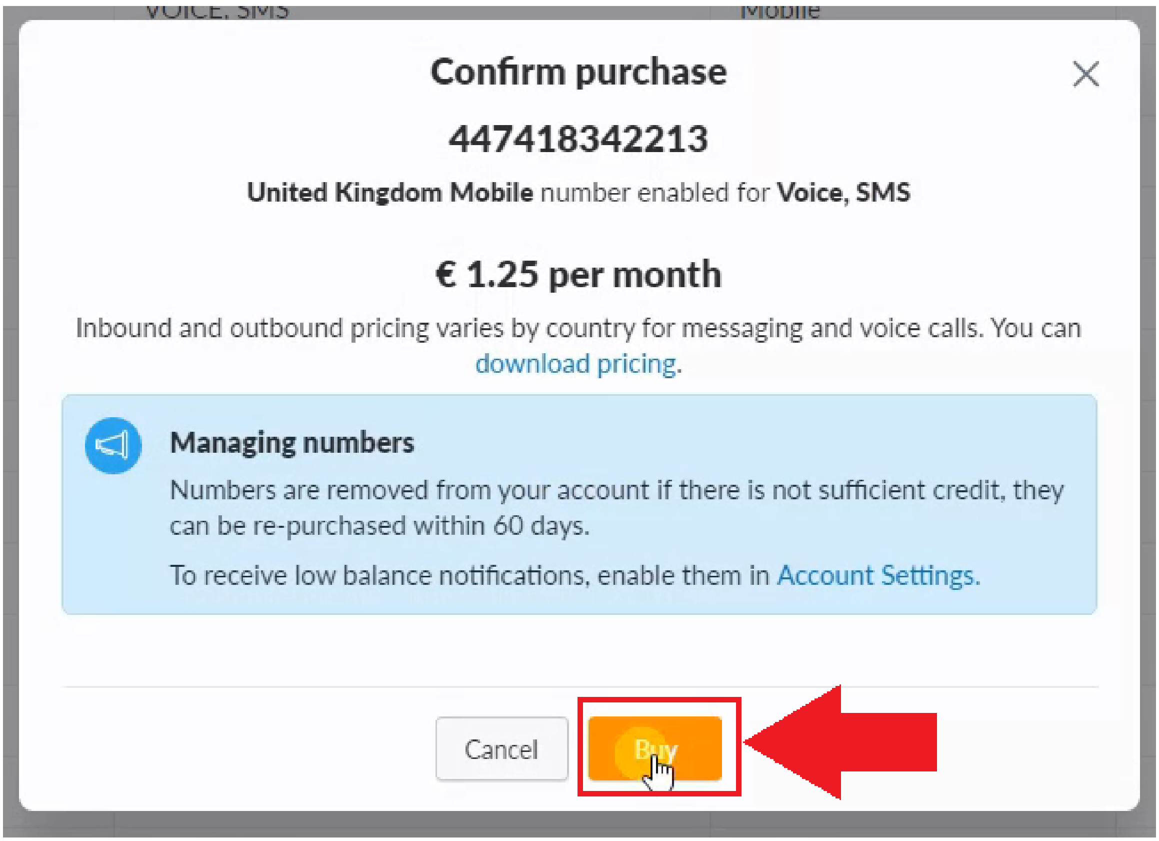
Buy Numbers

Country: United Kingdom (+44)
Features: SMS
Type: Mobile
Number: Contains (4 digits)

| Numbers | Features | Type | Cost per month | |
|---|---|---|---|---|
| 447418342213 | VOICE, SMS | Mobile | | Buy |
| 447418342305 | VOICE, SMS | Mobile | €1.25 | Buy |
| 447418342316 | VOICE, SMS | Mobile | €1.25 | Buy |
| [illegible] | VOICE, SMS | Mobile | €1.25 | Buy |
| 447418342342 | VOICE, SMS | Mobile | €1.25 | Buy |
| [illegible] | VOICE, SMS | Mobile | €1.25 | Buy |
| 447418343185 | VOICE, SMS | Mobile | €1.25 | Buy |
| 447418343186 | VOICE, SMS | Mobile | €1.25 | Buy |
| 447418343187 | VOICE, SMS | Mobile | €1.25 | Buy |
| 447418343191 | VOICE, SMS | Mobile | €1.25 | Buy |
| 447418343192 | VOICE, SMS | Mobile | €1.25 | Buy |
| 447418343193 | VOICE, SMS | Mobile | €1.25 | Buy |
| 447418343195 | VOICE, SMS | Mobile | €1.25 | Buy |

Figure 19 - Selecting a number

After having the number selected, you need to confirm the puchase.



Confirm purchase

447418342213

**United Kingdom Mobile** number enabled for **Voice, SMS**

**€ 1.25 per month**

Inbound and outbound pricing varies by country for messaging and voice calls. You can download pricing.

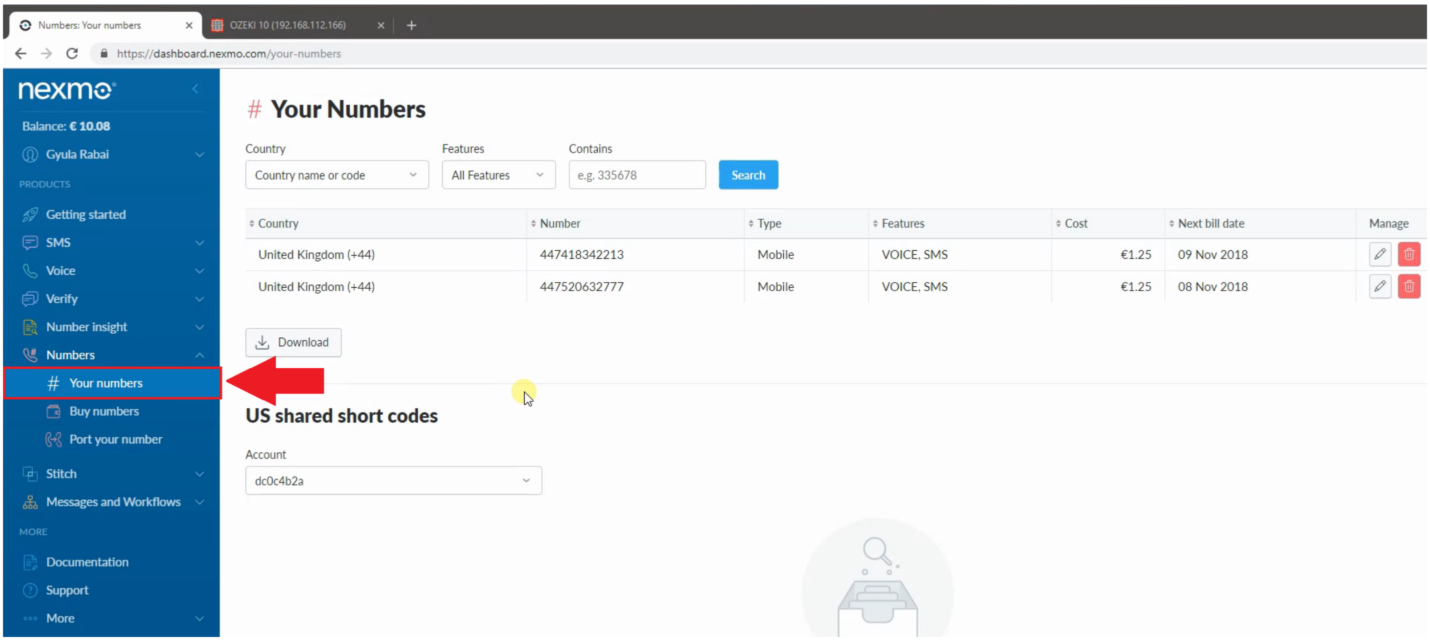
**Managing numbers**

Numbers are removed from your account if there is not sufficient credit, they can be re-purchased within 60 days.

To receive low balance notifications, enable them in Account Settings.

Figure 20 - Confirming the purchase

In the **'Your numbers'** menu you can see your own Nexmo phone numbers.



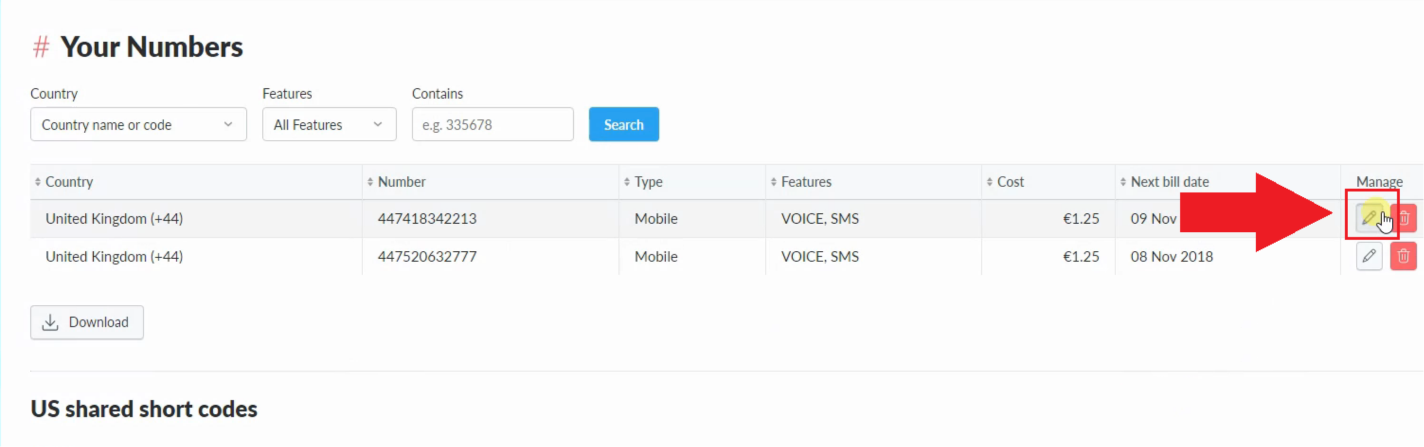
| Country | Number | Type | Features | Cost | Next bill date | Manage |
|---|---|---|---|---|---|---|
| United Kingdom (+44) | 447418342213 | Mobile | VOICE, SMS | €1.25 | 09 Nov 2018 | |
| United Kingdom (+44) | 447520632777 | Mobile | VOICE, SMS | €1.25 | 08 Nov 2018 | |

Figure 21 - Viewing your numbers

# How to setup Ozeki 10 and Nexmo for incoming SMS messages

After you have purchased a Nexmo number you can set it up to forward the messages to the Ozeki 10 machine.

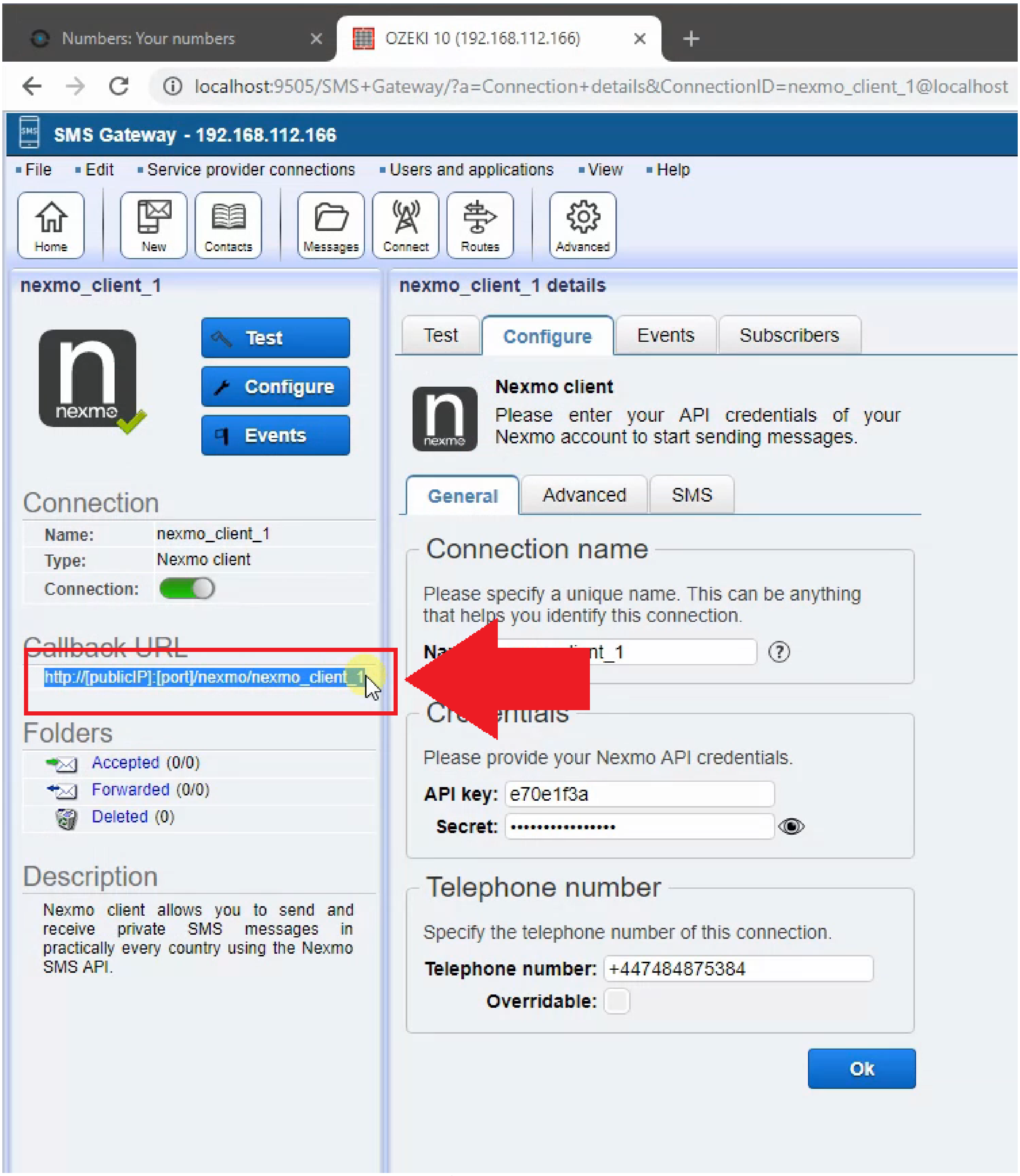
In the **'Your Numbers'** menu click 'Manage' to edit the number you would like to receive SMS messages to.



| Country | Number | Type | Features | Cost | Next bill date | Manage |
|---|---|---|---|---|---|---|
| United Kingdom (+44) | 447418342213 | Mobile | VOICE, SMS | €1.25 | 09 Nov | |
| United Kingdom (+44) | 447520632777 | Mobile | VOICE, SMS | €1.25 | 08 Nov 2018 | |

Figure 22 - Editing your number

Copy the callback URL from the Nexmo connection.



Connection

Name: nexmo_client_1

Type: Nexmo client

Connection:

Callback URL

http://[publicIP]:[port]/nexmo/nexmo_client_1

Folders

Accepted (0/0)

Forwarded (0/0)

Deleted (0)

Description

Nexmo client allows you to send and receive private SMS messages in practically every country using the Nexmo SMS API.

Nexmo client

Please enter your API credentials of your Nexmo account to start sending messages.

Connection name

Please specify a unique name. This can be anything that helps you identify this connection.

Name: nexmo_client_1

Credentials

Please provide your Nexmo API credentials.

API key: e70e1f3a

Secret: ••••••••••••••••

Telephone number

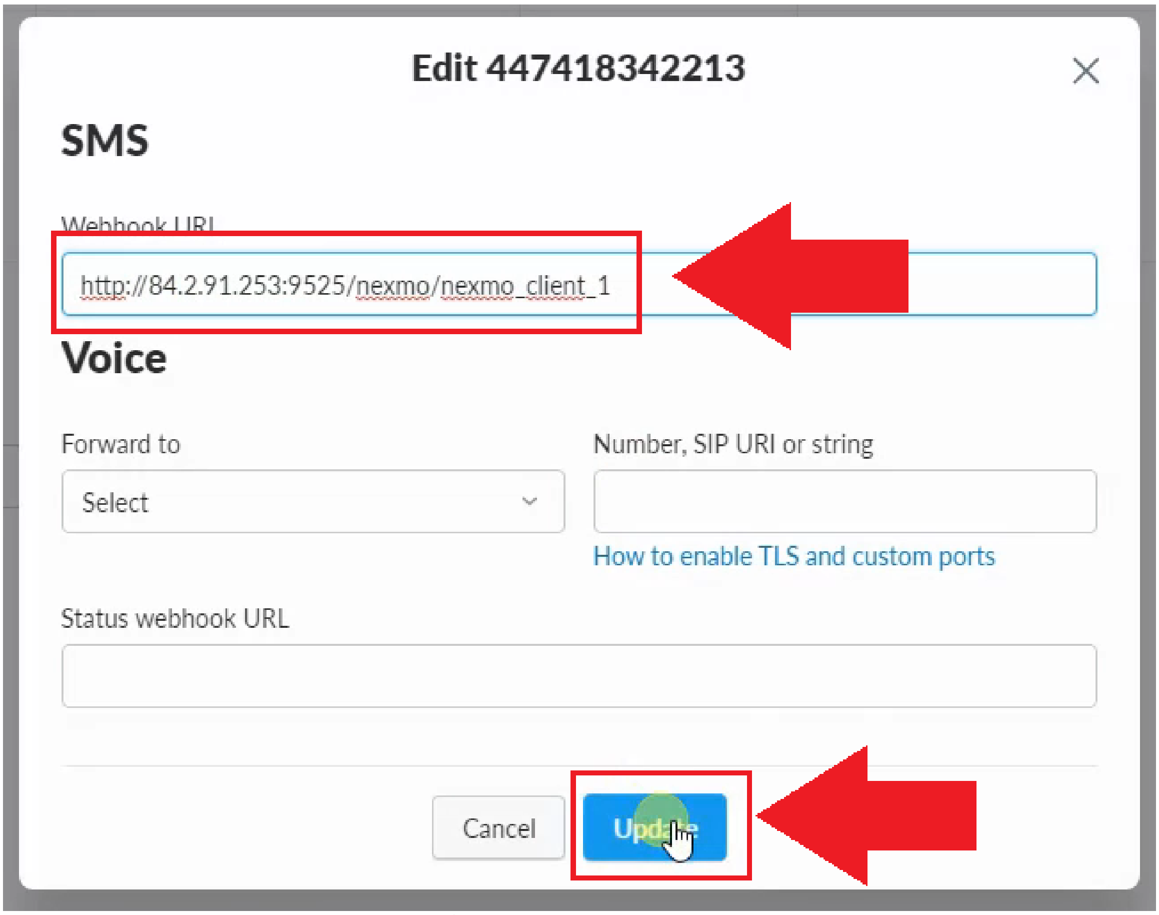
Specify the telephone number of this connection.

Telephone number: +447484875384

Overridable: □

Figure 23 - Copying the callback URL

Enter your Pubblic IP and the port you forwarded to the Ozeki 10 PC then click 'Update'.



Edit 447418342213

SMS

Webhook URL: http://84.2.91.253:9525/nexmo/nexmo_client_1

Voice

Forward to: Select

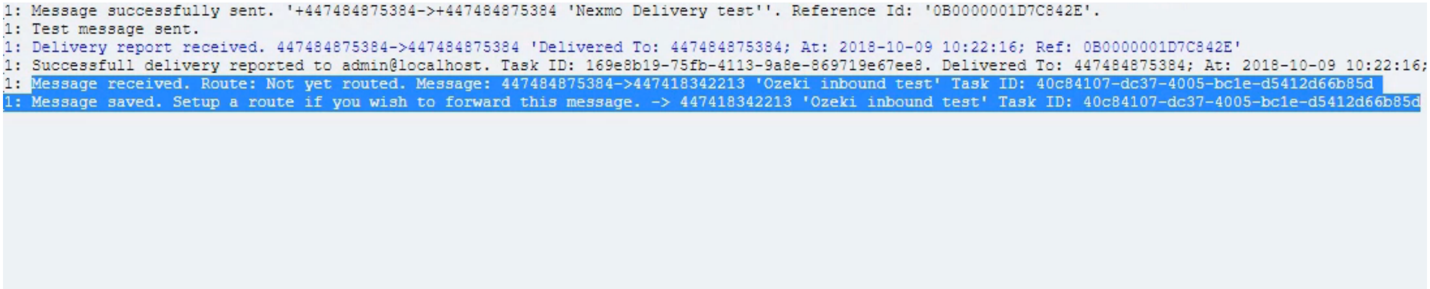
Number, SIP URI or string:

How to enable TLS and custom ports

Status webhook URL:

Figure 24 - Providing webhook URL

Send a Test Message to this number and you will see the message received in Ozeki 10.



```
1: Message successfully sent. '+447484875384->+447484875384 'Nexmo Delivery test''. Reference Id: '0B0000001D7C842E'.
1: Test message sent.
1: Delivery report received. 447484875384->447484875384 'Delivered To: 447484875384; At: 2018-10-09 10:22:16; Ref: 0B0000001D7C842E'
1: Successfull delivery reported to admin@localhost. Task ID: 169e8b19-75fb-4113-9a8e-869719e67ee8. Delivered To: 447484875384; At: 2018-10-09 10:22:16;
1: Message received. Route: Not yet routed. Message: 447484875384->447418342213 'Ozeki inbound test' Task ID: 40c84107-dc37-4005-bc1e-d5412d66b85d
1: Message saved. Setup a route if you wish to forward this message. -> 447418342213 'Ozeki inbound test' Task ID: 40c84107-dc37-4005-bc1e-d5412d66b85d
```

Figure 25 - Test message received

# MessageBird SMS service provider

## How to create a MessageBird account

In order to use the MessageBird service you need to create an account. In this section we guide you how to do it.

On the main page please select the '**Sign up**' button.

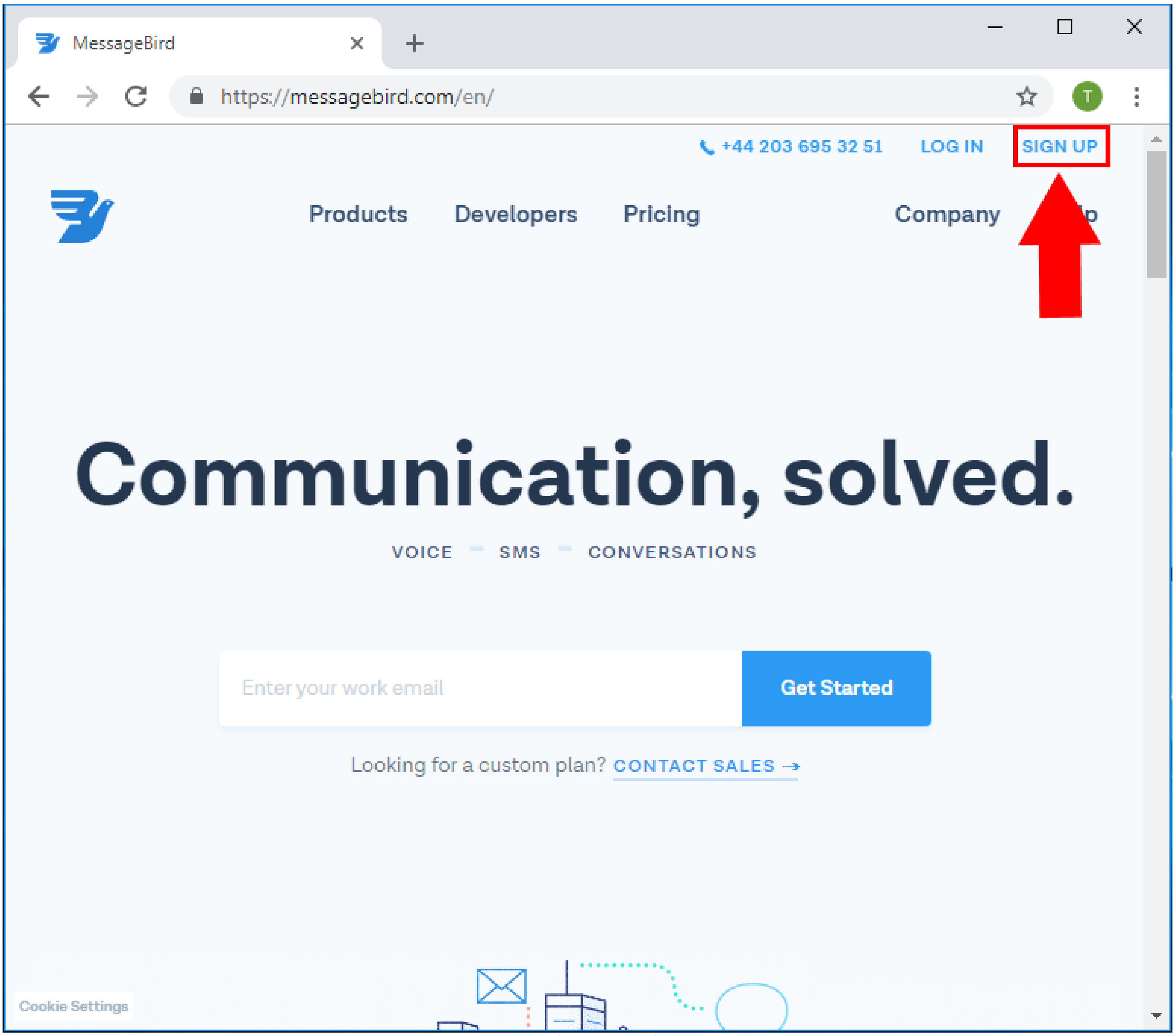
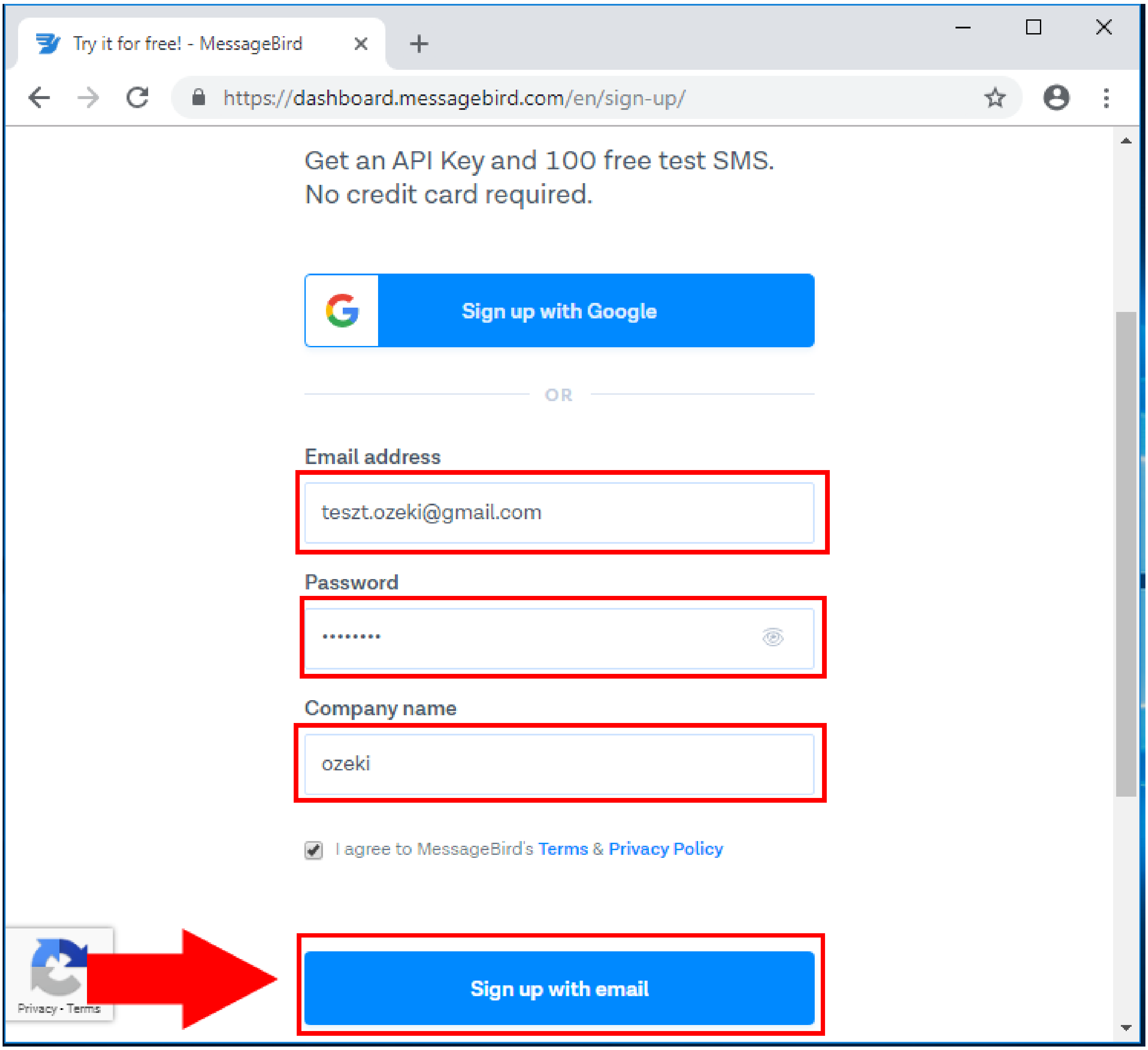


Figure 1 - Clicking **'Sign up'**

Then provide the login details and click '**Sign up with email**' button.



Try it for free! - MessageBird

https://dashboard.messagebird.com/en/sign-up/

Get an API Key and 100 free test SMS.
No credit card required.

Email address: teszt.ozeki@gmail.com

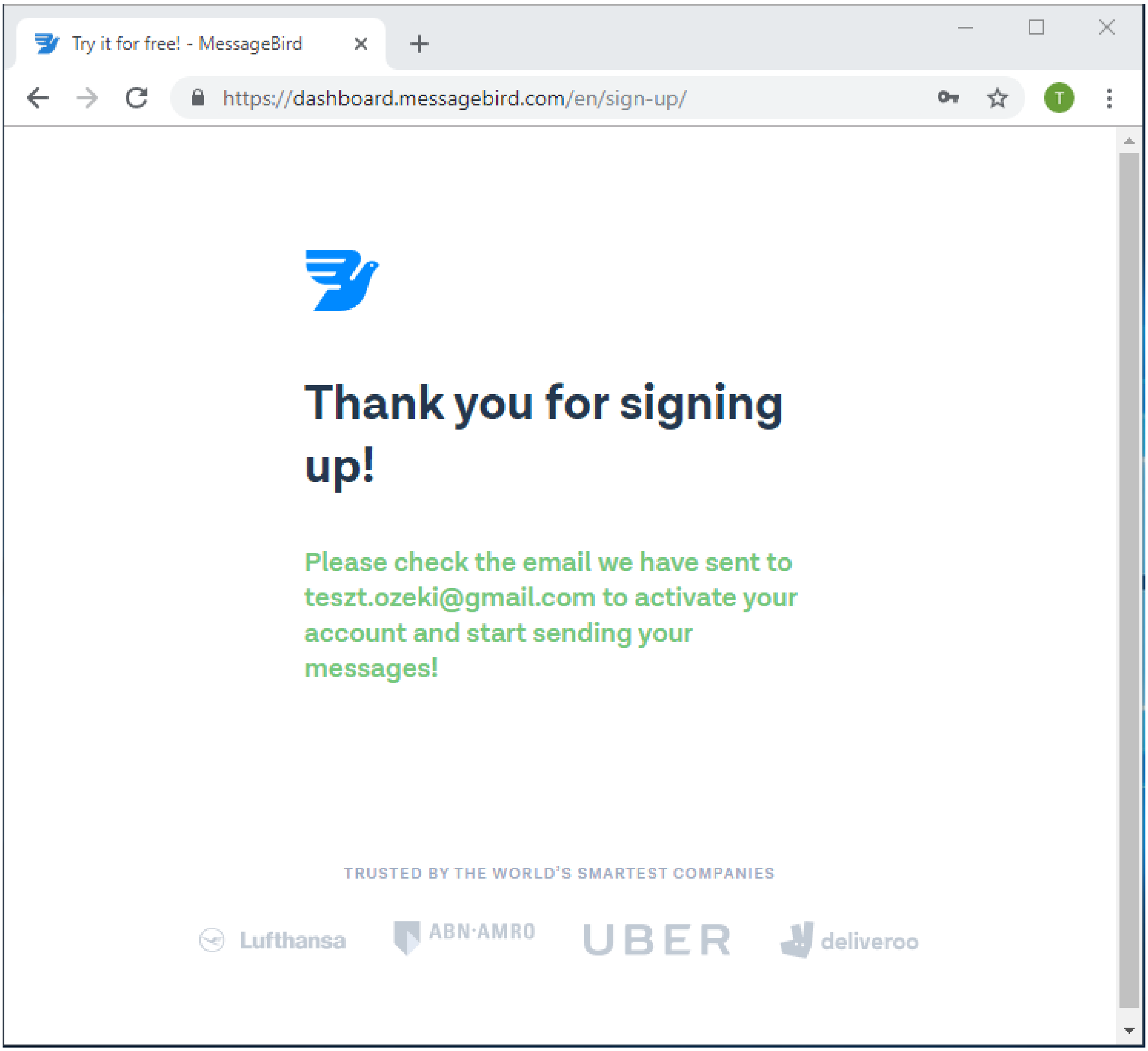
Password: ••••••••

Company name: ozeki

☑ I agree to MessageBird's Terms & Privacy Policy

Figure 2 - Provide account details

Check your email and activate your account.



Try it for free! - MessageBird
https://dashboard.messagebird.com/en/sign-up/

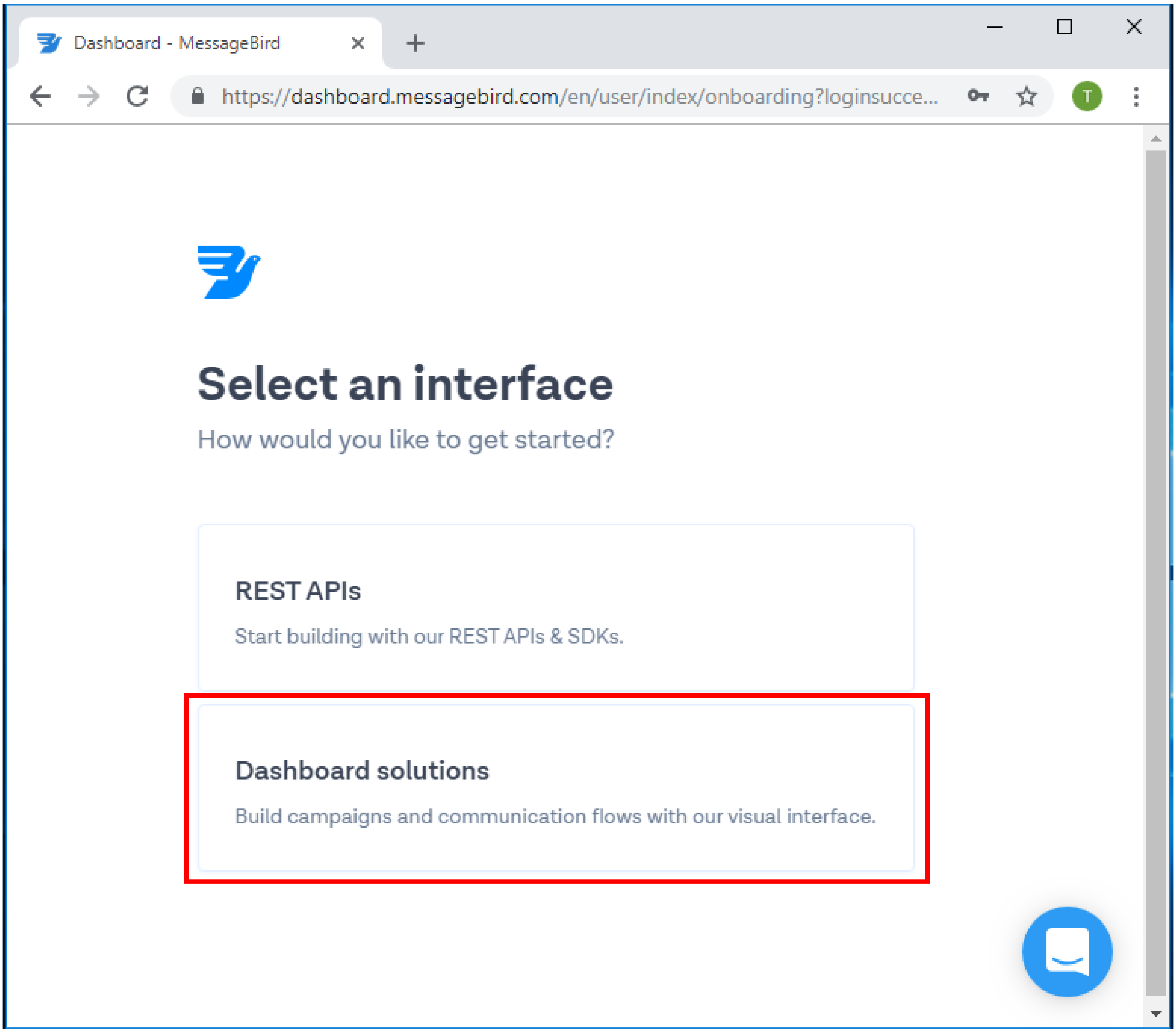
Thank you for signing up!

Please check the email we have sent to teszt.ozeki@gmail.com to activate your account and start sending your messages!

TRUSTED BY THE WORLD'S SMARTEST COMPANIES
Lufthansa
ABN·AMRO
UBER
deliveroo

Figure 3 - Successful registration

After the login select the "**Dashboard solutions**" interface.



Dashboard - MessageBird
https://dashboard.messagebird.com/en/user/index/onboarding?loginsucce...

Select an interface

How would you like to get started?

REST APIs

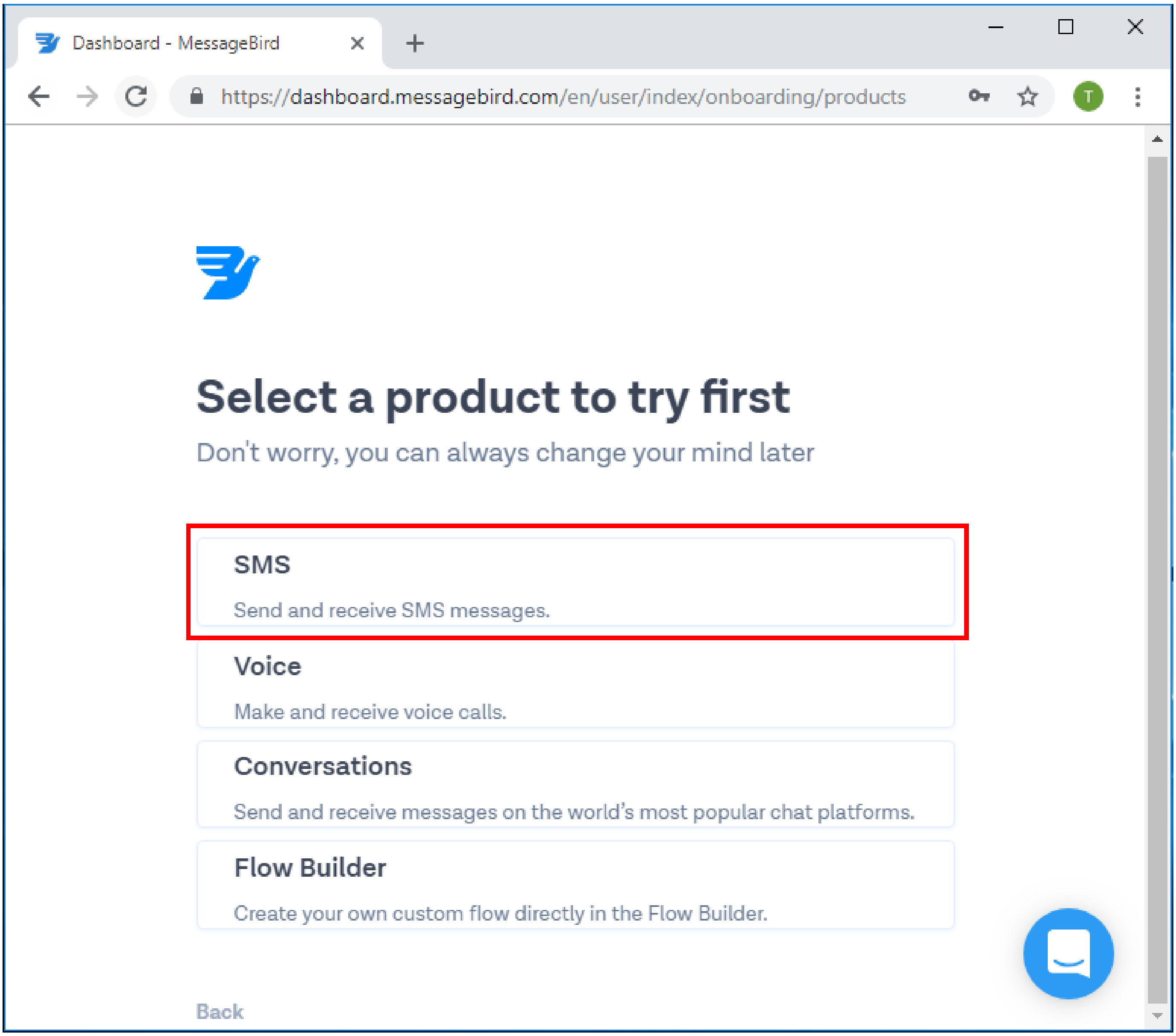
Start building with our REST APIs & SDKs.

Dashboard solutions

Build campaigns and communication flows with our visual interface.

Figure 4 - Select Dashbord solutins

And select **SMS** product.



Dashboard - MessageBird
https://dashboard.messagebird.com/en/user/index/onboarding/products

Select a product to try first

Don't worry, you can always change your mind later

SMS

Send and receive SMS messages.

Voice

Make and receive voice calls.

Conversations

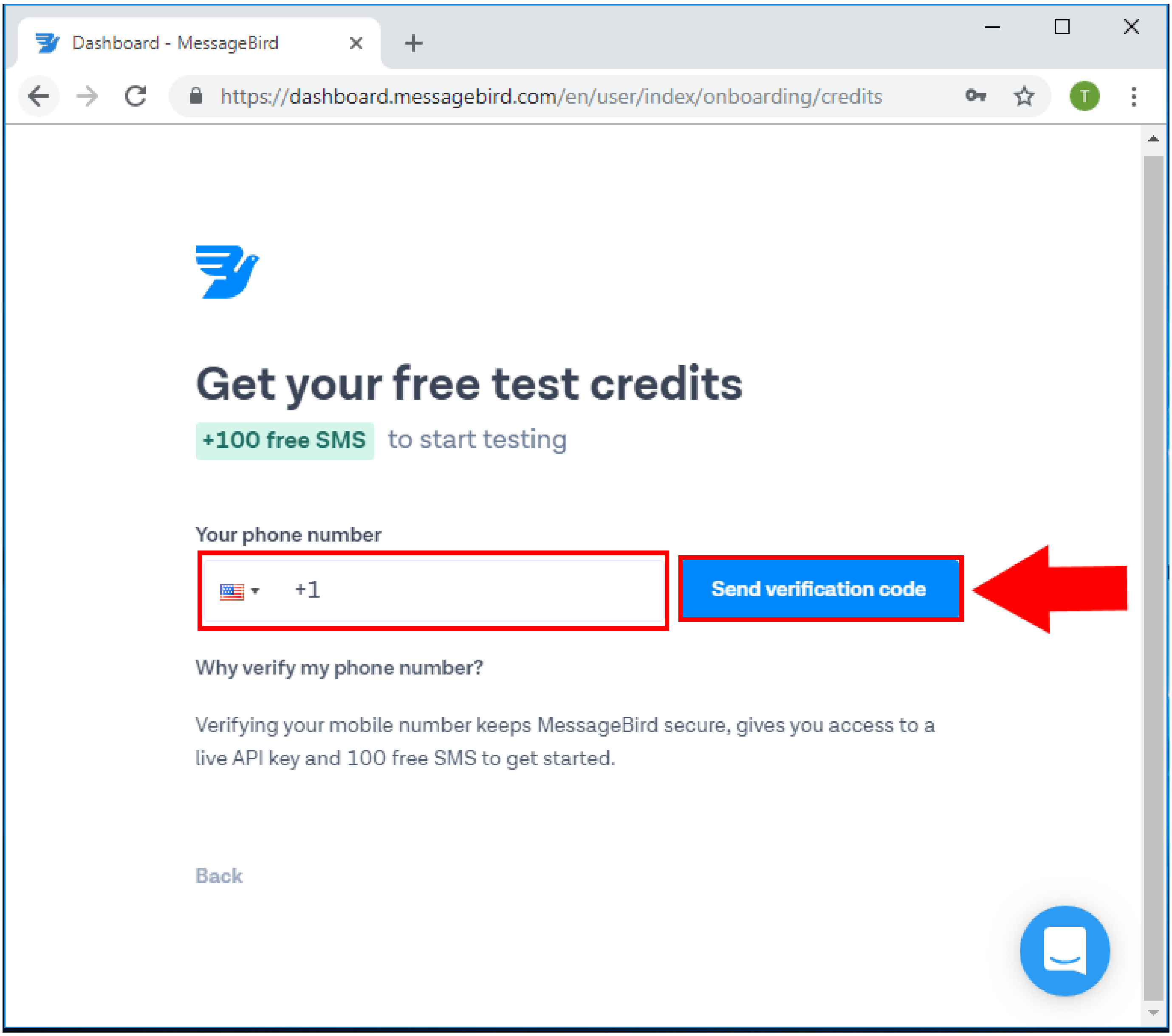
Send and receive messages on the world's most popular chat platforms.

Flow Builder

Create your own custom flow directly in the Flow Builder.

Figure 5 - Choose SMS

Send verification code to your phone number.


Dashboard - MessageBird
https://dashboard.messagebird.com/en/user/index/onboarding/credits

Get your free test credits

+100 free SMS to start testing

Your phone number

+1

Why verify my phone number?

Verifying your mobile number keeps MessageBird secure, gives you access to a live API key and 100 free SMS to get started.

Figure 6 - Send verification code

After the verification, you're all set up.

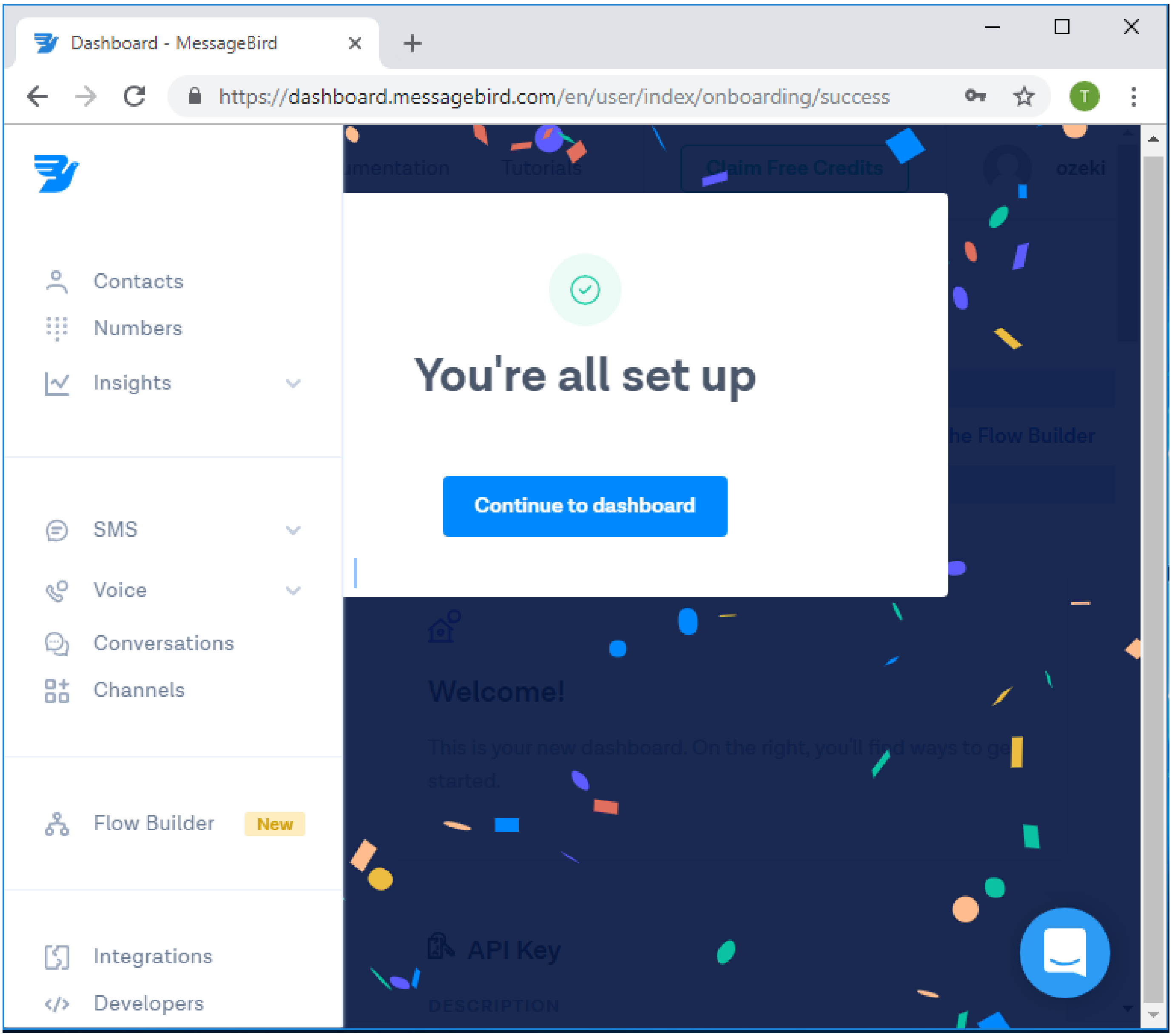
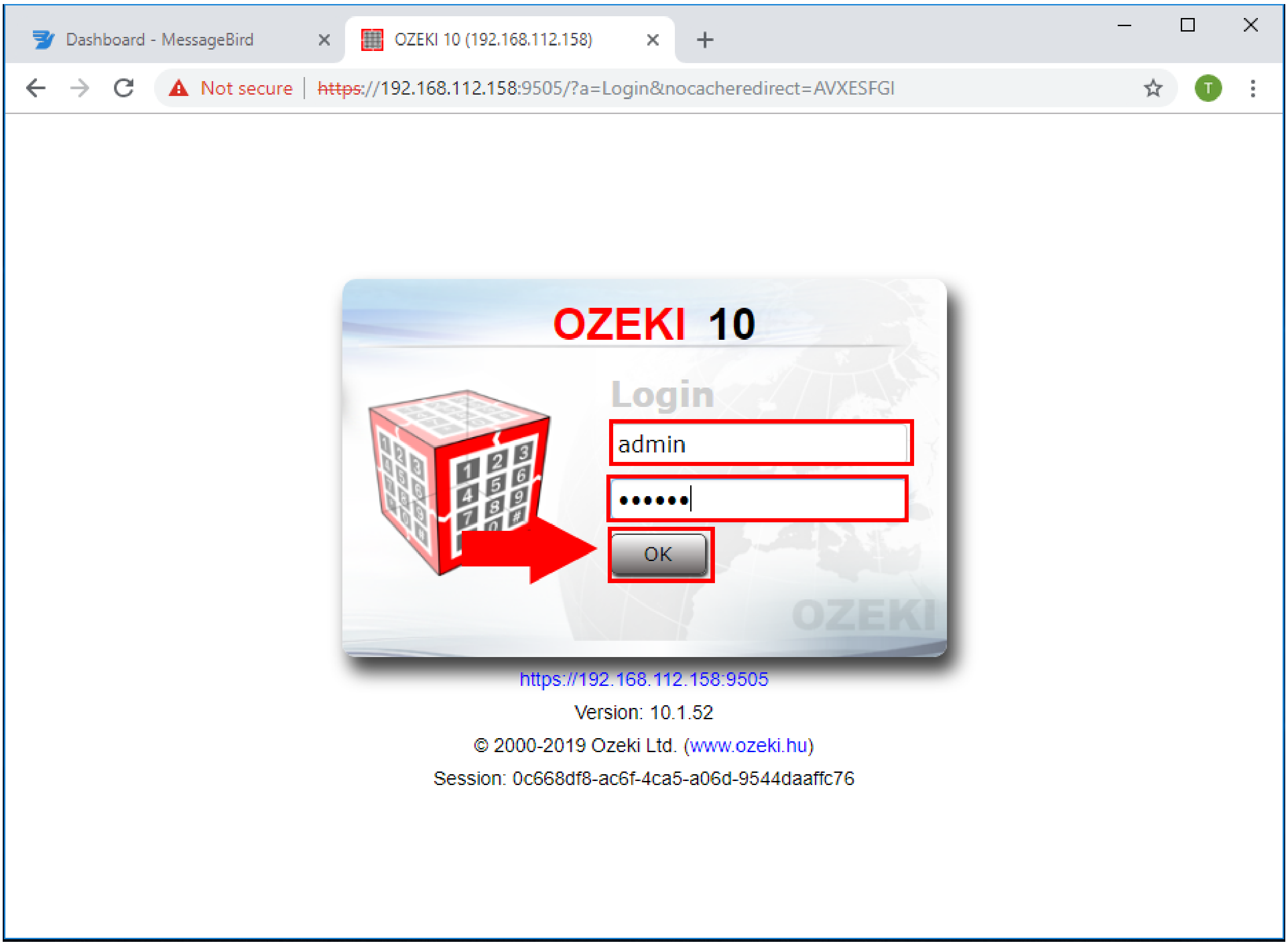


Figure 7 - Successful Log in

# How to setup Ozeki 10 and MessageBird for outbound SMS messages

This chapter provides you detailed information on how to configure MessageBird connection for SMS sending in the Ozeki 10 SMS Gateway software.

At first, please Log in to Ozeki 10 with your username and password.


Dashboard - MessageBird
OZEKI 10 (192.168.112.158)
Not secure
https://192.168.112.158:9505/?a=Login&nocacheredirect=AVXESFGI
OZEKI 10
Login
OK
OZEKI

admin

••••••

https://192.168.112.158:9505

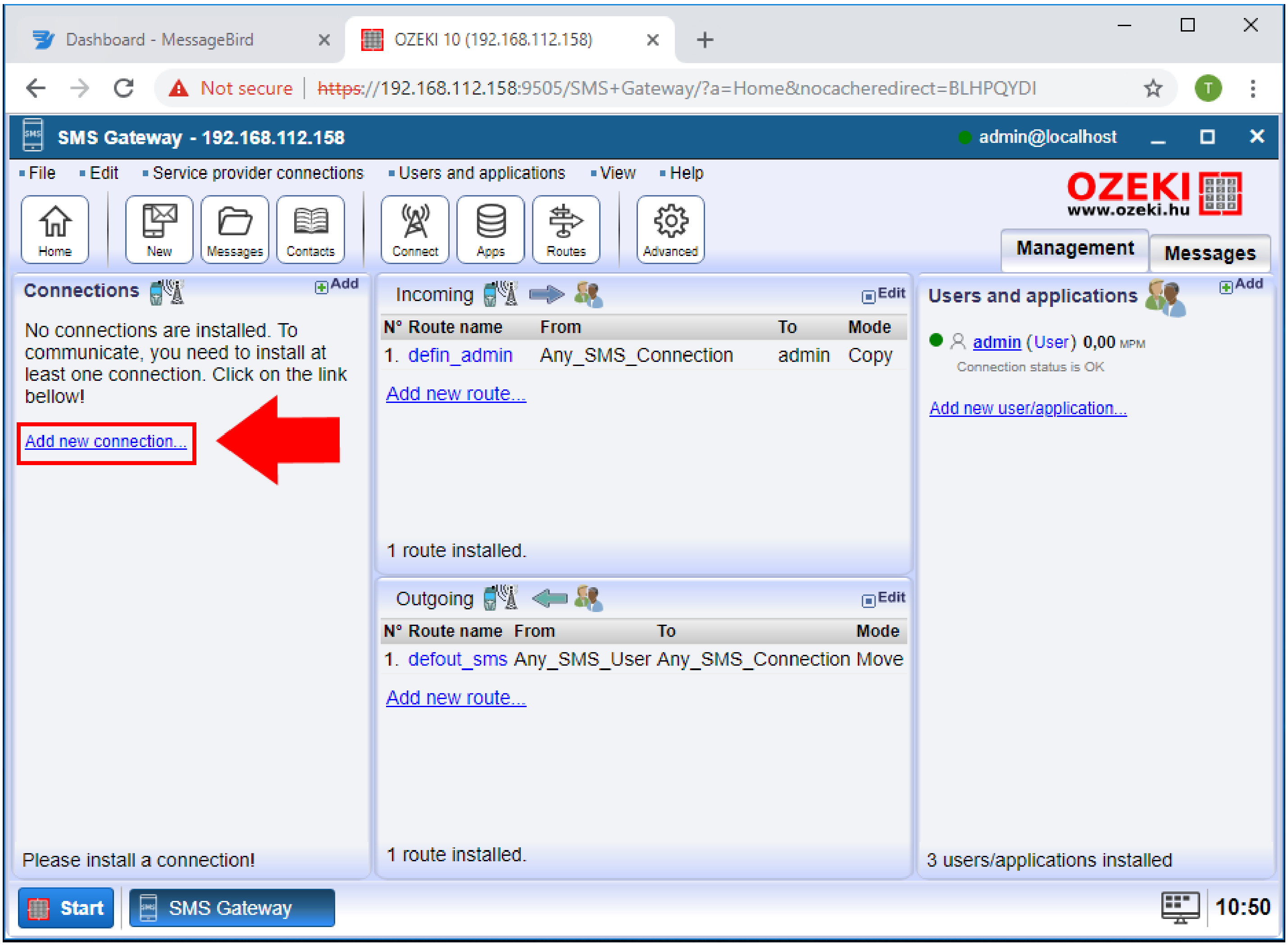
Version: 10.1.52

© 2000-2019 Ozeki Ltd. (www.ozeki.hu)

Session: 0c668df8-ac6f-4ca5-a06d-9544daaffc76

Figure 8 - Login to Ozeki 10

Click on '**Add new connection**' on the left.



No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

Please install a connection!

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defin_admin | Any_SMS_Connection | admin | Copy |

Add new route...

1 route installed.

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move |

Add new route...

1 route installed.

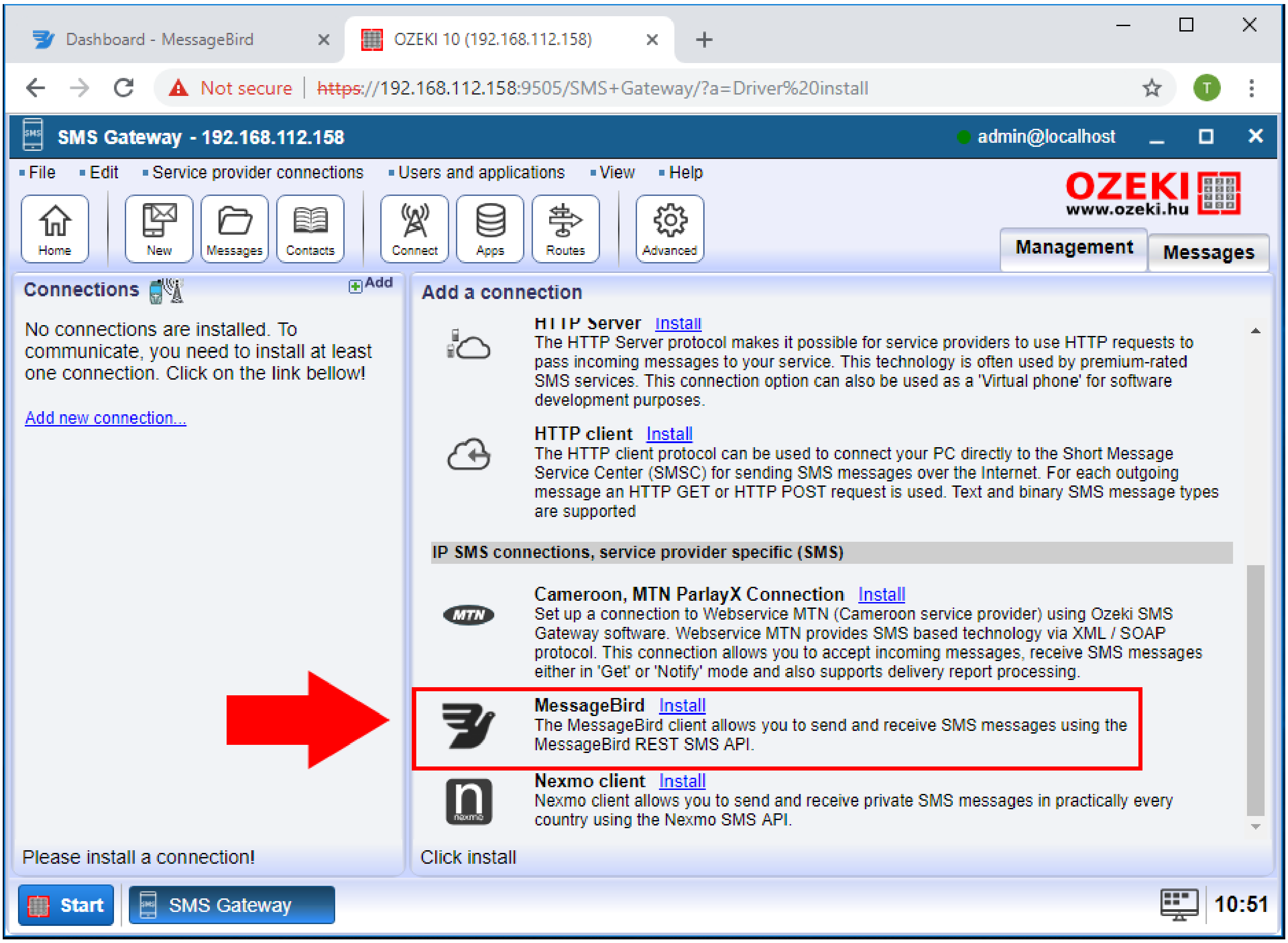
● admin (User) 0,00 MPM
Connection status is OK

Add new user/application...

3 users/applications installed

Figure 9 - Add new connection

Select the '**MessageBird**' Connection and click on install next to it.



No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

Please install a connection!

**HTTP Server** Install
The HTTP Server protocol makes it possible for service providers to use HTTP requests to pass incoming messages to your service. This technology is often used by premium-rated SMS services. This connection option can also be used as a 'Virtual phone' for software development purposes.

**HTTP client** Install
The HTTP client protocol can be used to connect your PC directly to the Short Message Service Center (SMSC) for sending SMS messages over the Internet. For each outgoing message an HTTP GET or HTTP POST request is used. Text and binary SMS message types are supported

**IP SMS connections, service provider specific (SMS)**

**Cameroon, MTN ParlayX Connection** Install
Set up a connection to Webservice MTN (Cameroon service provider) using Ozeki SMS Gateway software. Webservice MTN provides SMS based technology via XML / SOAP protocol. This connection allows you to accept incoming messages, receive SMS messages either in 'Get' or 'Notify' mode and also supports delivery report processing.

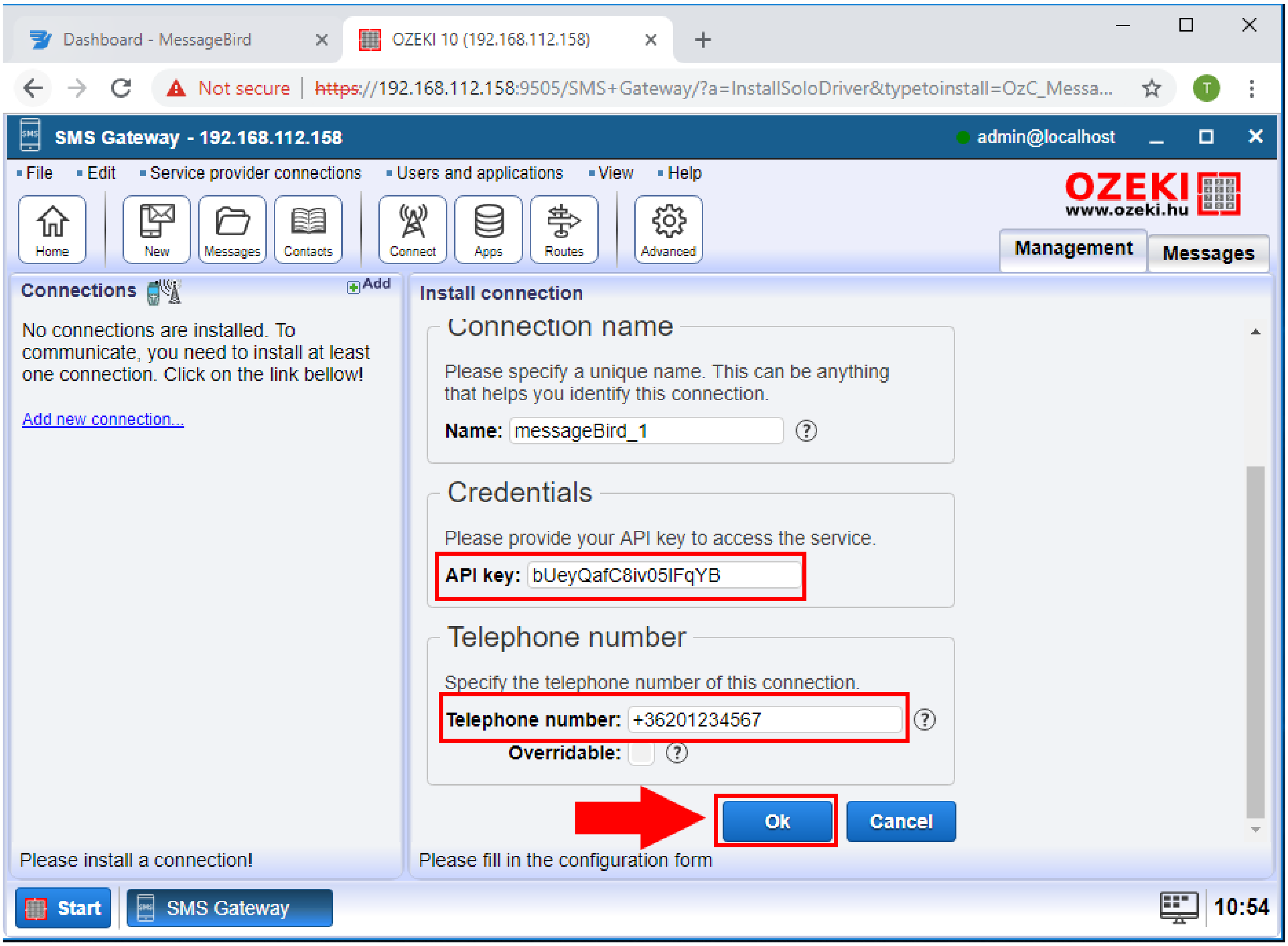
**MessageBird** Install
The MessageBird client allows you to send and receive SMS messages using the MessageBird REST SMS API.

**Nexmo client** Install
Nexmo client allows you to send and receive private SMS messages in practically every country using the Nexmo SMS API.

Click install

Figure 10 - Install MessageBird

Provide your **API key** and **telephone number** on the Install connection page.



No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

Please install a connection!

Install connection

Connection name

Please specify a unique name. This can be anything that helps you identify this connection.

Name: messageBird_1

Credentials

Please provide your API key to access the service.

API key: bUeyQafC8iv05IFqYB

Telephone number

Specify the telephone number of this connection.

Telephone number: +36201234567

Overridable: □

Ok Cancel

Please fill in the configuration form

Figure 11 - Provide API key

You can find your **API key** in the MessageBird Dashboard.

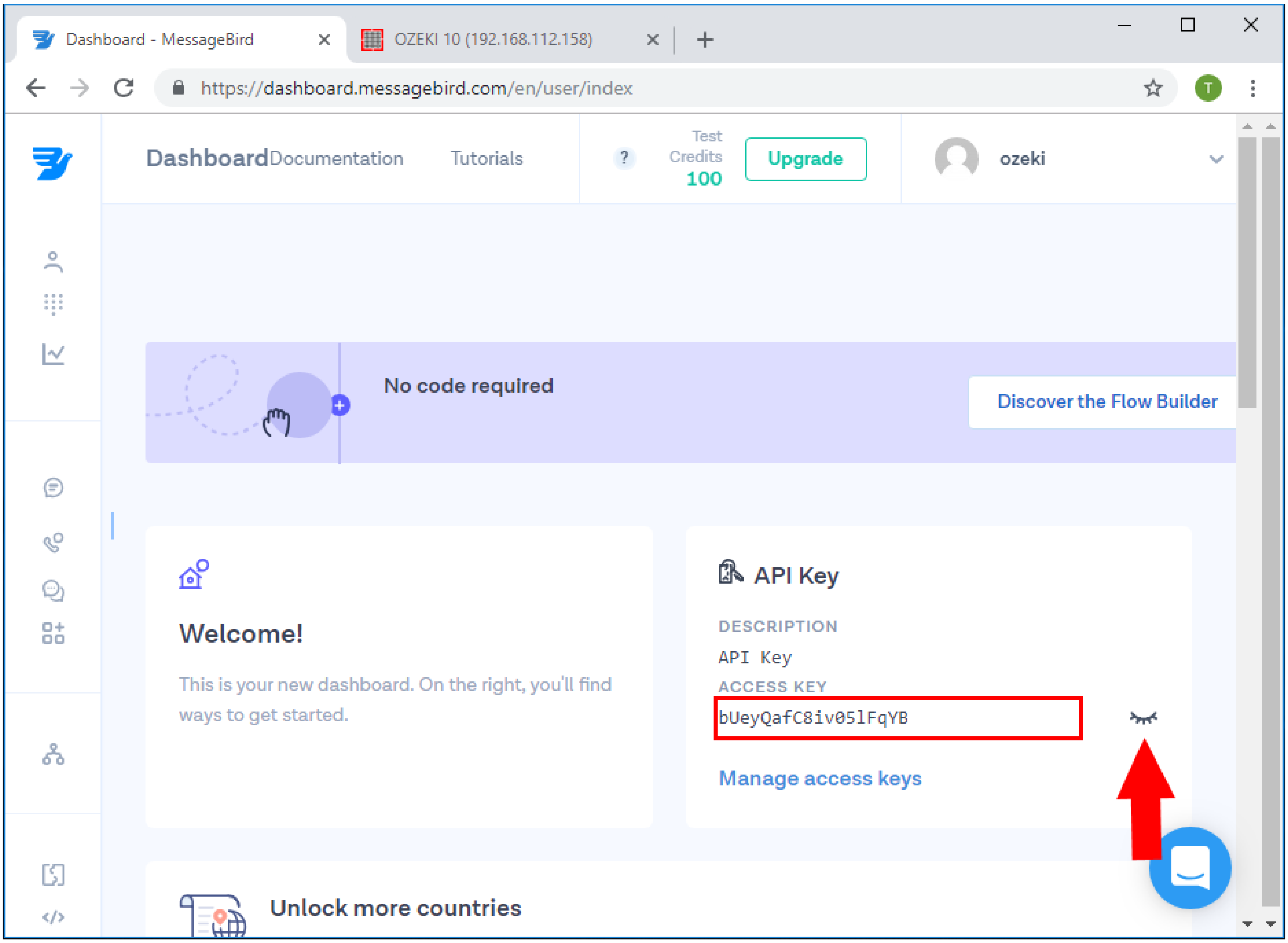
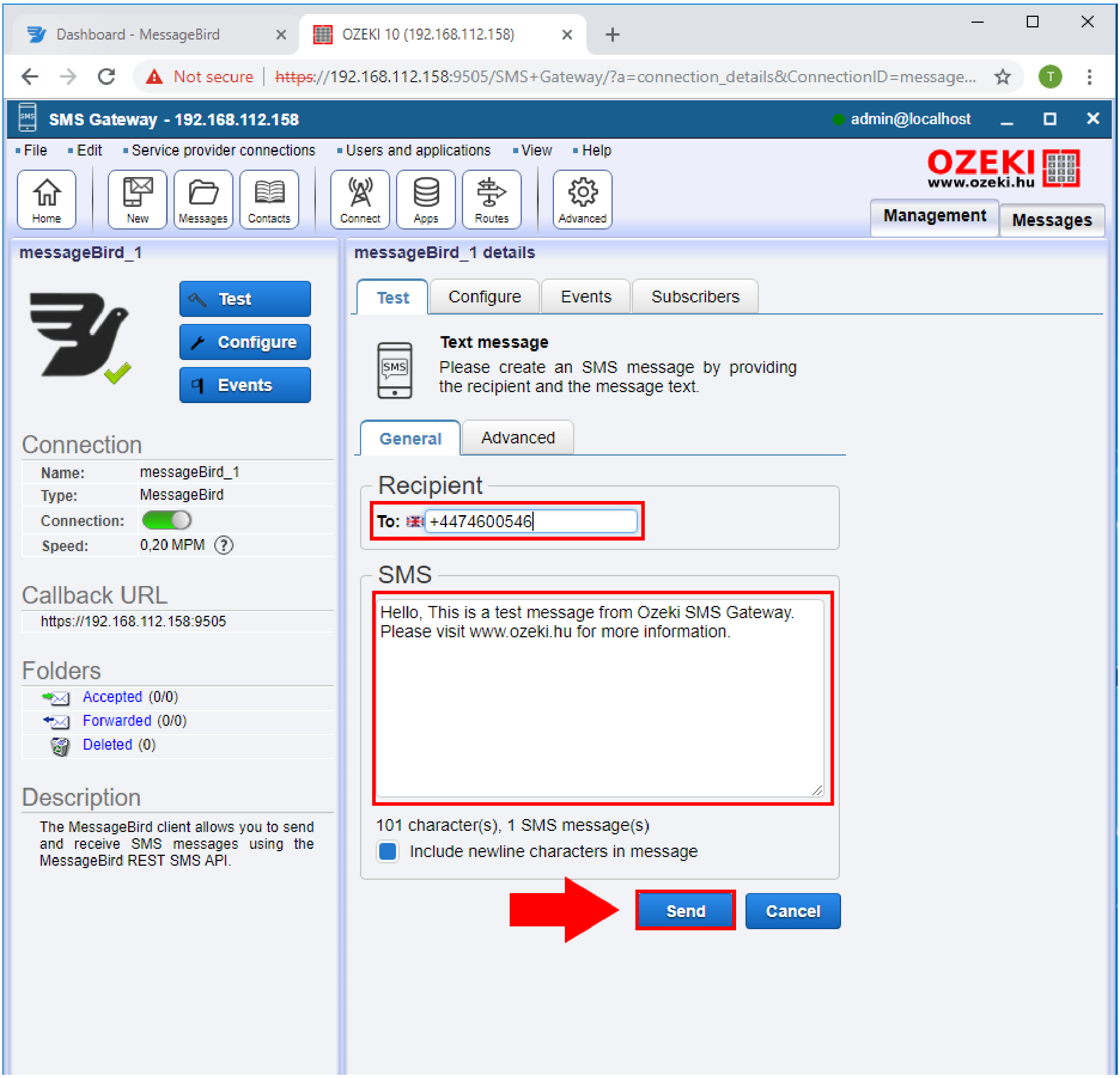


Figure 12 - MessageBird API key

After it you can send a test message.



Connection

Name: messageBird_1
Type: MessageBird
Connection:
Speed: 0,20 MPM

Callback URL

https://192.168.112.158:9505

Folders

Accepted (0/0)
Forwarded (0/0)
Deleted (0)

Description

The MessageBird client allows you to send and receive SMS messages using the MessageBird REST SMS API.

**Text message**
Please create an SMS message by providing the recipient and the message text.

To: +4474600546

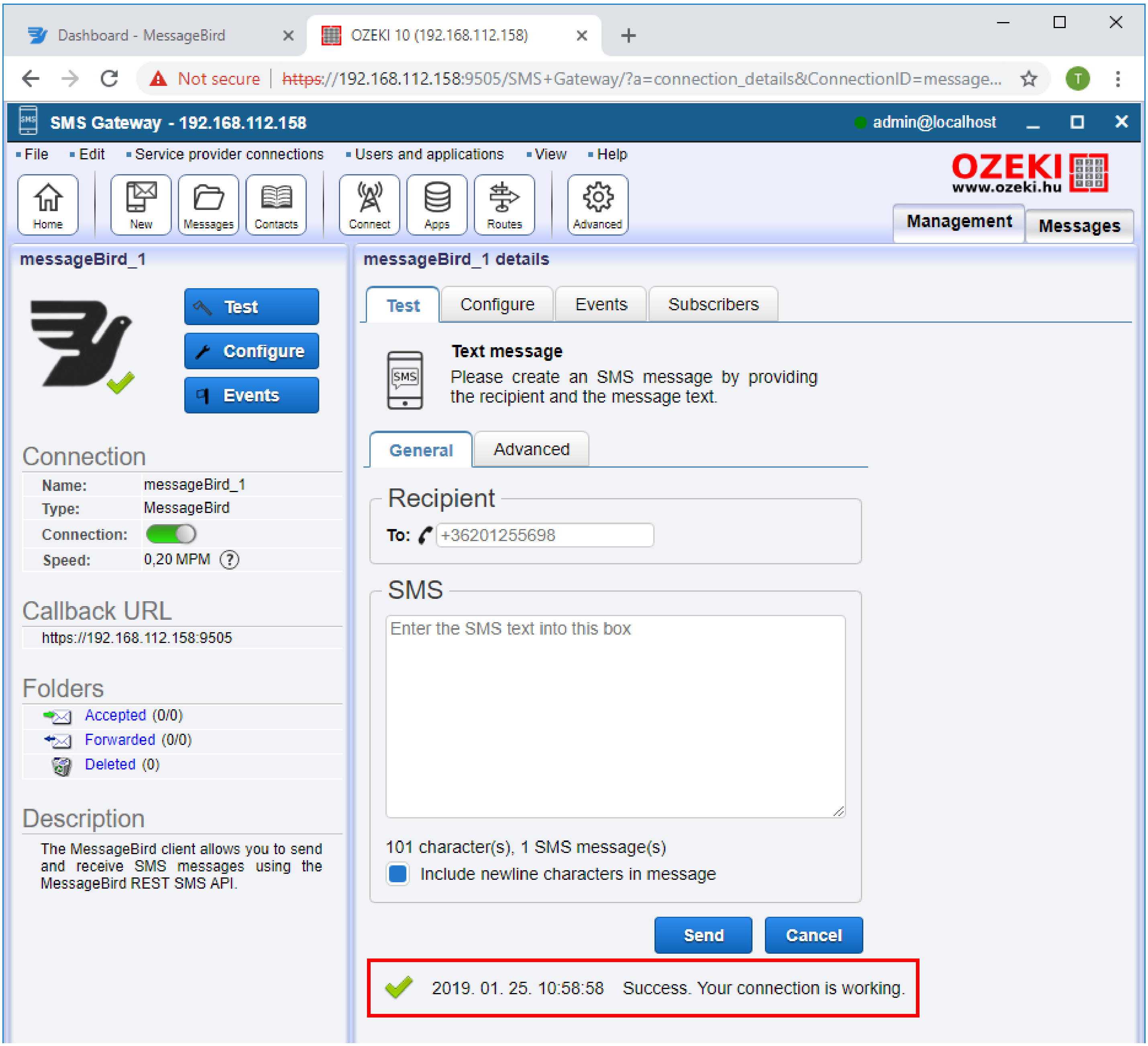
Hello, This is a test message from Ozeki SMS Gateway.
Please visit www.ozeki.hu for more information.

101 character(s), 1 SMS message(s)

☑ Include newline characters in message

Figure 13 - Send test message

If everything fine the message successfully sent.



Connection

Name: messageBird_1
Type: MessageBird
Connection:
Speed: 0,20 MPM

Callback URL

https://192.168.112.158:9505

Folders

Accepted (0/0)
Forwarded (0/0)
Deleted (0)

Description

The MessageBird client allows you to send and receive SMS messages using the MessageBird REST SMS API.

Text message

Please create an SMS message by providing the recipient and the message text.

Recipient

To: +36201255698

SMS

Enter the SMS text into this box

101 character(s), 1 SMS message(s)

☑ Include newline characters in message

2019. 01. 25. 10:58:58 Success. Your connection is working.

Figure 14 - Message successfully sent

# How to setup Ozeki 10 and MessageBird for incoming SMS and delivery reports

Before you can receive incoming delivery reports and inbound SMS, you need to forward ports in order to route the external traffic to your network.

1. Open a Web browser and type your Default Gateway number into the address bar then press Enter.

2. Enter your username and password to access the interface of your router. The default username and password should be listed in the documentation of your router or on a sticker on the side of your router. If the default username and password have been changed and you do not remember them, you will need to reset your router.

3. To forward ports on your router, look for a tab or menu named 'Port Forwarding/Port Triggering' or something similar.

4. No matter what type of router or interface you have, you will need to enter the same basic information. Enter the port you want to open under External (Port 1) and Internal (Port 2), or enter a range of ports to open under Start and End.
In this case Ozeki 10 uses port 9505 by default so you may also provide that in the 'Internal' section.

5. Select the Protocol (TCP, UDP, or both).

6. Enter the Private IP address of your PC where your Ozeki 10 installed.

7. Be sure you saved the changes.

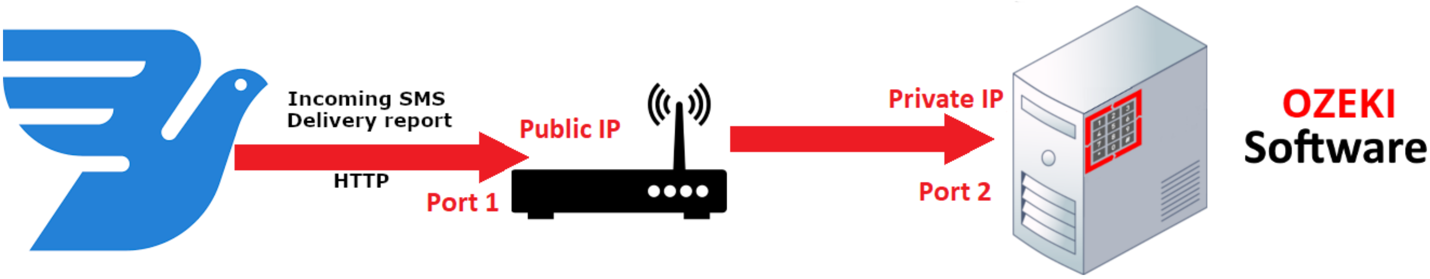
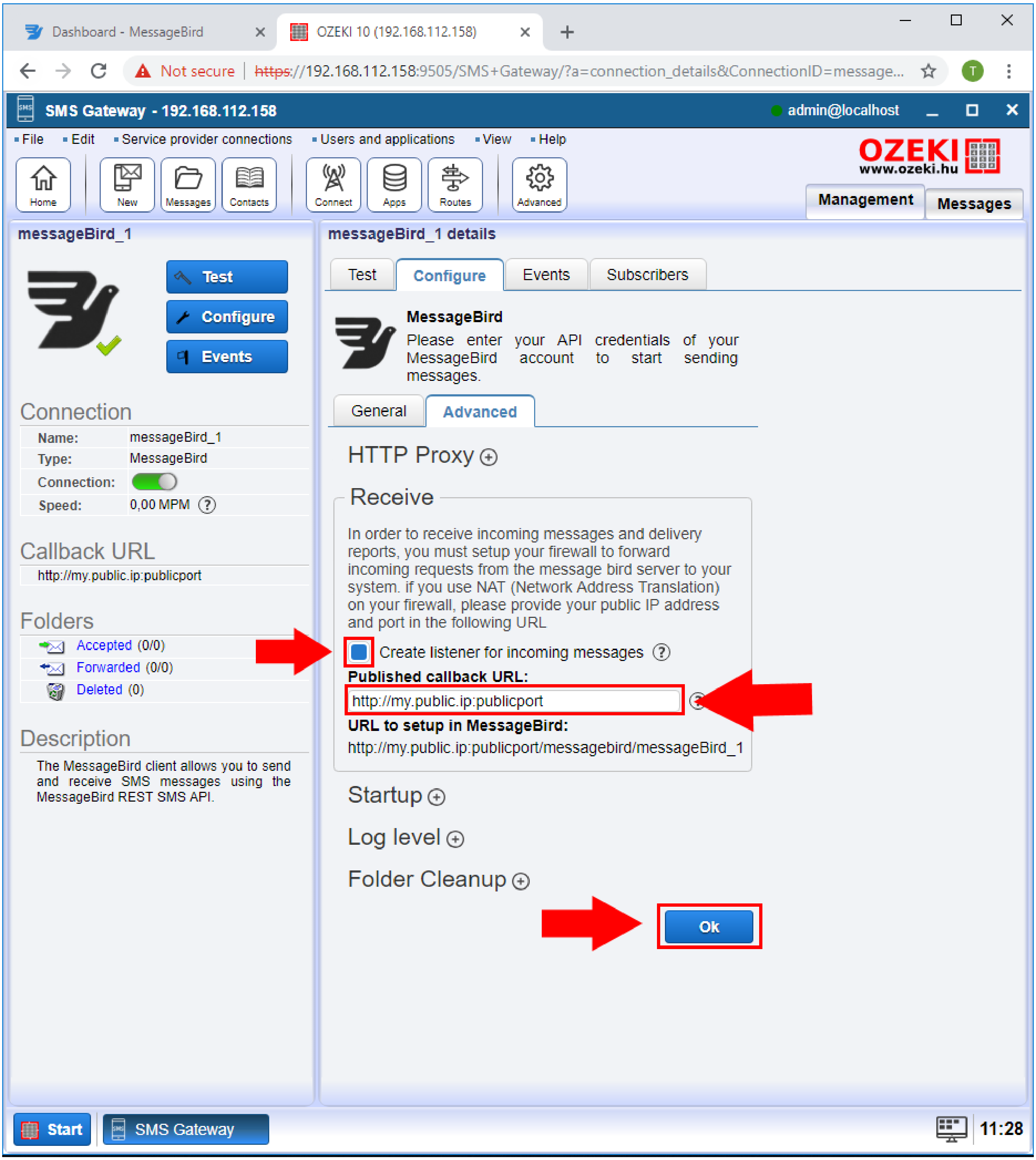


Figure 15 - Port Forwarding

After having the changes saved, please specify the callback URL on the Advanced tab's Receive section.



Connection

| | |
|---|---|
| Name: | messageBird_1 |
| Type: | MessageBird |
| Connection: | |
| Speed: | 0,00 MPM |

Callback URL

http://my.public.ip:publicport

Folders

Accepted (0/0)
Forwarded (0/0)
Deleted (0)

Description

The MessageBird client allows you to send and receive SMS messages using the MessageBird REST SMS API.

MessageBird

Please enter your API credentials of your MessageBird account to start sending messages.

Receive

In order to receive incoming messages and delivery reports, you must setup your firewall to forward incoming requests from the message bird server to your system. if you use NAT (Network Address Translation) on your firewall, please provide your public IP address and port in the following URL

☑ Create listener for incoming messages

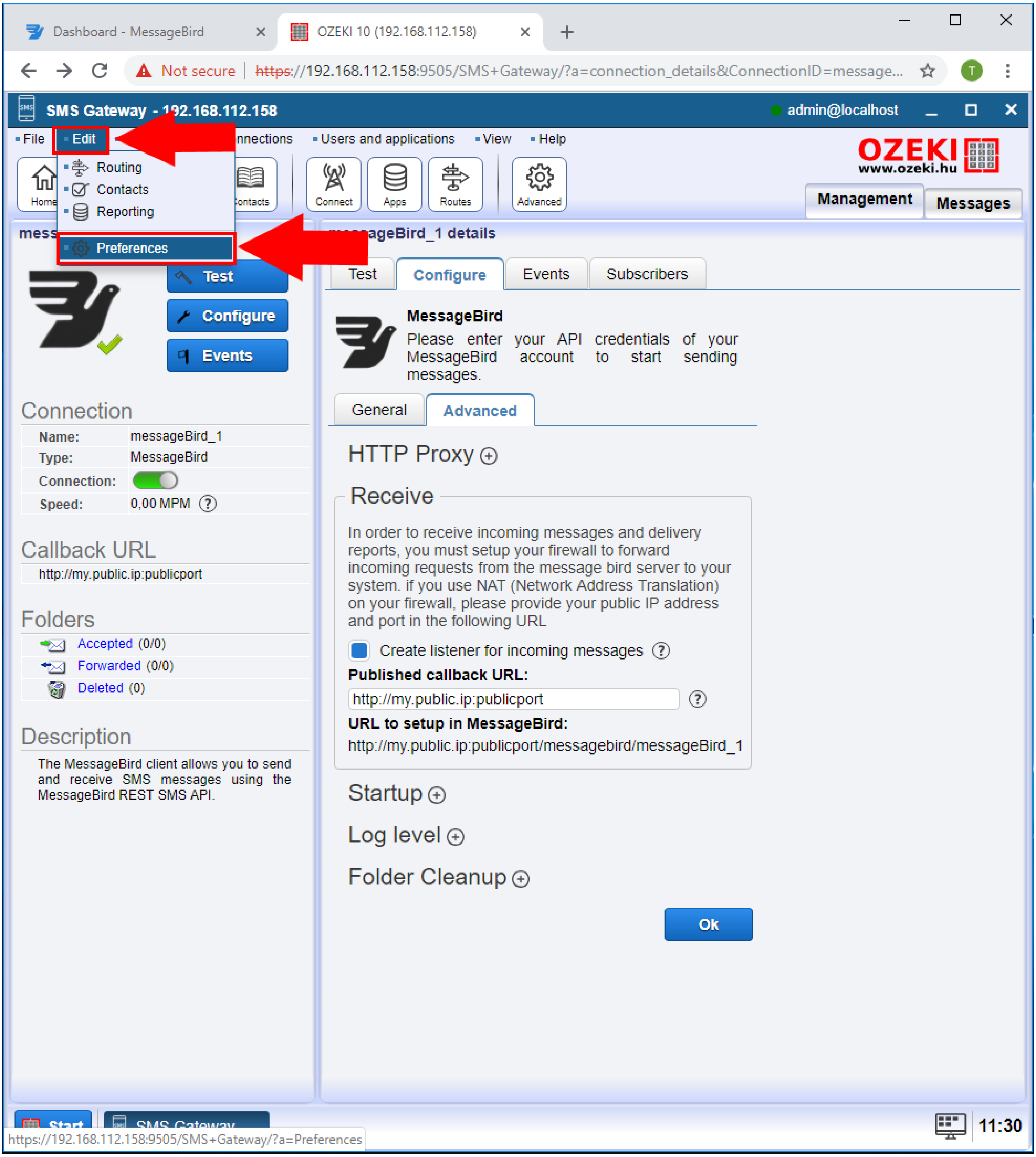
Published callback URL: http://my.public.ip:publicport

URL to setup in MessageBird: http://my.public.ip:publicport/messagebird/messageBird_1

Figure 16 - Receive message

In order to receive messages from MessageBird you need to change the webserver protocol to HTTP.

Select **Edit** and **Preferences**.



messageBird_1 details

**MessageBird**

Please enter your API credentials of your MessageBird account to start sending messages.

Connection

| | |
|---|---|
| Name: | messageBird_1 |
| Type: | MessageBird |
| Connection: | ● |
| Speed: | 0,00 MPM ? |

Callback URL

http://my.public.ip:publicport

Folders

- Accepted (0/0)
- Forwarded (0/0)
- Deleted (0)

Description

The MessageBird client allows you to send and receive SMS messages using the MessageBird REST SMS API.

Receive

In order to receive incoming messages and delivery reports, you must setup your firewall to forward incoming requests from the message bird server to your system. if you use NAT (Network Address Translation) on your firewall, please provide your public IP address and port in the following URL

☑ Create listener for incoming messages ?

**Published callback URL:**

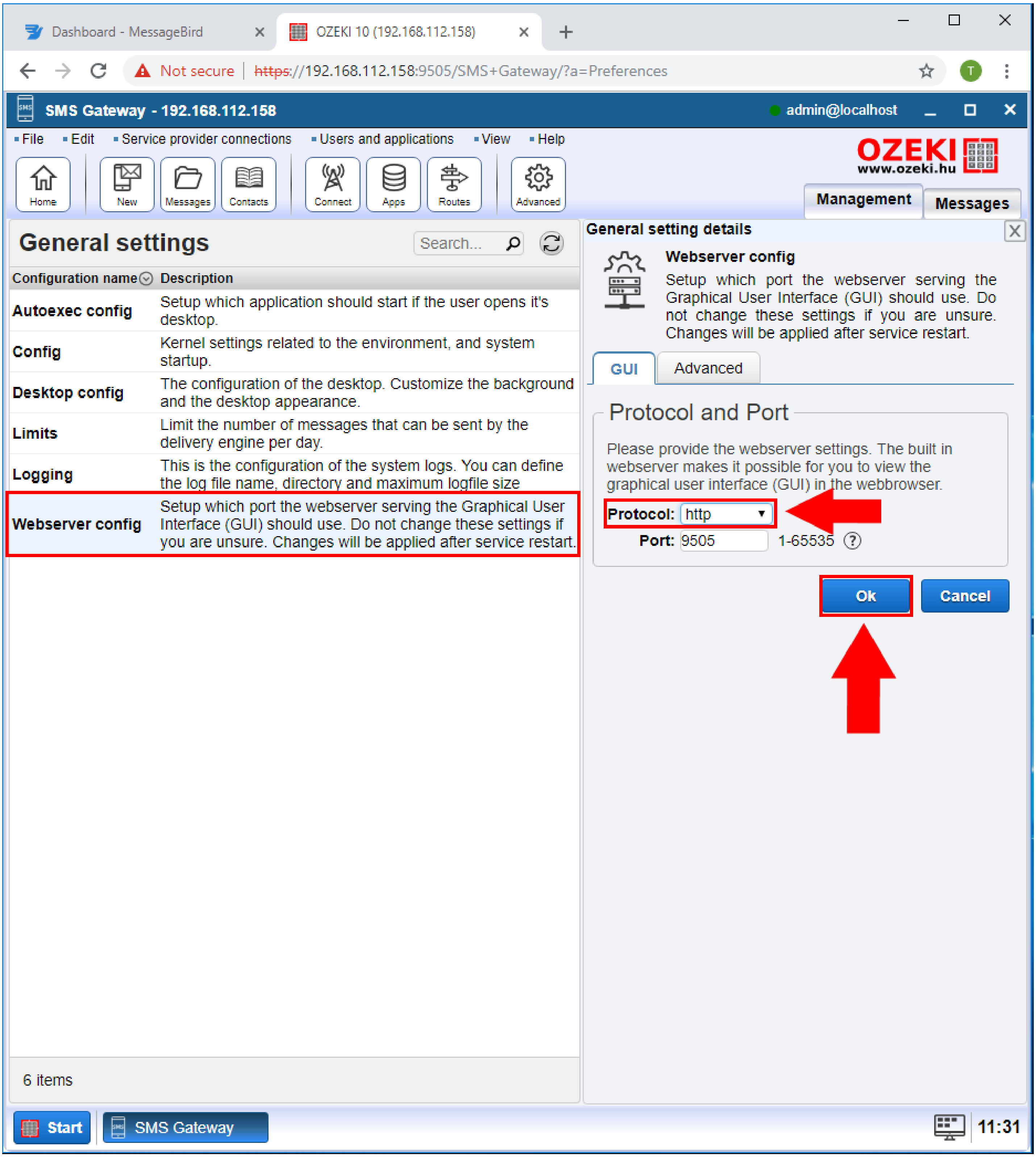
http://my.public.ip:publicport ?

**URL to setup in MessageBird:**

http://my.public.ip:publicport/messagebird/messageBird_1

Figure 17 - Preferences

Under the **Webserver configuration** choose http protocol and click on the **OK** button.



General settings

| Configuration name | Description |
|---|---|
| Autoexec config | Setup which application should start if the user opens it's desktop. |
| Config | Kernel settings related to the environment, and system startup. |
| Desktop config | The configuration of the desktop. Customize the background and the desktop appearance. |
| Limits | Limit the number of messages that can be sent by the delivery engine per day. |
| Logging | This is the configuration of the system logs. You can define the log file name, directory and maximum logfile size |
| Webserver config | Setup which port the webserver serving the Graphical User Interface (GUI) should use. Do not change these settings if you are unsure. Changes will be applied after service restart. |

6 items

General setting details

Webserver config

Setup which port the webserver serving the Graphical User Interface (GUI) should use. Do not change these settings if you are unsure. Changes will be applied after service restart.

GUI | Advanced

Protocol and Port

Please provide the webserver settings. The built in webserver makes it possible for you to view the graphical user interface (GUI) in the webbrowser.

Protocol: http

Port: 9505 1-65535

Ok | Cancel

Figure 18 - Webserver config

To apply the changes please restart the Ozeki 10 service

Right-click on your **Start** button and select **Run**.

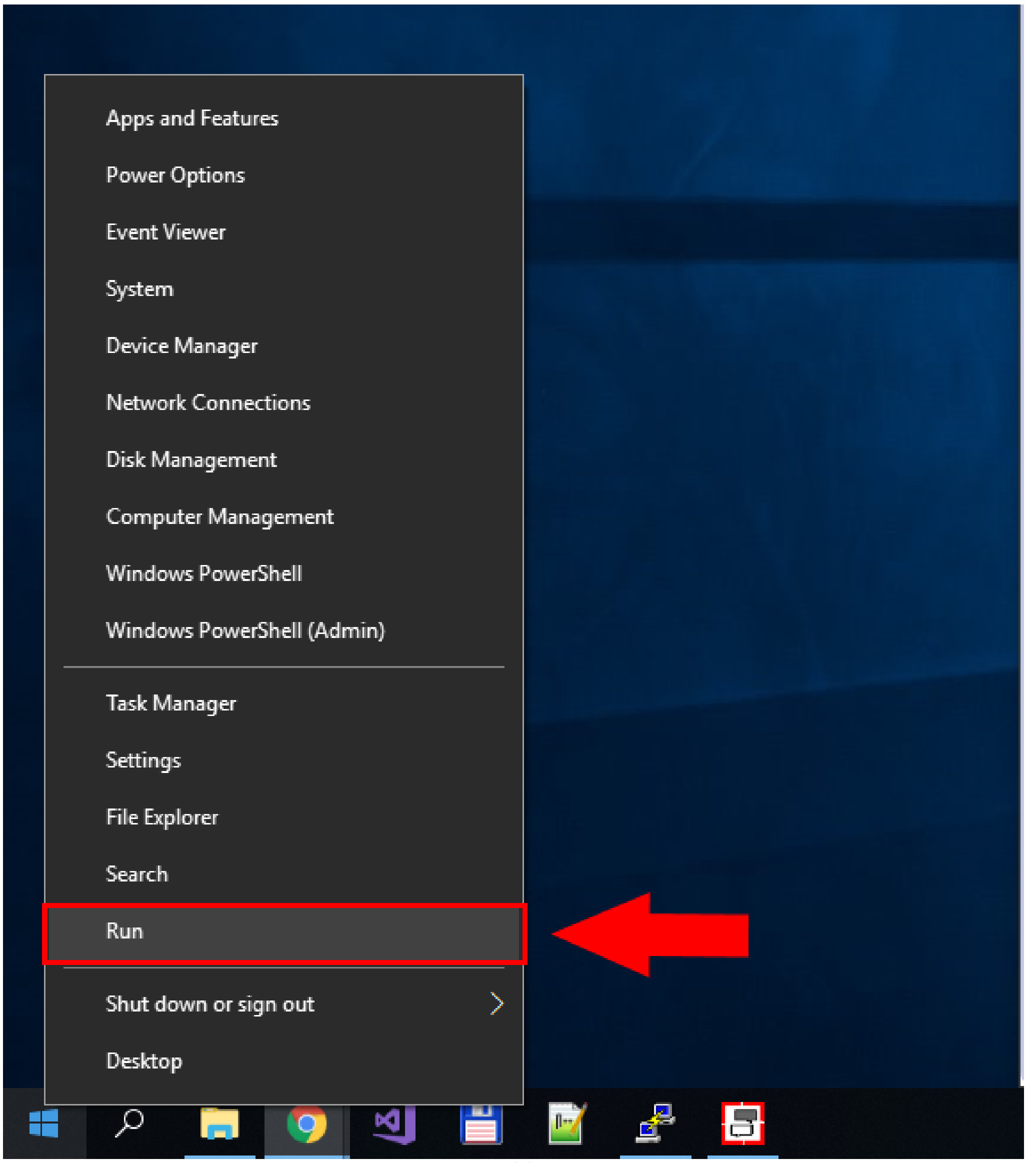
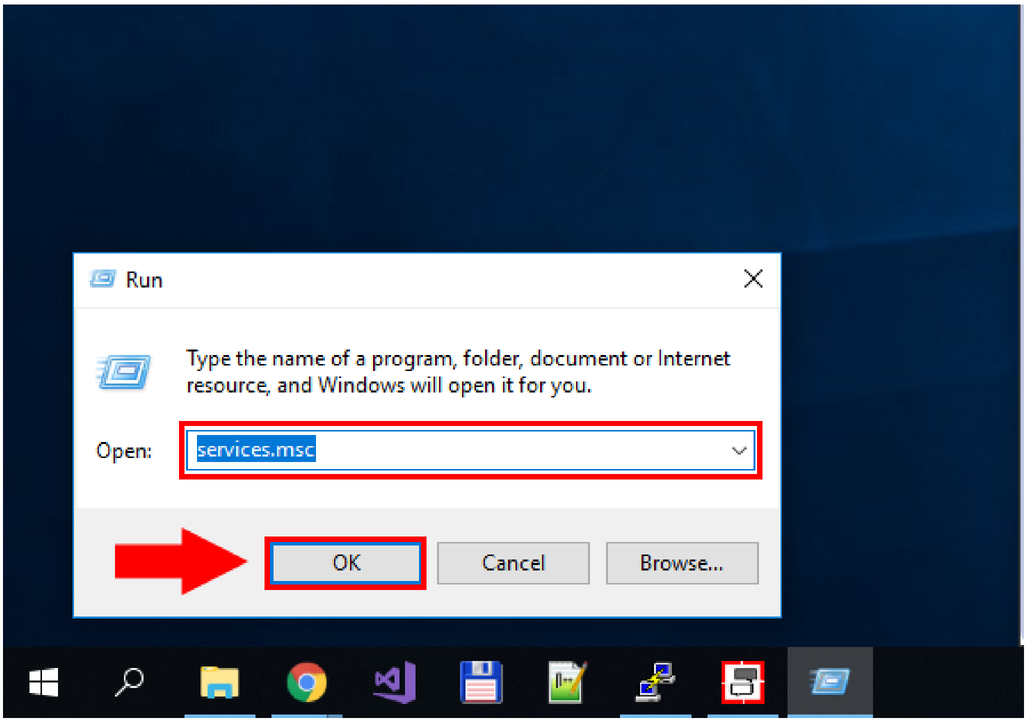


Figure 19 - Run

Type **services.msc** in the Run box.

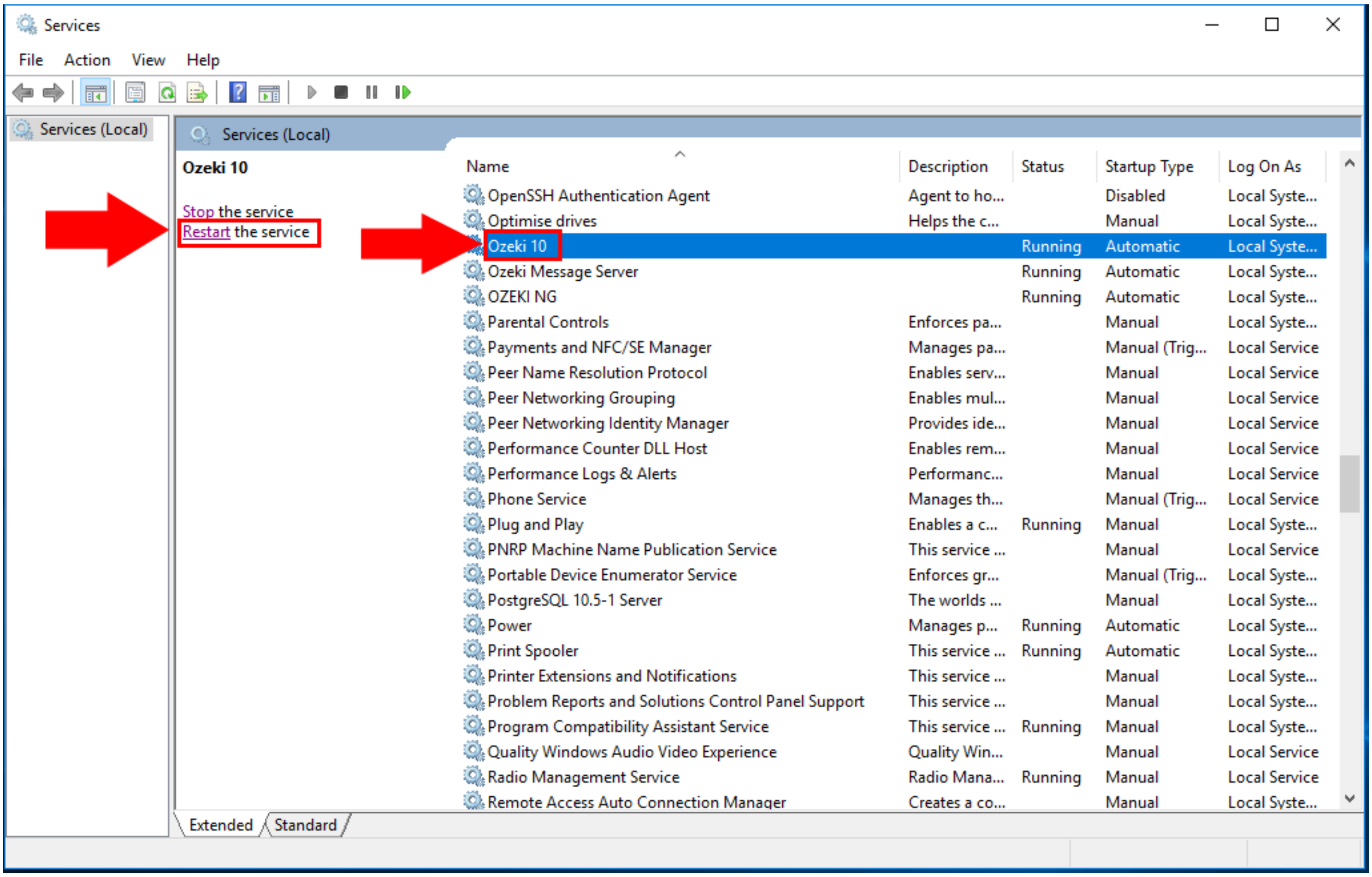


Type the name of a program, folder, document or Internet resource, and Windows will open it for you.

Open: services.msc

Figure 20 - services.msc

To restart the Ozeki 10 service, select it in the service list and click **Restart** service.



Ozeki 10

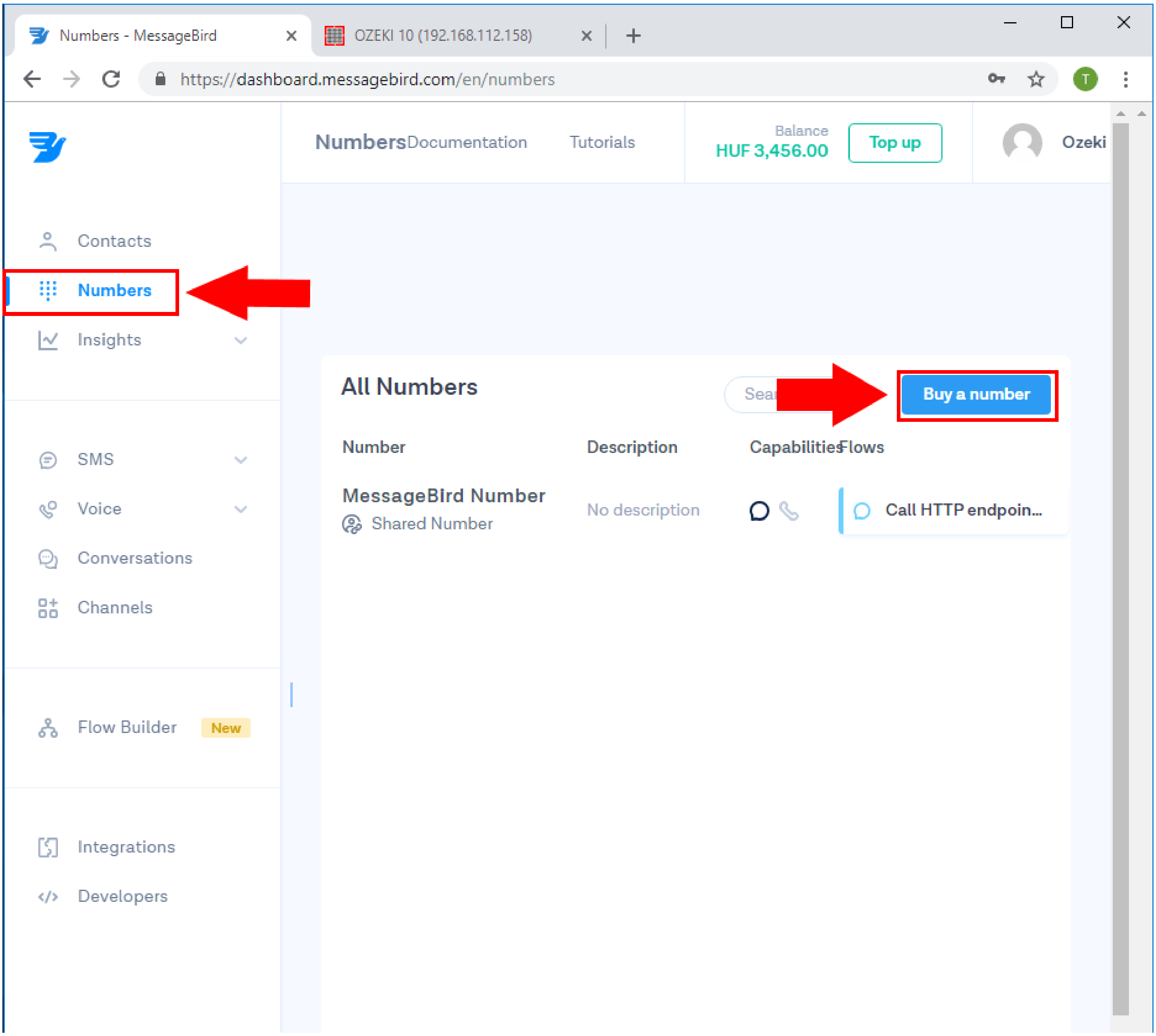
Stop the service
Restart the service

| Name | Description | Status | Startup Type | Log On As |
|---|---|---|---|---|
| OpenSSH Authentication Agent | Agent to ho... | | Disabled | Local Syste... |
| Optimise drives | Helps the c... | | Manual | Local Syste... |
| Ozeki 10 | | Running | Automatic | Local Syste... |
| Ozeki Message Server | | Running | Automatic | Local Syste... |
| OZEKI NG | | Running | Automatic | Local Syste... |
| Parental Controls | Enforces pa... | | Manual | Local Syste... |
| Payments and NFC/SE Manager | Manages pa... | | Manual (Trig... | Local Service |
| Peer Name Resolution Protocol | Enables serv... | | Manual | Local Service |
| Peer Networking Grouping | Enables mul... | | Manual | Local Service |
| Peer Networking Identity Manager | Provides ide... | | Manual | Local Service |
| Performance Counter DLL Host | Enables rem... | | Manual | Local Service |
| Performance Logs & Alerts | Performanc... | | Manual | Local Service |
| Phone Service | Manages th... | | Manual (Trig... | Local Service |
| Plug and Play | Enables a c... | Running | Manual | Local Syste... |
| PNRP Machine Name Publication Service | This service ... | | Manual | Local Service |
| Portable Device Enumerator Service | Enforces gr... | | Manual (Trig... | Local Syste... |
| PostgreSQL 10.5-1 Server | The worlds ... | | Manual | Local Syste... |
| Power | Manages p... | Running | Automatic | Local Syste... |
| Print Spooler | This service ... | Running | Automatic | Local Syste... |
| Printer Extensions and Notifications | This service ... | | Manual | Local Syste... |
| Problem Reports and Solutions Control Panel Support | This service ... | | Manual | Local Syste... |
| Program Compatibility Assistant Service | This service ... | Running | Manual | Local Syste... |
| Quality Windows Audio Video Experience | Quality Win... | | Manual | Local Service |
| Radio Management Service | Radio Mana... | Running | Manual | Local Service |
| Remote Access Auto Connection Manager | Creates a co... | | Manual | Local Syste... |

Figure 21 - Restart Ozeki 10

After the service restart you are able to receive messages form the MessageBird. Now we will demonstrate how you can buy MessageBird number and configure MessageBird to receive SMS messages.

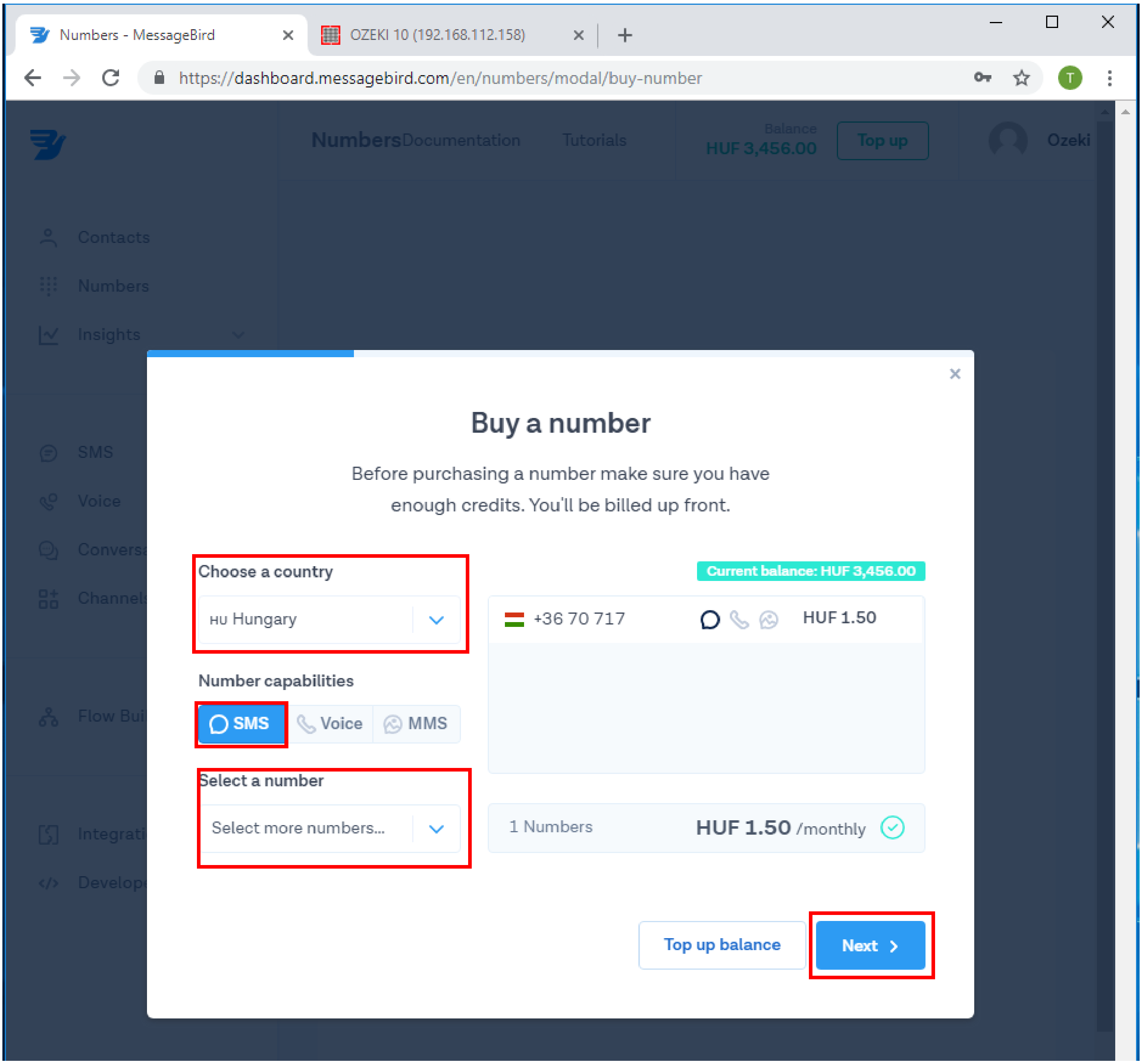
Select the **Numbers** menu and click on the **Buy a number** button.



| Number | Description | Capabilities | Flows |
|---|---|---|---|
| MessageBird Number<br>Shared Number | No description | | Call HTTP endpoin... |

Figure 22 - Buy a number

In the pop up menu choose your country, select the SMS capability and select a number from the list.



Buy a number

Before purchasing a number make sure you have enough credits. You'll be billed up front.

Choose a country: HU Hungary

Number capabilities: SMS [✓] Voice MMS

Select a number: Select more numbers...

Current balance: HUF 3,456.00

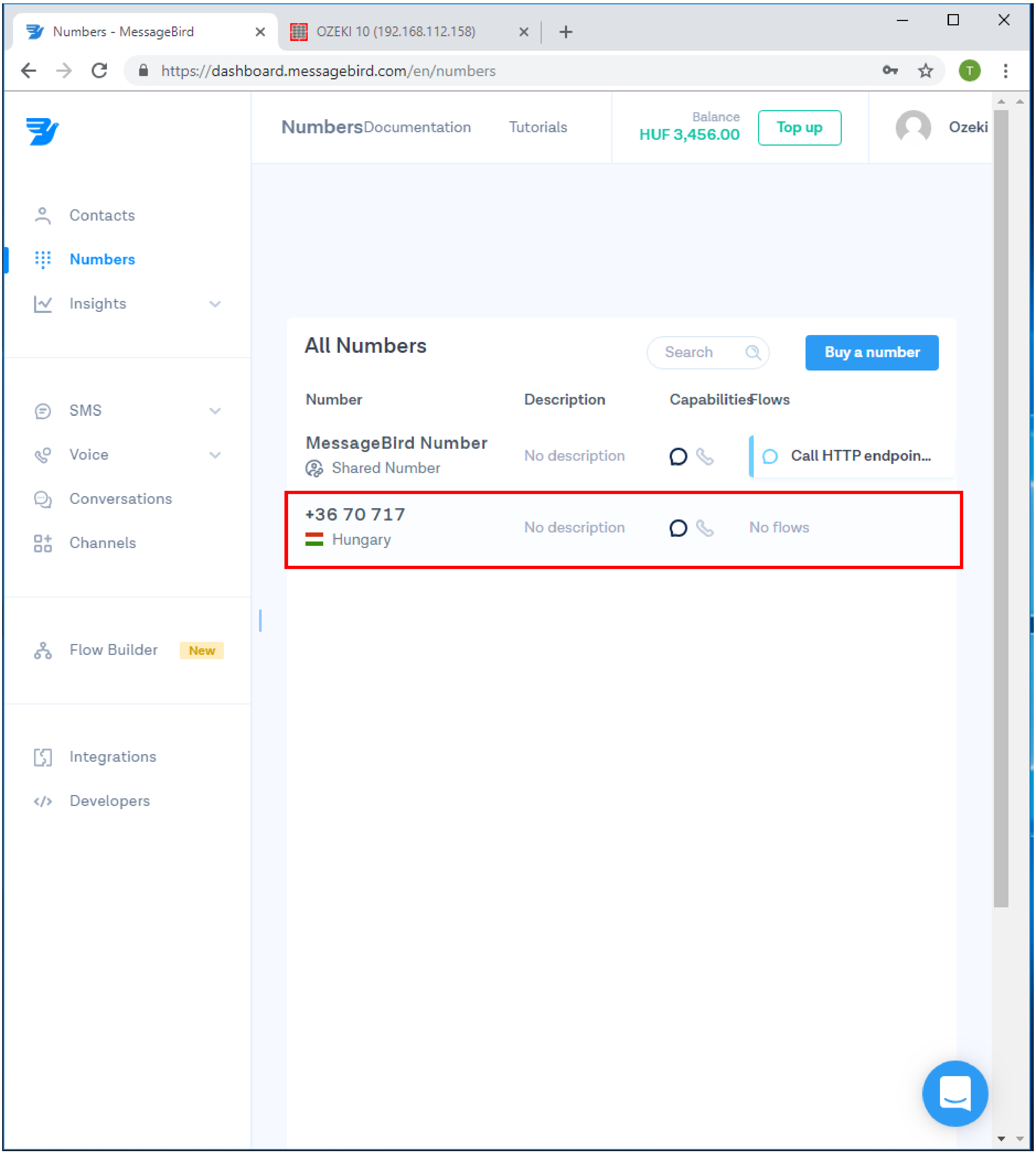
+36 70 717 HUF 1.50

1 Numbers HUF 1.50 /monthly ✓

Balance HUF 3,456.00

Figure 23 - Select a number

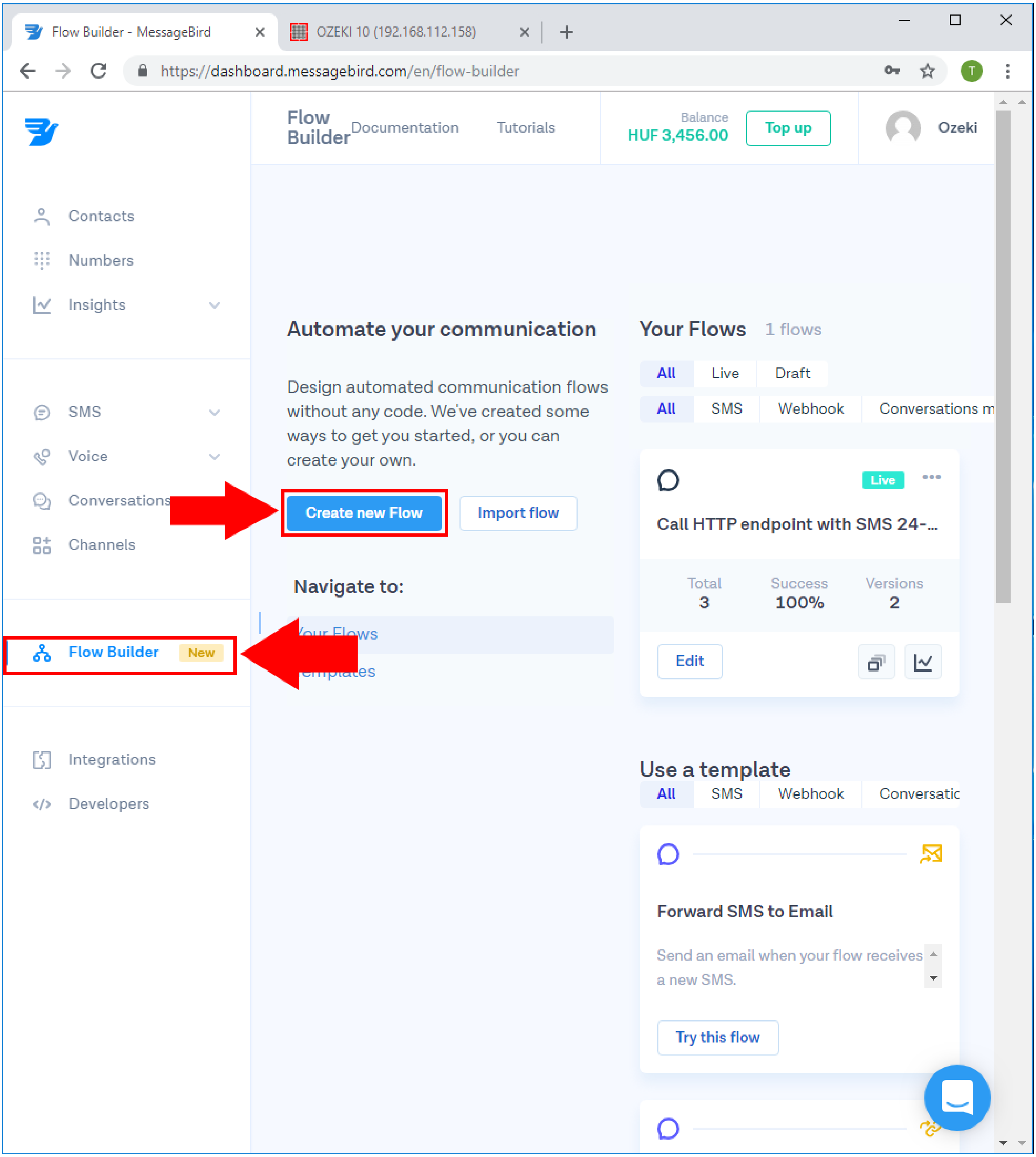
After the purchase you can see your number int he All Numbers list.



| Number | Description | Capabilities | Flows |
|---|---|---|---|
| MessageBird Number<br>Shared Number | No description | | Call HTTP endpoin... |
| +36 70 717<br>Hungary | No description | | No flows |

Figure 24 - Number purchased

Now select the **Flow Builder** and click on the **Create new Flow**.



Flow Builder - MessageBird

OZEKI 10 (192.168.112.158)

https://dashboard.messagebird.com/en/flow-builder

Flow Builder | Documentation | Tutorials | Balance HUF 3,456.00 | Top up | Ozeki

Contacts
Numbers
Insights
SMS
Voice
Conversations
Channels
Flow Builder New
Integrations
Developers

Automate your communication

Design automated communication flows without any code. We've created some ways to get you started, or you can create your own.

Create new Flow | Import flow

Navigate to:
Your Flows
Templates

Your Flows 1 flows

All | Live | Draft
All | SMS | Webhook | Conversations m

Live
Call HTTP endpoint with SMS 24-...

| Total | Success | Versions |
|---|---|---|
| 3 | 100% | 2 |

Edit

Use a template
All | SMS | Webhook | Conversatic

Forward SMS to Email

Send an email when your flow receives a new SMS.

Try this flow

Figure 25 - Create new Flow

Choose **Create Custom Flow**

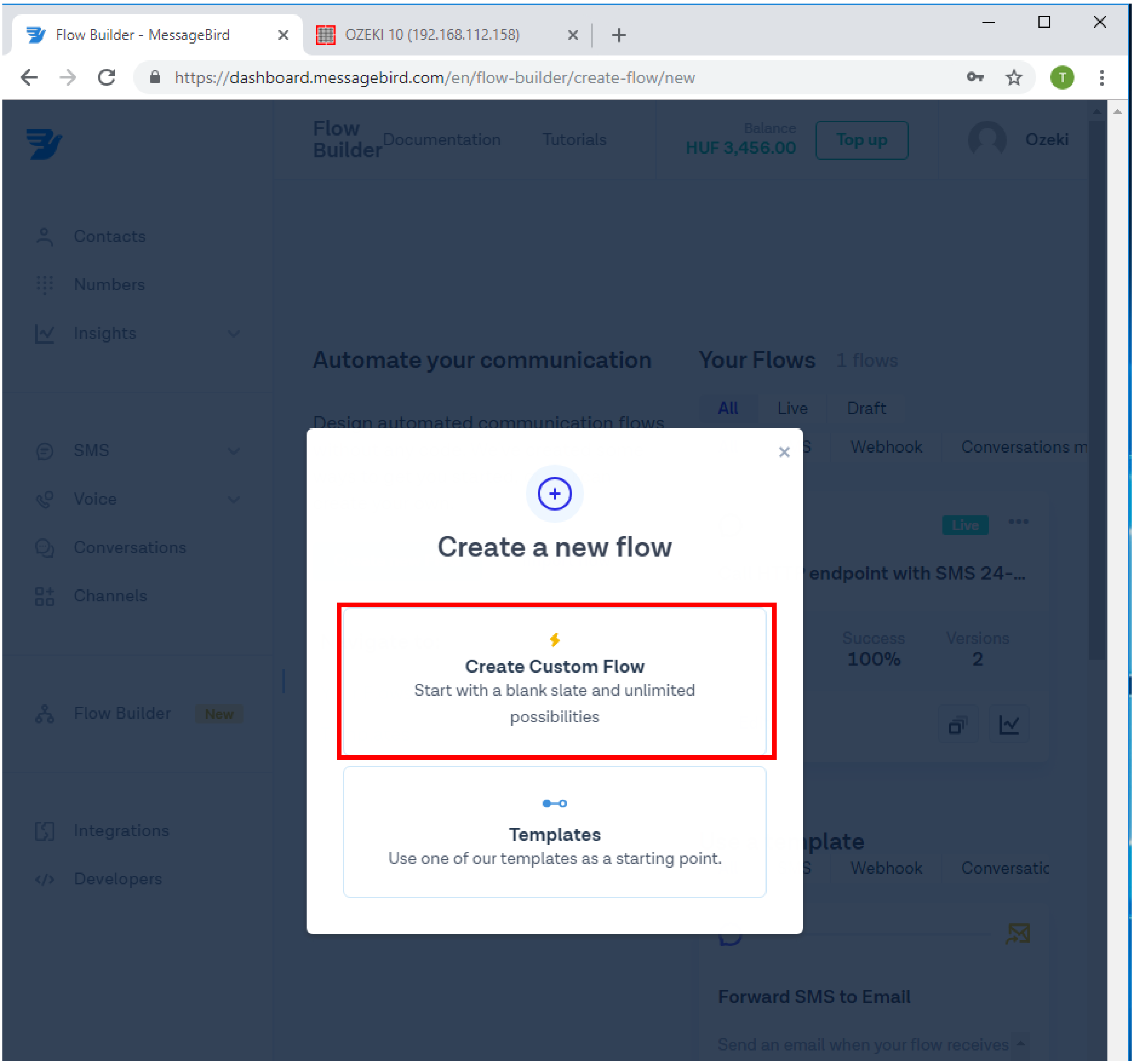
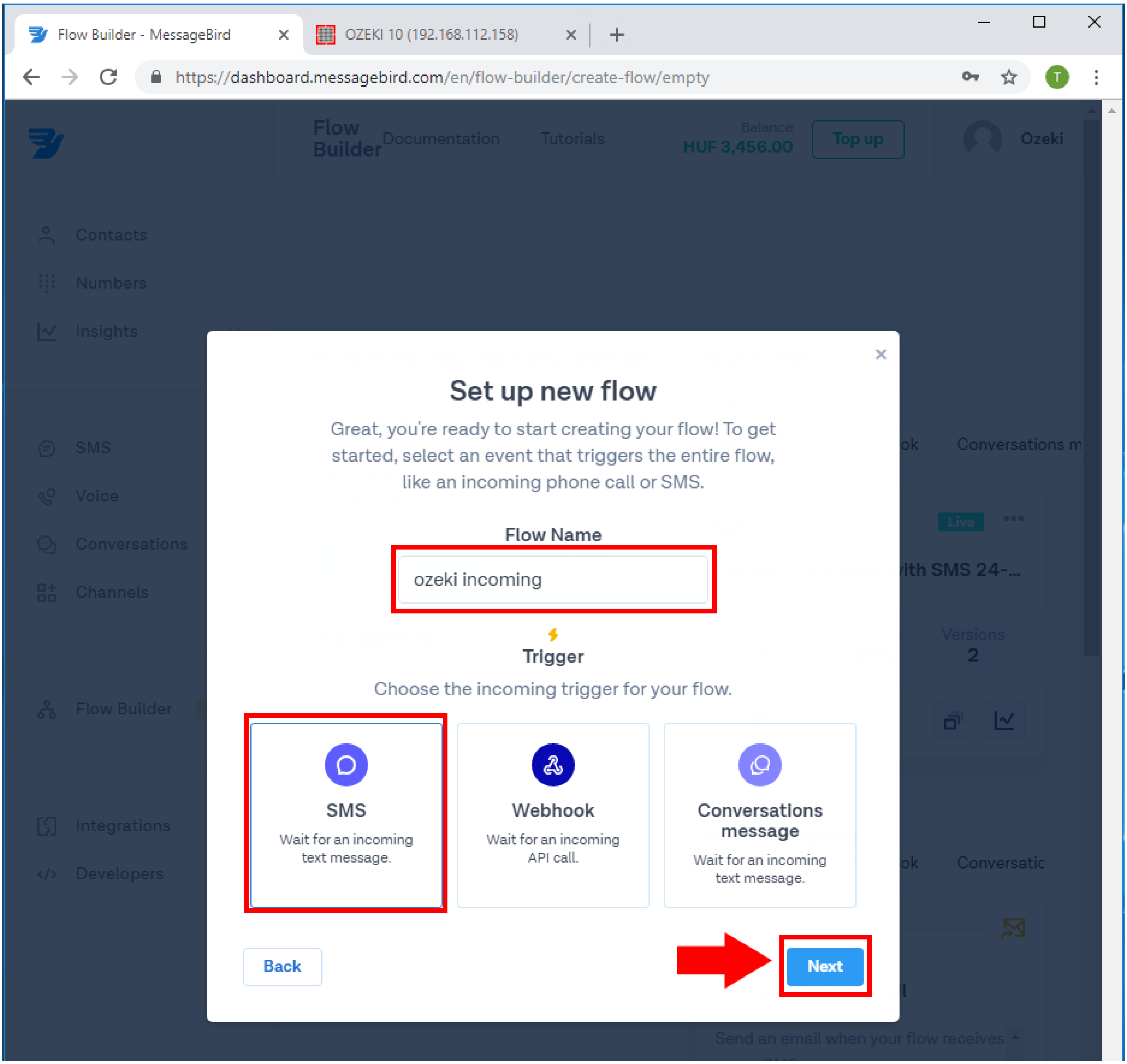


Figure 26 - Custom Flow

In the Set up new Flow menu specify the **Flow name** and the **SMS** trigger.



https://dashboard.messagebird.com/en/flow-builder/create-flow/empty

Set up new flow

Great, you're ready to start creating your flow! To get started, select an event that triggers the entire flow, like an incoming phone call or SMS.

Flow Name: ozeki incoming

Trigger

Choose the incoming trigger for your flow.

SMS – Wait for an incoming text message.

Webhook – Wait for an incoming API call.

Conversations message – Wait for an incoming text message.

Figure 27 - Setup new flow

Then select your phone number and drag the **HTTP Request** under the SMS.

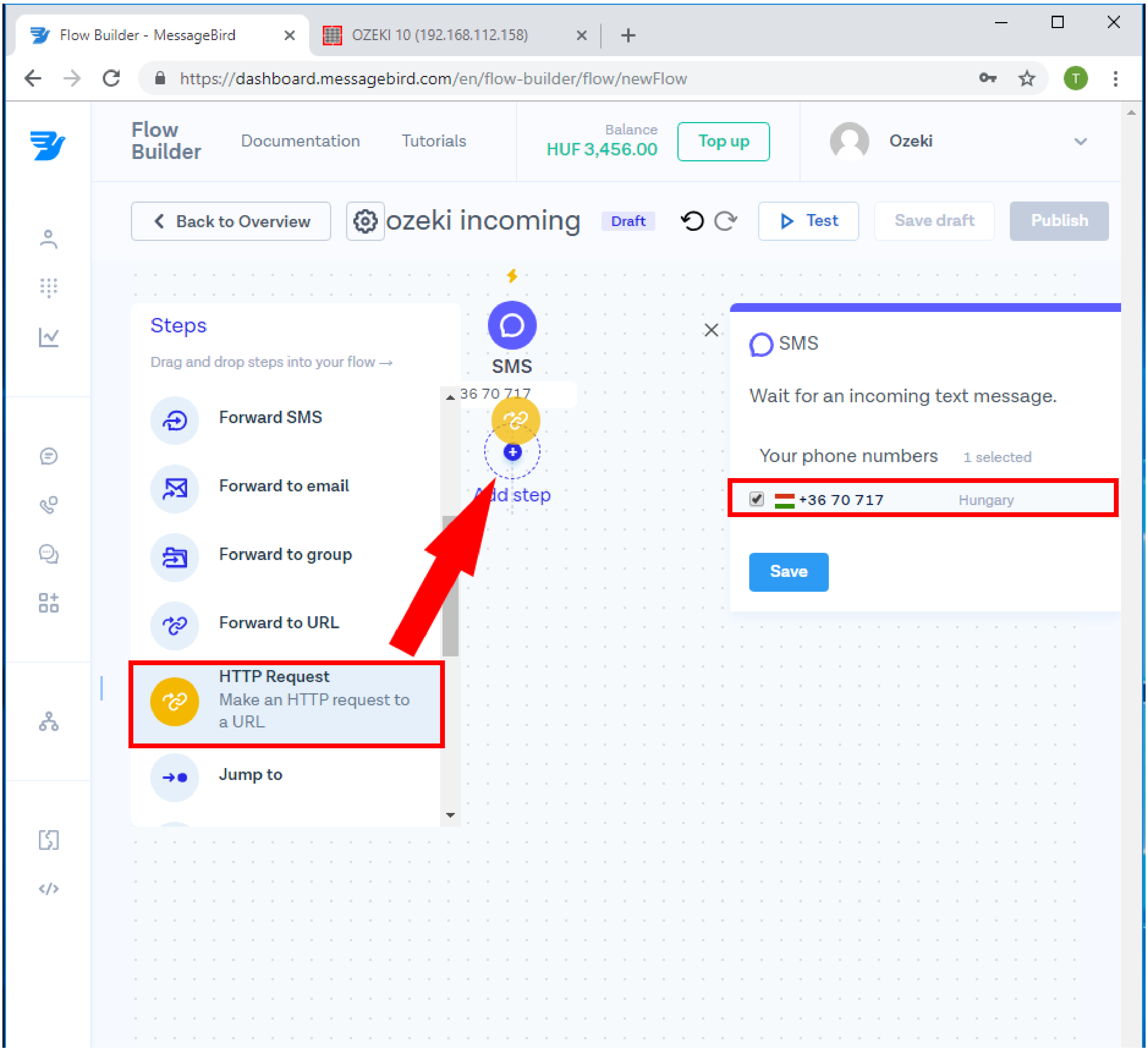
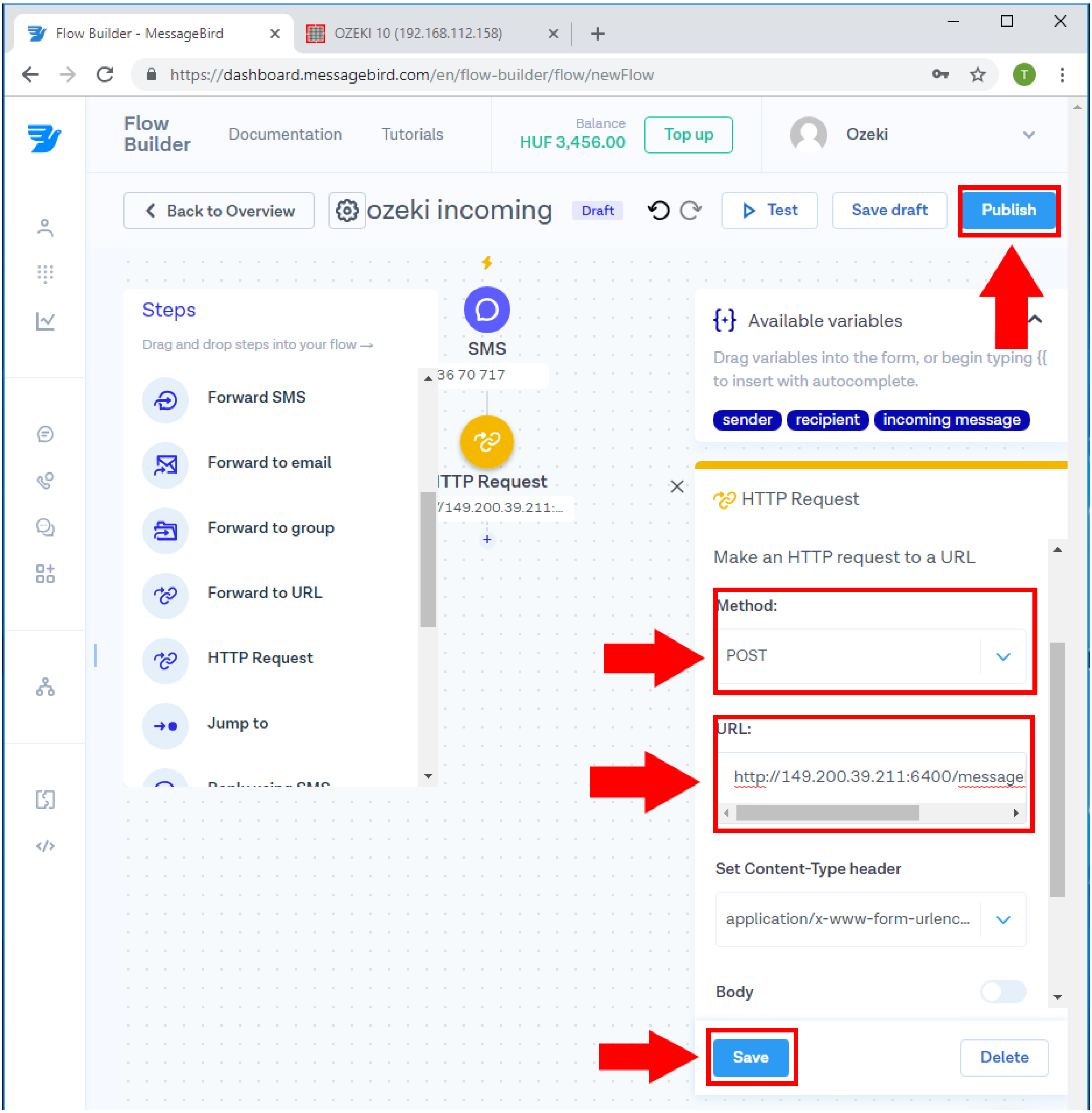


Figure 28 - Add HTTP Request

In the HTTP Request select the **POST Method** and paste the **URL** from the Ozeki 10. Finally click on teh Save and Publish.



Method: POST

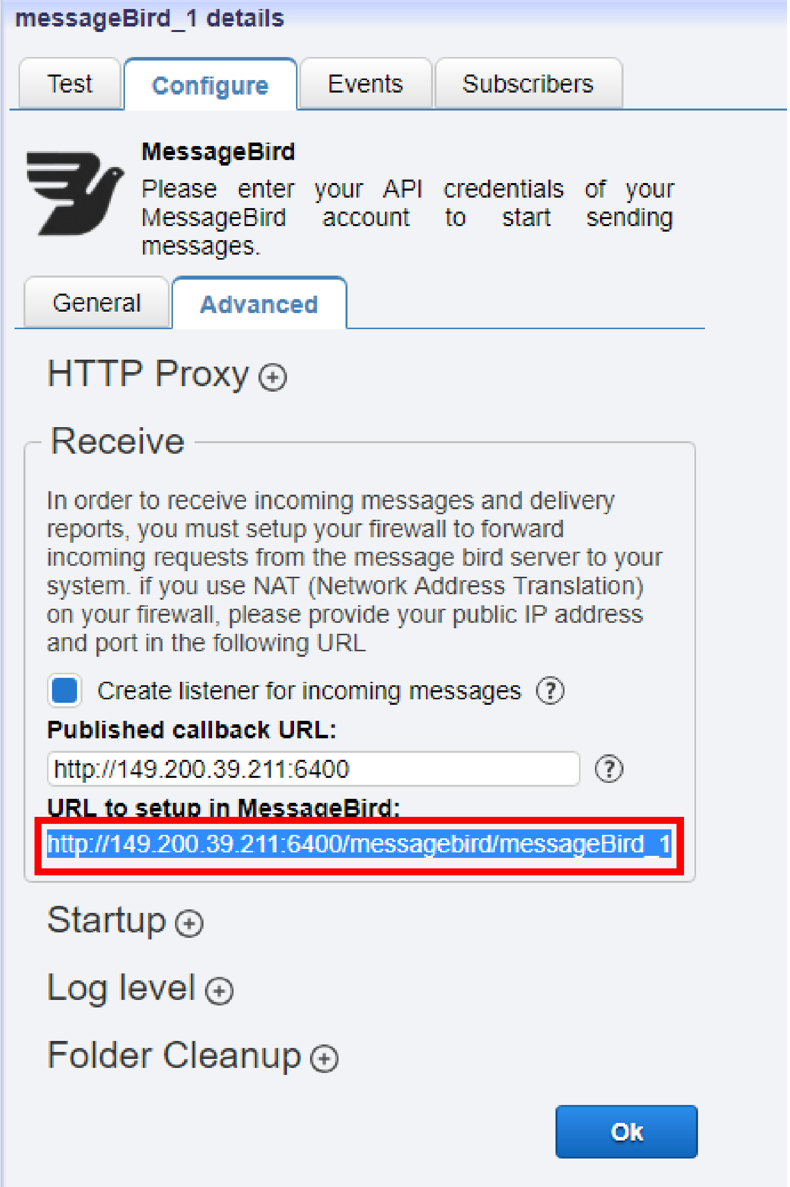
URL: http://149.200.39.211:6400/message

Set Content-Type header: application/x-www-form-urlenc...

Body

Figure 29 - Set up HTTP Request

You can find the URL on the Advanced tab's Receive section.



messageBird_1 details

**MessageBird**

Please enter your API credentials of your MessageBird account to start sending messages.

Receive

In order to receive incoming messages and delivery reports, you must setup your firewall to forward incoming requests from the message bird server to your system. if you use NAT (Network Address Translation) on your firewall, please provide your public IP address and port in the following URL

■ Create listener for incoming messages ?

**Published callback URL:** http://149.200.39.211:6400

**URL to setup in MessageBird:** http://149.200.39.211:6400/messagebird/messageBird_1

Figure 30 - Callback URL

After the Publish you can see the new Flow in the Flow builder.

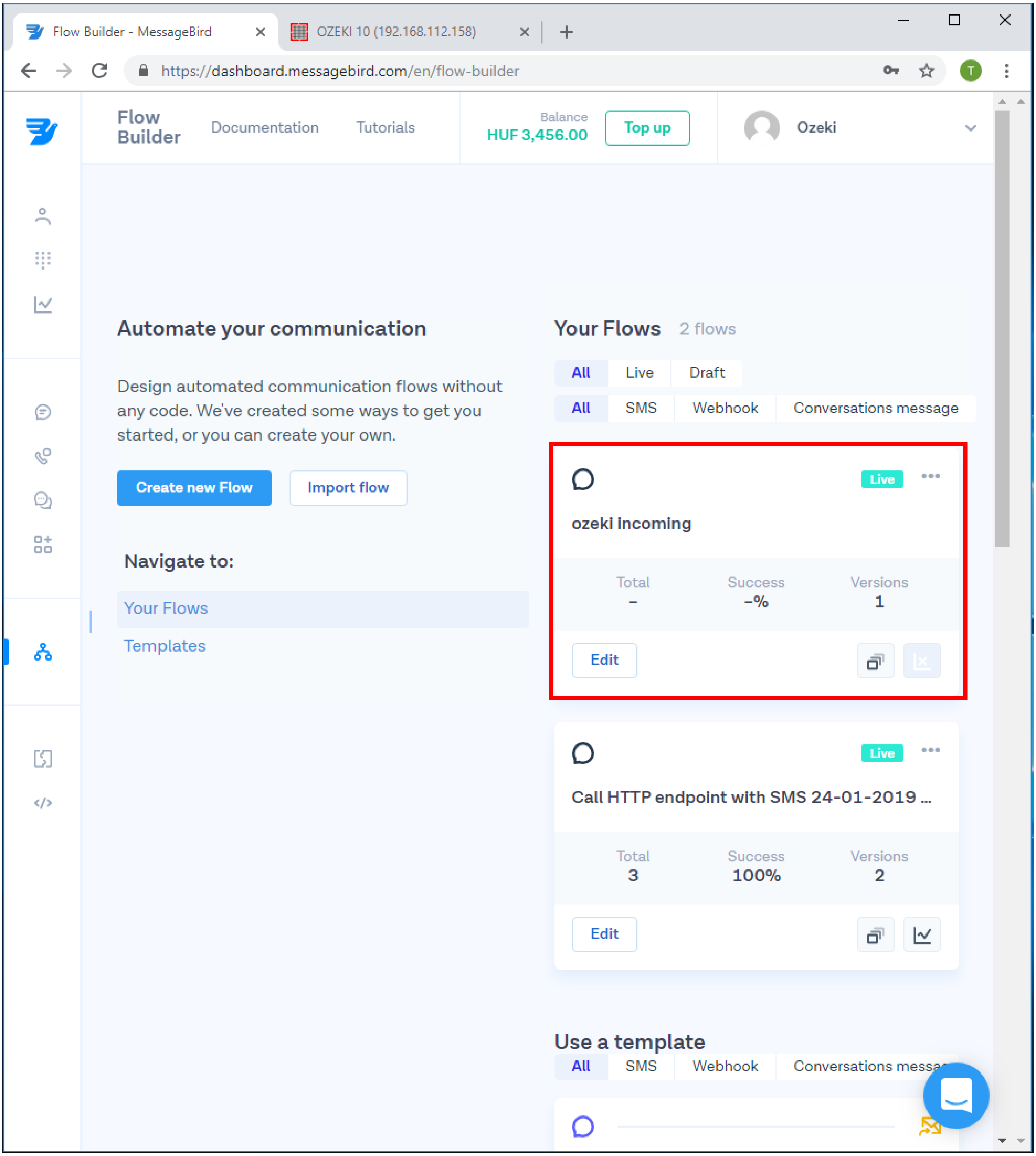
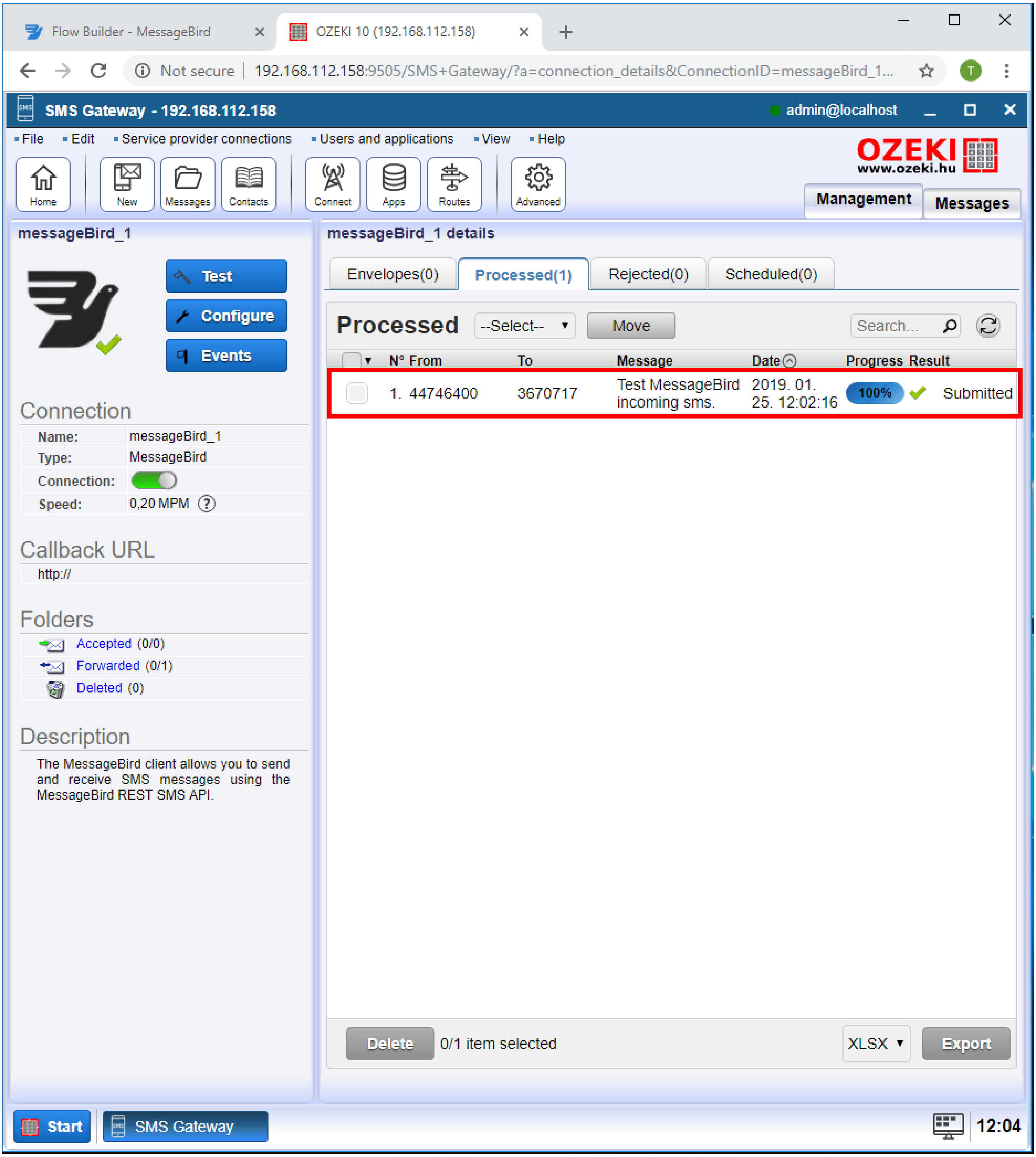


Figure 31 - Flow created

Send a Test Message to this number and you will see the message received in Ozeki 10.



messageBird_1

Connection

Name: messageBird_1
Type: MessageBird
Connection:
Speed: 0,20 MPM

Callback URL

http://

Folders

Accepted (0/0)
Forwarded (0/1)
Deleted (0)

Description

The MessageBird client allows you to send and receive SMS messages using the MessageBird REST SMS API.

messageBird_1 details

Processed

| N° | From | To | Message | Date | Progress | Result |
|---|---|---|---|---|---|---|
| 1. | 44746400 | 3670717 | Test MessageBird incoming sms. | 2019. 01. 25. 12:02:16 | 100% ✓ | Submitted |

0/1 item selected

Figure 32 - Test message received

# User guide

***Ozeki SMS Gateway can manage almost every SMS messaging solution. This page overviews of some of the problems Ozeki SMS Gateway can solve, so you can reach efficient and convenient SMS flow. You can find great ideas to reduce costs and handle your preferred connections.***

**Standard User**

With Ozeki SMS Gateway Standard user you can log in to the grafical user interface and you are able to send SMS. And you can investigate the details of the delivery of the message.

Learn More

**Autoreply SMS**

Ozeki SMS Gateway has 3 type of Autoreply Users which are used for automatically replying SMS messages or forwarding them to any recipient. These users can work in 3 simple ways. You will see how to install and configure these users on Ozeki SMS Gateway to serve your needs.

Learn More

**E-mail to SMS**

Ozeki SMS Gateway offers various methods to setup E-mail to SMS functionality. You can use IMAP, POP3 or SMTP to send and receive E-mails and convert them to and from SMS messages. You can put the phone number(s) into the subject line of the e-mails or you may send an e-mail to an address containing the phone nubmer, such as +441234657@smsgw.yourcompany.com

Learn More

**SMS from/to Email through your Email Account**

Ozeki SMS Gateway's E-mail User can be used for sending or downloading emails from a mailbox. For sending emails it should connect to your email account's SMTP server. For downloading emails from a mailbox it should connect to the POP3 server through your email account.

Learn More

**SMS from/to File**

Ozeki SMS Gateway's File User is capable to send and receive SMS messages in different file formats. Simple, Text, Verbose, List, CSV, XML, SAP, ATF and KAL file formats are supported by the Ozeki SMS Gateway's File User. Your application can place these files in proper directories to send SMS messages.

Learn More

**SMS from/to FTP**

Ozeki SMS Gateway's FTP to SMS Extended User can synchronize directories with SMS Gateway through standard FTP, FTPS or SFTP protocols. Incoming SMS messages will be uploaded and outgoing SMS messages will be downloaded from the FTP server. You can also look at the accepted file formats.

Learn More

**Start your Applications with SMS**

Ozeki SMS Gateway's Application Starter User can run any process or application in case an SMS message arrives. You simply need to provide the file path. You can also fetch parameters from received SMS messages to use them as process parameters or command line arguments.

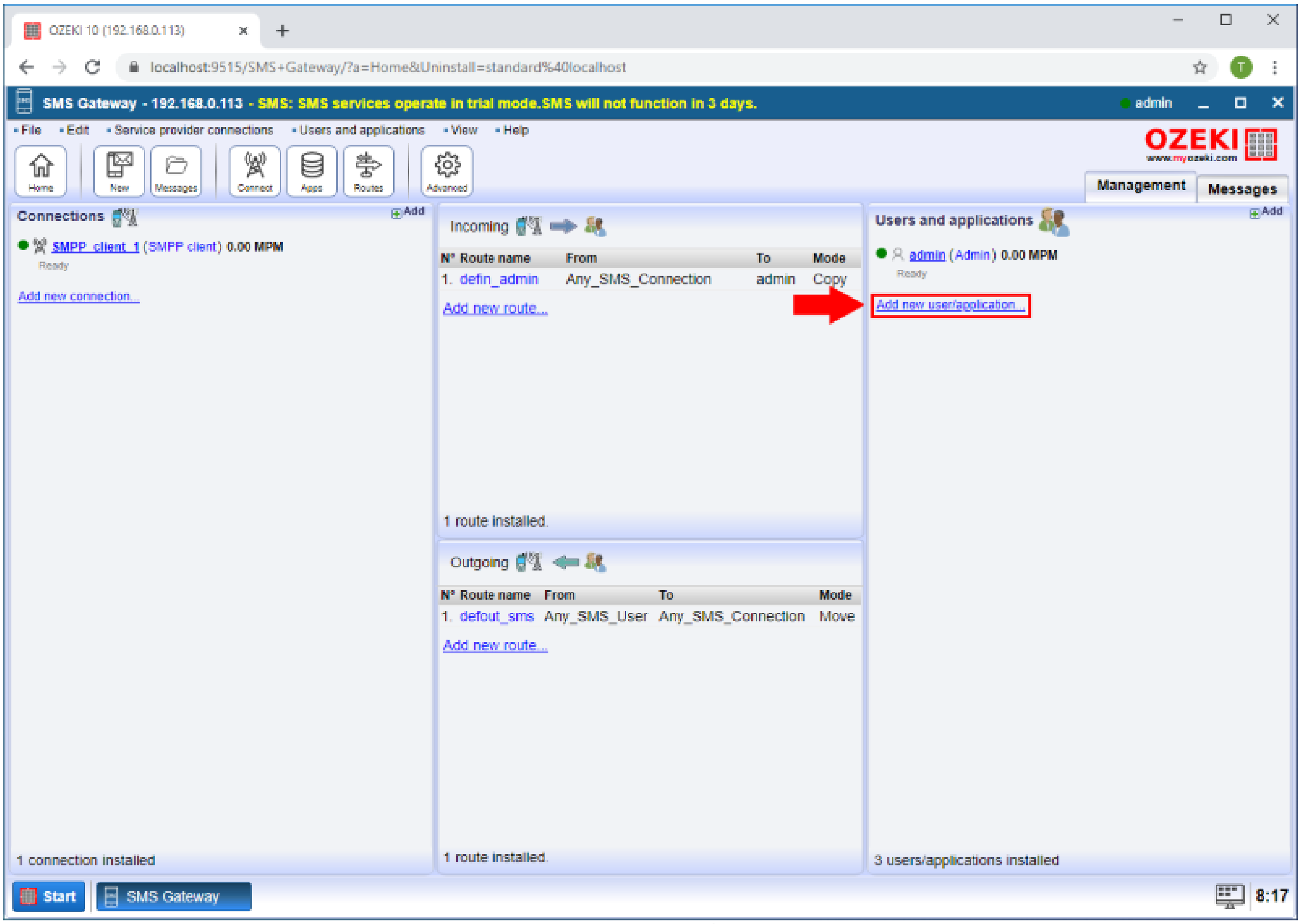
Learn More

# How create a standard SMS user account

***This chapter explains how to install and configure a standard user for SMS messaging using the graphical user interface of the Ozeki SMS Gateway software.***

## Step 1 - Add new user or application

To start installing and configuring a standard user, click the Add new user/application... link on the right of the Management Console(Figure 1).



localhost:9515/SMS+Gateway/?a=Home&Uninstall=standard%40localhost

SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.SMS will not function in 3 days.

Connections

SMPP_client_1 (SMPP client) 0.00 MPM
Ready

Add new connection...

1 connection installed

Incoming

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defin_admin | Any_SMS_Connection | admin | Copy |

Add new route...

1 route installed.

Outgoing

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defaut_sms | Any_SMS_User | Any_SMS_Connection | Move |

Add new route...

1 route installed.

Users and applications

admin (Admin) 0.00 MPM
Ready

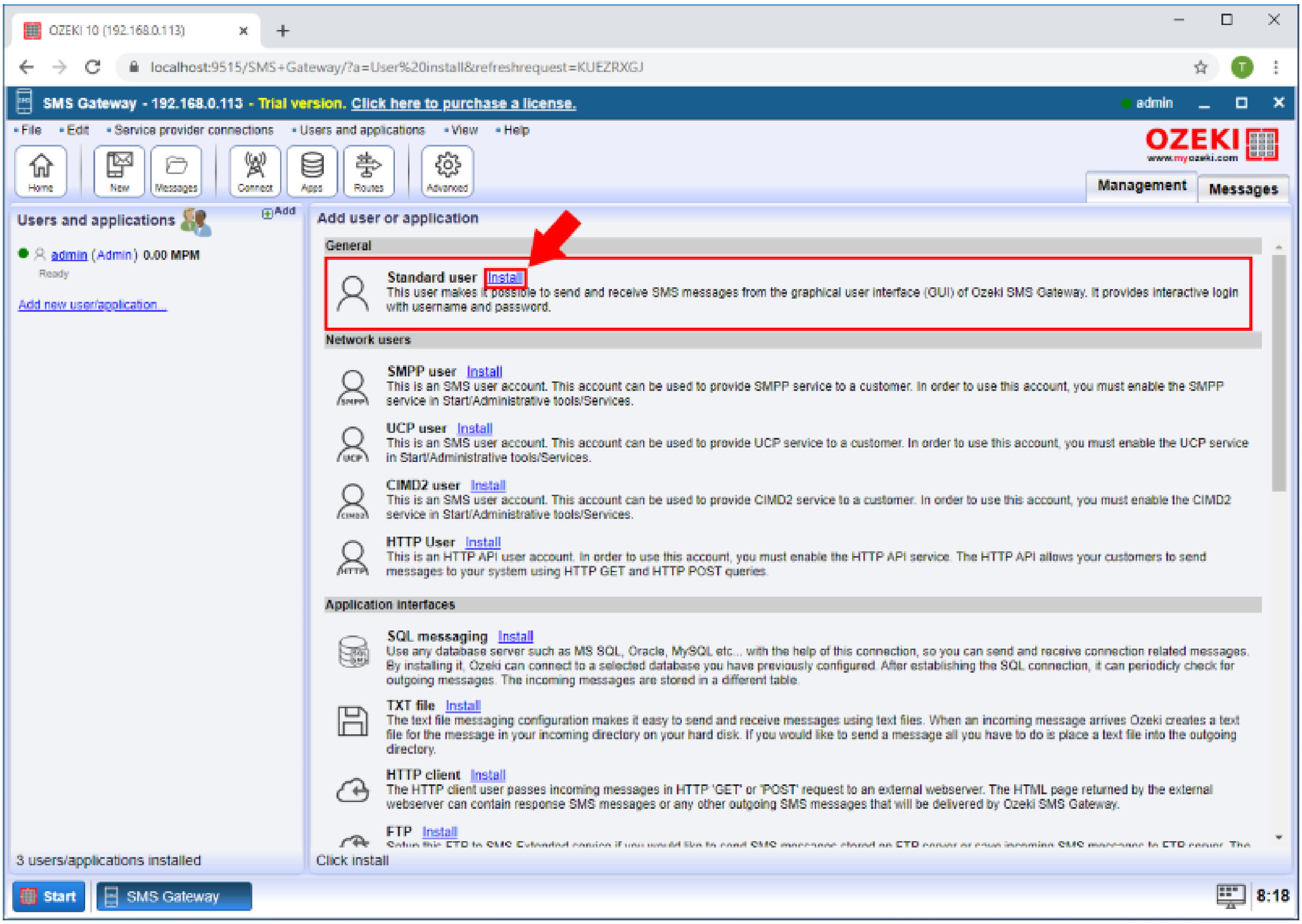
Add new user/application...

3 users/applications installed

Figure 1 - Add new user/application...

## Step 2 - Install Standard user

An interface will open consisting of two panels. The left side panel contains the already installed users and applications. The right side panel contains the users and applications you can install with a brief description next to them. Search the Standard User and click the blue 'install' button next to it (Figure 2).



Standard user Install
This user makes it possible to send and receive SMS messages from the graphical user interface (GUI) of Ozeki SMS Gateway. It provides interactive login with username and password.

SMPP user Install
This is an SMS user account. This account can be used to provide SMPP service to a customer. In order to use this account, you must enable the SMPP service in Start/Administrative tools/Services.

UCP user Install
This is an SMS user account. This account can be used to provide UCP service to a customer. In order to use this account, you must enable the UCP service in Start/Administrative tools/Services.

CIMD2 user Install
This is an SMS user account. This account can be used to provide CIMD2 service to a customer. In order to use this account, you must enable the CIMD2 service in Start/Administrative tools/Services.

HTTP User Install
This is an HTTP API user account. In order to use this account, you must enable the HTTP API service. The HTTP API allows your customers to send messages to your system using HTTP GET and HTTP POST queries.

SQL messaging Install
Use any database server such as MS SQL, Oracle, MySQL etc... with the help of this connection, so you can send and receive connection related messages. By installing it, Ozeki can connect to a selected database you have previously configured. After establishing the SQL connection, it can periodicly check for outgoing messages. The incoming messages are stored in a different table.

TXT file Install
The text file messaging configuration makes it easy to send and receive messages using text files. When an incoming message arrives Ozeki creates a text file for the message in your incoming directory on your hard disk. If you would like to send a message all you have to do is place a text file into the outgoing directory.

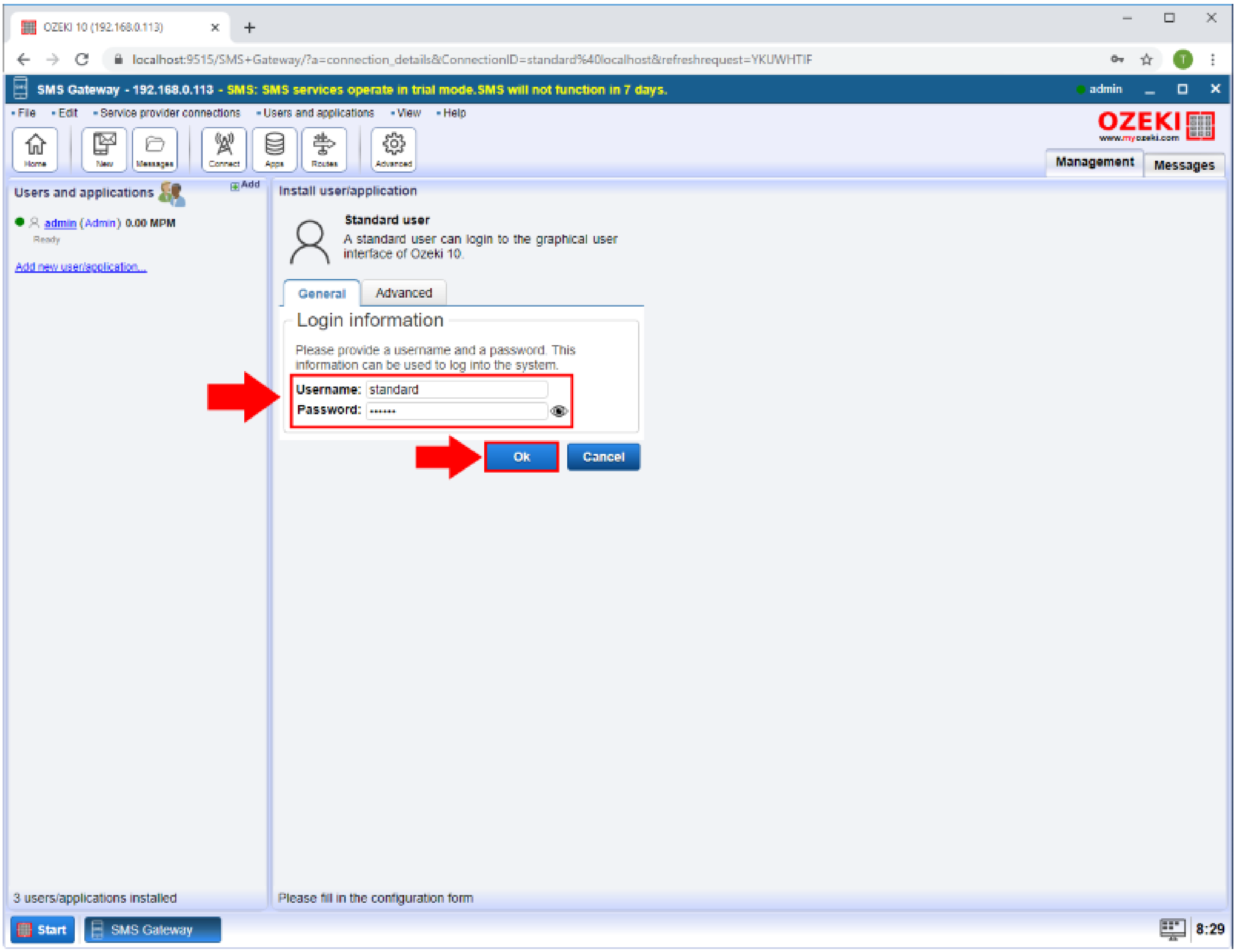
HTTP client Install
The HTTP client user passes incoming messages in HTTP 'GET' or 'POST' request to an external webserver. The HTML page returned by the external webserver can contain response SMS messages or any other outgoing SMS messages that will be delivered by Ozeki SMS Gateway.

FTP Install
Setup this FTP to SMS Extended service if you would like to send SMS messages stored on FTP server or save incoming SMS messages to FTP server. The

Figure 2 - Install Standard user

# Step 3 - Configure username and password

Clicking the Install link will bring up the Standard user installation panel. Here, you need to enter a unique username in the Username field and a password in the Passeowd filed (Figure 3).



SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.

Install user/application

**Standard user**

A standard user can login to the graphical user interface of Ozeki 10.

Login information

Please provide a username and a password. This information can be used to log into the system.

Username: standard

Password: ••••••

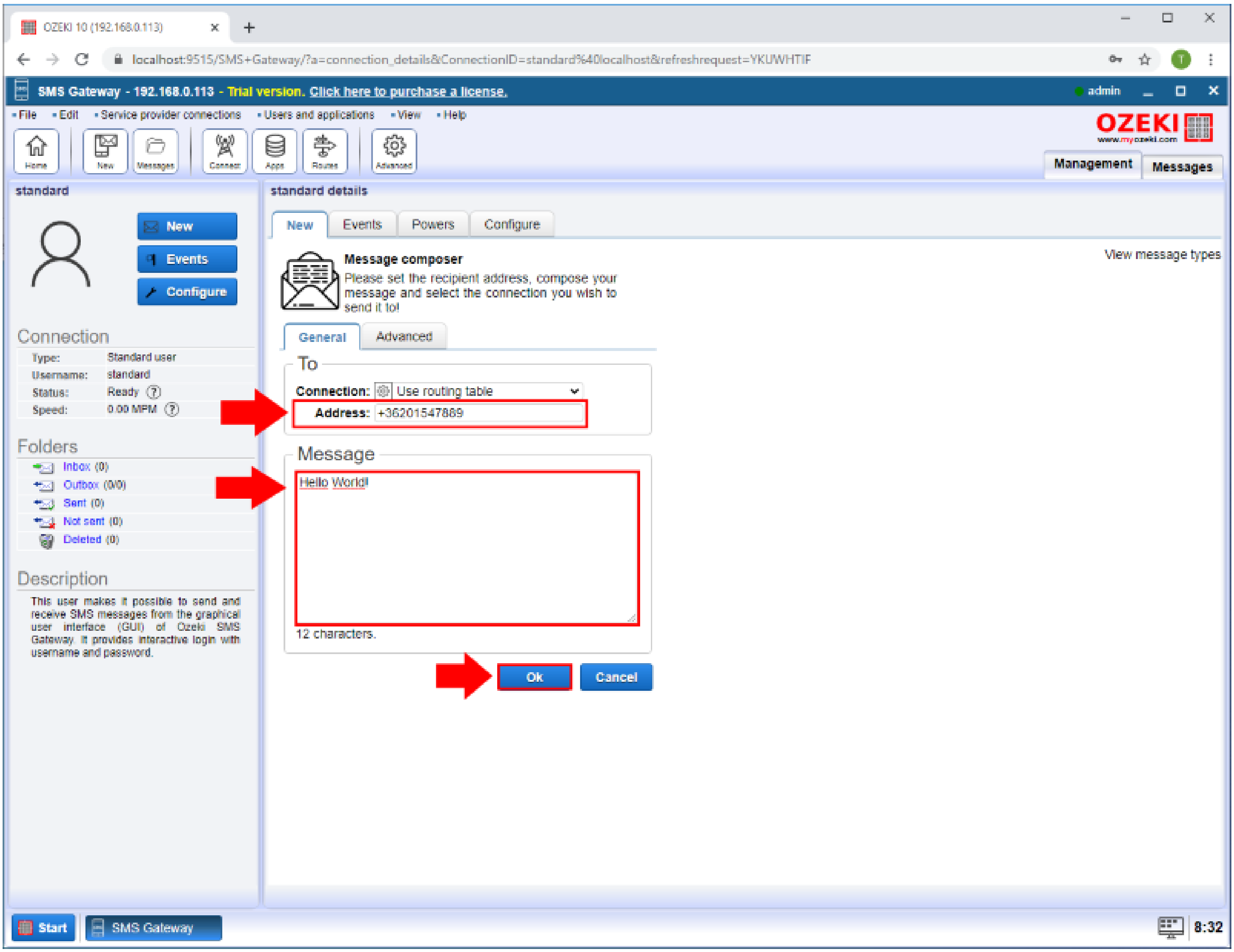
3 users/applications installed

Please fill in the configuration form

Figure 3 - Configure username and password

## Step 4 - Send test message

From the Standard user GUI you can send SMS message. Provide the recipient address, the message and click on the OK button (Figure 4).


SMS Gateway - 192.168.0.113 - Trial version. Click here to purchase a license.
standard details
New
Events
Powers
Configure
Message composer
Please set the recipient address, compose your message and select the connection you wish to send it to!
View message types
General
Advanced
To
Message
Ok
Cancel

Connection: Use routing table
Address: +36201547889

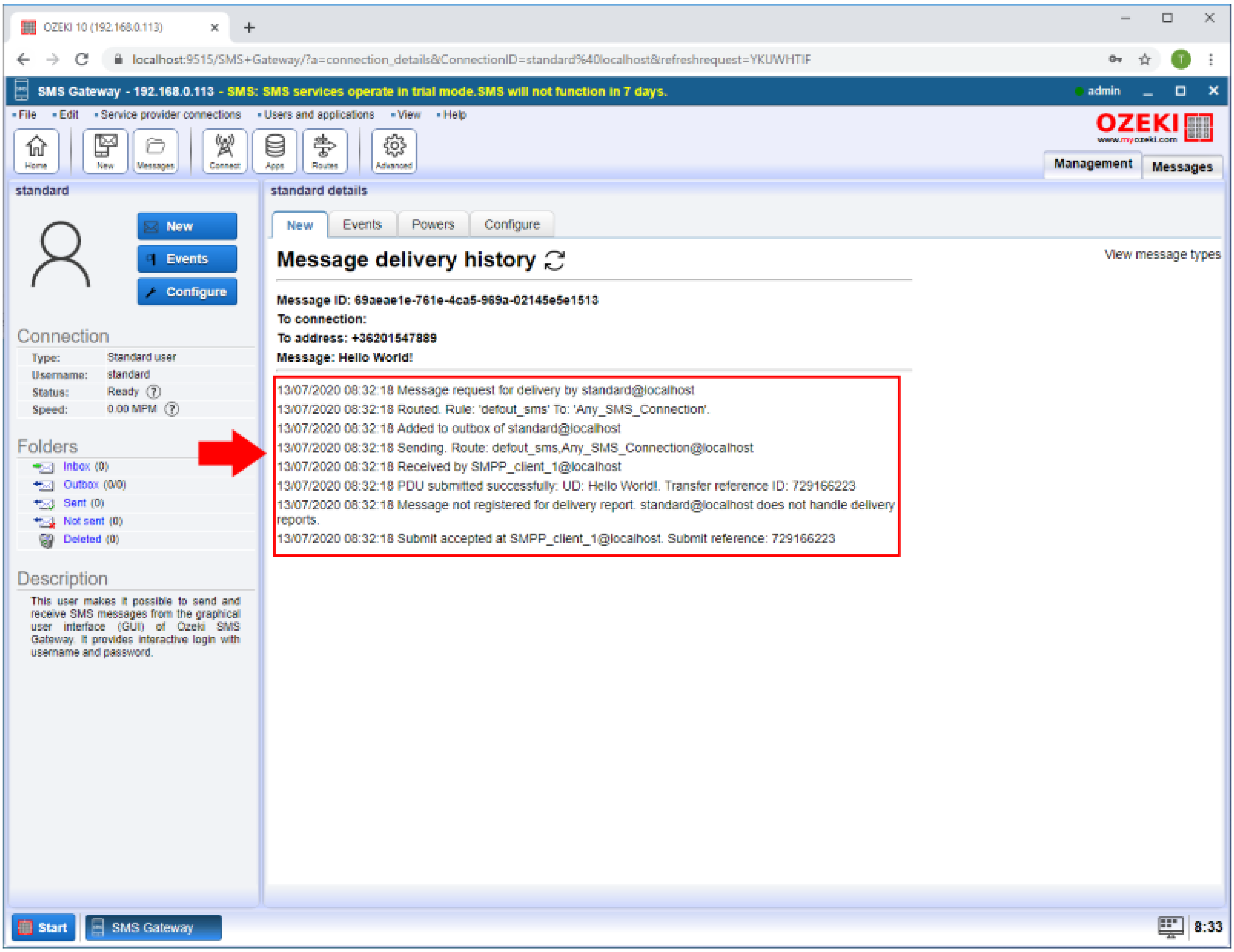
Hello World!

12 characters.

Figure 4 - Send test message

# Step 5 - Message sent

After the message is sent you can see the message history for it (Figure 5).



| | |
|---|---|
| Type: | Standard user |
| Username: | standard |
| Status: | Ready |
| Speed: | 0.00 MPM |

This user makes it possible to send and receive SMS messages from the graphical user interface (GUI) of Ozeki SMS Gateway. It provides interactive login with username and password.

Message delivery history

Message ID: 69aeae1e-761e-4ca5-969a-02145e5e1513
To connection:
To address: +36201547889
Message: Hello World!

```
13/07/2020 08:32:18 Message request for delivery by standard@localhost
13/07/2020 08:32:18 Routed. Rule: 'defout_sms' To: 'Any_SMS_Connection'.
13/07/2020 08:32:18 Added to outbox of standard@localhost
13/07/2020 08:32:18 Sending. Route: defout_sms,Any_SMS_Connection@localhost
13/07/2020 08:32:18 Received by SMPP_client_1@localhost
13/07/2020 08:32:18 PDU submitted successfully: UD: Hello World!. Transfer reference ID: 729166223
13/07/2020 08:32:18 Message not registered for delivery report. standard@localhost does not handle delivery
reports.
13/07/2020 08:32:18 Submit accepted at SMPP_client_1@localhost. Submit reference: 729166223
```

Figure 5 - Message sent

# Autoreply SMS messages

***Ozeki SMS Gateway can be configured to replay to incoming SMS messages automatically. To achieve this goal, it offers multiple options. The autoreply easy option will return a response SMS to every message received. The Autoreply by script option allows you to write a script, and create a reply and to create and send new messages when an incomnig SMS arrives.***

**Autoreply Easy**
Ozeki SMS Gateway's Autoreply Easy User can reply to every incoming SMS message with a text of your choice. A txt file stores the text, so you can modify it from your own application or from Ozeki SMS Gateway. This guide explains how to install and configure your Autoreply Easy User.
Read about Ozeki SMS Gateway's Autoreply Easy User

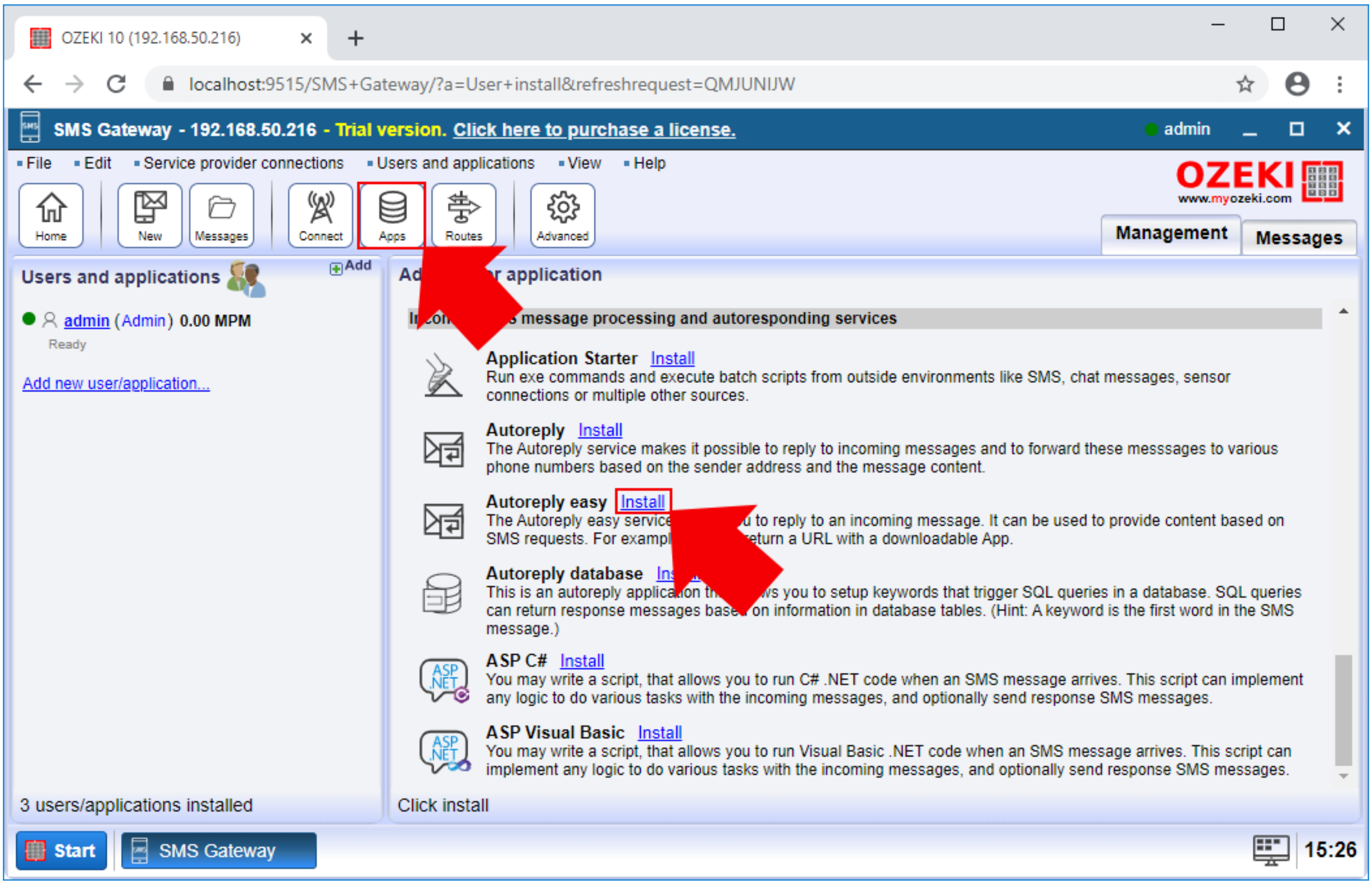
**Autoreply by Script**
Ozeki SMS Gateway's Autoreply User can automatically reply to SMS messages or forward incoming messages. This guide explains how to install and configure an Autoreply User on Ozeki SMS Gateway. Each Autoreply User is controlled by their own script. You will see example scripts too by reading this guide.
Read about Ozeki SMS Gateway's Autoreply User

# The most simple way to send SMS replies automatically

***This guide demonstrates how easy you can create an autoreply service using Ozeki SMS Gateway. This autoreply service is capable of sending predefined SMS reply messages to the sender in case the system receives a message. This document demonstrates the process of creating such an autoreply service which takes just a few clicks. So let's begin right now.***

## Step 1 - Create an Autoreply Easy connection

The first step of this guide is to create the connection for the autoreply service in SMS Gateway. So first, open the SMS Gateway, and click on the Apps button on the toolbar. Here, in this menu, scroll down to the Incoming SMS message processing and autoresponding services submenu, and like in Figure 1, select Autoreply easy by clicking on Install.


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=User+install&refreshrequest=QMJUNIJW
SMS Gateway - 192.168.50.216 - Trial version. Click here to purchase a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
OZEKI
www.myozeki.com
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Users and applications
Add
admin (Admin) 0.00 MPM
Ready
Add new user/application...
3 users/applications installed
Ad r application
In co s message processing and autoresponding services
Click install
Start
SMS Gateway
15:26

**Application Starter** Install
Run exe commands and execute batch scripts from outside environments like SMS, chat messages, sensor connections or multiple other sources.

**Autoreply** Install
The Autoreply service makes it possible to reply to incoming messages and to forward these messsages to various phone numbers based on the sender address and the message content.

**Autoreply easy** Install
The Autoreply easy service u to reply to an incoming message. It can be used to provide content based on SMS requests. For exampl eturn a URL with a downloadable App.

**Autoreply database** Ins
This is an autoreply application th ws you to setup keywords that trigger SQL queries in a database. SQL queries can return response messages base on information in database tables. (Hint: A keyword is the first word in the SMS message.)

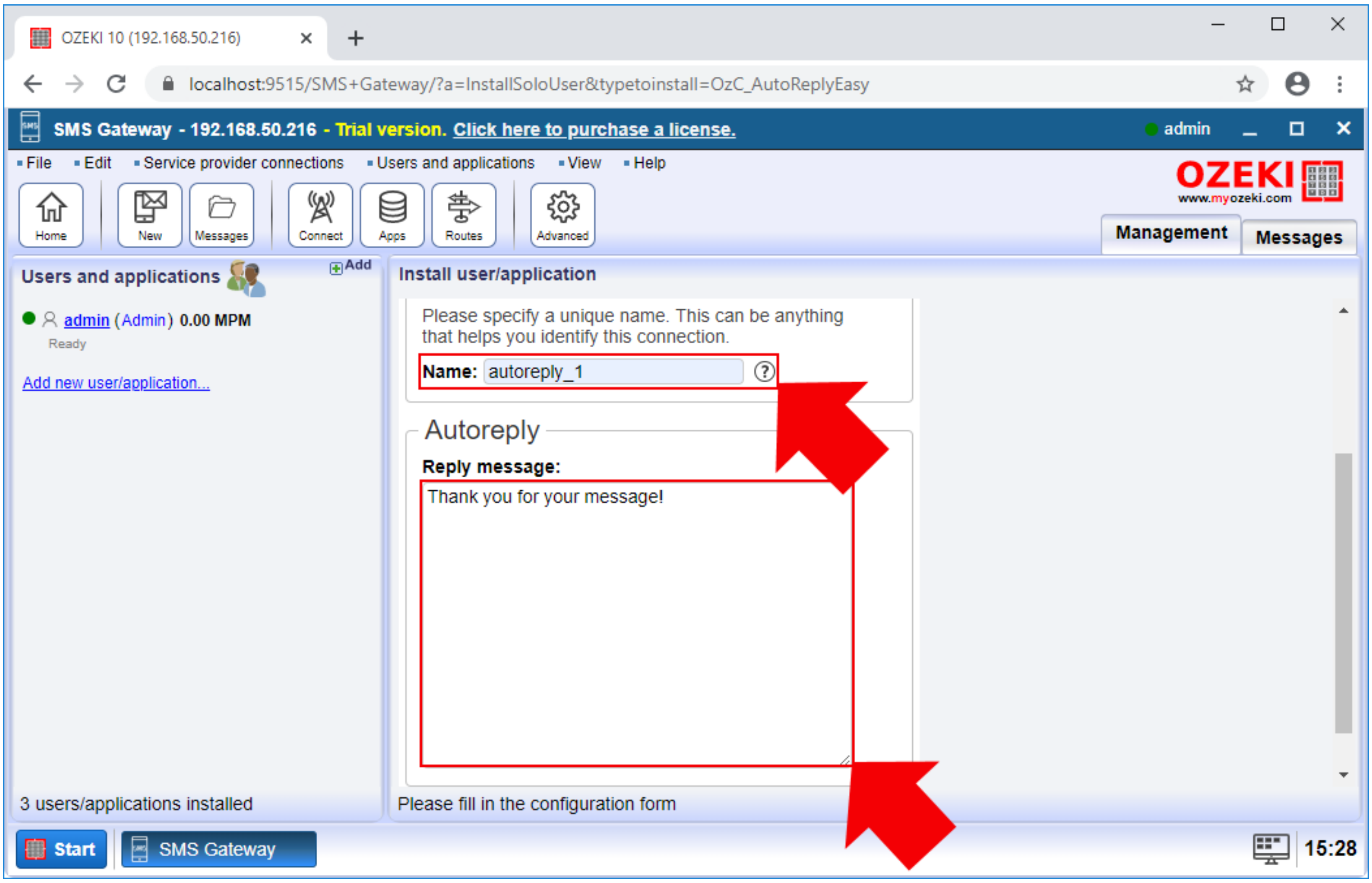
**ASP C#** Install
You may write a script, that allows you to run C# .NET code when an SMS message arrives. This script can implement any logic to do various tasks with the incoming messages, and optionally send response SMS messages.

**ASP Visual Basic** Install
You may write a script, that allows you to run Visual Basic .NET code when an SMS message arrives. This script can implement any logic to do various tasks with the incoming messages, and optionally send response SMS messages.

Figure 1 - Create the Autoreply easy connection

## Step 2 - Configure the Autoreply easy connection

The configuration of the Autoreply easy connection is quite easy, all you have to do here is to provide some basic details for the connection. The first thing, that you need to enter is the name of the connection. With this name, the connection can be identified in the SMS Gateway. Next, you need to enter the reply message itself (Figure 2), which will be sent as a reply message when someone sends a text message to you.


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=InstallSoloUser&typetoinstall=OzC_AutoReplyEasy
SMS Gateway - 192.168.50.216 - Trial version. Click here to purchase a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Users and applications
Add
admin (Admin) 0.00 MPM
Ready
Add new user/application...
3 users/applications installed
Start
SMS Gateway
15:28

Install user/application

Please specify a unique name. This can be anything that helps you identify this connection.

Name: autoreply_1

Autoreply

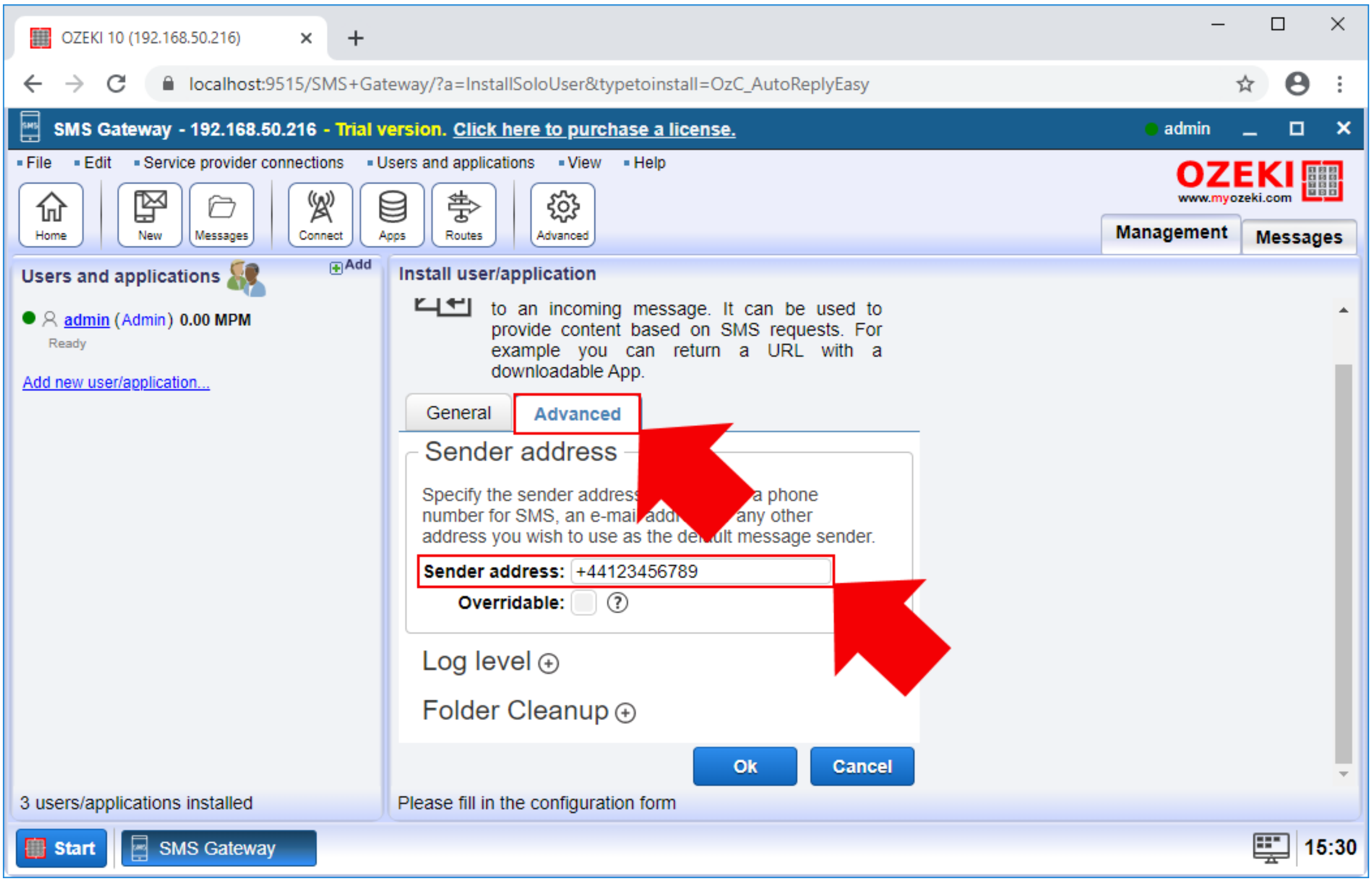
Reply message: Thank you for your message!

Please fill in the configuration form

Figure 2 - Configure the Autoreply easy connection

# Step 3 - Advanced configuration of Autoreply easy

The Autoreply easy connection can be further configured by setting a default sender address. This address will show up at the recipient side who receives the reply message. As Figure 3 shows, you can type here a phone number or an e-mail address which helps to identify the autoreply easy connection. If you finished the configuration, just click on OK.


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=InstallSoloUser&typetoinstall=OzC_AutoReplyEasy
SMS Gateway - 192.168.50.216 - Trial version. Click here to purchase a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Users and applications
Add
admin (Admin) 0.00 MPM
Ready
Add new user/application...
3 users/applications installed
Install user/application
General
Advanced
Log level
Folder Cleanup
Ok
Cancel
Start
SMS Gateway
15:30

to an incoming message. It can be used to provide content based on SMS requests. For example you can return a URL with a downloadable App.

Sender address

Specify the sender address, for example a phone number for SMS, an e-mail address or any other address you wish to use as the default message sender.

Sender address: +44123456789

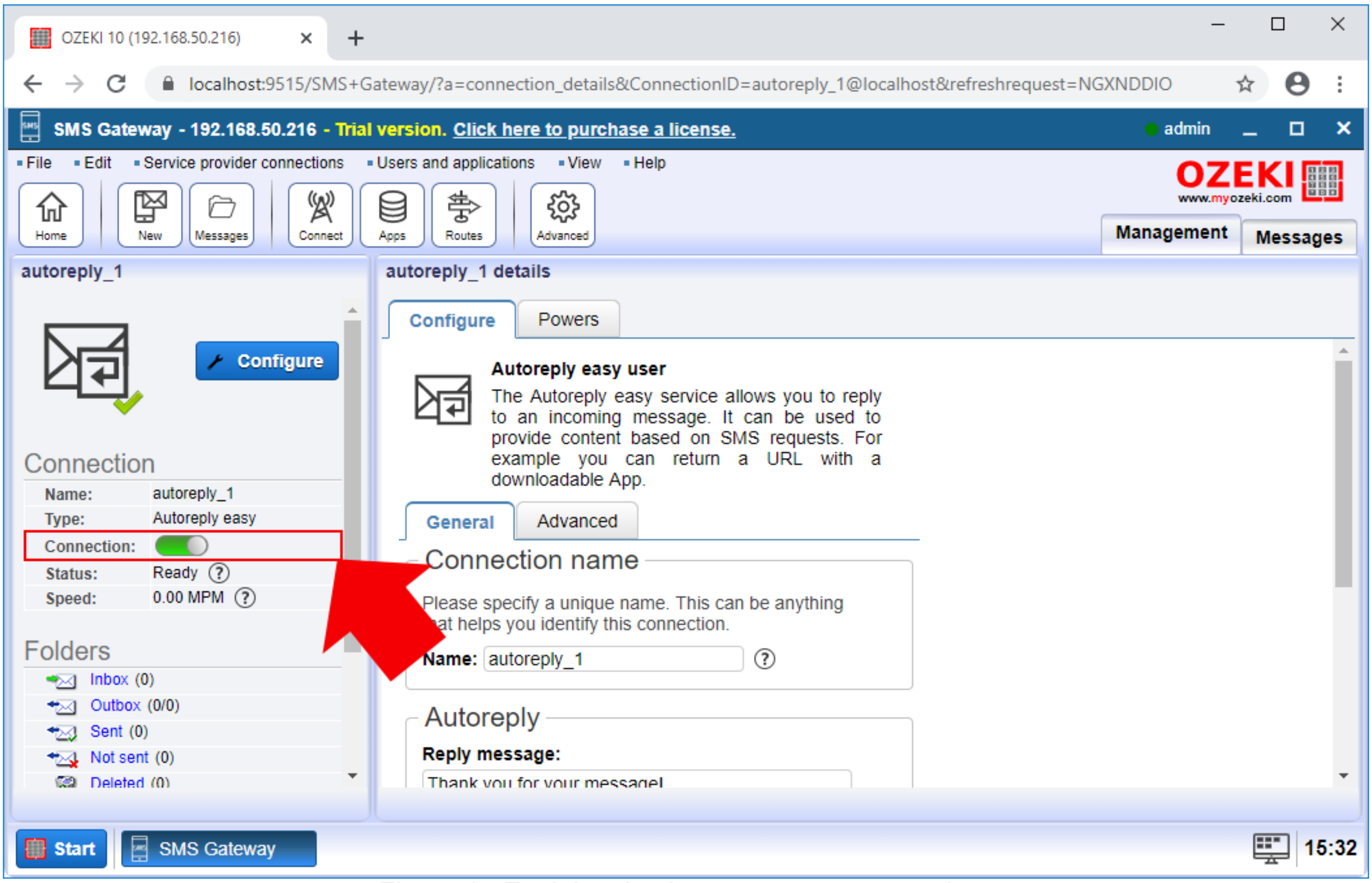
Overridable: □

Please fill in the configuration form

Figure 3 - Advanced configuration of the Autoreply easy connection

# Step 4 - Enable the Autoreply easy connection

After you created the Autoreply easy connection, the next window is the main menu of the autoreply service. Here, you can modify the details of the connection anytime you want. To activate the service, you need to enable the connection here. For that, just click on the toggle as Figure 4 shows. The toggle is green now that indicates that the autoreply service is now active.



Name: autoreply_1
Type: Autoreply easy
Connection:
Status: Ready
Speed: 0.00 MPM

**Autoreply easy user**

The Autoreply easy service allows you to reply to an incoming message. It can be used to provide content based on SMS requests. For example you can return a URL with a downloadable App.

Connection name

Please specify a unique name. This can be anything that helps you identify this connection.

**Name:** autoreply_1

Autoreply

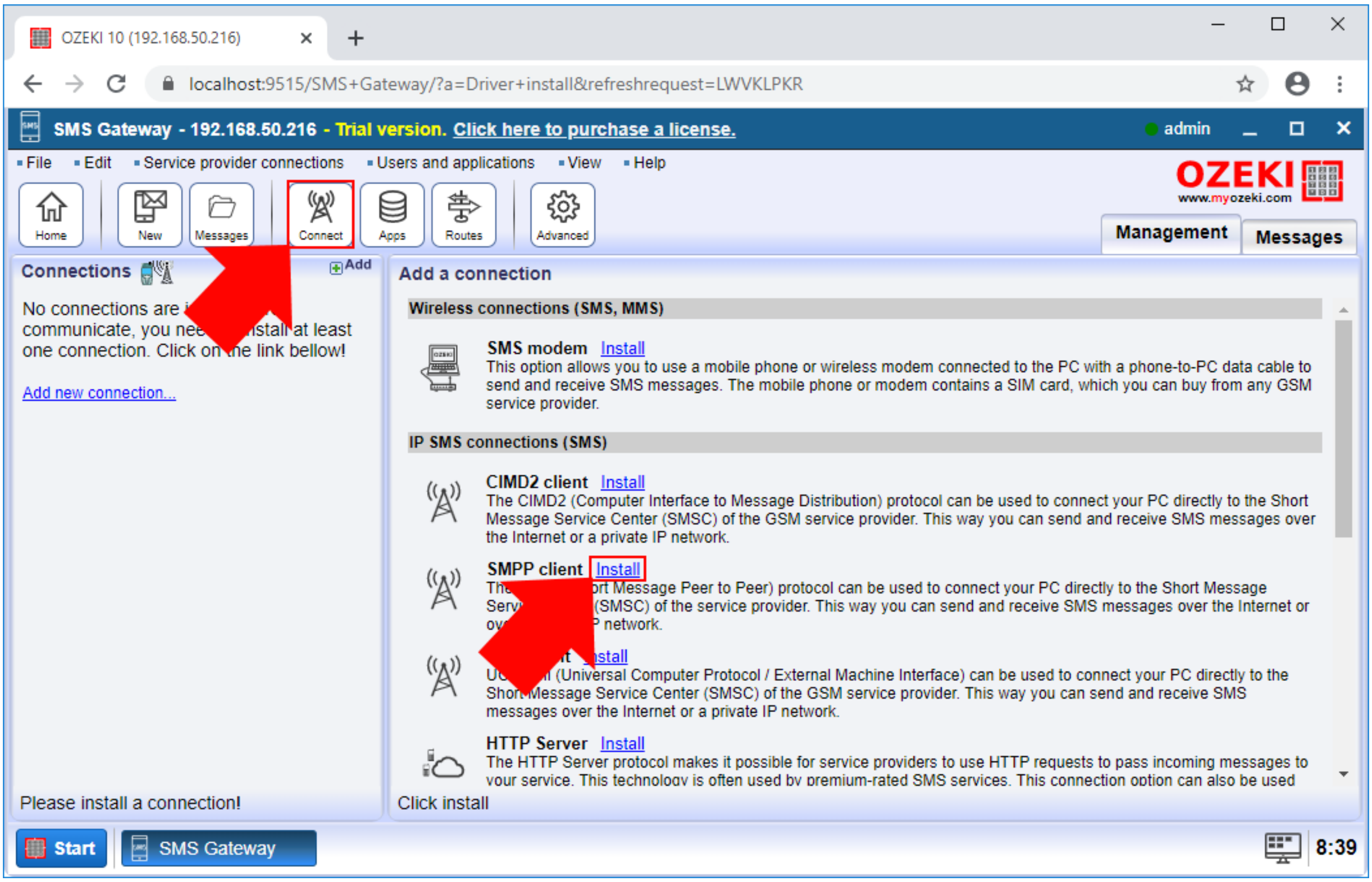
**Reply message:**

Thank you for your message!

Figure 4 - Enabling the Autoreply easy connection

## Step 5 - Connect to the mobile network

So far, you created and enabled your autoreply service, but to receive messages, you also need a connection that can connect to the mobile network and forward messages to your Ozeki system. To do that, click on Connect on the toolbar, and select the SMPP client as you can see it in Figure 5.


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=Driver+install&refreshrequest=LWVKLPKR
SMS Gateway - 192.168.50.216 - Trial version. Click here to purchase a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
OZEKI
www.myozeki.com
Management
Messages
Connections
Add
Start
SMS Gateway
8:39

No connections are i... communicate, you nee... install at least one connection. Click on the link bellow!

Add new connection...

Please install a connection!

Add a connection

Wireless connections (SMS, MMS)

SMS modem Install
This option allows you to use a mobile phone or wireless modem connected to the PC with a phone-to-PC data cable to send and receive SMS messages. The mobile phone or modem contains a SIM card, which you can buy from any GSM service provider.

IP SMS connections (SMS)

CIMD2 client Install
The CIMD2 (Computer Interface to Message Distribution) protocol can be used to connect your PC directly to the Short Message Service Center (SMSC) of the GSM service provider. This way you can send and receive SMS messages over the Internet or a private IP network.

SMPP client Install
The ...rt Message Peer to Peer) protocol can be used to connect your PC directly to the Short Message Serv... (SMSC) of the service provider. This way you can send and receive SMS messages over the Internet or ov... P network.

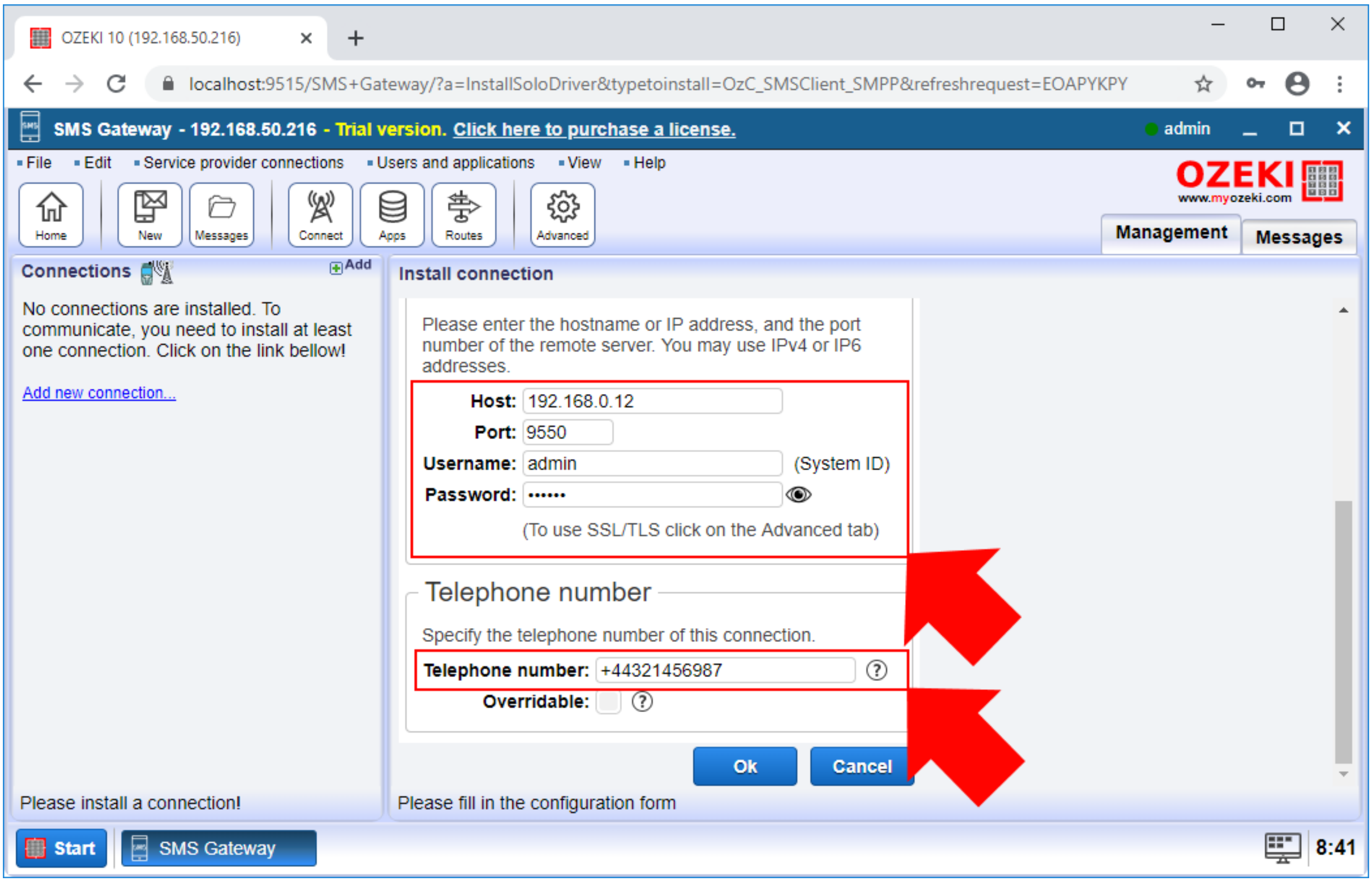
...t ...stall
UC... (Universal Computer Protocol / External Machine Interface) can be used to connect your PC directly to the Short Message Service Center (SMSC) of the GSM service provider. This way you can send and receive SMS messages over the Internet or a private IP network.

HTTP Server Install
The HTTP Server protocol makes it possible for service providers to use HTTP requests to pass incoming messages to your service. This technology is often used by premium-rated SMS services. This connection option can also be used

Click install

Figure 5 - Create an SMPP client connection

In the configuration menu, you have to provide some details about the server. This is the server, that this SMPP client is going to use to handle SMS messages. So here, you need to type the host, port and user account details. Then, you need to specify a telephone number for this connection (Figure 6). With this number, the connection can be identified, and able to send or receive messages. Lastly, you just need zo click on OK to create the SMPP client connection.



OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=InstallSoloDriver&typetoinstall=OzC_SMSClient_SMPP&refreshrequest=EOAPYKPY
SMS Gateway - 192.168.50.216 - Trial version. Click here to purchase a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Connections
Add
Install connection
Add new connection...
Ok
Cancel
Start
SMS Gateway
8:41

No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Please enter the hostname or IP address, and the port number of the remote server. You may use IPv4 or IP6 addresses.

Host: 192.168.0.12
Port: 9550
Username: admin (System ID)
Password: ••••••
(To use SSL/TLS click on the Advanced tab)

Telephone number

Specify the telephone number of this connection.

Telephone number: +44321456987
Overridable: □

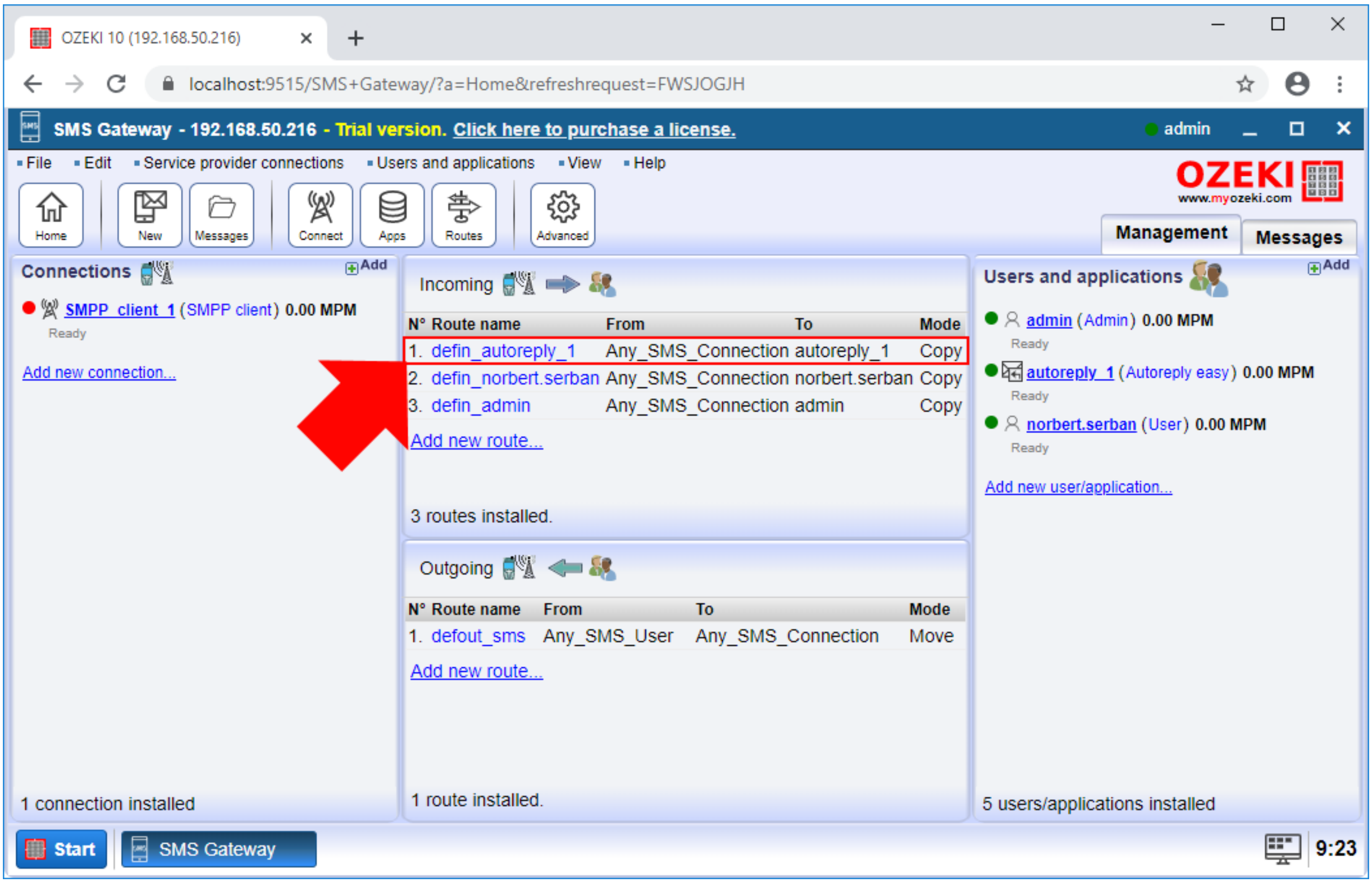
Please install a connection!

Please fill in the configuration form

Figure 6 - Configure an SMPP client connection

## Step 6 - Wait for the messages

Now your autoreply service is active and ready for the incoming messages. You can check that easily in the main menu of the SMS Gateway. Figure 7 shows that the creation of the autoreply service established a routing rule as well. This rule routes all the incoming messages to the autoreply connection and it will just do its job and sends the reply message.


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=Home&refreshrequest=FWSJOGJH
SMS Gateway - 192.168.50.216 - Trial version. Click here to purchase a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
OZEKI
www.myozeki.com
Management
Messages
Start
SMS Gateway
9:23

Connections
Add

SMPP_client_1 (SMPP client) 0.00 MPM
Ready

Add new connection...

1 connection installed

Incoming

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defin_autoreply_1 | Any_SMS_Connection | autoreply_1 | Copy |
| 2. | defin_norbert.serban | Any_SMS_Connection | norbert.serban | Copy |
| 3. | defin_admin | Any_SMS_Connection | admin | Copy |

Add new route...

3 routes installed.

Outgoing

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move |

Add new route...

1 route installed.

Users and applications
Add

admin (Admin) 0.00 MPM
Ready

autoreply_1 (Autoreply easy) 0.00 MPM
Ready

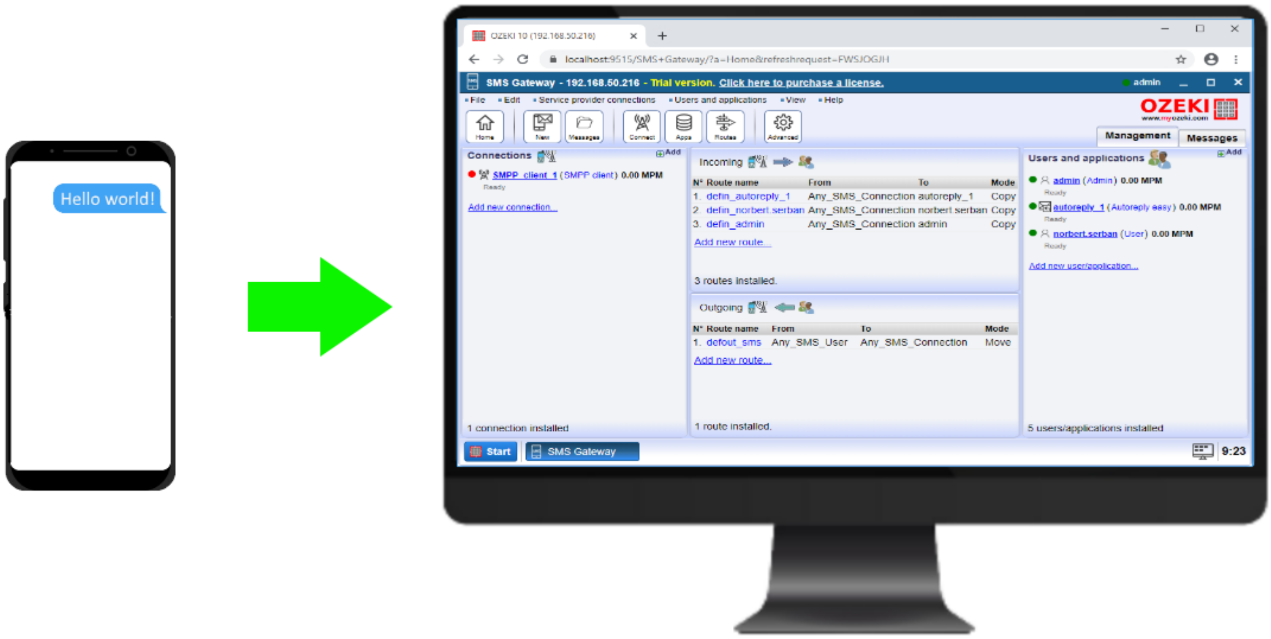
norbert.serban (User) 0.00 MPM
Ready

Add new user/application...

5 users/applications installed

Figure 7 - The incoming messages routed to the autoreply service

Figure 8 demonstrates that how simple an autoreply service works. All you need to do here is to wait for a message forwarded to the telephone number, that you gave for the SMPP client connection. As soon as the client received the message, it forwards it straight to the autoreply service.



Hello world!
OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=Home&refreshrequest=FWSJOGJH
SMS Gateway - 192.168.50.216 - Trial version. Click here to purchase a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
OZEKI
www.myozeki.com
Management
Messages
Start
SMS Gateway
9:23

Connections
SMPP_client_1 (SMPP client) 0.00 MPM
Ready
Add new connection...
1 connection installed

Incoming

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defin_autoreply_1 | Any_SMS_Connection | autoreply_1 | Copy |
| 2. | defin_norbert.serban | Any_SMS_Connection | norbert.serban | Copy |
| 3. | defin_admin | Any_SMS_Connection | admin | Copy |

Add new route...
3 routes installed.

Outgoing

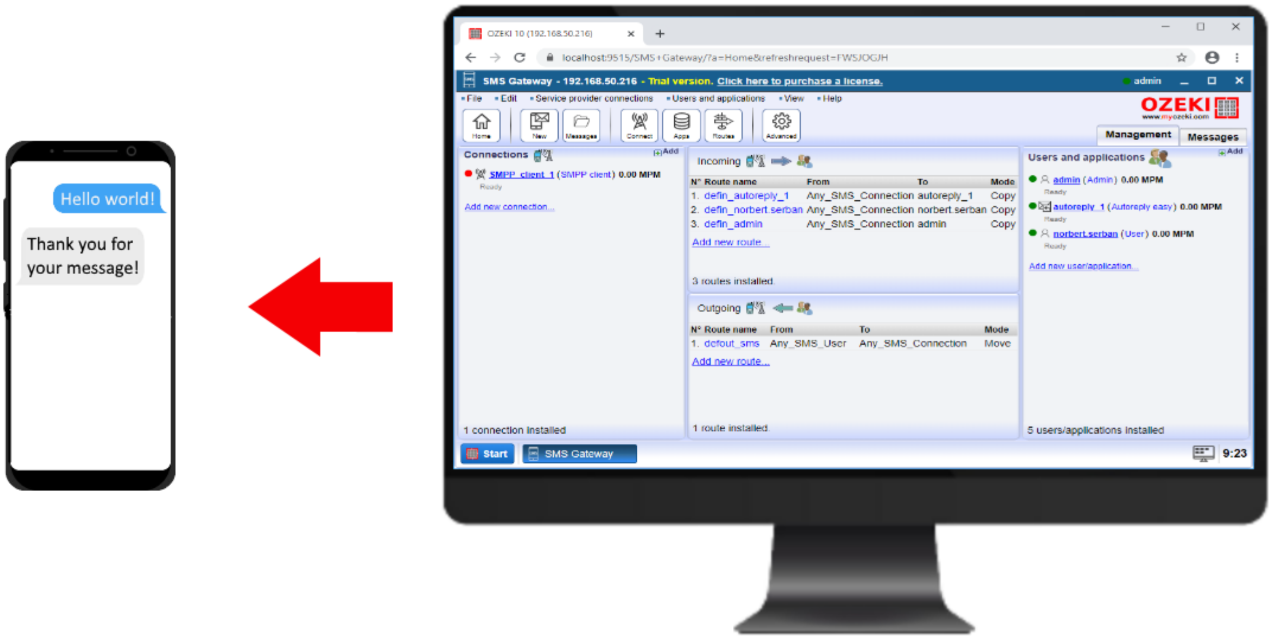
| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move |

Add new route...
1 route installed.

Users and applications
admin (Admin) 0.00 MPM
Ready
autoreply_1 (Autoreply easy) 0.00 MPM
Ready
norbert.serban (User) 0.00 MPM
Ready
Add new user/application...
5 users/applications installed

Figure 8 - Incoming messages routed to the autoreply service

At this point, the autoreply service is going to reply to the received message with the predefined text message. The process of how the sender receives the reply message is demonstrated in Figure 9.



Hello world!

Thank you for your message!

OZEKI 10 (192.168.50.216)

localhost:9515/SMS+Gateway/?a=Home&refreshrequest=FWSJOGJH

SMS Gateway - 192.168.50.216 - Trial version. Click here to purchase a license.

Connections

Add

● SMPP_client_1 (SMPP client) 0.00 MPM
Ready

Add new connection...

1 connection installed

Incoming

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defin_autoreply_1 | Any_SMS_Connection | autoreply_1 | Copy |
| 2. | defin_norbert.serban | Any_SMS_Connection | norbert.serban | Copy |
| 3. | defin_admin | Any_SMS_Connection | admin | Copy |

Add new route...

3 routes installed.

Outgoing

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move |

Add new route...

1 route installed.

Users and applications

Add

● admin (Admin) 0.00 MPM
Ready

● autoreply_1 (Autoreply easy) 0.00 MPM
Ready

● norbert.serban (User) 0.00 MPM
Ready

Add new user/application...

5 users/applications installed

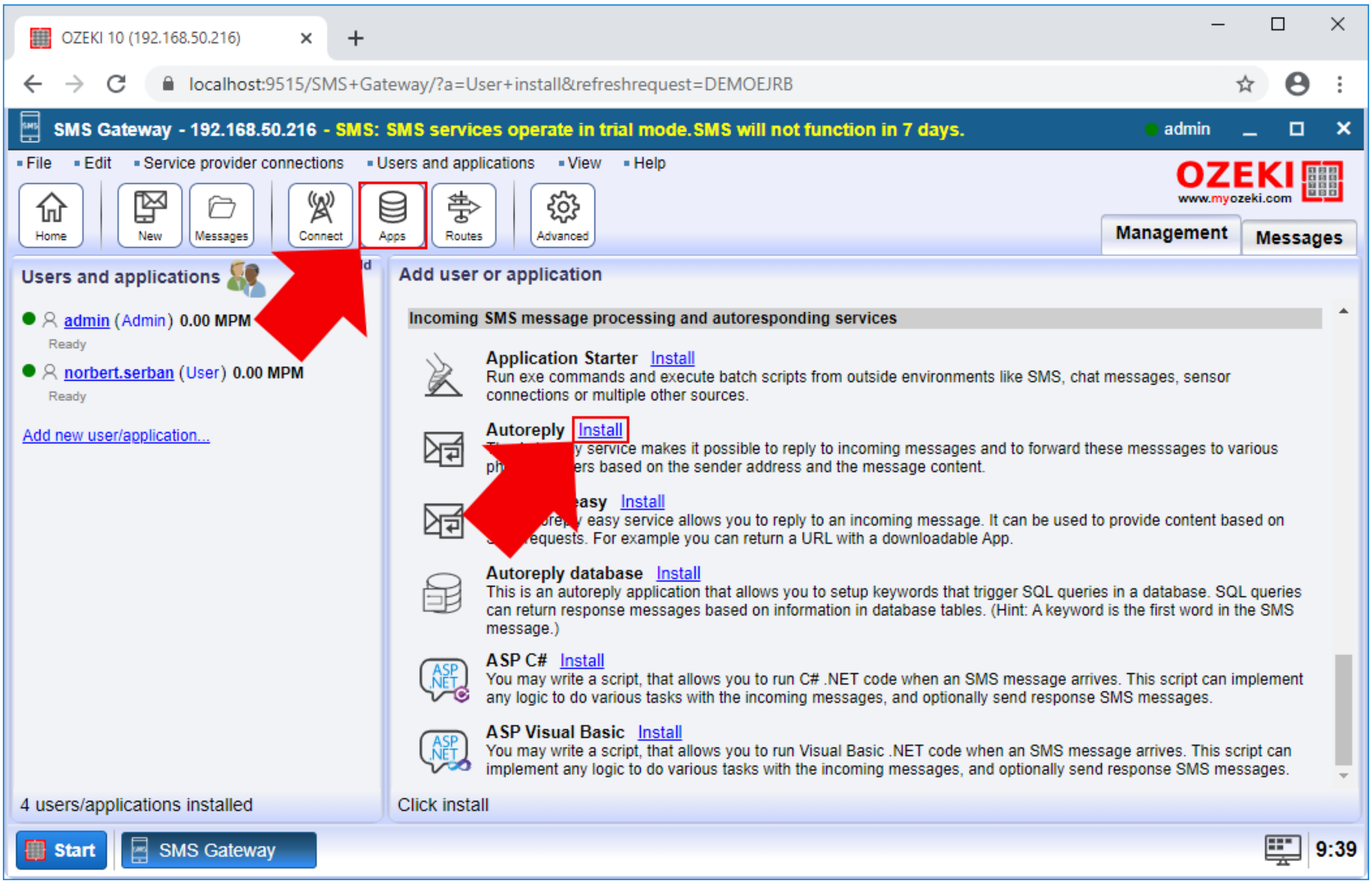
Figure 9 - The autoreply service sends the predefined message

# Automatic SMS replies by script

***The guide on this page is going to give you a brief introduction to how easy you can create an autoreply service that uses a script to answer by a defined action to the received messages. This service is capable of just reply with a predefined message, but you can write a script to forward the incoming messages to a specific phone number or do actions in case the sender phone number or the message itself matches with your predefined details. So, let's start it right now.***

## Step 1 - Create an Autoreply connection

The first thing that you need to do in this guide, is to create the autoreply service connection in the SMS Gateway. So for that, first, click on the Apps icon on the toolbar in the main menu. Here you can see all the available connections, that you can create in the SMS Gateway. Now, but need to scroll down to the 'Incoming SMS message processing and autoresponding services' section, and here, like in Figure 1, click on the Install button of the Autoreply connection.


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=User+install&refreshrequest=DEMOEJRB
SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
OZEKI
www.myozeki.com
Management
Messages
Users and applications
admin (Admin) 0.00 MPM
Ready
norbert.serban (User) 0.00 MPM
Ready
Add new user/application...
4 users/applications installed
Add user or application
Click install
Start
SMS Gateway
9:39

Incoming SMS message processing and autoresponding services

Application Starter Install
Run exe commands and execute batch scripts from outside environments like SMS, chat messages, sensor connections or multiple other sources.

Autoreply Install
Th y service makes it possible to reply to incoming messages and to forward these messsages to various ph ers based on the sender address and the message content.

easy Install
ore y easy service allows you to reply to an incoming message. It can be used to provide content based on equests. For example you can return a URL with a downloadable App.

Autoreply database Install
This is an autoreply application that allows you to setup keywords that trigger SQL queries in a database. SQL queries can return response messages based on information in database tables. (Hint: A keyword is the first word in the SMS message.)

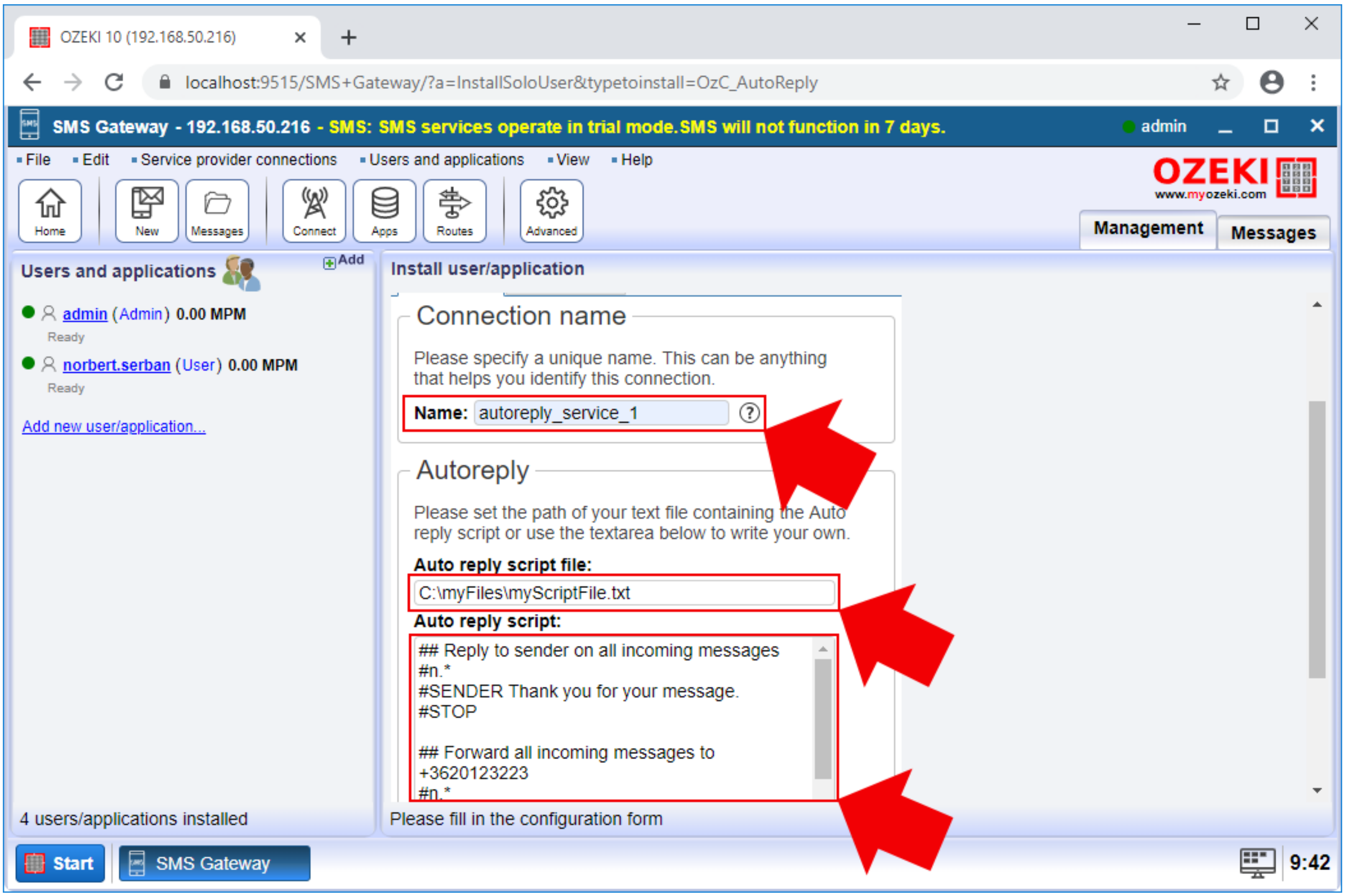
ASP C# Install
You may write a script, that allows you to run C# .NET code when an SMS message arrives. This script can implement any logic to do various tasks with the incoming messages, and optionally send response SMS messages.

ASP Visual Basic Install
You may write a script, that allows you to run Visual Basic .NET code when an SMS message arrives. This script can implement any logic to do various tasks with the incoming messages, and optionally send response SMS messages.

Figure 1 - Create an Autoreply connection

## Step 2 - Configure the Autoreply connection

In the configuration menu of the Autoreply connection, you need to specify a name for the connection in the first step. Then, you select the way, that the connection uses the script to handle the incoming messages. As you can see it in Figure 2, you can set the path for a script file, that the connection can use, or you can just type the script in the textbox below.


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=InstallSoloUser&typetoinstall=OzC_AutoReply
SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
OZEKI
www.myozeki.com
Management
Messages
Users and applications
Add
admin (Admin) 0.00 MPM
Ready
norbert.serban (User) 0.00 MPM
Ready
Add new user/application...
4 users/applications installed
Install user/application
Please fill in the configuration form
Start
SMS Gateway
9:42

Connection name

Please specify a unique name. This can be anything that helps you identify this connection.

Name: autoreply_service_1

Autoreply

Please set the path of your text file containing the Auto reply script or use the textarea below to write your own.

Auto reply script file: C:\myFiles\myScriptFile.txt

Auto reply script:

```
## Reply to sender on all incoming messages
#n.*
#SENDER Thank you for your message.
#STOP

## Forward all incoming messages to
+3620123223
#n.*
```

Figure 2 - Configure the Autoreply connection

# Step 3 - Select an example script

The easiest way to create this autoreply service is to just select one from the available four example scripts from the textbox. Right now they can't be executed since all lines commented out by an '#' sign. The first example script sends a 'Thank you' message back to the sender.

```
n.*
SENDER Thank you for your message.
STOP
```

The second example script is capable of forwarding all the incoming messages to the defined phone number. The second line of the script shows how you can do that by first, defining the phone number, and then, MSG variable stores the incoming message which is in this case will be sent to that phone number.

```
n.*
+3620123223 MSG
STOP
```

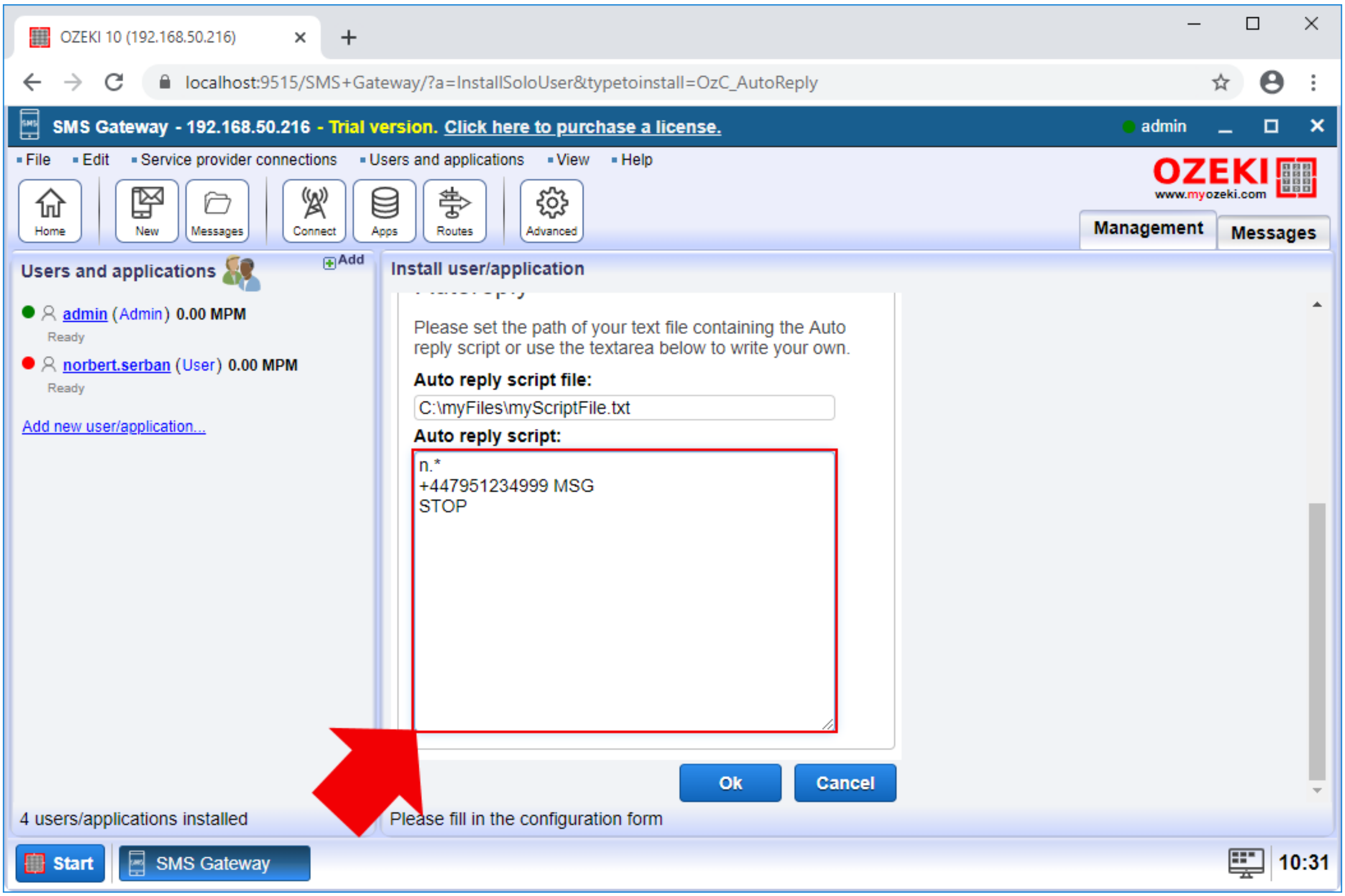
The next example script activates only when the message comes from a specified phone number. This phone number can be defined in the first line of the script. Then, the script forwards the message to the phone number that is defined in the second line. The MSG variable stores the message.

```
n^+362000001
+3620123223 MSG
STOP
```

The last example shows how you can do an action in case the message starts with a specific word. This example forwards the message to two other phone numbers defined in the second and third row. But it happens only in case when the message starts with the word 'important'. The word can be defined as the first row of the script demonstrates it.

```
m^important.*
+3620123223 SENDER: MSG
+3670322321 MSG
STOP
```

To follow this guide, just select the script that will forward the message to the defined phone number. Figure 3 demonstrates, how you need to write the script into the textbox to be able to operate with that script in case of an incoming message.



Please set the path of your text file containing the Auto reply script or use the textarea below to write your own.

Auto reply script file: C:\myFiles\myScriptFile.txt

Auto reply script:

```
n.*
+447951234999 MSG
STOP
```

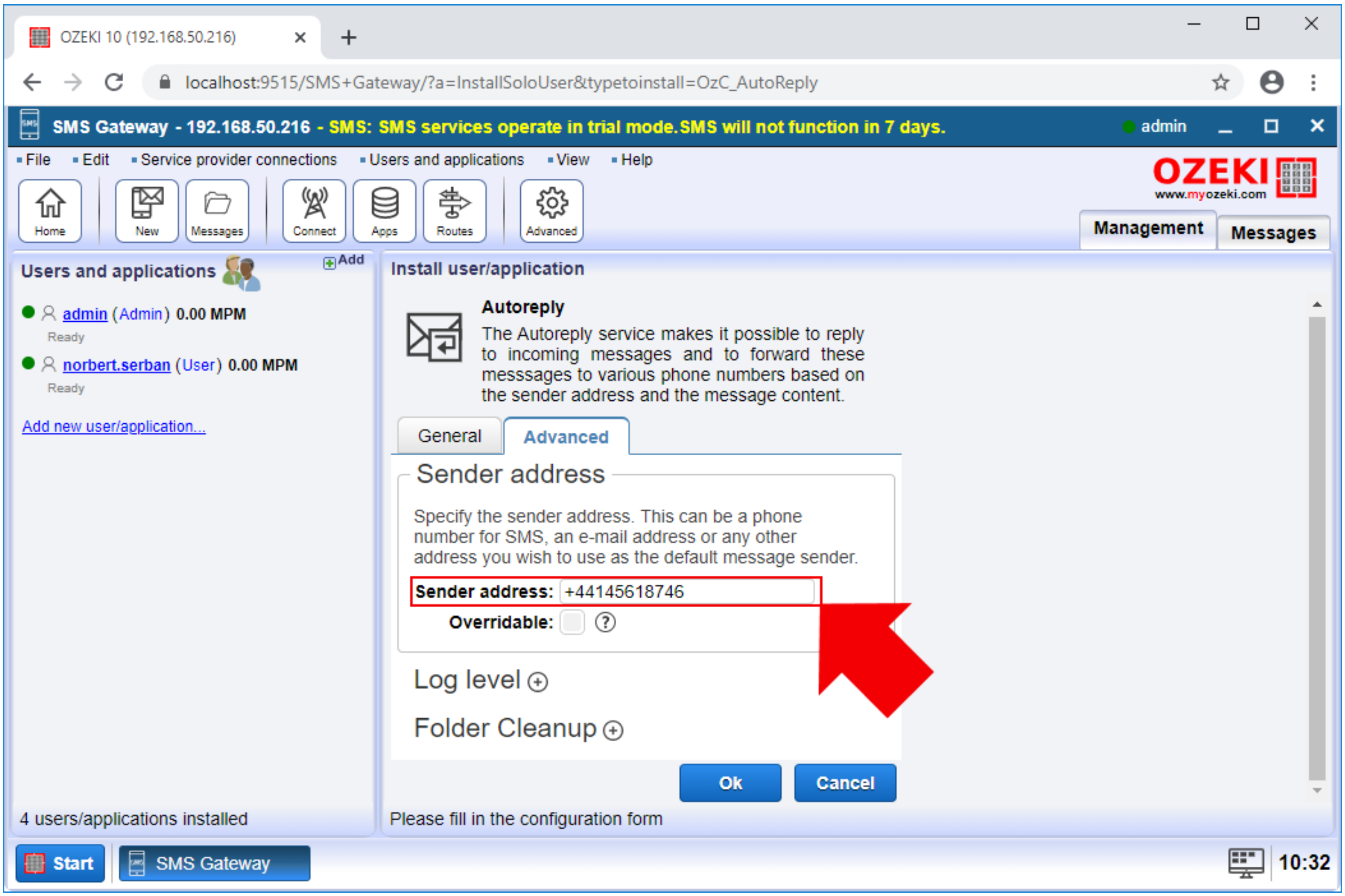
Please fill in the configuration form

OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=InstallSoloUser&typetoinstall=OzC_AutoReply
SMS Gateway - 192.168.50.216 - Trial version. Click here to purchase a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Users and applications
Add
admin (Admin) 0.00 MPM
Ready
norbert.serban (User) 0.00 MPM
Ready
Add new user/application...
4 users/applications installed
Install user/application
Ok
Cancel
Start
SMS Gateway
10:31

Figure 3 - Write a script for the Autoreply service

# Step 4 - Advanced configuration of Autoreply connection

The Autoreply connection can be further configured by setting a default sender address. This address will show up at the recipient side who receives the reply message. As you can see it in Figure 4, you can type here a phone number or an e-mail address which helps to identify the autoreply easy connection. If you finished the configuration, just click on OK.


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=InstallSoloUser&typetoinstall=OzC_AutoReply
SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Users and applications
Add
admin (Admin) 0.00 MPM
Ready
norbert.serban (User) 0.00 MPM
Ready
Add new user/application...
4 users/applications installed
Install user/application
General
Advanced
Log level
Folder Cleanup
Ok
Cancel
Please fill in the configuration form
Start
SMS Gateway
10:32

**Autoreply**

The Autoreply service makes it possible to reply to incoming messages and to forward these messsages to various phone numbers based on the sender address and the message content.

Sender address

Specify the sender address. This can be a phone number for SMS, an e-mail address or any other address you wish to use as the default message sender.

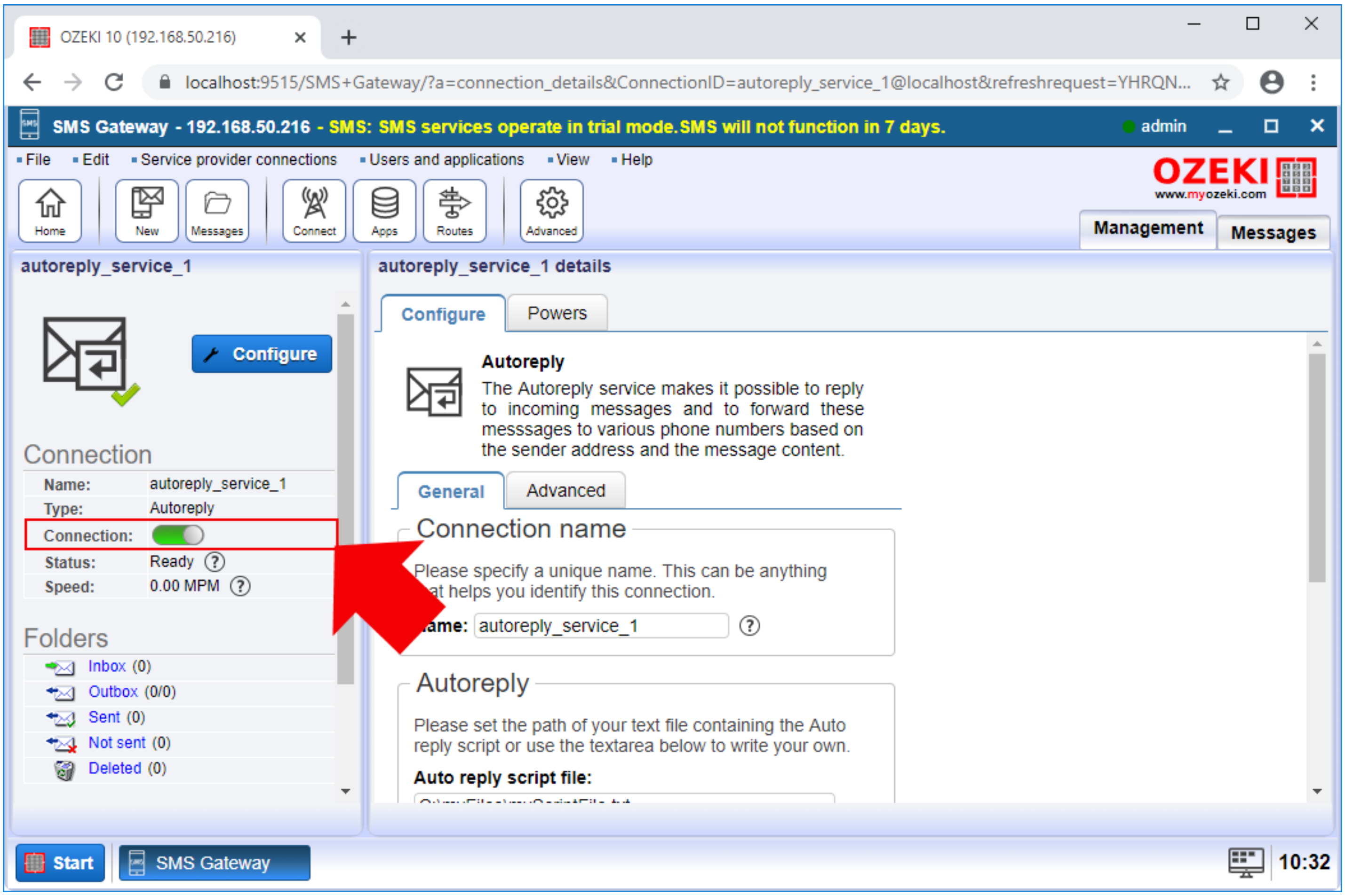
Sender address: +44145618746

Overridable: □

Figure 4 - Advanced configuration of the Autoreply connection

# Step 5 - Enable the Autoreply connection

After you created the Autoreply connection, the next window is the main menu of the autoreply service. Here, you can modify the details of the connection anytime you want. To activate the service, you need to enable the connection here. For that, just click on the toggle as Figure 5 shows. The toggle is green now that indicates that the autoreply service is now active.


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=connection_details&ConnectionID=autoreply_service_1@localhost&refreshrequest=YHRQN...
SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Configure
Start
SMS Gateway
10:32

autoreply_service_1

Connection

Name: autoreply_service_1
Type: Autoreply
Connection:
Status: Ready
Speed: 0.00 MPM

Folders

Inbox (0)
Outbox (0/0)
Sent (0)
Not sent (0)
Deleted (0)

autoreply_service_1 details

Configure | Powers

**Autoreply**
The Autoreply service makes it possible to reply to incoming messages and to forward these messsages to various phone numbers based on the sender address and the message content.

General | Advanced

Connection name

Please specify a unique name. This can be anything at helps you identify this connection.

Name: autoreply_service_1

Autoreply

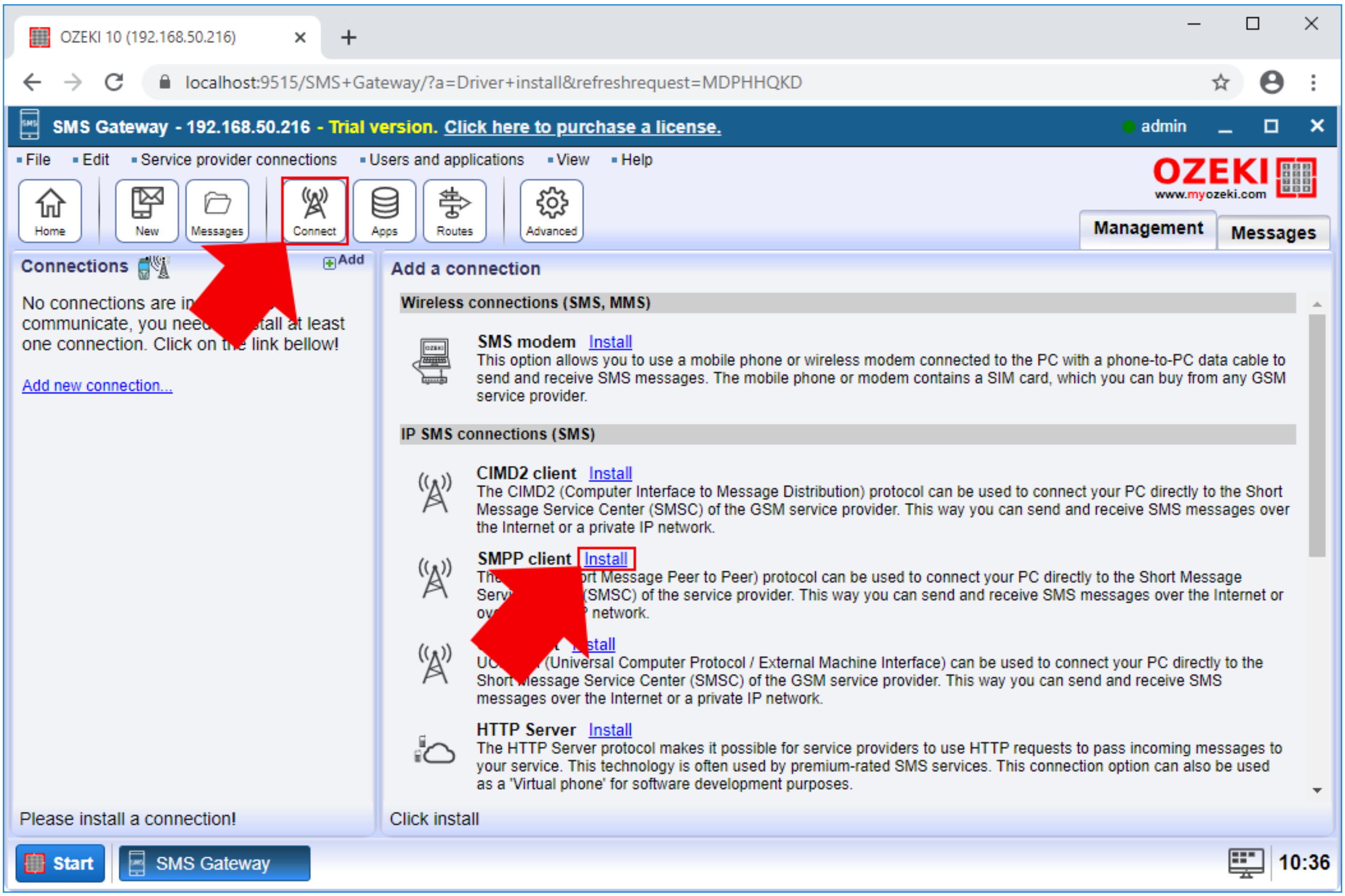
Please set the path of your text file containing the Auto reply script or use the textarea below to write your own.

**Auto reply script file:**

Figure 5 - Enabling the Autoreply connection

# Step 6 - Connect to the mobile network

By this point, you created and enabled your autoreply service, but to receive messages, you also need a connection that can connect to the mobile network and forward messages to your Ozeki system. To do that, click on Connect on the toolbar, and select the SMPP client as you can see it in Figure 6.


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=Driver+install&refreshrequest=MDPHHQKD
SMS Gateway - 192.168.50.216 - Trial version. Click here to purchase a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
OZEKI
www.myozeki.com
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Connections
Add
Add a connection
Start
SMS Gateway
10:36

No connections are in... communicate, you nee... tall at least one connection. Click on the link bellow!

Add new connection...

Please install a connection!

**Wireless connections (SMS, MMS)**

**SMS modem** Install
This option allows you to use a mobile phone or wireless modem connected to the PC with a phone-to-PC data cable to send and receive SMS messages. The mobile phone or modem contains a SIM card, which you can buy from any GSM service provider.

**IP SMS connections (SMS)**

**CIMD2 client** Install
The CIMD2 (Computer Interface to Message Distribution) protocol can be used to connect your PC directly to the Short Message Service Center (SMSC) of the GSM service provider. This way you can send and receive SMS messages over the Internet or a private IP network.

**SMPP client** Install
The ...ort Message Peer to Peer) protocol can be used to connect your PC directly to the Short Message Serv... (SMSC) of the service provider. This way you can send and receive SMS messages over the Internet or ov... network.

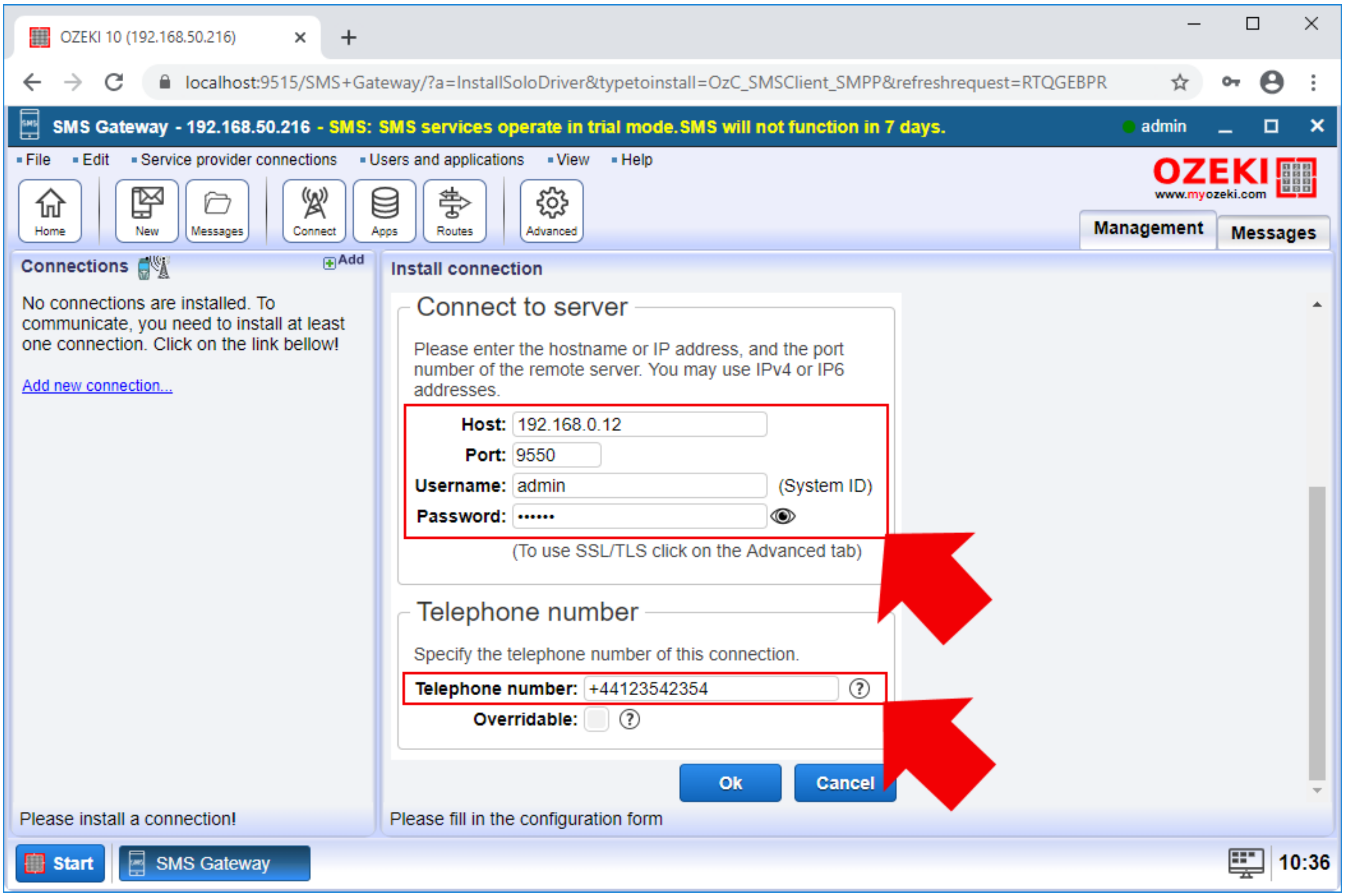
...stall
UC... (Universal Computer Protocol / External Machine Interface) can be used to connect your PC directly to the Short ...essage Service Center (SMSC) of the GSM service provider. This way you can send and receive SMS messages over the Internet or a private IP network.

**HTTP Server** Install
The HTTP Server protocol makes it possible for service providers to use HTTP requests to pass incoming messages to your service. This technology is often used by premium-rated SMS services. This connection option can also be used as a 'Virtual phone' for software development purposes.

Click install

Figure 6 - Create an SMPP client connection

In the configuration menu, you have to provide some details about the server. This is the server, that this SMPP client is going to use to handle SMS messages. So here, you need to type the host, port, and user account details. Then, you need to specify a telephone number for this connection (Figure 6). With this number, the connection can be identified, and able to send or receive messages. Lastly, you just need to click on OK to create the SMPP client connection.


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=InstallSoloDriver&typetoinstall=OzC_SMSClient_SMPP&refreshrequest=RTQGEBPR
SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Connections
Add
Start
SMS Gateway
10:36

No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

Please install a connection!

Install connection

Connect to server

Please enter the hostname or IP address, and the port number of the remote server. You may use IPv4 or IP6 addresses.

**Host:** 192.168.0.12
**Port:** 9550
**Username:** admin (System ID)
**Password:** ••••••

(To use SSL/TLS click on the Advanced tab)

Telephone number

Specify the telephone number of this connection.

**Telephone number:** +44123542354
**Overridable:** □

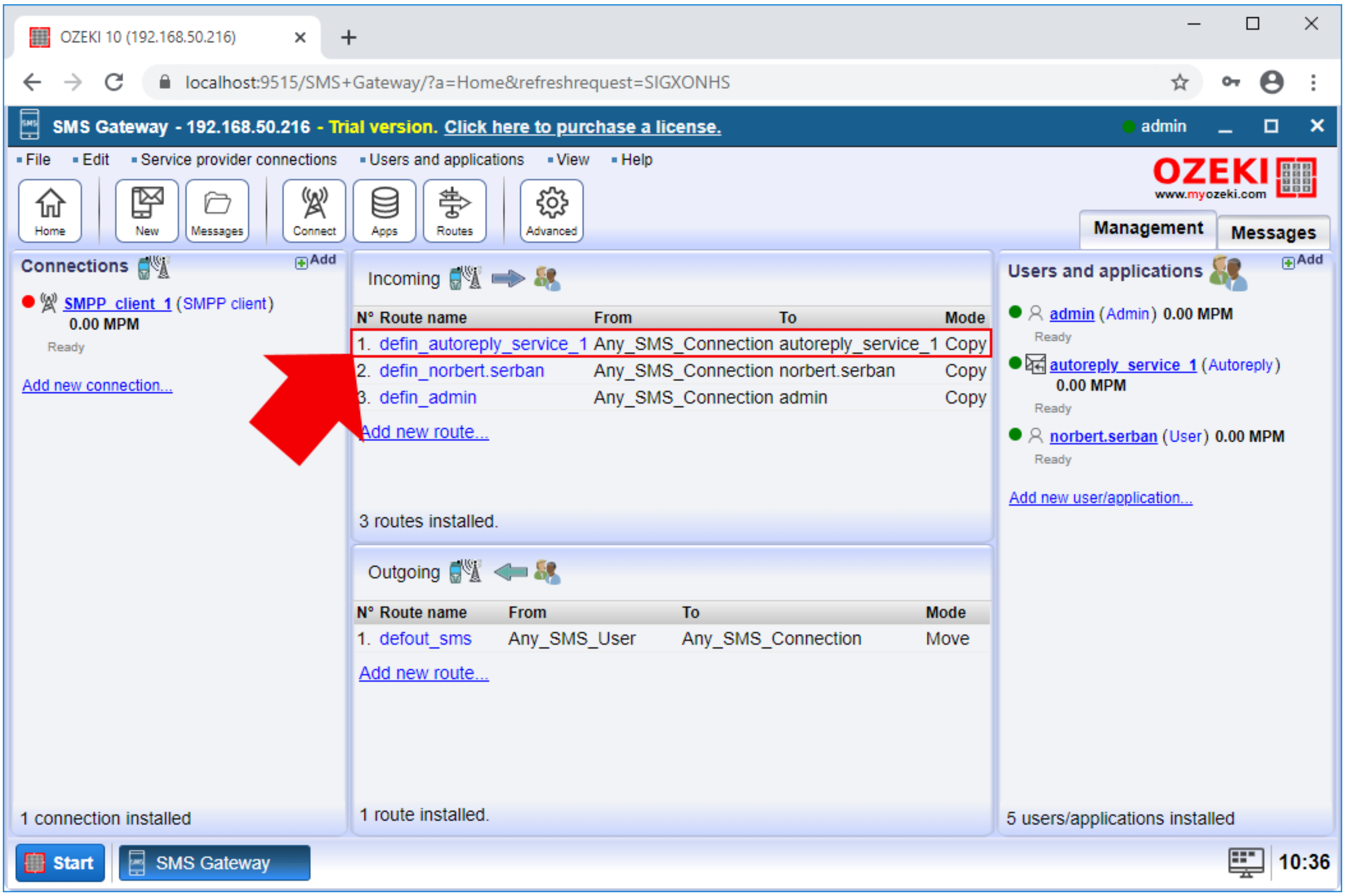
Ok Cancel

Please fill in the configuration form

Figure 7 - Configure an SMPP client connection

# Step 7 - Wait for the messages

After setting up and enabling the autoreply service, now it is ready for the incoming messages. You can check that easily in the main menu of the SMS Gateway. Figure 8 shows that the creation of the autoreply service established a routing rule as well. This rule routes all the incoming messages to the autoreply connection and it will just do its job and forward the messages to the dedicated phone number.


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=Home&refreshrequest=SIGXONHS
SMS Gateway - 192.168.50.216 - Trial version. Click here to purchase a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
OZEKI
www.myozeki.com
Management
Messages
Start
SMS Gateway
10:36

Connections

SMPP_client_1 (SMPP client)
0.00 MPM
Ready

Add new connection...

1 connection installed

Incoming

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defin_autoreply_service_1 | Any_SMS_Connection | autoreply_service_1 | Copy |
| 2. | defin_norbert.serban | Any_SMS_Connection | norbert.serban | Copy |
| 3. | defin_admin | Any_SMS_Connection | admin | Copy |

Add new route...

3 routes installed.

Outgoing

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move |

Add new route...

1 route installed.

Users and applications

admin (Admin) 0.00 MPM
Ready

autoreply_service_1 (Autoreply)
0.00 MPM
Ready

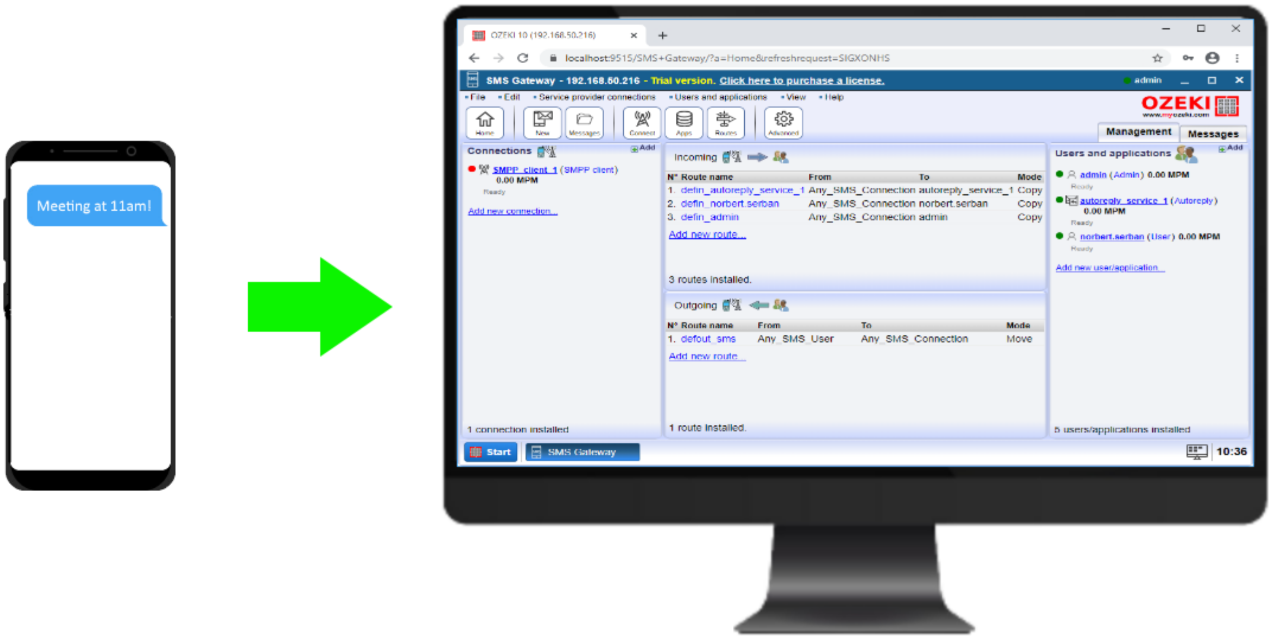
norbert.serban (User) 0.00 MPM
Ready

Add new user/application...

5 users/applications installed

Figure 8 - Route incoming messages to the Autoreply connection

Figure 9 demonstrates how simple an autoreply service works. All you need to do here is to wait for a message forwarded to the telephone number, that you gave for the SMPP client connection. As soon as the client received the message, it forwards it straight to the autoreply service.



OZEKI 10 (192.168.50.216)

localhost:9515/SMS+Gateway/?a=Home&refreshrequest=SIGXONHS

SMS Gateway - 192.168.50.216 - Trial version. Click here to purchase a license.

Connections

SMPP_client_1 (SMPP client)
0.00 MPM
Ready

Add new connection...

1 connection installed

Incoming

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defin_autoreply_service_1 | Any_SMS_Connection | autoreply_service_1 | Copy |
| 2. | defin_norbert.serban | Any_SMS_Connection | norbert.serban | Copy |
| 3. | defin_admin | Any_SMS_Connection | admin | Copy |

Add new route...

3 routes installed.

Outgoing

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move |

Add new route...

1 route installed.

Users and applications

admin (Admin) 0.00 MPM
Ready

autoreply_service_1 (Autoreply)
0.00 MPM
Ready

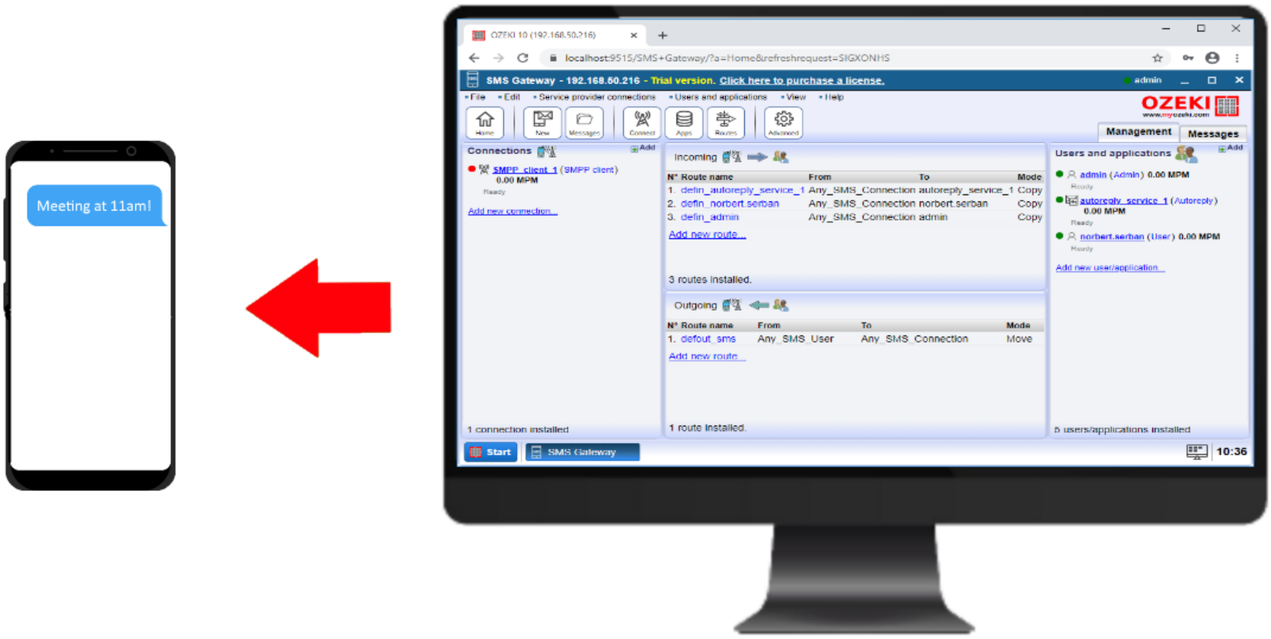
norbert.serban (User) 0.00 MPM
Ready

Add new user/application...

5 users/applications installed

Figure 9 - Incoming messages routed to the autoreply service

When the Autoreply service receives the message, the script, that you wrote for the service will be executed. In this case, it will forwards the incoming message to the specified phone number as Figure 10 shows that.



Meeting at 11am!
OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=Home&refreshrequest=SIGXONHS
SMS Gateway - 192.168.50.216 - Trial version. Click here to purchase a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
OZEKI
www.myozeki.com
Management
Messages
Start
SMS Gateway
10:36

Connections

SMPP_client_1 (SMPP client)
0.00 MPM
Ready

Add new connection...

1 connection installed

Incoming

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defin_autoreply_service_1 | Any_SMS_Connection | autoreply_service_1 | Copy |
| 2. | defin_norbert.serban | Any_SMS_Connection | norbert.serban | Copy |
| 3. | defin_admin | Any_SMS_Connection | admin | Copy |

Add new route...

3 routes installed.

Outgoing

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move |

Add new route...

1 route installed.

Users and applications

admin (Admin) 0.00 MPM
Ready

autoreply_service_1 (Autoreply)
0.00 MPM
Ready

norbert.serban (User) 0.00 MPM
Ready

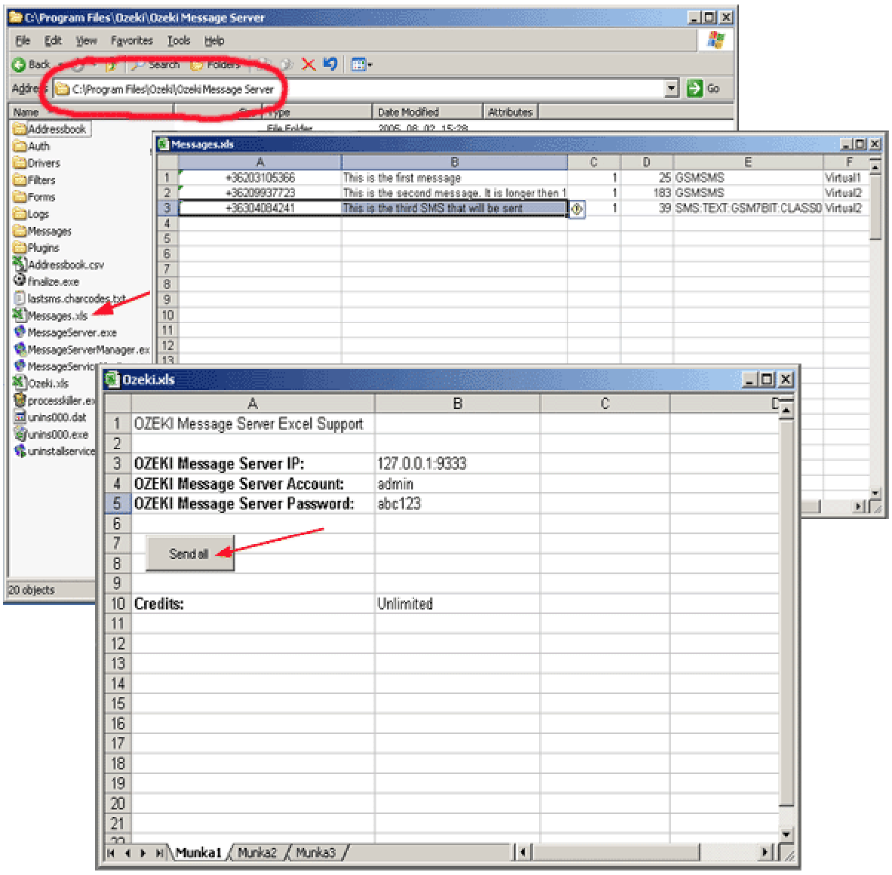
Add new user/application...

5 users/applications installed

Figure 10 - The autoreply service forwards the message

# How to send SMS messages from Excel

***The following guide is about to demonstrate how you can send multiple SMS messages by just one click from an Excel document using the SMS Gateway. The guide provides a step by step guide which shows what components you need to set up this solution. It presents how you can establish an HTTP API connection in Ozeki SMS Gateway that can forward the messages received from the Excel document. The guide takes about ten minutes to be successfully completed and does not require any specific knowledge. So, let's get started.***



|   | A | B | C | D | E | F |
|---|---|---|---|---|---|---|
| 1 | +36203105366 | This is the first message | 1 | 25 | GSMSMS | Virtual1 |
| 2 | +36209937723 | This is the second message. It is longer then 1 | 1 | 183 | GSMSMS | Virtual2 |
| 3 | +36304084241 | This is the third SMS that will be sent | 1 | 39 | SMS:TEXT:GSM7BIT:CLASS0 | Virtual2 |

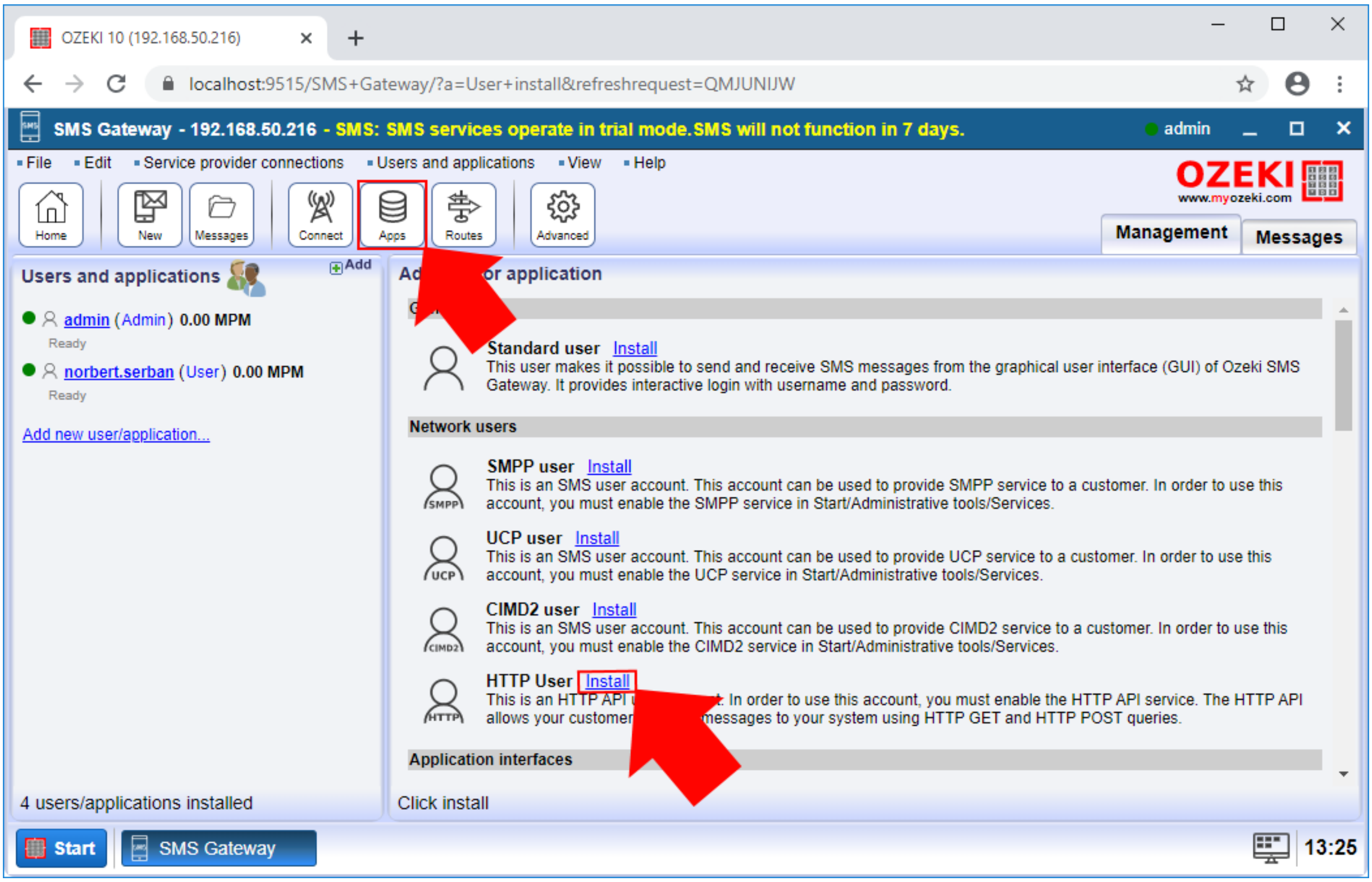
|   | A | B |
|---|---|---|
| 1 | OZEKI Message Server Excel Support | |
| 2 | | |
| 3 | **OZEKI Message Server IP:** | 127.0.0.1:9333 |
| 4 | **OZEKI Message Server Account:** | admin |
| 5 | **OZEKI Message Server Password:** | abc123 |
| 10 | **Credits:** | Unlimited |

## Prerequisites

- **Ozeki Excel Client:** Download Ozeki Excel client
- **Template Excel file for messages:** Download template messages

## Step 1 - Create an HTTP user connection

The first step of this guide is to create an HTTP user connection that will be able to handle the SMS messages from the Excel file. So, to create this connection, first open SMS Gateway from your Ozeki desktop. Here, click on the Apps icon on the toolbar and select the HTTP User option as you can see it in Figure 1.



Standard user Install
This user makes it possible to send and receive SMS messages from the graphical user interface (GUI) of Ozeki SMS Gateway. It provides interactive login with username and password.

Network users

SMPP user Install
This is an SMS user account. This account can be used to provide SMPP service to a customer. In order to use this account, you must enable the SMPP service in Start/Administrative tools/Services.

UCP user Install
This is an SMS user account. This account can be used to provide UCP service to a customer. In order to use this account, you must enable the UCP service in Start/Administrative tools/Services.

CIMD2 user Install
This is an SMS user account. This account can be used to provide CIMD2 service to a customer. In order to use this account, you must enable the CIMD2 service in Start/Administrative tools/Services.

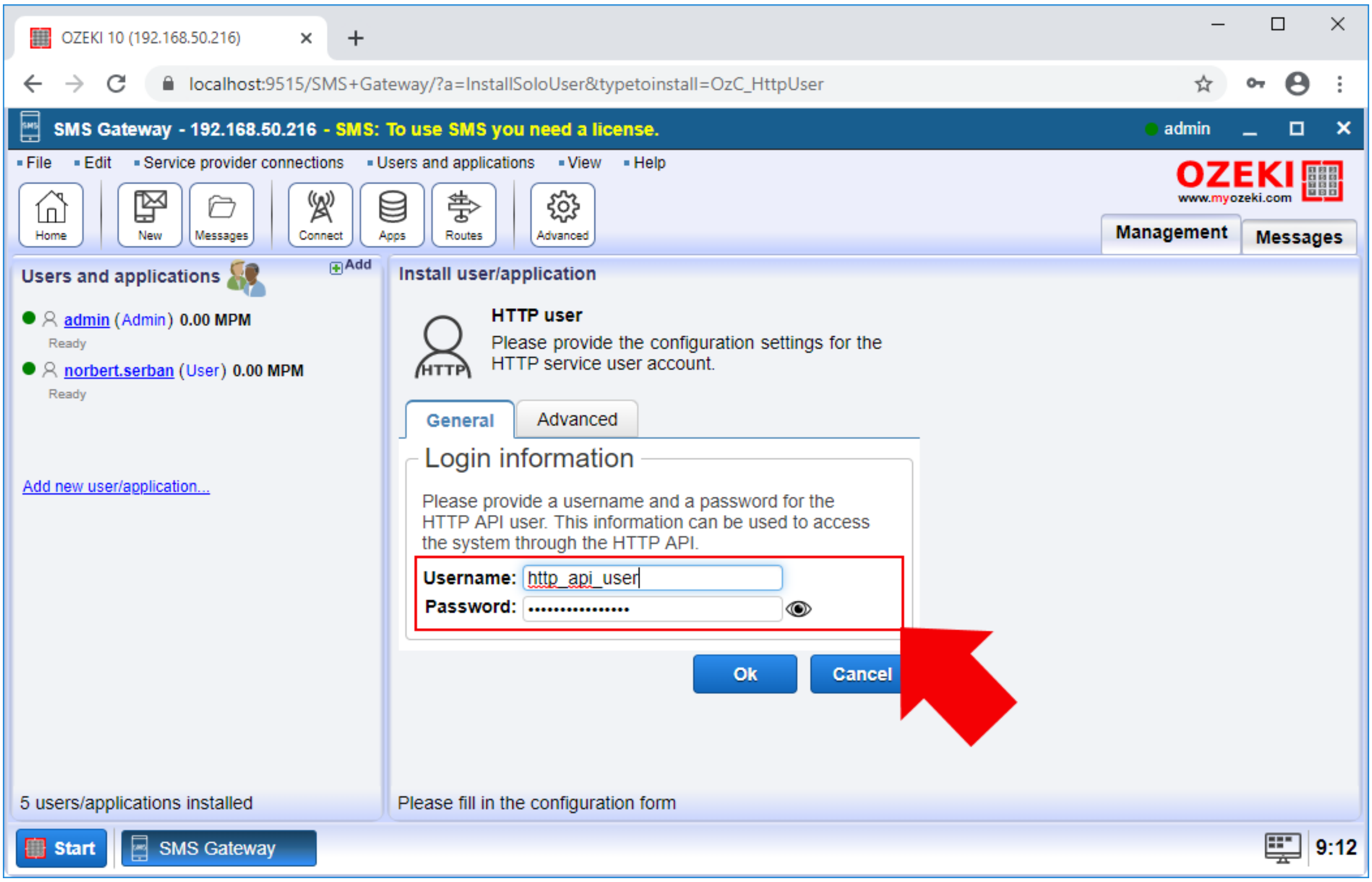
HTTP User Install
This is an HTTP API u... In order to use this account, you must enable the HTTP API service. The HTTP API allows your customer... messages to your system using HTTP GET and HTTP POST queries.

OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=User+install&refreshrequest=QMJUNIJW
SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
OZEKI
www.myozeki.com
Management
Messages
Users and applications
Add
admin (Admin) 0.00 MPM
Ready
norbert.serban (User) 0.00 MPM
Ready
Add new user/application...
Ad... or application
Application interfaces
4 users/applications installed
Click install
Start
SMS Gateway
13:25

Figure 1 - Select HTTP User connection from the list

## Step 2 - Configure the HTTP User connection

In the configuration menu, you need to specify some identifiers for the HTTP User connection to be able to handle messages from the Excel file. So, just type a username and a password for the connection (Figure 2), but remember, you need to provide these details for the Ozeki Excel Client as well. If you filled all the required fields, just click on Ok to create the HTTP User connection.


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=InstallSoloUser&typetoinstall=OzC_HttpUser
SMS Gateway - 192.168.50.216 - SMS: To use SMS you need a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
OZEKI
www.myozeki.com
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Users and applications
Add
admin (Admin) 0.00 MPM
Ready
norbert.serban (User) 0.00 MPM
Ready
Add new user/application...
5 users/applications installed
Install user/application
HTTP
General
Advanced
Ok
Cancel
Please fill in the configuration form
Start
SMS Gateway
9:12

HTTP user

Please provide the configuration settings for the HTTP service user account.

Login information

Please provide a username and a password for the HTTP API user. This information can be used to access the system through the HTTP API.

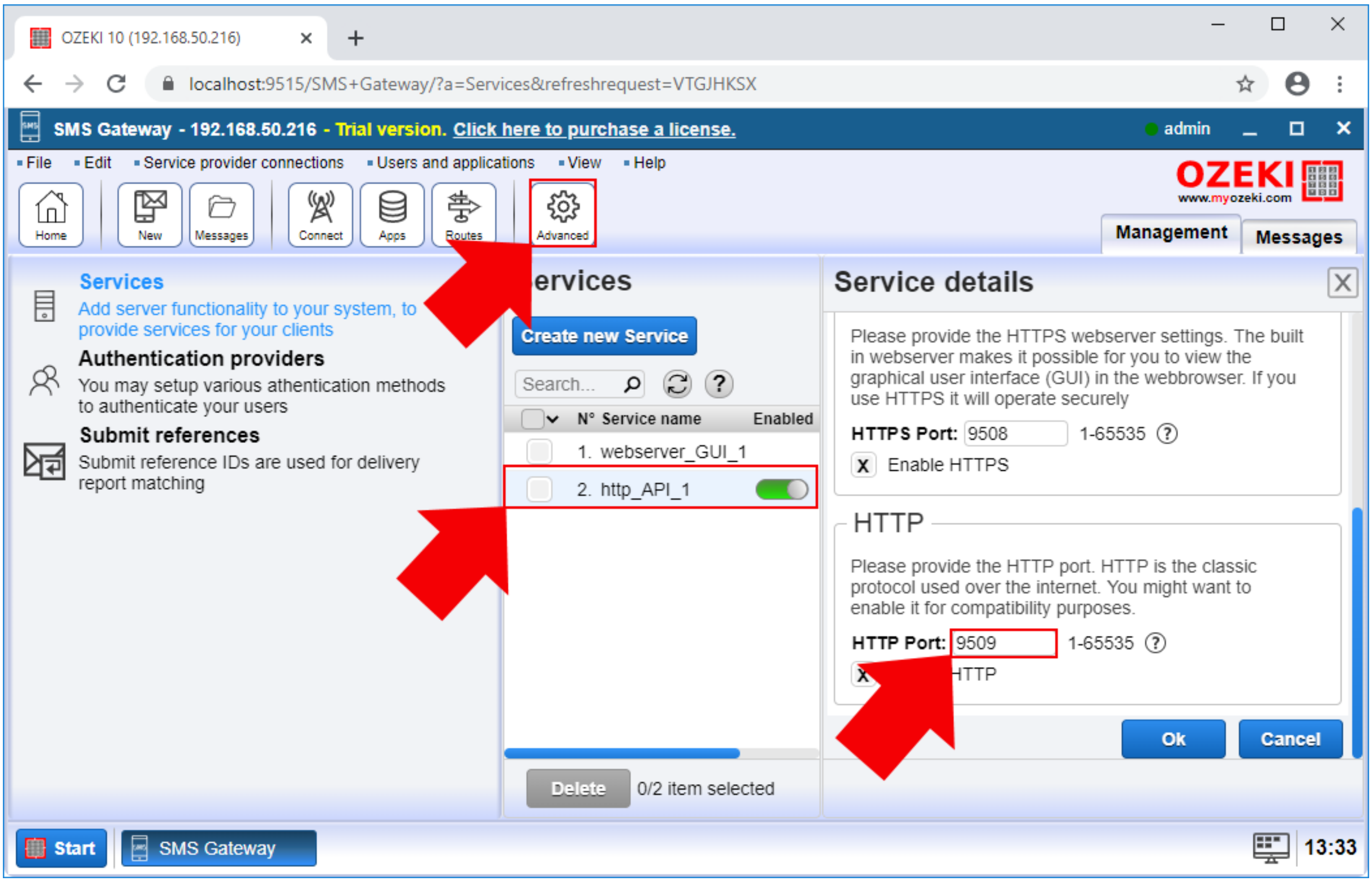
Username: http_api_user

Password: ••••••••••••••••

Figure 2 - Configure the HTTP User connection

# Step 3 - Check the HTTP API service

By creating the HTTP User connection, the SMS Gateway also created an HTTP API service. This service can be viewed in the Advanced menu (Figure 3). If you click on the HTTP API service, you can modify the port for the HTTP and the HTTPS as well. To follow the guide, change the port of the HTTP to 9509 and click on OK.


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=Services&refreshrequest=VTGJHKSX
SMS Gateway - 192.168.50.216 - Trial version. Click here to purchase a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Start
SMS Gateway
13:33

Services
Add server functionality to your system, to provide services for your clients

Authentication providers
You may setup various athentication methods to authenticate your users

Submit references
Submit reference IDs are used for delivery report matching

Services

Create new Service

| N° | Service name | Enabled |
|---|---|---|
| 1. | webserver_GUI_1 | |
| 2. | http_API_1 | ● |

Delete 0/2 item selected

Service details

Please provide the HTTPS webserver settings. The built in webserver makes it possible for you to view the graphical user interface (GUI) in the webbrowser. If you use HTTPS it will operate securely

HTTPS Port: 9508 1-65535

☒ Enable HTTPS

HTTP

Please provide the HTTP port. HTTP is the classic protocol used over the internet. You might want to enable it for compatibility purposes.

HTTP Port: 9509 1-65535

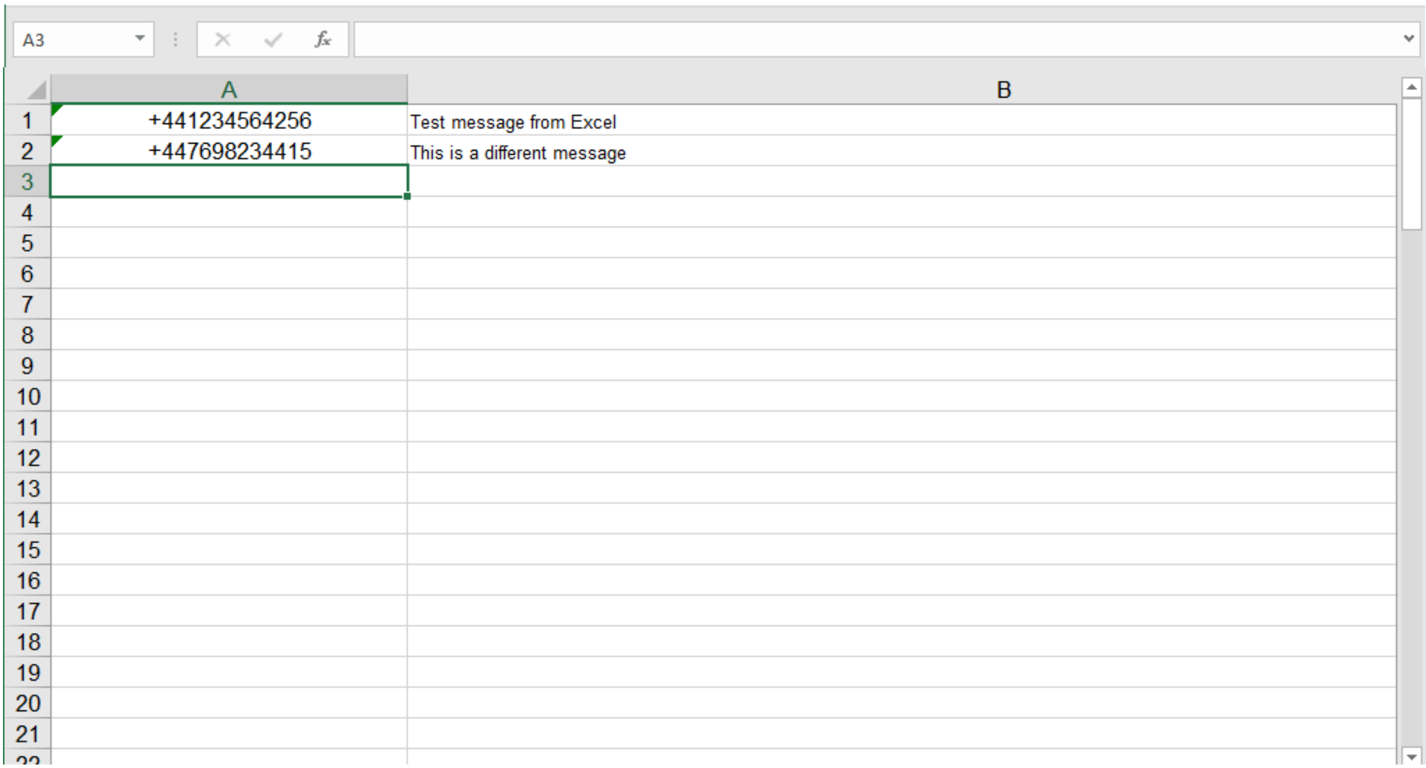
☒ HTTP

Ok Cancel

Figure 3 - HTTP API Service

# Step 4 - Create a list of messages in Excel

After setting up all connections and services that needed to handle SMS messages from Excel, now you can set up the messages itself. For testing, you can use the provided Excel file above, but you can easily create a new file for the messages. In this file, each row represents the details of a message. The first column in each row is dedicated to the phone number of the recipient and the second column is for the message itself (Figure 4). If you have done writing the messages, just save the modifications.

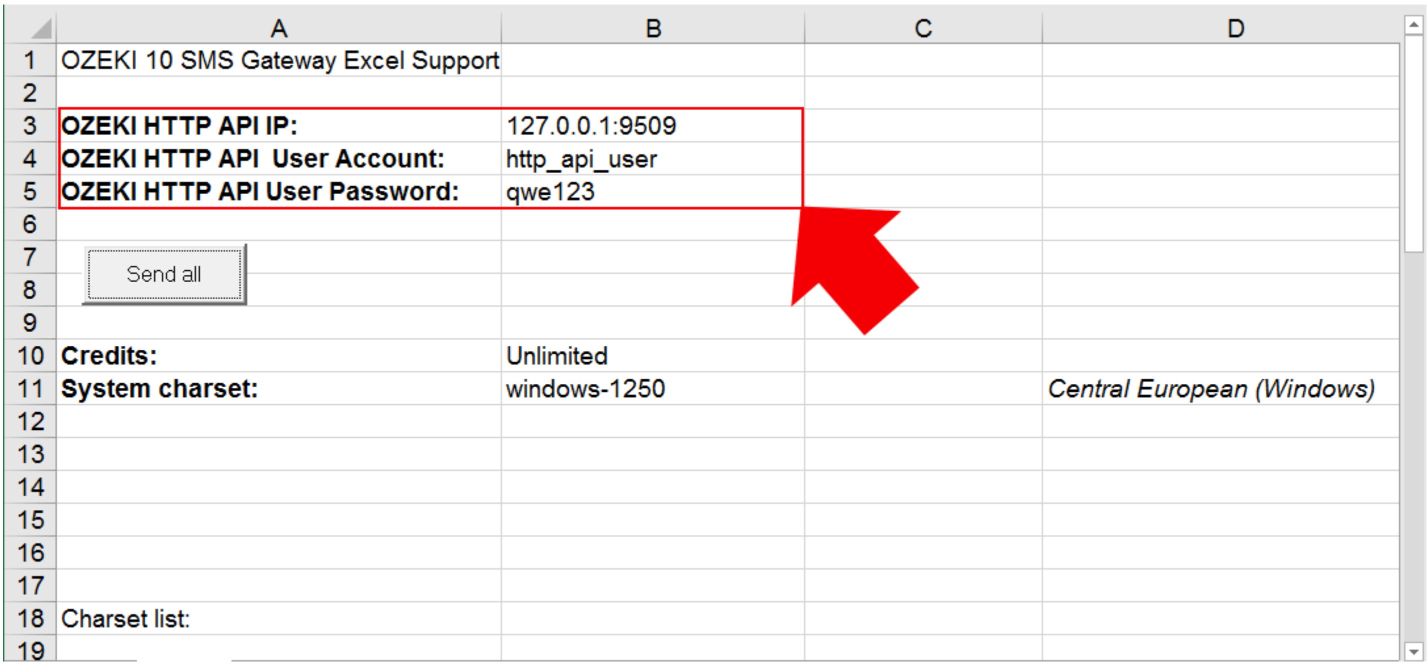


| | A | B |
|---|---|---|
| 1 | +441234564256 | Test message from Excel |
| 2 | +447698234415 | This is a different message |
| 3 | | |
| 4 | | |
| 5 | | |
| 6 | | |
| 7 | | |
| 8 | | |
| 9 | | |
| 10 | | |
| 11 | | |
| 12 | | |
| 13 | | |
| 14 | | |
| 15 | | |
| 16 | | |
| 17 | | |
| 18 | | |
| 19 | | |
| 20 | | |
| 21 | | |

Figure 4 - Excel file contains the SMS messages

# Step 5 - Open the Ozeki Excel SMS spreadsheet

Next, you need to open the downloaded Ozeki Excel Client. Here, you have to provide the details of the HTTP API service and the HTTP User connection, that you created in the SMS Gateway. So, for the Ozeki HTTP API IP field, enter the IP of the localhost (127.0.0.1) and the port number of the HTTP API service separated by a ':' sign. Next, type the username and the password of the HTTP User connection as Figure 5 demonstrates it.

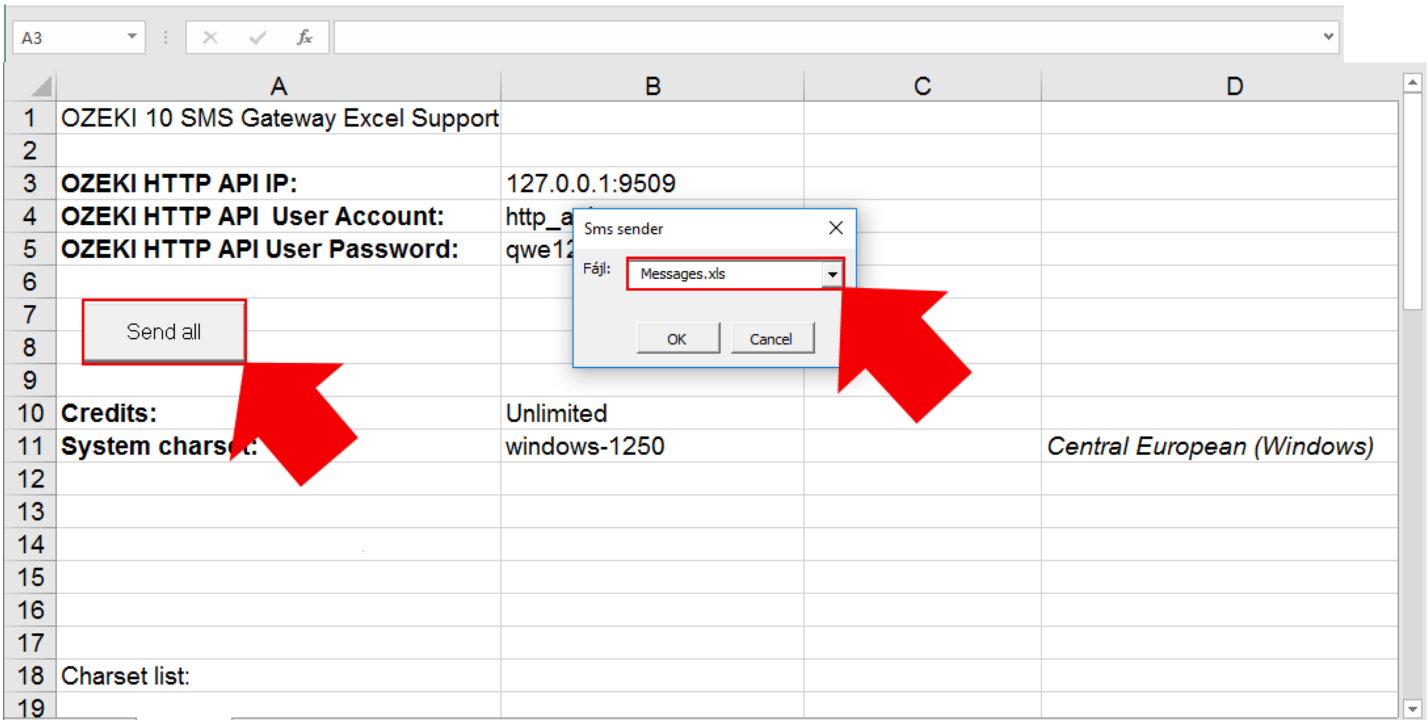


| | A | B | C | D |
|---|---|---|---|---|
| 1 | OZEKI 10 SMS Gateway Excel Support | | | |
| 2 | | | | |
| 3 | **OZEKI HTTP API IP:** | 127.0.0.1:9509 | | |
| 4 | **OZEKI HTTP API User Account:** | http_api_user | | |
| 5 | **OZEKI HTTP API User Password:** | qwe123 | | |
| 6 | | | | |
| 7 | Send all | | | |
| 8 | | | | |
| 9 | | | | |
| 10 | **Credits:** | Unlimited | | |
| 11 | **System charset:** | windows-1250 | | *Central European (Windows)* |
| 12 | | | | |
| 13 | | | | |
| 14 | | | | |
| 15 | | | | |
| 16 | | | | |
| 17 | | | | |
| 18 | Charset list: | | | |
| 19 | | | | |

Figure 5 - Ozeki Excel Client

# Step 6 - Send the SMS messages

After you set up all details that needed, you can just click on Send all button to send all the messages from the Excel file. If the Excel file of the SMS messages opened as well, Ozeki Excel Client automatically detects it, and you can select it from the pop up window (Figure 6). All you need to do here, is to click on OK.



| | A | B | C | D |
|---|---|---|---|---|
| 1 | OZEKI 10 SMS Gateway Excel Support | | | |
| 2 | | | | |
| 3 | **OZEKI HTTP API IP:** | 127.0.0.1:9509 | | |
| 4 | **OZEKI HTTP API User Account:** | http_a | | |
| 5 | **OZEKI HTTP API User Password:** | qwe12 | | |
| 6 | | | | |
| 7 | Send all | | | |
| 8 | | | | |
| 9 | | | | |
| 10 | **Credits:** | Unlimited | | |
| 11 | **System chars:** | windows-1250 | | *Central European (Windows)* |
| 12 | | | | |
| 13 | | | | |
| 14 | | | | |
| 15 | | | | |
| 16 | | | | |
| 17 | | | | |
| 18 | Charset list: | | | |
| 19 | | | | |

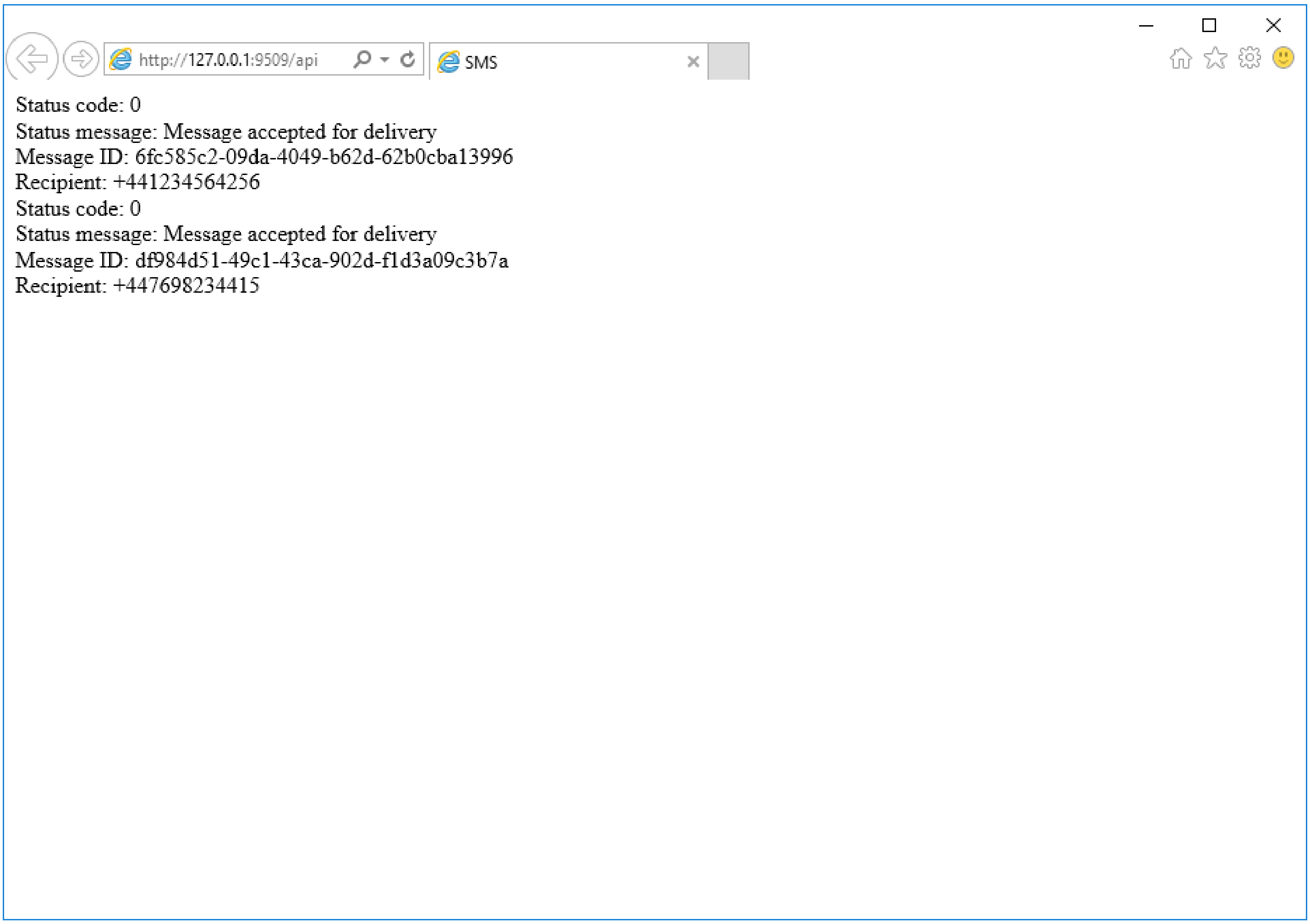
Sms sender

Fájl: Messages.xls

OK Cancel

Figure 6 - Select the Excel file with the SMS messages

When Ozeki Excel Client finished with sending all the message from the selected Excel file, a delivery report shows up in the browser (Figure 7). Here, you can check if the messages sent successfully, or a problem occurred during the sending process.



```
Status code: 0
Status message: Message accepted for delivery
Message ID: 6fc585c2-09da-4049-b62d-62b0cba13996
Recipient: +441234564256
Status code: 0
Status message: Message accepted for delivery
Message ID: df984d51-49c1-43ca-902d-f1d3a09c3b7a
Recipient: +447698234415
```

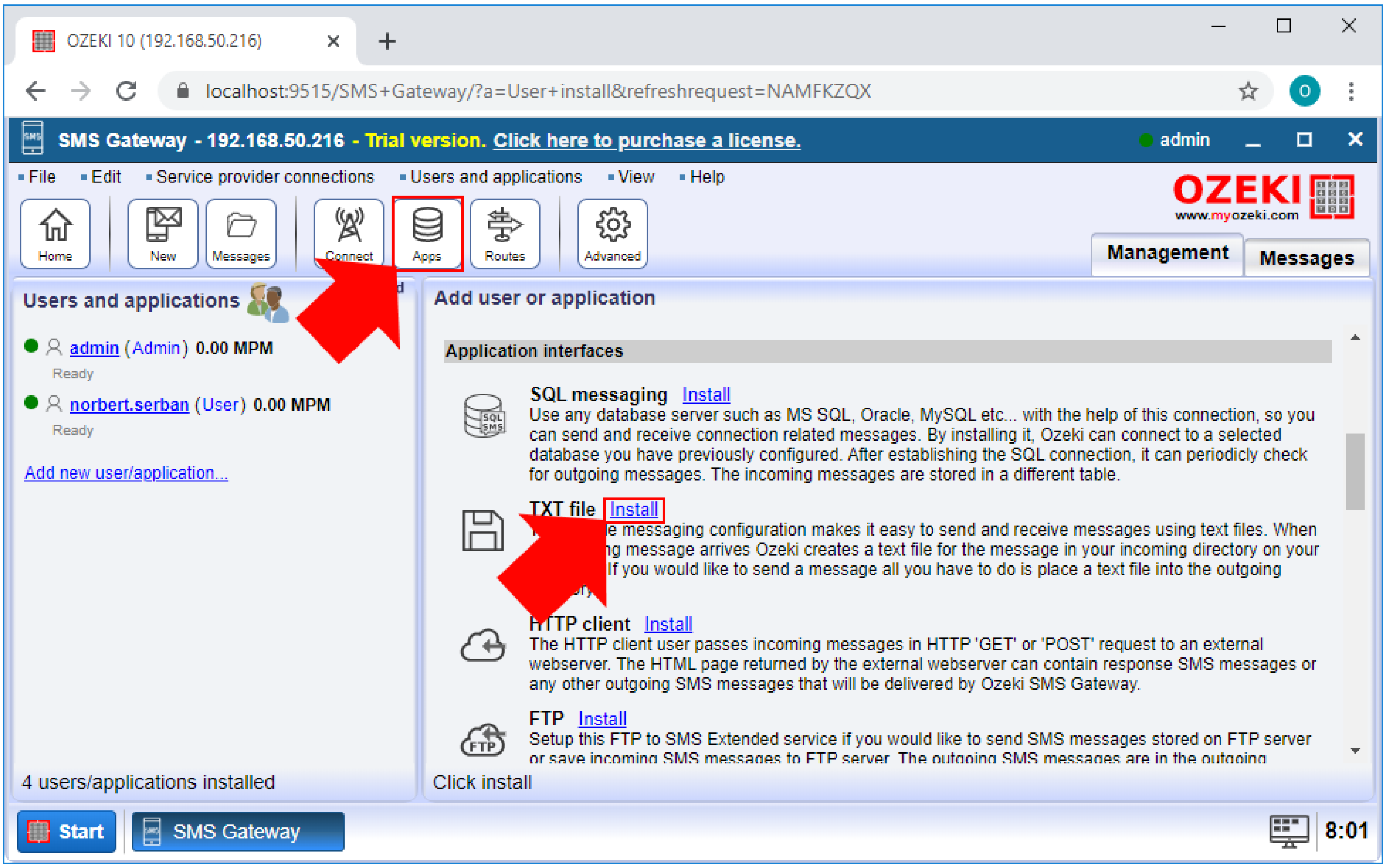
Figure 7 - Delivery report

# How to send SMS from TXT files

***The following guide is about to show how you can send message from the file or collecting the received messages in a file as well. This guide will show you how to install and configure a TXT File connection in SMS Gateway. This connection is capable of handling multiple file formats and you can also select the location of the following message boxes: Inbox, Outbox, Sent, Failed, Delivered.***

## Step 1 - Create a TXT File connection

The first step of the guide is to create the TXT File connection in SMS Gateway. First of all, you need to open the SMS Gateway application, and here, select the Apps menu from the toolbar. In this menu, go to the 'Application interfaces' section, and like in Figure 1, click on the Install button of the TXT File connection.



Use any database server such as MS SQL, Oracle, MySQL etc... with the help of this connection, so you can send and receive connection related messages. By installing it, Ozeki can connect to a selected database you have previously configured. After establishing the SQL connection, it can periodicly check for outgoing messages. The incoming messages are stored in a different table.

...e messaging configuration makes it easy to send and receive messages using text files. When ...ng message arrives Ozeki creates a text file for the message in your incoming directory on your ... If you would like to send a message all you have to do is place a text file into the outgoing ...ory.

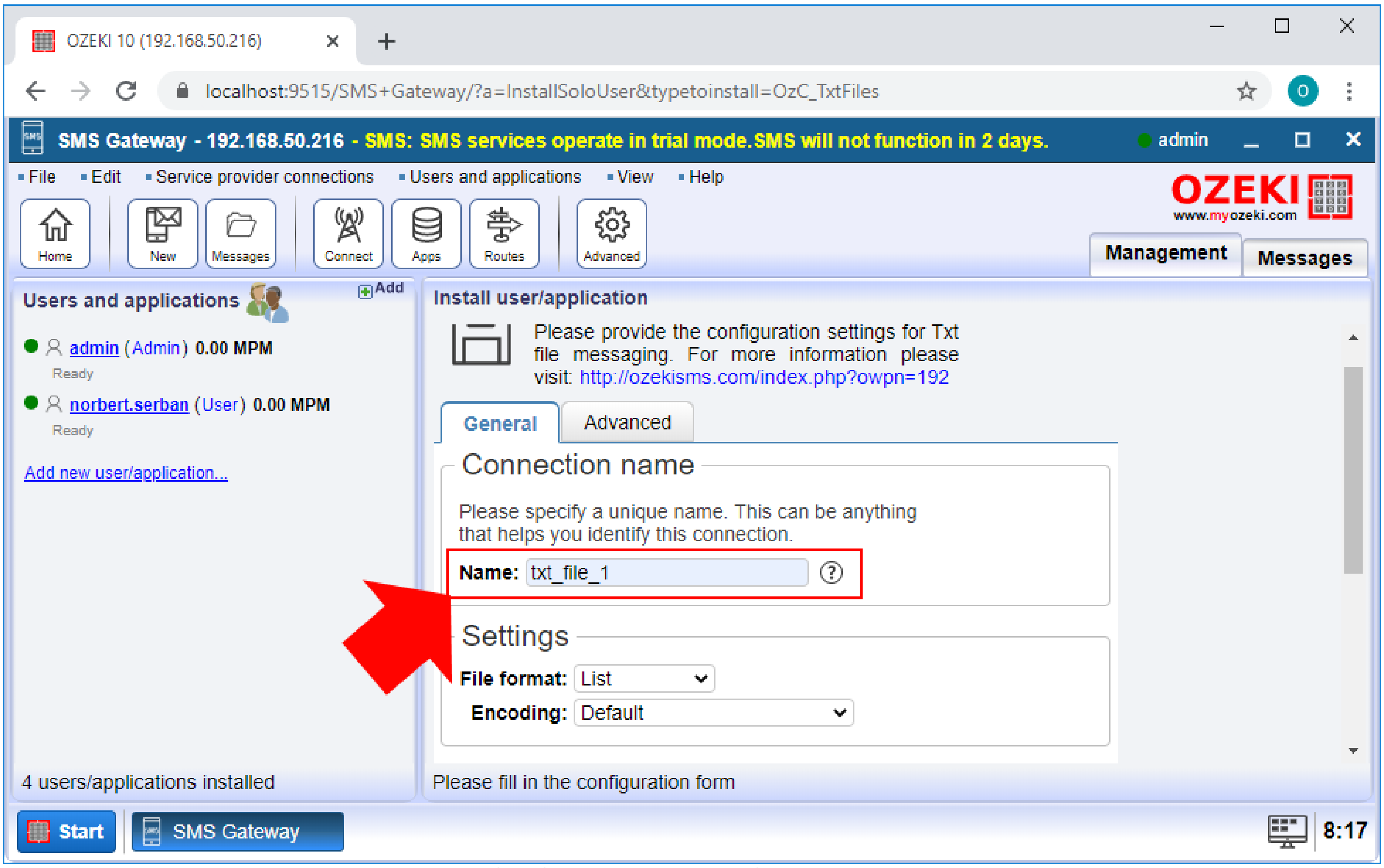
The HTTP client user passes incoming messages in HTTP 'GET' or 'POST' request to an external webserver. The HTML page returned by the external webserver can contain response SMS messages or any other outgoing SMS messages that will be delivered by Ozeki SMS Gateway.

Setup this FTP to SMS Extended service if you would like to send SMS messages stored on FTP server or save incoming SMS messages to FTP server. The outgoing SMS messages are in the outgoing

Figure 1 - Create a TXT File connection

## Step 2 - Configure the TXT File connection

After you selected the TXT File connection, you are going to be navigated to the configuration menu of the connection. In that menu, first, you need to define a name for that connection (Figure 2). Then, you have to select a file format for the SMS messages. These file formats will be discussed in the next step. Lastly, you can select folders for the messages.


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=InstallSoloUser&typetoinstall=OzC_TxtFiles
SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode.SMS will not function in 2 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Users and applications
Add
admin (Admin) 0.00 MPM
Ready
norbert.serban (User) 0.00 MPM
Ready
Add new user/application...
4 users/applications installed
Start
SMS Gateway
8:17

Install user/application

Please provide the configuration settings for Txt file messaging. For more information please visit: http://ozekisms.com/index.php?owpn=192

General | Advanced

Connection name

Please specify a unique name. This can be anything that helps you identify this connection.

Name: txt_file_1

Settings

File format: List

Encoding: Default

Please fill in the configuration form

Figure 2 - Configuration menu of TXT File connection

# Step 3 - Select a file format

For that connection, you can select from multiple file formats. These formats handle the messages differently. The first format is the 'Simple'. The format can handle one message per file. The name of the file needs to be the recipient's phone number and the message contains the SMS message itself.

c:\smsout\+3620310536

```
Hello world!
```

The next file format is the 'Verbose' format. This format also handles one message per file but you can specify the phone number of the sender in the file. The file extension of the Verbose format needs to be .sms.

c:\smsout\anyname.sms

```
+3620310536
+3620937723
Hello world!
```

The 'List' file format is capable of handle multiple messages and multiple recipients in the same file. Each row of the file contains the phone number of the recipient and the message. The file extension of the List format is .sml.

c:\smsout\list.sml

```
+3620454536 This is the first message
+3620993772 This is the second message
+3630778972 Use Ozeki SMS Gateway!
```

The 'Text' file format handles the messages the same way as the List format. The only difference between these two formats is the file extension. The List uses .sml format to manage the messages, the Text format uses the simple .txt format, so it is a little bit easier to create and manage.

c:\smsout\outbox.txt

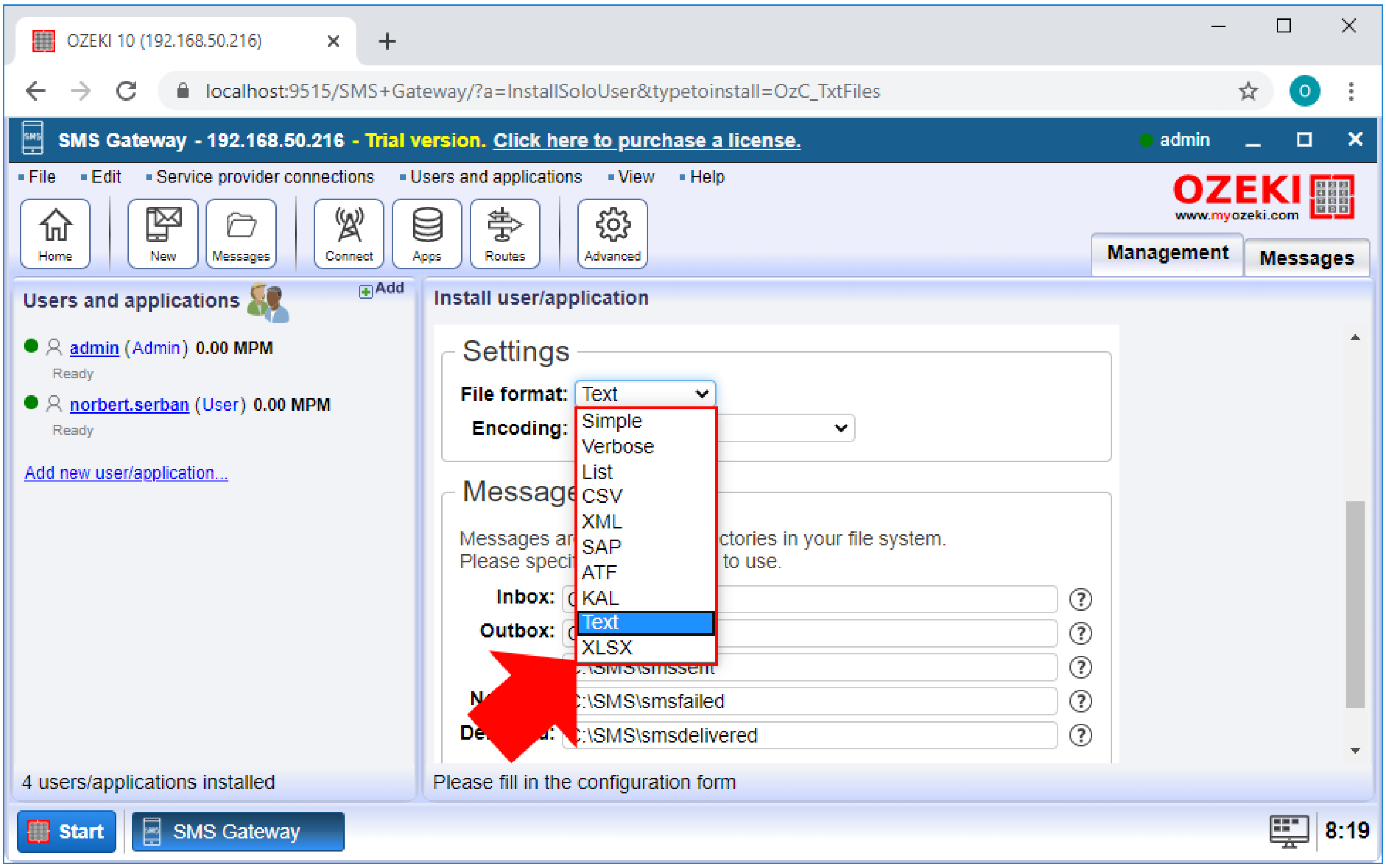
```
+3620454536 This is the first message
+3620993772 This is the second message
+3630778972 Use Ozeki SMS Gateway!
```

The 'XML' format is a great way to clearly organize your message in the dedicated XML file. This format uses tags to define the parts of a message. The 'originator' stores the phone number of the sender and the 'recipient' stores the recipient's phone number. The 'text' tag contains the message, the 'time' tag shows the time when the message should be sent. Lastly, the 'messagetype' is for defining the type of the message. This format is also capable of handling many messages in one file.

c:\smsout\outbox.xml

```
<messages>
  <message>
      <originator>ORINIGATOR_NUMBER</originator>
      <recipient>RECIPIENT_NUMBER</recipient>
      <text>Message text</text>
      <time>2020.07.07. 7:56:31</time>
      <messagetype>SMS:TEXT:FORMATTED</messagetype>
  </message>
</messages>
```

Figure 3 below shows all the file formats, that are available in SMS Gateway.


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=InstallSoloUser&typetoinstall=OzC_TxtFiles
SMS Gateway - 192.168.50.216 - Trial version. Click here to purchase a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
OZEKI
www.myozeki.com
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Users and applications
Add
Start
SMS Gateway
8:19

admin (Admin) 0.00 MPM
Ready

norbert.serban (User) 0.00 MPM
Ready

Add new user/application...

4 users/applications installed

Install user/application

Settings

File format: Text

Simple
Verbose
List
CSV
XML
SAP
ATF
KAL
Text
XLSX

Encoding:

Message

Messages ar ctories in your file system.
Please speci to use.

Inbox: C

Outbox: C

:\SMS\smssent

N :\SMS\smsfailed

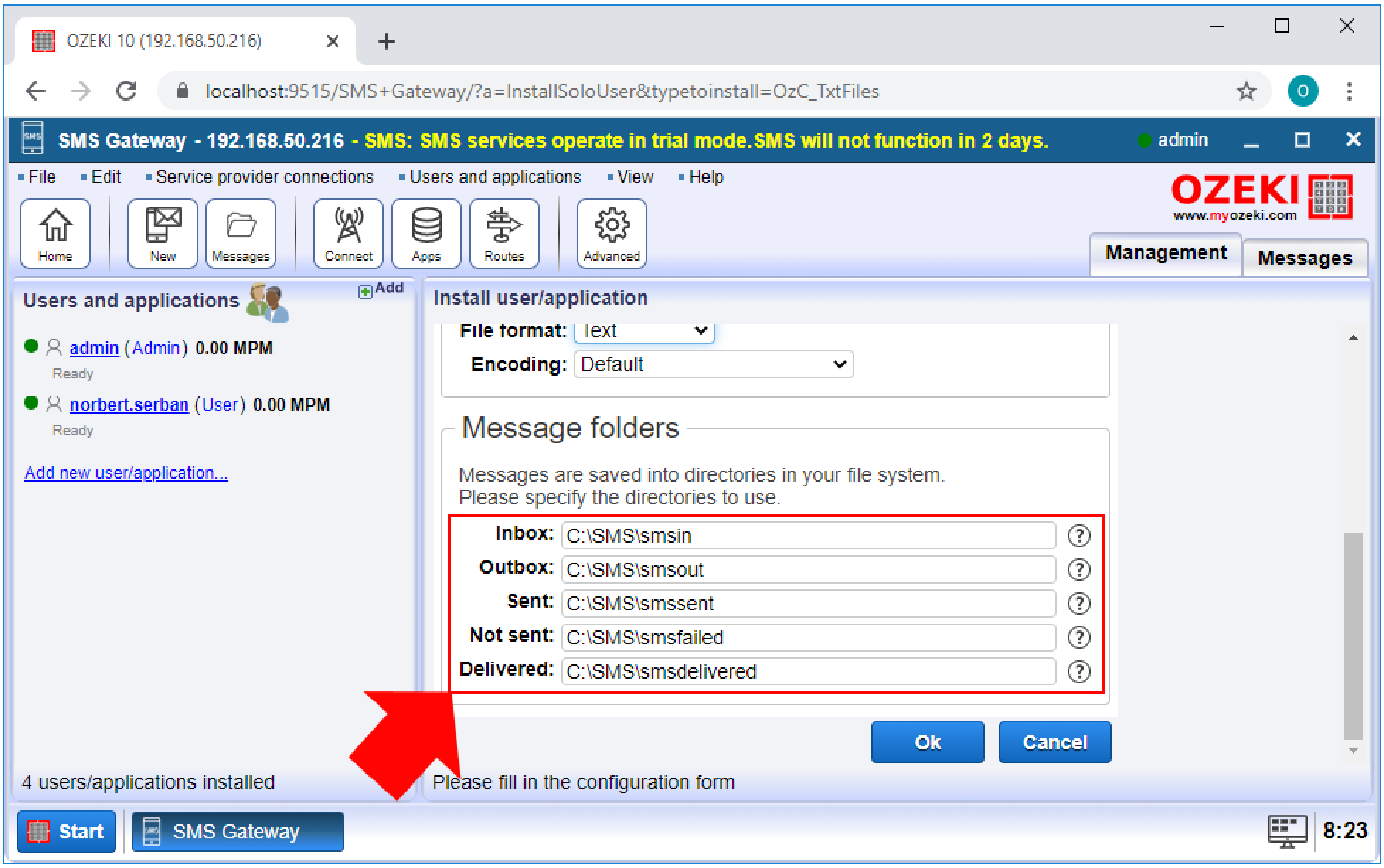
De d: :\SMS\smsdelivered

Please fill in the configuration form

Figure 3 - File formats for TXT File connection

## Step 4 - Select folders for the messages

In the next step, you need to set the location for the different types of messages regarding to they are about to be sent or received. The messages stored into five folders as you can see it in Figure 4: Inbox, Outbox, Sent, Not Sent and Delivered.


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=InstallSoloUser&typetoinstall=OzC_TxtFiles
SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode.SMS will not function in 2 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Users and applications
Add
admin (Admin) 0.00 MPM
Ready
norbert.serban (User) 0.00 MPM
Ready
Add new user/application...
4 users/applications installed
Install user/application
Ok
Cancel
Please fill in the configuration form
Start
SMS Gateway
8:23

File format: Text

Encoding: Default

Message folders

Messages are saved into directories in your file system.
Please specify the directories to use.

Inbox: C:\SMS\smsin

Outbox: C:\SMS\smsout

Sent: C:\SMS\smssent

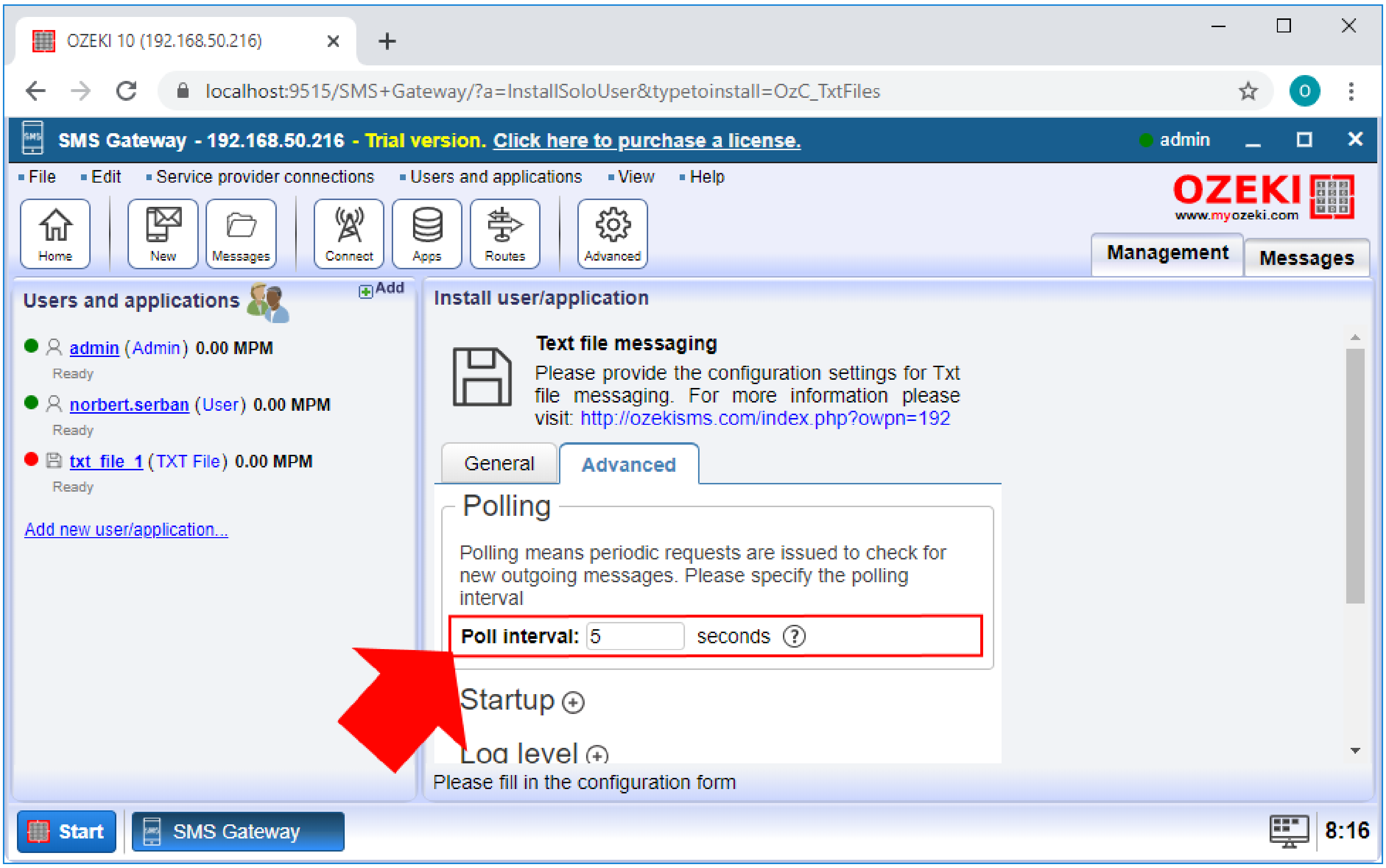
Not sent: C:\SMS\smsfailed

Delivered: C:\SMS\smsdelivered

Figure 4 - Configure the location for SMS files

# Step 5 - Advanced configuration of the TXT File connection

The TXT File connection can be further configured by clicking on the Advanced tab as Figure 5 shows that. Here, the main setting is the Polling. This option defines the time interval when the outbox folder will be checked, and the connection tries to deliver the messages from all files that are in that folder at that point. If you finished with the configuration, just click on OK.


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=InstallSoloUser&typetoinstall=OzC_TxtFiles
SMS Gateway - 192.168.50.216 - Trial version. Click here to purchase a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Users and applications
Add
admin (Admin) 0.00 MPM
Ready
norbert.serban (User) 0.00 MPM
Ready
txt_file_1 (TXT File) 0.00 MPM
Ready
Add new user/application...
Install user/application
General
Advanced
Startup
Log level
Start
SMS Gateway
8:16

**Text file messaging**

Please provide the configuration settings for Txt file messaging. For more information please visit: http://ozekisms.com/index.php?owpn=192

Polling

Polling means periodic requests are issued to check for new outgoing messages. Please specify the polling interval

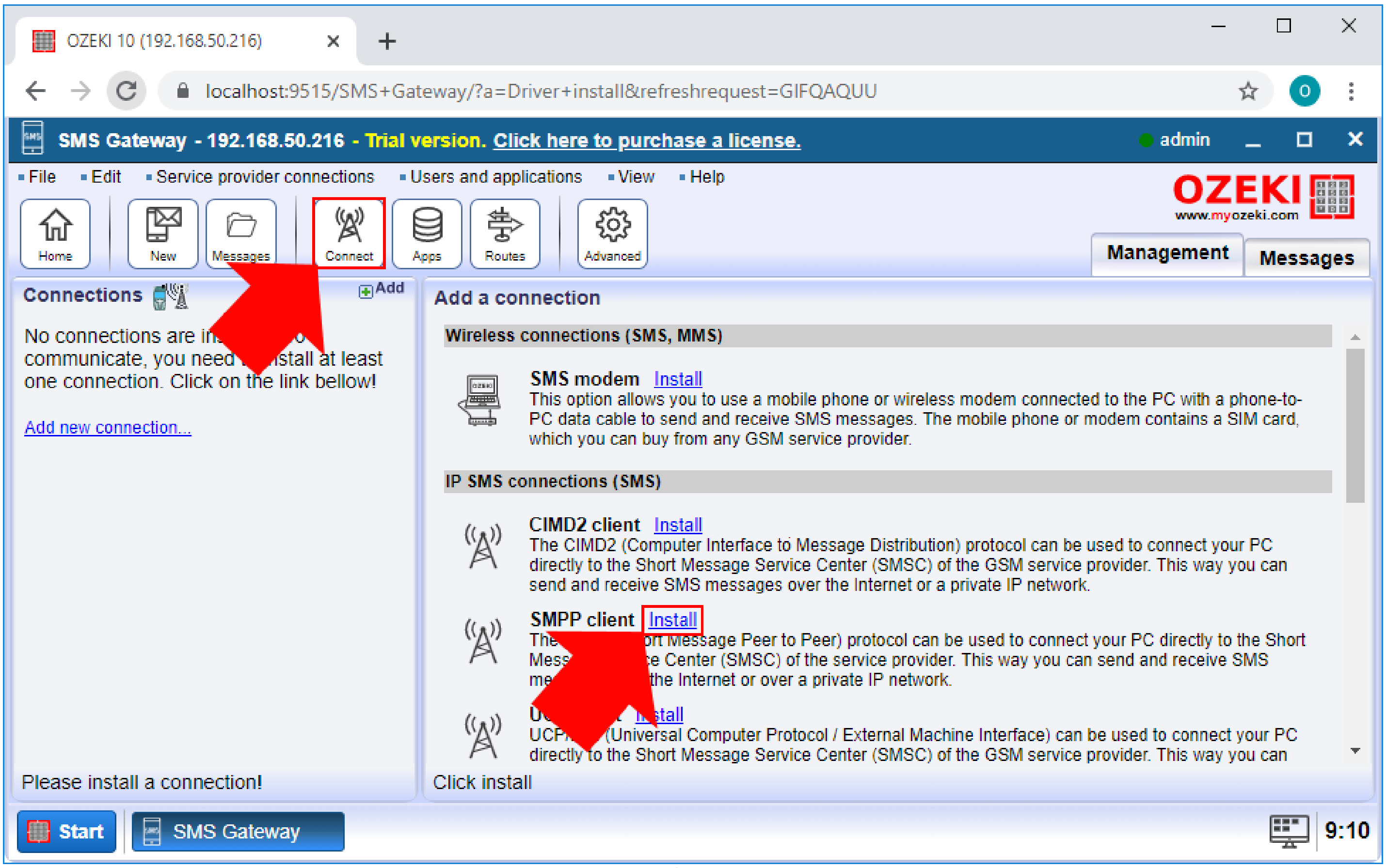
Poll interval: 5 seconds

Please fill in the configuration form

Figure 5 - Advanced configuration of TXT File connection

## Step 6 - Create an SMPP service provider connection

To be able to communicate with the members of the mobile network, you need a connection that creates that link. This connection will be the SMPP client connection. To create this connection, just select the Connect menu from the toolbar, and here as you can see it in Figure 6, click on the Install button of the SMPP client connection.


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=Driver+install&refreshrequest=GIFQAQUU
SMS Gateway - 192.168.50.216 - Trial version. Click here to purchase a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
OZEKI
www.myozeki.com
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Connections
Add
Start
SMS Gateway
9:10

No connections are in... communicate, you need ...stall at least one connection. Click on the link bellow!

Add new connection...

Please install a connection!

Add a connection

Wireless connections (SMS, MMS)

SMS modem Install
This option allows you to use a mobile phone or wireless modem connected to the PC with a phone-to-PC data cable to send and receive SMS messages. The mobile phone or modem contains a SIM card, which you can buy from any GSM service provider.

IP SMS connections (SMS)

CIMD2 client Install
The CIMD2 (Computer Interface to Message Distribution) protocol can be used to connect your PC directly to the Short Message Service Center (SMSC) of the GSM service provider. This way you can send and receive SMS messages over the Internet or a private IP network.

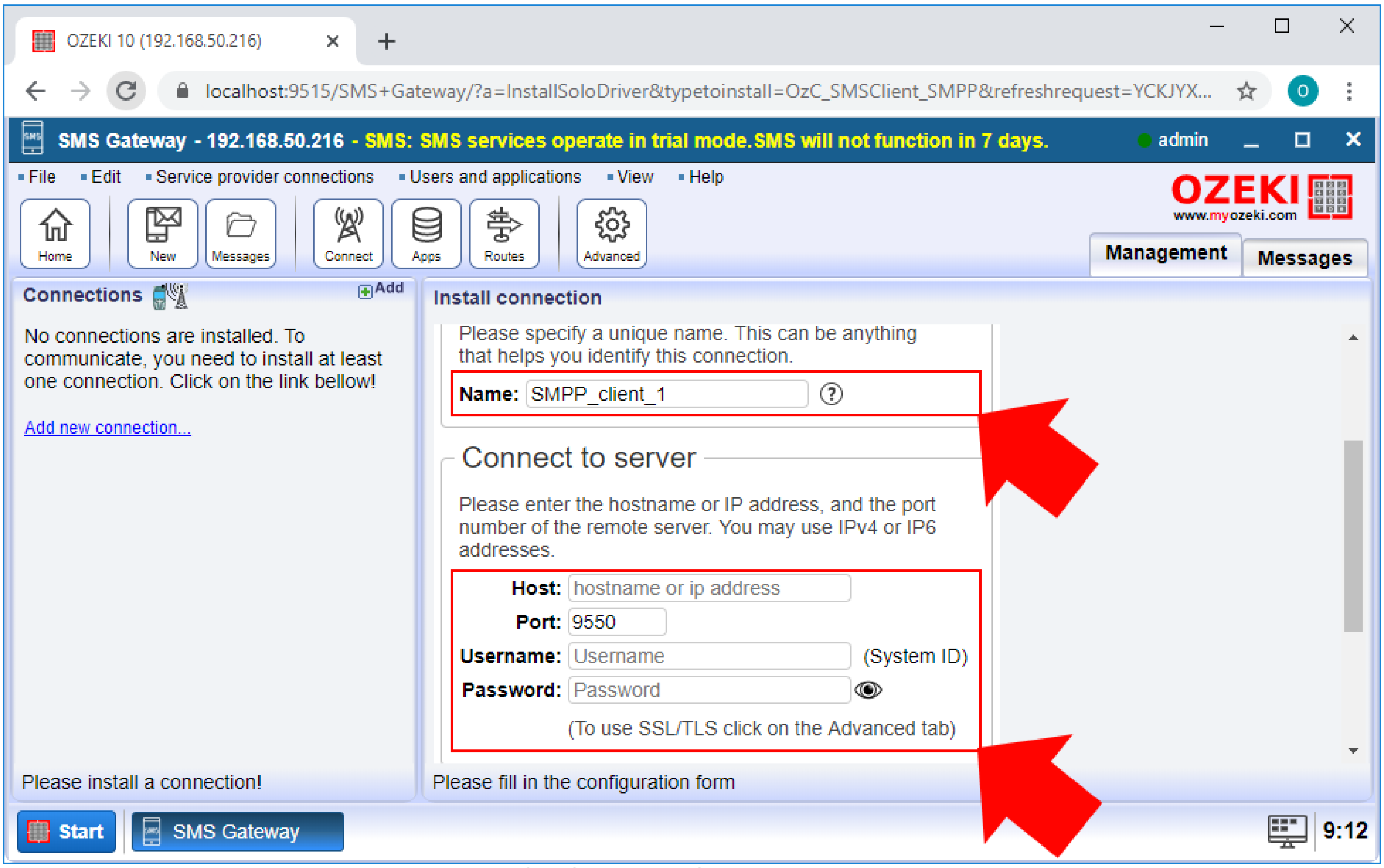
SMPP client Install
The ...ort Message Peer to Peer) protocol can be used to connect your PC directly to the Short Mess...e Center (SMSC) of the service provider. This way you can send and receive SMS me... the Internet or over a private IP network.

U... ...stall
UCP... (Universal Computer Protocol / External Machine Interface) can be used to connect your PC directly to the Short Message Service Center (SMSC) of the GSM service provider. This way you can

Click install

Figure 6 - Install the SMPP client connection

In the configuration menu of the connection, first, you need to type a name for the connection. Next, you have to provide the details of the SMPP server where you want that connection to connect to (Figure 7). Lastly, just type the phone number with that the connection is going to communicate with the mobile network. If you finished with everything, you can just click on OK.


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=InstallSoloDriver&typetoinstall=OzC_SMSClient_SMPP&refreshrequest=YCKJYX...
SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
OZEKI
www.myozeki.com
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Connections
Add
Start
SMS Gateway
9:12

No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

Please install a connection!

Install connection

Please specify a unique name. This can be anything that helps you identify this connection.

Name: SMPP_client_1

Connect to server

Please enter the hostname or IP address, and the port number of the remote server. You may use IPv4 or IP6 addresses.

Host: hostname or ip address
Port: 9550
Username: Username (System ID)
Password: Password

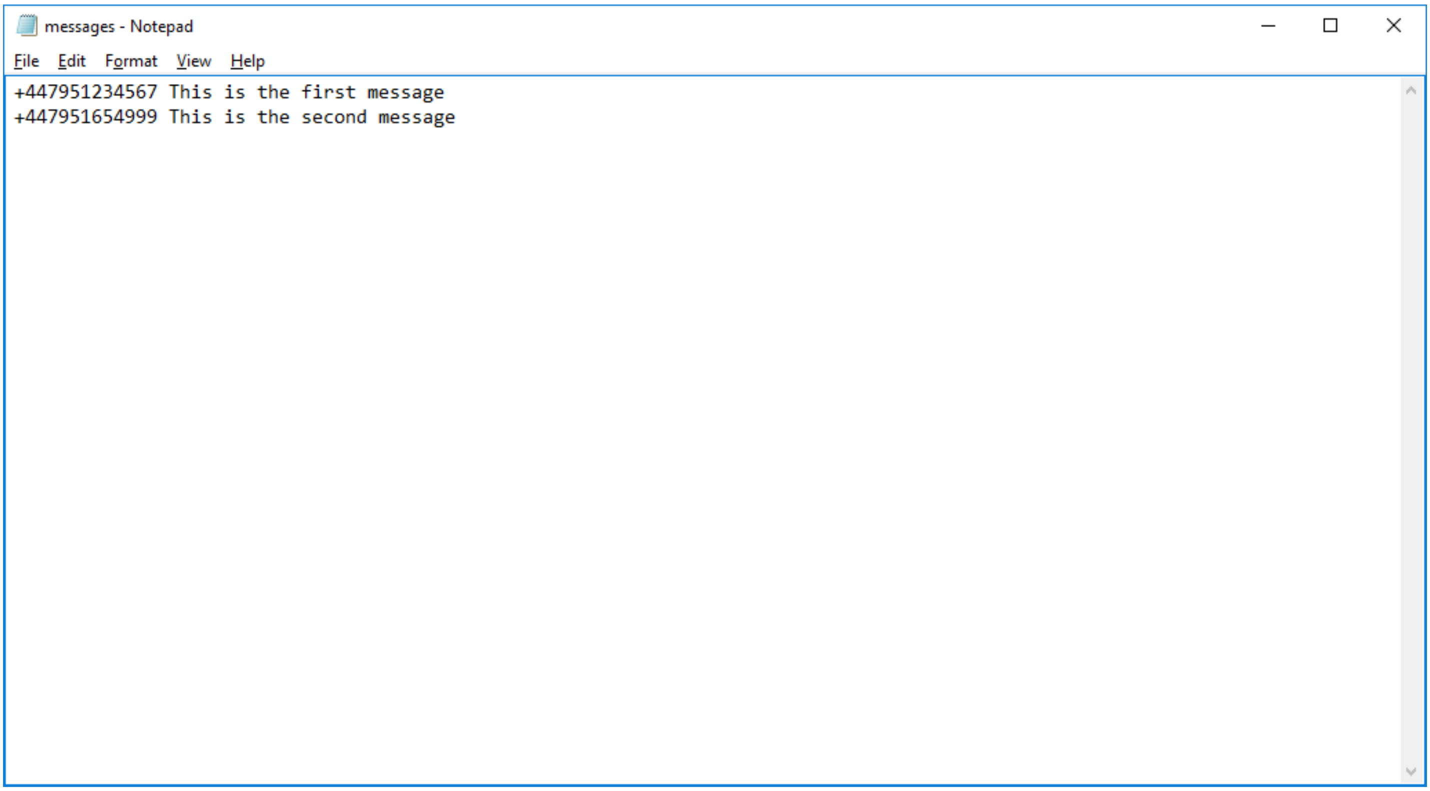
(To use SSL/TLS click on the Advanced tab)

Please fill in the configuration form

Figure 7 - Configure the SMPP client connection

# Step 7 - Send a message from file

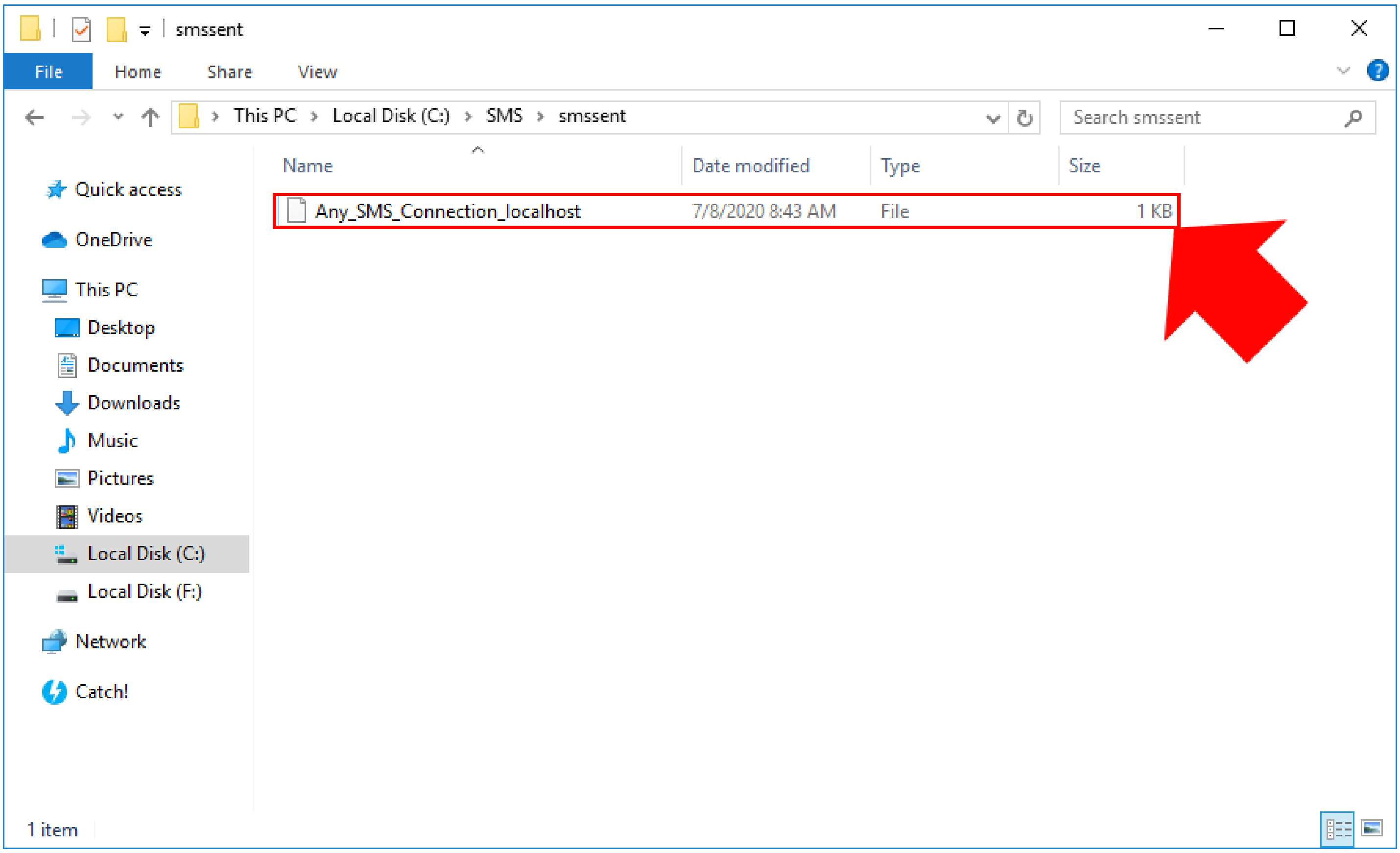
After you created the TXT file connection, now you will be able to send SMS messages using this connection. To do that, open the outbox folder in the File Explorer. Here, you can create a new text file, and as the format shows above, you can write multiple messages to multiple recipients in that file (Figure 8). When you finished, just save the file.



```
+447951234567 This is the first message
+447951654999 This is the second message
```

Figure 8 - Outbox folder with the message file

As soon as the TXT File connection polls the folders for files, the file from the outbox folder will disappear. This means the connection now processing the message file. As soon as it delivers the message, the message file will appear in the Sent folder as you can see it in Figure 9.

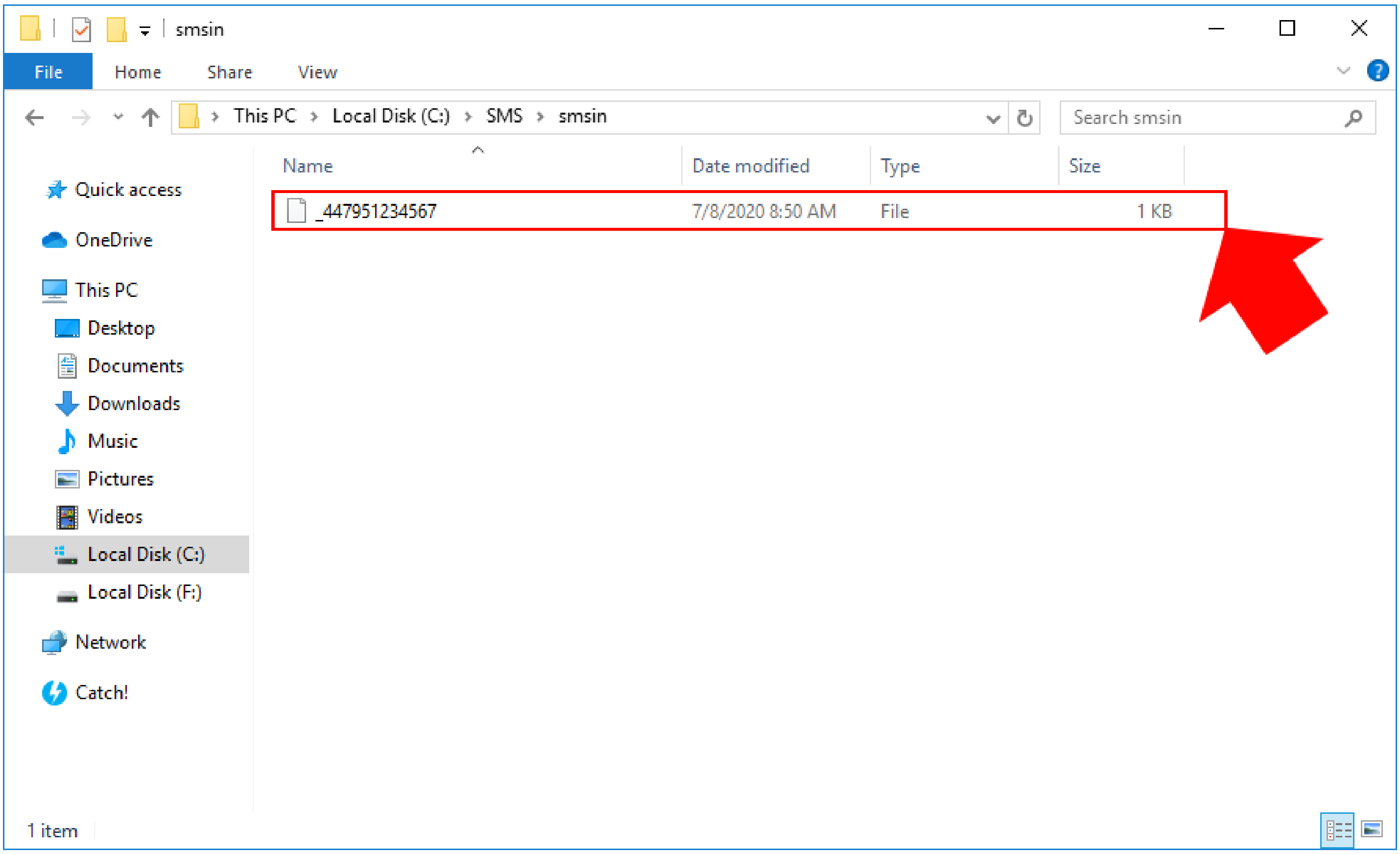


| Name | Date modified | Type | Size |
|---|---|---|---|
| Any_SMS_Connection_localhost | 7/8/2020 8:43 AM | File | 1 KB |

Figure 9 - Message file moved to the Sent folder

## Step 8 - Receive message with the TXT File connection

The creation of the TXT File connection also created a routing rule. This routing rule ensures that all the messages that SMS Gateway receives will be routed to that TXT File connection. The connection in case of an incoming message will create a file in the Inbox folder as you can see it in Figure 10. The format of this file is the same that you defined in the settings of the connection.



| Name | Date modified | Type | Size |
|---|---|---|---|
| _447951234567 | 7/8/2020 8:50 AM | File | 1 KB |

Figure 10 - Message file in the Inbox folder

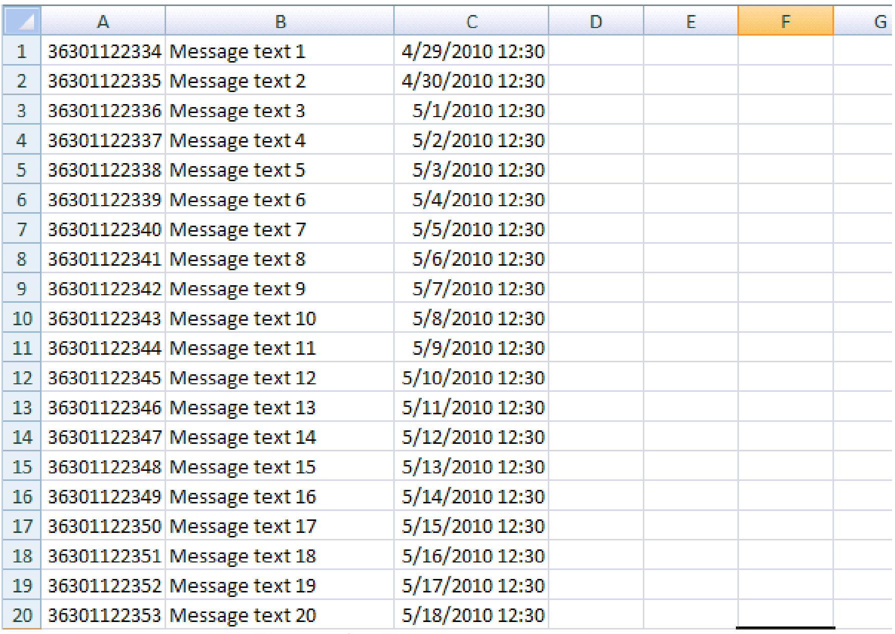
# How to use CSV files for SMS messaging

***See how to send SMS messages from excel sheets. You will only need a '.csv' file, a TXT File connection in Ozeki SMS Gateway and an SMS service provider connection. Keep in mind to write the excel sheet and export it to CSV format***

CSV is a much simpler format than an excel table. Each row in the CSV file represents a row in the excel sheet and the columns are separated by commas. You can send data in CSV file format with the help of the TXT File connection.

## Step 1 - Create a CSV file

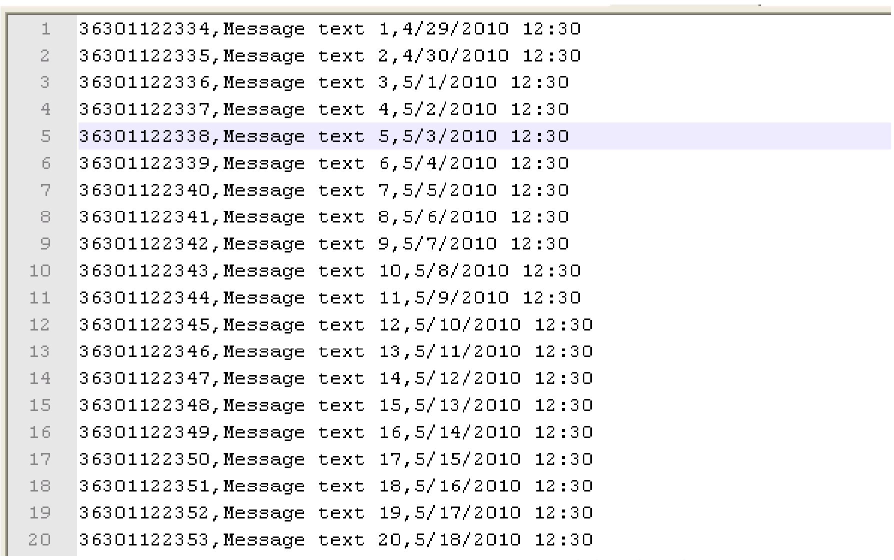
Write your own excel datasheet or your company's application can do it for you. It is important that the first column is the recipient's address, the second parameter should be the text message itself, while the third parameter is the time when to send it. You can see an example excel sheet on Figure 1.



|  | A | B | C | D | E | F | G |
|---|---|---|---|---|---|---|---|
| 1 | 36301122334 | Message text 1 | 4/29/2010 12:30 |  |  |  |  |
| 2 | 36301122335 | Message text 2 | 4/30/2010 12:30 |  |  |  |  |
| 3 | 36301122336 | Message text 3 | 5/1/2010 12:30 |  |  |  |  |
| 4 | 36301122337 | Message text 4 | 5/2/2010 12:30 |  |  |  |  |
| 5 | 36301122338 | Message text 5 | 5/3/2010 12:30 |  |  |  |  |
| 6 | 36301122339 | Message text 6 | 5/4/2010 12:30 |  |  |  |  |
| 7 | 36301122340 | Message text 7 | 5/5/2010 12:30 |  |  |  |  |
| 8 | 36301122341 | Message text 8 | 5/6/2010 12:30 |  |  |  |  |
| 9 | 36301122342 | Message text 9 | 5/7/2010 12:30 |  |  |  |  |
| 10 | 36301122343 | Message text 10 | 5/8/2010 12:30 |  |  |  |  |
| 11 | 36301122344 | Message text 11 | 5/9/2010 12:30 |  |  |  |  |
| 12 | 36301122345 | Message text 12 | 5/10/2010 12:30 |  |  |  |  |
| 13 | 36301122346 | Message text 13 | 5/11/2010 12:30 |  |  |  |  |
| 14 | 36301122347 | Message text 14 | 5/12/2010 12:30 |  |  |  |  |
| 15 | 36301122348 | Message text 15 | 5/13/2010 12:30 |  |  |  |  |
| 16 | 36301122349 | Message text 16 | 5/14/2010 12:30 |  |  |  |  |
| 17 | 36301122350 | Message text 17 | 5/15/2010 12:30 |  |  |  |  |
| 18 | 36301122351 | Message text 18 | 5/16/2010 12:30 |  |  |  |  |
| 19 | 36301122352 | Message text 19 | 5/17/2010 12:30 |  |  |  |  |
| 20 | 36301122353 | Message text 20 | 5/18/2010 12:30 |  |  |  |  |

Figure 1 - Example excel sheet

Please export your excel sheet to CSV file format (Figure 2).

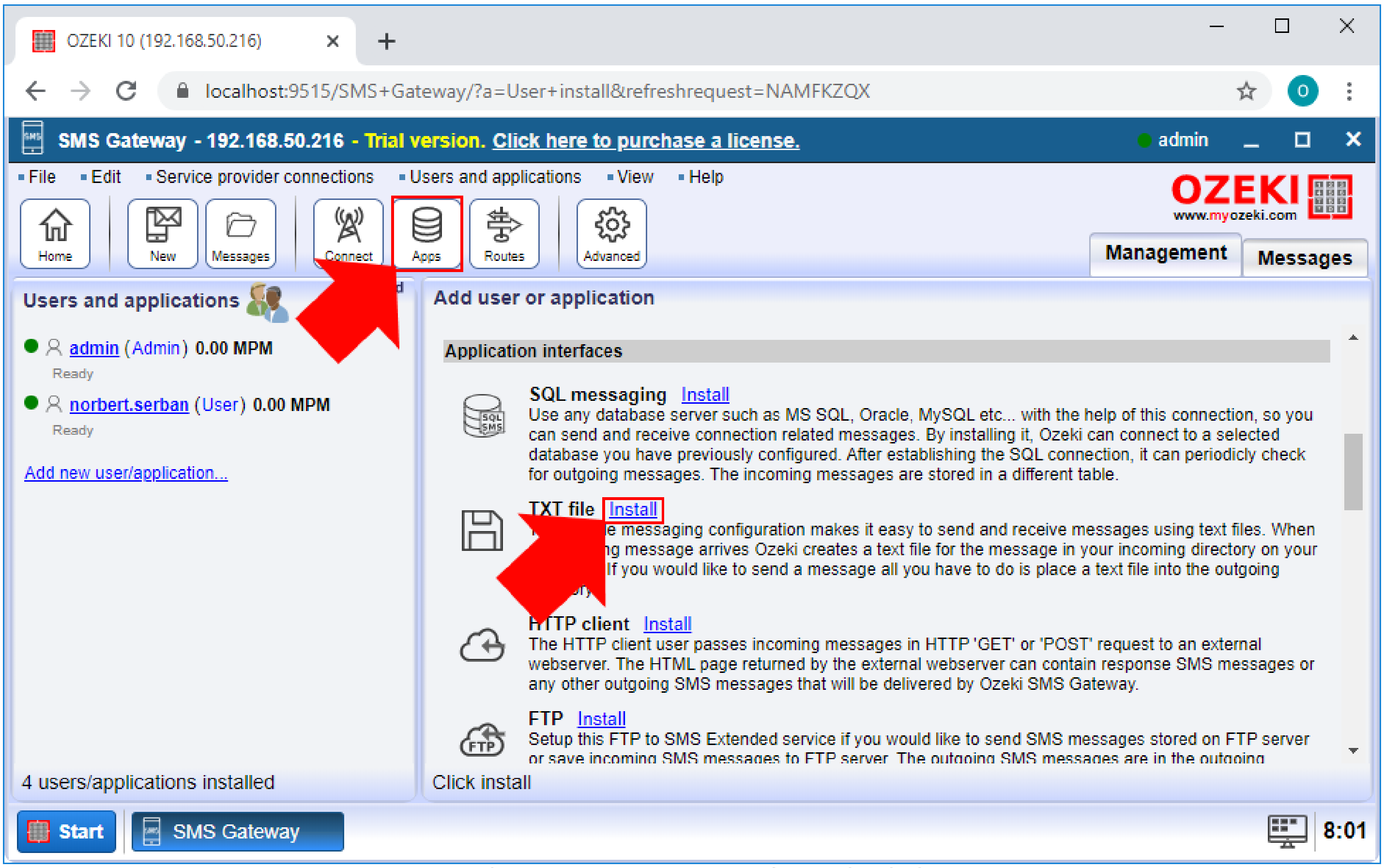


```
36301122334,Message text 1,4/29/2010 12:30
36301122335,Message text 2,4/30/2010 12:30
36301122336,Message text 3,5/1/2010 12:30
36301122337,Message text 4,5/2/2010 12:30
36301122338,Message text 5,5/3/2010 12:30
36301122339,Message text 6,5/4/2010 12:30
36301122340,Message text 7,5/5/2010 12:30
36301122341,Message text 8,5/6/2010 12:30
36301122342,Message text 9,5/7/2010 12:30
36301122343,Message text 10,5/8/2010 12:30
36301122344,Message text 11,5/9/2010 12:30
36301122345,Message text 12,5/10/2010 12:30
36301122346,Message text 13,5/11/2010 12:30
36301122347,Message text 14,5/12/2010 12:30
36301122348,Message text 15,5/13/2010 12:30
36301122349,Message text 16,5/14/2010 12:30
36301122350,Message text 17,5/15/2010 12:30
36301122351,Message text 18,5/16/2010 12:30
36301122352,Message text 19,5/17/2010 12:30
36301122353,Message text 20,5/18/2010 12:30
```

Figure 2 - Excel sheet exported to CSV file format

# Step 2 - Setup a TXT File connection in Ozeki

The next step is to create the TXT File connection. For that, you need to open the SMS Gateway application, and here, click on the Apps menu on the toolbar. In that menu, go to the 'Application interfaces' section, and as Figure 3 shows that, click on the Install button of the TXT File connection.


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=User+install&refreshrequest=NAMFKZQX
SMS Gateway - 192.168.50.216 - Trial version. Click here to purchase a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
OZEKI
www.myozeki.com
Management
Messages
Users and applications
Add user or application
Start
SMS Gateway
8:01

admin (Admin) 0.00 MPM
Ready
norbert.serban (User) 0.00 MPM
Ready
Add new user/application...

4 users/applications installed

Application interfaces

**SQL messaging** Install
Use any database server such as MS SQL, Oracle, MySQL etc... with the help of this connection, so you can send and receive connection related messages. By installing it, Ozeki can connect to a selected database you have previously configured. After establishing the SQL connection, it can periodicly check for outgoing messages. The incoming messages are stored in a different table.

**TXT file** Install
e messaging configuration makes it easy to send and receive messages using text files. When ng message arrives Ozeki creates a text file for the message in your incoming directory on your If you would like to send a message all you have to do is place a text file into the outgoing

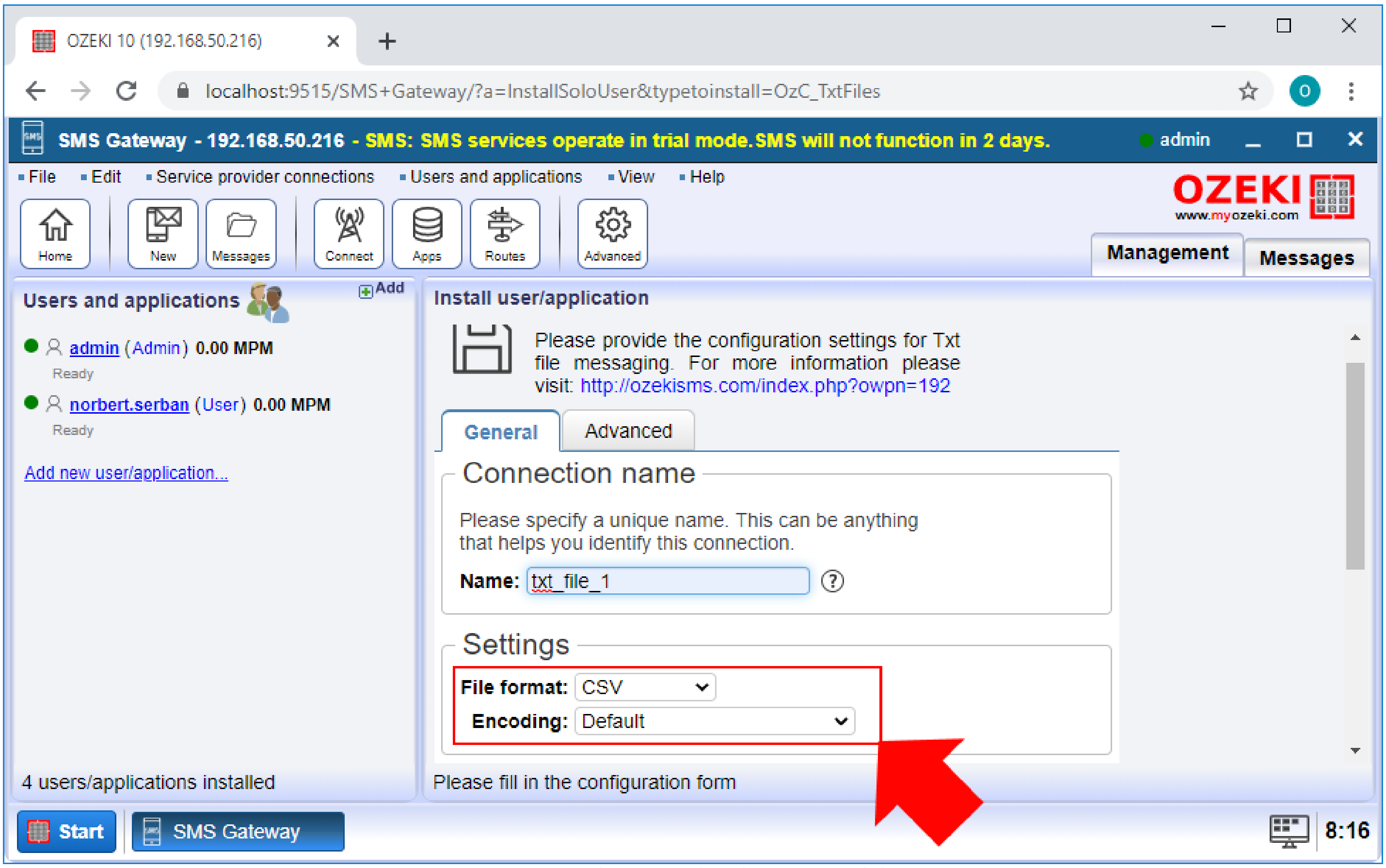
**HTTP client** Install
The HTTP client user passes incoming messages in HTTP 'GET' or 'POST' request to an external webserver. The HTML page returned by the external webserver can contain response SMS messages or any other outgoing SMS messages that will be delivered by Ozeki SMS Gateway.

**FTP** Install
Setup this FTP to SMS Extended service if you would like to send SMS messages stored on FTP server or save incoming SMS messages to FTP server. The outgoing SMS messages are in the outgoing

Click install

Figure 3 - Creating a File User in Ozeki SMS Gateway

To be able to handle CSV files with this connection, the main part of the configuration is to select the CSV file format from the list of file formats as you can see it in Figure 4. You also need to give a name for the connection here, you set the folders for the messages as well. When you finished with these settings, you can just click OK.


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=InstallSoloUser&typetoinstall=OzC_TxtFiles
SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode.SMS will not function in 2 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
OZEKI
www.myozeki.com
Management
Messages
Users and applications
Add
Start
SMS Gateway
8:16

admin (Admin) 0.00 MPM
Ready

norbert.serban (User) 0.00 MPM
Ready

Add new user/application...

4 users/applications installed

Install user/application

Please provide the configuration settings for Txt file messaging. For more information please visit: http://ozekisms.com/index.php?owpn=192

General | Advanced

Connection name

Please specify a unique name. This can be anything that helps you identify this connection.

Name: txt_file_1

Settings

File format: CSV

Encoding: Default

Please fill in the configuration form

Figure 4 - Change file format to CSV

Finally, you just have to copy the CSV you have created in the beginning of this tutorialto the default Outbox directory at C:\SMS\smsout . This is the path used by the TXT File connection. You can modify this location anytime, if you want to.

# How to send SMS from an FTP server

***Ozeki SMS Gateway's FTP User can synchronize directories with SMS Gateway through standard FTP, FTPS (over SSL/TLS) or SFTP (SSH FTP) protocols. Incoming SMS messages will be uploaded and outgoing SMS messages will be downloaded from the FTP server. You can also look at the accepted file formats.***

FTP User can synchronize directories. Synchronization is made between Ozeki SMS Gateway's machine and your FTP server on the following folder types: 'Inbox', 'Outbox', 'Sent', 'Failed', 'Delivered'. Please create these folders before providing the paths to the FTP User. This user is basicly used for sending or receiving text message files.

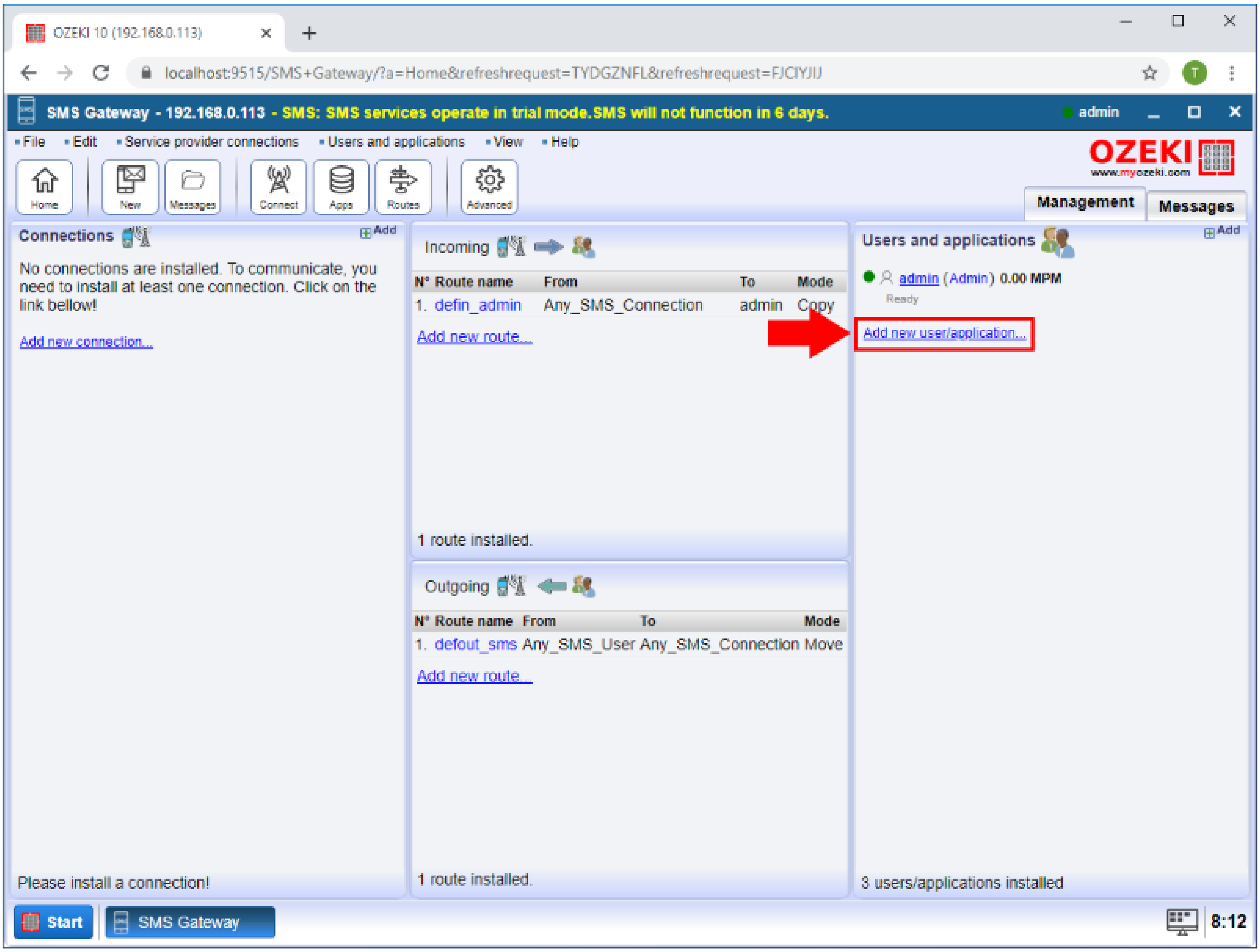
**It works in both directions:**

- If the user receives an incoming message, it creates a text file in the 'Inbox' folder of your FTP server and also a file in the 'Inbox' folder of your SMS Gateway's machine since the folders are synchronized.
- The user will send the messages from the FTP server's 'Outbox' folder. The message text file will be placed accordingly into the 'Sent', 'Failed' or 'Delivered' folder. The 'Outbox' folder will be checked periodically.

You can quickly install an FTP User through the browser GUI of Ozeki SMS Gateway.

## Step 1 - Install the FTP user

You can simply install the FTP User on the 'Management' console by clicking 'Add new user/application...' in the 'Users/Applications' panel (Figure 1).



No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defin_admin | Any_SMS_Connection | admin | Copy |

Add new route...

1 route installed.

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move |

Add new route...

1 route installed.

● admin (Admin) 0.00 MPM
Ready

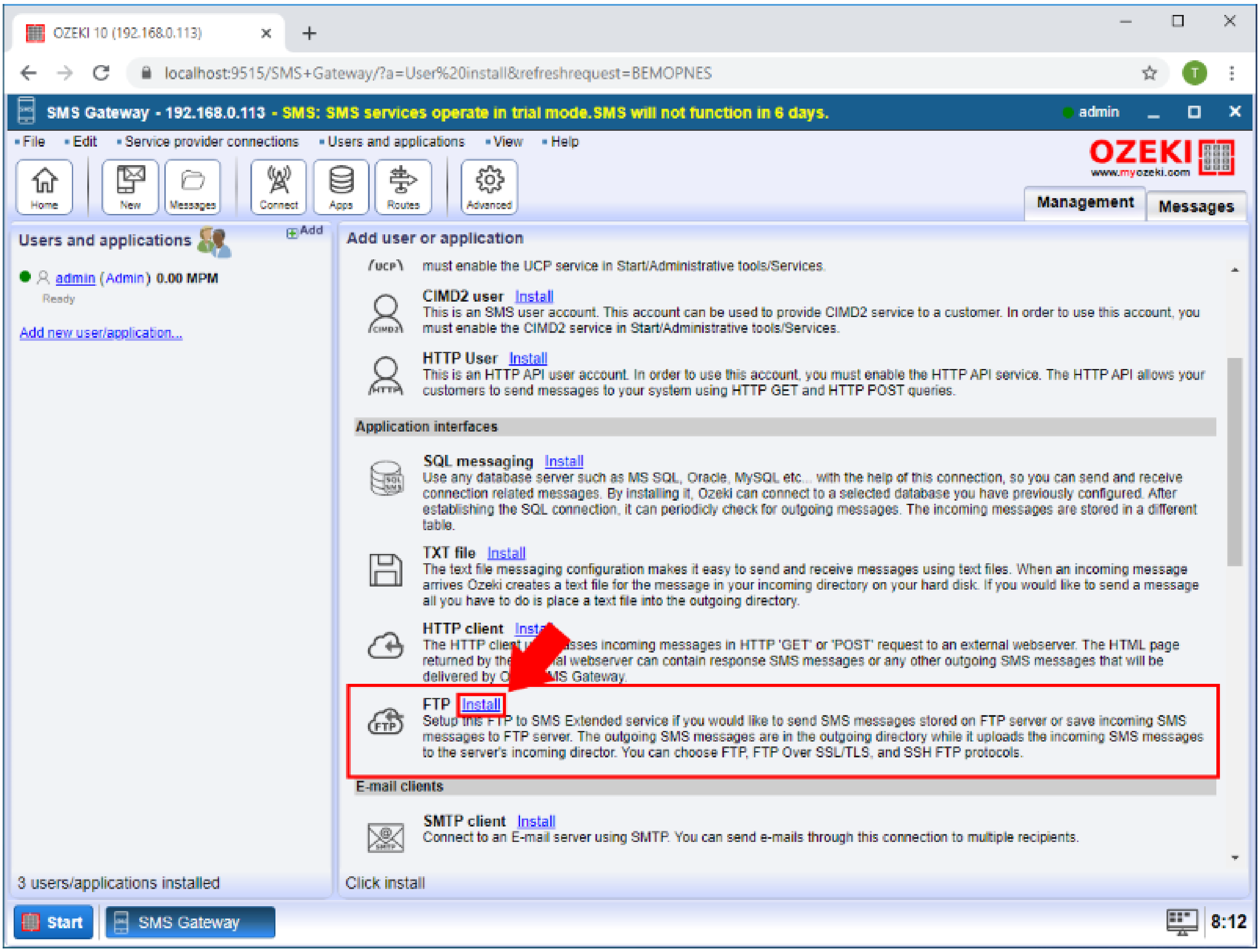
Add new user/application...

Please install a connection!

3 users/applications installed

Figure 1 - Click 'Add new user/application...'

An interface will open consisting of two panels. The left side panel contains the already installed users and applications. The right side panel contains the users and applications you can install with a brief description next to them. Search the FTP User and click the blue 'install' button next to it (Figure 2).



must enable the UCP service in Start/Administrative tools/Services.

CIMD2 user Install
This is an SMS user account. This account can be used to provide CIMD2 service to a customer. In order to use this account, you must enable the CIMD2 service in Start/Administrative tools/Services.

HTTP User Install
This is an HTTP API user account. In order to use this account, you must enable the HTTP API service. The HTTP API allows your customers to send messages to your system using HTTP GET and HTTP POST queries.

SQL messaging Install
Use any database server such as MS SQL, Oracle, MySQL etc... with the help of this connection, so you can send and receive connection related messages. By installing it, Ozeki can connect to a selected database you have previously configured. After establishing the SQL connection, it can periodicly check for outgoing messages. The incoming messages are stored in a different table.

TXT file Install
The text file messaging configuration makes it easy to send and receive messages using text files. When an incoming message arrives Ozeki creates a text file for the message in your incoming directory on your hard disk. If you would like to send a message all you have to do is place a text file into the outgoing directory.

HTTP client Install
The HTTP client passes incoming messages in HTTP 'GET' or 'POST' request to an external webserver. The HTML page returned by the external webserver can contain response SMS messages or any other outgoing SMS messages that will be delivered by Ozeki SMS Gateway.

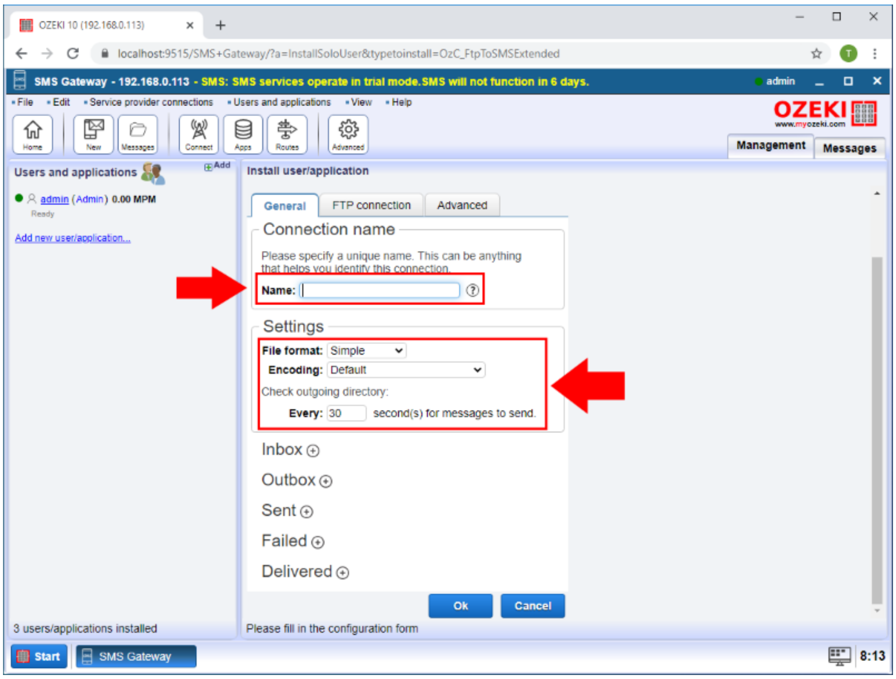
FTP Install
Setup this FTP to SMS Extended service if you would like to send SMS messages stored on FTP server or save incoming SMS messages to FTP server. The outgoing SMS messages are in the outgoing directory while it uploads the incoming SMS messages to the server's incoming director. You can choose FTP, FTP Over SSL/TLS, and SSH FTP protocols.

SMTP client Install
Connect to an E-mail server using SMTP. You can send e-mails through this connection to multiple recipients.

Figure 2 - Install User

# Step 2 - Configure the FTP connection

The 'Configuration' panel has a 'General' tab which contains the basic file settings. First please provide a unique username. In the Settings section of tabpage contains a combobox where you can select the file format 'Simple', 'Verbose', 'List', 'CSV' or 'XML'. Please check the File format section below for more information. There is another combobox where you can select the character encoding. You can also set how much time to wait before rechecking the 'Outbox'. The default value is 30, but you can change it to any positive whole number (Figure 3).


OZEKI 10 (192.168.0.113)
localhost:9515/SMS+Gateway/?a=InstallSoloUser&typetoinstall=OzC_FtpToSMSExtended
SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.SMS will not function in 6 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Users and applications
Add
admin (Admin) 0.00 MPM
Ready
Add new user/application...
3 users/applications installed
Start
SMS Gateway
8:13

Install user/application

General | FTP connection | Advanced

Connection name

Please specify a unique name. This can be anything that helps you identify this connection.

Name:

Settings

File format: Simple

Encoding: Default

Check outgoing directory:

Every: 30 second(s) for messages to send.

Inbox ⊕

Outbox ⊕

Sent ⊕

Failed ⊕

Delivered ⊕

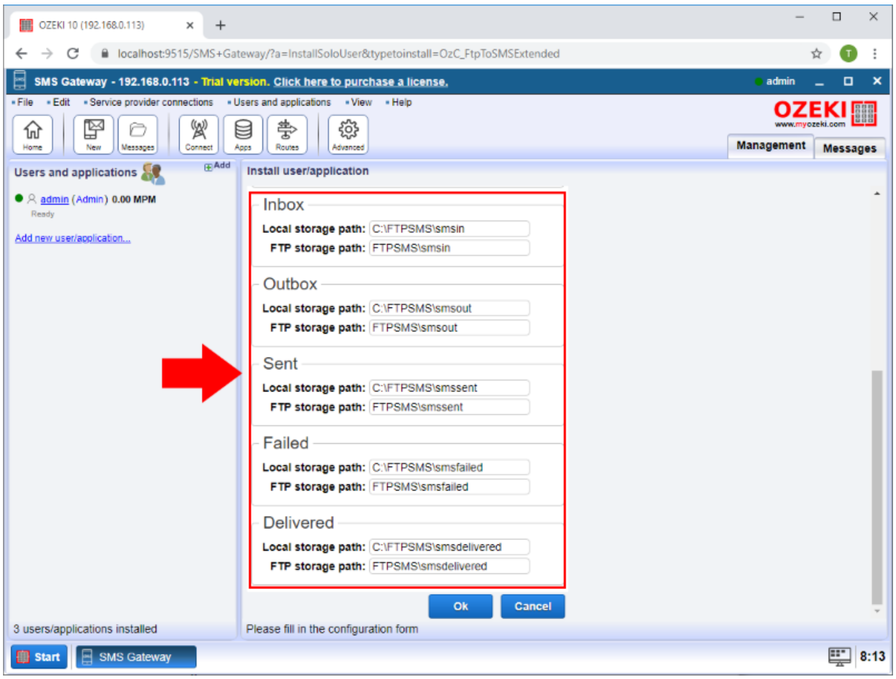
Ok | Cancel

Please fill in the configuration form

Figure 3 - Select the file format you prefer

# Step 3 - Configure the folders for messages

The lower section of the tabpages contains five sub-section referring to the five message folder types: 'Inbox', 'Outbox', 'Sent', 'Failed' or 'Delivered' (Figure 4). You can select each message folder's directory path on Ozeki SMS Gateway's machine and your FTP server. The FTP User will keep the directory pairs synchronized.


OZEKI 10 (192.168.0.113)
localhost:9515/SMS+Gateway/?a=InstallSoloUser&typetoinstall=OzC_FtpToSMSExtended
SMS Gateway - 192.168.0.113 - Trial version. Click here to purchase a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Users and applications
Add
admin (Admin) 0.00 MPM
Ready
Add new user/application...
3 users/applications installed
Start
SMS Gateway
8:13

Install user/application

Inbox

Local storage path: C:\FTPSMS\smsin

FTP storage path: FTPSMS\smsin

Outbox

Local storage path: C:\FTPSMS\smsout

FTP storage path: FTPSMS\smsout

Sent

Local storage path: C:\FTPSMS\smssent

FTP storage path: FTPSMS\smssent

Failed

Local storage path: C:\FTPSMS\smsfailed

FTP storage path: FTPSMS\smsfailed

Delivered

Local storage path: C:\FTPSMS\smsdelivered

FTP storage path: FTPSMS\smsdelivered

Ok | Cancel

Please fill in the configuration form

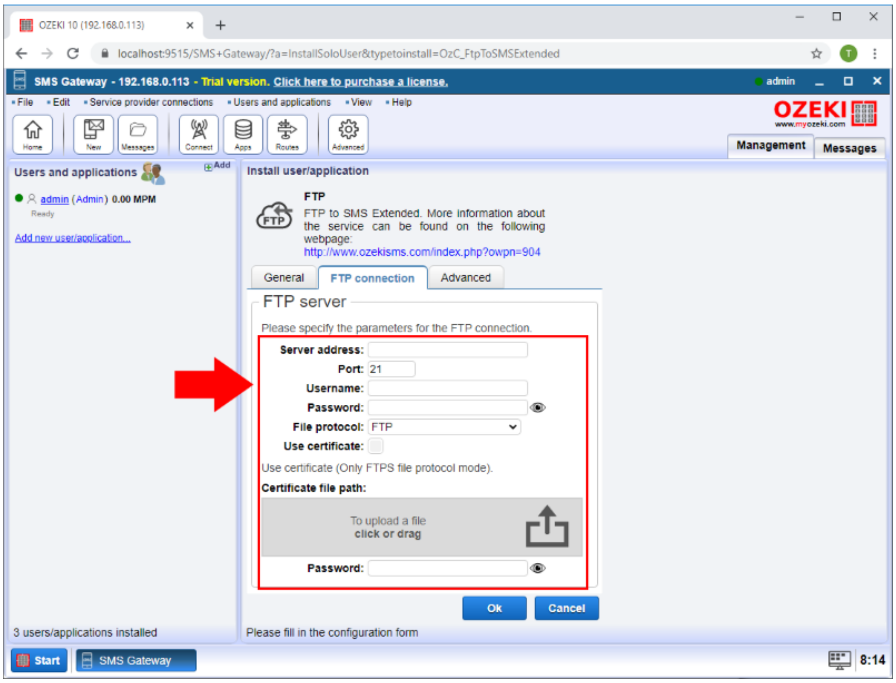
Figure 4 - Select the file path for each 'Inbox', 'Outbox', 'Sent', 'Failed' or 'Delivered' folder

**What kind of messages are contained in each folder**

- **Inbox:** Aready delivered messages to the user. Set it by setting Ozeki SMS Gateway's inbound routing.
- **Outbox:** Messages to be sent by the user. Set it by setting Ozeki SMS Gateway's outbound routing.
- **Sent:** Accepted messages for delivery by Ozeki SMS Gateway and the SMS service provider.
- **Failed:** Refused messages by Ozeki SMS Gateway or the SMS service provider.
- **Delivered:** Messages that received delivery success report by the SMS service provider.

# Step 4 - Configure FTP server details

If you have set the folders, you should switch to the **FTP Connection** tab to configure the FTP server connections. First provide the server's IP address, port number and the required username and password. Then select the protocol mode you prefer (FTP, FTPS, STFP) (Figure 5). In the lower section you can provide a certificate if you are using the FTPS file protocol mode. Please provide the certificate path and certificate password to add the certificate.



Install user/application

FTP

FTP to SMS Extended. More information about the service can be found on the following webpage: http://www.ozekisms.com/index.php?owpn=904

FTP server

Please specify the parameters for the FTP connection.

Server address:
Port: 21
Username:
Password:
File protocol: FTP
Use certificate: □

Use certificate (Only FTPS file protocol mode).

Certificate file path:

To upload a file
click or drag

Password:

Please fill in the configuration form

Figure 5 - Switch to the 'FTP connection' tab to configure your FTP server connection

# Step 5 - Learn about file formats

File formats are used to define how to store messages in text files. Various file formats are supported by Ozeki SMS Gateway's FTP to SMS Extended User: simple, verbose, list, XML, CSV. These formats can be used for sending or receiving SMS messages through a GSM network.

'Simple' is set as default on all FTP to SMS Extended User. The filename itself stores the sender or recipient, depending on if it is an incoming or outgoing message. The file contains the message content. Simple is the default format. To change it, please click on the 'General' tab. For example:

c:\SMS\outbox\+3620310366

```
Hello world!
```

c:\SMS\inbox\+3630310300

```
Hello Ozeki SMS Gateway
```

It has a side effect hidden in the name convention. More files cannot have the same name in a single directory. This means only one message can be sent at a time to the same recipient. If sending a new message to the same recipient, please wait until the old message gets deleted by Ozeki SMS Gateway. This works vice versa. As an example only one message at a time can be received by the sender, since older received message will be deleted. Please consider these facts before keeping this default file format or choose another format from the dropdown menu. It is a very useful tool for position tracking, since every new coordinate overwrites the old one. This helps you easily track vehicles that send coordinates in SMS messages.

The Verbose format stores every SMS message in a separate file, although each file can have a unique filename. Since unique filenames are supported, it can store multiple message files from the same sender and multiple message files from the same recipient as well. This means it can contain more information then a 'Simple' file format. Each file has an extension. A verbose file's extension is '.sms'. Do not forget that each message file contains a single message. Now let's check the structure. The first line contains the sender's phone number, while the second line is the recipient's phone number. The message text is contained in the third line. Look at a simple Verbose example:

c:\SMS\outbox\anyname.sms

```
+36209937723
The message text
```

The List format is capable of storing multiple SMS messages in a single file. A single '.sml' file can store multiple SMS messages in a simple list. This is the most comfortable way to send and receive a massive amount of text messages. Please look at the following file example:

c:\SMS\outbox\list.sml

```
+3620993723 This is the second message
+3630778972 Use Ozeki SMS Gateway!
```

Each line represents an SMS message. A single line consists of two important information separated by space. The first is the phone number of the recipient while the second is the message content. To send SMS message please place your '.sml' file into the Outbox folder.

You can send and receive SMS messages using 'XML' file format as you can see in the XML message below. Do not forget to use '.xml' file extension.

**<originator>:** The sender's phone number is between these tags.
**<recipient>:** The recipient's phone number is between these tags.
**<text>:** Type text message between these tags.
**<time>:** Set when the message should be sent.
**<messagetype>:** Declare the message type between these tags.

c:\SMS\outbox\mysms.xml

```
  <message>
      <originator>ORINIGATOR_NUMBER</originator>
      <recipient>RECIPIENT_NUMBER</recipient>
      <text>Message text</text>
      <time>2011.08.25. 7:56:31</time>
      <messagetype>SMS:TEXT:FORMATTED</messagetype>
  </message>
</messages>
```

You can export any excel file in CSV file format. This CSV structure is a text file format where the original columns are separated by commas. Each line corresponds to a row in the original excel datasheet. Finally the FTP to SMS Extended User is capable to read each row as you can see in the following example.

c:\SMS\outbox\mysms_messages.CSV

```
36205555366, Message Text 1, 3/30/2016 12:30
36209937723, Message Text 2, 3/30/2016 12:30
36307789723, Message Text 3, 3/30/2016 12:30
```

The first parameter is the recipient's or sender's address, the second parameter is the message text while the third parameter is when to send the message or when the message have arrived.

# How to start an Application with SMS

***Ozeki SMS Gateway's Application Starter User can run any process or application in case an SMS message arrives. You simply need to provide the file path. You can also fetch parameters from received SMS messages to use them as process parameters or command line arguments.***

Look at the workflow diagram in Figure 1 before scrolling through this step-by-step guide. Keep in mind the incoming message must arrive at Ozeki SMS Gateway's Application Starter User, so it can start any desired process or application.

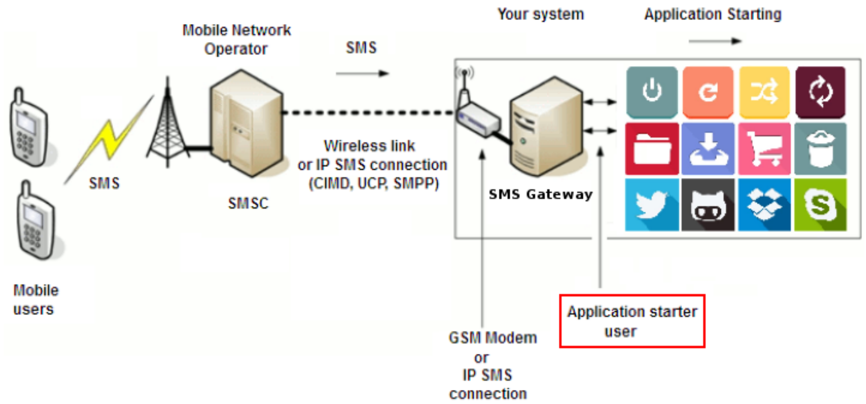
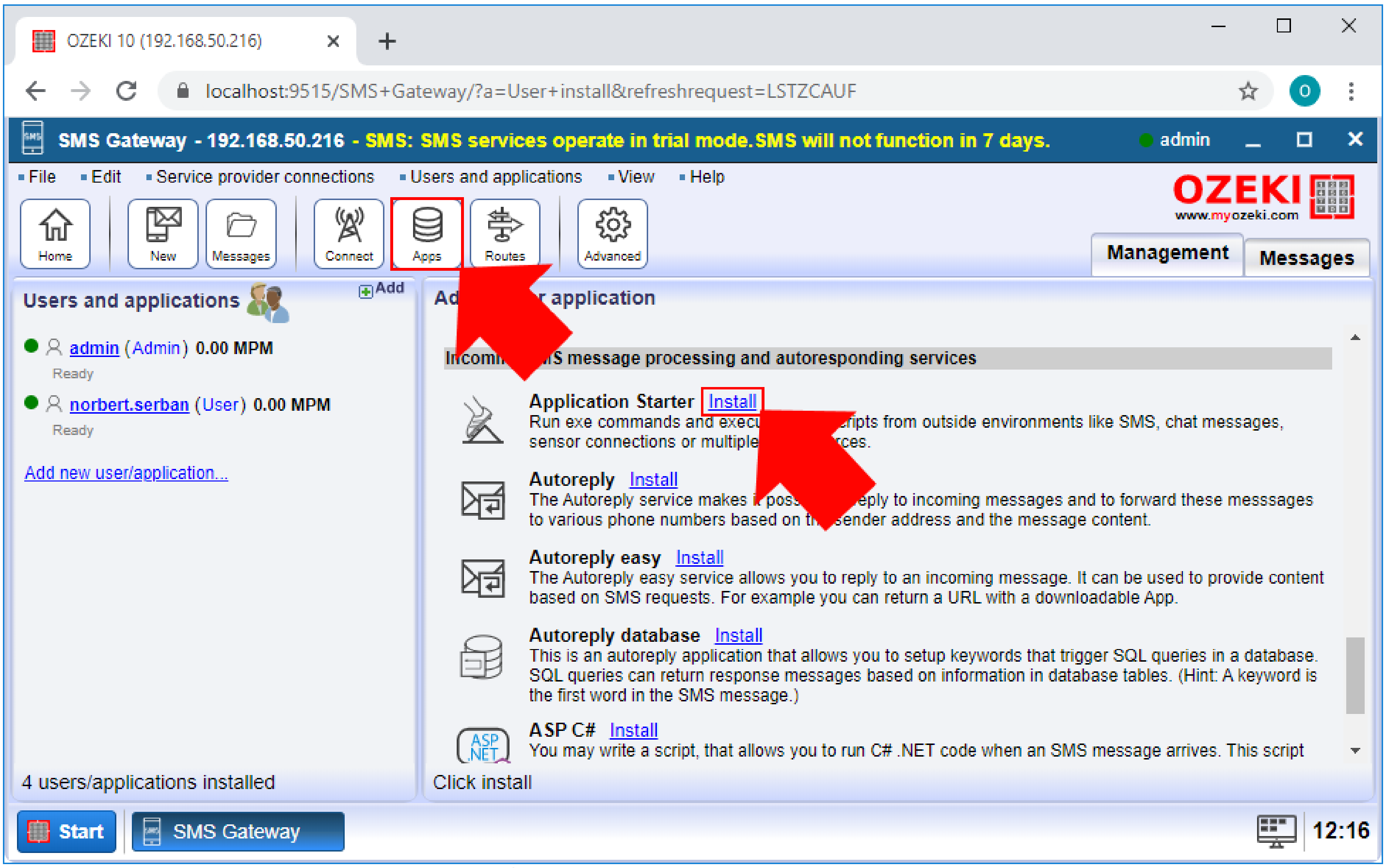


Figure 1 - How an application gets started after receiving a message

You will find out how your application starts with a glimpse of an SMS arrival. First, it is advised to install and configure your own Application Starter User in case an SMS is received. You should make sure your Ozeki SMS Gateway is ready to receive any SMS from an SMPP provider or a GSM modem.

## Step 1 - Create the Application starter connection

The first step is to open the SMS Gateway application. Here, in the main menu, you should select the Apps menu on the toolbar. In this menu, just scroll down to the 'Incoming SMS message processing and autoresponding services' section, and here, click on the Install button of the Application starter connection (Figure 2).


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=User+install&refreshrequest=LSTZCAUF
SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode. SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Users and applications
Add
admin (Admin) 0.00 MPM
Ready
norbert.serban (User) 0.00 MPM
Ready
Add new user/application...
4 users/applications installed
Click install
Start
SMS Gateway
12:16

Incoming SMS message processing and autoresponding services

Application Starter Install
Run exe commands and execute scripts from outside environments like SMS, chat messages, sensor connections or multiple other sources.

Autoreply Install
The Autoreply service makes it possible to reply to incoming messages and to forward these messsages to various phone numbers based on the sender address and the message content.

Autoreply easy Install
The Autoreply easy service allows you to reply to an incoming message. It can be used to provide content based on SMS requests. For example you can return a URL with a downloadable App.

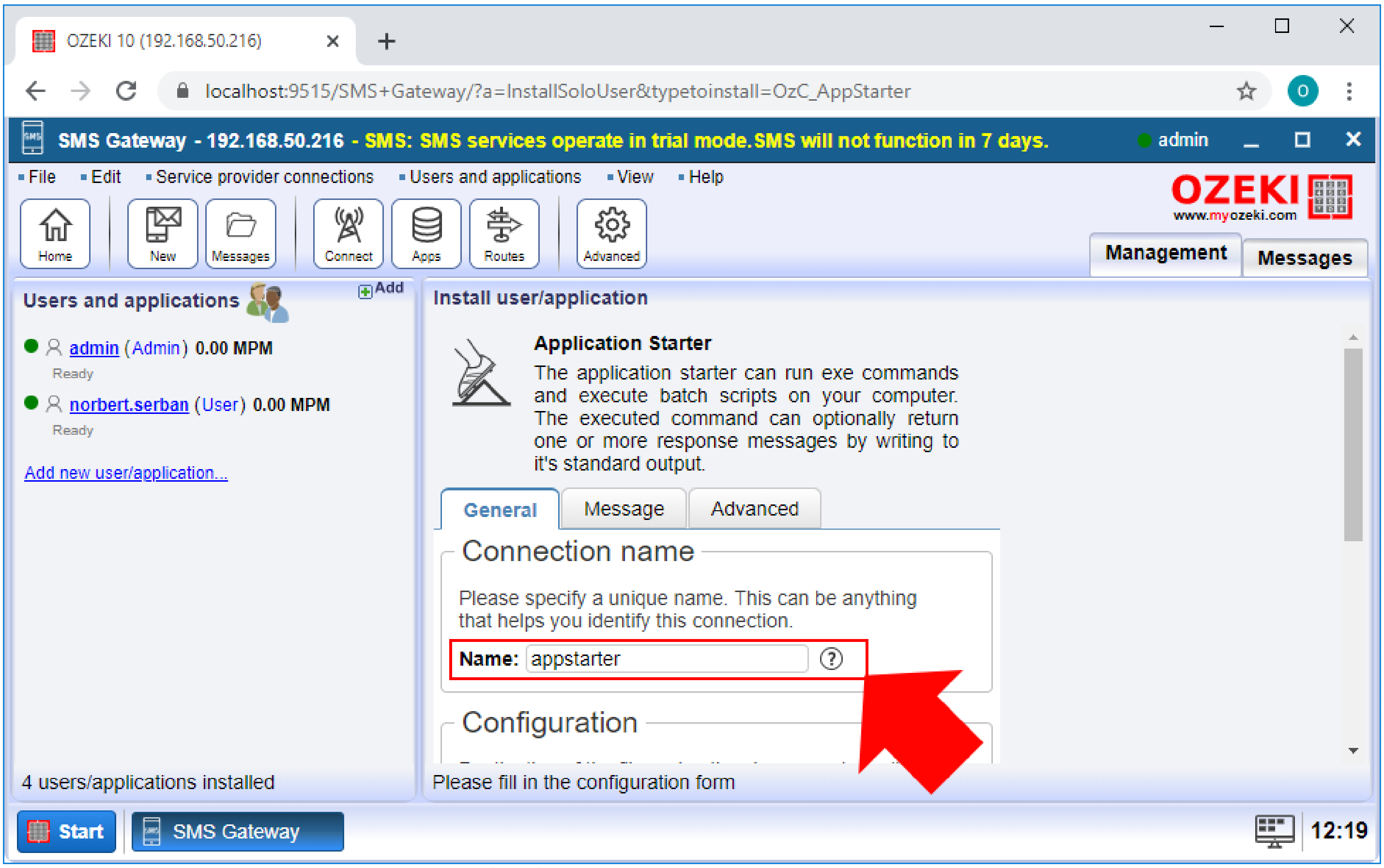
Autoreply database Install
This is an autoreply application that allows you to setup keywords that trigger SQL queries in a database. SQL queries can return response messages based on information in database tables. (Hint: A keyword is the first word in the SMS message.)

ASP C# Install
You may write a script, that allows you to run C# .NET code when an SMS message arrives. This script

Figure 2 - Select the Application starter connection

# Step 2 - Configure the Application Starter connection

Next, you need to configure your application starter connection. First, you need to type a name for the connection (Figure 3). This will identify the connection in your SMS Gateway system. Next, you need to configure the program, that the connection is going to execute. The two main ways will be discussed in the next step.


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=InstallSoloUser&typetoinstall=OzC_AppStarter
SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Users and applications
Add
admin (Admin) 0.00 MPM
Ready
norbert.serban (User) 0.00 MPM
Ready
Add new user/application...
4 users/applications installed
Install user/application
General
Message
Advanced
Start
SMS Gateway
12:19

**Application Starter**

The application starter can run exe commands and execute batch scripts on your computer. The executed command can optionally return one or more response messages by writing to it's standard output.

Connection name

Please specify a unique name. This can be anything that helps you identify this connection.

**Name:** appstarter

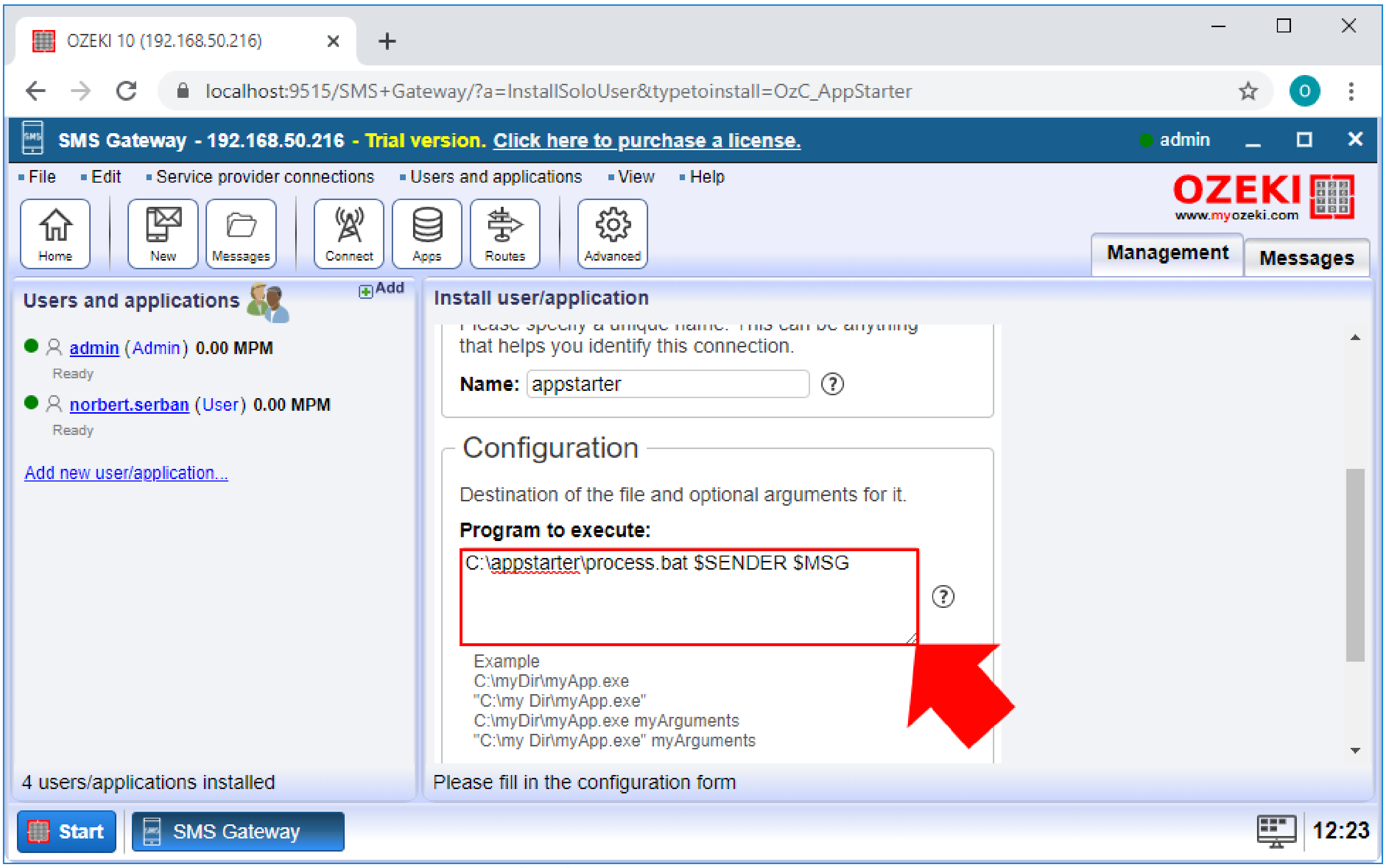
Configuration

Please fill in the configuration form

Figure 3 - Configure the Application starter connection

## Step 3 - Select a program to execute

The Application starter connection can process both bash scripts or the executable files of the external applications as well. So, the first way is to run a bash script. In this example, you need to create a bash script with the '.bat' file extension. As Figure 4 shows below, you need to provide the full path for the bash script. You can also add arguments for the execution. Here, there are two arguments. The '$SENDER' stores the address of the sender who sent the message, the '$MSG' argument stores the message itself.


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=InstallSoloUser&typetoinstall=OzC_AppStarter
SMS Gateway - 192.168.50.216 - Trial version. Click here to purchase a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Users and applications
Add
admin (Admin) 0.00 MPM
Ready
norbert.serban (User) 0.00 MPM
Ready
Add new user/application...
4 users/applications installed
Install user/application
Start
SMS Gateway
12:23

Please specify a unique name. This can be anything that helps you identify this connection.

**Name:** appstarter

Configuration

Destination of the file and optional arguments for it.

**Program to execute:**

C:\appstarter\process.bat $SENDER $MSG

Example
C:\myDir\myApp.exe
"C:\my Dir\myApp.exe"
C:\myDir\myApp.exe myArguments
"C:\my Dir\myApp.exe" myArguments

Please fill in the configuration form

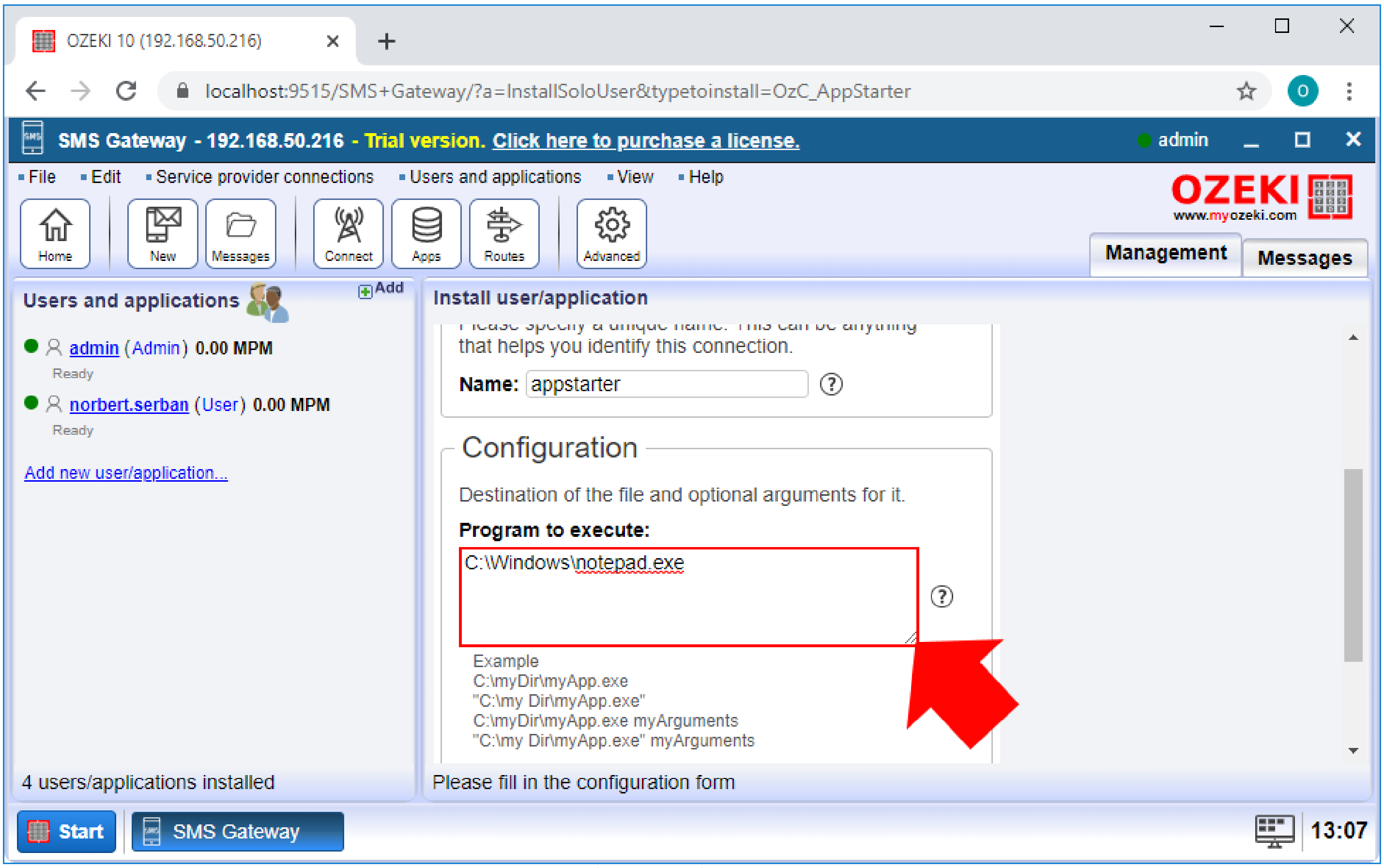
Figure 4 - Configure a bash script with arguments

Now, you need to write the bash script. This example below is going to take the two arguments that you set in the Application starter connection and write them in the inputsms.txt file. If that file does not exist yet, a script creates it, if the file exists, the script appends the file with the new message.

C:\process.bat

```
echo %1 %2 >> C:\inputsms.txt
```

The second way to configure the application starter is to run an external application. For that, you need to specify the full path of the application that you want to run in case of an incoming message. This guide shows you, how you can start the default notepad application with an SMS as you can see it in Figure 5.


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=InstallSoloUser&typetoinstall=OzC_AppStarter
SMS Gateway - 192.168.50.216 - Trial version. Click here to purchase a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Users and applications
Add
admin (Admin) 0.00 MPM
Ready
norbert.serban (User) 0.00 MPM
Ready
Add new user/application...
4 users/applications installed
Start
SMS Gateway
13:07

Install user/application

Please specify a unique name. This can be anything that helps you identify this connection.

**Name:** appstarter

Configuration

Destination of the file and optional arguments for it.

**Program to execute:**

C:\Windows\notepad.exe

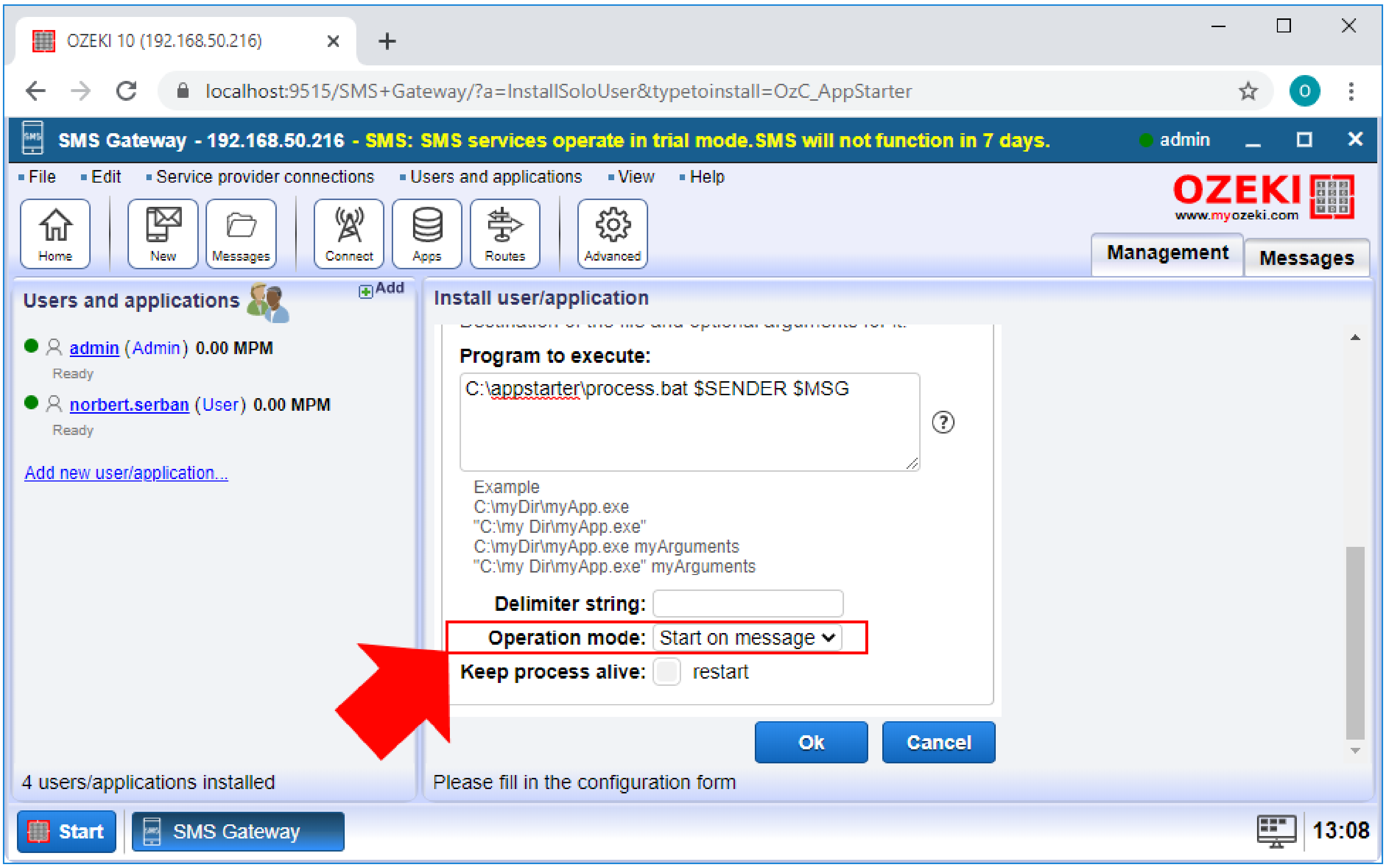
Example
C:\myDir\myApp.exe
"C:\my Dir\myApp.exe"
C:\myDir\myApp.exe myArguments
"C:\my Dir\myApp.exe" myArguments

Please fill in the configuration form

Figure 5 - Type the full path for the notepad application

# Step 3 - Further configuration of Application Starter

The last thing, that you clearly need to configure before creating the connection, is the mode that the application is going to run. This can be set by the 'Operation mode' option. Here, as you can see it in Figure 6, select the 'Start on message' option. This will ensure, that the program that you set before to execute, will only run if an incoming message occurs. If you finished with everything, you can just click OK.


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=InstallSoloUser&typetoinstall=OzC_AppStarter
SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode. SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
OZEKI
www.myozeki.com
Management
Messages
Users and applications
Add
admin (Admin) 0.00 MPM
Ready
norbert.serban (User) 0.00 MPM
Ready
Add new user/application...
4 users/applications installed
Install user/application
Ok
Cancel
Please fill in the configuration form
Start
SMS Gateway
13:08

Program to execute:
```
C:\appstarter\process.bat $SENDER $MSG
```

Example
C:\myDir\myApp.exe
"C:\my Dir\myApp.exe"
C:\myDir\myApp.exe myArguments
"C:\my Dir\myApp.exe" myArguments

Delimiter string:

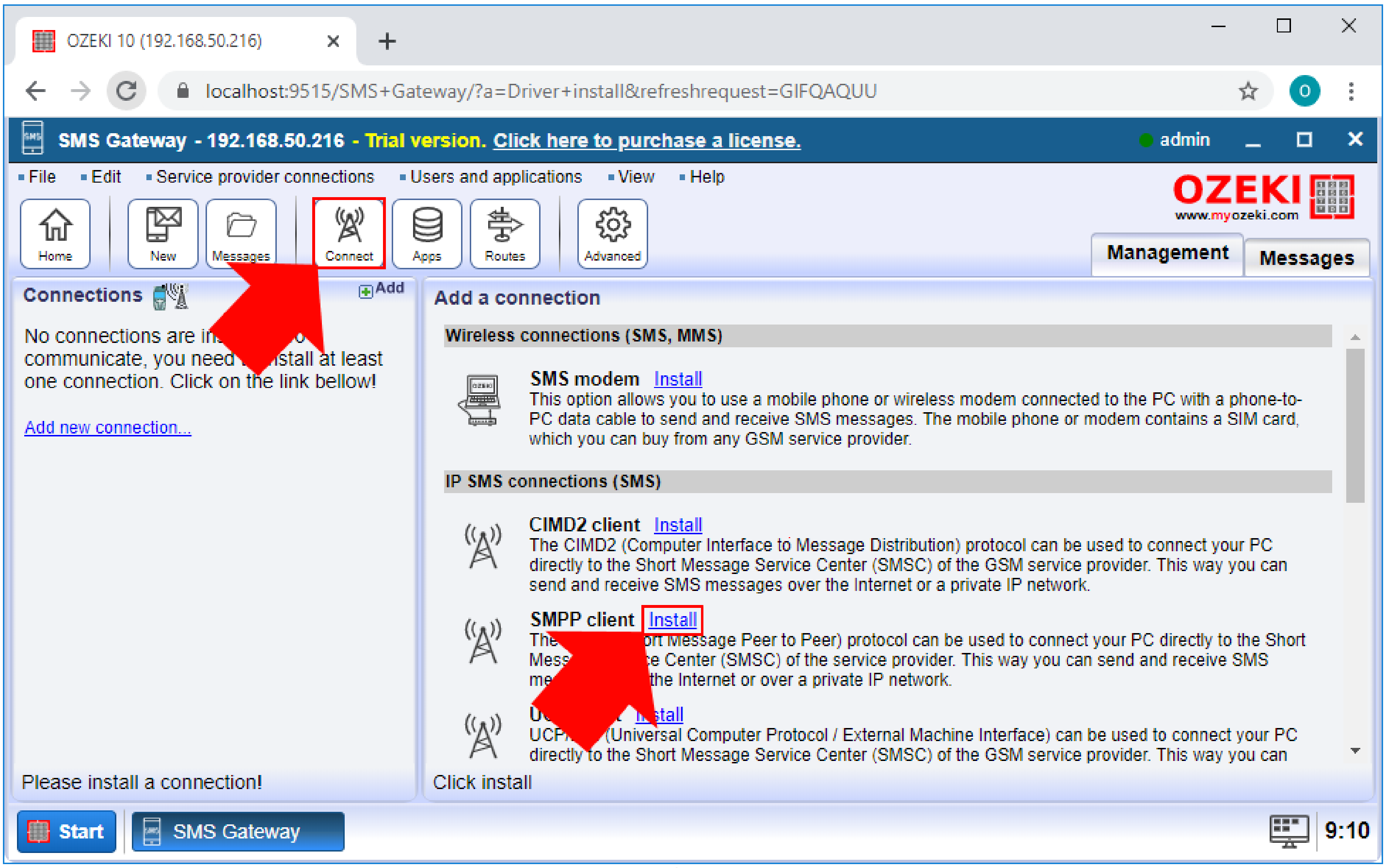
Operation mode: Start on message

Keep process alive: □ restart

Figure 6 - Set the opeartion mode

## Step 4 - Create an SMPP client connection

To be able to communicate with the members of the mobile network, and receive the starting message, you need a connection that creates that link. This connection will be the SMPP client connection. To create this connection, just select the Connect menu from the toolbar, and here as you can see it in Figure 7, click on the Install button of the SMPP client connection.


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=Driver+install&refreshrequest=GIFQAQUU
SMS Gateway - 192.168.50.216 - Trial version. Click here to purchase a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
OZEKI
www.myozeki.com
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Connections
Add
Add a connection
Start
SMS Gateway
9:10

No connections are in... communicate, you need ...stall at least one connection. Click on the link bellow!

Add new connection...

Please install a connection!

Wireless connections (SMS, MMS)

**SMS modem** Install
This option allows you to use a mobile phone or wireless modem connected to the PC with a phone-to-PC data cable to send and receive SMS messages. The mobile phone or modem contains a SIM card, which you can buy from any GSM service provider.

IP SMS connections (SMS)

**CIMD2 client** Install
The CIMD2 (Computer Interface to Message Distribution) protocol can be used to connect your PC directly to the Short Message Service Center (SMSC) of the GSM service provider. This way you can send and receive SMS messages over the Internet or a private IP network.

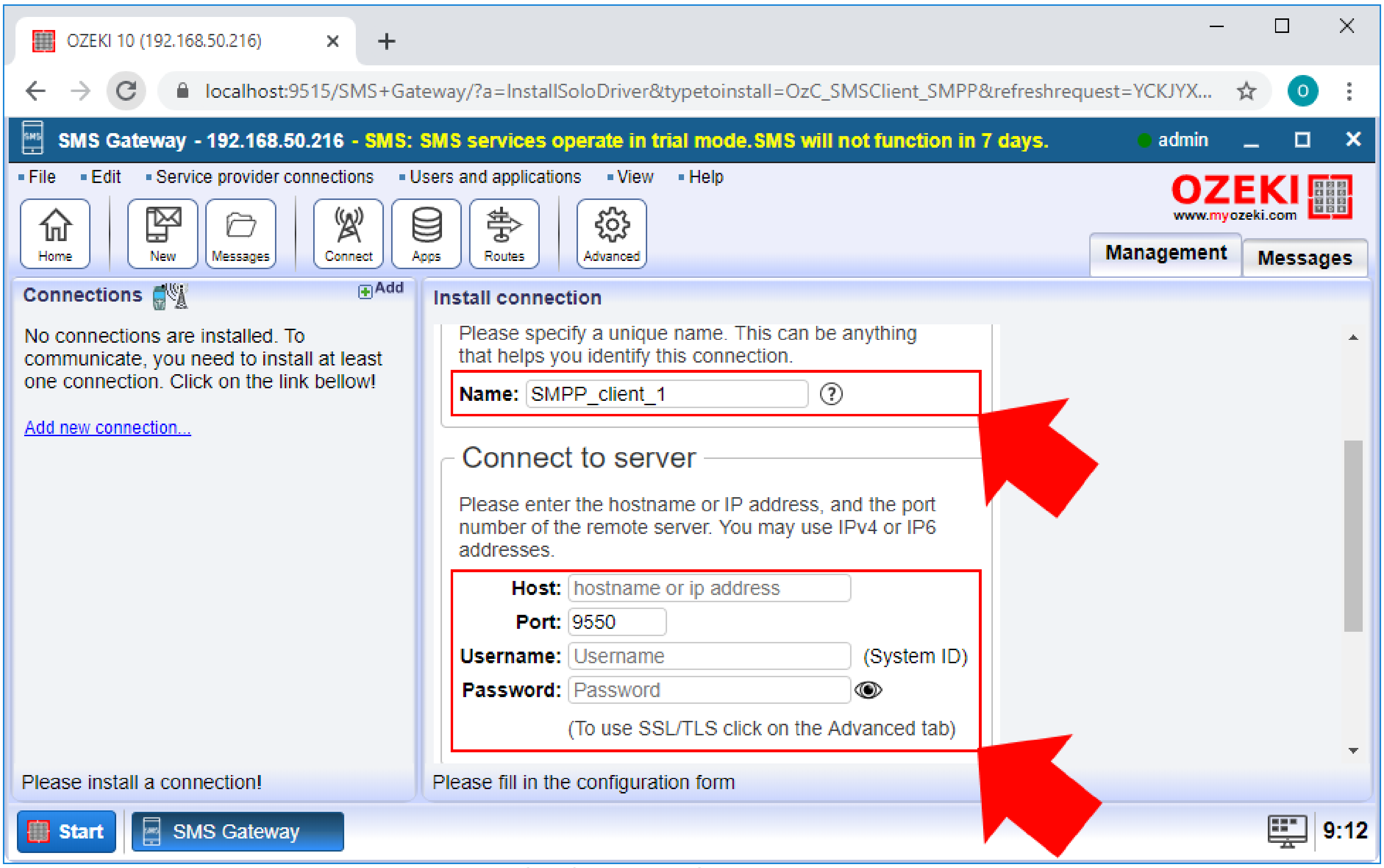
**SMPP client** Install
The ...ort Message Peer to Peer) protocol can be used to connect your PC directly to the Short Mess...e Center (SMSC) of the service provider. This way you can send and receive SMS me... the Internet or over a private IP network.

**U...** ...stall
UCP... (Universal Computer Protocol / External Machine Interface) can be used to connect your PC directly to the Short Message Service Center (SMSC) of the GSM service provider. This way you can

Click install

Figure 7 - Install the SMPP client connection

In the configuration menu of the connection, first, you need to type a name for the connection. Next, you have to provide the details of the SMPP server where you want that connection to connect to (Figure 8). Lastly, just type the phone number with that the connection is going to communicate with the mobile network. If you finished with everything, you can just click on OK.


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=InstallSoloDriver&typetoinstall=OzC_SMSClient_SMPP&refreshrequest=YCKJYX...
SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
OZEKI
www.myozeki.com
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Connections
Add
Install connection
Start
SMS Gateway
9:12

No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

Please install a connection!

Please specify a unique name. This can be anything that helps you identify this connection.

**Name:** SMPP_client_1

Connect to server

Please enter the hostname or IP address, and the port number of the remote server. You may use IPv4 or IP6 addresses.

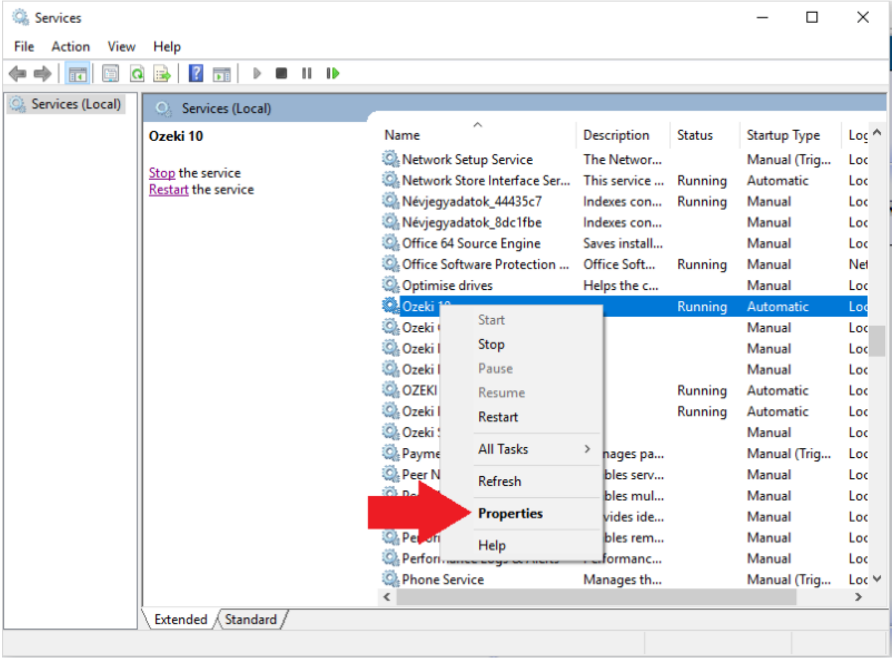
**Host:** hostname or ip address
**Port:** 9550
**Username:** Username (System ID)
**Password:** Password
(To use SSL/TLS click on the Advanced tab)

Please fill in the configuration form

Figure 8 - Configure the SMPP client connection

# Step 5 - Configure your Ozeki service

Ozeki service runs in the background when you use Ozeki SMS Gateway. It is essential to allow Ozeki 10 to run files with Application Starter User. Configure this service by selecting 'Control Panel' from your Windows Start menu and reach 'Administrative Tools/Services'. Find 'Ozeki 10' on the list, right click, and select 'Properties' (Figure 9).



Services (Local)

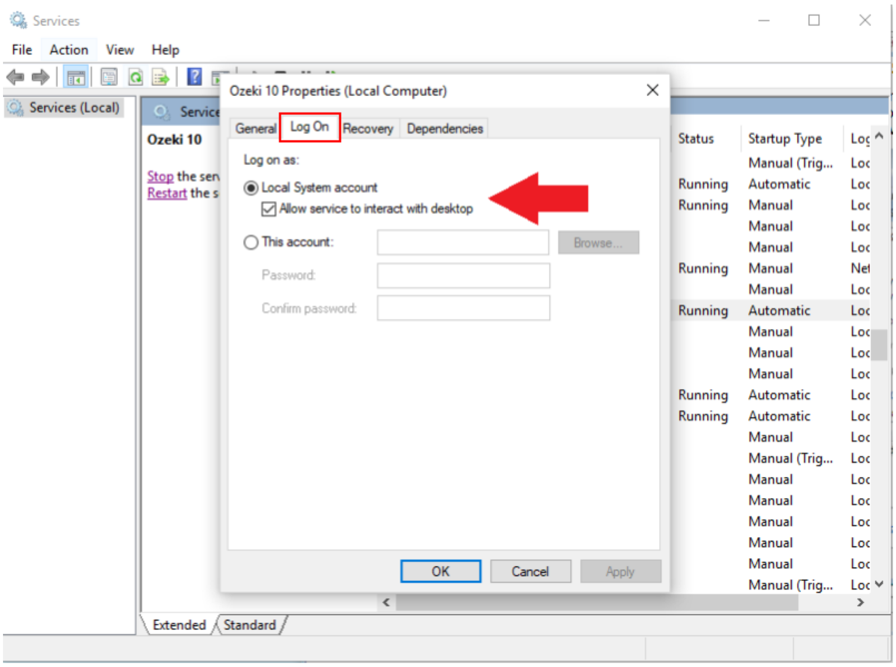
Ozeki 10

Stop the service
Restart the service

| Name | Description | Status | Startup Type | Log |
|---|---|---|---|---|
| Network Setup Service | The Networ... | | Manual (Trig... | Loc |
| Network Store Interface Ser... | This service ... | Running | Automatic | Loc |
| Névjegyadatok_44435c7 | Indexes con... | Running | Manual | Loc |
| Névjegyadatok_8dc1fbe | Indexes con... | | Manual | Loc |
| Office 64 Source Engine | Saves install... | | Manual | Loc |
| Office Software Protection ... | Office Soft... | Running | Manual | Net |
| Optimise drives | Helps the c... | | Manual | Loc |
| Ozeki 10 | | Running | Automatic | Loc |
| Ozeki ( | | | Manual | Loc |
| Ozeki I | | | Manual | Loc |
| Ozeki I | | | Manual | Loc |
| OZEKI | | Running | Automatic | Loc |
| Ozeki I | | Running | Automatic | Loc |
| Ozeki S | | | Manual | Loc |
| Payme | ...nages pa... | | Manual (Trig... | Loc |
| Peer N | ...bles serv... | | Manual | Loc |
| Pe | ...bles mul... | | Manual | Loc |
| | ...vides ide... | | Manual | Loc |
| Pe | ...bles rem... | | Manual | Loc |
| Perfor... | ...formanc... | | Manual | Loc |
| Phone Service | Manages th... | | Manual (Trig... | Loc |

Figure 9 - Set the service properties of Ozeki

On the new window, called 'Ozeki 10 Properties', please select the 'Log On' tab page and tick the 'Allow service to interact with desktop' checkbox (Figure 10).



Ozeki 10 Properties (Local Computer)

Log on as:

● Local System account

☑ Allow service to interact with desktop

○ This account:

Password:

Confirm password:

Figure 8 - 'Allow service to interact with desktop'

From now on Ozeki SMS Gateway's Application Starter User will be able to start any process or application if you provide the proper path to the file. The file on the chosen path will start in case of an SMS message is received.

# SMS to E-mail and E-mail to SMS

***If you wish to send (and receive) SMS text messages to mobile phones using E-mail Ozeki 10 SMS gateway is a great choice. You can configure it as an E-mail client to interact with your company's E-mail server as any other E-mail client would or it can act as a standalone SMTP service. Check out the following options and select the solution that best fits your needs***

## Option 1 - Setup Ozeki as an E-mail client

This option is the most popular setup, because in this configuration Ozeki will connect to your existing E-mail infrastructure, and you don't have to make any changes to your existing systems. In this setup Ozeki will download the E-mail messages to be sent periodically using the POP3 protocol and will forward them an SMS to the mobile network. If an incoming SMS message comes in, it will connect to your E-mail sever as an SMTP client and pass the messages to it through the standard SMTP protocol. This solution will work will with your company e-mail system (and it can even be used through G-mail).
Read more about how to setup E-mail to SMS forwarding using POP3 and SMTP client.

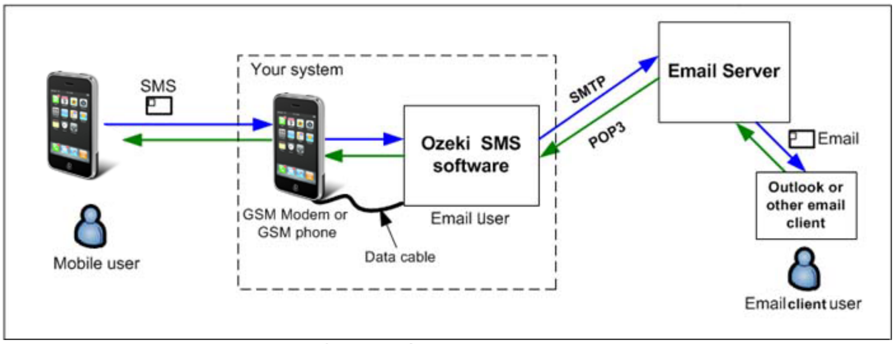


Figure 1 - Setup Ozeki as an E-mail client

## Option 2 - Setup Ozeki as an E-mail server

This option is recommended if you can control your Domain Name Service (DNS). If you can create an MX record in your DNS server that points a domain name to your SMS gateway, you can forward E-mail messages to the SMTP server built into Ozeki SMS Gateway. For example you can setup smsgw.mycompany.com to point to your SMS gateawy, and you can send e-mails to this domain in the format +441234567@smsgw.mycompany.com. In this case the built in SMTP server will convert the e-mail messages to SMS, and it will send these messages to the phone number provided in the e-mail address prefix.
Read more about how to use the built in SMTP server for E-mail to SMS forwarding.

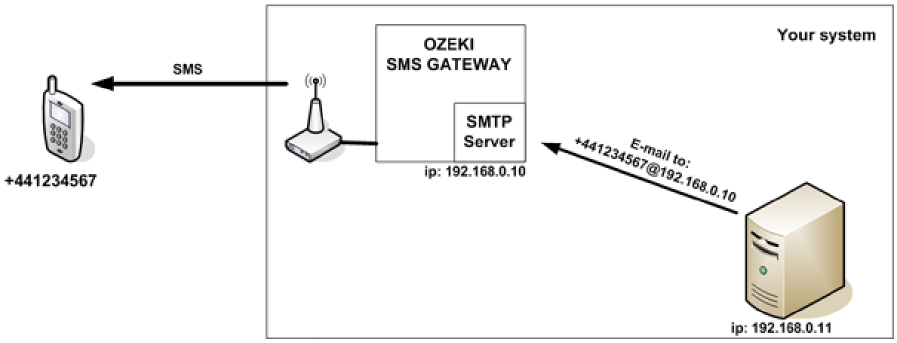


Figure 2 - Setup Ozeki as an E-mail server

## Option 3 - Direct e-mail clients

This option provides you a setup that makes it possible to use Ozeki SMS Gateway with other e-mail clients liek Mozilla Thunderbird or Windows mail. In this setup, the SMTP and POP3 servers integrated in Ozeki SMS Gateway need to be connected to the e-mail client. After you connected them to the e-mail client, it allows you the forward SMS to e-mail using the SMTP server and forward the e-mails as SMS messages with the bulit-in

POP3 server in SMS Gateway.
See how you can configure direct e-mail client to forward e-mail and SMS messages

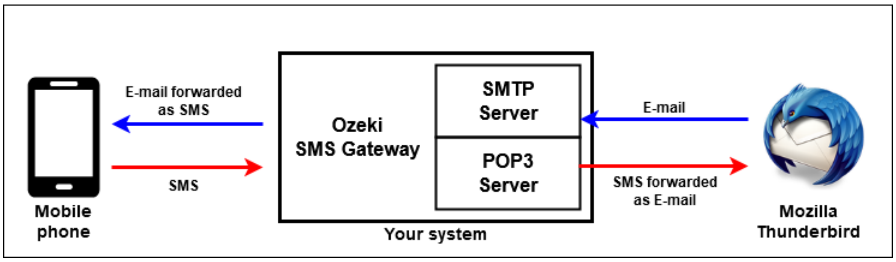


Figure 3 - Direct e-mail clients

# E-mail to SMS and SMS to E-mail

***This guide explains how you can setup Ozeki 10 SMS gateway to act as an e-mail client. This setup offers both e-mail to sms and sms to e-mail forwarding. For e-mail to SMS forwarding Ozeki will download e-mail messages from an e-mail server (using the POP3 protocol) periodically, and it will send these e-mails as SMS messages to mobile phones. For SMS to e-mail forwarding Ozeki will accept incoming SMS messages from the mobile network and will forward these to an e-mail address (using the SMTP protocol).***

## Introduction

This solution consists of two sections. The first section is E-mail to SMS forwarding. If you look at Figure 1, you will see the route of the message of this section by following the green arrow. The second section is SMS to E-mail forwarding. The route of the messages for SMS to E-mail forwarding is shown in blue.

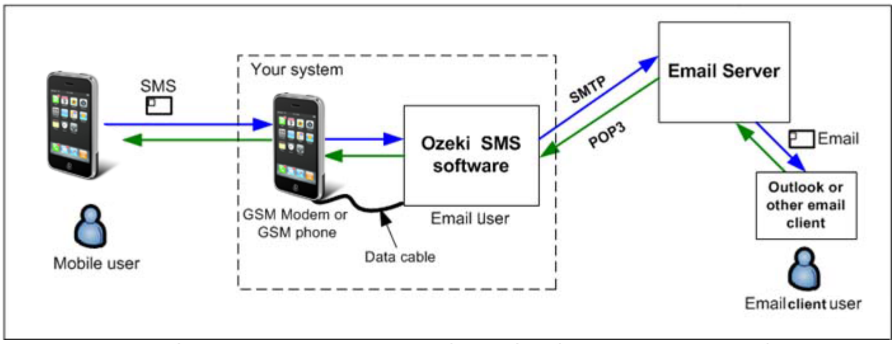


Figure 1 - System architecture of the SMS through Email function

## How to setup E-mail to SMS forwarding

Let's start by setting up e-mail to SMS forwarding. The following configuration steps will show how the system should be configured to collect the incoming e-mails from an e-mail account using a POP3 client connection. You will also see how these emails are converted into SMS and how they are sent as SMS (Figure 2).

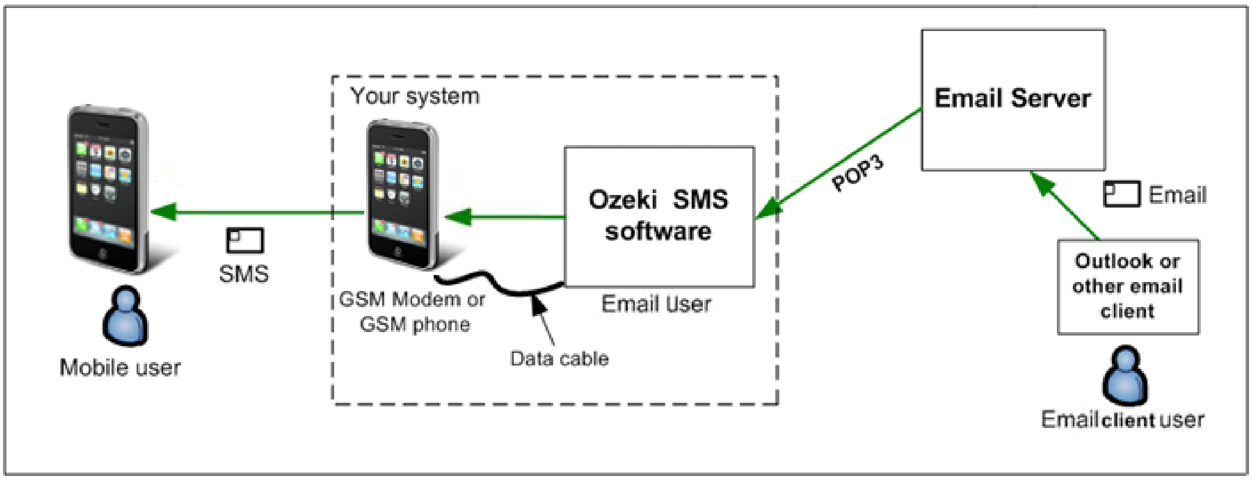
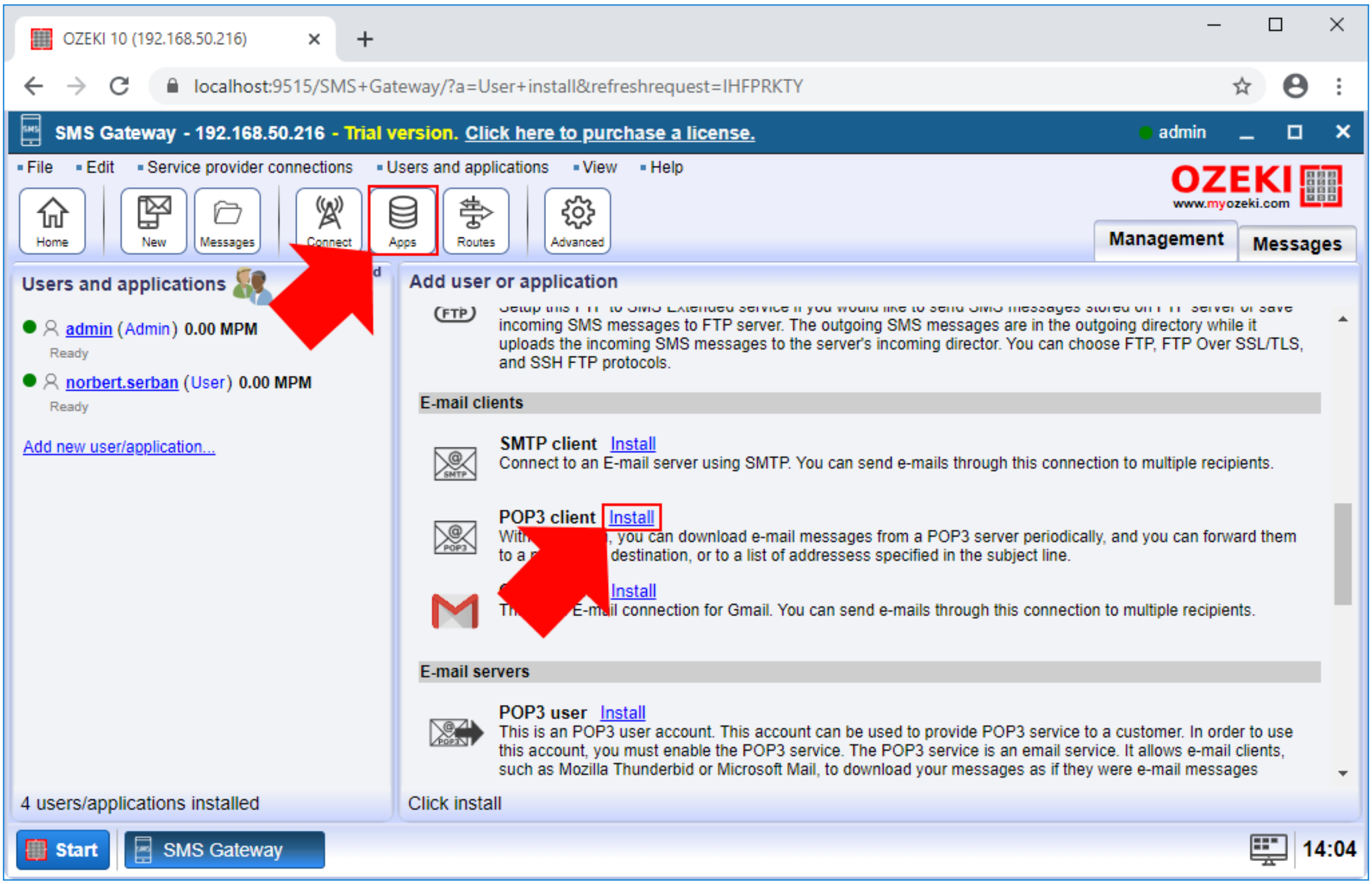


Figure 2 - Incoming email sent as SMS

## Step 1 - Create a POP3 client connection

The first step of this guide is to create the POP3 client connection. For that, you need to open the SMS Gateway application, and here, select the Apps menu from the toolbar. In this menu, like in Figure 3, select the POP3 client from the E-mail clients section by clicking on the Install button.


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=User+install&refreshrequest=IHFPRKTY
SMS Gateway - 192.168.50.216 - Trial version. Click here to purchase a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
OZEKI
www.myozeki.com
Management
Messages
Start
SMS Gateway
14:04

Users and applications

admin (Admin) 0.00 MPM
Ready

norbert.serban (User) 0.00 MPM
Ready

Add new user/application...

4 users/applications installed

Add user or application

Setup this FTP to SMS Extended service if you would like to send SMS messages stored on FTP server or save incoming SMS messages to FTP server. The outgoing SMS messages are in the outgoing directory while it uploads the incoming SMS messages to the server's incoming director. You can choose FTP, FTP Over SSL/TLS, and SSH FTP protocols.

E-mail clients

SMTP client Install
Connect to an E-mail server using SMTP. You can send e-mails through this connection to multiple recipients.

POP3 client Install
With , you can download e-mail messages from a POP3 server periodically, and you can forward them to a destination, or to a list of addresses specified in the subject line.

Install
Th E-mail connection for Gmail. You can send e-mails through this connection to multiple recipients.

E-mail servers

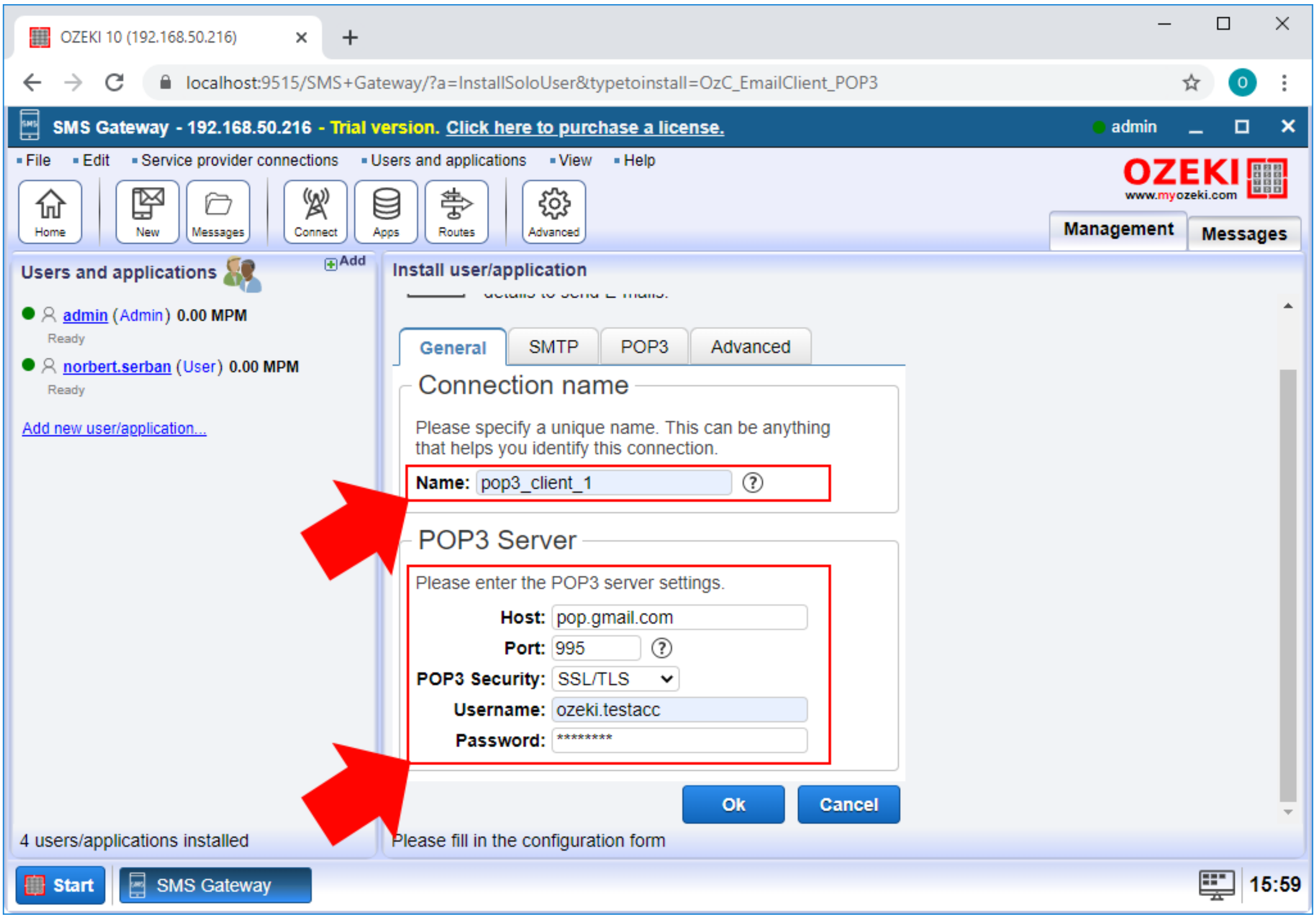
POP3 user Install
This is an POP3 user account. This account can be used to provide POP3 service to a customer. In order to use this account, you must enable the POP3 service. The POP3 service is an email service. It allows e-mail clients, such as Mozilla Thunderbid or Microsoft Mail, to download your messages as if they were e-mail messages

Click install

Figure 3 - Select the POP3 client connection

# Step 2 - Configure the POP3 server details

The first step of the configuration of the POP3 client connection is to provide the details of the POP3 server. This guide uses Gmail to demonstrate the process, so here, you need to enter the details of Gmail's POP3 server. As you can see it in Figure 4, type 'pop.gamil.com' as Host and type 995 as Port number. Then, select SSL/TLS as POP3 Security and lastly, provide the username and password of your e-mail account, which is in that case, a Gmail account.


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=InstallSoloUser&typetoinstall=OzC_EmailClient_POP3
SMS Gateway - 192.168.50.216 - Trial version. Click here to purchase a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
OZEKI
www.myozeki.com
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Users and applications
Add
admin (Admin) 0.00 MPM
Ready
norbert.serban (User) 0.00 MPM
Ready
Add new user/application...
4 users/applications installed
Install user/application
General
SMTP
POP3
Advanced
Ok
Cancel
Start
SMS Gateway
15:59

Connection name

Please specify a unique name. This can be anything that helps you identify this connection.

Name: pop3_client_1

POP3 Server

Please enter the POP3 server settings.

Host: pop.gmail.com

Port: 995

POP3 Security: SSL/TLS

Username: ozeki.testacc

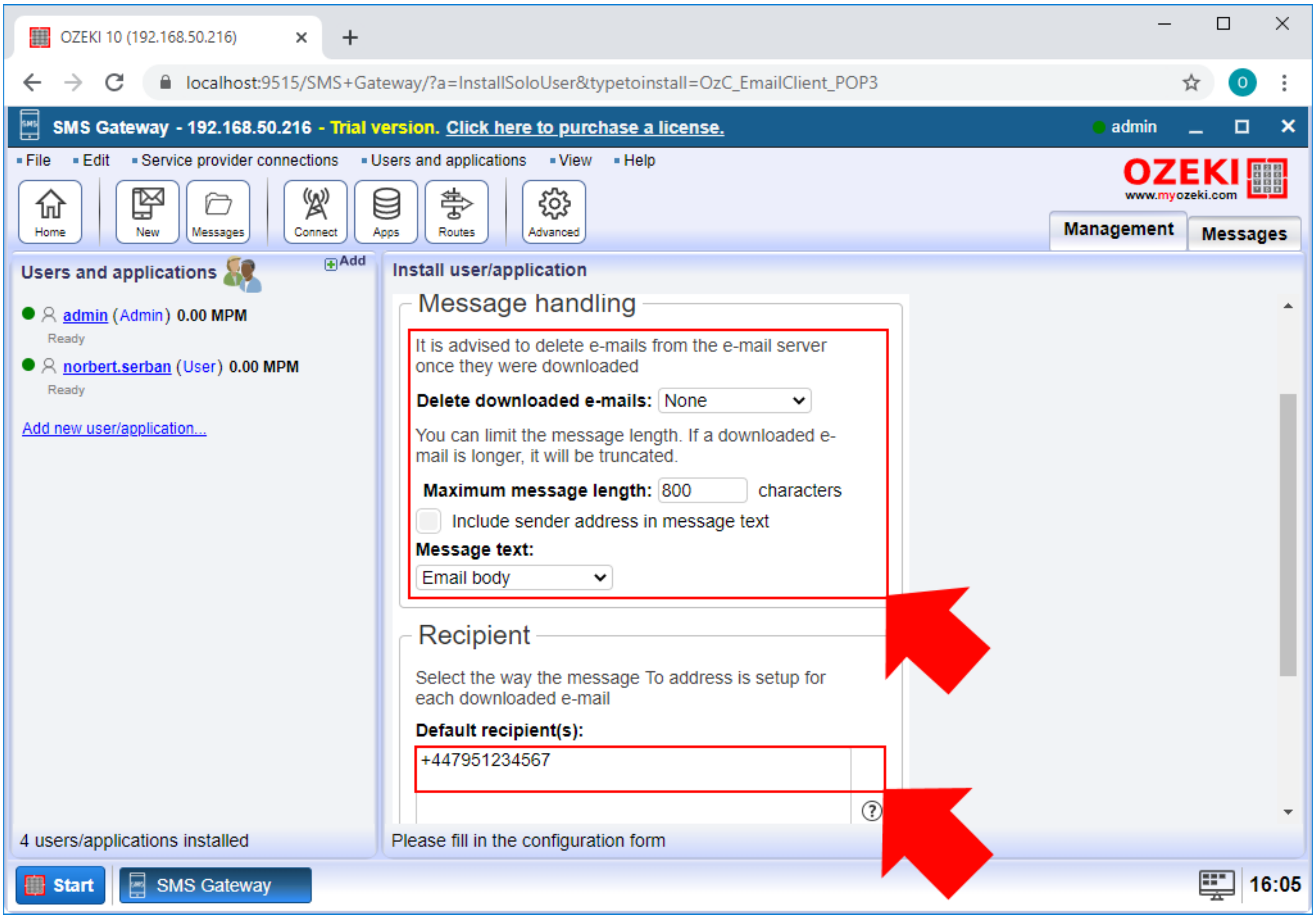
Password: ********

Please fill in the configuration form

Figure 4 - POP3 server details for the connection

## Step 3 - Configure the advanced details of the connection

In the next step, you need to configure the way, that the connection is going to forward the messages. For that, first, click on the POP3 tab. Here, in the 'Message handling' section, you can set the maximum length of the message and select the way, how the e-mail will be forwarded. Next, as Figure 5 demonstrates, you can set the recipients for the e-mail. The addresses, that you type here will get the forwarded e-mails. If you finished with the configuration, you can just click on OK.



OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=InstallSoloUser&typetoinstall=OzC_EmailClient_POP3
SMS Gateway - 192.168.50.216 - Trial version. Click here to purchase a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
OZEKI
www.myozeki.com
Management
Messages
Users and applications
Add
admin (Admin) 0.00 MPM
Ready
norbert.serban (User) 0.00 MPM
Ready
Add new user/application...
4 users/applications installed
Install user/application
Please fill in the configuration form
Start
SMS Gateway
16:05

Message handling

It is advised to delete e-mails from the e-mail server once they were downloaded

**Delete downloaded e-mails:** None

You can limit the message length. If a downloaded e-mail is longer, it will be truncated.

**Maximum message length:** 800 characters

□ Include sender address in message text

**Message text:** Email body

Recipient

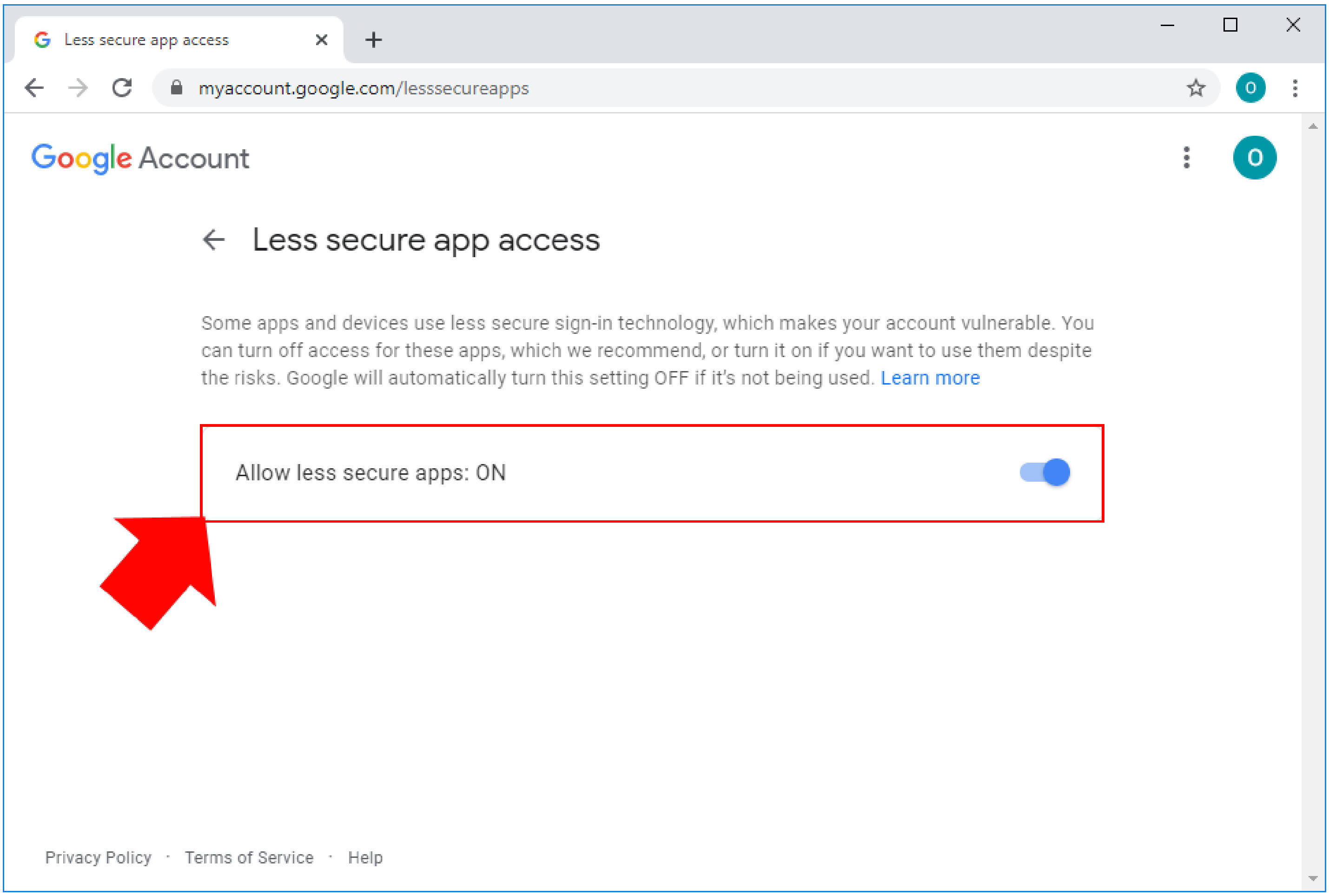
Select the way the message To address is setup for each downloaded e-mail

**Default recipient(s):** +447951234567

Figure 5 - Advanced configuration of the POP3 client connection

## Step 4 - Configure the security settings of Gmail account

After you created the POP3 client connection, you need to do some modifications to the security settings of your Gmail account. For that, just type 'https://myaccount.google.com/lesssecureapps' in your browser and hit Enter. Here (Figure 6), you just need to turn the 'Less secure apps access' on. This will ensure that SMS Gateway is going to have access to download the e-mails.



Less secure app access
myaccount.google.com/lesssecureapps
Google Account

Less secure app access

Some apps and devices use less secure sign-in technology, which makes your account vulnerable. You can turn off access for these apps, which we recommend, or turn it on if you want to use them despite the risks. Google will automatically turn this setting OFF if it's not being used. Learn more

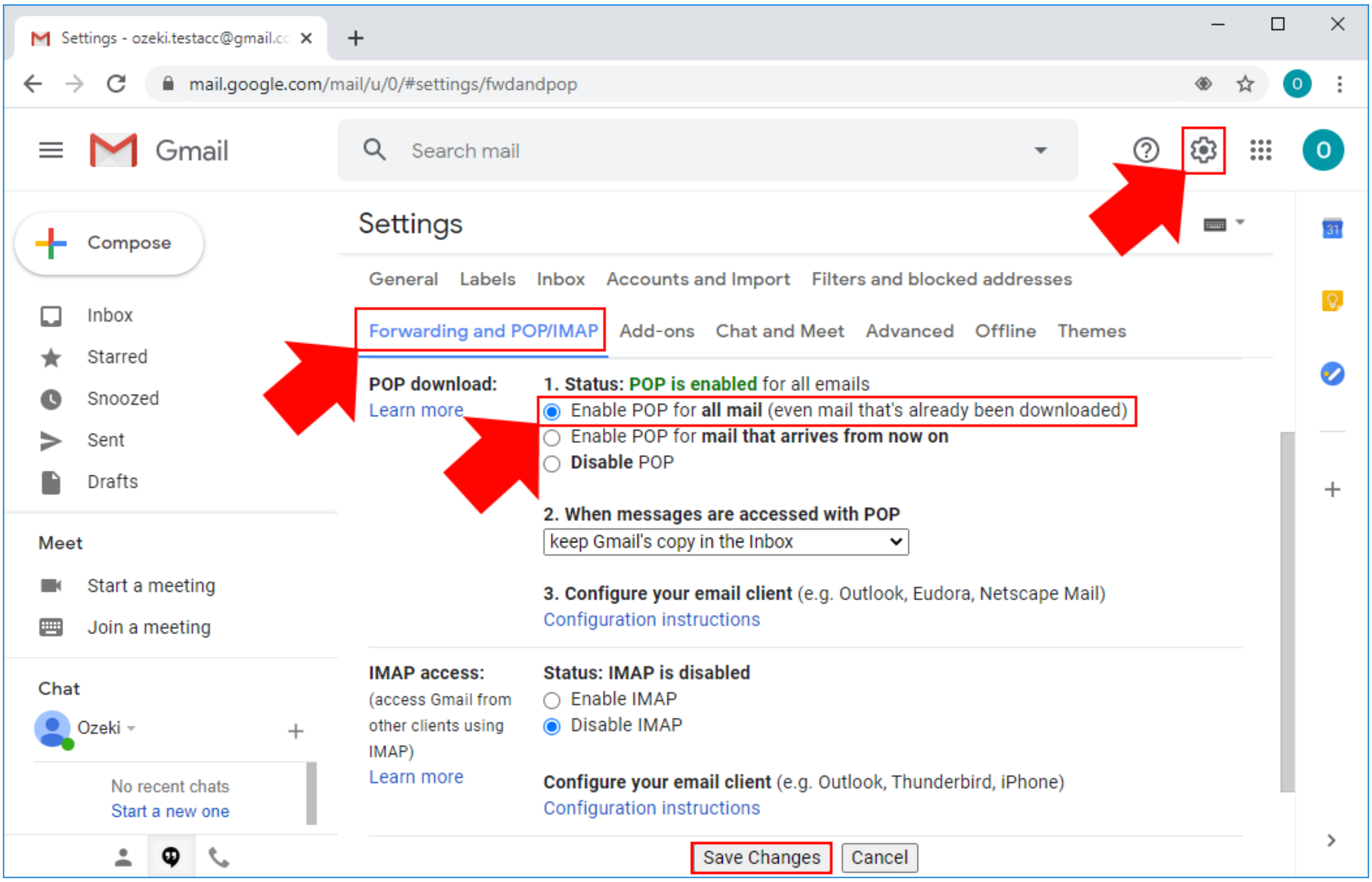
Allow less secure apps: ON

Privacy Policy · Terms of Service · Help

Figure 6 - Turning on access for less secure apps

## Step 5 - Modify the Gmail POP3 settings

In this step, you need to modify the forwarding setting of your Gmail account. For that, open Gmail and here, click on Settings. In the Settings menu, you need to select the Forwarding and POP/IMAP tab. Here, like Figure 7 shows that, select 'Enable POP for all mail' and finally, click on Save Changes.



Settings

**POP download:**
Learn more

**1. Status: POP is enabled** for all emails

● Enable POP for **all mail** (even mail that's already been downloaded)

○ Enable POP for **mail that arrives from now on**

○ **Disable** POP

**2. When messages are accessed with POP**

keep Gmail's copy in the Inbox

**3. Configure your email client** (e.g. Outlook, Eudora, Netscape Mail)
Configuration instructions

**IMAP access:**
(access Gmail from other clients using IMAP)
Learn more

**Status: IMAP is disabled**

○ Enable IMAP

● Disable IMAP

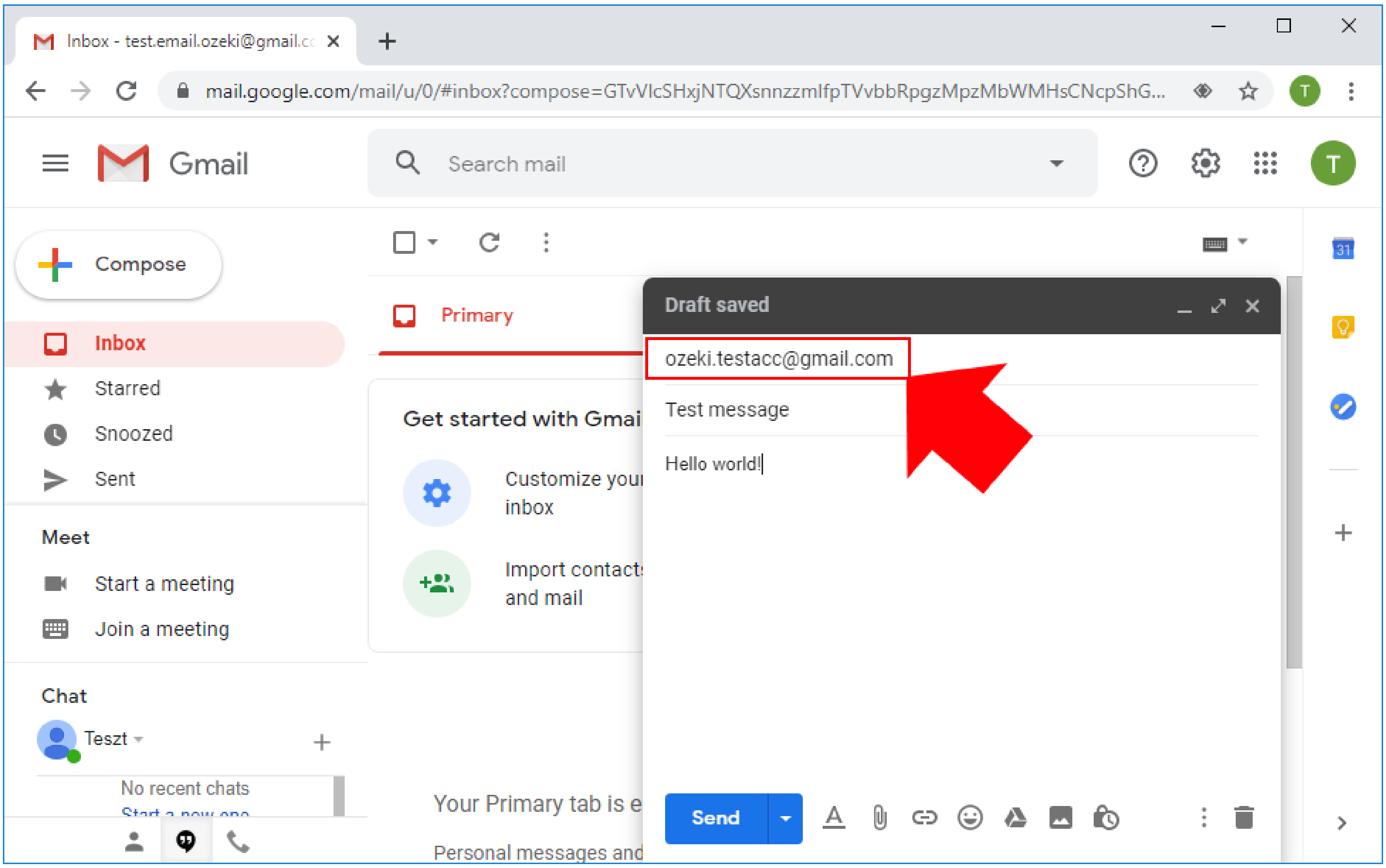
**Configure your email client** (e.g. Outlook, Thunderbird, iPhone)
Configuration instructions

Save Changes Cancel

Figure 7 - Modify forwarding settings of your Gmail account

# Step 6 - Send a test e-mail

At this point, you can write an e-mail message to the Gmail account to test the solution. As you can see it in Figure 9, all you need to do is to write a simple 'Hello world!' to the e-mail address of the Gmail account that you set in the POP3 client connection.



ozeki.testacc@gmail.com

Test message

Hello world!

Figure 8 - Write an e-mail message to the Gmail account

With everything is set up correctly, all you need to do now is to wait for the incoming e-mail to your Gmail account. In case of an incoming e-mail, the POP3 client connection is going to download the e-mail and forward it instantly to the specified recipient address. Figure 9 demonstrates the procedure of forwarding an incoming e-mail by using the POP3 client connection.

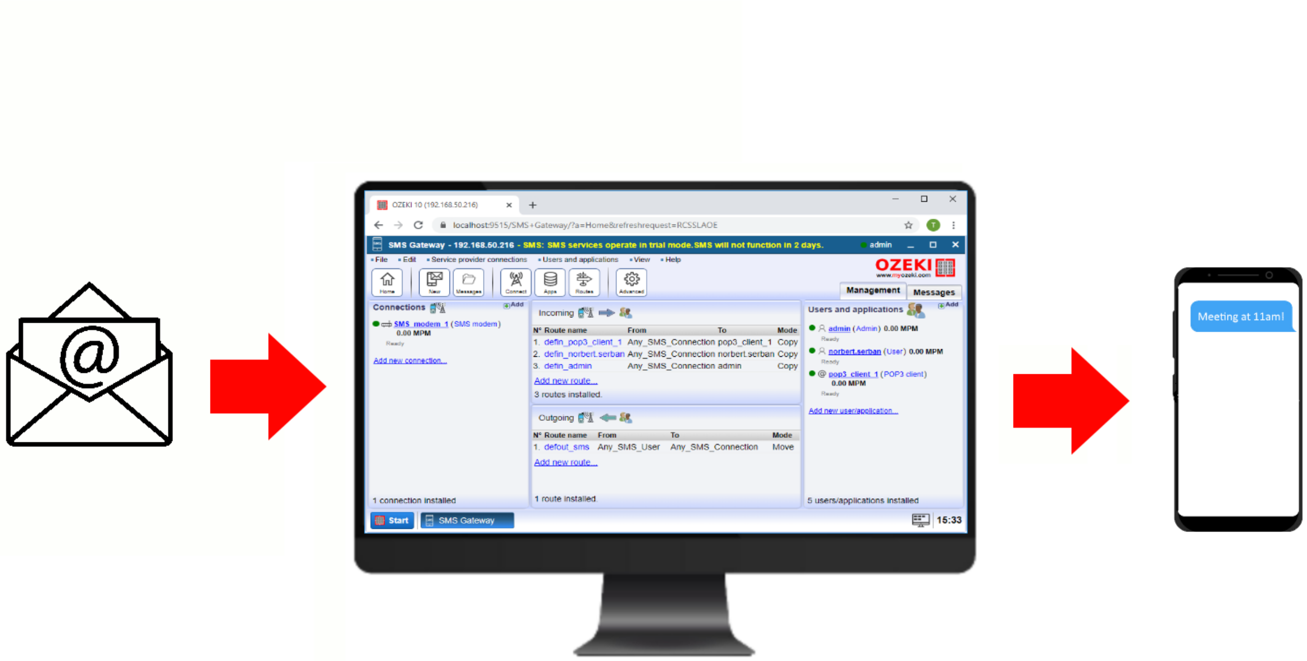
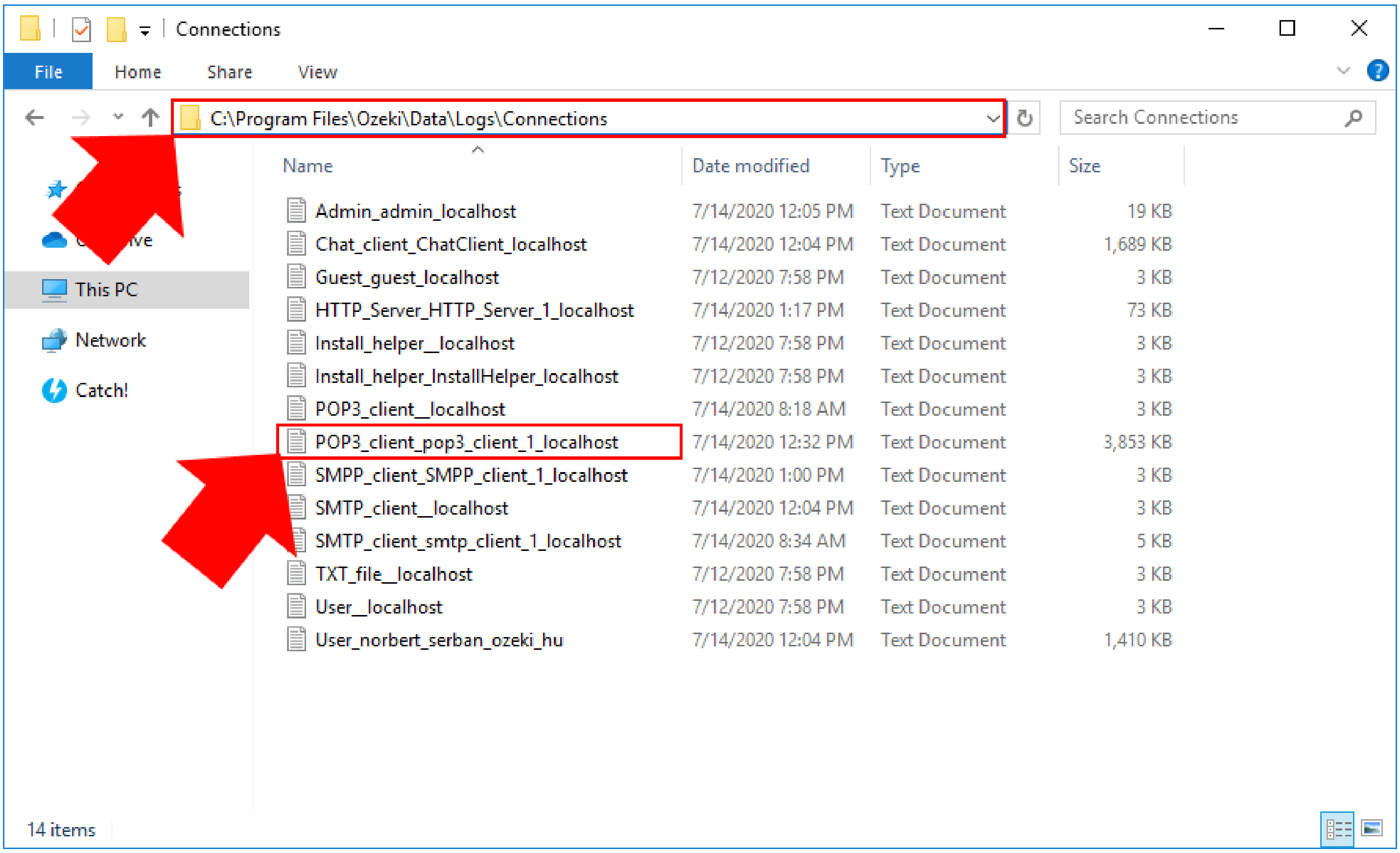


Figure 9 - See how SMS Gateway forwards the message as an SMS

## Step 7 - Check the POP3 logs

You can easily check the logs of the POP3 client on your computer. These logs store every event that occurred with your POP3 client connection. Figure 10 shows the path, you need to follow in File Explorer to open the log file of the POP3 client connection. If you open that log file, you will be able to see all events of the POP3 client connection.

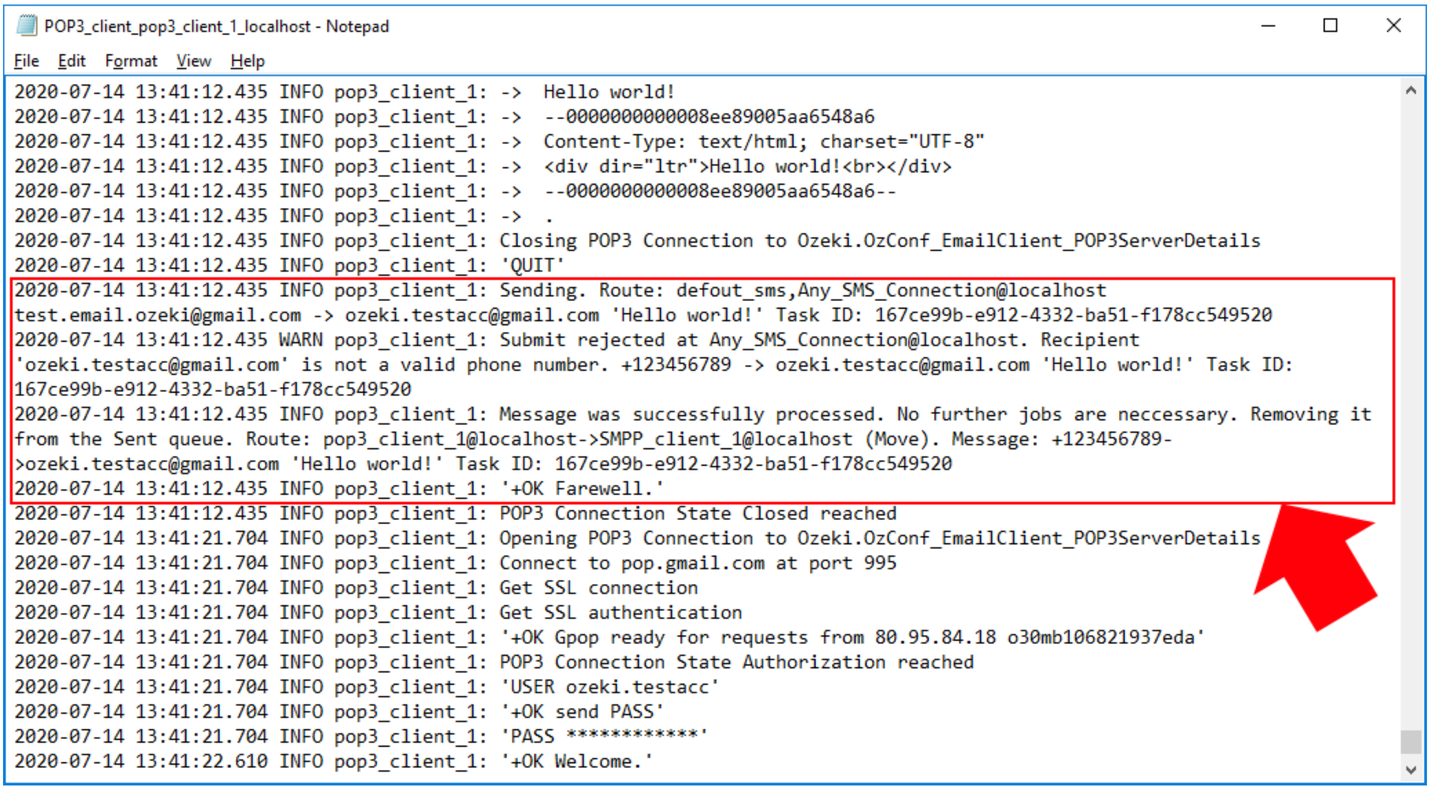


C:\Program Files\Ozeki\Data\Logs\Connections

| Name | Date modified | Type | Size |
|---|---|---|---|
| Admin_admin_localhost | 7/14/2020 12:05 PM | Text Document | 19 KB |
| Chat_client_ChatClient_localhost | 7/14/2020 12:04 PM | Text Document | 1,689 KB |
| Guest_guest_localhost | 7/12/2020 7:58 PM | Text Document | 3 KB |
| HTTP_Server_HTTP_Server_1_localhost | 7/14/2020 1:17 PM | Text Document | 73 KB |
| Install_helper__localhost | 7/12/2020 7:58 PM | Text Document | 3 KB |
| Install_helper_InstallHelper_localhost | 7/12/2020 7:58 PM | Text Document | 3 KB |
| POP3_client__localhost | 7/14/2020 8:18 AM | Text Document | 3 KB |
| POP3_client_pop3_client_1_localhost | 7/14/2020 12:32 PM | Text Document | 3,853 KB |
| SMPP_client_SMPP_client_1_localhost | 7/14/2020 1:00 PM | Text Document | 3 KB |
| SMTP_client__localhost | 7/14/2020 12:04 PM | Text Document | 3 KB |
| SMTP_client_smtp_client_1_localhost | 7/14/2020 8:34 AM | Text Document | 5 KB |
| TXT_file__localhost | 7/12/2020 7:58 PM | Text Document | 3 KB |
| User__localhost | 7/12/2020 7:58 PM | Text Document | 3 KB |
| User_norbert_serban_ozeki_hu | 7/14/2020 12:04 PM | Text Document | 1,410 KB |

Figure 10 - File path for the log file of the POP3 client connection

The log file contains details of every event that occurred with your POP3 client connection. Here you can see a timestamp that shows the date of each event and the name of the connection. Lastly, each event has got a message that describes the event. Figure 11 shows the event messages of downloading an e-mail from the Gmail account and forward it to the given recipient address.



```
2020-07-14 13:41:12.435 INFO pop3_client_1: ->  Hello world!
2020-07-14 13:41:12.435 INFO pop3_client_1: ->  --0000000000008ee89005aa6548a6
2020-07-14 13:41:12.435 INFO pop3_client_1: ->  Content-Type: text/html; charset="UTF-8"
2020-07-14 13:41:12.435 INFO pop3_client_1: ->  <div dir="ltr">Hello world!<br></div>
2020-07-14 13:41:12.435 INFO pop3_client_1: ->  --0000000000008ee89005aa6548a6--
2020-07-14 13:41:12.435 INFO pop3_client_1: ->  .
2020-07-14 13:41:12.435 INFO pop3_client_1: Closing POP3 Connection to Ozeki.OzConf_EmailClient_POP3ServerDetails
2020-07-14 13:41:12.435 INFO pop3_client_1: 'QUIT'
2020-07-14 13:41:12.435 INFO pop3_client_1: Sending. Route: defout_sms,Any_SMS_Connection@localhost
test.email.ozeki@gmail.com -> ozeki.testacc@gmail.com 'Hello world!' Task ID: 167ce99b-e912-4332-ba51-f178cc549520
2020-07-14 13:41:12.435 WARN pop3_client_1: Submit rejected at Any_SMS_Connection@localhost. Recipient
'ozeki.testacc@gmail.com' is not a valid phone number. +123456789 -> ozeki.testacc@gmail.com 'Hello world!' Task ID:
167ce99b-e912-4332-ba51-f178cc549520
2020-07-14 13:41:12.435 INFO pop3_client_1: Message was successfully processed. No further jobs are neccessary. Removing it
from the Sent queue. Route: pop3_client_1@localhost->SMPP_client_1@localhost (Move). Message: +123456789-
>ozeki.testacc@gmail.com 'Hello world!' Task ID: 167ce99b-e912-4332-ba51-f178cc549520
2020-07-14 13:41:12.435 INFO pop3_client_1: '+OK Farewell.'
2020-07-14 13:41:12.435 INFO pop3_client_1: POP3 Connection State Closed reached
2020-07-14 13:41:21.704 INFO pop3_client_1: Opening POP3 Connection to Ozeki.OzConf_EmailClient_POP3ServerDetails
2020-07-14 13:41:21.704 INFO pop3_client_1: Connect to pop.gmail.com at port 995
2020-07-14 13:41:21.704 INFO pop3_client_1: Get SSL connection
2020-07-14 13:41:21.704 INFO pop3_client_1: Get SSL authentication
2020-07-14 13:41:21.704 INFO pop3_client_1: '+OK Gpop ready for requests from 80.95.84.18 o30mb106821937eda'
2020-07-14 13:41:21.704 INFO pop3_client_1: POP3 Connection State Authorization reached
2020-07-14 13:41:21.704 INFO pop3_client_1: 'USER ozeki.testacc'
2020-07-14 13:41:21.704 INFO pop3_client_1: '+OK send PASS'
2020-07-14 13:41:21.704 INFO pop3_client_1: 'PASS ************'
2020-07-14 13:41:22.610 INFO pop3_client_1: '+OK Welcome.'
```

Figure 11 - Log file of the POP3 client connection

# How to configure SMS to E-mail forwarding

After E-mail to SMS is working, you should configure the other direction: SMS to E-mail forwarding. In this section, you will see the steps that will allow you to configure the SMTP server details, and you will see how incoming SMS messages are forwarded to an e-mail address.

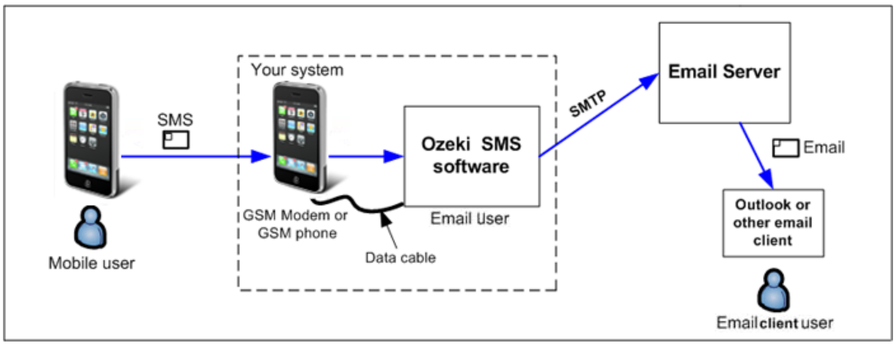
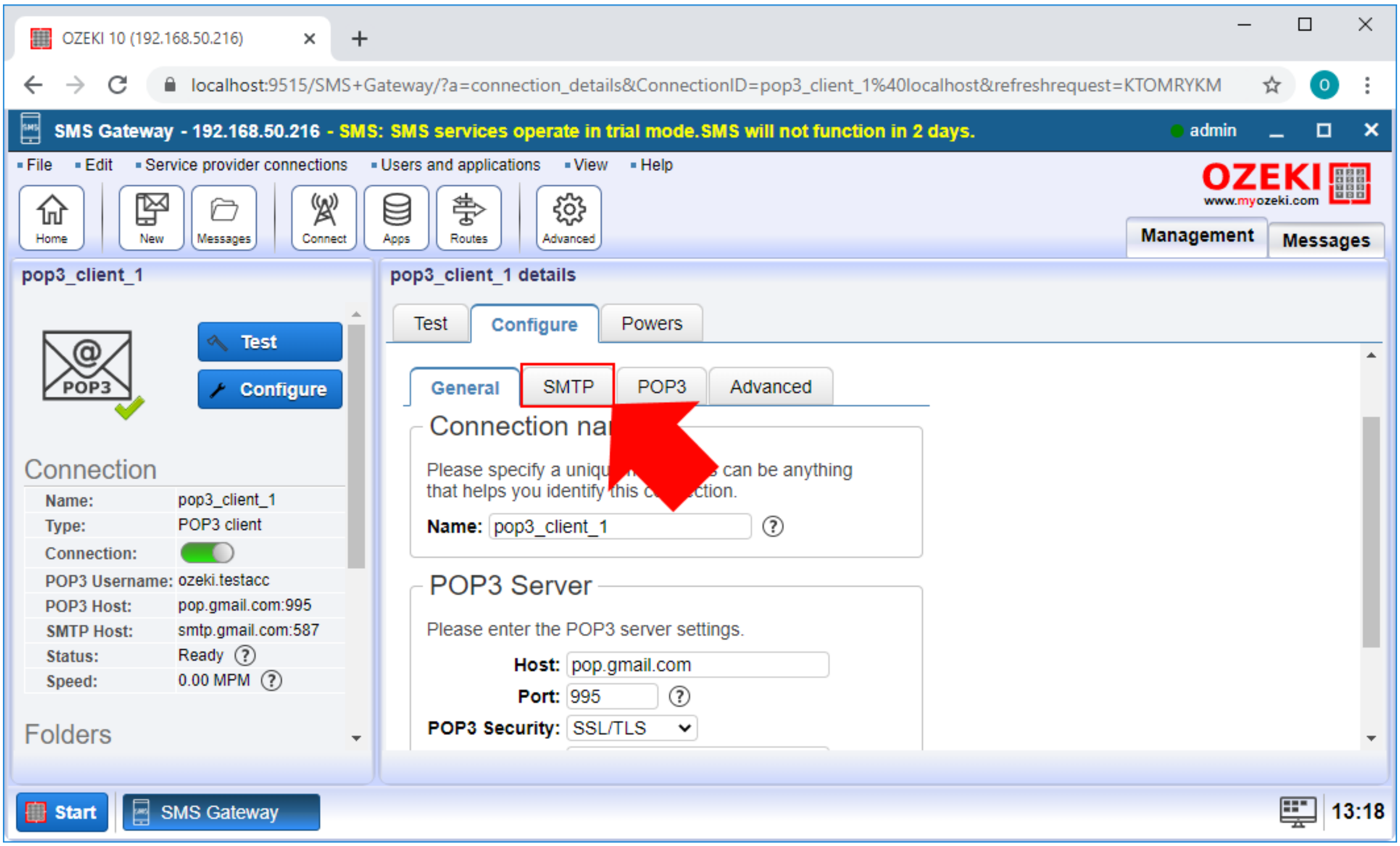


Figure 12 - Incoming SMS sent as email

## Step 8 - Configure the SMTP server

To be able to forward SMS messages as an e-mail, you need to configure an SMTP server that can send the e-mail message. This configuration can be done with the POP3 client connection. If you have created this connection before, you can just open it in SMS Gateway, but if you haven't created the connection, check Step 1, how you can do that. As Figure 13 shows that you need to click on the SMTP tab to be able to configure the details of the SMTP server.


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=connection_details&ConnectionID=pop3_client_1%40localhost&refreshrequest=KTOMRYKM
SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode.SMS will not function in 2 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
OZEKI
www.myozeki.com
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
POP3
Test
Configure
Powers
General
SMTP
POP3
Advanced
Folders
Start
SMS Gateway
13:18

pop3_client_1

Connection

| | |
|---|---|
| Name: | pop3_client_1 |
| Type: | POP3 client |
| Connection: | |
| POP3 Username: | ozeki.testacc |
| POP3 Host: | pop.gmail.com:995 |
| SMTP Host: | smtp.gmail.com:587 |
| Status: | Ready |
| Speed: | 0.00 MPM |

pop3_client_1 details

Connection na

Please specify a uniqu... can be anything that helps you identify this c...ction.

Name: pop3_client_1

POP3 Server

Please enter the POP3 server settings.

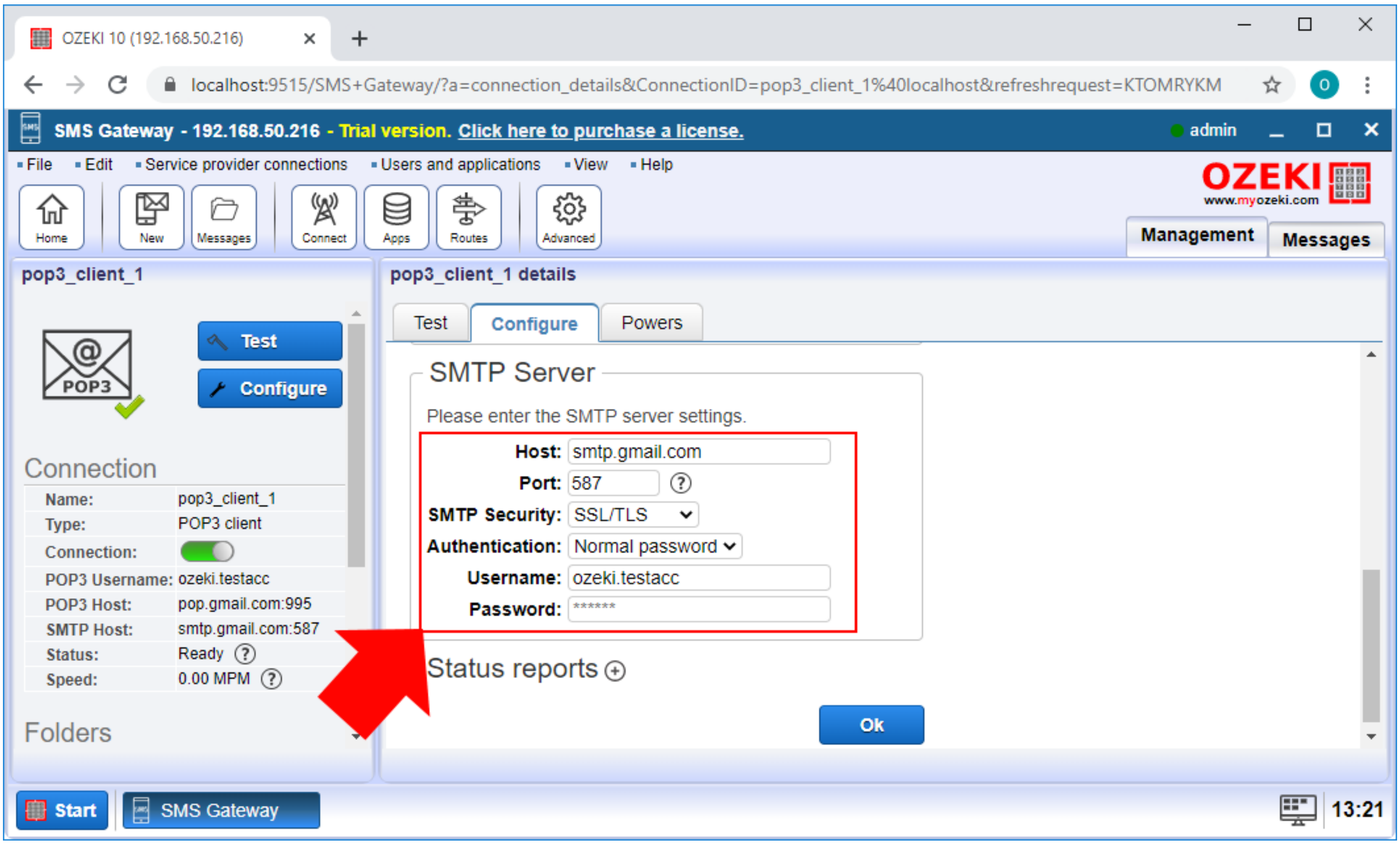
Host: pop.gmail.com

Port: 995

POP3 Security: SSL/TLS

Figure 13 - Select SMTP menu in POP3 client connection

In the SMTP menu, you can configure the details of the SMTP server. Figure 14 shows what settings you need to do to use the Gmail SMTP server. So, type 'smtp.gmail.com' as Host and 587 as the port number. For SMTP security, select 'SSL/TLS' and also select the 'Normal password' option for Authentication. Lastly, just type the username and password of your Gmail account.


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=connection_details&ConnectionID=pop3_client_1%40localhost&refreshrequest=KTOMRYKM
SMS Gateway - 192.168.50.216 - Trial version. Click here to purchase a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
pop3_client_1
Test
Configure
pop3_client_1 details
Test
Configure
Powers
Folders
Start
SMS Gateway
13:21

Connection

| | |
|---|---|
| Name: | pop3_client_1 |
| Type: | POP3 client |
| Connection: | |
| POP3 Username: | ozeki.testacc |
| POP3 Host: | pop.gmail.com:995 |
| SMTP Host: | smtp.gmail.com:587 |
| Status: | Ready |
| Speed: | 0.00 MPM |

SMTP Server

Please enter the SMTP server settings.

Host: smtp.gmail.com
Port: 587
SMTP Security: SSL/TLS
Authentication: Normal password
Username: ozeki.testacc
Password: ******

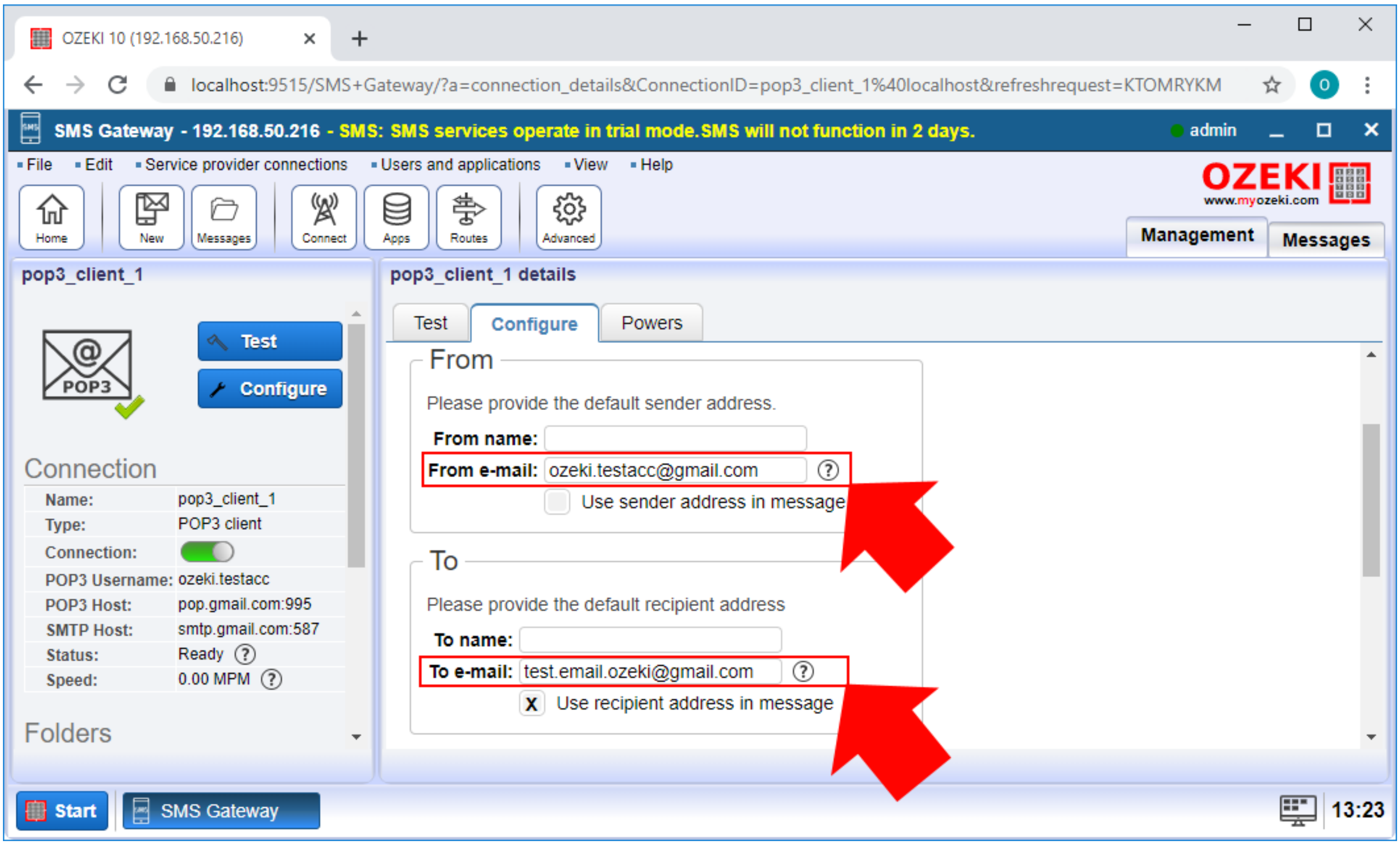
Status reports

Ok

Figure 14 - Configure the SMTP server details

# Step 9 - Configure the recipient e-mail address

The next configuration that you need to perform is to set up a recipient address which will receive the forwarded message. In the SMTP menu, type the recipient e-mail address to the 'To e-mail' field, as Figure 15 shows that. To the 'From e-mail' field, you can type the e-mail address of your Gmail account as well. This ensures that this address will show up as a sender address when the recipient gets the forwarded message. If you finished with all configuration, you can just click OK.


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=connection_details&ConnectionID=pop3_client_1%40localhost&refreshrequest=KTOMRYKM
SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode.SMS will not function in 2 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
OZEKI
www.myozeki.com
Management
Messages
Test
Configure
Powers
Start
SMS Gateway
13:23

pop3_client_1

Connection

| | |
|---|---|
| Name: | pop3_client_1 |
| Type: | POP3 client |
| Connection: | |
| POP3 Username: | ozeki.testacc |
| POP3 Host: | pop.gmail.com:995 |
| SMTP Host: | smtp.gmail.com:587 |
| Status: | Ready |
| Speed: | 0.00 MPM |

Folders

pop3_client_1 details

From

Please provide the default sender address.

From name:

From e-mail: ozeki.testacc@gmail.com

□ Use sender address in message

To

Please provide the default recipient address

To name:

To e-mail: test.email.ozeki@gmail.com

☒ Use recipient address in message

Figure 15 - Configure the recipient e-mail address

# Step 10 - Send a test SMS

With the configured SMTP server, now your system is capable of receive SMS message and forward it as an e-mail message. Figure 16 demonstrates how the solution works. The SMS arrives at your system using a service provider connection. If you don't have that kind of connection you can check how to create an SMPP client connection. Then, the message routed to the POP3 client connection, which uses its SMTP server and the given details to forward the message to the recipient e-mail address.

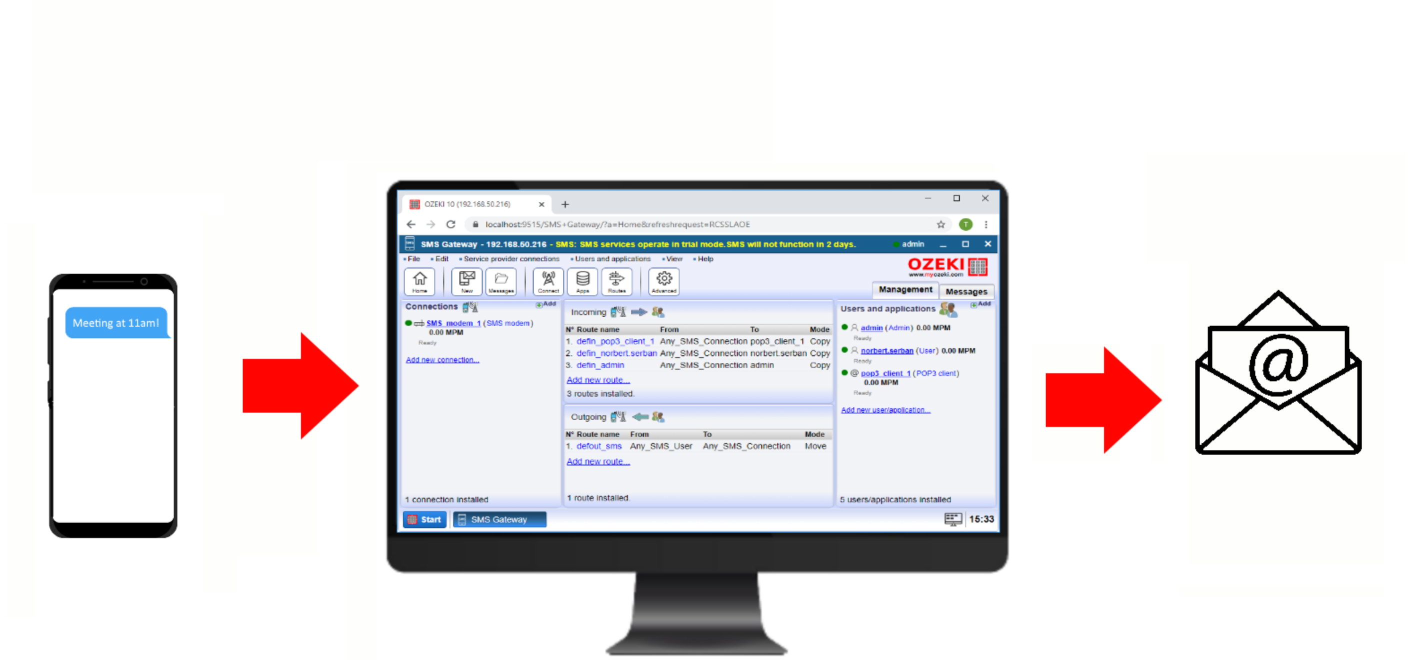
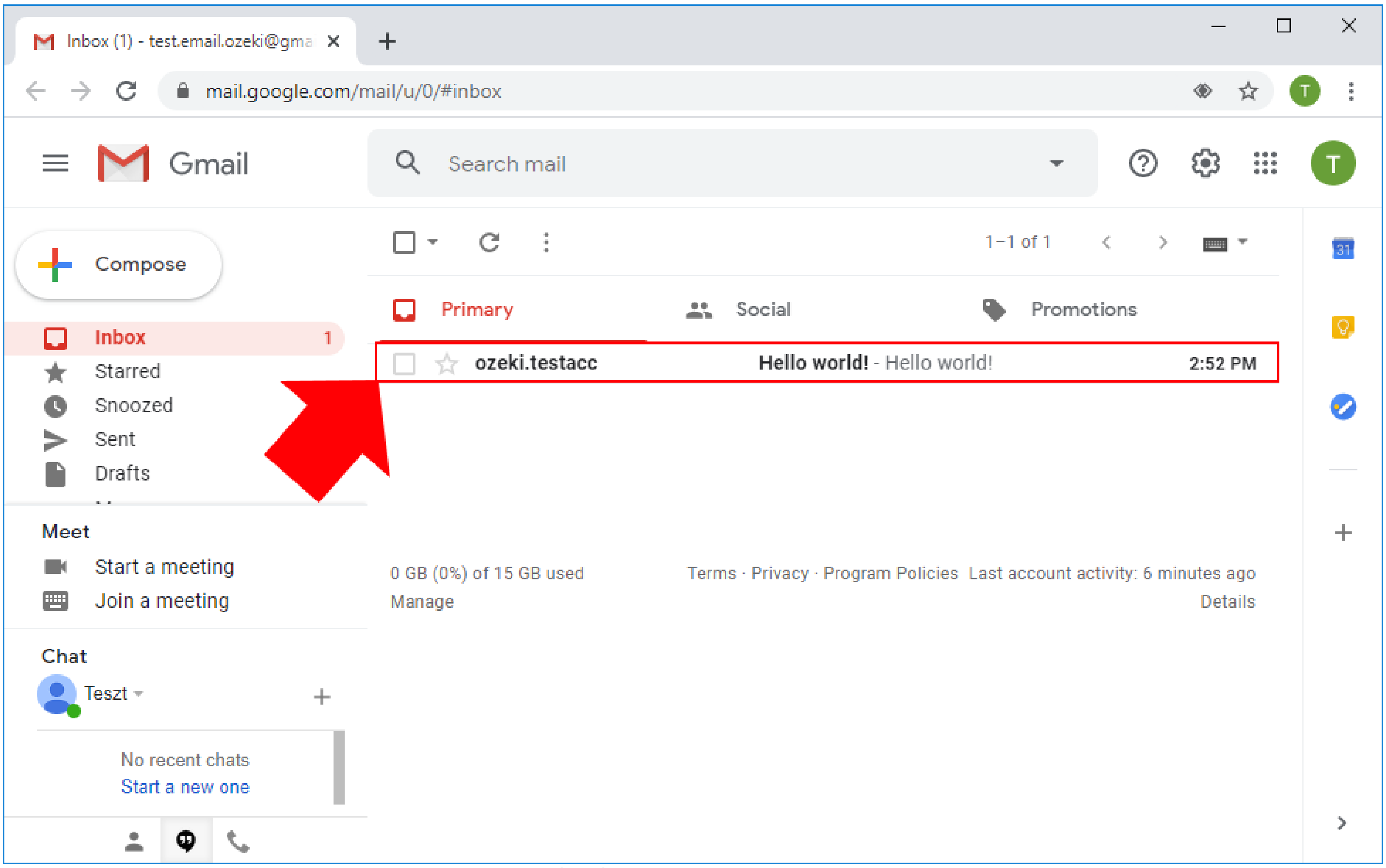


Figure 16 - Workflow of SMS to e-mail forwarding

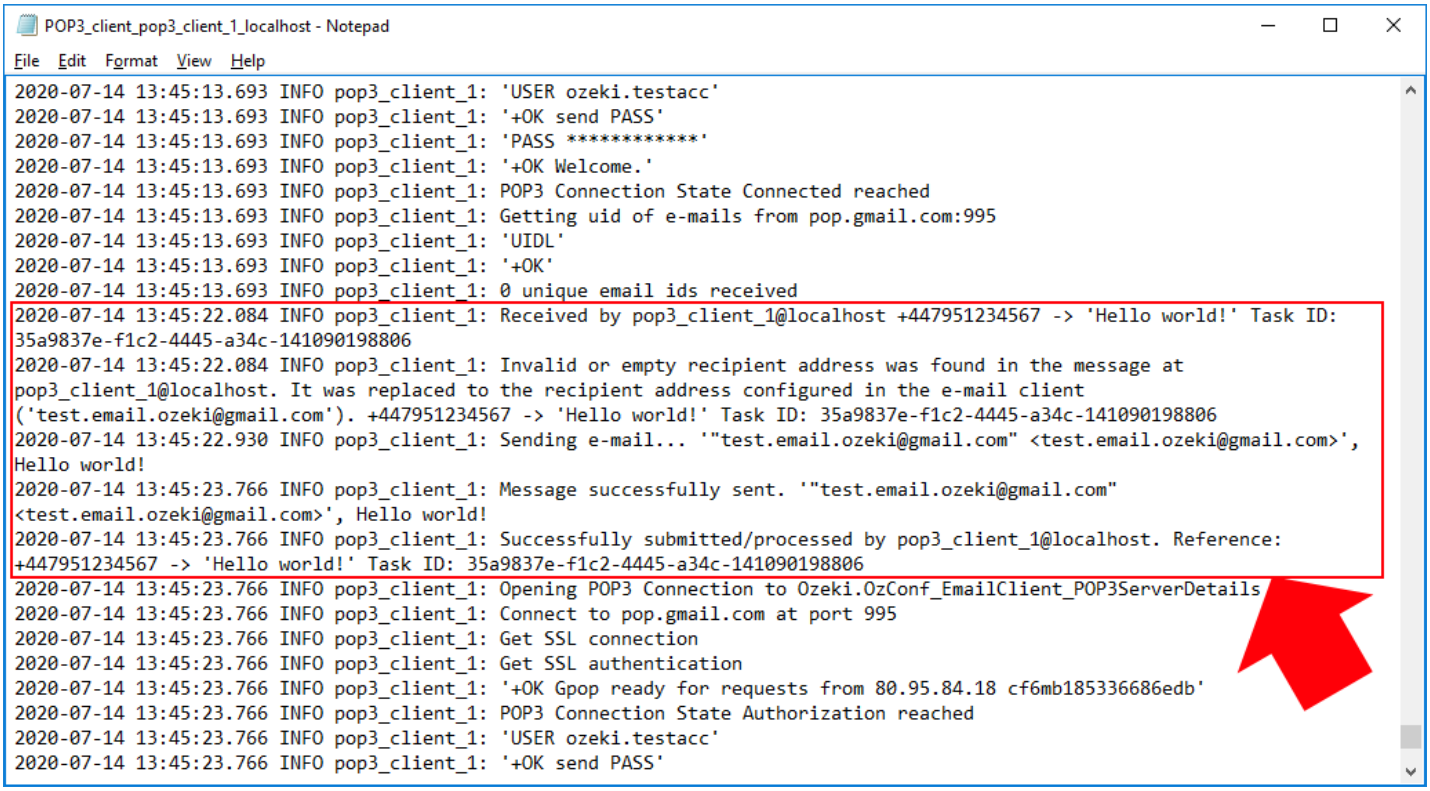


ozeki.testacc Hello world! - Hello world! 2:52 PM

Figure 16 - SMS message received as e-mail

# Step 11 - Check the logs

To check if your POP3 client connection works as expected, you can open its log file. Since you are using the same connection, the log file is located in the same folder that was described in Step 7. If you open the log file, you will be able to see every event of the connection. Figure 17 demonstrates that what events logged in case of a successful SMS to e-mail forwarding.



POP3_client_pop3_client_1_localhost - Notepad
File Edit Format View Help

```
2020-07-14 13:45:13.693 INFO pop3_client_1: 'USER ozeki.testacc'
2020-07-14 13:45:13.693 INFO pop3_client_1: '+OK send PASS'
2020-07-14 13:45:13.693 INFO pop3_client_1: 'PASS ************'
2020-07-14 13:45:13.693 INFO pop3_client_1: '+OK Welcome.'
2020-07-14 13:45:13.693 INFO pop3_client_1: POP3 Connection State Connected reached
2020-07-14 13:45:13.693 INFO pop3_client_1: Getting uid of e-mails from pop.gmail.com:995
2020-07-14 13:45:13.693 INFO pop3_client_1: 'UIDL'
2020-07-14 13:45:13.693 INFO pop3_client_1: '+OK'
2020-07-14 13:45:13.693 INFO pop3_client_1: 0 unique email ids received
2020-07-14 13:45:22.084 INFO pop3_client_1: Received by pop3_client_1@localhost +447951234567 -> 'Hello world!' Task ID:
35a9837e-f1c2-4445-a34c-141090198806
2020-07-14 13:45:22.084 INFO pop3_client_1: Invalid or empty recipient address was found in the message at
pop3_client_1@localhost. It was replaced to the recipient address configured in the e-mail client
('test.email.ozeki@gmail.com'). +447951234567 -> 'Hello world!' Task ID: 35a9837e-f1c2-4445-a34c-141090198806
2020-07-14 13:45:22.930 INFO pop3_client_1: Sending e-mail... '"test.email.ozeki@gmail.com" <test.email.ozeki@gmail.com>',
Hello world!
2020-07-14 13:45:23.766 INFO pop3_client_1: Message successfully sent. '"test.email.ozeki@gmail.com"
<test.email.ozeki@gmail.com>', Hello world!
2020-07-14 13:45:23.766 INFO pop3_client_1: Successfully submitted/processed by pop3_client_1@localhost. Reference:
+447951234567 -> 'Hello world!' Task ID: 35a9837e-f1c2-4445-a34c-141090198806
2020-07-14 13:45:23.766 INFO pop3_client_1: Opening POP3 Connection to Ozeki.OzConf_EmailClient_POP3ServerDetails
2020-07-14 13:45:23.766 INFO pop3_client_1: Connect to pop.gmail.com at port 995
2020-07-14 13:45:23.766 INFO pop3_client_1: Get SSL connection
2020-07-14 13:45:23.766 INFO pop3_client_1: Get SSL authentication
2020-07-14 13:45:23.766 INFO pop3_client_1: '+OK Gpop ready for requests from 80.95.84.18 cf6mb185336686edb'
2020-07-14 13:45:23.766 INFO pop3_client_1: POP3 Connection State Authorization reached
2020-07-14 13:45:23.766 INFO pop3_client_1: 'USER ozeki.testacc'
2020-07-14 13:45:23.766 INFO pop3_client_1: '+OK send PASS'
```

Figure 17 - Log events of SMS to e-mail forwarding

# How to setup SMS to E-mail forwarding

***This step-by-step guide shows how you can configure an E-mail User, so it can forward incoming SMS messages from a standard email client. The recipient's email address can be configured in the E-mail User or it can be the first word in the received SMS message.***

An E-mail User can be installed and configured on Ozeki SMS Gateway to send and receive email messages. This page is about sending, so you should connect this user to an SMTP server (Figure 1).

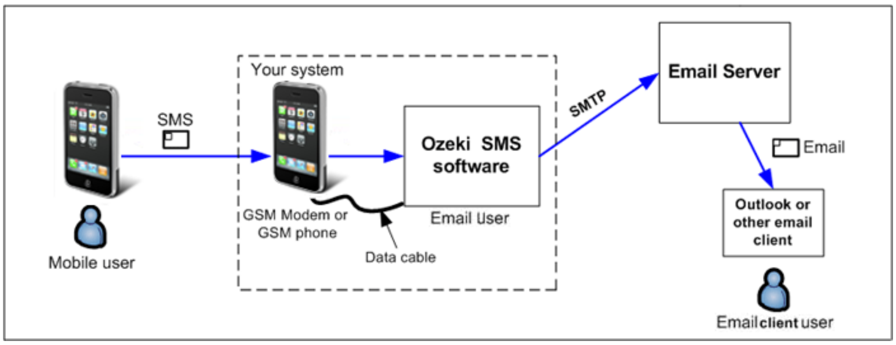
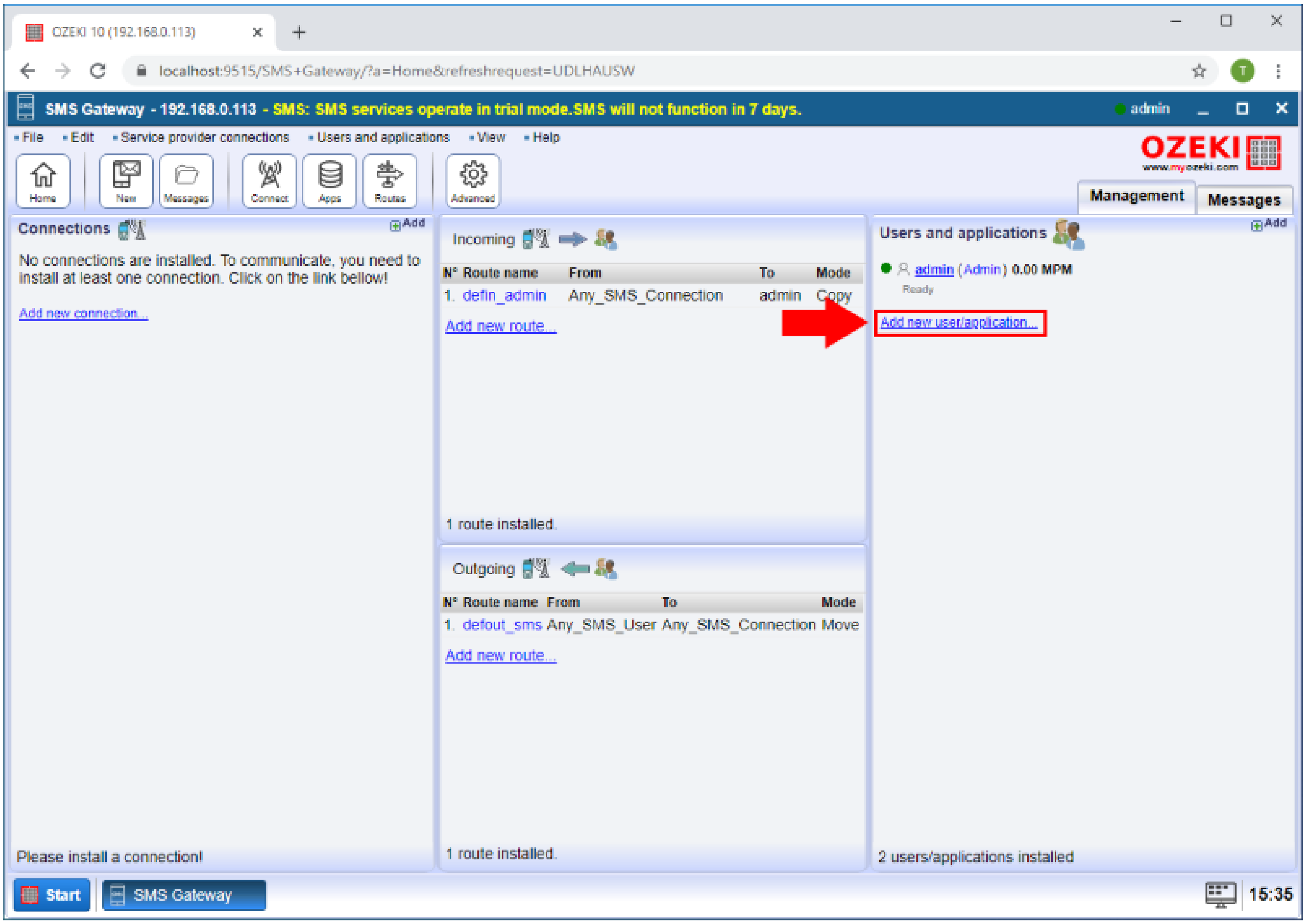


Figure 1 - Incoming SMS sent as email

E-mail Users can be easily installed and configured through the browser GUI of Ozeki SMS Gateway. So first, open the SMS Gateway, and click on the Add new user/application (Figure 2).



Connections

No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

Please install a connection!

Incoming

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defin_admin | Any_SMS_Connection | admin | Copy |

Add new route...

1 route installed.

Outgoing

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move |

Add new route...

1 route installed.

Users and applications

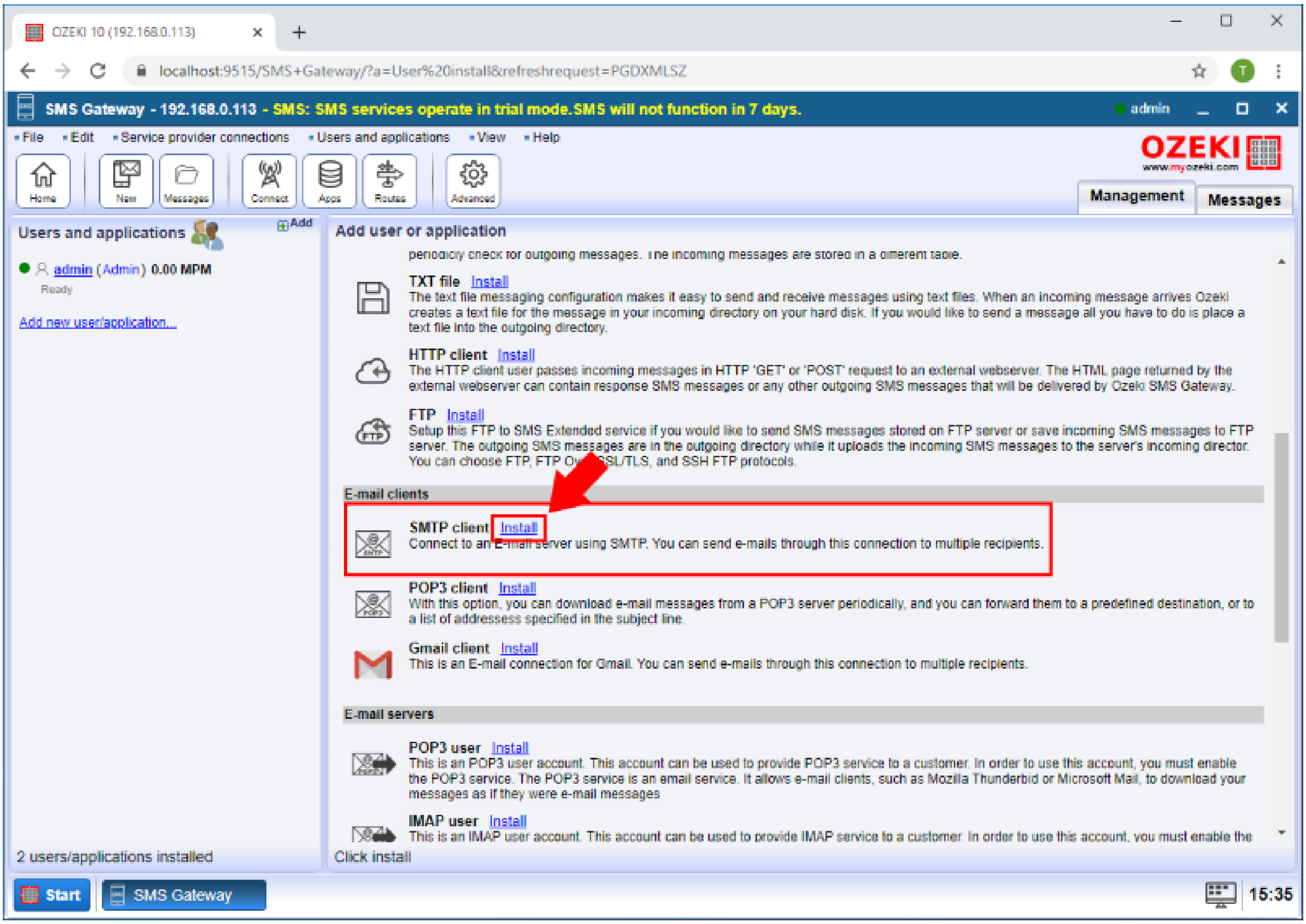
● admin (Admin) 0.00 MPM
Ready

Add new user/application...

2 users/applications installed

Figure 2 - Add new user

Here, in this menu, scroll down to the E-mail clients submenu, and like in Figure 3, select SMTP client by clicking on Install.



periodicly check for outgoing messages. The incoming messages are stored in a different table.

**TXT file** Install
The text file messaging configuration makes it easy to send and receive messages using text files. When an incoming message arrives Ozeki creates a text file for the message in your incoming directory on your hard disk. If you would like to send a message all you have to do is place a text file into the outgoing directory.

**HTTP client** Install
The HTTP client user passes incoming messages in HTTP 'GET' or 'POST' request to an external webserver. The HTML page returned by the external webserver can contain response SMS messages or any other outgoing SMS messages that will be delivered by Ozeki SMS Gateway.

**FTP** Install
Setup this FTP to SMS Extended service if you would like to send SMS messages stored on FTP server or save incoming SMS messages to FTP server. The outgoing SMS messages are in the outgoing directory while it uploads the incoming SMS messages to the server's incoming director. You can choose FTP, FTP Over SSL/TLS, and SSH FTP protocols.

**SMTP client** Install
Connect to an E-mail server using SMTP. You can send e-mails through this connection to multiple recipients.

**POP3 client** Install
With this option, you can download e-mail messages from a POP3 server periodically, and you can forward them to a predefined destination, or to a list of addressess specified in the subject line.

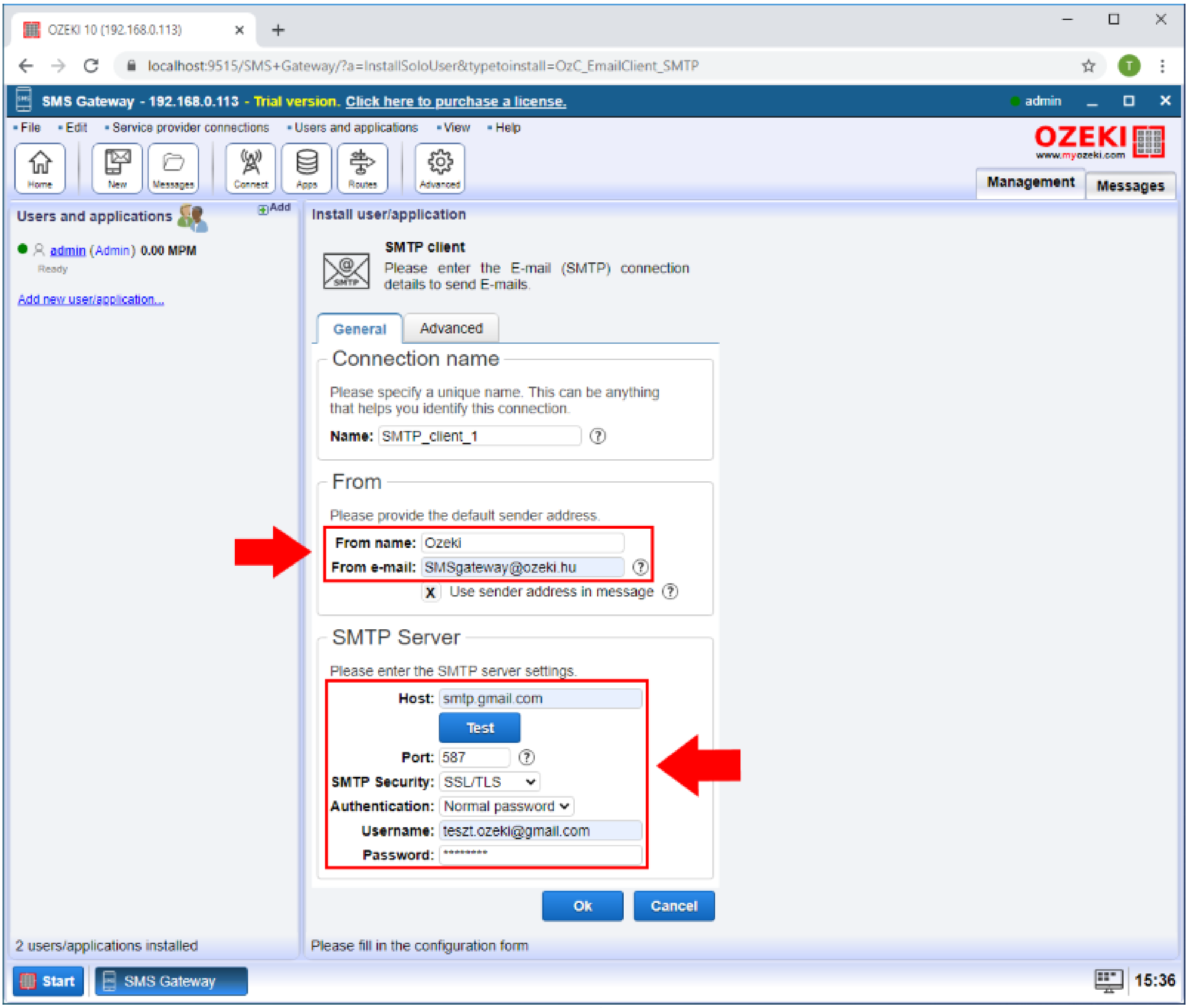
**Gmail client** Install
This is an E-mail connection for Gmail. You can send e-mails through this connection to multiple recipients.

**POP3 user** Install
This is an POP3 user account. This account can be used to provide POP3 service to a customer. In order to use this account, you must enable the POP3 service. The POP3 service is an email service. It allows e-mail clients, such as Mozilla Thunderbid or Microsoft Mail, to download your messages as if they were e-mail messages

**IMAP user** Install
This is an IMAP user account. This account can be used to provide IMAP service to a customer. In order to use this account, you must enable the

Figure 3 - Install SMTP Client

SMTP is an internet standard for email transmission. SMTP servers are used for relaying outgoing emails. On the From section provide the default sender address and On the SMTP server section, please enter the SMTP server's IP address or domain and the port number as you can see in the Figure 3. In case your SMTP server requires authentication, provide the username and password. You can also choose SSL connection if your SMTP server has a more secure connection.


OZEKI 10 (192.168.0.113)
localhost:9515/SMS+Gateway/?a=InstallSoloUser&typetoinstall=OzC_EmailClient_SMTP
SMS Gateway - 192.168.0.113 - Trial version. Click here to purchase a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Users and applications
Add
admin (Admin) 0.00 MPM
Ready
Add new user/application...
2 users/applications installed
Start
SMS Gateway
15:36

Install user/application

SMTP client

Please enter the E-mail (SMTP) connection details to send E-mails.

General | Advanced

Connection name

Please specify a unique name. This can be anything that helps you identify this connection.

Name: SMTP_client_1

From

Please provide the default sender address.

From name: Ozeki

From e-mail: SMSgateway@ozeki.hu

☒ Use sender address in message

SMTP Server

Please enter the SMTP server settings.

Host: smtp.gmail.com

Test

Port: 587

SMTP Security: SSL/TLS

Authentication: Normal password

Username: teszt.ozeki@gmail.com

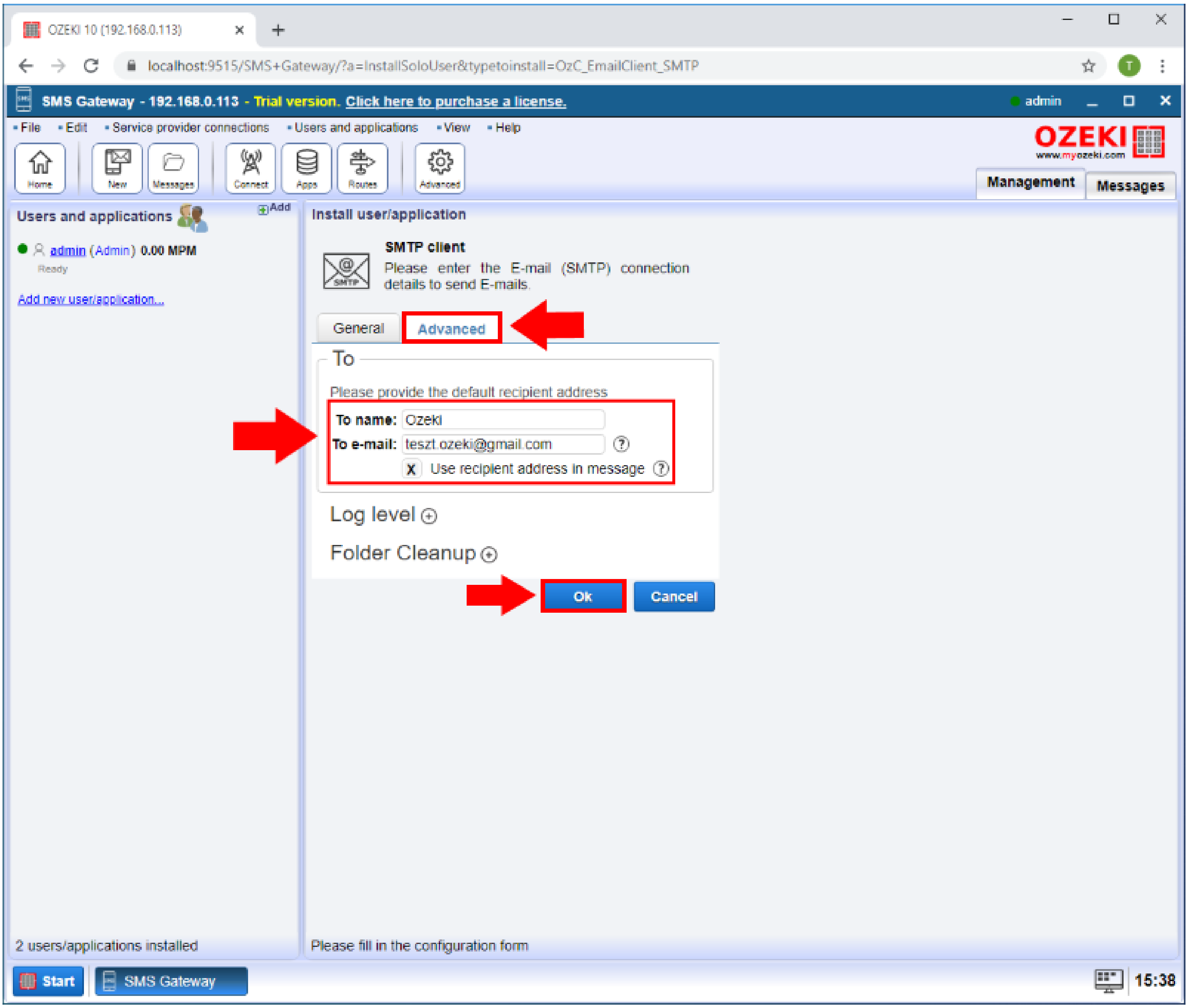
Password: ********

Ok | Cancel

Please fill in the configuration form

Figure 3 - Provide the SMTP server settings

Now switch to the Advanced tab (Figure 4) and make sure the emails are sent to the correct address. In the textboxes you can set the default recipient name and address.


OZEKI 10 (192.168.0.113)
localhost:9515/SMS+Gateway/?a=InstallSoloUser&typetoinstall=OzC_EmailClient_SMTP
SMS Gateway - 192.168.0.113 - Trial version. Click here to purchase a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
OZEKI
www.myozeki.com
Management
Messages
Users and applications
Add
admin (Admin) 0.00 MPM
Ready
Add new user/application...
Install user/application
SMTP client
Please enter the E-mail (SMTP) connection details to send E-mails.
General
Advanced
Log level
Folder Cleanup
Ok
Cancel
2 users/applications installed
Please fill in the configuration form
Start
SMS Gateway
15:38

To

Please provide the default recipient address

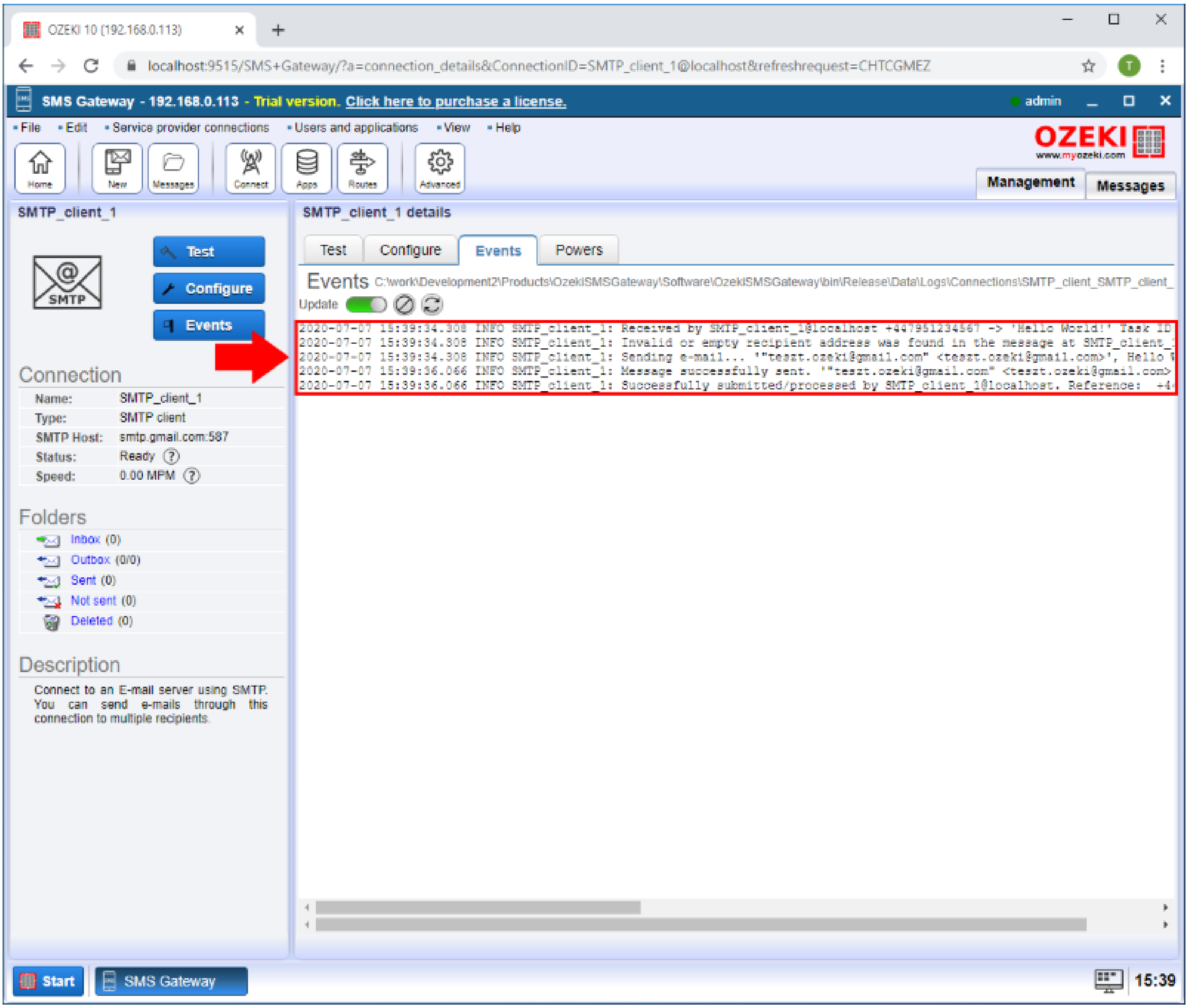
To name: Ozeki

To e-mail: teszt.ozeki@gmail.com

☒ Use recipient address in message

Figure 4 - Enter recipient address

Finally in the Events tab you can see that when an sms arrives on this SMTP client it is forwarded to the SMTP server which sends it to the recipient's email address.



OZEKI 10 (192.168.0.113)
localhost:9515/SMS+Gateway/?a=connection_details&ConnectionID=SMTP_client_1@localhost&refreshrequest=CHTCGMEZ
SMS Gateway - 192.168.0.113 - Trial version. Click here to purchase a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
OZEKI
www.myozeki.com
Management
Messages
SMTP_client_1
Test
Configure
Events
SMTP_client_1 details
Test
Configure
Events
Powers
Events C:\work\Development2\Products\OzekiSMSGateway\Software\OzekiSMSGateway\bin\Release\Data\Logs\Connections\SMTP_client_SMTP_client_
Update
Connection
Folders
Inbox (0)
Outbox (0/0)
Sent (0)
Not sent (0)
Deleted (0)
Description
Start
SMS Gateway
15:39

```
2020-07-07 15:39:34.308 INFO SMTP_client_1: Received by SMTP_client_1@localhost +447951234567 -> 'Hello World!' Task ID
2020-07-07 15:39:34.308 INFO SMTP_client_1: Invalid or empty recipient address was found in the message at SMTP_client_
2020-07-07 15:39:34.308 INFO SMTP_client_1: Sending e-mail... '"teszt.ozeki@gmail.com" <teszt.ozeki@gmail.com>', Hello W
2020-07-07 15:39:36.066 INFO SMTP_client_1: Message successfully sent. '"teszt.ozeki@gmail.com" <teszt.ozeki@gmail.com>
2020-07-07 15:39:36.066 INFO SMTP_client_1: Successfully submitted/processed by SMTP_client_1@localhost. Reference:  +4
```

| | |
|---|---|
| Name: | SMTP_client_1 |
| Type: | SMTP client |
| SMTP Host: | smtp.gmail.com:587 |
| Status: | Ready |
| Speed: | 0.00 MPM |

Connect to an E-mail server using SMTP. You can send e-mails through this connection to multiple recipients.

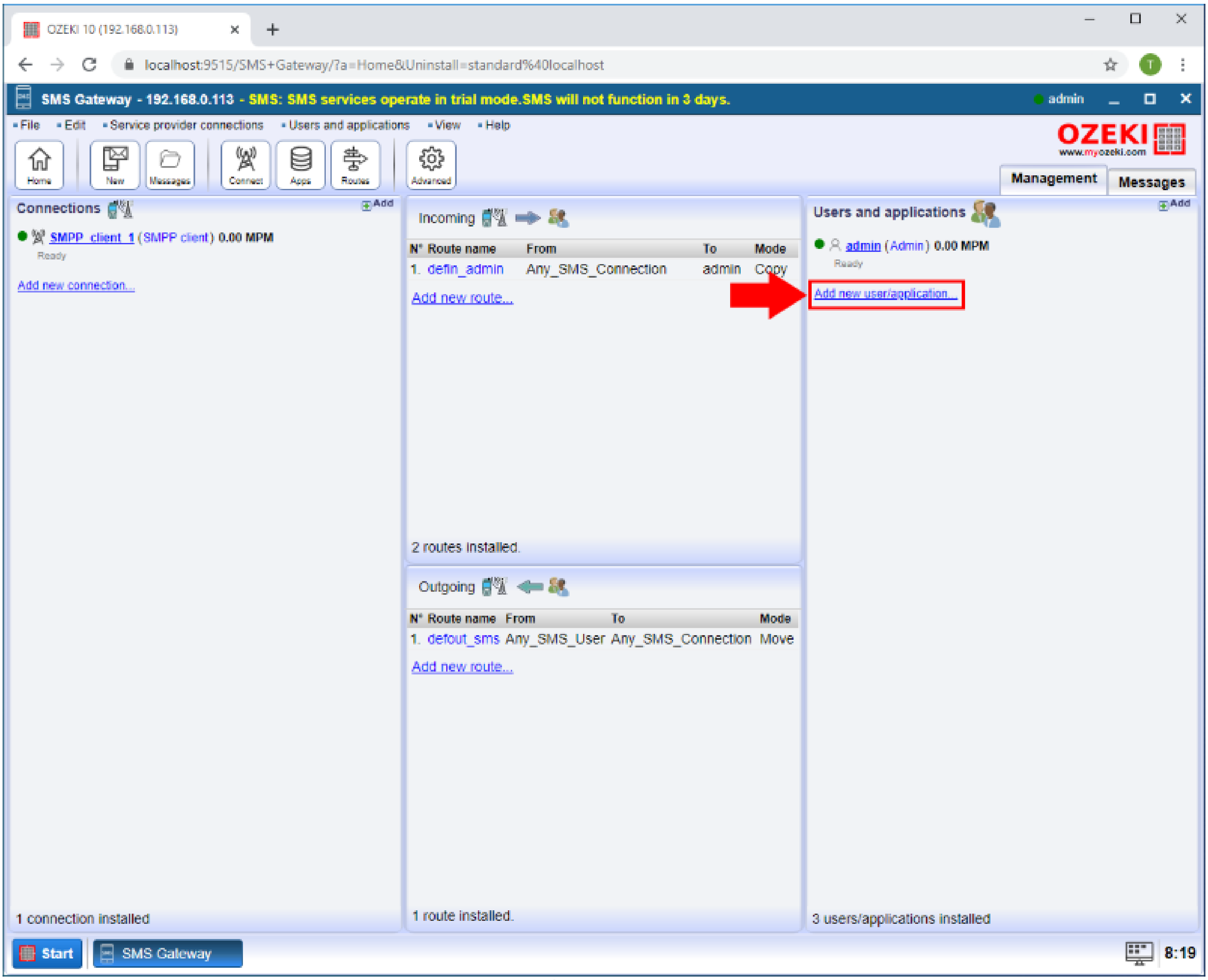
Figure 5 - SMS sent as e-mail

# SMS as E-mail through Gmail

***The Ozeki SMS Gateway can send messages through your Gmail's SMTP connection so it can forward incoming SMS messages by email.***

## Step 1 - Add new Gmail user

Gmail Users can be easily installed and configured through the browser GUI of Ozeki SMS Gateway. So first, open the SMS Gateway, and click on the Add new user/application (Figure 1).



SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.SMS will not function in 3 days.

Connections

● SMPP_client_1 (SMPP client) 0.00 MPM
Ready

Add new connection...

1 connection installed

Incoming

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defin_admin | Any_SMS_Connection | admin | Copy |

Add new route...

2 routes installed.

Outgoing

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move |

Add new route...

1 route installed.

Users and applications

● admin (Admin) 0.00 MPM
Ready

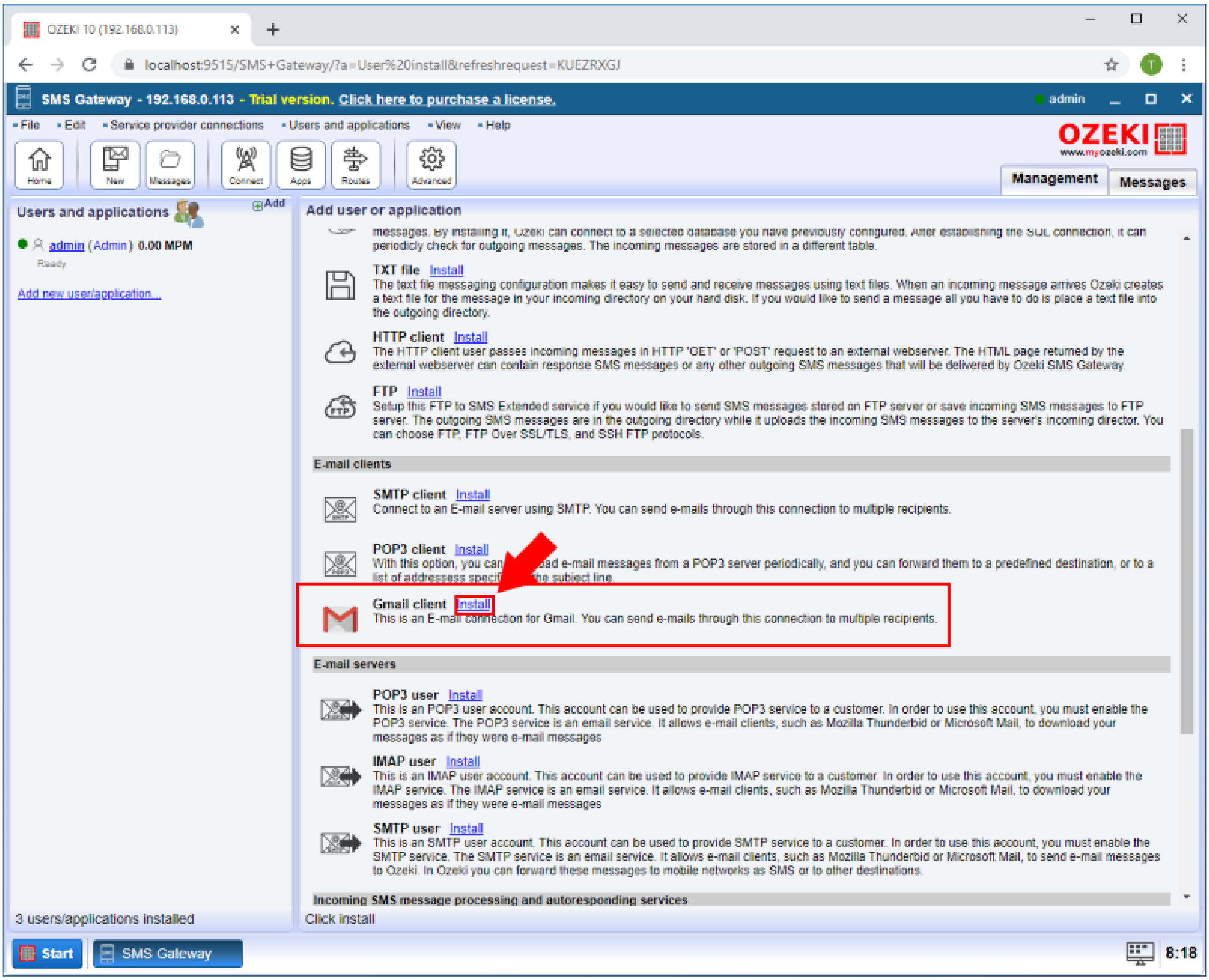
Add new user/application...

3 users/applications installed

Figure 1 - Add new user

## Step 2 - Add Gmail client connection

Here, in this menu, scroll down to the E-mail clients submenu, and like in Figure 2, select Gmail client by clicking on Install.


OZEKI 10 (192.168.0.113)
localhost:9515/SMS+Gateway/?a=User%20install&refreshrequest=KUEZRXGJ
SMS Gateway - 192.168.0.113 - Trial version. Click here to purchase a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
OZEKI
www.myozeki.com
Management
Messages
Users and applications
Add
admin (Admin) 0.00 MPM
Ready
Add new user/application...
3 users/applications installed
Start
SMS Gateway
8:18

Add user or application

messages. By installing it, Ozeki can connect to a selected database you have previously configured. After establishing the SQL connection, it can periodicly check for outgoing messages. The incoming messages are stored in a different table.

TXT file Install
The text file messaging configuration makes it easy to send and receive messages using text files. When an incoming message arrives Ozeki creates a text file for the message in your incoming directory on your hard disk. If you would like to send a message all you have to do is place a text file into the outgoing directory.

HTTP client Install
The HTTP client user passes incoming messages in HTTP 'GET' or 'POST' request to an external webserver. The HTML page returned by the external webserver can contain response SMS messages or any other outgoing SMS messages that will be delivered by Ozeki SMS Gateway.

FTP Install
Setup this FTP to SMS Extended service if you would like to send SMS messages stored on FTP server or save incoming SMS messages to FTP server. The outgoing SMS messages are in the outgoing directory while it uploads the incoming SMS messages to the server's incoming director. You can choose FTP, FTP Over SSL/TLS, and SSH FTP protocols.

E-mail clients

SMTP client Install
Connect to an E-mail server using SMTP. You can send e-mails through this connection to multiple recipients.

POP3 client Install
With this option, you can download e-mail messages from a POP3 server periodically, and you can forward them to a predefined destination, or to a list of addresses specified in the subject line.

Gmail client Install
This is an E-mail connection for Gmail. You can send e-mails through this connection to multiple recipients.

E-mail servers

POP3 user Install
This is an POP3 user account. This account can be used to provide POP3 service to a customer. In order to use this account, you must enable the POP3 service. The POP3 service is an email service. It allows e-mail clients, such as Mozilla Thunderbird or Microsoft Mail, to download your messages as if they were e-mail messages

IMAP user Install
This is an IMAP user account. This account can be used to provide IMAP service to a customer. In order to use this account, you must enable the IMAP service. The IMAP service is an email service. It allows e-mail clients, such as Mozilla Thunderbird or Microsoft Mail, to download your messages as if they were e-mail messages

SMTP user Install
This is an SMTP user account. This account can be used to provide SMTP service to a customer. In order to use this account, you must enable the SMTP service. The SMTP service is an email service. It allows e-mail clients, such as Mozilla Thunderbird or Microsoft Mail, to send e-mail messages to Ozeki. In Ozeki you can forward these messages to mobile networks as SMS or to other destinations.

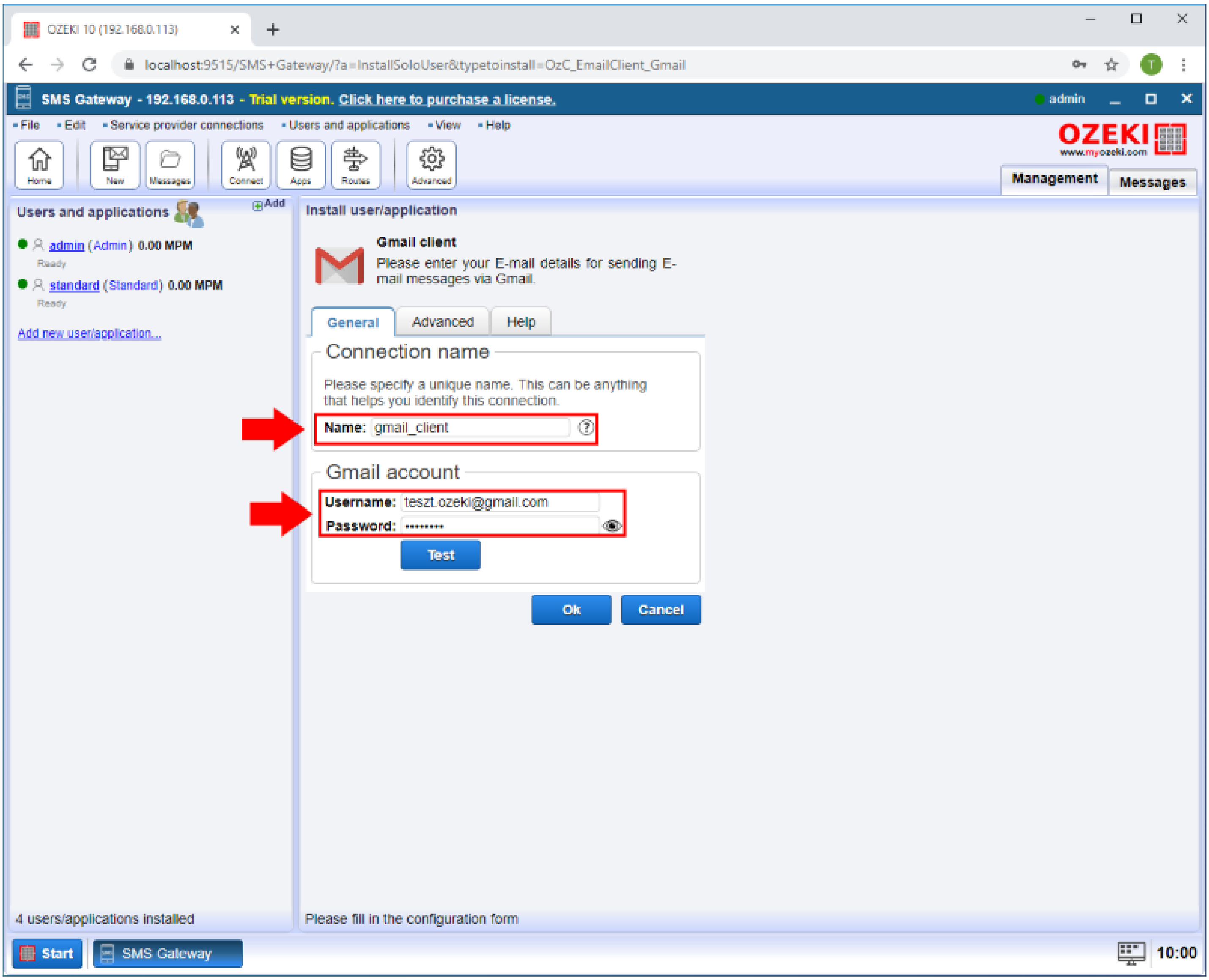
Incoming SMS message processing and autoresponding services

Click install

Figure 2 - Install Gmail Client

## Step 3 - Provide the Gmail settings

On the Name section provide the unique name for the Gmail client and for the authentication, provide the username and password of you Gmail account.


OZEKI 10 (192.168.0.113)
localhost:9515/SMS+Gateway/?a=InstallSoloUser&typetoinstall=OzC_EmailClient_Gmail
SMS Gateway - 192.168.0.113 - Trial version. Click here to purchase a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Users and applications
Add
admin (Admin) 0.00 MPM
Ready
standard (Standard) 0.00 MPM
Ready
Add new user/application...
4 users/applications installed
Start
SMS Gateway
10:00

Install user/application

**Gmail client**

Please enter your E-mail details for sending E-mail messages via Gmail.

General | Advanced | Help

Connection name

Please specify a unique name. This can be anything that helps you identify this connection.

**Name:** gmail_client

Gmail account

**Username:** teszt.ozeki@gmail.com

**Password:** ••••••••

Test

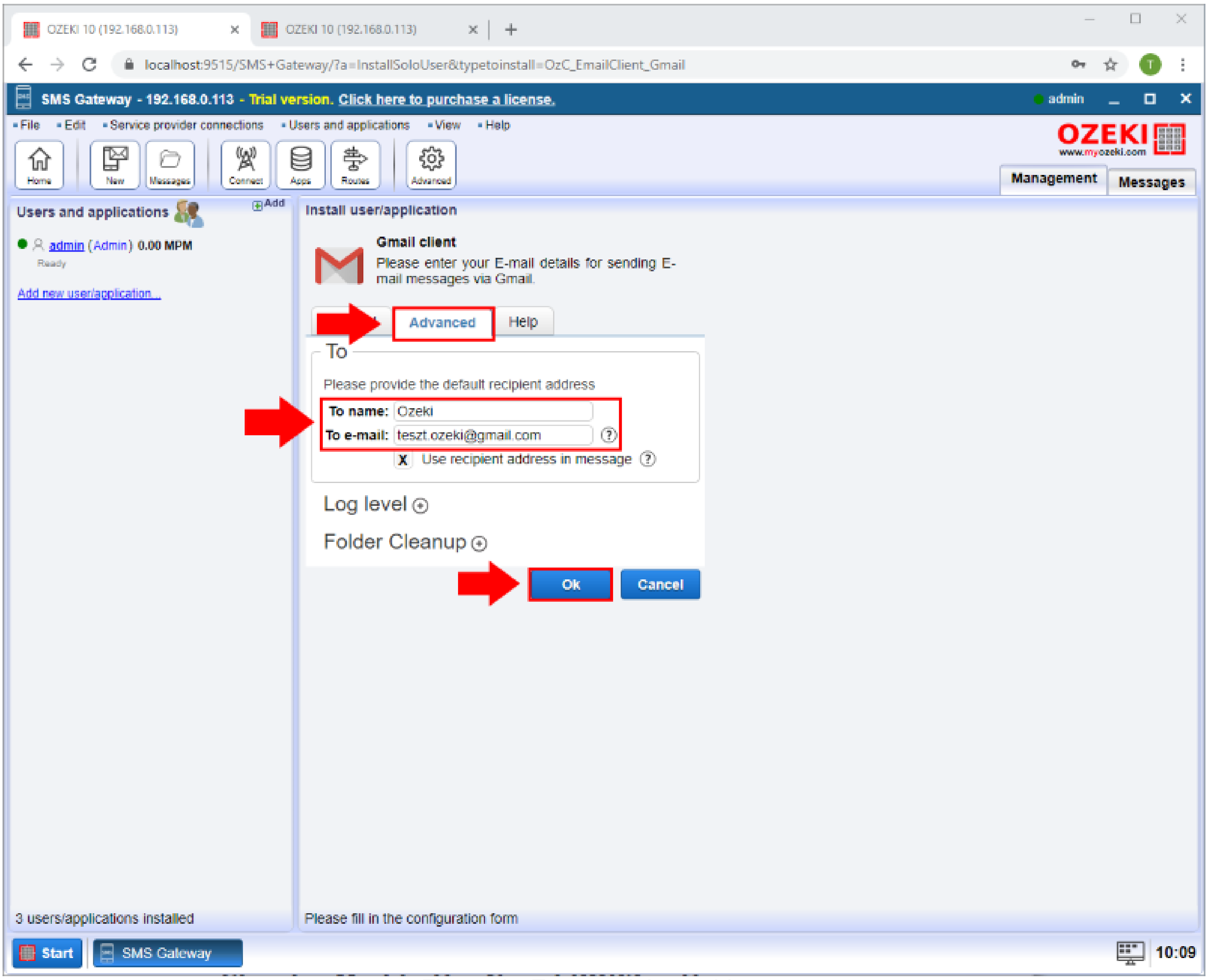
Ok | Cancel

Please fill in the configuration form

Figure 3 - Provide the Gmail settings

# Step 4 - Enter recipient address

Now switch to the Advanced tab (Figure 4) and make sure the emails are sent to the correct address. In the textboxes you can set the default recipient name and address.



Gmail client

Please enter your E-mail details for sending E-mail messages via Gmail.

To

Please provide the default recipient address

To name: Ozeki

To e-mail: teszt.ozeki@gmail.com

☒ Use recipient address in message

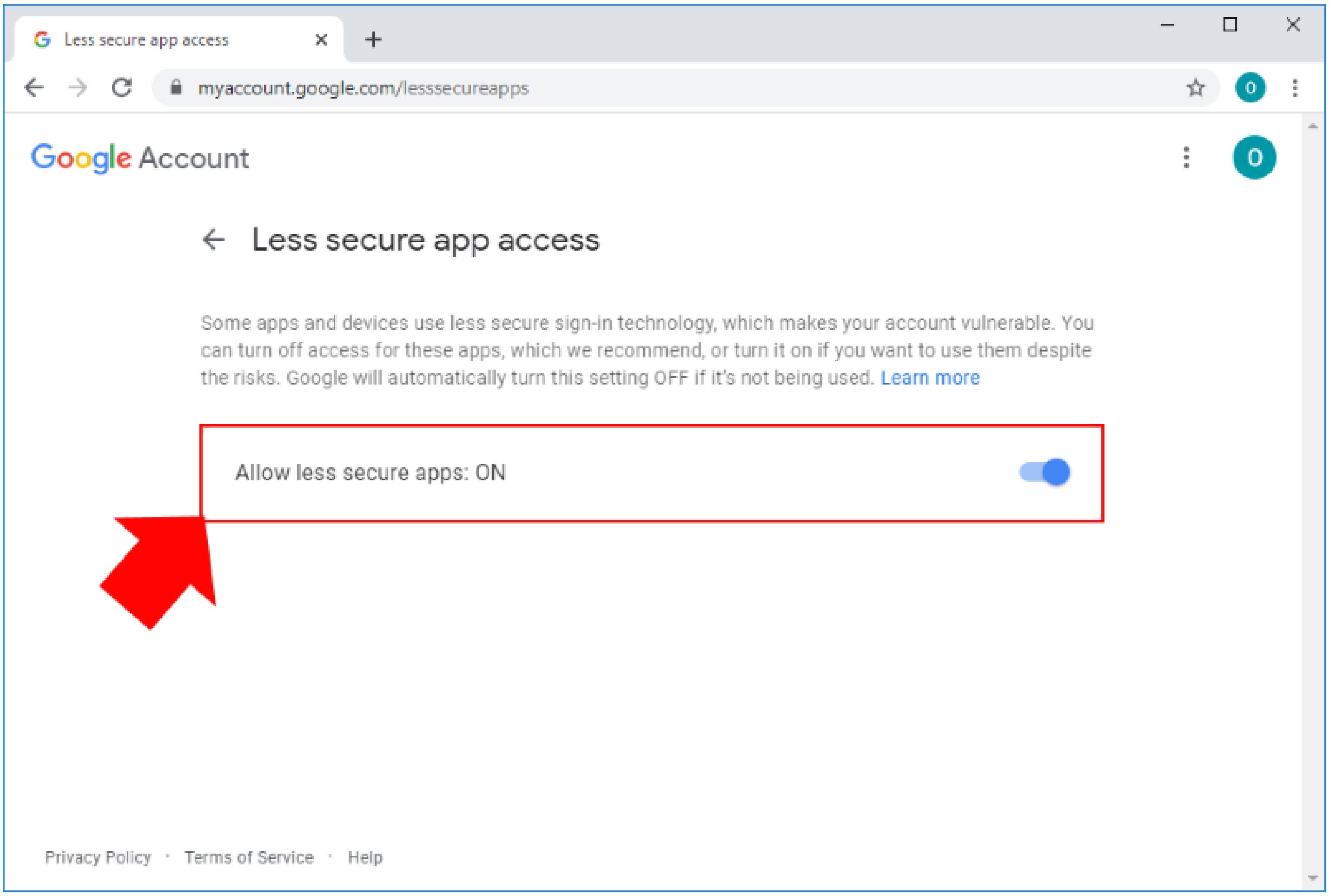
3 users/applications installed

Please fill in the configuration form

Figure 4 - Enter recipient address

## Step 5 - Configure the security settings of Gmail account

After you created the Gmail client connection, you need to do some modifications to the security settings of your Gmail account. For that, just type 'https://myaccount.google.com/lesssecureapps' in your browser and hit Enter. Here (Figure 5), you just need to turn the 'Less secure apps access' on. This will ensure that SMS Gateway is going to have access to Gmail's server.



Less secure app access

Some apps and devices use less secure sign-in technology, which makes your account vulnerable. You can turn off access for these apps, which we recommend, or turn it on if you want to use them despite the risks. Google will automatically turn this setting OFF if it's not being used. Learn more

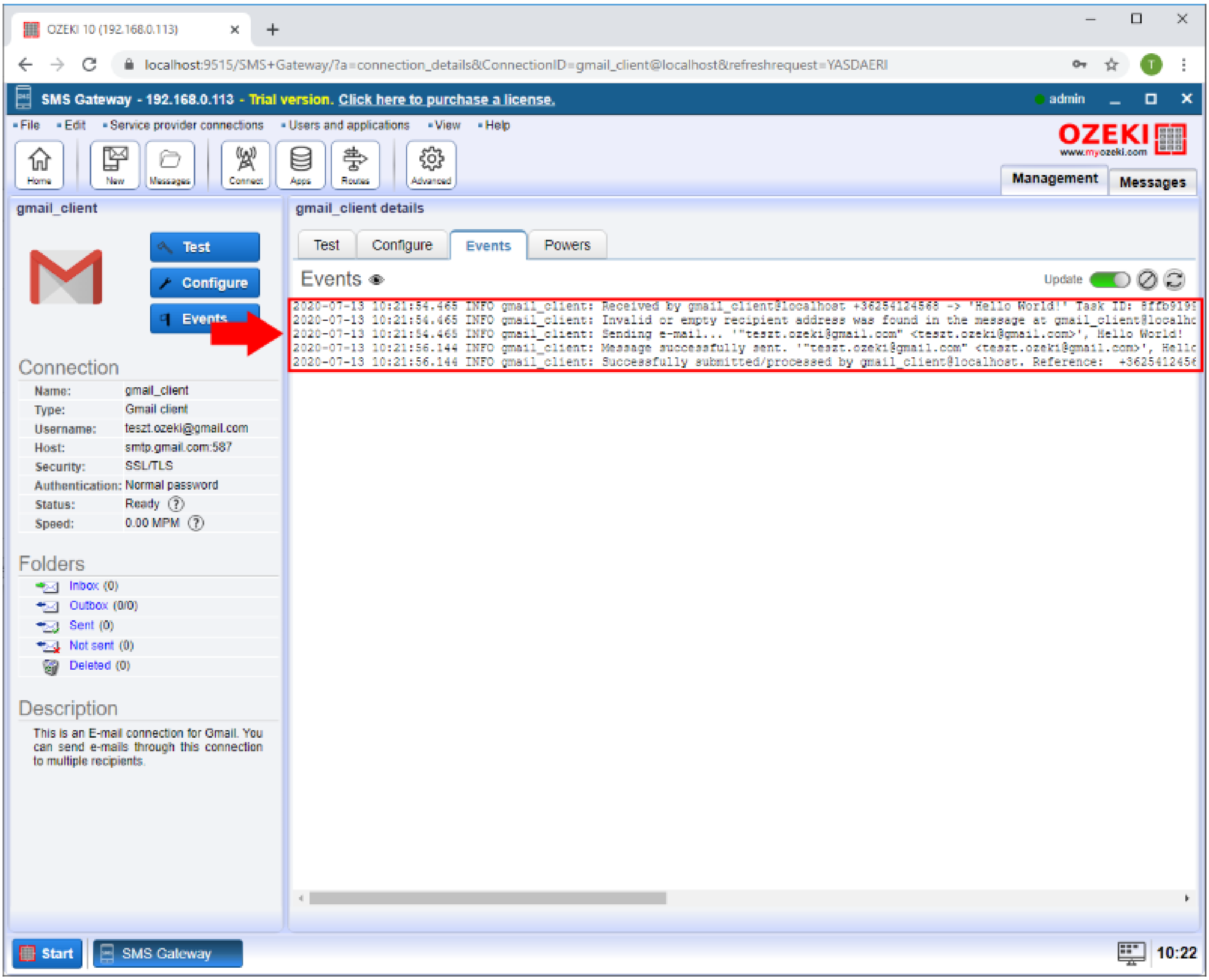
Allow less secure apps: ON

Less secure app access
myaccount.google.com/lesssecureapps
Google Account
Privacy Policy · Terms of Service · Help

Figure 5 - Turning on access for less secure apps

## Step 6 - SMS sent as e-mail

Finally in the Events tab you can see that when an sms arrives on this SMTP client it is forwarded to the SMTP server which sends it to the recipient's email address (Figure 6).



```
2020-07-13 10:21:54.465 INFO gmail_client: Received by gmail_client@localhost +36254124568 -> 'Hello World!' Task ID: 8ffb919
2020-07-13 10:21:54.465 INFO gmail_client: Invalid or empty recipient address was found in the message at gmail_client@localhc
2020-07-13 10:21:54.465 INFO gmail_client: Sending e-mail... '"teszt.ozeki@gmail.com" <teszt.ozeki@gmail.com>', Hello World!
2020-07-13 10:21:56.144 INFO gmail_client: Message successfully sent. '"teszt.ozeki@gmail.com" <teszt.ozeki@gmail.com>', Hellc
2020-07-13 10:21:56.144 INFO gmail_client: Successfully submitted/processed by gmail_client@localhost. Reference:  +3625412456
```

Connection

| | |
|---|---|
| Name: | gmail_client |
| Type: | Gmail client |
| Username: | teszt.ozeki@gmail.com |
| Host: | smtp.gmail.com:587 |
| Security: | SSL/TLS |
| Authentication: | Normal password |
| Status: | Ready ? |
| Speed: | 0.00 MPM ? |

Folders

- Inbox (0)
- Outbox (0/0)
- Sent (0)
- Not sent (0)
- Deleted (0)

Description

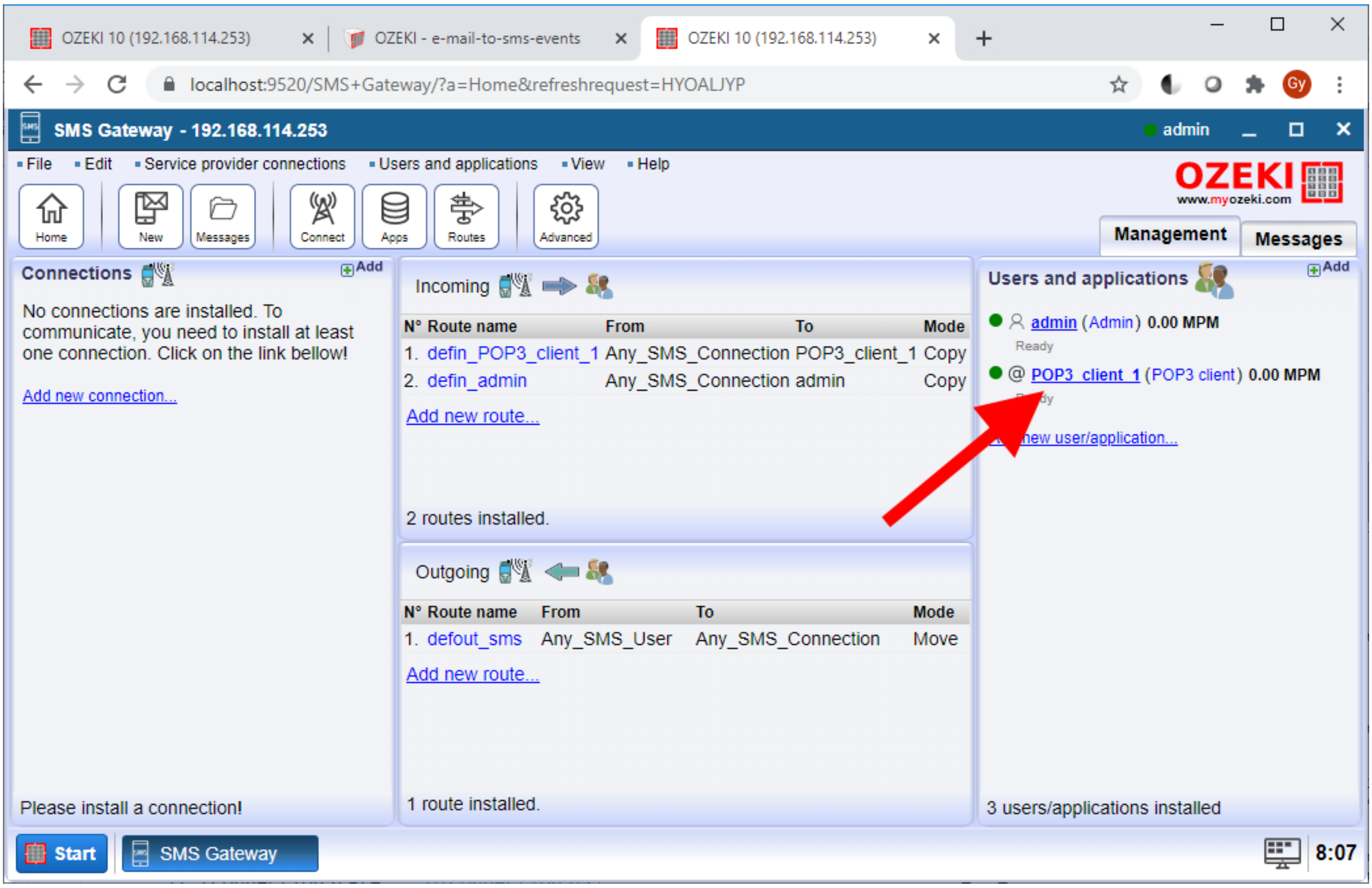
This is an E-mail connection for Gmail. You can send e-mails through this connection to multiple recipients.

Figure 6 - SMS sent as e-mail

# E-mail to SMS events

*This guide gives you information on how to view the event log of e-mail to SMS connection. It focuses on the built in SMTP/POP3 client of Ozeki 10 SMS gateway. It shows you how you can open the "Powers" tab, and how you can enable the event viewer functionality. The steps are relatively simple. The setup should not take more than 3 minutes.*

## Step 1 - Open the e-mail to SMS connection


OZEKI 10 (192.168.114.253)
OZEKI - e-mail-to-sms-events
OZEKI 10 (192.168.114.253)
localhost:9520/SMS+Gateway/?a=Home&refreshrequest=HYOALJYP
SMS Gateway - 192.168.114.253
admin
File
Edit
Service provider connections
Users and applications
View
Help
OZEKI
www.myozeki.com
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Start
SMS Gateway
8:07

Connections — Add

No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

Please install a connection!

Incoming

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defin_POP3_client_1 | Any_SMS_Connection | POP3_client_1 | Copy |
| 2. | defin_admin | Any_SMS_Connection | admin | Copy |

Add new route...

2 routes installed.

Outgoing

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move |

Add new route...

1 route installed.

Users and applications — Add

● admin (Admin) 0.00 MPM
Ready

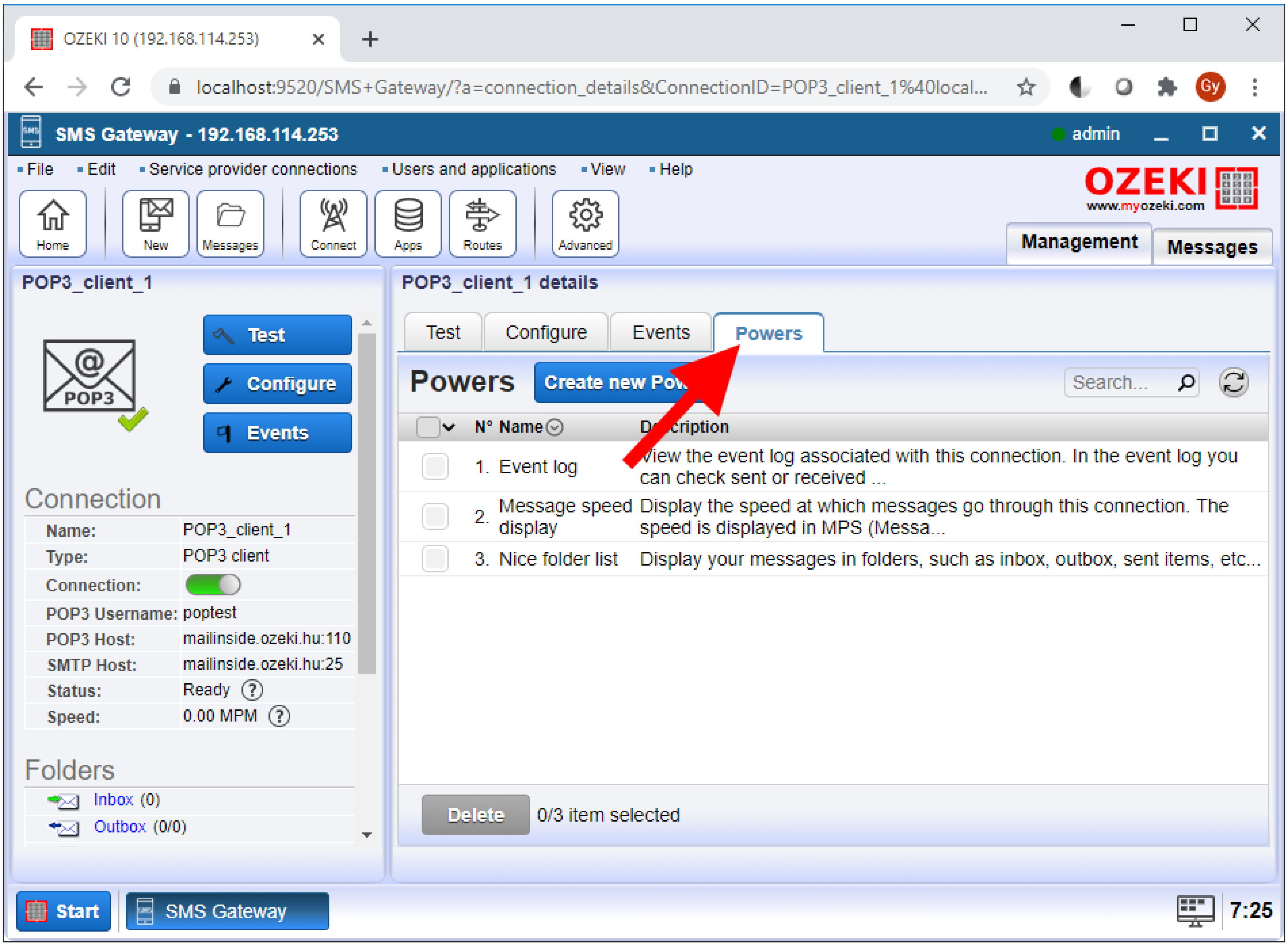
● POP3_client_1 (POP3 client) 0.00 MPM
Ready

Add new user/application...

3 users/applications installed

Figure 1 - Open the POP3 connection

## Step 2 - Select the powers tab of the e-mail connection



POP3_client_1

Connection

| | |
|---|---|
| Name: | POP3_client_1 |
| Type: | POP3 client |
| Connection: | |
| POP3 Username: | poptest |
| POP3 Host: | mailinside.ozeki.hu:110 |
| SMTP Host: | mailinside.ozeki.hu:25 |
| Status: | Ready |
| Speed: | 0.00 MPM |

Folders

Inbox (0)

Outbox (0/0)

POP3_client_1 details

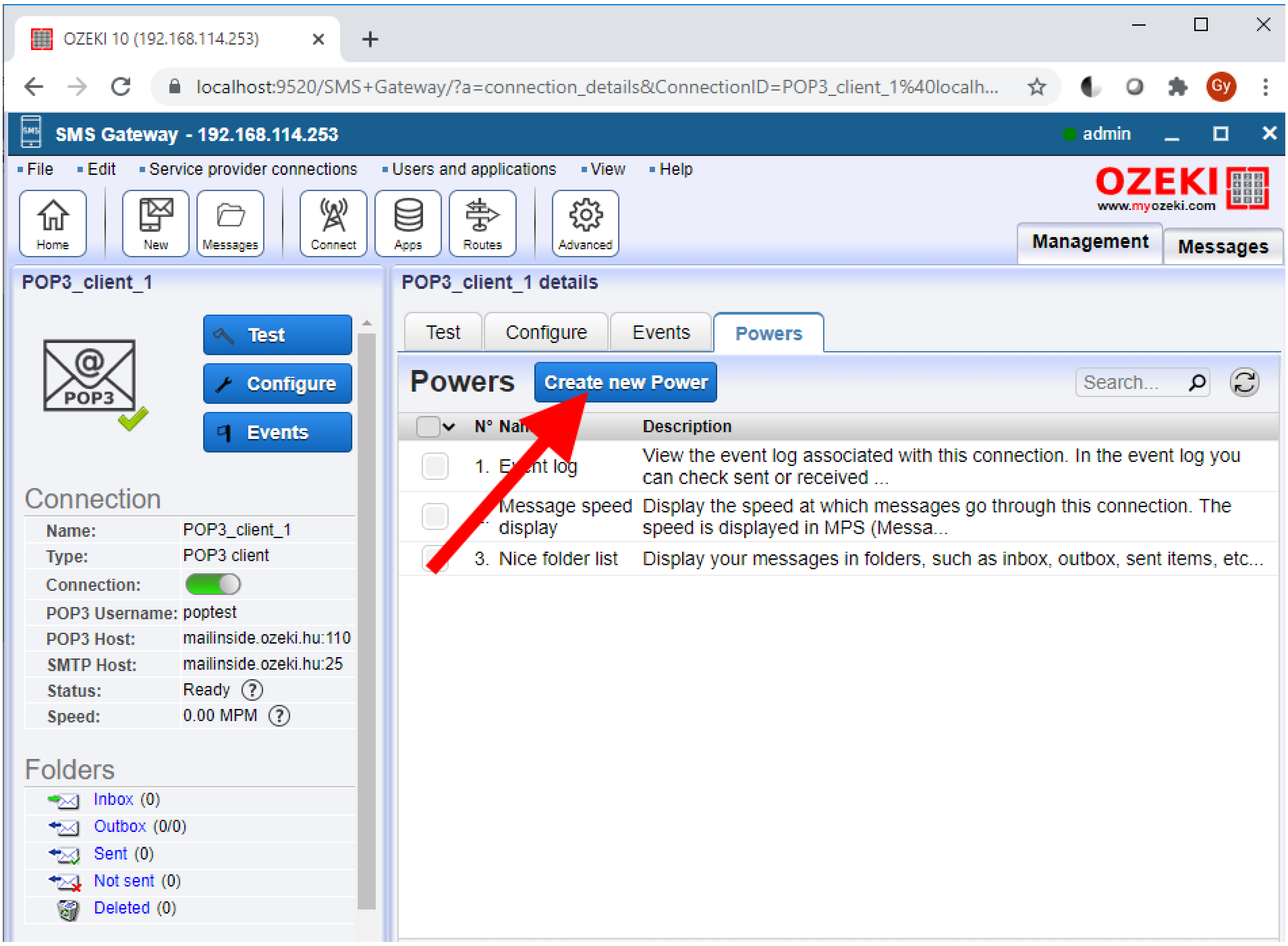
Powers

| N° | Name | Description |
|---|---|---|
| 1. | Event log | View the event log associated with this connection. In the event log you can check sent or received ... |
| 2. | Message speed display | Display the speed at which messages go through this connection. The speed is displayed in MPS (Messa... |
| 3. | Nice folder list | Display your messages in folders, such as inbox, outbox, sent items, etc... |

0/3 item selected

Figure 2 - Select the powers tab

## Step 3 - Click on create new power


OZEKI 10 (192.168.114.253)
localhost:9520/SMS+Gateway/?a=connection_details&ConnectionID=POP3_client_1%40localh...
SMS Gateway - 192.168.114.253
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Test
Configure
Events
Powers
Create new Power
Search...

POP3_client_1

Connection

| | |
|---|---|
| Name: | POP3_client_1 |
| Type: | POP3 client |
| Connection: | |
| POP3 Username: | poptest |
| POP3 Host: | mailinside.ozeki.hu:110 |
| SMTP Host: | mailinside.ozeki.hu:25 |
| Status: | Ready |
| Speed: | 0.00 MPM |

Folders

- Inbox (0)
- Outbox (0/0)
- Sent (0)
- Not sent (0)
- Deleted (0)

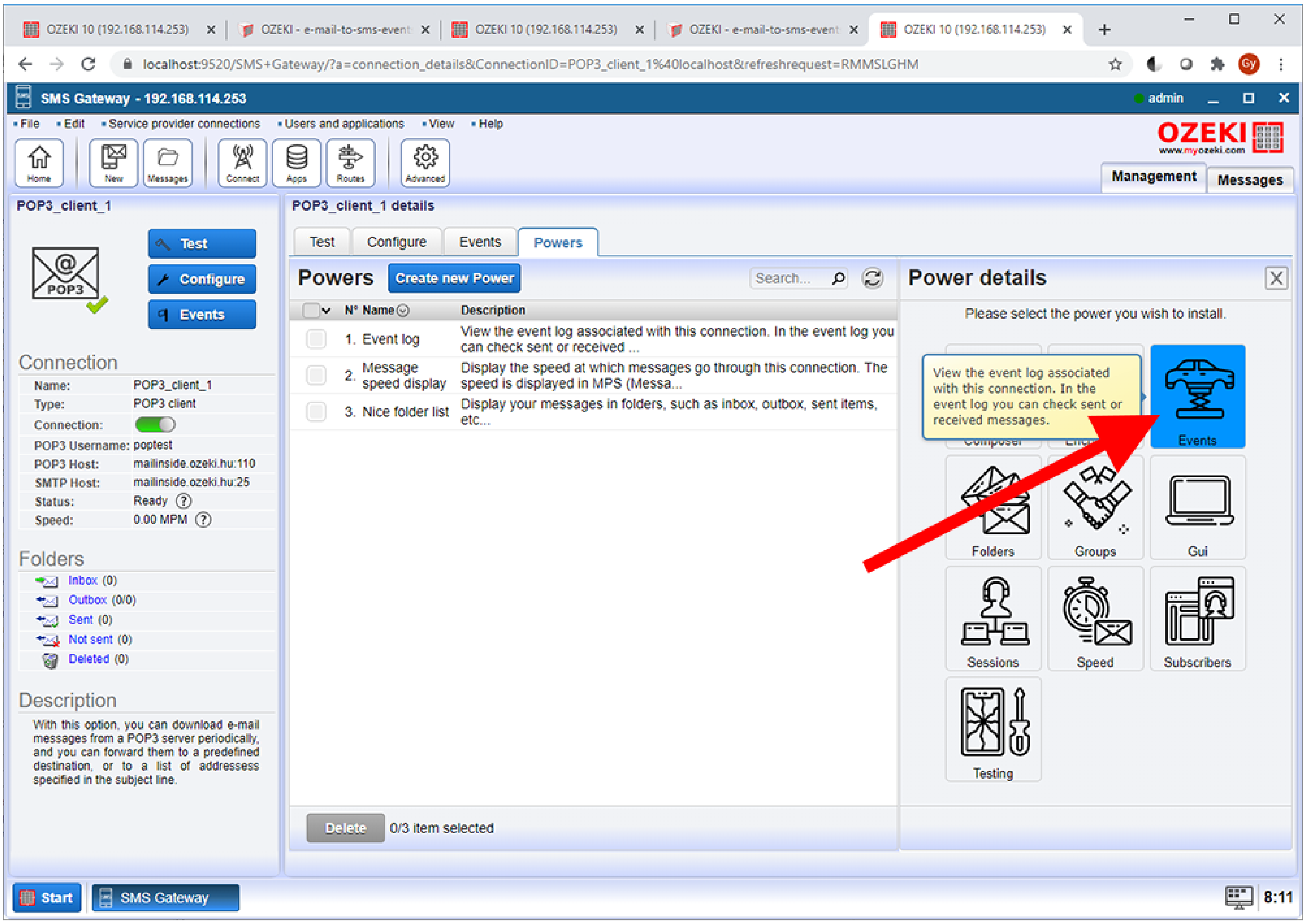
POP3_client_1 details

Powers

| N° | Name | Description |
|---|---|---|
| 1. | Event log | View the event log associated with this connection. In the event log you can check sent or received ... |
| 2. | Message speed display | Display the speed at which messages go through this connection. The speed is displayed in MPS (Messa... |
| 3. | Nice folder list | Display your messages in folders, such as inbox, outbox, sent items, etc... |

Figure 3 - Click on Create new power

## Step 4 - Select events



localhost:9520/SMS+Gateway/?a=connection_details&ConnectionID=POP3_client_1%40localhost&refreshrequest=RMMSLGHM

**POP3_client_1 details** — Test | Configure | Events | Powers

**Powers** — Create new Power

| N° | Name | Description |
|---|---|---|
| 1. | Event log | View the event log associated with this connection. In the event log you can check sent or received ... |
| 2. | Message speed display | Display the speed at which messages go through this connection. The speed is displayed in MPS (Messa... |
| 3. | Nice folder list | Display your messages in folders, such as inbox, outbox, sent items, etc... |

Delete 0/3 item selected

**Power details**

Please select the power you wish to install.

View the event log associated with this connection. In the event log you can check sent or received messages.

**Connection**

| | |
|---|---|
| Name: | POP3_client_1 |
| Type: | POP3 client |
| Connection: | |
| POP3 Username: | poptest |
| POP3 Host: | mailinside.ozeki.hu:110 |
| SMTP Host: | mailinside.ozeki.hu:25 |
| Status: | Ready |
| Speed: | 0.00 MPM |

**Folders**

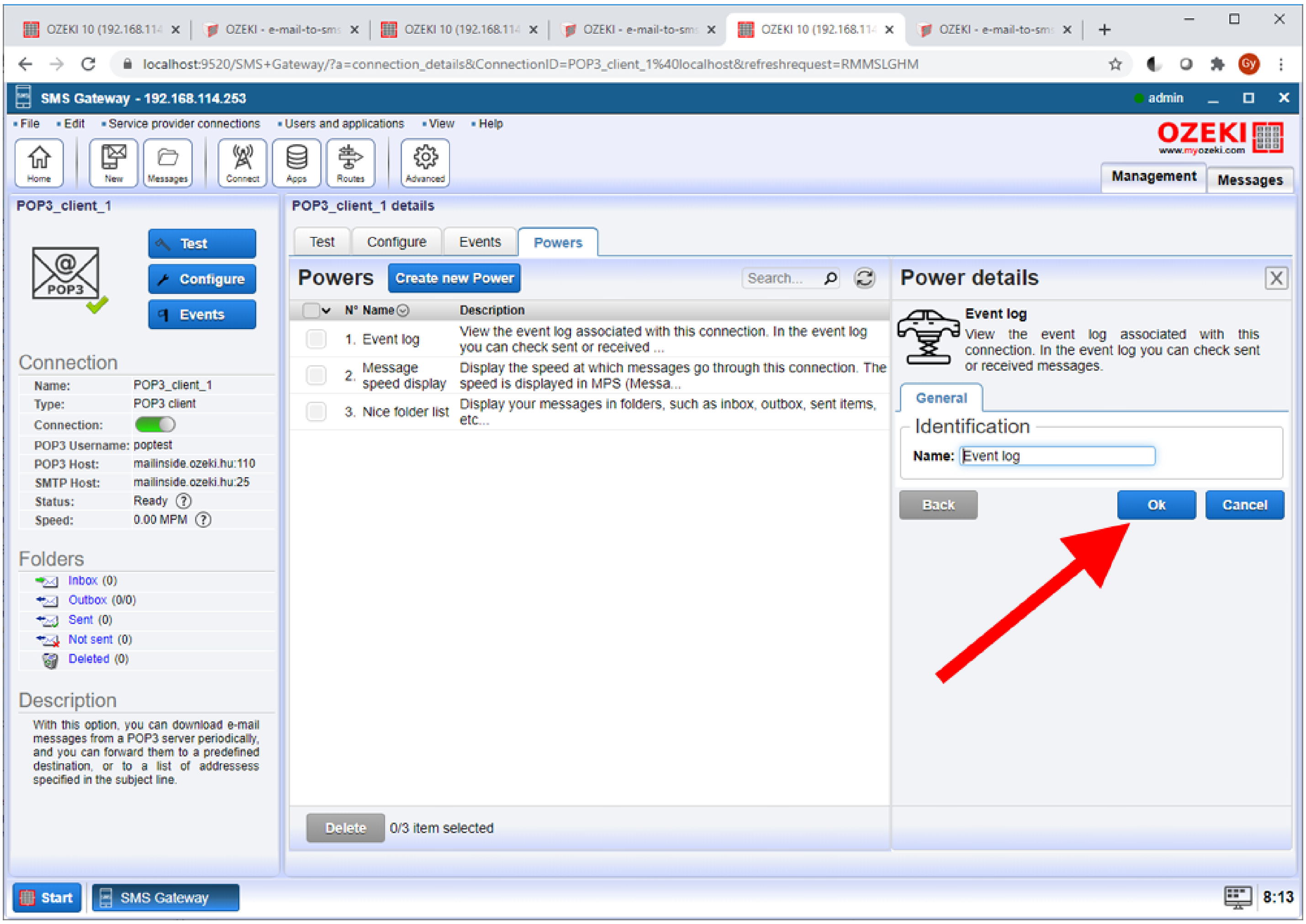
- Inbox (0)
- Outbox (0/0)
- Sent (0)
- Not sent (0)
- Deleted (0)

**Description**

With this option, you can download e-mail messages from a POP3 server periodically, and you can forward them to a predefined destination, or to a list of addresses specified in the subject line.

Figure 4 - Select events

# Step 5 - Approve your selection



POP3_client_1 details

Test | Configure | Events | Powers

Powers — Create new Power

| N° | Name | Description |
|---|---|---|
| 1. | Event log | View the event log associated with this connection. In the event log you can check sent or received ... |
| 2. | Message speed display | Display the speed at which messages go through this connection. The speed is displayed in MPS (Messa... |
| 3. | Nice folder list | Display your messages in folders, such as inbox, outbox, sent items, etc... |

Delete 0/3 item selected

Connection

| | |
|---|---|
| Name: | POP3_client_1 |
| Type: | POP3 client |
| Connection: | |
| POP3 Username: | poptest |
| POP3 Host: | mailinside.ozeki.hu:110 |
| SMTP Host: | mailinside.ozeki.hu:25 |
| Status: | Ready |
| Speed: | 0.00 MPM |

Folders

- Inbox (0)
- Outbox (0/0)
- Sent (0)
- Not sent (0)
- Deleted (0)

Description

With this option, you can download e-mail messages from a POP3 server periodically, and you can forward them to a predefined destination, or to a list of addresses specified in the subject line.

Power details

Event log
View the event log associated with this connection. In the event log you can check sent or received messages.

General

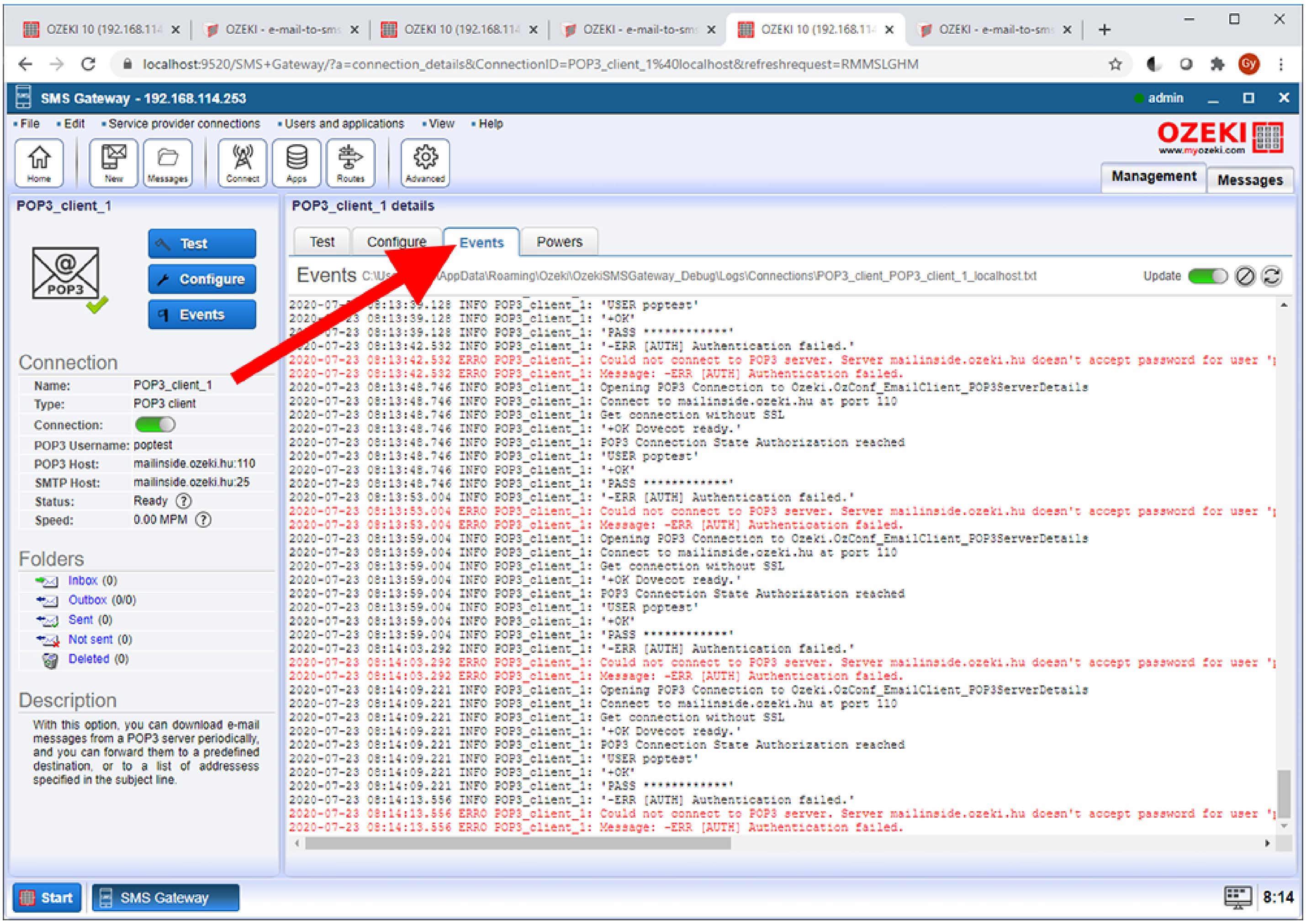
Identification

Name: Event log

Back | Ok | Cancel

Figure 5 - Click OK

# Step 6 - View the event log



Events C:\Us \AppData\Roaming\Ozeki\OzekiSMSGateway_Debug\Logs\Connections\POP3_client_POP3_client_1_localhost.txt

Connection

Name: POP3_client_1
Type: POP3 client
Connection:
POP3 Username: poptest
POP3 Host: mailinside.ozeki.hu:110
SMTP Host: mailinside.ozeki.hu:25
Status: Ready
Speed: 0.00 MPM

Folders

Inbox (0)
Outbox (0/0)
Sent (0)
Not sent (0)
Deleted (0)

Description

With this option, you can download e-mail messages from a POP3 server periodically, and you can forward them to a predefined destination, or to a list of addresses specified in the subject line.

```
2020-07-23 08:13:39.128 INFO POP3_client_1: 'USER poptest'
2020-07-23 08:13:39.128 INFO POP3_client_1: '+OK'
2020-07-23 08:13:39.128 INFO POP3_client_1: 'PASS ************'
2020-07-23 08:13:42.532 INFO POP3_client_1: '-ERR [AUTH] Authentication failed.'
2020-07-23 08:13:42.532 ERRO POP3_client_1: Could not connect to POP3 server. Server mailinside.ozeki.hu doesn't accept password for user ';
2020-07-23 08:13:42.532 ERRO POP3_client_1: Message: -ERR [AUTH] Authentication failed.
2020-07-23 08:13:48.746 INFO POP3_client_1: Opening POP3 Connection to Ozeki.OzConf_EmailClient_POP3ServerDetails
2020-07-23 08:13:48.746 INFO POP3_client_1: Connect to mailinside.ozeki.hu at port 110
2020-07-23 08:13:48.746 INFO POP3_client_1: Get connection without SSL
2020-07-23 08:13:48.746 INFO POP3_client_1: '+OK Dovecot ready.'
2020-07-23 08:13:48.746 INFO POP3_client_1: POP3 Connection State Authorization reached
2020-07-23 08:13:48.746 INFO POP3_client_1: 'USER poptest'
2020-07-23 08:13:48.746 INFO POP3_client_1: '+OK'
2020-07-23 08:13:48.746 INFO POP3_client_1: 'PASS ************'
2020-07-23 08:13:53.004 INFO POP3_client_1: '-ERR [AUTH] Authentication failed.'
2020-07-23 08:13:53.004 ERRO POP3_client_1: Could not connect to POP3 server. Server mailinside.ozeki.hu doesn't accept password for user ';
2020-07-23 08:13:53.004 ERRO POP3_client_1: Message: -ERR [AUTH] Authentication failed.
2020-07-23 08:13:59.004 INFO POP3_client_1: Opening POP3 Connection to Ozeki.OzConf_EmailClient_POP3ServerDetails
2020-07-23 08:13:59.004 INFO POP3_client_1: Connect to mailinside.ozeki.hu at port 110
2020-07-23 08:13:59.004 INFO POP3_client_1: Get connection without SSL
2020-07-23 08:13:59.004 INFO POP3_client_1: '+OK Dovecot ready.'
2020-07-23 08:13:59.004 INFO POP3_client_1: POP3 Connection State Authorization reached
2020-07-23 08:13:59.004 INFO POP3_client_1: 'USER poptest'
2020-07-23 08:13:59.004 INFO POP3_client_1: '+OK'
2020-07-23 08:13:59.004 INFO POP3_client_1: 'PASS ************'
2020-07-23 08:14:03.292 INFO POP3_client_1: '-ERR [AUTH] Authentication failed.'
2020-07-23 08:14:03.292 ERRO POP3_client_1: Could not connect to POP3 server. Server mailinside.ozeki.hu doesn't accept password for user ';
2020-07-23 08:14:03.292 ERRO POP3_client_1: Message: -ERR [AUTH] Authentication failed.
2020-07-23 08:14:09.221 INFO POP3_client_1: Opening POP3 Connection to Ozeki.OzConf_EmailClient_POP3ServerDetails
2020-07-23 08:14:09.221 INFO POP3_client_1: Connect to mailinside.ozeki.hu at port 110
2020-07-23 08:14:09.221 INFO POP3_client_1: Get connection without SSL
2020-07-23 08:14:09.221 INFO POP3_client_1: '+OK Dovecot ready.'
2020-07-23 08:14:09.221 INFO POP3_client_1: POP3 Connection State Authorization reached
2020-07-23 08:14:09.221 INFO POP3_client_1: 'USER poptest'
2020-07-23 08:14:09.221 INFO POP3_client_1: '+OK'
2020-07-23 08:14:09.221 INFO POP3_client_1: 'PASS ************'
2020-07-23 08:14:13.556 INFO POP3_client_1: '-ERR [AUTH] Authentication failed.'
2020-07-23 08:14:13.556 ERRO POP3_client_1: Could not connect to POP3 server. Server mailinside.ozeki.hu doesn't accept password for user ';
2020-07-23 08:14:13.556 ERRO POP3_client_1: Message: -ERR [AUTH] Authentication failed.
```

Figure 6 - Select events tab

# How to setup the built-in SMTP server

***This guide explains how you can setup Ozeki SMS Gateway's built-in SMTP server. This configuration will allow you to setup e-mail to SMS forwarding. Ozeki SMS Gateway's SMTP server will accepts your e-mail messages and will use the phone number in the e-mail address to send the message. For example if you use +441234657@smsgw.mycompany.com as the recipient e-mail address, the SMS will be sent to +441234567. The SMS text will be the e-mail subject (or optionally the e-mail body).***

## Introduction

You can setup e-mail to SMS forwarding by configuring Ozeki SMS Gateway's built-in SMTP server. After this configuration Ozeki SMS Gateway will listen on port 25 (the standard STMP port), and will accept incoming SMTP (e-mail submit) requests. To get the best result it is recommended to setup a subdomain name entry (MX record) in your DNS system that points to your SMS gateway. For example if your SMS gateway operates at IP address 192.168.0.10, you would create an MX record called smsgw.mycompany.com, that would point to 192.168.0.10. After creating this entry an e-mail sent to +44123467@smsgw.mycompany.com would be automatically forwarded to the smsgw.

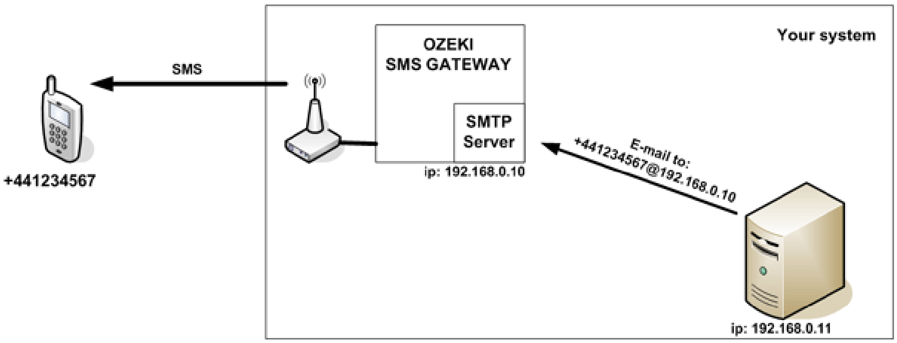


Figure 1 - Forwarding E-mail directly as an SMS alert

Keep in mind that the built-in SMTP server of Ozeki SMS Gateway can only be used for sms to e-mail forwarding. It cannot be used for e-mail to e-mail forwarding. It will NOT forward e-mail messages to other Email servers.

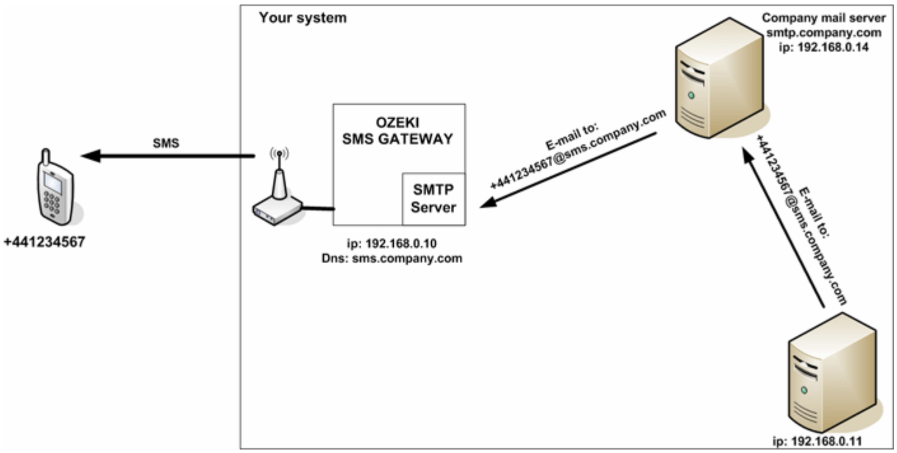


Figure 2 - An email forwarded through a company's mail server to Ozeki SMS Gateway

## Video tutorial

The following video tutorial shows the steps to take to setup the built in SMTP server of Ozeki 10 SMS gateway. These steps are also explained in the section following the video. Please follow the steps accuretly to get your system up and running.

## Step 1 - Create the SMTP service

The creation of on SMTP service in the SMS Gateway is super easy, it takes just a few clicks. First, open the SMS Gateway application, and on the toolbar, click on the icon of the Advanced option. Here, as you can see it in Figure 2, just click on the Create new Service button.

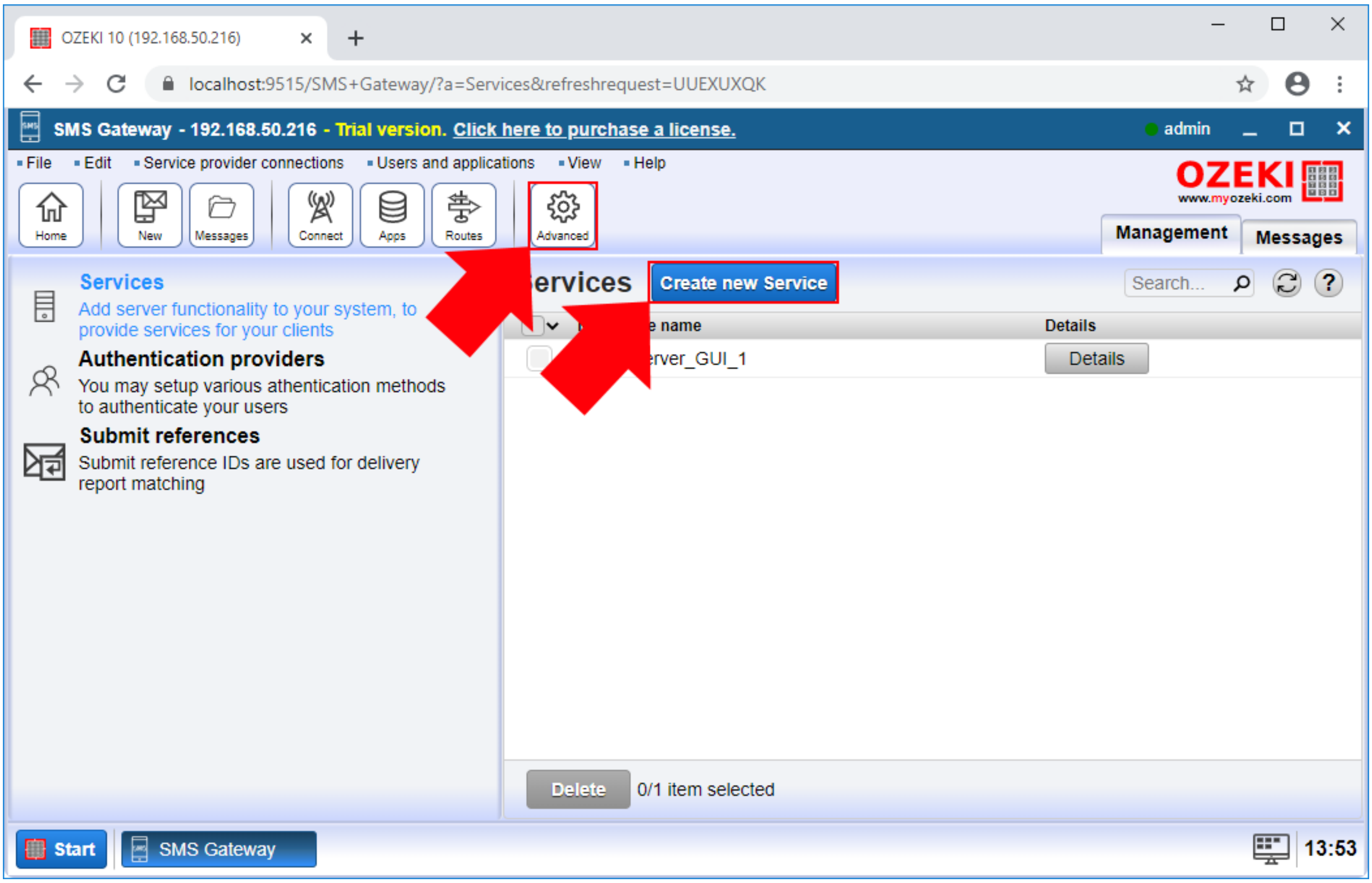

OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=Services&refreshrequest=UUEXUXQK
SMS Gateway - 192.168.50.216 - Trial version. Click here to purchase a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
OZEKI
www.myozeki.com
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Services
Add server functionality to your system, to provide services for your clients
Authentication providers
You may setup various athentication methods to authenticate your users
Submit references
Submit reference IDs are used for delivery report matching
ervices
Create new Service
Search...
e name
Details
rver_GUI_1
Details
Delete
0/1 item selected
Start
SMS Gateway
13:53

Figure 3 - Create a new SMTP service in SMS Gateway

Next, you need to select the E-mail option from the list of avalable service types. After you selected the E-mail option, like Figure 3 demonstrates it, you need to click on the SMTP service type.

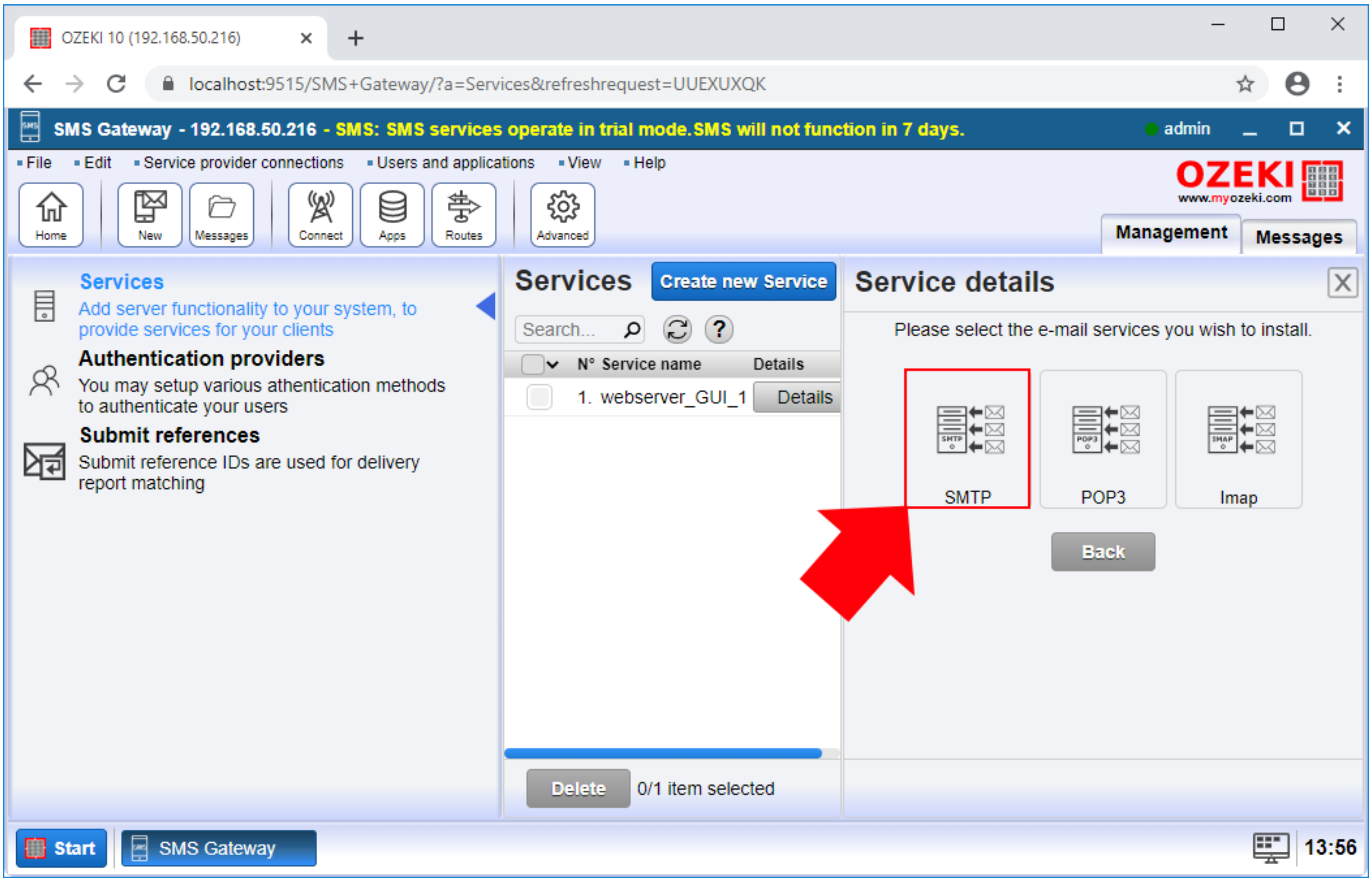
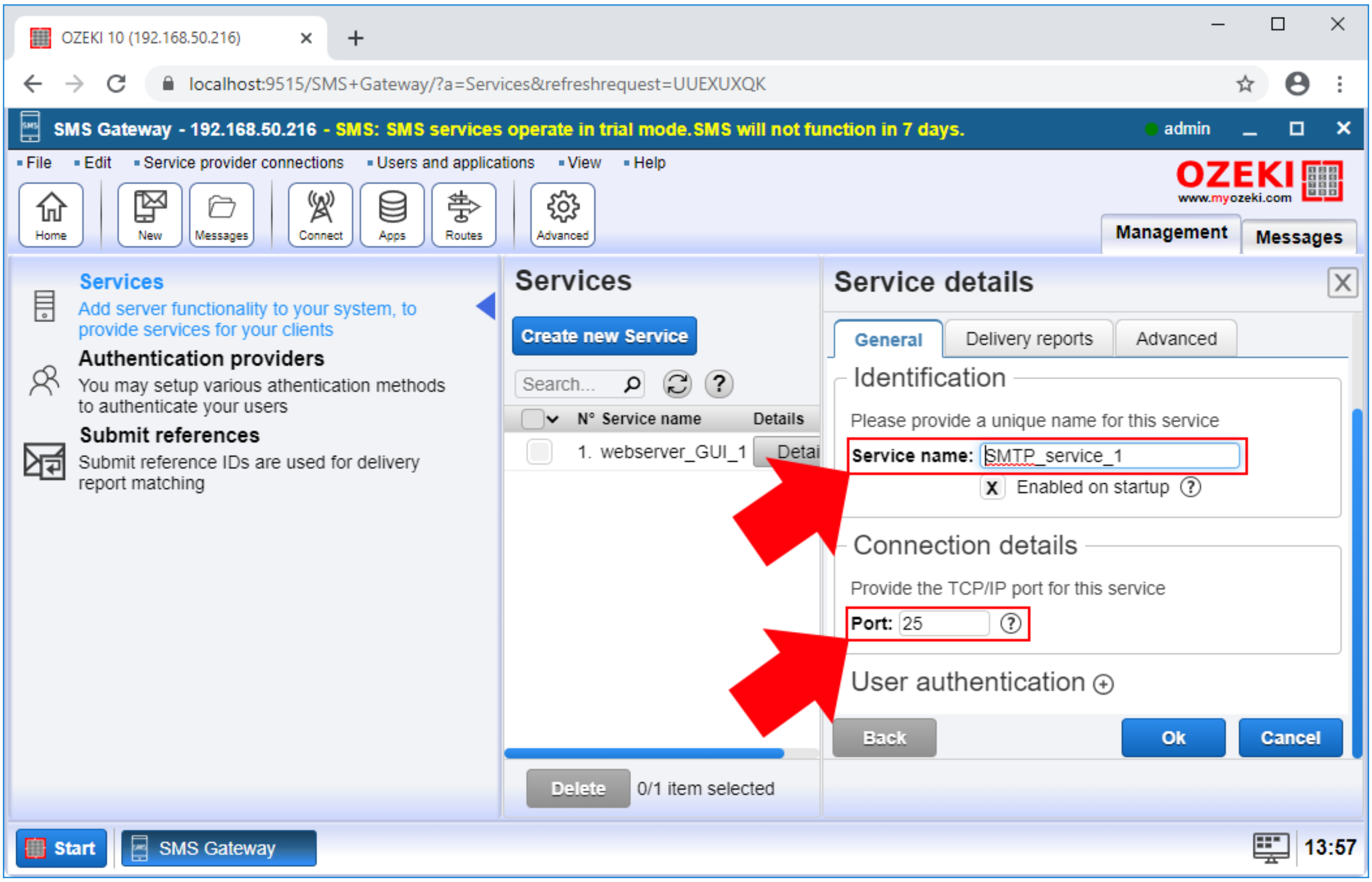

OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=Services&refreshrequest=UUEXUXQK
SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Services
Add server functionality to your system, to provide services for your clients
Authentication providers
You may setup various athentication methods to authenticate your users
Submit references
Submit reference IDs are used for delivery report matching
Services
Create new Service
Search...
N° Service name
Details
1. webserver_GUI_1
Details
Delete
0/1 item selected
Service details
Please select the e-mail services you wish to install.
SMTP
POP3
Imap
Back
Start
SMS Gateway
13:56

Figure 4 - Select the SMTP service type

# Step 2 - Configure the SMTP service

Here, in the configuration menu, you need to provide a name for the service. This name will be used by the SMS Gateway to identify the connection. The next thing, that you need to do here is to specify a port number for the SMTP service (Figure 4). This is 25 by default, and you can leave it there.


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=Services&refreshrequest=UUEXUXQK
SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
OZEKI
www.myozeki.com
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Start
SMS Gateway
13:57

**Services**

Add server functionality to your system, to provide services for your clients

**Authentication providers**

You may setup various athentication methods to authenticate your users

**Submit references**

Submit reference IDs are used for delivery report matching

**Services**

Create new Service

Search...

| N° | Service name | Details |
|---|---|---|
| 1. | webserver_GUI_1 | Detai |

Delete 0/1 item selected

**Service details**

General | Delivery reports | Advanced

Identification

Please provide a unique name for this service

Service name: SMTP_service_1

☒ Enabled on startup

Connection details

Provide the TCP/IP port for this service

Port: 25

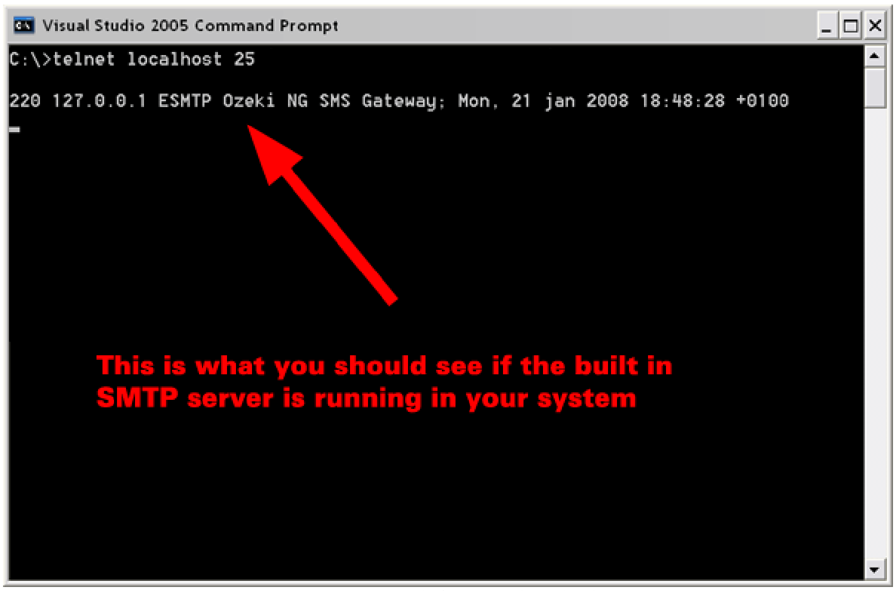
User authentication ⊕

Back | Ok | Cancel

Figure 5 - Provide details for the SMTP service

## Step 3 - Test your SMTP server

It is a good idea to check if your SMTP server is up and running. **Test your SMTP server with the Telnet application of Windows.** Keep in mind that you should manually activate Telnet in Windows 10 ('Turn Windows features on or off'). Please telnet to port 25 of the computer running Ozeki SMS Gateway. If the SMTP server is running on port 25, the SMS Gateway software should greet you (Figure 6).



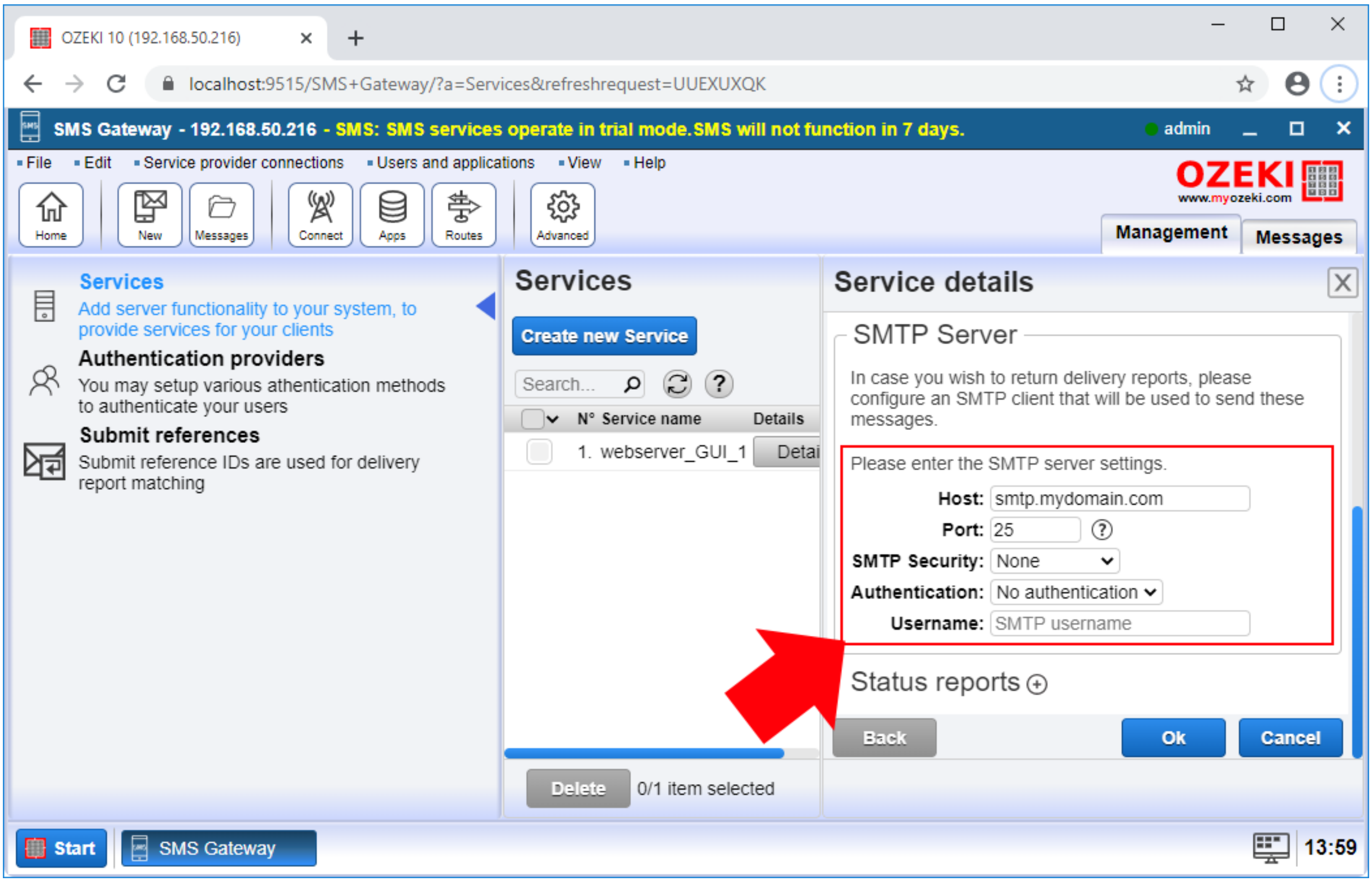
```
C:\>telnet localhost 25

220 127.0.0.1 ESMTP Ozeki NG SMS Gateway; Mon, 21 jan 2008 18:48:28 +0100
_
```

This is what you should see if the built in SMTP server is running in your system

**Figure 6** - Checking the status of Ozeki SMS Gateway's SMTP server

## Step 4 - Advanced configuration of STMP service

The Delivery Report tab contains some other setting that you can modify as well. As you can see it in Figure 7, you can modify the details of the SMTP server. Here, you can set the name of the Host, modify the SMTP security and authentication details.


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=Services&refreshrequest=UUEXUXQK
SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Services
Add server functionality to your system, to provide services for your clients
Authentication providers
You may setup various athentication methods to authenticate your users
Submit references
Submit reference IDs are used for delivery report matching
Services
Create new Service
Search...
N° Service name
Details
1. webserver_GUI_1
Detai
Delete
0/1 item selected
Service details
Status reports
Back
Ok
Cancel
Start
SMS Gateway
13:59

SMTP Server

In case you wish to return delivery reports, please configure an SMTP client that will be used to send these messages.

Please enter the SMTP server settings.

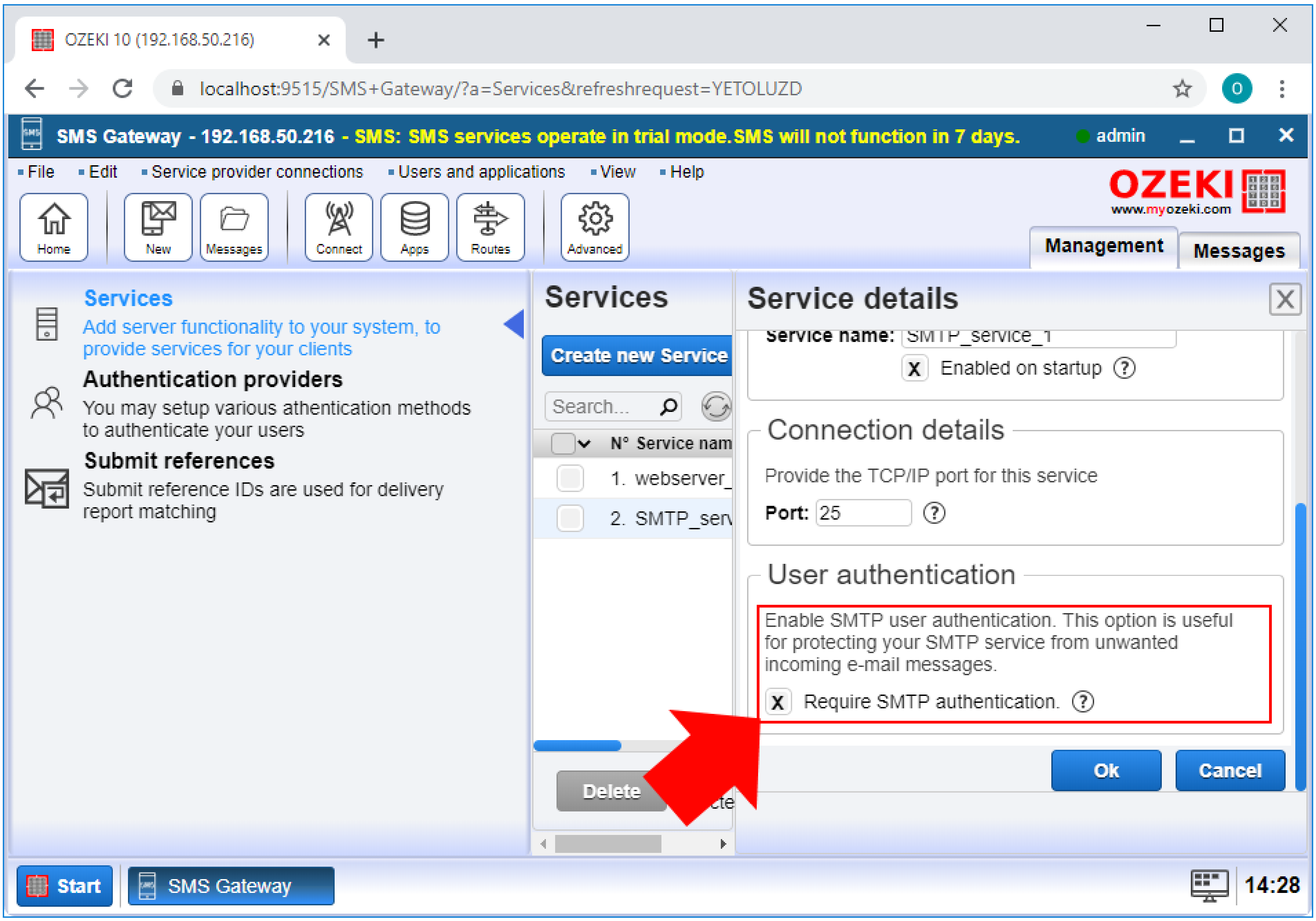
Host: smtp.mydomain.com
Port: 25
SMTP Security: None
Authentication: No authentication
Username: SMTP username

Figure 7 - Advanced configuration of the SMTP service

# Step 5 - Configure SMTP authentication

This step is optional.

SMTP server authentication provides a safer connection where you can connect with the username and password. To do this, you need to setup a standard user account in Ozeki 10 SMS Gateawy and use the username and password of this user account to connect. After the standard user account was created, you can turn on the SMTP authentication provider (Figure 8). Remember, you will have to configure SMTP authentication in your e-mail client as well.



SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.

**Services**
Add server functionality to your system, to provide services for your clients

**Authentication providers**
You may setup various athentication methods to authenticate your users

**Submit references**
Submit reference IDs are used for delivery report matching

**Service details**

Service name: SMTP_service_1
☒ Enabled on startup ?

Connection details
Provide the TCP/IP port for this service
Port: 25 ?

User authentication
Enable SMTP user authentication. This option is useful for protecting your SMTP service from unwanted incoming e-mail messages.
☒ Require SMTP authentication. ?

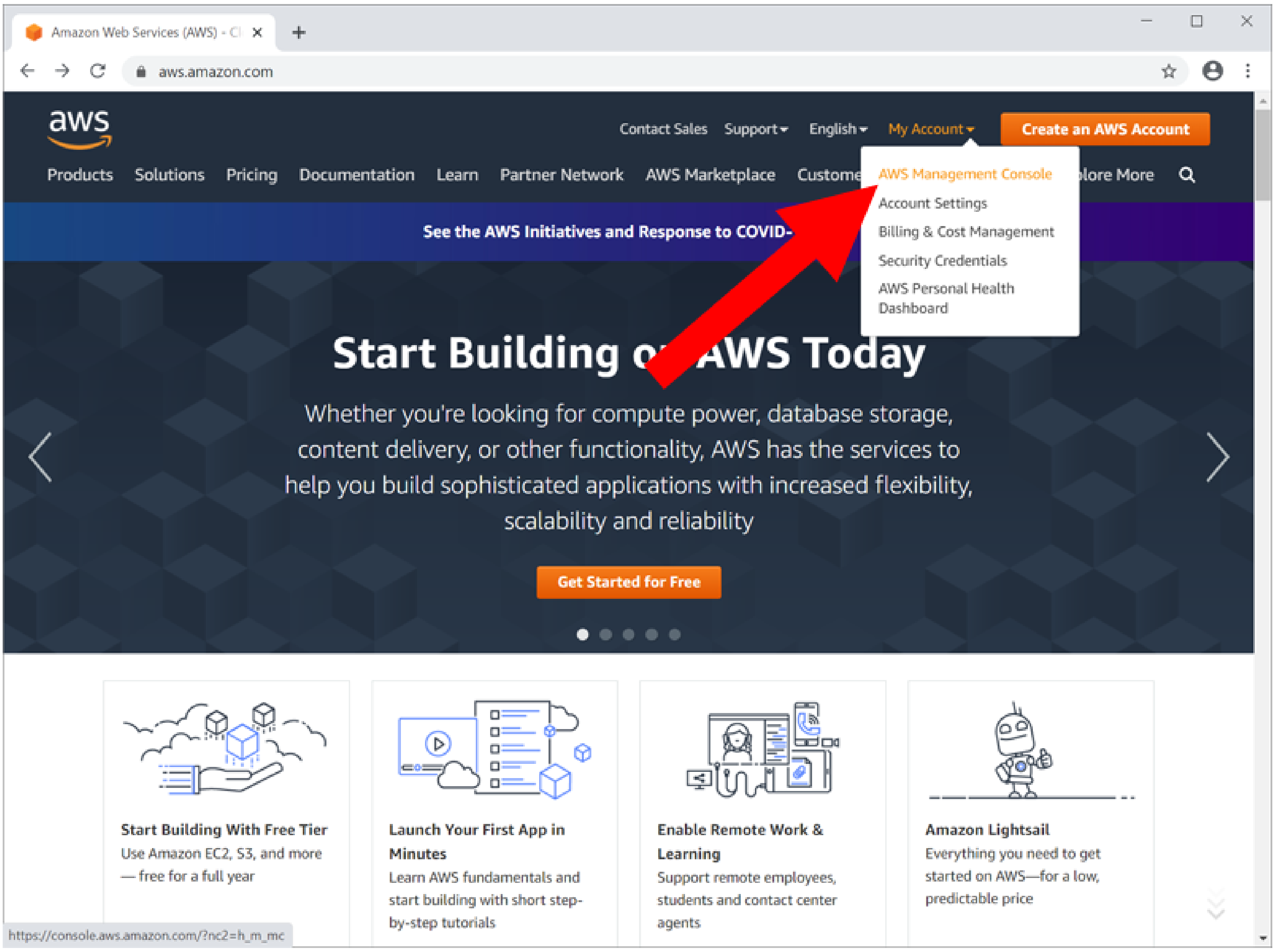
Figure 8 - Activating SMTP server authentication

# Email to SMS DNS setup

***This guide gives you instructions on how to setup an e-mail to SMS gateway that allows you to send SMS messages from the Internet by sending e-mails to a domain and using the username part of the e-mail as the recipient phone number. E.g. 0123456798@emailtosms.ozeki.hu.***

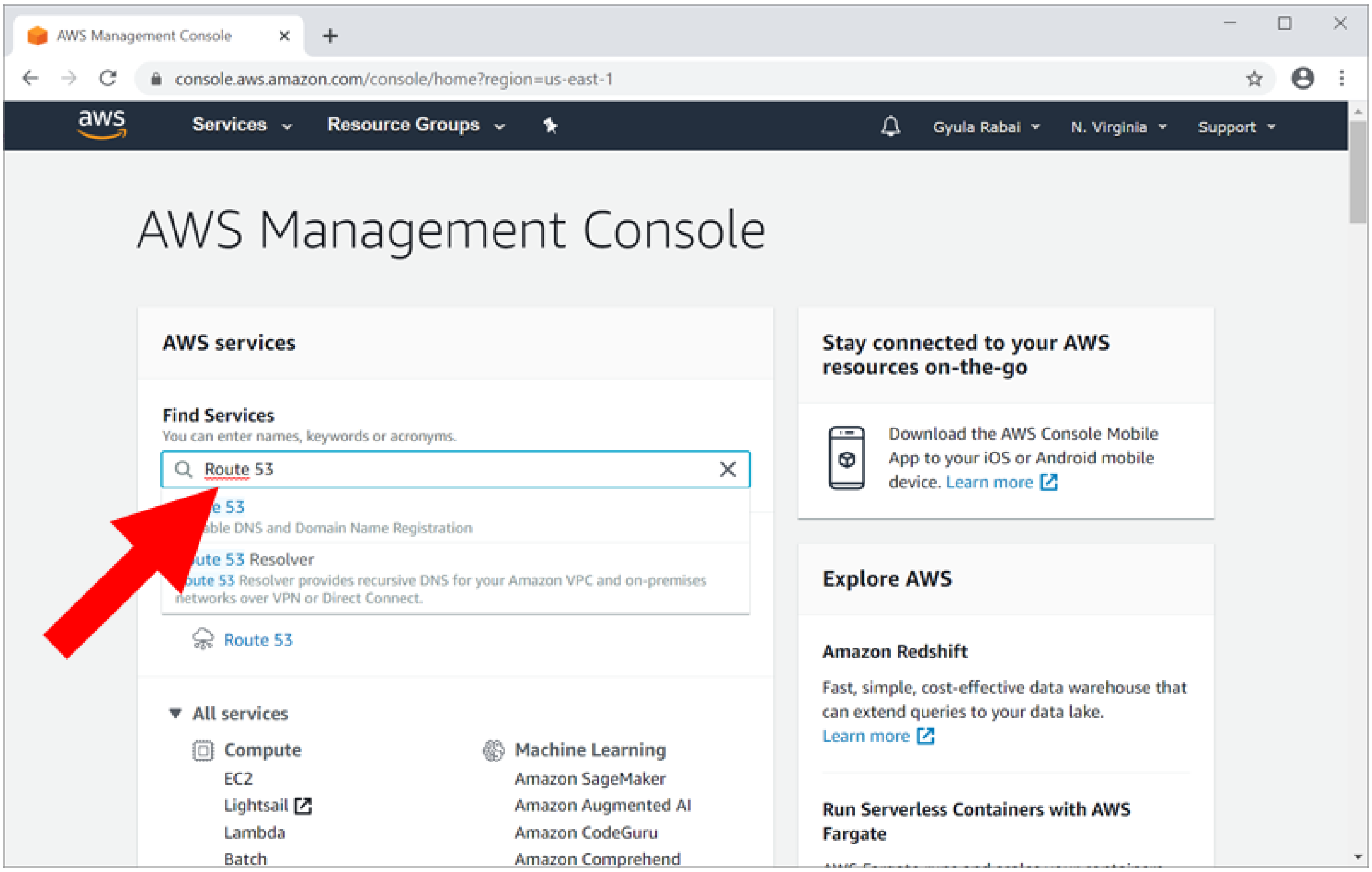
## Step 1: Create a DNS zone called emailtosms.ozeki.hu.

In order to accept e-mails in the emailtosms.ozeki.hu subdomain, a new DNS zone should be created for this subdomain. In our example we use Amazon web services to create this subdomain. You can also use Amazon web services, even if you host your main domain at a different DNS provider.



Amazon Web Services (AWS) - Cl
aws.amazon.com
aws
Contact Sales
Support
English
My Account
Create an AWS Account
Products
Solutions
Pricing
Documentation
Learn
Partner Network
AWS Marketplace
Custome
AWS Management Console
Account Settings
Billing & Cost Management
Security Credentials
AWS Personal Health Dashboard
plore More
See the AWS Initiatives and Response to COVID-
Start Building on AWS Today
Whether you're looking for compute power, database storage, content delivery, or other functionality, AWS has the services to help you build sophisticated applications with increased flexibility, scalability and reliability
Get Started for Free
Start Building With Free Tier
Use Amazon EC2, S3, and more — free for a full year
Launch Your First App in Minutes
Learn AWS fundamentals and start building with short step-by-step tutorials
Enable Remote Work & Learning
Support remote employees, students and contact center agents
Amazon Lightsail
Everything you need to get started on AWS—for a low, predictable price
https://console.aws.amazon.com/?nc2=h_m_mc

Figure 1 - Open Amazon Web Services



AWS Management Console

console.aws.amazon.com/console/home?region=us-east-1

# AWS Management Console

**AWS services**

**Find Services**

You can enter names, keywords or acronyms.

Route 53

e 53
able DNS and Domain Name Registration

ute 53 Resolver
oute 53 Resolver provides recursive DNS for your Amazon VPC and on-premises networks over VPN or Direct Connect.

Route 53

All services

| Compute | Machine Learning |
|---|---|
| EC2 | Amazon SageMaker |
| Lightsail | Amazon Augmented AI |
| Lambda | Amazon CodeGuru |
| Batch | Amazon Comprehend |

**Stay connected to your AWS resources on-the-go**

Download the AWS Console Mobile App to your iOS or Android mobile device. Learn more

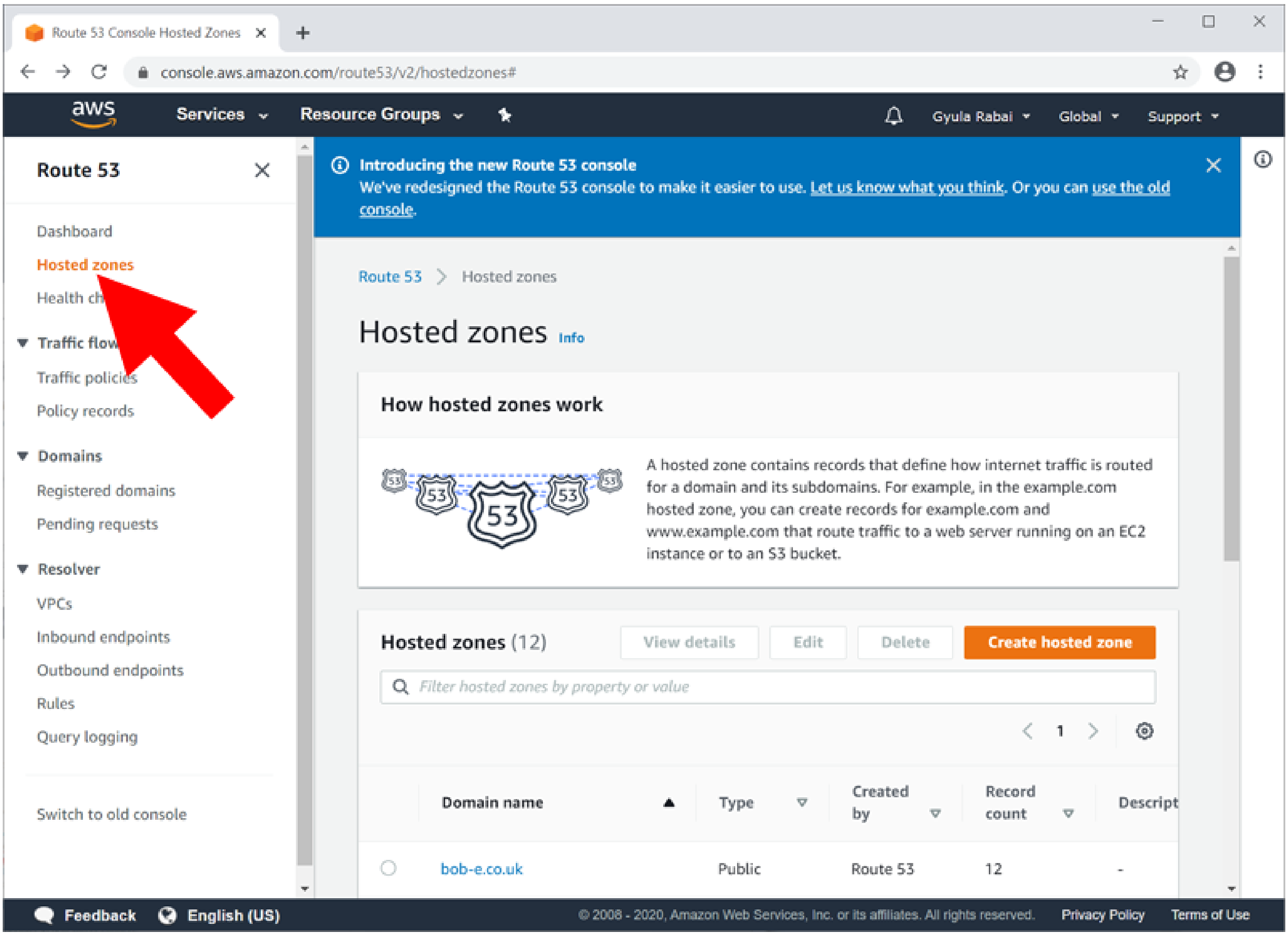
**Explore AWS**

**Amazon Redshift**

Fast, simple, cost-effective data warehouse that can extend queries to your data lake.
Learn more

**Run Serverless Containers with AWS Fargate**

Figure 2 - Select Route 53. This is the DNS service



Route 53 Console Hosted Zones

console.aws.amazon.com/route53/v2/hostedzones#

**Route 53**

- Dashboard
- Hosted zones
- Health ch
- Traffic flow
  - Traffic policies
  - Policy records
- Domains
  - Registered domains
  - Pending requests
- Resolver
  - VPCs
  - Inbound endpoints
  - Outbound endpoints
  - Rules
  - Query logging
- Switch to old console

**Introducing the new Route 53 console**
We've redesigned the Route 53 console to make it easier to use. Let us know what you think. Or you can use the old console.

Route 53 > Hosted zones

## Hosted zones Info

**How hosted zones work**

A hosted zone contains records that define how internet traffic is routed for a domain and its subdomains. For example, in the example.com hosted zone, you can create records for example.com and www.example.com that route traffic to a web server running on an EC2 instance or to an S3 bucket.

**Hosted zones** (12)

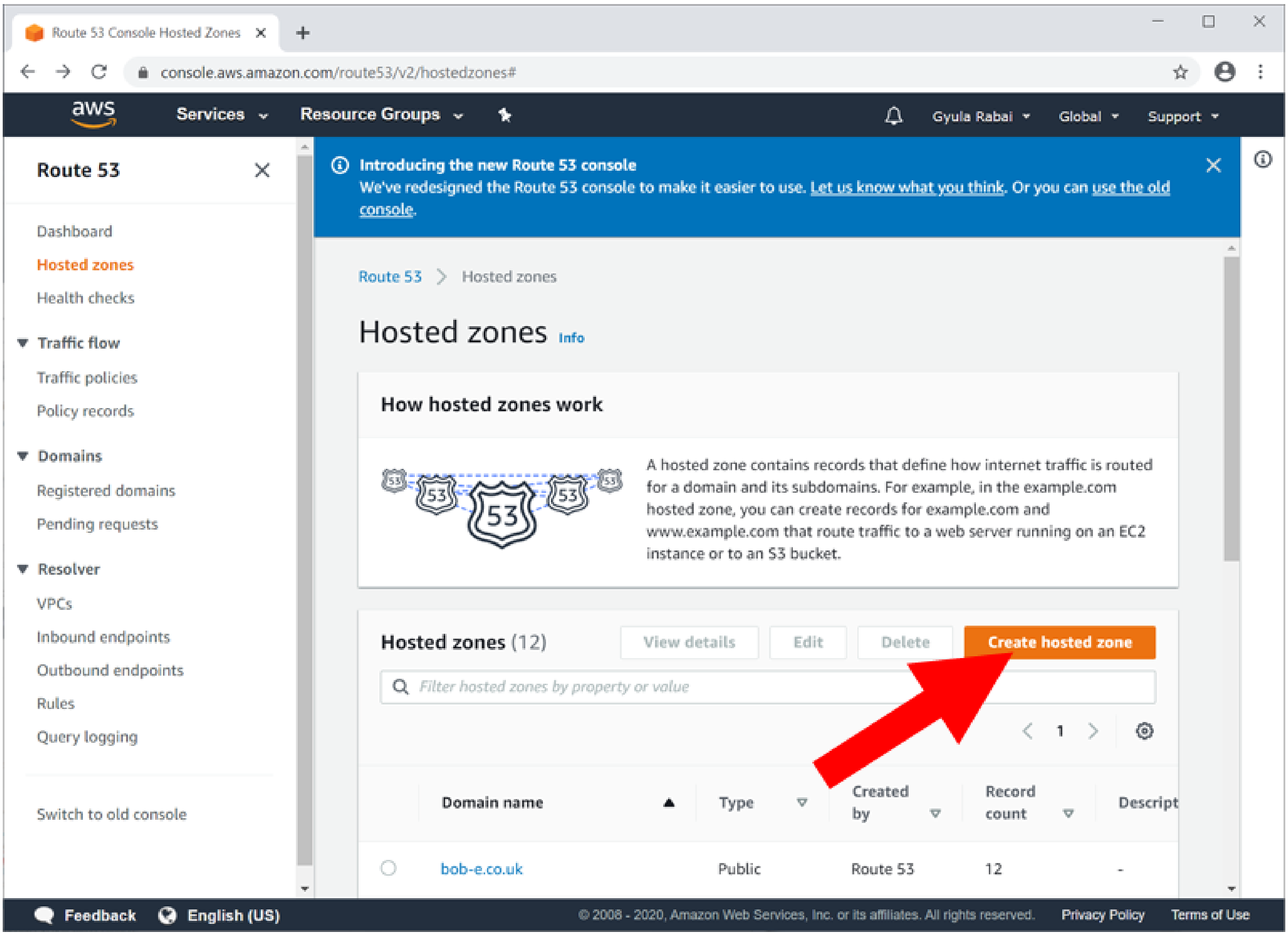
View details | Edit | Delete | Create hosted zone

Filter hosted zones by property or value

| | Domain name | Type | Created by | Record count | Descript |
|---|---|---|---|---|---|
| ○ | bob-e.co.uk | Public | Route 53 | 12 | - |

Feedback English (US) © 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

Figure 3 - Select Hosted Zones



**Introducing the new Route 53 console**

We've redesigned the Route 53 console to make it easier to use. Let us know what you think. Or you can use the old console.

Hosted zones

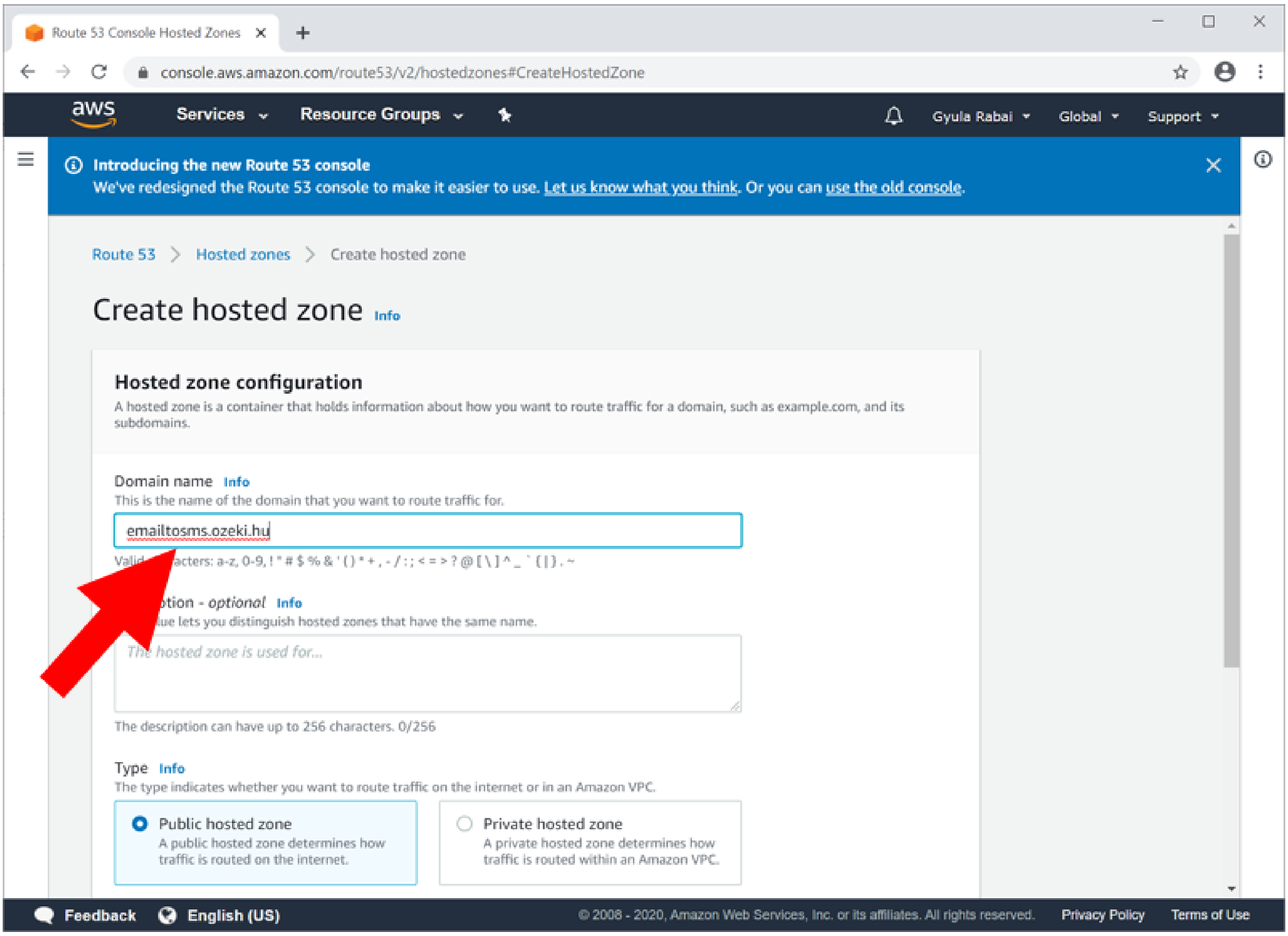
How hosted zones work

A hosted zone contains records that define how internet traffic is routed for a domain and its subdomains. For example, in the example.com hosted zone, you can create records for example.com and www.example.com that route traffic to a web server running on an EC2 instance or to an S3 bucket.

Hosted zones (12)

| Domain name | Type | Created by | Record count | Descript |
|---|---|---|---|---|
| ○ bob-e.co.uk | Public | Route 53 | 12 | - |

Figure 4 - Create Hosted Zone



Route 53 Console Hosted Zones
console.aws.amazon.com/route53/v2/hostedzones#CreateHostedZone
aws
Services
Resource Groups
Gyula Rabai
Global
Support

**Introducing the new Route 53 console**
We've redesigned the Route 53 console to make it easier to use. Let us know what you think. Or you can use the old console.

Route 53 > Hosted zones > Create hosted zone

Create hosted zone Info

Hosted zone configuration
A hosted zone is a container that holds information about how you want to route traffic for a domain, such as example.com, and its subdomains.

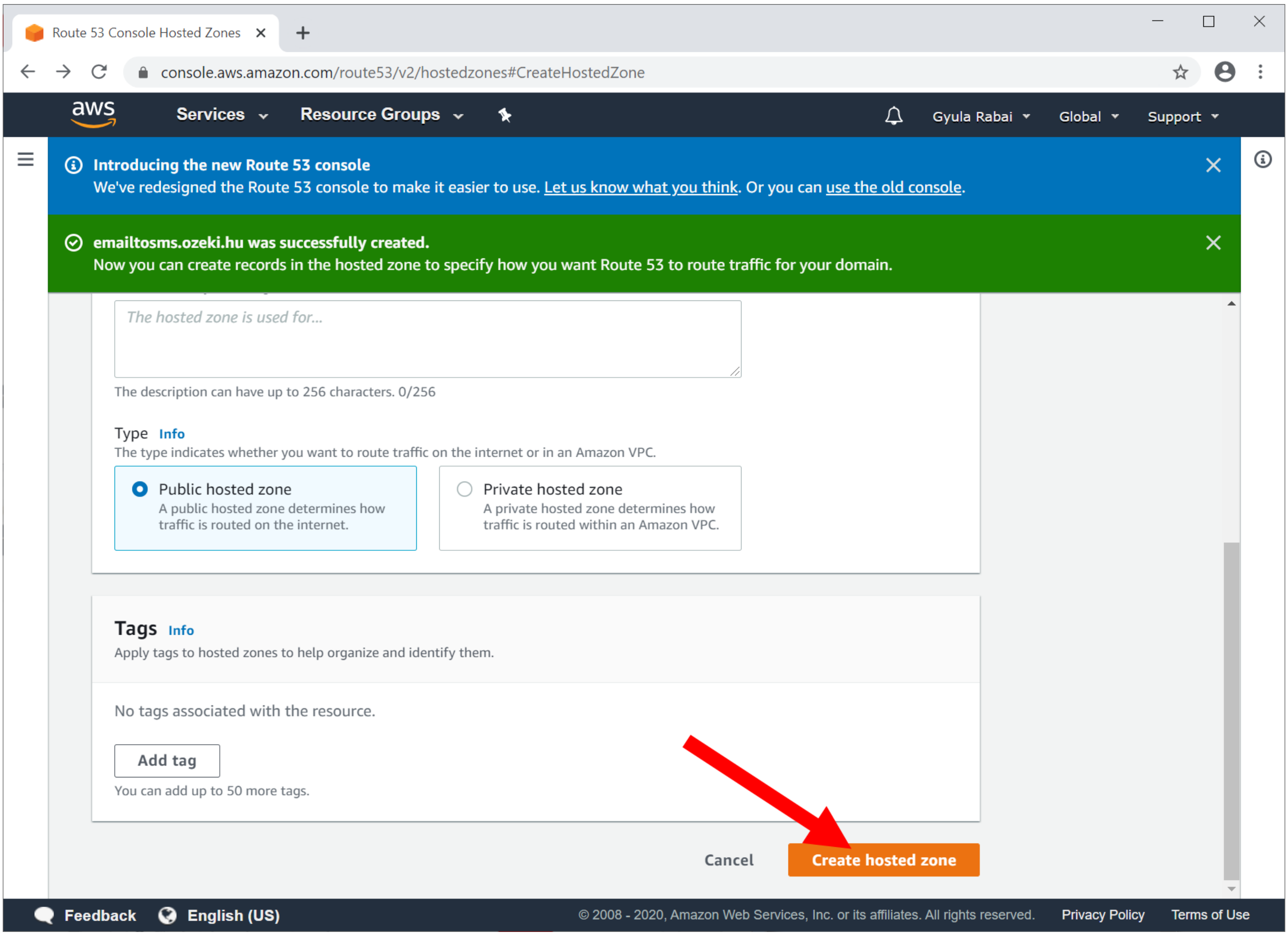
Domain name Info
This is the name of the domain that you want to route traffic for.
emailtosms.ozeki.hu
Valid characters: a-z, 0-9, ! " # $ % & ' ( ) * + , - / : ; < = > ? @ [ \ ] ^ _ ` { | } . ~

Description - optional Info
This value lets you distinguish hosted zones that have the same name.
The hosted zone is used for...
The description can have up to 256 characters. 0/256

Type Info
The type indicates whether you want to route traffic on the internet or in an Amazon VPC.
● Public hosted zone
A public hosted zone determines how traffic is routed on the internet.
○ Private hosted zone
A private hosted zone determines how traffic is routed within an Amazon VPC.

Feedback
English (US)
© 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use

Figure 5 - Enter the domain name: emailtosms.ozeki.hu



Route 53 Console Hosted Zones

console.aws.amazon.com/route53/v2/hostedzones#CreateHostedZone

**Introducing the new Route 53 console**
We've redesigned the Route 53 console to make it easier to use. Let us know what you think. Or you can use the old console.

**emailtosms.ozeki.hu was successfully created.**
Now you can create records in the hosted zone to specify how you want Route 53 to route traffic for your domain.

The hosted zone is used for...

The description can have up to 256 characters. 0/256

Type Info
The type indicates whether you want to route traffic on the internet or in an Amazon VPC.

● Public hosted zone
A public hosted zone determines how traffic is routed on the internet.

○ Private hosted zone
A private hosted zone determines how traffic is routed within an Amazon VPC.

Tags Info
Apply tags to hosted zones to help organize and identify them.

No tags associated with the resource.

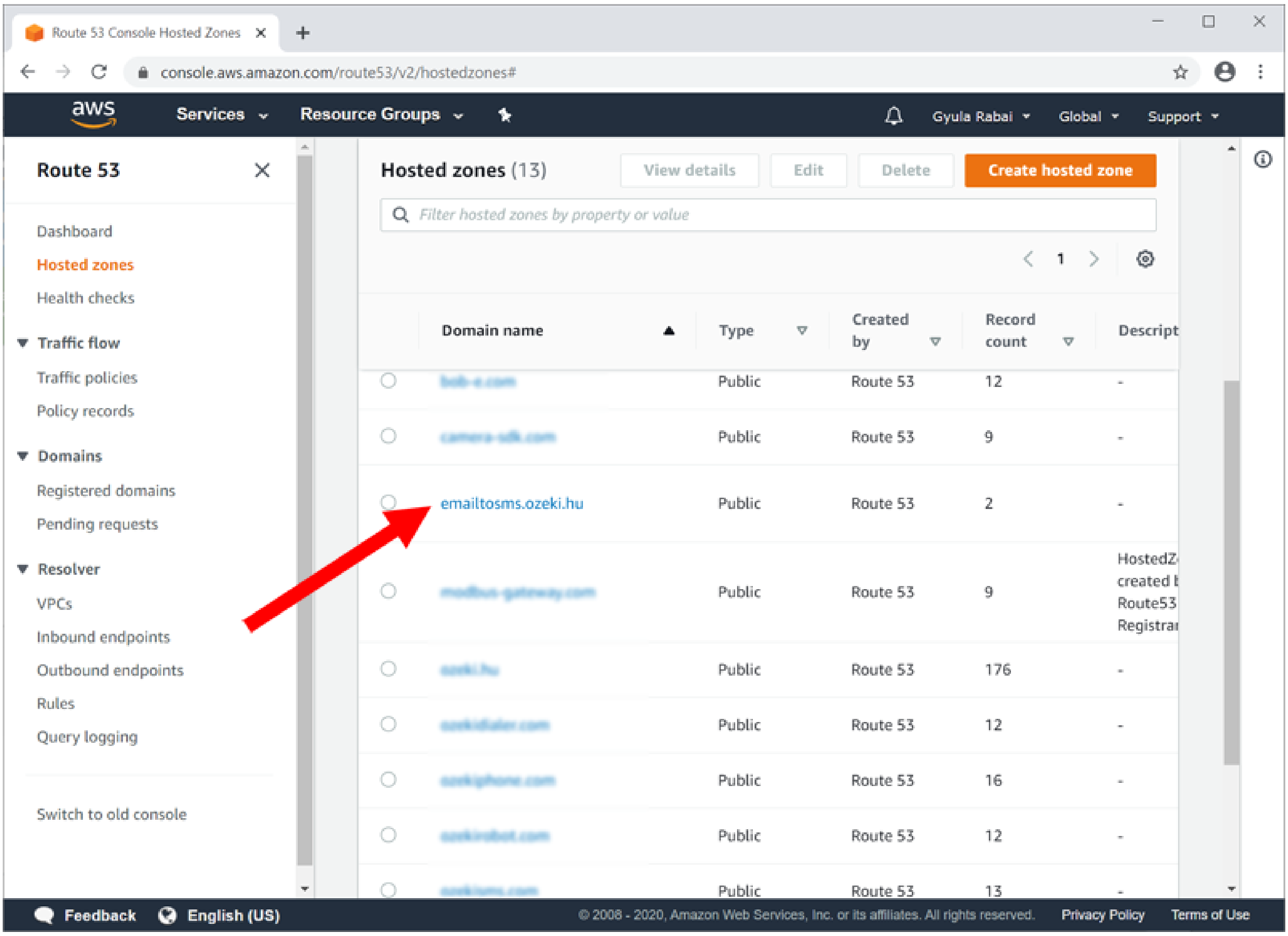
You can add up to 50 more tags.

© 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Figure 6 - Click Create Hosted Zone

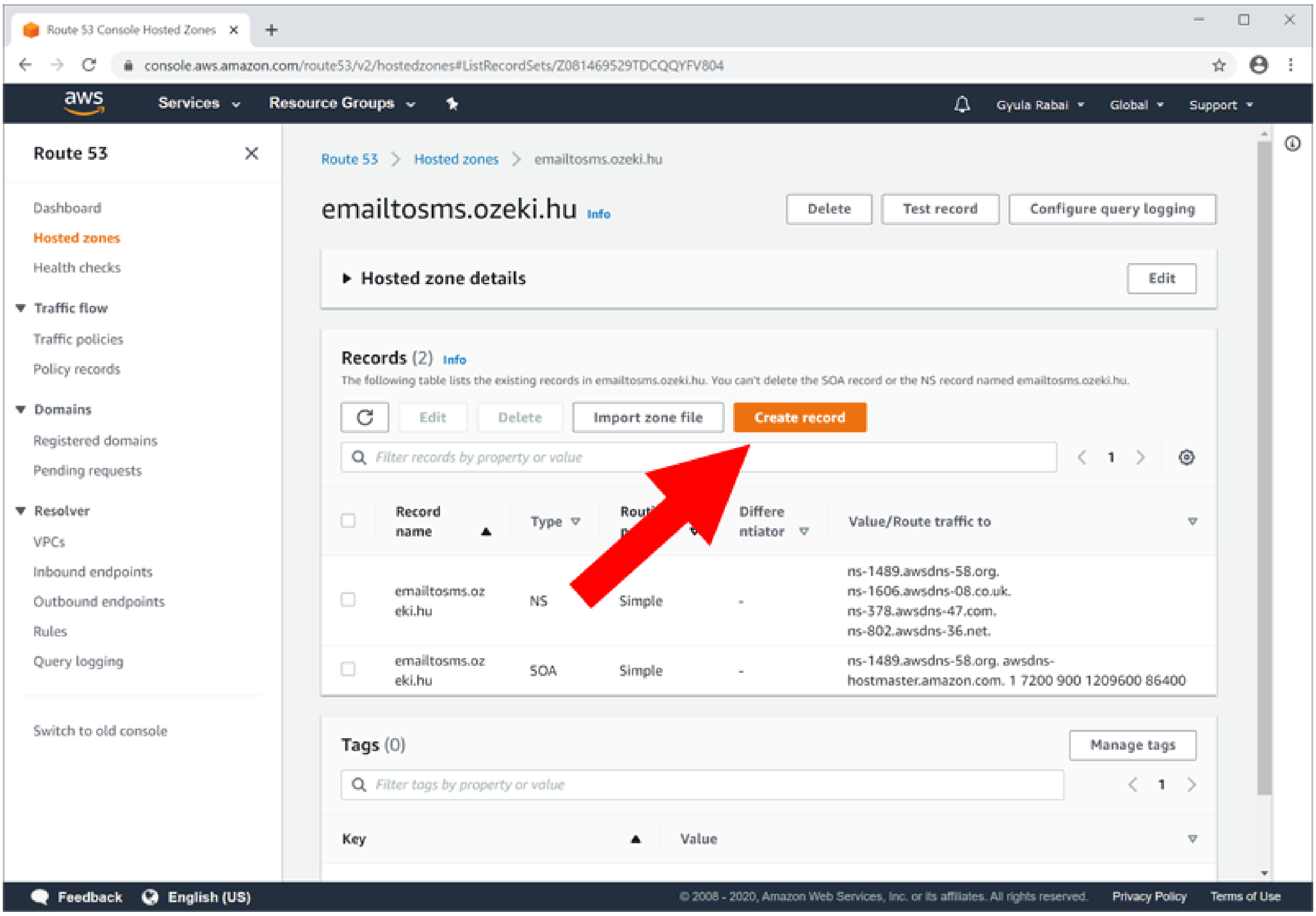
# Step 2: Create an MX record pointing to your SMS gateway

This step is about creating two records in your newly created domain. One should be an A record pointing to your sms gateway, the other should be an MX record telling the email servers to forward emails to your sms gateway. The way the system works, is that email servers on the Internet will lookup the MX record, which point to your mail.emailtosms.ozeki.hu A record which points to the IP address of your SMS gateway. After the email servers found out the IP address of your SMS gateway, they will forward the e-mail messages to it's SMTP port (25).



| | Domain name | Type | Created by | Record count | Descript |
|---|---|---|---|---|---|
| ○ | [redacted] | Public | Route 53 | 12 | - |
| ○ | [redacted] | Public | Route 53 | 9 | - |
| ○ | emailtosms.ozeki.hu | Public | Route 53 | 2 | - |
| ○ | [redacted] | Public | Route 53 | 9 | HostedZ<br>created b<br>Route53<br>Registrar |
| ○ | [redacted] | Public | Route 53 | 176 | - |
| ○ | [redacted] | Public | Route 53 | 12 | - |
| ○ | [redacted] | Public | Route 53 | 16 | - |
| ○ | [redacted] | Public | Route 53 | 12 | - |
| ○ | [redacted] | Public | Route 53 | 13 | - |

Figure 7 - Open your domain



Route 53 Console Hosted Zones

console.aws.amazon.com/route53/v2/hostedzones#ListRecordSets/Z081469529TDCQQYFV804

Route 53

- Dashboard
- Hosted zones
- Health checks

▼ Traffic flow

- Traffic policies
- Policy records

▼ Domains

- Registered domains
- Pending requests

▼ Resolver

- VPCs
- Inbound endpoints
- Outbound endpoints
- Rules
- Query logging

Switch to old console

Route 53 > Hosted zones > emailtosms.ozeki.hu

emailtosms.ozeki.hu Info

▶ Hosted zone details

Records (2) Info

The following table lists the existing records in emailtosms.ozeki.hu. You can't delete the SOA record or the NS record named emailtosms.ozeki.hu.

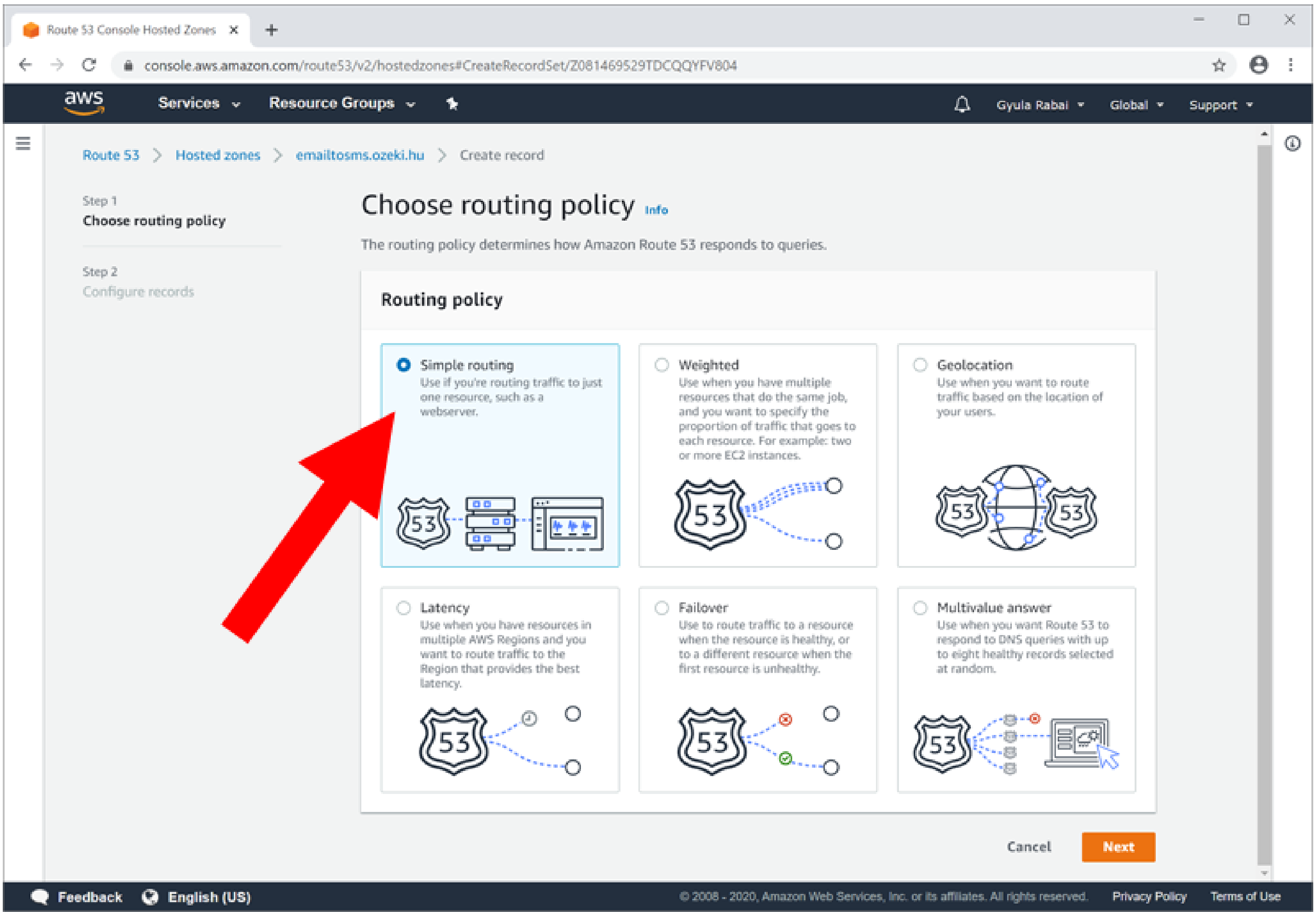
| □ | Record name ▲ | Type ▽ | Routing policy ▽ | Differentiator ▽ | Value/Route traffic to |
|---|---|---|---|---|---|
| □ | emailtosms.ozeki.hu | NS | Simple | - | ns-1489.awsdns-58.org.<br>ns-1606.awsdns-08.co.uk.<br>ns-378.awsdns-47.com.<br>ns-802.awsdns-36.net. |
| □ | emailtosms.ozeki.hu | SOA | Simple | - | ns-1489.awsdns-58.org. awsdns-hostmaster.amazon.com. 1 7200 900 1209600 86400 |

Tags (0)

| Key ▲ | Value ▽ |
|---|---|

Feedback
English (US)
© 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use

Figure 8 - Create MX record



Route 53 Console Hosted Zones

console.aws.amazon.com/route53/v2/hostedzones#CreateRecordSet/Z081469529TDCQQYFV804

Route 53 > Hosted zones > emailtosms.ozeki.hu > Create record

Step 1
Choose routing policy

Step 2
Configure records

Choose routing policy Info

The routing policy determines how Amazon Route 53 responds to queries.

Routing policy

● Simple routing
Use if you're routing traffic to just one resource, such as a webserver.

○ Weighted
Use when you have multiple resources that do the same job, and you want to specify the proportion of traffic that goes to each resource. For example: two or more EC2 instances.

○ Geolocation
Use when you want to route traffic based on the location of your users.

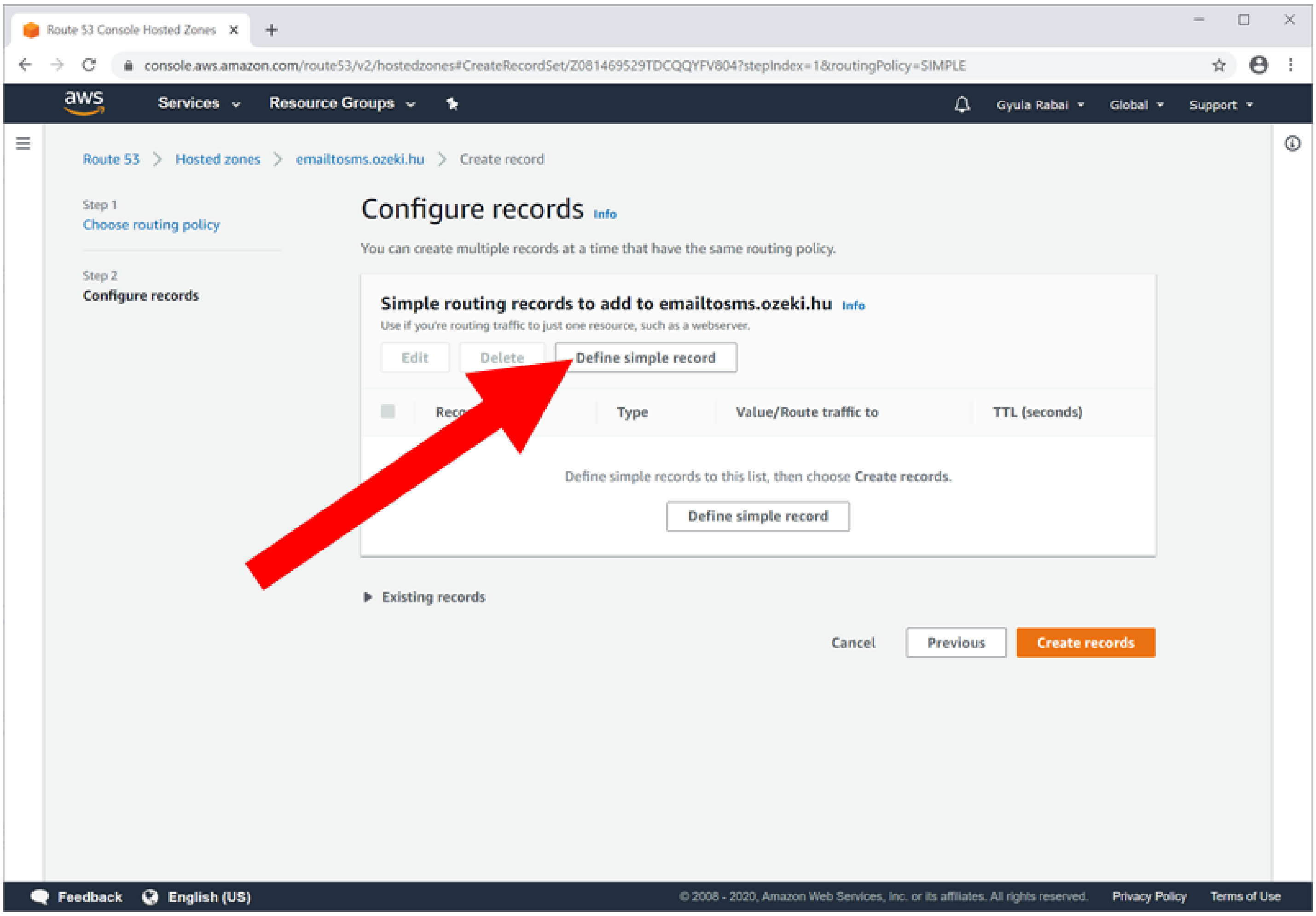
○ Latency
Use when you have resources in multiple AWS Regions and you want to route traffic to the Region that provides the best latency.

○ Failover
Use to route traffic to a resource when the resource is healthy, or to a different resource when the first resource is unhealthy.

○ Multivalue answer
Use when you want Route 53 to respond to DNS queries with up to eight healthy records selected at random.

53
Cancel
Next
Feedback
English (US)
© 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use

Figure 9 - Select simple routing



Route 53 > Hosted zones > emailtosms.ozeki.hu > Create record

Step 1
Choose routing policy

Step 2
Configure records

Configure records Info

You can create multiple records at a time that have the same routing policy.

Simple routing records to add to emailtosms.ozeki.hu Info

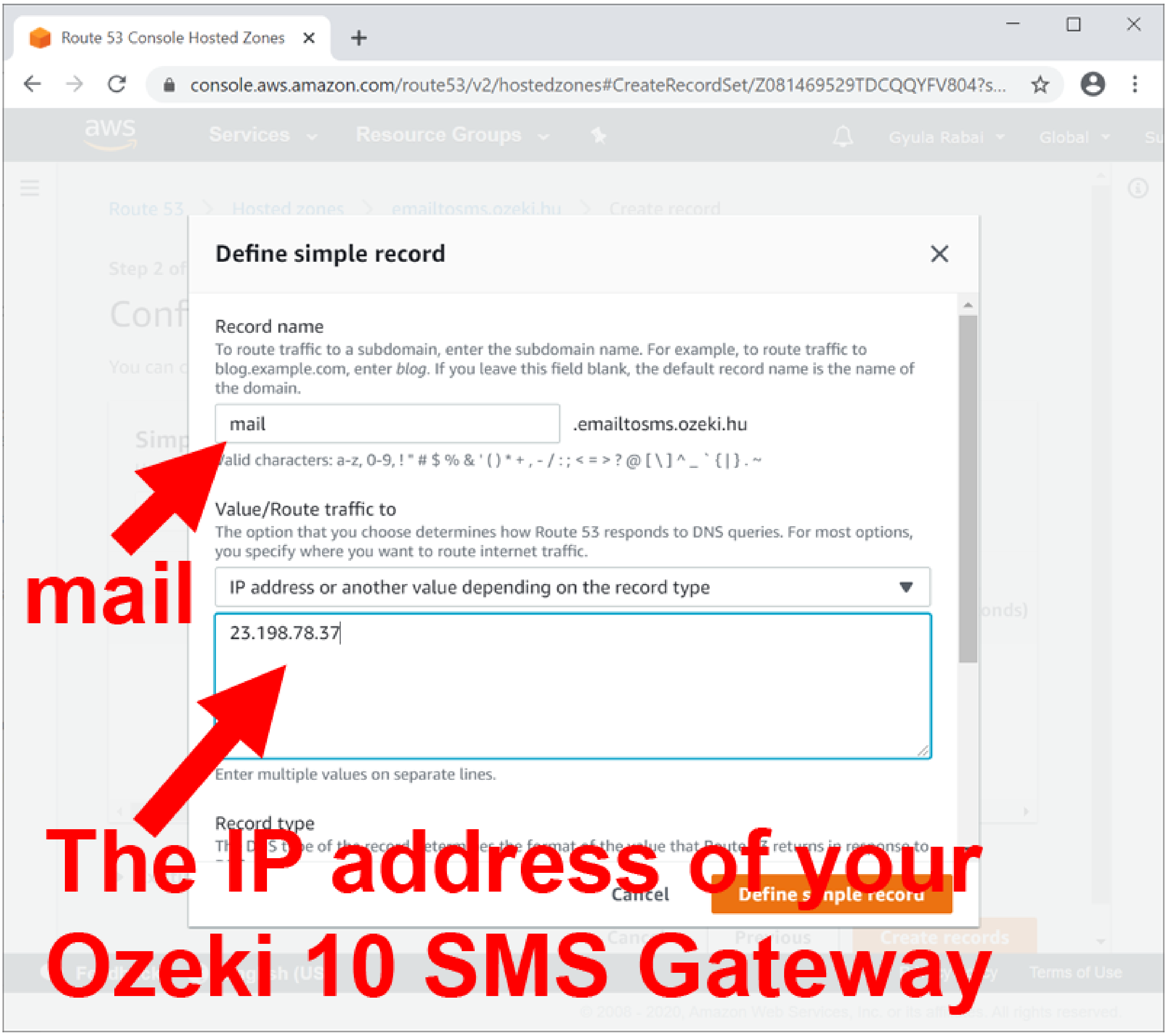
Use if you're routing traffic to just one resource, such as a webserver.

| Record name | Type | Value/Route traffic to | TTL (seconds) |
|---|---|---|---|

Define simple records to this list, then choose **Create records**.

Route 53 Console Hosted Zones
console.aws.amazon.com/route53/v2/hostedzones#CreateRecordSet/Z081469529TDCQQYFV804?stepIndex=1&routingPolicy=SIMPLE
aws
Services
Resource Groups
Gyula Rabai
Global
Support
Edit
Delete
Define simple record
Define simple record
Existing records
Cancel
Previous
Create records
Feedback
English (US)
© 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use

Figure 10 - Select define simple record



Route 53 Console Hosted Zones
console.aws.amazon.com/route53/v2/hostedzones#CreateRecordSet/Z081469529TDCQQYFV804?s...
aws
Services
Resource Groups
Gyula Rabai
Global
Route 53 > Hosted zones > emailtosms.ozeki.hu > Create record
Step 2 of
Conf
Cancel
Previous
Create records
Terms of Use
mail
The IP address of your Ozeki 10 SMS Gateway

Define simple record

Record name

To route traffic to a subdomain, enter the subdomain name. For example, to route traffic to blog.example.com, enter *blog*. If you leave this field blank, the default record name is the name of the domain.

mail .emailtosms.ozeki.hu

Valid characters: a-z, 0-9, ! " # $ % & ' ( ) * + , - / : ; < = > ? @ [ \ ] ^ _ ` { | } . ~

Value/Route traffic to

The option that you choose determines how Route 53 responds to DNS queries. For most options, you specify where you want to route internet traffic.

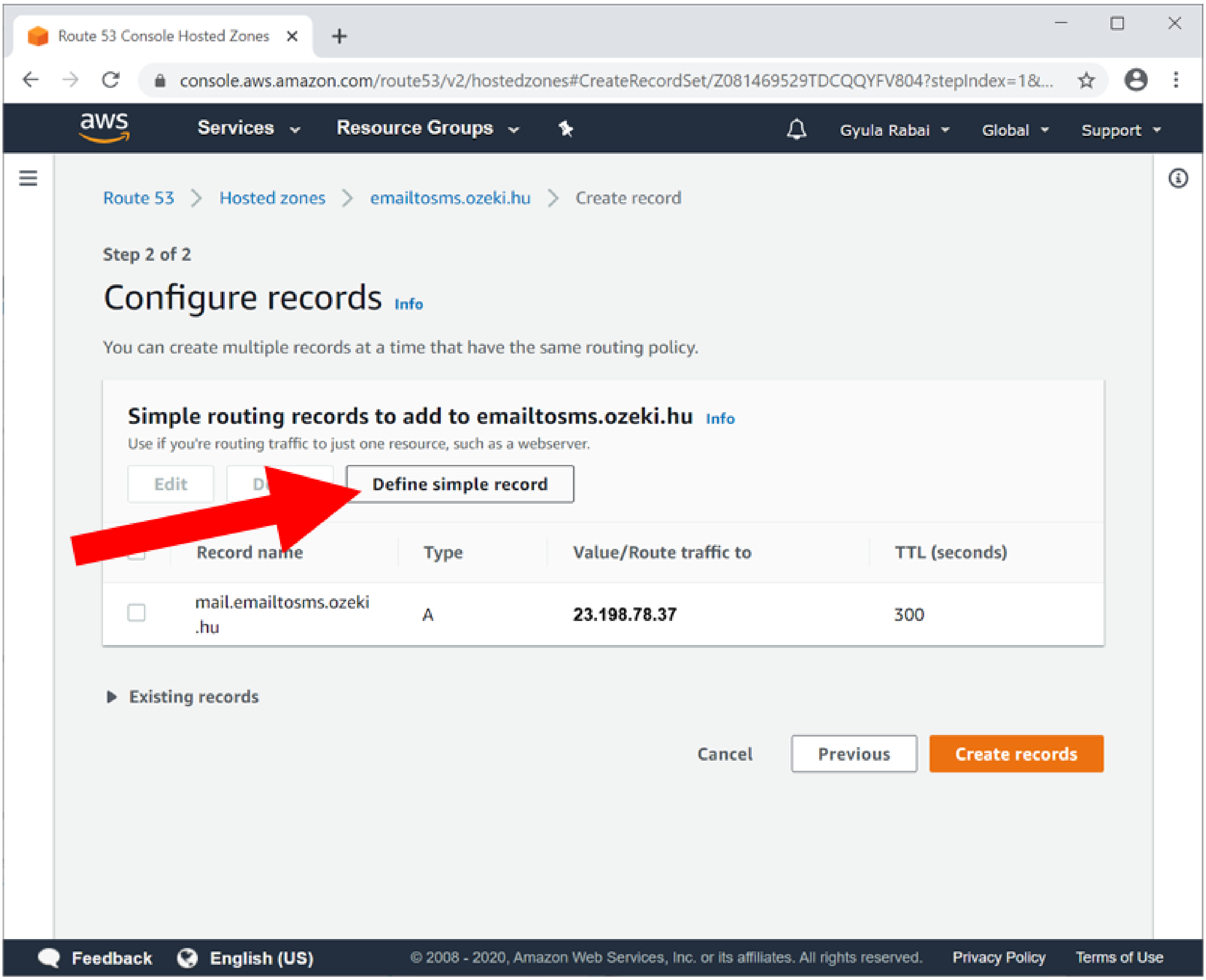
IP address or another value depending on the record type

23.198.78.37

Enter multiple values on separate lines.

Record type

Figure 11 - Create an A record pointing to your SMS gateway



Route 53 Console Hosted Zones
console.aws.amazon.com/route53/v2/hostedzones#CreateRecordSet/Z081469529TDCQQYFV804?stepIndex=1&...
aws
Services
Resource Groups
Gyula Rabai
Global
Support
Route 53
Hosted zones
emailtosms.ozeki.hu
Create record

Step 2 of 2

Configure records Info

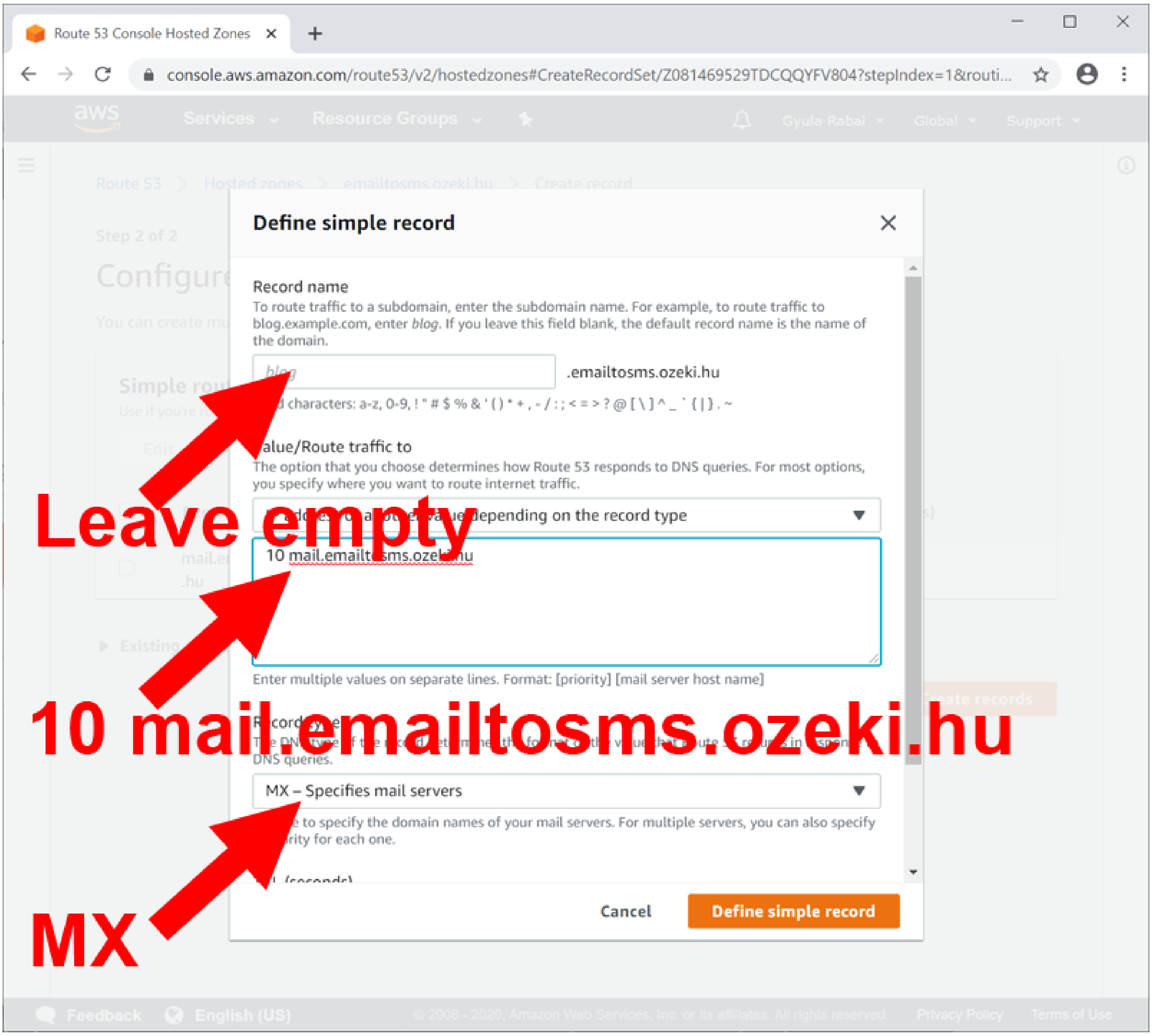
You can create multiple records at a time that have the same routing policy.

Simple routing records to add to emailtosms.ozeki.hu Info

Use if you're routing traffic to just one resource, such as a webserver.

| □ | Record name | Type | Value/Route traffic to | TTL (seconds) |
|---|---|---|---|---|
| □ | mail.emailtosms.ozeki.hu | A | 23.198.78.37 | 300 |

Figure 12 - Select define simple record again


Route 53 Console Hosted Zones
console.aws.amazon.com/route53/v2/hostedzones#CreateRecordSet/Z081469529TDCQQYFV804?stepIndex=1&routi...
aws
Services
Resource Groups
Gyula Rabai
Global
Support
Route 53
Hosted zones
emailtosms.ozeki.hu
Create record
Step 2 of 2
Configure
Simple rou
Existing
Create records
Feedback
English (US)
© 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use

Define simple record

Record name

To route traffic to a subdomain, enter the subdomain name. For example, to route traffic to blog.example.com, enter *blog*. If you leave this field blank, the default record name is the name of the domain.

*blog* .emailtosms.ozeki.hu

d characters: a-z, 0-9, ! " # $ % & ' ( ) * + , - / : ; < = > ? @ [ \ ] ^ _ ` { | } . ~

alue/Route traffic to

The option that you choose determines how Route 53 responds to DNS queries. For most options, you specify where you want to route internet traffic.

depending on the record type ▼

10 mail.emailt sms.ozek hu

Enter multiple values on separate lines. Format: [priority] [mail server host name]

DNS queries.

MX – Specifies mail servers ▼

e to specify the domain names of your mail servers. For multiple servers, you can also specify rity for each one.

Cancel

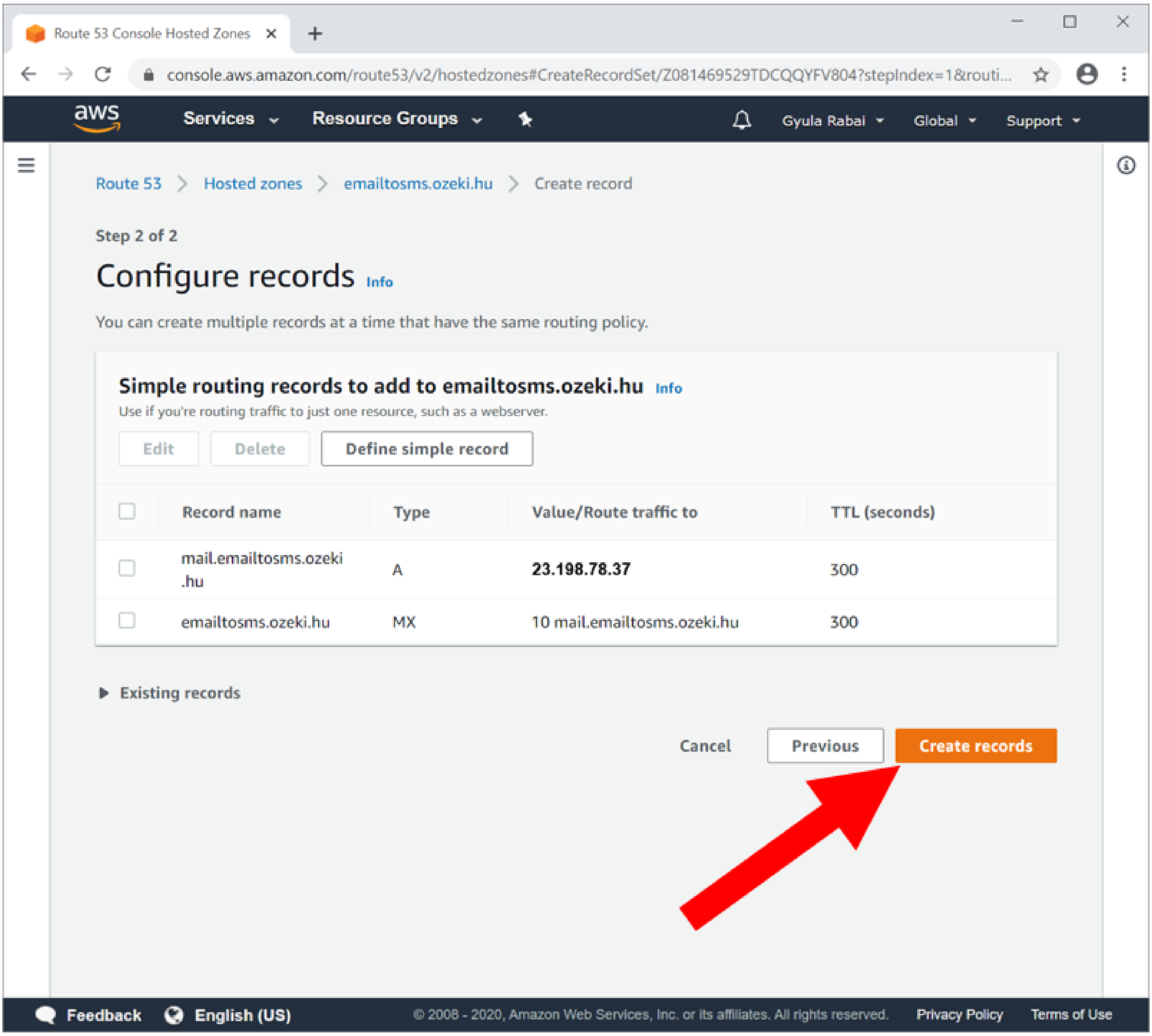
Define simple record

Leave empty

10 mail.emailtosms.ozeki.hu

MX

Figure 14 - Create the MX record



Route 53 Console Hosted Zones
console.aws.amazon.com/route53/v2/hostedzones#CreateRecordSet/Z081469529TDCQQYFV804?stepIndex=1&routi...
aws
Services
Resource Groups
Gyula Rabai
Global
Support
Route 53 > Hosted zones > emailtosms.ozeki.hu > Create record

Step 2 of 2

Configure records Info

You can create multiple records at a time that have the same routing policy.

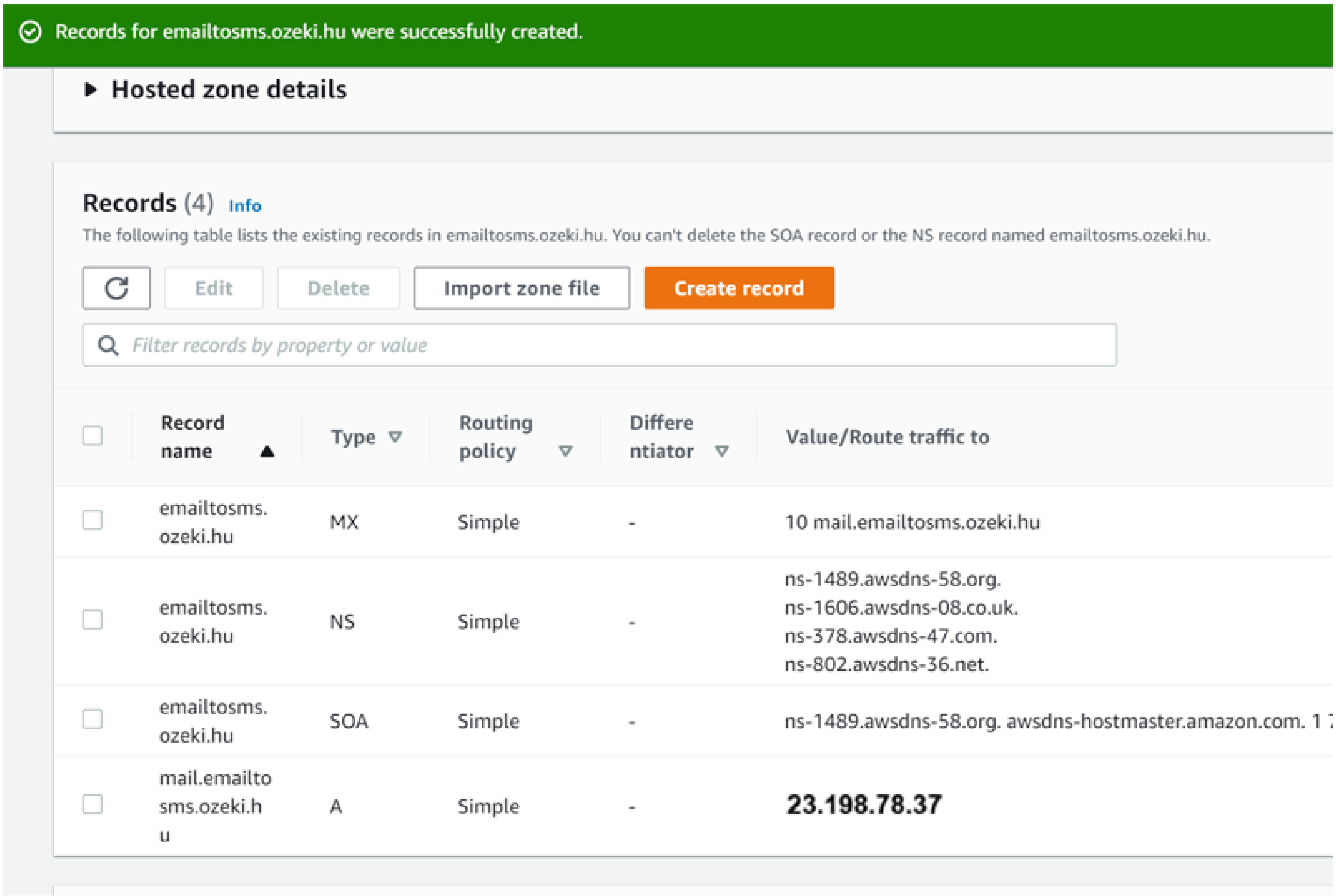
Simple routing records to add to emailtosms.ozeki.hu Info

Use if you're routing traffic to just one resource, such as a webserver.

Edit
Delete
Define simple record

| □ | Record name | Type | Value/Route traffic to | TTL (seconds) |
|---|---|---|---|---|
| □ | mail.emailtosms.ozeki.hu | A | **23.198.78.37** | 300 |
| □ | emailtosms.ozeki.hu | MX | 10 mail.emailtosms.ozeki.hu | 300 |

Figure 15 - Configured Zone



Records for emailtosms.ozeki.hu were successfully created.

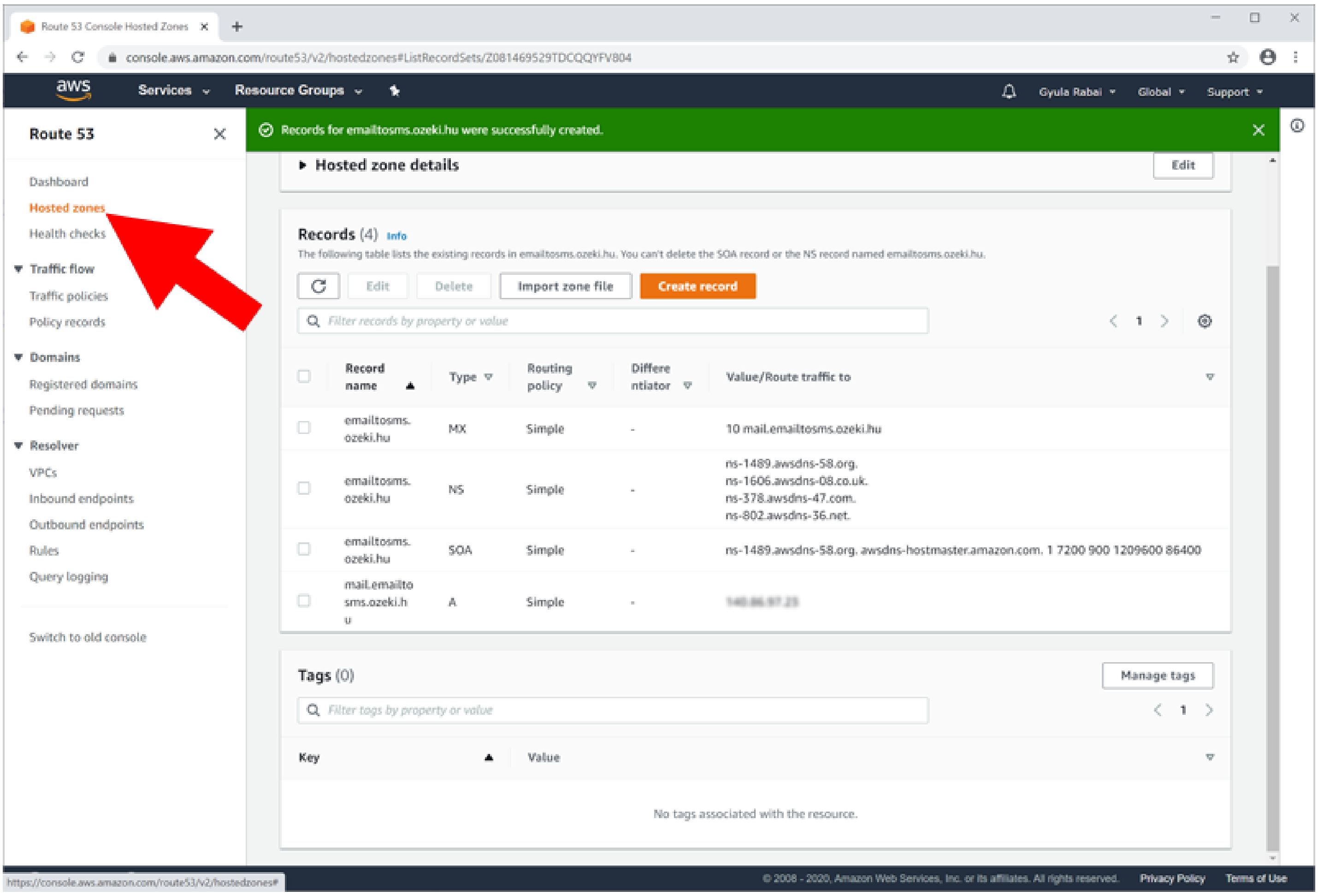
The following table lists the existing records in emailtosms.ozeki.hu. You can't delete the SOA record or the NS record named emailtosms.ozeki.hu.

| □ | Record name ▲ | Type ▽ | Routing policy ▽ | Differentiator ▽ | Value/Route traffic to |
|---|---|---|---|---|---|
| □ | emailtosms.ozeki.hu | MX | Simple | - | 10 mail.emailtosms.ozeki.hu |
| □ | emailtosms.ozeki.hu | NS | Simple | - | ns-1489.awsdns-58.org.<br>ns-1606.awsdns-08.co.uk.<br>ns-378.awsdns-47.com.<br>ns-802.awsdns-36.net. |
| □ | emailtosms.ozeki.hu | SOA | Simple | - | ns-1489.awsdns-58.org. awsdns-hostmaster.amazon.com. 1 7 |
| □ | mail.emailtosms.ozeki.hu | A | Simple | - | 23.198.78.37 |

Figure 16 - Fully configured DNS zone

## Step 3 - Create an NS record pointing to your DNS zone

When your subdomain is created, the next step is to point to it from your main domain. In our exmaple we have create a subdomain called emailtosms.ozeki.hu, so we have to setup an NS record in the domain ozeki.hu to point to it.



Records for emailtosms.ozeki.hu were successfully created.

Hosted zone details

Edit

Records (4) Info

The following table lists the existing records in emailtosms.ozeki.hu. You can't delete the SOA record or the NS record named emailtosms.ozeki.hu.

| □ | Record name | Type | Routing policy | Differentiator | Value/Route traffic to |
|---|---|---|---|---|---|
| □ | emailtosms.ozeki.hu | MX | Simple | - | 10 mail.emailtosms.ozeki.hu |
| □ | emailtosms.ozeki.hu | NS | Simple | - | ns-1489.awsdns-58.org.<br>ns-1606.awsdns-08.co.uk.<br>ns-378.awsdns-47.com.<br>ns-802.awsdns-36.net. |
| □ | emailtosms.ozeki.hu | SOA | Simple | - | ns-1489.awsdns-58.org. awsdns-hostmaster.amazon.com. 1 7200 900 1209600 86400 |
| □ | mail.emailtosms.ozeki.hu | A | Simple | - | [redacted] |

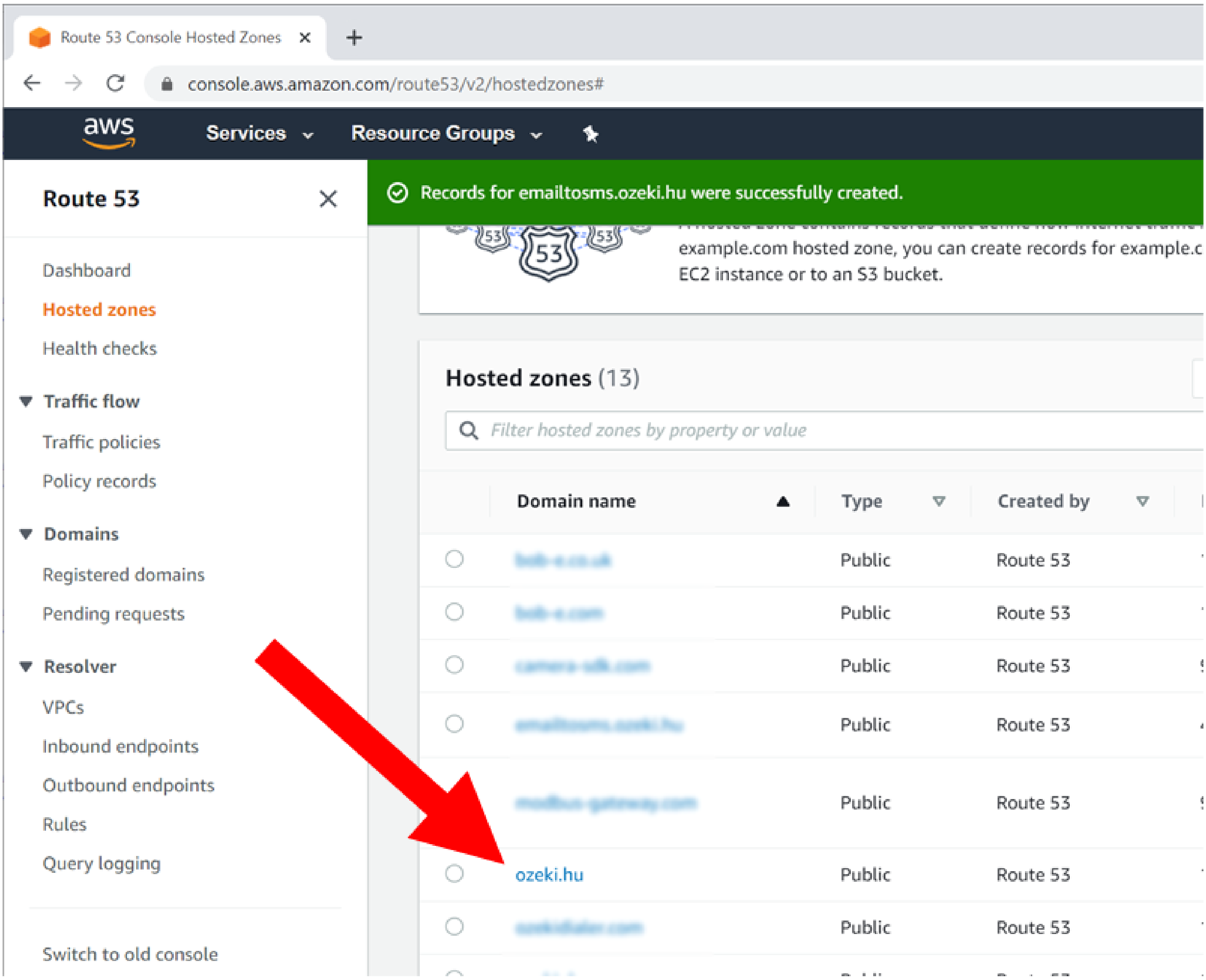
Tags (0)

| Key | Value |
|---|---|

No tags associated with the resource.

https://console.aws.amazon.com/route53/v2/hostedzones#
© 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use

Figure 17 - Back to hosted zones


Route 53 Console Hosted Zones
console.aws.amazon.com/route53/v2/hostedzones#
aws
Services
Resource Groups

Records for emailtosms.ozeki.hu were successfully created.

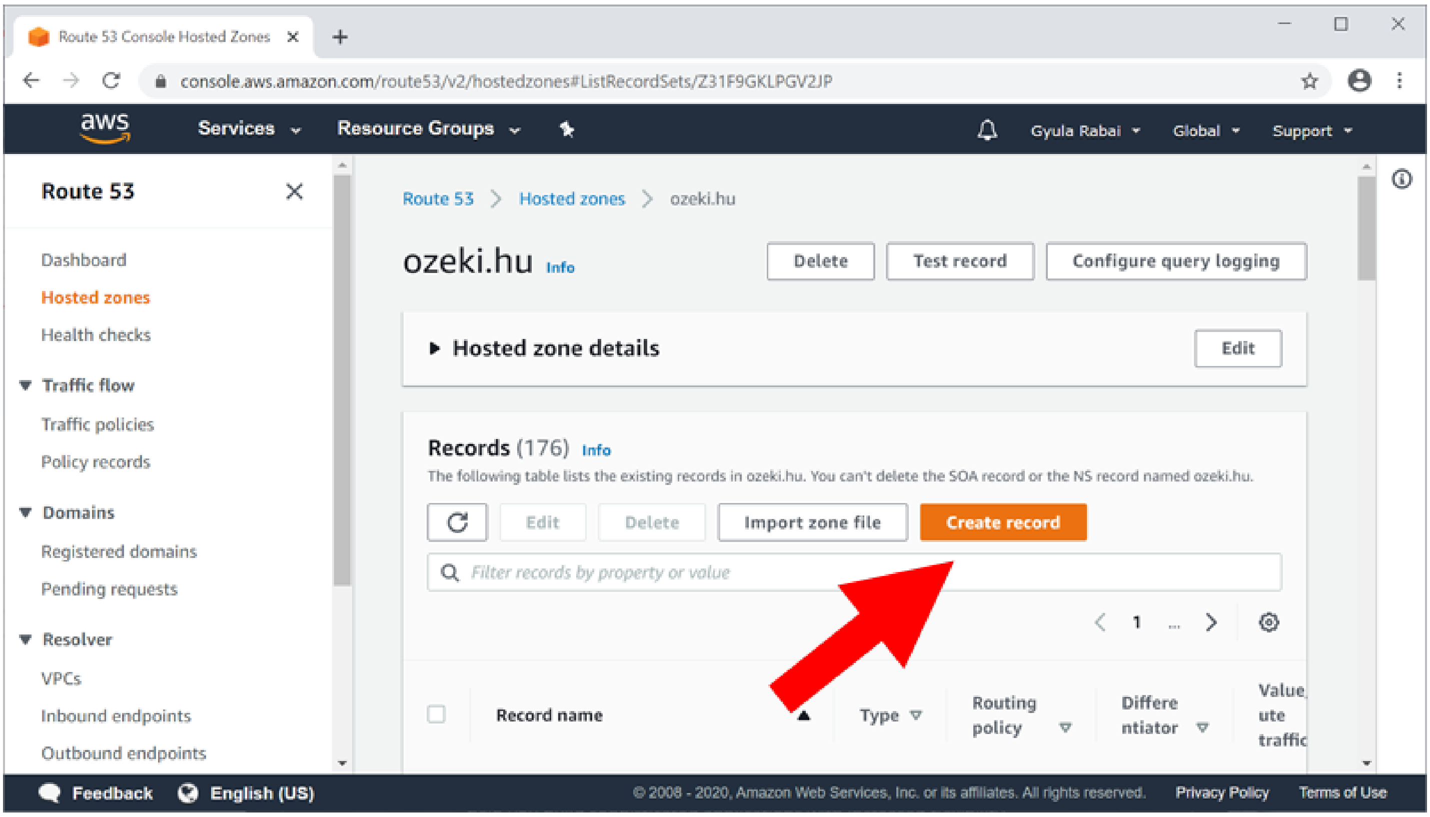
example.com hosted zone, you can create records for example.c EC2 instance or to an S3 bucket.

Hosted zones (13)

Filter hosted zones by property or value

|  | Domain name ▲ | Type ▽ | Created by ▽ |
|---|---|---|---|
| ○ | [redacted] | Public | Route 53 |
| ○ | [redacted] | Public | Route 53 |
| ○ | [redacted] | Public | Route 53 |
| ○ | [redacted] | Public | Route 53 |
| | [redacted] | Public | Route 53 |
| ○ | ozeki.hu | Public | Route 53 |
| ○ | [redacted] | Public | Route 53 |

Figure 18 - Select your main domain



ozeki.hu Info

▸ Hosted zone details

Records (176) Info

The following table lists the existing records in ozeki.hu. You can't delete the SOA record or the NS record named ozeki.hu.

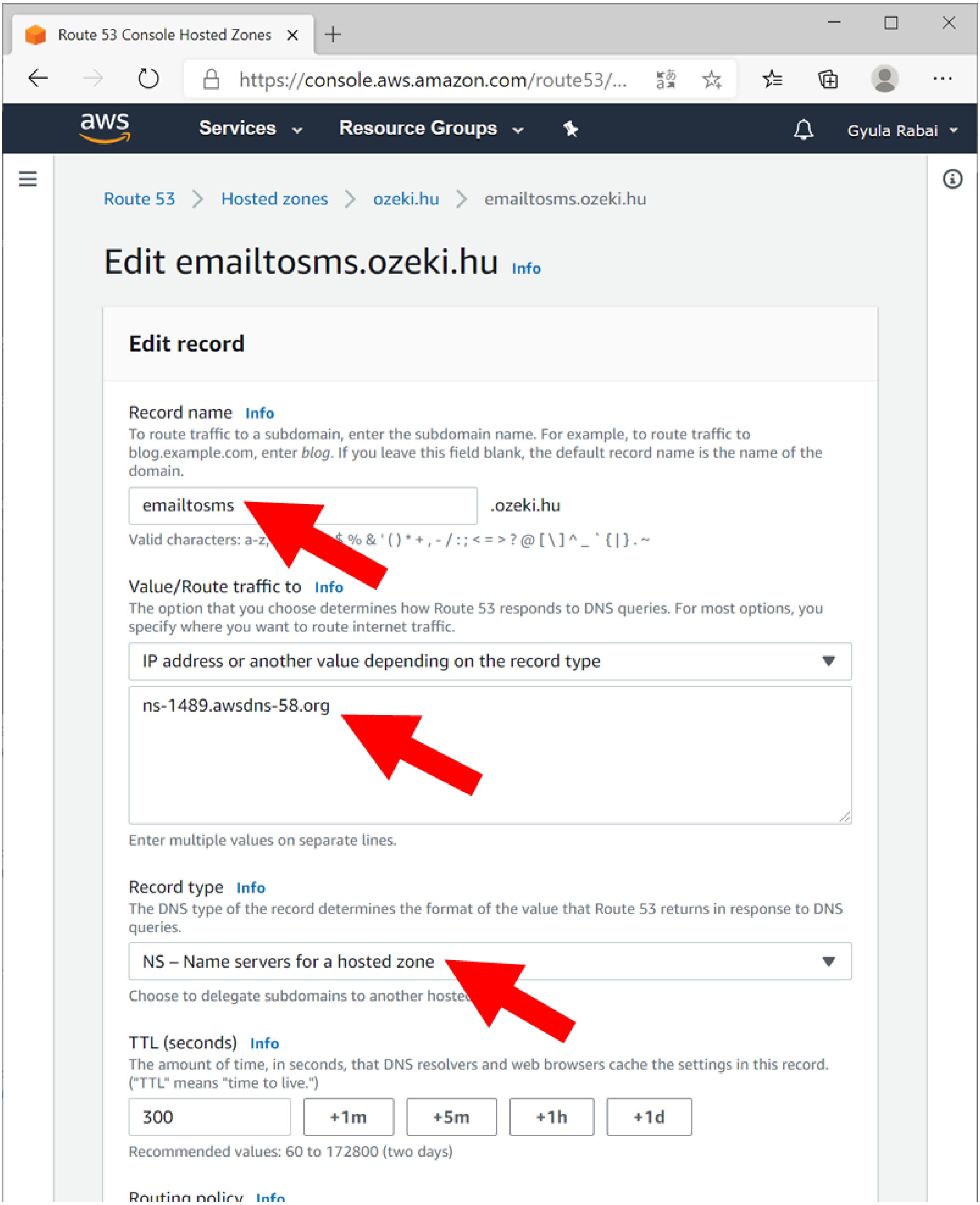
| ☐ | Record name ▲ | Type ▽ | Routing policy ▽ | Differe ntiator ▽ | Value ute traffic |
|---|---|---|---|---|---|

Feedback
English (US)
© 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use

Figure 19 - Select create record in your main domain

The next step is to create the NS record that points to your newly created zone. Note, that the DNS server you provide in this step must match the DNS server serving your new zone. In our exmaple you will see if you look at Figure 16, that the DNS server of emailtosms.ozeki.hu is ns-1489.awsdns-58.org. This is why we provided this value in the created zone. (Of course your DNS server four your subdomain will be different, just make sure it matches.)

You must also note that the domain name emailtosms is used in the Record name of the NS record. This will ensure, that queries coming to the zone emailtosms.ozeki.hu will be directed to the appropriate DNS server serving the emailtosms.ozeki.hu subdomain.


Route 53 Console Hosted Zones
https://console.aws.amazon.com/route53/...
aws
Services
Resource Groups
Gyula Rabai
Route 53 > Hosted zones > ozeki.hu > emailtosms.ozeki.hu

Edit emailtosms.ozeki.hu Info

Edit record

Record name Info

To route traffic to a subdomain, enter the subdomain name. For example, to route traffic to blog.example.com, enter *blog*. If you leave this field blank, the default record name is the name of the domain.

emailtosms .ozeki.hu

Valid characters: a-z, $ % & ' ( ) * + , - / : ; < = > ? @ [ \ ] ^ _ ` { | } . ~

Value/Route traffic to Info

The option that you choose determines how Route 53 responds to DNS queries. For most options, you specify where you want to route internet traffic.

IP address or another value depending on the record type

ns-1489.awsdns-58.org

Enter multiple values on separate lines.

Record type Info

The DNS type of the record determines the format of the value that Route 53 returns in response to DNS queries.

NS – Name servers for a hosted zone

Choose to delegate subdomains to another hoste

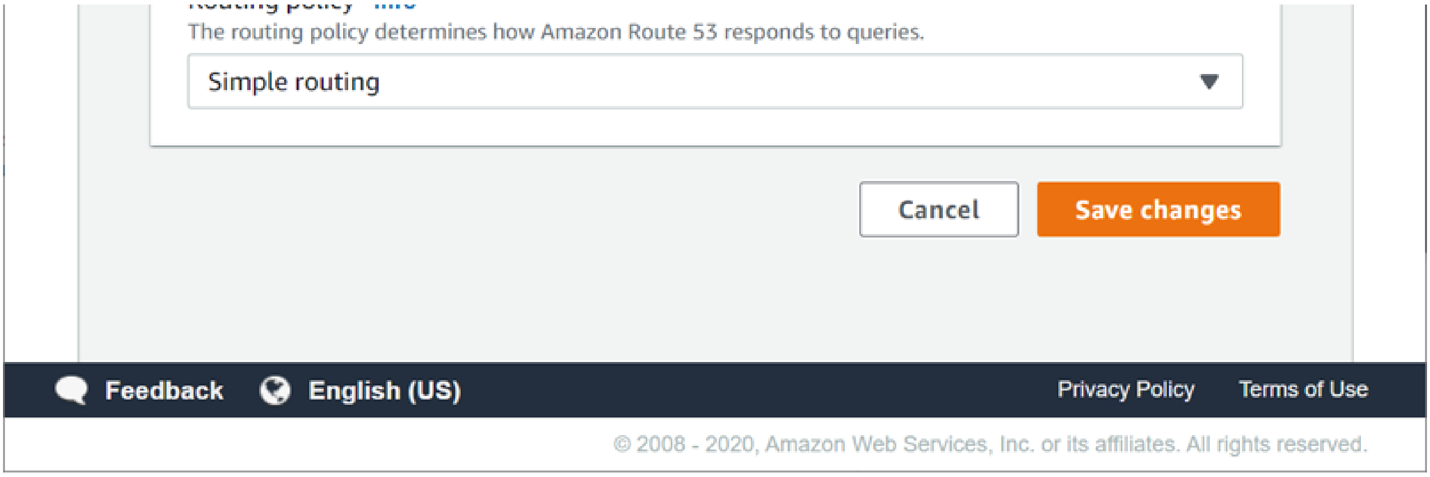
TTL (seconds) Info

The amount of time, in seconds, that DNS resolvers and web browsers cache the settings in this record. ("TTL" means "time to live.")

300 +1m +5m +1h +1d

Recommended values: 60 to 172800 (two days)

Routing policy Info

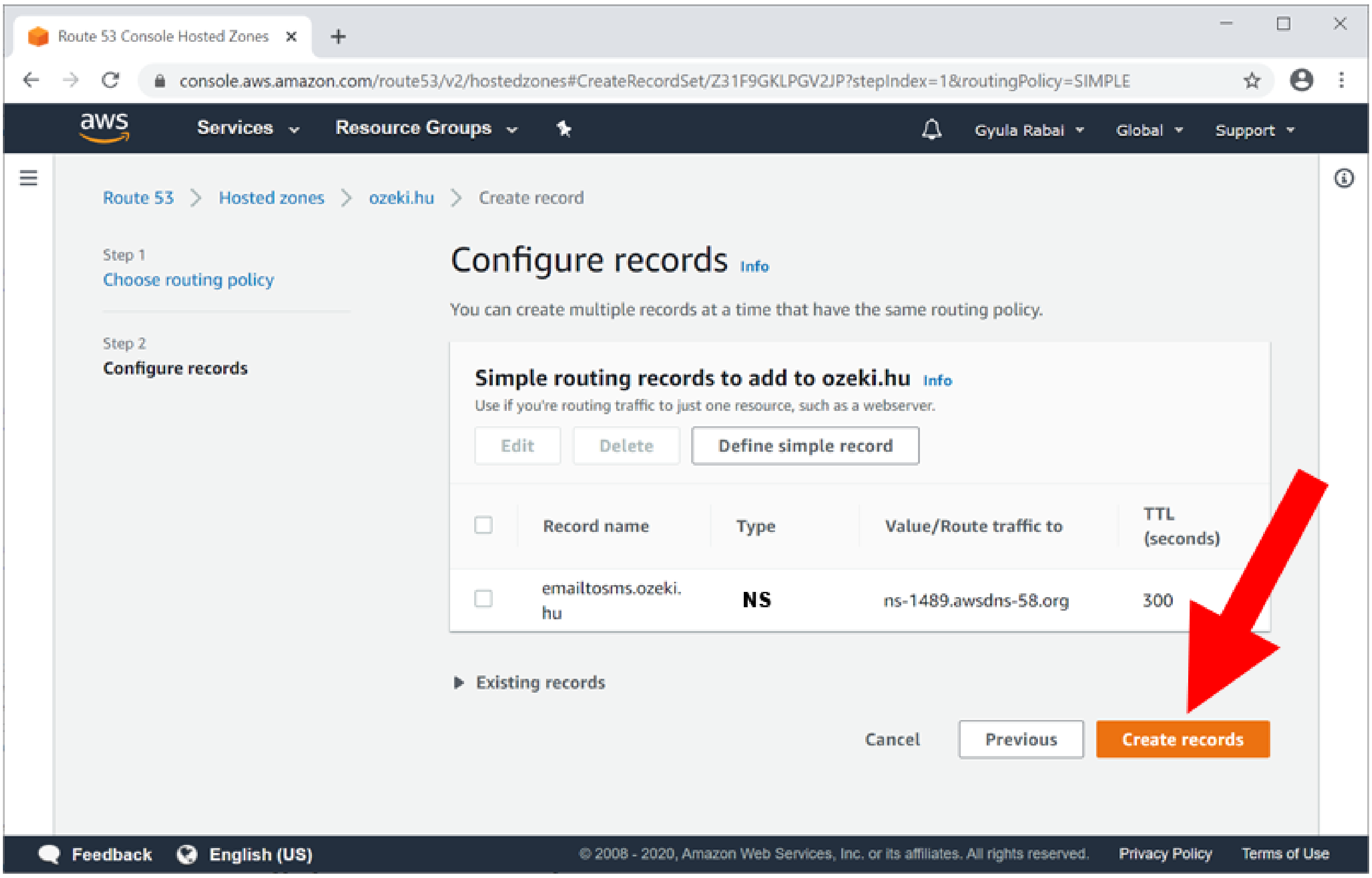


The routing policy determines how Amazon Route 53 responds to queries.

Routing policy: Simple routing

Cancel
Save changes
Feedback
English (US)
Privacy Policy
Terms of Use
© 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Figure 20 - Create the NS record pointing to the subdomain



console.aws.amazon.com/route53/v2/hostedzones#CreateRecordSet/Z31F9GKLPGV2JP?stepIndex=1&routingPolicy=SIMPLE

Configure records

You can create multiple records at a time that have the same routing policy.

Simple routing records to add to ozeki.hu

Use if you're routing traffic to just one resource, such as a webserver.

| □ | Record name | Type | Value/Route traffic to | TTL (seconds) |
|---|---|---|---|---|
| □ | emailtosms.ozeki.hu | NS | ns-1489.awsdns-58.org | 300 |

Figure 21 - Finalize the E-mail to SMS DNS settings.

After these settings your DNS system will forward e-mails coming to phonenubmer@emailtosms.ozeki.hu to your SMS gateway's (23.198.78.37 in our example) SMTP port, which is the TCP/IP port 25.

# Direct e-mail client connections

***Ozeki SMS Gateway provides the opportunity to use an e-mail client for sending or receiving SMS messages. These clients can be also used for forwarding an e-mail to SMS message or the SMS message to an e-mail. This list shows the available e-mail client that can perform the above-mentioned actions with Ozeki SMS Gateway.***

**Mozilla thunderbird**

Mozilla Thunderbird is a popular e-mail client and used by a lot of people. That's why it is a great feature to connect it to your Ozeki SMS Gateway and be able to send or receive SMS messages with the e-mail client. This guide shows you how you can setup SMTP and POP3 users in Ozeki and connect them to Mozilla Thunderbird to be able to use SMS to e-mail forvarding or vice versa.

Check how to send or receive SMS messages using Thunderbird

**Windows mail**

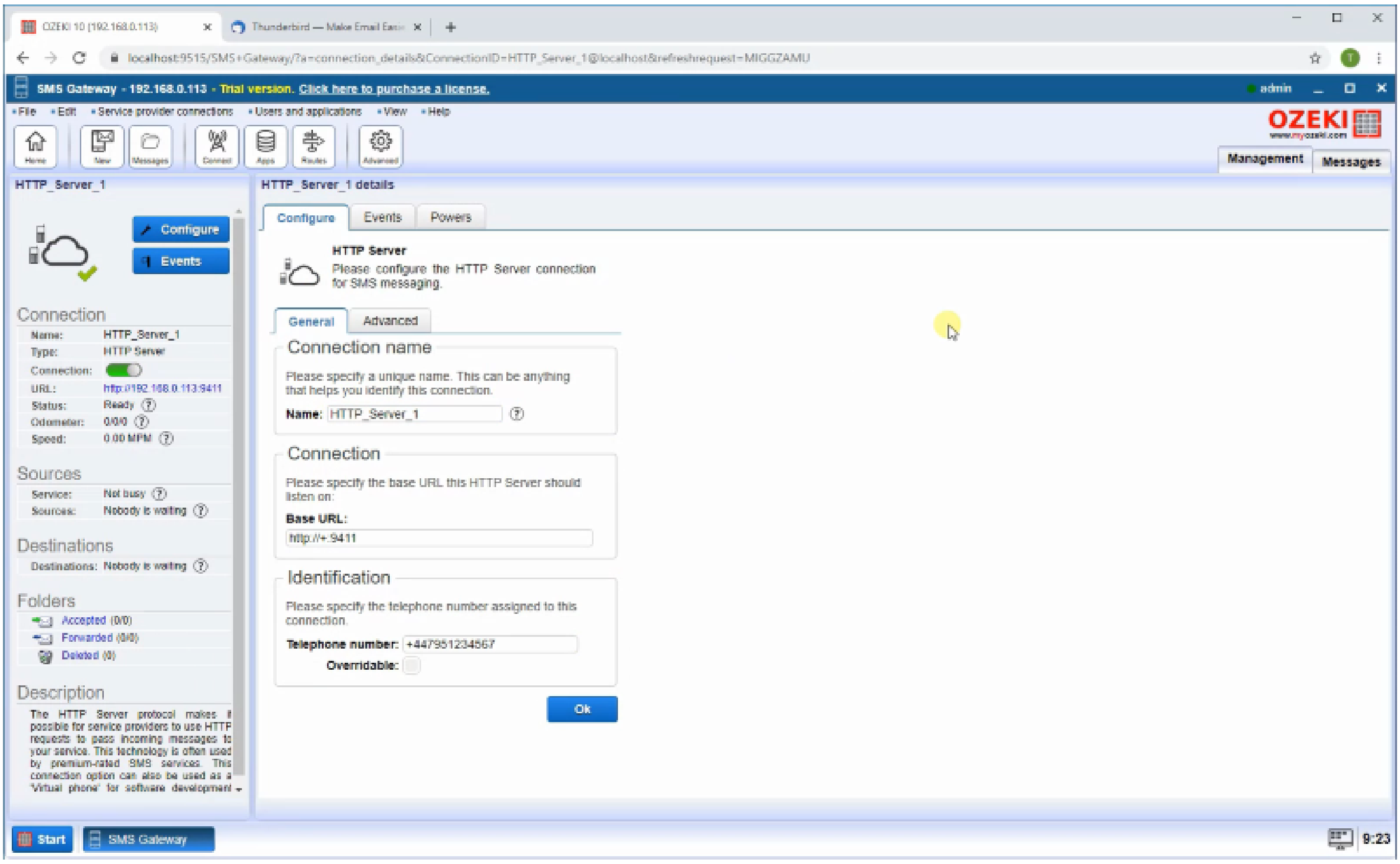
Windows mail is the default e-mail client which is installed on the most of the computers run on Windows operating system. This e-mail client can be connected to Ozeki SMS Gateway using a SMTP and a POP3 user connection. The details of these connections need to be provided in Windows mail, and then, it can be used for forwarding e-mail or SMS messages.

Learn more about how you can use Windows mail to handle SMS messages

# How to use mozilla thunderbird to send/receive sms

***This guide explains how you can setup Ozeki SMS Gateway and Mozilla Thunderbird to send/receive sms. This configuration will allow you to setup e-mail to SMS and sms to e-mail forwarding.***

The first step in order to be able to send and receive SMS using Thunderbird is to establish a Service provider connection to which the message will be forwarded. In this documentation, the HTTP server plays this role so install an HTTP server in the Ozeki SMS Gateway (Figure 1).


SMS Gateway - 192.168.0.113 - Trial version. Click here to purchase a license.

HTTP_Server_1

Connection

Name: HTTP_Server_1
Type: HTTP Server
URL: http://192.168.0.113:9411
Status: Ready
Odometer: 0/0/0
Speed: 0.00 MPM

Sources

Service: Not busy
Sources: Nobody is waiting

Destinations

Destinations: Nobody is waiting

Folders

Accepted (0/0)
Forwarded (0/0)
Deleted (0)

Description

The HTTP Server protocol makes it possible for service providers to use HTTP requests to pass incoming messages to your service. This technology is often used by premium-rated SMS services. This connection option can also be used as a 'Virtual phone' for software development

HTTP_Server_1 details

HTTP Server
Please configure the HTTP Server connection for SMS messaging.

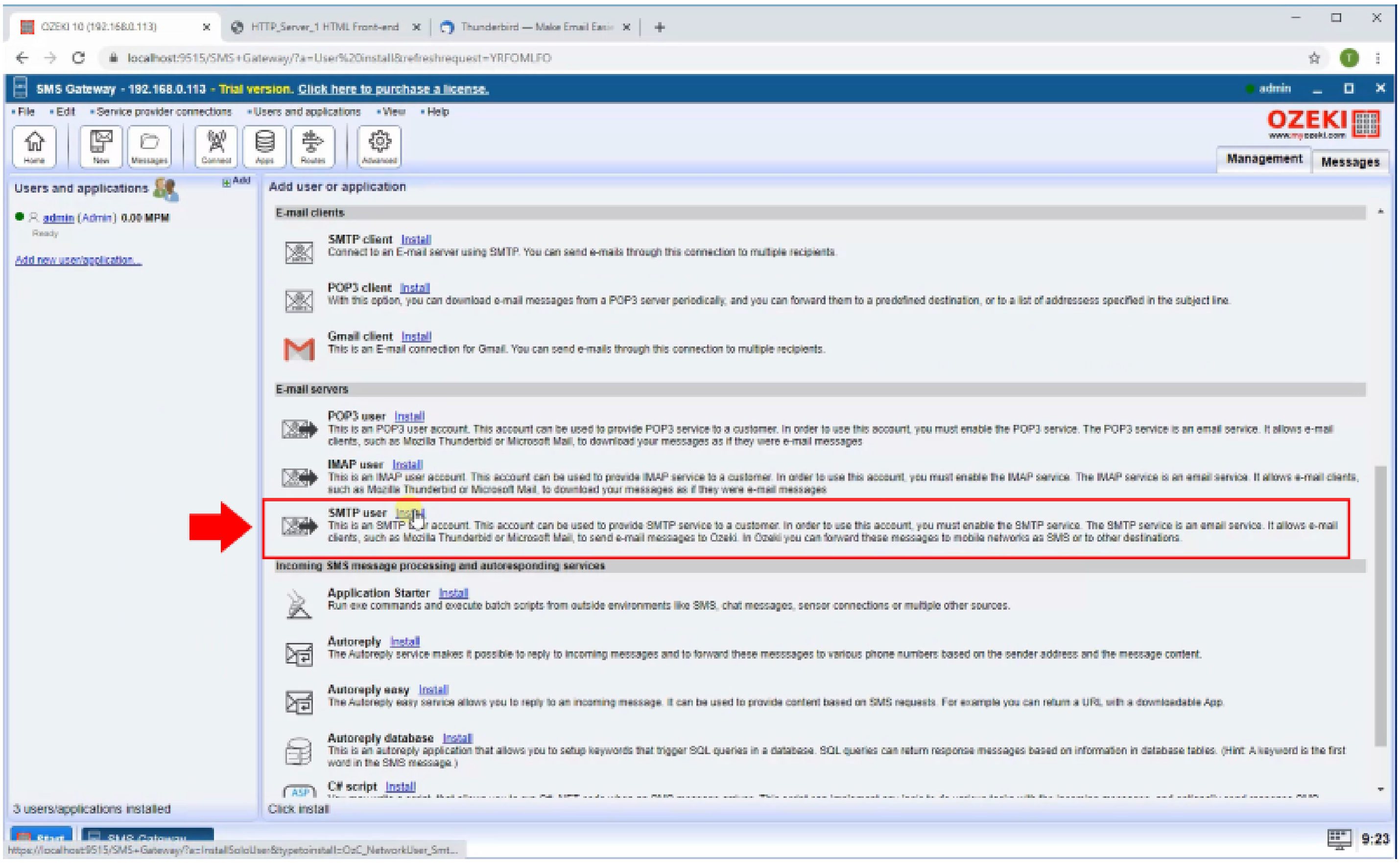
Connection name
Please specify a unique name. This can be anything that helps you identify this connection.
Name: HTTP_Server_1

Connection
Please specify the base URL this HTTP Server should listen on:
Base URL: http://+:9411

Identification
Please specify the telephone number assigned to this connection.
Telephone number: +447951234567
Overridable: □

Figure 1 - Install HTTP Server

The next step is to create an SMTP User to receive emails from Thunderbird. Open the Apps menu, the left side panel contains the already installed users and applications. The right side panel contains the users and applications you can install with a brief description next to them. Search the SMTP User and click the blue 'install' button next to it (Figure 2).


OZEKI 10 (192.168.0.113)
HTTP_Server_1 HTML Front-end
Thunderbird — Make Email Easi
localhost:9515/SMS+Gateway/?a=User%20install&refreshrequest=YRFOMLFO
SMS Gateway - 192.168.0.113 - Trial version. Click here to purchase a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
OZEKI
www.myozeki.com
Management
Messages
Users and applications
Add
admin (Admin) 0.00 MPM
Ready
Add new user/application...
3 users/applications installed
Start
SMS Gateway
https://localhost:9515/SMS+Gateway/?a=InstallSoloUser&typetoinstall=OzC_NetworkUser_Smt...
9:23

Add user or application

E-mail clients

SMTP client Install
Connect to an E-mail server using SMTP. You can send e-mails through this connection to multiple recipients.

POP3 client Install
With this option, you can download e-mail messages from a POP3 server periodically, and you can forward them to a predefined destination, or to a list of addressess specified in the subject line.

Gmail client Install
This is an E-mail connection for Gmail. You can send e-mails through this connection to multiple recipients.

E-mail servers

POP3 user Install
This is an POP3 user account. This account can be used to provide POP3 service to a customer. In order to use this account, you must enable the POP3 service. The POP3 service is an email service. It allows e-mail clients, such as Mozilla Thunderbid or Microsoft Mail, to download your messages as if they were e-mail messages

IMAP user Install
This is an IMAP user account. This account can be used to provide IMAP service to a customer. In order to use this account, you must enable the IMAP service. The IMAP service is an email service. It allows e-mail clients, such as Mozilla Thunderbid or Microsoft Mail, to download your messages as if they were e-mail messages

SMTP user Install
This is an SMTP user account. This account can be used to provide SMTP service to a customer. In order to use this account, you must enable the SMTP service. The SMTP service is an email service. It allows e-mail clients, such as Mozilla Thunderbid or Microsoft Mail, to send e-mail messages to Ozeki. In Ozeki you can forward these messages to mobile networks as SMS or to other destinations.

Incoming SMS message processing and autoresponding services

Application Starter Install
Run exe commands and execute batch scripts from outside environments like SMS, chat messages, sensor connections or multiple other sources.

Autoreply Install
The Autoreply service makes it possible to reply to incoming messages and to forward these messsages to various phone numbers based on the sender address and the message content.

Autoreply easy Install
The Autoreply easy service allows you to reply to an incoming message. It can be used to provide content based on SMS requests. For example you can return a URL with a downloadable App.

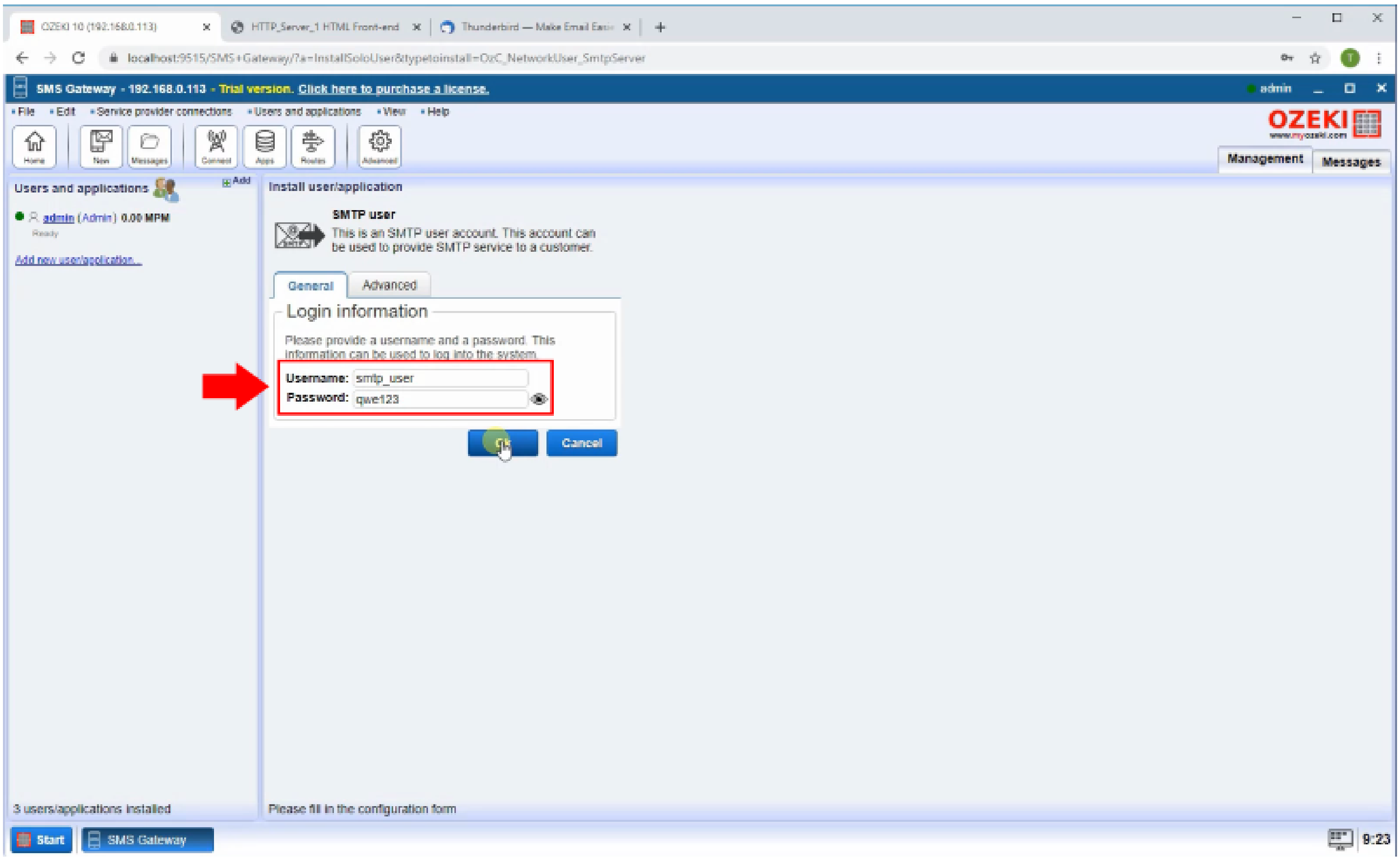
Autoreply database Install
This is an autoreply application that allows you to setup keywords that trigger SQL queries in a database. SQL queries can return response messages based on information in database tables. (Hint: A keyword is the first word in the SMS message.)

C# script Install

Click install

Figure 2 - SMTP User install

Clicking the Install link will bring up the SMTP User installation panel. Here, you need to enter a unique username in the Username field and a password in the Passeowd filed (Figure 3).


OZEKI 10 (192.168.0.113)
HTTP_Server_1 HTML Front-end
Thunderbird — Make Email Easi
localhost:9515/SMS+Gateway/?a=InstallSoloUser&typetoinstall=OzC_NetworkUser_SmtpServer
SMS Gateway - 192.168.0.113 - Trial version. Click here to purchase a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
OZEKI
www.myozeki.com
Management
Messages
Users and applications
Add
admin (Admin) 0.00 MPM
Ready
Add new user/application...
3 users/applications installed
General
Advanced
Ok
Cancel
Start
SMS Gateway
9:23

Install user/application

SMTP user
This is an SMTP user account. This account can be used to provide SMTP service to a customer.

Login information

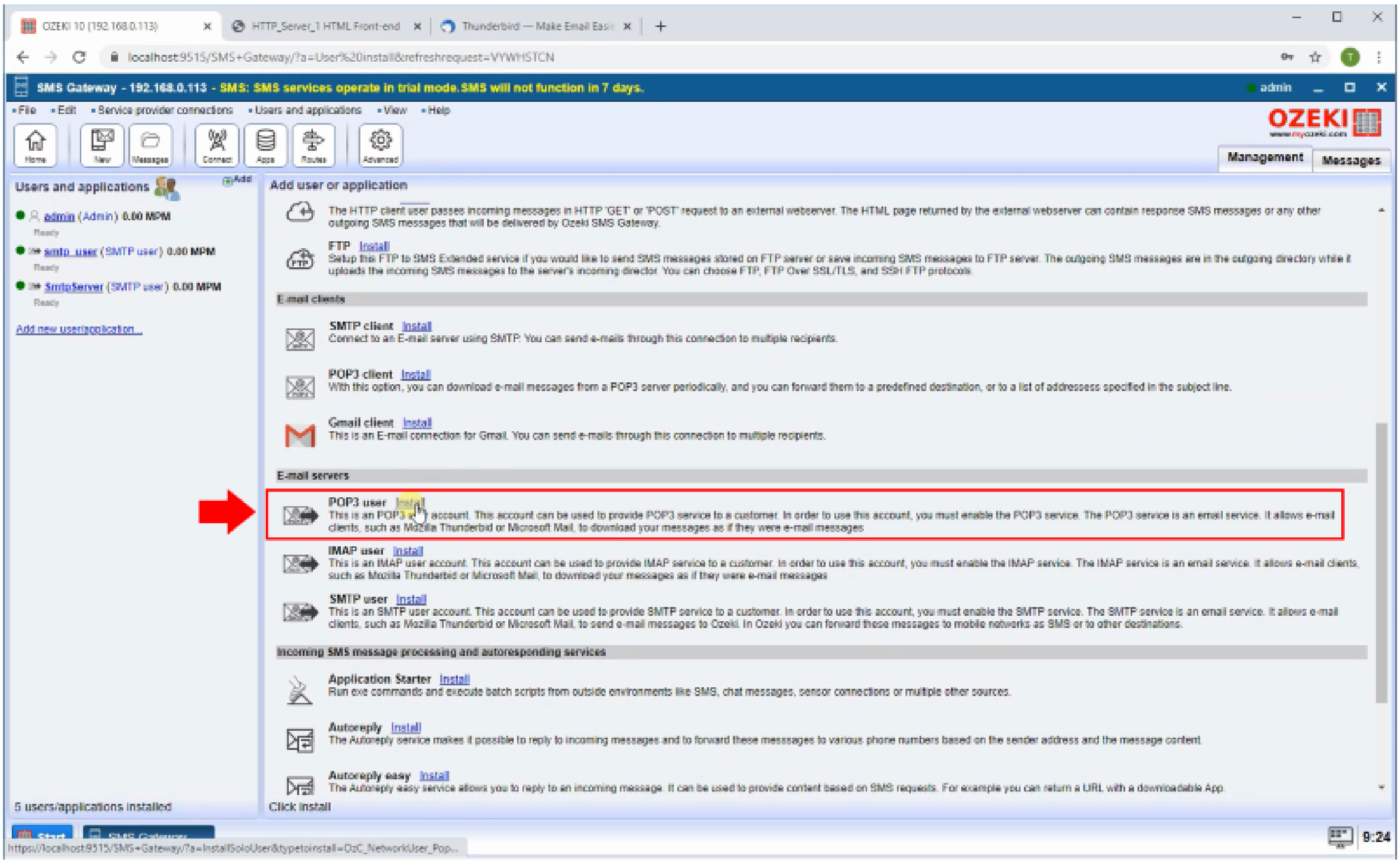
Please provide a username and a password. This information can be used to log into the system.

Username: smtp_user
Password: qwe123

Please fill in the configuration form

Figure 3 - SMTP username and password

Now create a POP3 User to send emails to the Thunderbird. Open the Apps menu, the left side panel contains the already installed users and applications. The right side panel contains the users and applications you can install with a brief description next to them. Search the POP3 User and click the blue 'install' button next to it (Figure 4).


localhost:9515/SMS+Gateway/?a=User%20install&refreshrequest=VYWHSTCN
SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
OZEKI
Management
Messages
Users and applications
Add
admin (Admin) 0.00 MPM
smtp_user (SMTP user) 0.00 MPM
SmtpServer (SMTP user) 0.00 MPM
Ready
Add new user/application...
5 users/applications installed
Start
SMS Gateway
9:24

Add user or application

The HTTP client user passes incoming messages in HTTP 'GET' or 'POST' request to an external webserver. The HTML page returned by the external webserver can contain response SMS messages or any other outgoing SMS messages that will be delivered by Ozeki SMS Gateway.

FTP Install
Setup this FTP to SMS Extended service if you would like to send SMS messages stored on FTP server or save incoming SMS messages to FTP server. The outgoing SMS messages are in the outgoing directory while it uploads the incoming SMS messages to the server's incoming director. You can choose FTP, FTP Over SSL/TLS, and SSH FTP protocols.

E-mail clients

SMTP client Install
Connect to an E-mail server using SMTP. You can send e-mails through this connection to multiple recipients.

POP3 client Install
With this option, you can download e-mail messages from a POP3 server periodically, and you can forward them to a predefined destination, or to a list of addressess specified in the subject line.

Gmail client Install
This is an E-mail connection for Gmail. You can send e-mails through this connection to multiple recipients.

E-mail servers

POP3 user Install
This is an POP3 user account. This account can be used to provide POP3 service to a customer. In order to use this account, you must enable the POP3 service. The POP3 service is an email service. It allows e-mail clients, such as Mozilla Thunderbid or Microsoft Mail, to download your messages as if they were e-mail messages

IMAP user Install
This is an IMAP user account. This account can be used to provide IMAP service to a customer. In order to use this account, you must enable the IMAP service. The IMAP service is an email service. It allows e-mail clients, such as Mozilla Thunderbid or Microsoft Mail, to download your messages as if they were e-mail messages

SMTP user Install
This is an SMTP user account. This account can be used to provide SMTP service to a customer. In order to use this account, you must enable the SMTP service. The SMTP service is an email service. It allows e-mail clients, such as Mozilla Thunderbid or Microsoft Mail, to send e-mail messages to Ozeki. In Ozeki you can forward these messages to mobile networks as SMS or to other destinations.

Incoming SMS message processing and autoresponding services

Application Starter Install
Run exe commands and execute batch scripts from outside environments like SMS, chat messages, sensor connections or multiple other sources.

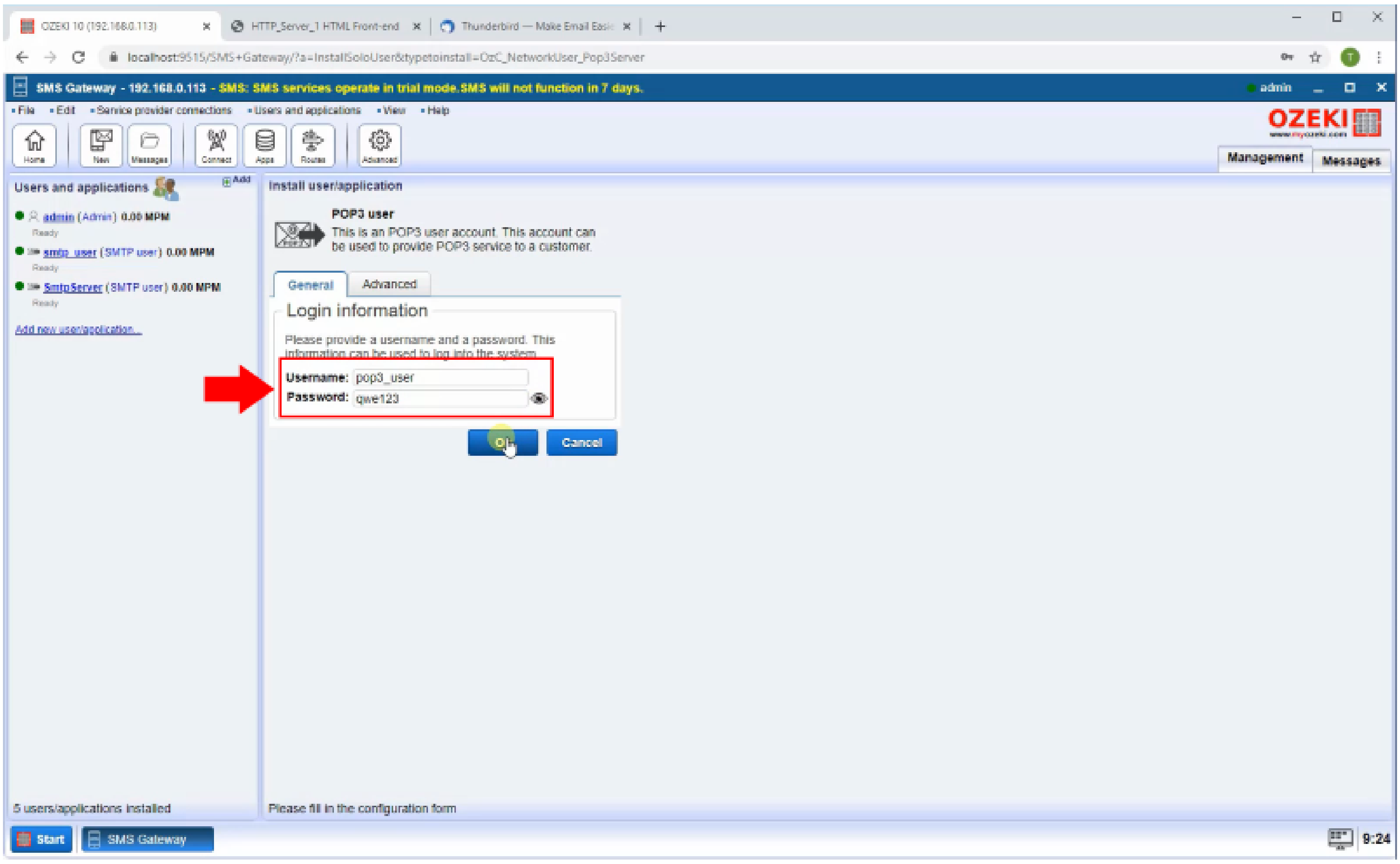
Autoreply Install
The Autoreply service makes it possible to reply to incoming messages and to forward these messsages to various phone numbers based on the sender address and the message content.

Autoreply easy Install
The Autoreply easy service allows you to reply to an incoming message. It can be used to provide content based on SMS requests. For example you can return a URL with a downloadable App.

Click install

Figure 4 - POP3 User install

Clicking the Install link will bring up the POP3 User installation panel. Here, you need to enter a unique username in the Username field and a password in the Passeowd filed (Figure 5).


localhost:9515/SMS+Gateway/?a=InstallSoloUser&typetoinstall=OzC_NetworkUser_Pop3Server
SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
OZEKI
Management
Messages
Users and applications
Add
admin (Admin) 0.00 MPM
smtp_user (SMTP user) 0.00 MPM
SmtpServer (SMTP user) 0.00 MPM
Ready
Add new user/application...
5 users/applications installed
General
Advanced
Ok
Cancel
Start
SMS Gateway
9:24

Install user/application

POP3 user
This is an POP3 user account. This account can be used to provide POP3 service to a customer.

Login information

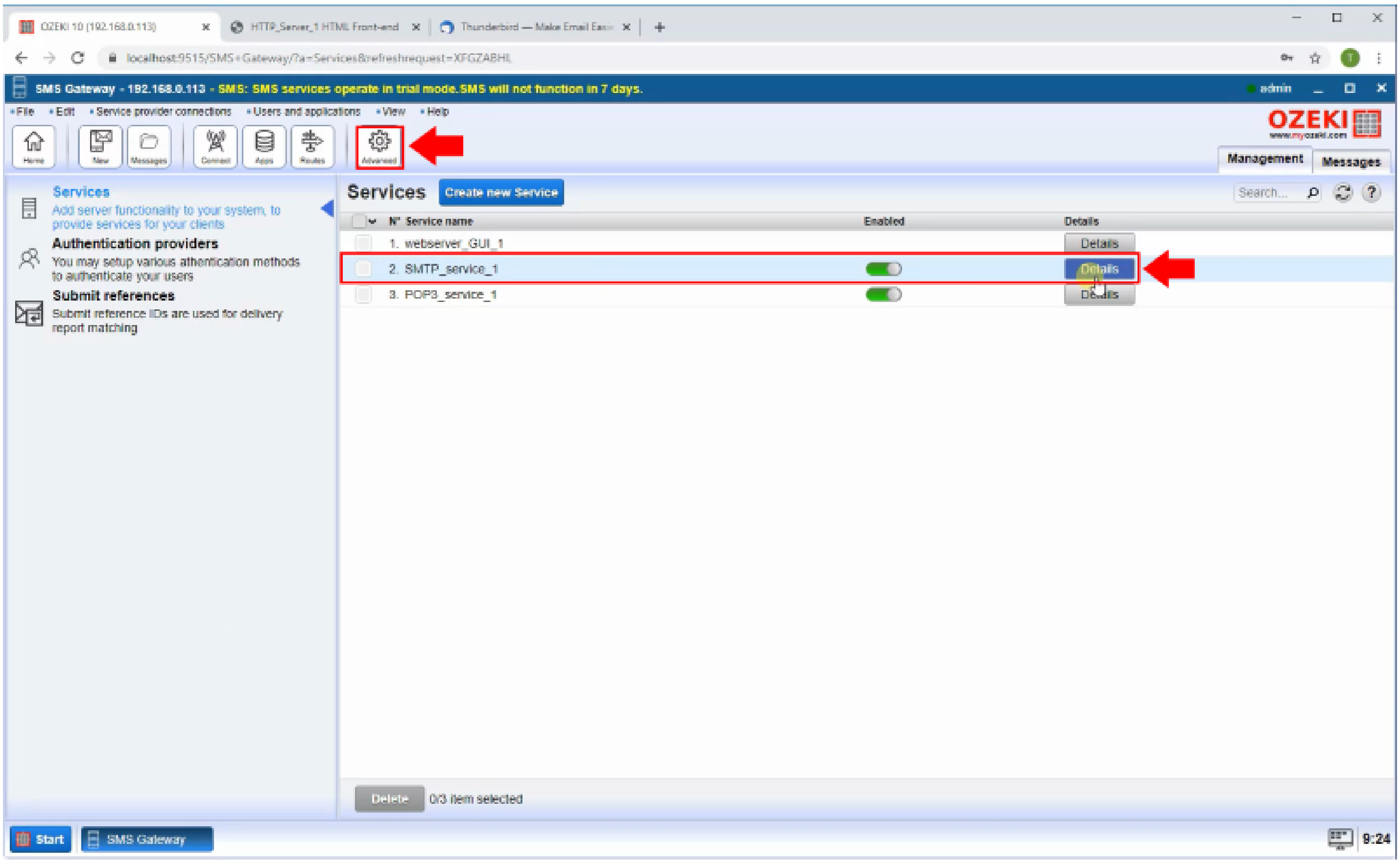
Please provide a username and a password. This information can be used to log into the system.

Username: pop3_user
Password: qwe123

Please fill in the configuration form

Figure 5 - POP3 username and password

Then you must then enable authentication on the SMTP server. To do this, open the SMTP server details page in the advanced menu as the Figure 6 shows.



SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.

**Services**
Add server functionality to your system, to provide services for your clients

**Authentication providers**
You may setup various athentication methods to authenticate your users

**Submit references**
Submit reference IDs are used for delivery report matching

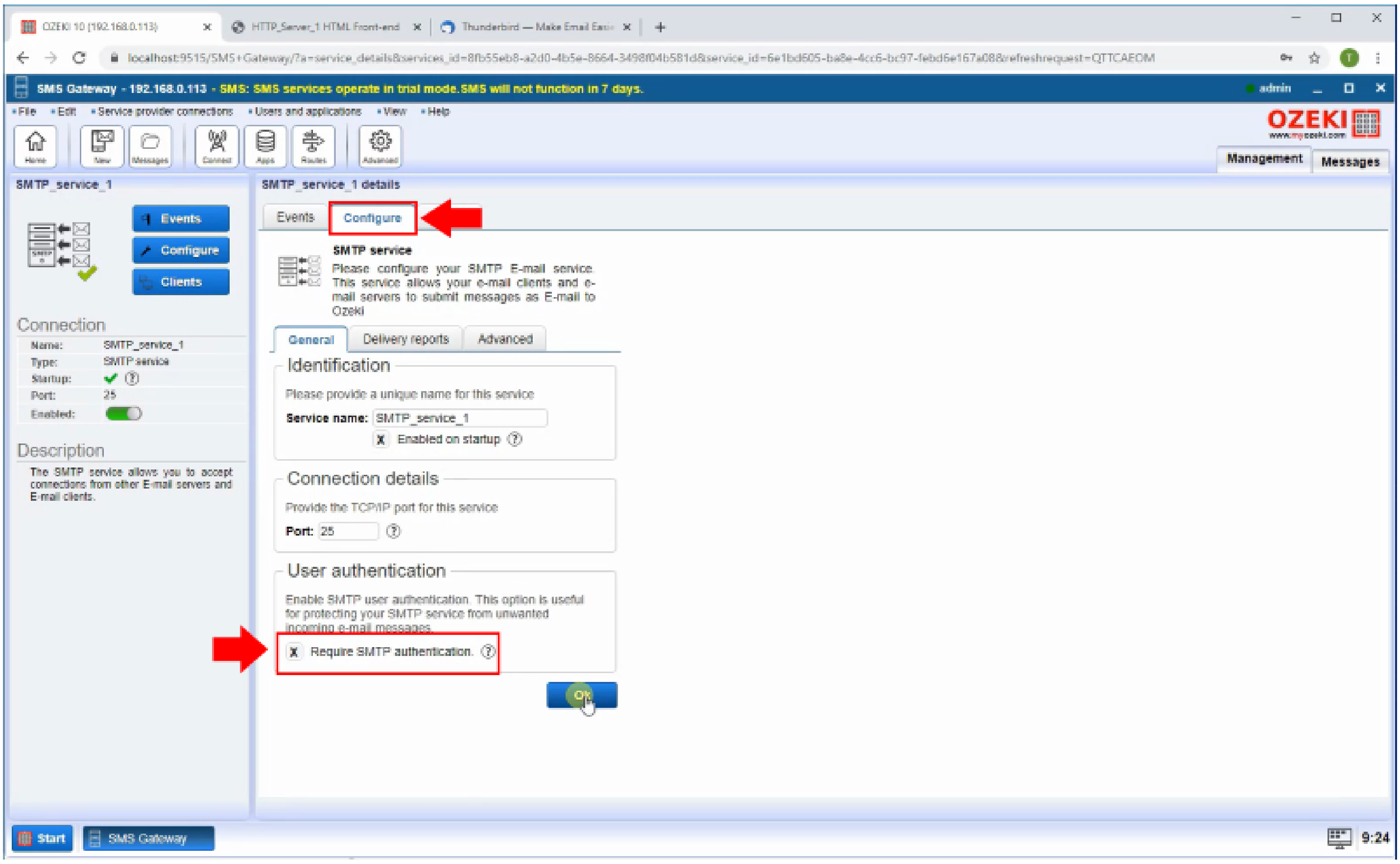
## Services

| N° | Service name | Enabled | Details |
|---|---|---|---|
| 1. | webserver_GUI_1 | | Details |
| 2. | SMTP_service_1 | ● | Details |
| 3. | POP3_service_1 | ● | Details |

0/3 item selected

Figure 6 - SMTP server details

On the SMTP server Details page, select the Configure tab and enable 'Require SMTP authentication' in the User Authentication section as you can see in the Figure 7.



SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.

SMTP_service_1

Connection

| | |
|---|---|
| Name: | SMTP_service_1 |
| Type: | SMTP service |
| Startup: | ✓ |
| Port: | 25 |
| Enabled: | ● |

Description

The SMTP service allows you to accept connections from other E-mail servers and E-mail clients.

SMTP_service_1 details

**SMTP service**
Please configure your SMTP E-mail service. This service allows your e-mail clients and e-mail servers to submit messages as E-mail to Ozeki

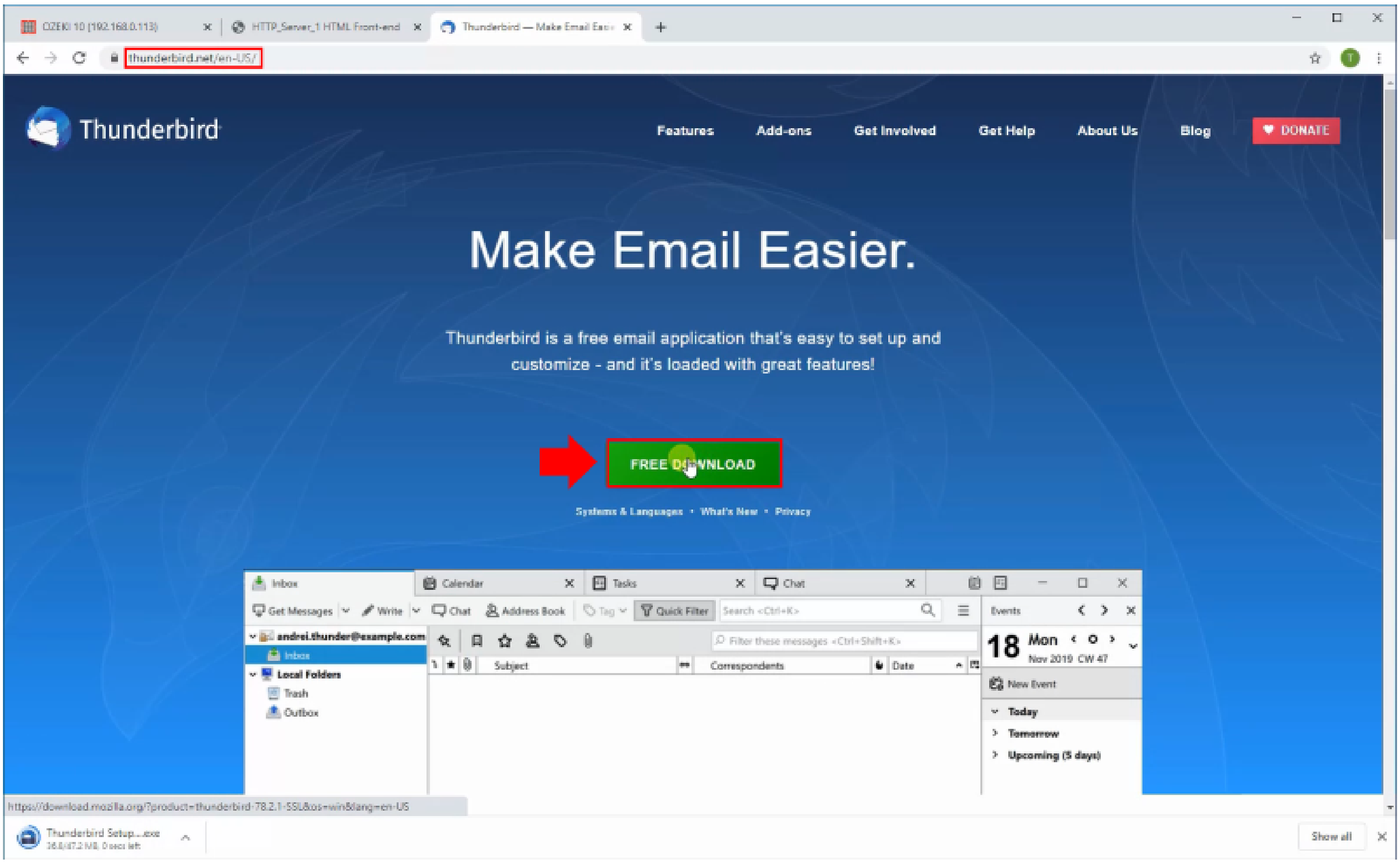
Identification
Please provide a unique name for this service
**Service name:** SMTP_service_1
☒ Enabled on startup

Connection details
Provide the TCP/IP port for this service
**Port:** 25

User authentication
Enable SMTP user authentication. This option is useful for protecting your SMTP service from unwanted incoming e-mail messages.
☒ Require SMTP authentication.

Figure 7 - Enable Require Authentication

Now download the latest version of Thunderbird from tunderbird.net (Figure 8).



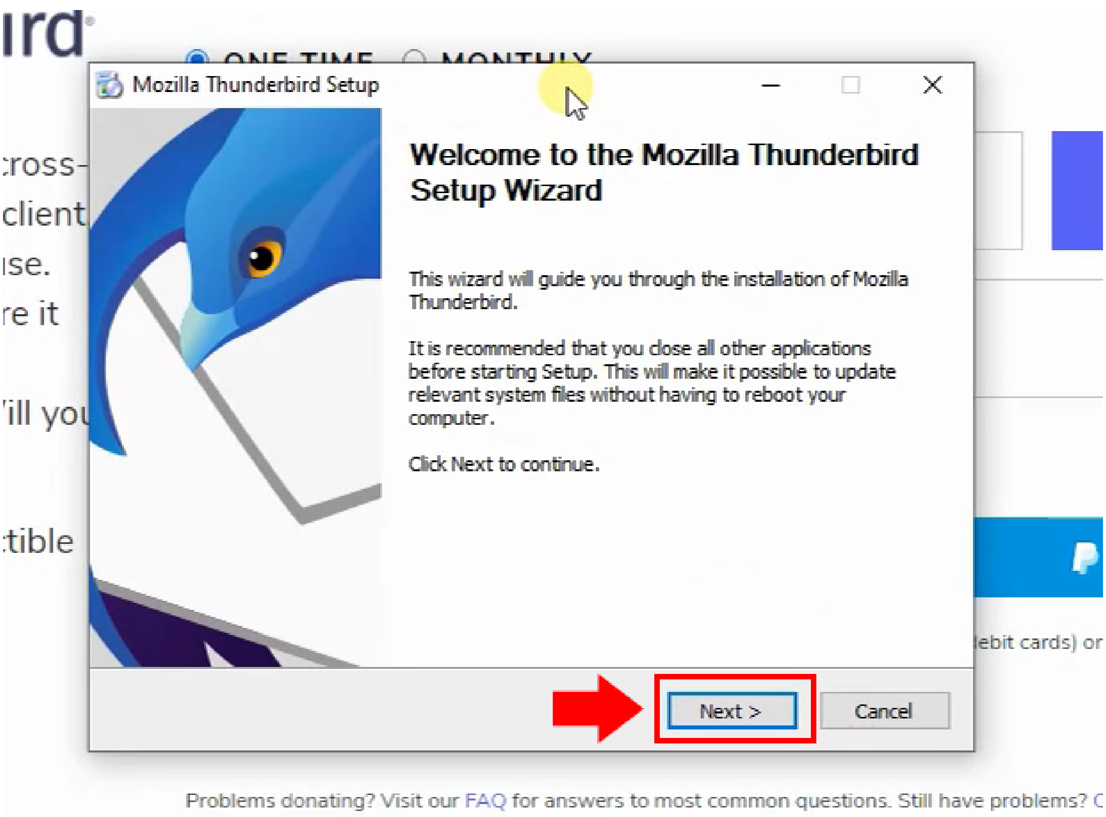
Make Email Easier.

Thunderbird is a free email application that's easy to set up and customize - and it's loaded with great features!

https://download.mozilla.org/?product=thunderbird-78.2.1-SSL&os=win&lang=en-US

Figure 8 - Download Mozilla Thunderbird

Open the installer and click the 'Next' button on the Welcome page to begin the installation (Figure 9).



Welcome to the Mozilla Thunderbird Setup Wizard

This wizard will guide you through the installation of Mozilla Thunderbird.

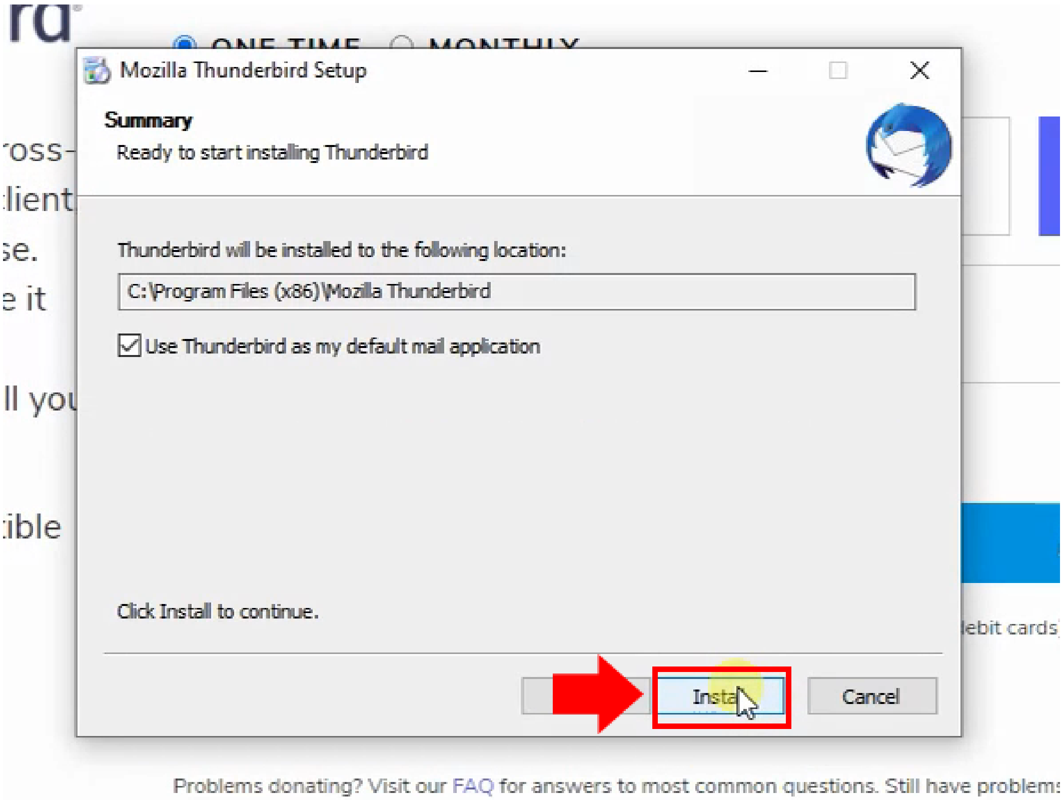
It is recommended that you close all other applications before starting Setup. This will make it possible to update relevant system files without having to reboot your computer.

Click Next to continue.

Problems donating? Visit our FAQ for answers to most common questions. Still have problems? C

Figure 9 - Install Mozilla Thunderbird

Select the installation location and start the installation by clicking the Install button (Figure 10).



Mozilla Thunderbird Setup

**Summary**

Ready to start installing Thunderbird

Thunderbird will be installed to the following location:

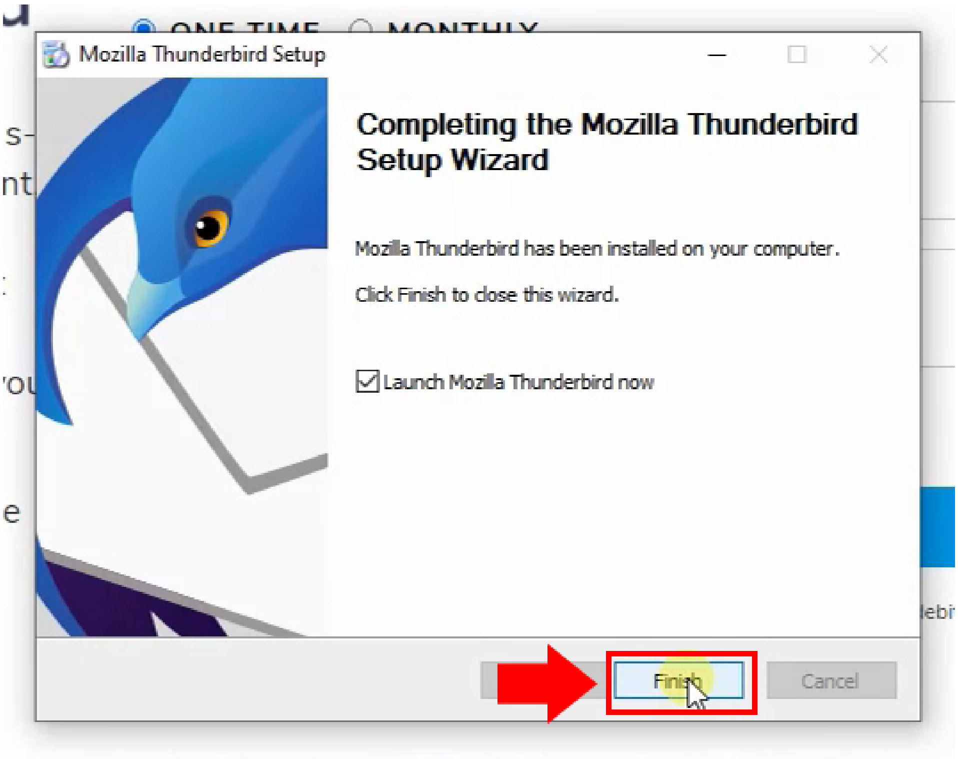
C:\Program Files (x86)\Mozilla Thunderbird

☑ Use Thunderbird as my default mail application

Click Install to continue.

Figure 10 - Start Thunderbird installation

When the installation is complete, click Finish button and Thundrebird will start as the Figure 11 shows.



Mozilla Thunderbird Setup

**Completing the Mozilla Thunderbird Setup Wizard**

Mozilla Thunderbird has been installed on your computer.

Click Finish to close this wizard.

☑ Launch Mozilla Thunderbird now

Figure 11 - Installation Finished

Now you need to create a new email account in Tunderbird. Select New section in the Menu and then the Existing Mail Account option (Figure 12).

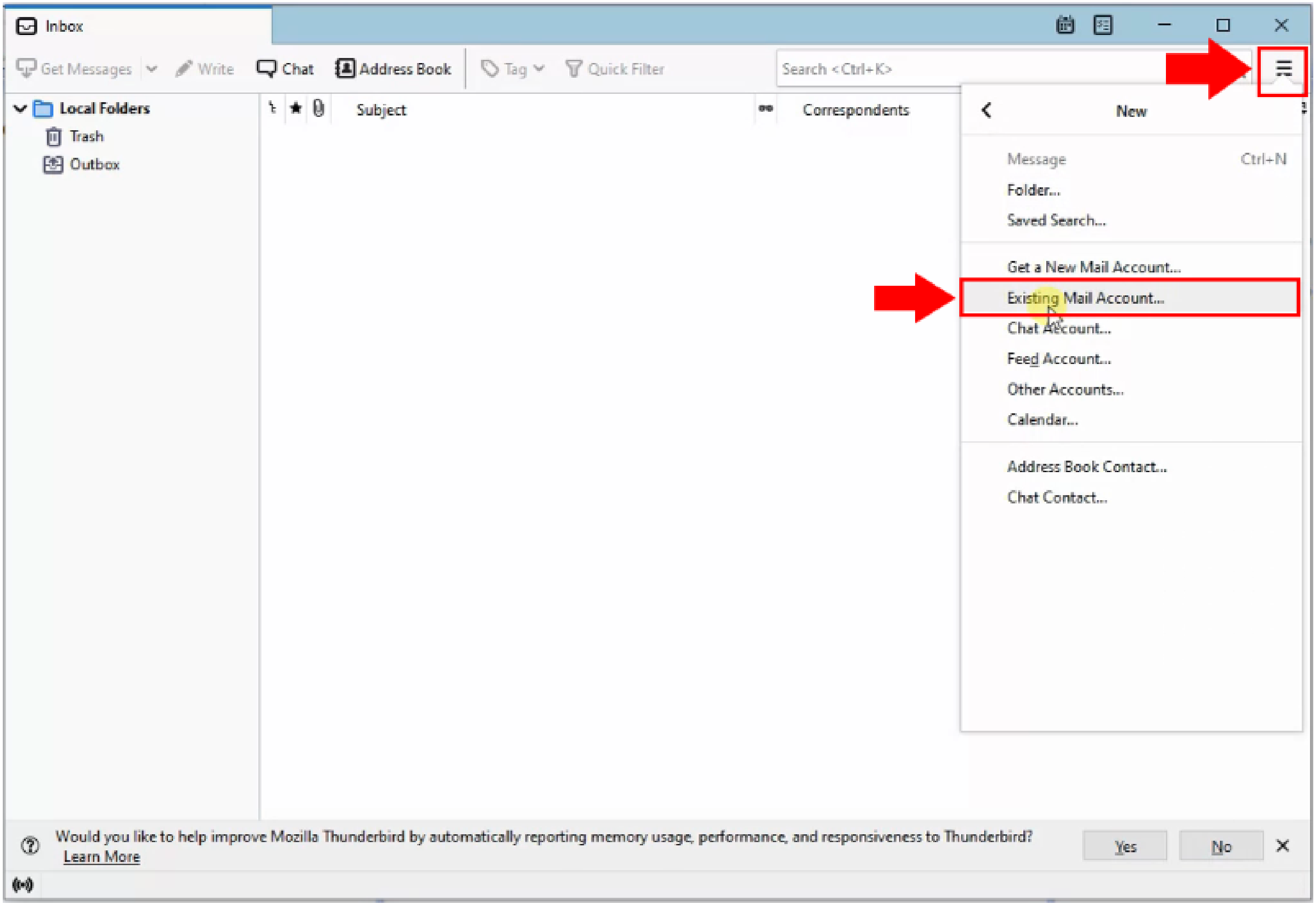
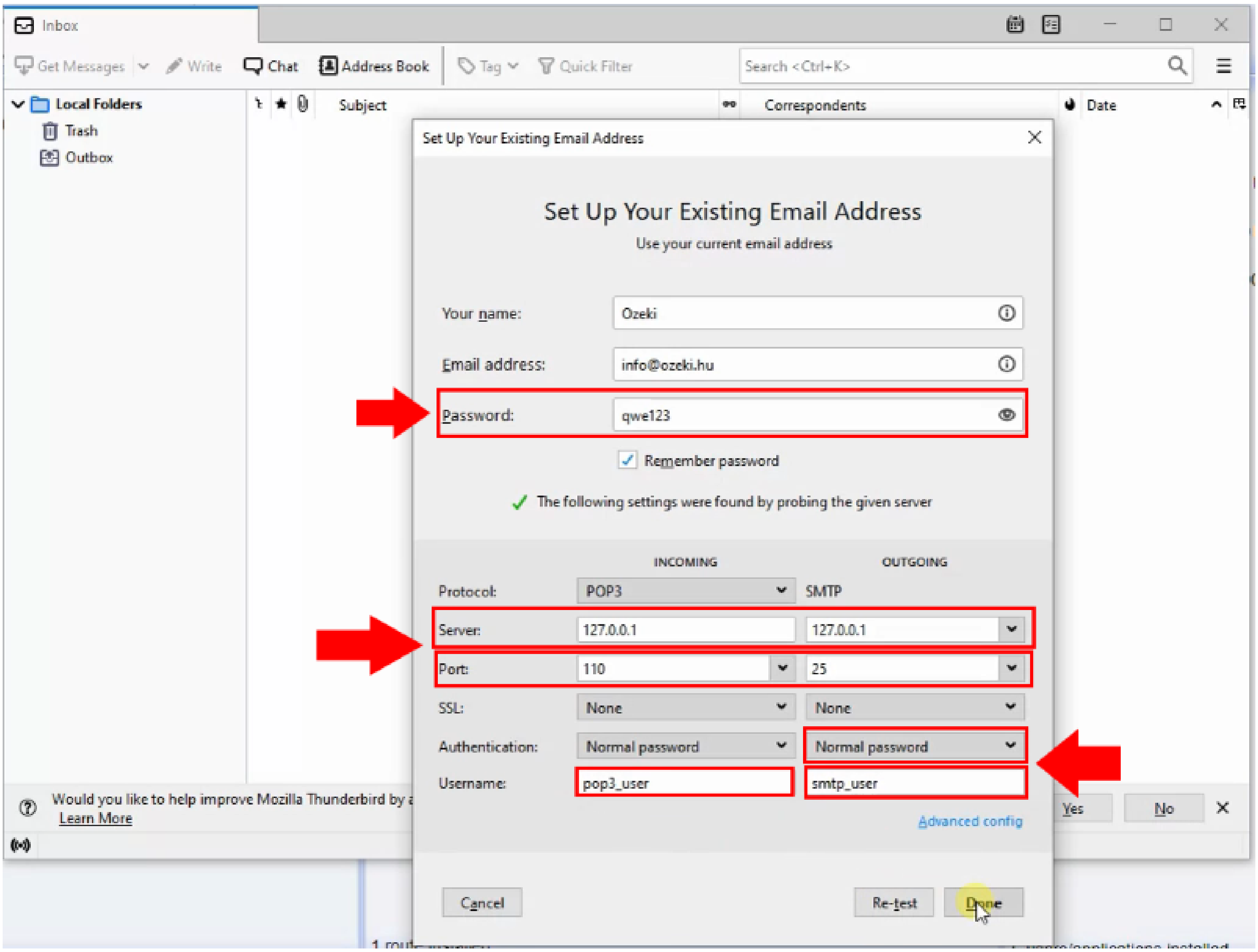


Figure 12 - Add new mail account

Then enter your name and email address and then the password you used for SMTP User. In the Advanced configure section. Enter the IP address of the SMS Gateway on the server and the default POP3 and SMTP ports on the port, which are 110 and 25. Finally, for SMTP, select Normal Passoword for Authentication. Username should be SMTP and POP3 user names (Figure 13).



Set Up Your Existing Email Address

Set Up Your Existing Email Address

Use your current email address

Your name: Ozeki

Email address: info@ozeki.hu

Password: qwe123

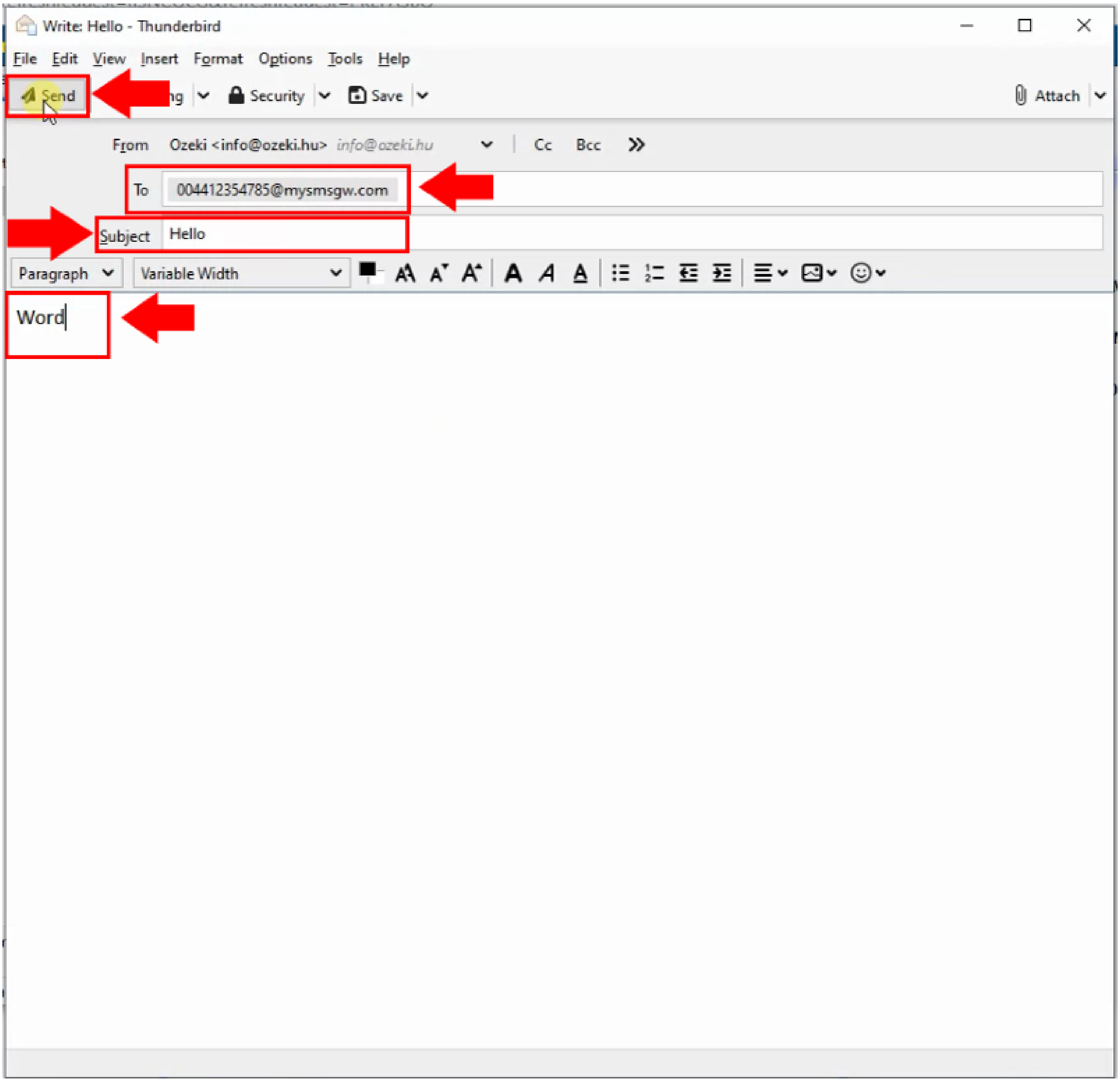
☑ Remember password

✓ The following settings were found by probing the given server

| | INCOMING | OUTGOING |
|---|---|---|
| Protocol: | POP3 | SMTP |
| Server: | 127.0.0.1 | 127.0.0.1 |
| Port: | 110 | 25 |
| SSL: | None | None |
| Authentication: | Normal password | Normal password |
| Username: | pop3_user | smtp_user |

Figure 13 - Setup e-mail details

The next step is to send a message from Thunderbird to the SMS Gateway. To do this, select the Write menu. You can enter the recipient in the form phone number@domain where the phone number is the recipient's phone number and the domain is the SMS Gateway domain name. Both the subject and the body part will be sent in the SMS (Figure 14).



Write: Hello - Thunderbird
File Edit View Insert Format Options Tools Help
Send
Security
Save
Attach
Paragraph
Variable Width
Cc
Bcc

From Ozeki <info@ozeki.hu> info@ozeki.hu

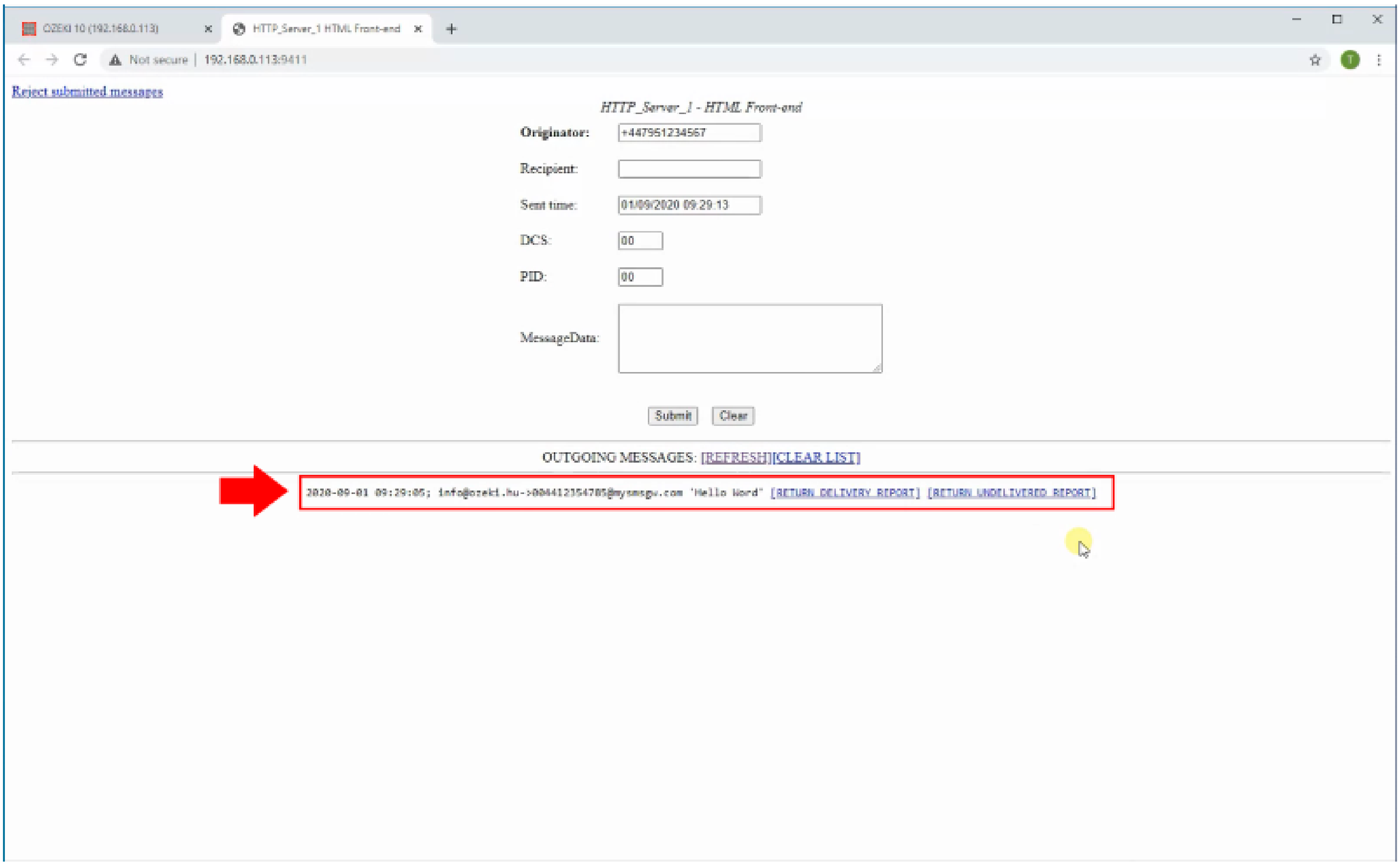
To 004412354785@mysmsgw.com

Subject Hello

Word

Figure 14 - Send Message

You will see that both parts of the message appear on the HTTP server as the Figure 15 shows.


OZEKI 10 (192.168.0.113)
HTTP_Server_1 HTML Front-end
Not secure | 192.168.0.113:9411
Reject submitted messages
Submit
Clear

*HTTP_Server_1 - HTML Front-end*

**Originator:** +447951234567

Recipient:

Sent time: 01/09/2020 09:29:13

DCS: 00

PID: 00

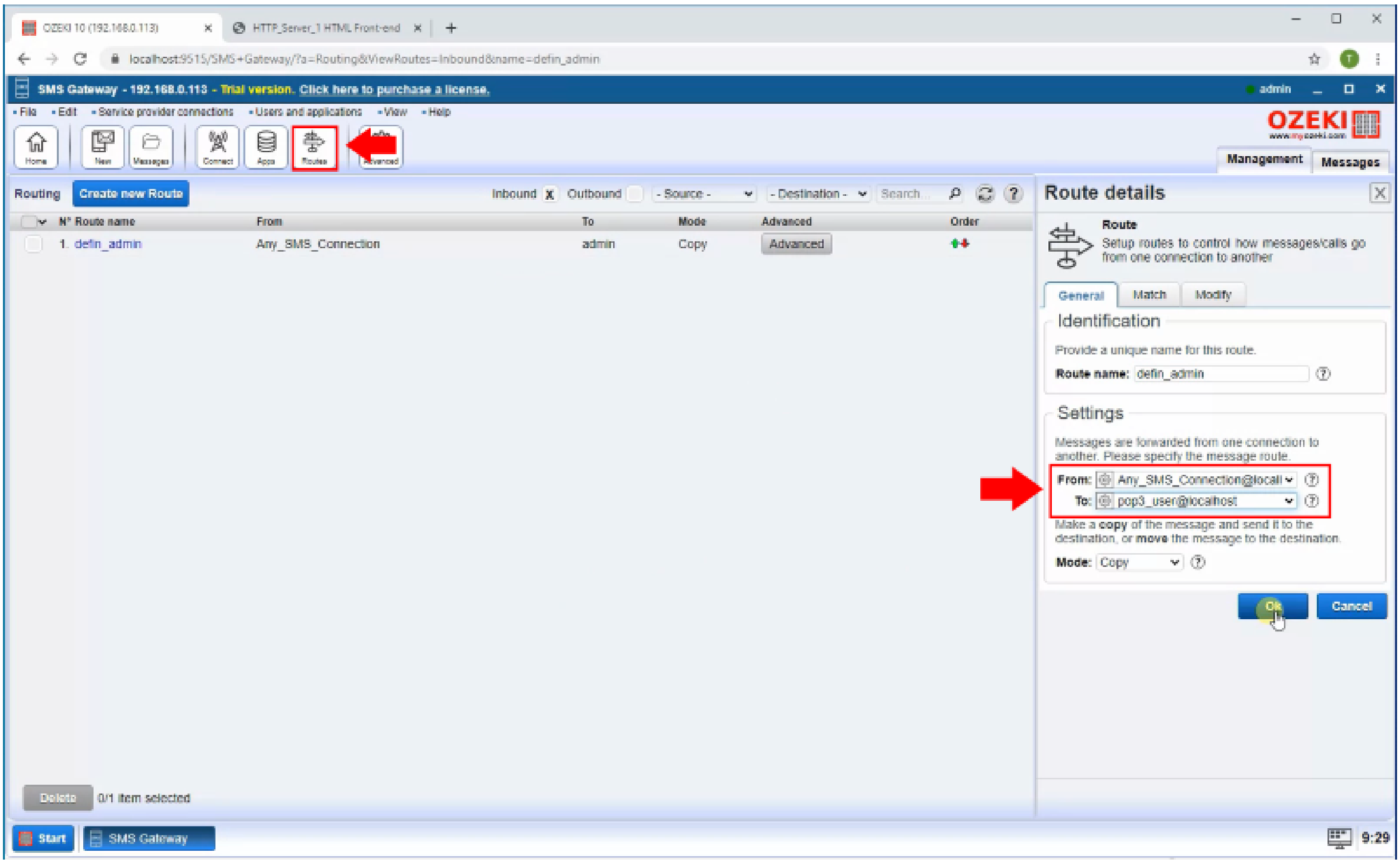
MessageData:

OUTGOING MESSAGES: [REFRESH][CLEAR LIST]

```
2020-09-01 09:29:05; info@ozeki.hu->004412354705@mysmsgw.com 'Hello Word' [RETURN DELIVERY REPORT] [RETURN UNDELIVERED REPORT]
```

Figure 15 - Message Received by the HTTP Server

In order to receive a message on the POP3 user, the routing table must be set to forward incoming messages here. Open the routing menu and for the TO connection, select the POP3 user (Figure 16).



| N° | Route name | From | To | Mode | Advanced | Order |
|---|---|---|---|---|---|---|
| 1. | defin_admin | Any_SMS_Connection | admin | Copy | Advanced | |

Route details

**Route**
Setup routes to control how messages/calls go from one connection to another

Identification

Provide a unique name for this route.

**Route name:** defin_admin

Settings

Messages are forwarded from one connection to another. Please specify the message route.

**From:** Any_SMS_Connection@locall

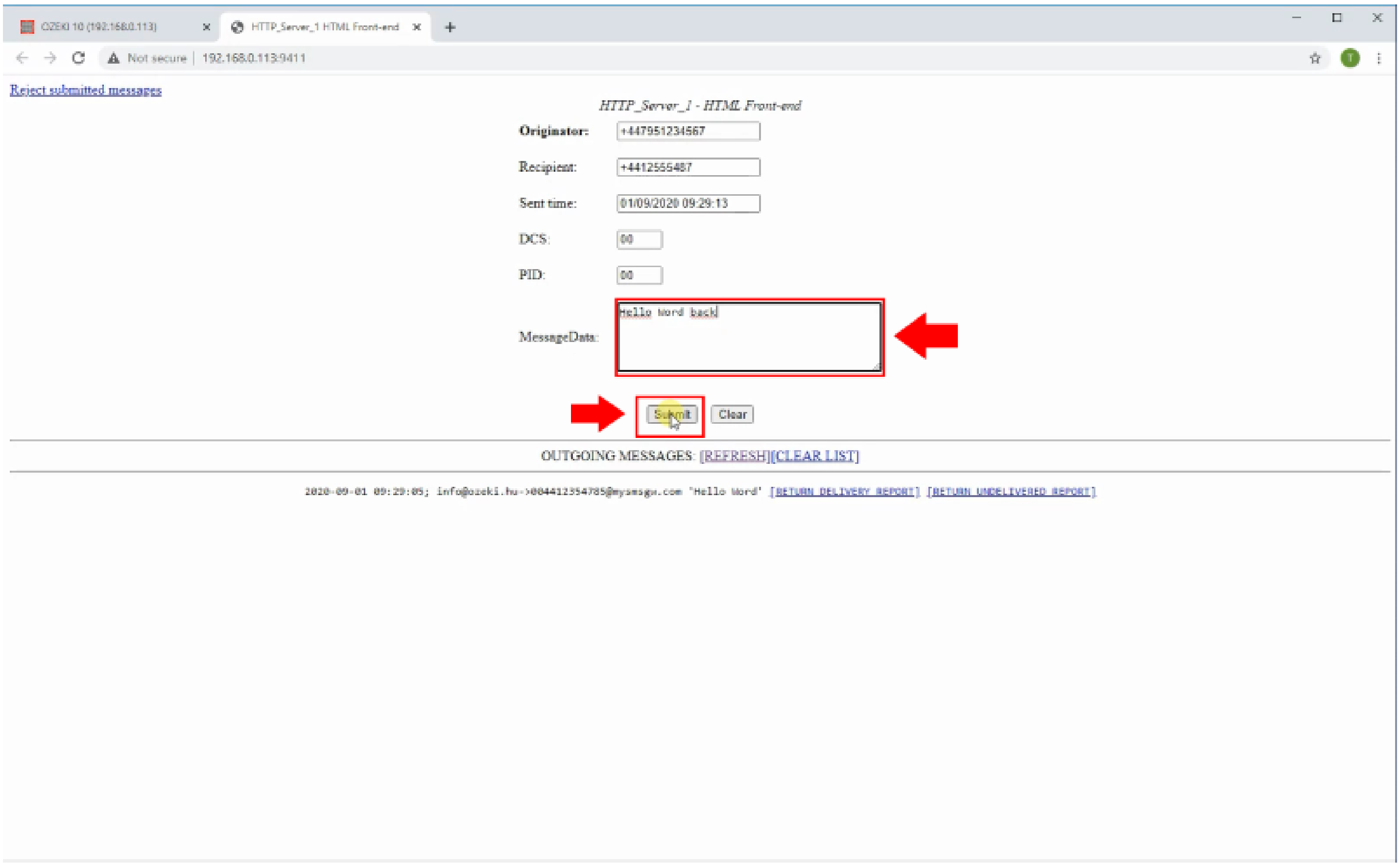
**To:** pop3_user@localhost

Make a **copy** of the message and send it to the destination, or **move** the message to the destination.

**Mode:** Copy

Figure 16 - Edit incoming Route

Now send a message back from the HTTP server to the SMS Gateway.


OZEKI 10 (192.168.0.113)
HTTP_Server_1 HTML Front-end
Not secure | 192.168.0.113:9411
Reject submitted messages

*HTTP_Server_1 - HTML Front-end*

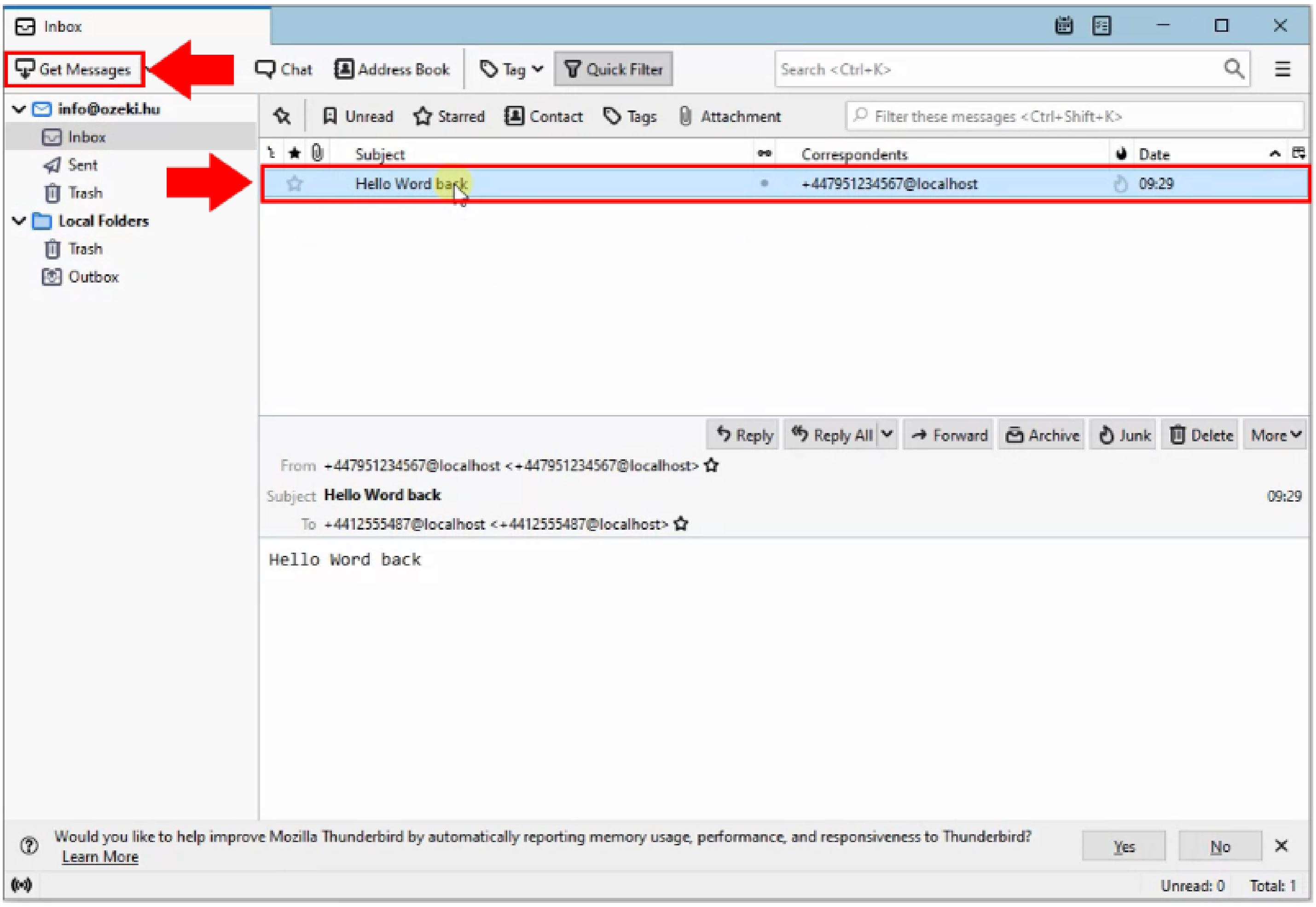
| | |
|---|---|
| **Originator:** | +447951234567 |
| Recipient: | +4412555487 |
| Sent time: | 01/09/2020 09:29:13 |
| DCS: | 00 |
| PID: | 00 |
| MessageData: | Hello Word back |

OUTGOING MESSAGES: [REFRESH][CLEAR LIST]

```
2020-09-01 09:29:05; info@ozeki.hu->004412354785@mysmsgw.com 'Hello Word' [RETURN DELIVERY REPORT] [RETURN UNDELIVERED REPORT]
```

Figure 17 - Send Message back to Thunderbird

Finally, if you press the Get Messages button in Thunderbird you will see that the message arrives in your inbox as you can see in the Figure 18.



| Subject | Correspondents | Date |
|---|---|---|
| Hello Word back | +447951234567@localhost | 09:29 |

From +447951234567@localhost <+447951234567@localhost>

Subject **Hello Word back** 09:29

To +4412555487@localhost <+4412555487@localhost>

```
Hello Word back
```

Would you like to help improve Mozilla Thunderbird by automatically reporting memory usage, performance, and responsiveness to Thunderbird?
Learn More

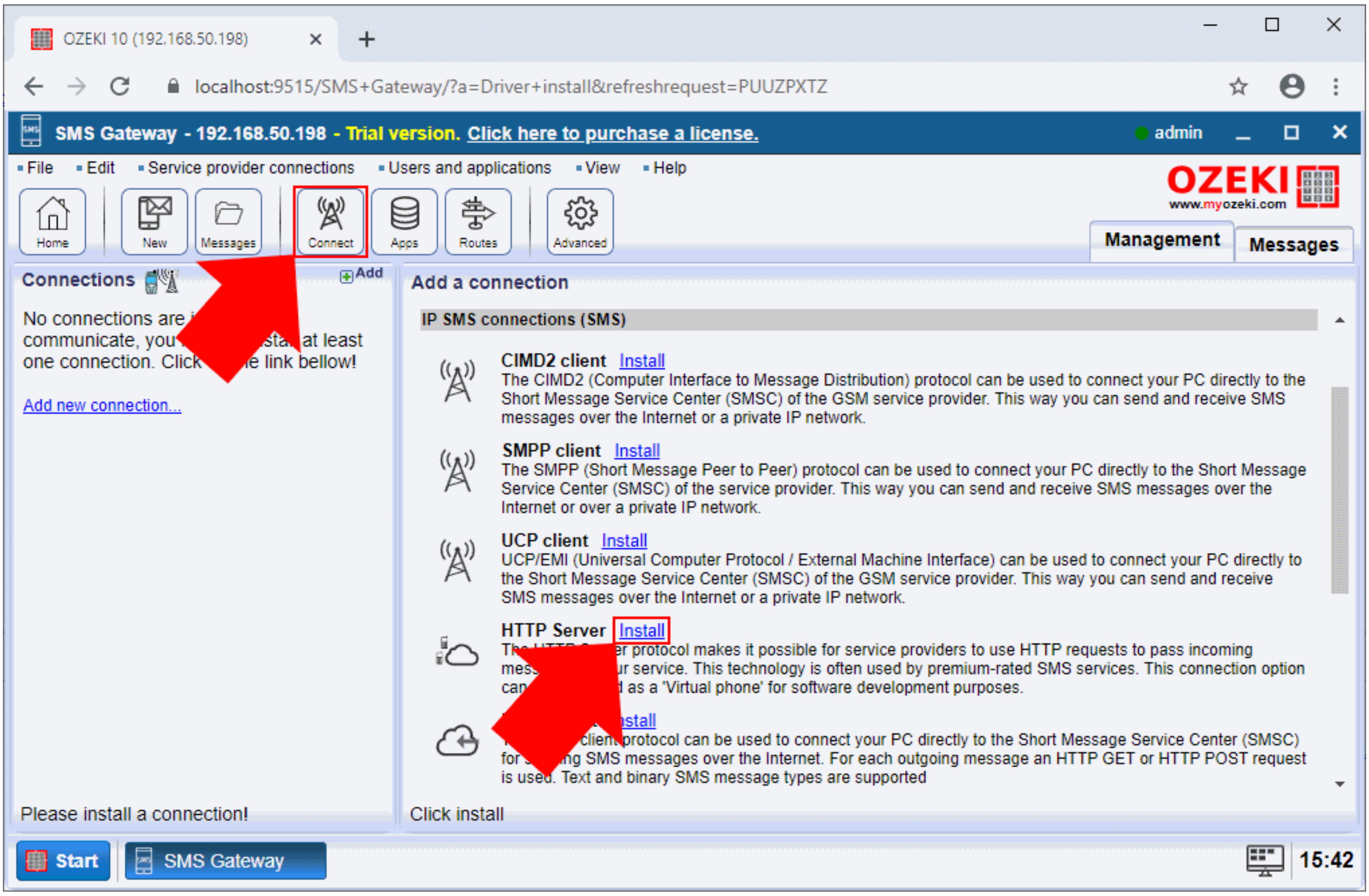
Figure 18 - Message Received by the Thunderbird

# How to send SMS from Windows mail

***The following document is about to demonstrate how you can configure your Ozeki SMS Gateway and Windows Mail to be able to send or receive SMS messages. By performing this configuration on your system, you will be allowed to use e-mail to SMS and SMS to e-mail forwarding. The step by step guide on this page shows you what connections that you need to create in SMS Gateway and how you need to configure Windows Mail to be able to forward e-mails and SMS messages. The guide does not take more than 10 minutes to complete, so let's start right now!***

## Step 1 - Create HTTP server connection

The first step of this guide is to create a HTTP Server connection. This connection is going to help us to send and receive SMS messages to test if the forwarding working well to both directions. So, after you opened SMS Gateway, click on the icon of Connect on the toolbar (Figure 1), and here from the list, click on the Install button of the HTTP Server connection.


OZEKI 10 (192.168.50.198)
localhost:9515/SMS+Gateway/?a=Driver+install&refreshrequest=PUUZPXTZ
SMS Gateway - 192.168.50.198 - Trial version. Click here to purchase a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
OZEKI
www.myozeki.com
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Connections
Add
Add a connection
IP SMS connections (SMS)
Start
SMS Gateway
15:42

No connections are i... communicate, you ... sta... at least one connection. Click ... e link bellow!

Add new connection...

CIMD2 client Install
The CIMD2 (Computer Interface to Message Distribution) protocol can be used to connect your PC directly to the Short Message Service Center (SMSC) of the GSM service provider. This way you can send and receive SMS messages over the Internet or a private IP network.

SMPP client Install
The SMPP (Short Message Peer to Peer) protocol can be used to connect your PC directly to the Short Message Service Center (SMSC) of the service provider. This way you can send and receive SMS messages over the Internet or over a private IP network.

UCP client Install
UCP/EMI (Universal Computer Protocol / External Machine Interface) can be used to connect your PC directly to the Short Message Service Center (SMSC) of the GSM service provider. This way you can send and receive SMS messages over the Internet or a private IP network.

HTTP Server Install
The HTTP S... er protocol makes it possible for service providers to use HTTP requests to pass incoming mes... ur service. This technology is often used by premium-rated SMS services. This connection option can ... l as a 'Virtual phone' for software development purposes.

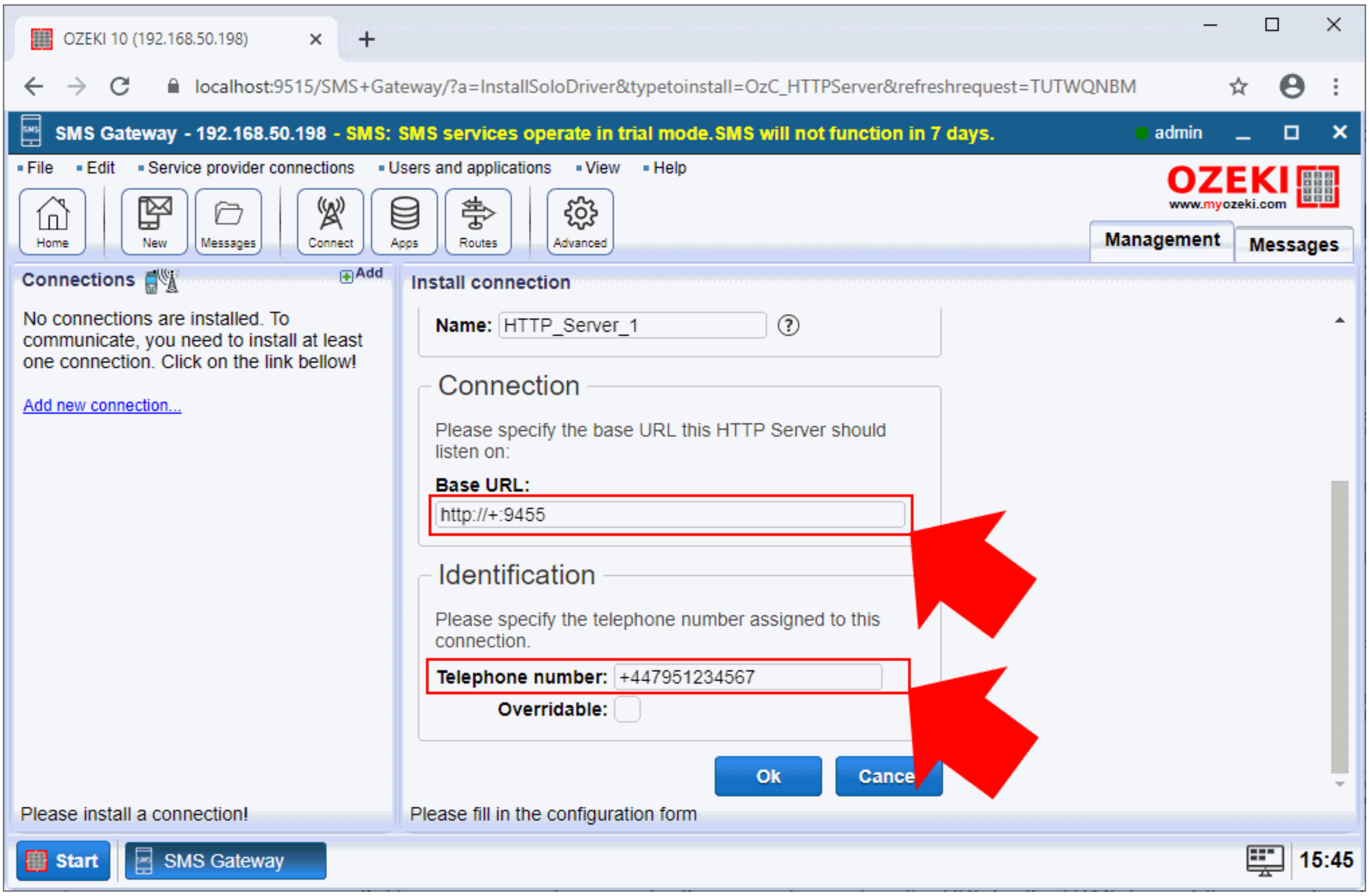
...stall
... client protocol can be used to connect your PC directly to the Short Message Service Center (SMSC) for ... ng SMS messages over the Internet. For each outgoing message an HTTP GET or HTTP POST request is used. Text and binary SMS message types are supported

Please install a connection!

Click install

Figure 1 - Select the HTTP Server connection

Next, you will be able to see the configuration menu of the HTTP Server connection as you can see it in Figure 2. Here, you can set a name for the connection, set up the URL for the HTML form of the connection. You will need to use this form to handle the SMS messages. Lastly, you can specify a phone number for this connection. If you finished with the configuration, just click on OK to create the HTTP Server connection.


OZEKI 10 (192.168.50.198)
localhost:9515/SMS+Gateway/?a=InstallSoloDriver&typetoinstall=OzC_HTTPServer&refreshrequest=TUTWQNBM
SMS Gateway - 192.168.50.198 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
OZEKI
www.myozeki.com
Management
Messages
Connections
Add
Start
SMS Gateway
15:45

No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

Please install a connection!

Install connection

Name: HTTP_Server_1

Connection

Please specify the base URL this HTTP Server should listen on:

Base URL: http://+:9455

Identification

Please specify the telephone number assigned to this connection.

Telephone number: +447951234567

Overridable: □

Ok

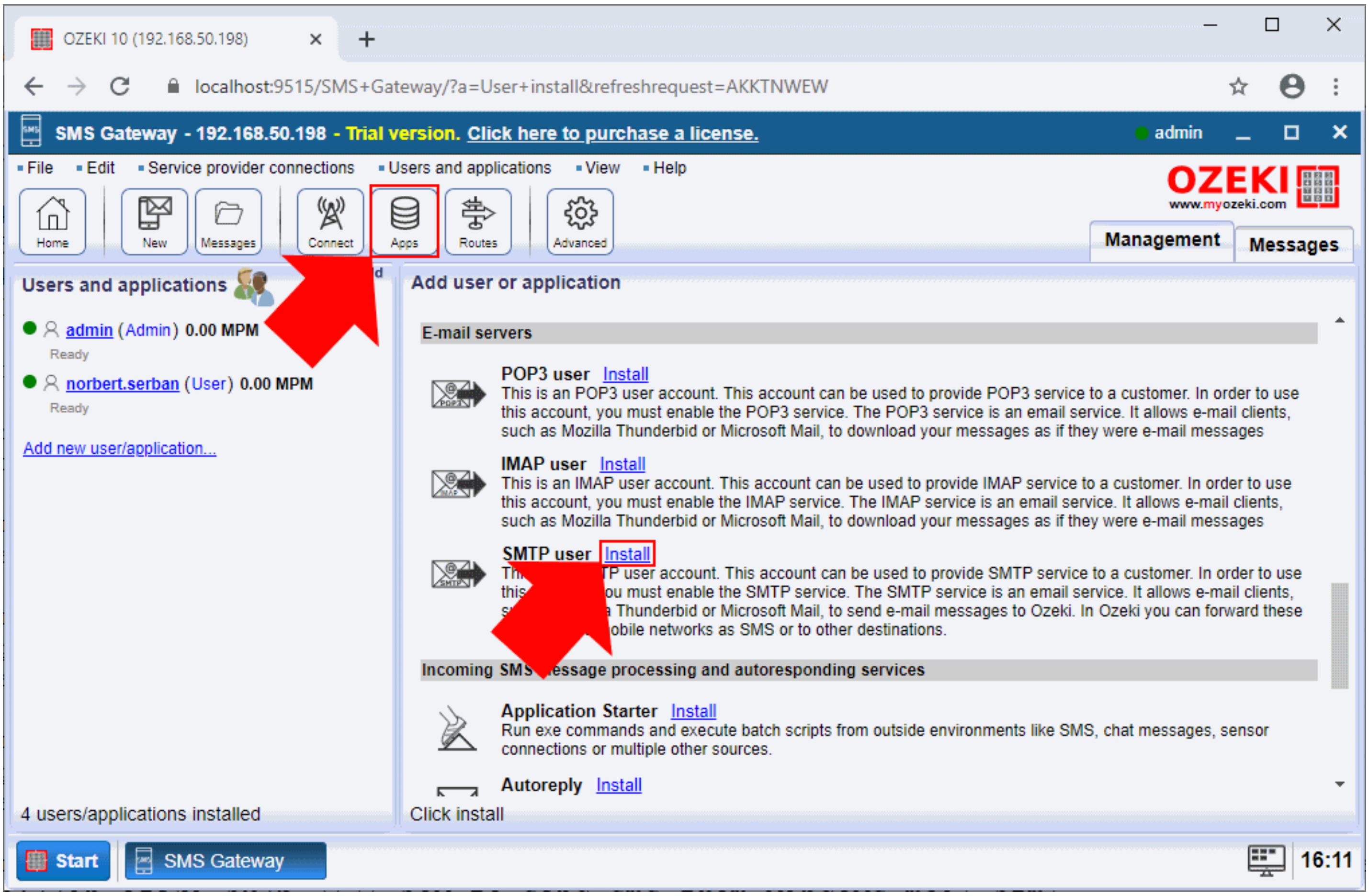
Cancel

Please fill in the configuration form

Figure 2 - Configure the HTTP Server connection

## Step 2 - Install SMTP User connection

Next, you need to set up a SMTP user connection in Ozeki SMS Gateway in order to receive the e-mail messages from the Windows Mail application. To create a new user connection in SMS Gateway, first, you need to open the Apps menu by clicking on its icon on the toolbar. Next, scrool down to the SMTP user connection, and like in Figure 3, click on the Install button.


OZEKI 10 (192.168.50.198)
localhost:9515/SMS+Gateway/?a=User+install&refreshrequest=AKKTNWEW
SMS Gateway - 192.168.50.198 - Trial version. Click here to purchase a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Users and applications
admin (Admin) 0.00 MPM
Ready
norbert.serban (User) 0.00 MPM
Ready
Add new user/application...
4 users/applications installed
Start
SMS Gateway
16:11

Add user or application

E-mail servers

POP3 user Install
This is an POP3 user account. This account can be used to provide POP3 service to a customer. In order to use this account, you must enable the POP3 service. The POP3 service is an email service. It allows e-mail clients, such as Mozilla Thunderbid or Microsoft Mail, to download your messages as if they were e-mail messages

IMAP user Install
This is an IMAP user account. This account can be used to provide IMAP service to a customer. In order to use this account, you must enable the IMAP service. The IMAP service is an email service. It allows e-mail clients, such as Mozilla Thunderbid or Microsoft Mail, to download your messages as if they were e-mail messages

SMTP user Install
This is an SMTP user account. This account can be used to provide SMTP service to a customer. In order to use this account, you must enable the SMTP service. The SMTP service is an email service. It allows e-mail clients, such as Mozilla Thunderbid or Microsoft Mail, to send e-mail messages to Ozeki. In Ozeki you can forward these messages to mobile networks as SMS or to other destinations.

Incoming SMS message processing and autoresponding services

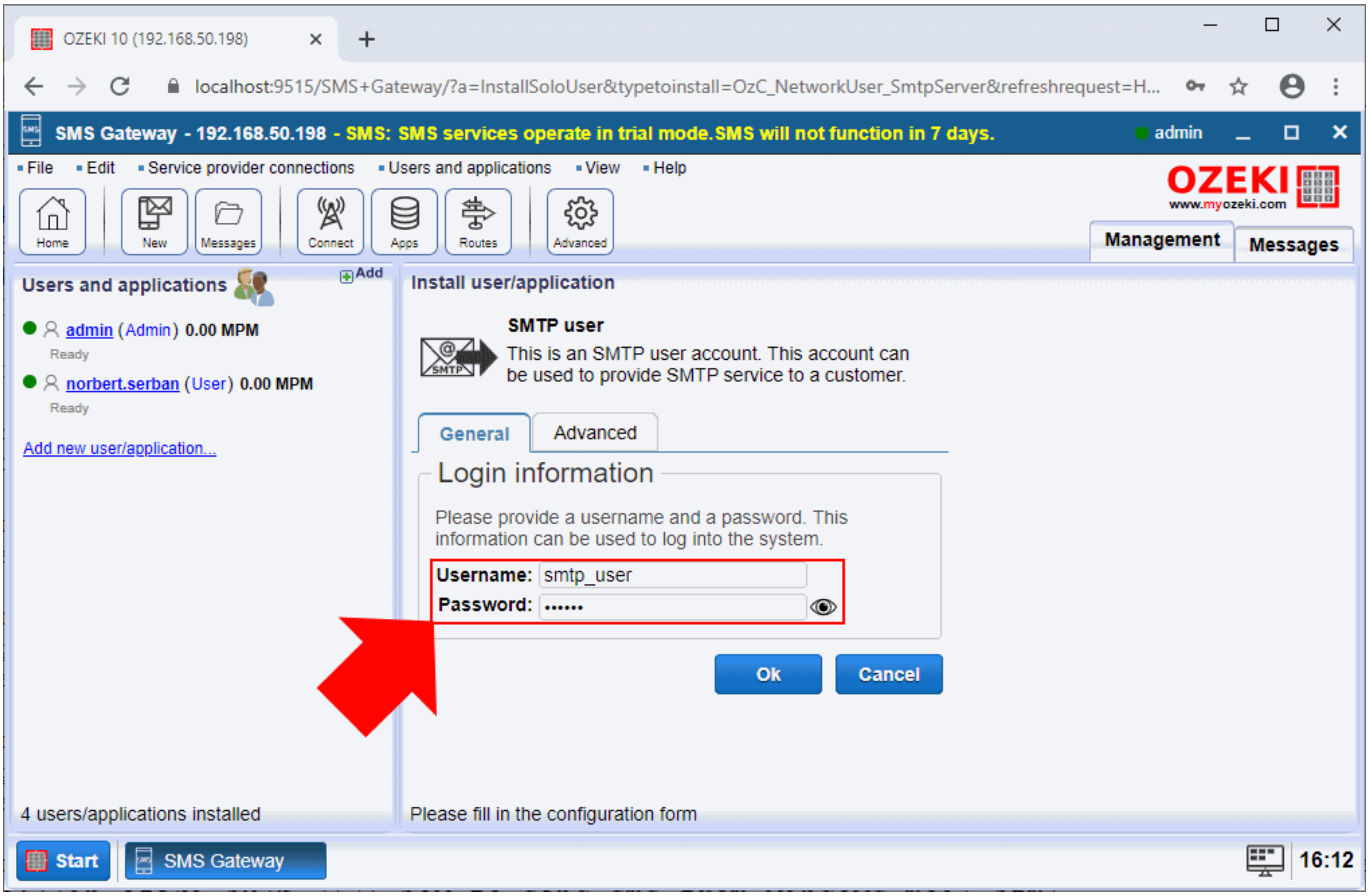
Application Starter Install
Run exe commands and execute batch scripts from outside environments like SMS, chat messages, sensor connections or multiple other sources.

Autoreply Install

Click install

Figure 3 - Select the SMTP User connection

The configuration of the SMTP User connection does not require any specific knowledge from your side. All you need to do here is to specify a username and a password for the SMTP user connection as Figure 4 demonstrates it. After you finished with the configuration, just click Ok to create the SMTP user connection.


OZEKI 10 (192.168.50.198)
localhost:9515/SMS+Gateway/?a=InstallSoloUser&typetoinstall=OzC_NetworkUser_SmtpServer&refreshrequest=H...
SMS Gateway - 192.168.50.198 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Users and applications
Add
admin (Admin) 0.00 MPM
Ready
norbert.serban (User) 0.00 MPM
Ready
Add new user/application...
4 users/applications installed
Start
SMS Gateway
16:12

Install user/application

SMTP user
This is an SMTP user account. This account can be used to provide SMTP service to a customer.

General | Advanced

Login information

Please provide a username and a password. This information can be used to log into the system.

Username: smtp_user
Password: ••••••

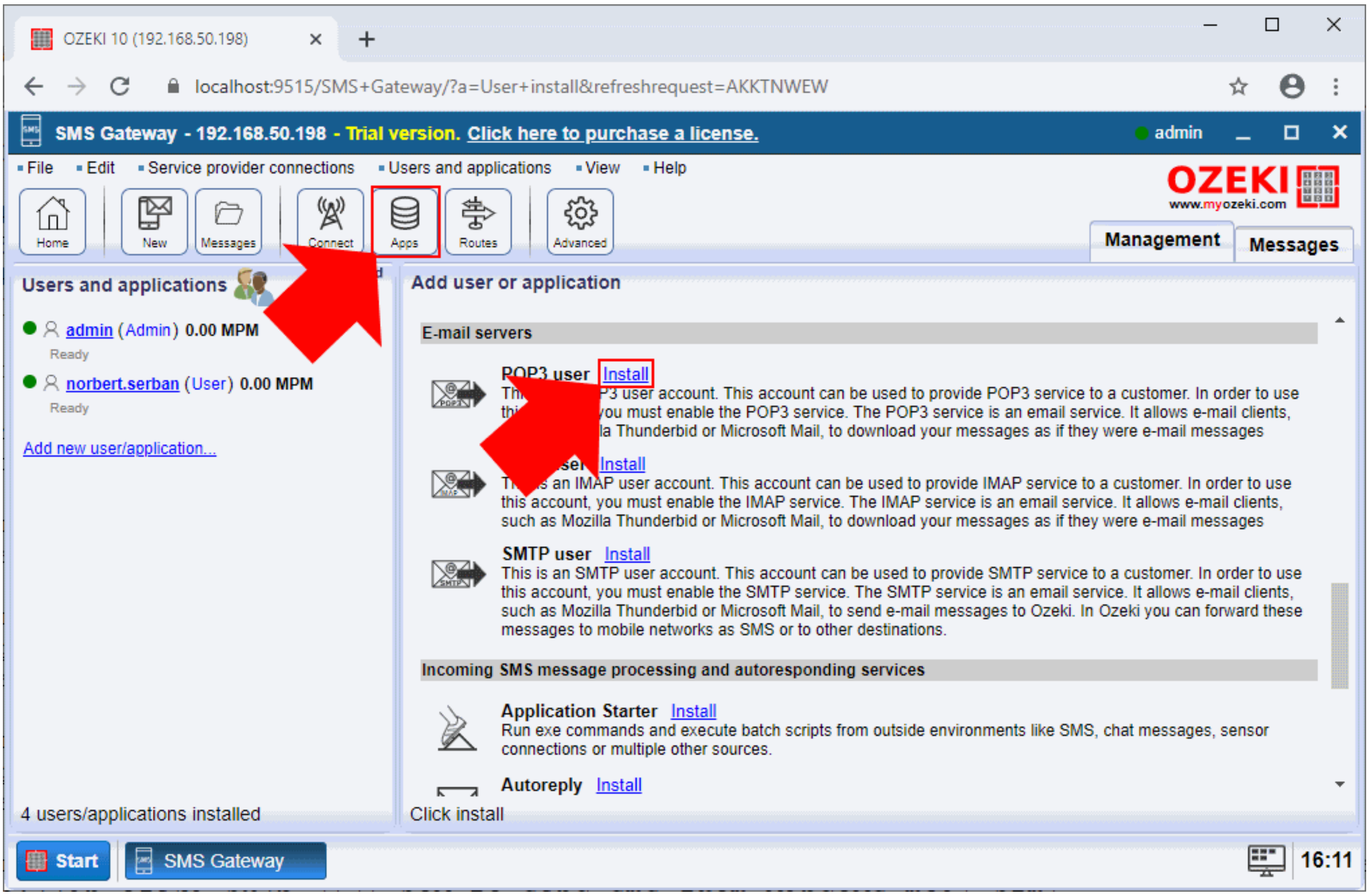
Ok Cancel

Please fill in the configuration form

Figure 4 - Add username and password for the connection

# Step 3 - Install POP3 User connection

The next step of the guide is to install a POP user connection on your SMS Gateway. This POP3 user connection allows you to send the SMS messages received by the SMS Gateway to the Windows Mail application. To install this connection, open the Apps menu by clicking on its icon on the toolbar, and next, as you can see it in Figure 5, select the POP3 user from the list, by clicking on the Install button.



OZEKI 10 (192.168.50.198)
localhost:9515/SMS+Gateway/?a=User+install&refreshrequest=AKKTNWEW
SMS Gateway - 192.168.50.198 - Trial version. Click here to purchase a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
OZEKI
www.myozeki.com
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Start
SMS Gateway
16:11

Users and applications

admin (Admin) 0.00 MPM
Ready

norbert.serban (User) 0.00 MPM
Ready

Add new user/application...

4 users/applications installed

Add user or application

E-mail servers

POP3 user Install
Th 3 user account. This account can be used to provide POP3 service to a customer. In order to use thi ou must enable the POP3 service. The POP3 service is an email service. It allows e-mail clients, la Thunderbird or Microsoft Mail, to download your messages as if they were e-mail messages

ser Install
T s an IMAP user account. This account can be used to provide IMAP service to a customer. In order to use this account, you must enable the IMAP service. The IMAP service is an email service. It allows e-mail clients, such as Mozilla Thunderbird or Microsoft Mail, to download your messages as if they were e-mail messages

SMTP user Install
This is an SMTP user account. This account can be used to provide SMTP service to a customer. In order to use this account, you must enable the SMTP service. The SMTP service is an email service. It allows e-mail clients, such as Mozilla Thunderbird or Microsoft Mail, to send e-mail messages to Ozeki. In Ozeki you can forward these messages to mobile networks as SMS or to other destinations.

Incoming SMS message processing and autoresponding services

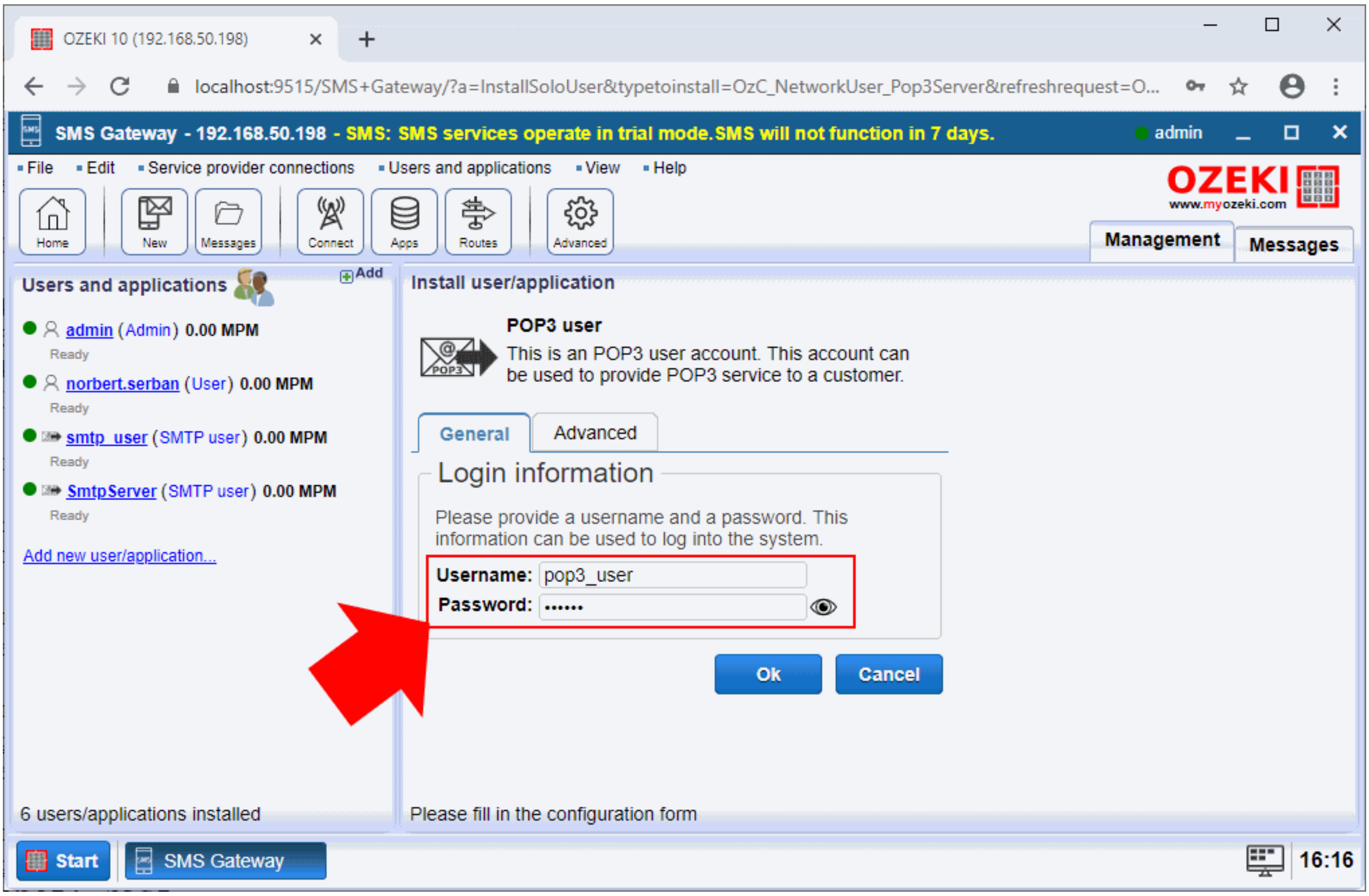
Application Starter Install
Run exe commands and execute batch scripts from outside environments like SMS, chat messages, sensor connections or multiple other sources.

Autoreply Install

Click install

Figure 5 - Select the POP3 user connection

After you clicked on the Install button, the next windows that shows up, is the configuration menu of the POP3 user connection. Here, you need to enter a unique username in the Username field and a password for the user in the Password field like in Figure 6. Lastly, just click on Ok to create the POP3 user connection.


OZEKI 10 (192.168.50.198)
localhost:9515/SMS+Gateway/?a=InstallSoloUser&typetoinstall=OzC_NetworkUser_Pop3Server&refreshrequest=O...
SMS Gateway - 192.168.50.198 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Users and applications
Add
admin (Admin) 0.00 MPM
Ready
norbert.serban (User) 0.00 MPM
Ready
smtp_user (SMTP user) 0.00 MPM
Ready
SmtpServer (SMTP user) 0.00 MPM
Ready
Add new user/application...
6 users/applications installed
Install user/application
General
Advanced
Ok
Cancel
Please fill in the configuration form
Start
SMS Gateway
16:16

**POP3 user**

This is an POP3 user account. This account can be used to provide POP3 service to a customer.

Login information

Please provide a username and a password. This information can be used to log into the system.

**Username:** pop3_user

**Password:** ••••••

Figure 6 - Configure POP3 user connection

# Step 4 - Configure the SMTP service

Before connecting and configuring the Windows Mail with Ozeki SMS Gateway, you need to configure the SMTP service in the SMS Gateway by turning on the SMTP authentication. This SMTP service is created automatically when you created the SMTP user connection. To find that service, just select the Advanced menu from the toolbar, and as you can see it in Figure 7, and then, click on the Details button of the SMTP service.

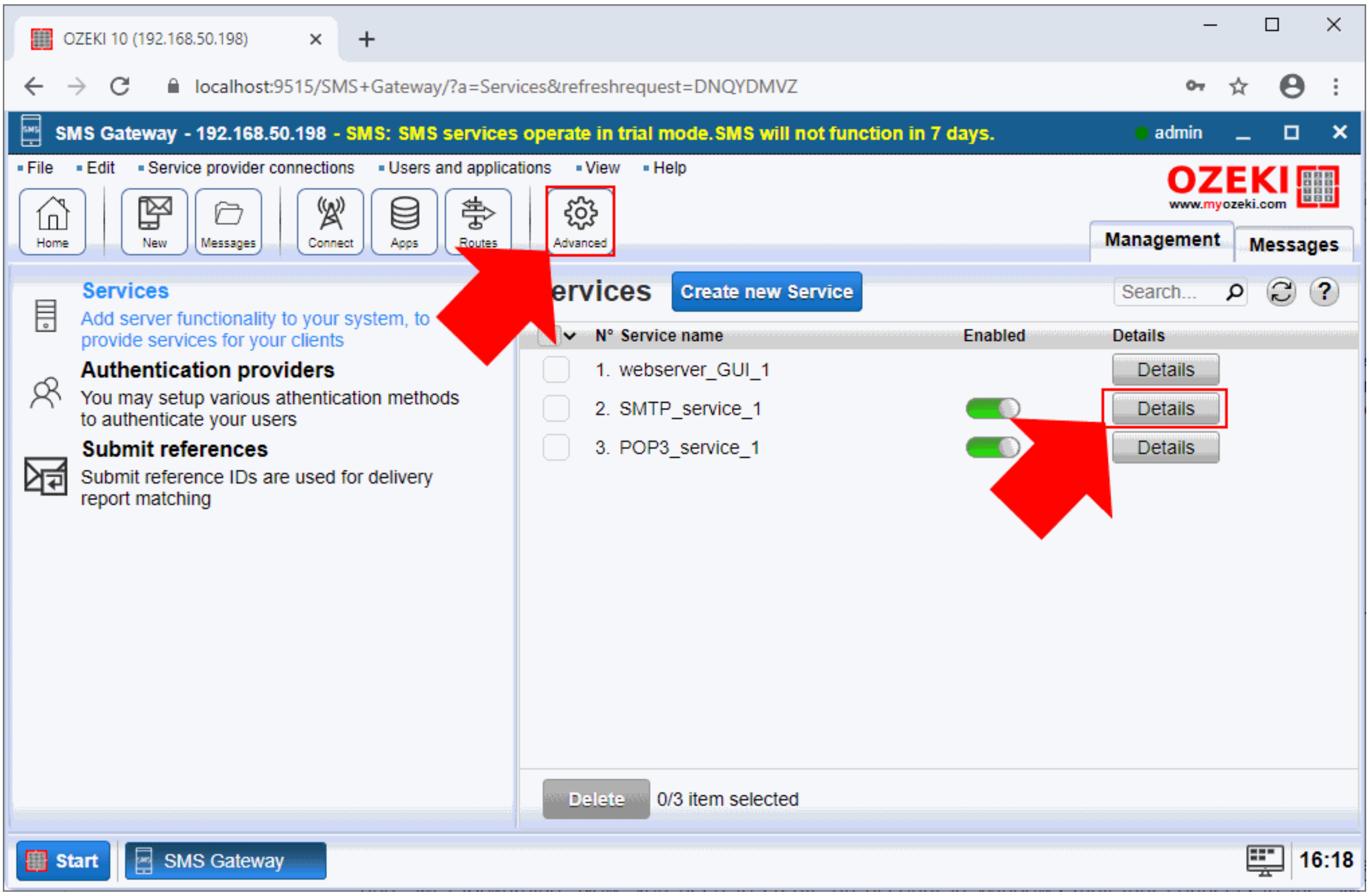
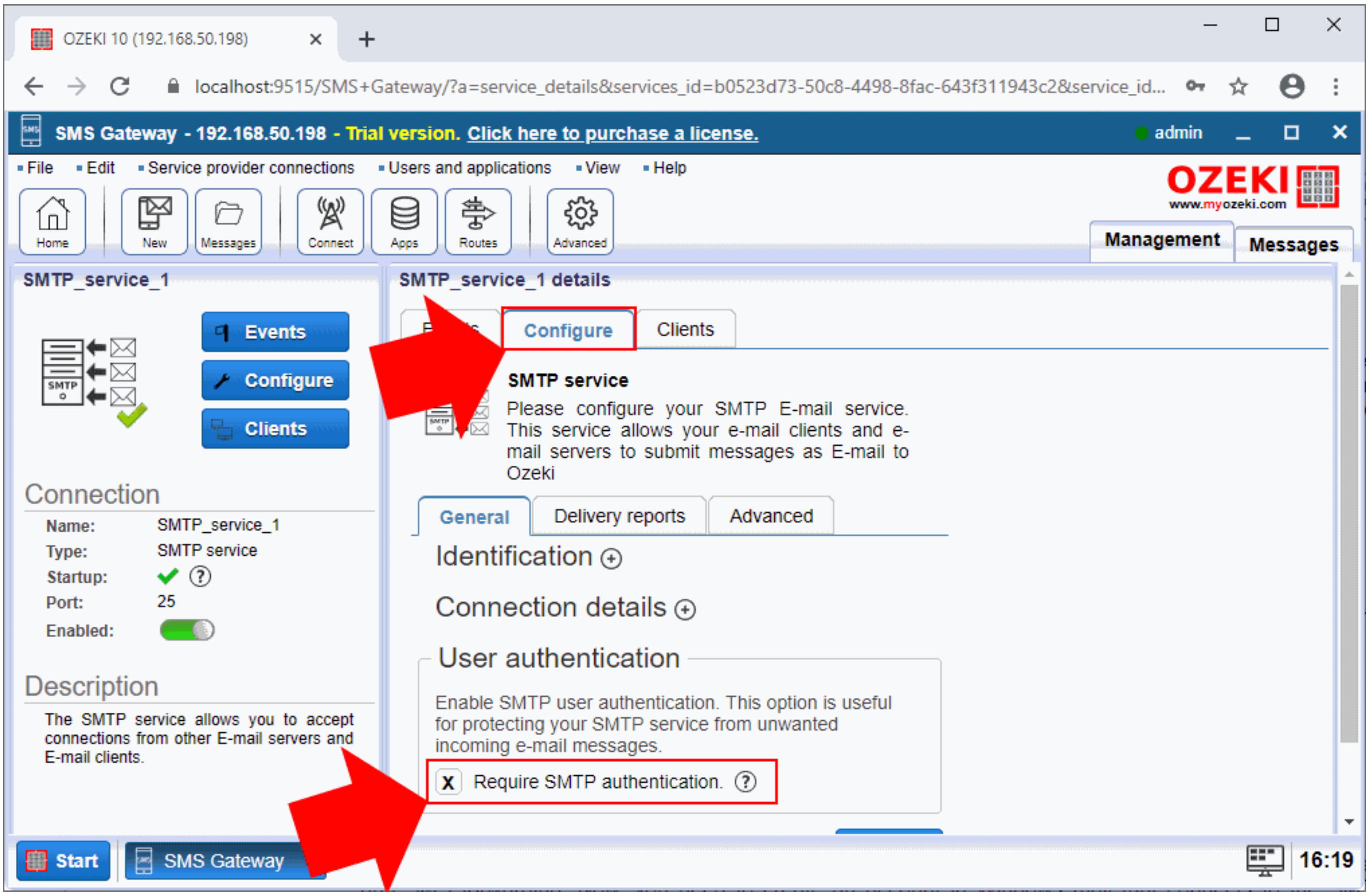

OZEKI 10 (192.168.50.198)
localhost:9515/SMS+Gateway/?a=Services&refreshrequest=DNQYDMVZ
SMS Gateway - 192.168.50.198 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Services
Add server functionality to your system, to provide services for your clients
Authentication providers
You may setup various athentication methods to authenticate your users
Submit references
Submit reference IDs are used for delivery report matching
Services
Create new Service
Search...
N° Service name
Enabled
Details
1. webserver_GUI_1
2. SMTP_service_1
3. POP3_service_1
Details
Delete
0/3 item selected
Start
SMS Gateway
16:18

Figure 7 - Select the SMTP service

In the Details menu of the SMTP service, you can perform some modifications on the service. To turn on the 'Require SMTP authentication' option, first, select the Configure tab as you can see it in Figure 8. Here, in the User authentication submenu, just check the 'Require SMTP authentication' option. If you finished, click on OK to save the modifications of the SMTP service.



SMTP service

Please configure your SMTP E-mail service. This service allows your e-mail clients and e-mail servers to submit messages as E-mail to Ozeki

Connection

Name: SMTP_service_1

Type: SMTP service

Startup: ✓

Port: 25

Enabled: ●

Description

The SMTP service allows you to accept connections from other E-mail servers and E-mail clients.

Identification ⊕

Connection details ⊕

User authentication

Enable SMTP user authentication. This option is useful for protecting your SMTP service from unwanted incoming e-mail messages.

☒ Require SMTP authentication. ?

Figure 8 - Turn on 'Require SMTP authentication'

## Step 5 - Create new account in Windows mail

At this point, you configured every connection and service in Ozeki SMS Gateway to be able to handle the e-mail and SMS forwarding. Now, you need to create an account in Windows mail that connects to the SMTP and POP3 servers integrated in SMS Gateway. First, open Windows mail on your computer. This application is installed on every Windows computer by default, and you can open it by clicking on its icon on the taskbar (Figure 9).

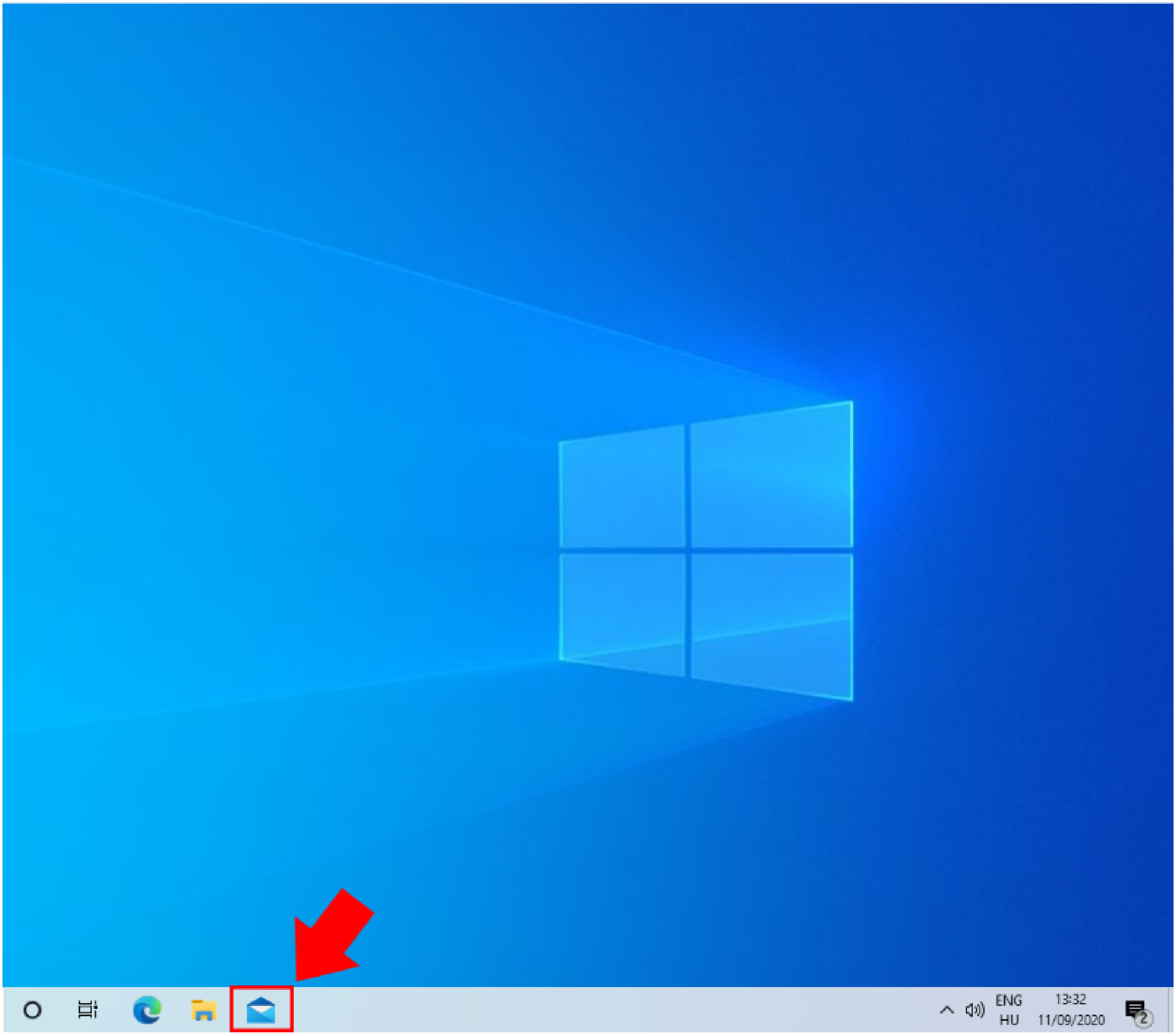


Figure 9 - Open Windows mail

After you opened the Windows mail application, you will be able to see the mian menu, where you can check the incoming and outgoing e-mail messages. To create a new account, first, click on the Settings icon in the bottom left corner as Figure 10 shows that, and on the right-hand side, select the 'Manage accounts' option. After opening this menu, you need to select the 'Add account' option to be able to create a new account.

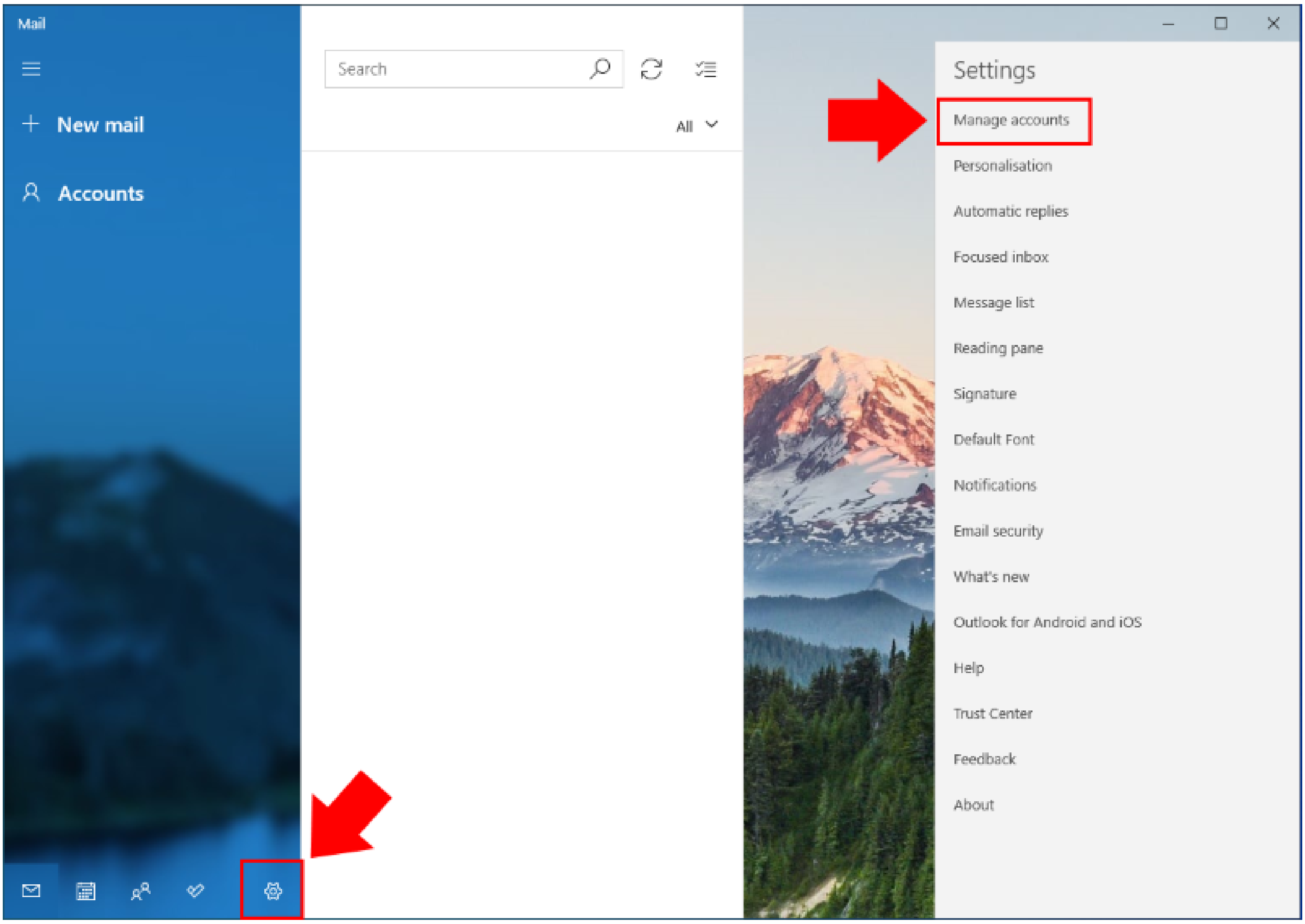
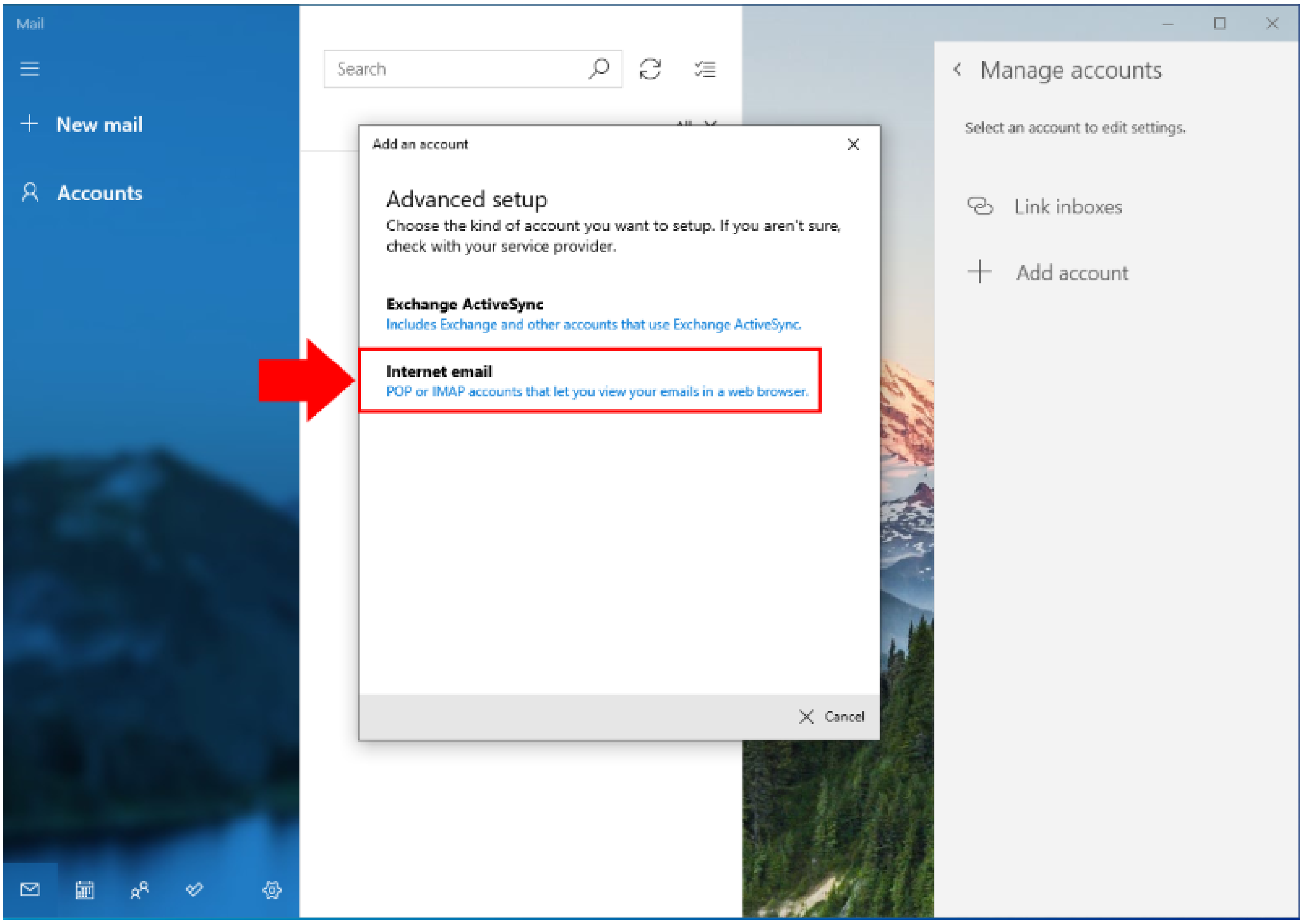


Figure 10 - Add a new account

## Step 6 - Select Internet e-mail account

In the list of accounts, you can create or connect many types of e-mail accounts to Windows mail. Now, you need to select the 'Advanced setup' option to be able to configure the SMTP and POP3 servers. As you can see it in Figure 11, you need to select the 'Internet email' option.



Advanced setup

Choose the kind of account you want to setup. If you aren't sure, check with your service provider.

**Exchange ActiveSync**
Includes Exchange and other accounts that use Exchange ActiveSync.

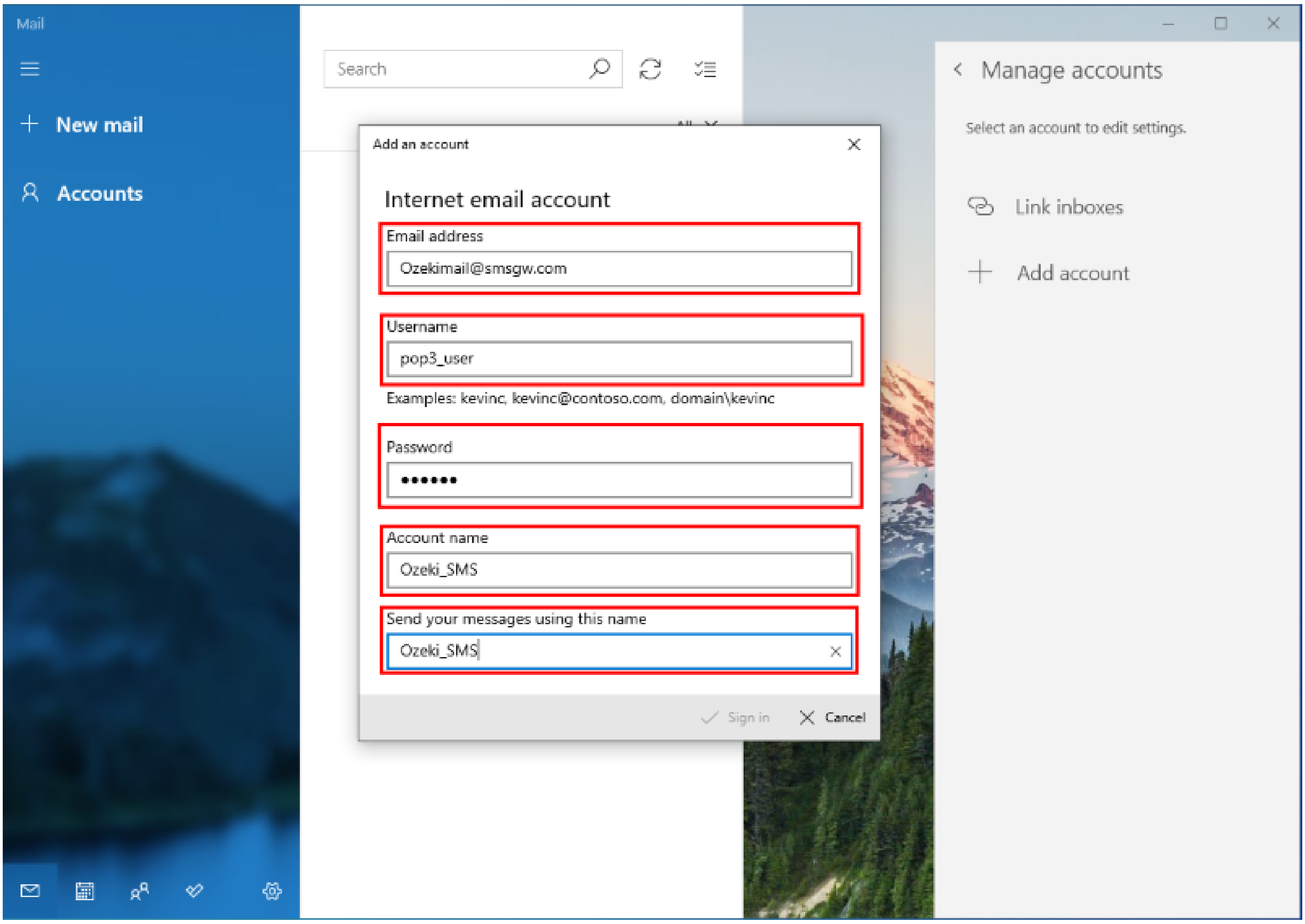
**Internet email**
POP or IMAP accounts that let you view your emails in a web browser.

Select an account to edit settings.

Figure 11 - Select Internet e-mail

## Step 7 - Configure Internet e-mail account

The next step is to specify the details of the account to be able to send and receive e-mails and SMS messages. First, type an e-mail address that you would like to use for this account. Next, in the Username and Password fields, type the username and password of your POP3 user connection that you created in SMS Gateway. Then, type an Account name, that Windws mail uses to identify the account and to the next field, type a name that you would like to use as a sender name when you send an e-mail.



Add an account

Internet email account

Email address: Ozekimail@smsgw.com

Username: pop3_user

Examples: kevinc, kevinc@contoso.com, domain\kevinc

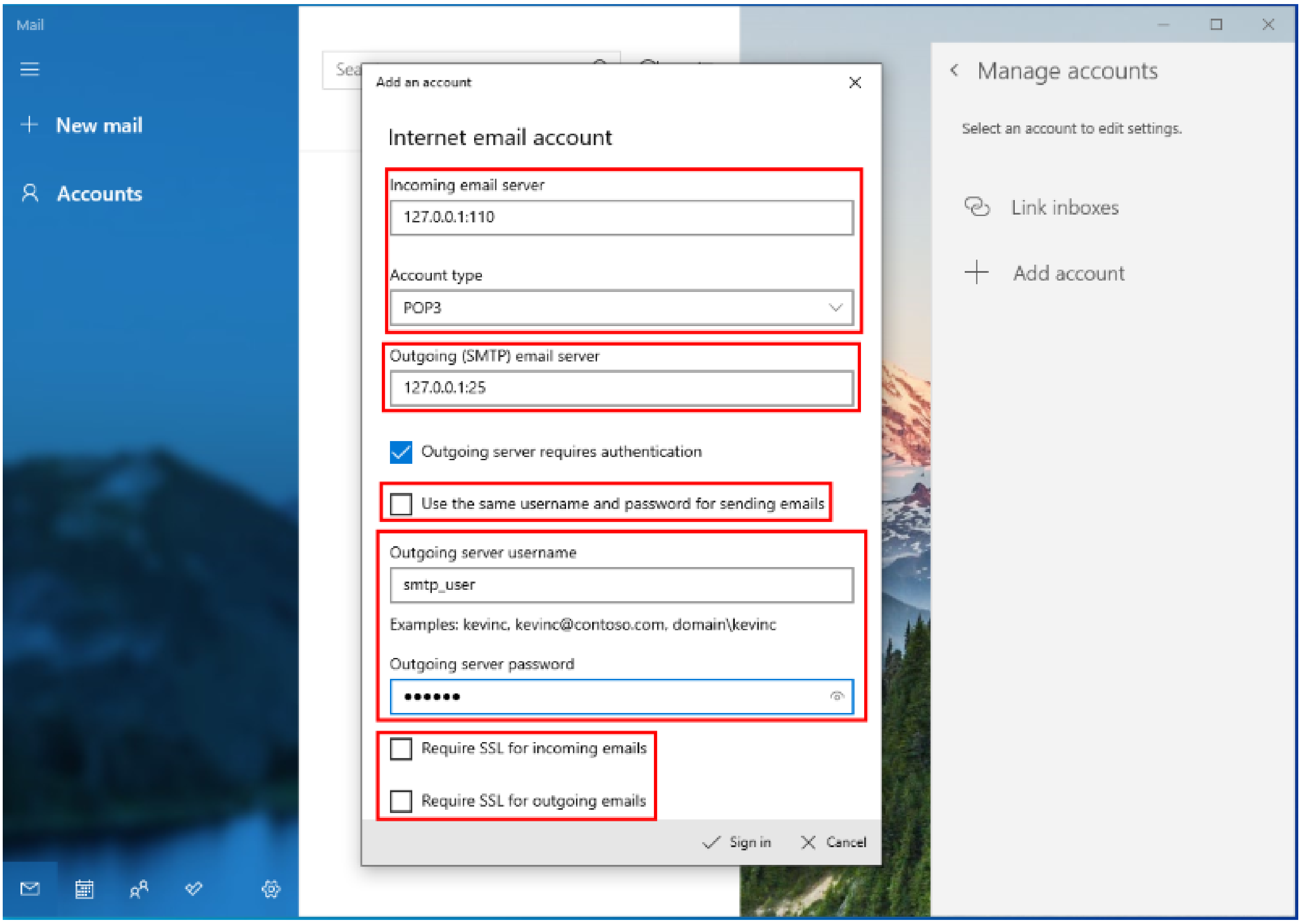
Password: ••••••

Account name: Ozeki_SMS

Send your messages using this name: Ozeki_SMS

Figure 11 - Specify the details of the new account

Next, you need to specify the details of the SMTP and POP3 servers. The incoming e-mail server will be a POP3 server. Here, you need to type 'localhost:110' to use the built-in POP3 server of SMS Gateway. Next, select the POP3 option as an 'Account type'. Then, in the field of 'Outgoing (SMTP) email server', type 'localhost:25' to connect to the SMTP server of the SMS Gateway (Figure 12). In the next step, disbale the 'Use the same name and password for sending email' and here, type the username and password of the SMTP user connection. Lastly, disable SSL for both incoming and outgoing messages, and click on 'Sign in'.



Add an account

Internet email account

Incoming email server

127.0.0.1:110

Account type

POP3

Outgoing (SMTP) email server

127.0.0.1:25

☑ Outgoing server requires authentication

□ Use the same username and password for sending emails

Outgoing server username

smtp_user

Examples: kevinc, kevinc@contoso.com, domain\kevinc

Outgoing server password

••••••

□ Require SSL for incoming emails

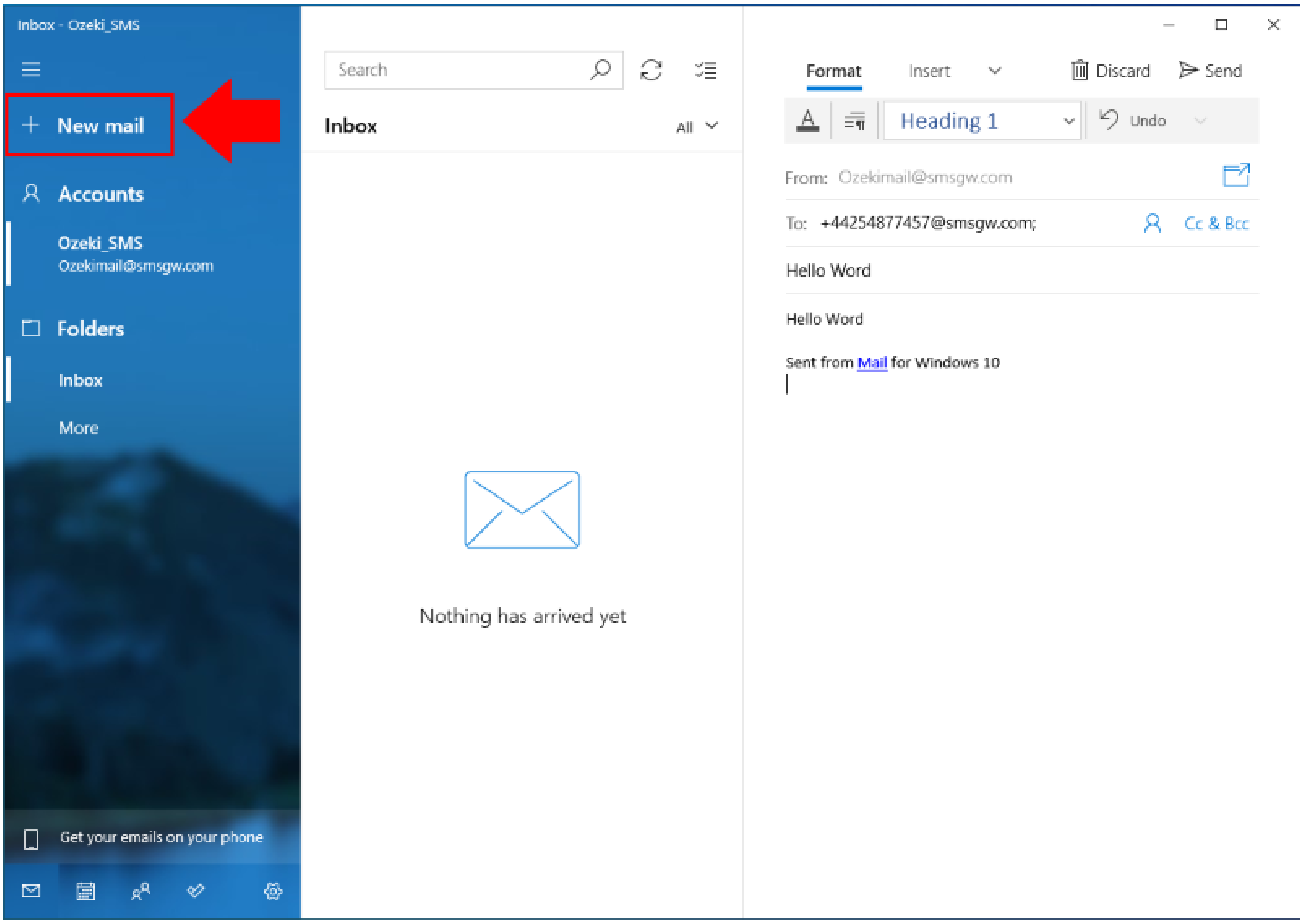
□ Require SSL for outgoing emails

✓ Sign in ✕ Cancel

Figure 12 - Specify the server details of the new account

# Step 8 - Send e-mail to test e-mail to SMS forwarding

In the previous steps, you configured both the SMS Gateway and Windows mail, so now, it is ready to forward the e-mail and SMS messages. First, let's try the e-mail to SMS forwarding. For that, open Windows mail, and click on the 'Plus' button to write a new e-mail. Here, you need to enter the recipient in phone number@domain form, where the phone number is the recipient's phone number, and the domain is the SMS Gateway domain name. Then, just type the subject and the body od the message. Lastly, click on Send to send the e-mail.



From: Ozekimail@smsgw.com

To: +44254877457@smsgw.com;

Cc & Bcc

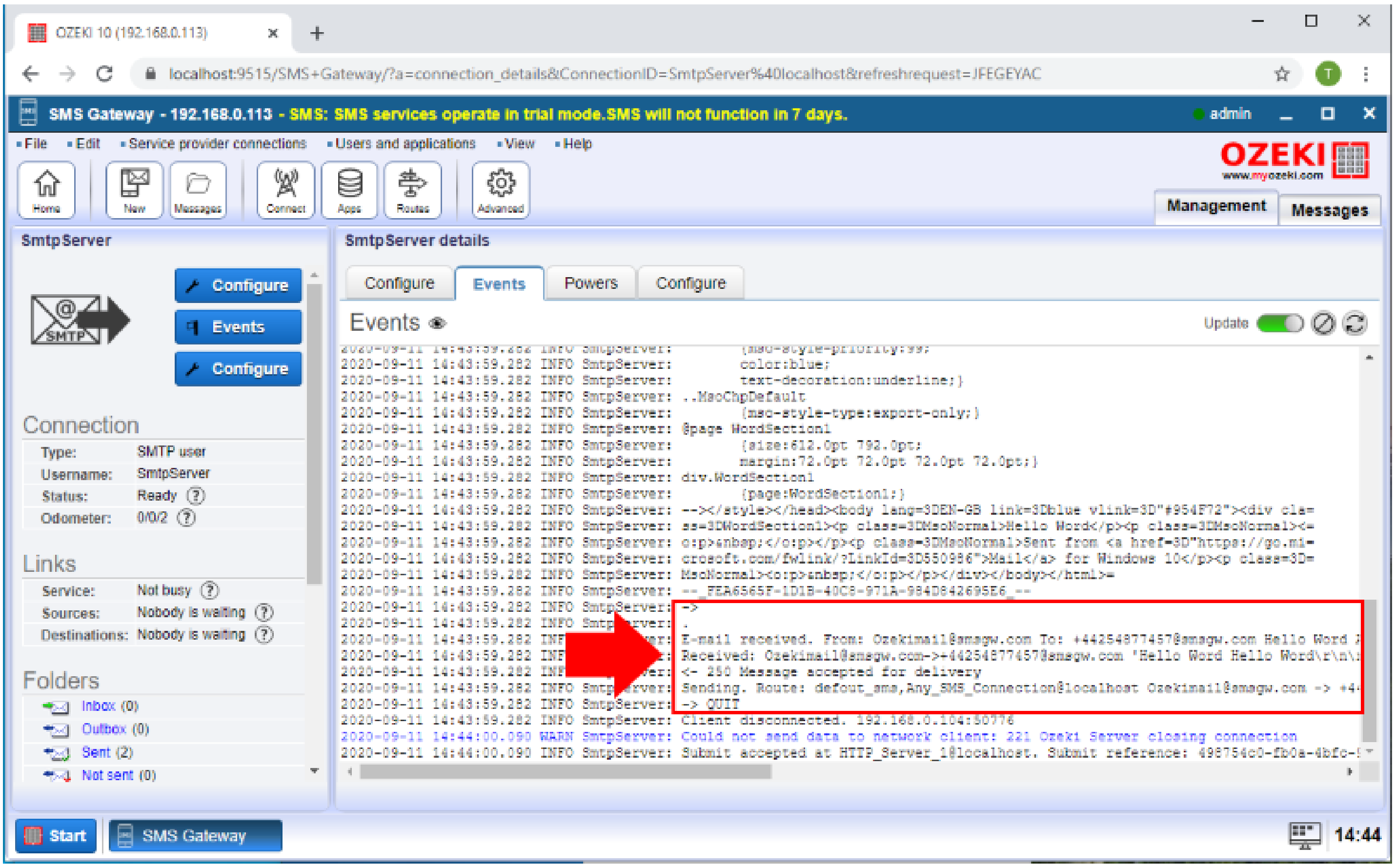
Hello Word

Hello Word

Sent from Mail for Windows 10

Figure 13 - Send test e-mail

After you sent the e-mail message, you can check if the solution works by seeing the logs of the SMTP User connection. You can open it, by opening the menu of the connection in SMS Gateway, and here, select the Events option as Figure 14 shows that. Here, you can see that the e-mail was received by the connection and it forwarded to the HTTP Server connection which can send the SMS message.



Connection

Type: SMTP user
Username: SmtpServer
Status: Ready
Odometer: 0/0/2

Links

Service: Not busy
Sources: Nobody is waiting
Destinations: Nobody is waiting

Folders

Inbox (0)
Outbox (0)
Sent (2)
Not sent (0)

```
2020-09-11 14:43:59.282 INFO SmtpServer:        {mso-style-priority:99;
2020-09-11 14:43:59.282 INFO SmtpServer:        color:blue;
2020-09-11 14:43:59.282 INFO SmtpServer:        text-decoration:underline;}
2020-09-11 14:43:59.282 INFO SmtpServer: ..MsoChpDefault
2020-09-11 14:43:59.282 INFO SmtpServer:        {mso-style-type:export-only;}
2020-09-11 14:43:59.282 INFO SmtpServer: @page WordSection1
2020-09-11 14:43:59.282 INFO SmtpServer:        {size:612.0pt 792.0pt;
2020-09-11 14:43:59.282 INFO SmtpServer:        margin:72.0pt 72.0pt 72.0pt 72.0pt;}
2020-09-11 14:43:59.282 INFO SmtpServer: div.WordSection1
2020-09-11 14:43:59.282 INFO SmtpServer:        {page:WordSection1;}
2020-09-11 14:43:59.282 INFO SmtpServer: --></style></head><body lang=3DEN-GB link=3Dblue vlink=3D"#954F72"><div cla=
2020-09-11 14:43:59.282 INFO SmtpServer: ss=3DWordSection1><p class=3DMsoNormal>Hello Word</p><p class=3DMsoNormal><=
2020-09-11 14:43:59.282 INFO SmtpServer: o:p>&nbsp;</o:p></p><p class=3DMsoNormal>Sent from <a href=3D"https://go.mi=
2020-09-11 14:43:59.282 INFO SmtpServer: crosoft.com/fwlink/?LinkId=3D550986">Mail</a> for Windows 10</p><p class=3D=
2020-09-11 14:43:59.282 INFO SmtpServer: MsoNormal><o:p>&nbsp;</o:p></p></div></body></html>=
2020-09-11 14:43:59.282 INFO SmtpServer: --_FEA6565F-1D1B-40C8-971A-984D842695E6_--
2020-09-11 14:43:59.282 INFO SmtpServer: ->
2020-09-11 14:43:59.282 INFO SmtpServer: .
2020-09-11 14:43:59.282 INFO SmtpServer: E-mail received. From: Ozekimail@smsgw.com To: +44254877457@smsgw.com Hello Word 
2020-09-11 14:43:59.282 INFO SmtpServer: Received: Ozekimail@smsgw.com->+44254877457@smsgw.com 'Hello Word Hello Word\r\n\
2020-09-11 14:43:59.282 INFO SmtpServer: <- 250 Message accepted for delivery
2020-09-11 14:43:59.282 INFO SmtpServer: Sending. Route: defout_sms,Any_SMS_Connection@localhost Ozekimail@smsgw.com -> +4
2020-09-11 14:43:59.282 INFO SmtpServer: -> QUIT
2020-09-11 14:43:59.282 INFO SmtpServer: Client disconnected. 192.168.0.104:50776
2020-09-11 14:44:00.090 WARN SmtpServer: Could not send data to network client: 221 Ozeki Server closing connection
2020-09-11 14:44:00.090 INFO SmtpServer: Submit accepted at HTTP_Server_1@localhost. Submit reference: 498754c0-fb0a-4bfc-
```

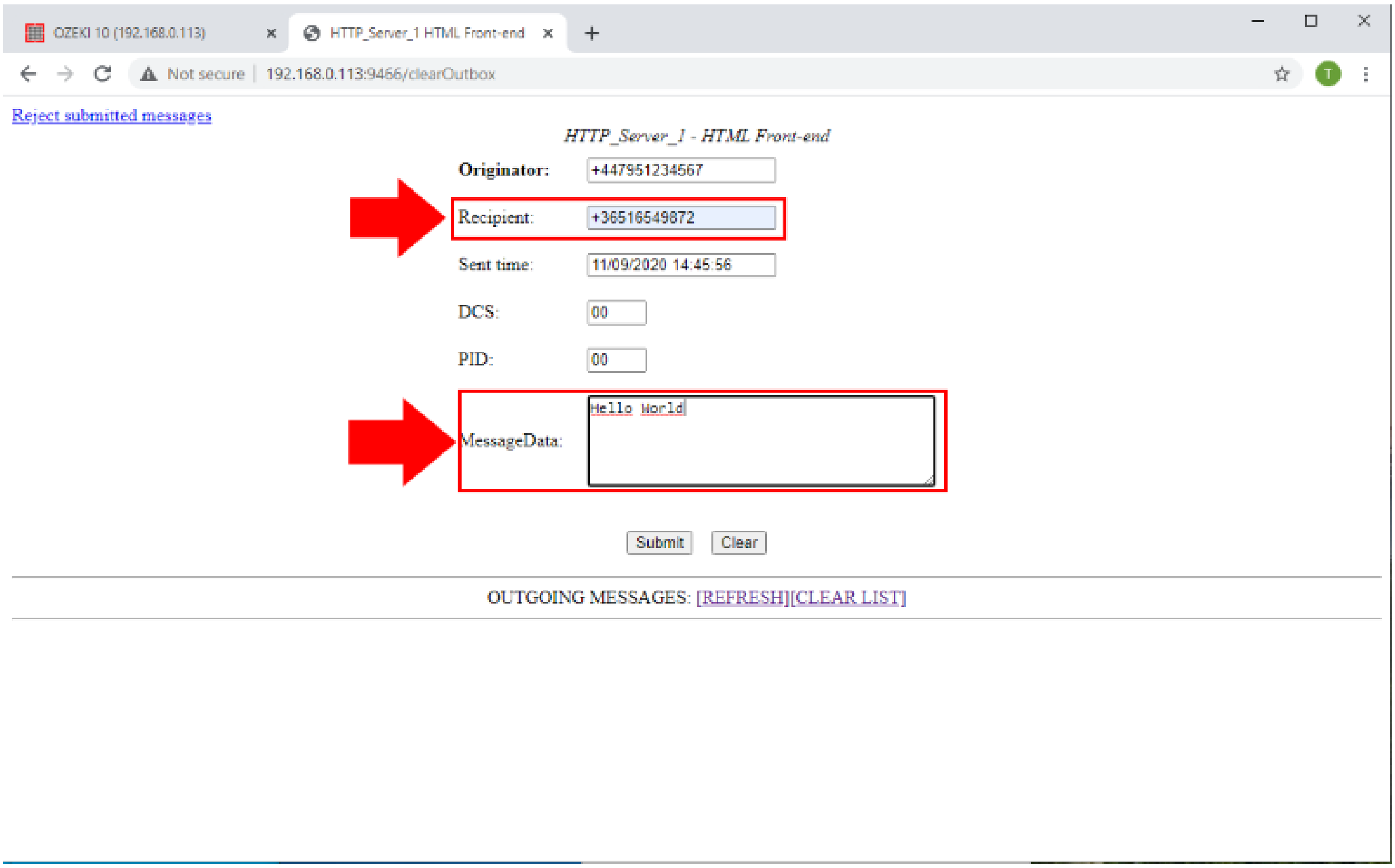
Figure 13 - Events menu of the SMTP User connection

# Step 9 - Set up route to POP3 user connection

Before testing the SMS to e-mail forwarding, you need to do one more configuration to make sure, the solution works perfectly. At this point, you need to create a routing rule from any SMS connection to the POP3 user connection. This allows the POP3 user connection to collect all incoming SMS messages and send them to the connected Windows mail account. So, open the Routes menu from the toolbar and select 'Create new Route'. Here, as Figure 14 shows that, select 'Any SMS connection' for the 'From' field, and the POP3 user connection for the 'To' field. To create the route, just click OK.

# Step 10 - Send SMS to test SMS to e-mail forwarding

You can test your solution in the other way as well, and see if it can forward SMS messages as e-mails. For that, open the HTML form of the HTTP Server connection. Here, you can send an SMS message (Figure 15). Just type the phone number of the recipient, and the message itself. If you are ready, just click on 'Submit' to send the SMS message.



OZEKI 10 (192.168.0.113)
HTTP_Server_1 HTML Front-end
Not secure
192.168.0.113:9466/clearOutbox
Reject submitted messages

*HTTP_Server_1 - HTML Front-end*

**Originator:** +447951234567

Recipient: +36516549872

Sent time: 11/09/2020 14:45:56

DCS: 00

PID: 00

MessageData: Hello World

OUTGOING MESSAGES: [REFRESH][CLEAR LIST]

Figure 15 - HTML form of th HTTP Server connection

As soon as you sent the SMS message, the SMS Gateway uses the created routing rule to help the POP3 user connection to receive the SMS message. After the POP3 user connection received the SMS message, it forwards the message to the connected Windows mail account as a new e-mail. As you can see it in Figure 16, the message arrives as a new e-mail in the Inbox folder.

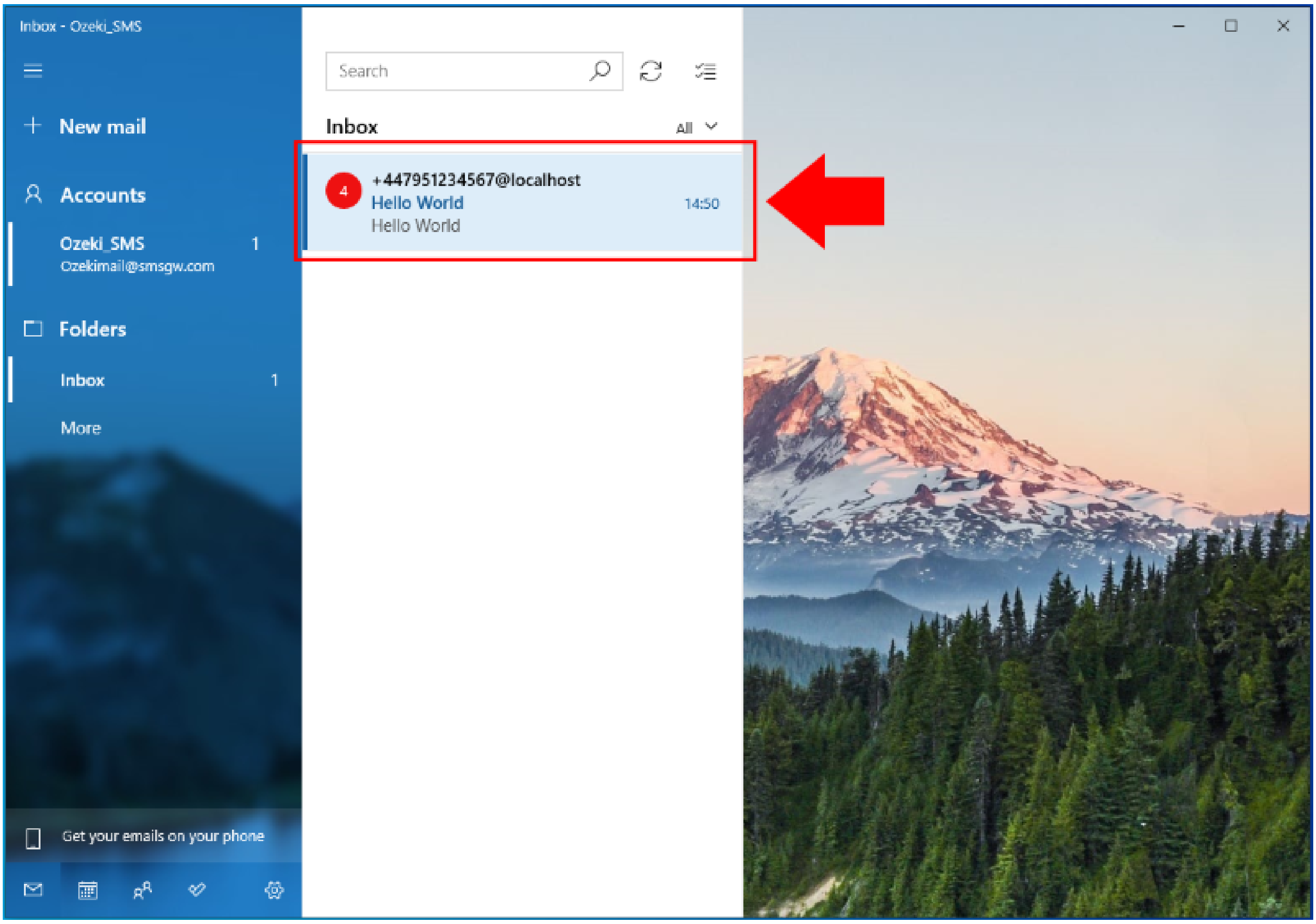


Figure 16 - Inbox folder in Windows mail

# Change the message

***This guide is going to demonstrate how you can manipulate messages using the routing rules of the SMS Gateway. The routing rules are used in the system to direct the messages between connections. You can setup rules to take a message form one connection and to pass it to another connection. For example you can forward in incoming e-mail messages from an e-mail connection to an SMS connection and send it to a mobile phone as SMS. A routing rule is also great tool to manipulate a message. This guide shows how to modify the sender or recipient phone number, or how you can modify the message text. So, let's start right now!***

### Change SMS recipient

SMS Gateway routing rules allow you to do some modification in the messages. One of them is that you can change the recipient phone number of the message. It can be useful when you have got a new phone number and you would like to redirect the messages that still to be delivered to your old phone number to that new phone number.

Learn more about how you can change the recipient number of the SMS messages

### Change SMS sender

The following guide is going to show you how you can change and manipulate the sender phone number of an outgoing message in the SMS Gateway. By changing the sender phone number and define a new phone number, you can ensure that every outgoing message is going to have the same predefined phone number as a sender address.

See how to change the sender of the outgoing messages

### Regexp manipulation

The regular expressions can be great if you would like to modify some details of the sender or recipient phone number but still keep the original phone number as well. This guide demonstrates how you can add prefix, remove or replace some numbers of the sender or recipient phone number.

Check how you can manipulate the messages using regular expressions

### Change SMS text

By following the link to this guide, you will be able to learn about how you can change the text of an outgoing message. By performing this action, all of your outgoing message going to have got the same text since the routing rule replace the original text with the predefined text message.

Learn more about how to change the text of the outgoing messages

### Append SMS message

You can learn about how you can add text to your outgoing message but keeping the original text too at the same time. Appending the message is quite useful when you are using a template for your messages and you wouldn't like to type it in every message. By applying this setting, the routing rule appends the message by the specified text.

See how to add text to your outgoing messages

### Swap sender and recipient

This solution allows you to create an autoresponding service using your routing rules. By activating this modification, you can create a two way communication by both the sender and the recipient receives a message. You can replace the original message as well with a response message.

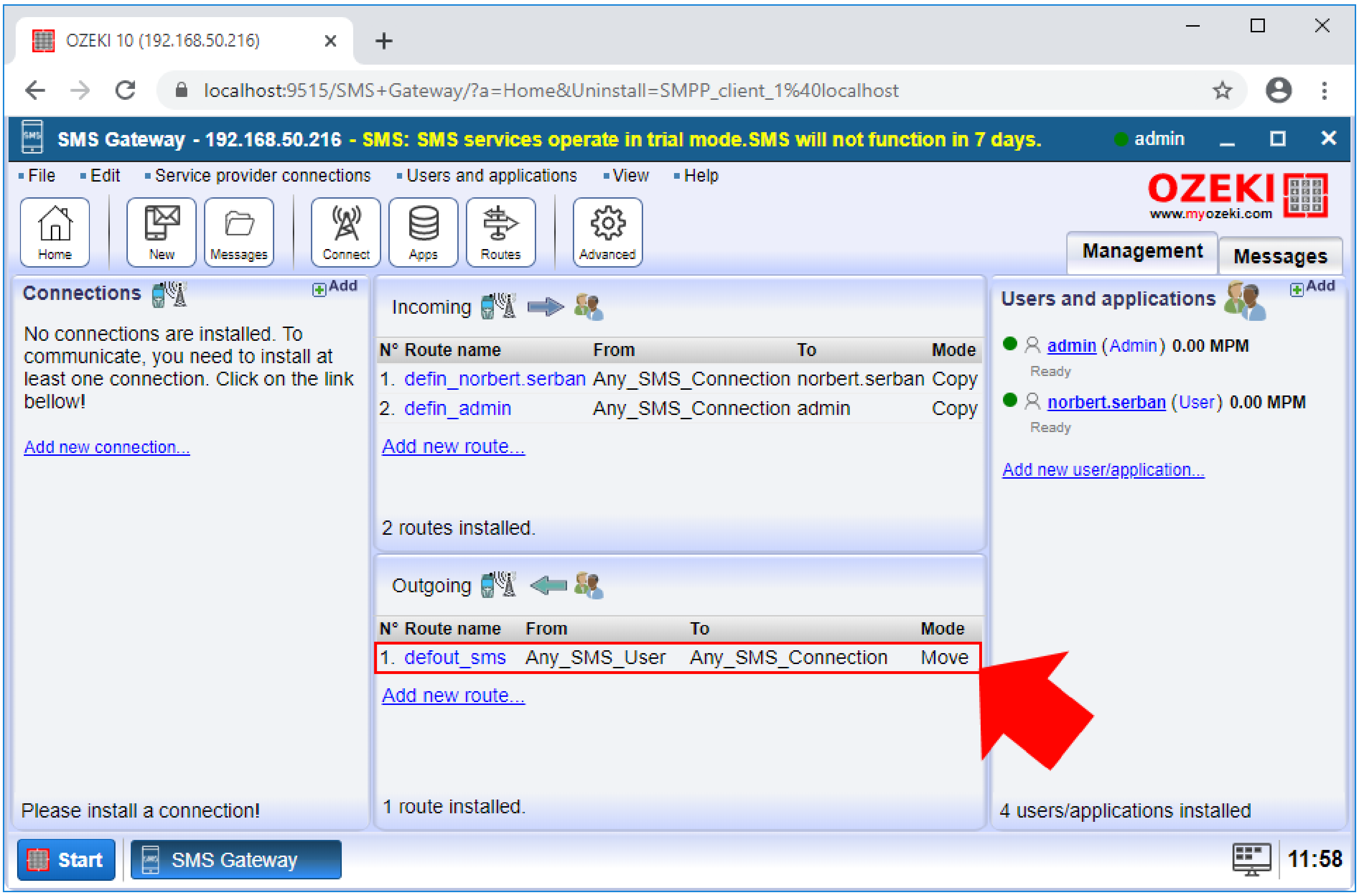
Learn more about how to swap sender and recipient of your messages

# How to change the recipient phone number

***This document is going to show how you can change the recipient phone number using the routing table. With this operation, you can ensure for example that any outgoing message from your SMS Gateway system will be received by a specified phone number. This feature is also useful when you changed you phone number and you would like to redirect the messages sent to you old phone number to the new phone number. The guide demonstrates how you nee to configure the routing rule to change the recipient number and how to test the solution. It does not take more than five minutes to complete this guide, so let's start right now!***

## Step 1 - Open a routing rule

The first step is to select a route that you want to modify. If you are not familiar with routing rules, here you can learn about SMS routing in SMS Gateway. SMS Gateway creates default routes, so you can simply select the default outgoing route as you can see it in Figure 1.



OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=Home&Uninstall=SMPP_client_1%40localhost
SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
OZEKI
www.myozeki.com
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Connections
Add
Incoming
Outgoing
Users and applications
Add
Start
SMS Gateway
11:58

No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

Please install a connection!

Incoming

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defin_norbert.serban | Any_SMS_Connection | norbert.serban | Copy |
| 2. | defin_admin | Any_SMS_Connection | admin | Copy |

Add new route...

2 routes installed.

Outgoing

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move |

Add new route...

1 route installed.

admin (Admin) 0.00 MPM
Ready

norbert.serban (User) 0.00 MPM
Ready

Add new user/application...

4 users/applications installed

Figure 1 - Select a routing rule in SMS Gateway

Here, you can see the setting menu of the routing rule. To be able to modify the messages on this route, you just need to select the Modify tab as you can see it in Figure 2.

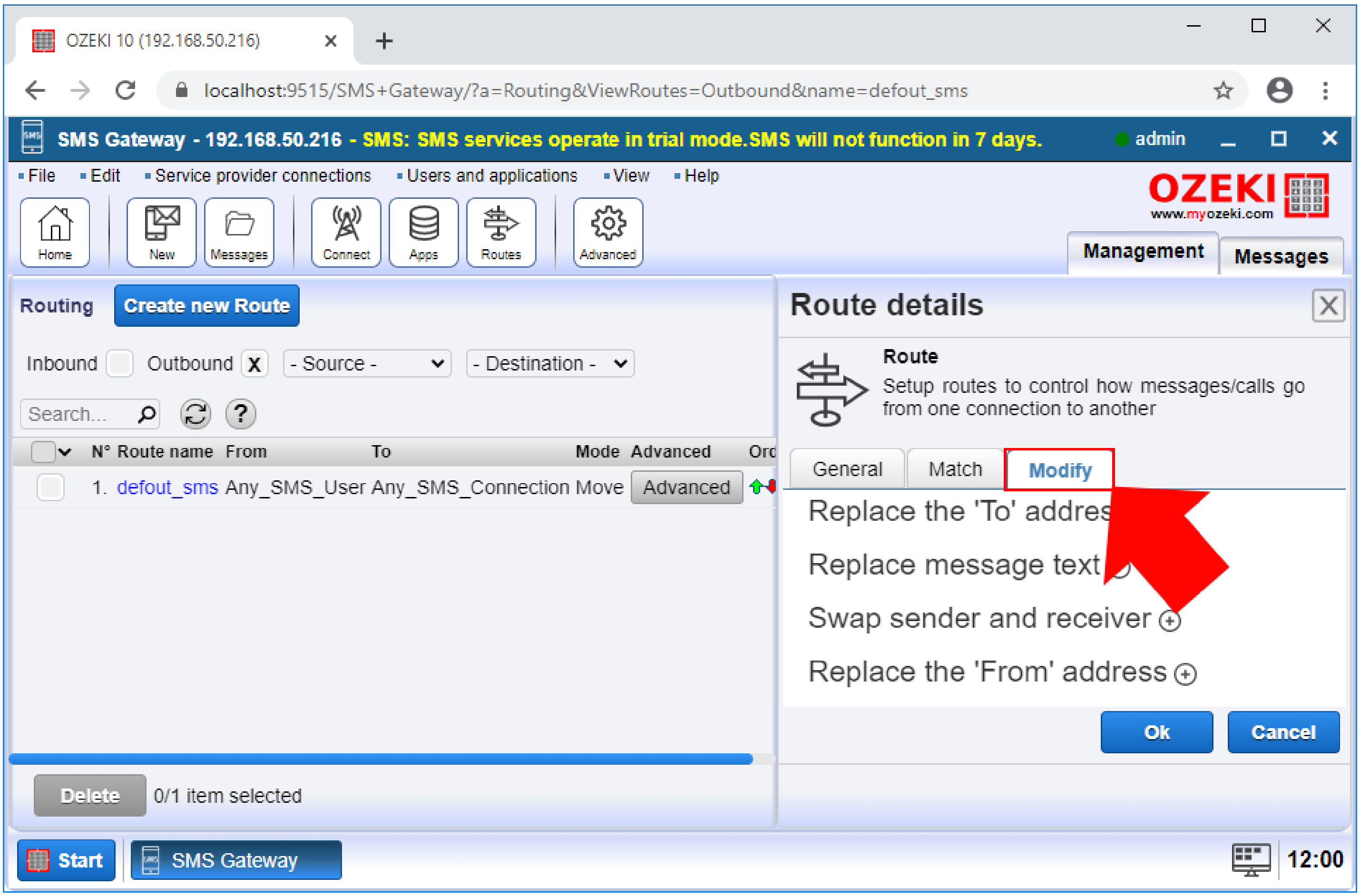
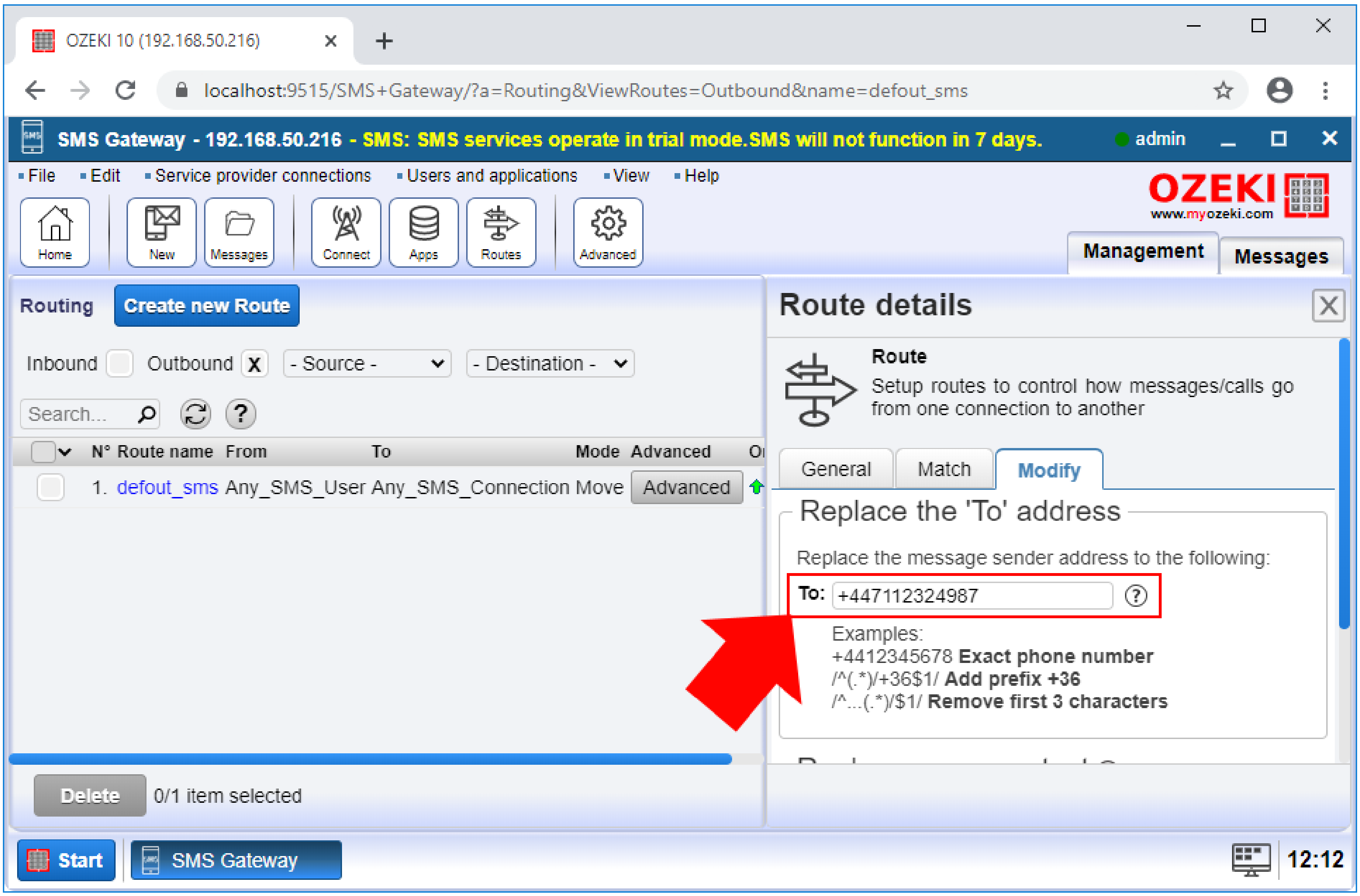

OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=Routing&ViewRoutes=Outbound&name=defout_sms
SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
OZEKI
www.myozeki.com
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Routing
Create new Route
Inbound
Outbound X
- Source -
- Destination -
Search...
N° Route name From To Mode Advanced Ord
1. defout_sms Any_SMS_User Any_SMS_Connection Move Advanced
Delete
0/1 item selected
Route details
Route
Setup routes to control how messages/calls go from one connection to another
General
Match
Modify
Replace the 'To' address
Replace message text
Swap sender and receiver
Replace the 'From' address
Ok
Cancel
Start
SMS Gateway
12:00

Figure 2 - Select the Modify menu

## Step 2 - Configure the recipient phone number

In the Modify menu, you can perform all modifying operation for the outgoing messages. At this point, you need to select the 'Replace the 'To' address' submenu like in Figure 3, and here, you can just type the phone number that you want to receive every outgoing message. Finally, just click on OK to save the modifications.


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=Routing&ViewRoutes=Outbound&name=defout_sms
SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Routing
Create new Route
Inbound
Outbound
- Source -
- Destination -
Search...
N° Route name From To Mode Advanced
Delete
Start
SMS Gateway
12:12

1. defout_sms Any_SMS_User Any_SMS_Connection Move Advanced

0/1 item selected

Route details

Route
Setup routes to control how messages/calls go from one connection to another

General | Match | Modify

Replace the 'To' address

Replace the message sender address to the following:

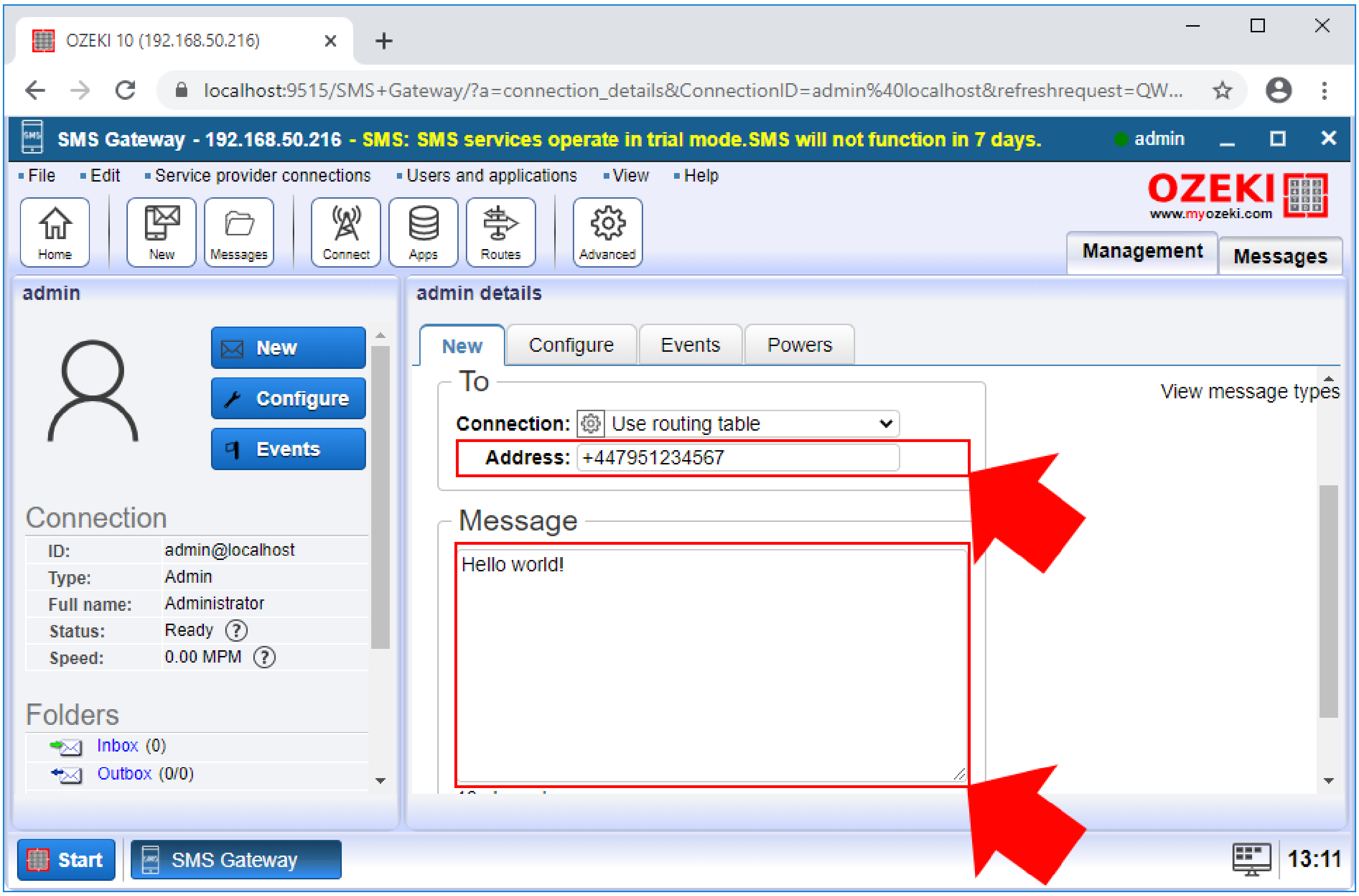
To: +447112324987

Examples:
+4412345678 **Exact phone number**
/^(.*)/+36$1/ **Add prefix +36**
/^...(.*)/$1/ **Remove first 3 characters**

Figure 3 - Configure the recipient phone number

# Step 3 - Send a test message

After you configured the outgoing routing rule, you just need to write a test message. For that purpose, you can set up a SMPP client, which will forward the message to the recipient. Here, you can learn how to set up a SMPP client connection. To write the message, just open the admin account, and like in Figure 4, just write a simple message to test the solution.


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=connection_details&ConnectionID=admin%40localhost&refreshrequest=QW...
SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
admin
New
Configure
Events
Connection
Folders
Inbox (0)
Outbox (0/0)
admin details
New
Configure
Events
Powers
View message types
Start
SMS Gateway
13:11

To

Connection: Use routing table

Address: +447951234567

Message

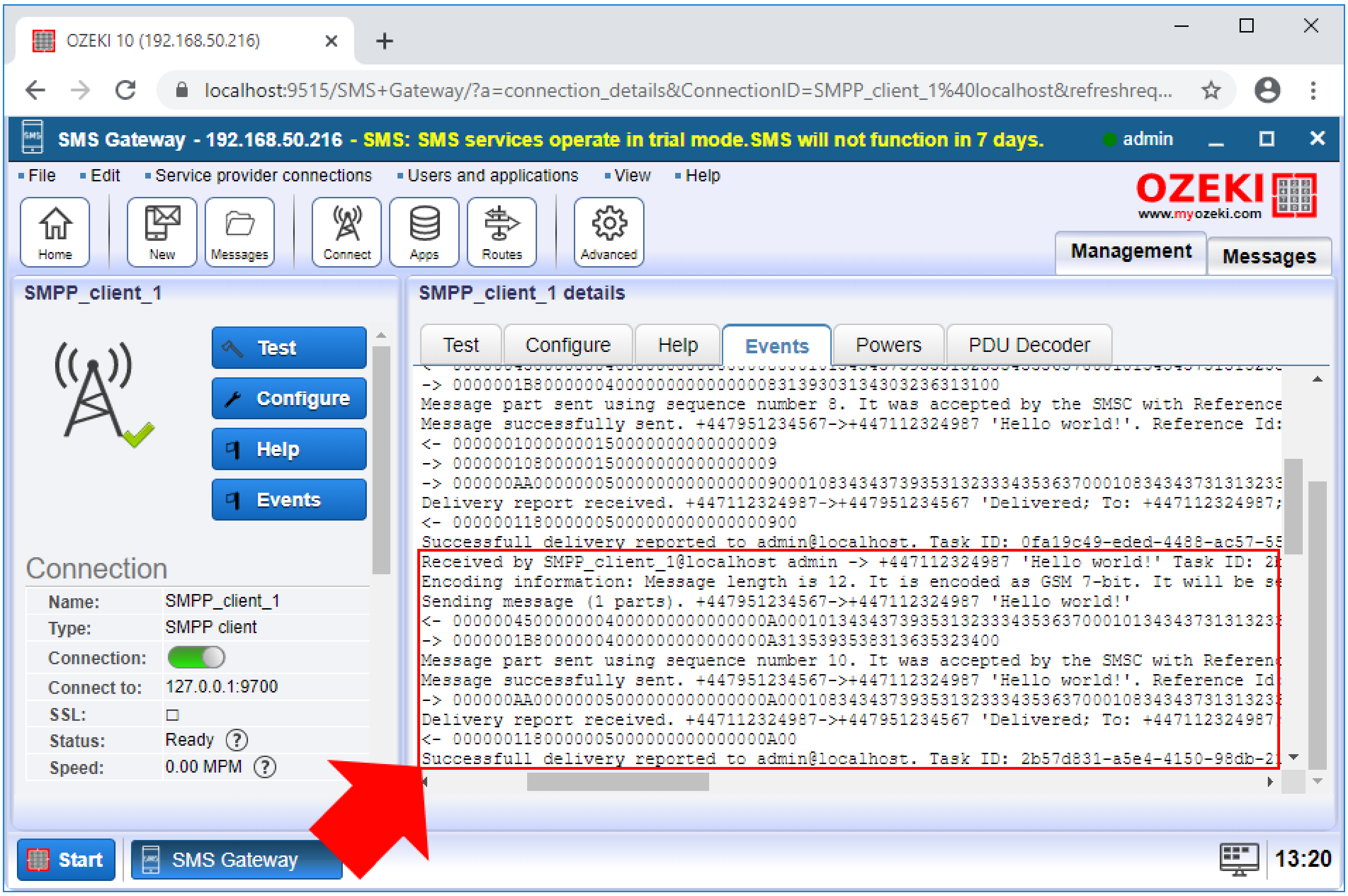
Hello world!

| | |
|---|---|
| ID: | admin@localhost |
| Type: | Admin |
| Full name: | Administrator |
| Status: | Ready |
| Speed: | 0.00 MPM |

Figure 4 - Write a simple test message

## Step 4 - View the results in the event log

To check if the modifications worked, you need to open the SMPP client and select the Events tab. Here, you can see every event of the SMPP client connection. As Figure 5 shows that, the routing rule changed the recipient phone number, so the SMPP client forwarded the message to the modified phone number.


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=connection_details&ConnectionID=SMPP_client_1%40localhost&refreshreq...
SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
OZEKI
www.myozeki.com
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
SMPP_client_1
Test
Configure
Help
Events
SMPP_client_1 details
Test
Configure
Help
Events
Powers
PDU Decoder
Start
SMS Gateway
13:20

Connection

Name: SMPP_client_1
Type: SMPP client
Connection:
Connect to: 127.0.0.1:9700
SSL: □
Status: Ready
Speed: 0.00 MPM

```
-> 0000001B80000004000000000000000831393031343032363131 00
Message part sent using sequence number 8. It was accepted by the SMSC with Reference
Message successfully sent. +447951234567->+447112324987 'Hello world!'. Reference Id:
<- 00000010000000150000000000000009
-> 00000010800000150000000000000009
-> 000000AA00000005000000000000000090001083434373935313233343536370001083434373131323З
Delivery report received. +447112324987->+447951234567 'Delivered; To: +447112324987;
<- 000000118000000500000000000000000900
Successfull delivery reported to admin@localhost. Task ID: 0fa19c49-eded-4488-ac57-55
Received by SMPP_client_1@localhost admin -> +447112324987 'Hello world!' Task ID: 2
Encoding information: Message length is 12. It is encoded as GSM 7-bit. It will be s
Sending message (1 parts). +447951234567->+447112324987 'Hello world!'
<- 000000450000000400000000000000000A0001013434373935313233343536370001013434373131323
-> 0000001B800000040000000000000000A313539353831363532 3400
Message part sent using sequence number 10. It was accepted by the SMSC with Referen
Message successfully sent. +447951234567->+447112324987 'Hello world!'. Reference Id
-> 000000AA0000000500000000000000000A0001083434373935313233343536370001083434373131323
Delivery report received. +447112324987->+447951234567 'Delivered; To: +447112324987
<- 0000001180000005000000000000000000A00
Successfull delivery reported to admin@localhost. Task ID: 2b57d831-a5e4-4150-98db-2
```

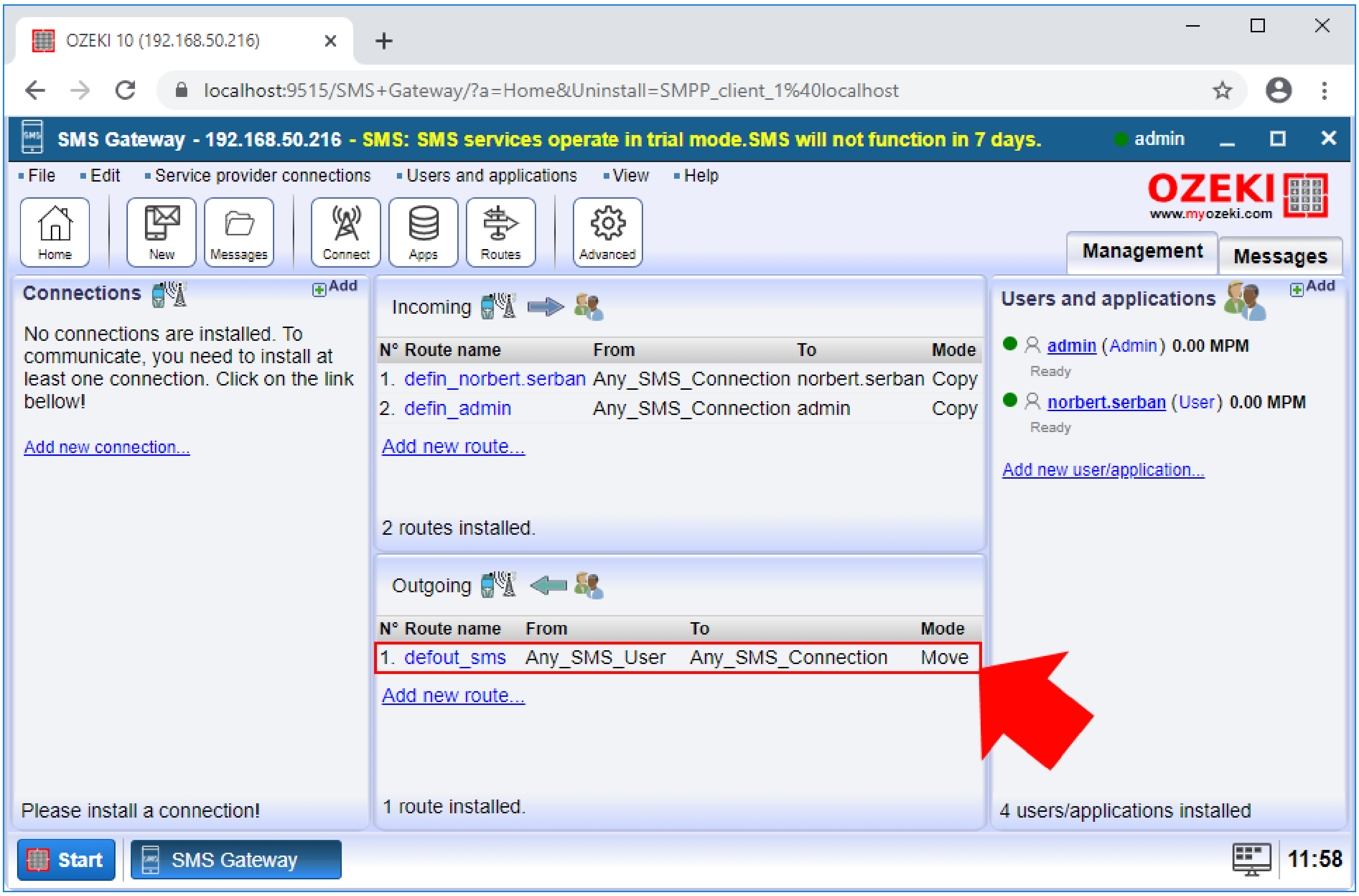
Figure 5 - See that the routing rule changed the recipient phone number

# How to change the sender phone number

***This section is about to show you how easy you can change the sender phone number in a routing rule. This modification process makes sure that every message sent from your SMS Gateway system is going to have the same predefined sender phone number. The document shows you all the configuration work that you have to perform in the routing rule the set a new sender phone number. It also demonstrates how you can test your system by sending a test message and see if the routing rule works fine.***

## Step 1 - Open a routing rule

The first step that you need to perform is to select a route that you want to modify. If you are not familiar with routing rules, here you can learn about SMS routing in SMS Gateway. SMS Gateway creates default routes, so you can simply select the default outgoing route as Figure 1 demonstrates it for you.



SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.

Connections

No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

Please install a connection!

Incoming

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defin_norbert.serban | Any_SMS_Connection | norbert.serban | Copy |
| 2. | defin_admin | Any_SMS_Connection | admin | Copy |

Add new route...

2 routes installed.

Outgoing

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move |

Add new route...

1 route installed.

Users and applications

- admin (Admin) 0.00 MPM Ready
- norbert.serban (User) 0.00 MPM Ready

Add new user/application...

4 users/applications installed

Figure 1 - Select a routing rule in SMS Gateway

After you selected the routing rule from the routing table, you can see the settings menu of the selected routing rule. To be able to modify the details of the messages going through this route, you just need to select the Modify tab as you can see it in Figure 2.

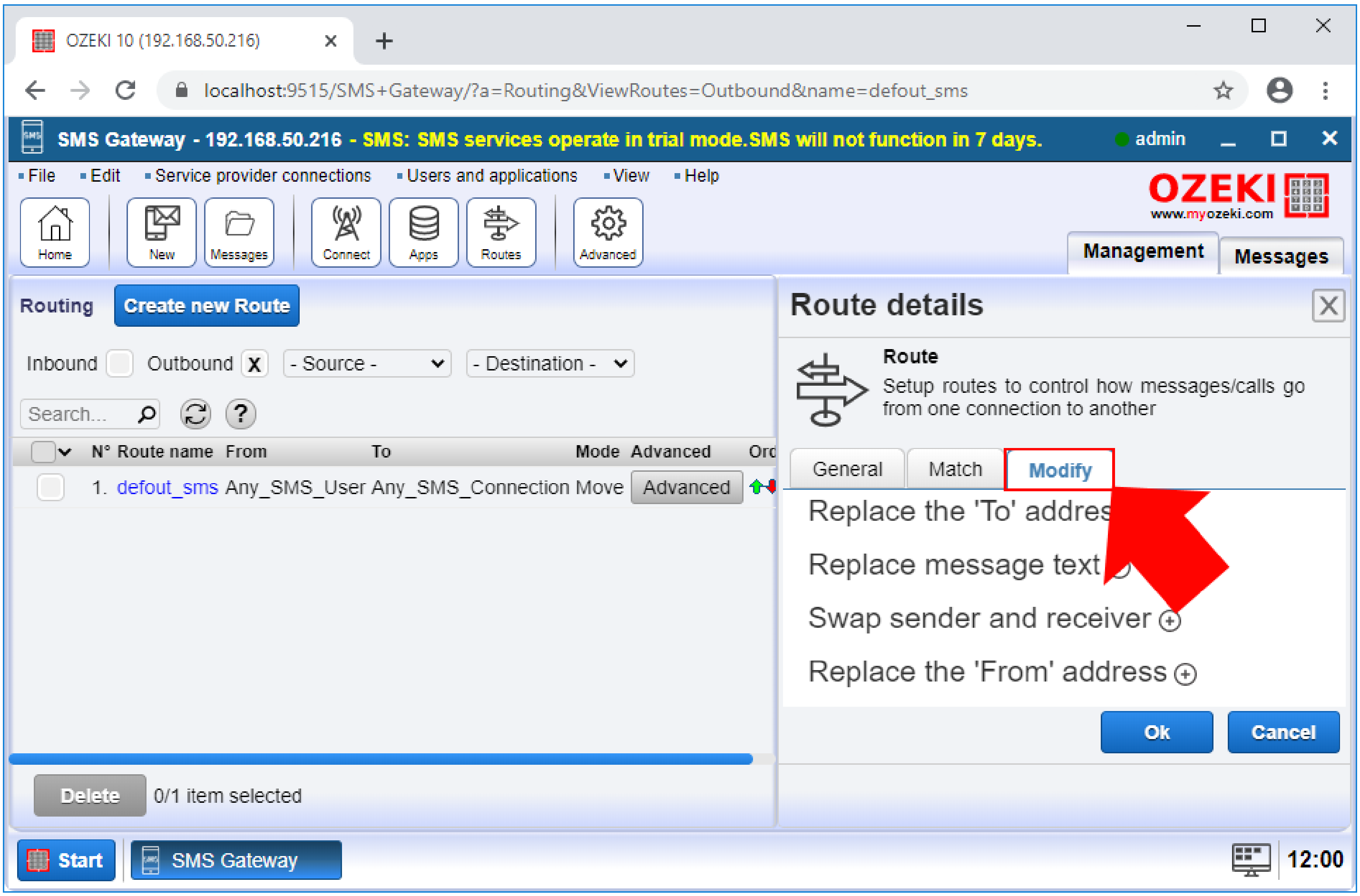
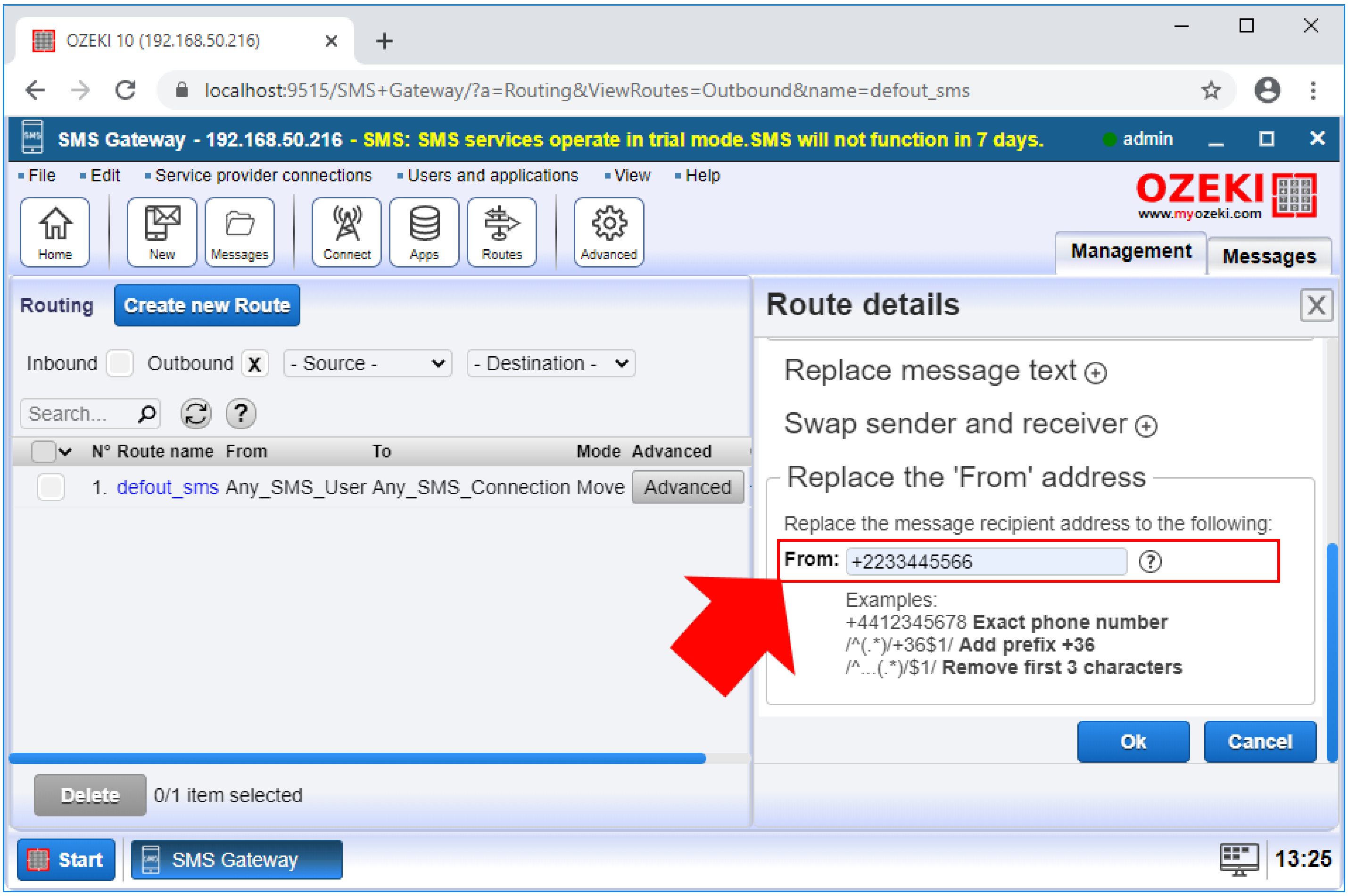

OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=Routing&ViewRoutes=Outbound&name=defout_sms
SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Routing
Create new Route
Inbound
Outbound
- Source -
- Destination -
Search...
N° Route name From To Mode Advanced Ord
1. defout_sms Any_SMS_User Any_SMS_Connection Move Advanced
Delete
0/1 item selected
Route details
Route
Setup routes to control how messages/calls go from one connection to another
General
Match
Modify
Replace the 'To' addres
Replace message text
Swap sender and receiver
Replace the 'From' address
Ok
Cancel
Start
SMS Gateway
12:00

Figure 2 - Select the Modify menu

## Step 2 - Configure the sender phone number

In the Modify menu, you can perform all operation that can modify outgoing messages. At this point, you need to select the 'Replace the 'From' address' submenu as Figure 3 demonstrates, and here, you can just type the phone number that you want to be the sender phone number of all outgoing messages.


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=Routing&ViewRoutes=Outbound&name=defout_sms
SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
OZEKI
www.myozeki.com
Management
Messages
Routing
Create new Route
Inbound
Outbound
- Source -
- Destination -
Search...
N° Route name From To Mode Advanced
1. defout_sms Any_SMS_User Any_SMS_Connection Move Advanced
Delete
0/1 item selected
Route details
Replace message text
Swap sender and receiver
Replace the 'From' address
Ok
Cancel
Start
SMS Gateway
13:25

Replace the message recipient address to the following:

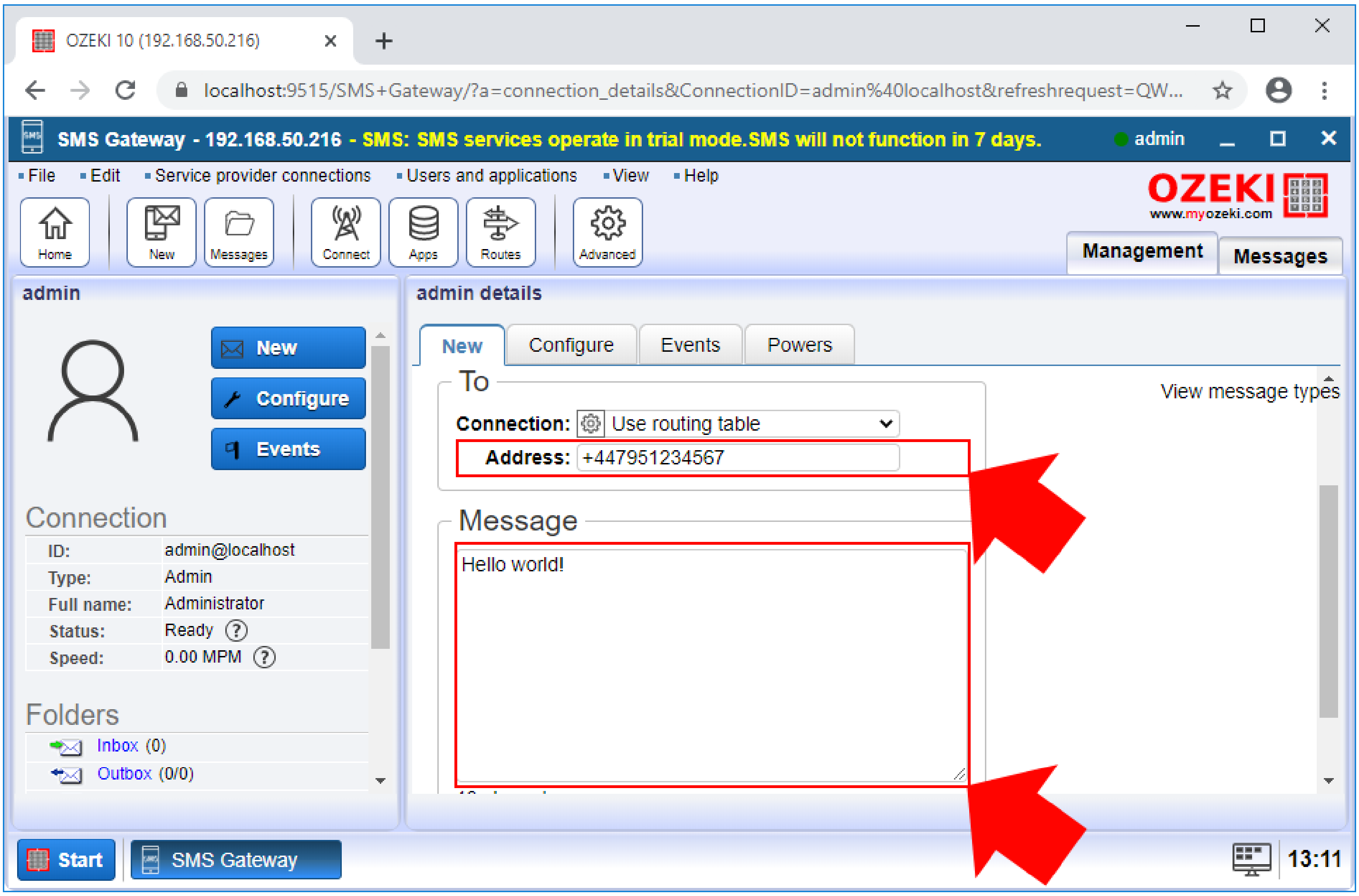
From: +2233445566

Examples:
+4412345678 **Exact phone number**
/^(.*)/+36$1/ **Add prefix +36**
/^...(.*)/$1/ **Remove first 3 characters**

Figure 3 - Configure the sender phone number

# Step 3 - Send a test message

After you configured the outgoing routing rule, you just need to write a test message. For that purpose, you can set up a SMPP client, which will forward the message to the recipient. Here, you can learn how to set up a SMPP client connection. To write the message, just open the admin account, and as you can see it in Figure 4, just write a simple test message.



OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=connection_details&ConnectionID=admin%40localhost&refreshrequest=QW...
SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
admin
New
Configure
Events
Start
SMS Gateway
13:11

admin details

New | Configure | Events | Powers

To

Connection: Use routing table

Address: +447951234567

View message types

Message

Hello world!

Connection

| | |
|---|---|
| ID: | admin@localhost |
| Type: | Admin |
| Full name: | Administrator |
| Status: | Ready |
| Speed: | 0.00 MPM |

Folders

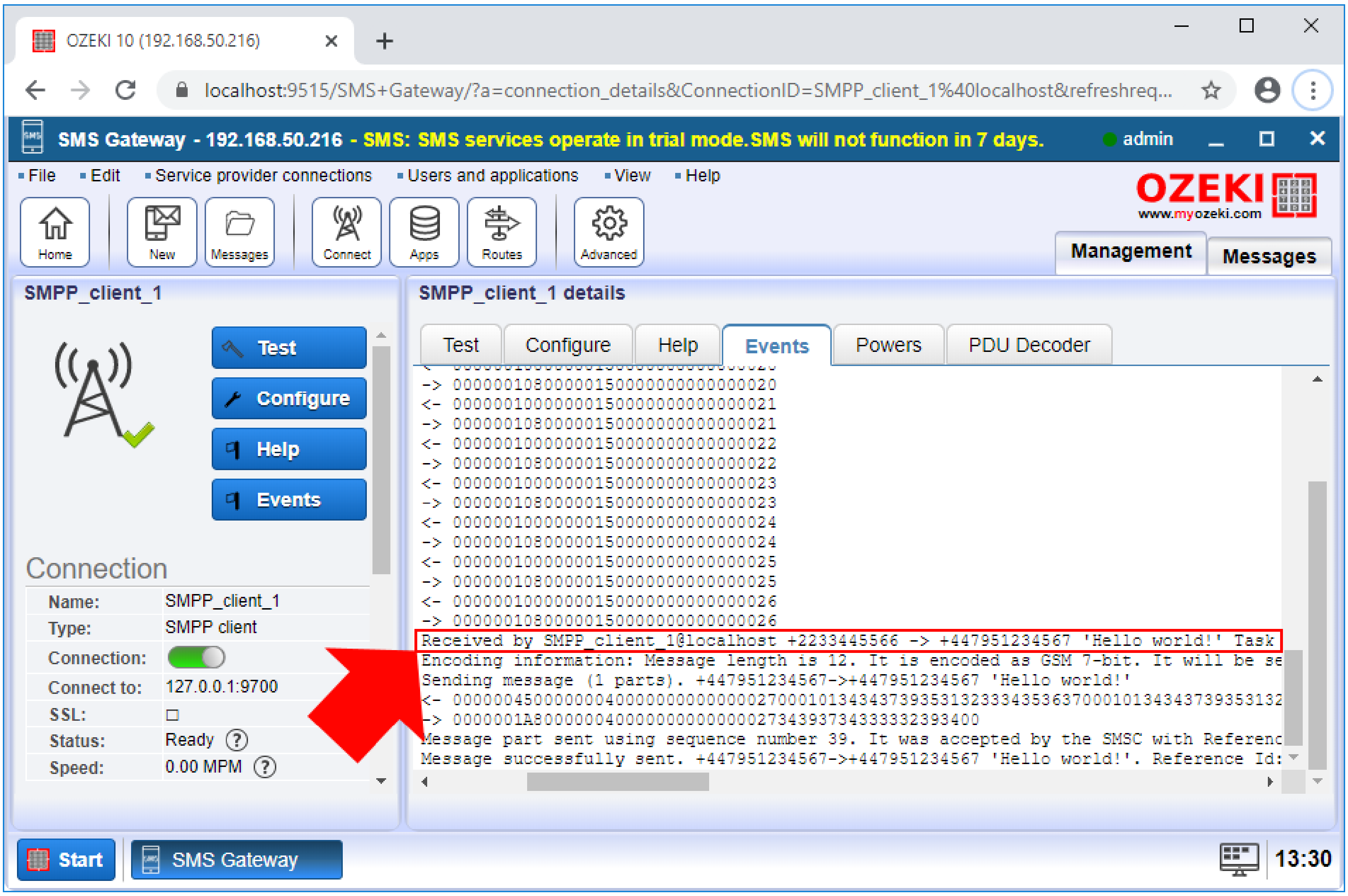
Inbox (0)

Outbox (0/0)

Figure 4 - Write a simple test message

## Step 4 - View the results in the event log

If you would like to check if the modifications worked, you need to open the SMPP client and select the Events tab. Here, you can see every event of the SMPP client connection. As you can see it in Figure 5, the routing rule changed the phone number of the sender.



OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=connection_details&ConnectionID=SMPP_client_1%40localhost&refreshreq...
SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
OZEKI
www.myozeki.com
Management
Messages
SMPP_client_1
Test
Configure
Help
Events
SMPP_client_1 details
Test
Configure
Help
Events
Powers
PDU Decoder
Start
SMS Gateway
13:30

Connection

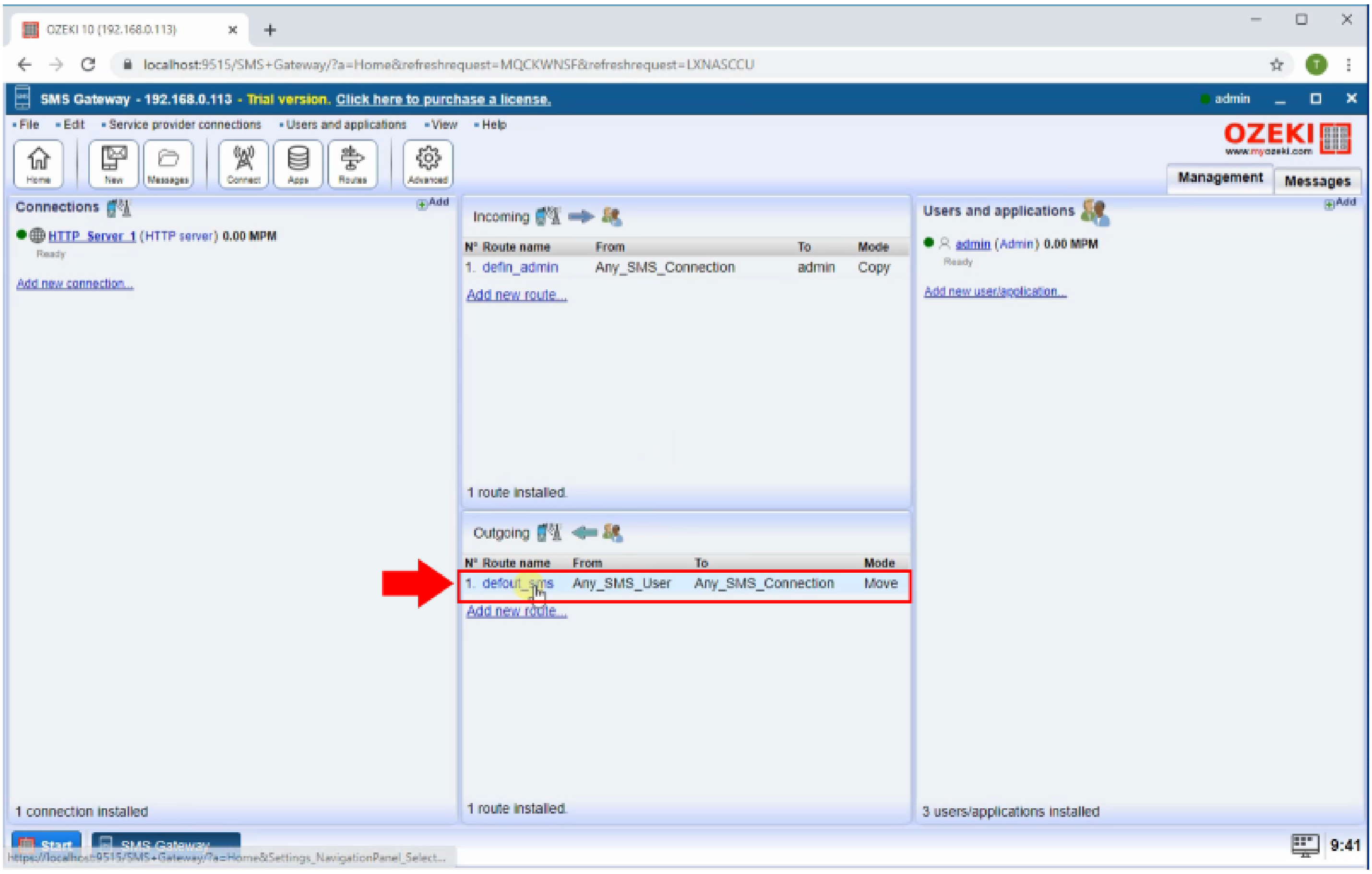
| | |
|---|---|
| Name: | SMPP_client_1 |
| Type: | SMPP client |
| Connection: | |
| Connect to: | 127.0.0.1:9700 |
| SSL: | □ |
| Status: | Ready |
| Speed: | 0.00 MPM |

```
-> 00000010800000150000000000000020
<- 00000010000000150000000000000021
-> 00000010800000150000000000000021
<- 00000010000000150000000000000022
-> 00000010800000150000000000000022
<- 00000010000000150000000000000023
-> 00000010800000150000000000000023
<- 00000010000000150000000000000024
-> 00000010800000150000000000000024
<- 00000010000000150000000000000025
-> 00000010800000150000000000000025
<- 00000010000000150000000000000026
-> 00000010800000150000000000000026
Received by SMPP_client_1@localhost +2233445566 -> +447951234567 'Hello world!' Task
Encoding information: Message length is 12. It is encoded as GSM 7-bit. It will be se
Sending message (1 parts). +447951234567->+447951234567 'Hello world!'
<- 0000004500000004000000000000002700010134343739353132333435363700010134343739353132
-> 0000001A80000004000000000000002734393734333332393400
Message part sent using sequence number 39. It was accepted by the SMSC with Referenc
Message successfully sent. +447951234567->+447951234567 'Hello world!'. Reference Id:
```

Figure 5 - See that the routing rule changed the sender phone number

# How to modify the recipient phone using regular expressions

In the Ozeki SMS Gateway you can change the recipient phone number in the outgoing routing rule. You are able to modify the rule by open it from the SMS Gateway Management consol Outigoing section as the Figure 1 shows.

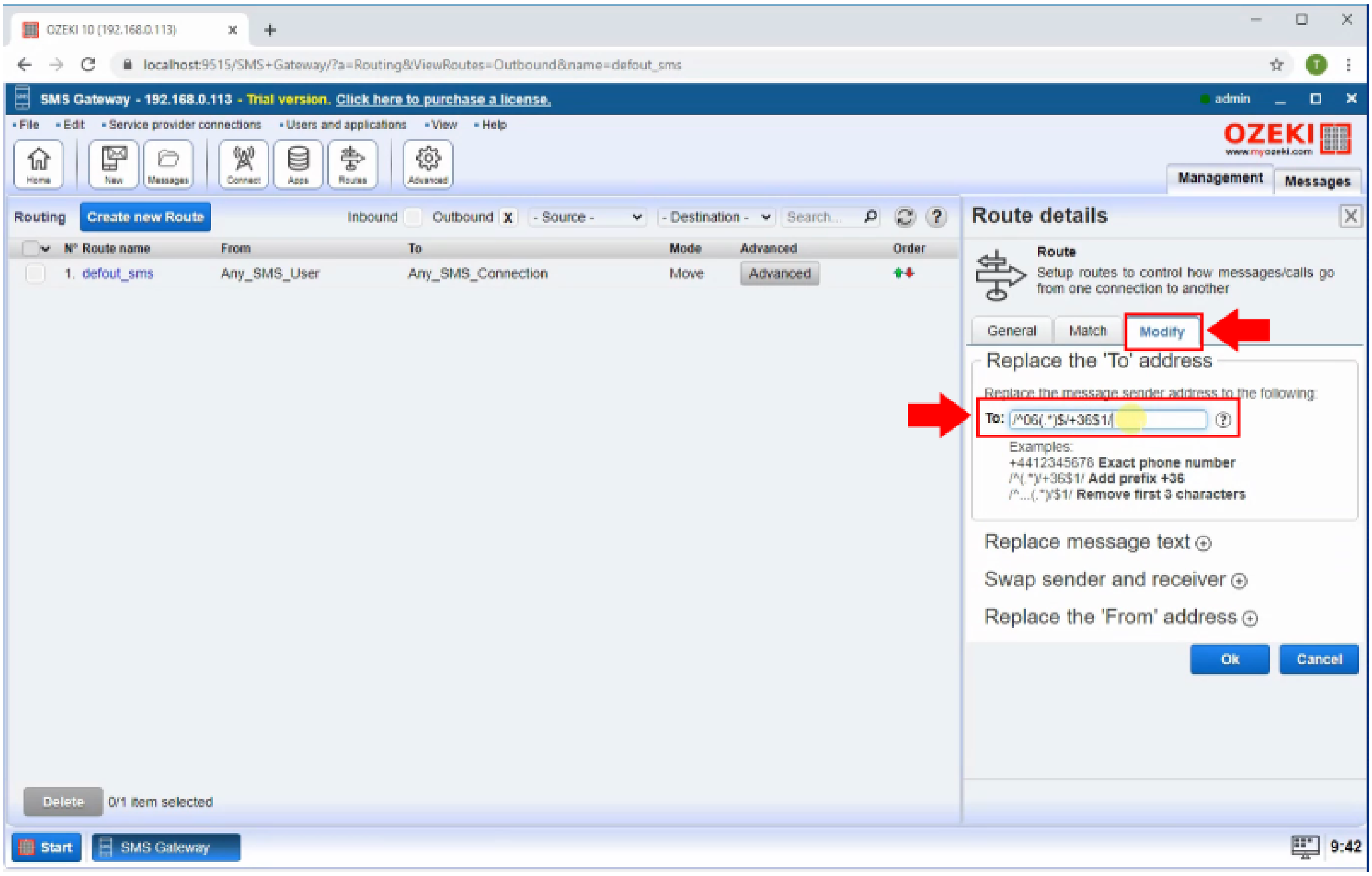


| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defin_admin | Any_SMS_Connection | admin | Copy |

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | default_sms | Any_SMS_User | Any_SMS_Connection | Move |

Figure 1 - Open outgoing route

In order to manipulat the recipient phone number open the actual routing rule's Modify tab and in it select the "Replace the 'TO' address" section. Here You can define an exact phone number where the messages will be send or you are able to add a regular expresson as you can see in the Figure 2. With this regexp if a phone number start with '06' it will replace to '+36'.



| | N° Route name | From | To | Mode | Advanced | Order |
|---|---|---|---|---|---|---|
| | 1. defout_sms | Any_SMS_User | Any_SMS_Connection | Move | Advanced | |

Route details

Route
Setup routes to control how messages/calls go from one connection to another

Replace the 'To' address

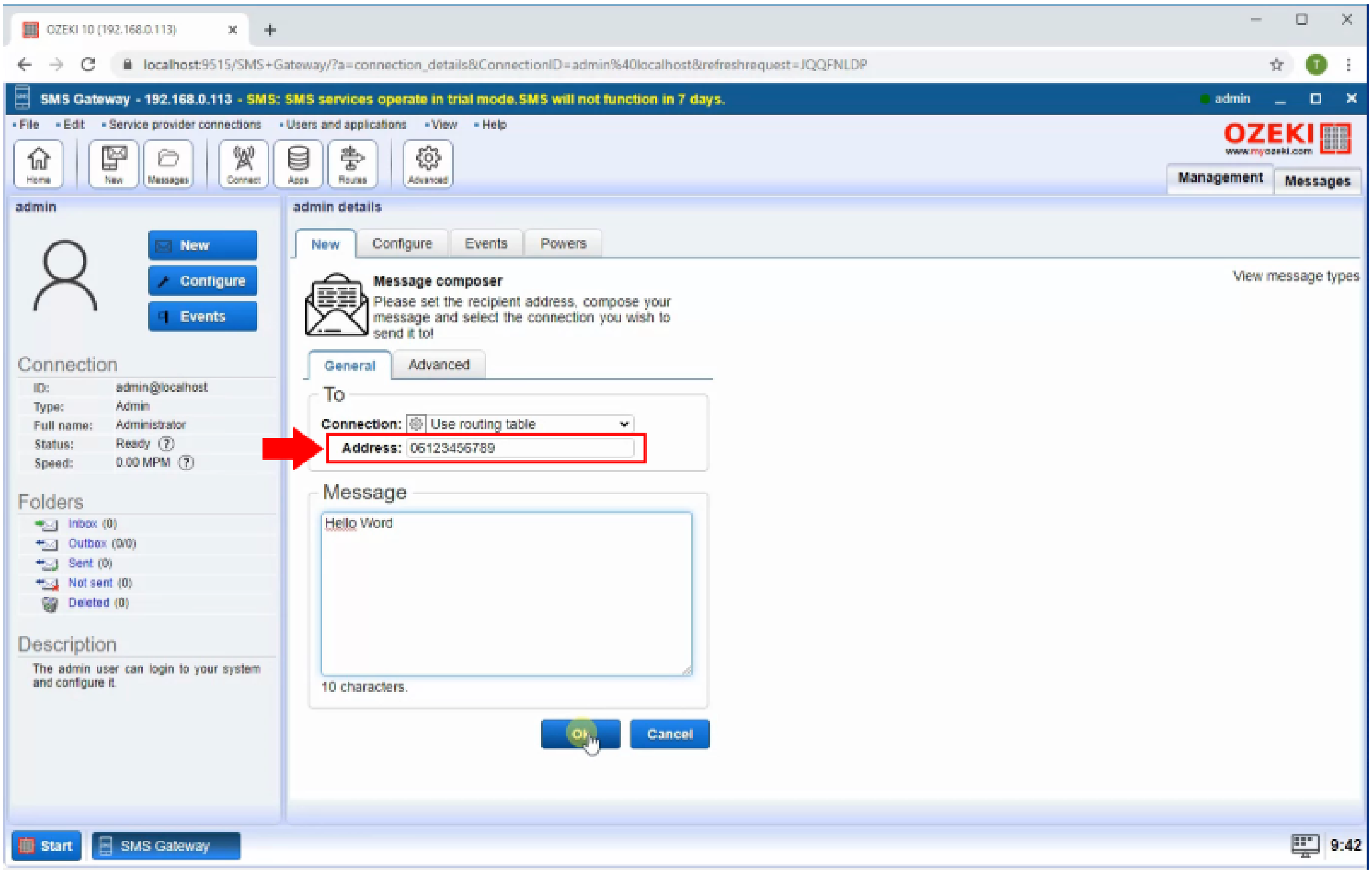
Replace the message sender address to the following:

To: /^06(.*)$/+36$1/

Examples:
+4412345678 **Exact phone number**
/^(.*)/+36$1/ **Add prefix +36**
/^...(.*)/$1/ **Remove first 3 characters**

Figure 2 - Manipulate 'TO' address

To test the phone number manipulation from the admin user send a test message to a phone number what is start with 06 (Figure 3).


OZEKI 10 (192.168.0.113)
localhost:9515/SMS+Gateway/?a=connection_details&ConnectionID=admin%40localhost&refreshrequest=JQQFNLDP
SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
OZEKI
www.myozeki.com
Management
Messages
admin
New
Configure
Events

Connection

| | |
|---|---|
| ID: | admin@localhost |
| Type: | Admin |
| Full name: | Administrator |
| Status: | Ready |
| Speed: | 0.00 MPM |

Folders

- Inbox (0)
- Outbox (0/0)
- Sent (0)
- Not sent (0)
- Deleted (0)

Description

The admin user can login to your system and configure it.

admin details

New
Configure
Events
Powers
View message types

Message composer
Please set the recipient address, compose your message and select the connection you wish to send it to!

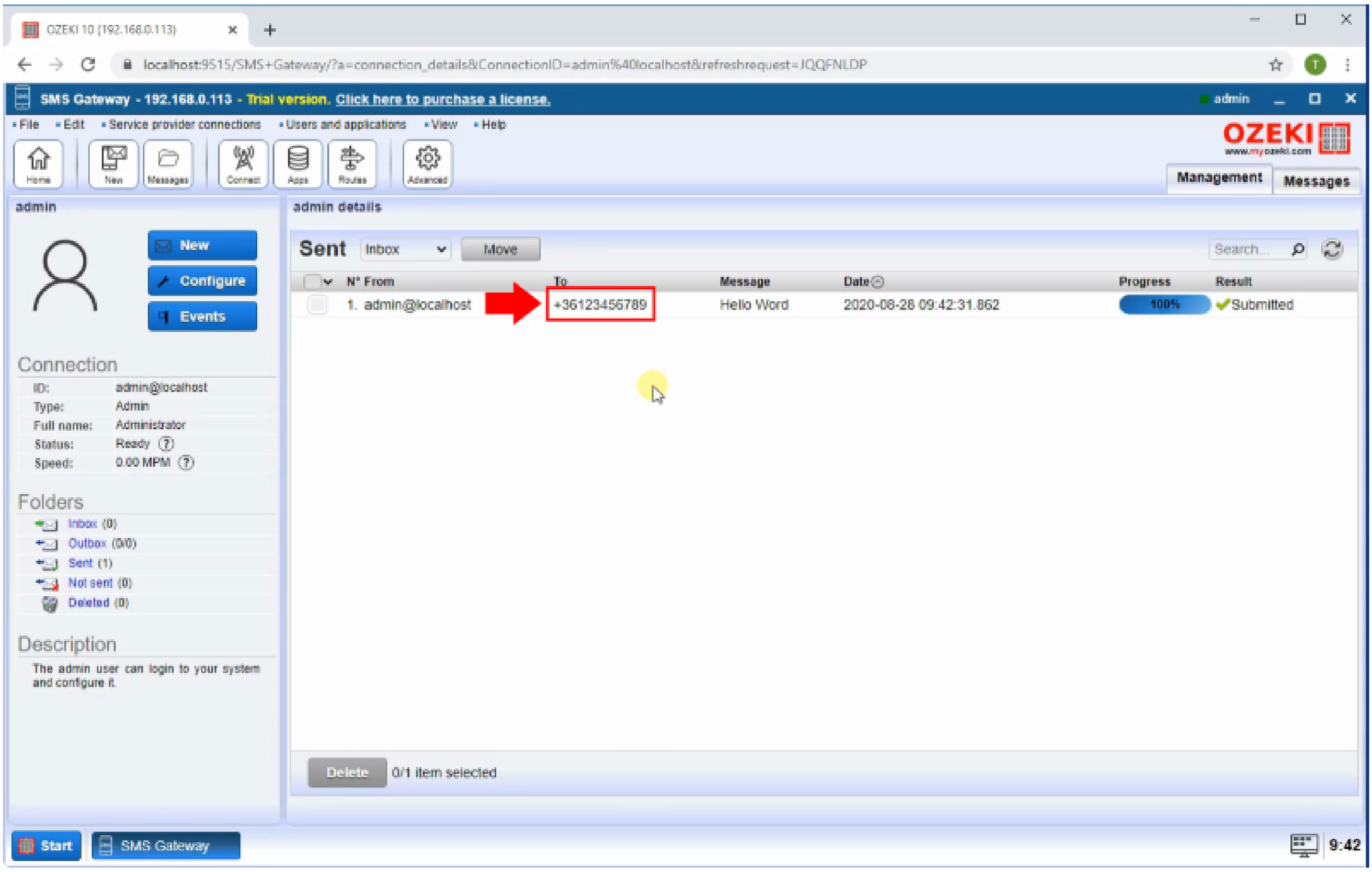
To
Connection: Use routing table
Address: 06123456789

Message
Hello Word

10 characters.

Figure 3 - Send test message

After the message is sent you will see in the admin user's sent folder the TO address will be changed to a phone number what starts with +36 as the Figure 4 shows.



localhost:9515/SMS+Gateway/?a=connection_details&ConnectionID=admin%40localhost&refreshrequest=JQQFNLDP

SMS Gateway - 192.168.0.113 - Trial version. Click here to purchase a license.

admin details

Sent

| N° | From | To | Message | Date | Progress | Result |
|---|---|---|---|---|---|---|
| 1. | admin@localhost | +36123456789 | Hello Word | 2020-08-28 09:42:31.862 | 100% | ✓Submitted |

0/1 item selected

Connection

| | |
|---|---|
| ID: | admin@localhost |
| Type: | Admin |
| Full name: | Administrator |
| Status: | Ready |
| Speed: | 0.00 MPM |

Folders

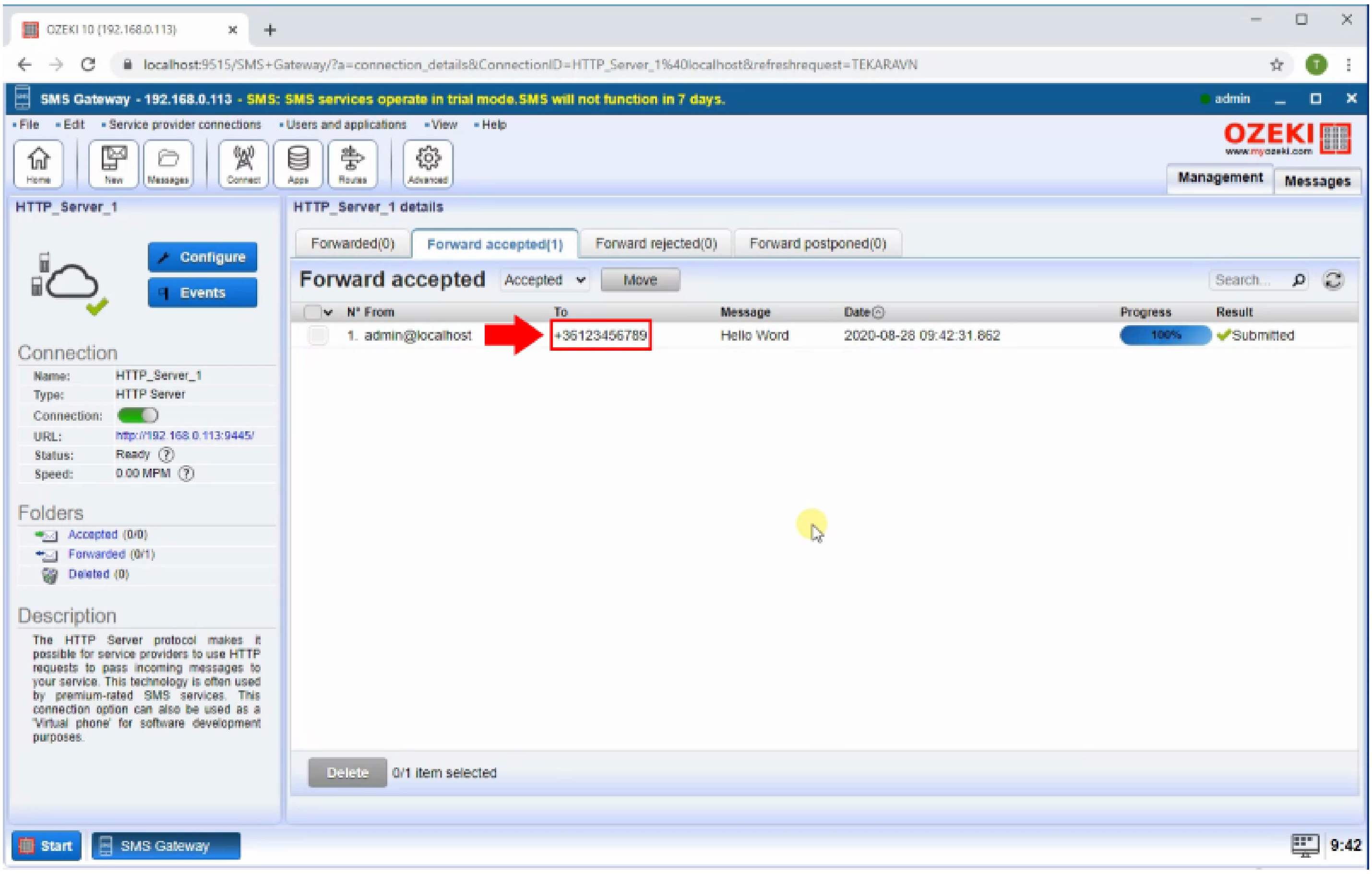
- Inbox (0)
- Outbox (0/0)
- Sent (1)
- Not sent (0)
- Deleted (0)

Description

The admin user can login to your system and configure it.

Figure 4 - TO address modified

You will also see in the service provider's Forwarded folder that the message is sent to the new phone number as you can see in the Figure 5.



Connection

Name: HTTP_Server_1
Type: HTTP Server
Connection:
URL: http://192.168.0.113:9445/
Status: Ready
Speed: 0.00 MPM

Folders

Accepted (0/0)
Forwarded (0/1)
Deleted (0)

Description

The HTTP Server protocol makes it possible for service providers to use HTTP requests to pass incoming messages to your service. This technology is often used by premium-rated SMS services. This connection option can also be used as a 'Virtual phone' for software development purposes.

| N° | From | To | Message | Date | Progress | Result |
|---|---|---|---|---|---|---|
| 1. | admin@localhost | +36123456789 | Hello Word | 2020-08-28 09:42:31.862 | 100% | Submitted |

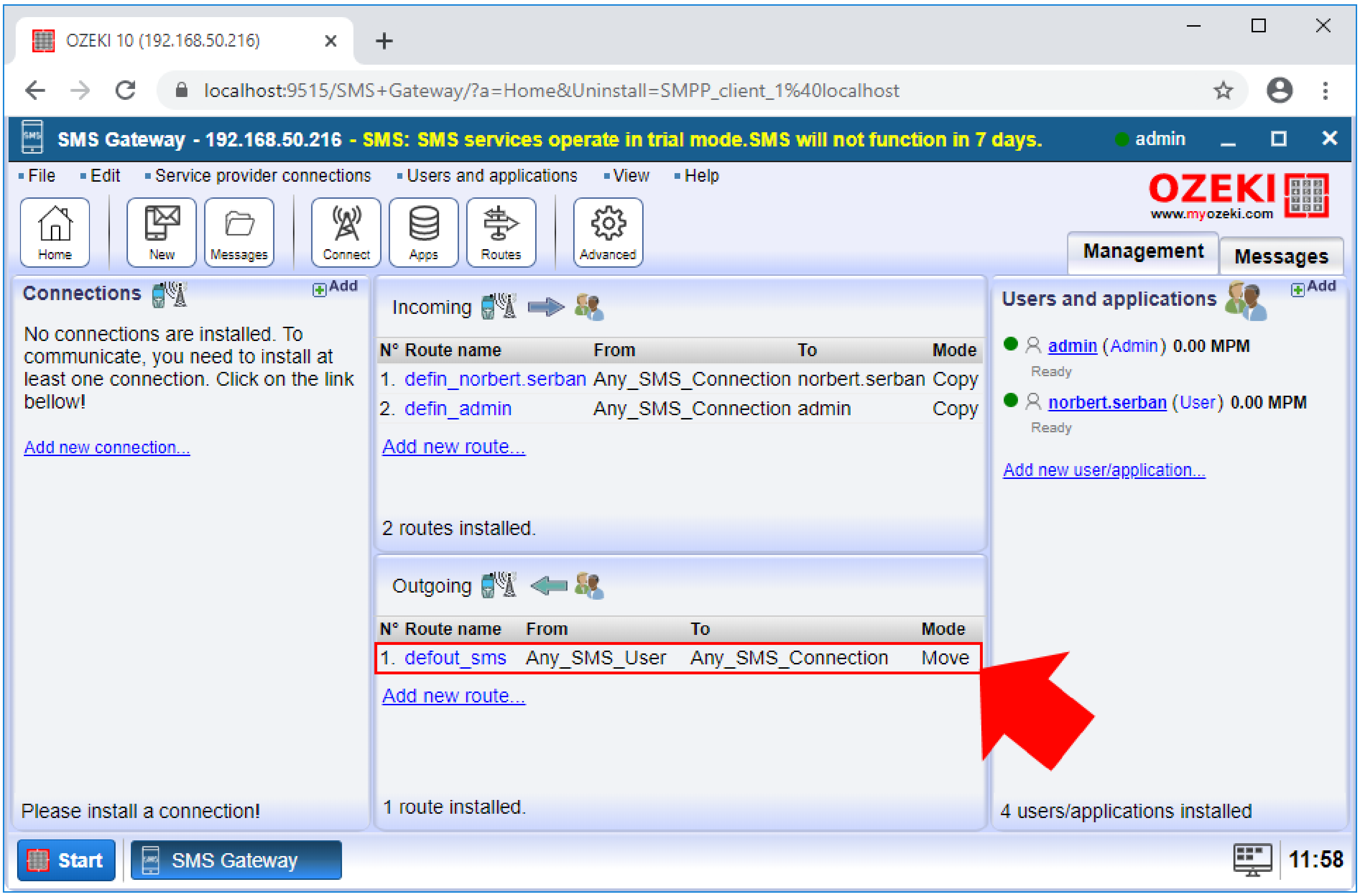
Figure 5 - Forwarded to the new phone number

# How to change the message text

***In this guide, you will be able to learn how you can modify the message by just replacing it with a predefined text message. This operation ensures that all message sent from your system is going to have the same text since the routing rule replaces the original message with the predefined one. The document makes it so easy for you to complete the modification setting since it contains a step by step guide that clearly describes every step of the operation and also demonstrates every step with a screenshot for the better understanding. So, let's start right now!***

## Step 1 - Open a routing rule

The first step is to select a route that you want to modify. If you are not familiar with routing rules, here you can learn about SMS routing in SMS Gateway. SMS Gateway creates default routes, so you can simply select the default outgoing route as you can see it in Figure 1.



SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode.SMS will not function in 7 days. admin

Connections

No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

Please install a connection!

Incoming

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defin_norbert.serban | Any_SMS_Connection | norbert.serban | Copy |
| 2. | defin_admin | Any_SMS_Connection | admin | Copy |

Add new route...

2 routes installed.

Outgoing

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move |

Add new route...

1 route installed.

Users and applications

admin (Admin) 0.00 MPM
Ready

norbert.serban (User) 0.00 MPM
Ready

Add new user/application...

4 users/applications installed

Figure 1 - Select a routing rule in SMS Gateway

By selecting a routing rule from the list of available routes, the next menu that you will be able to see is the details menu of the selected route. Here, you need to select the 'Modify' tab, as you can see it in Figure 2.

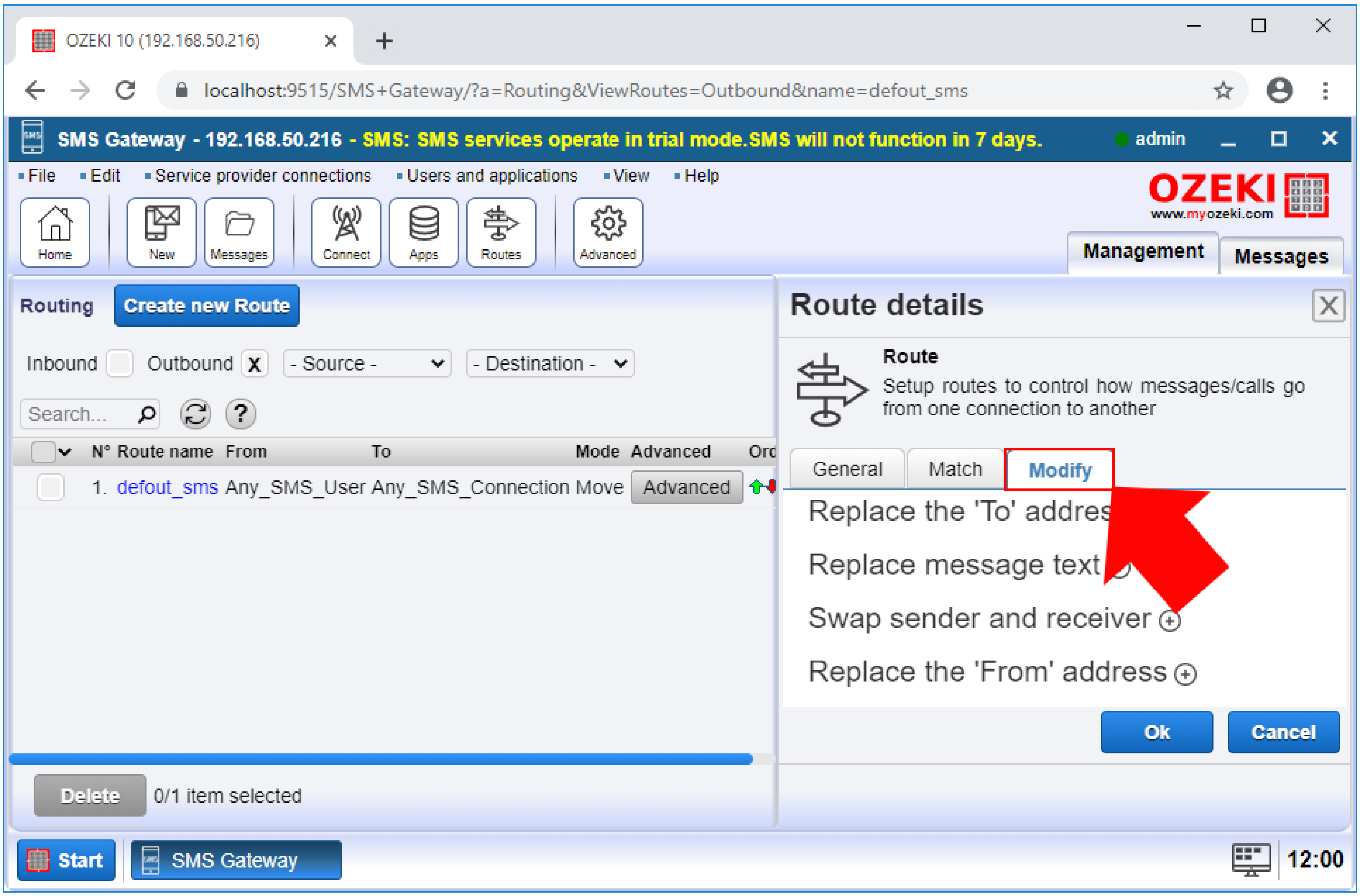
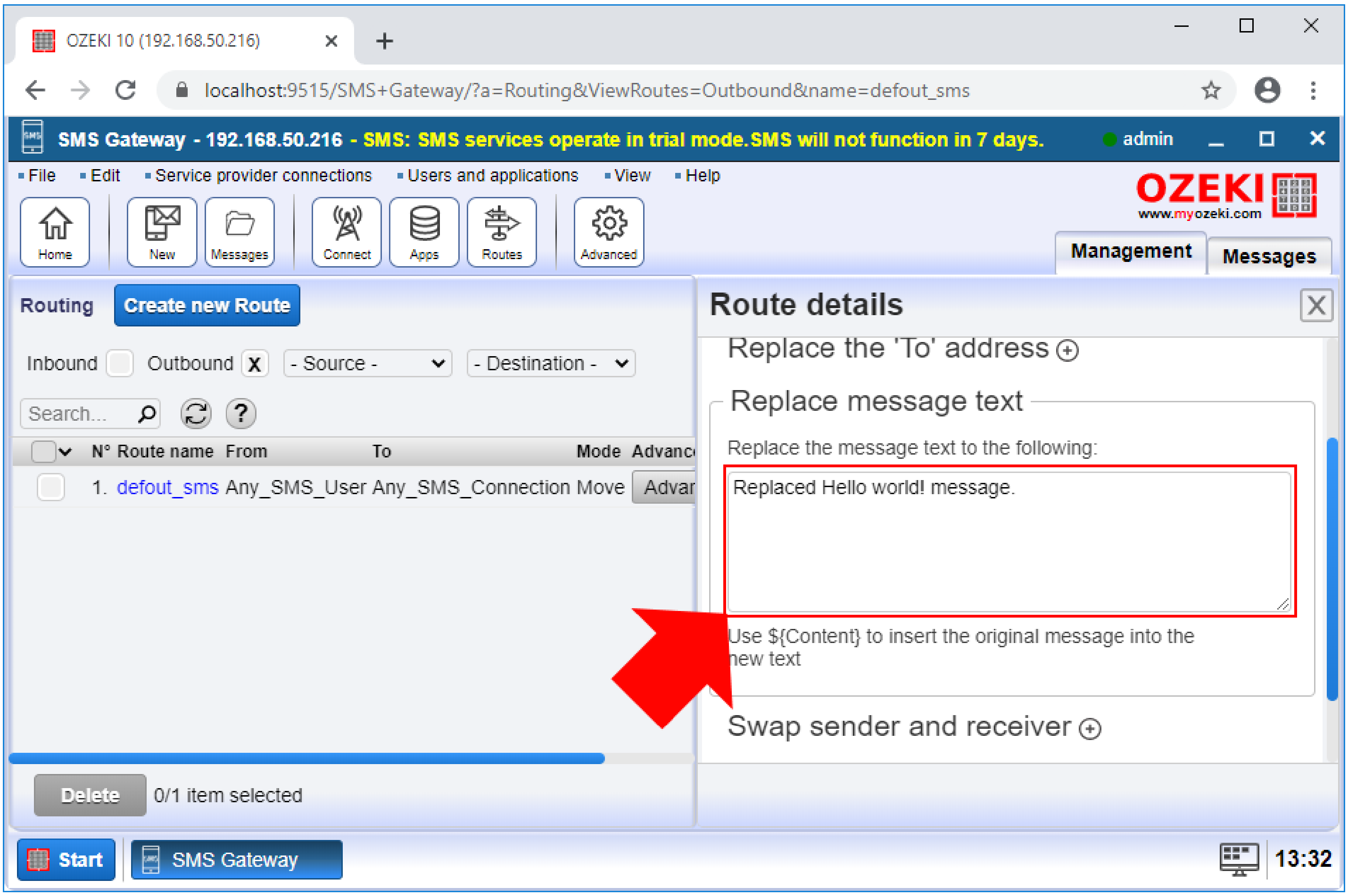

OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=Routing&ViewRoutes=Outbound&name=defout_sms
SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Routing
Create new Route
Inbound
Outbound
- Source -
- Destination -
Search...
N° Route name From To Mode Advanced Ord
1. defout_sms Any_SMS_User Any_SMS_Connection Move Advanced
Delete
0/1 item selected
Route details
Route
Setup routes to control how messages/calls go from one connection to another
General
Match
Modify
Replace the 'To' addres
Replace message text
Swap sender and receiver
Replace the 'From' address
Ok
Cancel
Start
SMS Gateway
12:00

Figure 2 - Select the Modify menu

## Step 2 - Configure the message text

In the Modify menu, you can configure the modification of the message text for the outgoing messages. To be able to change the text of the outgoing messages, select the 'Replace message text' submenu. Here, you can write a message like in Figure 3, so this text is going to replace every message that is going through that route. If you finished, you can just click OK.


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=Routing&ViewRoutes=Outbound&name=defout_sms
SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Routing
Create new Route
Inbound
Outbound
- Source -
- Destination -
Search...
N° Route name From To Mode Advanc
Delete
0/1 item selected
Start
SMS Gateway
13:32

Route details

Replace the 'To' address

Replace message text

Replace the message text to the following:

Replaced Hello world! message.

Use ${Content} to insert the original message into the new text

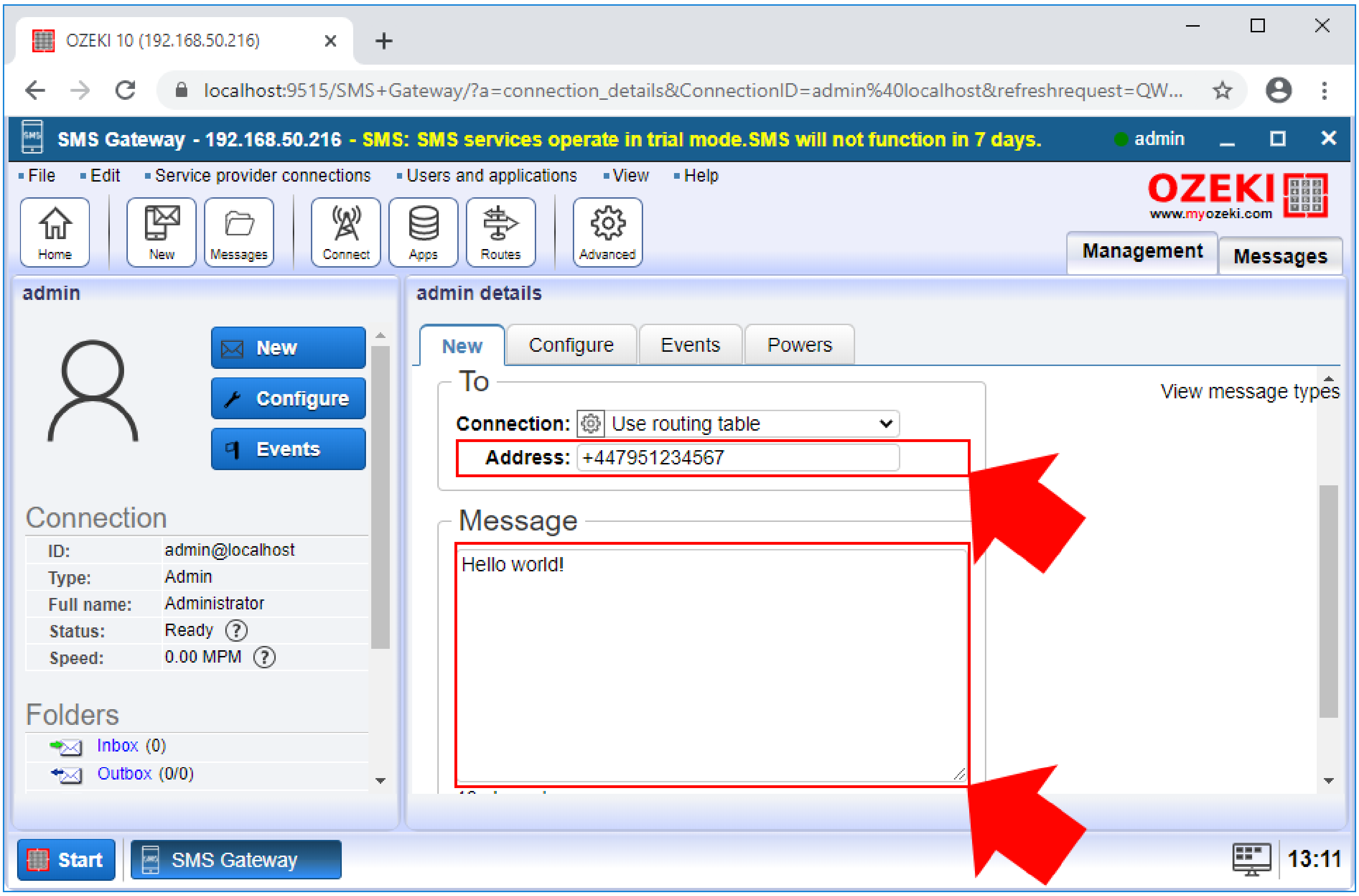
Swap sender and receiver

1. defout_sms Any_SMS_User Any_SMS_Connection Move Advar

Figure 3 - Configure the sender phone number

## Step 3 - Send a test message

After you changed the text of the message in the outgoing routing rule, you just need to write a test message. For that purpose, you can set up a SMPP client, which will forward the message to the recipient. Here, you can learn how to set up a SMPP client connection. To write the message, just open the admin account, and like in Figure 4, just write a simple test message.



ID: admin@localhost
Type: Admin
Full name: Administrator
Status: Ready
Speed: 0.00 MPM

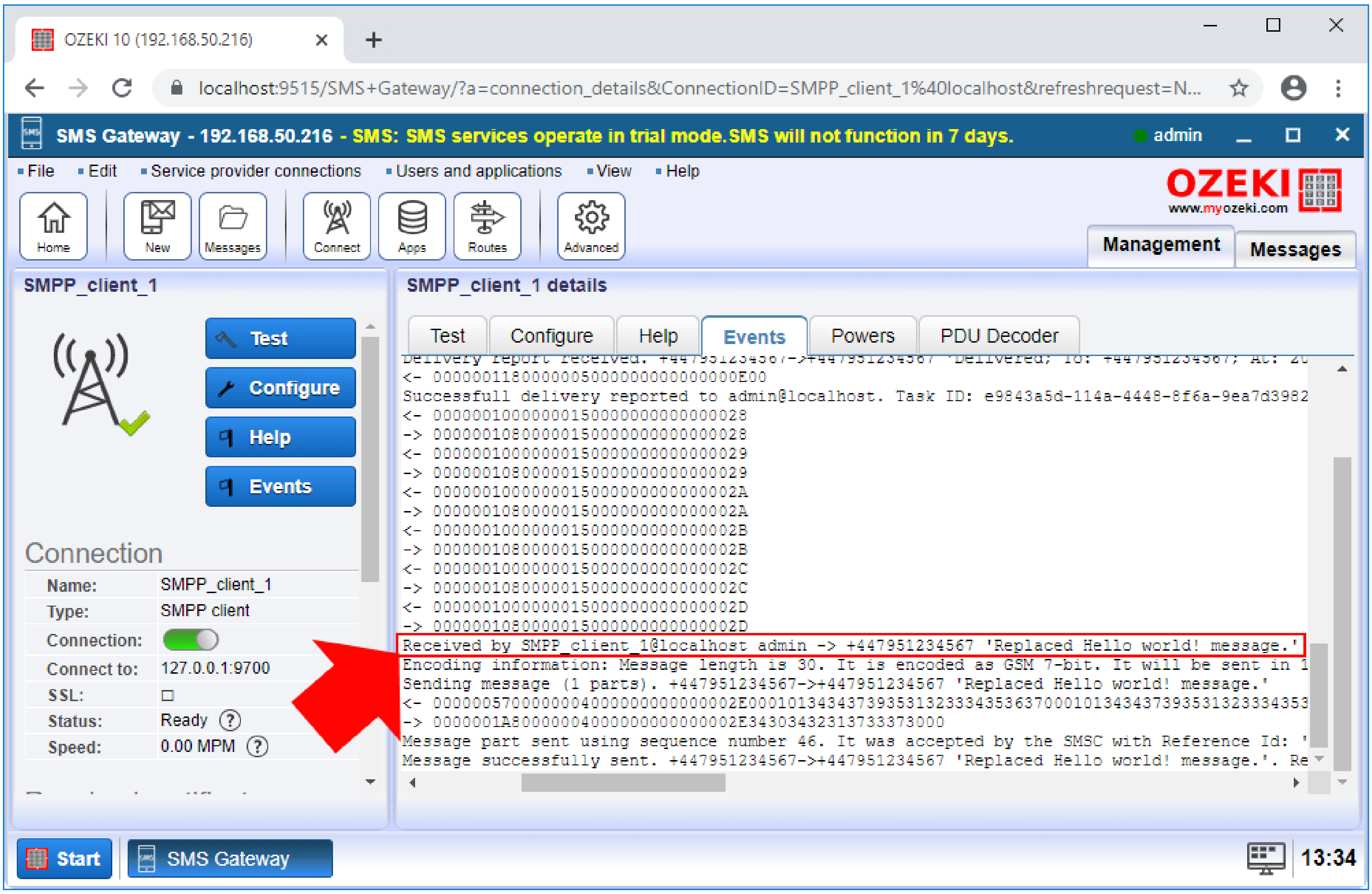
Connection: Use routing table
Address: +447951234567

Hello world!

Figure 4 - Write a simple test message

## Step 4 - View the results in the event log

To check if you have done everything right with modifying the message, you need to open the SMPP client and select the Events tab. Here, you can see every event of the SMPP client connection. Figure 5 shows that the original text is replaced by the specified message text.



OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=connection_details&ConnectionID=SMPP_client_1%40localhost&refreshrequest=N...
SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
SMPP_client_1
Test
Configure
Help
Events
SMPP_client_1 details
Test
Configure
Help
Events
Powers
PDU Decoder
Start
SMS Gateway
13:34

Connection

Name: SMPP_client_1
Type: SMPP client
Connection:
Connect to: 127.0.0.1:9700
SSL: □
Status: Ready
Speed: 0.00 MPM

```
Delivery report received. +447951234567->+447951234567 'Delivered; ID: +447951234567; At: 20
<- 00000011800000050000000000000000E00
Successfull delivery reported to admin@localhost. Task ID: e9843a5d-114a-4448-8f6a-9ea7d3982
<- 00000010000000150000000000000028
-> 00000010800000150000000000000028
<- 00000010000000150000000000000029
-> 00000010800000150000000000000029
<- 0000001000000015000000000000002A
-> 0000001080000015000000000000002A
<- 0000001000000015000000000000002B
-> 0000001080000015000000000000002B
<- 0000001000000015000000000000002C
-> 0000001080000015000000000000002C
<- 0000001000000015000000000000002D
-> 0000001080000015000000000000002D
Received by SMPP_client_1@localhost admin -> +447951234567 'Replaced Hello world! message.'
Encoding information: Message length is 30. It is encoded as GSM 7-bit. It will be sent in 1
Sending message (1 parts). +447951234567->+447951234567 'Replaced Hello world! message.'
<- 000000570000000400000000000000002E00010134343739353132333435363700010134343739353132333435З
-> 0000001A80000004000000000000002E343034323137333373000
Message part sent using sequence number 46. It was accepted by the SMSC with Reference Id: '
Message successfully sent. +447951234567->+447951234567 'Replaced Hello world! message.'. Re
```

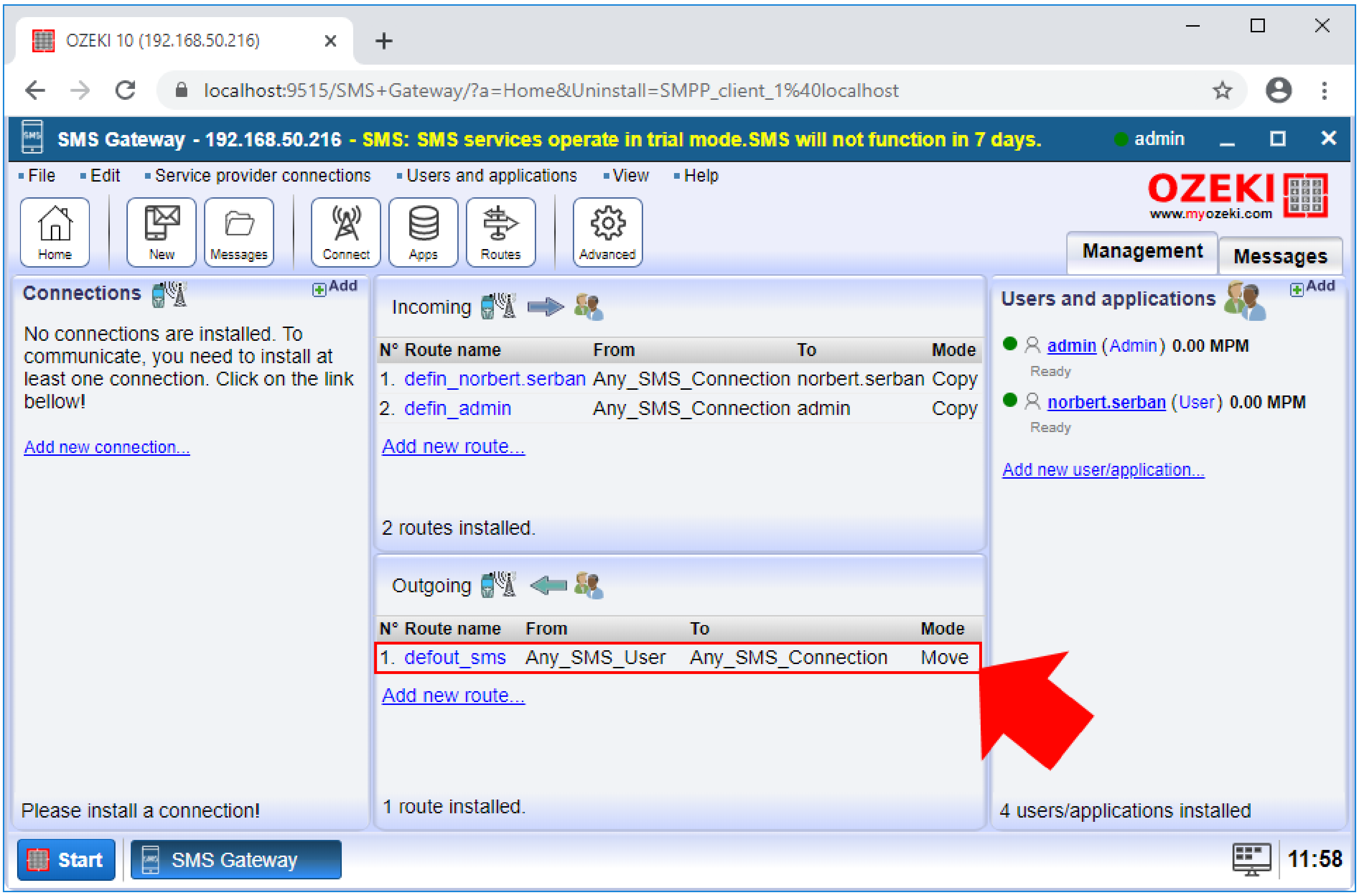
Figure 5 - See that the routing rule changed the message text

# How to add text to the end of the message

***This section is going to demonstrate that you can modify the message text by keeping the original text too. This solution is quite useful when you want every message to have the same ending sentence, so you don't need to write that to every single message, the routing rule adds it automatically. You will be able to see how you need to set up the routing rule to add text at the end of the message. The document demonstrates every step with a short description about what you have to do next. You can also find screenshots at each step to be able to visually check if you have done everything correct.***

## Step 1 - Open a routing rule

The first step is to select a route that you want to modify. If you are not familiar with routing rules, here you can learn about SMS routing in SMS Gateway. SMS Gateway creates default routes, so you can simply select the default outgoing route as you can see it in Figure 1.



OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=Home&Uninstall=SMPP_client_1%40localhost
SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
OZEKI
www.myozeki.com
Management
Messages
Start
SMS Gateway
11:58

Connections

No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

Please install a connection!

Incoming

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defin_norbert.serban | Any_SMS_Connection | norbert.serban | Copy |
| 2. | defin_admin | Any_SMS_Connection | admin | Copy |

Add new route...

2 routes installed.

Outgoing

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move |

Add new route...

1 route installed.

Users and applications

● admin (Admin) 0.00 MPM
Ready

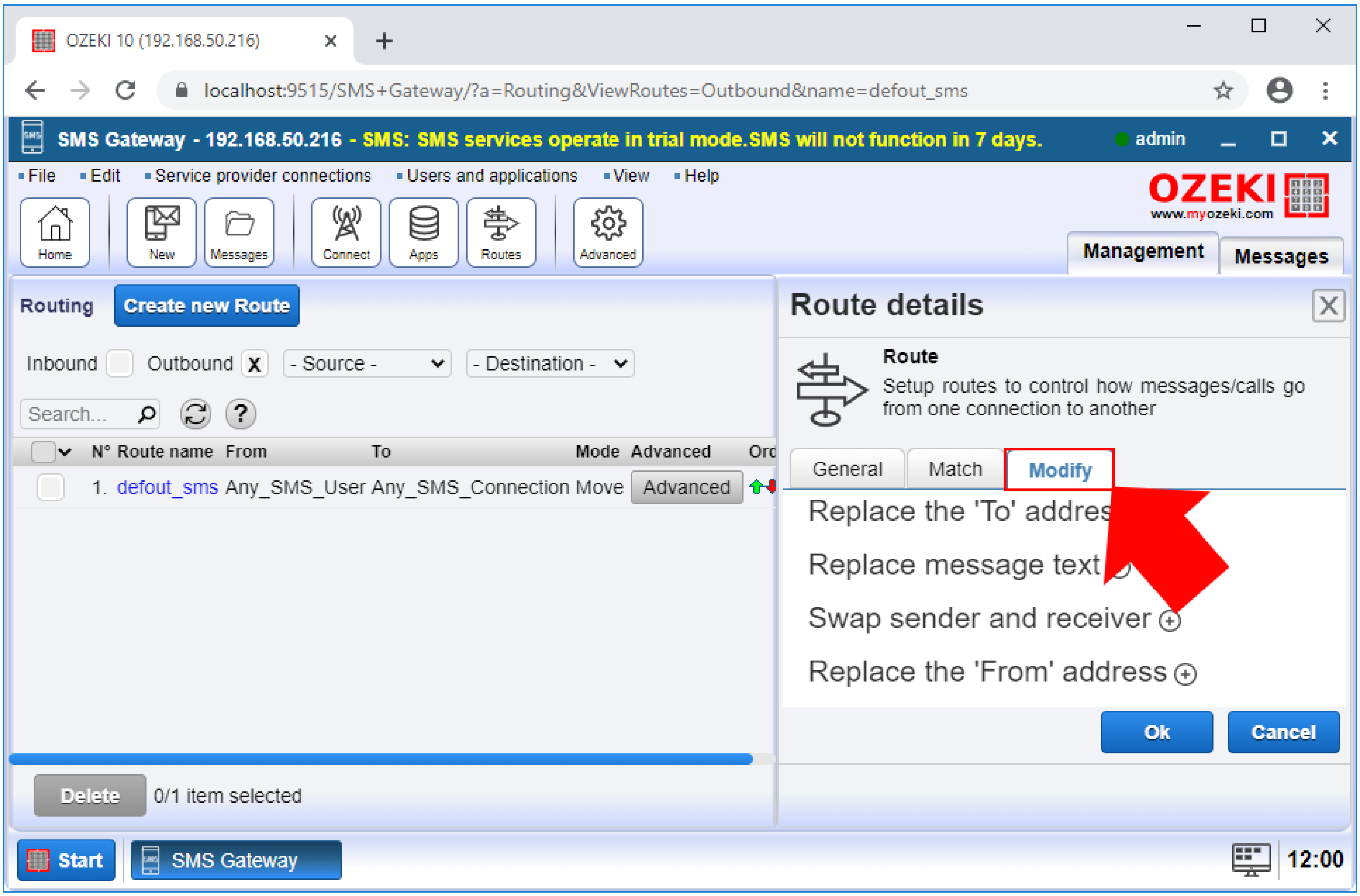
● norbert.serban (User) 0.00 MPM
Ready

Add new user/application...

4 users/applications installed

Figure 1 - Select a routing rule in SMS Gateway

Here, you can see the setting menu of the routing rule. To be able to modify the messages on this route, you just need to select the Modify tab as you can see it in Figure 2.

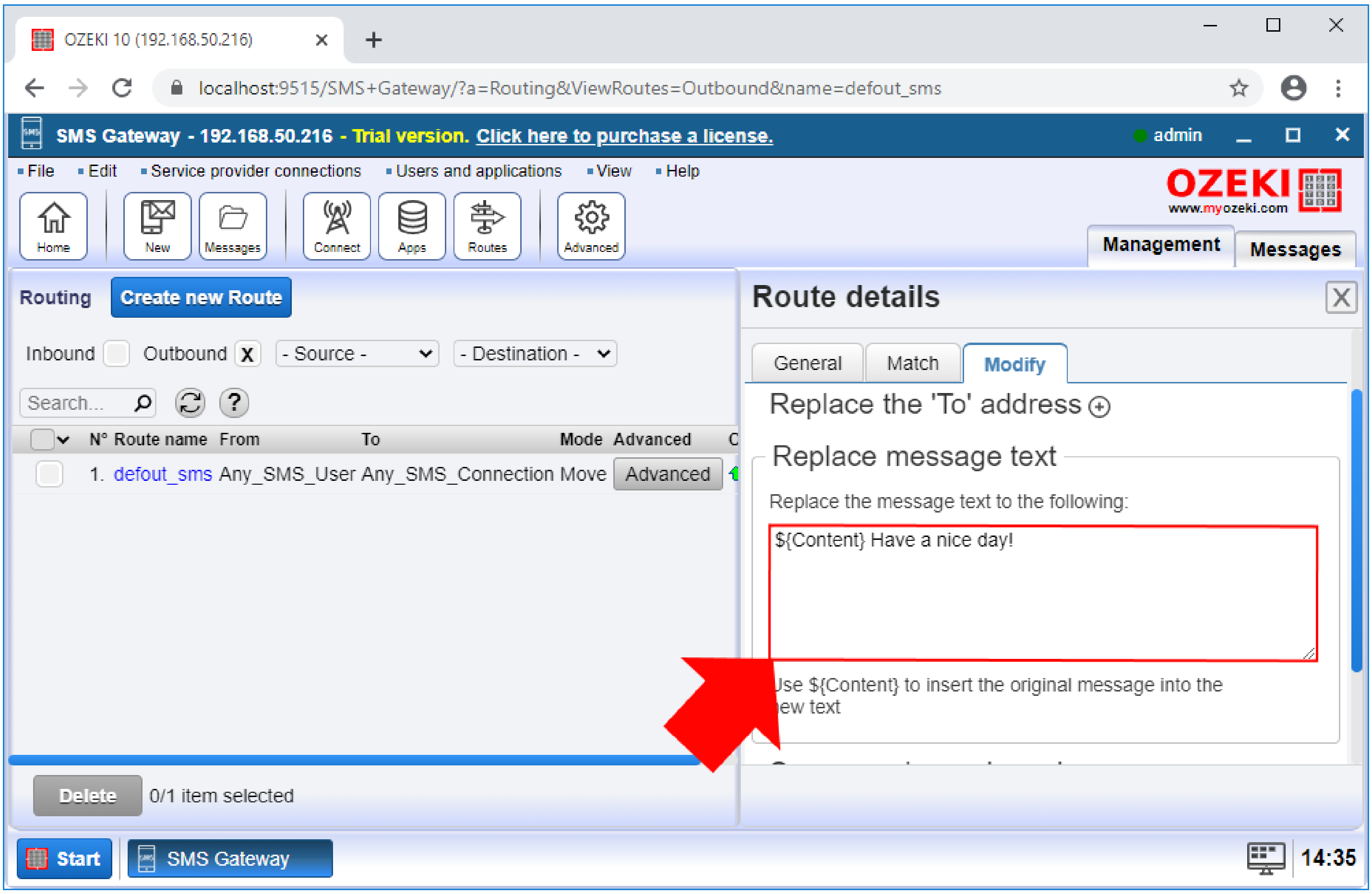


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=Routing&ViewRoutes=Outbound&name=defout_sms
SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Routing
Create new Route
Inbound
Outbound
- Source -
- Destination -
Search...
N° Route name From To Mode Advanced Ord
1. defout_sms Any_SMS_User Any_SMS_Connection Move Advanced
Delete
0/1 item selected
Route details
Route
Setup routes to control how messages/calls go from one connection to another
General
Match
Modify
Replace the 'To' addres
Replace message text
Swap sender and receiver
Replace the 'From' address
Ok
Cancel
Start
SMS Gateway
12:00

Figure 2 - Select the Modify menu

## Step 2 - Configure the message to add a postfix

In the Modify menu, you have to select the 'Replace message text' submenu to be able to modify the message. To keep the original message, you can use the '${Content}' tag as in Figure 3. This tag represents the original message, so if you write something after that, the routing rule appends it to the original message. To save this setting, click on OK.


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=Routing&ViewRoutes=Outbound&name=defout_sms
SMS Gateway - 192.168.50.216 - Trial version. Click here to purchase a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
OZEKI
www.myozeki.com
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Routing
Create new Route
Inbound
Outbound
- Source -
- Destination -
Search...
N° Route name From To Mode Advanced
Delete
Start
SMS Gateway
14:35

1. defout_sms Any_SMS_User Any_SMS_Connection Move Advanced

0/1 item selected

Route details

General | Match | Modify

Replace the 'To' address

Replace message text

Replace the message text to the following:

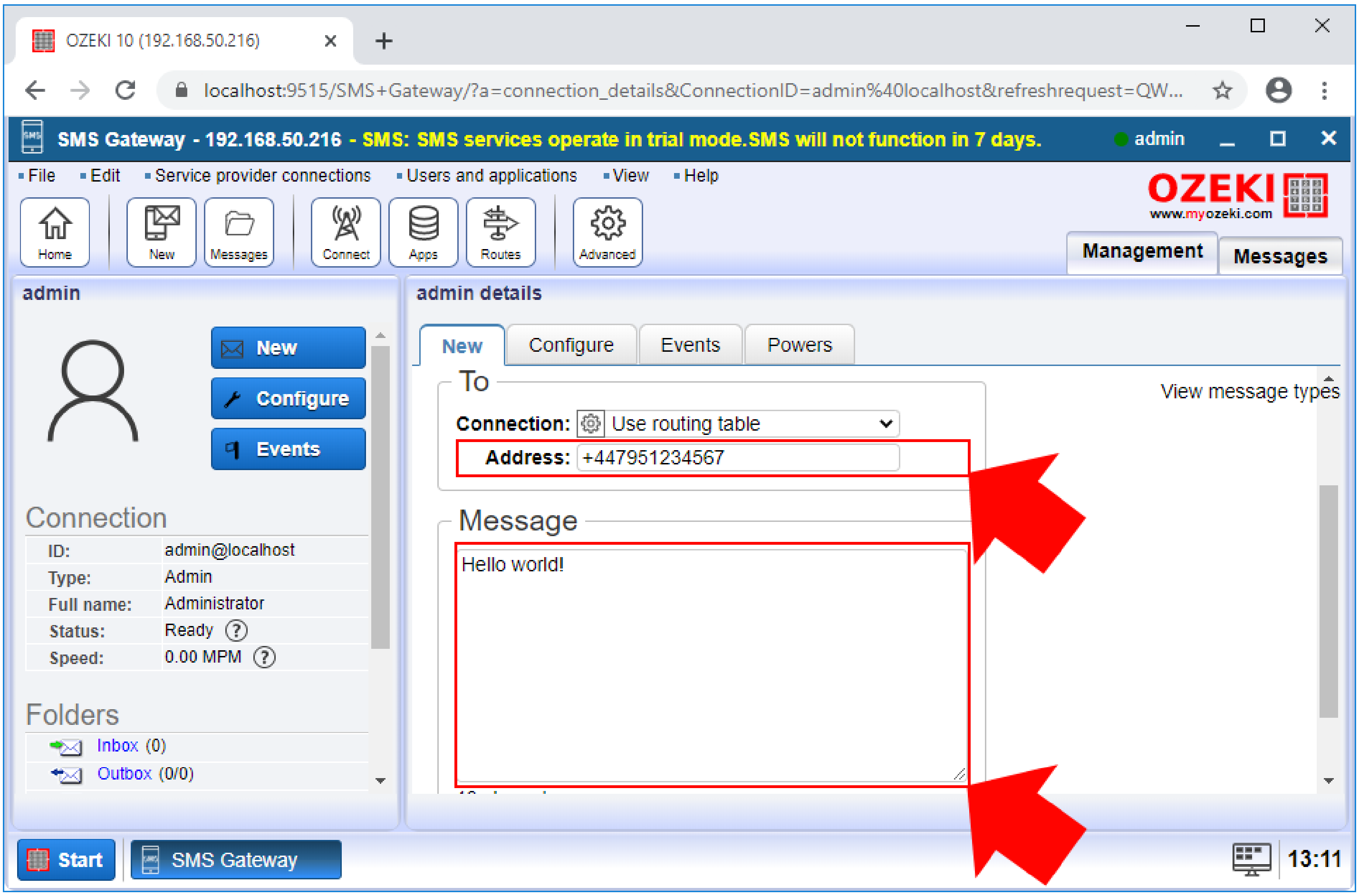
${Content} Have a nice day!

Use ${Content} to insert the original message into the new text

Figure 3 - Configure the swap of sender and receiver

# Step 3 - Send a test message

After you configured the outgoing routing rule, you just need to write a test message. For that purpose, you can set up a SMPP client, which will forward the message to the recipient. Here, you can learn how to set up a SMPP client connection. To write the message, just open the admin account, and like in Figure 4, just write a simple test message.



OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=connection_details&ConnectionID=admin%40localhost&refreshrequest=QW...
SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
admin
New
Configure
Events
admin details
New
Configure
Events
Powers
View message types
Connection
Folders
Inbox (0)
Outbox (0/0)
Start
SMS Gateway
13:11

To

Connection: Use routing table

Address: +447951234567

Message

Hello world!

ID: admin@localhost

Type: Admin

Full name: Administrator

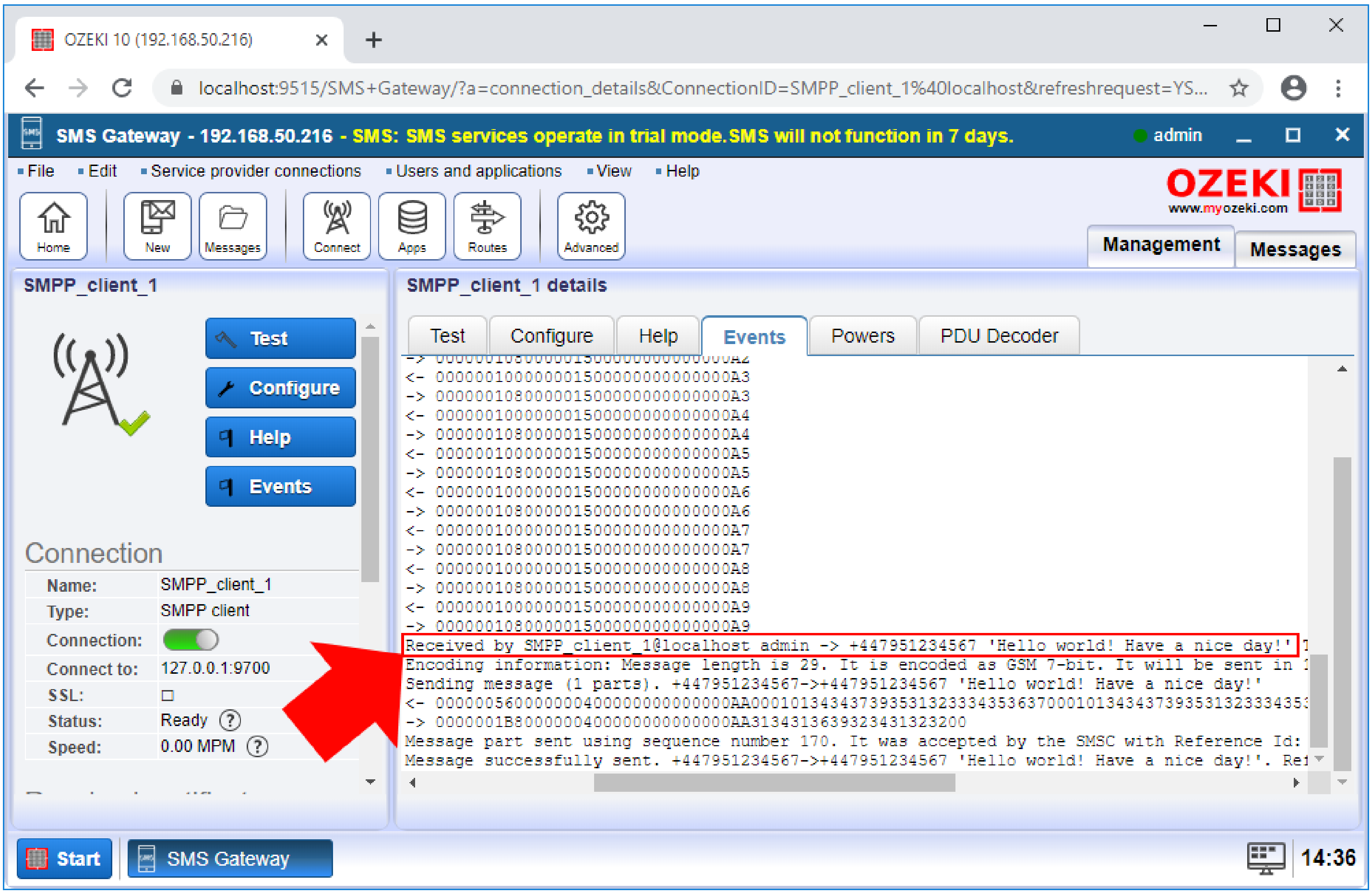
Status: Ready

Speed: 0.00 MPM

Figure 4 - Write a simple test message

## Step 4 - View the results in the event log

You can simply check if you have done everything correctly by just opening the SMPP client and here, select the Events tab. Here, you can see every event of the SMPP client connection. Figure 5 shows that the original text is merged with the text, that you wrote before so it will be sent together.

 
OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=connection_details&ConnectionID=SMPP_client_1%40localhost&refreshrequest=YS...
SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
OZEKI
www.myozeki.com
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
SMPP_client_1
Test
Configure
Help
Events
SMPP_client_1 details
Test
Configure
Help
Events
Powers
PDU Decoder
Start
SMS Gateway
14:36

Connection

| | |
|---|---|
| Name: | SMPP_client_1 |
| Type: | SMPP client |
| Connection: | ● |
| Connect to: | 127.0.0.1:9700 |
| SSL: | □ |
| Status: | Ready ⓘ |
| Speed: | 0.00 MPM ⓘ |

```
-> 000000108000001500000000000000A2
<- 000000100000001500000000000000A3
-> 000000108000001500000000000000A3
<- 000000100000001500000000000000A4
-> 000000108000001500000000000000A4
<- 000000100000001500000000000000A5
-> 000000108000001500000000000000A5
<- 000000100000001500000000000000A6
-> 000000108000001500000000000000A6
<- 000000100000001500000000000000A7
-> 000000108000001500000000000000A7
<- 000000100000001500000000000000A8
-> 000000108000001500000000000000A8
<- 000000100000001500000000000000A9
-> 000000108000001500000000000000A9
Received by SMPP_client_1@localhost admin -> +447951234567 'Hello world! Have a nice day!' T
Encoding information: Message length is 29. It is encoded as GSM 7-bit. It will be sent in 1
Sending message (1 parts). +447951234567->+447951234567 'Hello world! Have a nice day!'
<- 000000560000000400000000000000AA0001013434373935313233343536370001013434373935313233343535
-> 0000001B8000000400000000000000AA3134313639323431323200
Message part sent using sequence number 170. It was accepted by the SMSC with Reference Id:
Message successfully sent. +447951234567->+447951234567 'Hello world! Have a nice day!'. Ref
```

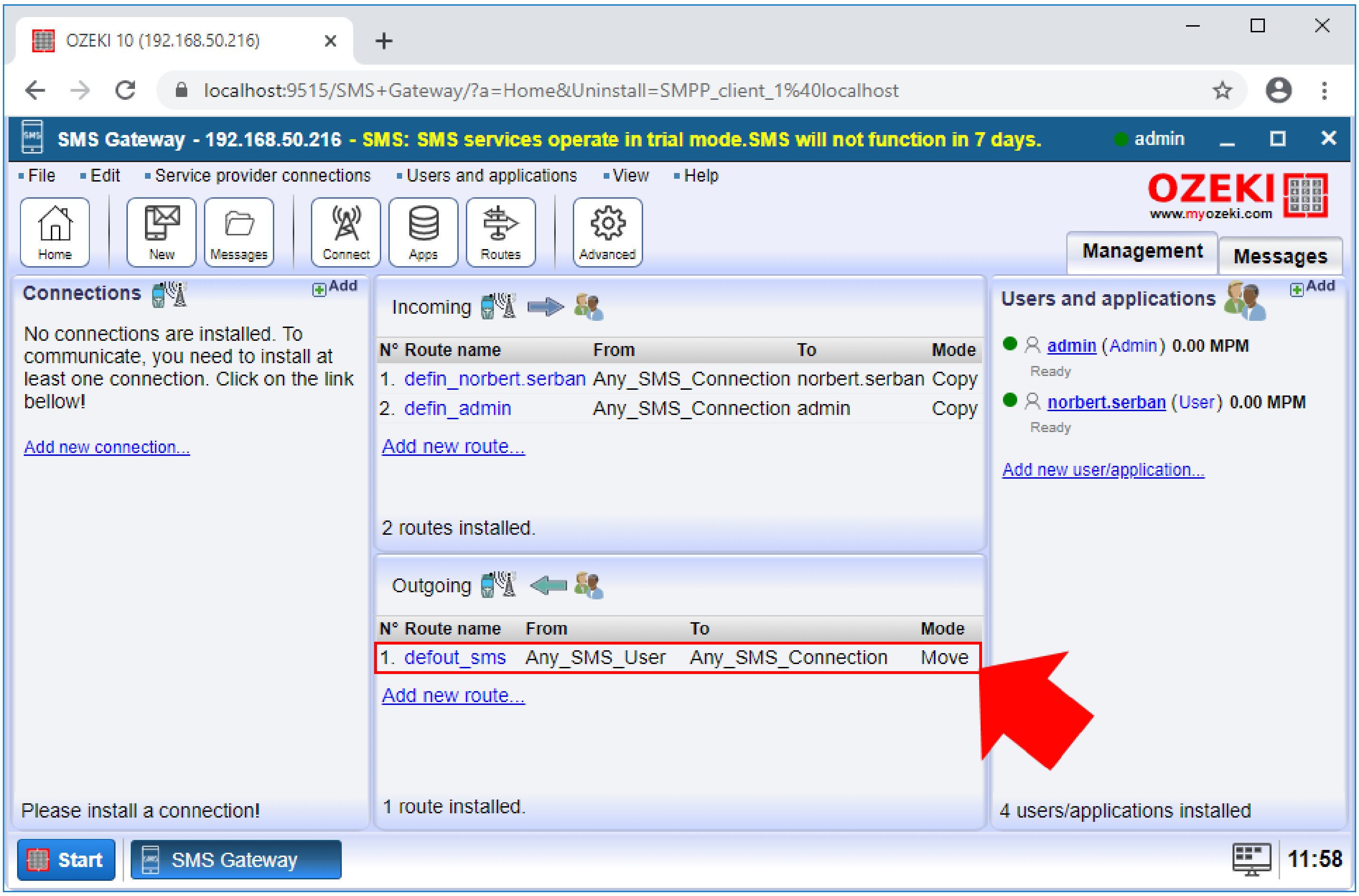
Figure 5 - See that the routing rule changed the message text

# How to swap the sender and recipient phone numbers

***The next document is about to show how easy you can swap the sender and the recipient phone numbers, so the sender becomes the recipient and the recipient becomes the sender. This operation is great to generate an autoreply message by changing the text as well and send it back to the sender as a response. The guide does not take more than five minutes to complete because it contains step by step instructions and each step demonstrated with a screenshot as well which makes it easier to complete.***

## Step 1 - Open a routing rule

The first step is to select a route that you want to modify. If you are not familiar with routing rules, here you can learn about SMS routing in SMS Gateway. SMS Gateway creates default routes, so you can simply select the default outgoing route as you can see it in Figure 1.



SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.

Connections

No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

Please install a connection!

Incoming

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defin_norbert.serban | Any_SMS_Connection | norbert.serban | Copy |
| 2. | defin_admin | Any_SMS_Connection | admin | Copy |

Add new route...

2 routes installed.

Outgoing

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move |

Add new route...

1 route installed.

Users and applications

● admin (Admin) 0.00 MPM
Ready

● norbert.serban (User) 0.00 MPM
Ready

Add new user/application...

4 users/applications installed

Figure 1 - Select a routing rule in SMS Gateway

The next menu, that you will be able to see is the details menu of the selected routing rule. Here you can configure it and specify how it should handle your outgoing messages. To swap the sender and recipient phone number, first you need to select the 'Modify' tab, like in Figure 2.

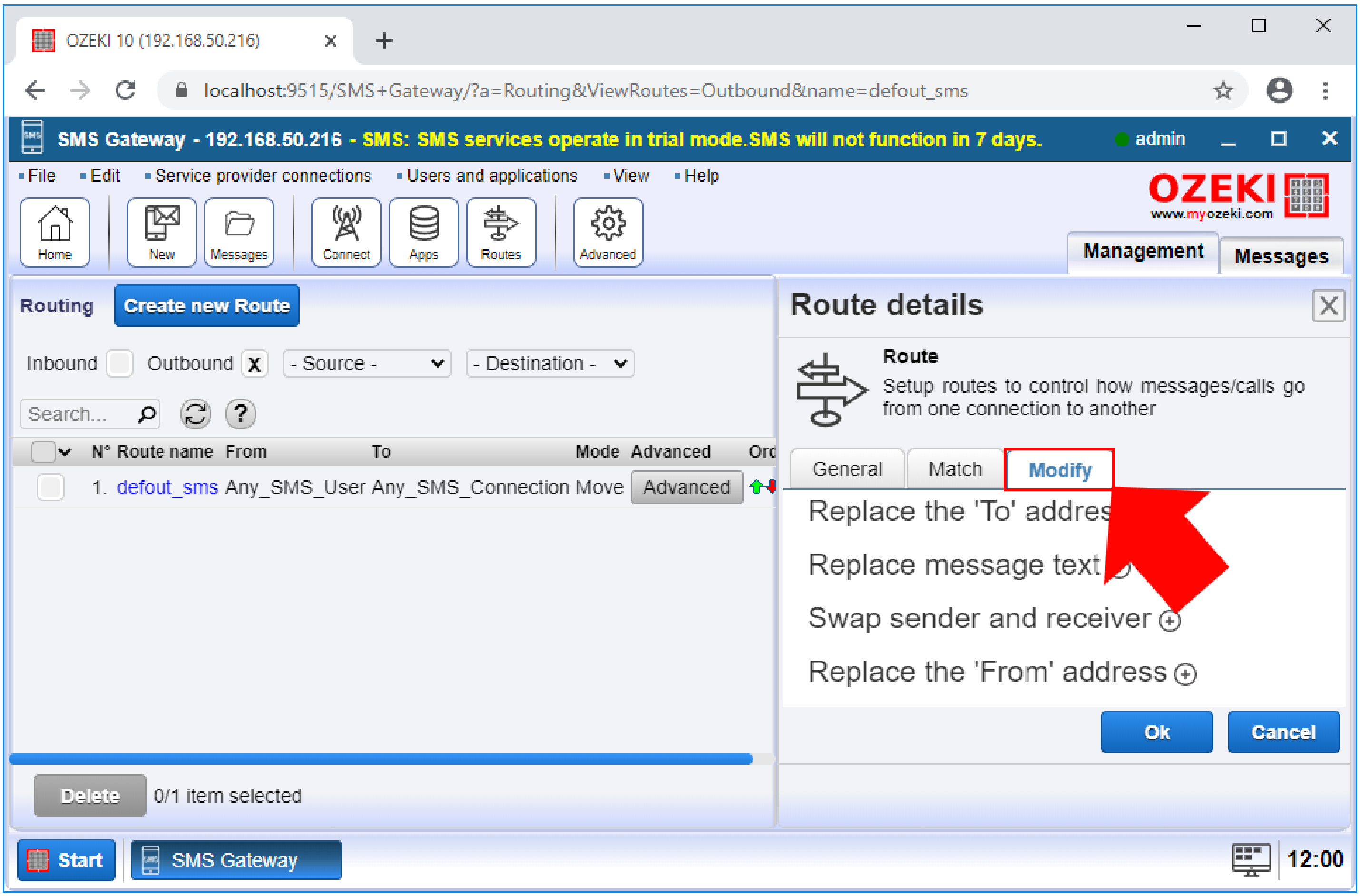
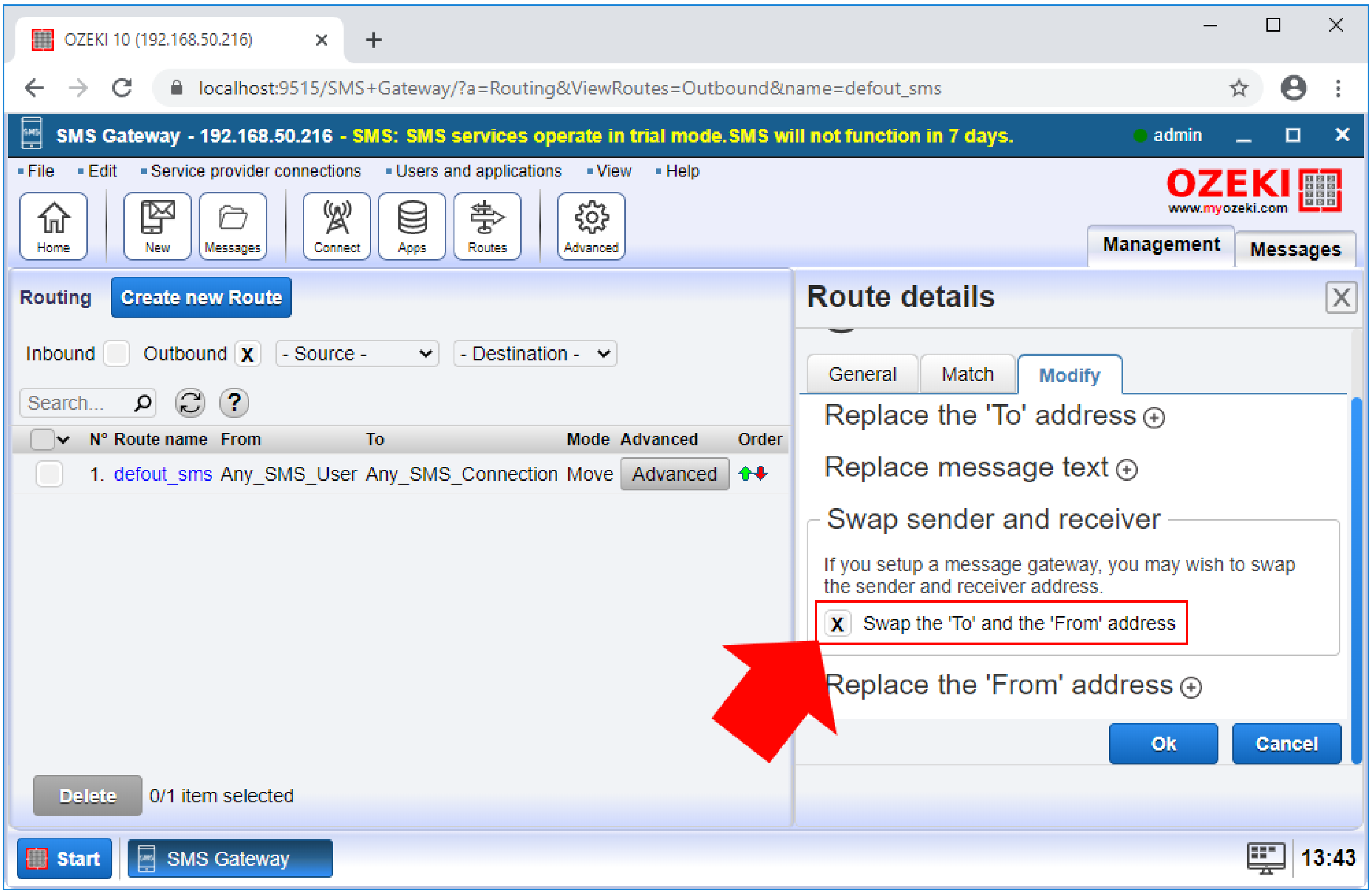

OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=Routing&ViewRoutes=Outbound&name=defout_sms
SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
OZEKI
www.myozeki.com
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Routing
Create new Route
Inbound
Outbound X
- Source -
- Destination -
Search...
N° Route name From To Mode Advanced Ord
1. defout_sms Any_SMS_User Any_SMS_Connection Move Advanced
Delete
0/1 item selected
Route details
Route
Setup routes to control how messages/calls go from one connection to another
General
Match
Modify
Replace the 'To' addres
Replace message text
Swap sender and receiver
Replace the 'From' address
Ok
Cancel
Start
SMS Gateway
12:00

Figure 2 - Select the Modify menu

## Step 2 - Swap the sender and receiver addresses

In the Modify menu, you need to select the 'Swap sender and receiver' option, where you can set up the swapping of the sender and recipient address. This operation can be done easily by just ticking the 'Swap the 'To' and the 'From' address' option like in Figure 3. Then, you can modify or replace the original message. If you are not familiar with these operations, you can learn more about how to change the message text or how to add text to the message. To save all modifications, just click OK.



| | N° | Route name | From | To | Mode | Advanced | Order |
|---|---|---|---|---|---|---|---|
| □ | 1. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move | Advanced | |

Swap sender and receiver

If you setup a message gateway, you may wish to swap the sender and receiver address.

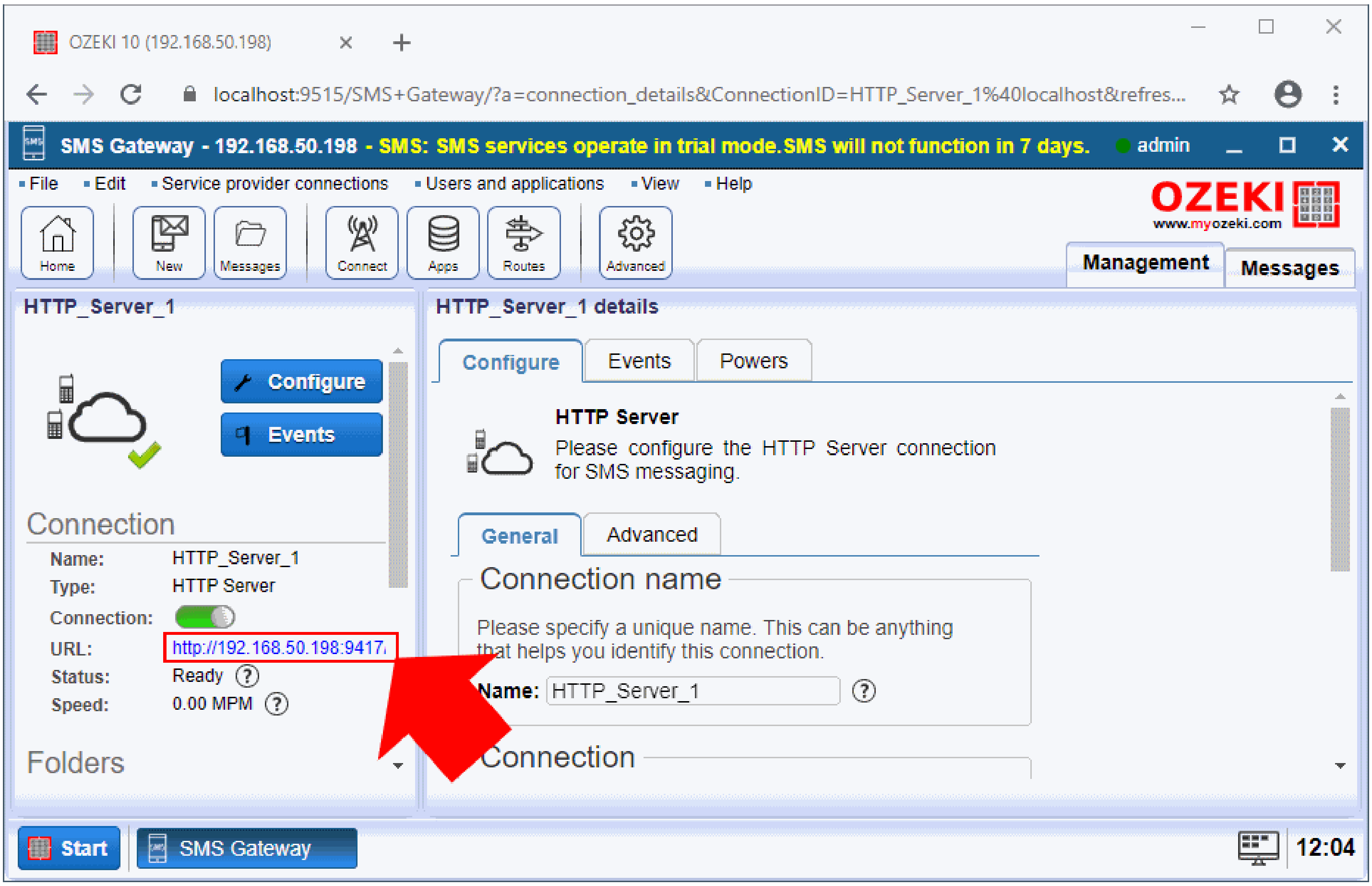
☒ Swap the 'To' and the 'From' address

Replace the 'From' address

Figure 3 - Configure the swap of sender and receiver

# Step 3 - Send a test message

Next, you can test the solution. For that, you need to create two connections to see if the response message arrived without any problem. These connections can be HTTP server connections. If you don't know about how to create the HTTP Server connection, you can check it at this point. After you created the two HTTP server connections, open the first connection, and open the HTML form like in Figure 4.


OZEKI 10 (192.168.50.198)
localhost:9515/SMS+Gateway/?a=connection_details&ConnectionID=HTTP_Server_1%40localhost&refres...
SMS Gateway - 192.168.50.198 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
OZEKI
www.myozeki.com
Management
Messages
Configure
Events
Powers
General
Advanced
Start
SMS Gateway
12:04

HTTP_Server_1

Connection

Name: HTTP_Server_1
Type: HTTP Server
Connection:
URL: http://192.168.50.198:9417/
Status: Ready
Speed: 0.00 MPM

Folders

HTTP_Server_1 details

HTTP Server
Please configure the HTTP Server connection for SMS messaging.

Connection name

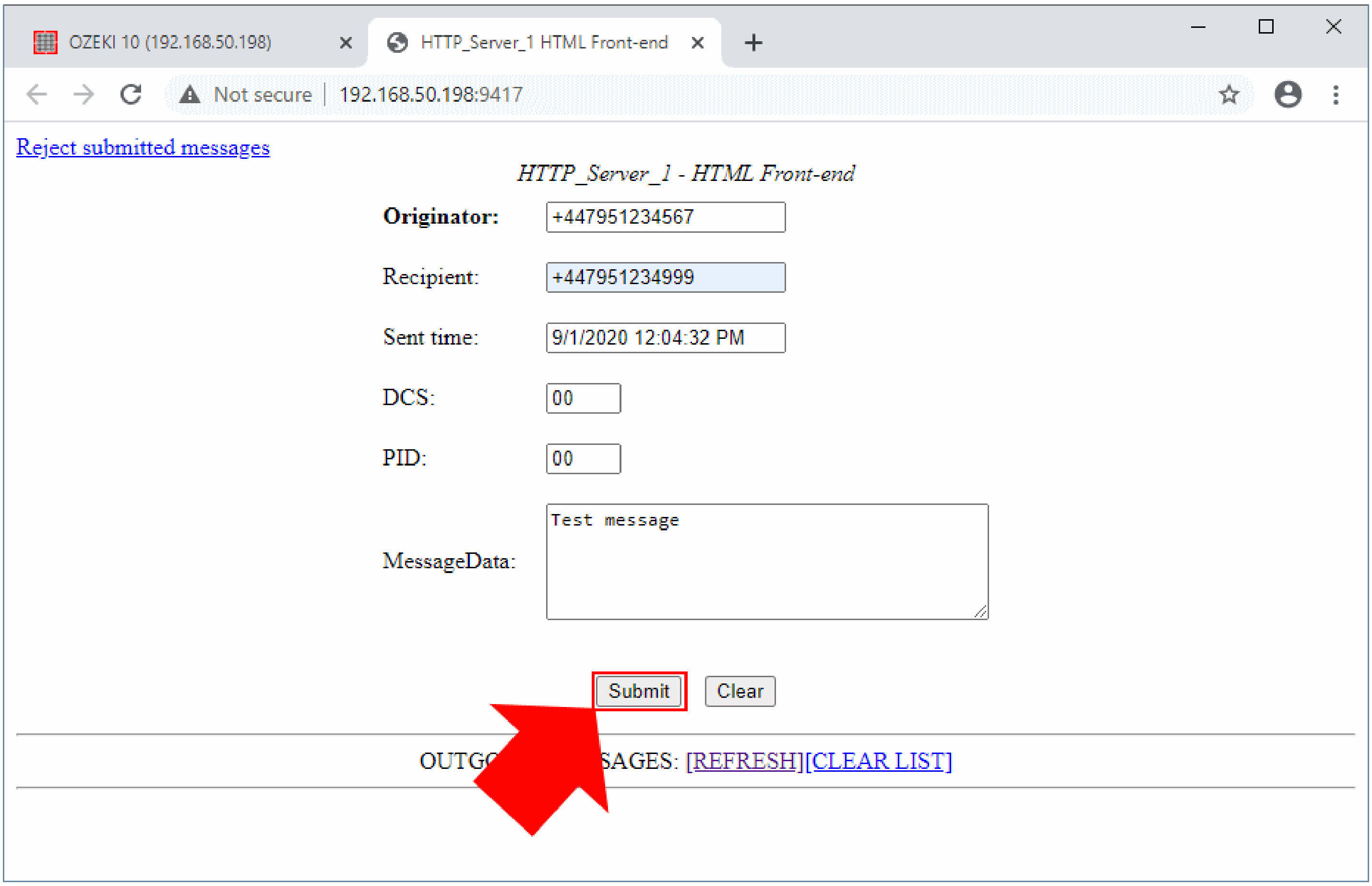
Please specify a unique name. This can be anything that helps you identify this connection.

Name: HTTP_Server_1

Connection

Figure 4 - Open the HTML form of the connection

The HTML form allows you to send test messages to other connections or phone numbers. Here, type the recipient phone number that you specified for the second HTTP server connection as Figure 5 demonstrates it. Then, just write a simple test message in the textbox. If you finished, just click on 'Submit' to send the message.


OZEKI 10 (192.168.50.198)
HTTP_Server_1 HTML Front-end
Not secure
192.168.50.198:9417
Reject submitted messages
Submit
Clear

*HTTP_Server_1 - HTML Front-end*

**Originator:** +447951234567

Recipient: +447951234999

Sent time: 9/1/2020 12:04:32 PM

DCS: 00

PID: 00

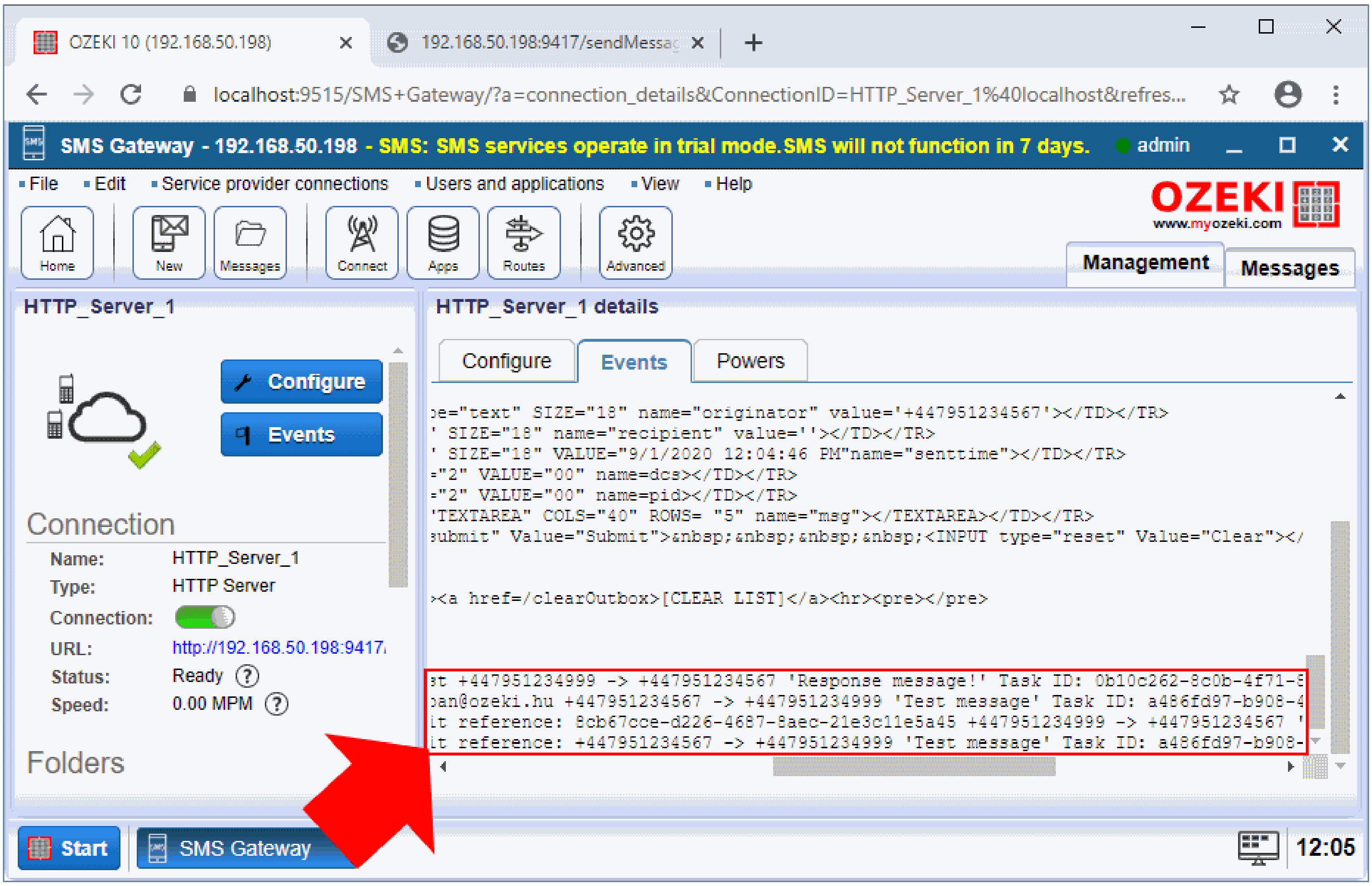
MessageData: Test message

OUTGO... SAGES: [REFRESH][CLEAR LIST]

Figure 5 - Write and send the test message

# Step 4 - View the results in the event log

The last thing that you need to perform is to check if both the original and the response message arrived successfully to the sender and the recipient as well. For that, you can check the Event menu of the HTTP server connection. Here, as you can see it in Figure 6, the connection lists every event that occurred during its uptime. Here you can see the two lines that logged, first the delivery of the original message to the recipient, then the delivery of the response as well.



OZEKI 10 (192.168.50.198)
192.168.50.198:9417/sendMessag
localhost:9515/SMS+Gateway/?a=connection_details&ConnectionID=HTTP_Server_1%40localhost&refres...
SMS Gateway - 192.168.50.198 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
OZEKI
www.myozeki.com
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
HTTP_Server_1
Configure
Events
HTTP_Server_1 details
Configure
Events
Powers
Folders
Start
SMS Gateway
12:05

Connection

Name: HTTP_Server_1
Type: HTTP Server
Connection:
URL: http://192.168.50.198:9417/
Status: Ready
Speed: 0.00 MPM

```
pe="text" SIZE="18" name="originator" value='+447951234567'></TD></TR>
' SIZE="18" name="recipient" value=''></TD></TR>
' SIZE="18" VALUE="9/1/2020 12:04:46 PM"name="senttime"></TD></TR>
="2" VALUE="00" name=dcs></TD></TR>
="2" VALUE="00" name=pid></TD></TR>
'TEXTAREA" COLS="40" ROWS= "5" name="msg"></TEXTAREA></TD></TR>
submit" Value="Submit">&nbsp;&nbsp;&nbsp;&nbsp;<INPUT type="reset" Value="Clear"></


><a href=/clearOutbox>[CLEAR LIST]</a><hr><pre></pre>



st +447951234999 -> +447951234567 'Response message!' Task ID: 0b10c262-8c0b-4f71-8
pan@ozeki.hu +447951234567 -> +447951234999 'Test message' Task ID: a486fd97-b908-4
it reference: 8cb67cce-d226-4687-8aec-21e3c11e5a45 +447951234999 -> +447951234567 '
it reference: +447951234567 -> +447951234999 'Test message' Task ID: a486fd97-b908-
```

Figure 6 - Check the results in the event log

# SMS Developers Guide

***The Ozeki SMS Gateway is a developer friendly software platform. You can use it to integrate and quickly deploy SMS messaging solutions. It can operate in your server computer, or you can install it on your customers' or suppliers' system. Save time and money and use the most popular SMS gateway software in the world.***

This page is designed to help software developers, IT and Business leaders and solution designers to better understand the SMS technology and SMS developer tools offered by Ozeki. Add communication capabilities to your products with Ozeki ready-to-integrate SMS APIs to send and receive SMS messages. In addition to sending and receiving, it also makes it possible to handle delivery reports and manage delivery times.

Send out hundreds of thousands of text messages automatically with ease. SMS automation enables you to plan and execute surveys, notifications and campaigns or simply send alerts and reminders. You can also integrate SMS messaging into your workflow using these SMS APIs.

The OZEKI SMS gateway will give you powerful developer tools needed to build great applications. You can connect your systems to Ozeki SMS using HTTP, or a database or using our C#.NET, Visual Basic.NET examples.

### SMS from/to Database

This is a helpful guide, so you can easily connect Ozeki SMS Gateway and a selected database server. Follow the steps in this guide to learn how to send or receive SMS messages from a database application. A widely used method can help you to achieve this by inserting or reading data from the tables. You can also find easy configuration steps in this tutorial.

Learn More

### HTTP SMS API

Ozeki introduced a new HTTP SMS API in Ozeki SMS Gateway to address the requirements of the SMS software developer community. This new HTTP SMS API provides more features, flexibility and simplicity compared to previous versions. It was designed to help developers in productivity and to be able to add SMS functionality to any project.

Learn More

### PHP SMS API

PHP developers can use PHS SMS API to add SMS functionality to their website. In a few lines of code, your PHP application can send, receive, and reply to text messages with Ozeki SMS API. This tutorial shows you how to use Ozeki SMS Gateway's PHP SMS API. It will take you less then an hour to integrate the solution that can send and receive messages from a MySQL database.

Learn More

### C# SMS API

Your C# application can connect to a Standard User or any other user created in Ozeki SMS Gateway. You can easily prepare your C# codes to send or receive SMS messages through SMS Gateway. Below you can download source codes for two C#.NET SMS demo projects which demonstrates how easy it is to connect C# with Ozeki SMS Gateway.

Learn More

### VB.NET SMS API

See how to send and receive SMS messages by using your Visual Basic.NET applications. It is a great choice to communicate with your clients or employees with simple text messages. Choose which is the best method for your system by downloading a database and a HTTP API example.

Learn More

# How to setup an SQL to SMS gateway

***The following guide provides you information about how you can send and receive messages from an SQL database server with the help of Ozeki 10 SMS Gateway. This document divided into 3 main segments: First the supported database servers are listed. Second, the procedure of sending SMS from a database server is demonstrated using MySQL. Third, an autoresponding SMS service is created with the help of a database trigger. To complete this guide successfully, you need to set up a database server as a prerequisite and you must have some basic understanding of SQL. The completion of the steps in this guide takes about 10 to 15 minutes. So, let's start!***

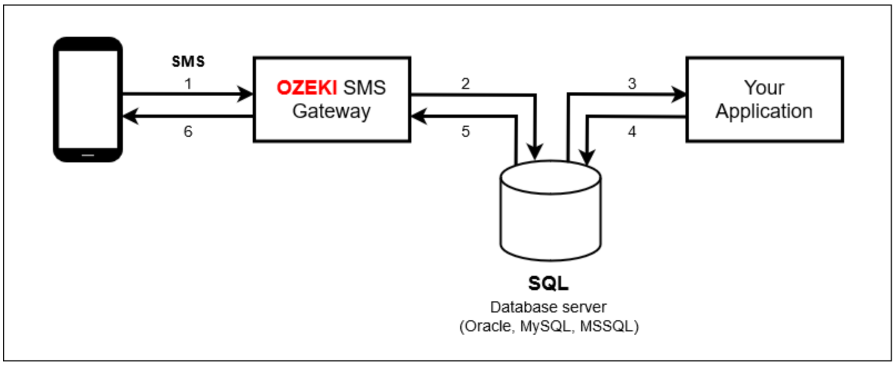


Figure 1 - SMS messaging using a database server

## Database servers supported by Ozeki 10 SMS Gateway

**Microsoft SQL Express**

You can use an MSSQL Express database for SMS messaging. Follow these pages to send or receive SMS messages through Ozeki SMS Gateway's Database User. All you have to do is insert or read data rows from the appropriate SQL table by setting INSERT INTO and SELECT statements.

See how to send SMS from an MSSQL Express database

**Oracle**

The connection between Oracle Database and Ozeki SMS Gateway is a perfect solution for SMS messaging. Ozeki SMS Gateway's Database User makes sure that incoming SMS messages can be inserted while outgoing SMS messages can be selected from database tables.

See how to send SMS from an Oracle SQL database

**MySQL**

You can combine Ozeki SMS Gateway with a MySQL database to send, receive and store SMS messages. On these pages you can see how to install and configure the connection with the proper connection string and how to create the recommended database structure.

See how to send SMS from a MySQL database

**PostgreSQL**

This chapter gives you great opportunity to see how to send and receive SMS messages through pre-created PostgreSQL database tables. You just need to connect to them with a Database User of Ozeki SMS Gateway. Do not forget to provide the ODBC driver connection string for the user.

See how to send SMS from a PostgreSQL database

**SAP SQL Anywhere**

See how to send and receive SMS messages through an SAP SQL Anywhere server with the Database User of Ozeki SMS Gateway. Here you can find a connection string and short CREATE TABLE statements to get started. You can send messages by inserting new message records.

See how to send SMS from a SAP SQL Anywhere database

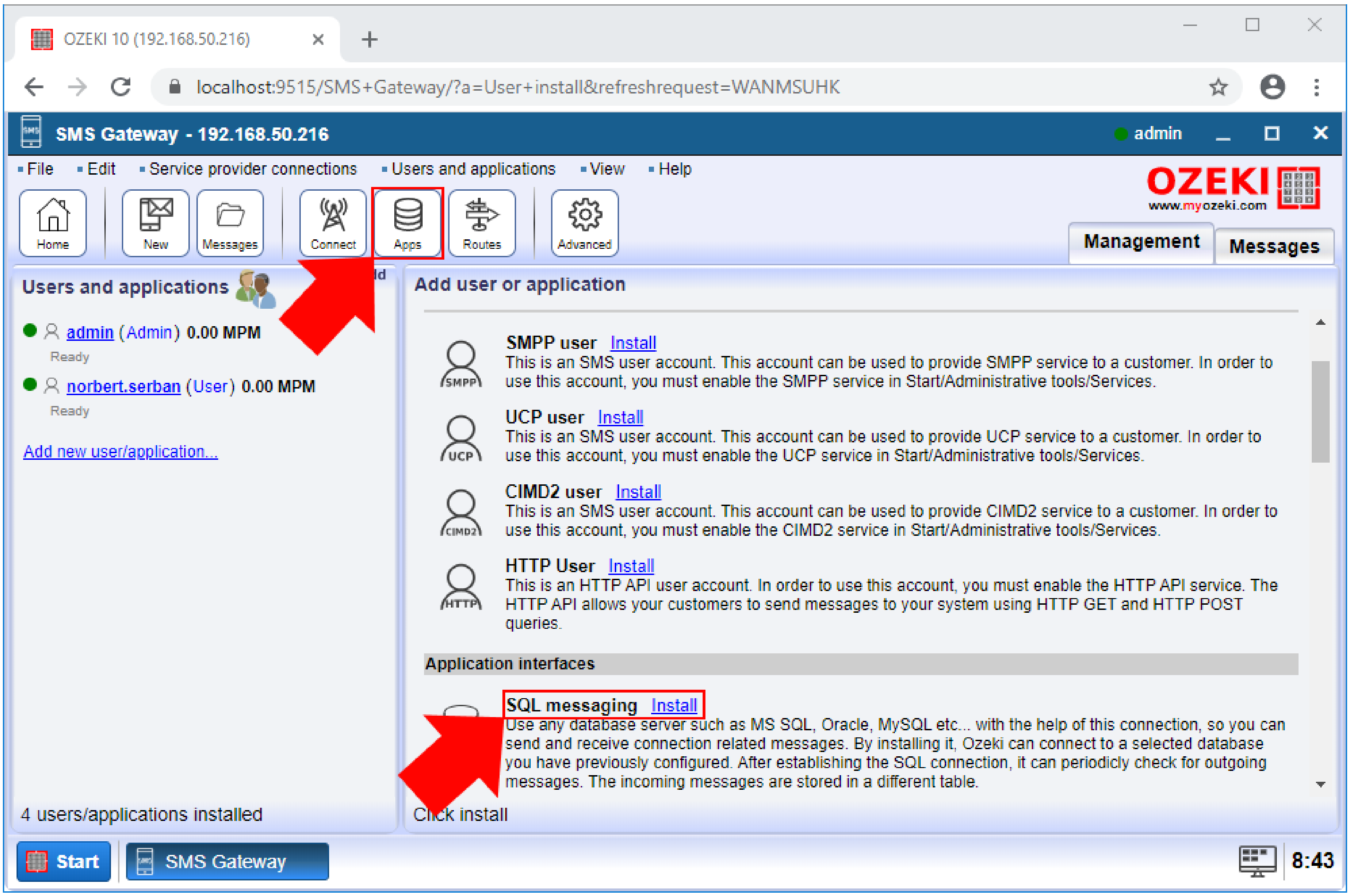
**Microsoft Access**

You can simply start to send and receive SMS messages through Ozeki SMS Gateway's Database User by using Microsoft Access database tables. All you have to do is insert or read data rows from the appropriate SQL table by configuring INSERT INTO and SELECT statements for the user.

See how to send SMS from a Microsoft Access database

## Step 1 - Setup a database connection

The first thing you need to do is is to setup a database connection in Ozeki SMS Gateway. To do this, first, open the SMS Gateway, and select the **Apps** icon from the toolbar. Next, from the list of applications (Figure 2), you have to select **SQL messaging** by clicking on the Install button.


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=User+install&refreshrequest=WANMSUHK
SMS Gateway - 192.168.50.216
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Users and applications
admin (Admin) 0.00 MPM
Ready
norbert.serban (User) 0.00 MPM
Ready
Add new user/application...
4 users/applications installed
Add user or application
Start
SMS Gateway
8:43

**SMPP user** Install
This is an SMS user account. This account can be used to provide SMPP service to a customer. In order to use this account, you must enable the SMPP service in Start/Administrative tools/Services.

**UCP user** Install
This is an SMS user account. This account can be used to provide UCP service to a customer. In order to use this account, you must enable the UCP service in Start/Administrative tools/Services.

**CIMD2 user** Install
This is an SMS user account. This account can be used to provide CIMD2 service to a customer. In order to use this account, you must enable the CIMD2 service in Start/Administrative tools/Services.

**HTTP User** Install
This is an HTTP API user account. In order to use this account, you must enable the HTTP API service. The HTTP API allows your customers to send messages to your system using HTTP GET and HTTP POST queries.

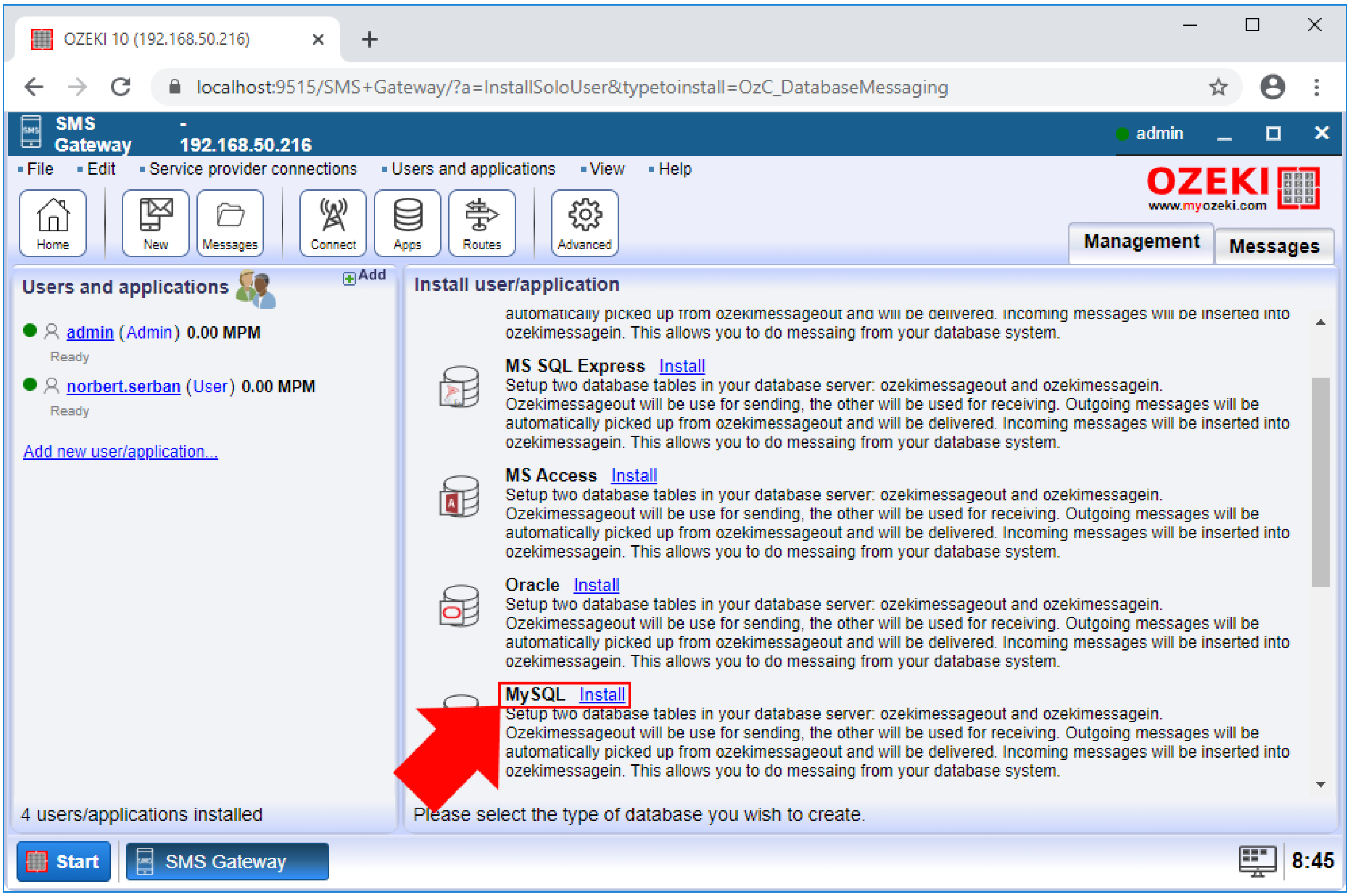
**Application interfaces**

**SQL messaging** Install
Use any database server such as MS SQL, Oracle, MySQL etc... with the help of this connection, so you can send and receive connection related messages. By installing it, Ozeki can connect to a selected database you have previously configured. After establishing the SQL connection, it can periodicly check for outgoing messages. The incoming messages are stored in a different table.

Click install

Figure 2 - Create a SQL messaging application

After you selected SQL messaging, you can see all available options that you can create in SMS Gateway. By following this guide, you will be able to see how you can create an SQL connection with a MySQL database, so to follow that guide, just click on the Install button of the MySQL option as you can see it in Figure 3.



OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=InstallSoloUser&typetoinstall=OzC_DatabaseMessaging
SMS Gateway - 192.168.50.216
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
OZEKI
www.myozeki.com
Management
Messages
Users and applications
Add
admin (Admin) 0.00 MPM
Ready
norbert.serban (User) 0.00 MPM
Ready
Add new user/application...
4 users/applications installed
Install user/application
Start
SMS Gateway
8:45

automatically picked up from ozekimessageout and will be delivered. Incoming messages will be inserted into ozekimessagein. This allows you to do messaging from your database system.

MS SQL Express Install
Setup two database tables in your database server: ozekimessageout and ozekimessagein. Ozekimessageout will be use for sending, the other will be used for receiving. Outgoing messages will be automatically picked up from ozekimessageout and will be delivered. Incoming messages will be inserted into ozekimessagein. This allows you to do messaging from your database system.

MS Access Install
Setup two database tables in your database server: ozekimessageout and ozekimessagein. Ozekimessageout will be use for sending, the other will be used for receiving. Outgoing messages will be automatically picked up from ozekimessageout and will be delivered. Incoming messages will be inserted into ozekimessagein. This allows you to do messaging from your database system.

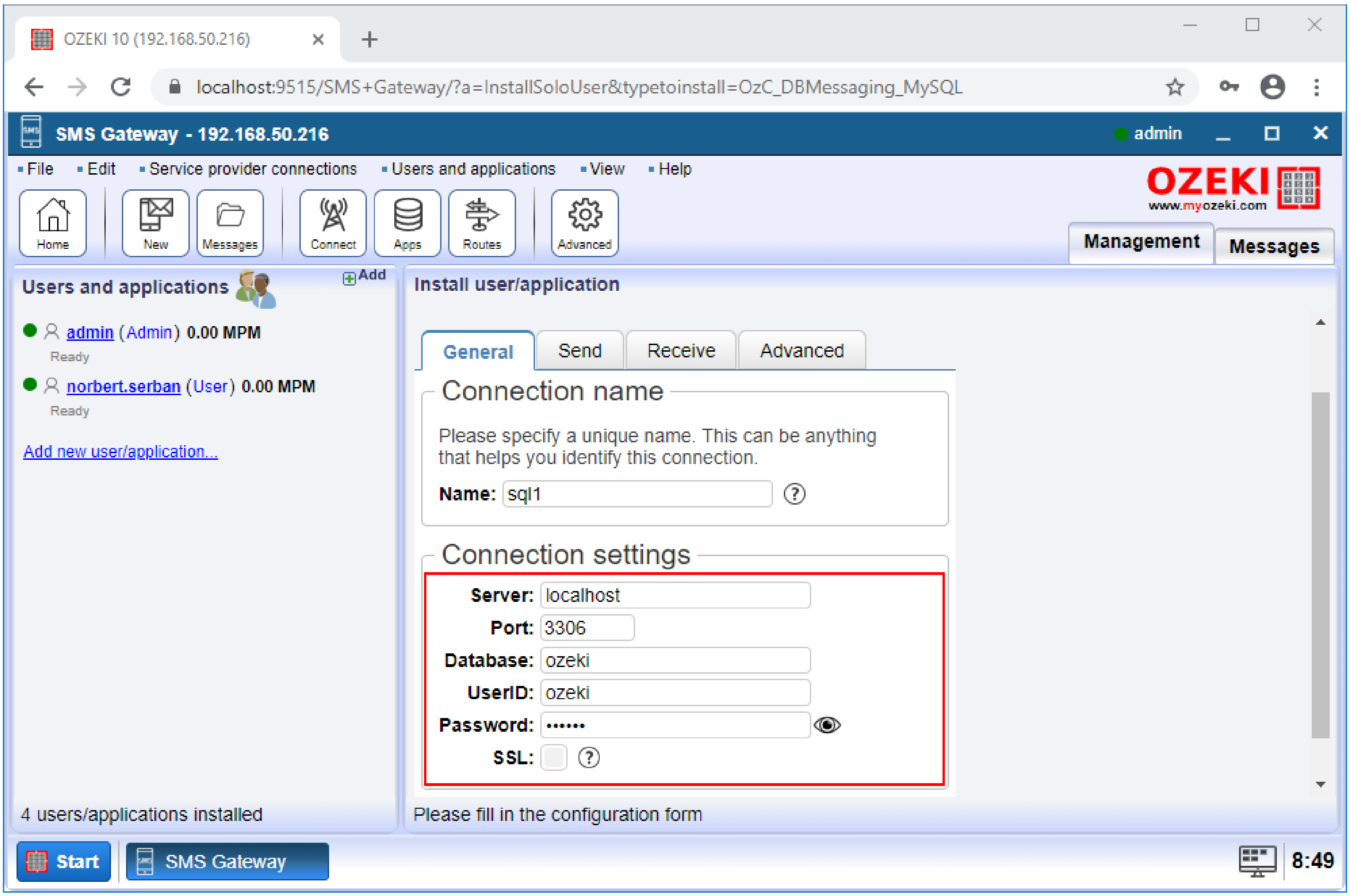
Oracle Install
Setup two database tables in your database server: ozekimessageout and ozekimessagein. Ozekimessageout will be use for sending, the other will be used for receiving. Outgoing messages will be automatically picked up from ozekimessageout and will be delivered. Incoming messages will be inserted into ozekimessagein. This allows you to do messaging from your database system.

MySQL Install
Setup two database tables in your database server: ozekimessageout and ozekimessagein. Ozekimessageout will be use for sending, the other will be used for receiving. Outgoing messages will be automatically picked up from ozekimessageout and will be delivered. Incoming messages will be inserted into ozekimessagein. This allows you to do messaging from your database system.

Please select the type of database you wish to create.

Figure 3 - Install MySQL database connection

The last step of the creation of the database application is to connect it to your database server by filling the fields of the Connection Settings (Figure 4). Here you have to give all details about the database that you want to connect to. The first two fields are about to identify the database server. If you left them to default when setting up the database server, you can leave the values here default as well. Then you have to type the name of the database that you want to connect to and your user ID with a password that you use within the database server. If you filled all fields, you can just click OK to create the database application.



OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=InstallSoloUser&typetoinstall=OzC_DBMessaging_MySQL
SMS Gateway - 192.168.50.216
admin
File
Edit
Service provider connections
Users and applications
View
Help
OZEKI
www.myozeki.com
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Users and applications
Add
admin (Admin) 0.00 MPM
Ready
norbert.serban (User) 0.00 MPM
Ready
Add new user/application...
4 users/applications installed
Start
SMS Gateway
8:49

Install user/application

General | Send | Receive | Advanced

Connection name

Please specify a unique name. This can be anything that helps you identify this connection.

Name: sql1

Connection settings

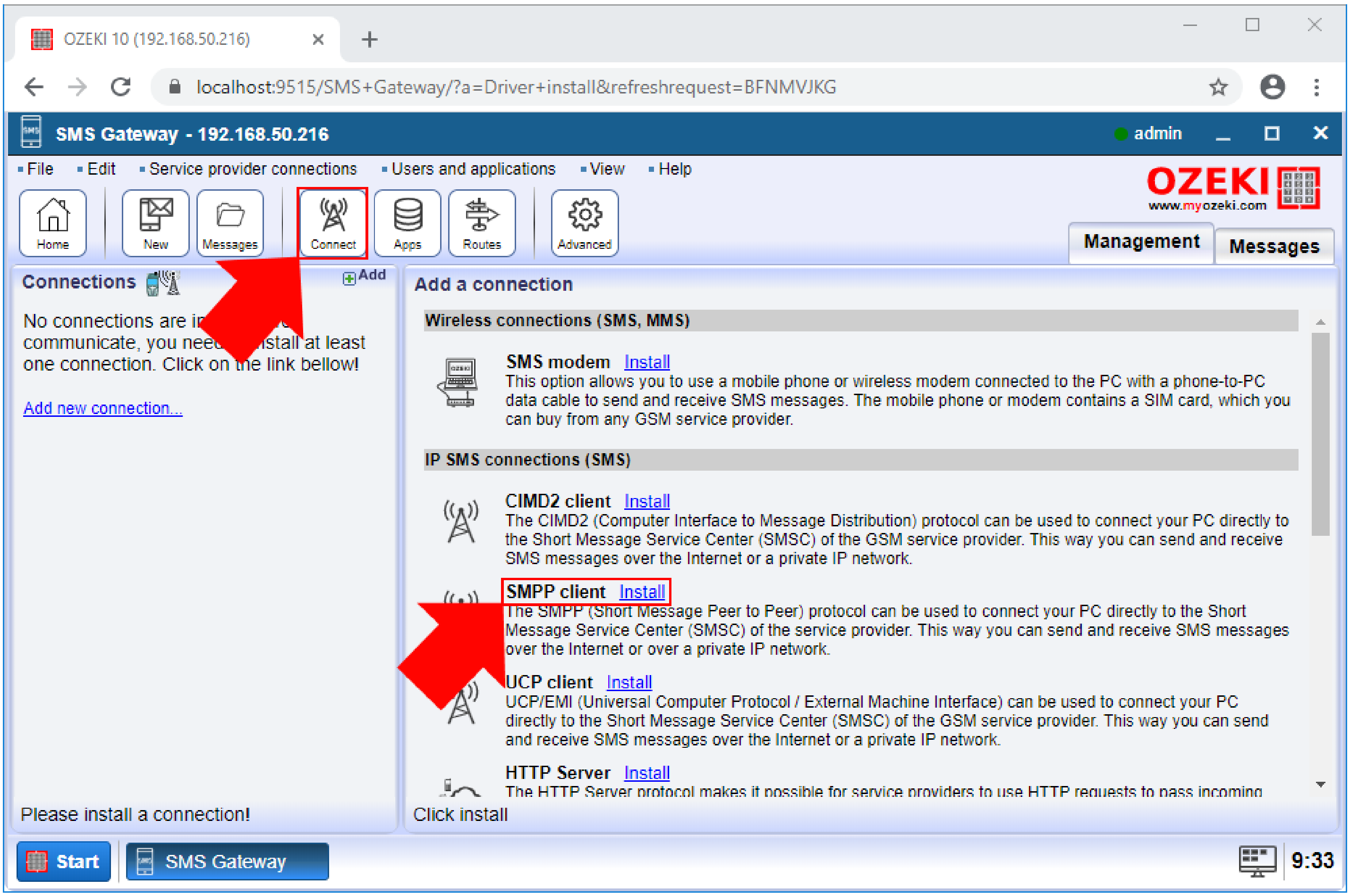
Server: localhost
Port: 3306
Database: ozeki
UserID: ozeki
Password: ••••••
SSL: □

Please fill in the configuration form

Figure 4 - Configure the database application

## Step 2 - Setup an SMPP connection

At this point, you have got a database application, but you also need a service provider connection to handle the SMS messages. In this guide, you will be able to create an SMPP client connection. For that, you just have to click on **Connect** on the toolbar, and select **SMPP client** (Figure 5) from the list of all available service provider connections.



No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

Please install a connection!

**Wireless connections (SMS, MMS)**

**SMS modem** Install
This option allows you to use a mobile phone or wireless modem connected to the PC with a phone-to-PC data cable to send and receive SMS messages. The mobile phone or modem contains a SIM card, which you can buy from any GSM service provider.

**IP SMS connections (SMS)**

**CIMD2 client** Install
The CIMD2 (Computer Interface to Message Distribution) protocol can be used to connect your PC directly to the Short Message Service Center (SMSC) of the GSM service provider. This way you can send and receive SMS messages over the Internet or a private IP network.

**SMPP client** Install
The SMPP (Short Message Peer to Peer) protocol can be used to connect your PC directly to the Short Message Service Center (SMSC) of the service provider. This way you can send and receive SMS messages over the Internet or over a private IP network.

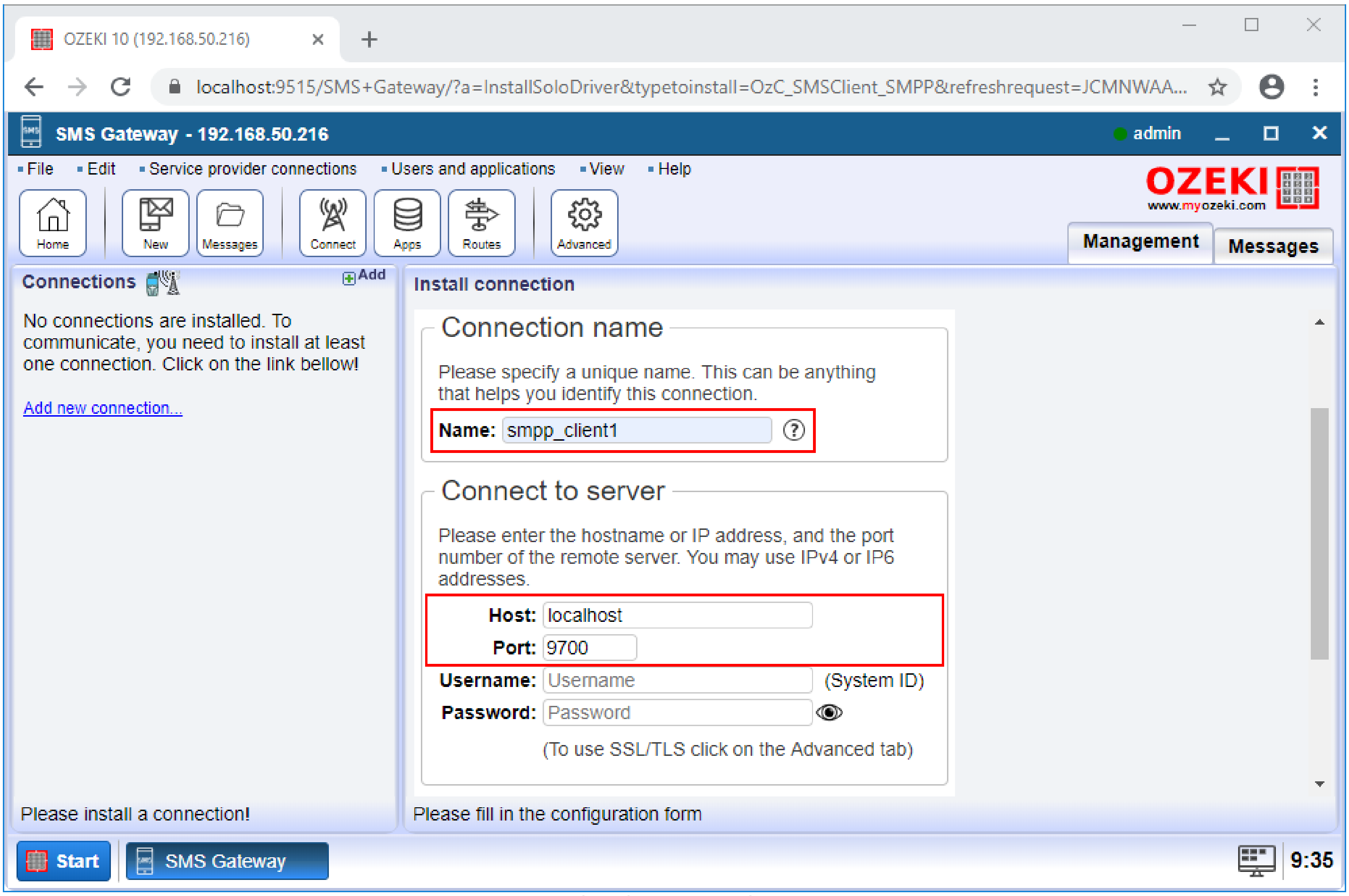
**UCP client** Install
UCP/EMI (Universal Computer Protocol / External Machine Interface) can be used to connect your PC directly to the Short Message Service Center (SMSC) of the GSM service provider. This way you can send and receive SMS messages over the Internet or a private IP network.

**HTTP Server** Install
The HTTP Server protocol makes it possible for service providers to use HTTP requests to pass incoming

Click install

Figure 5 - Install a SMPP client connection

Before establishing the connection, you need to provide some details regarding the connection. First, you have to type a name for the connection, then you have to define the server that you want to connect to as Figure 6 demonstrates it. To finish the creation of the service provider connection, just click on OK.


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=InstallSoloDriver&typetoinstall=OzC_SMSClient_SMPP&refreshrequest=JCMNWAA...
SMS Gateway - 192.168.50.216
admin
File
Edit
Service provider connections
Users and applications
View
Help
OZEKI
www.myozeki.com
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Connections
Add
Start
SMS Gateway
9:35

No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

Please install a connection!

Install connection

Connection name

Please specify a unique name. This can be anything that helps you identify this connection.

Name: smpp_client1

Connect to server

Please enter the hostname or IP address, and the port number of the remote server. You may use IPv4 or IP6 addresses.

Host: localhost

Port: 9700

Username: Username (System ID)

Password: Password

(To use SSL/TLS click on the Advanced tab)

Please fill in the configuration form

Figure 6 - Configure the SMPP client

# Step 3 - Send SMS from a database

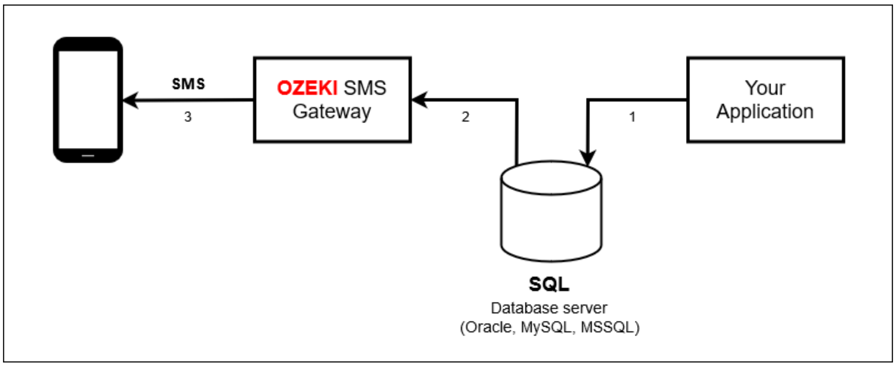
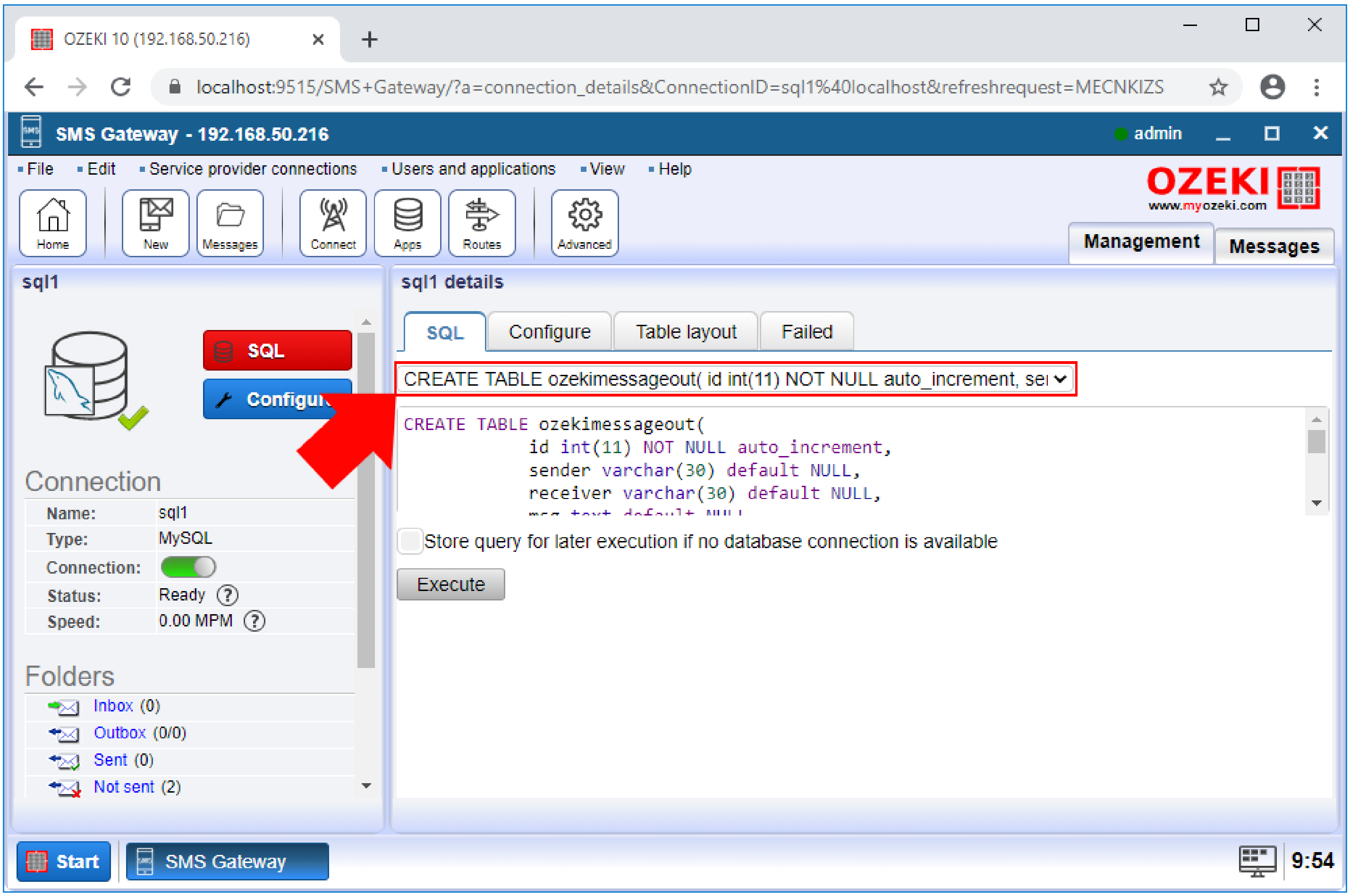


Figure 7 - Sending a SMS using the database server

With all connections set up properly in Ozeki SMS Gateway, now you can start managing SMS messages from a database. First, you will be able to see, how you can send an SMS message by using only one SQL command. By default, SMS Gateway uses **ozekimessageout** table to send messages. You do not have to worry if you have not got this table in your database, you can create this table easily. Just open your SQL application in SMS Gateway and as you can see it on Figure 8, you can select a SQL command to create the **ozekimessageout** table. So, just select it, and click on **Execute** to create the table.



OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=connection_details&ConnectionID=sql1%40localhost&refreshrequest=MECNKIZS
SMS Gateway - 192.168.50.216
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
sql1
SQL
Configure
sql1 details
SQL
Configure
Table layout
Failed
Start
SMS Gateway
9:54

Connection

| | |
|---|---|
| Name: | sql1 |
| Type: | MySQL |
| Connection: | |
| Status: | Ready |
| Speed: | 0.00 MPM |

Folders

- Inbox (0)
- Outbox (0/0)
- Sent (0)
- Not sent (2)

CREATE TABLE ozekimessageout( id int(11) NOT NULL auto_increment, se

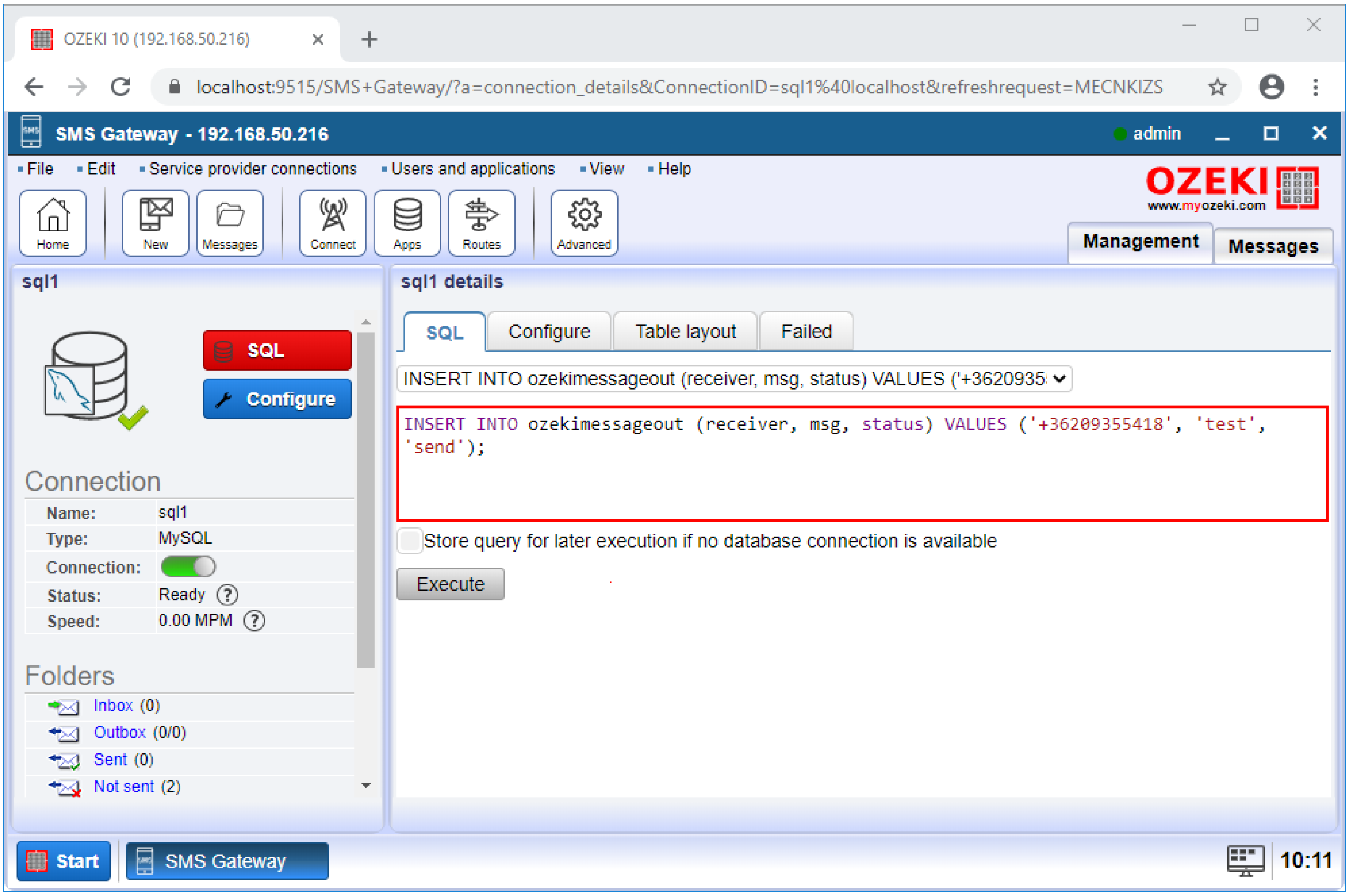
```
CREATE TABLE ozekimessageout(
            id int(11) NOT NULL auto_increment,
            sender varchar(30) default NULL,
            receiver varchar(30) default NULL,
```

□ Store query for later execution if no database connection is available

Execute

Figure 8 - Execute the SQL command to create a table

Sending a message can be done by simply just inserting the message into the **ozekimessageout** table. The SMS Gateway periodically checks this table and sends the newly added messages. To insert a message into this table, just select the corresponding INSERT command as you can see it in Figure 9. In the textbox, you can change the values which represent the receiver and the message itself. When you finished with writing the message, just click on **Execute** and your message is about to be sent.



INSERT INTO ozekimessageout (receiver, msg, status) VALUES ('+3620935

```
INSERT INTO ozekimessageout (receiver, msg, status) VALUES ('+36209355418', 'test',
'send');
```

□ Store query for later execution if no database connection is available

Connection

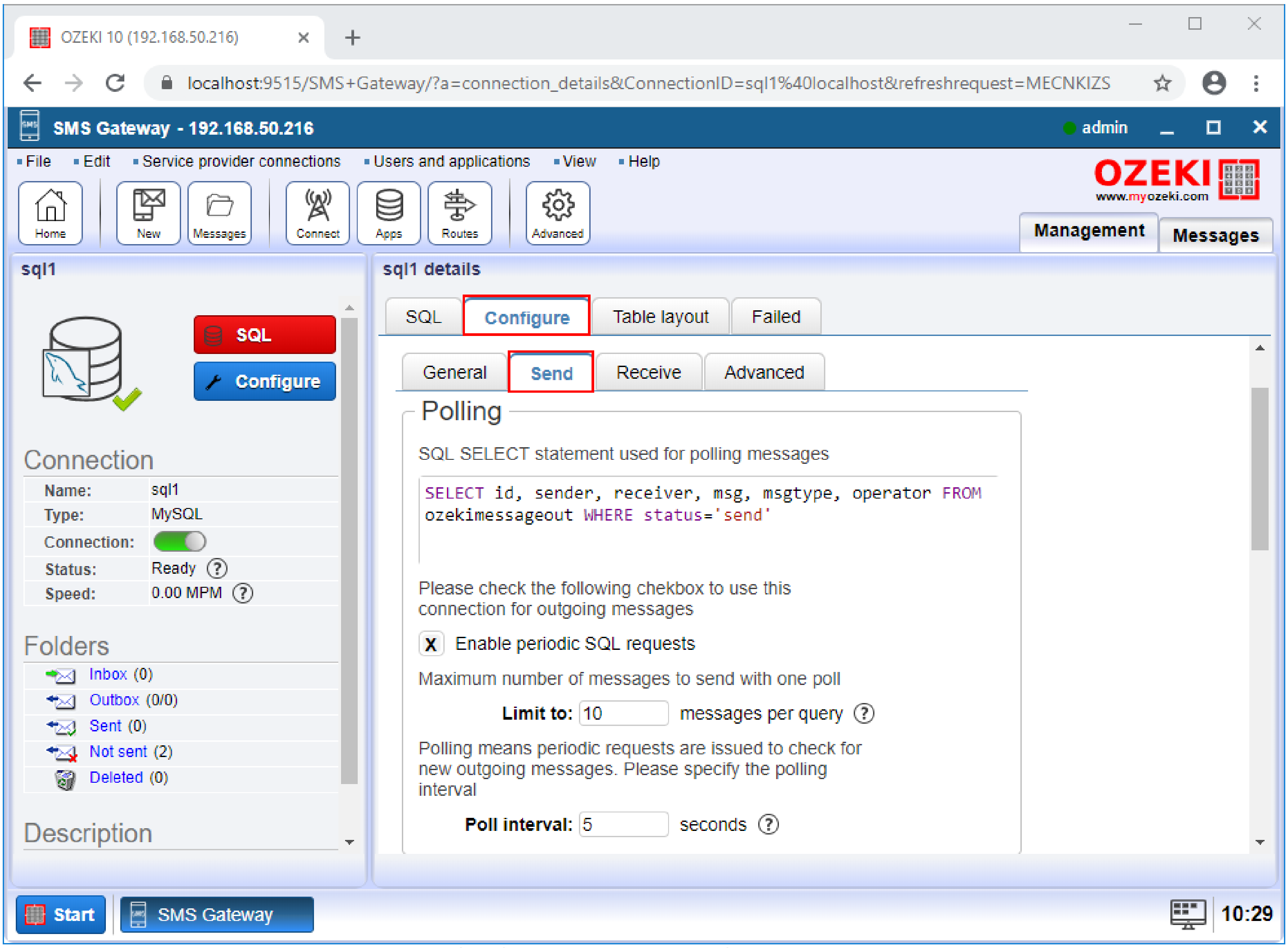
| | |
|---|---|
| Name: | sql1 |
| Type: | MySQL |
| Connection: | ● |
| Status: | Ready |
| Speed: | 0.00 MPM |

Folders

- Inbox (0)
- Outbox (0/0)
- Sent (0)
- Not sent (2)

Figure 9 - Insert the message into the database

To modify the setting of sending SMS messages, you only have to select the **Configure** tab in the menu of the database application, and here, select the **Send** tab. Here, you will be able to see the settings that define, how the outgoing messages are handled. As Figure 10 shows, here you can modify the SQL statement that queries and processes the outgoing messages. You can also set the maximum number of outgoing messages per query and the interval of polling.


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=connection_details&ConnectionID=sql1%40localhost&refreshrequest=MECNKIZS
SMS Gateway - 192.168.50.216
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
sql1
SQL
Configure
Start
SMS Gateway
10:29

sql1

Connection

Name: sql1
Type: MySQL
Connection:
Status: Ready
Speed: 0.00 MPM

Folders

Inbox (0)
Outbox (0/0)
Sent (0)
Not sent (2)
Deleted (0)

Description

sql1 details

SQL | Configure | Table layout | Failed

General | Send | Receive | Advanced

Polling

SQL SELECT statement used for polling messages

```
SELECT id, sender, receiver, msg, msgtype, operator FROM
ozekimessageout WHERE status='send'
```

Please check the following chekbox to use this connection for outgoing messages

☒ Enable periodic SQL requests

Maximum number of messages to send with one poll

**Limit to:** 10 messages per query

Polling means periodic requests are issued to check for new outgoing messages. Please specify the polling interval

**Poll interval:** 5 seconds

Figure 10 - Configure the settings of outgoing messages

# Step 4 - Receive SMS from a database

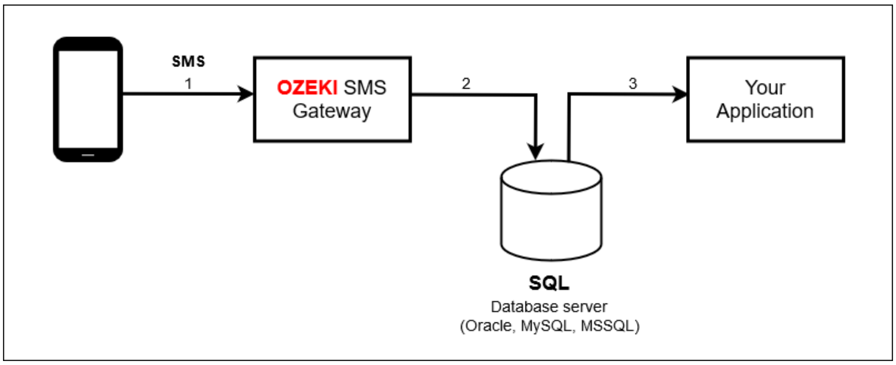
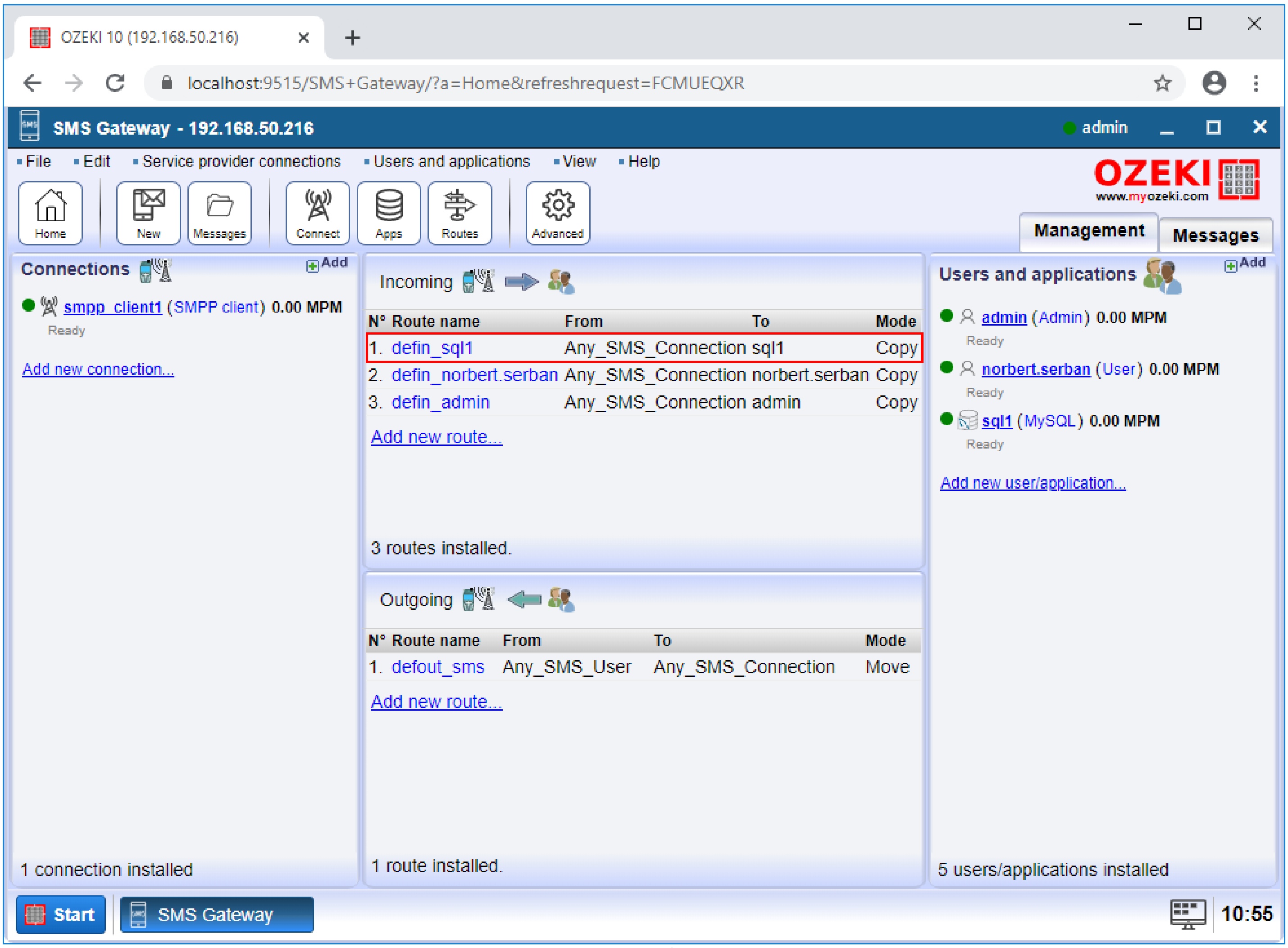


Figure 11 - Receiving messages using the database server

When you created the database application, not only the application was created, SMS Gateway created a routing rule as well, which defines that all incoming SMS messages will be copied into the database. This rule can be found in the list of routing rules as you can see it in Figure 12.



Connections

● smpp_client1 (SMPP client) 0.00 MPM
Ready

Add new connection...

1 connection installed

Incoming

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defin_sql1 | Any_SMS_Connection | sql1 | Copy |
| 2. | defin_norbert.serban | Any_SMS_Connection | norbert.serban | Copy |
| 3. | defin_admin | Any_SMS_Connection | admin | Copy |

Add new route...

3 routes installed.

Outgoing

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move |

Add new route...

1 route installed.

Users and applications

● admin (Admin) 0.00 MPM
Ready

● norbert.serban (User) 0.00 MPM
Ready

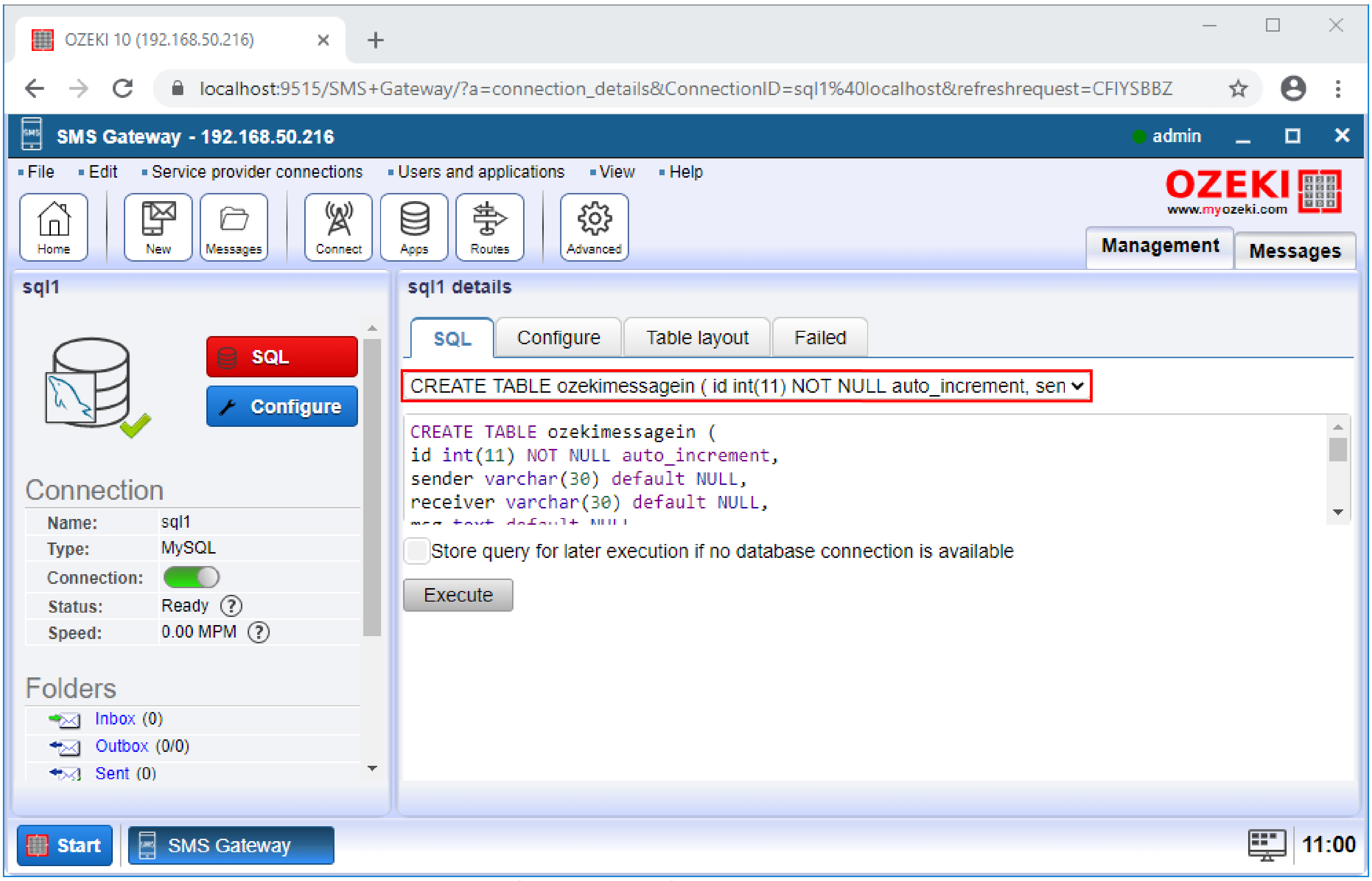
● sql1 (MySQL) 0.00 MPM
Ready

Add new user/application...

5 users/applications installed

Figure 12 - Routing rule created for the database application

The incoming messages from any SMS connection will be routed into the **ozekimessagein** table by default. You can create this table easily by opening your database application and in the SQL menu, select the SQL command from the list (Figure 13) which will create the **ozekimessagein** table in your database after executing that command.



CREATE TABLE ozekimessagein ( id int(11) NOT NULL auto_increment, sen

```
CREATE TABLE ozekimessagein (
id int(11) NOT NULL auto_increment,
sender varchar(30) default NULL,
receiver varchar(30) default NULL,
```

□ Store query for later execution if no database connection is available

Connection

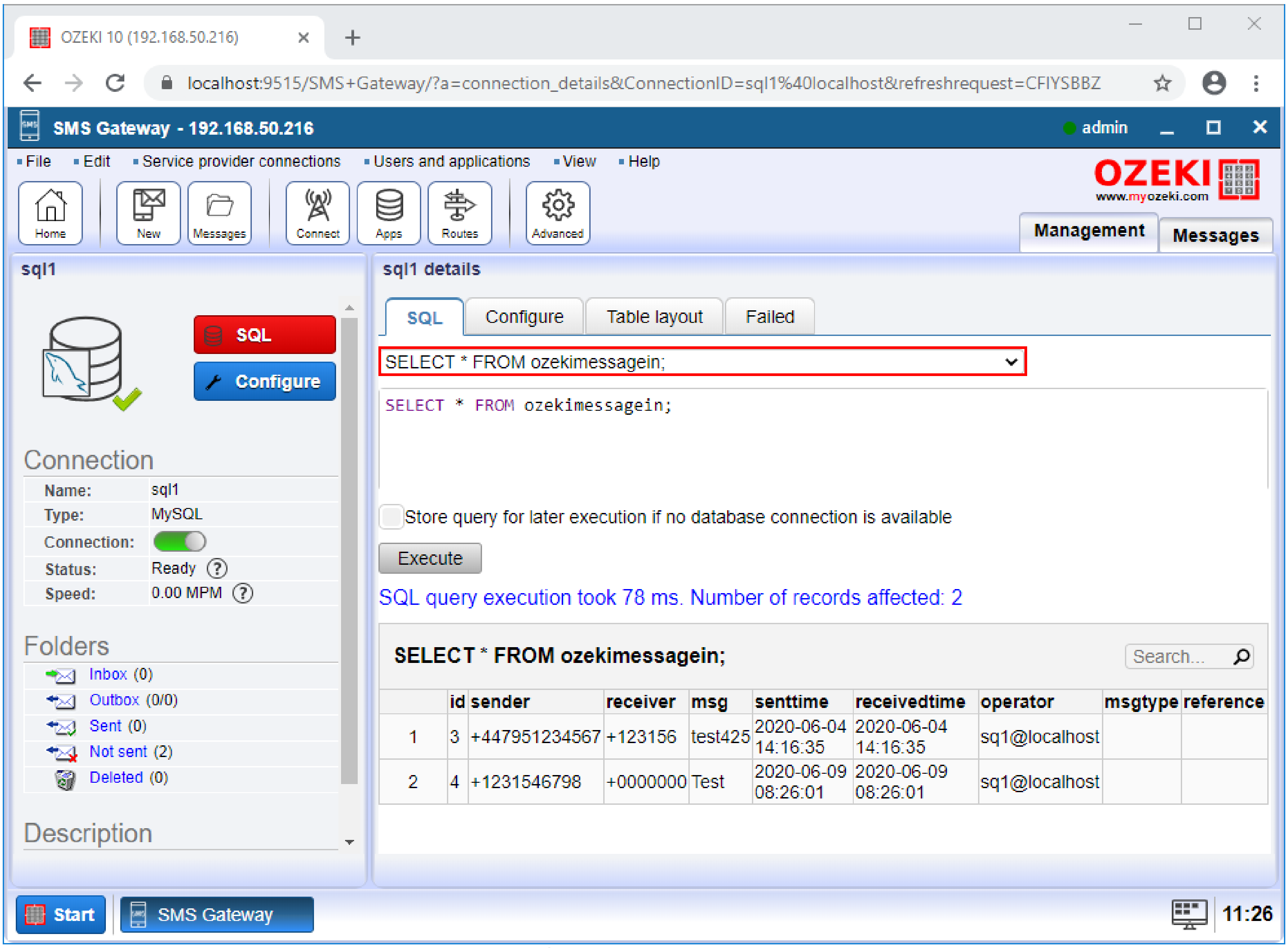
| | |
|---|---|
| Name: | sql1 |
| Type: | MySQL |
| Connection: | ● |
| Status: | Ready ? |
| Speed: | 0.00 MPM ? |

Folders

- Inbox (0)
- Outbox (0/0)
- Sent (0)

Figure 13 - Create table for the incoming messages

After you created the **ozekimessagein** table, all received messages will be inserted into that table. To check these messages, you can start a query in the database application. So, open the application, and in the SQL tab, select the command that will query the incoming messages (Figure 14) and click on **Execute**. If the execution was successful, you will be able to see the incoming messages.



| | |
|---|---|
| Name: | sql1 |
| Type: | MySQL |
| Connection: | |
| Status: | Ready |
| Speed: | 0.00 MPM |

```
SELECT * FROM ozekimessagein;
```

□ Store query for later execution if no database connection is available

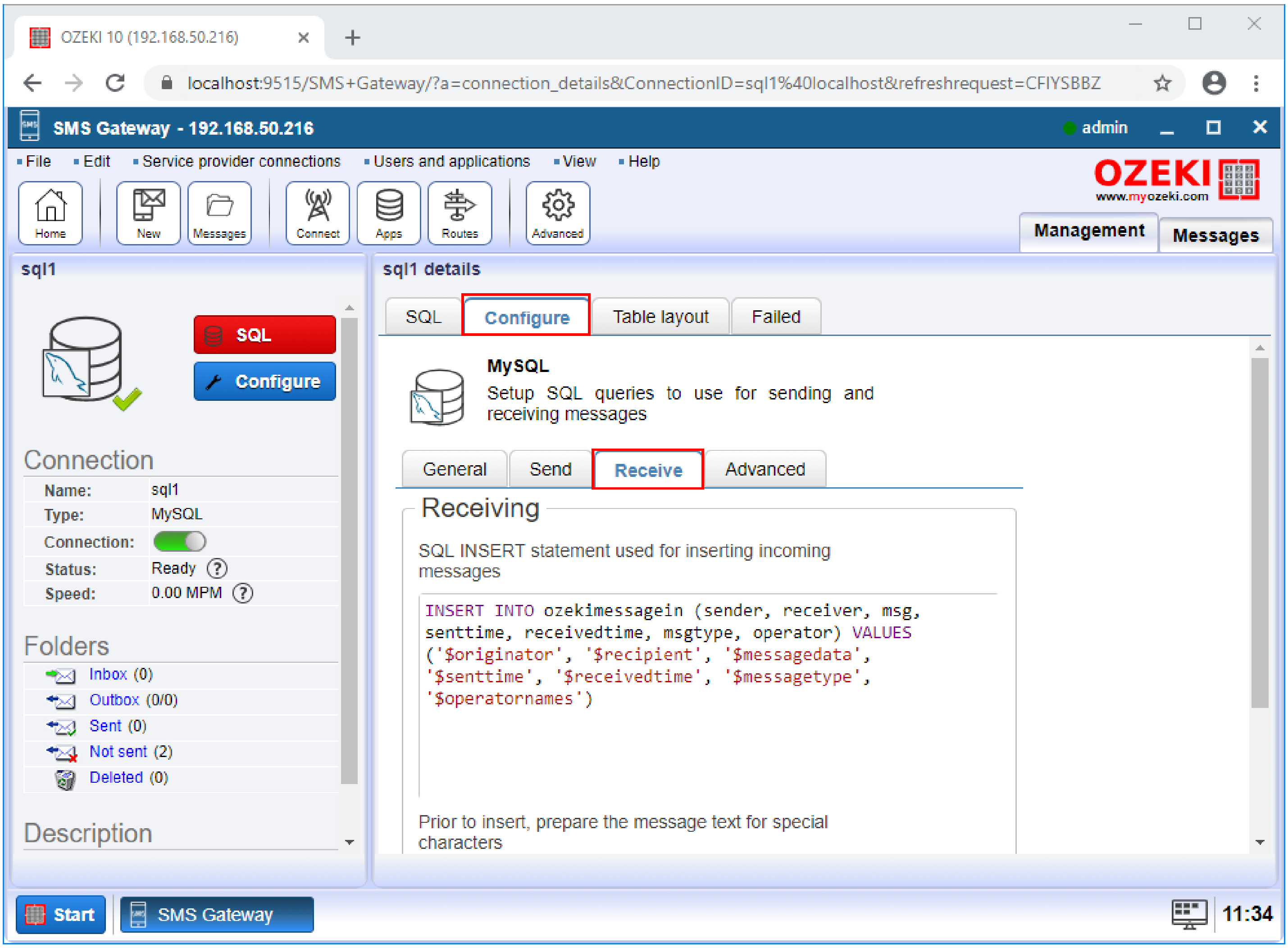
SQL query execution took 78 ms. Number of records affected: 2

**SELECT * FROM ozekimessagein;**

| | id | sender | receiver | msg | senttime | receivedtime | operator | msgtype | reference |
|---|---|---|---|---|---|---|---|---|---|
| 1 | 3 | +447951234567 | +123156 | test425 | 2020-06-04 14:16:35 | 2020-06-04 14:16:35 | sq1@localhost | | |
| 2 | 4 | +1231546798 | +0000000 | Test | 2020-06-09 08:26:01 | 2020-06-09 08:26:01 | sq1@localhost | | |

Figure 14 - Query all received messages

To modify the settings of receiving messages, you can just select **Configure** tab and here, click on **Receive** as Figure 15 demonstrates it. In this menu, you can set a SQL command, that inserts the incoming messages into a table. In the command, you can also set the values which will be inserted into the table.



Connection

Name: sql1
Type: MySQL
Connection:
Status: Ready
Speed: 0.00 MPM

Folders

Inbox (0)
Outbox (0/0)
Sent (0)
Not sent (2)
Deleted (0)

Description

**MySQL**
Setup SQL queries to use for sending and receiving messages

Receiving

SQL INSERT statement used for inserting incoming messages

```
INSERT INTO ozekimessagein (sender, receiver, msg,
senttime, receivedtime, msgtype, operator) VALUES
('$originator', '$recipient', '$messagedata',
'$senttime', '$receivedtime', '$messagetype',
'$operatornames')
```

Prior to insert, prepare the message text for special characters

Figure 15 - Modify the SQL command for the incoming messages

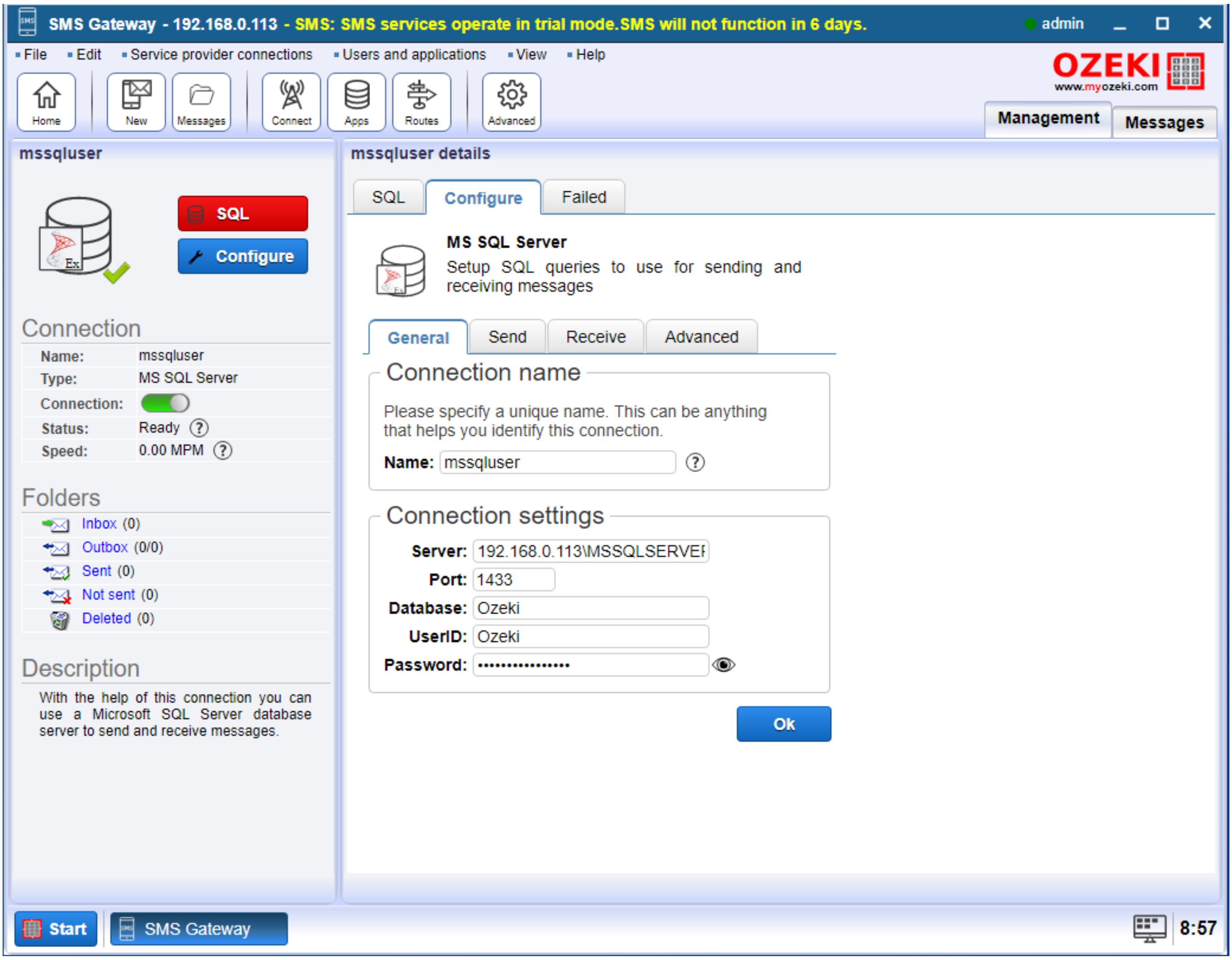
# SMS from/to MSSQL

***You can use Microsoft SQL Server or Microsoft SQL Express for sending and receiving SMS. Follow this page to send or receive SMS messages through Ozeki SMS Gateway's Database User. All you have to do is insert or read data rows from the appropriate SQL table by setting INSERT INTO and SELECT statements. The solution uses Ozeki SMS Gateway installed on your PC. This solution is divided into 4 sections:***

MSSQL Installation
Create the database tables for SMS sending and receiving
Setup the MSSQL connection in Ozeki
Send a test SMS message

Please make sure you read the general SQL to SMS howto before continuing.

To use this messaging function, please install and configure Ozeki SMS Gateway. After software configuration, please open it's management consol so you can install a database user. You will be able to send and receive SMS messages with an SQL Express database server. It is required to add the database connection type as seen in the examples on this page. **In SQL Express you should provide the following parameters:**



SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.SMS will not function in 6 days.

mssqluser

Connection

Name: mssqluser
Type: MS SQL Server
Connection:
Status: Ready
Speed: 0.00 MPM

Folders

Inbox (0)
Outbox (0/0)
Sent (0)
Not sent (0)
Deleted (0)

Description

With the help of this connection you can use a Microsoft SQL Server database server to send and receive messages.

mssqluser details

**MS SQL Server**
Setup SQL queries to use for sending and receiving messages

Connection name

Please specify a unique name. This can be anything that helps you identify this connection.

Name: mssqluser

Connection settings

Server: 192.168.0.113\MSSQLSERVEI
Port: 1433
Database: Ozeki
UserID: Ozeki
Password: ••••••••••••••••

Figure 1 - MSSQL Configuration

Congratulation for configuring a database user! Now you should create a database layout by using SQL Express. Receiving and sending text messages will work on separate SQL tables ('ozekimessagein' and 'ozekimessageout'). You can control these tables by using the INSERT and SELECT statements. To create a database layout, please open a windows command prompt and run 'cmd.exe' to start the SQL Command interpreter.

**sqlcmd**

SQL Express will start running, so you can type your statements in the command interpreter. If the command interpreter cannot connect to SQL Express, please read How to connect to SQL Express using the SQL command line utility. **Please type the following statements to create tables and add 'sysadmin' rights to your database user** (**Figure 1**). Although you can give any table name you wish.

```
CREATE DATABASE ozeki
GO

USE ozeki
GO

CREATE TABLE ozekimessagein (
 id int IDENTITY (1,1),
 sender varchar(30),
 receiver varchar(30),
 msg nvarchar(160),
 senttime varchar(100),
 receivedtime varchar(100),
 operator varchar(30),
 msgtype varchar(30),
 reference varchar(30),
);

CREATE TABLE ozekimessageout (
 id int IDENTITY (1,1),
 sender varchar(30),
 receiver varchar(30),
 msg nvarchar(160),
 senttime varchar(100),
 receivedtime varchar(100),
 operator varchar(100),
 msgtype varchar(30),
 reference varchar(30),
 status varchar(30),
 errormsg varchar(250)
);

GO

sp_addLogin 'ozekiuser', 'ozekipass'
GO

sp_addsrvrolemember 'ozekiuser', 'sysadmin'
GO
```

**Figure 1** - CREATE TABLE statement. It also adds rights to 'ozekiuser'

Later on you can use the command prompt to modify the size or data type of the SMS messages.

Now you are ready to send your first SMS message by inserting a record into 'ozekimessageout'.
**Use the following SQL statement example:**

```
insert into ozekimessageout (receiver,msg,status) values ("+44111223465","Hello
 world","Send");
GO
```

Tip: SQL Express is free and can be downloaded from the following URL:
https://www.microsoft.com/en-us/sql-server/sql-server-editions-express

# Trouble shooting

In some systems it is harder to connect to SQL Express. If this is the case you can see the following error message:

ERROR 6001: Database connection error: The 'SQLNCLI' provider is not registered on the local machine..

SQLNCLI should be installed with SQL Express. This error shows that it is not installed on your machine. Please try to change the provider to 'sqloledb', so you can use OLE DB. By changing the provider the connections string changes as well:

**Connection type:**

**OLE DB**

**Connection string:**

**Provider=SQLOLEDB;Data Source=.\SQLEXPRESS;User ID=ozekiuser;password=ozekipass;Database=ozeki;Persist Security Info=True**

---

**Or change the whole connection type to:**

**ODBC**

**and use the following connection string:**

**Driver={SQL Server};Server=.\SQLEXPRESS;User ID=ozekiuser;password=ozekipass;Database=ozeki;Persist Security Info=True;**

---

**Or change the whole connection type to:**

**SQLServer**

**and use the following connection string:**

**Server=.\SQLEXPRESS;User ID=ozekiuser;password=ozekipass;Database=ozeki;Persist Security Info=True;**

One of the 3 strings above should fix this ERROR.

# Microsoft SQL Server 2008

Try the following string if you need to connect to Microsoft SQL Server 2008:

**Provider=SQLOLEDB.1;Data Source=YourHostName;Persist Security Info=False;Integrated Security=SSPI;User ID=UserName;Password=User'sPassword;Initial Catalog=DatabaseName**

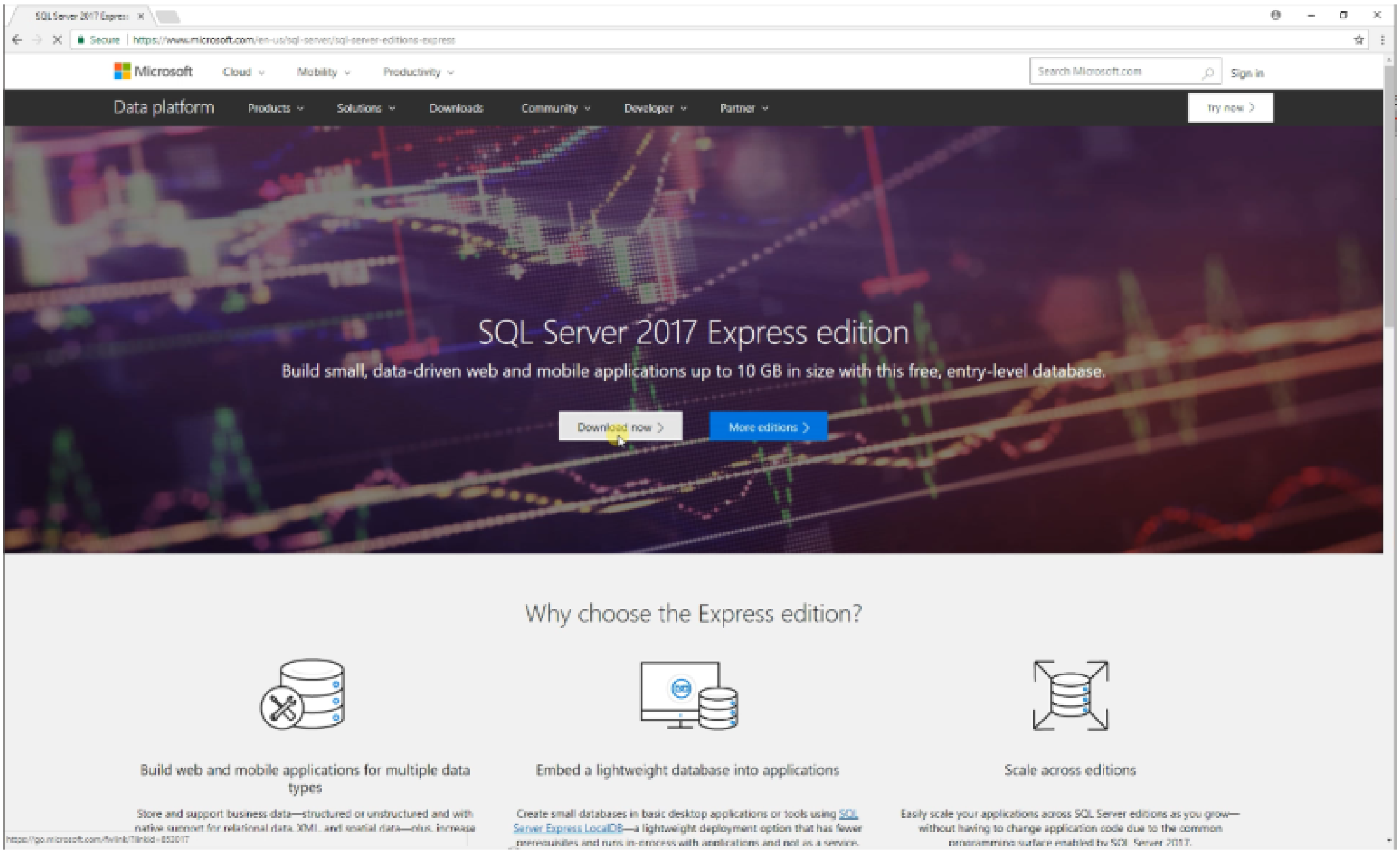
# How to setup MSSQL

***This page gives you the steps to take to download, install and configure Microsoft SQL Server (Express edition). Microsoft SQL Server and Micrsoft SQL Express are basically the same products, so the steps to get them up and running are pretty much the same.***

**Content**

---

Please jump to the next video or if you scroll down you can find screenshots that describe the installation details of MSSQL from the downloading and installing the package, through instance creation until creating the administrator account's password.



SQL Server 2017 Express edition

Build small, data-driven web and mobile applications up to 10 GB in size with this free, entry-level database.

Why choose the Express edition?

Build web and mobile applications for multiple data types

Store and support business data—structured or unstructured and with native support for relational data, XML, and spatial data—plus, increase

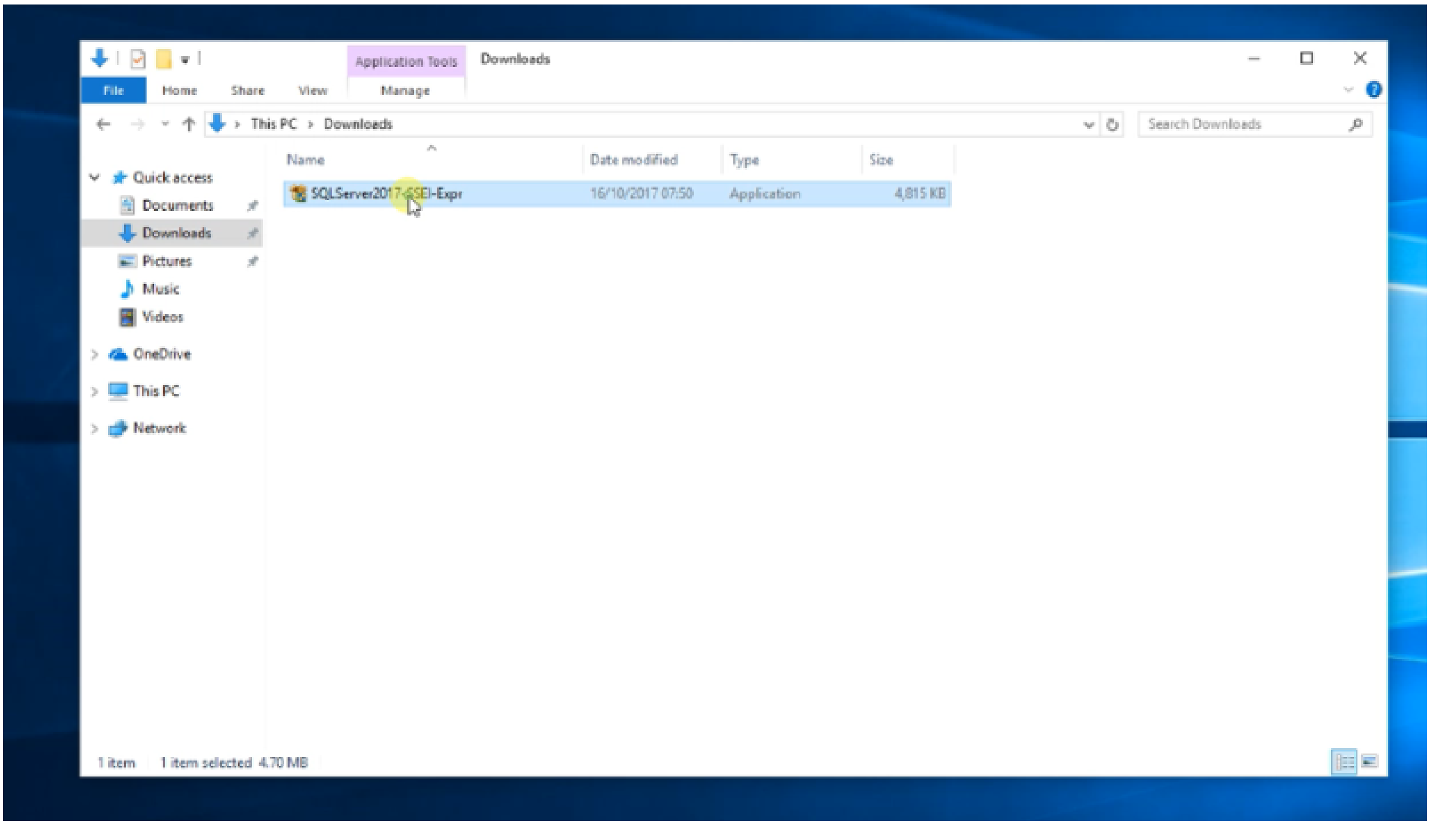
Embed a lightweight database into applications

Create small databases in basic desktop applications or tools using SQL Server Express LocalDB—a lightweight deployment option that has fewer prerequisites and runs in-process with applications and not as a service.

Scale across editions

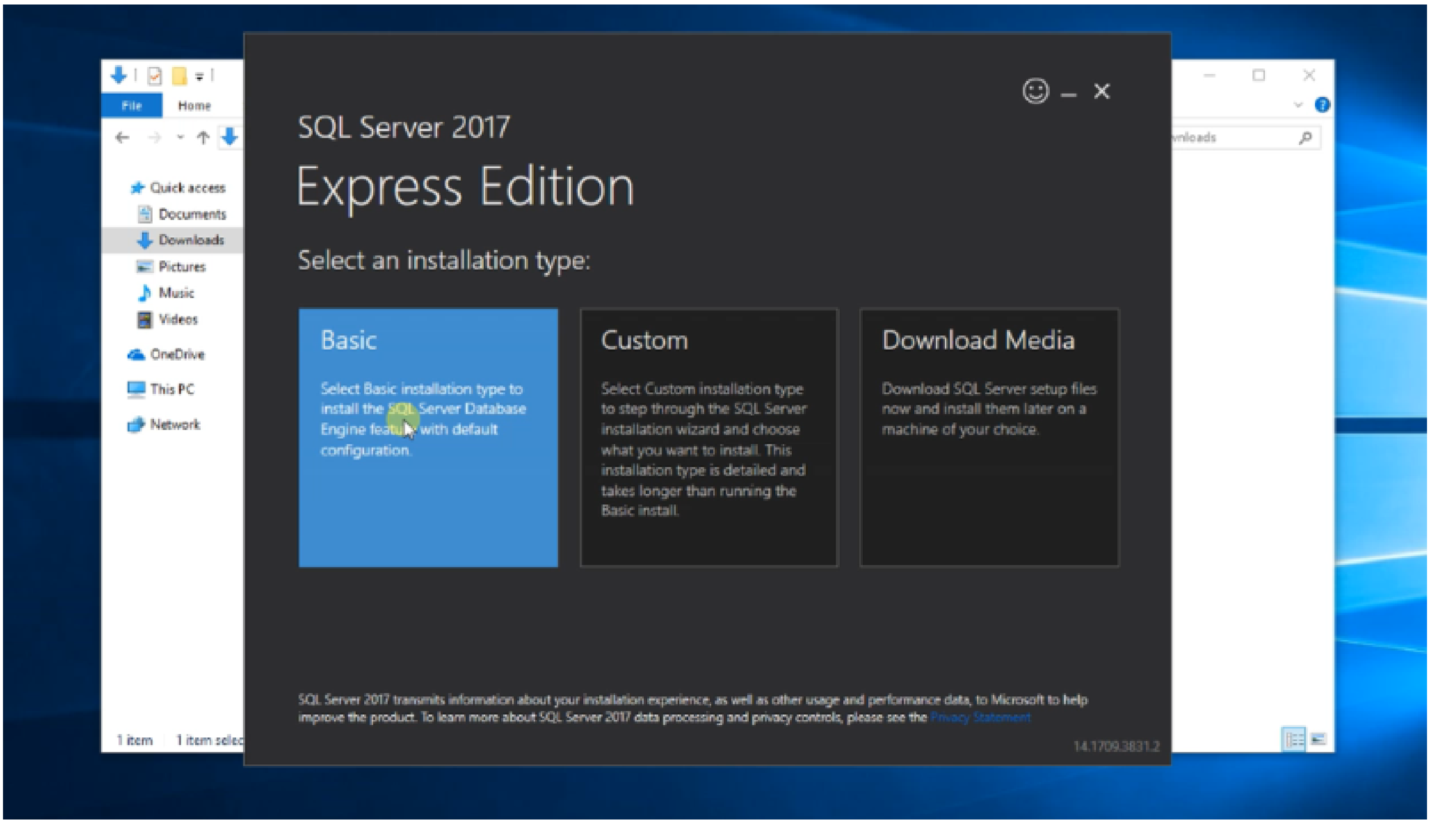
Easily scale your applications across SQL Server editions as you grow—without having to change application code due to the common programming surface enabled by SQL Server 2017.

**Figure 1 -** Download SQL Express from Microsoft's webpage



| Name | Date modified | Type | Size |
|---|---|---|---|
| SQLServer2017-SSEI-Expr | 16/10/2017 07:50 | Application | 4,815 KB |

**Figure 2 -** Start the installation



SQL Server 2017

Express Edition

Select an installation type:

Basic

Select Basic installation type to install the SQL Server Database Engine feature with default configuration.

Custom

Select Custom installation type to step through the SQL Server installation wizard and choose what you want to install. This installation type is detailed and takes longer than running the Basic install.

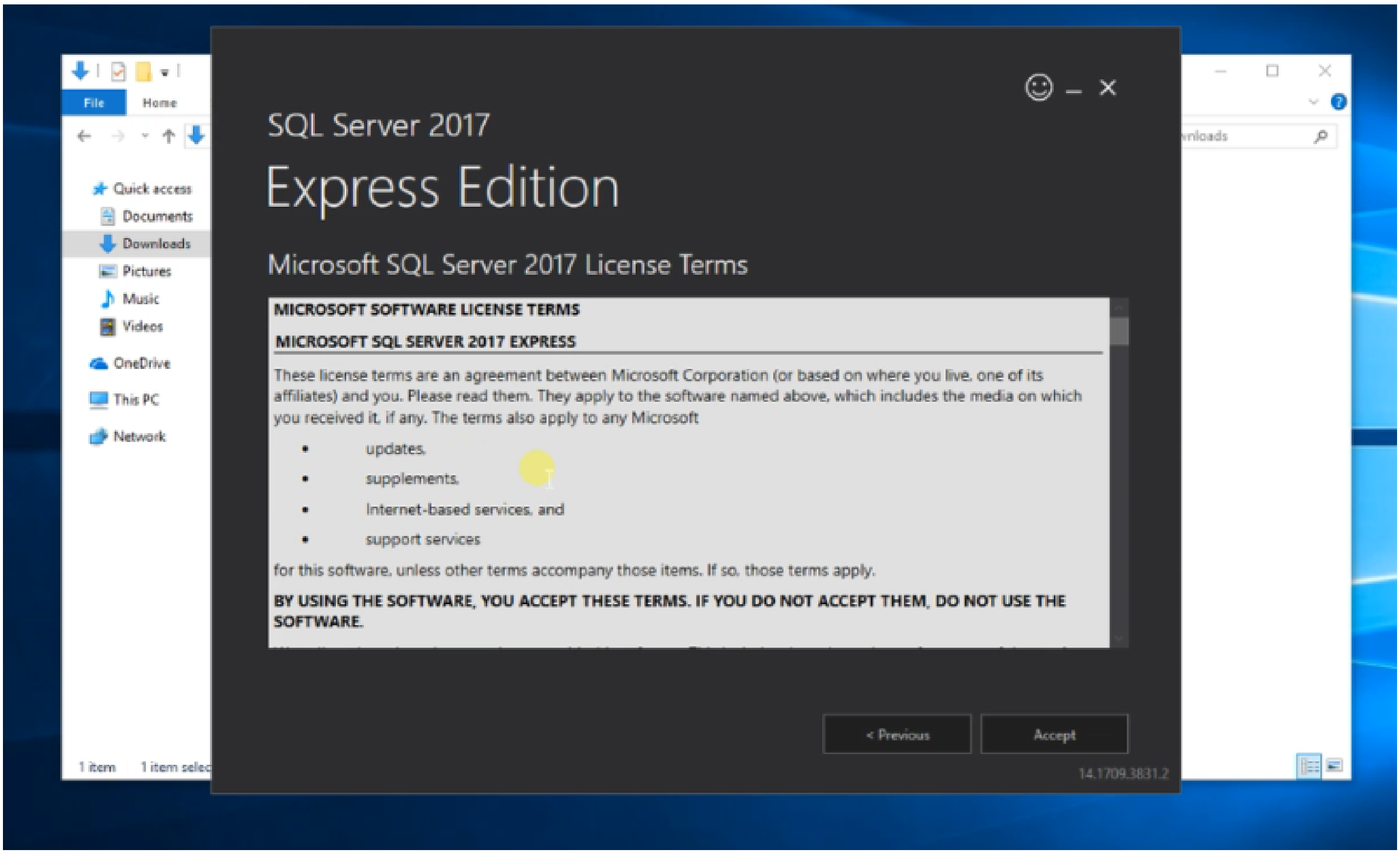
Download Media

Download SQL Server setup files now and install them later on a machine of your choice.

SQL Server 2017 transmits information about your installation experience, as well as other usage and performance data, to Microsoft to help improve the product. To learn more about SQL Server 2017 data processing and privacy controls, please see the Privacy Statement

14.1709.3831.2

**Figure 3 -** Select an installation type



SQL Server 2017

Express Edition

Microsoft SQL Server 2017 License Terms

**MICROSOFT SOFTWARE LICENSE TERMS**

**MICROSOFT SQL SERVER 2017 EXPRESS**

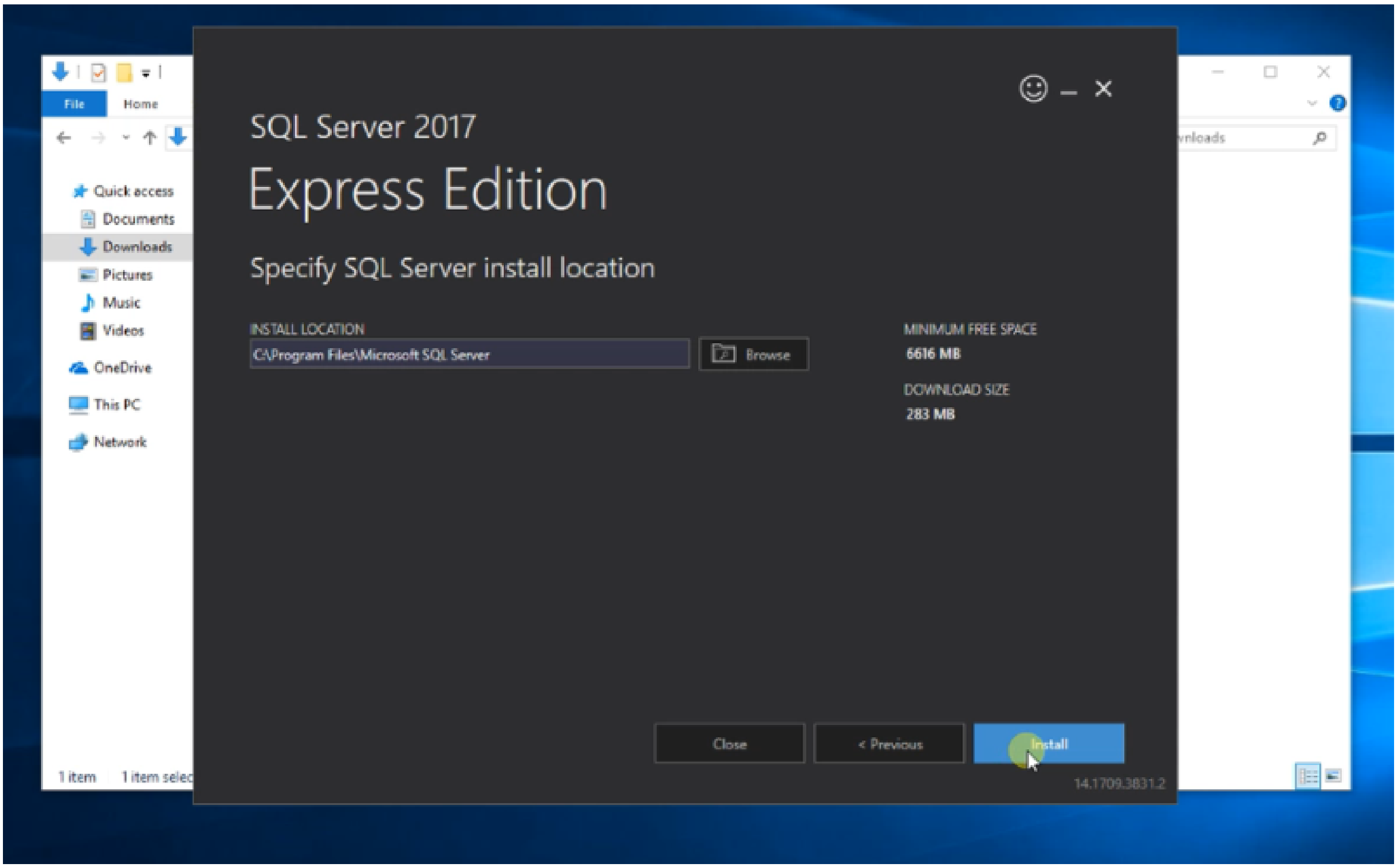
These license terms are an agreement between Microsoft Corporation (or based on where you live, one of its affiliates) and you. Please read them. They apply to the software named above, which includes the media on which you received it, if any. The terms also apply to any Microsoft

- updates,
- supplements,
- Internet-based services, and
- support services

for this software, unless other terms accompany those items. If so, those terms apply.

**BY USING THE SOFTWARE, YOU ACCEPT THESE TERMS. IF YOU DO NOT ACCEPT THEM, DO NOT USE THE SOFTWARE.**

**Figure 4 -** Accept license terms



SQL Server 2017

Express Edition

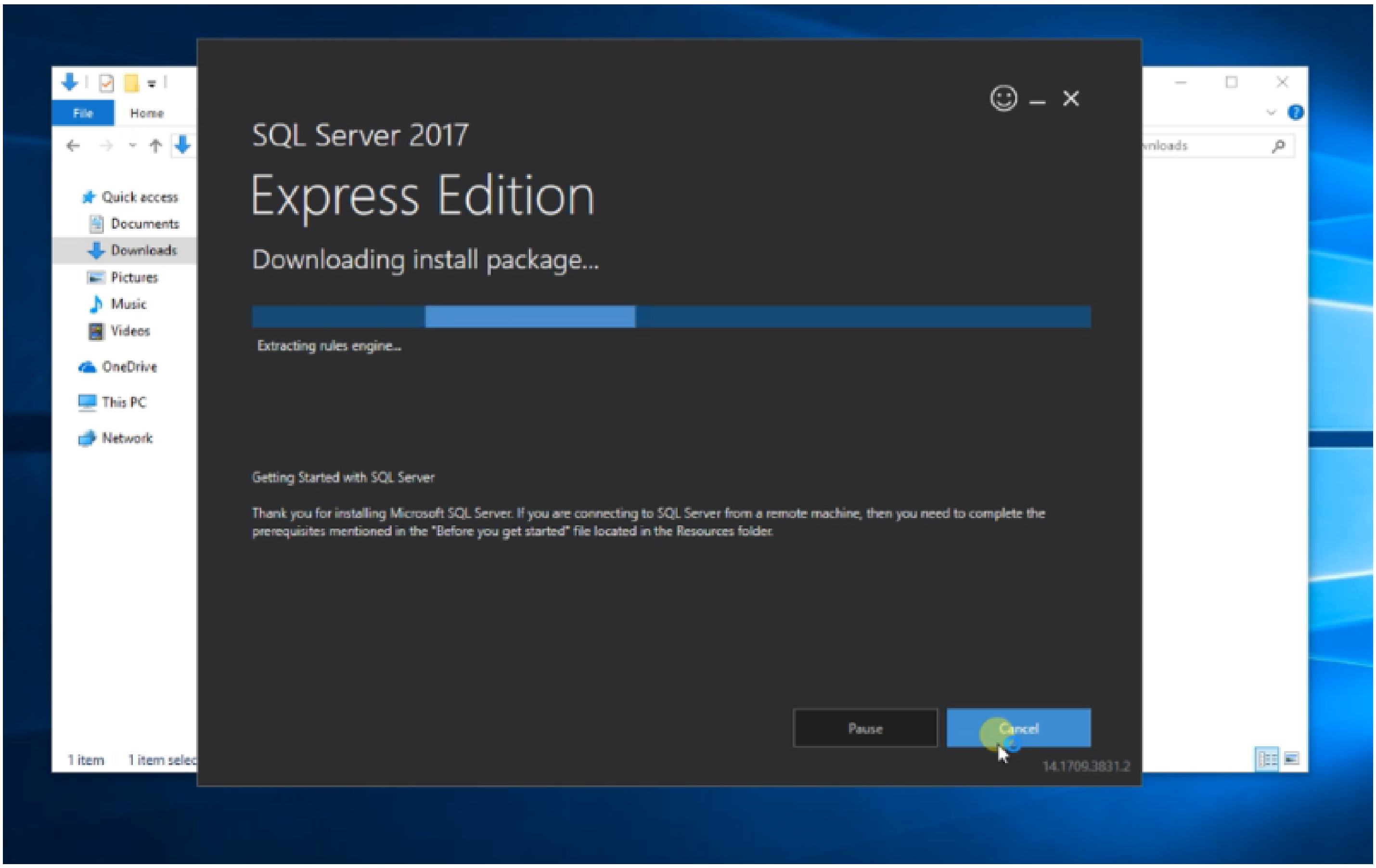
Specify SQL Server install location

INSTALL LOCATION: C:\Program Files\Microsoft SQL Server

MINIMUM FREE SPACE: 6616 MB

DOWNLOAD SIZE: 283 MB

**Figure 5 -** Click the 'Install' button



SQL Server 2017

Express Edition

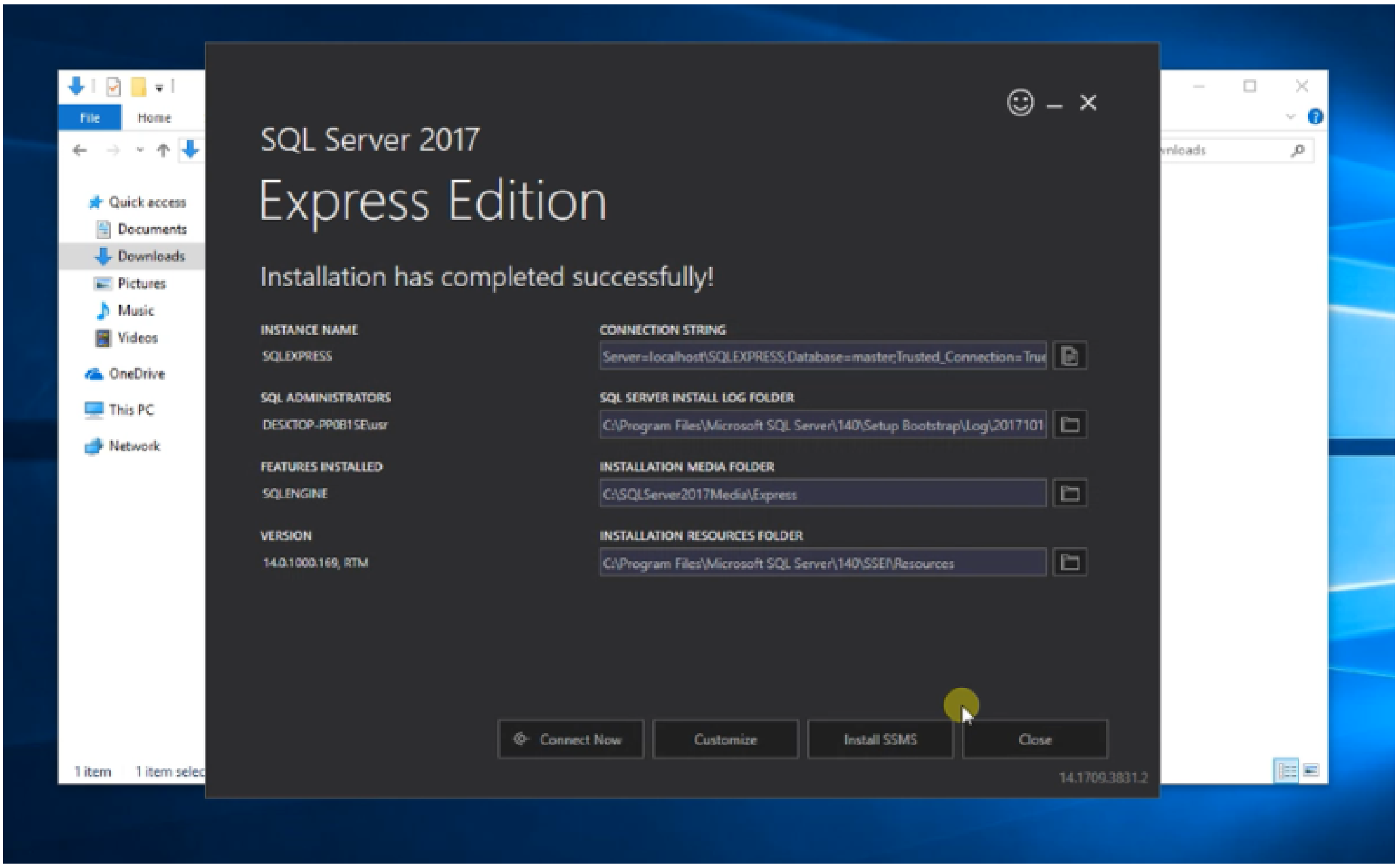
Downloading install package...

Extracting rules engine...

Getting Started with SQL Server

Thank you for installing Microsoft SQL Server. If you are connecting to SQL Server from a remote machine, then you need to complete the prerequisites mentioned in the "Before you get started" file located in the Resources folder.

**Figure 6 -** Wait until the installation is finished



SQL Server 2017

Express Edition

Installation has completed successfully!

INSTANCE NAME
SQLEXPRESS

CONNECTION STRING
Server=localhost\SQLEXPRESS;Database=master;Trusted_Connection=True

SQL ADMINISTRATORS
DESKTOP-PP0B15E\usr

SQL SERVER INSTALL LOG FOLDER
C:\Program Files\Microsoft SQL Server\140\Setup Bootstrap\Log\2017101

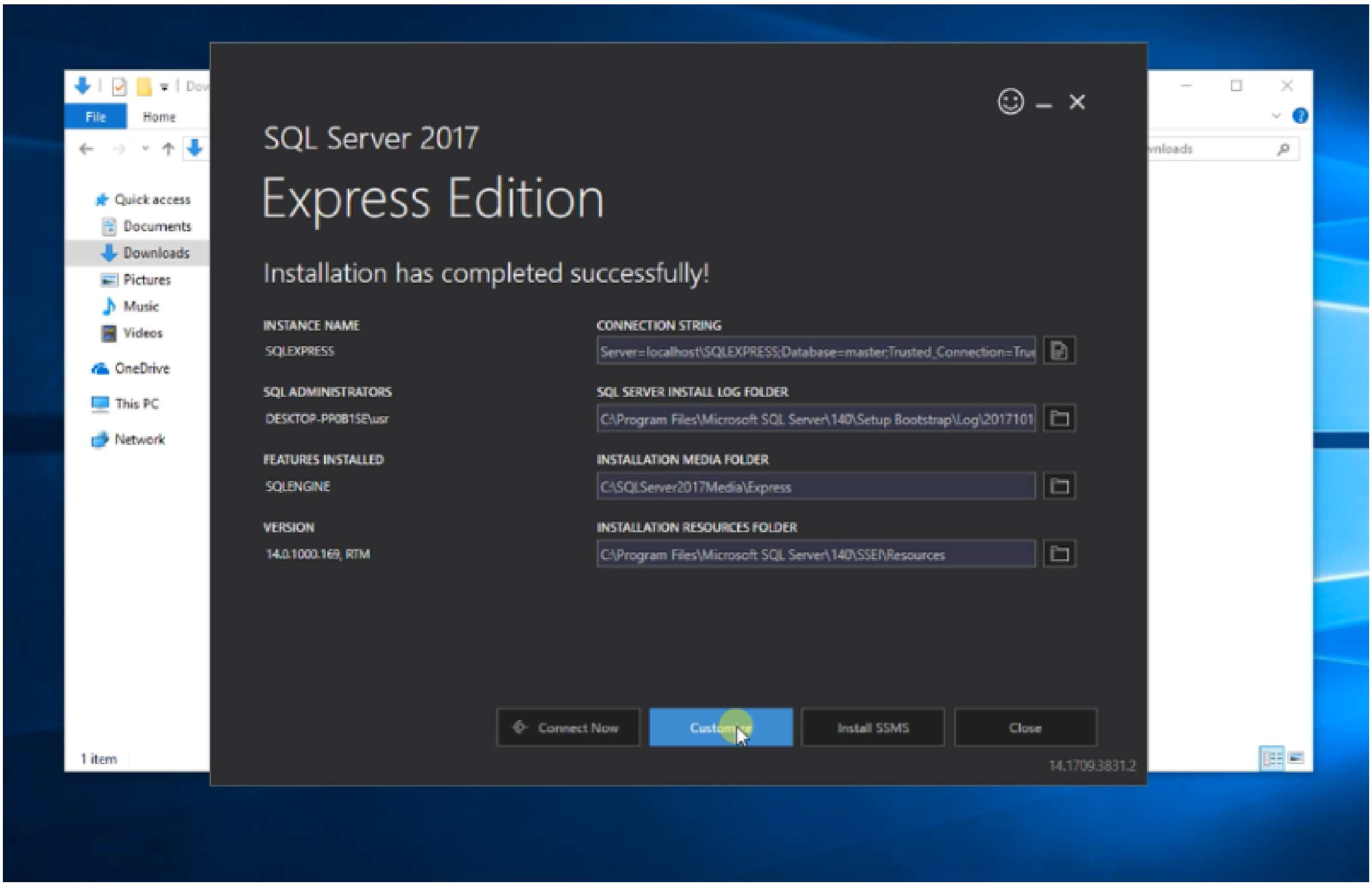
FEATURES INSTALLED
SQLENGINE

INSTALLATION MEDIA FOLDER
C:\SQLServer2017Media\Express

VERSION
14.0.1000.169, RTM

INSTALLATION RESOURCES FOLDER
C:\Program Files\Microsoft SQL Server\140\SSEI\Resources

**Figure 7 -** Installation has been completed



SQL Server 2017

Express Edition

Installation has completed successfully!

INSTANCE NAME
SQLEXPRESS

CONNECTION STRING
Server=localhost\SQLEXPRESS;Database=master;Trusted_Connection=Tru

SQL ADMINISTRATORS
DESKTOP-PP0B15E\usr

SQL SERVER INSTALL LOG FOLDER
C:\Program Files\Microsoft SQL Server\140\Setup Bootstrap\Log\2017101

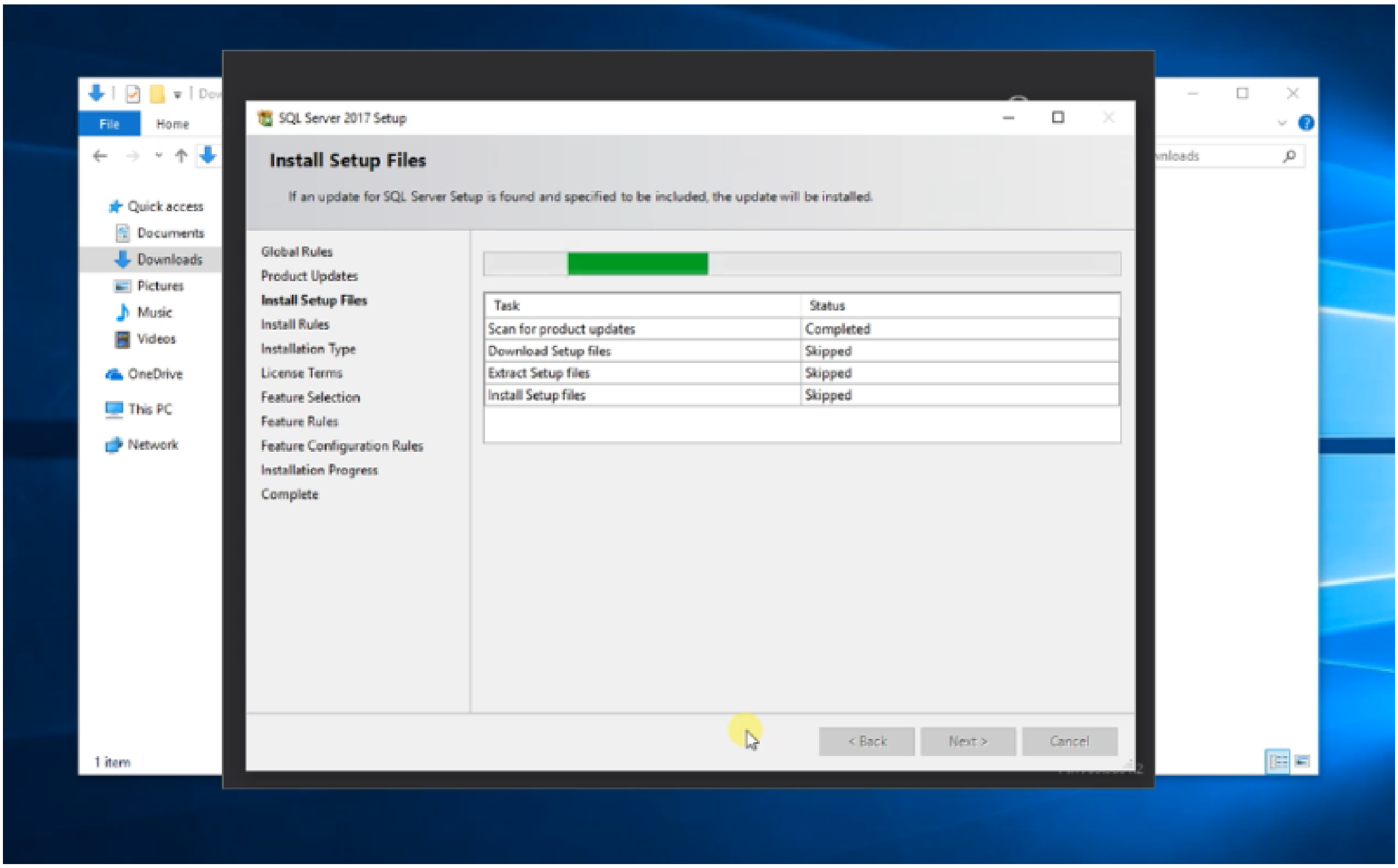
FEATURES INSTALLED
SQLENGINE

INSTALLATION MEDIA FOLDER
C:\SQLServer2017Media\Express

VERSION
14.0.1000.169, RTM

INSTALLATION RESOURCES FOLDER
C:\Program Files\Microsoft SQL Server\140\SSEI\Resources

**Figure 8 -** Click the 'Customize' button



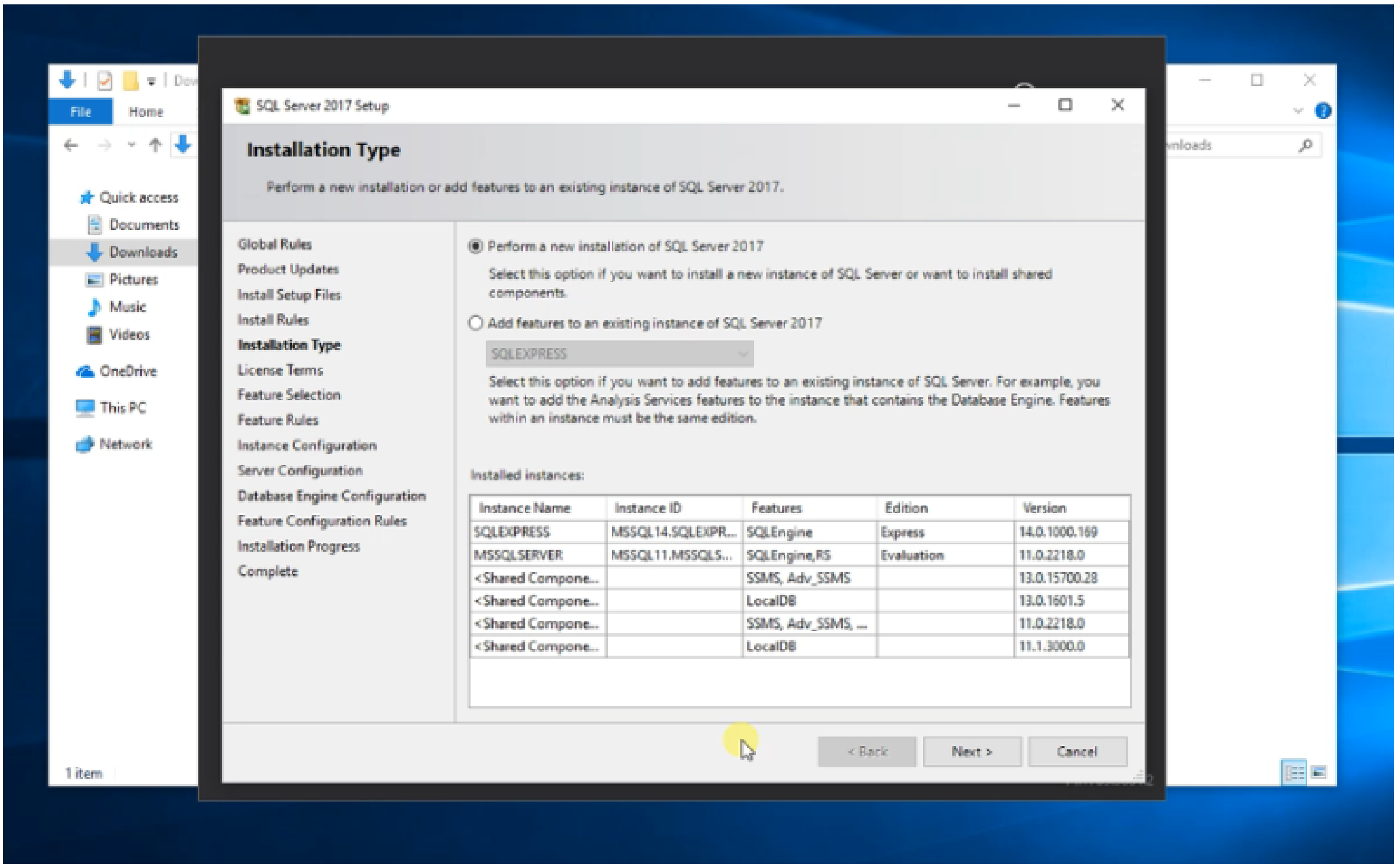
SQL Server 2017 Setup

**Install Setup Files**

If an update for SQL Server Setup is found and specified to be included, the update will be installed.

| Task | Status |
|---|---|
| Scan for product updates | Completed |
| Download Setup files | Skipped |
| Extract Setup files | Skipped |
| Install Setup files | Skipped |

**Figure 9 -** Install setup files



**Installation Type**

Perform a new installation or add features to an existing instance of SQL Server 2017.

● Perform a new installation of SQL Server 2017

Select this option if you want to install a new instance of SQL Server or want to install shared components.

○ Add features to an existing instance of SQL Server 2017

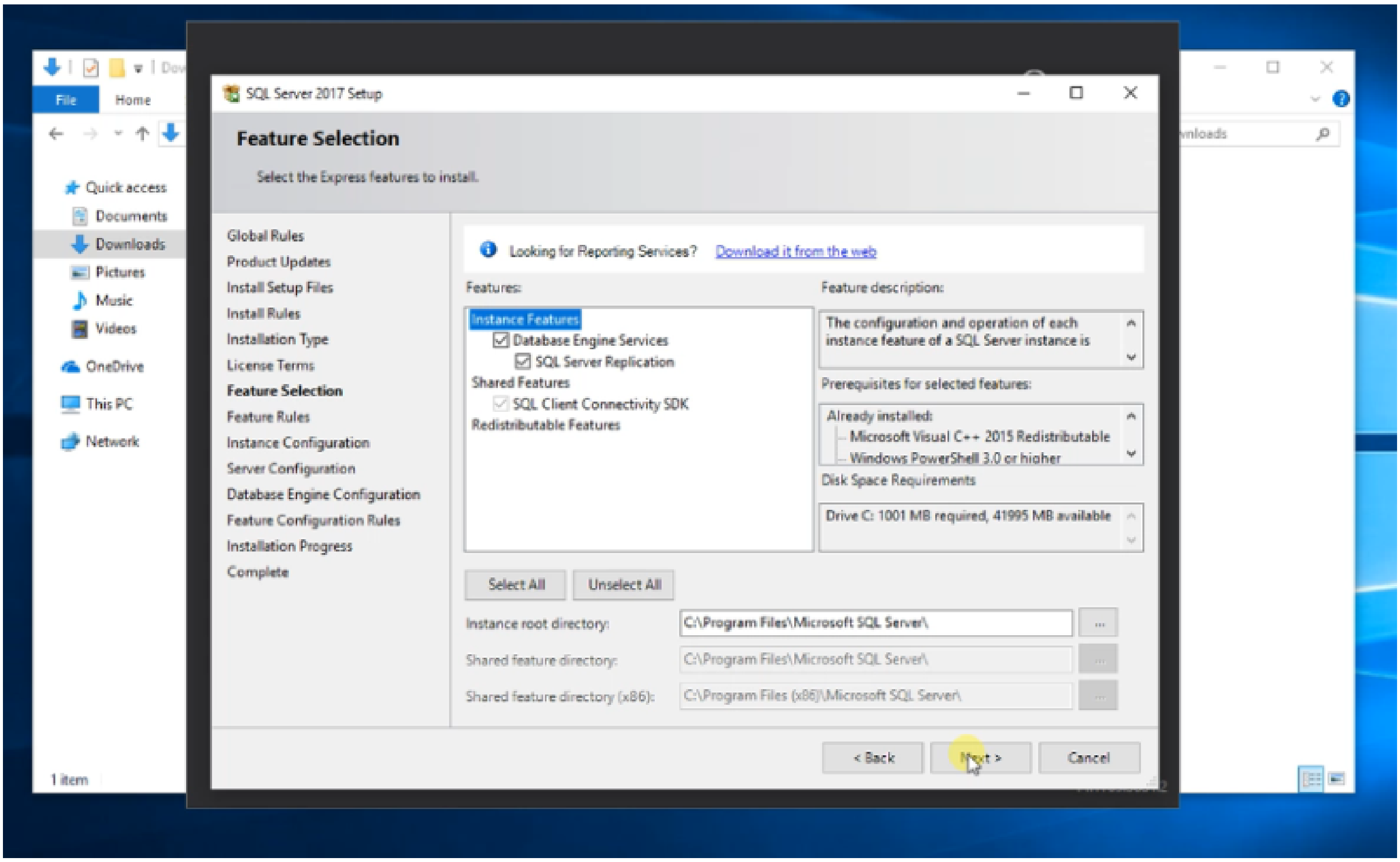
SQLEXPRESS

Select this option if you want to add features to an existing instance of SQL Server. For example, you want to add the Analysis Services features to the instance that contains the Database Engine. Features within an instance must be the same edition.

Installed instances:

| Instance Name | Instance ID | Features | Edition | Version |
|---|---|---|---|---|
| SQLEXPRESS | MSSQL14.SQLEXPR... | SQLEngine | Express | 14.0.1000.169 |
| MSSQLSERVER | MSSQL11.MSSQLS... | SQLEngine,RS | Evaluation | 11.0.2218.0 |
| <Shared Compone... | | SSMS, Adv_SSMS | | 13.0.15700.28 |
| <Shared Compone... | | LocalDB | | 13.0.1601.5 |
| <Shared Compone... | | SSMS, Adv_SSMS, ... | | 11.0.2218.0 |
| <Shared Compone... | | LocalDB | | 11.1.3000.0 |

**Figure 10 -** Select installation type



**Feature Selection**

Select the Express features to install.

Looking for Reporting Services? Download it from the web

Features:

Instance Features
☑ Database Engine Services
☑ SQL Server Replication
Shared Features
☑ SQL Client Connectivity SDK
Redistributable Features

Feature description:

The configuration and operation of each instance feature of a SQL Server instance is

Prerequisites for selected features:

Already installed:
Microsoft Visual C++ 2015 Redistributable
Windows PowerShell 3.0 or higher

Disk Space Requirements

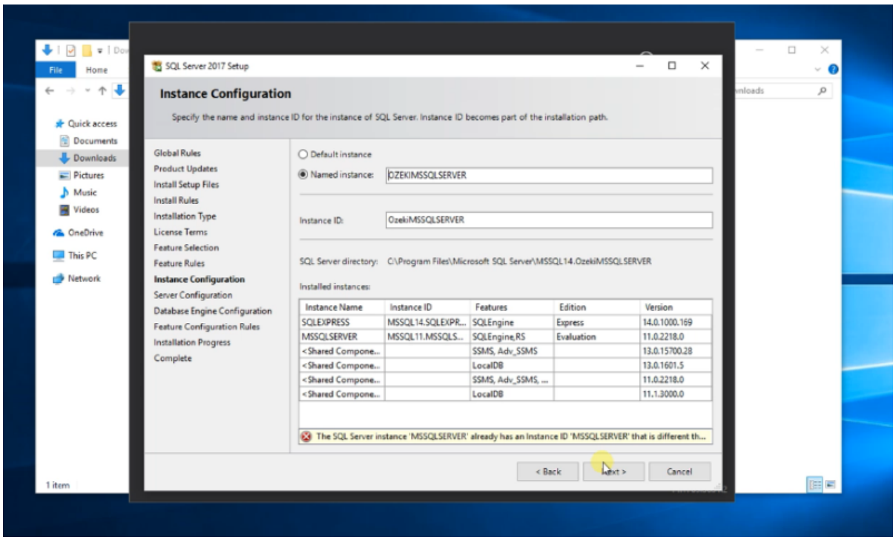
Drive C: 1001 MB required, 41995 MB available

Instance root directory: C:\Program Files\Microsoft SQL Server\

Shared feature directory: C:\Program Files\Microsoft SQL Server\

Shared feature directory (x86): C:\Program Files (x86)\Microsoft SQL Server\

**Figure 11 -** Select instance features



**Instance Configuration**

Specify the name and instance ID for the instance of SQL Server. Instance ID becomes part of the installation path.

○ Default instance

● Named instance: OZEKIMSSQLSERVER

Instance ID: OzekiMSSQLSERVER

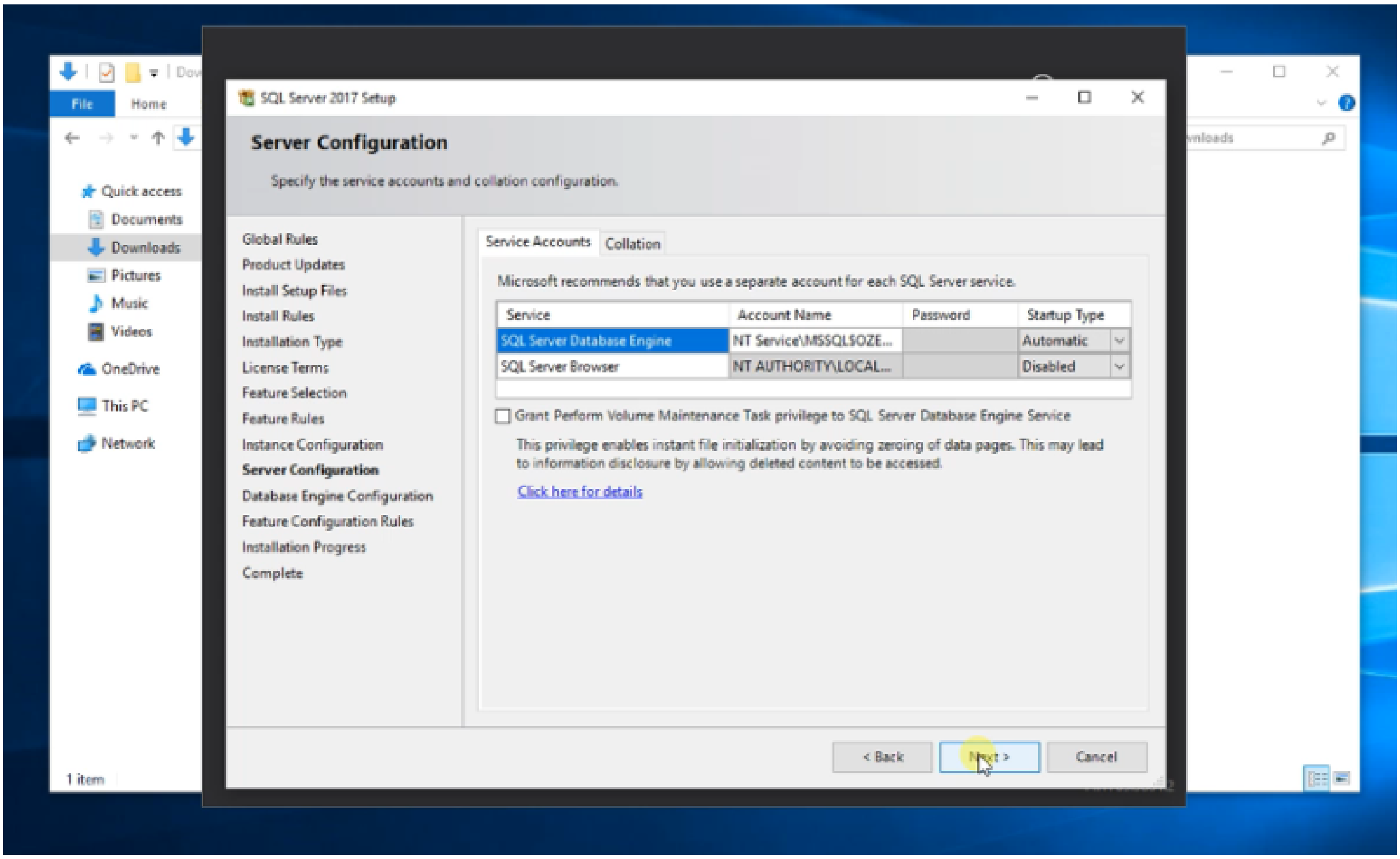
SQL Server directory: C:\Program Files\Microsoft SQL Server\MSSQL14.OzekiMSSQLSERVER

Installed instances:

| Instance Name | Instance ID | Features | Edition | Version |
|---|---|---|---|---|
| SQLEXPRESS | MSSQL14.SQLEXPR... | SQLEngine | Express | 14.0.1000.169 |
| MSSQLSERVER | MSSQL11.MSSQLS... | SQLEngine,RS | Evaluation | 11.0.2218.0 |
| <Shared Compone... | | SSMS, Adv_SSMS | | 13.0.15700.28 |
| <Shared Compone... | | LocalDB | | 13.0.1601.5 |
| <Shared Compone... | | SSMS, Adv_SSMS, ... | | 11.0.2218.0 |
| <Shared Compone... | | LocalDB | | 11.1.3000.0 |

The SQL Server instance 'MSSQLSERVER' already has an Instance ID 'MSSQLSERVER' that is different th...

**Figure 12 -** Specify the name for the instance



**Server Configuration**

Specify the service accounts and collation configuration.

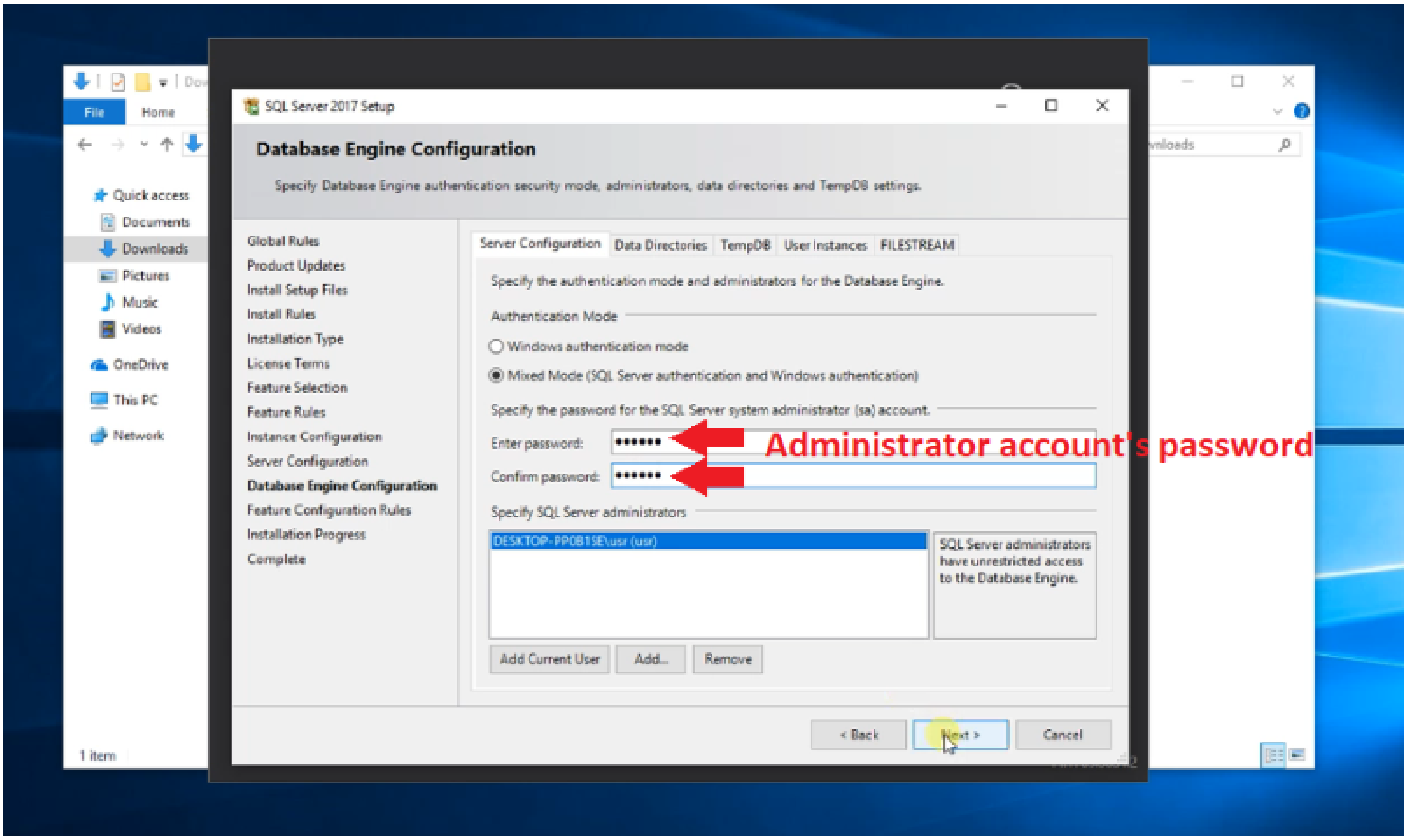
Microsoft recommends that you use a separate account for each SQL Server service.

| Service | Account Name | Password | Startup Type |
|---|---|---|---|
| SQL Server Database Engine | NT Service\MSSQL$OZE... | | Automatic |
| SQL Server Browser | NT AUTHORITY\LOCAL... | | Disabled |

□ Grant Perform Volume Maintenance Task privilege to SQL Server Database Engine Service

This privilege enables instant file initialization by avoiding zeroing of data pages. This may lead to information disclosure by allowing deleted content to be accessed.

**Figure 13 -** Specify the service accounts



**Database Engine Configuration**

Specify Database Engine authentication security mode, administrators, data directories and TempDB settings.

Specify the authentication mode and administrators for the Database Engine.

Authentication Mode

○ Windows authentication mode

● Mixed Mode (SQL Server authentication and Windows authentication)

Specify the password for the SQL Server system administrator (sa) account.

Enter password: ••••••

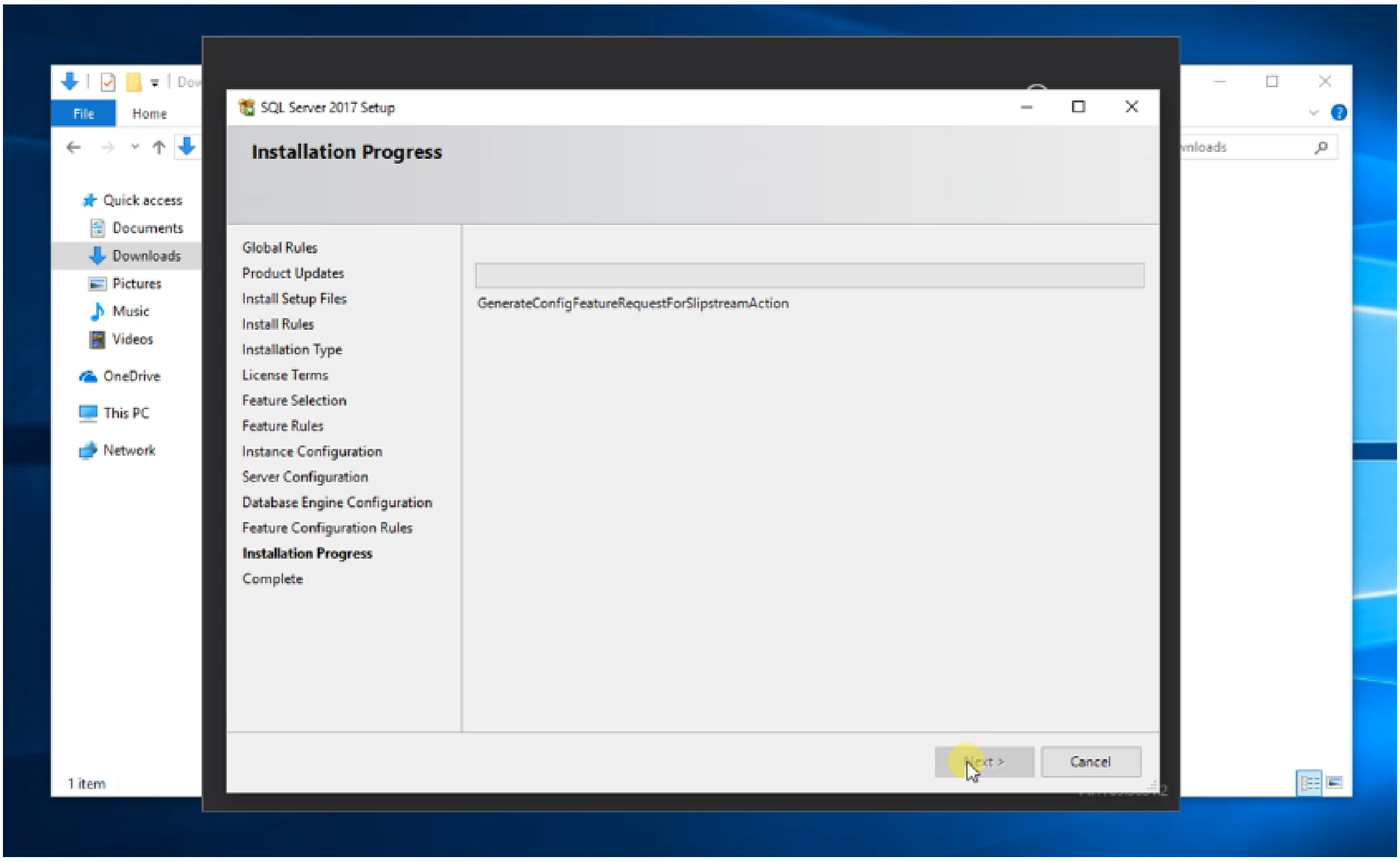
Confirm password: ••••••

Specify SQL Server administrators

DESKTOP-PP0B15E\usr (usr)

SQL Server administrators have unrestricted access to the Database Engine.

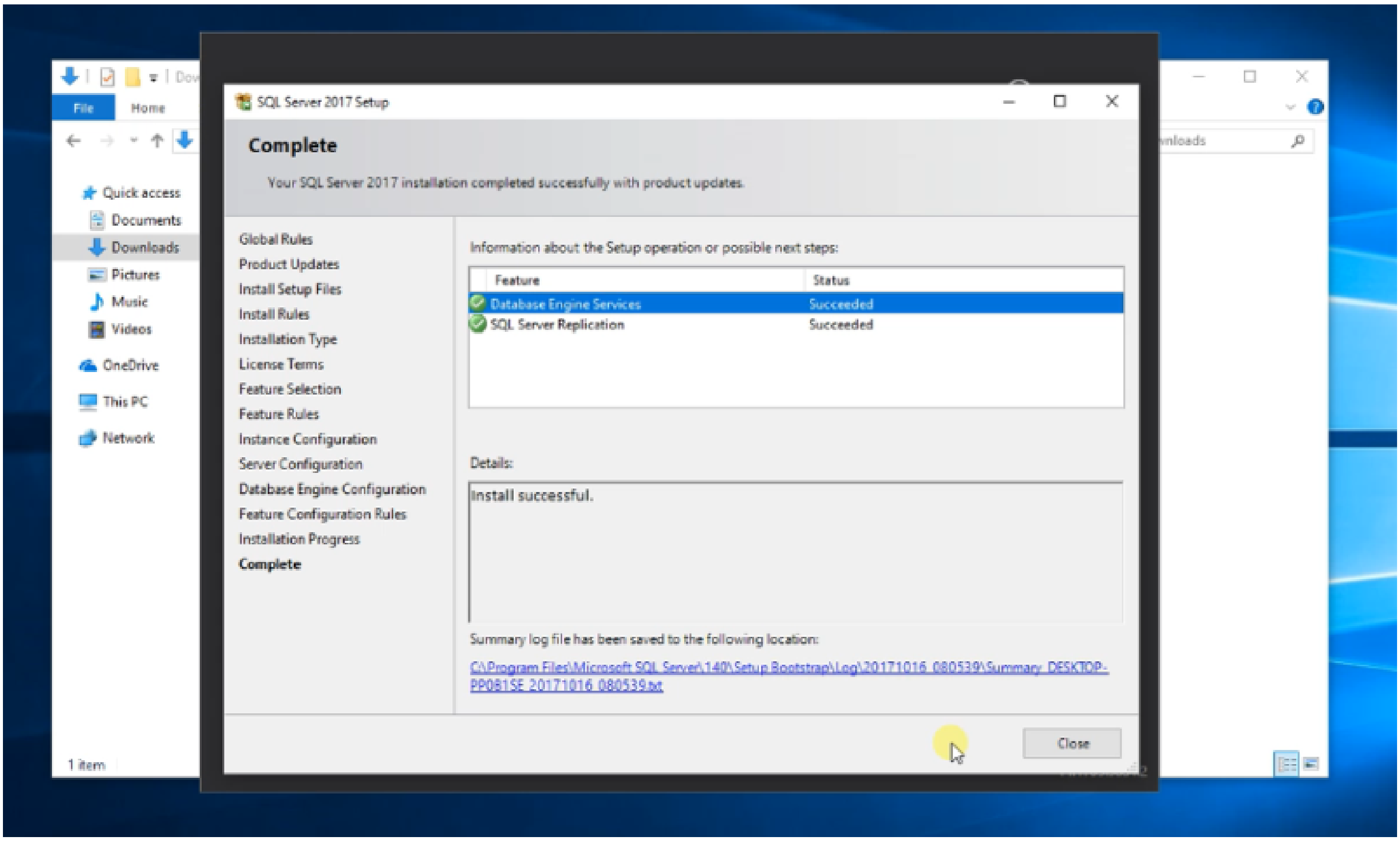
**Figure 14 -** Specify the authentication mode and the password for the administrator account



**Installation Progress**

GenerateConfigFeatureRequestForSlipstreamAction

**Figure 15 -** Wait until the installation is completed



Complete

Your SQL Server 2017 installation completed successfully with product updates.

Information about the Setup operation or possible next steps:

| Feature | Status |
|---|---|
| Database Engine Services | Succeeded |
| SQL Server Replication | Succeeded |

Details:

Install successful.

Summary log file has been saved to the following location:

C:\Program Files\Microsoft SQL Server\140\Setup Bootstrap\Log\20171016_080539\Summary_DESKTOP-PP081SE_20171016_080539.txt

**Figure 16 -** Installation has been completed

# MSSQL Database layout for SMS

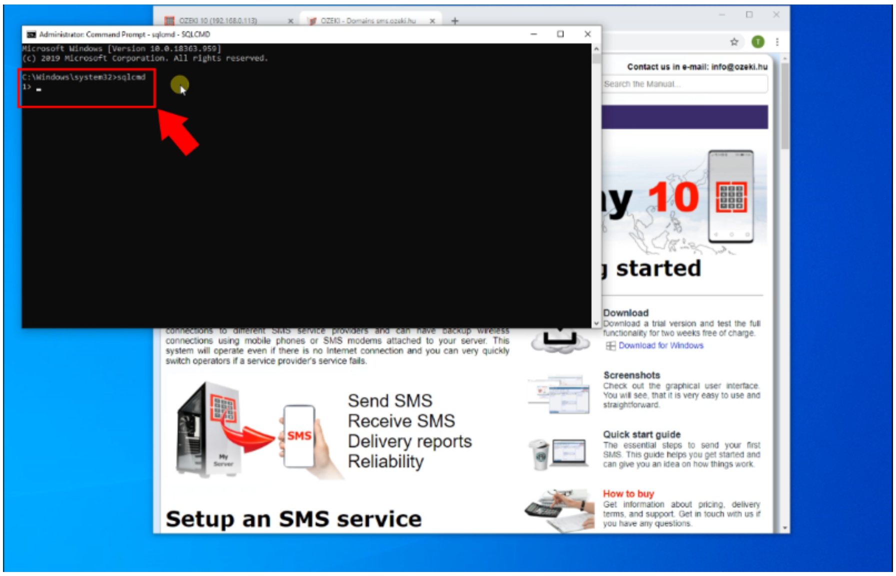
***This page gives you the database layout to use when you wish to setup an SQL to SMS solution. You will see that two database tables need to be created. One will be used for sending SMS messages and the other for receiving. This guide shows you how to connect to your MS SQL server and how to create these two database tables.***

**Content**

1. Connect to MSSQL Express
2. Copy CREATE TABLE statement
3. Run CREATE TABLE statement

---

Please scroll down to copy the CREATE TABLE statements used in the video. If you have created the database in MSSQL Express, you can jump to the next video.



```
Microsoft Windows [Version 10.0.18363.959]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Windows\system32>sqlcmd
1> _
```

connections to different SMS service providers and can have backup wireless connections using mobile phones or SMS modems attached to your server. This system will operate even if there is no Internet connection and you can very quickly switch operators if a service provider's service fails.

Download a trial version and test the full functionality for two weeks free of charge.

Check out the graphical user interface. You will see, that it is very easy to use and straightforward.

The essential steps to send your first SMS. This guide helps you get started and can give you an idea on how things work.

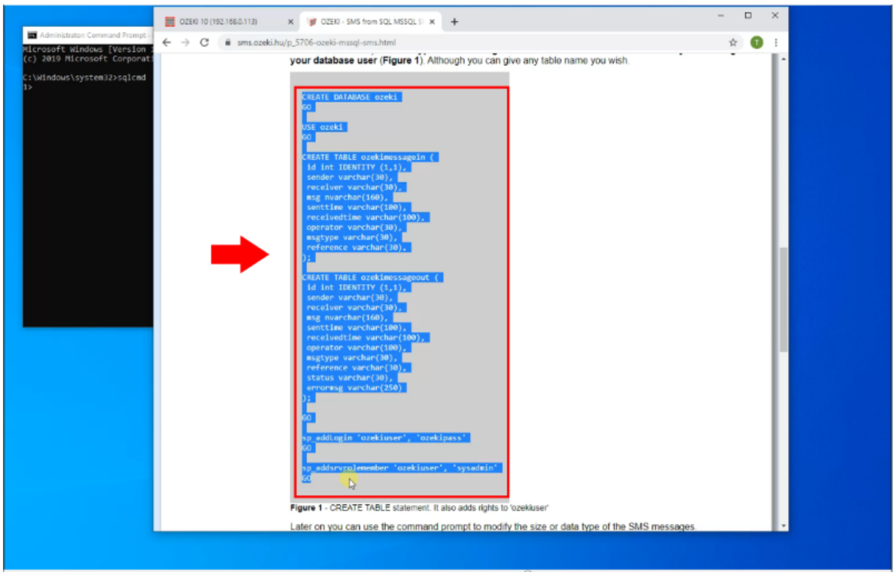
Get information about pricing, delivery terms, and support. Get in touch with us if you have any questions.

**Figure 1** - Connect to the Microsoft SQL Express database

```sql
create database ozeki
GO

use ozeki
GO

CREATE TABLE ozekimessagein (
 id int IDENTITY (1,1),
 sender varchar(30),
 receiver varchar(30),
 msg nvarchar(160),
 senttime varchar(100),
 receivedtime varchar(100),
 operator varchar(30),
 msgtype varchar(30),
 reference varchar(30),
);

CREATE TABLE ozekimessageout (
 id int IDENTITY (1,1),
 sender varchar(30),
 receiver varchar(30),
 msg nvarchar(160),
 senttime varchar(100),
 receivedtime varchar(100),
 operator varchar(100),
 msgtype varchar(30),
 reference varchar(30),
 status varchar(30),
 errormsg varchar(250)
);


GO
```

```
sp_addLogin 'ozekiuser', 'ozekipass'
GO

sp_addsrvrolemember 'ozekiuser', 'sysadmin'
GO
```

**Figure 2** - CREATE TABLE statement


Administrator: Command Prompt
OZEKI 10 (192.168.0.113)
OZEKI - SMS from SQL MSSQL S
sms.ozeki.hu/p_5706-ozeki-mssql-sms.html

```
Microsoft Windows [Version
(c) 2019 Microsoft Corporat

C:\Windows\system32>sqlcmd
1>
```

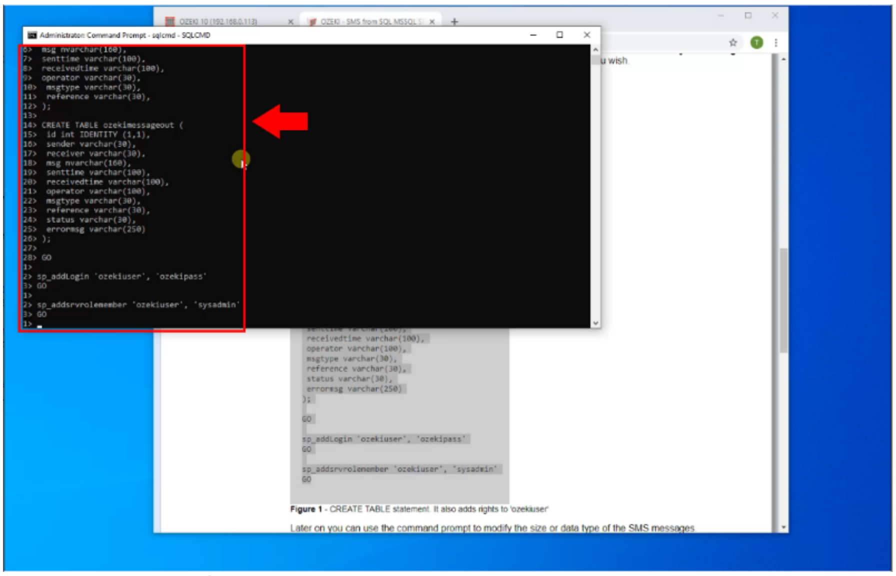
**your database user** (**Figure 1**). Although you can give any table name you wish.

```
CREATE DATABASE ozeki
GO

USE ozeki
GO

CREATE TABLE ozekimessagein (
 id int IDENTITY (1,1),
 sender varchar(30),
 receiver varchar(30),
 msg nvarchar(160),
 senttime varchar(100),
 receivedtime varchar(100),
 operator varchar(30),
 msgtype varchar(30),
 reference varchar(30),
);

CREATE TABLE ozekimessageout (
 id int IDENTITY (1,1),
 sender varchar(30),
 receiver varchar(30),
 msg nvarchar(160),
 senttime varchar(100),
 receivedtime varchar(100),
 operator varchar(100),
 msgtype varchar(30),
 reference varchar(30),
 status varchar(30),
 errormsg varchar(250)
);

GO

sp_addLogin 'ozekiuser', 'ozekipass'
GO

sp_addsrvrolemember 'ozekiuser', 'sysadmin'
GO
```

**Figure 1** - CREATE TABLE statement. It also adds rights to 'ozekiuser'

Later on you can use the command prompt to modify the size or data type of the SMS messages

**Figure 3 -** Copy CREATE TABLE statement from Figure 2


Administrator: Command Prompt - sqlcmd - SQLCMD
OZEKI 10 (192.168.0.113)
OZEKI - SMS from SQL MSSQL S

```
6>  msg nvarchar(160),
7>  senttime varchar(100),
8>  receivedtime varchar(100),
9>  operator varchar(30),
10>  msgtype varchar(30),
11>  reference varchar(30),
12> );
13>
14> CREATE TABLE ozekimessageout (
15>  id int IDENTITY (1,1),
16>  sender varchar(30),
17>  receiver varchar(30),
18>  msg nvarchar(160),
19>  senttime varchar(100),
20>  receivedtime varchar(100),
21>  operator varchar(100),
22>  msgtype varchar(30),
23>  reference varchar(30),
24>  status varchar(30),
25>  errormsg varchar(250)
26> );
27>
28> GO
1>
2> sp_addLogin 'ozekiuser', 'ozekipass'
3> GO
1>
2> sp_addsrvrolemember 'ozekiuser', 'sysadmin'
3> GO
1> _
```

u wish.

```
 receivedtime varchar(100),
 operator varchar(100),
 msgtype varchar(30),
 reference varchar(30),
 status varchar(30),
 errormsg varchar(250)
);

GO

sp_addLogin 'ozekiuser', 'ozekipass'
GO

sp_addsrvrolemember 'ozekiuser', 'sysadmin'
GO
```

**Figure 1** - CREATE TABLE statement. It also adds rights to 'ozekiuser'

Later on you can use the command prompt to modify the size or data type of the SMS messages

**Figure 4 -** Run CREATE TABLE statement on the database

# Create an MSSQL connection in Ozeki

***This guide explains, how you can create a database connection in Ozeki SMS Gateway to your MS SQL database server. This connection will be used to SELECT your outgoing messages from the outgoing databaes table, and to INSERT your incoming SMS messages into the incoming database table. If everything goes well these steps can be completed in about 5 minutes.***

**Content**

---

If you have created the database in MS SQL, you can jump to the next video. Although you can precisely examine the sequence by looking through the screenshots.

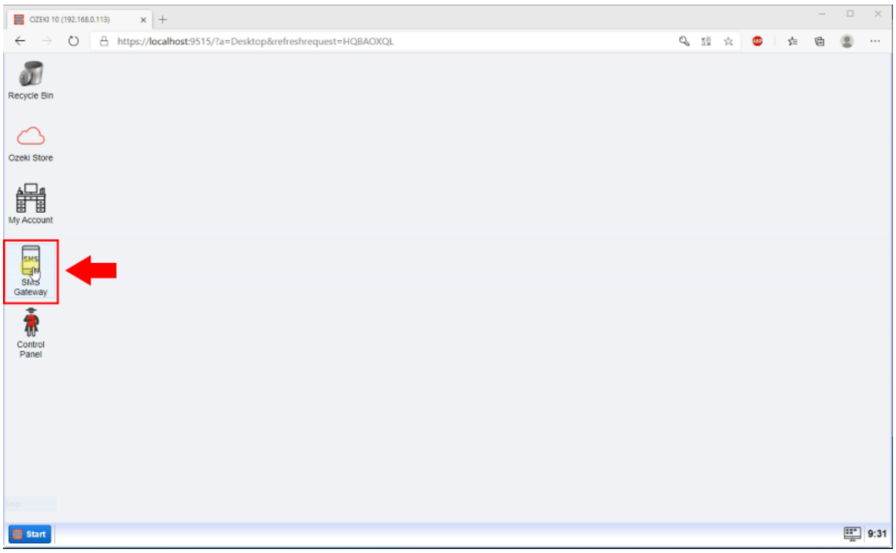
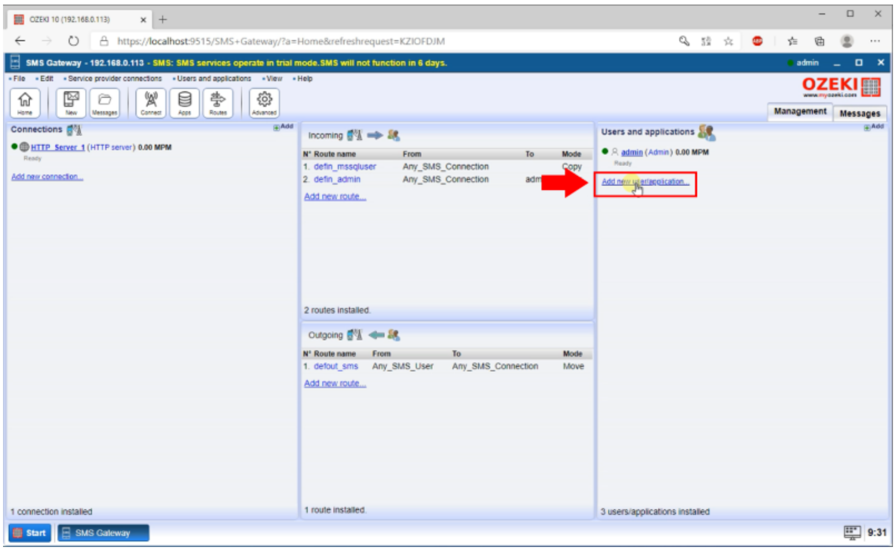


**Figure 1** - Open the SMS Gateway applicaton



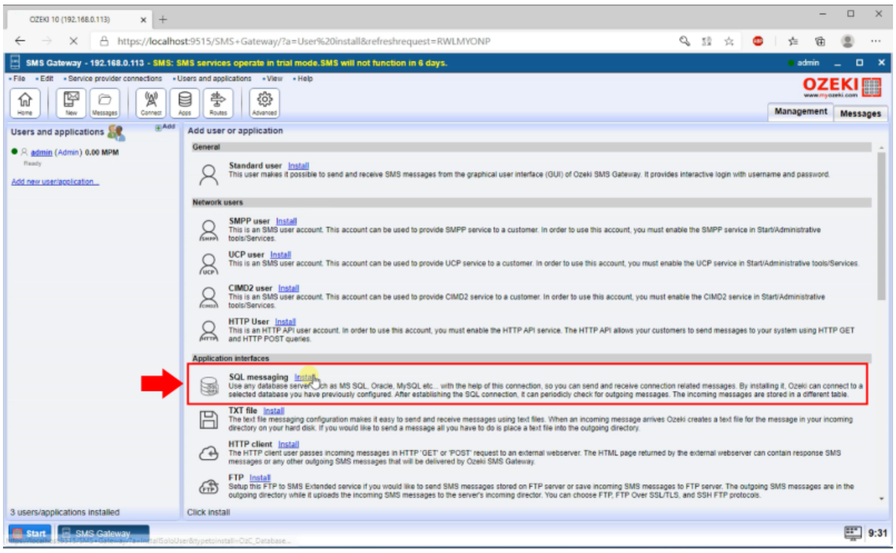
Incoming

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defin_mssqluser | Any_SMS_Connection | | Copy |
| 2. | defin_admin | Any_SMS_Connection | adm | |

Outgoing

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move |

**Figure 2** - Select 'Add new user or application'



Add user or application

General

Standard user Install
This user makes it possible to send and receive SMS messages from the graphical user interface (GUI) of Ozeki SMS Gateway. It provides interactive login with username and password.

Network users

SMPP user Install
This is an SMS user account. This account can be used to provide SMPP service to a customer. In order to use this account, you must enable the SMPP service in Start/Administrative tools/Services.

UCP user Install
This is an SMS user account. This account can be used to provide UCP service to a customer. In order to use this account, you must enable the UCP service in Start/Administrative tools/Services.

CIMD2 user Install
This is an SMS user account. This account can be used to provide CIMD2 service to a customer. In order to use this account, you must enable the CIMD2 service in Start/Administrative tools/Services.

HTTP User Install
This is an HTTP API user account. In order to use this account, you must enable the HTTP API service. The HTTP API allows your customers to send messages to your system using HTTP GET and HTTP POST queries.

Application interfaces

SQL messaging Install
Use any database server such as MS SQL, Oracle, MySQL etc... with the help of this connection, so you can send and receive connection related messages. By installing it, Ozeki can connect to a selected database you have previously configured. After establishing the SQL connection, it can periodicly check for outgoing messages. The incoming messages are stored in a different table.

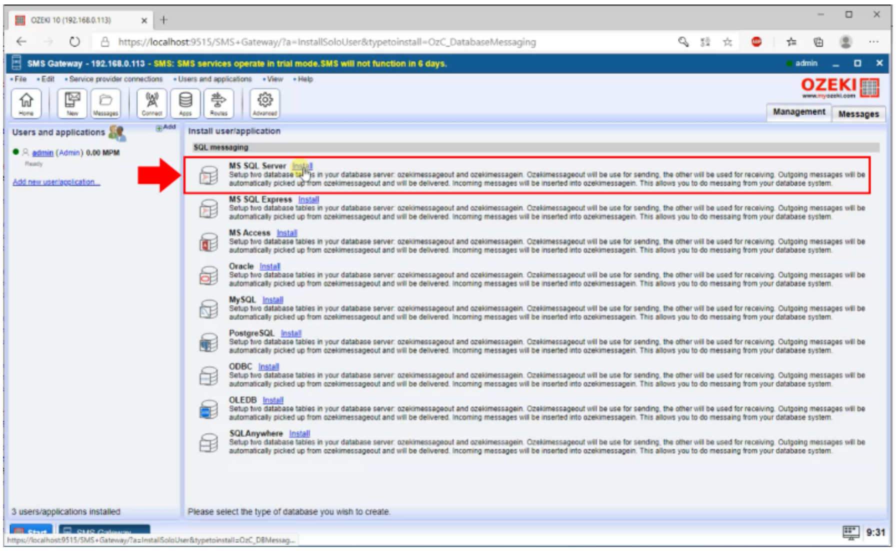
TXT file Install
The text file messaging configuration makes it easy to send and receive messages using text files. When an incoming message arrives Ozeki creates a text file for the message in your incoming directory on your hard disk. If you would like to send a message all you have to do is place a text file into the outgoing directory.

HTTP client Install
The HTTP client user passes incoming messages in HTTP 'GET' or 'POST' request to an external webserver. The HTML page returned by the external webserver can contain response SMS messages or any other outgoing SMS messages that will be delivered by Ozeki SMS Gateway.

FTP Install
Setup this FTP to SMS Extended service if you would like to send SMS messages stored on FTP server or save incoming SMS messages to FTP server. The outgoing SMS messages are in the outgoing directory while it uploads the incoming SMS messages to the server's incoming director. You can choose FTP, FTP Over SSL/TLS, and SSH FTP protocols.

Click install

**Figure 3** - Install SQL Messageing User



Install user/application

SQL messaging

MS SQL Server Install
Setup two database tables in your database server: ozekimessageout and ozekimessagein. Ozekimessageout will be use for sending, the other will be used for receiving. Outgoing messages will be automatically picked up from ozekimessageout and will be delivered. Incoming messages will be inserted into ozekimessagein. This allows you to do messaing from your database system.

MS SQL Express Install
Setup two database tables in your database server: ozekimessageout and ozekimessagein. Ozekimessageout will be use for sending, the other will be used for receiving. Outgoing messages will be automatically picked up from ozekimessageout and will be delivered. Incoming messages will be inserted into ozekimessagein. This allows you to do messaing from your database system.

MS Access Install
Setup two database tables in your database server: ozekimessageout and ozekimessagein. Ozekimessageout will be use for sending, the other will be used for receiving. Outgoing messages will be automatically picked up from ozekimessageout and will be delivered. Incoming messages will be inserted into ozekimessagein. This allows you to do messaing from your database system.

Oracle Install
Setup two database tables in your database server: ozekimessageout and ozekimessagein. Ozekimessageout will be use for sending, the other will be used for receiving. Outgoing messages will be automatically picked up from ozekimessageout and will be delivered. Incoming messages will be inserted into ozekimessagein. This allows you to do messaing from your database system.

MySQL Install
Setup two database tables in your database server: ozekimessageout and ozekimessagein. Ozekimessageout will be use for sending, the other will be used for receiving. Outgoing messages will be automatically picked up from ozekimessageout and will be delivered. Incoming messages will be inserted into ozekimessagein. This allows you to do messaing from your database system.

PostgreSQL Install
Setup two database tables in your database server: ozekimessageout and ozekimessagein. Ozekimessageout will be use for sending, the other will be used for receiving. Outgoing messages will be automatically picked up from ozekimessageout and will be delivered. Incoming messages will be inserted into ozekimessagein. This allows you to do messaing from your database system.

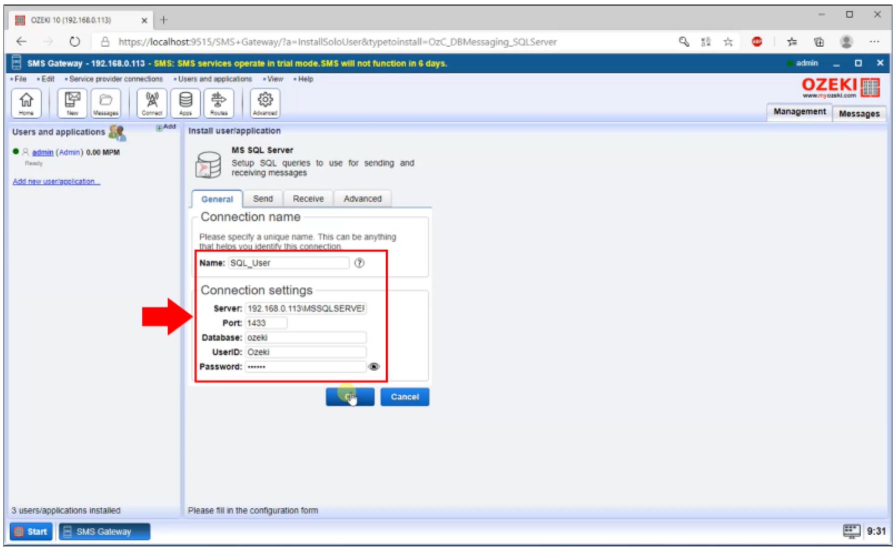
ODBC Install
Setup two database tables in your database server: ozekimessageout and ozekimessagein. Ozekimessageout will be use for sending, the other will be used for receiving. Outgoing messages will be automatically picked up from ozekimessageout and will be delivered. Incoming messages will be inserted into ozekimessagein. This allows you to do messaing from your database system.

OLEDB Install
Setup two database tables in your database server: ozekimessageout and ozekimessagein. Ozekimessageout will be use for sending, the other will be used for receiving. Outgoing messages will be automatically picked up from ozekimessageout and will be delivered. Incoming messages will be inserted into ozekimessagein. This allows you to do messaing from your database system.

SQLAnywhere Install
Setup two database tables in your database server: ozekimessageout and ozekimessagein. Ozekimessageout will be use for sending, the other will be used for receiving. Outgoing messages will be automatically picked up from ozekimessageout and will be delivered. Incoming messages will be inserted into ozekimessagein. This allows you to do messaing from your database system.

Please select the type of database you wish to create.

**Figure 4** - Install MS SQL Connection



Install user/application

MS SQL Server
Setup SQL queries to use for sending and receiving messages

General | Send | Receive | Advanced

Connection name

Please specify a unique name. This can be anything that helps you identify this connection.

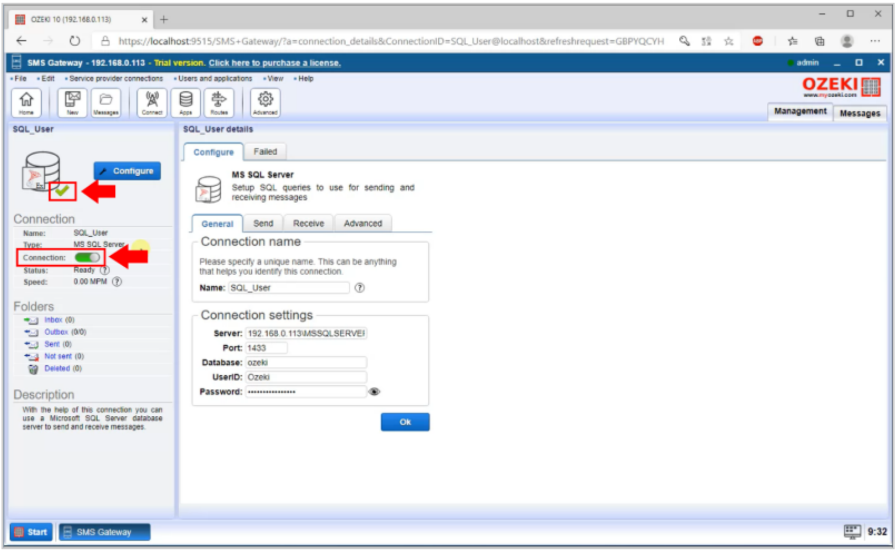
Name: SQL_User

Connection settings

Server: 192.168.0.113\MSSQLSERVER
Port: 1433
Database: ozeki
UserID: Ozeki
Password: ••••••

Please fill in the configuration form

**Figure 5** - Define the MS SQL database connection details



OZEKI 10 (192.168.0.113)
https://localhost:9515/SMS+Gateway/?a=connection_details&ConnectionID=SQL_User@localhost&refreshrequest=GBPYQCYH
SMS Gateway - 192.168.0.113 - Trial version. Click here to purchase a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
OZEKI
www.myozeki.com
Management
Messages
Configure
Failed
General
Send
Receive
Advanced
Ok
Start
SMS Gateway
9:32

SQL_User

Connection

Name: SQL_User
Type: MS SQL Server
Connection: ●
Status: Ready
Speed: 0.00 MPM

Folders

Inbox (0)
Outbox (0/0)
Sent (0)
Not sent (0)
Deleted (0)

Description

With the help of this connection you can use a Microsoft SQL Server database server to send and receive messages.

SQL_User details

MS SQL Server
Setup SQL queries to use for sending and receiving messages

Connection name

Please specify a unique name. This can be anything that helps you identify this connection.

Name: SQL_User

Connection settings

Server: 192.168.0.113\MSSQLSERVEI
Port: 1433
Database: ozeki
UserID: Ozeki
Password: ••••••••••••••••

**Figure 6 -** Enable MS SQL connection

# Send a test SMS from MSSQL

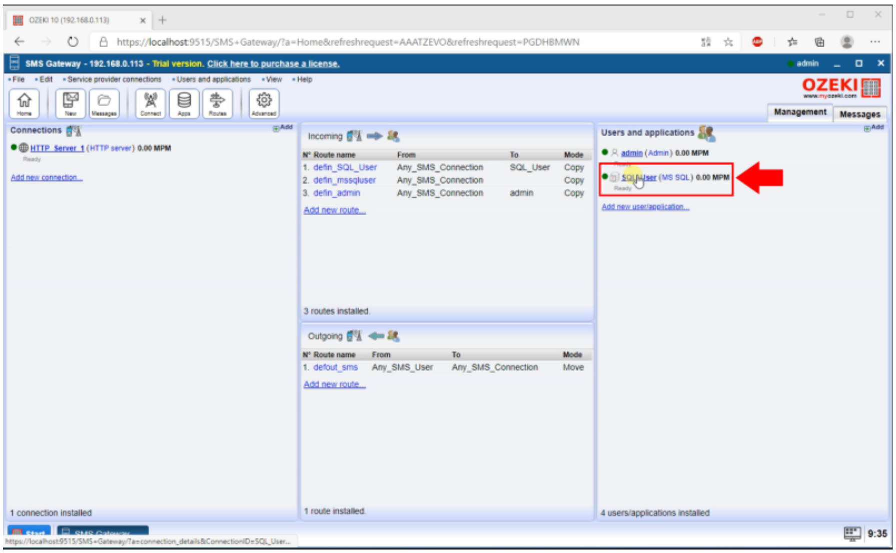
***This guide explains how you can send an SMS message from your MSSQL database server. The procedure is simple: you need to insert a record into the outgoin database table. The guide will give you the INSERT statment you should use. After the record is inserted, the Ozeki SMS Gateway will update it multiple times to give you information about the status of the message submission procedure.***

**Video content**

1. Insert message into database
2. Send test message

---

Look at the upcoming screenshots to thoroughly examine the final stage, which is SMS sending.



SMS Gateway - 192.168.0.113 - Trial version. Click here to purchase a license.
Connections
HTTP_Server_1 (HTTP server) 0.00 MPM
Add new connection...
1 connection installed
Management
Messages

Incoming

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defin_SQL_User | Any_SMS_Connection | SQL_User | Copy |
| 2. | defin_mssqluser | Any_SMS_Connection | | Copy |
| 3. | defin_admin | Any_SMS_Connection | admin | Copy |

Add new route...

3 routes installed.

Outgoing

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move |

Add new route...

1 route installed.

Users and applications

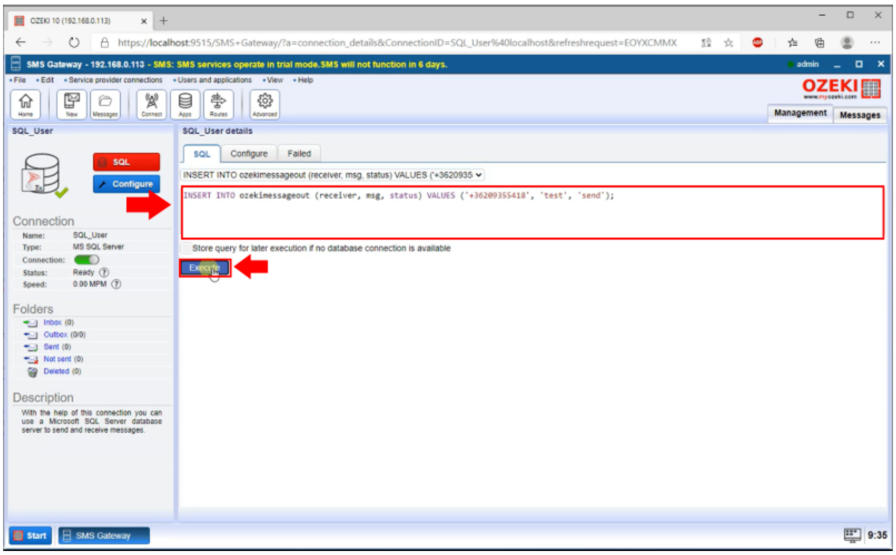
admin (Admin) 0.00 MPM

SQL_User (MS SQL) 0.00 MPM

Add new user/application...

4 users/applications installed

**Figure 1** - Open Database user



Connection

Name: SQL_User
Type: MS SQL Server
Status: Ready
Speed: 0.00 MPM

Description

With the help of this connection you can use a Microsoft SQL Server database server to send and receive messages.

```
INSERT INTO ozekimessageout (receiver, msg, status) VALUES ('+36209355418', 'test', 'send');
```

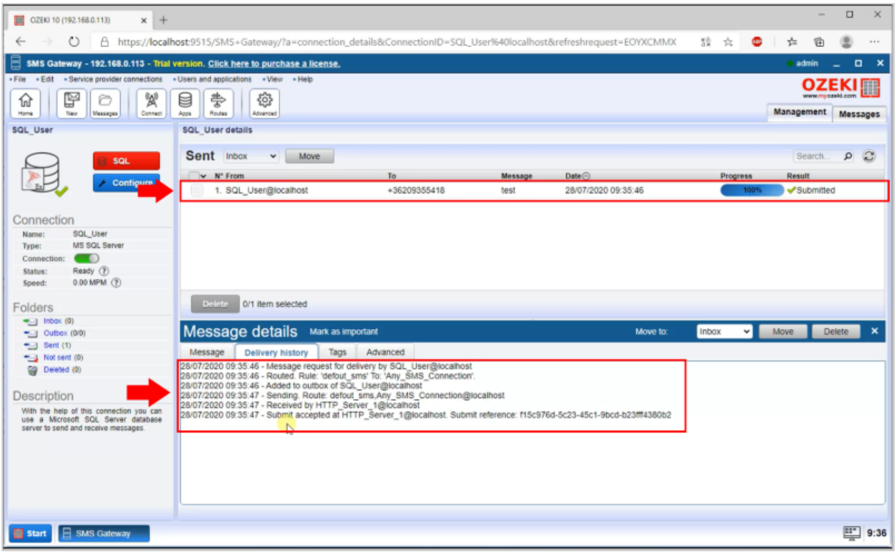
□ Store query for later execution if no database connection is available

**Figure 2** - Insert message to the database table

**INSERT message record** (example):

```
USE ozeki
INSERT INTO ozekimessageout (receiver,msg,status) values ("+44111223465","Helloworld","Send");
GO
```

**Figure 3** - Copy INSERT statement



SQL_User

Connection

Name: SQL_User
Type: MS SQL Server
Status: Ready
Speed: 0.00 MPM

Folders

- Inbox (0)
- Outbox (0/0)
- Sent (1)
- Not sent (0)
- Deleted (0)

Description

With the help of this connection you can use a Microsoft SQL Server database server to send and receive messages.

SQL_User details

Sent

| N° | From | To | Message | Date | Progress | Result |
|---|---|---|---|---|---|---|
| 1. | SQL_User@localhost | +36209355418 | test | 28/07/2020 09:35:46 | 100% | ✓Submitted |

0/1 item selected

Message details

Move to: Inbox

```
28/07/2020 09:35:46 - Message request for delivery by SQL_User@localhost
28/07/2020 09:35:46 - Routed. Rule: 'defout_sms' To: 'Any_SMS_Connection'.
28/07/2020 09:35:46 - Added to outbox of SQL_User@localhost
28/07/2020 09:35:47 - Sending. Route: defout_sms,Any_SMS_Connection@localhost
28/07/2020 09:35:47 - Received by HTTP_Server_1@localhost
28/07/2020 09:35:47 - Submit accepted at HTTP_Server_1@localhost. Submit reference: f15c976d-5c23-45c1-9bcd-b23fff4380b2
```

**Figure 3 -** The Database User's sent folder shows that your message has been sent by Ozeki SMS Gateway

# MS SQL connection troubleshooting

***This guide is intended to give you hints, that might help you fix a broken Microsoft SQL Server or Microsoft SQL Express connection. In general a reinstallation of the database server usually fixes the problem, but in most cases reinstallation is not an option. Here are the things to check if you cannot connect to your SQL server:***

1. Please use the SQL Server configuration manager to check if both TCPIP and Named Pipes are enabled for the **database server instance**.

2. Make sure SQLBrowser service is up and running.

3. Check if the TCPIP and Named Pipe protocols are enabled to be used in the **SQL Client**.

4. Open the error log to find the SQL database instance and make sure there are no ERROR messages for TCPIP and Named Pipes. The logs can be found at: C:\Program Files\Microsoft SQL Server\MSSQL.x\MSSQL\LOG

5.You should also find the port number, and check if the port number configured in the server and client are identical.

6. Please try different protocols by testing them with 'sqlcmd'. One of the following 'sqlcmd' commands should work.

```
sqlcmd -S np:\\127.0.0.1\pipe\MSSQL$instancename\sql\query
sqlcmd -S tcp:servername\instancename
sqlcmd -S tcp:servername\instancename,portnumber
```

Tip: Check that you use the correct servername, instancename and portnumber for your database server.

6. If it is still not possible to connect, please change this registry key:

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Microsoft SQL Server\MSSQL.1\MSSQLServer

LoginMode=2

# SMS from/to Oracle

***The connection between Oracle Database and Ozeki SMS Gateway is a perfect solution for SMS messaging. Ozeki SMS Gateway's Database User makes sure that incoming SMS messages can be inserted while outgoing SMS messages can be selected from database tables. Incoming and outgoing messages are handled in separate tables. The solution uses the Ozeki SMS Gateway software installed on your PC.This solution is divided into 4 sections:***

Install Oracle
Create the database tables in Oracle
Configure the Oracle database connection in Ozeki
Send a test SMS from Oracle

This page extends the general guide to SMS from/to Database.

You should also make sure that Ozeki SMS Gateway is installed. You should also check if Oracle Database is installed. If Oracle is not installed on your machine, please download the server and client from the oracle website. You will see how to store outgoing and incoming SMS messages in your Oracle Database's tables.

You can choose from a wide variety of options to send and receive SMS messages. You can send messages through a wireless mobile link, by using an Android mobile phone or an SMS modem attached to your computer. You can also send SMS messages over the Internet using in IP SMS service. In this case you will likely setup an SMPP, CIMD2 or UCP/EMI service provider connection.

To use either of these connection methods, you need to setup a Database link on Ozeki SMS Gateway. For Oracle connections, you need to configure this link to connect to your Oracle Database server. You will need to use a connection string to specify the connection details.

Each Database Connection needs it's own Database User.
The Database User belonging to an Oracle Database has the following connection string format:

| | |
|---|---|
| Connection string type: | **Oracle** |
| Connection string: | **Data Source=127.0.0.1; User ID=username; Password=pwd123;** |

(Other 'Connection strings for Oracle')

In the connection string the 127.0.0.1 IP address should be replaced with the IP address of the machine where Oracle Database server is running and do not forget to use the login credentials belonging to your Oracle Server's account.

By including 'Unicode=True' to the connection string, you can use unicode characters like NVARCHAR2. Make sure that NLS_CHARACTERSET and NLS_LANGUAGE parameters are set to your language. You can also try to set HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\HOMEID\NLS_LANG registry key to UTF8.

Now the connection should be configured between Ozeki SMS Gateway's Database User and your Oracle Server, so it is time to create two database tables both for incoming and outgoing message by creating a proper database table structure:

**<u>Oracle SQL statements:</u>**

### CREATE TABLESPACE:

```
CREATE TABLESPACE ozeki
DATAFILE 'C:\oraclexe\app\oracle\oradata\XE\ozeki.dbf'
SIZE 40M autoextend on;
```

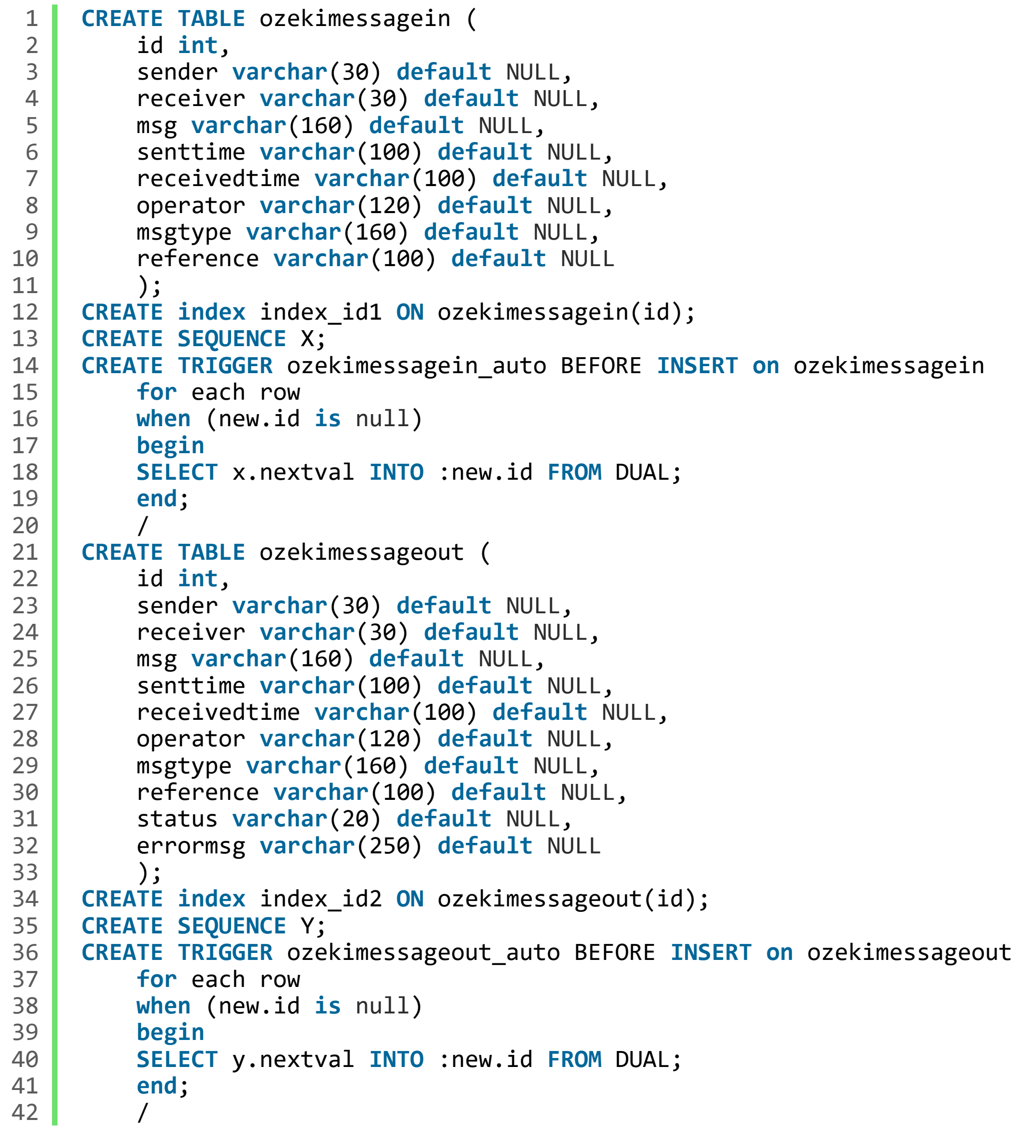
Figure 1 - Create tablespace 'ozeki'

### CREATE USER:

```
CREATE USER ozeki
IDENTIFIED BY qwe123
DEFAULT TABLESPACE ozeki;

GRANT DBA TO ozeki;
```

Figure 2 - Create user 'ozeki' and grant access to tablespace

**CREATE TABLE:**



```
CREATE TABLE ozekimessagein (
    id int,
    sender varchar(30) default NULL,
    receiver varchar(30) default NULL,
    msg varchar(160) default NULL,
    senttime varchar(100) default NULL,
    receivedtime varchar(100) default NULL,
    operator varchar(120) default NULL,
    msgtype varchar(160) default NULL,
    reference varchar(100) default NULL
    );
CREATE index index_id1 ON ozekimessagein(id);
CREATE SEQUENCE X;
CREATE TRIGGER ozekimessagein_auto BEFORE INSERT on ozekimessagein
    for each row
    when (new.id is null)
    begin
    SELECT x.nextval INTO :new.id FROM DUAL;
    end;
    /
CREATE TABLE ozekimessageout (
    id int,
    sender varchar(30) default NULL,
    receiver varchar(30) default NULL,
    msg varchar(160) default NULL,
    senttime varchar(100) default NULL,
    receivedtime varchar(100) default NULL,
    operator varchar(120) default NULL,
    msgtype varchar(160) default NULL,
    reference varchar(100) default NULL,
    status varchar(20) default NULL,
    errormsg varchar(250) default NULL
    );
CREATE index index_id2 ON ozekimessageout(id);
CREATE SEQUENCE Y;
CREATE TRIGGER ozekimessageout_auto BEFORE INSERT on ozekimessageout
    for each row
    when (new.id is null)
    begin
    SELECT y.nextval INTO :new.id FROM DUAL;
    end;
    /
```

Figure 3 - Create table structure

The '**id**' field should be maintained to support faster SQL updates.
The **size of the message can be set higher** than 160 character if it is necessary.
The '**msgtype**' field can also be changed.

# Connection strings for Oracle

***You can use these alternative connection strings to connect to your Oracle Database. These are useful if you have a connection problem or if you use another driver for example ODBC or OLE DB.***

## .NET Oracle connection providers

**Standard Security:**

| | |
|---|---|
| Connection string type: | **Oracle** |
| Connection string: | **Data Source=MyOracleDB; User Id=myUsername;Password=myPassword;** |

**Trusted Connection:**

- OS Authenticated connection ('Id=/'):

| | |
|---|---|
| Connection string type: | **Oracle** |
| Connection string: | **Data Source=MyOracleDB;User Id=/;Password=;** |

- OS Authenticated connection using OSAuthent:

| | |
|---|---|
| Connection string type: | **Oracle** |
| Connection string: | **Data Source=MyOracleDB;OSAuthent=1;** |

## ODBC Driver for Oracle

Using the current ODBC driver from Microsoft:

| | |
|---|---|
| Connection string type: | **ODBC** |
| Connection string: | **Driver={Microsoft ODBC for Oracle};Server=OracleServer.world; Uid=MyUsername;Pwd=MyPassword;** |

Using the older ODBC driver from Microsoft:

| | |
|---|---|
| Connection string type: | **ODBC** |
| Connection string: | **Driver={Microsoft ODBC Driver for Oracle};ConnectString=OracleServer.world;Uid=myUsername;Pwd=myPassword;** |

## OLE DB Provider for Oracle

**Standard Security:**

| | |
|---|---|
| Connection string type: | **OLE DB** |
| Connection string: | **Provider=OraOLEDB.Oracle;Data Source=MyOracleDB; User Id=myUsername;Password=myPassword;** |

**Trusted Connection:**

- OS Authenticated connection ('Id=/'):

| | |
|---|---|
| Connection string type: | **OLE DB** |
| Connection string: | **Provider=OraOLEDB.Oracle;Data Source=MyOracleDB;User Id=/;Password=;** |

- OS Authenticated connection using OSAuthent:

| | |
|---|---|
| Connection string type: | **OLE DB** |
| Connection string: | **Provider=OraOLEDB.Oracle;Data Source=MyOracleDB;OSAuthent=1;** |

Info: Please provide the 'Data Source=' a Net8 name which uses a standard naming method like Local Naming or Oracle Names. Local Naming is the alias in the tnsnames.ora file. The Oracle Name is the Net8 Service name.

# How to send SMS from Oracle

***Here you can see how to send SMS text messages from Oracle Express with your Ozeki SMS Gateway, which you have probably already downloaded and installed. This tutorial shows you a solution on how to configure Ozeki SMS Gateway to detect and send outgoing messages from your database to any mobile phone. You can find a detailed configuration instruction on this page with screenshots that will help you.***

Ozeki SMS Gateway is a powerful SMS gateway software that allows two-way SMS message communication from mobile phones to PC and vice versa. The messages can be stored on a database, such as an Oracle Server.

In this tutorial, you will create an 'ozekimessagein' and 'ozekimessageout' table, but you could give them any name you would like. The only thing is that you need to keep in mind the table names. Ozeki SMS gateway will use SQL queries on the 'ozekimessageout' table to send message.

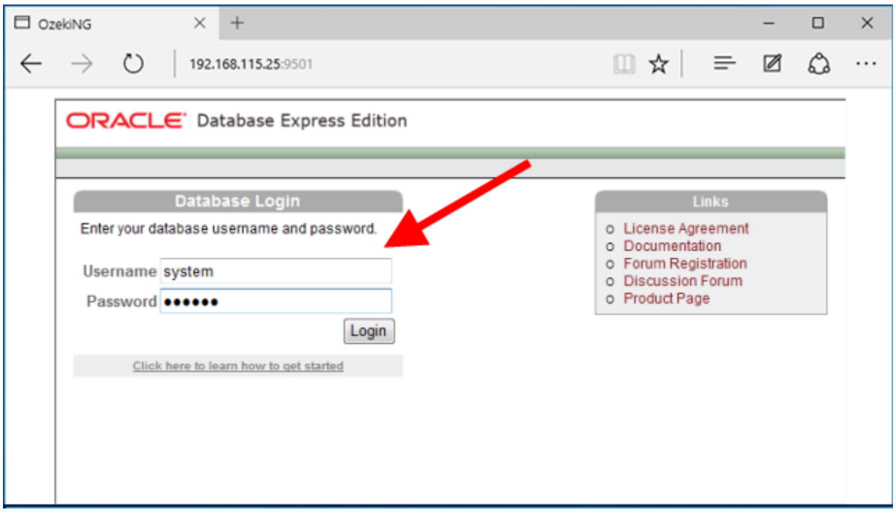
**You can connect Ozeki SMS Gateway to the GSM network in two way:**

1. A hardware solution: Mobile phone or GSM modem connection to PC
2. A software solution: IP SMS Service Provider over the internet (SMPP, CIMD, UCP/EMI)

Here you can read how to configure Ozeki SMS Gateway with browser GUI.

## Step 1 - Configure the Oracle database

The first step of this guide is to configure an Oracle database. For that, first, you need to install Oracle to your computer and log into the server's web interface (Figure 1).



Enter your database username and password.

Username system

Password ••••••

Figure 1 - Log into Oracle

In the main menu of the Oracle Database Express application, you need to select the SQL option the configure the Oracle database, so just click on its icon as you can see it in Figure 2.

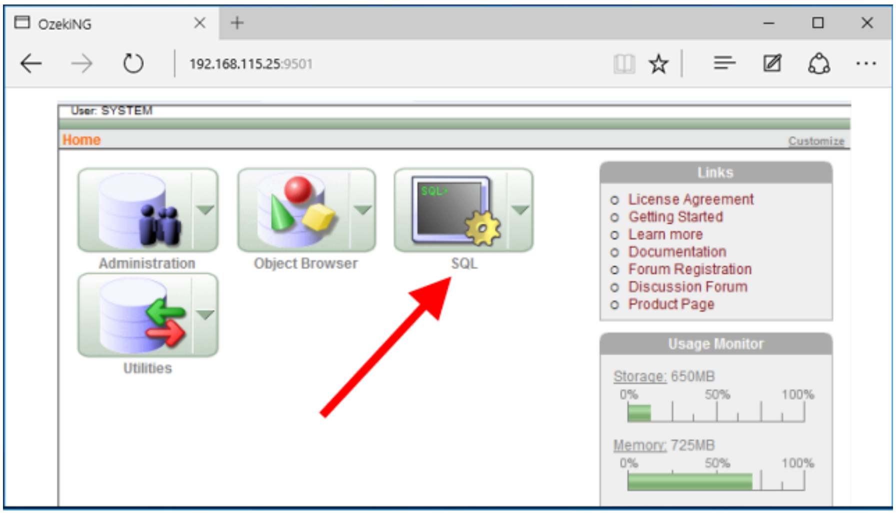


Figure 2 - Click on 'SQL'

Next, you need to click on the 'SQL Commands' as Figure 3 demonstrates it to be able to execute SQL commands in your Oracle database.

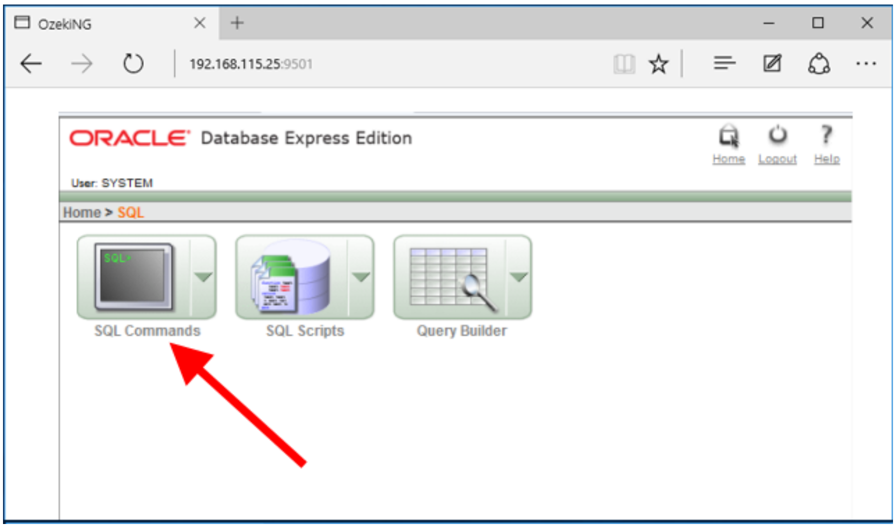
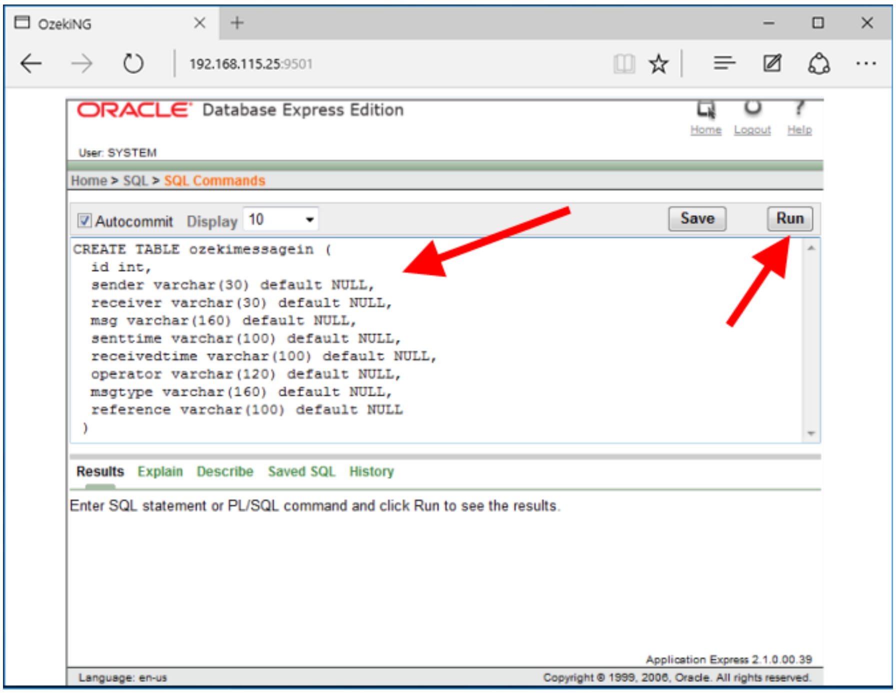


Figure 3 - Click on 'SQL Commands'

Please copy-paste the following SQL statements to create the table layouts (Figure 4-11).

```
CREATE TABLE ozekimessagein (
    id int,
    sender varchar(30) default NULL,
    receiver varchar(30) default NULL,
    msg varchar(160) default NULL,
    senttime varchar(100) default NULL,
    receivedtime varchar(100) default NULL,
    operator varchar(120) default NULL,
    msgtype varchar(160) default NULL,
    reference varchar(100) default NULL
)
```

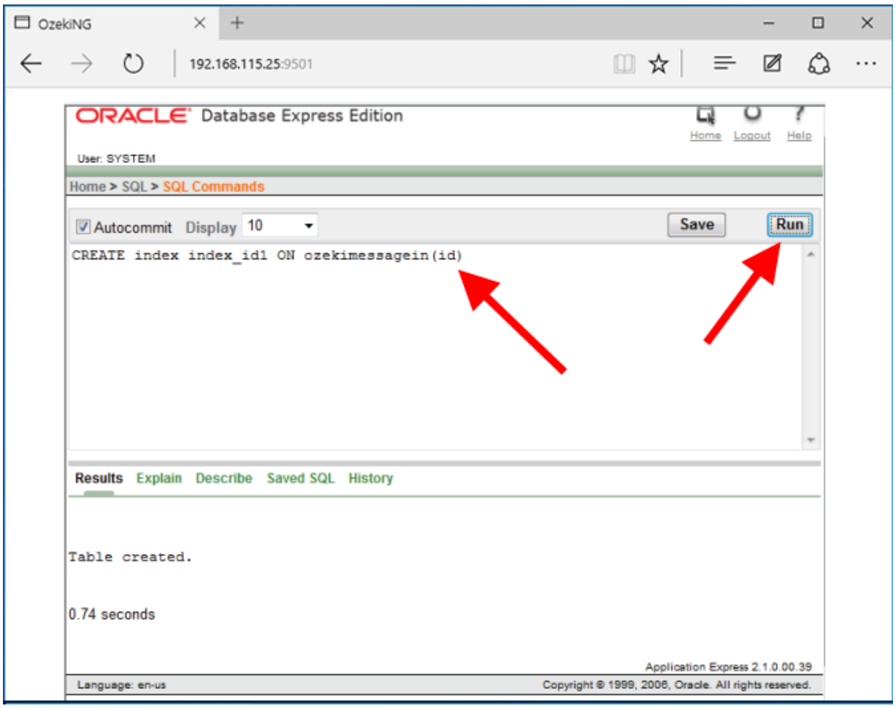

OzekiNG
192.168.115.25:9501
ORACLE Database Express Edition
Home Logout Help
User: SYSTEM
Home > SQL > SQL Commands
☑ Autocommit Display 10
Save
Run
Results Explain Describe Saved SQL History
Application Express 2.1.0.00.39
Language: en-us
Copyright © 1999, 2006, Oracle. All rights reserved.

```
CREATE TABLE ozekimessagein (
  id int,
  sender varchar(30) default NULL,
  receiver varchar(30) default NULL,
  msg varchar(160) default NULL,
  senttime varchar(100) default NULL,
  receivedtime varchar(100) default NULL,
  operator varchar(120) default NULL,
  msgtype varchar(160) default NULL,
  reference varchar(100) default NULL
 )
```

Enter SQL statement or PL/SQL command and click Run to see the results.

Figure 4 - Copy-paste CREATE TABLE statement

```
CREATE INDEX index_id1 ON ozekimessagein(id)
```


OzekiNG
192.168.115.25:9501
ORACLE Database Express Edition
Home Logout Help
User: SYSTEM
Home > SQL > SQL Commands
☑ Autocommit Display 10
Save
Run
Results Explain Describe Saved SQL History
Application Express 2.1.0.00.39
Language: en-us
Copyright © 1999, 2006, Oracle. All rights reserved.

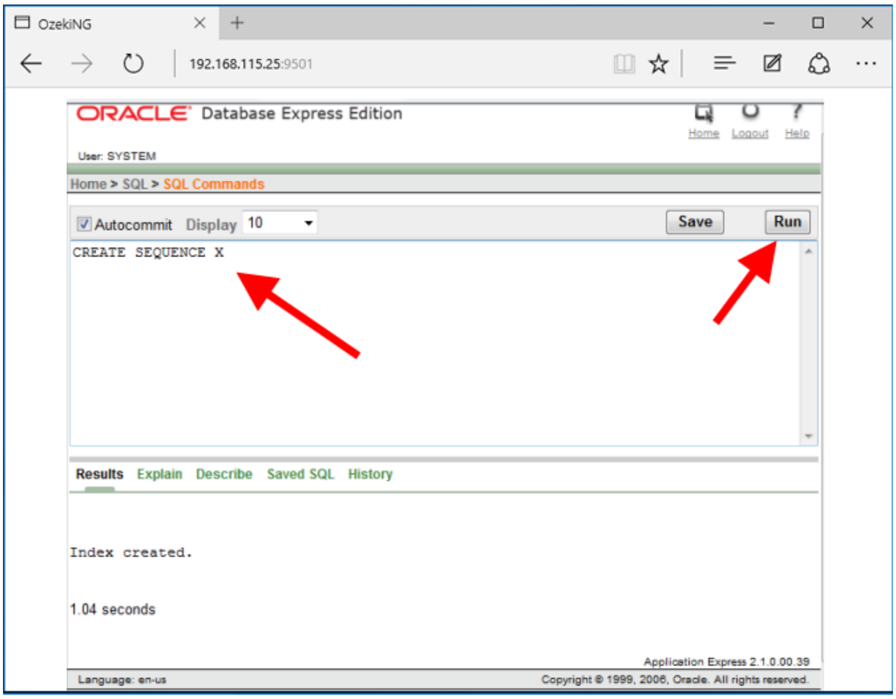
```
CREATE index index_id1 ON ozekimessagein(id)
```

```
Table created.
```

0.74 seconds

Figure 5 - Copy-paste CREATE INDEX statement

```
CREATE SEQUENCE X
```

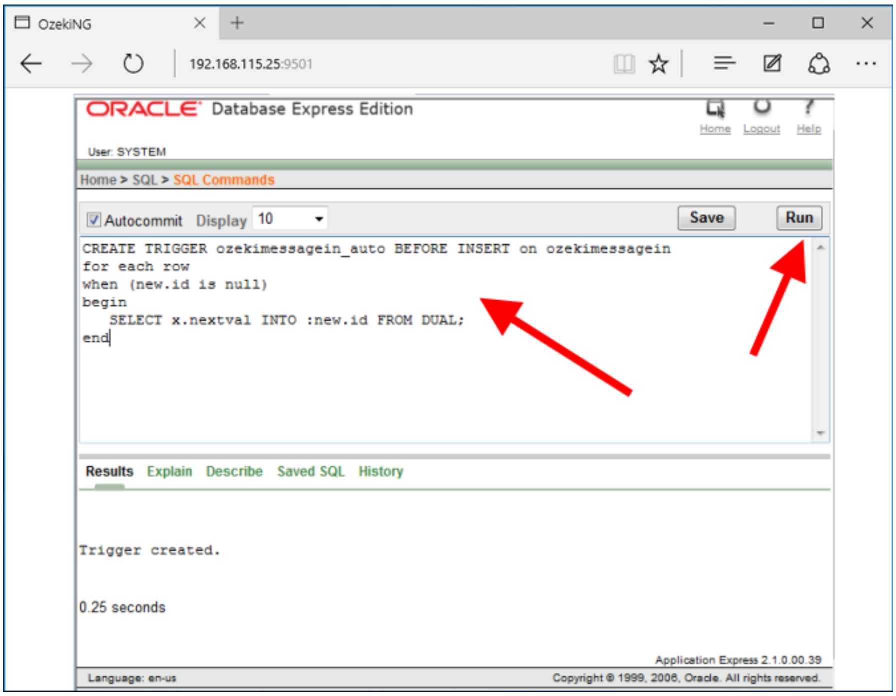


```
CREATE SEQUENCE X
```

```
Index created.

1.04 seconds
```

Figure 6 - Copy-paste CREATE SEQUENCE statement

```
CREATE TRIGGER ozekimessagein_auto BEFORE INSERT on ozekimessagein
    for each row
    when (new.id is null)
    begin
        SELECT x.nextval INTO :new.id FROM DUAL;
    end
```

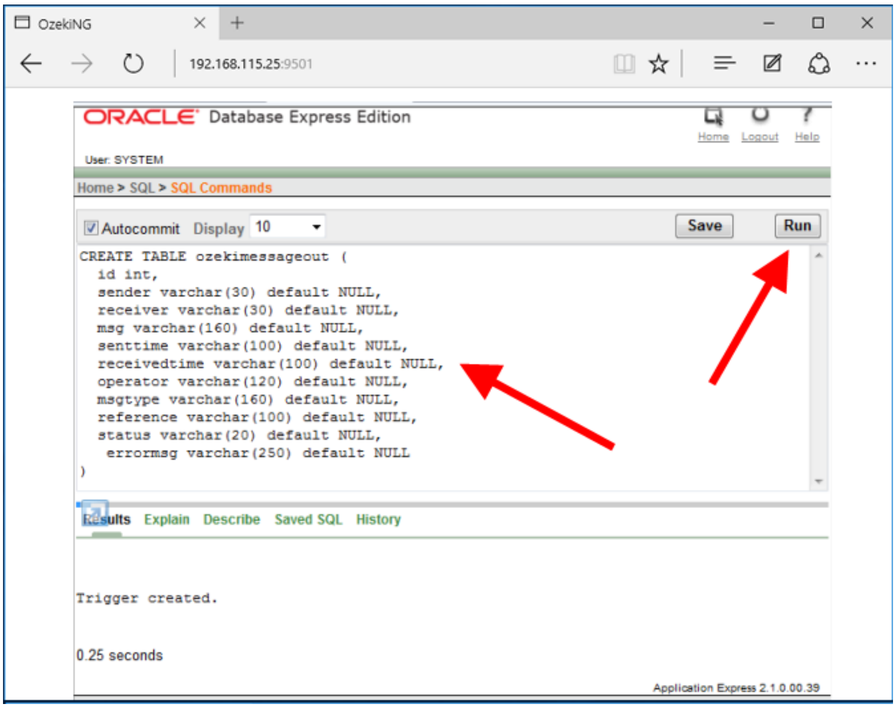


```
CREATE TRIGGER ozekimessagein_auto BEFORE INSERT on ozekimessagein
for each row
when (new.id is null)
begin
   SELECT x.nextval INTO :new.id FROM DUAL;
end
```

```
Trigger created.

0.25 seconds
```

Figure 7 - Copy-paste CREATE TRIGGER statement

```
CREATE TABLE ozekimessageout (
    id int,
    sender varchar(30) default NULL,
    receiver varchar(30) default NULL,
    msg varchar(160) default NULL,
    senttime varchar(100) default NULL,
    receivedtime varchar(100) default NULL,
    operator varchar(120) default NULL,
    msgtype varchar(160) default NULL,
    reference varchar(100) default NULL,
    status varchar(20) default NULL,
    errormsg varchar(250) default NULL
)
```

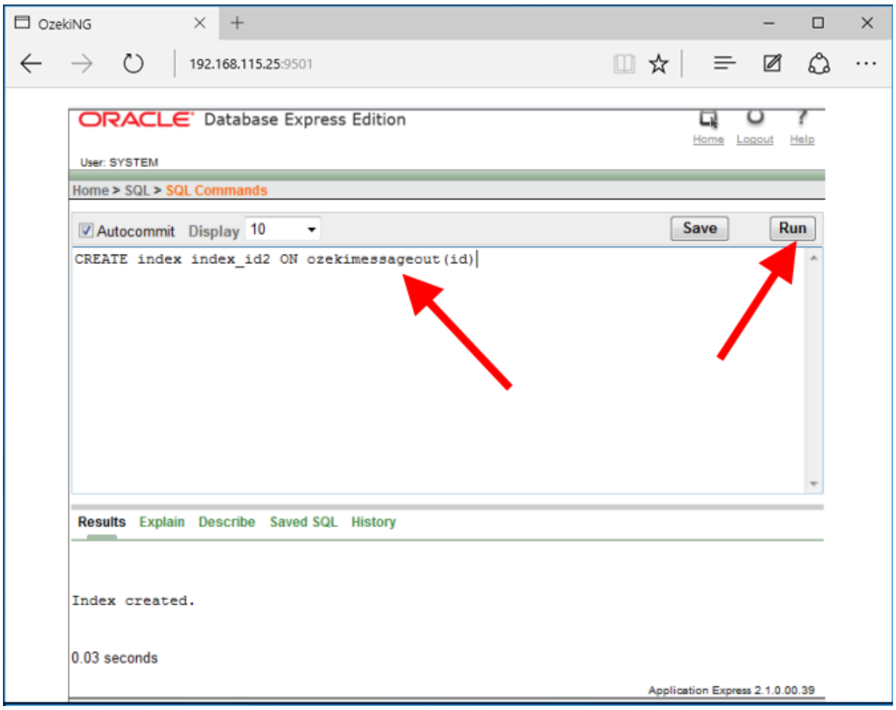


```
CREATE TABLE ozekimessageout (
  id int,
  sender varchar(30) default NULL,
  receiver varchar(30) default NULL,
  msg varchar(160) default NULL,
  senttime varchar(100) default NULL,
  receivedtime varchar(100) default NULL,
  operator varchar(120) default NULL,
  msgtype varchar(160) default NULL,
  reference varchar(100) default NULL,
  status varchar(20) default NULL,
   errormsg varchar(250) default NULL
)
```

```
Trigger created.

0.25 seconds
```

Figure 8 - Copy-paste CREATE TABLE statement

```
CREATE INDEX index_id2 ON ozekimessageout(id)!
```

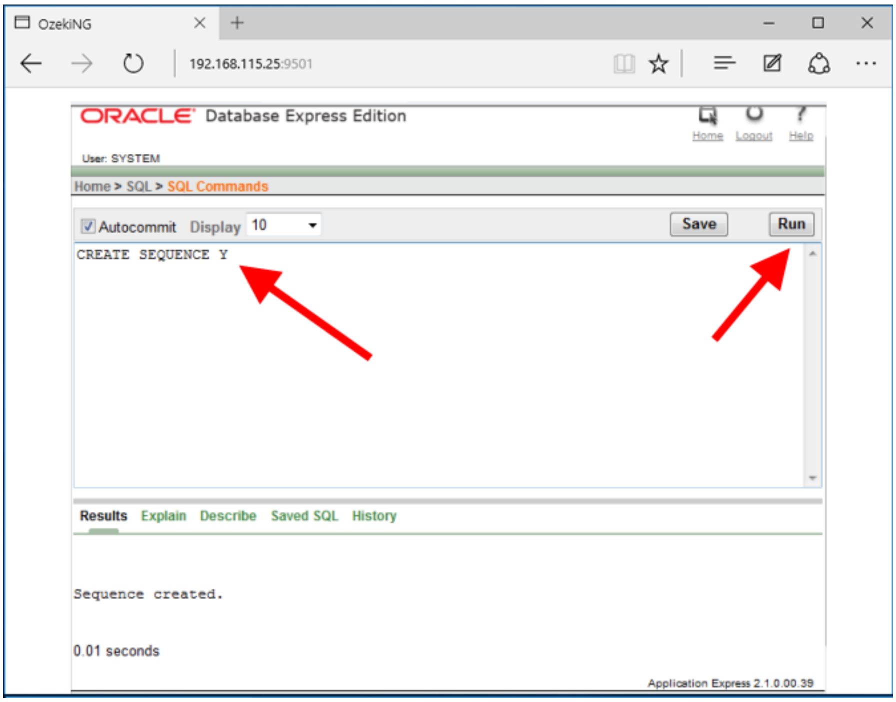


```
CREATE index index_id2 ON ozekimessageout(id)
```

```
Index created.

0.03 seconds
```

Figure 9 - Copy-paste CREATE INDEX statement

```
CREATE SEQUENCE Y
```

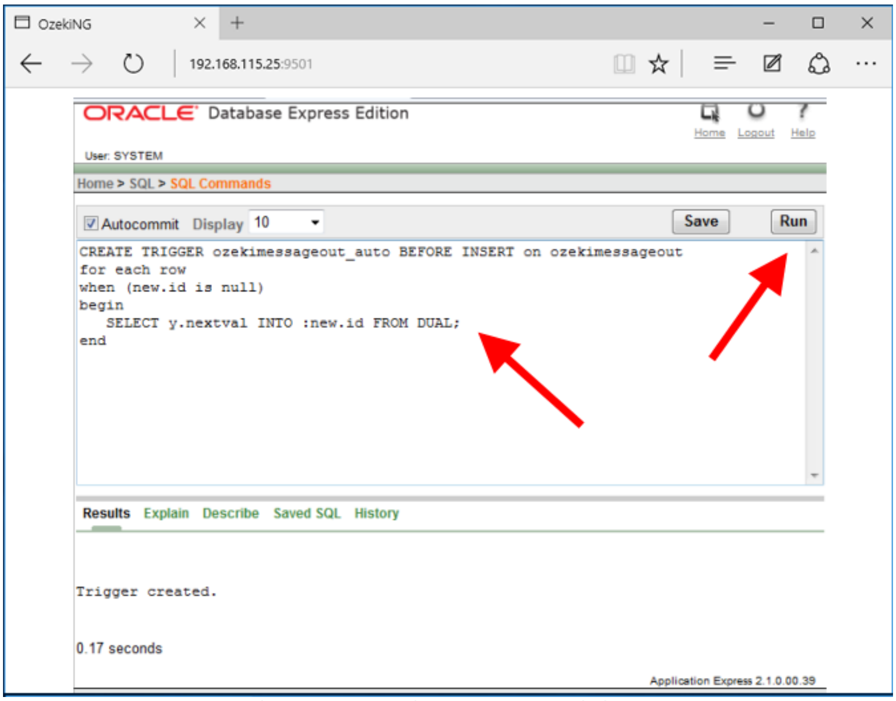


```
CREATE SEQUENCE Y
```

```
Sequence created.

0.01 seconds
```

Figure 10 - Copy-paste CREATE SEQUENCE statement

```sql
CREATE TRIGGER ozekimessageout_auto BEFORE INSERT on ozekimessageout
    for each row
    when (new.id is null)
    begin
        SELECT y.nextval INTO :new.id FROM DUAL;
    end
```



```
CREATE TRIGGER ozekimessageout_auto BEFORE INSERT on ozekimessageout
for each row
when (new.id is null)
begin
   SELECT y.nextval INTO :new.id FROM DUAL;
end
```

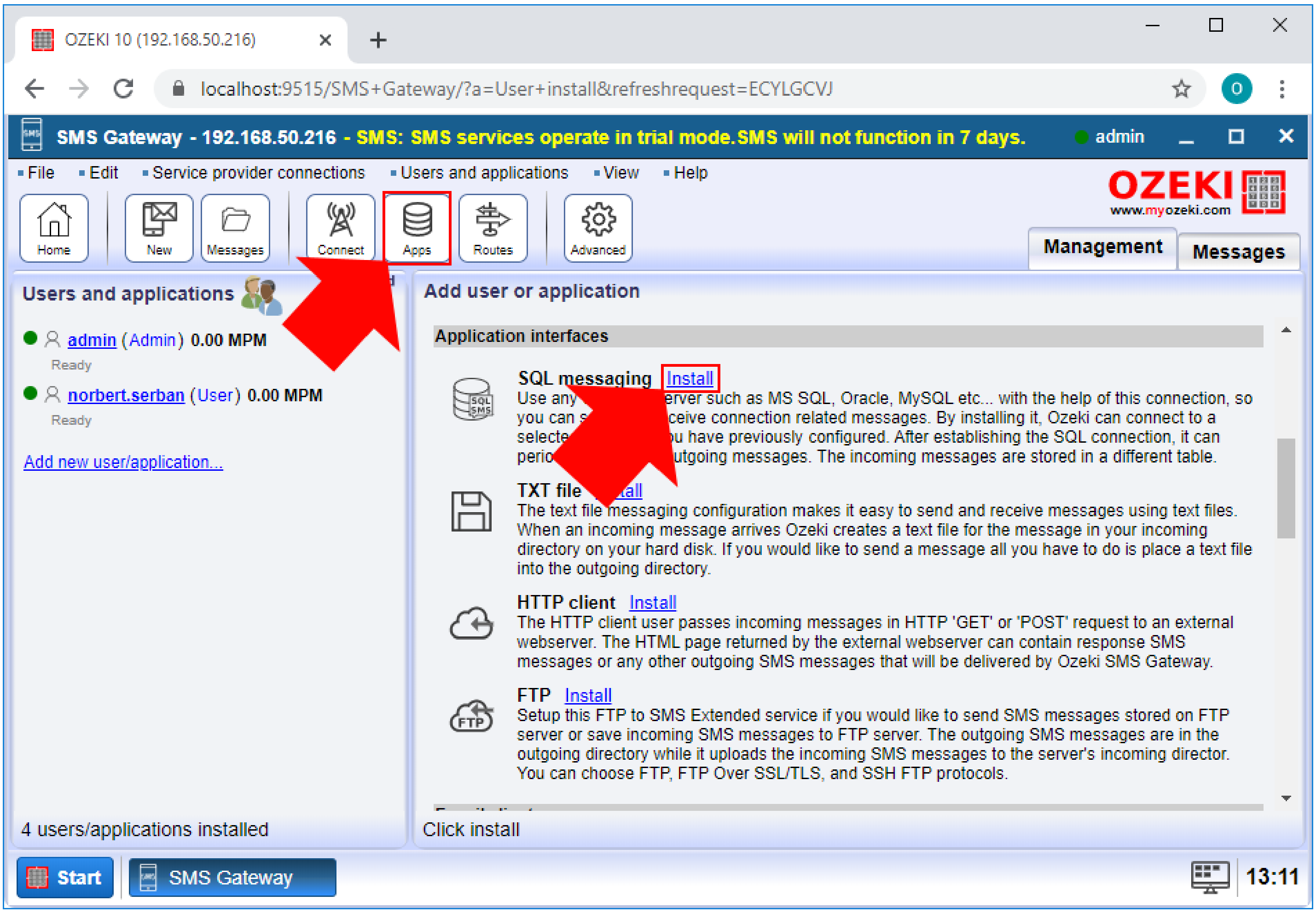
```
Trigger created.

0.17 seconds
```

Figure 11 - Copy-paste CREATE TRIGGER statement

Congratulations! You have finished creating your database structure.

# Step 2 - Create the Oracle database connection

The next step of the guide is to create the Oracle database connection which can connect to the database and handle its tables. For that, open SMS Gateway and first, select the Apps menu from the toolbar. In this menu, you need to scroll down to the 'Application interfaces' section, and here, like in Figure 12, just click on the install button of 'SQL messaging'.


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=User+install&refreshrequest=ECYLGCVJ
SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
OZEKI
www.myozeki.com
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Users and applications
admin (Admin) 0.00 MPM
Ready
norbert.serban (User) 0.00 MPM
Ready
Add new user/application...
4 users/applications installed
Add user or application
Application interfaces
Click install
Start
SMS Gateway
13:11

**SQL messaging** Install
Use any ... erver such as MS SQL, Oracle, MySQL etc... with the help of this connection, so you can s... ceive connection related messages. By installing it, Ozeki can connect to a selecte... u have previously configured. After establishing the SQL connection, it can perio... utgoing messages. The incoming messages are stored in a different table.

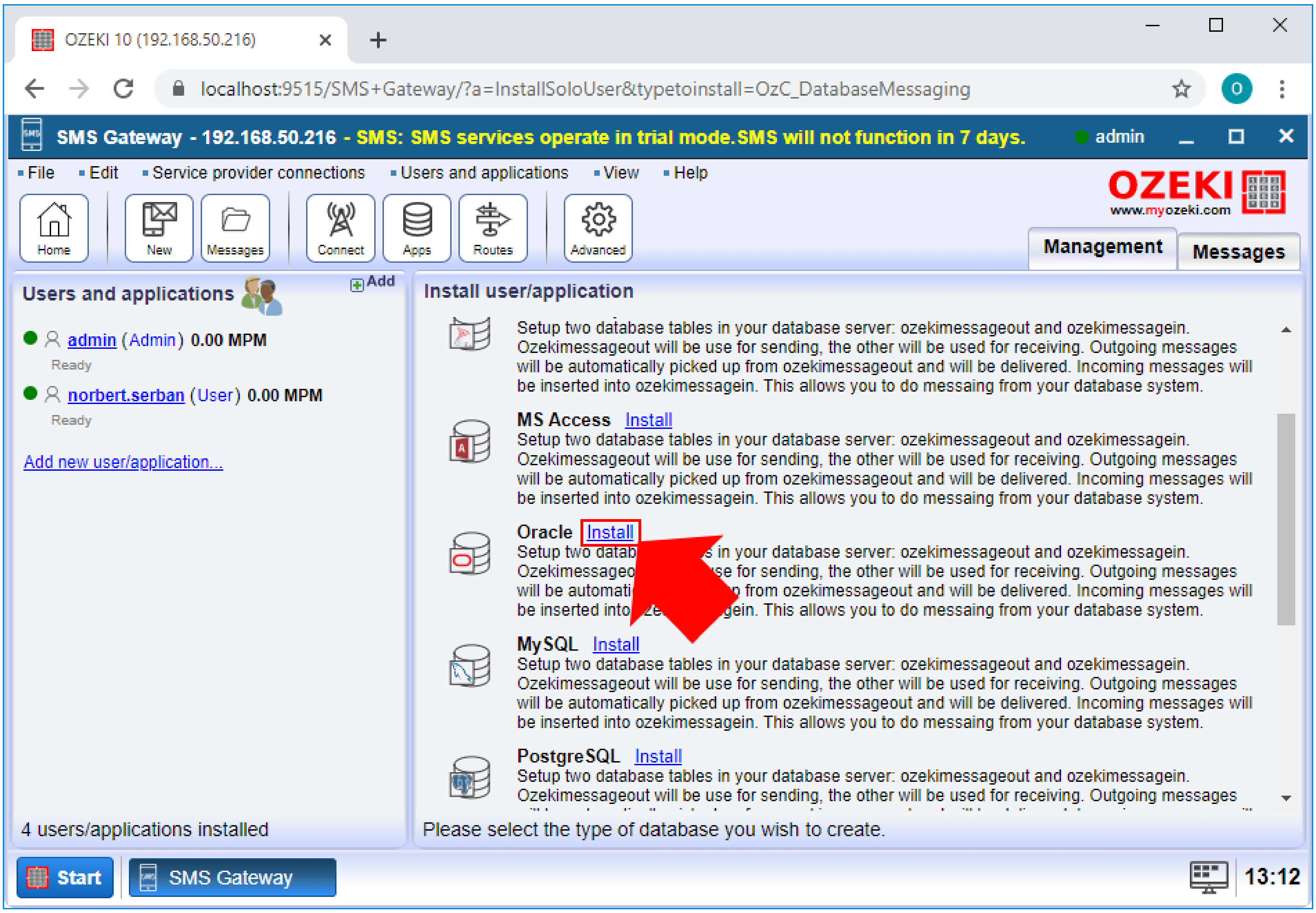
**TXT file** ...tall
The text file messaging configuration makes it easy to send and receive messages using text files. When an incoming message arrives Ozeki creates a text file for the message in your incoming directory on your hard disk. If you would like to send a message all you have to do is place a text file into the outgoing directory.

**HTTP client** Install
The HTTP client user passes incoming messages in HTTP 'GET' or 'POST' request to an external webserver. The HTML page returned by the external webserver can contain response SMS messages or any other outgoing SMS messages that will be delivered by Ozeki SMS Gateway.

**FTP** Install
Setup this FTP to SMS Extended service if you would like to send SMS messages stored on FTP server or save incoming SMS messages to FTP server. The outgoing SMS messages are in the outgoing directory while it uploads the incoming SMS messages to the server's incoming director. You can choose FTP, FTP Over SSL/TLS, and SSH FTP protocols.

Figure 12 - Click 'Add new user or application...'

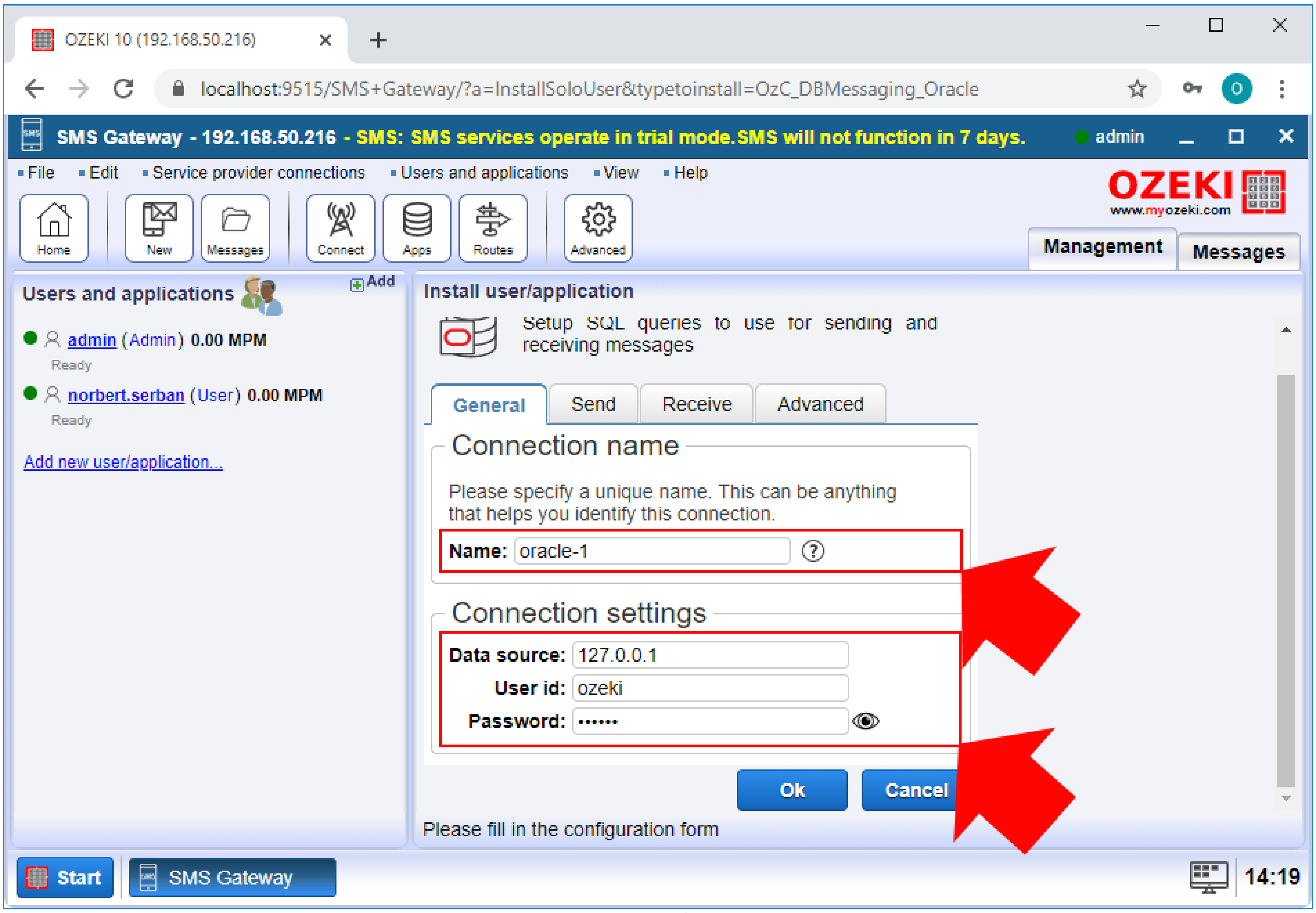
The 'SQL messaging' menu contains all types of database connections available in SMS Gateway. Now, to be able to connect to your Oracle database and its tables, just click on the Install button of the Oracle database connection as you can see it in Figure 13.


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=InstallSoloUser&typetoinstall=OzC_DatabaseMessaging
SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Users and applications
Add
admin (Admin) 0.00 MPM
Ready
norbert.serban (User) 0.00 MPM
Ready
Add new user/application...
4 users/applications installed
Install user/application
MS Access Install
Oracle Install
MySQL Install
PostgreSQL Install
Please select the type of database you wish to create.
Start
SMS Gateway
13:12

Setup two database tables in your database server: ozekimessageout and ozekimessagein. Ozekimessageout will be use for sending, the other will be used for receiving. Outgoing messages will be automatically picked up from ozekimessageout and will be delivered. Incoming messages will be inserted into ozekimessagein. This allows you to do messaing from your database system.

Figure 13 - Click on the 'install' button next to 'Database' user

Next, you need to provide some details to be able to connect to the database successfully. First, just give a name to the connection. After that, as in Figure 14, provide the details of the Oracle database. The 'Data source' id the IP address of the place where you installed the database. If it is on the same computer you can type '127.0.0.1' in this field. The 'User id' and 'Password' fields should represent your Oracle database user account. If you finished, you can just click OK.


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=InstallSoloUser&typetoinstall=OzC_DBMessaging_Oracle
SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Users and applications
Add
admin (Admin) 0.00 MPM
Ready
norbert.serban (User) 0.00 MPM
Ready
Add new user/application...
Install user/application
General
Send
Receive
Advanced
Ok
Cancel
Start
SMS Gateway
14:19

Setup SQL queries to use for sending and receiving messages

Connection name

Please specify a unique name. This can be anything that helps you identify this connection.

Name: oracle-1

Connection settings

Data source: 127.0.0.1

User id: ozeki

Password: ••••••

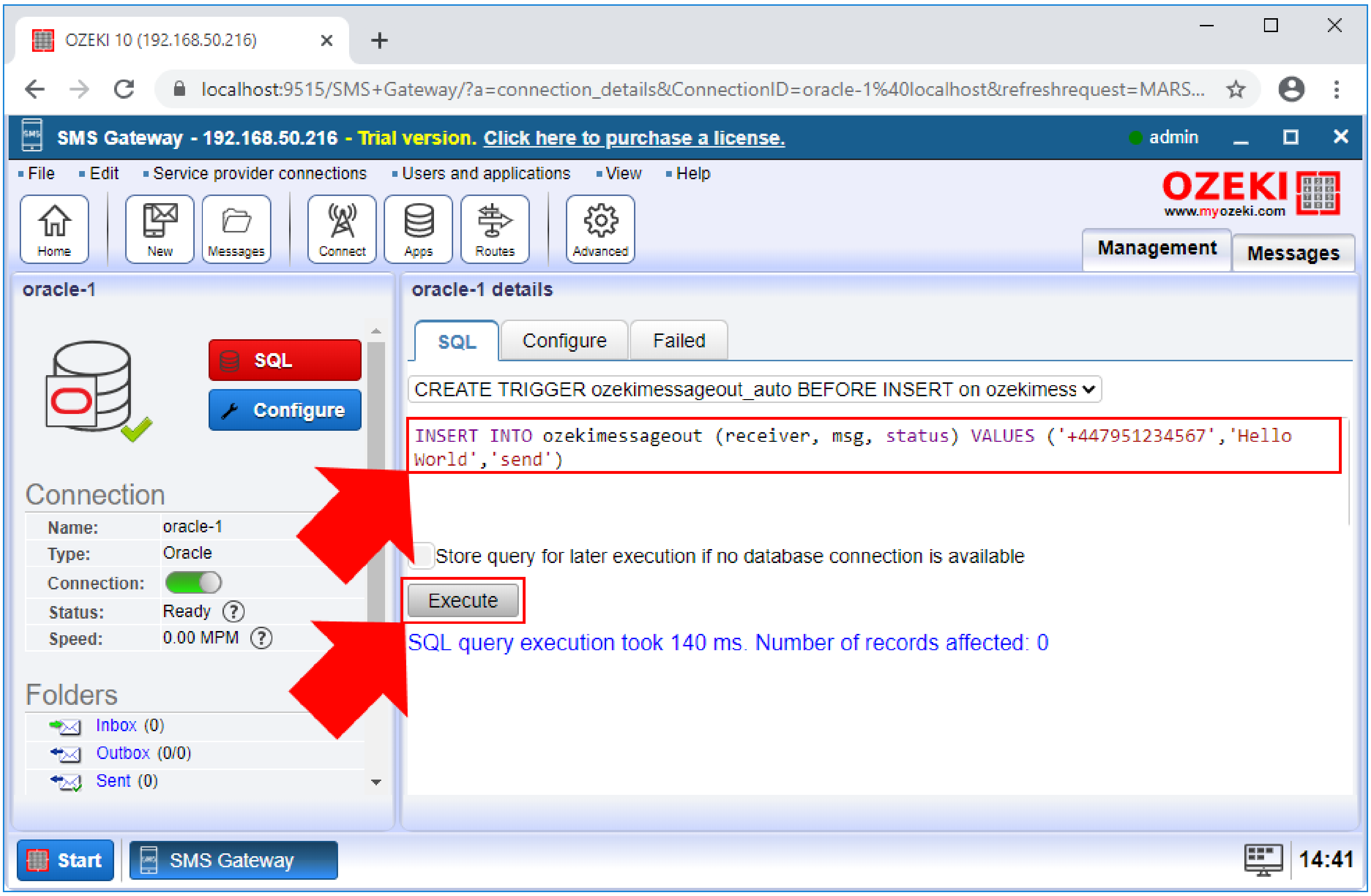
Please fill in the configuration form

Figure 14 - Provide a name for your Database User

# Step 3 - Send a test SMS

To test the database connection and the database itself, just open the created Oracle database connection. Here, you can see the SQL tab (Figure 15) where you can write the SQL commands that you want to execute. So, just copy-paste the SQL command below, write a valid phone number as a value and click 'Execute' to test the connection.

```
INSERT INTO ozekimessageout (receiver, msg, status) VALUES ('+36205460691','Hello World','send'
```


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=connection_details&ConnectionID=oracle-1%40localhost&refreshrequest=MARS...
SMS Gateway - 192.168.50.216 - Trial version. Click here to purchase a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
oracle-1
SQL
Configure
oracle-1 details
SQL
Configure
Failed
Connection
Folders
Inbox (0)
Outbox (0/0)
Sent (0)
Start
SMS Gateway
14:41

Name: oracle-1
Type: Oracle
Connection:
Status: Ready
Speed: 0.00 MPM

CREATE TRIGGER ozekimessageout_auto BEFORE INSERT on ozekimess

```
INSERT INTO ozekimessageout (receiver, msg, status) VALUES ('+447951234567','Hello
World','send')
```

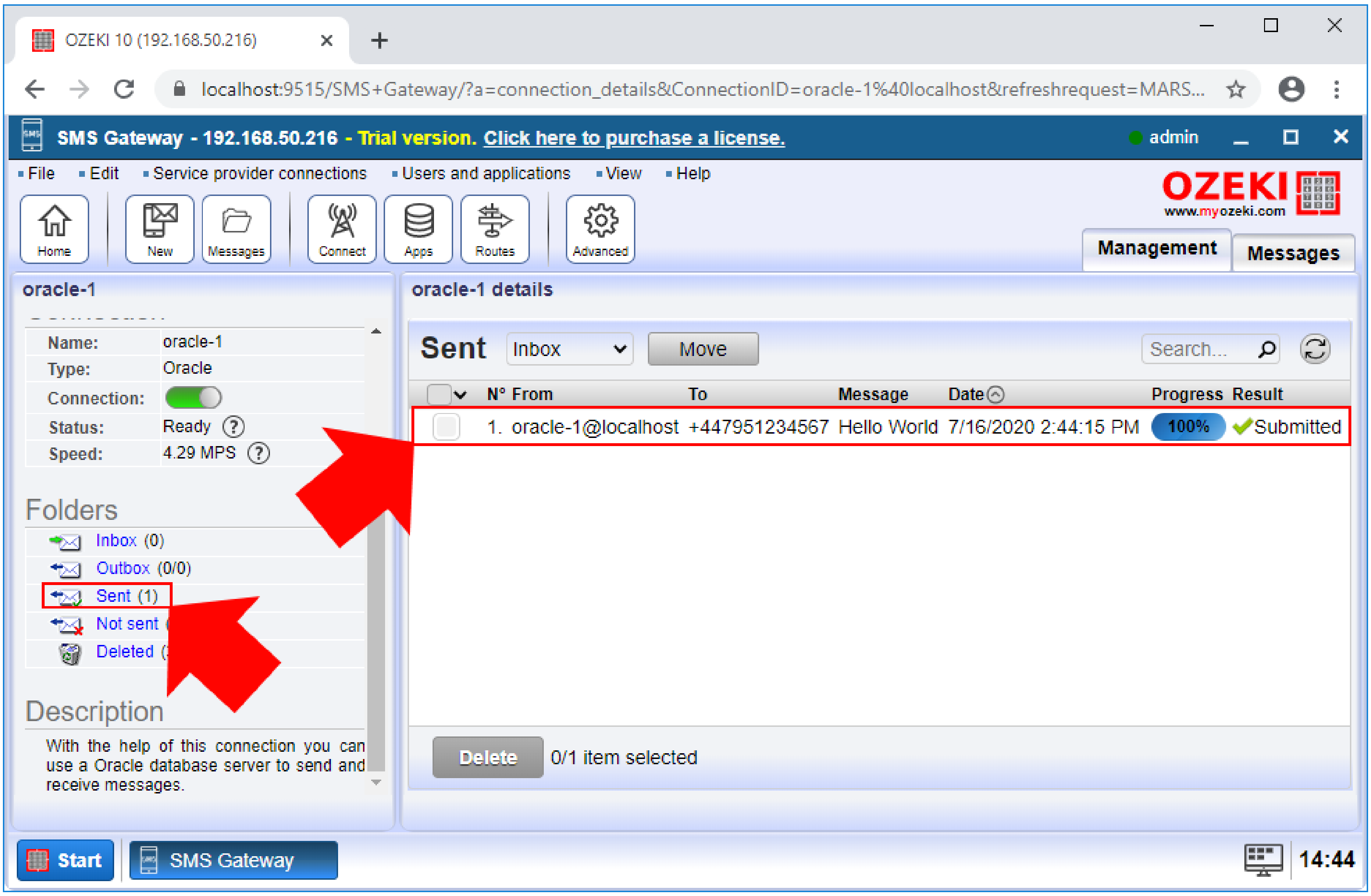
☐ Store query for later execution if no database connection is available

Execute

SQL query execution took 140 ms. Number of records affected: 0

Figure 15 - Execute the INSERT statement to create a new message record

The Oracle database connection is capable of reading the rows of the create tables, and if it detects a new message, it sends the message to the phone number that you provided as a value. Then, the message stored in the Sent folder of the connection as you can see it in Figure 16.


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=connection_details&ConnectionID=oracle-1%40localhost&refreshrequest=MARS...
SMS Gateway - 192.168.50.216 - Trial version. Click here to purchase a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
oracle-1
oracle-1 details
Folders
Inbox (0)
Outbox (0/0)
Sent (1)
Not sent
Deleted
Description
Sent
Inbox
Move
Search...
Delete
Start
SMS Gateway
14:44

| | |
|---|---|
| Name: | oracle-1 |
| Type: | Oracle |
| Connection: | ● |
| Status: | Ready |
| Speed: | 4.29 MPS |

With the help of this connection you can use a Oracle database server to send and receive messages.

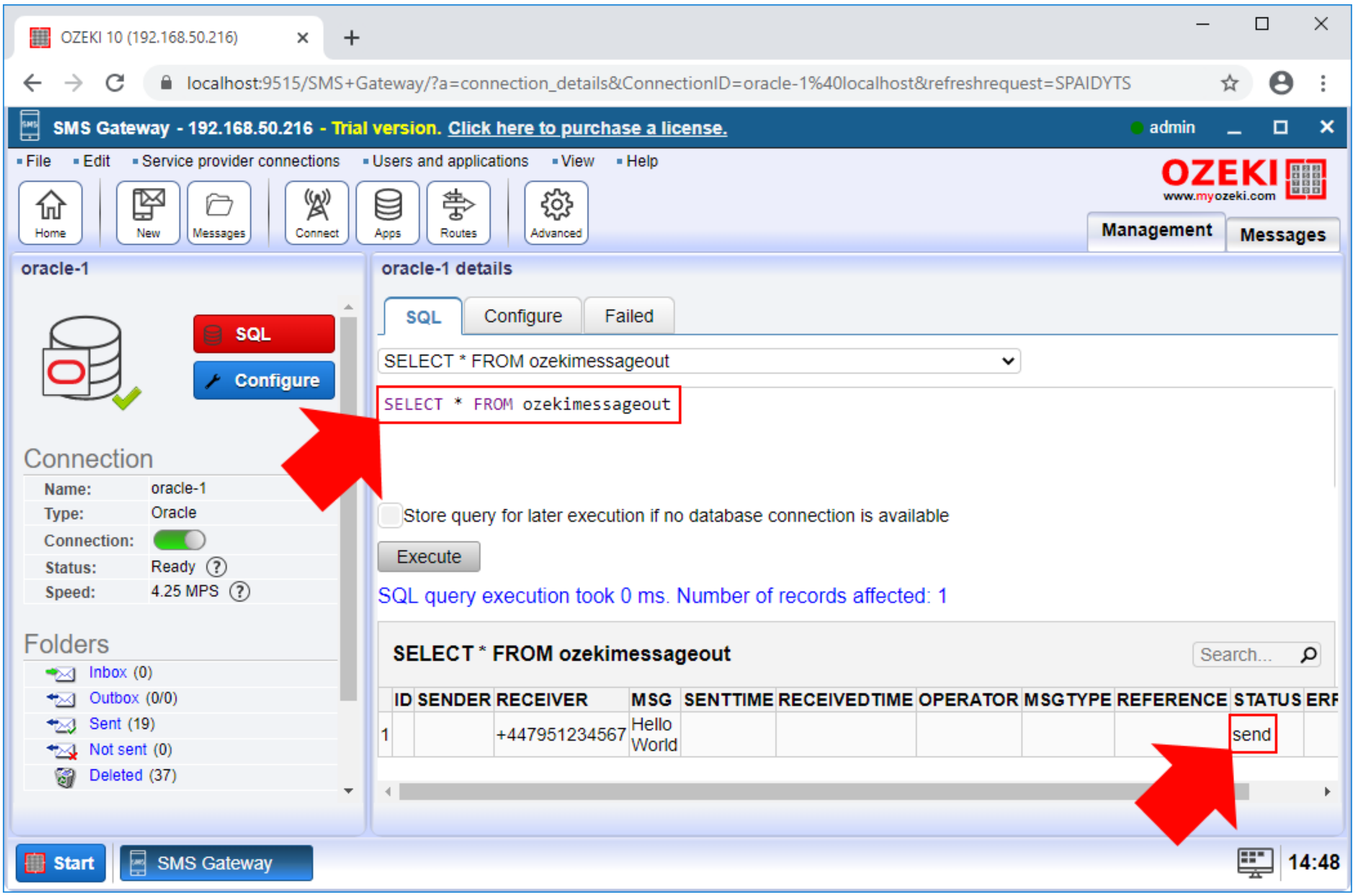
| N° | From | To | Message | Date | Progress | Result |
|---|---|---|---|---|---|---|
| 1. | oracle-1@localhost | +447951234567 | Hello World | 7/16/2020 2:44:15 PM | 100% | ✓Submitted |

0/1 item selected

Figure 16 - Message have been placed into the 'Sent' folder

You can check every message in the 'ozekimessageout' table by executing the SQL query below. This command lists every message with all its details like the phone number of the recipient, the message and the status of the message as Figure 17 demonstrates it.

```
SELECT * FROM ozekimessageout
```



| | |
|---|---|
| Name: | oracle-1 |
| Type: | Oracle |
| Connection: | |
| Status: | Ready |
| Speed: | 4.25 MPS |

SELECT * FROM ozekimessageout

```
SELECT * FROM ozekimessageout
```

□ Store query for later execution if no database connection is available

SQL query execution took 0 ms. Number of records affected: 1

**SELECT * FROM ozekimessageout**

| ID | SENDER | RECEIVER | MSG | SENTTIME | RECEIVEDTIME | OPERATOR | MSGTYPE | REFERENCE | STATUS | ERR |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | | +447951234567 | Hello World | | | | | | send | |

Figure 17 - The STATUS of the message is 'sent'

# Send SMS from Oracle SQL (part 1/4)

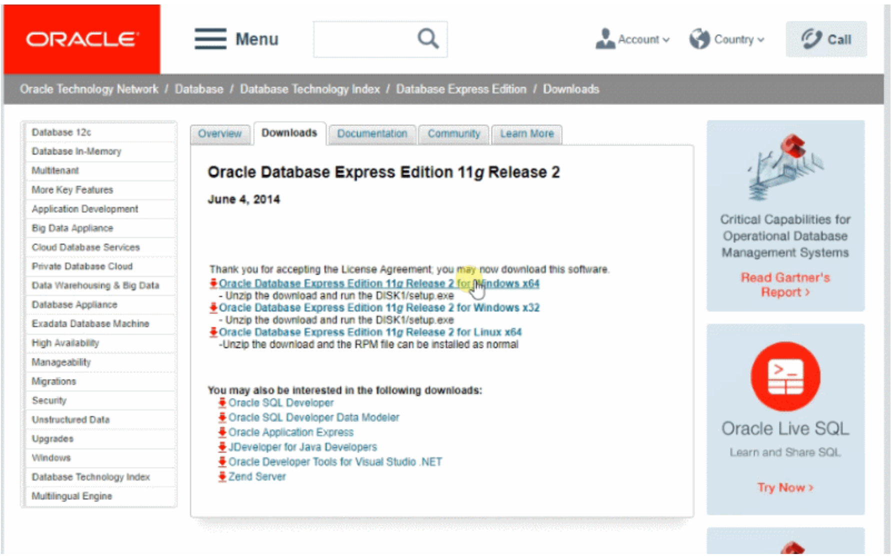
***In this video series you will see how to connect Ozeki SMS Gateway and Oracle SQL database server for SMS messaging. The first video shows how to download and install a Oracle SQL.***

**Video content**

1. Download Oracle
2. Install Oracle

---

Please jump to the next video or if you scroll down you can find screenshots that describe how to download and install **Oracle Database 11g Express** and how to set the port numbers and SYSTEM password.



**Oracle Database Express Edition 11*g* Release 2**

**June 4, 2014**

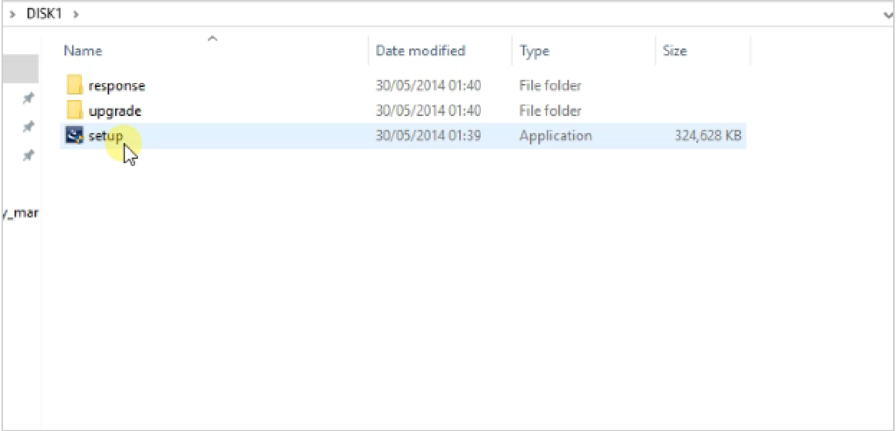
Thank you for accepting the License Agreement; you may now download this software.

- Oracle Database Express Edition 11*g* Release 2 for Windows x64
  - Unzip the download and run the DISK1/setup.exe
- Oracle Database Express Edition 11*g* Release 2 for Windows x32
  - Unzip the download and run the DISK1/setup.exe
- Oracle Database Express Edition 11*g* Release 2 for Linux x64
  -Unzip the download and the RPM file can be installed as normal

**You may also be interested in the following downloads:**

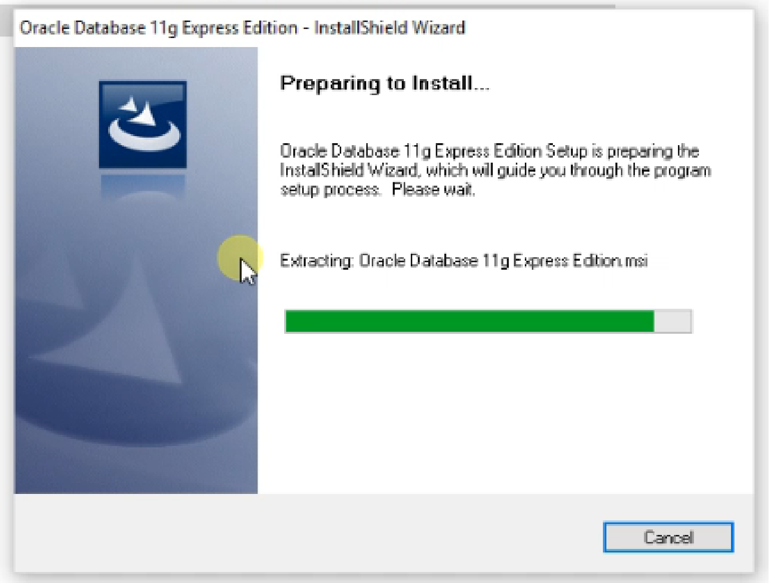
- Oracle SQL Developer
- Oracle SQL Developer Data Modeler
- Oracle Application Express
- JDeveloper for Java Developers
- Oracle Developer Tools for Visual Studio .NET
- Zend Server

Figure 1 - Download Oracle Database from Oracle's webpage



| Name | Date modified | Type | Size |
|---|---|---|---|
| response | 30/05/2014 01:40 | File folder | |
| upgrade | 30/05/2014 01:40 | File folder | |
| setup | 30/05/2014 01:39 | Application | 324,628 KB |

Figure 2 - Start the installation



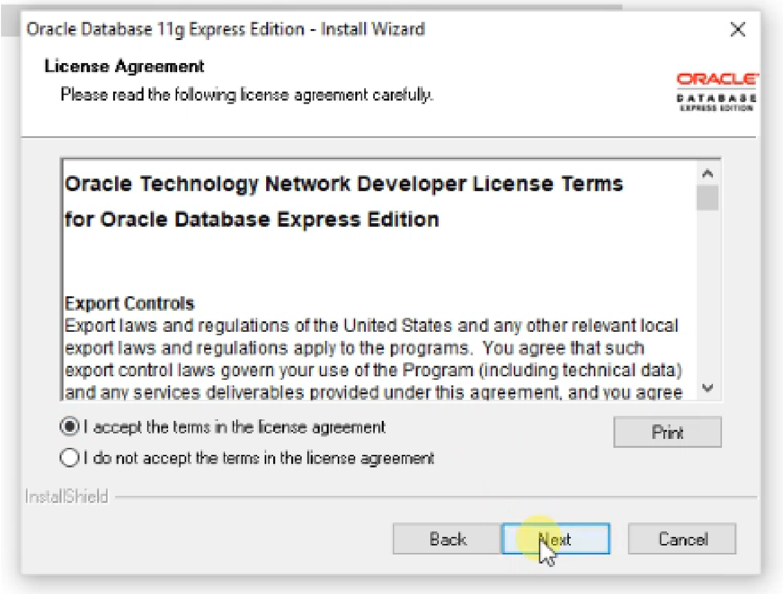
Oracle Database 11g Express Edition - InstallShield Wizard

**Preparing to Install...**

Oracle Database 11g Express Edition Setup is preparing the InstallShield Wizard, which will guide you through the program setup process. Please wait.

Extracting: Oracle Database 11g Express Edition.msi

Figure 3 - Wait until the Installation Wizard prepares installation



Oracle Database 11g Express Edition - Install Wizard

**License Agreement**

Please read the following license agreement carefully.

**Oracle Technology Network Developer License Terms for Oracle Database Express Edition**

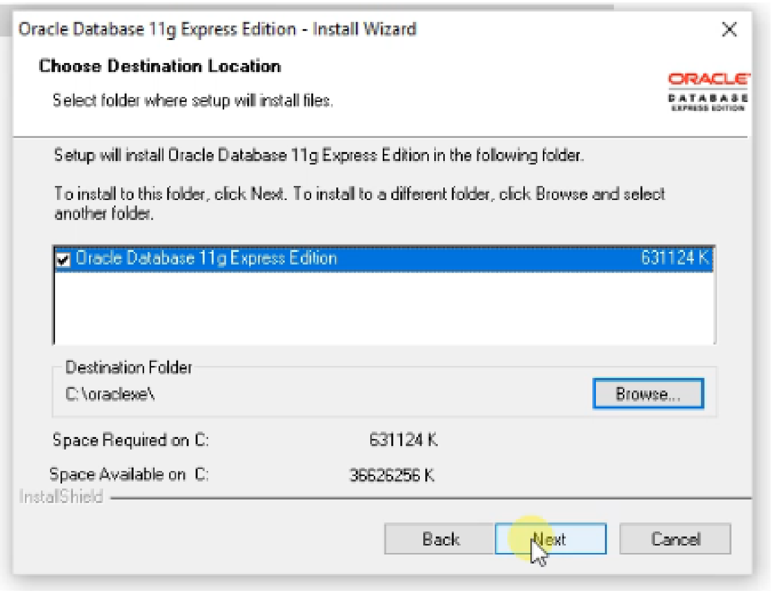
**Export Controls**

Export laws and regulations of the United States and any other relevant local export laws and regulations apply to the programs. You agree that such export control laws govern your use of the Program (including technical data) and any services deliverables provided under this agreement, and you agree

● I accept the terms in the license agreement

○ I do not accept the terms in the license agreement

Figure 4 - Accept license agreement



Oracle Database 11g Express Edition - Install Wizard

**Choose Destination Location**

Select folder where setup will install files.

Setup will install Oracle Database 11g Express Edition in the following folder.

To install to this folder, click Next. To install to a different folder, click Browse and select another folder.

☑ Oracle Database 11g Express Edition 631124 K

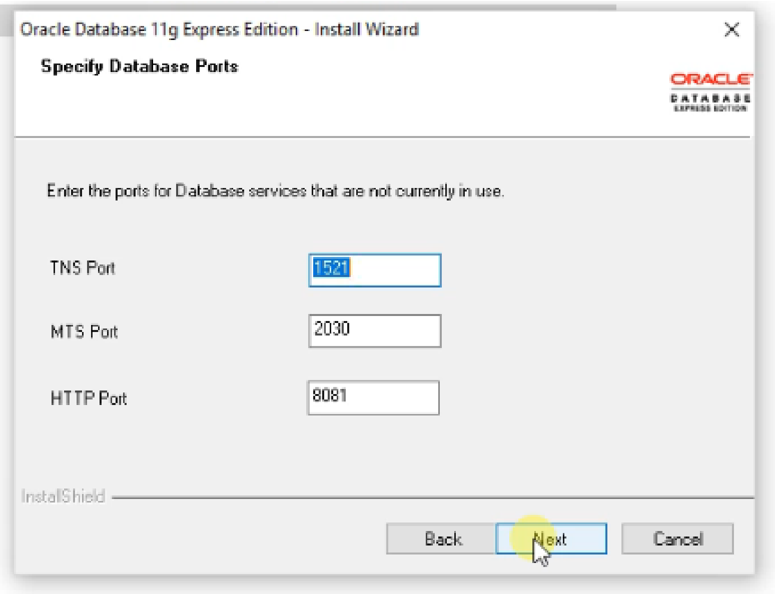
Destination Folder

C:\oraclexe\

Space Required on C: 631124 K

Space Available on C: 36626256 K

Figure 5 - Choose destination folder


Oracle Database 11g Express Edition - Install Wizard

Specify Database Ports

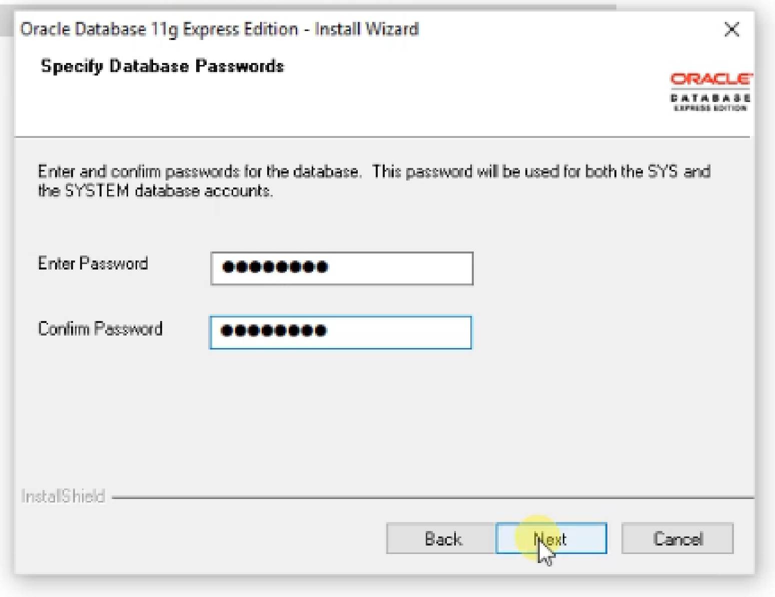
Enter the ports for Database services that are not currently in use.

TNS Port 1521

MTS Port 2030

HTTP Port 8081

Figure 6 - Specify Oracle Database 11g Express ports


Oracle Database 11g Express Edition - Install Wizard

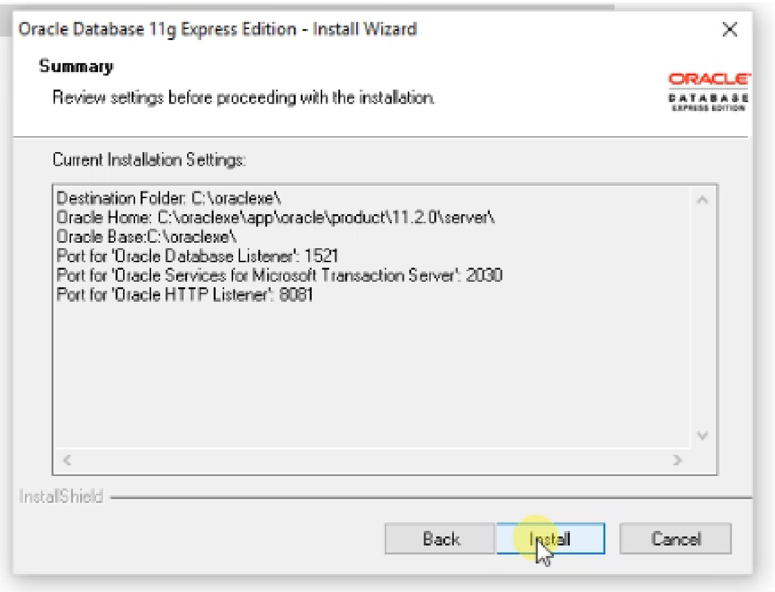
Specify Database Passwords

Enter and confirm passwords for the database. This password will be used for both the SYS and the SYSTEM database accounts.

Enter Password ●●●●●●●●

Confirm Password ●●●●●●●●

Figure 7 - Specify SYSTEM database account's password


Oracle Database 11g Express Edition - Install Wizard

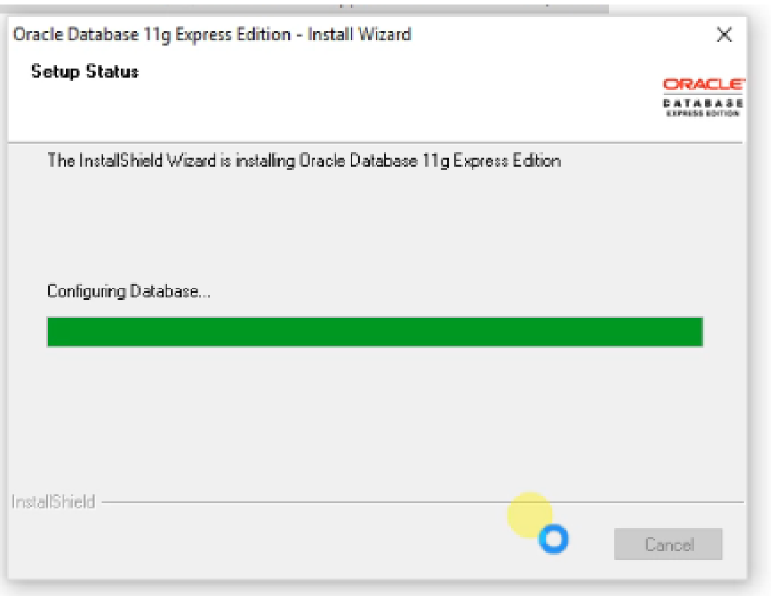
Summary

Review settings before proceeding with the installation.

Current Installation Settings:

```
Destination Folder: C:\oraclexe\
Oracle Home: C:\oraclexe\app\oracle\product\11.2.0\server\
Oracle Base:C:\oraclexe\
Port for 'Oracle Database Listener': 1521
Port for 'Oracle Services for Microsoft Transaction Server': 2030
Port for 'Oracle HTTP Listener': 8081
```

Figure 8 - Click the 'Install' button



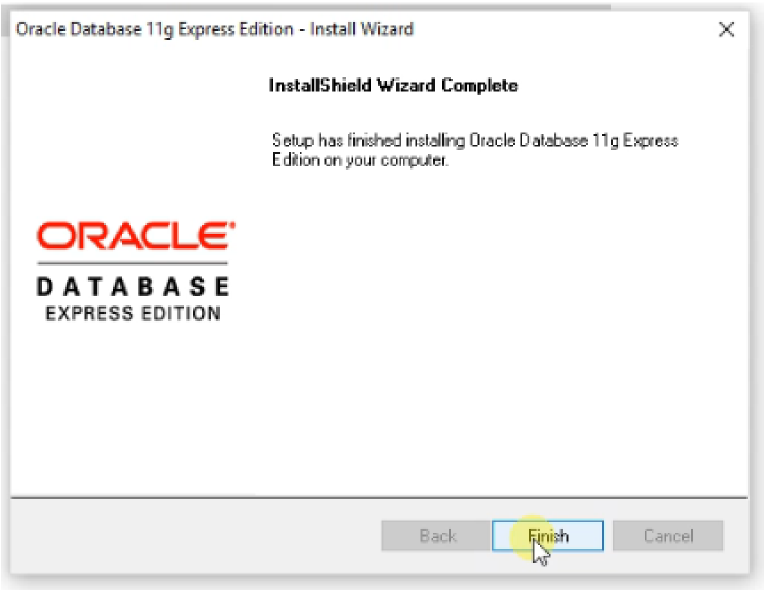
Oracle Database 11g Express Edition - Install Wizard

Setup Status

The InstallShield Wizard is installing Oracle Database 11g Express Edition

Configuring Database...

Figure 9 - Wait until the installation is finished



Oracle Database 11g Express Edition - Install Wizard

InstallShield Wizard Complete

Setup has finished installing Oracle Database 11g Express Edition on your computer.

Figure 10 - Oracle 11g Express's installation has been finished

# Send SMS from Oracle SQL (part 2/4)

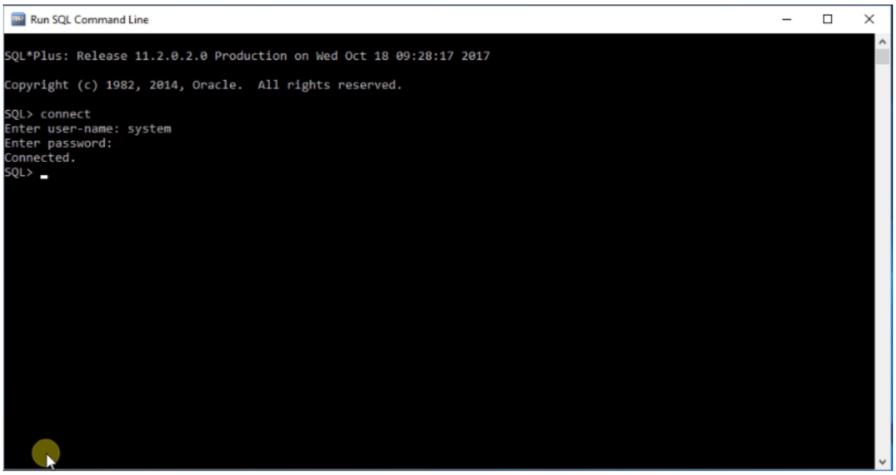
***In this video series you can see how to connect Ozeki SMS Gateway and Oracle SQL database server for SMS messaging. This video shows how to create the proper database table structure by using Oracle SQL through it's command line.***

**Video content**

1. Connect to Oracle
2. Copy CREATE TABLE statement

---

Please **scroll down to copy the SQL statements** (**Figure 2-4**) used in the video. If you have created the database in Oracle 11g Express, you can jump to the next video.



```
SQL*Plus: Release 11.2.0.2.0 Production on Wed Oct 18 09:28:17 2017

Copyright (c) 1982, 2014, Oracle.  All rights reserved.

SQL> connect
Enter user-name: system
Enter password:
Connected.
SQL> _
```

Figure 1 - Connect to the Oracle database

## Oracle SQL statements to copy:

### CREATE TABLESPACE:

```
DATAFILE 'C:\oraclexe\app\oracle\oradata\XE\ozeki.dbf'
SIZE 40M autoextend on;
```

Figure 2 - CREATE TABLESPACE ozeki

### CREATE USER:

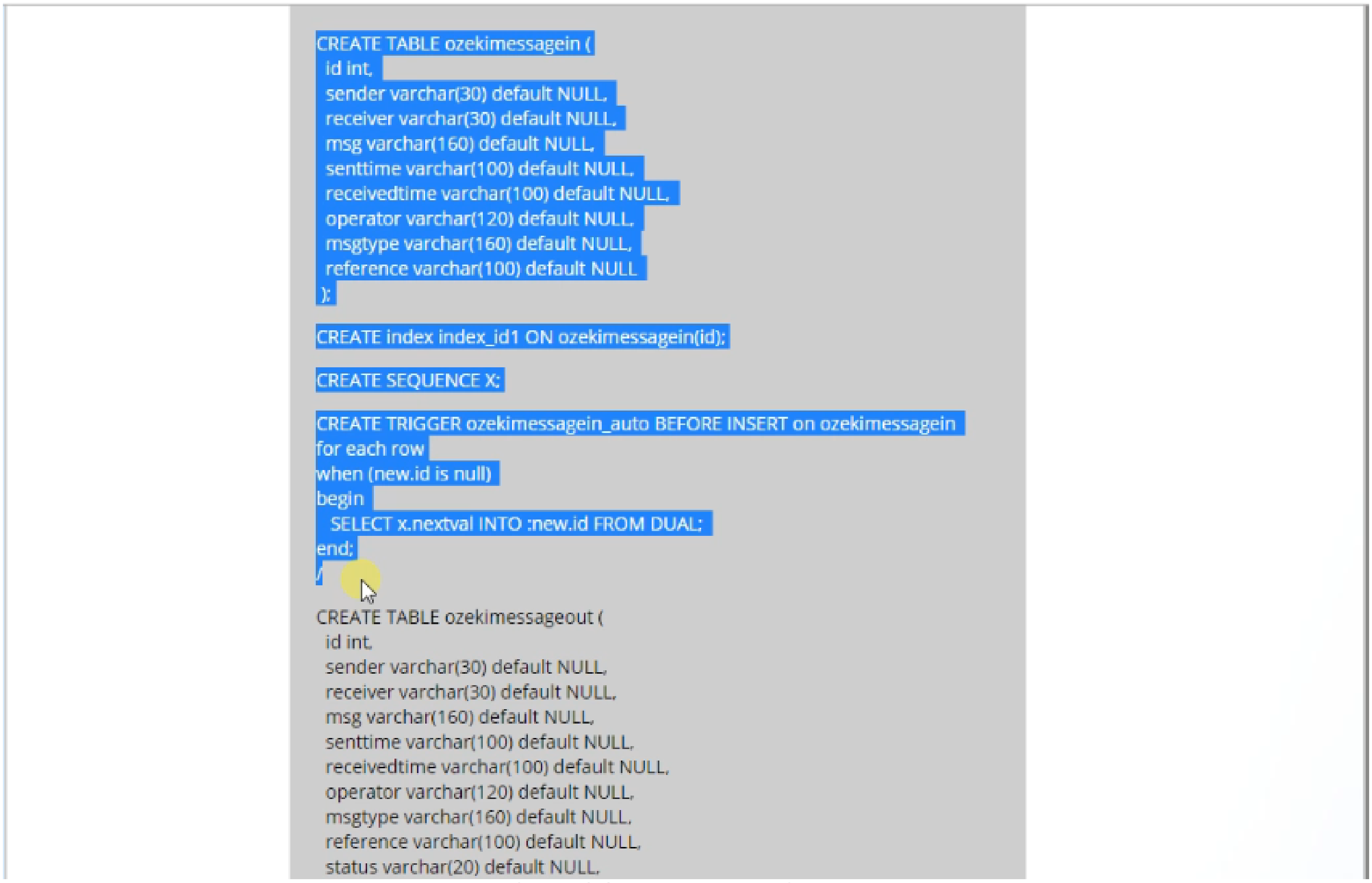
```
IDENTIFIED BY qwe123
DEFAULT TABLESPACE ozeki;

GRANT DBA TO ozeki;<br>
```

Figure 3 - CREATE USER ozeki and GRANT database access TO ozeki

### CREATE TABLE:

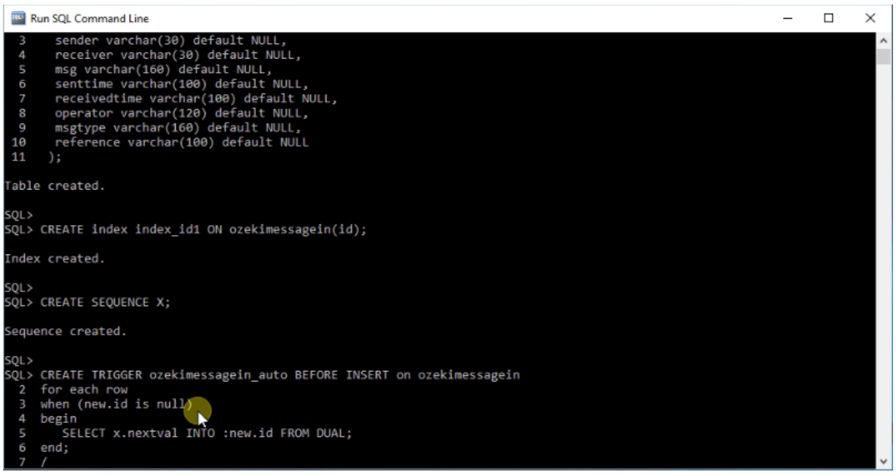
```
<textarea style="width:100%" readonly="" rows="41">CREATE TABLE ozekimessagein (
    id int,
    sender varchar(30) default NULL,
    receiver varchar(30) default NULL,
    msg varchar(160) default NULL,
    senttime varchar(100) default NULL,
    receivedtime varchar(100) default NULL,
    operator varchar(120) default NULL,
    msgtype varchar(160) default NULL,
    reference varchar(100) default NULL

CREATE index index_id1 ON ozekimessagein(id);
CREATE SEQUENCE X;
CREATE TRIGGER ozekimessagein_auto BEFORE INSERT on ozekimessagein
    for each row when (new.id is null)
    begin
    SELECT x.nextval INTO :new.id FROM DUAL;
    end;
    /
CREATE TABLE ozekimessageout (
    id int,
    sender varchar(30) default NULL,
    receiver varchar(30) default NULL,
    msg varchar(160) default NULL,
    senttime varchar(100) default NULL,
    receivedtime varchar(100) default NULL,
    operator varchar(120) default NULL,
    msgtype varchar(160) default NULL,
    reference varchar(100) default NULL,
    status varchar(20) default NULL,
    errormsg varchar(250) default NULL
    ;
CREATE index index_id2 ON ozekimessageout(id);
CREATE SEQUENCE Y;
CREATE TRIGGER ozekimessageout_auto BEFORE INSERT on ozekimessageout
    for each row
    when (new.id is null)
    begin
       SELECT y.nextval INTO :new.id FROM DUAL;
    end;
    /</textarea>

```

Figure 4 - CREATE TABLE ozekimessagein and ozekimessageout



```
CREATE TABLE ozekimessagein (
 id int,
 sender varchar(30) default NULL,
 receiver varchar(30) default NULL,
 msg varchar(160) default NULL,
 senttime varchar(100) default NULL,
 receivedtime varchar(100) default NULL,
 operator varchar(120) default NULL,
 msgtype varchar(160) default NULL,
 reference varchar(100) default NULL
);

CREATE index index_id1 ON ozekimessagein(id);

CREATE SEQUENCE X;

CREATE TRIGGER ozekimessagein_auto BEFORE INSERT on ozekimessagein
for each row
when (new.id is null)
begin
  SELECT x.nextval INTO :new.id FROM DUAL;
end;
/

CREATE TABLE ozekimessageout (
 id int,
 sender varchar(30) default NULL,
 receiver varchar(30) default NULL,
 msg varchar(160) default NULL,
 senttime varchar(100) default NULL,
 receivedtime varchar(100) default NULL,
 operator varchar(120) default NULL,
 msgtype varchar(160) default NULL,
 reference varchar(100) default NULL,
 status varchar(20) default NULL,
```

Figure 5 - Copy SQL statements from Figure 2-4



```
  3    sender varchar(30) default NULL,
  4    receiver varchar(30) default NULL,
  5    msg varchar(160) default NULL,
  6    senttime varchar(100) default NULL,
  7    receivedtime varchar(100) default NULL,
  8    operator varchar(120) default NULL,
  9    msgtype varchar(160) default NULL,
 10    reference varchar(100) default NULL
 11   );

Table created.

SQL>
SQL> CREATE index index_id1 ON ozekimessagein(id);

Index created.

SQL>
SQL> CREATE SEQUENCE X;

Sequence created.

SQL>
SQL> CREATE TRIGGER ozekimessagein_auto BEFORE INSERT on ozekimessagein
  2  for each row
  3  when (new.id is null)
  4  begin
  5     SELECT x.nextval INTO :new.id FROM DUAL;
  6  end;
  7  /
```

Figure 6 - Paste the statements and run them to create table structure

# Send SMS from Oracle SQL (part 3/4)

***In this video series you can see how to connect Ozeki SMS Gateway and Oracle SQL database server for SMS messaging. This video shows how to install a Oracle databse connection on Ozeki SMS Gateway.***

**Video content**

1. Open SMS Gateway
2. Create Oracle connection
3. Configure Oracle connection

If you have created the database in Oracle 11g Express, you can jump to the next video. Although you can precisely examine the sequence by looking through the screenshots.

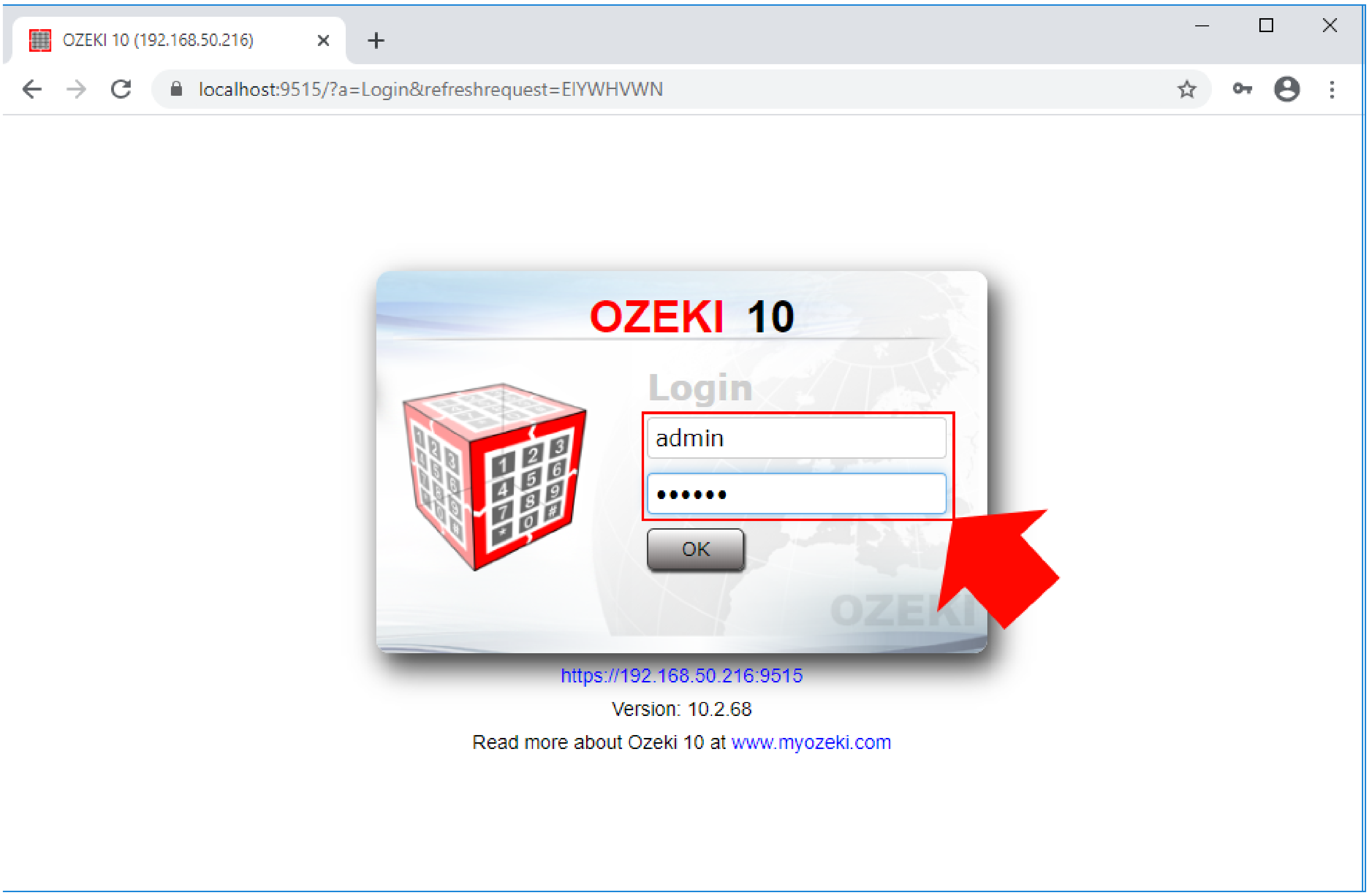

OZEKI 10 (192.168.50.216)
localhost:9515/?a=Login&refreshrequest=EIYWHVWN
OZEKI 10
Login
admin
••••••
OK
https://192.168.50.216:9515
Version: 10.2.68
Read more about Ozeki 10 at www.myozeki.com

Figure 1 - Login to Ozeki 10

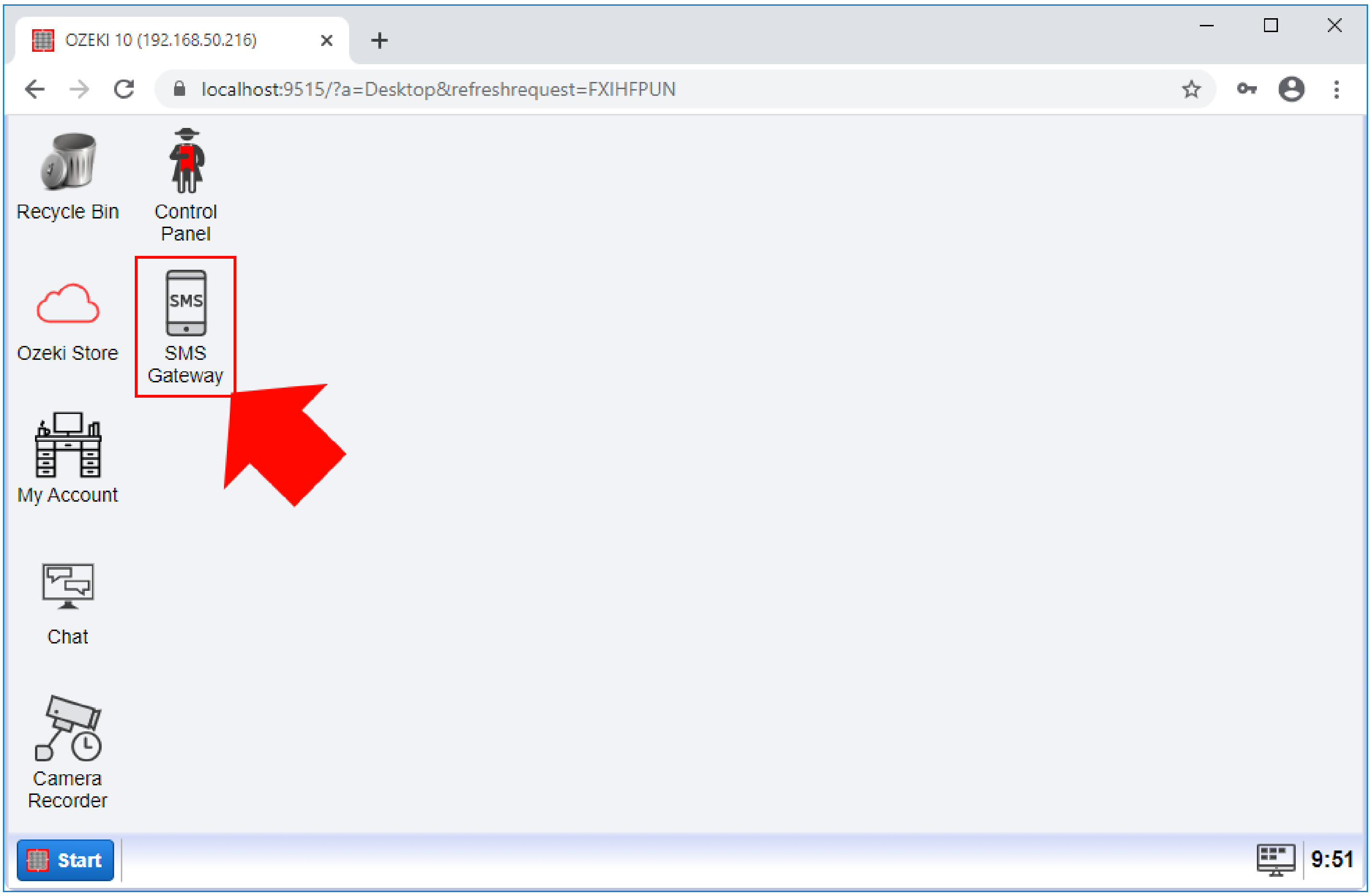
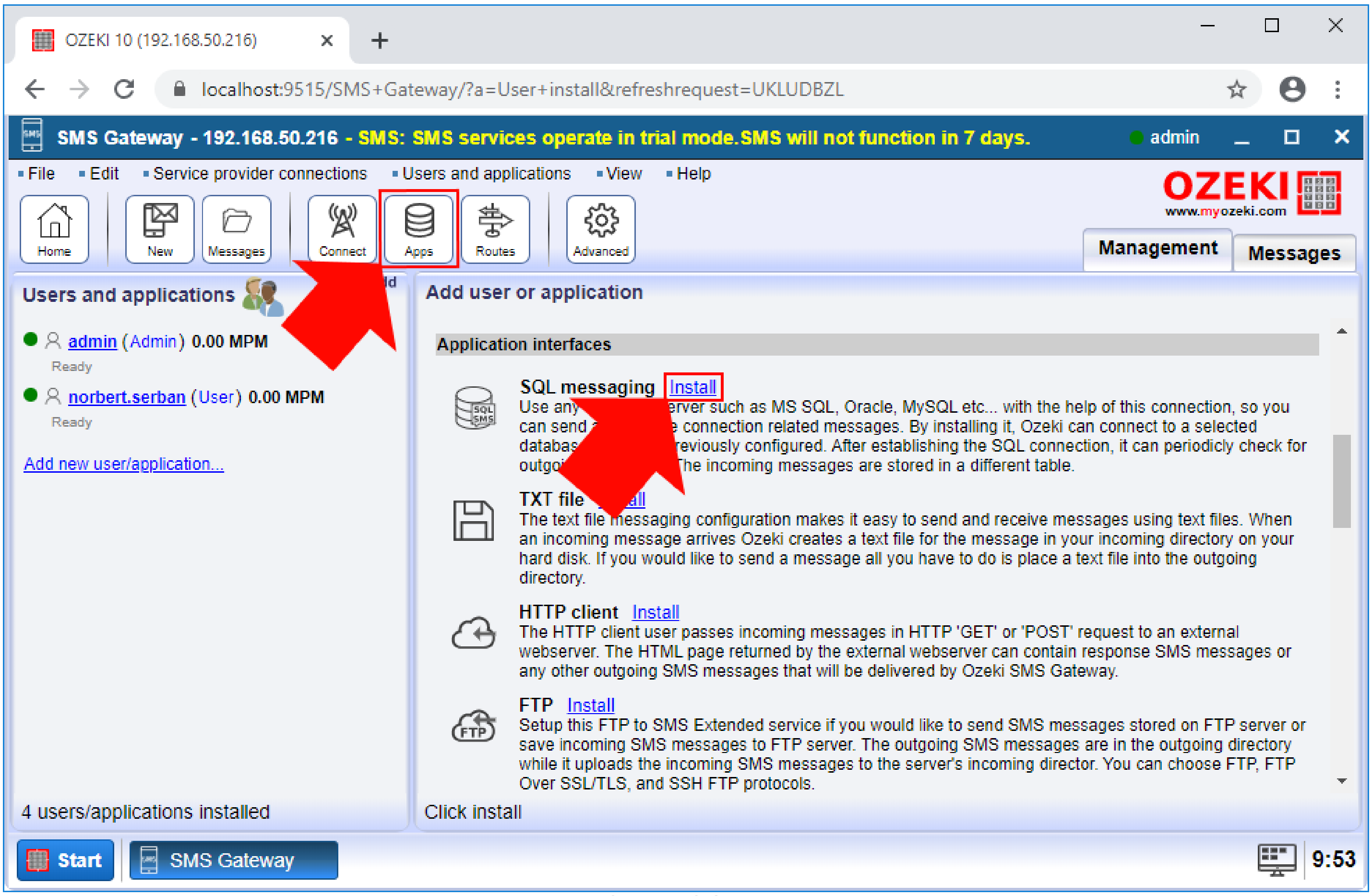


Figure 2 - Open the SMS Gateway applicaton from Ozeki 10's desktop screen


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=User+install&refreshrequest=UKLUDBZL
SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
OZEKI
www.myozeki.com
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Users and applications
admin (Admin) 0.00 MPM
Ready
norbert.serban (User) 0.00 MPM
Ready
Add new user/application...
4 users/applications installed
Add user or application
Application interfaces
Click install
Start
SMS Gateway
9:53

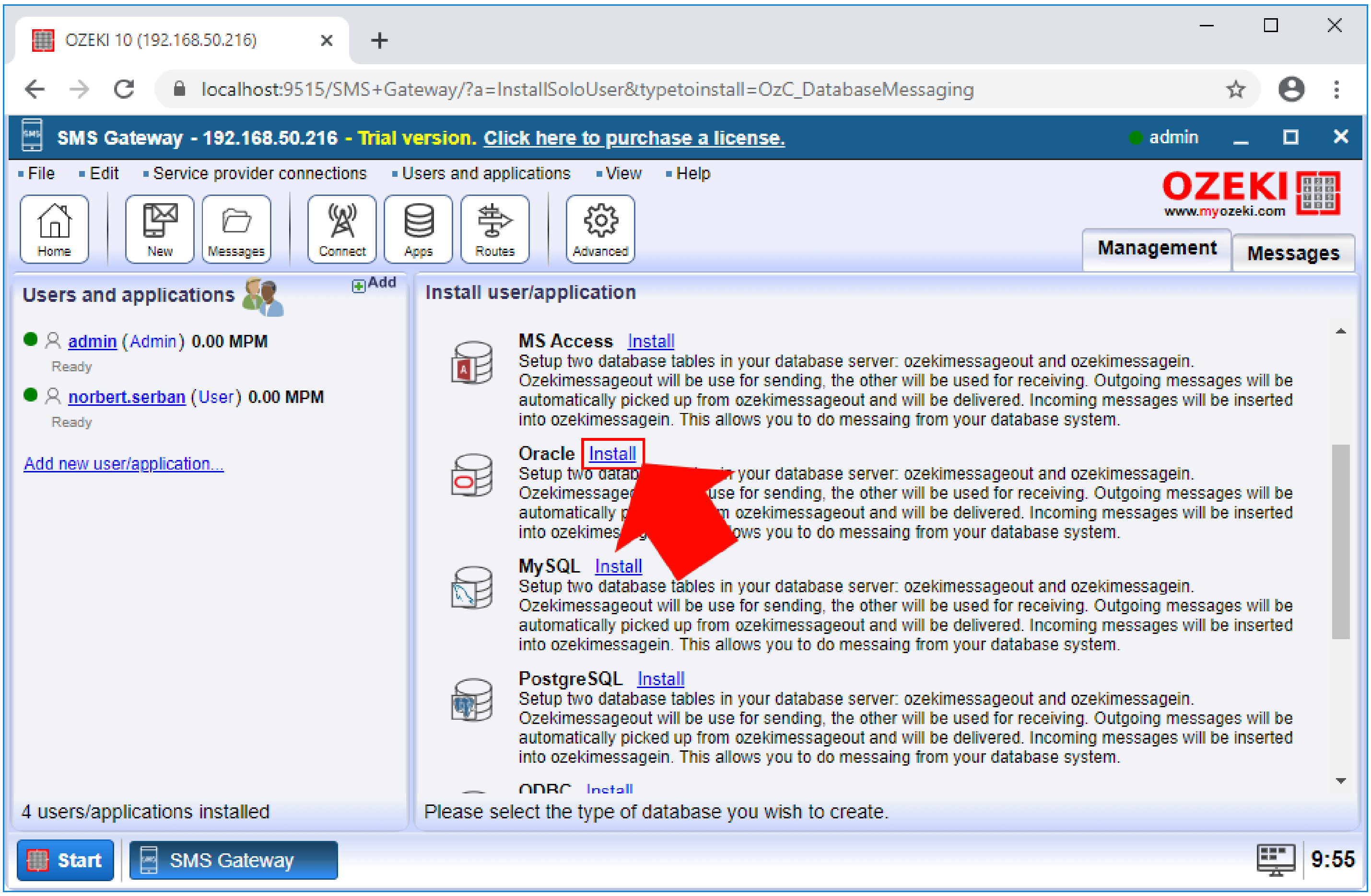
SQL messaging Install
Use any ... erver such as MS SQL, Oracle, MySQL etc... with the help of this connection, so you can send ... e connection related messages. By installing it, Ozeki can connect to a selected databas... reviously configured. After establishing the SQL connection, it can periodicly check for outgo... The incoming messages are stored in a different table.

TXT file ...all
The text file messaging configuration makes it easy to send and receive messages using text files. When an incoming message arrives Ozeki creates a text file for the message in your incoming directory on your hard disk. If you would like to send a message all you have to do is place a text file into the outgoing directory.

HTTP client Install
The HTTP client user passes incoming messages in HTTP 'GET' or 'POST' request to an external webserver. The HTML page returned by the external webserver can contain response SMS messages or any other outgoing SMS messages that will be delivered by Ozeki SMS Gateway.

FTP Install
Setup this FTP to SMS Extended service if you would like to send SMS messages stored on FTP server or save incoming SMS messages to FTP server. The outgoing SMS messages are in the outgoing directory while it uploads the incoming SMS messages to the server's incoming director. You can choose FTP, FTP Over SSL/TLS, and SSH FTP protocols.

Figure 3 - Select SQL messaging



OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=InstallSoloUser&typetoinstall=OzC_DatabaseMessaging
SMS Gateway - 192.168.50.216 - Trial version. Click here to purchase a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
OZEKI
www.myozeki.com
Management
Messages
Start
SMS Gateway
9:55

Users and applications

Add

admin (Admin) 0.00 MPM
Ready

norbert.serban (User) 0.00 MPM
Ready

Add new user/application...

4 users/applications installed

Install user/application

MS Access Install
Setup two database tables in your database server: ozekimessageout and ozekimessagein. Ozekimessageout will be use for sending, the other will be used for receiving. Outgoing messages will be automatically picked up from ozekimessageout and will be delivered. Incoming messages will be inserted into ozekimessagein. This allows you to do messaing from your database system.

Oracle Install
Setup two datab... your database server: ozekimessageout and ozekimessagein. Ozekimessage... use for sending, the other will be used for receiving. Outgoing messages will be automatically p... m ozekimessageout and will be delivered. Incoming messages will be inserted into ozekimes... ows you to do messaing from your database system.

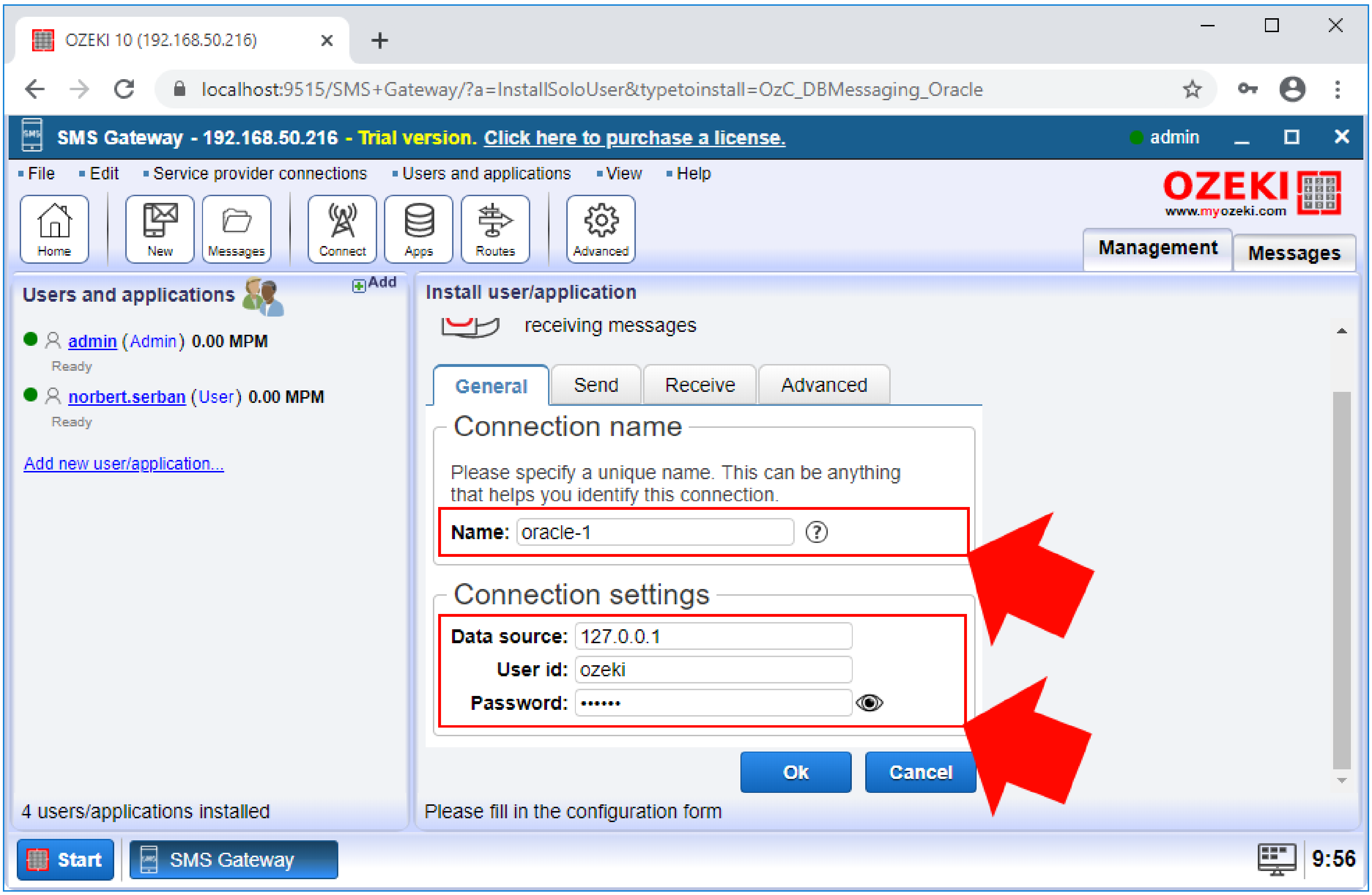
MySQL Install
Setup two database tables in your database server: ozekimessageout and ozekimessagein. Ozekimessageout will be use for sending, the other will be used for receiving. Outgoing messages will be automatically picked up from ozekimessageout and will be delivered. Incoming messages will be inserted into ozekimessagein. This allows you to do messaing from your database system.

PostgreSQL Install
Setup two database tables in your database server: ozekimessageout and ozekimessagein. Ozekimessageout will be use for sending, the other will be used for receiving. Outgoing messages will be automatically picked up from ozekimessageout and will be delivered. Incoming messages will be inserted into ozekimessagein. This allows you to do messaing from your database system.

ODBC Install

Please select the type of database you wish to create.

Figure 4 - Click on Install button of Oracle connection



OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=InstallSoloUser&typetoinstall=OzC_DBMessaging_Oracle
SMS Gateway - 192.168.50.216 - Trial version. Click here to purchase a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
OZEKI
www.myozeki.com
Management
Messages
General
Send
Receive
Advanced
Ok
Cancel
Start
SMS Gateway
9:56

Users and applications

Add

admin (Admin) 0.00 MPM
Ready

norbert.serban (User) 0.00 MPM
Ready

Add new user/application...

4 users/applications installed

Install user/application

receiving messages

Connection name

Please specify a unique name. This can be anything that helps you identify this connection.

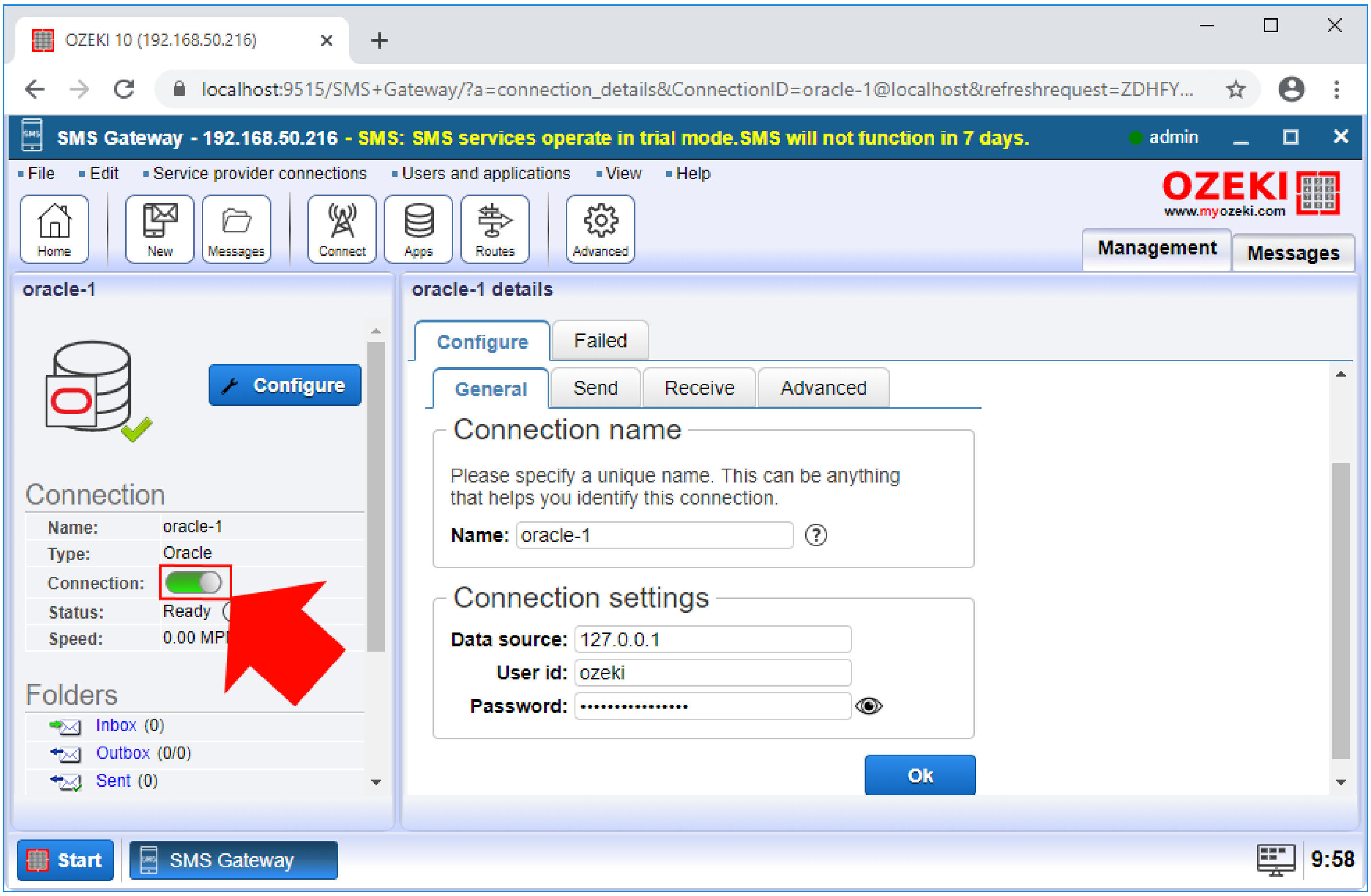
Name: oracle-1

Connection settings

Data source: 127.0.0.1
User id: ozeki
Password: ••••••

Please fill in the configuration form

Figure 5 - Provide details of Oracle database



OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=connection_details&ConnectionID=oracle-1@localhost&refreshrequest=ZDHFY...
SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
OZEKI
www.myozeki.com
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
oracle-1
Configure
Folders
Inbox (0)
Outbox (0/0)
Sent (0)
oracle-1 details
Configure
Failed
General
Send
Receive
Advanced
Ok
Start
SMS Gateway
9:58

Connection

Name: oracle-1
Type: Oracle
Connection:
Status: Ready
Speed: 0.00 MPI

Connection name

Please specify a unique name. This can be anything that helps you identify this connection.

Name: oracle-1

Connection settings

Data source: 127.0.0.1
User id: ozeki
Password: ••••••••••••••••

Figure 6 - Enable Oracle databse connection

# Send SMS from Oracle SQL (part 4/4)

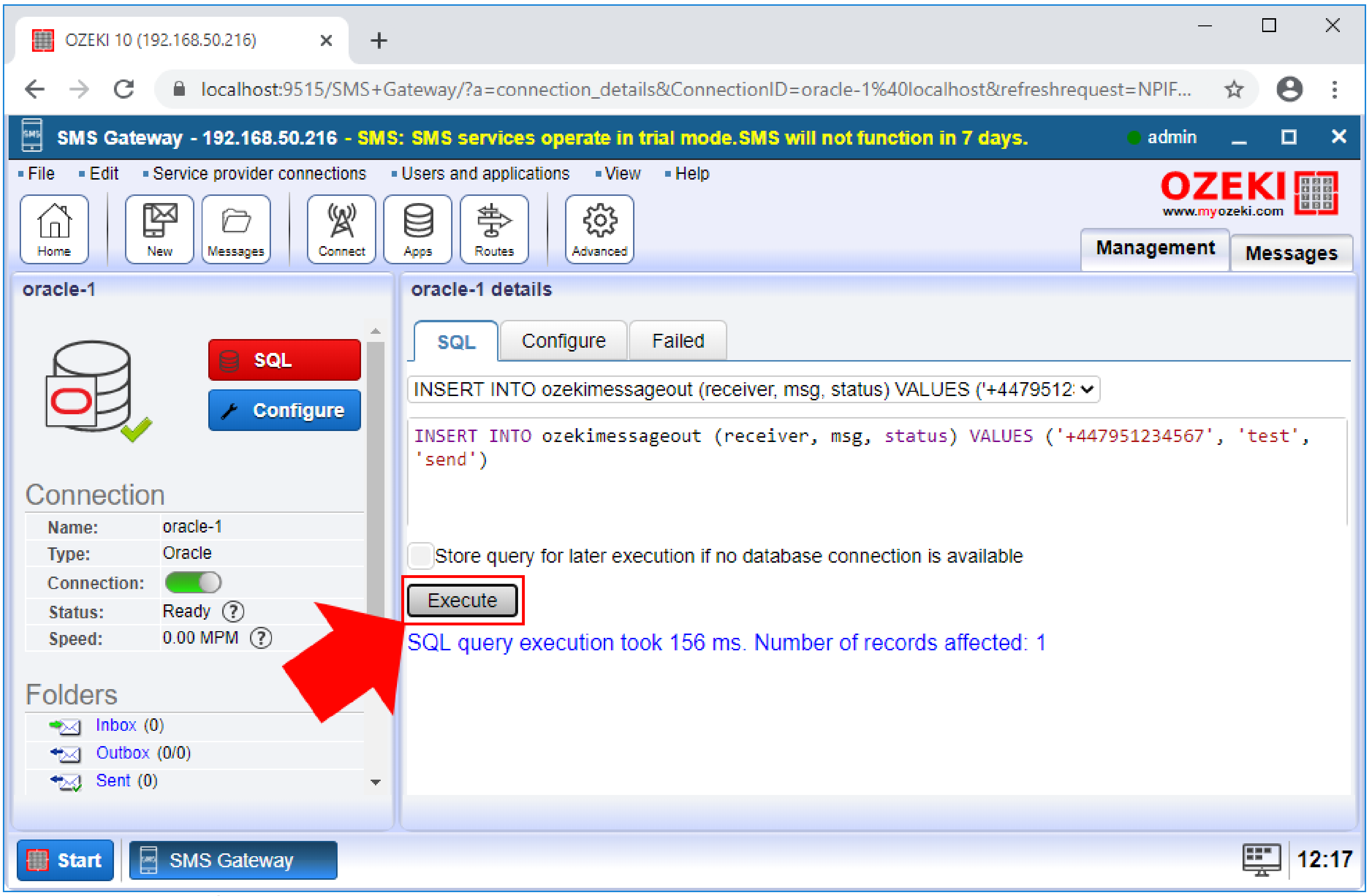
***This is the last video in this video series which shows how to connect Ozeki SMS Gateway and Oracle SQL database server for SMS messaging. See how to insert a simple row into the proper table, so Ozeki SMS Gateway can automatically send SMS messages to mobile phones.***

**Video content**

1. Execute SQL command
2. Check sent message

---

Look at the upcoming screenshots to thoroughly examine the final stage, which is SMS sending.


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=connection_details&ConnectionID=oracle-1%40localhost&refreshrequest=NPIF...
SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
oracle-1
SQL
Configure
oracle-1 details
SQL
Configure
Failed
Execute
Folders
Inbox (0)
Outbox (0/0)
Sent (0)
Start
SMS Gateway
12:17

INSERT INTO ozekimessageout (receiver, msg, status) VALUES ('+4479512

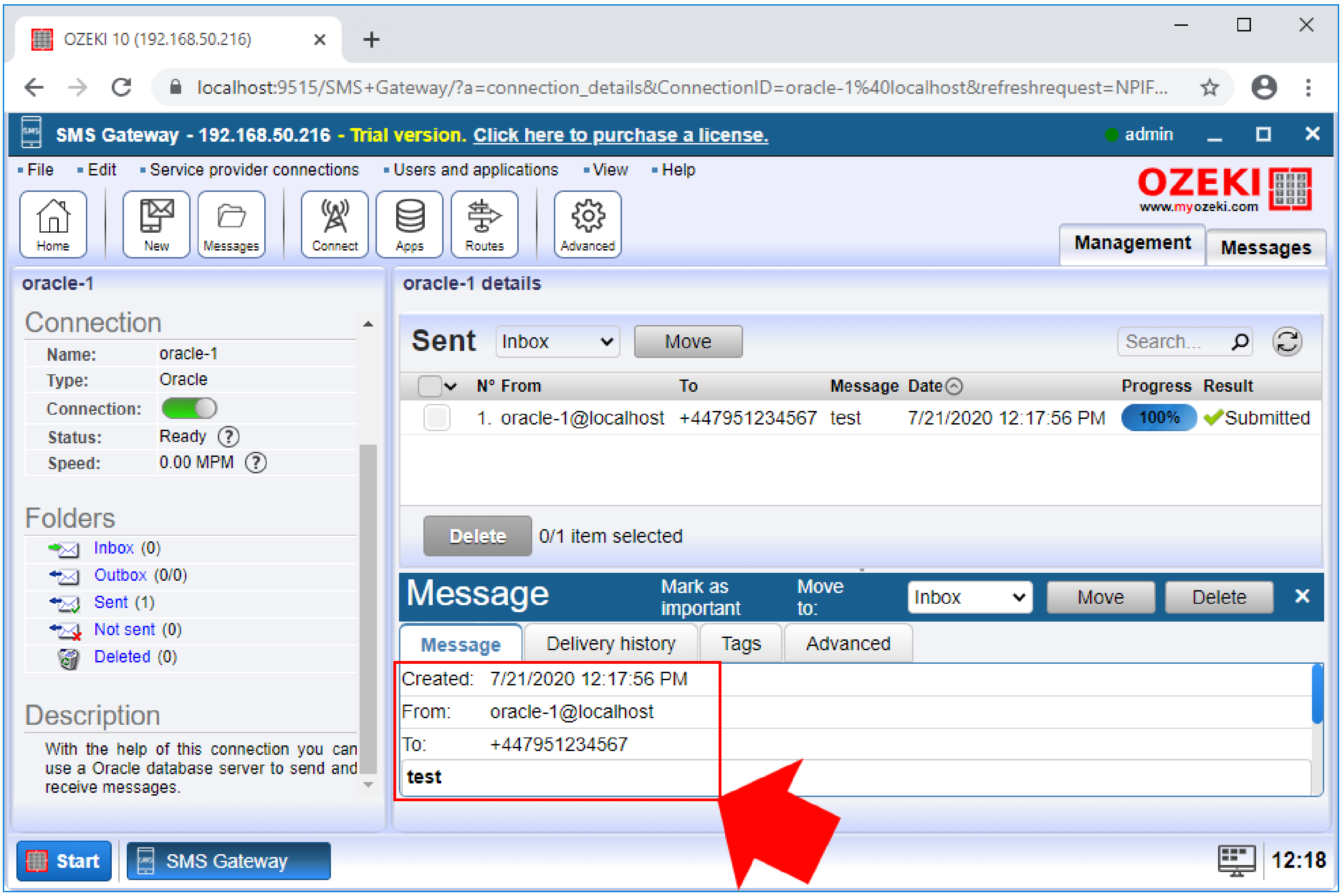
```
INSERT INTO ozekimessageout (receiver, msg, status) VALUES ('+447951234567', 'test',
'send')
```

□ Store query for later execution if no database connection is available

SQL query execution took 156 ms. Number of records affected: 1

Connection
Name: oracle-1
Type: Oracle
Connection:
Status: Ready
Speed: 0.00 MPM

Figure 1 - Use INSERT statement to send message


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=connection_details&ConnectionID=oracle-1%40localhost&refreshrequest=NPIF...
SMS Gateway - 192.168.50.216 - Trial version. Click here to purchase a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Start
SMS Gateway
12:18

oracle-1

Connection

Name: oracle-1
Type: Oracle
Connection:
Status: Ready
Speed: 0.00 MPM

Folders

Inbox (0)
Outbox (0/0)
Sent (1)
Not sent (0)
Deleted (0)

Description

With the help of this connection you can use a Oracle database server to send and receive messages.

oracle-1 details

Sent  Inbox  Move  Search...

| N° | From | To | Message | Date | Progress | Result |
|---|---|---|---|---|---|---|
| 1. | oracle-1@localhost | +447951234567 | test | 7/21/2020 12:17:56 PM | 100% | ✓Submitted |

Delete  0/1 item selected

Message  Mark as important  Move to: Inbox  Move  Delete

Message | Delivery history | Tags | Advanced

Created: 7/21/2020 12:17:56 PM
From: oracle-1@localhost
To: +447951234567
**test**

Figure 2 - The Sent folder stores each sent message with details

# SMS from/to MySQL

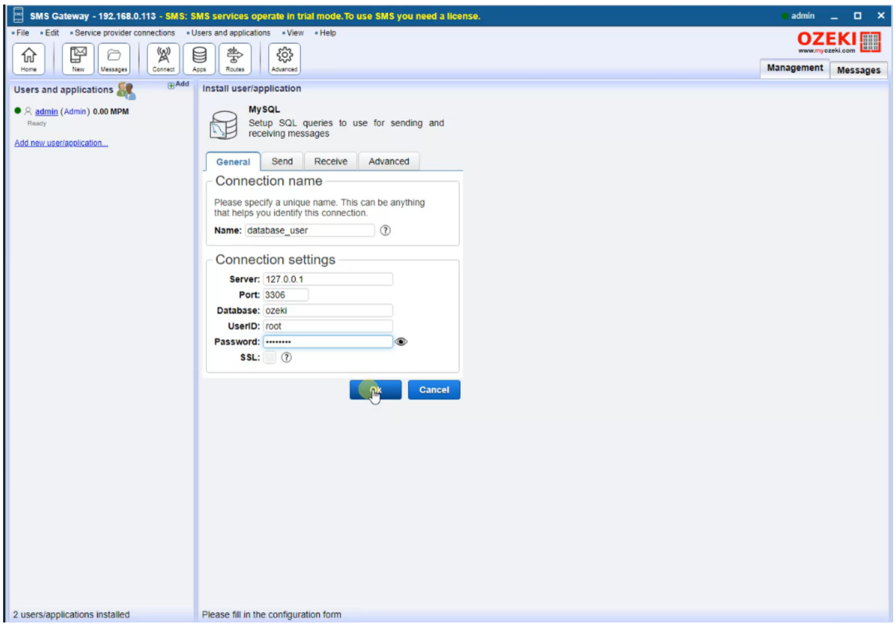
**You can combine Ozeki SMS Gateway with a MySQL database to send, receive and store SMS messages. In this tutorial you can see how to install and configure the MYSQL database connection and how to create the recommended database structure.**

## How to send SMS from MySQL

MySQL Installation (Video guide)
Create Database Tables (Video guide)
Configure Database User (Video guide)
Send Test Message (Video guide)

**The solution uses Ozeki SMS Gateway installed on your PC.** The Database User of SMS Gateway can easily communicate with your MySQL database user.

Please **define the MYSQL database connection details** onto the Database User's configuration form (**Figure 1**).



SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.To use SMS you need a license.

Install user/application

MySQL
Setup SQL queries to use for sending and receiving messages

Connection name

Please specify a unique name. This can be anything that helps you identify this connection.

Name: database_user

Connection settings

Server: 127.0.0.1
Port: 3306
Database: ozeki
UserID: root
Password: ••••••••
SSL: □

Please fill in the configuration form

**Figure 1** - Specifying the connection string for the MySQL database

After the connection details are set, it is very important to **create the database table structure**. The example database in this tutorial consists of two tables 'ozekimessageout' and 'ozekimessagein'. You can choose any table name, although you should keep them in mind. You are recommended to **use the following table layout:**

**MySQL create table script:**

```
CREATE TABLE ozekimessagein (
id int(11) NOT NULL
auto_increment,
sender varchar(30) default NULL,
receiver varchar(30) default NULL,
msg text default NULL,
senttime varchar(100) default NULL,
receivedtime varchar(100) default
NULL,
operator varchar(100) default NULL,
msgtype varchar(160) default NULL,
reference varchar(100) default
NULL,
PRIMARY KEY (id)
) charset=utf8;
ALTER TABLE ozekimessagein ADD
INDEX (id) ;
```

```
CREATE TABLE ozekimessageout (
id int(11) NOT NULL
auto_increment,
sender varchar(30) default NULL,
receiver varchar(30) default NULL,
msg text default NULL,
senttime varchar(100) default NULL,
receivedtime varchar(100) default
NULL,
reference varchar(100) default
NULL,
status varchar(20) default NULL,
msgtype varchar(160) default NULL,
operator varchar(100) default NULL,
errormsg varchar(250) default NULL,
PRIMARY KEY (id)
) charset=utf8;
ALTER TABLE ozekimessageout ADD
INDEX (id);
```

**Figure 2** - CREATE database tables

Please maintain the 'id' field in your database to support faster SQL updates.

If you modify database tables. For example if you add additional columns, please review the SQL templates used by Ozeki SMS Gateway. Please make sure that they are compatible with the customized database table layout.

# MySQL basics

***The following commands can be used to setup a MySQL server on an Ubuntu Linux distribution to be used with Ozeki 10. These commands should be executed in a root command shell. To get a root shell on the Ubuntu desktop, open the terminal app and type "sudo bash".***

## Install the mysql server

```
# apt-get install mysql-server
```

## Login to the mysql server

```
# mysql -u root
```

## Create the ozeki database

```
mysql> create database ozeki;
```

## Create the ozeki user with password abc123

```
mysql> create user ozeki identified by 'abc123';
```

## Let the ozeki user access to ozeki database

```
mysql> grant all privileges on ozeki.* to 'ozeki';
```

## Configure remote access for MySQL

```
# vi /etc/mysql/mysql.conf.d/mysqld.conf
```

change the bind address to 0.0.0.0

## Restart MySQL

```
# systemctl restart mysql
```

## Check if mysql listens on port 3306

```
# netstat -tulnp | grep mysql
```

## Allow port 3306 in your firewall

```
# ufw allow 3306/tcp
```

## Find out the IP address of your Linux box

```
# ifconfig
```

# Send SMS from MySQL (part 1/4)
## MySQL Installation

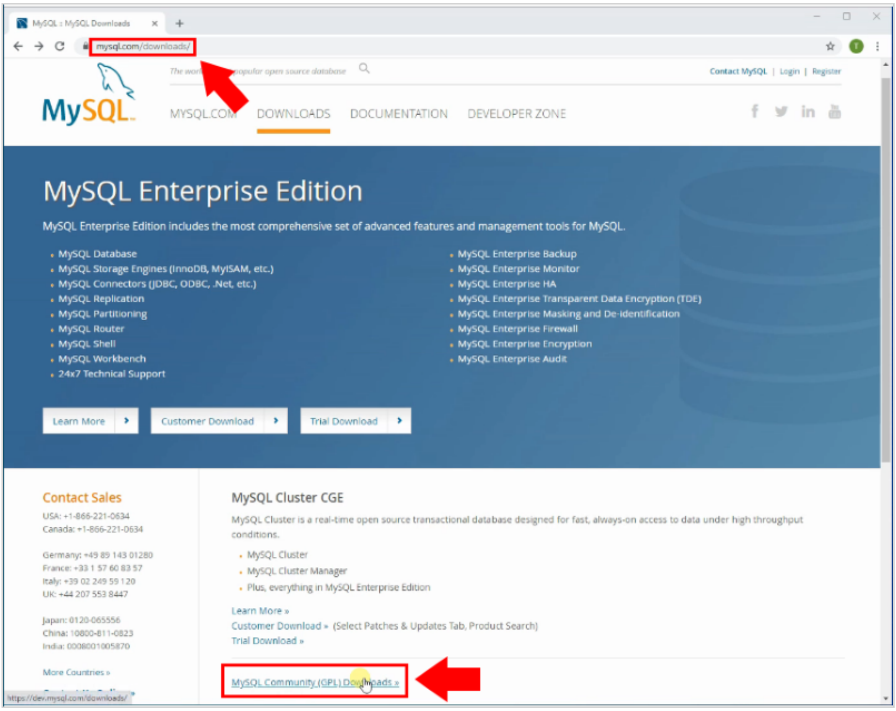
**In this video series you will see how to connect Ozeki SMS Gateway and MySQL database server for SMS messaging. The first video shows how to download and install a MySQL.**

**Video content**

1. Download MySQL
2. Install MySQL

---

Please jump to the next video or if you scroll down you can find screenshots that describe how to download and install **MySQL** database server.



MySQL Enterprise Edition

MySQL Enterprise Edition includes the most comprehensive set of advanced features and management tools for MySQL.

- MySQL Database
- MySQL Storage Engines (InnoDB, MyISAM, etc.)
- MySQL Connectors (JDBC, ODBC, .Net, etc.)
- MySQL Replication
- MySQL Partitioning
- MySQL Router
- MySQL Shell
- MySQL Workbench
- 24x7 Technical Support
- MySQL Enterprise Backup
- MySQL Enterprise Monitor
- MySQL Enterprise HA
- MySQL Enterprise Transparent Data Encryption (TDE)
- MySQL Enterprise Masking and De-identification
- MySQL Enterprise Firewall
- MySQL Enterprise Encryption
- MySQL Enterprise Audit

Contact Sales

USA: +1-866-221-0634
Canada: +1-866-221-0634

Germany: +49 89 143 01280
France: +33 1 57 60 83 57
Italy: +39 02 249 59 120
UK: +44 207 553 8447

Japan: 0120-065556
China: 10800-811-0823
India: 0008001005870

More Countries »

MySQL Cluster CGE

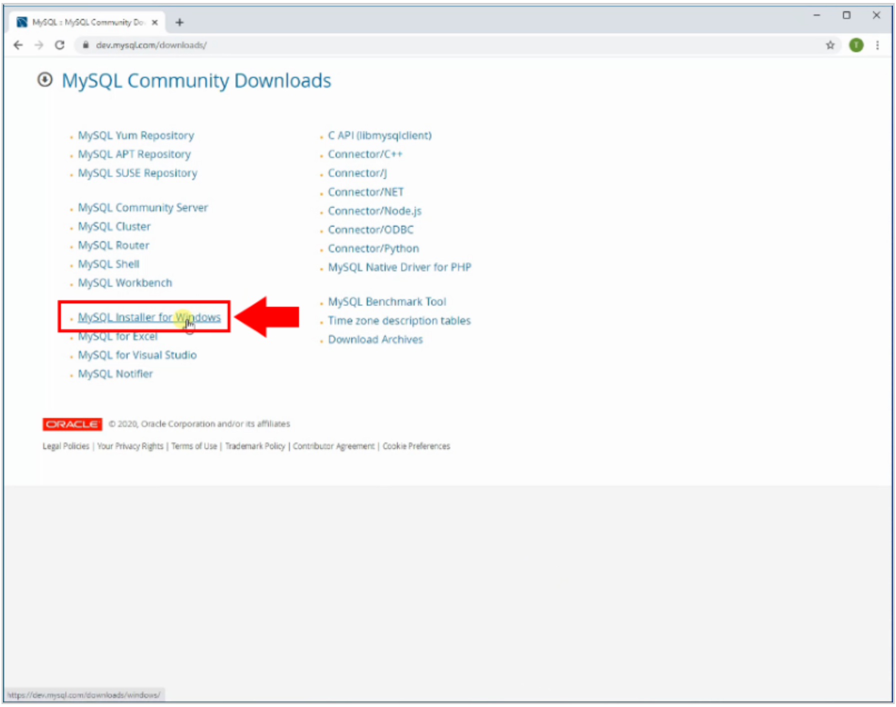
MySQL Cluster is a real-time open source transactional database designed for fast, always-on access to data under high throughput conditions.

- MySQL Cluster
- MySQL Cluster Manager
- Plus, everything in MySQL Enterprise Edition

Learn More »
Customer Download » (Select Patches & Updates Tab, Product Search)
Trial Download »

MySQL Community (GPL) Downloads »

**Figure 1** - MYSQL webpage


MySQL :: MySQL Community Do
dev.mysql.com/downloads/
https://dev.mysql.com/downloads/windows/

MySQL Community Downloads

- MySQL Yum Repository
- MySQL APT Repository
- MySQL SUSE Repository

- MySQL Community Server
- MySQL Cluster
- MySQL Router
- MySQL Shell
- MySQL Workbench

- MySQL Installer for Windows
- MySQL for Excel
- MySQL for Visual Studio
- MySQL Notifier

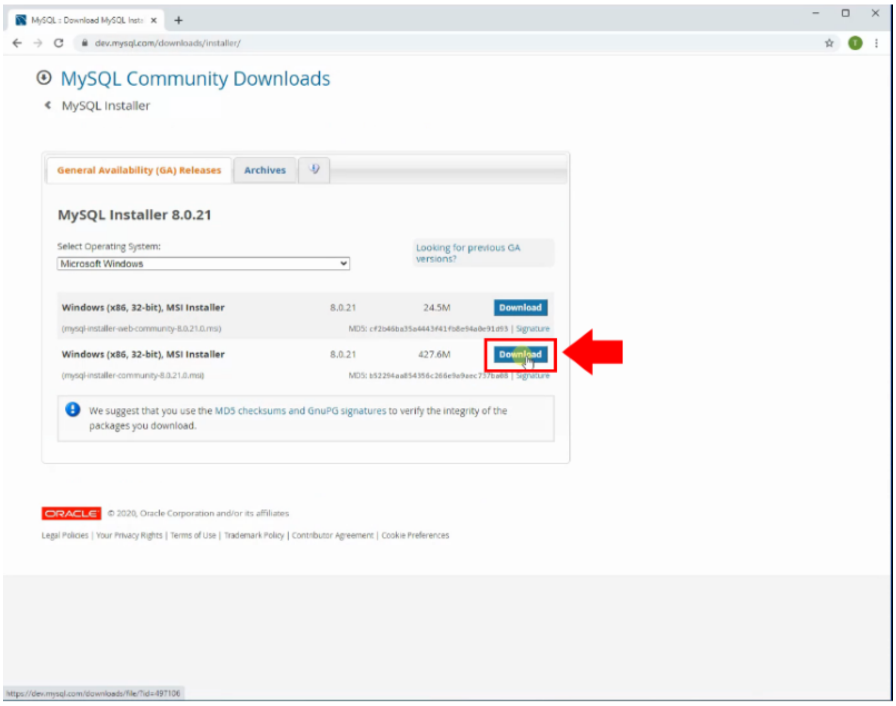
- C API (libmysqlclient)
- Connector/C++
- Connector/J
- Connector/NET
- Connector/Node.js
- Connector/ODBC
- Connector/Python
- MySQL Native Driver for PHP

- MySQL Benchmark Tool
- Time zone description tables
- Download Archives

ORACLE © 2020, Oracle Corporation and/or its affiliates

Legal Policies | Your Privacy Rights | Terms of Use | Trademark Policy | Contributor Agreement | Cookie Preferences

**Figure 2** - Select Windows installer


MySQL :: Download MySQL Insta
dev.mysql.com/downloads/installer/
General Availability (GA) Releases
Archives
Select Operating System:
Microsoft Windows
Looking for previous GA versions?
Download
Download
Signature
Signature
Legal Policies | Your Privacy Rights | Terms of Use | Trademark Policy | Contributor Agreement | Cookie Preferences
https://dev.mysql.com/downloads/file/?id=497106

MySQL Community Downloads

MySQL Installer

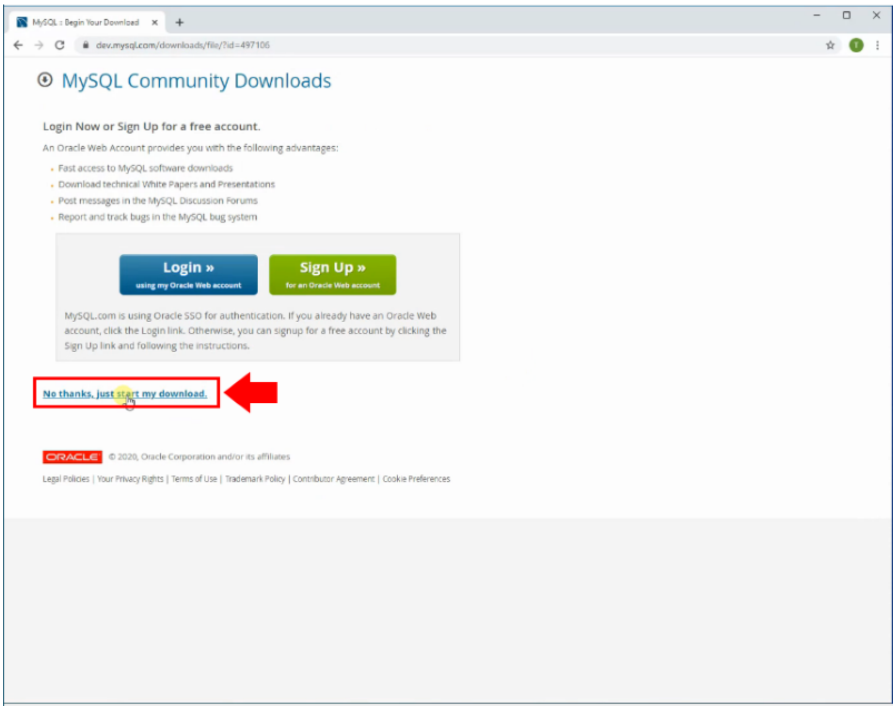
MySQL Installer 8.0.21

| | | |
|---|---|---|
| Windows (x86, 32-bit), MSI Installer<br>(mysql-installer-web-community-8.0.21.0.msi) | 8.0.21 | 24.5M<br>MD5: cf2b46ba35a4443f41fb8e94a0e91d93 |
| Windows (x86, 32-bit), MSI Installer<br>(mysql-installer-community-8.0.21.0.msi) | 8.0.21 | 427.6M<br>MD5: b52294aa854356c266e9a9aec737ba88 |

We suggest that you use the MD5 checksums and GnuPG signatures to verify the integrity of the packages you download.

© 2020, Oracle Corporation and/or its affiliates

**Figure 3** - Download the installer


MySQL :: Begin Your Download
dev.mysql.com/downloads/file/?id=497106
Login »
using my Oracle Web account
Sign Up »
for an Oracle Web account
No thanks, just start my download.
Legal Policies | Your Privacy Rights | Terms of Use | Trademark Policy | Contributor Agreement | Cookie Preferences

MySQL Community Downloads

Login Now or Sign Up for a free account.

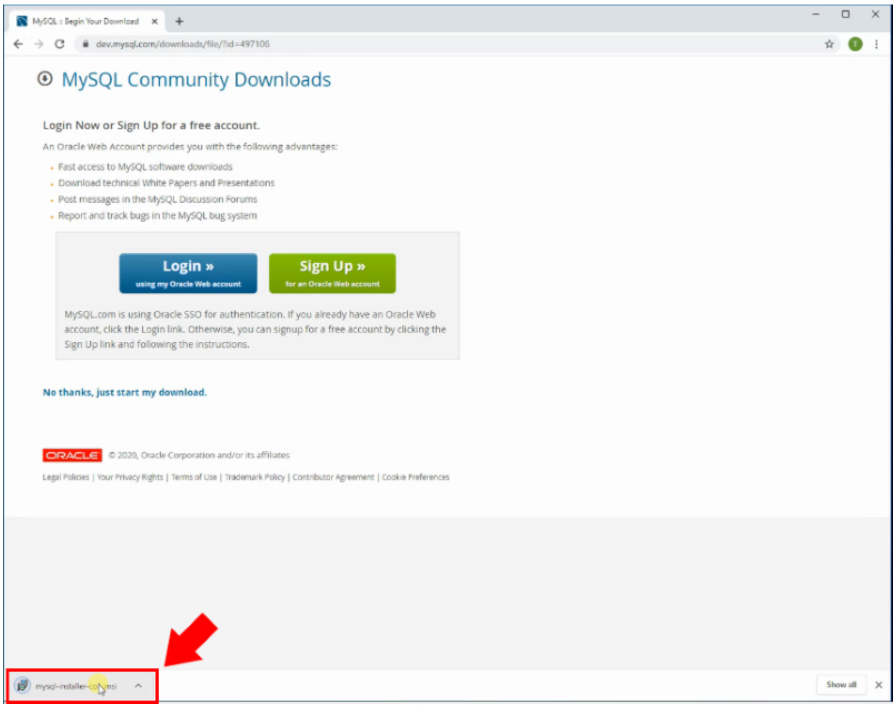
An Oracle Web Account provides you with the following advantages:

- Fast access to MySQL software downloads
- Download technical White Papers and Presentations
- Post messages in the MySQL Discussion Forums
- Report and track bugs in the MySQL bug system

MySQL.com is using Oracle SSO for authentication. If you already have an Oracle Web account, click the Login link. Otherwise, you can signup for a free account by clicking the Sign Up link and following the instructions.

© 2020, Oracle Corporation and/or its affiliates

**Figure 4** - Start Download



MySQL :: Begin Your Download
dev.mysql.com/downloads/file/?id=497106

MySQL Community Downloads

Login Now or Sign Up for a free account.

An Oracle Web Account provides you with the following advantages:

- Fast access to MySQL software downloads
- Download technical White Papers and Presentations
- Post messages in the MySQL Discussion Forums
- Report and track bugs in the MySQL bug system

Login »
using my Oracle Web account
Sign Up »
for an Oracle Web account

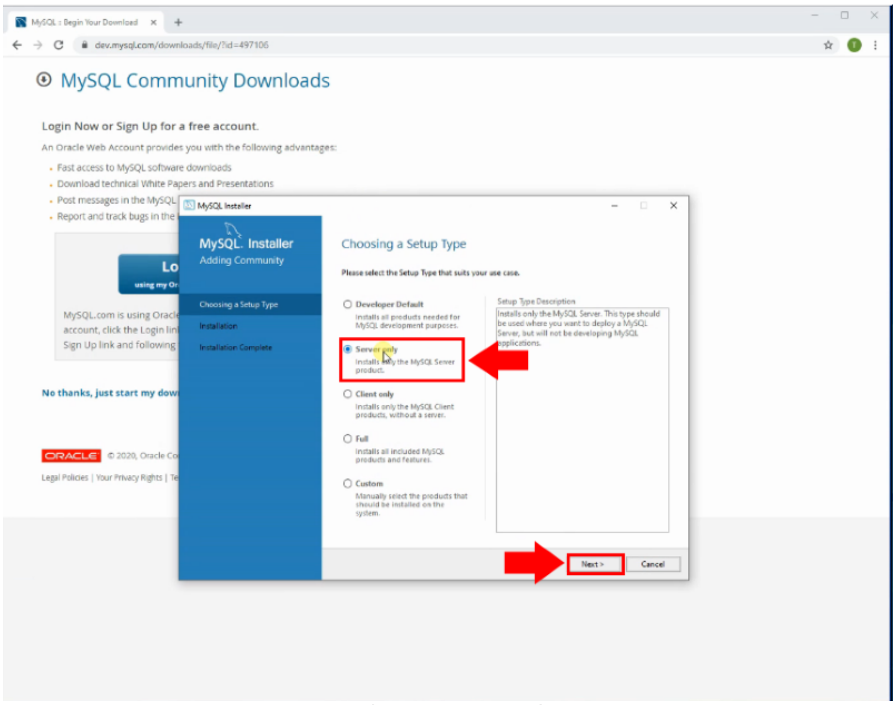
MySQL.com is using Oracle SSO for authentication. If you already have an Oracle Web account, click the Login link. Otherwise, you can signup for a free account by clicking the Sign Up link and following the instructions.

No thanks, just start my download.

© 2020, Oracle Corporation and/or its affiliates

Legal Policies | Your Privacy Rights | Terms of Use | Trademark Policy | Contributor Agreement | Cookie Preferences

**Figure 5 -** Open the Installer



Choosing a Setup Type

Please select the Setup Type that suits your use case.

○ Developer Default
Installs all products needed for MySQL development purposes.

● Server only
Installs only the MySQL Server product.

○ Client only
Installs only the MySQL Client products, without a server.

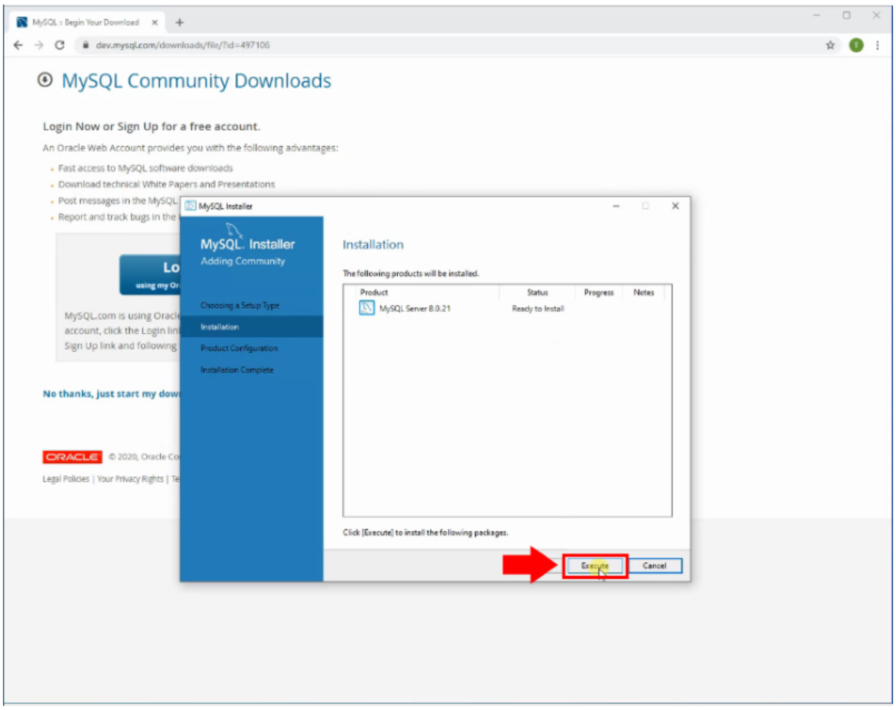
○ Full
Installs all included MySQL products and features.

○ Custom
Manually select the products that should be installed on the system.

Setup Type Description

Installs only the MySQL Server. This type should be used where you want to deploy a MySQL Server, but will not be developing MySQL applications.

**Figure 6 -** Select install Server only


MySQL :: Begin Your Download
dev.mysql.com/downloads/file/?id=497106

MySQL Community Downloads

Login Now or Sign Up for a free account.

An Oracle Web Account provides you with the following advantages:

- Fast access to MySQL software downloads
- Download technical White Papers and Presentations
- Post messages in the MySQL
- Report and track bugs in the

MySQL.com is using Oracle account, click the Login lin Sign Up link and following

No thanks, just start my dow

© 2020, Oracle Co

Legal Policies | Your Privacy Rights | Te

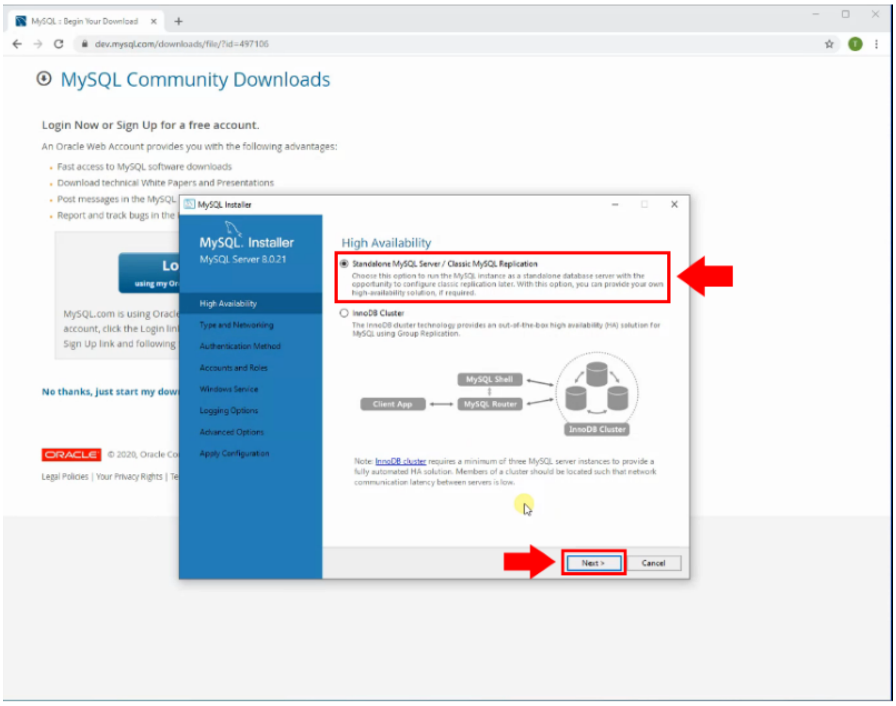
Installation

The following products will be installed.

| Product | Status | Progress | Notes |
|---|---|---|---|
| MySQL Server 8.0.21 | Ready to Install | | |

Click [Execute] to install the following packages.

**Figure 7 -** Execute installation


MySQL :: Begin Your Download
dev.mysql.com/downloads/file/?id=497106

MySQL Community Downloads

Login Now or Sign Up for a free account.

An Oracle Web Account provides you with the following advantages:

- Fast access to MySQL software downloads
- Download technical White Papers and Presentations
- Post messages in the MySQL
- Report and track bugs in the

MySQL.com is using Oracle account, click the Login lin Sign Up link and following

No thanks, just start my dow

© 2020, Oracle Co

Legal Policies | Your Privacy Rights | Te

High Availability

● Standalone MySQL Server / Classic MySQL Replication

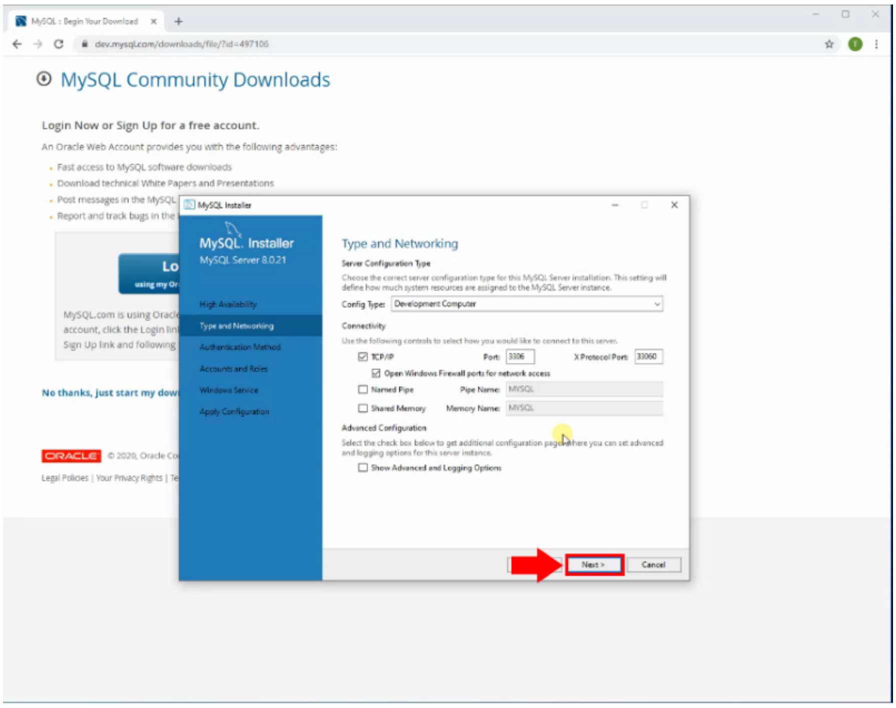
Choose this option to run the MySQL instance as a standalone database server with the opportunity to configure classic replication later. With this option, you can provide your own high-availability solution, if required.

○ InnoDB Cluster

The InnoDB cluster technology provides an out-of-the-box high availability (HA) solution for MySQL using Group Replication.

Note: InnoDB cluster requires a minimum of three MySQL server instances to provide a fully automated HA solution. Members of a cluster should be located such that network communication latency between servers is low.

**Figure 8 -** Configure Standalone server



Type and Networking

Server Configuration Type

Choose the correct server configuration type for this MySQL Server installation. This setting will define how much system resources are assigned to the MySQL Server instance.

Config Type: Development Computer

Connectivity

Use the following controls to select how you would like to connect to this server.

☑ TCP/IP Port: 3306 X Protocol Port: 33060

☑ Open Windows Firewall ports for network access

□ Named Pipe Pipe Name: MYSQL

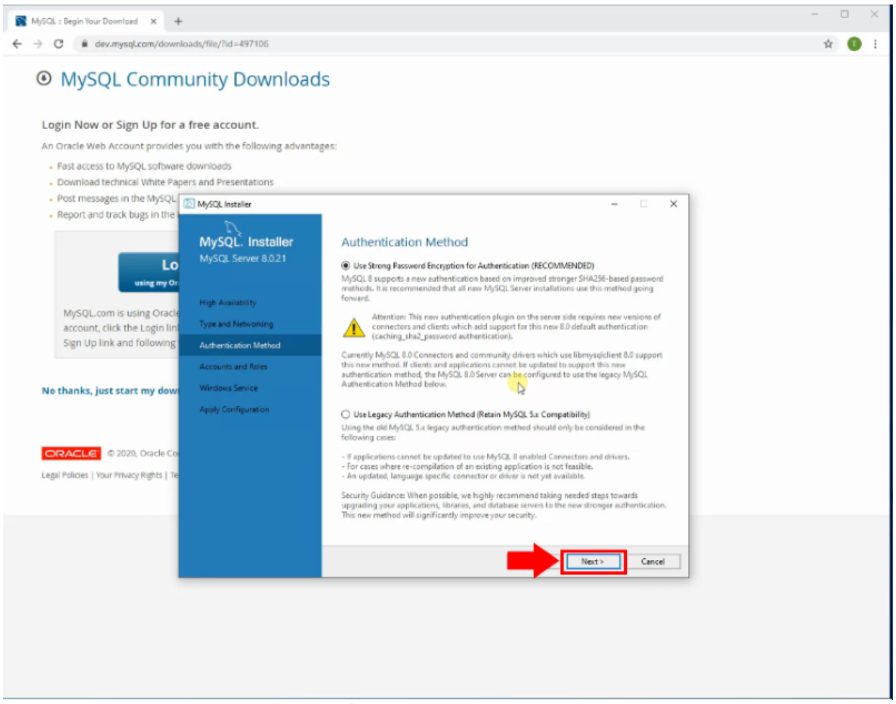
□ Shared Memory Memory Name: MYSQL

Advanced Configuration

Select the check box below to get additional configuration pages where you can set advanced and logging options for this server instance.

□ Show Advanced and Logging Options

**Figure 9** - Configure Type and Network settings



Authentication Method

● Use Strong Password Encryption for Authentication (RECOMMENDED)

MySQL 8 supports a new authentication based on improved stronger SHA256-based password methods. It is recommended that all new MySQL Server installations use this method going forward.

Attention: This new authentication plugin on the server side requires new versions of connectors and clients which add support for this new 8.0 default authentication (caching_sha2_password authentication).

Currently MySQL 8.0 Connectors and community drivers which use libmysqlclient 8.0 support this new method. If clients and applications cannot be updated to support this new authentication method, the MySQL 8.0 Server can be configured to use the legacy MySQL Authentication Method below.

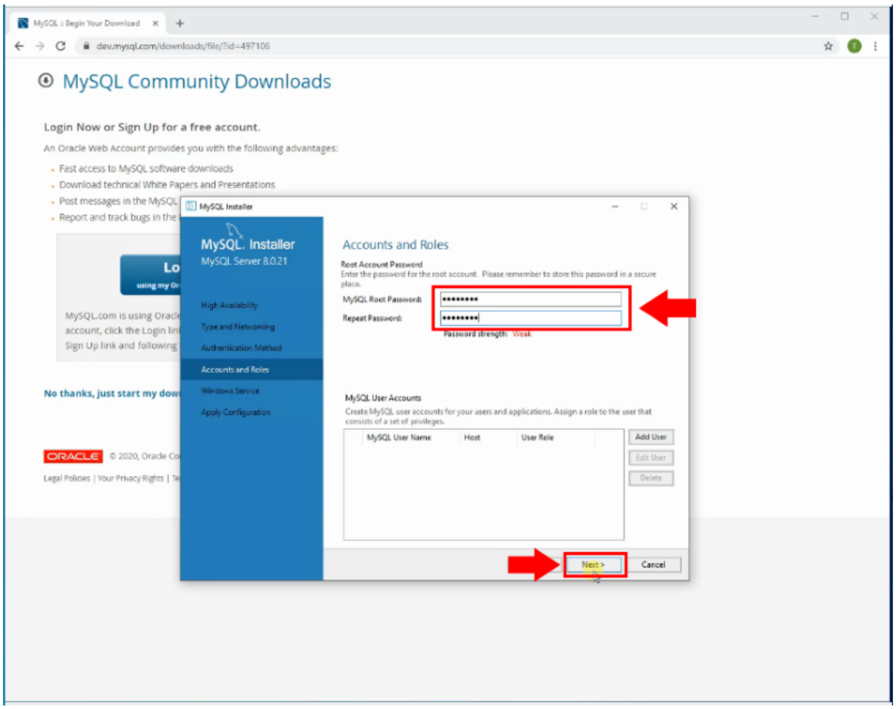
○ Use Legacy Authentication Method (Retain MySQL 5.x Compatibility)

Using the old MySQL 5.x legacy authentication method should only be considered in the following cases:

- If applications cannot be updated to use MySQL 8 enabled Connectors and drivers.
- For cases where re-compilation of an existing application is not feasible.
- An updated, language specific connector or driver is not yet available.

Security Guidance: When possible, we highly recommend taking needed steps towards upgrading your applications, libraries, and database servers to the new stronger authentication. This new method will significantly improve your security.

**Figure 10** - Select Authentication method



MySQL :: Begin Your Download
dev.mysql.com/downloads/file/?id=497106
MySQL Community Downloads
Login Now or Sign Up for a free account.
An Oracle Web Account provides you with the following advantages:
Fast access to MySQL software downloads
Download technical White Papers and Presentations
No thanks, just start my dow
MySQL Installer
MySQL. Installer
MySQL Server 8.0.21
High Availability
Type and Networking
Authentication Method
Accounts and Roles
Windows Service
Apply Configuration
Add User
Edit User
Delete
Next >
Cancel

Accounts and Roles

Root Account Password
Enter the password for the root account. Please remember to store this password in a secure place.

MySQL Root Password: ••••••••

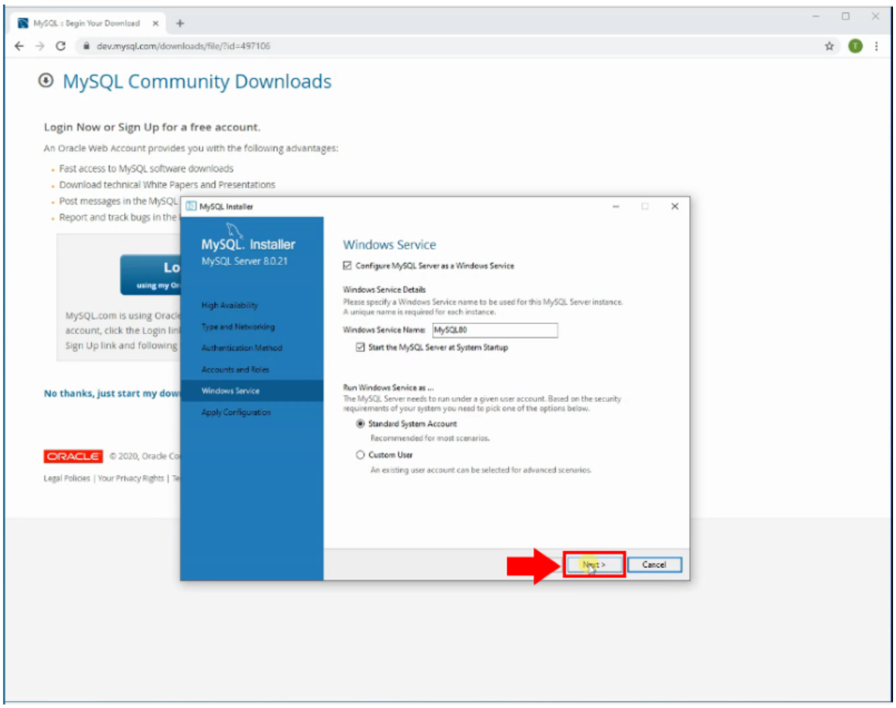
Repeat Password: ••••••••

Password strength: Weak

MySQL User Accounts
Create MySQL user accounts for your users and applications. Assign a role to the user that consists of a set of privileges.

| MySQL User Name | Host | User Role |
|---|---|---|

**Figure 11** - Define Root password



MySQL :: Begin Your Download
dev.mysql.com/downloads/file/?id=497106
MySQL Community Downloads
Login Now or Sign Up for a free account.
An Oracle Web Account provides you with the following advantages:
Fast access to MySQL software downloads
Download technical White Papers and Presentations
No thanks, just start my dow
MySQL Installer
MySQL. Installer
MySQL Server 8.0.21
High Availability
Type and Networking
Authentication Method
Accounts and Roles
Windows Service
Apply Configuration
Next >
Cancel

Windows Service

☑ Configure MySQL Server as a Windows Service

Windows Service Details
Please specify a Windows Service name to be used for this MySQL Server instance. A unique name is required for each instance.

Windows Service Name: MySQL80

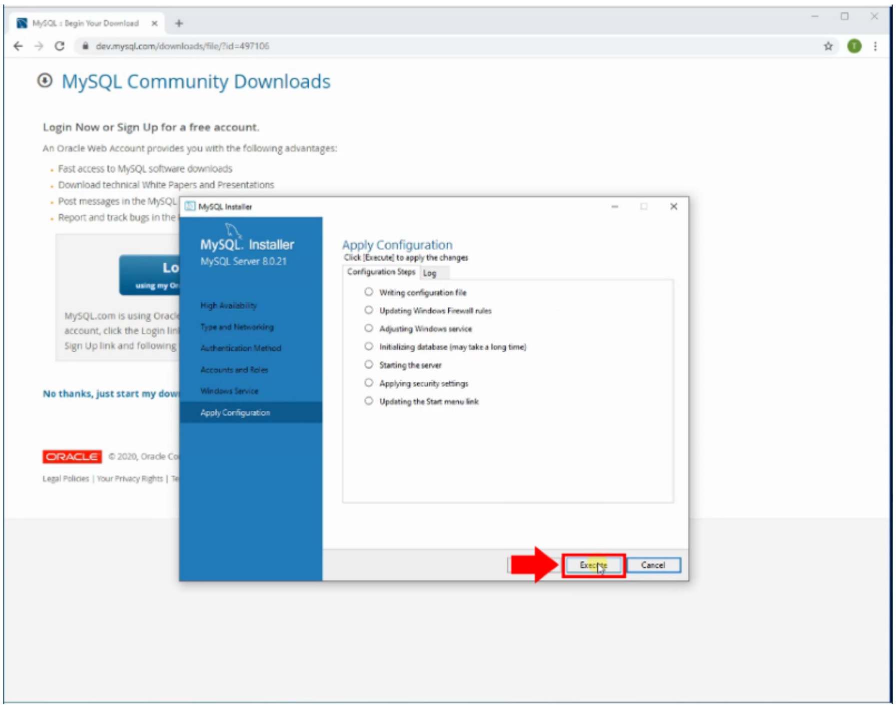
☑ Start the MySQL Server at System Startup

Run Windows Service as ...
The MySQL Server needs to run under a given user account. Based on the security requirements of your system you need to pick one of the options below.

● Standard System Account
Recommended for most scenarios.

○ Custom User
An existing user account can be selected for advanced scenarios.

**Figure 12** - Configure Mysql service



MySQL Community Downloads

Login Now or Sign Up for a free account.

An Oracle Web Account provides you with the following advantages:

- Fast access to MySQL software downloads
- Download technical White Papers and Presentations
- Post messages in the MySQL
- Report and track bugs in the

MySQL.com is using Oracle account, click the Login lin Sign Up link and following

No thanks, just start my dow

© 2020, Oracle Co

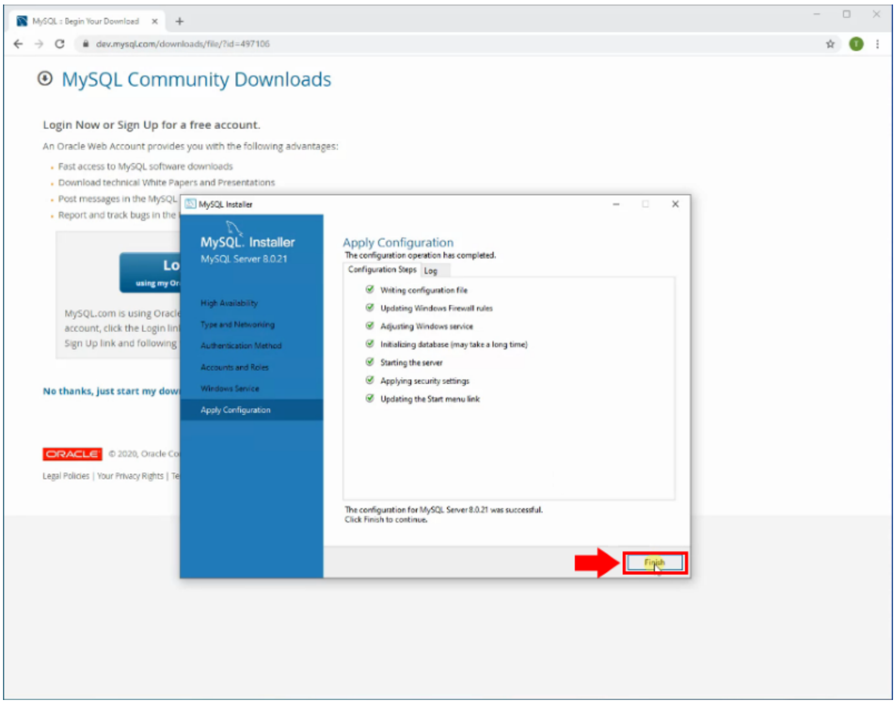
Legal Policies | Your Privacy Rights | Te

Apply Configuration

Click [Execute] to apply the changes

○ Writing configuration file
○ Updating Windows Firewall rules
○ Adjusting Windows service
○ Initializing database (may take a long time)
○ Starting the server
○ Applying security settings
○ Updating the Start menu link

**Figure 13 -** Apply configuration



MySQL Community Downloads

Login Now or Sign Up for a free account.

An Oracle Web Account provides you with the following advantages:

- Fast access to MySQL software downloads
- Download technical White Papers and Presentations
- Post messages in the MySQL
- Report and track bugs in the

MySQL.com is using Oracle account, click the Login lin Sign Up link and following

No thanks, just start my dow

© 2020, Oracle Co

Legal Policies | Your Privacy Rights | Te

Apply Configuration

The configuration operation has completed.

✓ Writing configuration file
✓ Updating Windows Firewall rules
✓ Adjusting Windows service
✓ Initializing database (may take a long time)
✓ Starting the server
✓ Applying security settings
✓ Updating the Start menu link

The configuration for MySQL Server 8.0.21 was successful.
Click Finish to continue.

**Figure 14 -** Finish installation

# Send SMS from MySQL (part 2/4)

## Create Database Tables

**In this video series you can see how to connect Ozeki SMS Gateway and MySQL database server for SMS messaging. This video shows how to create the proper database table structure by using the command line of MySQL.**

**Video content**

1. Connect to MySQL
2. Copy CREATE TABLE statement

Please **scroll down to copy the SQL statements** (**Figure 1**) used in the video. If you have created the database in MySQL, you can jump to the next video.

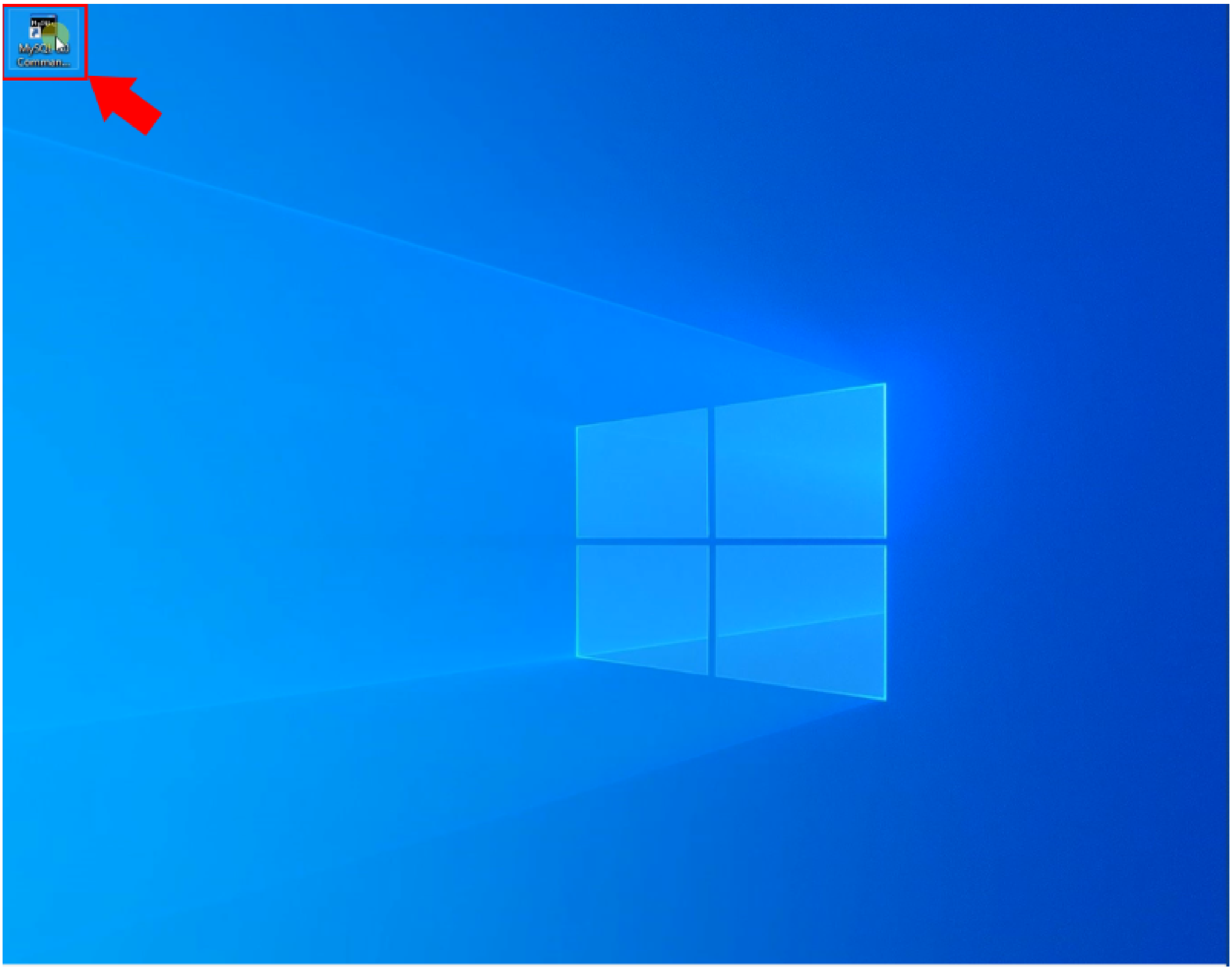
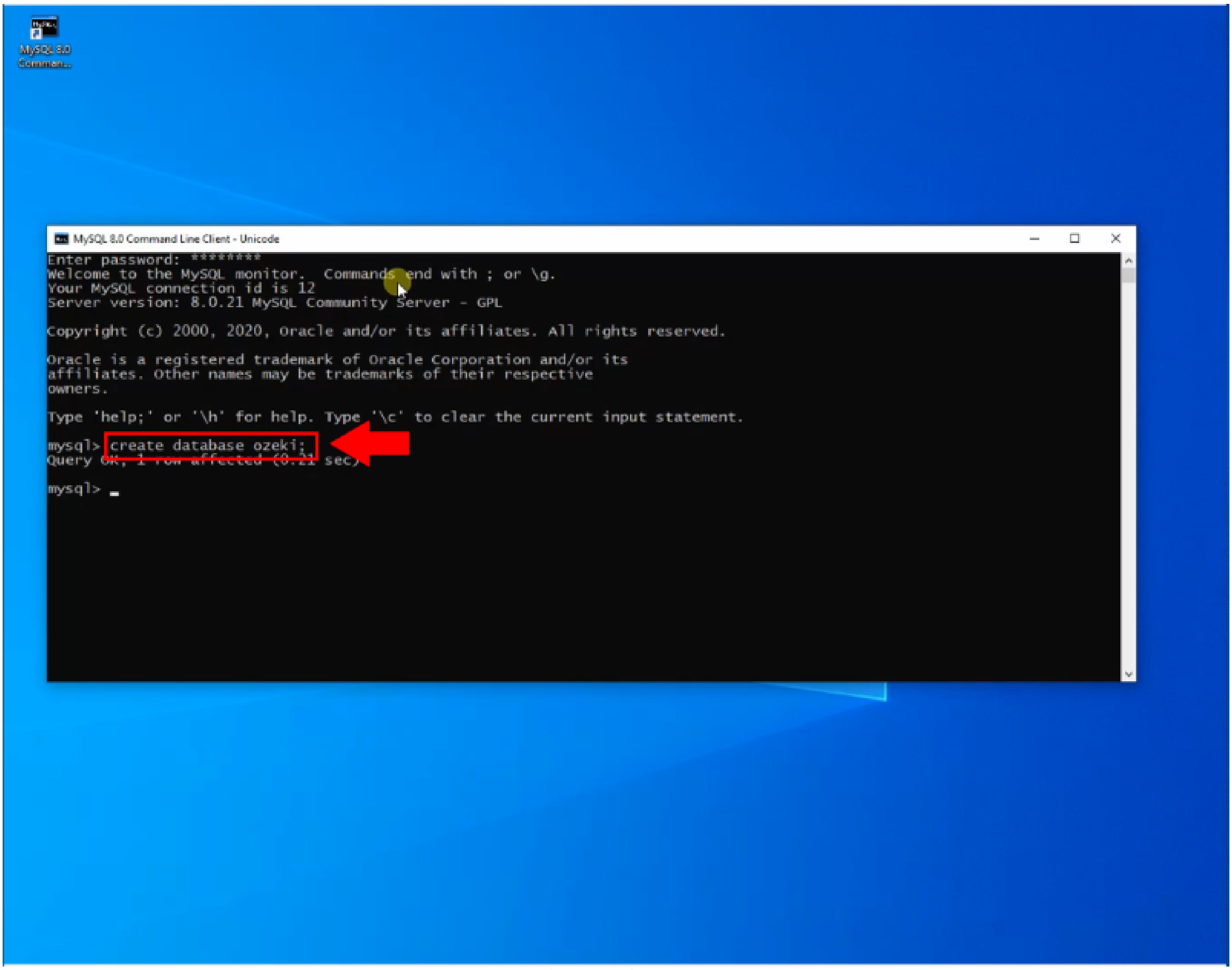


Figure 1 - Start Mysql client



```
Enter password: ********
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 12
Server version: 8.0.21 MySQL Community Server - GPL

Copyright (c) 2000, 2020, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> create database ozeki;
Query OK, 1 row affected (0.21 sec)

mysql> _
```

Figure 2 - Create Ozeki database

## MySQL CREATE TABLE statements to copy:

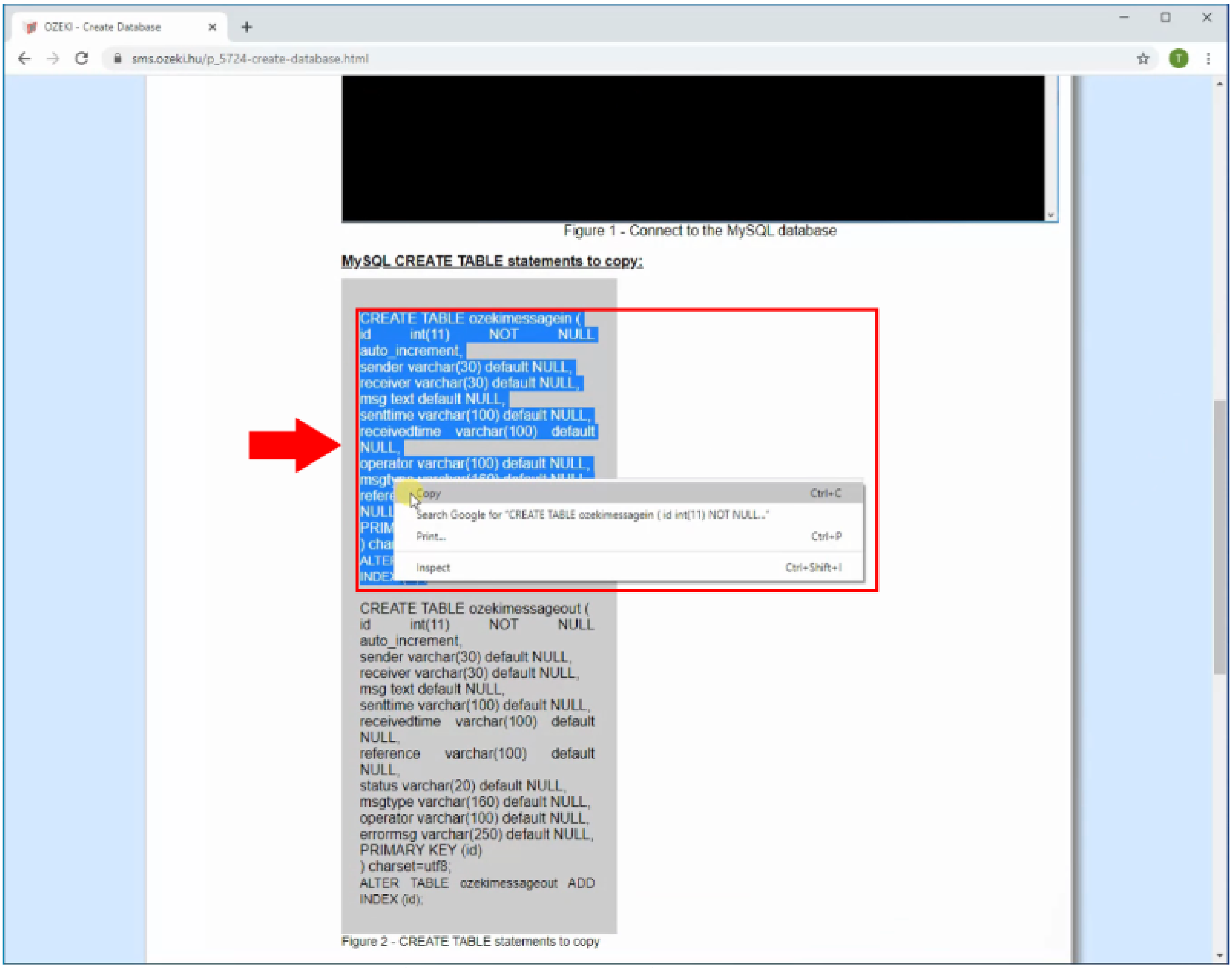
```
CREATE DATABASE Ozeki;
```

Figure 3 - Create database statement

## MySQL CREATE TABLE statements to copy:

```
  id int(11) NOT NULL auto_increment,<br>
  sender varchar(30) default NULL,<br>
  receiver varchar(30) default NULL,<br>
  msg text default NULL,<br>
  senttime varchar(100) default NULL,<br>
  receivedtime varchar(100) default NULL,<br>
  operator varchar(100) default NULL,<br>
  msgtype varchar(160) default NULL,<br>
  reference varchar(100) default NULL,<br>
  PRIMARY KEY (id)<br>
  ) charset=utf8;
  <br>
  <span class="syntax">
  <span class="syntax_alpha syntax_alpha_reservedWord">ALTER</span>
  <span class="syntax_alpha syntax_alpha_reservedWord">TABLE</span>
  <span class="syntax_quote syntax_quote_backtick">ozekimessagein</span>
  <span class="syntax_alpha syntax_alpha_reservedWord">ADD</span>
  <span class="syntax_alpha syntax_alpha_reservedWord">INDEX</span>
  <span class="syntax_punct syntax_punct_bracket_open_round">(id)</span> ;
</span>
<br>
<br>

CREATE TABLE ozekimessageout (<br>
  id int(11) NOT NULL auto_increment,<br>
  sender varchar(30) default NULL,<br>
  receiver varchar(30) default NULL,<br>
  msg text default NULL,<br>
  senttime varchar(100) default NULL,<br>
  receivedtime varchar(100) default NULL,<br>
  reference varchar(100) default NULL,<br>
  status varchar(20) default NULL,<br>
  msgtype varchar(160) default NULL,<br>
  operator varchar(100) default NULL,<br>
  errormsg varchar(250) default NULL,<br>
  PRIMARY KEY (id)<br>) charset=utf8;<br><span class="syntax">
  <span class="syntax_alpha syntax_alpha_reservedWord">ALTER</span>
  <span class="syntax_alpha syntax_alpha_reservedWord">TABLE</span>
  <span class="syntax_quote syntax_quote_backtick">ozekimessageout</span>
  <span class="syntax_alpha syntax_alpha_reservedWord">ADD</span>
  <span class="syntax_alpha syntax_alpha_reservedWord">INDEX</span>
  <span class="syntax_punct syntax_punct_bracket_open_round">(id);</span>
</span>
```

Figure 4 - CREATE TABLE statements to copy



OZEKI - Create Database
sms.ozeki.hu/p_5724-create-database.html
Copy Ctrl+C
Search Google for "CREATE TABLE ozekimessagein ( id int(11) NOT NULL..."
Print... Ctrl+P
Inspect Ctrl+Shift+I

Figure 1 - Connect to the MySQL database

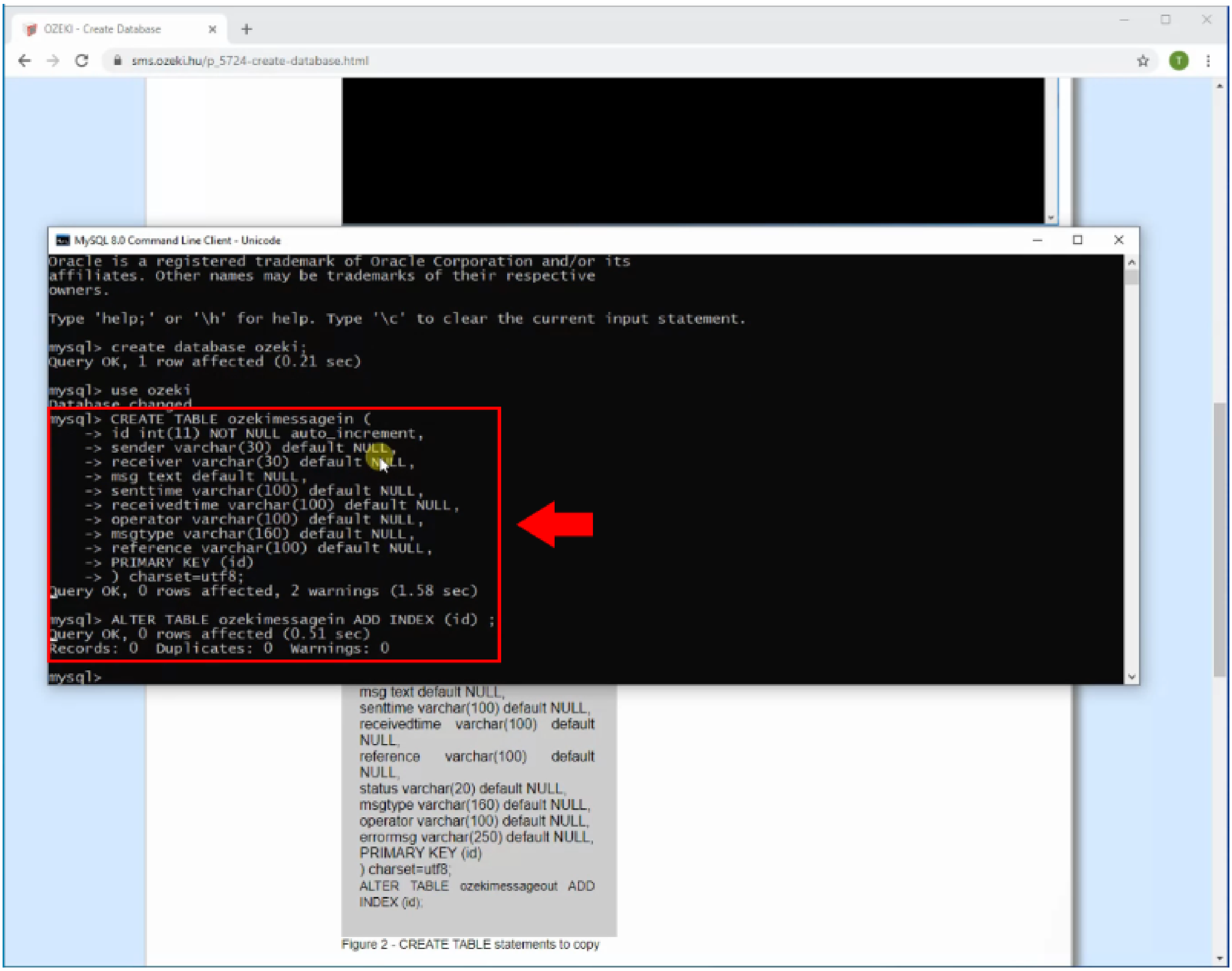
MySQL CREATE TABLE statements to copy:

```
CREATE TABLE ozekimessagein (
id        int(11)        NOT        NULL
auto_increment,
sender varchar(30) default NULL,
receiver varchar(30) default NULL,
msg text default NULL,
senttime varchar(100) default NULL,
receivedtime    varchar(100)    default
NULL,
operator varchar(100) default NULL,
msgt
refere
NULL
PRIM
) char
ALTER
INDEX
```

```
CREATE TABLE ozekimessageout (
id        int(11)        NOT        NULL
auto_increment,
sender varchar(30) default NULL,
receiver varchar(30) default NULL,
msg text default NULL,
senttime varchar(100) default NULL,
receivedtime    varchar(100)    default
NULL,
reference      varchar(100)      default
NULL,
status varchar(20) default NULL,
msgtype varchar(160) default NULL,
operator varchar(100) default NULL,
errormsg varchar(250) default NULL,
PRIMARY KEY (id)
) charset=utf8;
ALTER   TABLE   ozekimessageout   ADD
INDEX (id);
```

Figure 2 - CREATE TABLE statements to copy

Figure 3 - Copy ozekimessagein CREATE TABLE statement



OZEKI - Create Database
sms.ozeki.hu/p_5724-create-database.html
MySQL 8.0 Command Line Client - Unicode

```
Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> create database ozeki;
Query OK, 1 row affected (0.21 sec)

mysql> use ozeki
Database changed
mysql> CREATE TABLE ozekimessagein (
    -> id int(11) NOT NULL auto_increment,
    -> sender varchar(30) default NULL,
    -> receiver varchar(30) default NULL,
    -> msg text default NULL,
    -> senttime varchar(100) default NULL,
    -> receivedtime varchar(100) default NULL,
    -> operator varchar(100) default NULL,
    -> msgtype varchar(160) default NULL,
    -> reference varchar(100) default NULL,
    -> PRIMARY KEY (id)
    -> ) charset=utf8;
Query OK, 0 rows affected, 2 warnings (1.58 sec)

mysql> ALTER TABLE ozekimessagein ADD INDEX (id) ;
Query OK, 0 rows affected (0.51 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql>
```

```
msg text default NULL,
senttime varchar(100) default NULL,
receivedtime varchar(100) default NULL,
reference varchar(100) default NULL,
status varchar(20) default NULL,
msgtype varchar(160) default NULL,
operator varchar(100) default NULL,
errormsg varchar(250) default NULL,
PRIMARY KEY (id)
) charset=utf8;
ALTER TABLE ozekimessageout ADD INDEX (id);
```

Figure 2 - CREATE TABLE statements to copy

Figure 4 - Paste ozekimessagein CREATE TABLE statements and run it on the MySQL database server

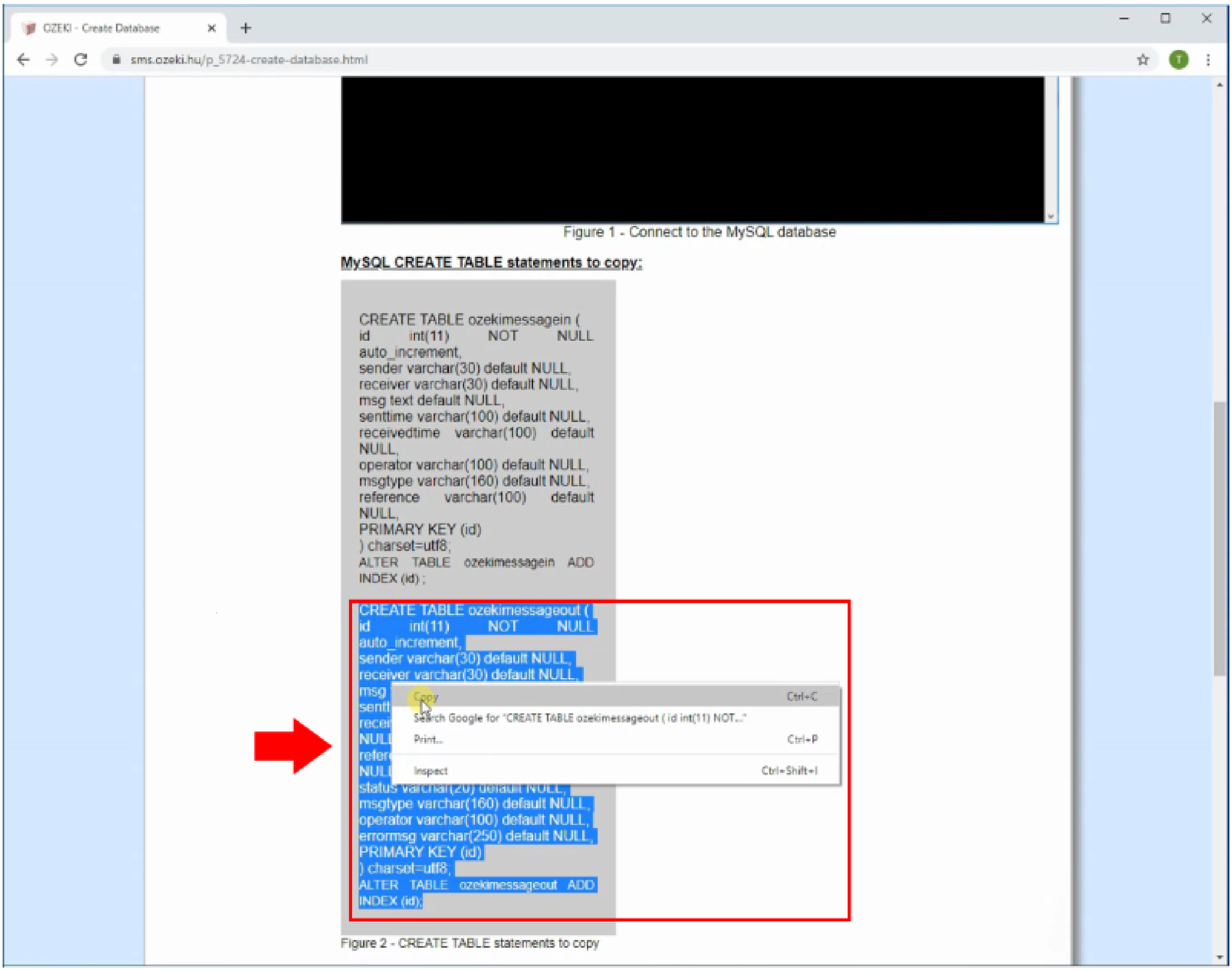

OZEKI - Create Database
sms.ozeki.hu/p_5724-create-database.html
Copy Ctrl+C
Search Google for "CREATE TABLE ozekimessageout ( id int(11) NOT..."
Print... Ctrl+P
Inspect Ctrl+Shift+I

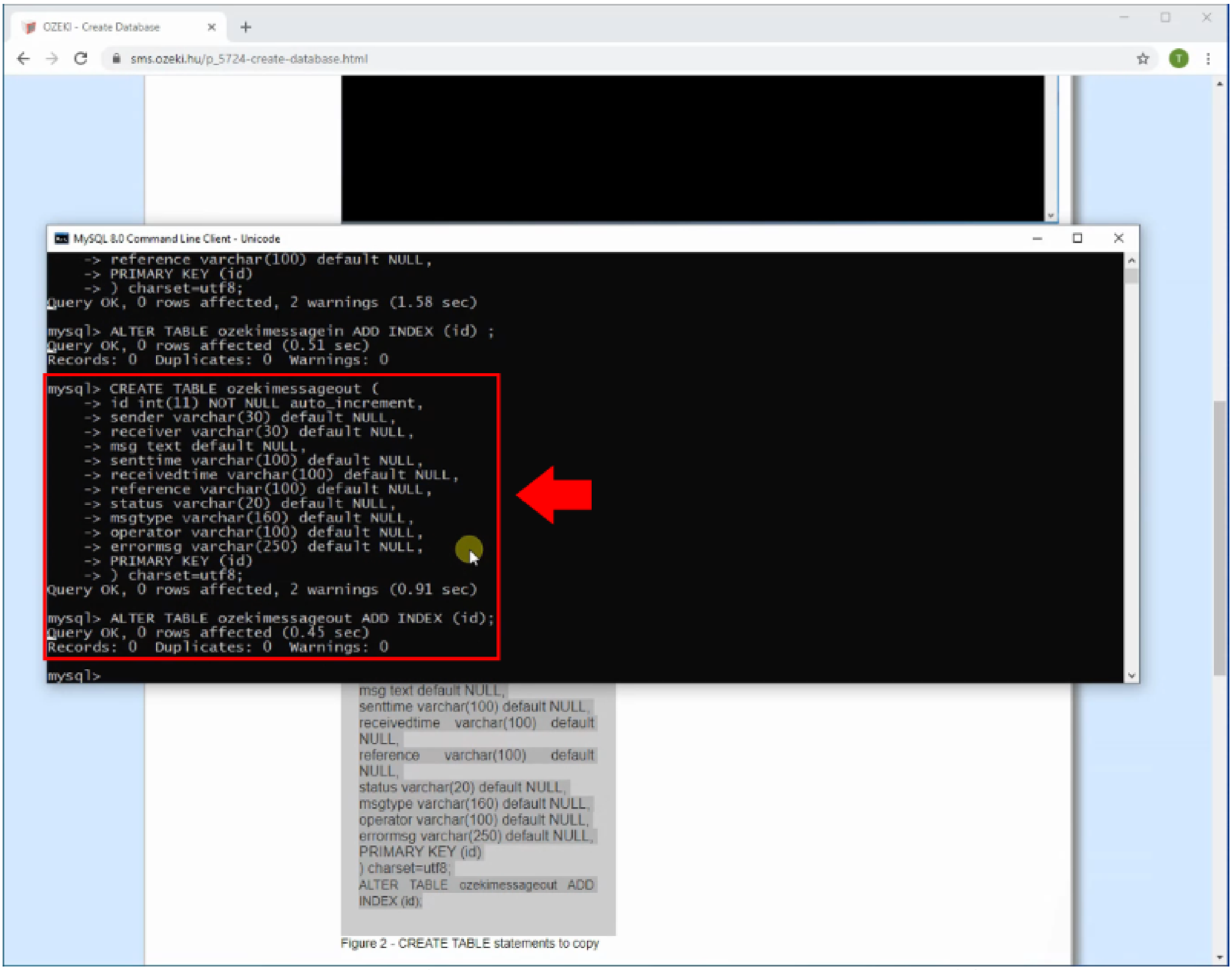
Figure 1 - Connect to the MySQL database

**MySQL CREATE TABLE statements to copy:**

```
CREATE TABLE ozekimessagein (
id        int(11)        NOT        NULL
auto_increment,
sender varchar(30) default NULL,
receiver varchar(30) default NULL,
msg text default NULL,
senttime varchar(100) default NULL,
receivedtime   varchar(100)   default
NULL,
operator varchar(100) default NULL,
msgtype varchar(160) default NULL,
reference     varchar(100)     default
NULL,
PRIMARY KEY (id)
) charset=utf8;
ALTER   TABLE   ozekimessagein   ADD
INDEX (id) ;

CREATE TABLE ozekimessageout (
id        int(11)        NOT        NULL
auto_increment,
sender varchar(30) default NULL,
receiver varchar(30) default NULL,
msg
sentt
recei
NULL
refer
NULL
status varchar(20) default NULL,
msgtype varchar(160) default NULL,
operator varchar(100) default NULL,
errormsg varchar(250) default NULL,
PRIMARY KEY (id)
) charset=utf8;
ALTER   TABLE   ozekimessageout   ADD
INDEX (id);
```

Figure 2 - CREATE TABLE statements to copy

Figure 5 - Copy ozekimessageout CREATE TABLE statement

 
OZEKI - Create Database
sms.ozeki.hu/p_5724-create-database.html
MySQL 8.0 Command Line Client - Unicode

```
    -> reference varchar(100) default NULL,
    -> PRIMARY KEY (id)
    -> ) charset=utf8;
Query OK, 0 rows affected, 2 warnings (1.58 sec)

mysql> ALTER TABLE ozekimessagein ADD INDEX (id) ;
Query OK, 0 rows affected (0.51 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> CREATE TABLE ozekimessageout (
    -> id int(11) NOT NULL auto_increment,
    -> sender varchar(30) default NULL,
    -> receiver varchar(30) default NULL,
    -> msg text default NULL,
    -> senttime varchar(100) default NULL,
    -> receivedtime varchar(100) default NULL,
    -> reference varchar(100) default NULL,
    -> status varchar(20) default NULL,
    -> msgtype varchar(160) default NULL,
    -> operator varchar(100) default NULL,
    -> errormsg varchar(250) default NULL,
    -> PRIMARY KEY (id)
    -> ) charset=utf8;
Query OK, 0 rows affected, 2 warnings (0.91 sec)

mysql> ALTER TABLE ozekimessageout ADD INDEX (id);
Query OK, 0 rows affected (0.45 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql>
```

```
msg text default NULL,
senttime varchar(100) default NULL,
receivedtime varchar(100) default NULL,
reference varchar(100) default NULL,
status varchar(20) default NULL,
msgtype varchar(160) default NULL,
operator varchar(100) default NULL,
errormsg varchar(250) default NULL,
PRIMARY KEY (id)
) charset=utf8;
ALTER TABLE ozekimessageout ADD INDEX (id);
```

Figure 2 - CREATE TABLE statements to copy

Figure 6 - Paste ozekimessageout CREATE TABLE statements and run it on the MySQL database server

# Send SMS from MySQL (part 3/4)

## Configure Database User

**In this video series you can see how to connect Ozeki SMS Gateway and MySQL database server for SMS messaging. This video shows how to install and configure a Database User on Ozeki SMS Gateway.**

### Video content

1. Install database user
2. Connect to database

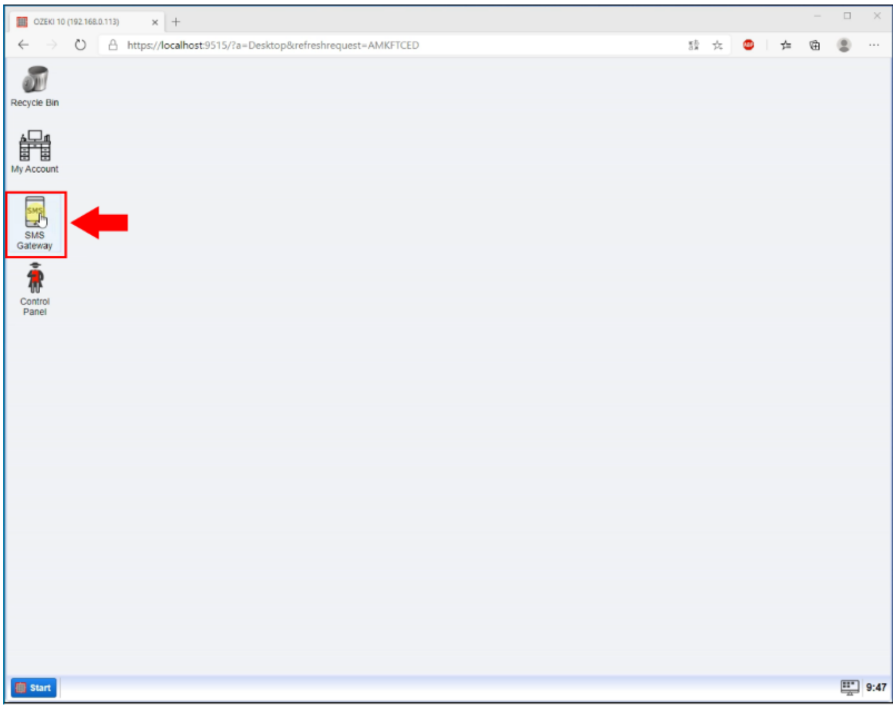


**Figure 1** - Open the SMS Gateway applicaton on Ozeki 10's desktop screen

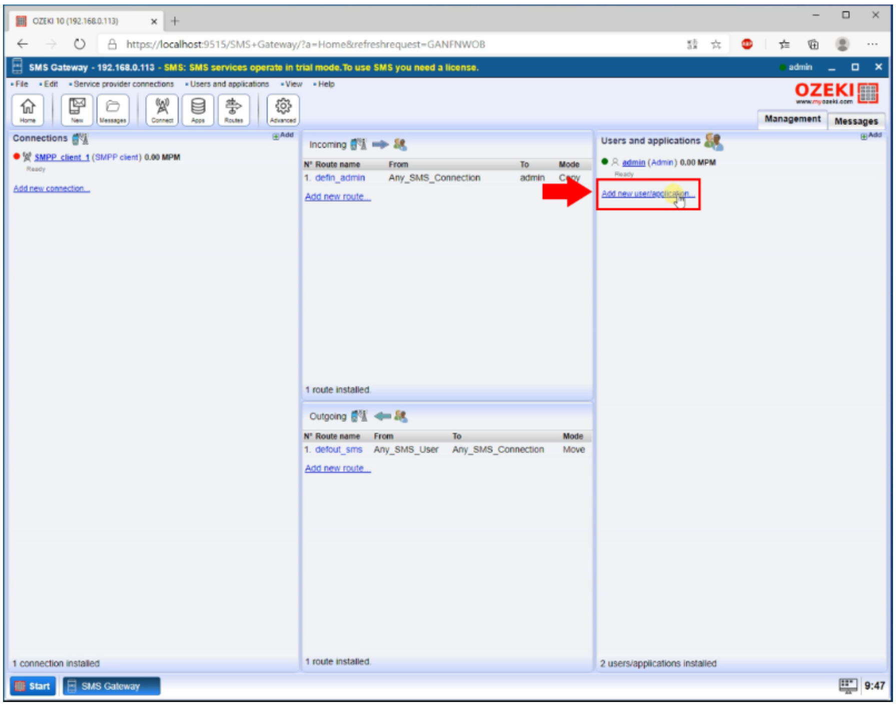
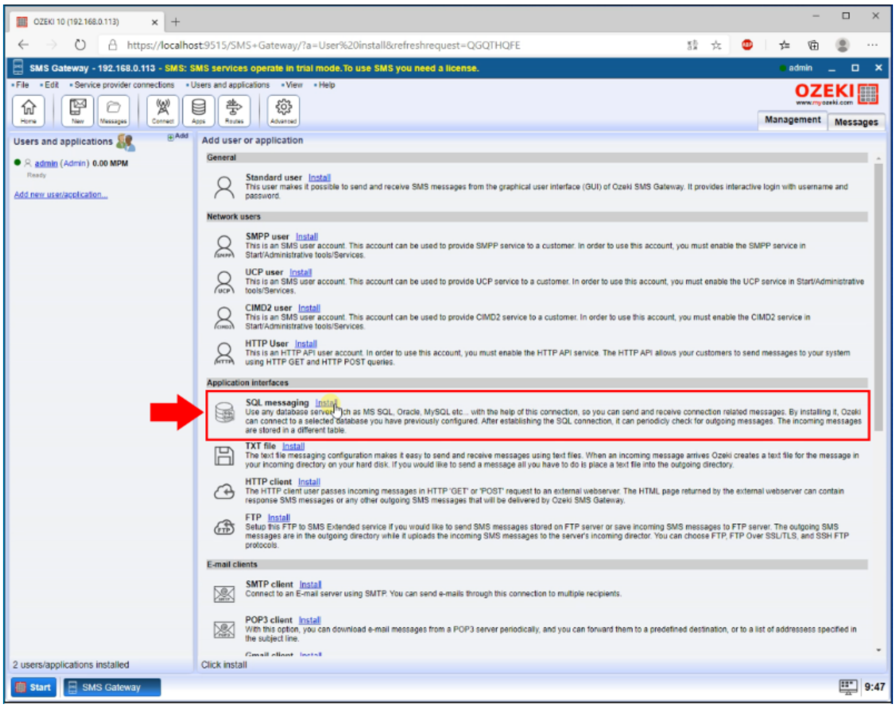


**Figure 2** - Click 'Add new user or application' on the right side panel



**Standard user** Install
This user makes it possible to send and receive SMS messages from the graphical user interface (GUI) of Ozeki SMS Gateway. It provides interactive login with username and password.

**SMPP user** Install
This is an SMS user account. This account can be used to provide SMPP service to a customer. In order to use this account, you must enable the SMPP service in Start/Administrative tools/Services.

**UCP user** Install
This is an SMS user account. This account can be used to provide UCP service to a customer. In order to use this account, you must enable the UCP service in Start/Administrative tools/Services.

**CIMD2 user** Install
This is an SMS user account. This account can be used to provide CIMD2 service to a customer. In order to use this account, you must enable the CIMD2 service in Start/Administrative tools/Services.

**HTTP User** Install
This is an HTTP API user account. In order to use this account, you must enable the HTTP API service. The HTTP API allows your customers to send messages to your system using HTTP GET and HTTP POST queries.

**SQL messaging** Install
Use any database server such as MS SQL, Oracle, MySQL etc... with the help of this connection, so you can send and receive connection related messages. By installing it, Ozeki can connect to a selected database you have previously configured. After establishing the SQL connection, it can periodicly check for outgoing messages. The incoming messages are stored in a different table.

**TXT file** Install
The text file messaging configuration makes it easy to send and receive messages using text files. When an incoming message arrives Ozeki creates a text file for the message in your incoming directory on your hard disk. If you would like to send a message all you have to do is place a text file into the outgoing directory.

**HTTP client** Install
The HTTP client user passes incoming messages in HTTP 'GET' or 'POST' request to an external webserver. The HTML page returned by the external webserver can contain response SMS messages or any other outgoing SMS messages that will be delivered by Ozeki SMS Gateway.

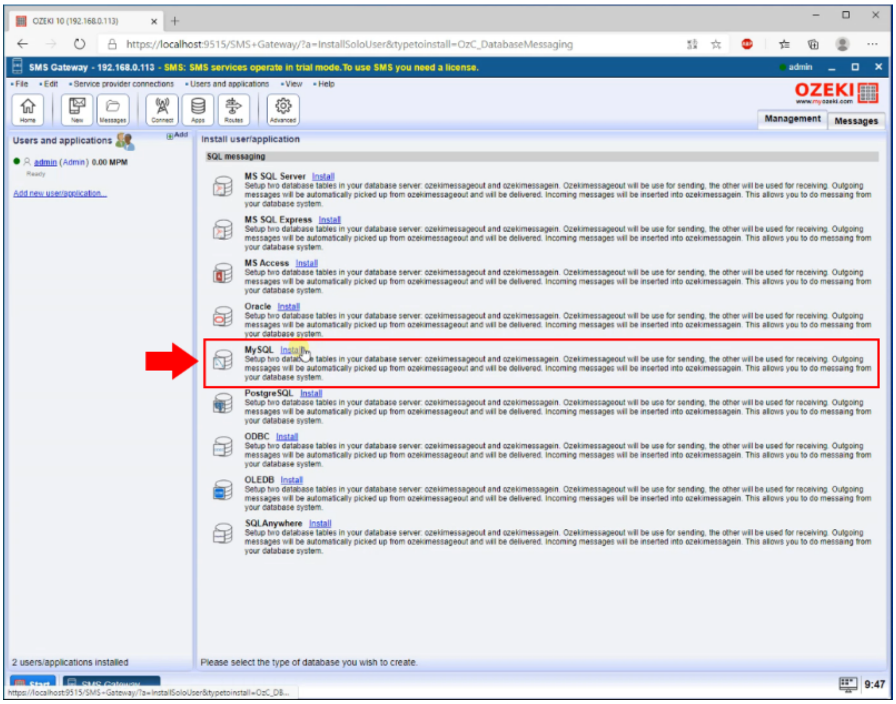
**FTP** Install
Setup this FTP to SMS Extended service if you would like to send SMS messages stored on FTP server or save incoming SMS messages to FTP server. The outgoing SMS messages are in the outgoing directory while it uploads the incoming SMS messages to the server's incoming director. You can choose FTP, FTP Over SSL/TLS, and SSH FTP protocols.

**SMTP client** Install
Connect to an E-mail server using SMTP. You can send e-mails through this connection to multiple recipients.

**POP3 client** Install
With this option, you can download e-mail messages from a POP3 server periodically, and you can forward them to a predefined destination, or to a list of addresses specified in the subject line.

**Gmail client** Install

**Figure 3** - Install SQL Messageing User


OZEKI 10 (192.168.0.113)
https://localhost:9515/SMS+Gateway/?a=InstallSoloUser&typetoinstall=OzC_DatabaseMessaging
SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.To use SMS you need a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
OZEKI
www.myozeki.com
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Users and applications
Add
admin (Admin) 0.00 MPM
Ready
Add new user/application...
Install user/application
SQL messaging
2 users/applications installed
Please select the type of database you wish to create.
Start
SMS Gateway
https://localhost:9515/SMS+Gateway/?a=InstallSoloUser&typetoinstall=OzC_DB...
9:47

**MS SQL Server** Install
Setup two database tables in your database server: ozekimessageout and ozekimessagein. Ozekimessageout will be use for sending, the other will be used for receiving. Outgoing messages will be automatically picked up from ozekimessageout and will be delivered. Incoming messages will be inserted into ozekimessagein. This allows you to do messaing from your database system.

**MS SQL Express** Install
Setup two database tables in your database server: ozekimessageout and ozekimessagein. Ozekimessageout will be use for sending, the other will be used for receiving. Outgoing messages will be automatically picked up from ozekimessageout and will be delivered. Incoming messages will be inserted into ozekimessagein. This allows you to do messaing from your database system.

**MS Access** Install
Setup two database tables in your database server: ozekimessageout and ozekimessagein. Ozekimessageout will be use for sending, the other will be used for receiving. Outgoing messages will be automatically picked up from ozekimessageout and will be delivered. Incoming messages will be inserted into ozekimessagein. This allows you to do messaing from your database system.

**Oracle** Install
Setup two database tables in your database server: ozekimessageout and ozekimessagein. Ozekimessageout will be use for sending, the other will be used for receiving. Outgoing messages will be automatically picked up from ozekimessageout and will be delivered. Incoming messages will be inserted into ozekimessagein. This allows you to do messaing from your database system.

**MySQL** Install
Setup two database tables in your database server: ozekimessageout and ozekimessagein. Ozekimessageout will be use for sending, the other will be used for receiving. Outgoing messages will be automatically picked up from ozekimessageout and will be delivered. Incoming messages will be inserted into ozekimessagein. This allows you to do messaing from your database system.

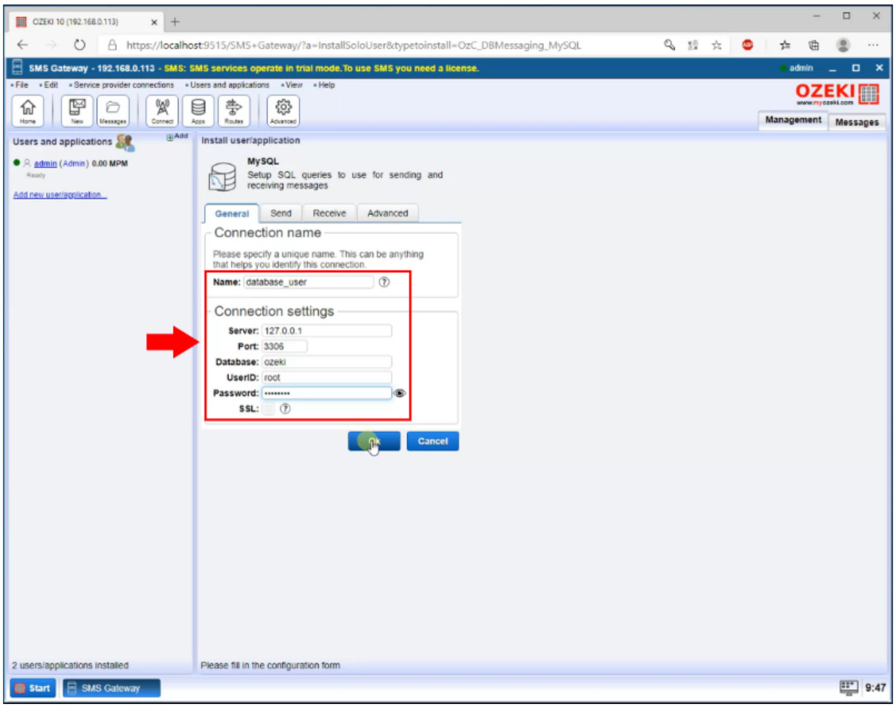
**PostgreSQL** Install
Setup two database tables in your database server: ozekimessageout and ozekimessagein. Ozekimessageout will be use for sending, the other will be used for receiving. Outgoing messages will be automatically picked up from ozekimessageout and will be delivered. Incoming messages will be inserted into ozekimessagein. This allows you to do messaing from your database system.

**ODBC** Install
Setup two database tables in your database server: ozekimessageout and ozekimessagein. Ozekimessageout will be use for sending, the other will be used for receiving. Outgoing messages will be automatically picked up from ozekimessageout and will be delivered. Incoming messages will be inserted into ozekimessagein. This allows you to do messaing from your database system.

**OLEDB** Install
Setup two database tables in your database server: ozekimessageout and ozekimessagein. Ozekimessageout will be use for sending, the other will be used for receiving. Outgoing messages will be automatically picked up from ozekimessageout and will be delivered. Incoming messages will be inserted into ozekimessagein. This allows you to do messaing from your database system.

**SQLAnywhere** Install
Setup two database tables in your database server: ozekimessageout and ozekimessagein. Ozekimessageout will be use for sending, the other will be used for receiving. Outgoing messages will be automatically picked up from ozekimessageout and will be delivered. Incoming messages will be inserted into ozekimessagein. This allows you to do messaing from your database system.

**Figure 4** - Install MYSQL Connection


OZEKI 10 (192.168.0.113)
https://localhost:9515/SMS+Gateway/?a=InstallSoloUser&typetoinstall=OzC_DBMessaging_MySQL
SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode. To use SMS you need a license.
File
Edit
Service provider connections
Users and applications
View
Help
Management
Messages
Users and applications
admin (Admin) 0.00 MPM
Add new user/application...
Install user/application
General
Send
Receive
Advanced
Ok
Cancel
Start
SMS Gateway
9:47

MySQL
Setup SQL queries to use for sending and receiving messages

Connection name

Please specify a unique name. This can be anything that helps you identify this connection.

Name: database_user

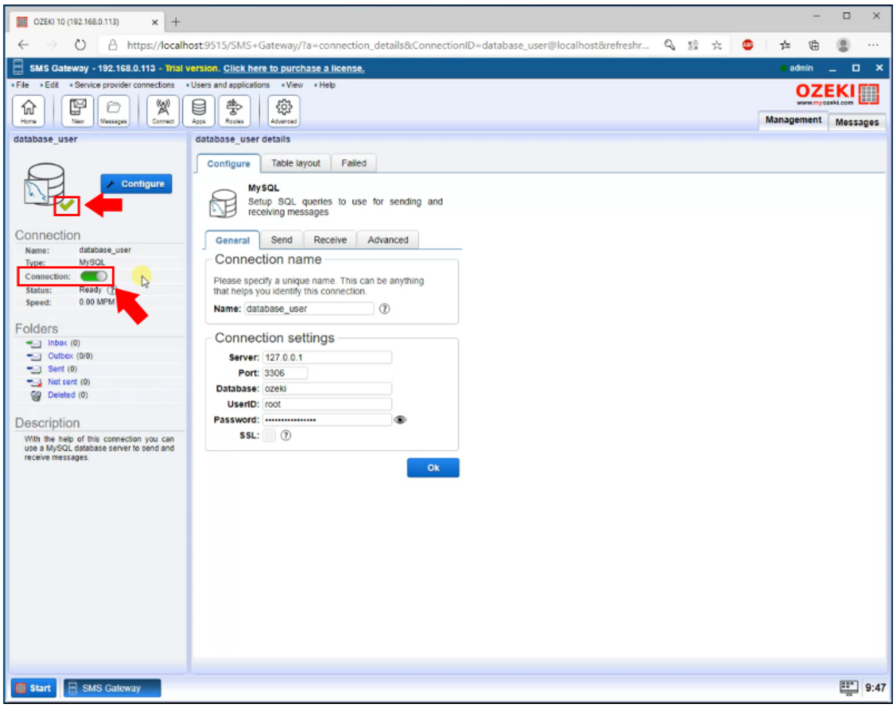
Connection settings

Server: 127.0.0.1
Port: 3306
Database: ozeki
UserID: root
Password: ••••••••
SSL: □

2 users/applications installed

Please fill in the configuration form

**Figure 5 -** Define the MYSQL database connection details


OZEKI 10 (192.168.0.113)
https://localhost:9515/SMS+Gateway/?a=connection_details&ConnectionID=database_user@localhost&refreshr...
SMS Gateway - 192.168.0.113 - Trial version. Click here to purchase a license.
File
Edit
Service provider connections
Users and applications
View
Help
Management
Messages
Configure
Table layout
Failed
General
Send
Receive
Advanced
Ok
Inbox (0)
Outbox (0/0)
Sent (0)
Not sent (0)
Deleted (0)
Start
SMS Gateway
9:47

database_user

database_user details

Connection
Name: database_user
Type: MySQL
Connection: ●
Status: Ready
Speed: 0.00 MPM

Folders

Description

With the help of this connection you can use a MySQL database server to send and receive messages.

MySQL
Setup SQL queries to use for sending and receiving messages

Connection name

Please specify a unique name. This can be anything that helps you identify this connection.

Name: database_user

Connection settings

Server: 127.0.0.1
Port: 3306
Database: ozeki
UserID: root
Password: ••••••••••••••••
SSL: □

**Figure 6 -** Enable MYSQL connection

# Send SMS from MySQL (part 4/4)
## Send Test Message

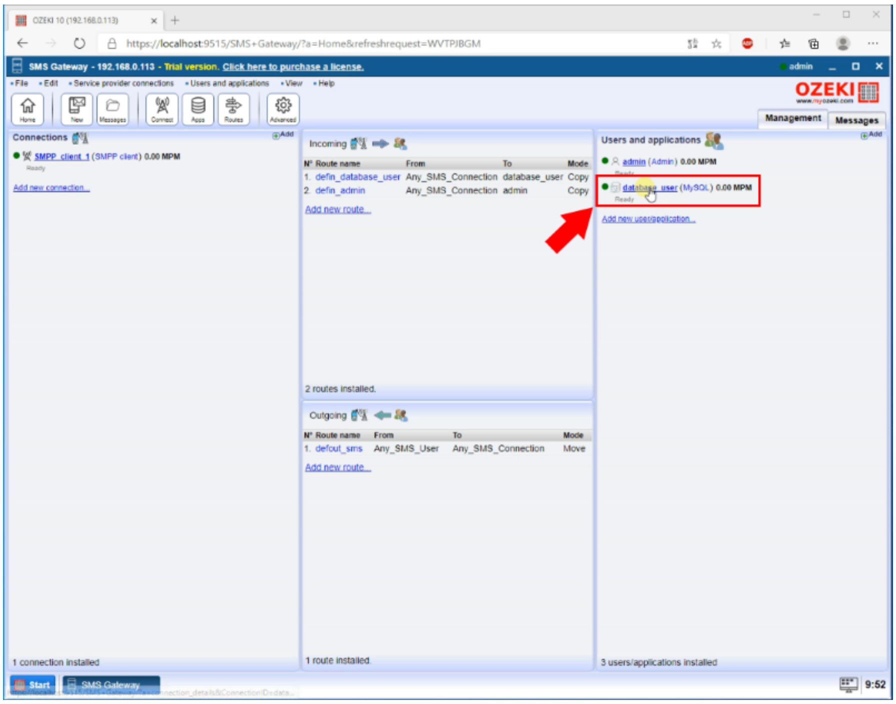
**This is the last video in this video series which shows how to connect Ozeki SMS Gateway and MySQL database server for SMS messaging. See how to insert a simple row into the proper table, so Ozeki SMS Gateway can automatically send SMS messages to mobile phones.**

### Video content

1. Insert message into database
2. Send test message

---

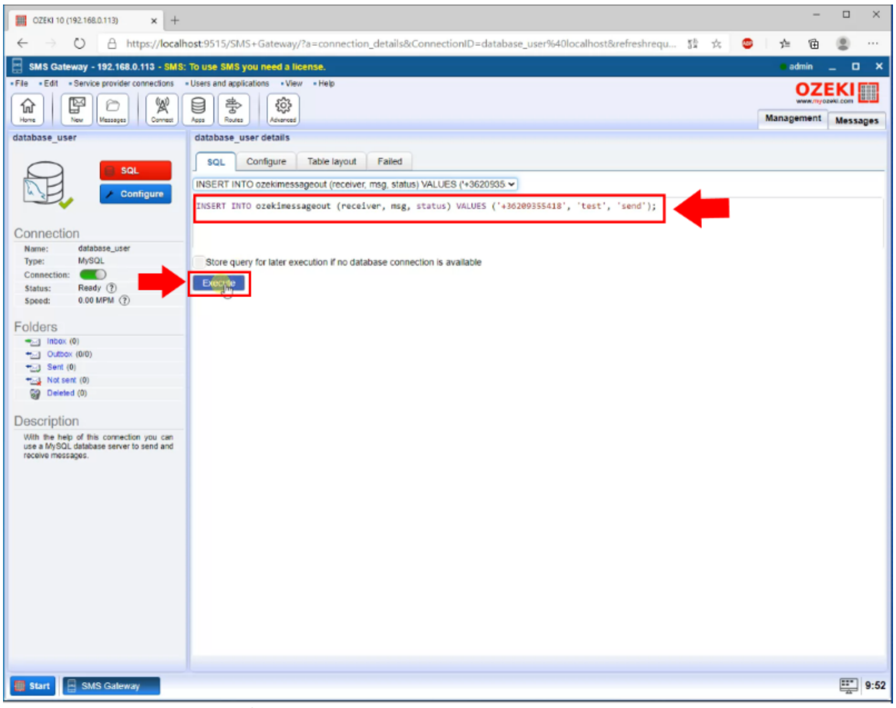
Look at the upcoming screenshots to thoroughly examine the final stage, which is SMS sending. You can start the whole process by INSERT-ing the SMS into the database.



| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defin_database_user | Any_SMS_Connection | database_user | Copy |
| 2. | defin_admin | Any_SMS_Connection | admin | Copy |

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move |

**Figure 1** - Open Database user



```
INSERT INTO ozekimessageout (receiver, msg, status) VALUES ('+36209355418', 'test', 'send');
```

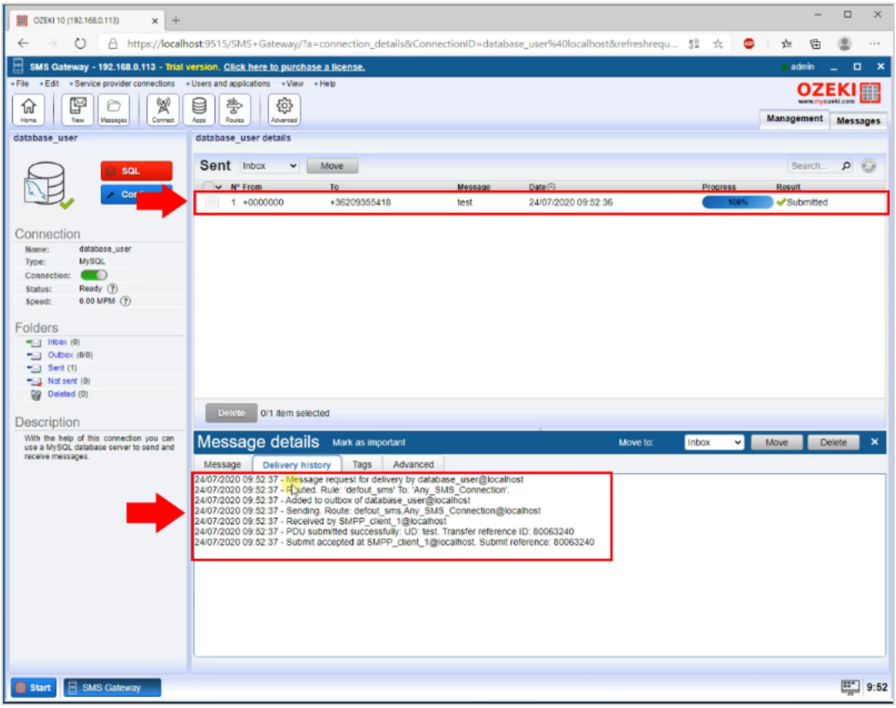
SMS Gateway - 192.168.0.113 - SMS: To use SMS you need a license.
database_user
SQL
Configure
database_user details
SQL
Configure
Table layout
Failed
INSERT INTO ozekimessageout (receiver, msg, status) VALUES ('+3620935
Store query for later execution if no database connection is available
Execute
Connection
Name: database_user
Type: MySQL
Connection:
Status: Ready
Speed: 0.00 MPM
Folders
Inbox (0)
Outbox (0/0)
Sent (0)
Not sent (0)
Deleted (0)
Description
With the help of this connection you can use a MySQL database server to send and receive messages.
Start
SMS Gateway
9:52

**Figure 2** - Insert message to the database table

**INSERT message record** (example):

```
USE ozeki
INSERT INTO ozekimessageout (receiver,msg,status) values ("+44111223465","Hello world","Send");
GO
```

**Figure 3** - Copy INSERT statement


OZEKI 10 (192.168.0.113)
https://localhost:9515/SMS+Gateway/?a=connection_details&ConnectionID=database_user%40localhost&refreshrequ...
SMS Gateway - 192.168.0.113 - Trial version. Click here to purchase a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
OZEKI
www.myozeki.com
Management
Messages
SQL
Con
Start
SMS Gateway
9:52

database_user

Connection

| | |
|---|---|
| Name: | database_user |
| Type: | MySQL |
| Connection: | |
| Status: | Ready |
| Speed: | 0.00 MPM |

Folders

- Inbox (0)
- Outbox (0/0)
- Sent (1)
- Not sent (0)
- Deleted (0)

Description

With the help of this connection you can use a MySQL database server to send and receive messages.

database_user details

Sent Inbox Move

| Nº | From | To | Message | Date | Progress | Result |
|---|---|---|---|---|---|---|
| 1. | +0000000 | +36209355418 | test | 24/07/2020 09:52:36 | 100% | ✓ Submitted |

Delete 0/1 item selected

Message details

```
24/07/2020 09:52:37 - Message request for delivery by database_user@localhost
24/07/2020 09:52:37 - Routed. Rule: 'defout_sms' To: 'Any_SMS_Connection'.
24/07/2020 09:52:37 - Added to outbox of database_user@localhost
24/07/2020 09:52:37 - Sending. Route: defout_sms,Any_SMS_Connection@localhost
24/07/2020 09:52:37 - Received by SMPP_client_1@localhost
24/07/2020 09:52:37 - PDU submitted successfully: UD: test. Transfer reference ID: 80063240
24/07/2020 09:52:37 - Submit accepted at SMPP_client_1@localhost. Submit reference: 80063240
```

**Figure 3 -** The Database User's sent folder shows that your message has been sent by Ozeki SMS Gateway

# SMS from/to PostgreSQL

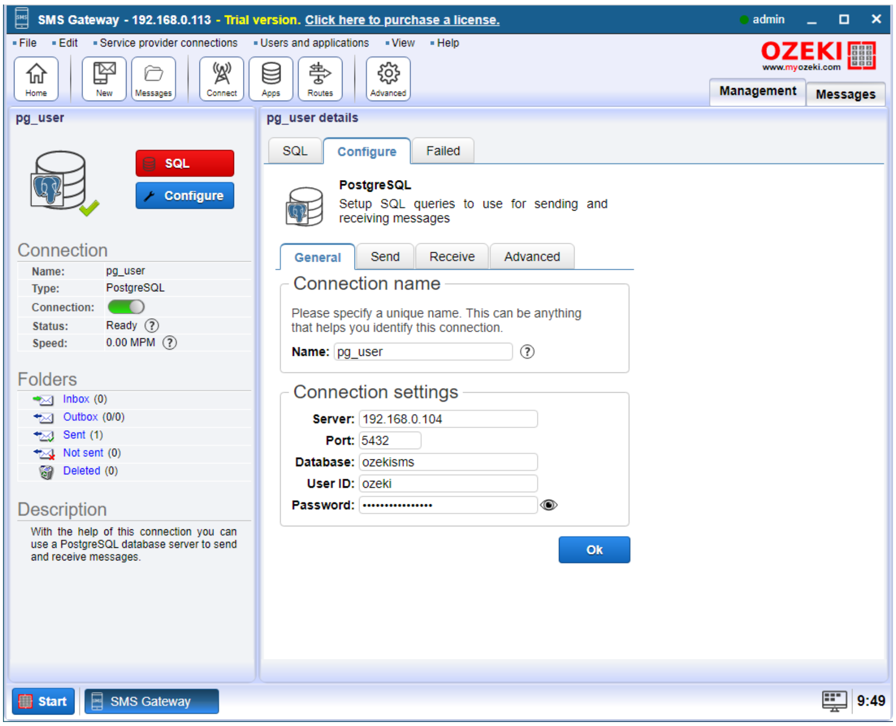
**This chapter gives you great opportunity to see how to send and receive SMS messages through pre-created PostgreSQL database tables. You just need to connect to them with a Database User of Ozeki SMS Gateway. Do not forget to provide the ODBC driver connection string for the user.**

How to send SMS from PostgreSQL
PostgreSQL Installation (Video guide)
Create Database Tables (Video guide)
Configure Database User (Video guide)
Send Test Message (Video guide)

Please **install two important programs**. A simple **PostgreSQL database server** and an **Ozeki SMS Gateway**, which is capable to create a Database User and connect to the PostgreSQL database using the connection string. Check this page how to install a Database User. Do not forget to create a separate table for the outgoing and incoming messages. These tables are called 'ozekimessagein' and 'ozekimessageout' in the current example. Send and receive SMS messages by using simple SELECT and INSERT SQL statements on the tables.

**Use Ozeki SMS Gateway's browser GUI to install a Database User and provide a connection details** similarly as you would do it for other databases (e.g. MSSQL, Oracle, MySQL etc.)

It works if you **modify the variables** by using the information of your PostgreSQL database server (**Figure 1**).



pg_user

Connection
Name: pg_user
Type: PostgreSQL
Connection:
Status: Ready
Speed: 0.00 MPM

Folders
Inbox (0)
Outbox (0/0)
Sent (1)
Not sent (0)
Deleted (0)

Description
With the help of this connection you can use a PostgreSQL database server to send and receive messages.

pg_user details

PostgreSQL
Setup SQL queries to use for sending and receiving messages

General | Send | Receive | Advanced

Connection name
Please specify a unique name. This can be anything that helps you identify this connection.
Name: pg_user

Connection settings
Server: 192.168.0.104
Port: 5432
Database: ozekisms
User ID: ozeki
Password: ••••••••••••••••

Ok

**Figure 1 -** Modify the connection string variables according to your PostgreSQL database

## CREATE tables in PostgreSQL

PostgreSQL accepts simple SQL statements. You can see how to CREATE two separate tables for incoming and outgoing messages in **Figure 3**.

**The CREATE TABLE script in PostgreSQL:**

```
CREATE TABLE ozekimessagein
(
id serial,
sender varchar(30),
receiver varchar(30),
msg varchar(160),
senttime varchar(100),
receivedtime varchar(100),
operator varchar(100),
msgtype varchar(160),
reference varchar(100)
);

CREATE TABLE ozekimessageout
(
id serial,
sender varchar(30),
receiver varchar(30),
msg varchar(160),
senttime varchar(100),
receivedtime varchar(100),
reference varchar(100),
status varchar(20),
operator varchar(100),
errormsg varchar(250),
msgtype varchar(160)
);
```

**Figure 2** - These statements CREATE two tables in your PostgreSQL database

You can always increase the size of the 'msg' field above 160 characters. You can also change it's data type as well.

It is strongly suggested to maintain the 'id' attribute in all tables. 'id' is basicly the index of each record.

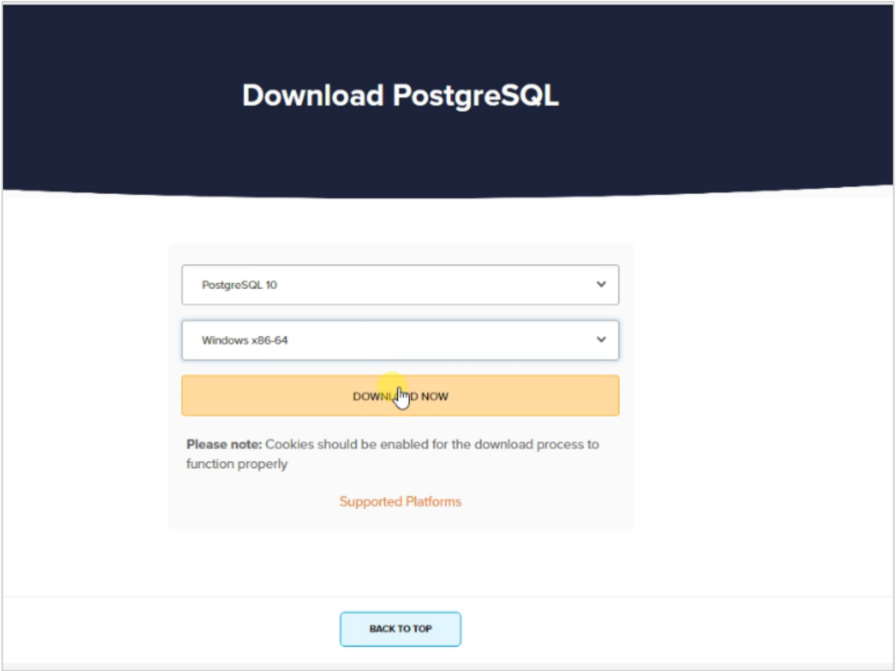
# Send SMS from PostgreSQL (part 1/4)
## PostgreSQL Installation

**In this video series you can see how to connect Ozeki SMS Gateway to a PostgreSQL database server for SMS messaging. The first video shows how to download and install PostgreSQL.**

**Video content**

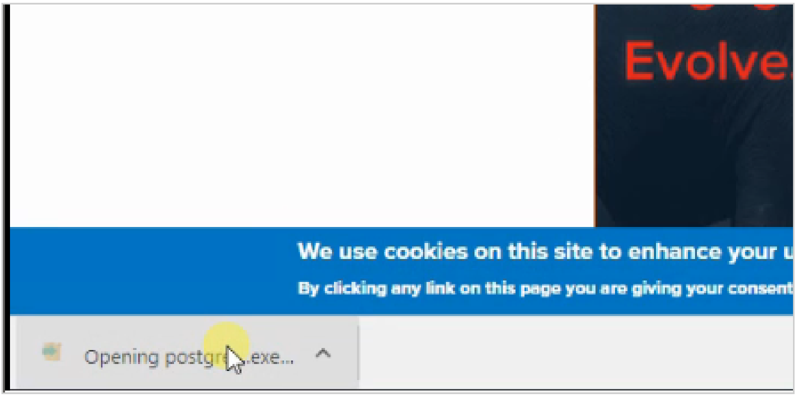
1. Download PostgreSQL
2. Install PostgreSQL
3. Install PostgreSQL's ODBC Driver

---



Please note: Cookies should be enabled for the download process to function properly

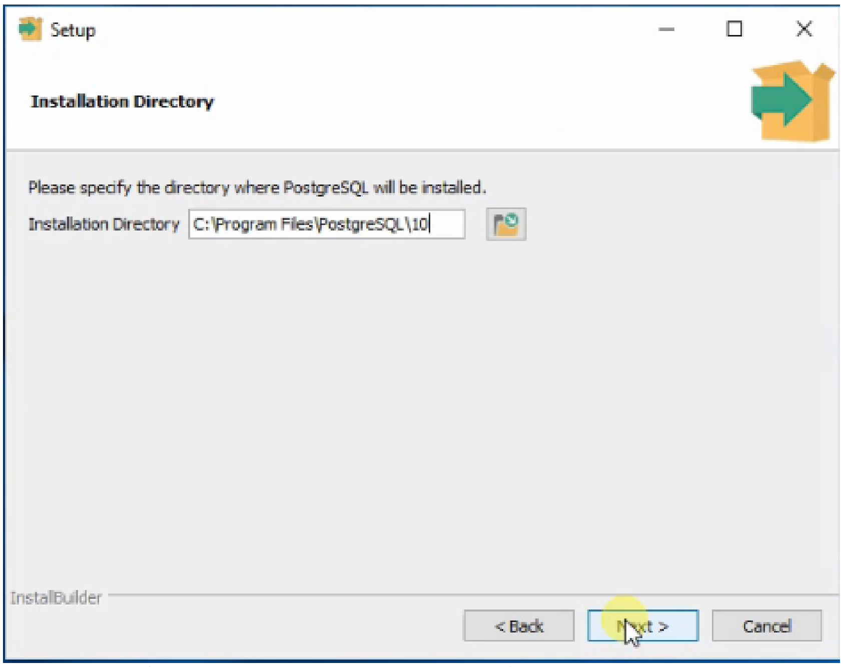
**Figure 1** - PostgreSQL's webpage



We use cookies on this site to enhance your u

By clicking any link on this page you are giving your consent

**Figure 2** - Start installation from webbrowser



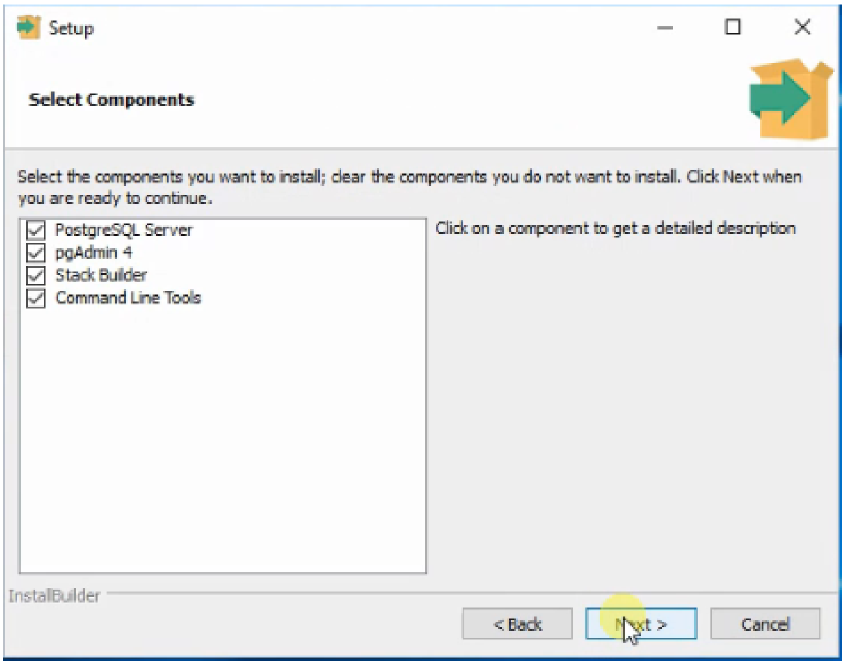
Setup

Installation Directory

Please specify the directory where PostgreSQL will be installed.

Installation Directory C:\Program Files\PostgreSQL\10

**Figure 3** - Choose installation directory for the database server



Setup

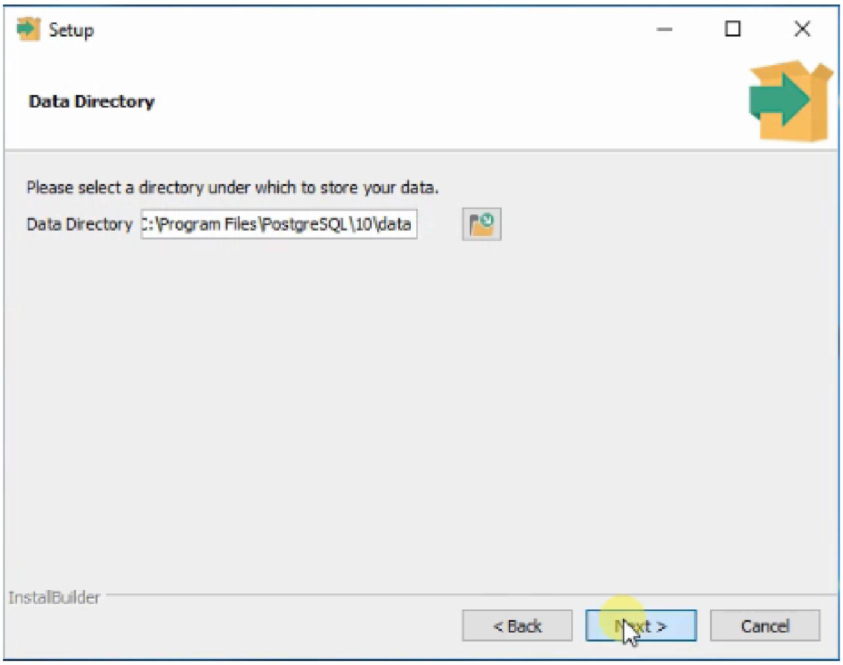
Select Components

Select the components you want to install; clear the components you do not want to install. Click Next when you are ready to continue.

☑ PostgreSQL Server
☑ pgAdmin 4
☑ Stack Builder
☑ Command Line Tools

Click on a component to get a detailed description

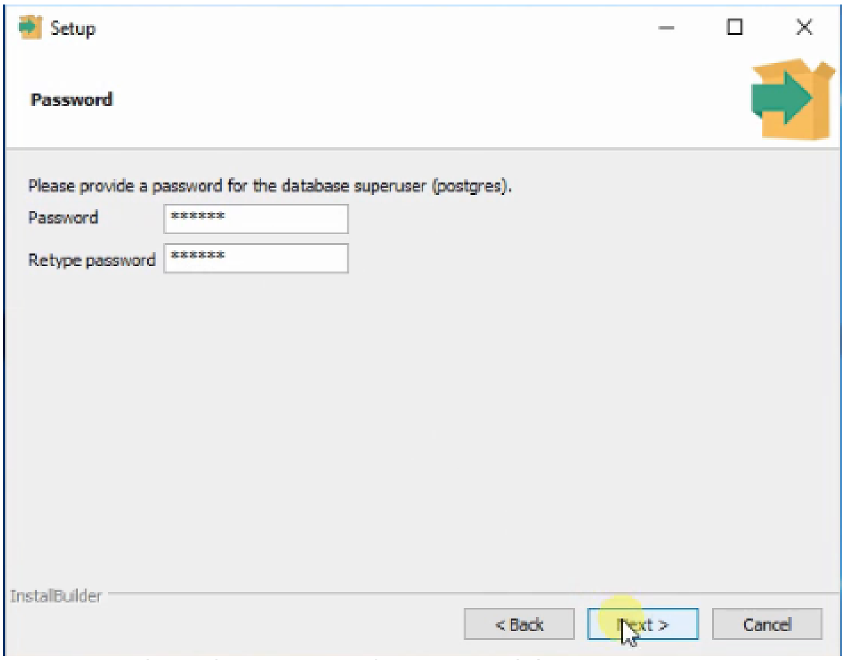
**Figure 4** - Select components to install



Please select a directory under which to store your data.

Data Directory :\Program Files\PostgreSQL\10\data

**Figure 5** - Choose data directory for the PostgreSQL server

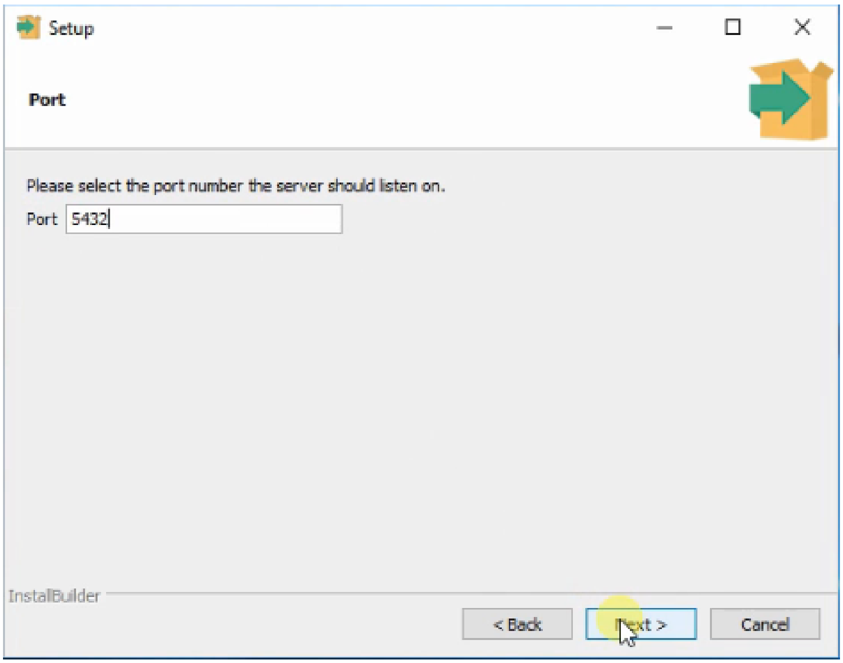


Please provide a password for the database superuser (postgres).

Password ******

Retype password ******

**Figure 6** - Specify password for PostgreSQL's superuser



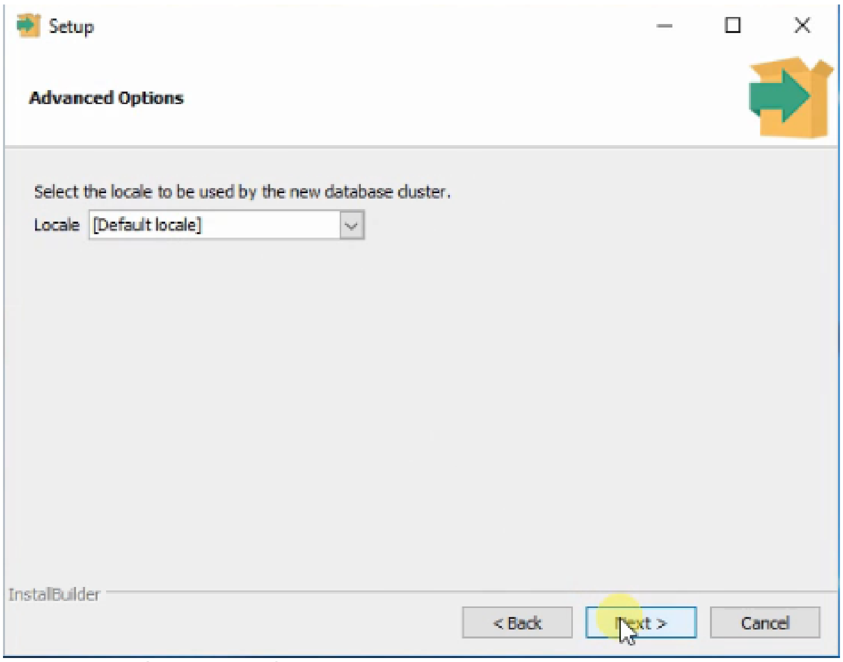
Setup

Port

Please select the port number the server should listen on.

Port 5432

**Figure 7** - Specify PostgreSQL's port number



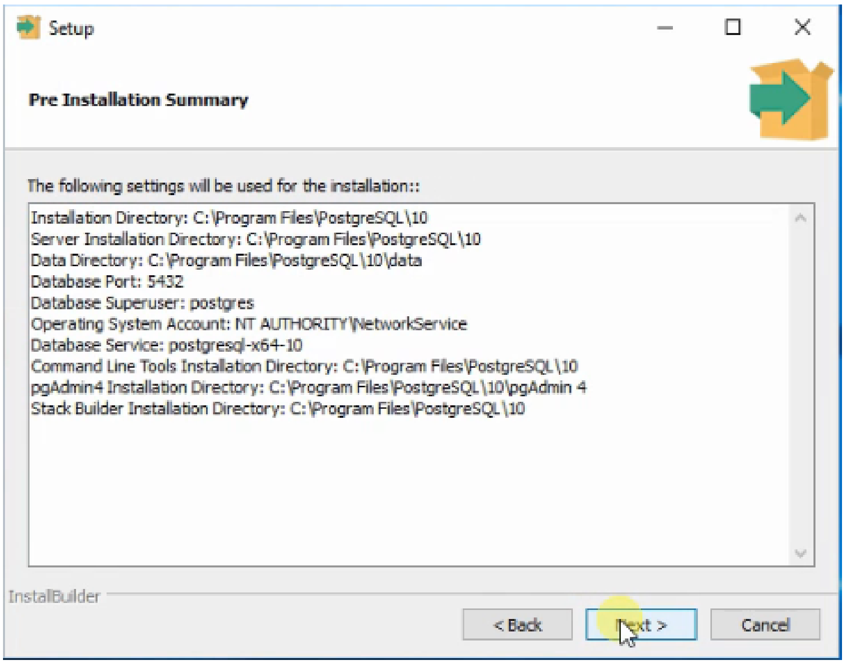
Setup

Advanced Options

Select the locale to be used by the new database cluster.

Locale [Default locale]

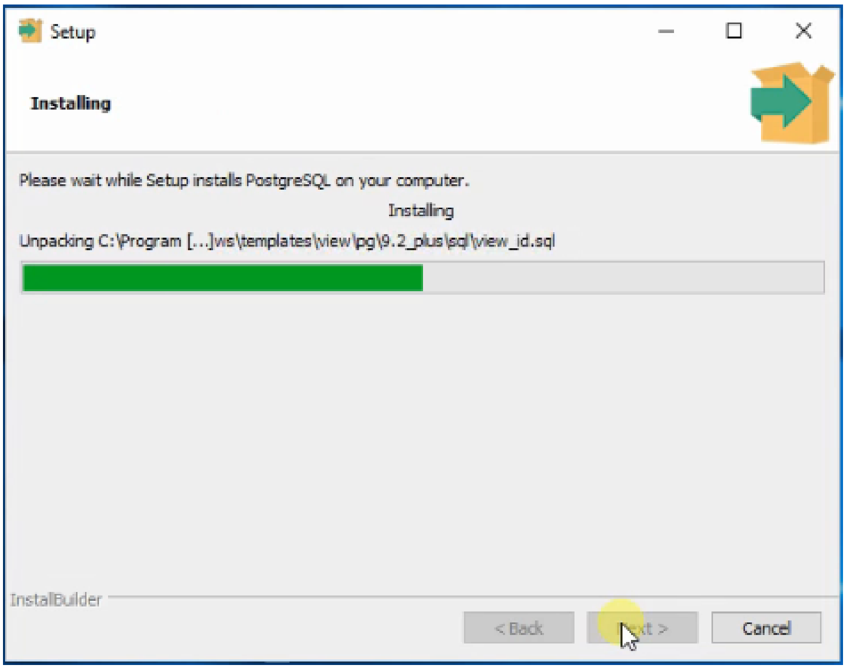
**Figure 8** - Select [Default local] database cluster



The following settings will be used for the installation::

```
Installation Directory: C:\Program Files\PostgreSQL\10
Server Installation Directory: C:\Program Files\PostgreSQL\10
Data Directory: C:\Program Files\PostgreSQL\10\data
Database Port: 5432
Database Superuser: postgres
Operating System Account: NT AUTHORITY\NetworkService
Database Service: postgresql-x64-10
Command Line Tools Installation Directory: C:\Program Files\PostgreSQL\10
pgAdmin4 Installation Directory: C:\Program Files\PostgreSQL\10\pgAdmin 4
Stack Builder Installation Directory: C:\Program Files\PostgreSQL\10
```

**Figure 9** - Read pre installation summary

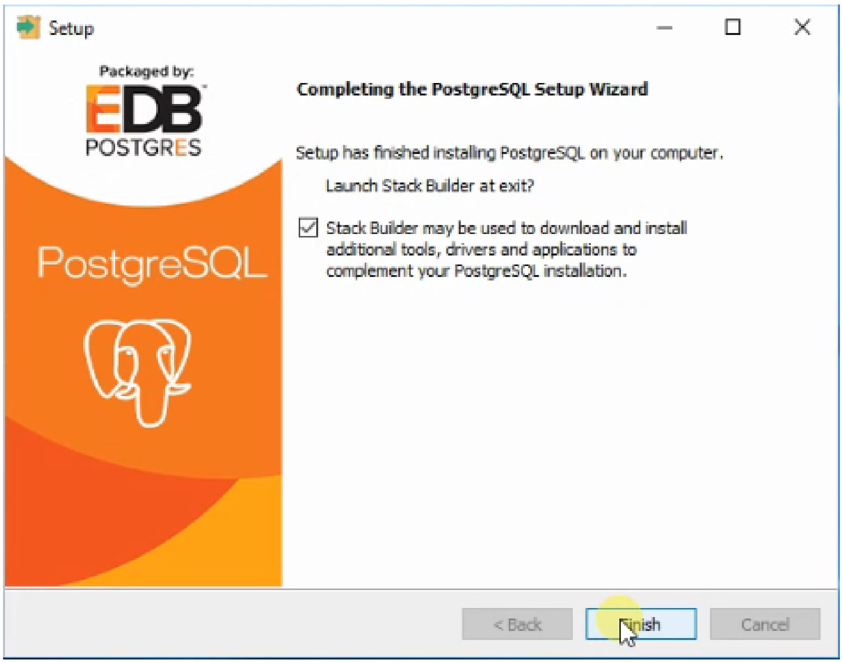


Please wait while Setup installs PostgreSQL on your computer.

Installing

Unpacking C:\Program [...]ws\templates\view\pg\9.2_plus\sql\view_id.sql

**Figure 10** - Wait until the installation is finished



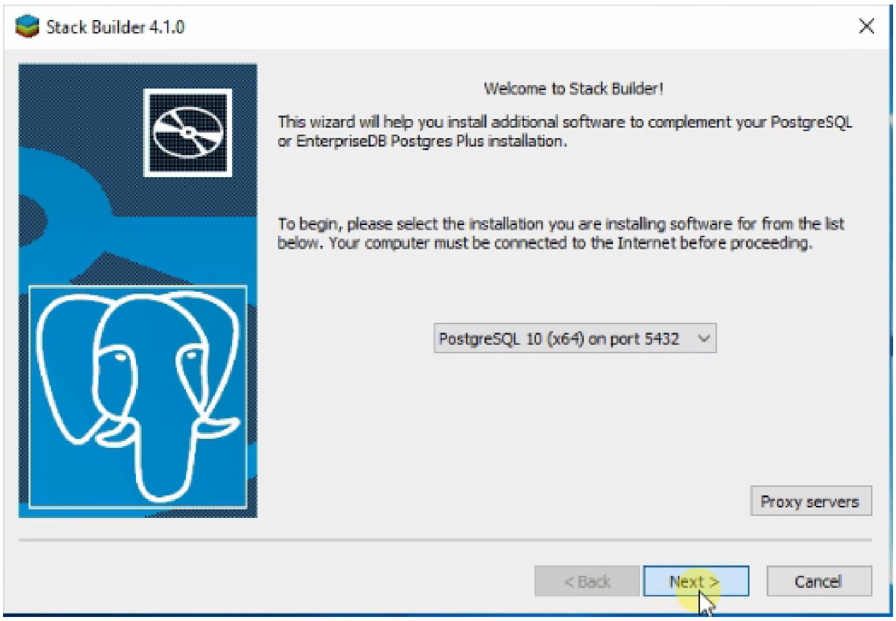
Completing the PostgreSQL Setup Wizard

Setup has finished installing PostgreSQL on your computer.

Launch Stack Builder at exit?

☑ Stack Builder may be used to download and install additional tools, drivers and applications to complement your PostgreSQL installation.

**Figure 11** - PostgreSQL database server's installation has been finished



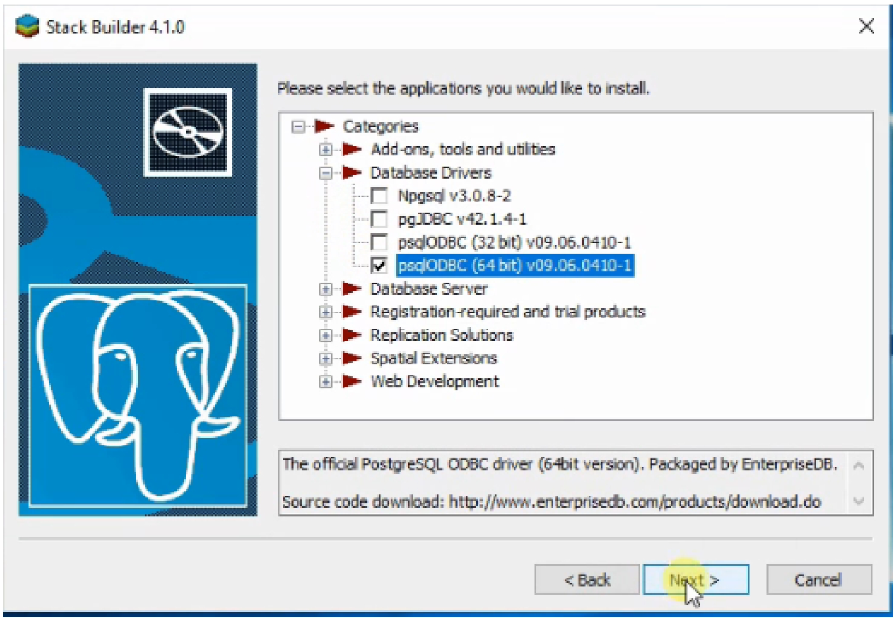
Welcome to Stack Builder!

This wizard will help you install additional software to complement your PostgreSQL or EnterpriseDB Postgres Plus installation.

To begin, please select the installation you are installing software for from the list below. Your computer must be connected to the Internet before proceeding.

PostgreSQL 10 (x64) on port 5432

**Figure 12** - Install PostgreSQL's ODBC Driver



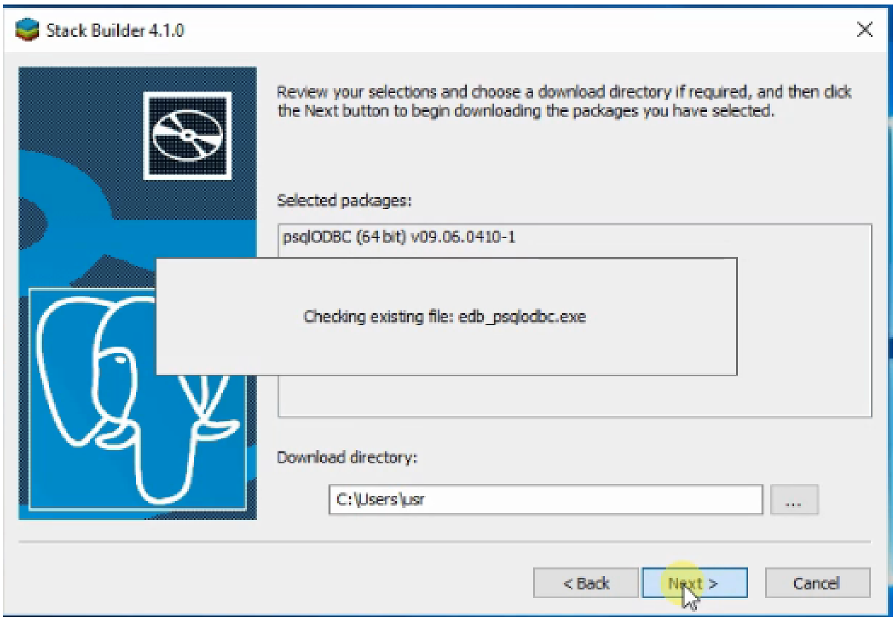
Please select the applications you would like to install.

- Categories
  - Add-ons, tools and utilities
  - Database Drivers
    - □ Npgsql v3.0.8-2
    - □ pgJDBC v42.1.4-1
    - □ psqlODBC (32 bit) v09.06.0410-1
    - ☑ psqlODBC (64 bit) v09.06.0410-1
  - Database Server
  - Registration-required and trial products
  - Replication Solutions
  - Spatial Extensions
  - Web Development

The official PostgreSQL ODBC driver (64bit version). Packaged by EnterpriseDB.

Source code download: http://www.enterprisedb.com/products/download.do

**Figure 13** - Select 'psqlODBC' driver



Review your selections and choose a download directory if required, and then click the Next button to begin downloading the packages you have selected.

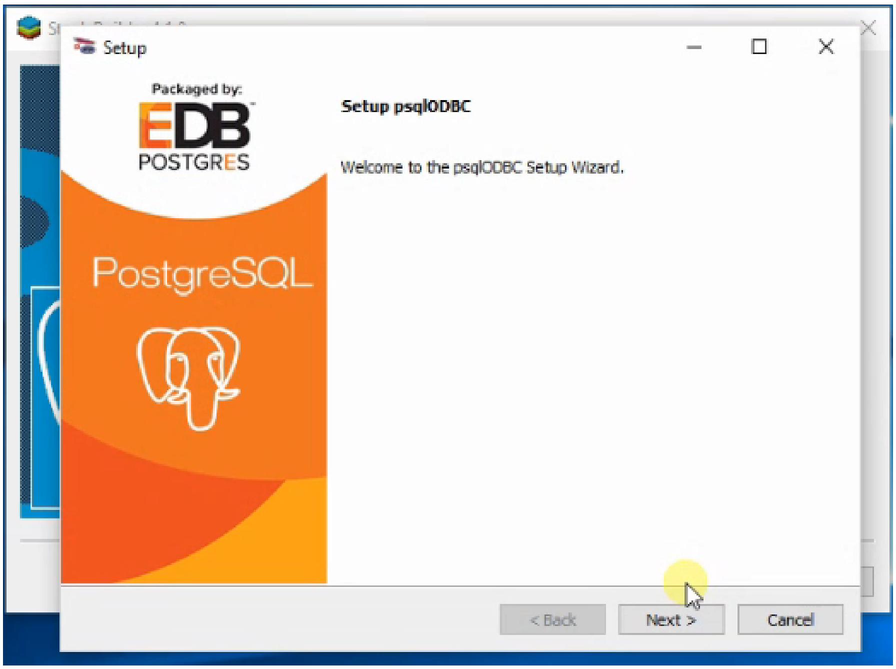
Selected packages:

psqlODBC (64 bit) v09.06.0410-1

Checking existing file: edb_psqlodbc.exe

Download directory: C:\Users\usr

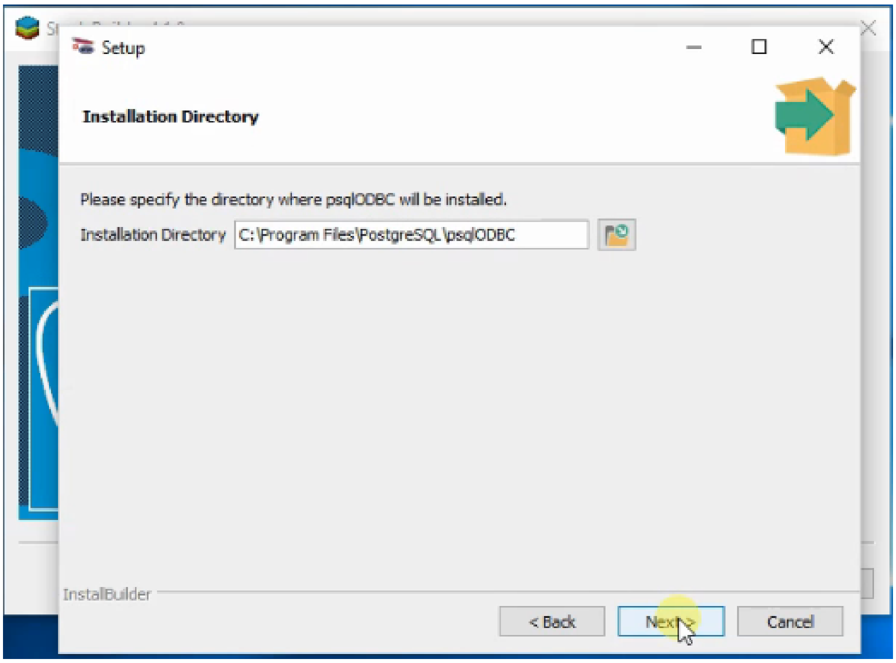
**Figure 14** - Select destination folder to download 'psqlODBC' driver packages



**Setup psqlODBC**

Welcome to the psqlODBC Setup Wizard.

**Figure 15** - Start the installer of the 'psqlODBC' driver

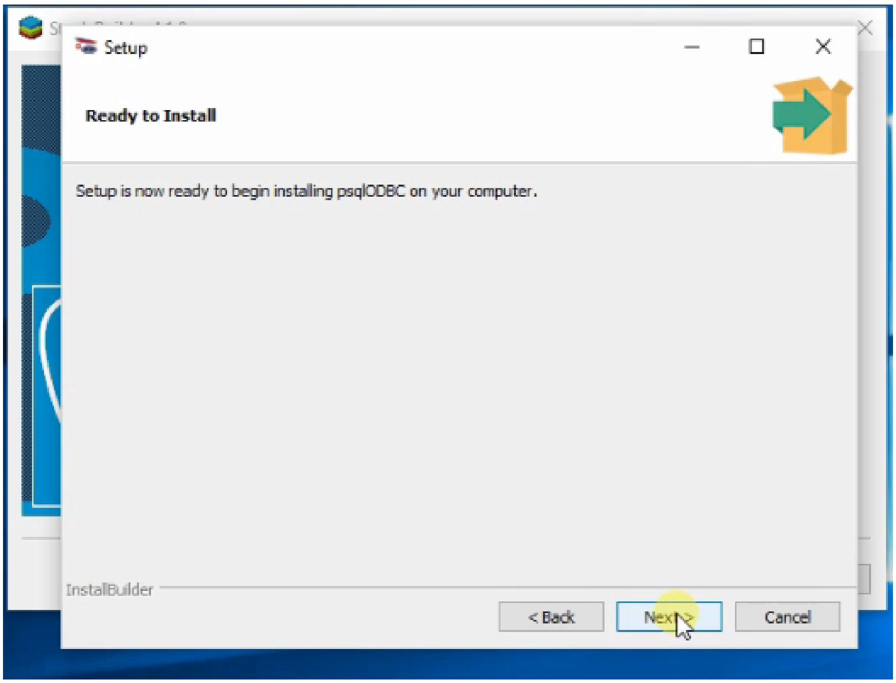


**Installation Directory**

Please specify the directory where psqlODBC will be installed.

Installation Directory C:\Program Files\PostgreSQL\psqlODBC

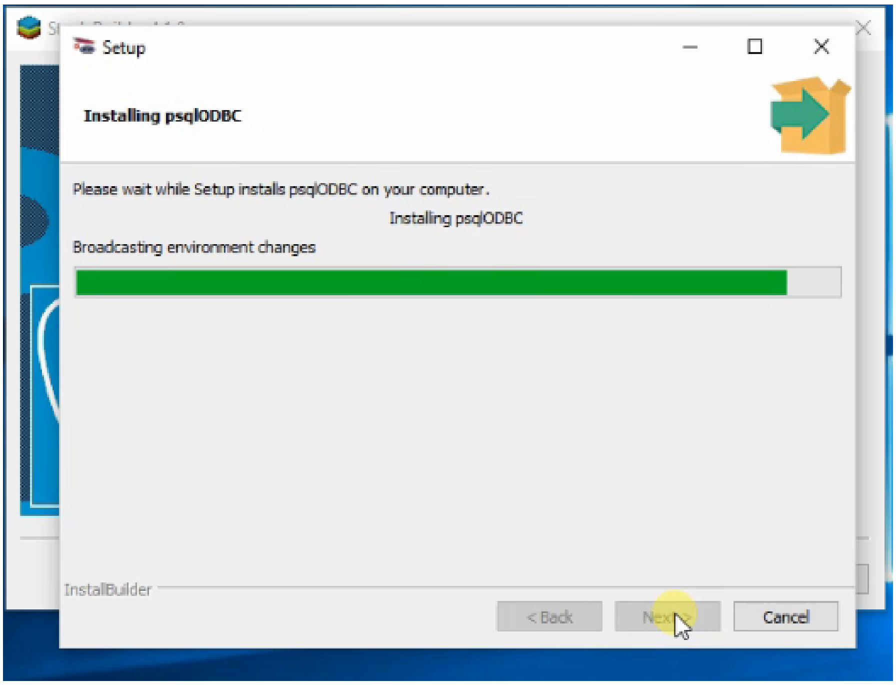
**Figure 16** - Choose installation directory for the driver


Setup

Ready to Install

Setup is now ready to begin installing psqlODBC on your computer.

**Figure 17 -** Start 'psqlODBC' driver installation


Setup

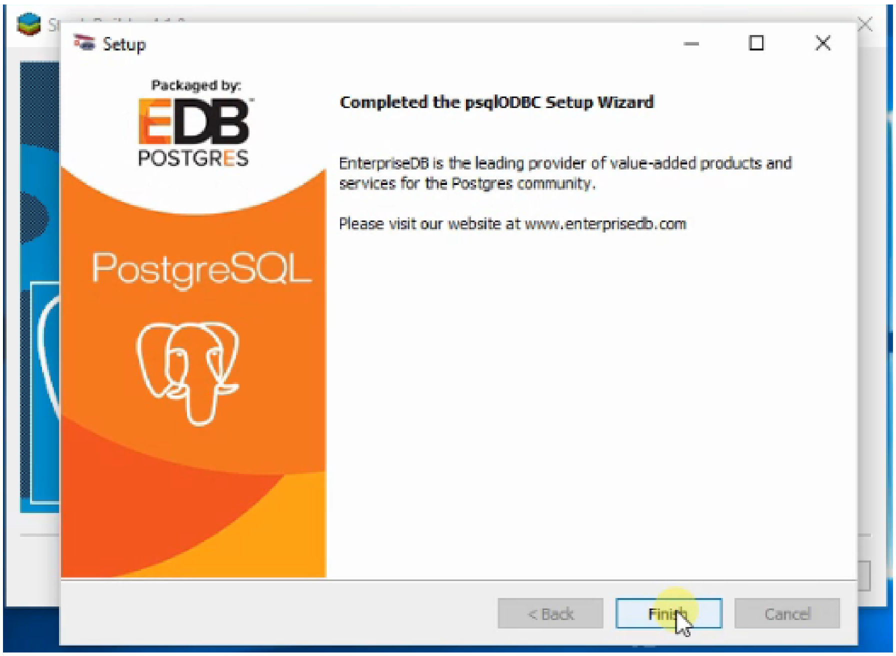
Installing psqlODBC

Please wait while Setup installs psqlODBC on your computer.

Installing psqlODBC

Broadcasting environment changes

**Figure 18 -** Wait until the driver's installation is finished



Completed the psqlODBC Setup Wizard

EnterpriseDB is the leading provider of value-added products and services for the Postgres community.

Please visit our website at www.enterprisedb.com

**Figure 19 -** psqlODBC driver's installation has been finished

# Send SMS from PostgreSQL (part 2/4)
## Create Database Tables

**In this video series you can see how to connect Ozeki SMS Gateway to a PostgreSQL database server for SMS messaging. This video shows how to create the proper database table structure by using the GUI of your PostgreSQL database server.**

**Video content**

1. Connect to PostgreSQL
2. Create User in PostgreSQL
3. Copy CREATE TABLE statements

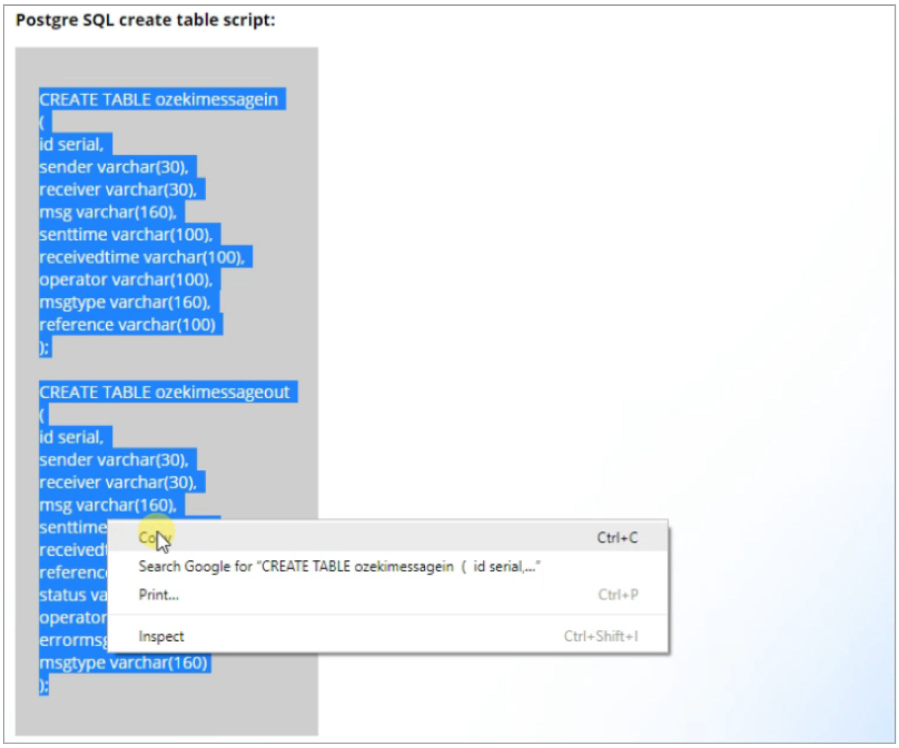
Please **scroll down to copy the SQL statements** (**Figure 2-3**) used in the video. If you have created the database in PostgreSQL, you can jump to the next video.

**Figure 1** - Connect to the PostgreSQL server by using the PgAdmin software

**CREATE TABLE statements to use on your PostgreSQL server**

```
CREATE TABLE ozekimessagein
<br>(
<br>
<font face="Times New Roman"></font>id serial,
<br><font face="Times New Roman"></font>sender varchar(30),
<br><font face="Times New Roman"></font>receiver varchar(30),
<br><font face="Times New Roman"></font>msg varchar(160),
<br><font face="Times New Roman"></font>senttime varchar(100),
<br><font face="Times New Roman"></font>receivedtime varchar(100),
<br>
<font face="Times New Roman"></font>operator varchar(100),
<br><font face="Times New Roman"></font>msgtype varchar(160),
<br><font face="Times New Roman"></font>reference varchar(100)
<br>);<br><br>CREATE TABLE ozekimessageout
<br>(<br><font face="Times New Roman"></font>id serial,
<br><font face="Times New Roman"></font>sender varchar(30),
<br><font face="Times New Roman"></font>receiver varchar(30),
<br><font face="Times New Roman"></font>msg varchar(160),
<br><font face="Times New Roman"></font>senttime varchar(100),
<br><font face="Times New Roman"></font>receivedtime varchar(100),
<br><font face="Times New Roman"></font>reference varchar(100),
<br><font face="Times New Roman"></font>status varchar(20),
<br><font face="Times New Roman"></font>operator varchar(100),
<br>errormsg varchar(250)<font face="Times New Roman">,
<br></font>msgtype varchar(160)<br>);
```

**Figure 2** - Copy CREATE TABLE statements

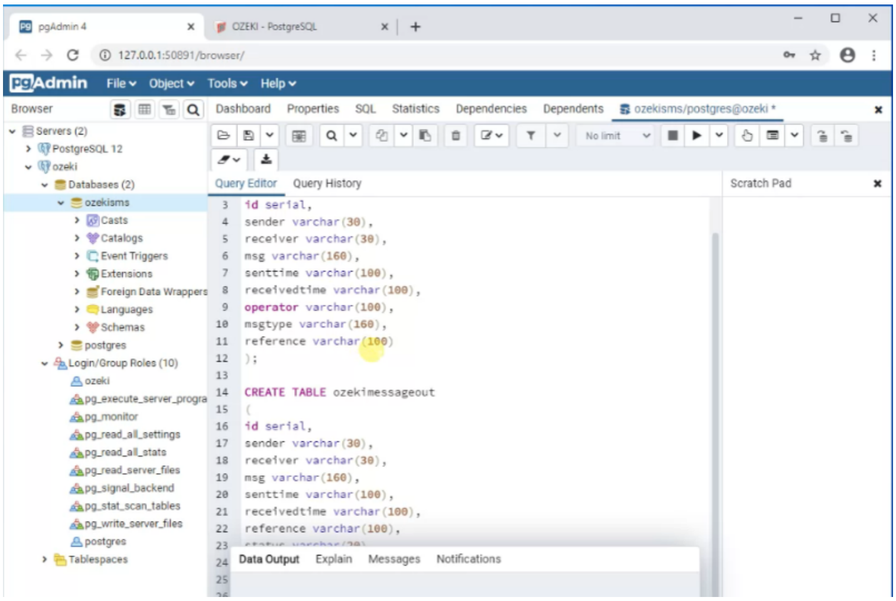


**Postgre SQL create table script:**

```
CREATE TABLE ozekimessagein
(
id serial,
sender varchar(30),
receiver varchar(30),
msg varchar(160),
senttime varchar(100),
receivedtime varchar(100),
operator varchar(100),
msgtype varchar(160),
reference varchar(100)
);

CREATE TABLE ozekimessageout
(
id serial,
sender varchar(30),
receiver varchar(30),
msg varchar(160),
senttime
received
referenc
status va
operator
errormsg
msgtype varchar(160)
);
```

Copy Ctrl+C
Search Google for "CREATE TABLE ozekimessagein ( id serial,..."
Print... Ctrl+P
Inspect Ctrl+Shift+I

**Figure 3 -** Copy CREATE TABLE statements from Figure 2



```
id serial,
sender varchar(30),
receiver varchar(30),
msg varchar(160),
senttime varchar(100),
receivedtime varchar(100),
operator varchar(100),
msgtype varchar(160),
reference varchar(100)
);

CREATE TABLE ozekimessageout
(
id serial,
sender varchar(30),
receiver varchar(30),
msg varchar(160),
senttime varchar(100),
receivedtime varchar(100),
reference varchar(100),
status varchar(30)
```

**Figure 4 -** Paste CREATE TABLE statements and run them on the PostgreSQL database server

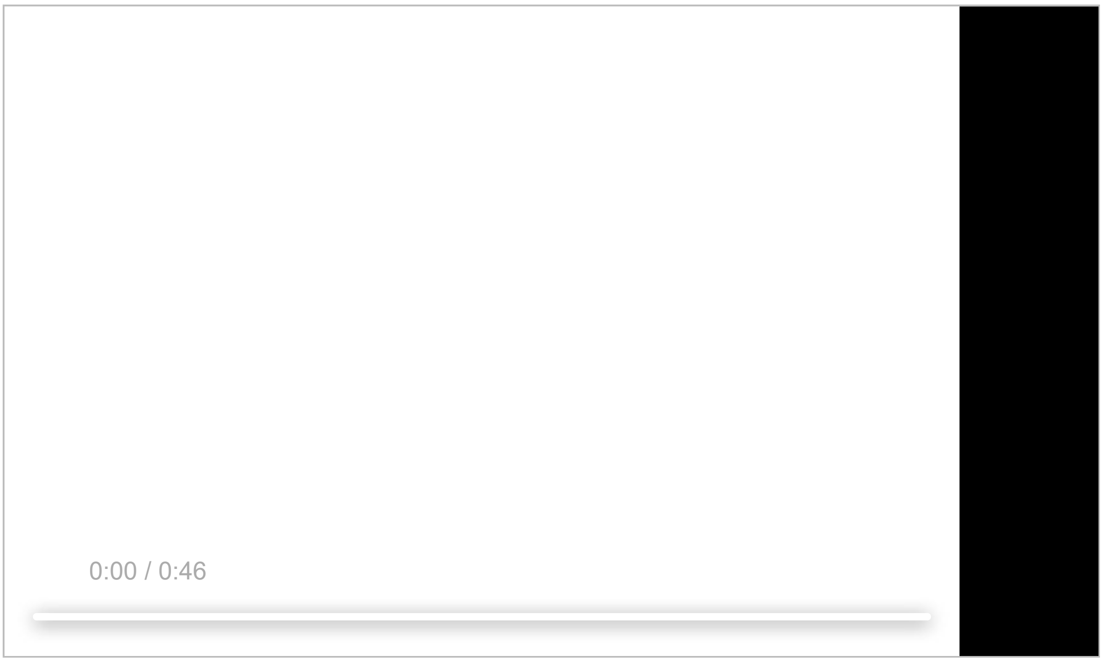
# Send SMS from PostgreSQL (part 3/4)

## Configure Database User

**In this video series you can see how to connect Ozeki SMS Gateway to your PostgreSQL database server for SMS messaging. This video shows how to install and configure a Database User on Ozeki SMS Gateway. It is necessary to provide the connection string.**

**Video content**

1. Install Database User
2. Provide connection String
3. Connect to Database



Please **scroll down to copy the PostgreSQL connection string** used in the video. If you have created the database in PostgreSQL, you can jump to the next video. Although you can precisely examine the sequence by looking through these screenshots.

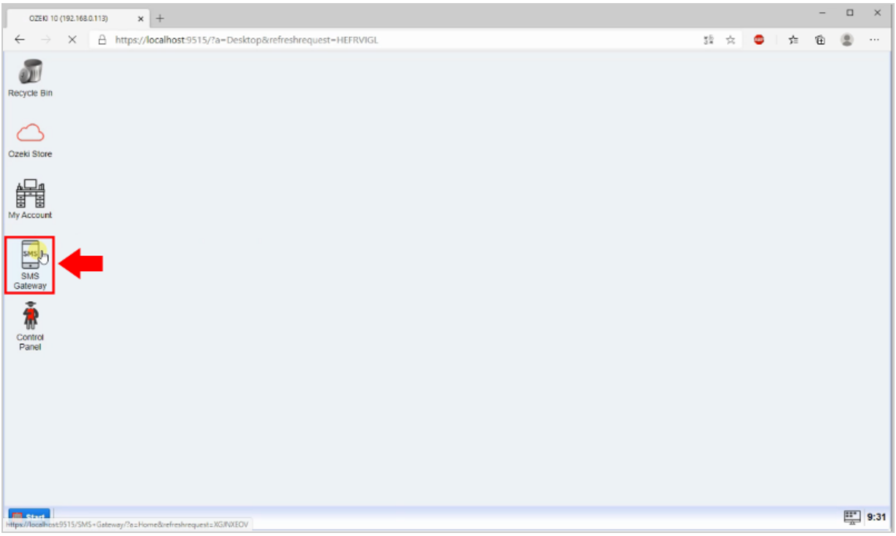
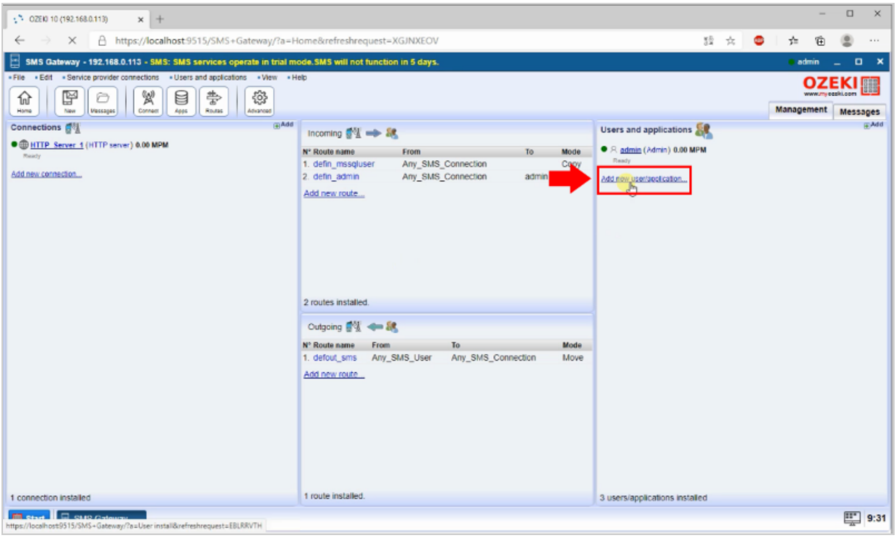


**Figure 1** - Open the SMS Gateway applicaton on Ozeki 10's desktop screen


OZEKI 10 (192.168.0.113)
https://localhost:9515/SMS+Gateway/?a=Home&refreshrequest=XGJNXEOV
SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.SMS will not function in 5 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
OZEKI
www.myozeki.com
Management
Messages
Add
Start
SMS Gateway
9:31
https://localhost:9515/SMS+Gateway/?a=User install&refreshrequest=EBLRRVTH

Connections

HTTP_Server_1 (HTTP server) 0.00 MPM
Ready

Add new connection...

1 connection installed

Incoming

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defin_mssqluser | Any_SMS_Connection | | Copy |
| 2. | defin_admin | Any_SMS_Connection | admin | |

Add new route...

2 routes installed.

Outgoing

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move |

Add new route...

1 route installed.

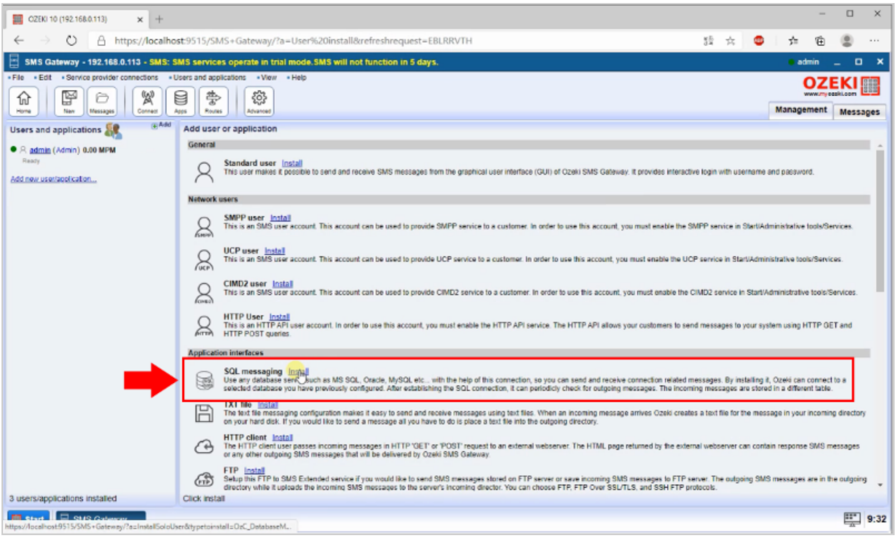
Users and applications

admin (Admin) 0.00 MPM
Ready

Add new user/application...

3 users/applications installed

**Figure 2** - Click 'Add new user or application' on the right side panel


OZEKI 10 (192.168.0.113)
https://localhost:9515/SMS+Gateway/?a=User%20install&refreshrequest=EBLRRVTH
SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.SMS will not function in 5 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
OZEKI
www.myozeki.com
Management
Messages
Add
Start
SMS Gateway
9:32
https://localhost:9515/SMS+Gateway/?a=InstallSoloUser&typetoinstall=OzC_DatabaseM...

Users and applications

admin (Admin) 0.00 MPM
Ready

Add new user/application...

3 users/applications installed

Add user or application

General

Standard user Install
This user makes it possible to send and receive SMS messages from the graphical user interface (GUI) of Ozeki SMS Gateway. It provides interactive login with username and password.

Network users

SMPP user Install
This is an SMS user account. This account can be used to provide SMPP service to a customer. In order to use this account, you must enable the SMPP service in Start/Administrative tools/Services.

UCP user Install
This is an SMS user account. This account can be used to provide UCP service to a customer. In order to use this account, you must enable the UCP service in Start/Administrative tools/Services.

CIMD2 user Install
This is an SMS user account. This account can be used to provide CIMD2 service to a customer. In order to use this account, you must enable the CIMD2 service in Start/Administrative tools/Services.

HTTP User Install
This is an HTTP API user account. In order to use this account, you must enable the HTTP API service. The HTTP API allows your customers to send messages to your system using HTTP GET and HTTP POST queries.

Application interfaces

SQL messaging Install
Use any database server such as MS SQL, Oracle, MySQL etc... with the help of this connection, so you can send and receive connection related messages. By installing it, Ozeki can connect to a selected database you have previously configured. After establishing the SQL connection, it can periodicly check for outgoing messages. The incoming messages are stored in a different table.

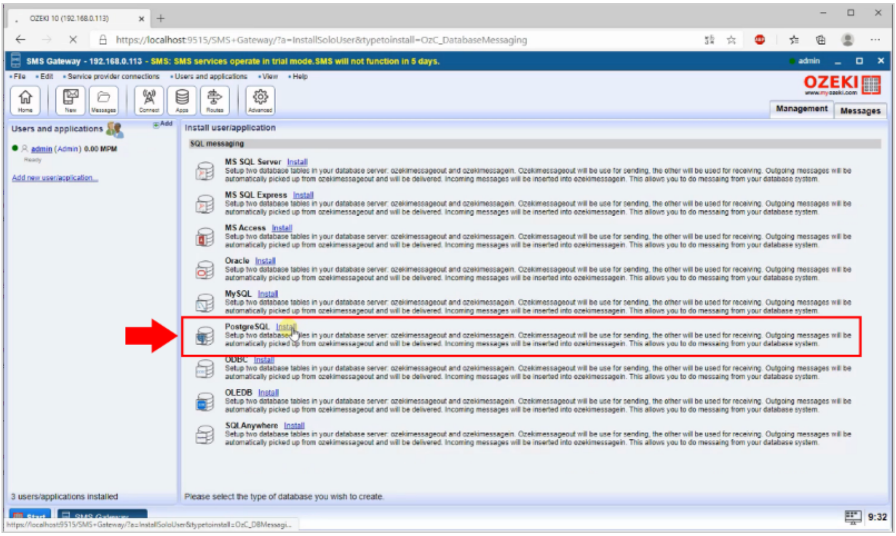
TXT file Install
The text file messaging configuration makes it easy to send and receive messages using text files. When an incoming message arrives Ozeki creates a text file for the message in your incoming directory on your hard disk. If you would like to send a message all you have to do is place a text file into the outgoing directory.

HTTP client Install
The HTTP client user passes incoming messages in HTTP 'GET' or 'POST' request to an external webserver. The HTML page returned by the external webserver can contain response SMS messages or any other outgoing SMS messages that will be delivered by Ozeki SMS Gateway.

FTP Install
Setup this FTP to SMS Extended service if you would like to send SMS messages stored on FTP server or save incoming SMS messages to FTP server. The outgoing SMS messages are in the outgoing directory while it uploads the incoming SMS messages to the server's incoming director. You can choose FTP, FTP Over SSL/TLS, and SSH FTP protocols.

Click install

**Figure 3** - Install SQL messaging User


OZEKI 10 (192.168.0.113)
https://localhost:9515/SMS+Gateway/?a=InstallSoloUser&typetoinstall=OzC_DatabaseMessaging
SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.SMS will not function in 5 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
OZEKI
www.myozeki.com
Management
Messages
Add
Start
SMS Gateway
9:32
https://localhost:9515/SMS+Gateway/?a=InstallSoloUser&typetoinstall=OzC_DBMessagi...

Users and applications

admin (Admin) 0.00 MPM
Ready

Add new user/application...

3 users/applications installed

Install user/application

SQL messaging

MS SQL Server Install
Setup two database tables in your database server: ozekimessageout and ozekimessagein. Ozekimessageout will be use for sending, the other will be used for receiving. Outgoing messages will be automatically picked up from ozekimessageout and will be delivered. Incoming messages will be inserted into ozekimessagein. This allows you to do messaing from your database system.

MS SQL Express Install
Setup two database tables in your database server: ozekimessageout and ozekimessagein. Ozekimessageout will be use for sending, the other will be used for receiving. Outgoing messages will be automatically picked up from ozekimessageout and will be delivered. Incoming messages will be inserted into ozekimessagein. This allows you to do messaing from your database system.

MS Access Install
Setup two database tables in your database server: ozekimessageout and ozekimessagein. Ozekimessageout will be use for sending, the other will be used for receiving. Outgoing messages will be automatically picked up from ozekimessageout and will be delivered. Incoming messages will be inserted into ozekimessagein. This allows you to do messaing from your database system.

Oracle Install
Setup two database tables in your database server: ozekimessageout and ozekimessagein. Ozekimessageout will be use for sending, the other will be used for receiving. Outgoing messages will be automatically picked up from ozekimessageout and will be delivered. Incoming messages will be inserted into ozekimessagein. This allows you to do messaing from your database system.

MySQL Install
Setup two database tables in your database server: ozekimessageout and ozekimessagein. Ozekimessageout will be use for sending, the other will be used for receiving. Outgoing messages will be automatically picked up from ozekimessageout and will be delivered. Incoming messages will be inserted into ozekimessagein. This allows you to do messaing from your database system.

PostgreSQL Install
Setup two database tables in your database server: ozekimessageout and ozekimessagein. Ozekimessageout will be use for sending, the other will be used for receiving. Outgoing messages will be automatically picked up from ozekimessageout and will be delivered. Incoming messages will be inserted into ozekimessagein. This allows you to do messaing from your database system.

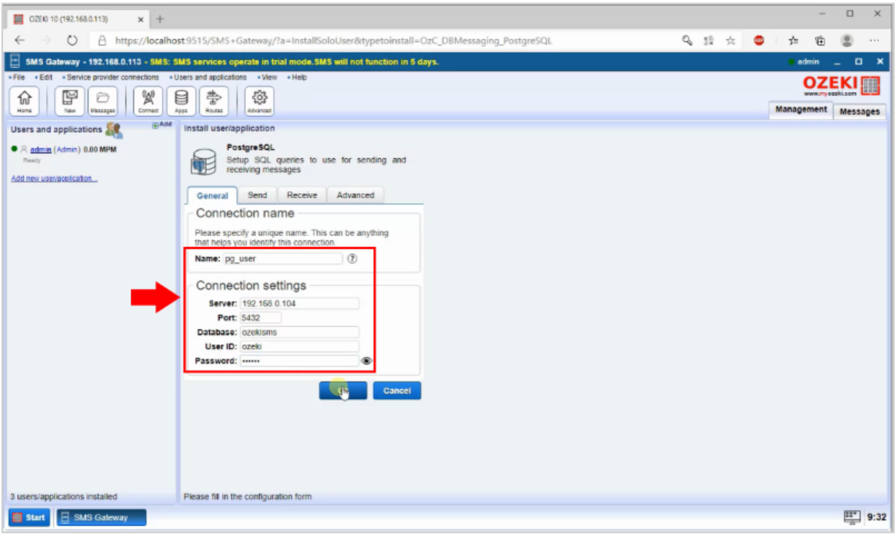
ODBC Install
Setup two database tables in your database server: ozekimessageout and ozekimessagein. Ozekimessageout will be use for sending, the other will be used for receiving. Outgoing messages will be automatically picked up from ozekimessageout and will be delivered. Incoming messages will be inserted into ozekimessagein. This allows you to do messaing from your database system.

OLEDB Install
Setup two database tables in your database server: ozekimessageout and ozekimessagein. Ozekimessageout will be use for sending, the other will be used for receiving. Outgoing messages will be automatically picked up from ozekimessageout and will be delivered. Incoming messages will be inserted into ozekimessagein. This allows you to do messaing from your database system.

SQL Anywhere Install
Setup two database tables in your database server: ozekimessageout and ozekimessagein. Ozekimessageout will be use for sending, the other will be used for receiving. Outgoing messages will be automatically picked up from ozekimessageout and will be delivered. Incoming messages will be inserted into ozekimessagein. This allows you to do messaing from your database system.

Please select the type of database you wish to create.

**Figure 4** - Install Postgre SQL User



SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.SMS will not function in 5 days.

Install user/application

PostgreSQL
Setup SQL queries to use for sending and receiving messages

General | Send | Receive | Advanced

Connection name

Please specify a unique name. This can be anything that helps you identify this connection.

Name: pg_user

Connection settings

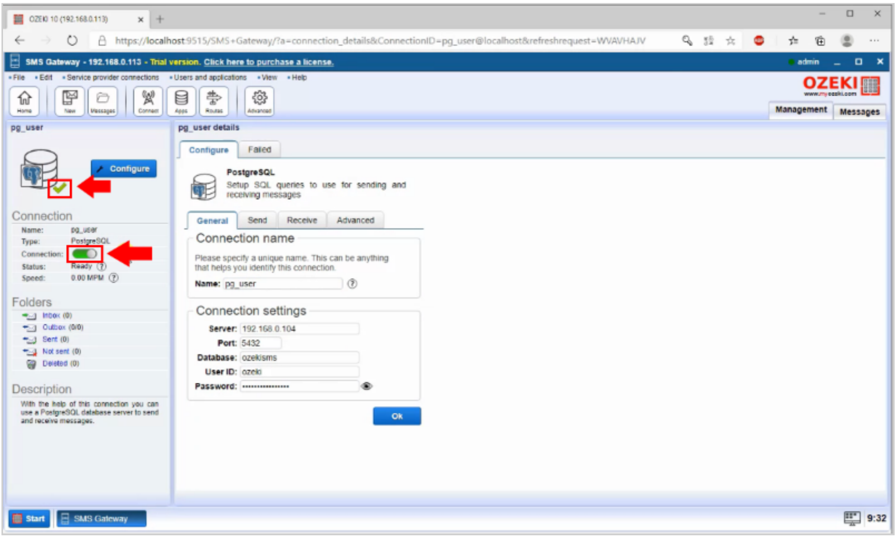
Server: 192.168.0.104
Port: 5432
Database: ozekisms
User ID: ozeki
Password: ••••••

Ok | Cancel

3 users/applications installed

Please fill in the configuration form

**Figure 5 -** Provide the connection details for the Postgre SQL server



SMS Gateway - 192.168.0.113 - Trial version. Click here to purchase a license.

pg_user

Configure

Connection
Name: pg_user
Type: PostgreSQL
Connection:
Status: Ready
Speed: 0.00 MPM

Folders
Inbox (0)
Outbox (0/0)
Sent (0)
Not sent (0)
Deleted (0)

Description
With the help of this connection you can use a PostgreSQL database server to send and receive messages.

pg_user details

Configure | Failed

PostgreSQL
Setup SQL queries to use for sending and receiving messages

General | Send | Receive | Advanced

Connection name

Please specify a unique name. This can be anything that helps you identify this connection.

Name: pg_user

Connection settings

Server: 192.168.0.104
Port: 5432
Database: ozekisms
User ID: ozeki
Password: ••••••••••••••••

Ok

**Figure 6 -** Enable Postgre SQL connection

# Send SMS from PostgreSQL (part 4/4)
## Send Test Message

**This is the last video in this video series which shows how to connect Ozeki SMS Gateway to a PostgreSQL database server for SMS messaging. See how to insert a simple row into the proper table, so Ozeki SMS Gateway can automatically send SMS messages to mobile phones.**

### Video content

1. Insert message into database
2. Send Test message

---

Look at the upcoming screenshots to thoroughly examine the final stage, which is SMS sending. You can start the whole process by INSERT-ing the SMS into the database (**Figure 2**).

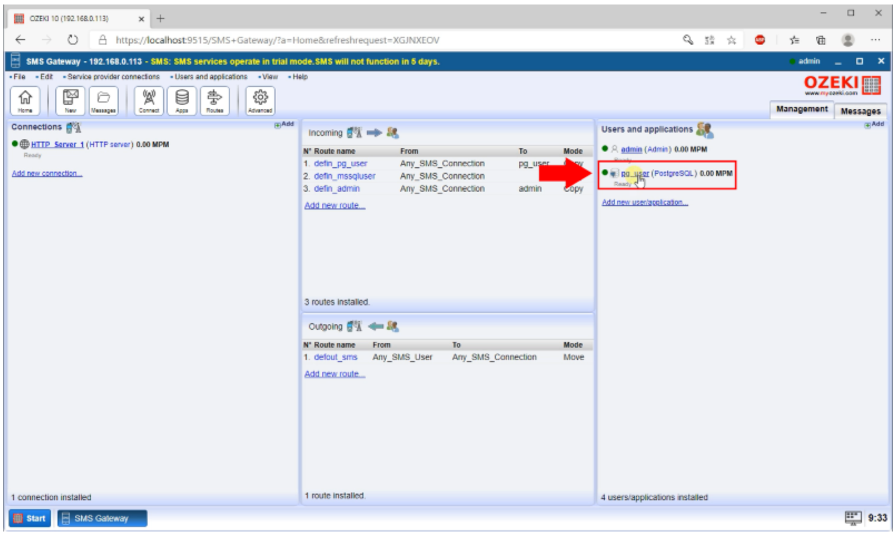
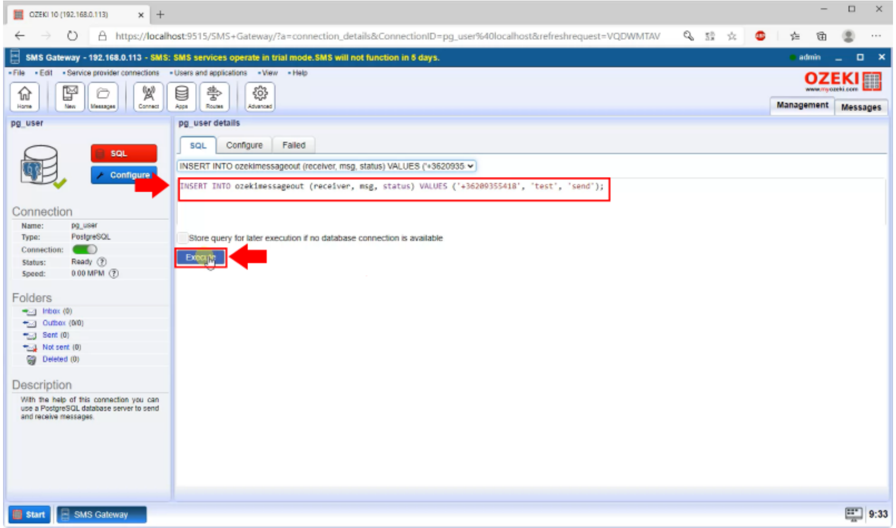

SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.SMS will not function in 5 days.
Connections
HTTP_Server_1 (HTTP server) 0.00 MPM
Add new connection...
Incoming
1. defin_pg_user Any_SMS_Connection pg_user
2. defin_mssqluser Any_SMS_Connection
3. defin_admin Any_SMS_Connection admin Copy
3 routes installed.
Outgoing
1. default_sms Any_SMS_User Any_SMS_Connection Move
1 route installed.
Users and applications
admin (Admin) 0.00 MPM
pg_user (PostgreSQL) 0.00 MPM
Add new user/application...
4 users/applications installed
1 connection installed
Start
SMS Gateway
9:33

**Figure 1** - Open Database user

**INSERT message record** (example):

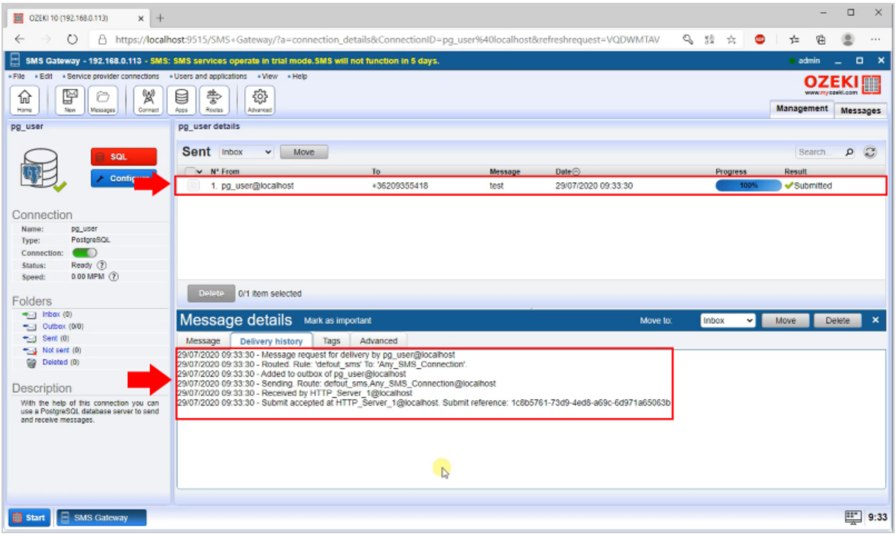
```
INSERT INTO ozekimessageout (receiver,msg,status) values ('+36201234567','Hello World','Send');
```

**Figure 2** - Copy INSERT statement


SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.SMS will not function in 5 days.
pg_user
SQL
Configure
Connection
Name: pg_user
Type: PostgreSQL
Status: Ready
Speed: 0.00 MPM
Folders
Inbox (0)
Outbox (0/0)
Sent (0)
Not sent (0)
Deleted (0)
Description
With the help of this connection you can use a PostgreSQL database server to send and receive messages.
pg_user details
SQL
Configure
Failed
Store query for later execution if no database connection is available
Execute
Start
SMS Gateway
9:33

```
INSERT INTO ozekimessageout (receiver, msg, status) VALUES ('+36209355418', 'test', 'send');
```

**Figure 3** - Paste INSERT statement to PostgreSQL's database table



Connection

| | |
|---|---|
| Name: | pg_user |
| Type: | PostgreSQL |
| Connection: | |
| Status: | Ready |
| Speed: | 0.00 MPM |

Folders

- Inbox (0)
- Outbox (0/0)
- Sent (0)
- Not sent (0)
- Deleted (0)

Description

With the help of this connection you can use a PostgreSQL database server to send and receive messages.

| N° | From | To | Message | Date | Progress | Result |
|---|---|---|---|---|---|---|
| 1. | pg_user@localhost | +36209355418 | test | 29/07/2020 09:33:30 | 100% | ✓Submitted |

0/1 item selected

```
29/07/2020 09:33:30 - Message request for delivery by pg_user@localhost
29/07/2020 09:33:30 - Routed. Rule: 'defout_sms' To: 'Any_SMS_Connection'.
29/07/2020 09:33:30 - Added to outbox of pg_user@localhost
29/07/2020 09:33:30 - Sending. Route: defout_sms,Any_SMS_Connection@localhost
29/07/2020 09:33:30 - Received by HTTP_Server_1@localhost
29/07/2020 09:33:30 - Submit accepted at HTTP_Server_1@localhost. Submit reference: 1c8b5761-73d9-4ed8-a69c-6d971a65063b
```

**Figure 4 -** The Database User's sent folder shows that your message has been sent by Ozeki SMS Gateway

# SMS from/to SAP SQL Anywhere

**See how to send and receive SMS messages through an SAP SQL Anywhere server with the Database User of Ozeki SMS Gateway. Here you can find a connection string and short CREATE TABLE statements to get started. You can send messages by inserting new message records.**

## Send SMS messages with SQL Anywhere

SQL Anywhere Installation(Video guide)
Create Database Tables (Video guide)
Configure Database User (Video guide)
Send Test Message (Video guide)

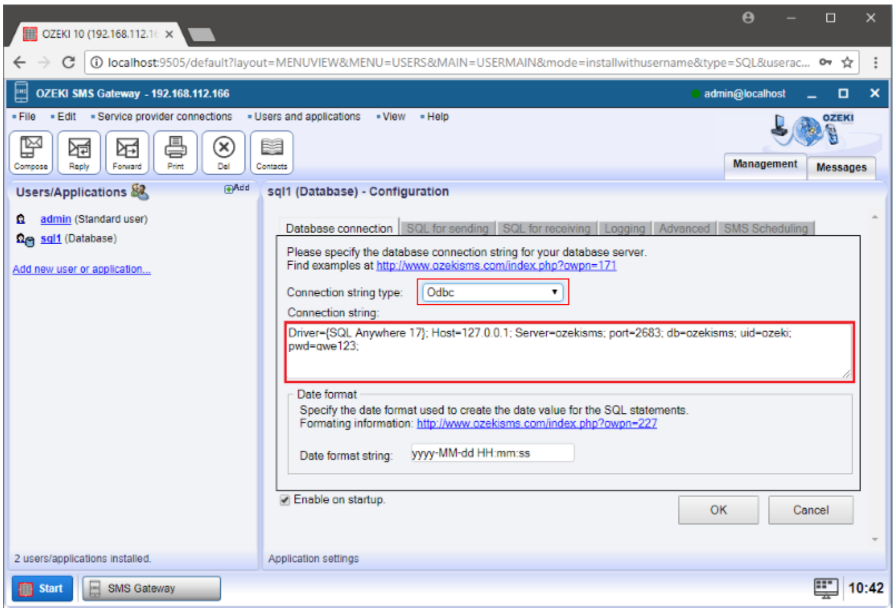
**You should have two important programs.** The **first one is an SAP SQL Anywhere server,** while the **second one is Ozeki SMS Gateway**, which must have a Database User installed, so it can connect to the SAP SQL Database Server using a connection string. Please create a table for the incoming messages and another table for the outgoing messages. You should call these tables as 'ozekimessagein' and 'ozekimessageout', which is used in the current example. You can see your received SMS messages by using a simple SELECT and you can send SMS message with a simple INSERT SQL statement.

The browser GUI of Ozeki SMS Gateway can be used for installing a Database User. Please **provide the ODBC connection string** for your SAP SQL Anywhere database. The Database User can be configured from the browser GUI. Keep in mind to select 'Odbc' and type the connection string (**Figure 2**).

| | |
|---|---|
| Connection Type: | **ODBC** |
| Connection String: | **Driver={SQL Anywhere 17}; Host=127.0.0.1;Server=ozekisms;port=2683;<br>db=ozekisms;uid=ozeki;pwd=qwe123;** |

**Figure 1 -** Example ODBC connection string, *which needs to be modified*

Please use the information of your SAP SQL Anywhere server and **change the connection string variables** (**Figure 2**).



sql1 (Database) - Configuration

Please specify the database connection string for your database server.
Find examples at http://www.ozekisms.com/index.php?owpn=171

Connection string type: Odbc

Connection string: Driver={SQL Anywhere 17}; Host=127.0.0.1; Server=ozekisms; port=2683; db=ozekisms; uid=ozeki; pwd=qwe123;

Date format
Specify the date format used to create the date value for the SQL statements.
Formating information: http://www.ozekisms.com/index.php?owpn=227

Date format string: yyyy-MM-dd HH:mm:ss

☑ Enable on startup.

**Figure 2 -** Modify the connection string variables according to your SAP SQL Anywhere database

# SAP SQL Anywhere CREATE TABLE script

Please create the 'ozekimessageout' and 'ozekimessagein' tables on your SAP SQL Anywhere database server.

```
CREATE TABLE "ozekimessagein"
(
 "id"       integer NOT NULL DEFAULT autoincrement ,
 "sender"          varchar(160) NULL ,
 "receiver"        varchar(160) NULL ,
 "msg"      varchar(160) NULL ,
 "senttime"        varchar(100) NULL ,
 "receivedtime"    varchar(100) NULL ,
 "operator"        varchar(100) NULL ,
 "msgtype"         varchar(160) NULL ,
 "reference"       varchar(100) NULL ,
  PRIMARY KEY ("id")
)
go
commit work
go

CREATE TABLE "ozekimessageout"
(
 "id"       integer NOT NULL DEFAULT autoincrement ,
 "sender"          varchar(160) NULL ,
 "receiver"        varchar(160) NULL ,
 "msg"      varchar(160) NULL ,
 "senttime"        varchar(100) NULL ,
 "receivedtime"    varchar(100) NULL ,
 "operator"        varchar(100) NULL ,
 "status"     varchar(20)  NULL ,
 "msgtype"         varchar(160) NULL ,
 "reference"       varchar(100) NULL ,
 "errormsg"       varchar(250) NULL ,
  PRIMARY KEY ("id")
)
go
commit work
go

CREATE INDEX "ozekimessageinindex"
ON "ozekimessagein"
(
 "id" ASC
)
go
commit work
go

CREATE INDEX "ozekimessageoutindex"
ON "ozekimessageout"
(
 "id" ASC
)
go
commit work
go
```

The size of the 'msg' field can be always increased above 160 characters. You can also change it's data type as well.

The index of each record is the 'id' attribute. Please maintain the 'id' in all tables.

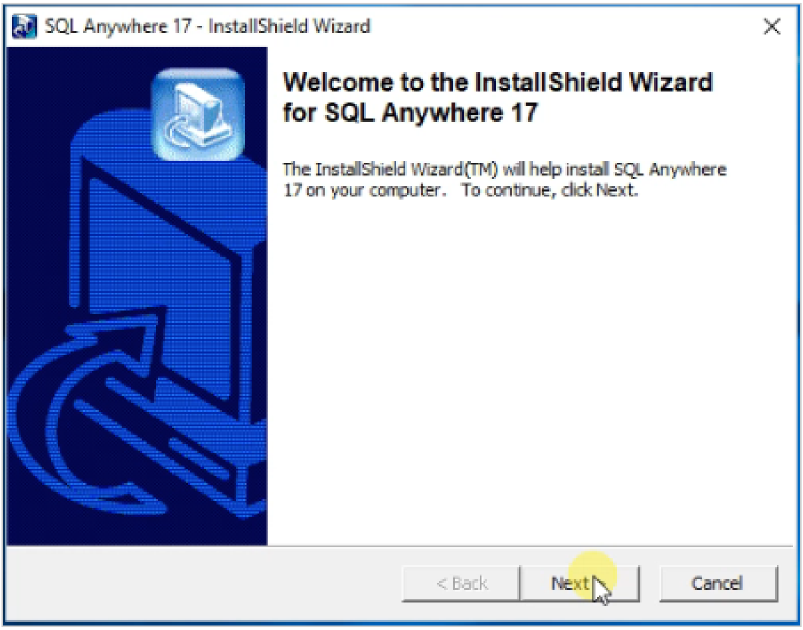
# Send SMS from SQL Anywhere (part 1/4)

## SQL Anywhere Installation

**In this video series you can see how to connect Ozeki SMS Gateway to an SAP SQL Anywhere database server for SMS messaging. The first video shows how to install SQL Anywhere. Please download it from the following page.**

**Video content**

1. Install SQL Anywhere

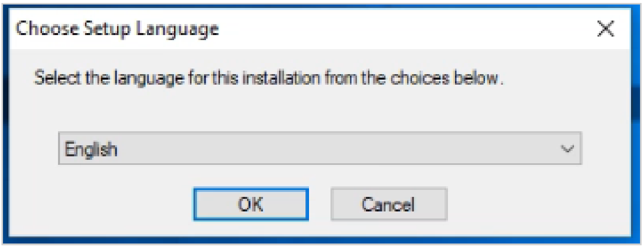


SQL Anywhere 17 - InstallShield Wizard

Welcome to the InstallShield Wizard for SQL Anywhere 17

The InstallShield Wizard(TM) will help install SQL Anywhere 17 on your computer. To continue, click Next.

**Figure 1** - Start installation

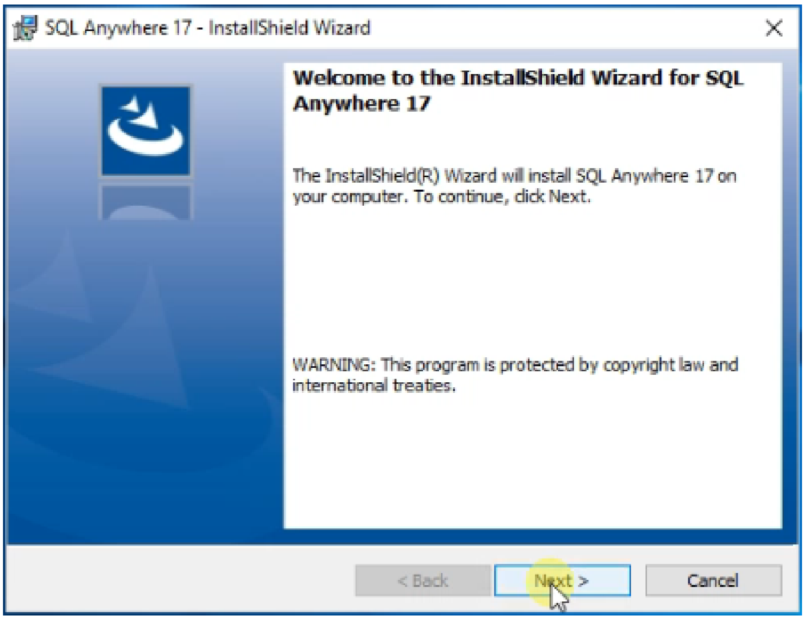


Choose Setup Language

Select the language for this installation from the choices below.

English

**Figure 2** - Select language



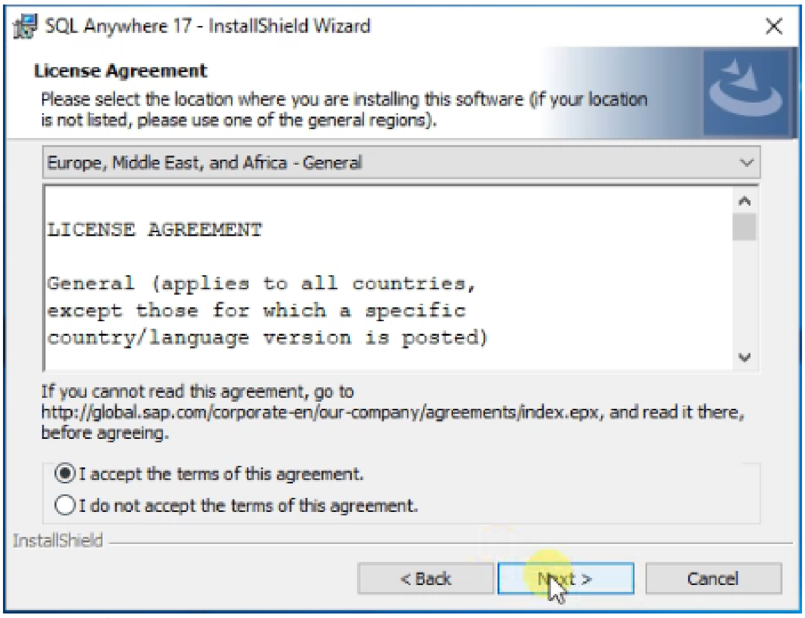
SQL Anywhere 17 - InstallShield Wizard

Welcome to the InstallShield Wizard for SQL Anywhere 17

The InstallShield(R) Wizard will install SQL Anywhere 17 on your computer. To continue, click Next.

WARNING: This program is protected by copyright law and international treaties.

**Figure 3** - Click 'Next'



SQL Anywhere 17 - InstallShield Wizard

**License Agreement**

Please select the location where you are installing this software (if your location is not listed, please use one of the general regions).

Europe, Middle East, and Africa - General

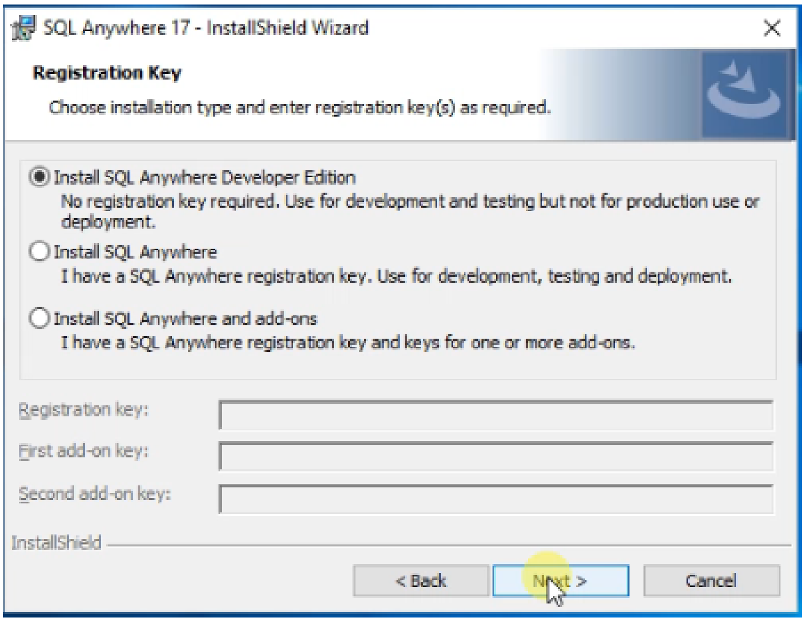
```
LICENSE AGREEMENT

General (applies to all countries,
except those for which a specific
country/language version is posted)
```

If you cannot read this agreement, go to http://global.sap.com/corporate-en/our-company/agreements/index.epx, and read it there, before agreeing.

● I accept the terms of this agreement.

○ I do not accept the terms of this agreement.

**Figure 4** - Accept license terms



SQL Anywhere 17 - InstallShield Wizard

**Registration Key**

Choose installation type and enter registration key(s) as required.

● Install SQL Anywhere Developer Edition
No registration key required. Use for development and testing but not for production use or deployment.

○ Install SQL Anywhere
I have a SQL Anywhere registration key. Use for development, testing and deployment.

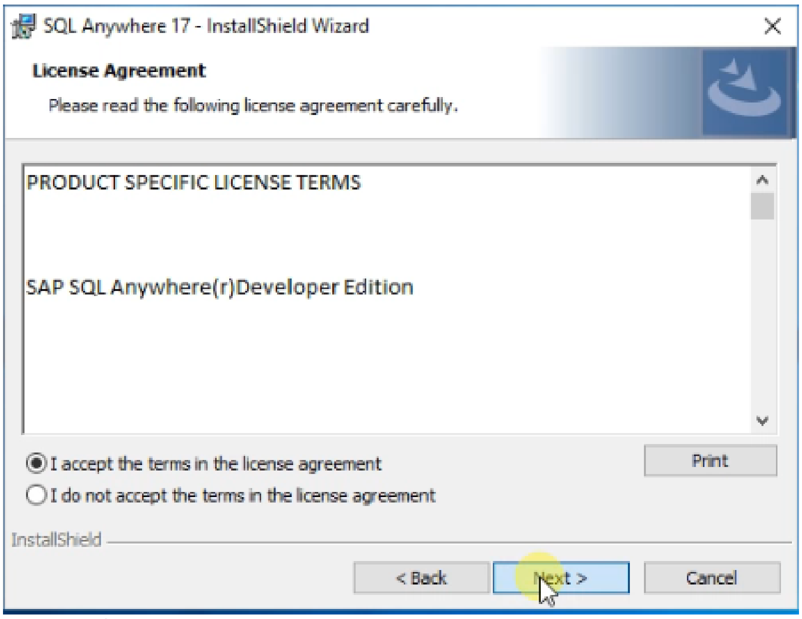
○ Install SQL Anywhere and add-ons
I have a SQL Anywhere registration key and keys for one or more add-ons.

Registration key:

First add-on key:

Second add-on key:

**Figure 5** - Select installation type



SQL Anywhere 17 - InstallShield Wizard

**License Agreement**

Please read the following license agreement carefully.

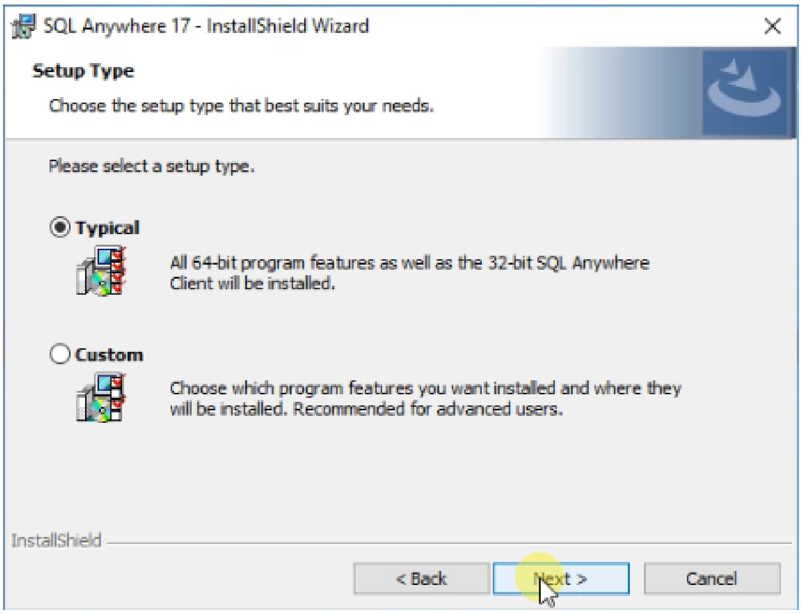
PRODUCT SPECIFIC LICENSE TERMS

SAP SQL Anywhere(r)Developer Edition

● I accept the terms in the license agreement

○ I do not accept the terms in the license agreement

**Figure 6** - Accept this other license agreement


SQL Anywhere 17 - InstallShield Wizard

**Setup Type**

Choose the setup type that best suits your needs.

Please select a setup type.

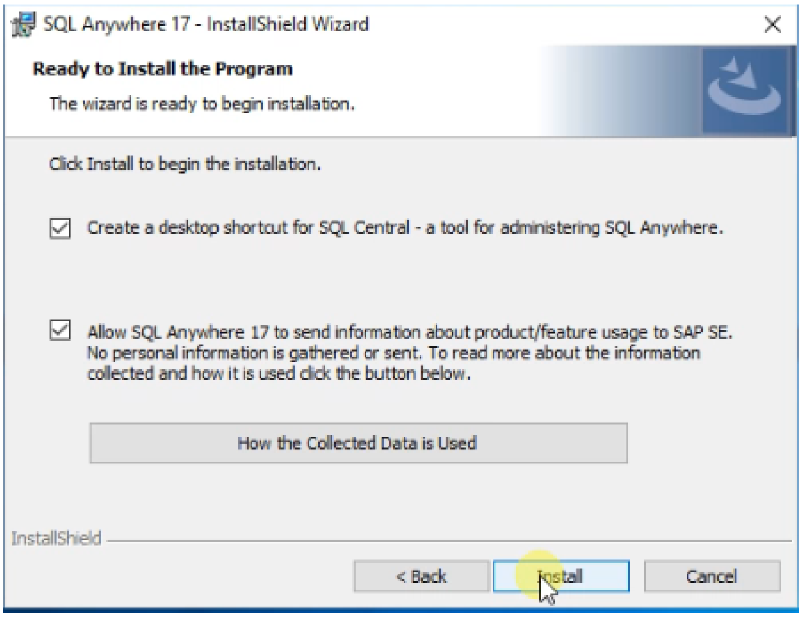
● **Typical**

All 64-bit program features as well as the 32-bit SQL Anywhere Client will be installed.

○ **Custom**

Choose which program features you want installed and where they will be installed. Recommended for advanced users.

**Figure 7** - Select setup typee


SQL Anywhere 17 - InstallShield Wizard

**Ready to Install the Program**

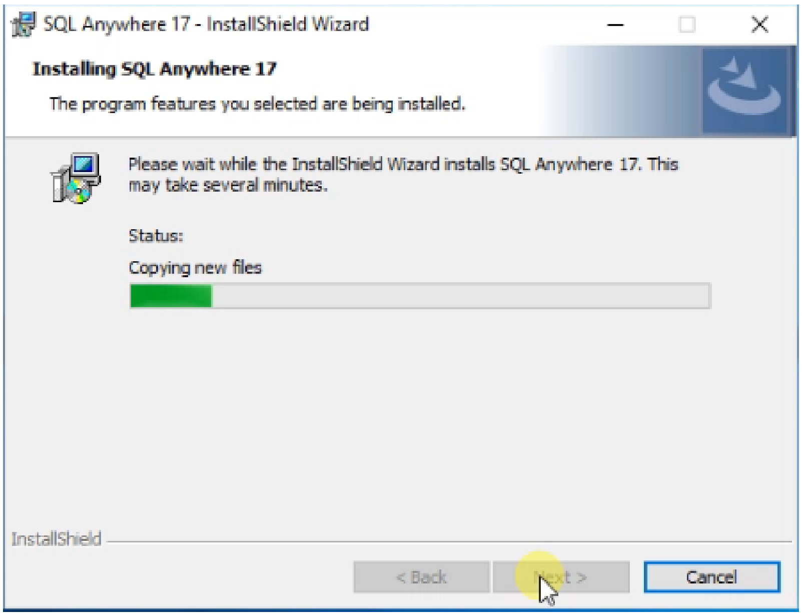
The wizard is ready to begin installation.

Click Install to begin the installation.

☑ Create a desktop shortcut for SQL Central - a tool for administering SQL Anywhere.

☑ Allow SQL Anywhere 17 to send information about product/feature usage to SAP SE. No personal information is gathered or sent. To read more about the information collected and how it is used click the button below.

**Figure 8** - Click 'Install' button


SQL Anywhere 17 - InstallShield Wizard

**Installing SQL Anywhere 17**

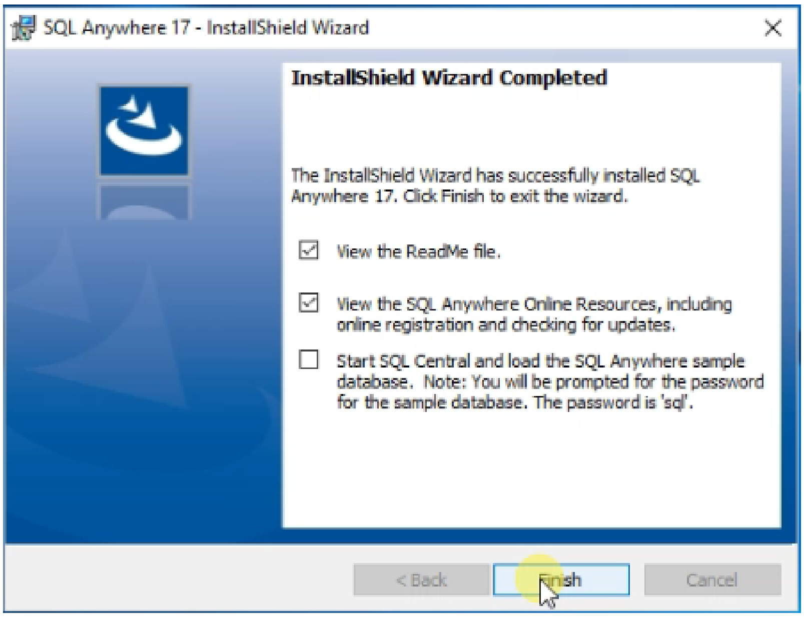
The program features you selected are being installed.

Please wait while the InstallShield Wizard installs SQL Anywhere 17. This may take several minutes.

Status:

Copying new files

**Figure 9** - Installation is in progress



SQL Anywhere 17 - InstallShield Wizard

**InstallShield Wizard Completed**

The InstallShield Wizard has successfully installed SQL Anywhere 17. Click Finish to exit the wizard.

☑ View the ReadMe file.

☑ View the SQL Anywhere Online Resources, including online registration and checking for updates.

□ Start SQL Central and load the SQL Anywhere sample database. Note: You will be prompted for the password for the sample database. The password is 'sql'.

**Figure 10 -** Installation has been finished

# Send SMS from SQL Anywhere (part 2/4)
## Create Database Tables

**In this video series you can see how to connect Ozeki SMS Gateway to an SQL Anywhere database server for SMS messaging. This video shows how to create the proper database table structure by using the GUI of your SQL Anywhere database server.**

**Video content**
1. Start SQL Central
2. Create Database
3. Connect to Database
4. Run CREATE TABLE statements

---

Please **scroll down to copy the SQL statements** (**Figure 1**) used in the video. If you have created the database in SQL Anywhere, you can jump to the next video.

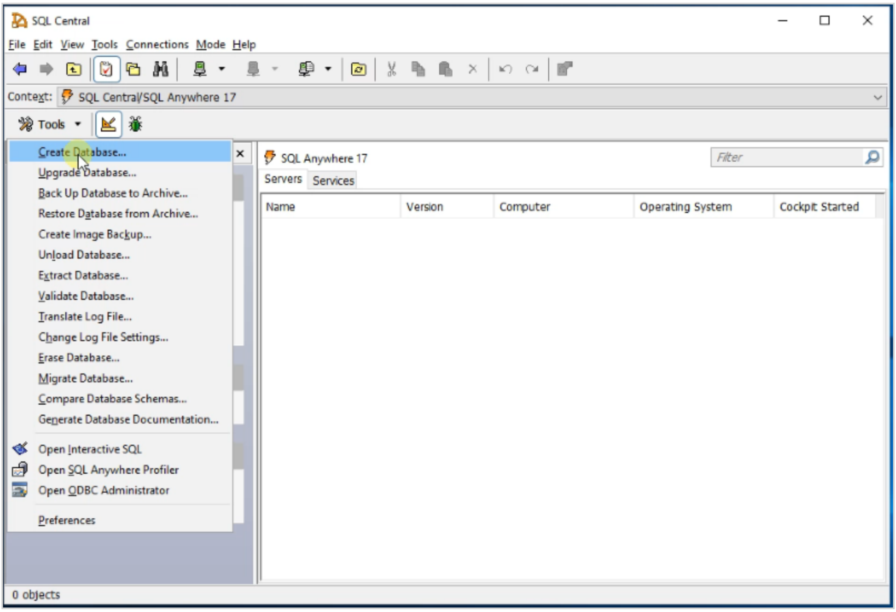
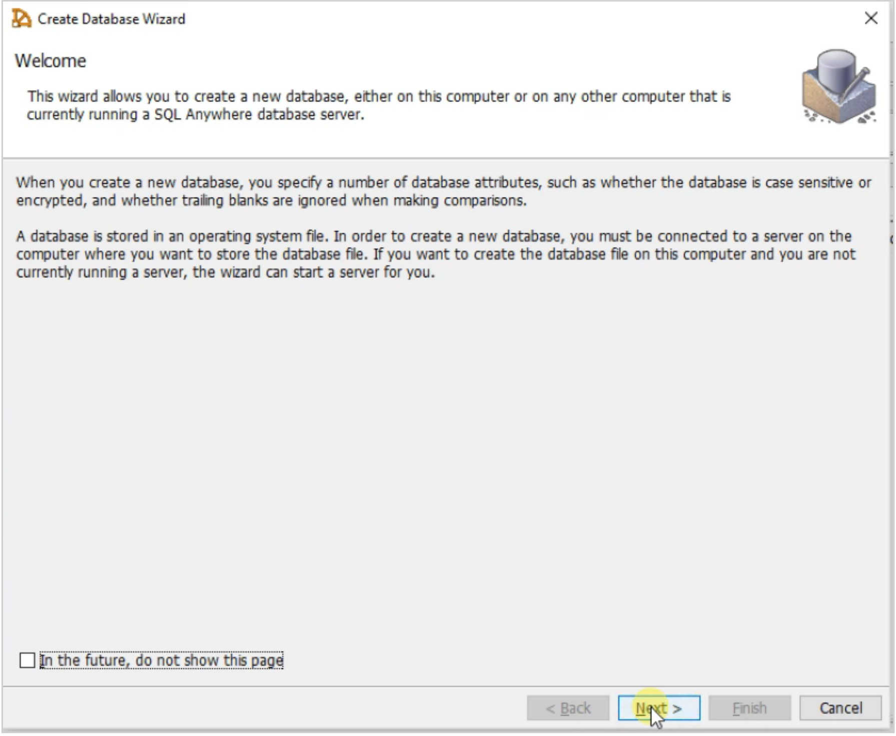


**Figure 1** - Create SQL Anywhere database by clicking 'Create database...'



Welcome

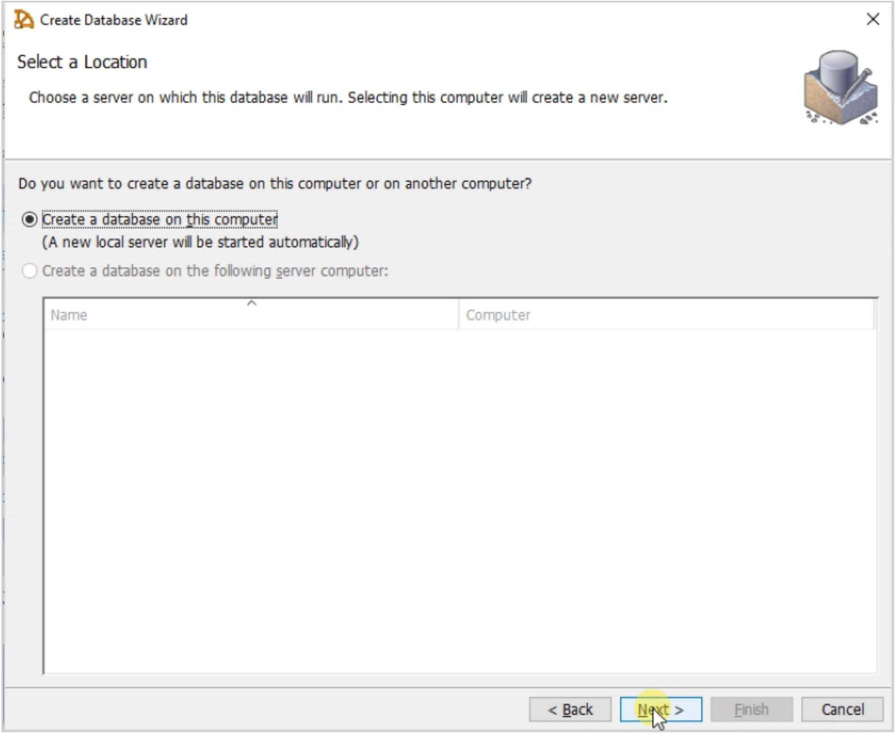
This wizard allows you to create a new database, either on this computer or on any other computer that is currently running a SQL Anywhere database server.

When you create a new database, you specify a number of database attributes, such as whether the database is case sensitive or encrypted, and whether trailing blanks are ignored when making comparisons.

A database is stored in an operating system file. In order to create a new database, you must be connected to a server on the computer where you want to store the database file. If you want to create the database file on this computer and you are not currently running a server, the wizard can start a server for you.

☐ In the future, do not show this page

**Figure 2** - Create Database Wizard



Select a Location

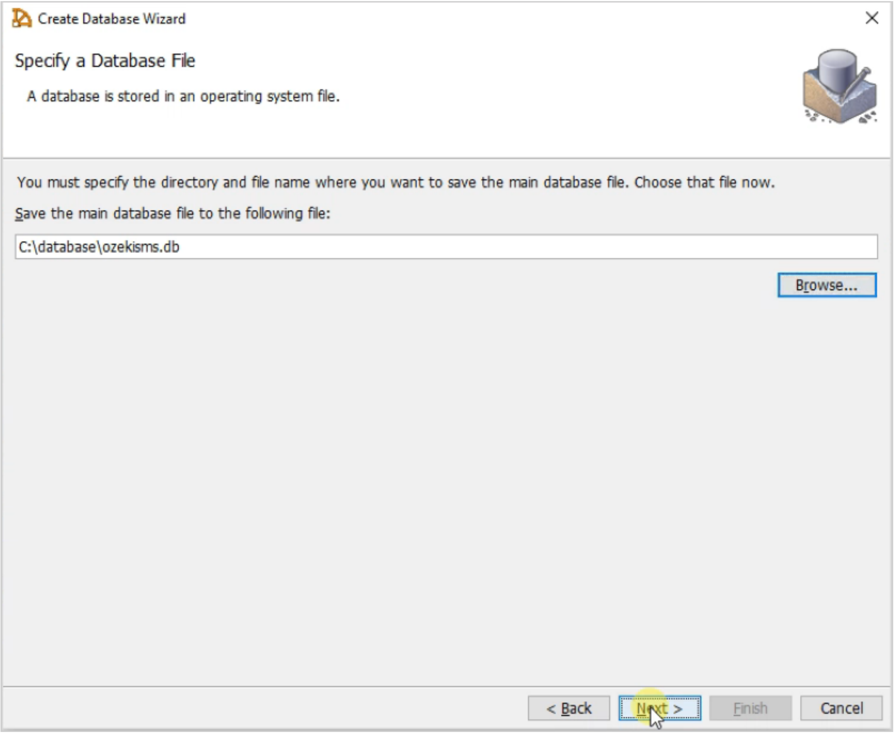
Choose a server on which this database will run. Selecting this computer will create a new server.

Do you want to create a database on this computer or on another computer?

● Create a database on this computer
(A new local server will be started automatically)

○ Create a database on the following server computer:

**Figure 3** - Select the machine where to store the database



Specify a Database File

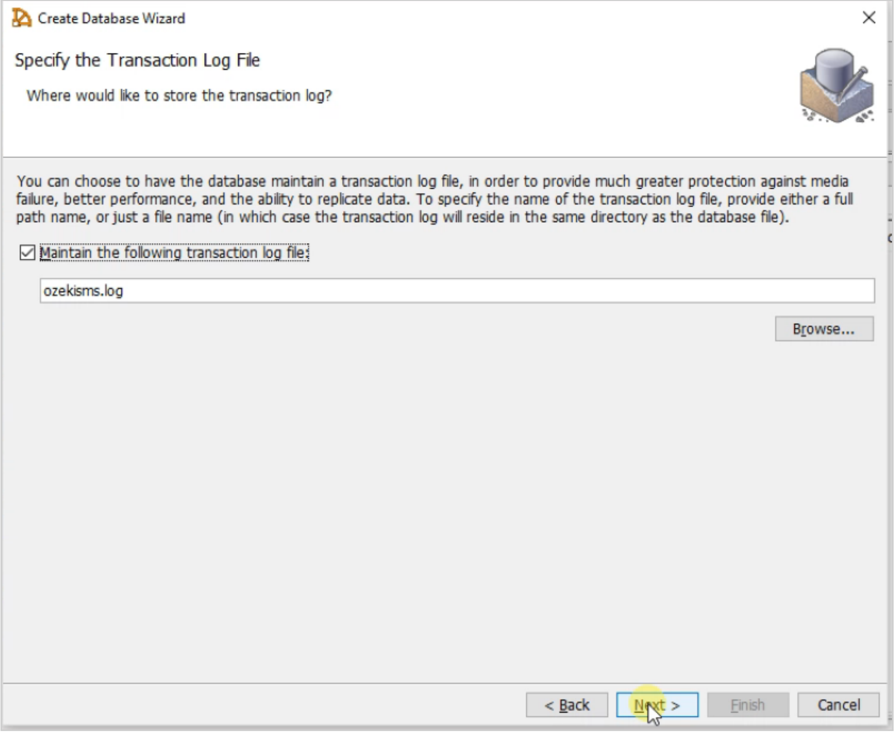
A database is stored in an operating system file.

You must specify the directory and file name where you want to save the main database file. Choose that file now.

Save the main database file to the following file:

C:\database\ozekisms.db

**Figure 4** - Specify the database's location in the filesystem


Create Database Wizard

Specify the Transaction Log File

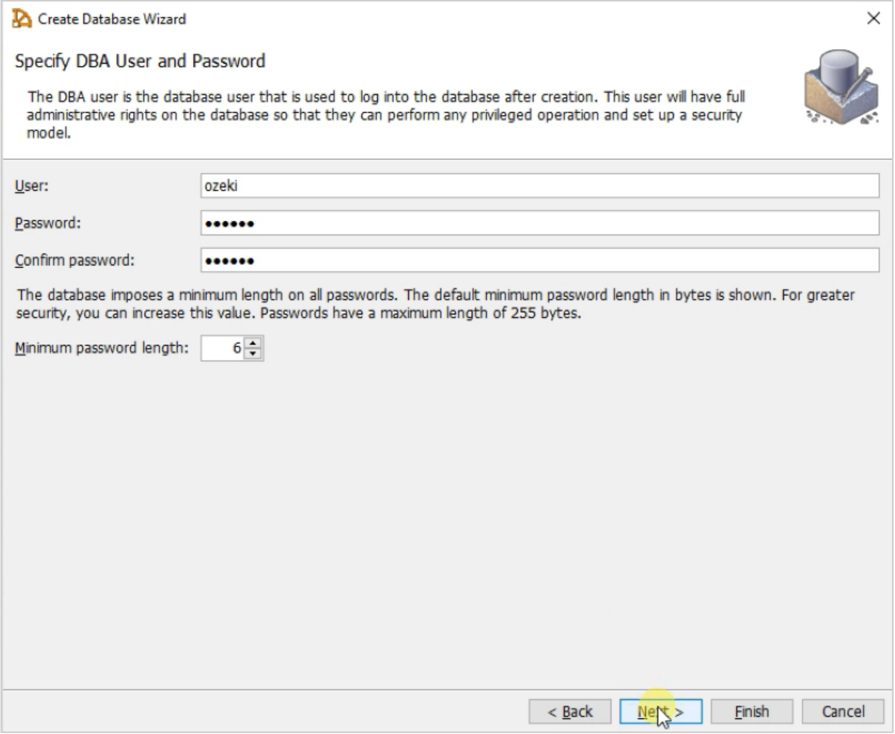
Where would like to store the transaction log?

You can choose to have the database maintain a transaction log file, in order to provide much greater protection against media failure, better performance, and the ability to replicate data. To specify the name of the transaction log file, provide either a full path name, or just a file name (in which case the transaction log will reside in the same directory as the database file).

☑ Maintain the following transaction log file:

ozekisms.log

**Figure 5 -** Select where to use the transaction log on your filesystem


Create Database Wizard

Specify DBA User and Password

The DBA user is the database user that is used to log into the database after creation. This user will have full administrative rights on the database so that they can perform any privileged operation and set up a security model.

User: ozeki

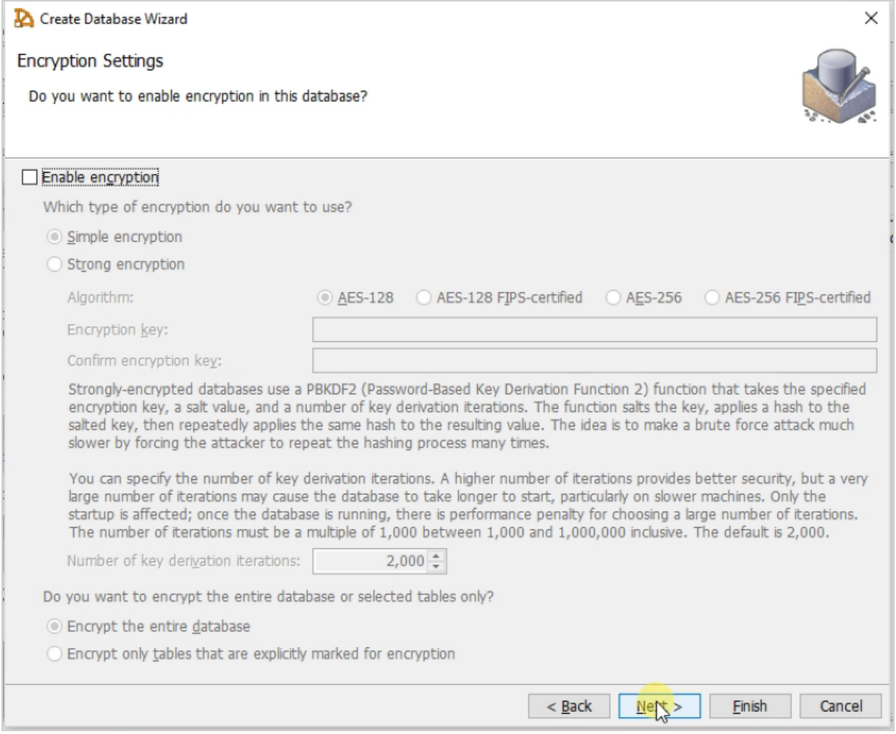
Password: ••••••

Confirm password: ••••••

The database imposes a minimum length on all passwords. The default minimum password length in bytes is shown. For greater security, you can increase this value. Passwords have a maximum length of 255 bytes.

Minimum password length: 6

**Figure 6 -** Create a DBA user. This user will have rights to the database



Encryption Settings

Do you want to enable encryption in this database?

□ Enable encryption

Which type of encryption do you want to use?

● Simple encryption

○ Strong encryption

Algorithm: ● AES-128 ○ AES-128 FIPS-certified ○ AES-256 ○ AES-256 FIPS-certified

Encryption key:

Confirm encryption key:

Strongly-encrypted databases use a PBKDF2 (Password-Based Key Derivation Function 2) function that takes the specified encryption key, a salt value, and a number of key derivation iterations. The function salts the key, applies a hash to the salted key, then repeatedly applies the same hash to the resulting value. The idea is to make a brute force attack much slower by forcing the attacker to repeat the hashing process many times.

You can specify the number of key derivation iterations. A higher number of iterations provides better security, but a very large number of iterations may cause the database to take longer to start, particularly on slower machines. Only the startup is affected; once the database is running, there is performance penalty for choosing a large number of iterations. The number of iterations must be a multiple of 1,000 between 1,000 and 1,000,000 inclusive. The default is 2,000.

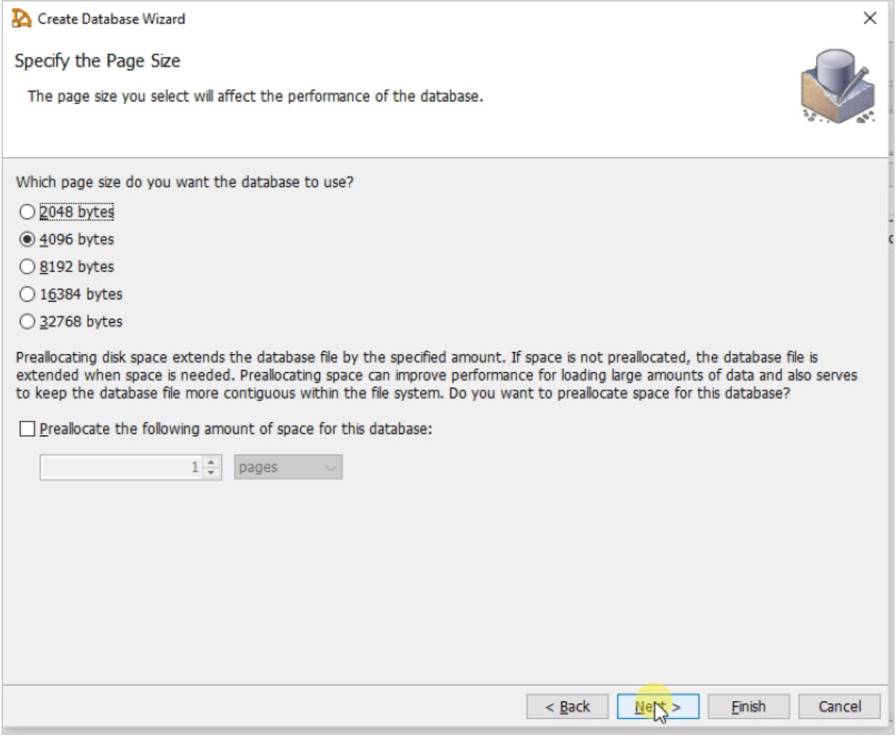
Number of key derivation iterations: 2,000

Do you want to encrypt the entire database or selected tables only?

● Encrypt the entire database

○ Encrypt only tables that are explicitly marked for encryption

**Figure 7 -** Select the type of encryption you prefer



Specify the Page Size

The page size you select will affect the performance of the database.

Which page size do you want the database to use?

○ 2048 bytes

● 4096 bytes

○ 8192 bytes

○ 16384 bytes

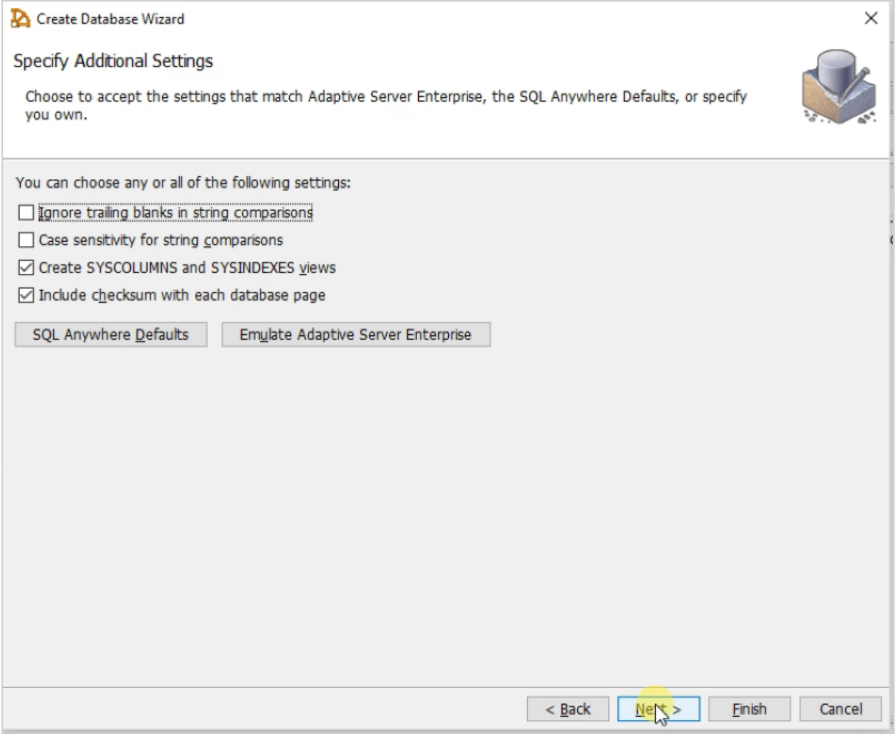
○ 32768 bytes

Preallocating disk space extends the database file by the specified amount. If space is not preallocated, the database file is extended when space is needed. Preallocating space can improve performance for loading large amounts of data and also serves to keep the database file more contiguous within the file system. Do you want to preallocate space for this database?

□ Preallocate the following amount of space for this database:

1 pages

**Figure 8 -** Specify the size of the database you wish to create



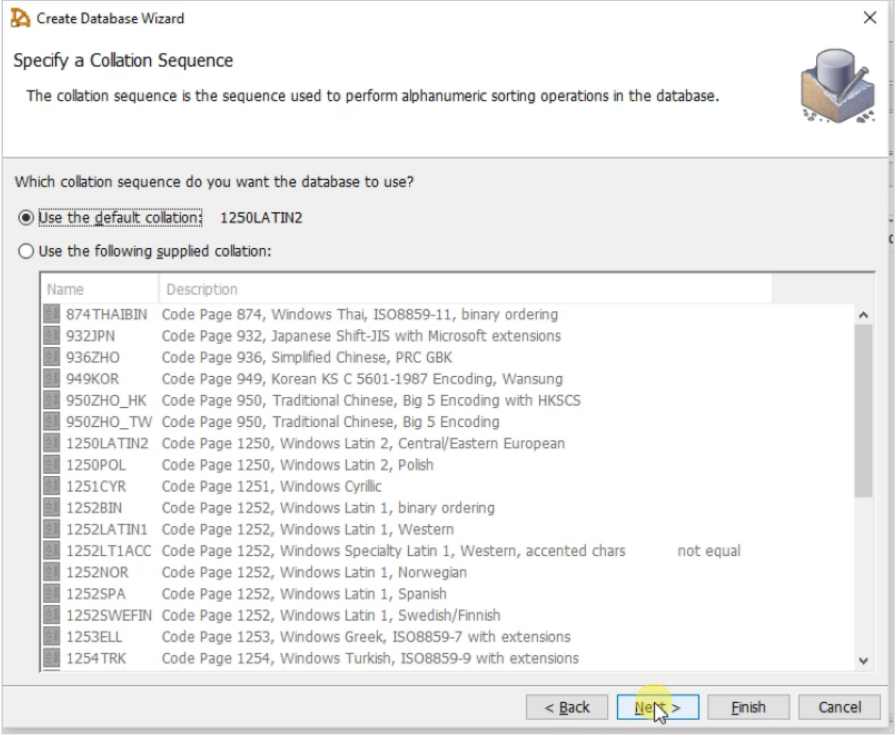
Specify Additional Settings

Choose to accept the settings that match Adaptive Server Enterprise, the SQL Anywhere Defaults, or specify you own.

You can choose any or all of the following settings:

- ☐ Ignore trailing blanks in string comparisons
- ☐ Case sensitivity for string comparisons
- ☑ Create SYSCOLUMNS and SYSINDEXES views
- ☑ Include checksum with each database page

**Figure 9 -** Choose additional database settings



Specify a Collation Sequence

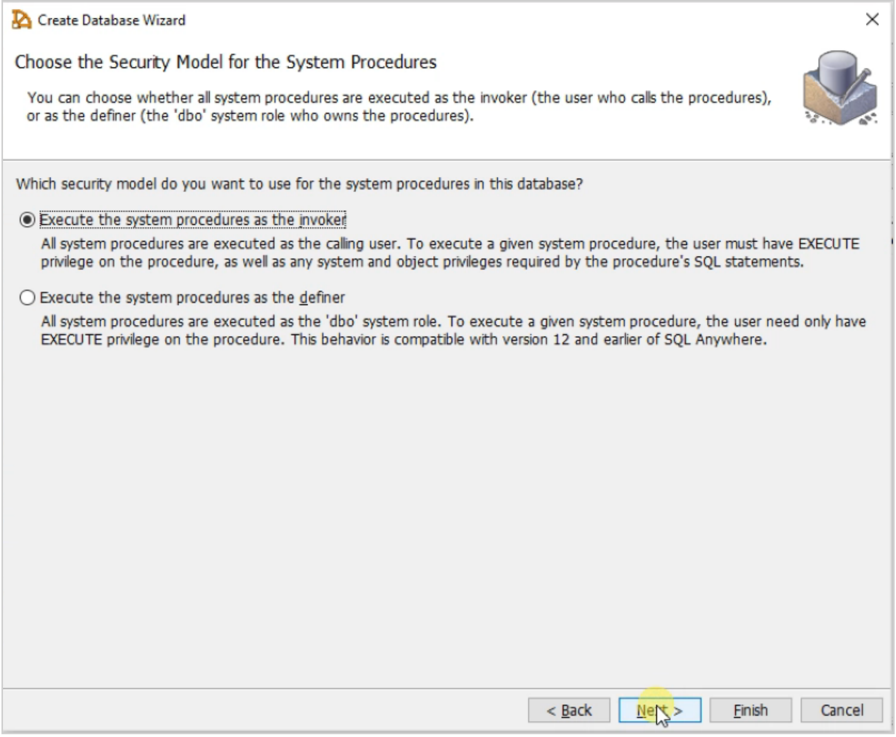
The collation sequence is the sequence used to perform alphanumeric sorting operations in the database.

Which collation sequence do you want the database to use?

- ● Use the default collation: 1250LATIN2
- ○ Use the following supplied collation:

| Name | Description | |
|---|---|---|
| 874THAIBIN | Code Page 874, Windows Thai, ISO8859-11, binary ordering | |
| 932JPN | Code Page 932, Japanese Shift-JIS with Microsoft extensions | |
| 936ZHO | Code Page 936, Simplified Chinese, PRC GBK | |
| 949KOR | Code Page 949, Korean KS C 5601-1987 Encoding, Wansung | |
| 950ZHO_HK | Code Page 950, Traditional Chinese, Big 5 Encoding with HKSCS | |
| 950ZHO_TW | Code Page 950, Traditional Chinese, Big 5 Encoding | |
| 1250LATIN2 | Code Page 1250, Windows Latin 2, Central/Eastern European | |
| 1250POL | Code Page 1250, Windows Latin 2, Polish | |
| 1251CYR | Code Page 1251, Windows Cyrillic | |
| 1252BIN | Code Page 1252, Windows Latin 1, binary ordering | |
| 1252LATIN1 | Code Page 1252, Windows Latin 1, Western | |
| 1252LT1ACC | Code Page 1252, Windows Specialty Latin 1, Western, accented chars | not equal |
| 1252NOR | Code Page 1252, Windows Latin 1, Norwegian | |
| 1252SPA | Code Page 1252, Windows Latin 1, Spanish | |
| 1252SWEFIN | Code Page 1252, Windows Latin 1, Swedish/Finnish | |
| 1253ELL | Code Page 1253, Windows Greek, ISO8859-7 with extensions | |
| 1254TRK | Code Page 1254, Windows Turkish, ISO8859-9 with extensions | |

**Figure 10 -** Specify the collation sequence to perform alphanumberic sorting operations in the database


Create Database Wizard

Choose the Security Model for the System Procedures

You can choose whether all system procedures are executed as the invoker (the user who calls the procedures), or as the definer (the 'dbo' system role who owns the procedures).

Which security model do you want to use for the system procedures in this database?

● Execute the system procedures as the invoker

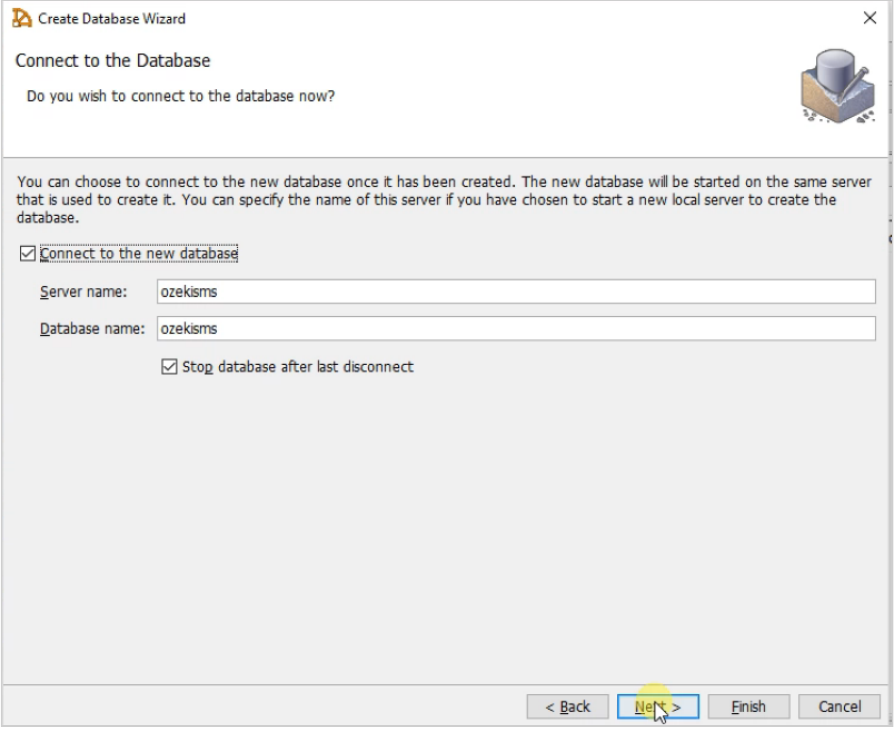
All system procedures are executed as the calling user. To execute a given system procedure, the user must have EXECUTE privilege on the procedure, as well as any system and object privileges required by the procedure's SQL statements.

○ Execute the system procedures as the definer

All system procedures are executed as the 'dbo' system role. To execute a given system procedure, the user need only have EXECUTE privilege on the procedure. This behavior is compatible with version 12 and earlier of SQL Anywhere.

**Figure 11** - Choose the security model for the system procedures

Your database have been created, so you can add a name and connect to it (**Figure 12**)


Create Database Wizard

Connect to the Database

Do you wish to connect to the database now?

You can choose to connect to the new database once it has been created. The new database will be started on the same server that is used to create it. You can specify the name of this server if you have chosen to start a new local server to create the database.

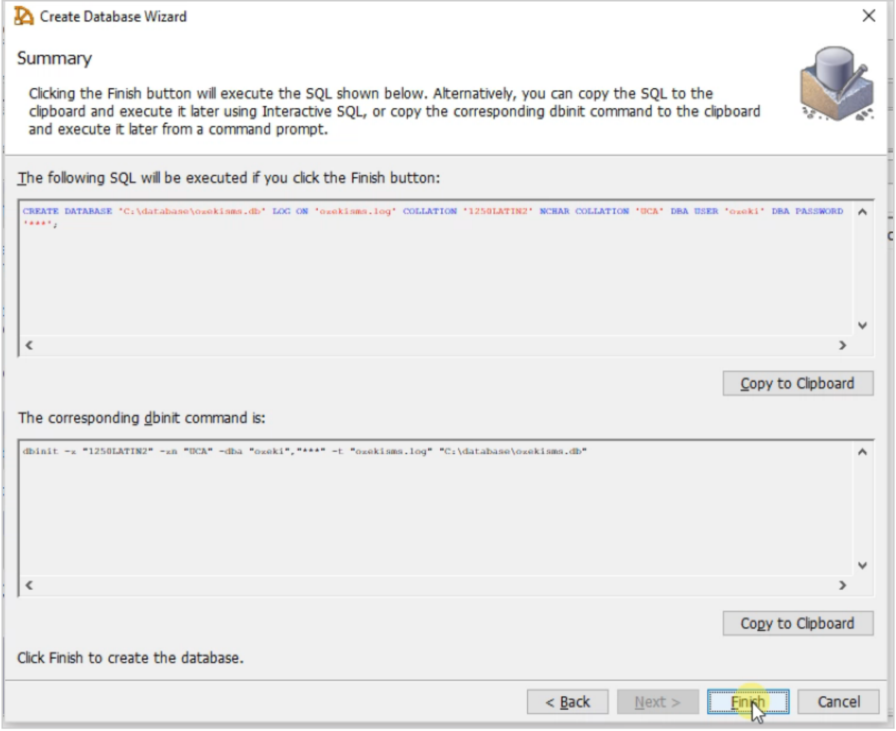
☑ Connect to the new database

Server name: ozekisms

Database name: ozekisms

☑ Stop database after last disconnect

**Figure 12** - Connect to database by using the server and database name



Summary

Clicking the Finish button will execute the SQL shown below. Alternatively, you can copy the SQL to the clipboard and execute it later using Interactive SQL, or copy the corresponding dbinit command to the clipboard and execute it later from a command prompt.

The following SQL will be executed if you click the Finish button:

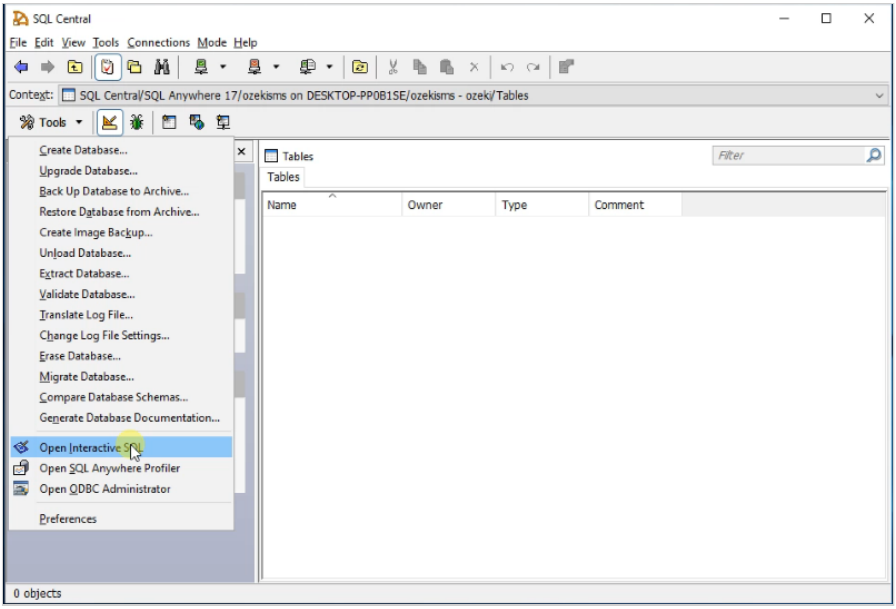
```
CREATE DATABASE 'C:\database\ozekisms.db' LOG ON 'ozekisms.log' COLLATION '1250LATIN2' NCHAR COLLATION 'UCA' DBA USER 'ozeki' DBA PASSWORD
'***';
```

The corresponding dbinit command is:

```
dbinit -z "1250LATIN2" -zn "UCA" -dba "ozeki","***" -t "ozekisms.log" "C:\database\ozekisms.db"
```

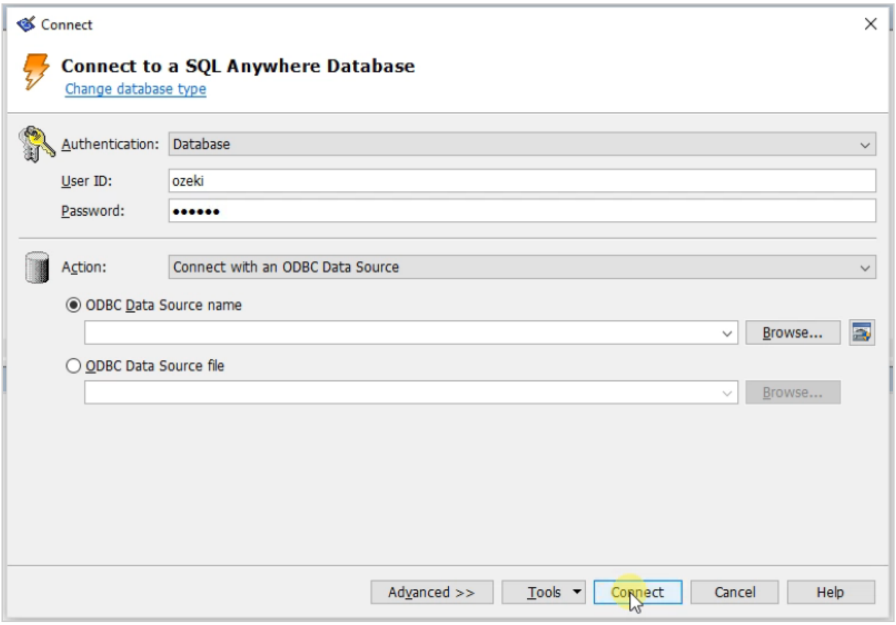
Click Finish to create the database.

**Figure 13 -** Run CREATE DATABASE SQL statement



Context: SQL Central/SQL Anywhere 17/ozekisms on DESKTOP-PP0B1SE/ozekisms - ozeki/Tables

**Figure 14 -** Open interactive SQL in the freshly opened SQL Central



Authentication: Database

User ID: ozeki

Password: ••••••

Action: Connect with an ODBC Data Source

● ODBC Data Source name

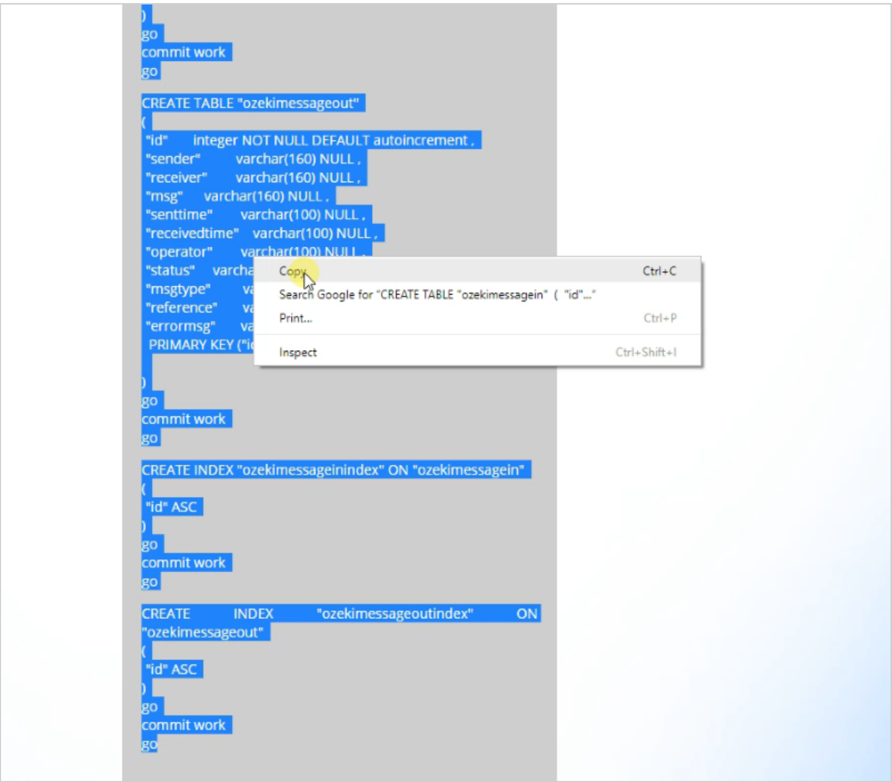
○ ODBC Data Source file

**Figure 15 -** Connect to your database with the DBA user credentials you have previously set

**<u>SQL Anywhere CREATE TABLE statements to copy:</u>**

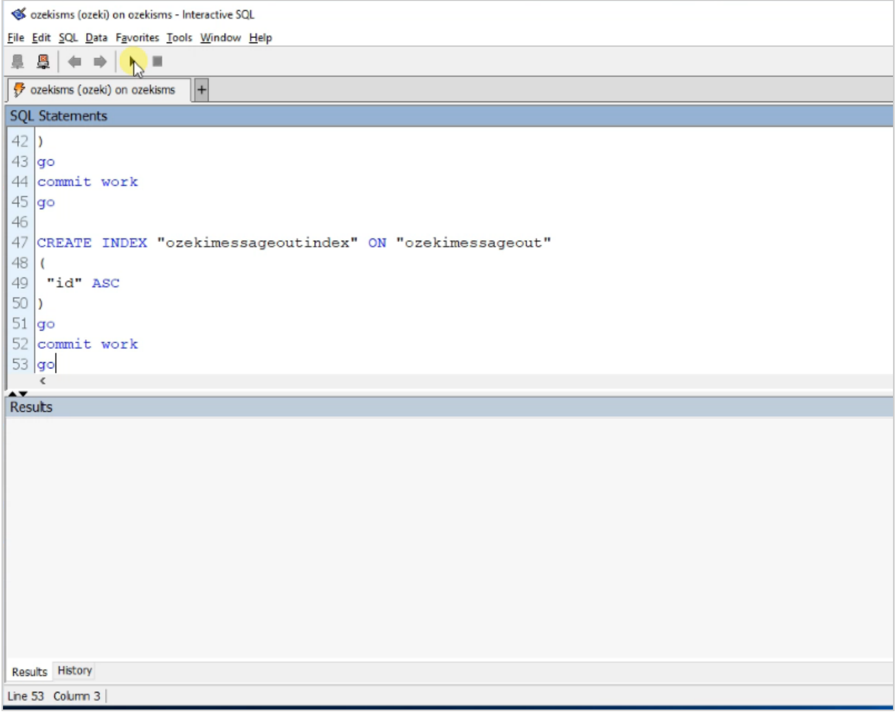
```
CREATE TABLE "ozekimessagein"
<br>(<br> "id"       integer NOT NULL DEFAULT autoincrement ,
<br> "sender"          varchar(160) NULL ,
<br> "receiver"        varchar(160) NULL ,
<br> "msg"      varchar(160) NULL ,
<br> "senttime"        varchar(100) NULL ,
<br> "receivedtime"    varchar(100) NULL ,
<br> "operator"        varchar(100) NULL ,
<br> "msgtype"         varchar(160) NULL ,
<br> "reference"       varchar(100) NULL ,
<br>  PRIMARY KEY ("id"),
<br>)
<br>go
<br>commit work
<br>go
<br>
<br>CREATE TABLE "ozekimessageout"
<br>(
<br> "id"       integer NOT NULL DEFAULT autoincrement ,
<br> "sender"          varchar(160) NULL ,
<br> "receiver"        varchar(160) NULL ,
<br> "msg"      varchar(160) NULL ,
<br> "senttime"        varchar(100) NULL ,
<br> "receivedtime"    varchar(100) NULL ,
<br> "operator"        varchar(100) NULL ,
<br> "status"     varchar(20)  NULL ,
<br> "msgtype"         varchar(160) NULL ,
<br> "reference"       varchar(100) NULL ,
<br> "errormsg"      varchar(250) NULL ,
<br>   PRIMARY KEY ("id"),
<br>)
<br>go
<br>commit work
<br>go
<br>
<br>CREATE INDEX "ozekimessageinindex" ON "ozekimessagein"
<br>(
<br> "id" ASC
<br>)
<br>go
<br>commit work
<br>go
<br>
<br>CREATE INDEX "ozekimessageoutindex" ON "ozekimessageout"
<br>(
<br> "id" ASC
<br>)
<br>go
<br>commit work
<br>go
```

**Figure 16** - CREATE TABLE statements to copy



```
)
go
commit work
go

CREATE TABLE "ozekimessageout"
(
 "id"    integer NOT NULL DEFAULT autoincrement ,
 "sender"      varchar(160) NULL ,
 "receiver"     varchar(160) NULL ,
 "msg"    varchar(160) NULL ,
 "senttime"     varchar(100) NULL ,
 "receivedtime"   varchar(100) NULL ,
 "operator"     varchar(100) NULL ,
 "status"    varcha
 "msgtype"      va
 "reference"    va
 "errormsg"     va
  PRIMARY KEY ("i

)
go
commit work
go

CREATE INDEX "ozekimessageinindex" ON "ozekimessagein"
(
 "id" ASC
)
go
commit work
go

CREATE        INDEX        "ozekimessageoutindex"        ON
"ozekimessageout"
(
 "id" ASC
)
go
commit work
go
```

**Figure 17** - Copy CREATE TABLE statements from Figure 16



ozekisms (ozeki) on ozekisms - Interactive SQL
File Edit SQL Data Favorites Tools Window Help
ozekisms (ozeki) on ozekisms
SQL Statements

```
)
go
commit work
go

CREATE INDEX "ozekimessageoutindex" ON "ozekimessageout"
(
  "id" ASC
)
go
commit work
go
```

Results
Results History
Line 53 Column 3

**Figure 18** - Paste CREATE TABLE statements and run them on the SQL Anywhere database server

# Send SMS from SQL Anywhere (part 3/4)

## Configure Database User

**In this video series you can see how to connect Ozeki SMS Gateway to an SQL Anywhere database server for SMS messaging. This video shows how to install and configure a Database User on Ozeki SMS Gateway. It is necessary to provide the connection string.**

**Video content**

1. Start SQL Anywhere server
2. Install Database User
3. Provide connection string
4. Connect to Database

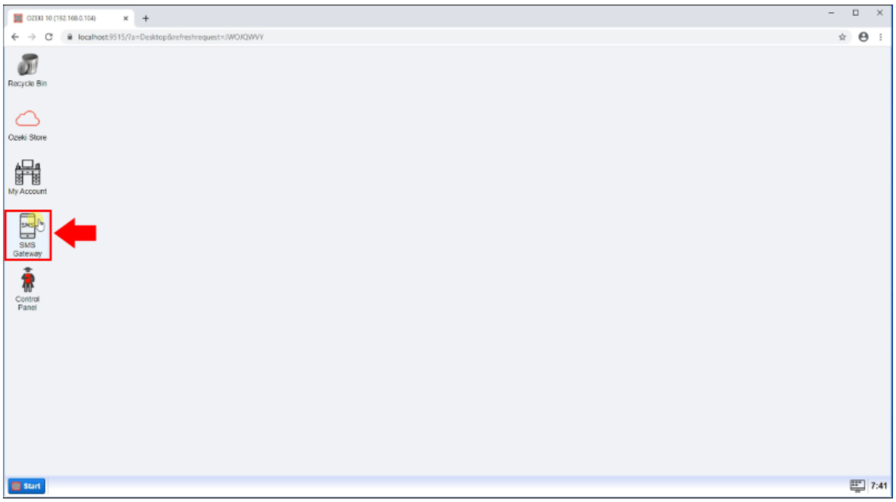


**Figure 1 -** Open the SMS Gateway applicaton on Ozeki 10's desktop screen

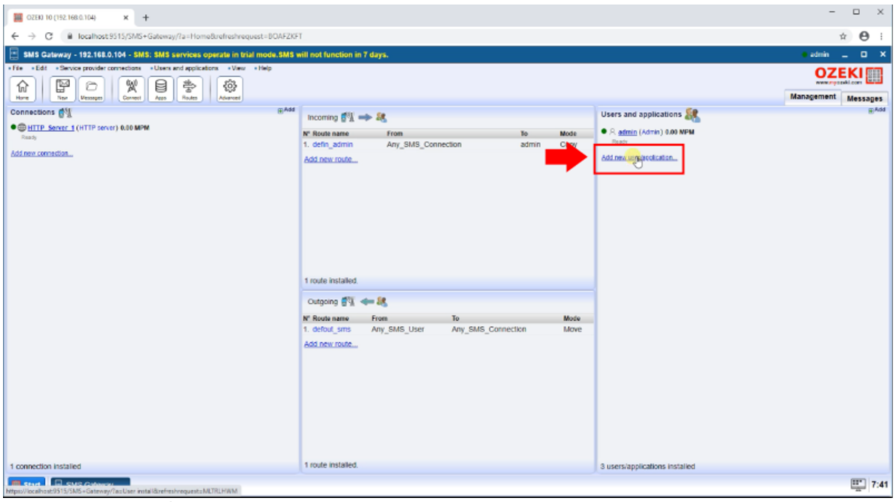
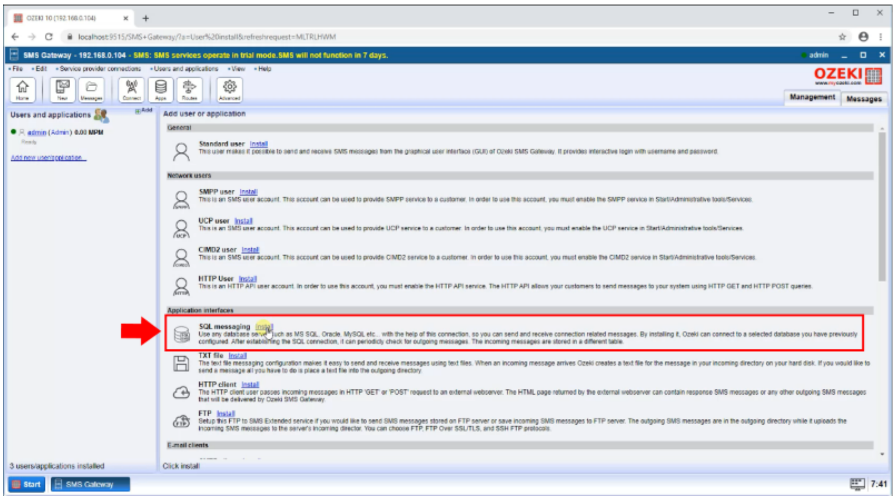


**Figure 2 -** Click 'Add new user or application' on the right side panel



Standard user Install
This user makes it possible to send and receive SMS messages from the graphical user interface (GUI) of Ozeki SMS Gateway. It provides interactive login with username and password.

SMPP user Install

UCP user Install
This is an SMS user account. This account can be used to provide UCP service to a customer. In order to use this account, you must enable the UCP service in Start/Administrative tools/Services.

CIMD2 user Install
This is an SMS user account. This account can be used to provide CIMD2 service to a customer. In order to use this account, you must enable the CIMD2 service in Start/Administrative tools/Services.

HTTP User Install
This is an HTTP API user account. In order to use this account, you must enable the HTTP API service. The HTTP API allows your customers to send messages to your system using HTTP GET and HTTP POST queries.

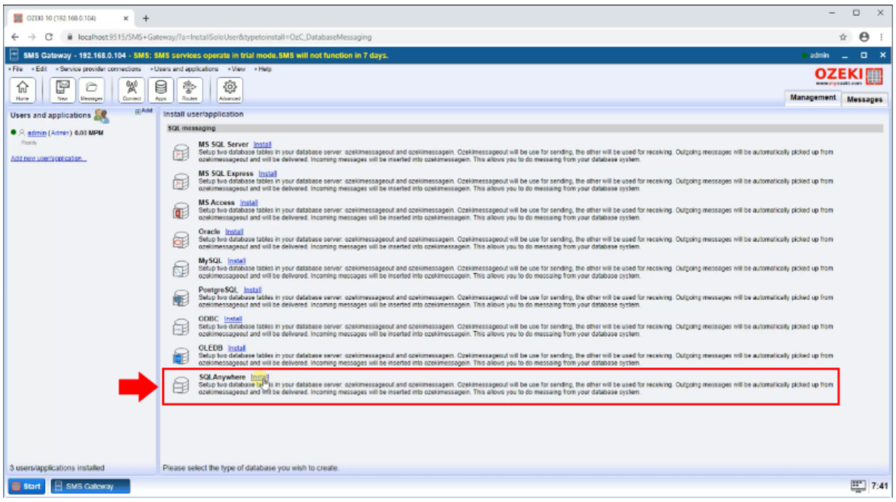
SQL messaging Install
Use any database server such as MS SQL, Oracle, MySQL etc... with the help of this connection, so you can send and receive connection related messages. By installing it, Ozeki can connect to a selected database you have previously configured. After establishing the SQL connection, it can periodicly check for outgoing messages. The incoming messages are stored in a different table.

TXT file Install
The text file messaging configuration makes it easy to send and receive messages using text files. When an incoming message arrives Ozeki creates a text file for the message in your incoming directory on your hard disk. If you would like to send a message all you have to do is place a text file into the outgoing directory.

HTTP client Install
The HTTP client user passes incoming messages in HTTP 'GET' or 'POST' request to an external webserver. The HTML page returned by the external webserver can contain response SMS messages or any other outgoing SMS messages that will be delivered by Ozeki SMS Gateway.

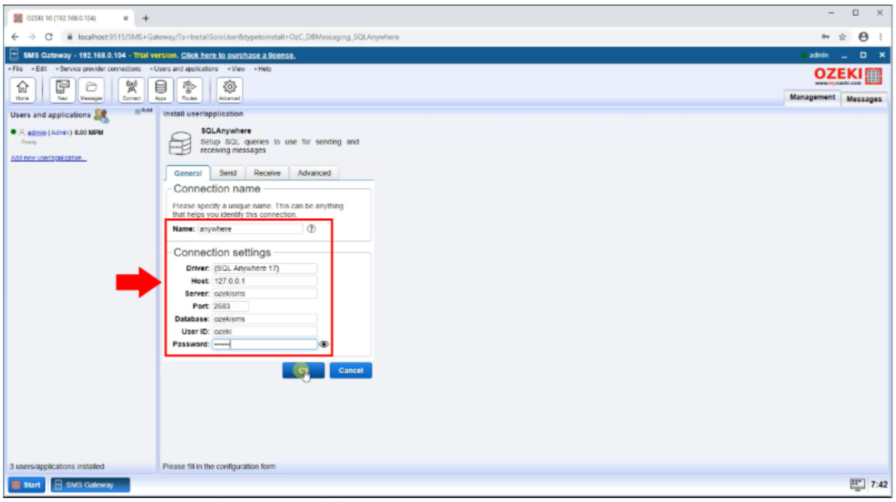
FTP Install
Setup this FTP to SMS Extended service if you would like to send SMS messages stored on FTP server or save incoming SMS messages to FTP server. The outgoing SMS messages are in the outgoing directory while it uploads the incoming SMS messages to the server's incoming director. You can choose FTP, FTP Over SSL/TLS, and SSH FTP protocols.

**Figure 3** - Install SQL Messageing User



SQLAnywhere Install
Setup two database tables in your database server: ozekimessageout and ozekimessagein. Ozekimessageout will be use for sending, the other will be used for receiving. Outgoing messages will be automatically picked up from ozekimessageout and will be delivered. Incoming messages will be inserted into ozekimessagein. This allows you to do messaing from your database system.

**Figure 4** - Install SQL Anywhere Connection



SQLAnywhere
Setup SQL queries to use for sending and receiving messages

Connection name

Please specify a unique name. This can be anything that helps you identify this connection.

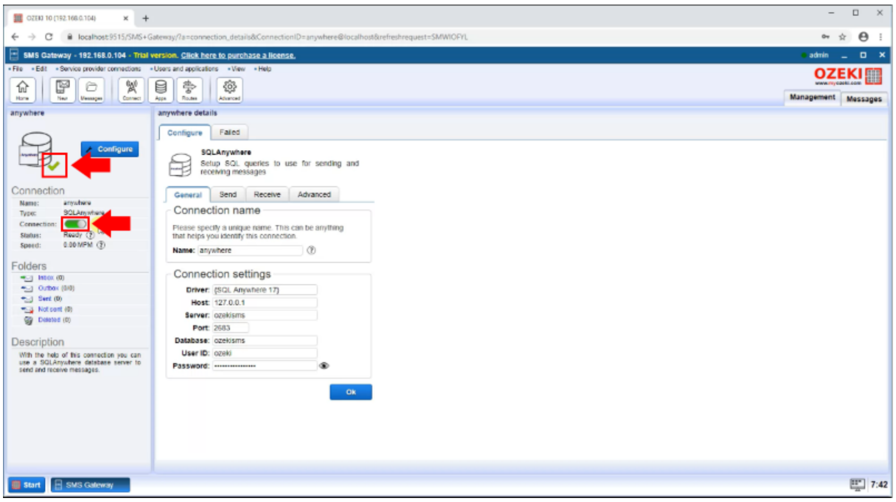
Name: anywhere

Connection settings

Driver: {SQL Anywhere 17}
Host: 127.0.0.1
Server: ozekisms
Port: 2683
Database: ozekisms
User ID: ozeki
Password: ••••••

Please fill in the configuration form

**Figure 5** - Define the SQL Anywhere database connection details



anywhere

Connection

Name: anywhere
Type: SQLAnywhere
Connection:
Status: Ready
Speed: 0.00 MPM

Description

With the help of this connection you can use a SQLAnywhere database server to send and receive messages.

SQLAnywhere
Setup SQL queries to use for sending and receiving messages

Connection name

Please specify a unique name. This can be anything that helps you identify this connection.

Name: anywhere

Connection settings

Driver: {SQL Anywhere 17}
Host: 127.0.0.1
Server: ozekisms
Port: 2683
Database: ozekisms
User ID: ozeki
Password: ••••••••••••••••

**Figure 6 -** Enable SQL Anywhere connection

# Send SMS from SQL Anywhere (part 4/4)

## Send Test Message

**In this video series you can see how to connect Ozeki SMS Gateway to an SQL Anywhere database server for SMS messaging. See how to insert a simple row into the proper table, so Ozeki SMS Gateway can automatically send SMS messages to mobile phones.**

**Video content**
1. INSERT message into database
2. Send Test message

Look at the upcoming screenshots to thoroughly examine the final stage, which is SMS sending. You can start the whole process by INSERT-ing the SMS into the database (**Figure 1**).

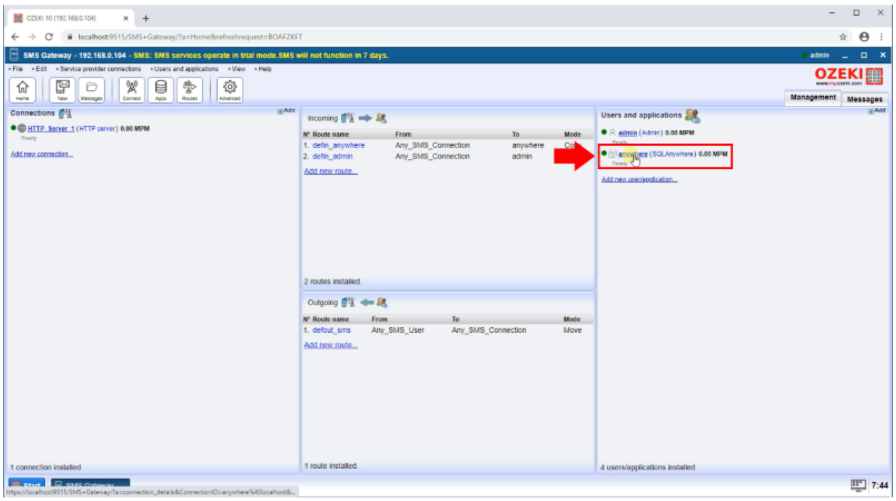
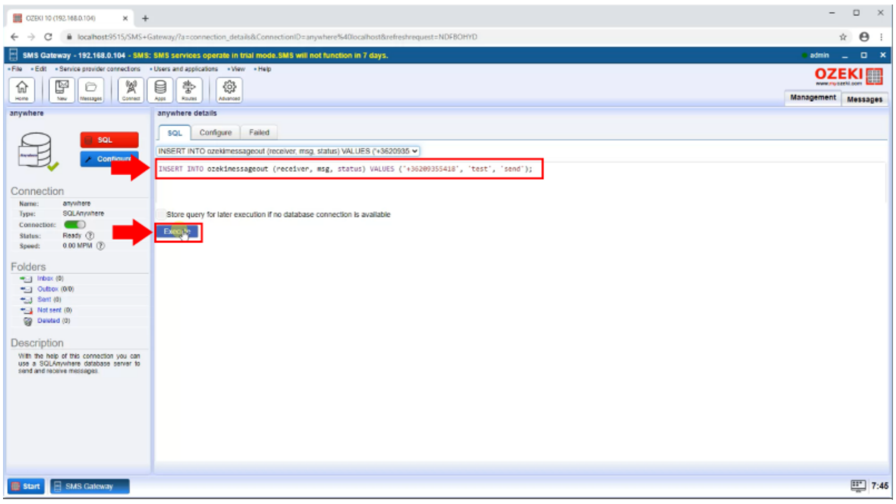

SMS Gateway - 192.168.0.104 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
OZEKI
Connections
HTTP_Server_1 (HTTP server) 0.00 MPM
Add new connection...
Incoming
Outgoing
Users and applications
admin (Admin) 0.00 MPM
anywhere (SQLAnywhere) 0.00 MPM
Add new user/application...
1 connection installed
2 routes installed
1 route installed
4 users/applications installed

**Figure 1** - Open Database user



```
INSERT INTO ozekimessageout (receiver, msg, status) VALUES ('+36209355418', 'test', 'send');
```

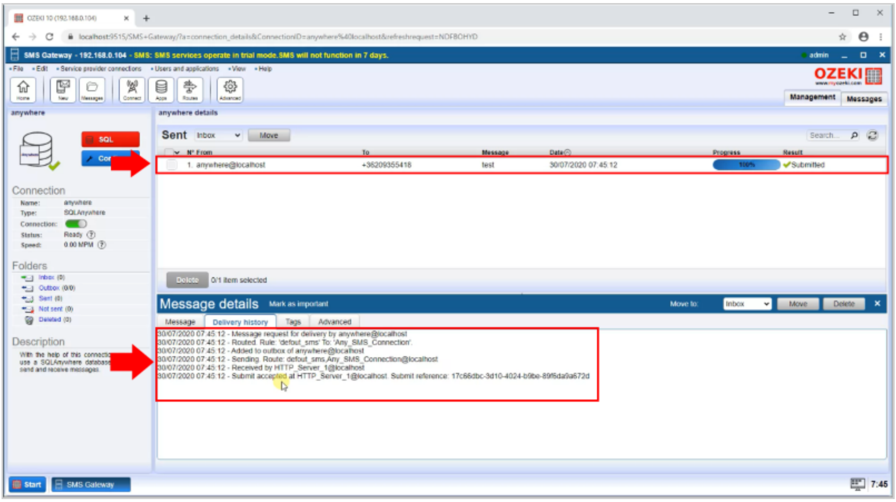
Store query for later execution if no database connection is available

**Figure 2** - Insert message to the database table

**INSERT message record** (example)**:**

```
USE ozeki
INSERT INTO ozekimessageout (receiver,msg,status) values ("+44111223465","Hello world","Send");
GO
```

**Figure 3** - Copy INSERT statement



Connection
Name: anywhere
Type: SQLAnywhere
Status: Ready
Speed: 0.00 MPM

Sent

| Nº | From | To | Message | Date | Progress | Result |
|---|---|---|---|---|---|---|
| 1. | anywhere@localhost | +36209355418 | test | 30/07/2020 07:45:12 | 100% | ✓Submitted |

0/1 item selected

```
30/07/2020 07:45:12 - Message request for delivery by anywhere@localhost
30/07/2020 07:45:12 - Routed. Rule: 'defout_sms' To: 'Any_SMS_Connection'.
30/07/2020 07:45:12 - Added to outbox of anywhere@localhost
30/07/2020 07:45:12 - Sending. Route: defout_sms,Any_SMS_Connection@localhost
30/07/2020 07:45:12 - Received by HTTP_Server_1@localhost
30/07/2020 07:45:12 - Submit accepted at HTTP_Server_1@localhost. Submit reference: 17c66dbc-3d10-4024-b9be-89f6da9a672d
```

**Figure 3 -** The Database User's sent folder shows that your message has been sent by Ozeki SMS Gateway

# SMS from/to Microsoft Access

***You can simply start to send and receive SMS messages through Ozeki SMS Gateway's Database User by using Microsoft Access database tables. All you have to do is insert or read data rows from the appropriate SQL table by configuring INSERT INTO and SELECT statements for the user.***

**This page shows how to install and prepare the Database User in the browser GUI of Ozeki SMS Gateway.** Additionally on the next page you can simply see how to use Microsoft Access with a few easy screenshots.

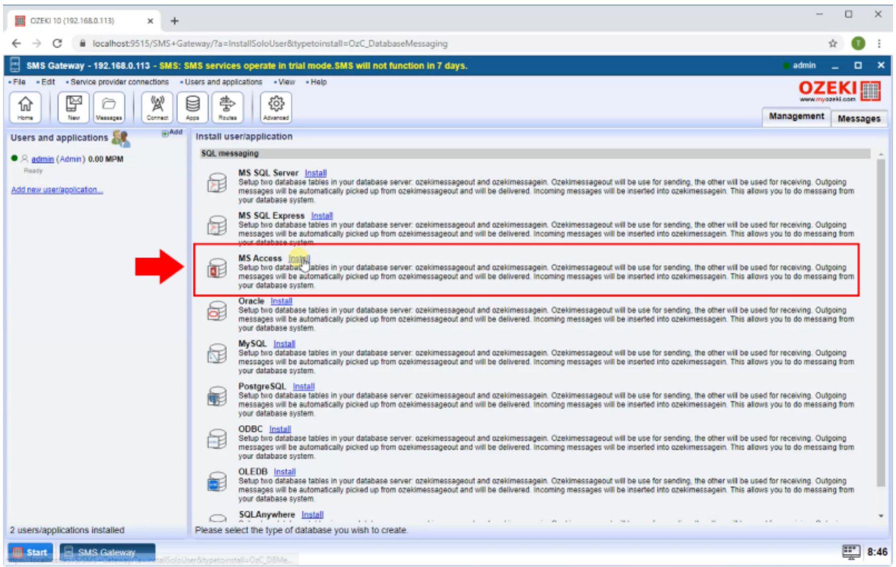
## Basics of Microsoft Access connection

The Database User can connect to your Access database tables. Keep in mind to create two tables. One for receiving and the other one for sending messages. The table structure should match the one in the MS SQL configuration, which basicly consists of an ozekimessagein and ozekimessageout table. Finally you should write SQL statements to work with these tables. At the bottom of the page you can find some UPDATE statements to use on the sent message records.

## Add Database User

*Make sure that your Ozeki SMS Gateway is connected to your GSM modem or IP SMS service provider*. For example SMPP, CIMD2 or UCP/EMI are accepted service provider connections. After you are able to manually send and receive SMS messages with SMS Gateway, you can install (**Figure 1**) and configure a Database User as you can see below.

You should start by **selecting 'Users and applications/Add new user or application' from the top menu** then **search the SQL messaging User and and then select the MS Access user. Click the blue 'install' button** next to it as you can **see on Figure 1**.



Install user/application

SQL messaging

MS SQL Server Install
Setup two database tables in your database server: ozekimessageout and ozekimessagein. Ozekimessageout will be use for sending, the other will be used for receiving. Outgoing messages will be automatically picked up from ozekimessageout and will be delivered. Incoming messages will be inserted into ozekimessagein. This allows you to do messaing from your database system.

MS SQL Express Install
Setup two database tables in your database server: ozekimessageout and ozekimessagein. Ozekimessageout will be use for sending, the other will be used for receiving. Outgoing messages will be automatically picked up from ozekimessageout and will be delivered. Incoming messages will be inserted into ozekimessagein. This allows you to do messaing from your database system.

MS Access Install
Setup two database tables in your database server: ozekimessageout and ozekimessagein. Ozekimessageout will be use for sending, the other will be used for receiving. Outgoing messages will be automatically picked up from ozekimessageout and will be delivered. Incoming messages will be inserted into ozekimessagein. This allows you to do messaing from your database system.

Oracle Install
Setup two database tables in your database server: ozekimessageout and ozekimessagein. Ozekimessageout will be use for sending, the other will be used for receiving. Outgoing messages will be automatically picked up from ozekimessageout and will be delivered. Incoming messages will be inserted into ozekimessagein. This allows you to do messaing from your database system.

MySQL Install
Setup two database tables in your database server: ozekimessageout and ozekimessagein. Ozekimessageout will be use for sending, the other will be used for receiving. Outgoing messages will be automatically picked up from ozekimessageout and will be delivered. Incoming messages will be inserted into ozekimessagein. This allows you to do messaing from your database system.

PostgreSQL Install
Setup two database tables in your database server: ozekimessageout and ozekimessagein. Ozekimessageout will be use for sending, the other will be used for receiving. Outgoing messages will be automatically picked up from ozekimessageout and will be delivered. Incoming messages will be inserted into ozekimessagein. This allows you to do messaing from your database system.

ODBC Install
Setup two database tables in your database server: ozekimessageout and ozekimessagein. Ozekimessageout will be use for sending, the other will be used for receiving. Outgoing messages will be automatically picked up from ozekimessageout and will be delivered. Incoming messages will be inserted into ozekimessagein. This allows you to do messaing from your database system.

OLEDB Install
Setup two database tables in your database server: ozekimessageout and ozekimessagein. Ozekimessageout will be use for sending, the other will be used for receiving. Outgoing messages will be automatically picked up from ozekimessageout and will be delivered. Incoming messages will be inserted into ozekimessagein. This allows you to do messaing from your database system.

SQLAnywhere Install

Please select the type of database you wish to create.

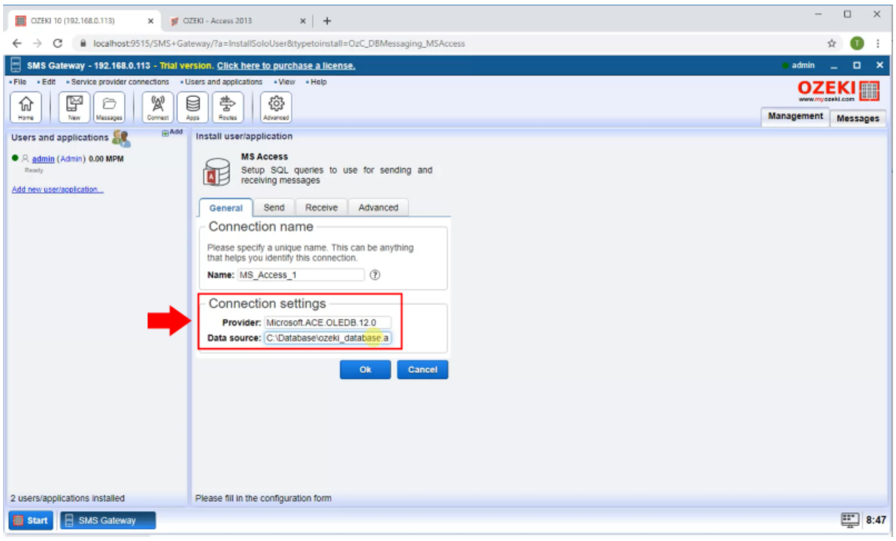
OZEKI 10 (192.168.0.113)
localhost:9515/SMS+Gateway/?a=InstallSoloUser&typetoinstall=OzC_DatabaseMessaging
SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
OZEKI
www.myozeki.com
Management
Messages
Users and applications
Add
admin (Admin) 0.00 MPM
Ready
Add new user/application...
2 users/applications installed
Start
SMS Gateway
8:46

**Figure 1** - Start installing your SMS Gateway's Database User

## Configure Database User

Database User is highly advised to connect to the most common databases. Here you can find the list of supported databases. After using a **connection string** to connect to your database, you can set which **SQL statements** to use in Ozeki SMS Gateway.

Connect to your Microsoft Access database with the help of an ODBC connection string, which points to your database through your ODBC driver. Please **type the path of the database file** as seen on **Figure 2**.



Install user/application

**MS Access**

Setup SQL queries to use for sending and receiving messages

Connection name

Please specify a unique name. This can be anything that helps you identify this connection.

Name: MS_Access_1

Connection settings

Provider: Microsoft.ACE.OLEDB.12.0

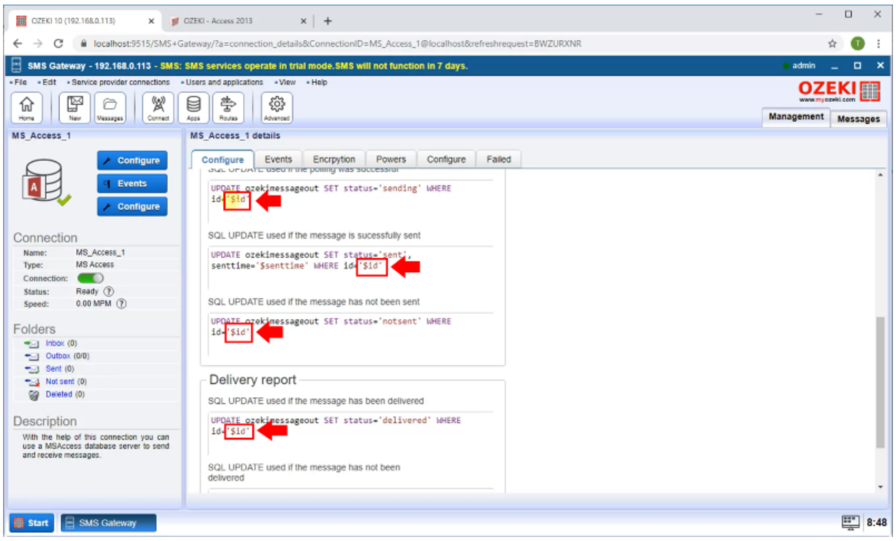
Data source: C:\Database\ozeki_database.a

Please fill in the configuration form

**Figure 2 -** Configure the database connection settings

**Set SQL templates**

After setting the connection string, it is time to **set the SQL templates**. These statements are required to update the outgoing message table, which is most likely called 'ozekimessageout'. ***The default statements** in Ozeki SMS Gateway **will not work with Microsoft Access databases**.* This is why you are highly suggested to **check the following figures** to make some minor changes like moving the apostrophe (' ') surrounding the $id keyword.



MS_Access_1 details

```
UPDATE ozekimessageout SET status='sending' WHERE
id='$id'
```

SQL UPDATE used if the message is sucessfully sent

```
UPDATE ozekimessageout SET status='sent',
senttime='$senttime' WHERE id='$id'
```

SQL UPDATE used if the message has not been sent

```
UPDATE ozekimessageout SET status='notsent' WHERE
id='$id'
```

Delivery report

SQL UPDATE used if the message has been delivered

```
UPDATE ozekimessageout SET status='delivered' WHERE
id='$id'
```

SQL UPDATE used if the message has not been delivered

**Figure 3 -** Sending template UPDATE-s your message in ozekimessageout

Congratulation! You have properly configured Ozeki SMS Gateway to send SMS messages from your Microsoft Access database. You are recommended to **test it by INSERT-ing your first SMS message to your outgoing message table**. The SMS Gateway's Database User is capable to read these message record and update them as described above from Figure 3. Please jump to the Microsoft Access 2013 tutorial for details.

# Microsoft Access 2013

**You will see detailed information on this page to show how to send and receive SMS messages with Microsoft Access 2013 database tables. Please follow these steps and screenshots to configure the connection to perfectly read lines from the outgoing message table and paste lines to the incoming message table. The default table names are 'ozekimessagein' and 'ozekimessageout'.**

On the previous page you might have read how to connect to the Access 2013 database and set the SQL statements, but on this page you can see it from the aspect of your Microsoft Access database. At the bottom of this page you can see how to send and receive messages with simple examples.

This example contains a database which you can download from here: ozeki_database.accdb

## How to connect SMS Gateway to this Access database example

After you have downloaded 'ozeki_database.accdb', please place it to a selected folder. In this tutorial it have been saved to the desktop (**Figure 1**) Although you can save it to any other location.

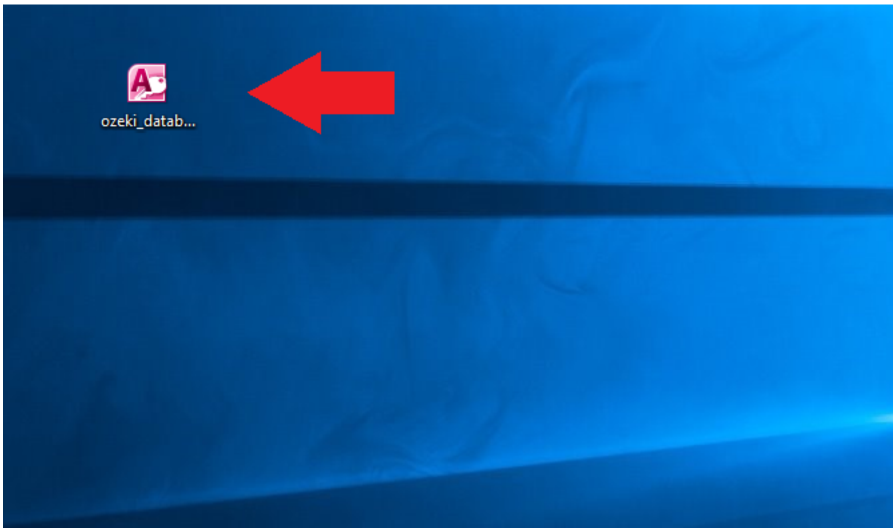
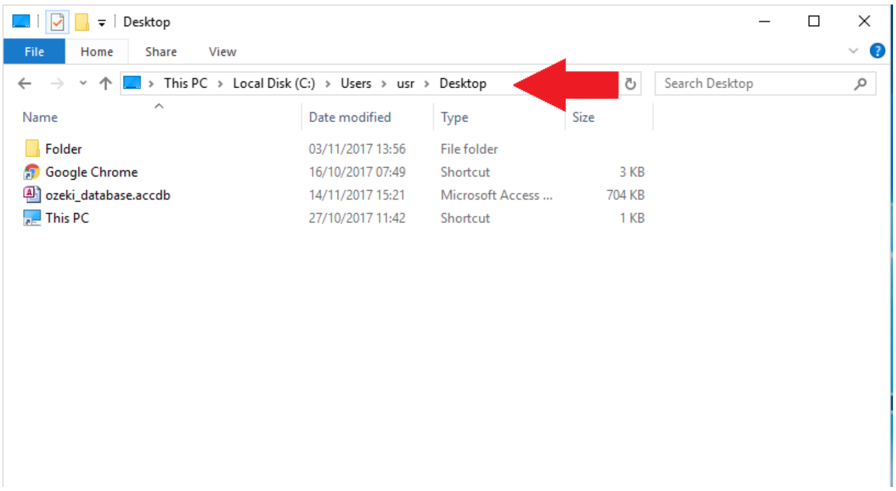


**Figure 1 -** ozeki_database.accdb placed on the laptop

Please memorize the file's location. Later on you will have to use this path in the connection string. You can check the path from the 'File Explorer' (**Figure 2**). The file is an Access 2013 database file.

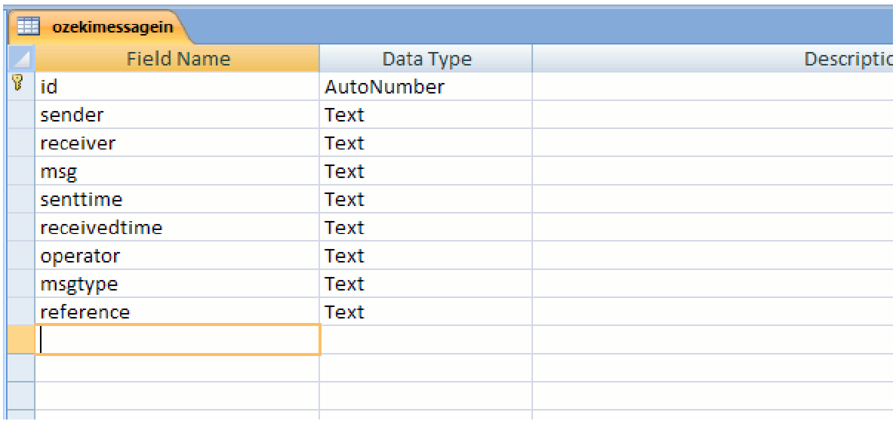


This PC › Local Disk (C:) › Users › usr › Desktop

| Name | Date modified | Type | Size |
|---|---|---|---|
| Folder | 03/11/2017 13:56 | File folder | |
| Google Chrome | 16/10/2017 07:49 | Shortcut | 3 KB |
| ozeki_database.accdb | 14/11/2017 15:21 | Microsoft Access ... | 704 KB |
| This PC | 27/10/2017 11:42 | Shortcut | 1 KB |

**Figure 2 -** Save the path to your database. *This will be useful for creating the connection string.*

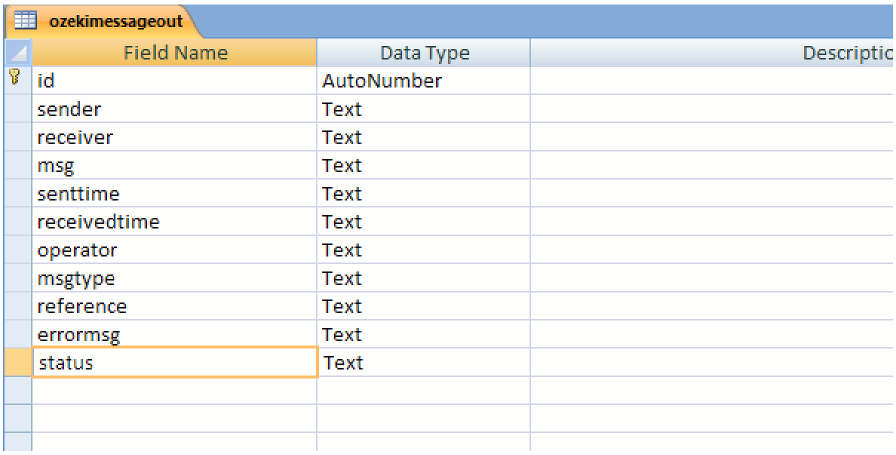
Just before you start configuring your database connection, please create these two database tables: 'ozekimessagein' and 'ozekimessageout' (**Figure 3** and **Figure 4**).



ozekimessagein

| Field Name | Data Type | Descriptic |
|---|---|---|
| id | AutoNumber | |
| sender | Text | |
| receiver | Text | |
| msg | Text | |
| senttime | Text | |
| receivedtime | Text | |
| operator | Text | |
| msgtype | Text | |
| reference | Text | |

**Figure 3** - 'ozekimessagein' table in Microsoft Access database



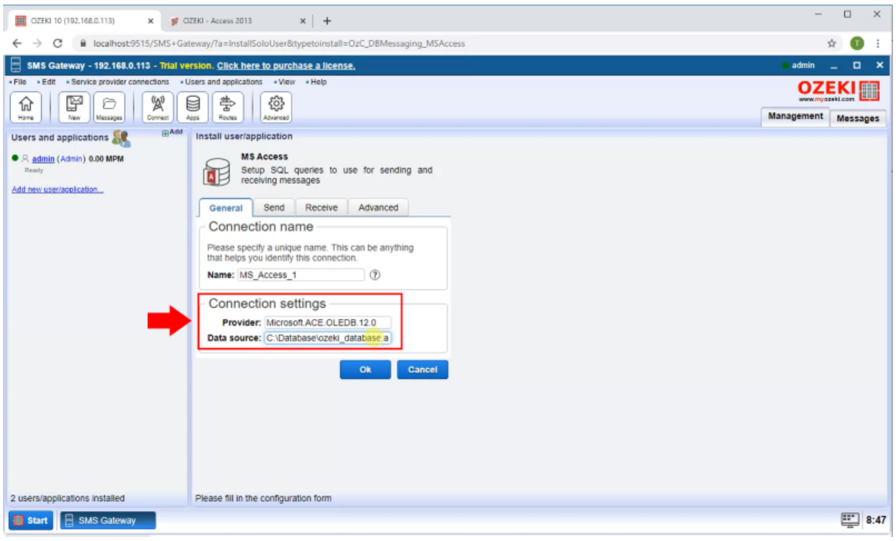
ozekimessageout

| Field Name | Data Type | Descriptic |
|---|---|---|
| id | AutoNumber | |
| sender | Text | |
| receiver | Text | |
| msg | Text | |
| senttime | Text | |
| receivedtime | Text | |
| operator | Text | |
| msgtype | Text | |
| reference | Text | |
| errormsg | Text | |
| status | Text | |

**Figure 4** - 'ozekimessageout' table in Microsoft Access database

It is assumed that you have already created the Database User. The Database User can be installed on your Ozeki SMS Gateway.

Keep in mind to change the 'Data Source=' parameter to the location of 'ozeki_database.accdb' (**Figure 6**).



Install user/application

MS Access

Setup SQL queries to use for sending and receiving messages

Connection name

Please specify a unique name. This can be anything that helps you identify this connection.

Name: MS_Access_1

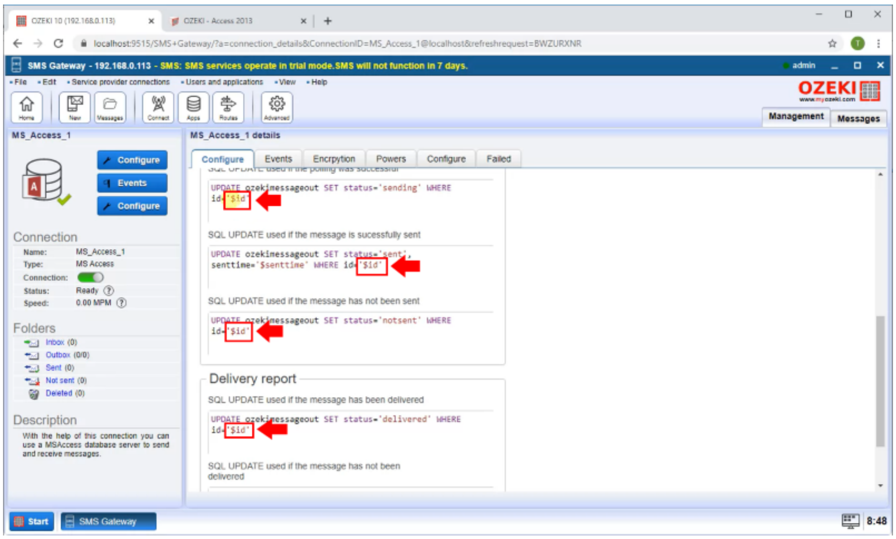
Connection settings

Provider: Microsoft.ACE.OLEDB.12.0

Data source: C:\Database\ozeki_database.a

**Figure 6** - Paste your database connection string at the Database User's configuration panel

Switch to the 'Sending' and tabpage and remove the apostrophes (' ') around $id (**Figure 7**).


SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.

MS_Access_1 details

SQL UPDATE used if the polling was successful

```
UPDATE ozekimessageout SET status='sending' WHERE
id='$id'
```

SQL UPDATE used if the message is sucessfully sent

```
UPDATE ozekimessageout SET status='sent',
senttime='$senttime' WHERE id='$id'
```

SQL UPDATE used if the message has not been sent

```
UPDATE ozekimessageout SET status='notsent' WHERE
id='$id'
```

Delivery report

SQL UPDATE used if the message has been delivered

```
UPDATE ozekimessageout SET status='delivered' WHERE
id='$id'
```

SQL UPDATE used if the message has not been delivered

**Figure 7 -** Remove apostrophes from all of the UPDATE statements

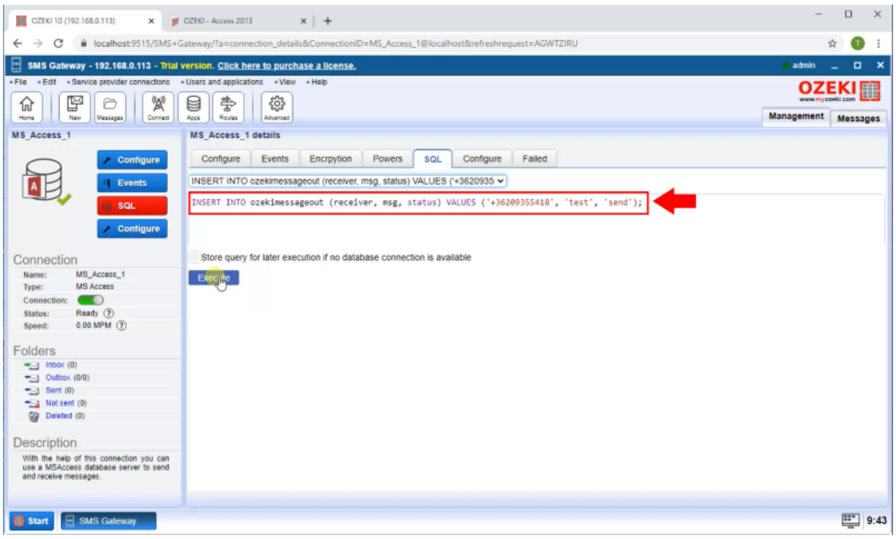
Now it is ready to use the configuration of the connection.

# Testing the configuration

When testing, you should use both tables. The one which was created for incoming messages and the one which was created for outgoing messages. In these examples these tables are called 'ozekimessageout' and 'ozekimessagein'.

## Sending messages

To **test the configuration**, you should **INSERT a message record into the outgoing messages table** (**Figure 8**), which is called 'ozekimessageout' in the current example. These messages can be SELECT-ed (polled) by configuring the Database User here:
'SQL for sending' tabpage and 'Polling' sub-tabpage. Please search these options in the SMS Gateway's browser GUI.


SMS Gateway - 192.168.0.113 - Trial version. Click here to purchase a license.

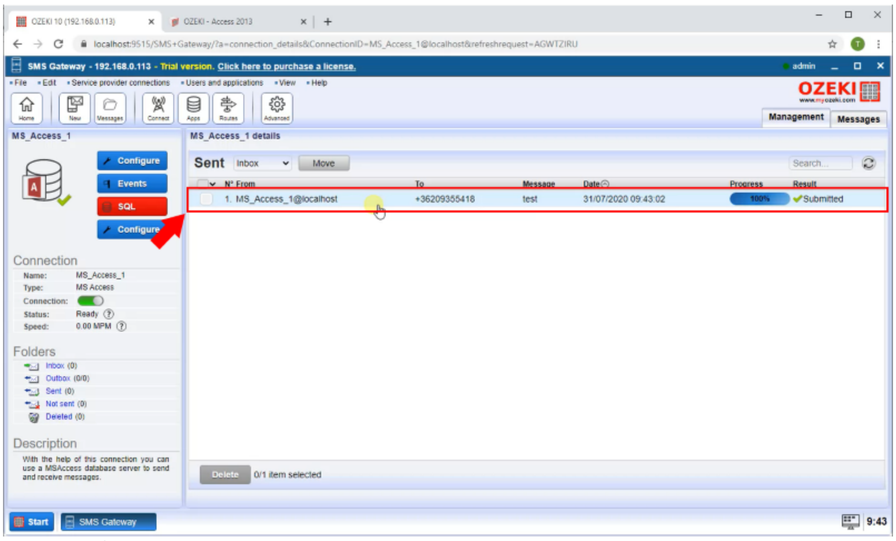
MS_Access_1 details

```
INSERT INTO ozekimessageout (receiver, msg, status) VALUES ('+36209355418', 'test', 'send');
```

□ Store query for later execution if no database connection is available

**Figure 8 -** INSERT a new row into the outgoing message table

The 'status=' parameter of the message record will change from 'Sending' to 'Sent' to 'Delivered'. The sent message will be visible in the Database User's **Sent folder** (**Figure 9**).



MS_Access_1 details

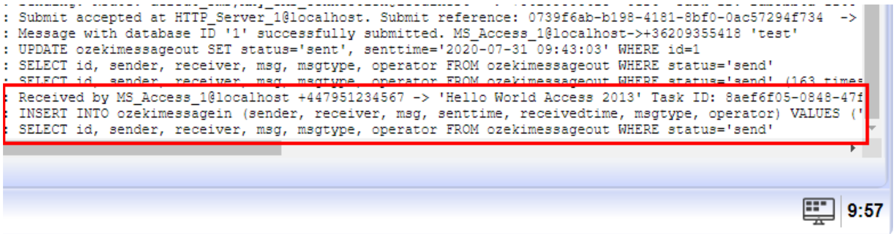
Sent

| N° From | To | Message | Date | Progress | Result |
|---|---|---|---|---|---|
| 1. MS_Access_1@localhost | +36209355418 | test | 31/07/2020 09:43:02 | 100% | ✓Submitted |

**Figure 9 -** The test message have been sent

## Receiving messages

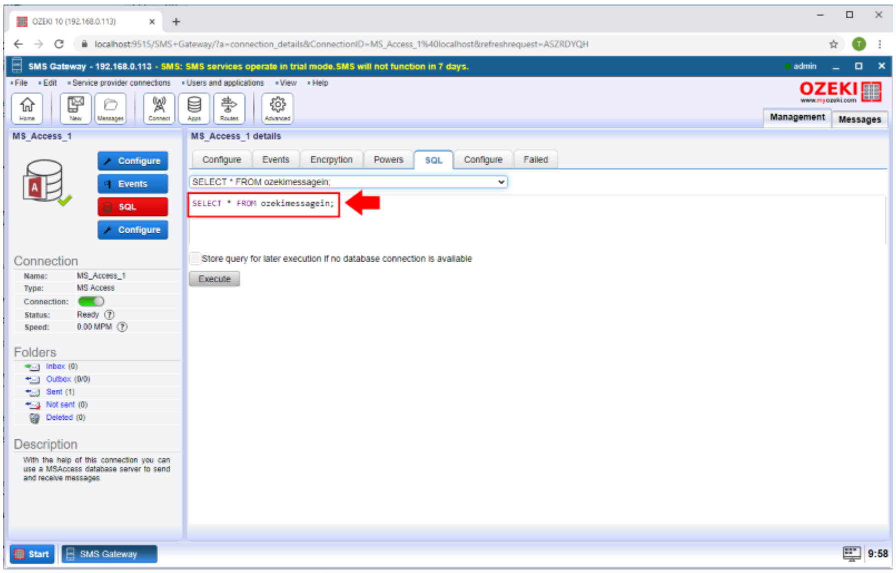
If an **incoming message log event** is displayed in the Database User's event log (**Figure 10**), the message entry will be INSERT -ed in the incoming message database table. The INSERT statement can be configured in the 'SQL for receiving' tabpage of this Database User's configuration panel. This is the statement which INSERT-s incoming messages.



```
: Submit accepted at HTTP_Server_1@localhost. Submit reference: 0739f6ab-b198-4181-8bf0-0ac57294f734  ->
: Message with database ID '1' successfully submitted. MS_Access_1@localhost->+36209355418 'test'
: UPDATE ozekimessageout SET status='sent', senttime='2020-07-31 09:43:03' WHERE id=1
: SELECT id, sender, receiver, msg, msgtype, operator FROM ozekimessageout WHERE status='send'
: SELECT id, sender, receiver, msg, msgtype, operator FROM ozekimessageout WHERE status='send' (163 times
: Received by MS_Access_1@localhost +447951234567 -> 'Hello World Access 2013' Task ID: 8aef6f05-0848-47f
: INSERT INTO ozekimessagein (sender, receiver, msg, senttime, receivedtime, msgtype, operator) VALUES ('
: SELECT id, sender, receiver, msg, msgtype, operator FROM ozekimessageout WHERE status='send'
```

**Figure 10 -** Message is received by Ozeki SMS Gateway. *Your Database User will INSERT it into a table.*

In this example the incoming message table is called 'ozekimessagein'. This is where those SMS messages are INSERT-ed, which are received by the Database User. You should run a SELECT statement on it (**Figure 11**) to read all of the messages (**Figure 12**).



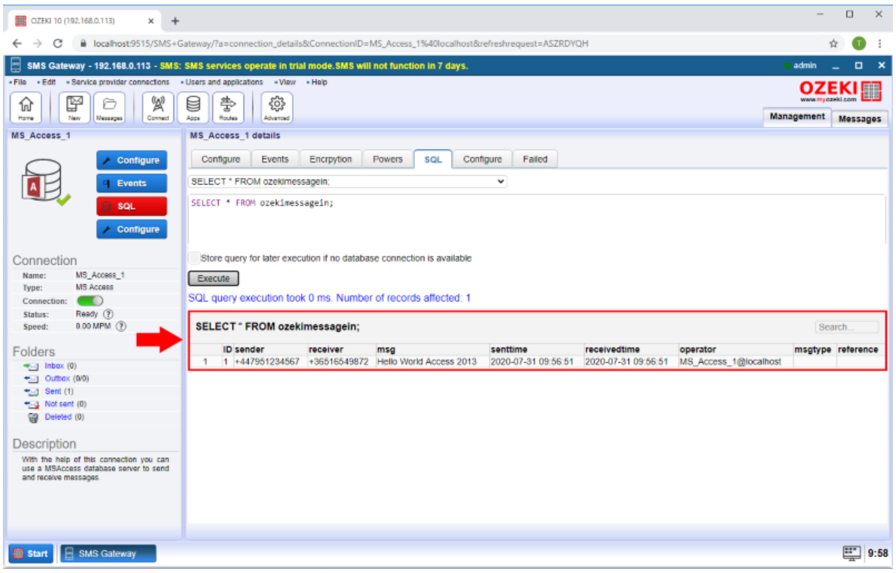
MS_Access_1 details

SQL

```
SELECT * FROM ozekimessagein;
```

**Figure 11 -** SELECT messages from the incoming message table

All of the incoming messages can be read after running a SELECT query on 'ozekimessagein' (**Figure 12**).



Connection

Name: MS_Access_1
Type: MS Access
Connection:
Status: Ready
Speed: 0.00 MPM

Folders

Inbox (0)
Outbox (0/0)
Sent (1)
Not sent (0)
Deleted (0)

Description

With the help of this connection you can use a MSAccess database server to send and receive messages.

SELECT * FROM ozekimessagein;

```
SELECT * FROM ozekimessagein;
```

□ Store query for later execution if no database connection is available

SQL query execution took 0 ms. Number of records affected: 1

SELECT * FROM ozekimessagein;

| | ID | sender | receiver | msg | senttime | receivedtime | operator | msgtype | reference |
|---|---|---|---|---|---|---|---|---|---|
| 1 | 1 | +447951234567 | +36516549872 | Hello World Access 2013 | 2020-07-31 09:56:51 | 2020-07-31 09:56:51 | MS_Access_1@localhost | | |

**Figure 12 -** The SMS messages received by Ozeki SMS Gateway's Database User

# HTTP SMS API

**The HTTP SMS API implemented by Ozeki 10 SMS Gateway provides a great, easy to use interface, that allows software developers to send and receive SMS messages. the followig pages give you information on how to use this technology, with several free to use / free to customize examples.**

## Send SMS

To send an SMS message using the HTTP API you need to issue an HTTP request (Figure 1). You can use the HTTP GET or the HTTP POST method, when you send this request to Ozeki 10 SMS Gateway. After the request is sent, you will receive a response.

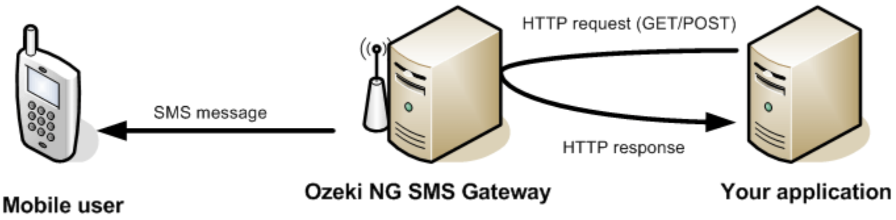


**Figure 1 -** Send an SMS messages through the built in webserver

To send an SMS messages you need to follow a few simple steps. First you need to create an HTTP API user, then you need to write an HTTP request, then you need to submit your request to the SMS Gateway.

Make sure Ozeki 10 SMS Gateway is installed
Create an HTTP API user
Write an HTTP request to send the SMS
Write an HTTP request to send many SMS messages

**HTTP SMS API examples**
How to send SMS from C#
How to send SMS from Java
How to send SMS from Python
How to send SMS from Ruby
How to send SMS from Perl
How to send SMS from PHP
How to send SMS from Chrome

## Receive SMS

When an SMS message arrives to the Ozeki SMS gateway, it is stored in a message folder, called inbox. If you want to check incoming SMS messages one of your options is to download the contents of the inbox folder using HTTP. The following URL gives more information about how to download the contents of a message folder:

How to download incoming SMS messages **How to send a reply SMS**

If you want to process incoming SMS messages as soon as they arrive to your system, you may want to setup an HTTP Client user in Ozeki SMS Gateway. This HTTP Client user can forward incoming messages to your web application(s) as HTTP GET or HTTP POST requests (**Figure 2**).

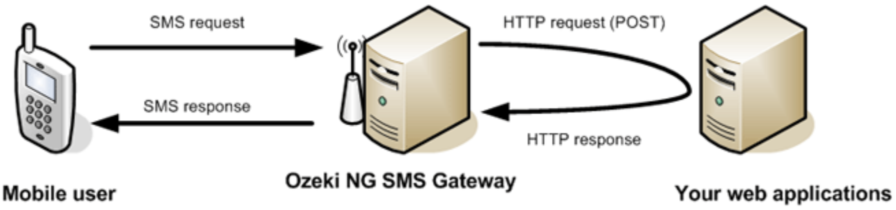


**Figure 2 -** HTTP Post on incoming SMS

The following webpage gives information on how to configure your SMS gateway to forward your SMS messages to an external web application:

HTTP Client user

Note that you can write this external web application in any programming language. We have examples for PHP SMS processing, ASP SMS processing and Java SMS processing.

Your web application can also return a response SMS message in the HTTP response. This response SMS messages will be forwarded back to the recipient handset. The following URL has information about the proper response format your web application can use:

HTTP response format for returning SMS response(s)

# How to create an HTTP API user

***In order to use the HTTP API, you need to create an HTTP API user. During this procedure you will be able to specify a username and a password. You can use this username and password to connect to the HTTP API service of Ozeki SMS Gateway to submit your messages, or to download the incoming messages. This guide gives you the steps to take to setup an HTTP API user.***

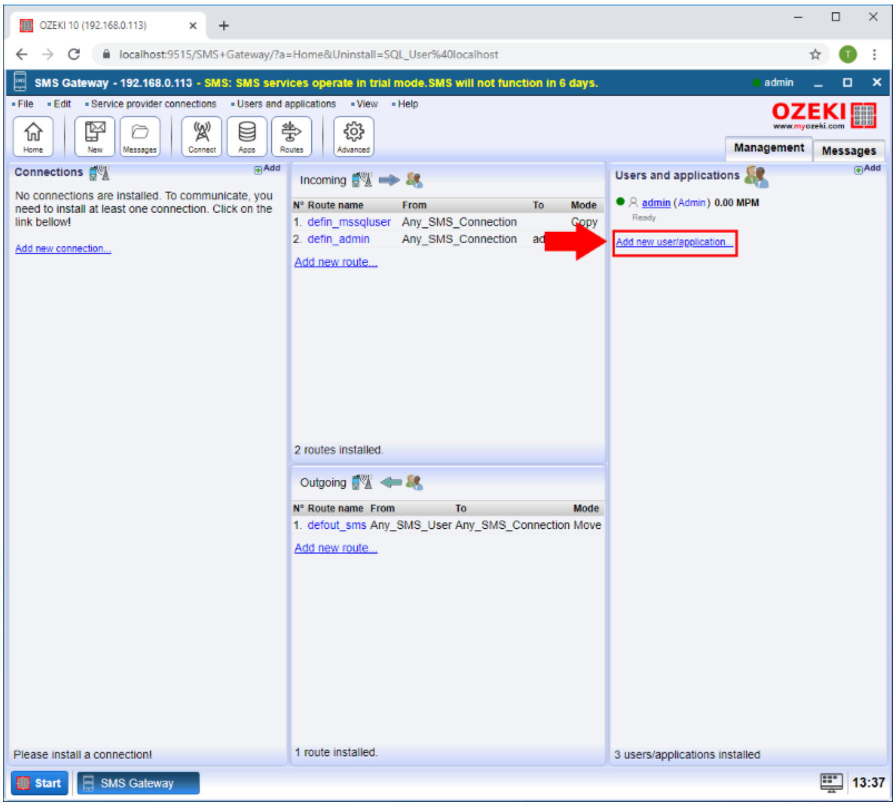
## Prerequisites

Ozeki 10 sms gateway needs to be installed

## Video tutorial

## Step 1 - Add new user/application

You can simply install the HTTP API User on the Management console by clicking Add new user/application... in the Users/Applications panel (Figure 1).



No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

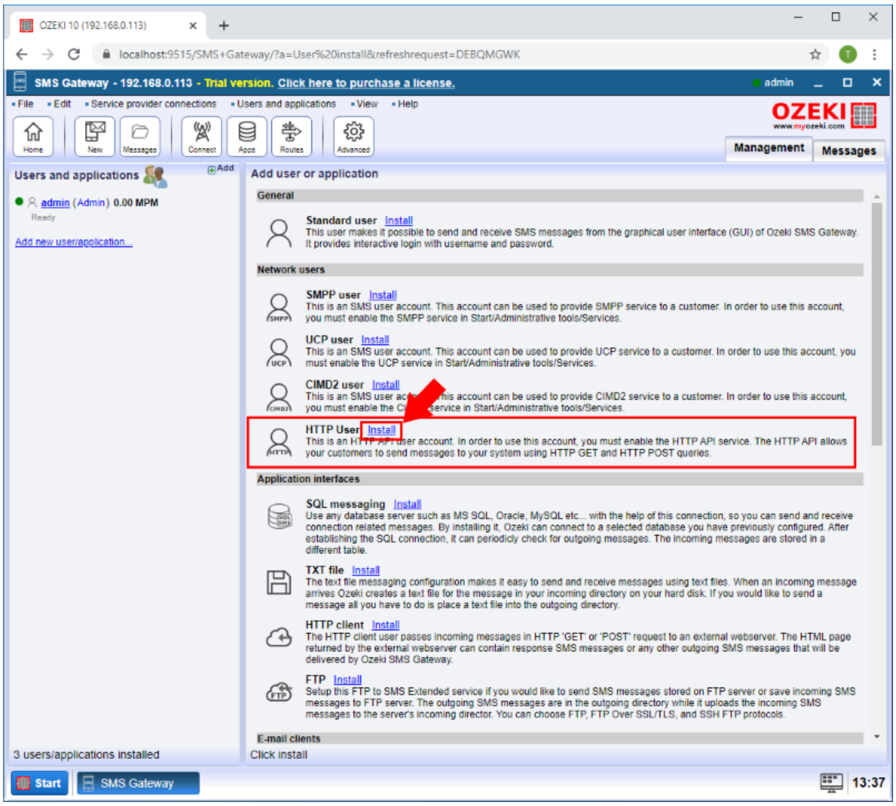
| Nº | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defin_mssqluser | Any_SMS_Connection | | Copy |
| 2. | defin_admin | Any_SMS_Connection | ad | |

| Nº | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move |

**Figure 1** - Add new user/application...

## Step 2 - Add HTTP API User

An interface will open consisting of two panels. The left side panel contains the already installed users and applications. The right side panel contains the users and applications you can install with a brief description next to them. Find the HTTP API User and click the blue 'install' button next to it (Figure 2).



Add user or application

General

Standard user Install
This user makes it possible to send and receive SMS messages from the graphical user interface (GUI) of Ozeki SMS Gateway. It provides interactive login with username and password.

Network users

SMPP user Install
This is an SMS user account. This account can be used to provide SMPP service to a customer. In order to use this account, you must enable the SMPP service in Start/Administrative tools/Services.

UCP user Install
This is an SMS user account. This account can be used to provide UCP service to a customer. In order to use this account, you must enable the UCP service in Start/Administrative tools/Services.

CIMD2 user Install
This is an SMS user account. This account can be used to provide CIMD2 service to a customer. In order to use this account, you must enable the CIMD2 service in Start/Administrative tools/Services.

HTTP User Install
This is an HTTP API user account. In order to use this account, you must enable the HTTP API service. The HTTP API allows your customers to send messages to your system using HTTP GET and HTTP POST queries.

Application interfaces

SQL messaging Install
Use any database server such as MS SQL, Oracle, MySQL etc... with the help of this connection, so you can send and receive connection related messages. By installing it, Ozeki can connect to a selected database you have previously configured. After establishing the SQL connection, it can periodicly check for outgoing messages. The incoming messages are stored in a different table.

TXT file Install
The text file messaging configuration makes it easy to send and receive messages using text files. When an incoming message arrives Ozeki creates a text file for the message in your incoming directory on your hard disk. If you would like to send a message all you have to do is place a text file into the outgoing directory.

HTTP client Install
The HTTP client user passes incoming messages in HTTP 'GET' or 'POST' request to an external webserver. The HTML page returned by the external webserver can contain response SMS messages or any other outgoing SMS messages that will be delivered by Ozeki SMS Gateway.

FTP Install
Setup this FTP to SMS Extended service if you would like to send SMS messages stored on FTP server or save incoming SMS messages to FTP server. The outgoing SMS messages are in the outgoing directory while it uploads the incoming SMS messages to the server's incoming director. You can choose FTP, FTP Over SSL/TLS, and SSH FTP protocols.

E-mail clients

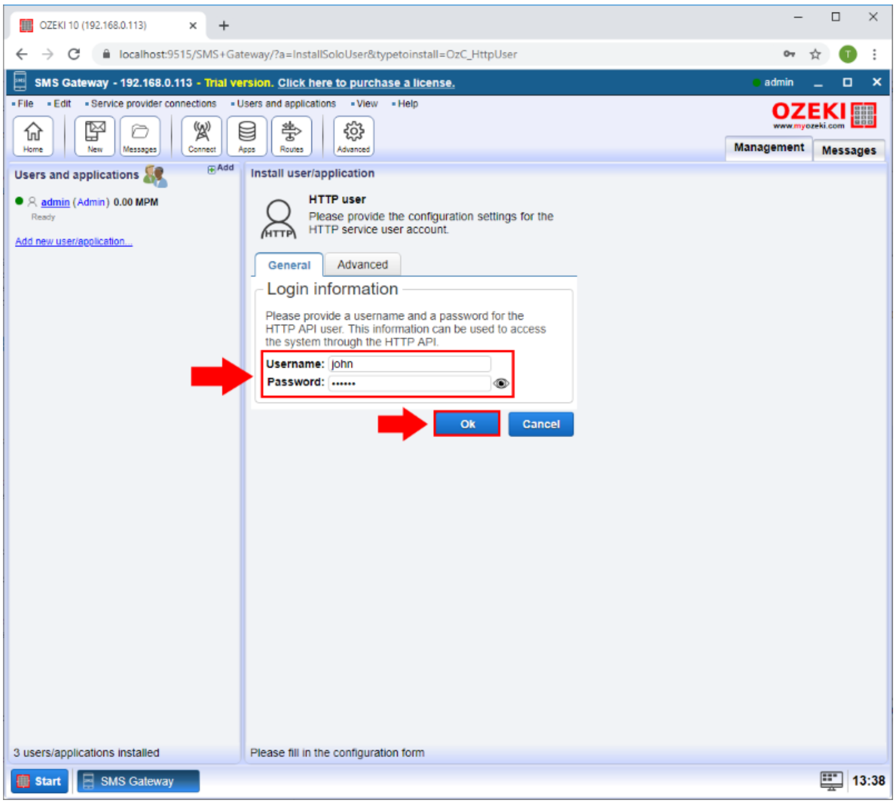
Click install

OZEKI 10 (192.168.0.113)
localhost:9515/SMS+Gateway/?a=User%20install&refreshrequest=DEBQMGWK
SMS Gateway - 192.168.0.113 - Trial version. Click here to purchase a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Users and applications
Add
admin (Admin) 0.00 MPM
Ready
Add new user/application...
3 users/applications installed
Start
SMS Gateway
13:37

**Figure 2** - Add HTTP API User

# Step 3 - Configure username and password

On the Name section provide the unique name for the HTTP User and provide the username and password for the authentication (Figure 3).



Install user/application

HTTP user
Please provide the configuration settings for the HTTP service user account.

Login information

Please provide a username and a password for the HTTP API user. This information can be used to access the system through the HTTP API.

Username: john
Password: ••••••

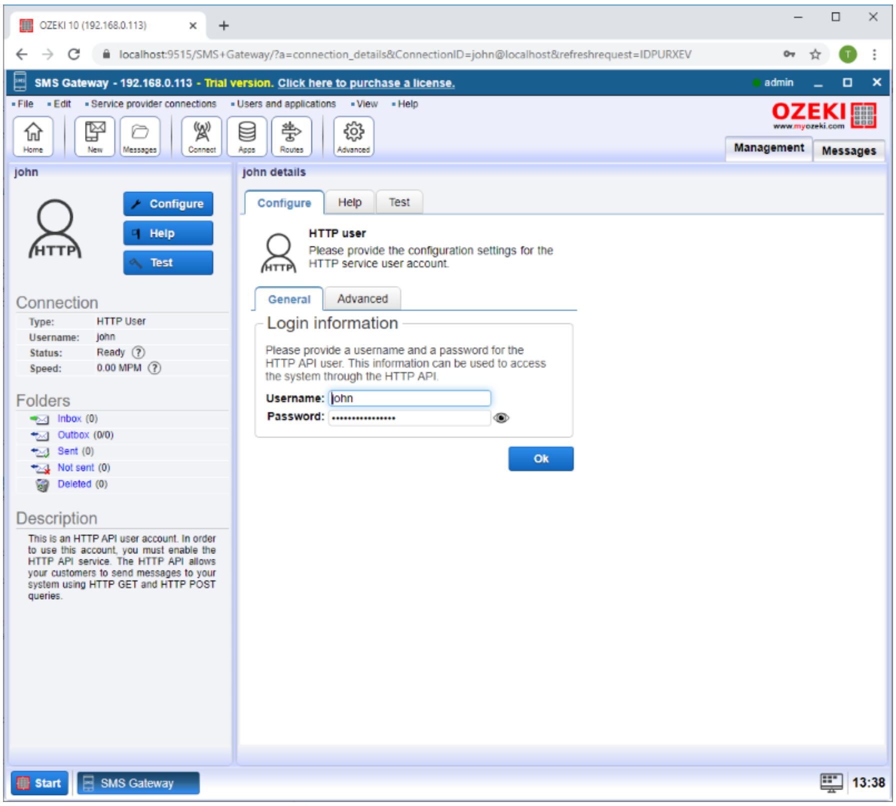
Please fill in the configuration form

OZEKI 10 (192.168.0.113)
localhost:9515/SMS+Gateway/?a=InstallSoloUser&typetoinstall=OzC_HttpUser
SMS Gateway - 192.168.0.113 - Trial version. Click here to purchase a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Users and applications
Add
admin (Admin) 0.00 MPM
Ready
Add new user/application...
General
Advanced
Ok
Cancel
3 users/applications installed
Start
SMS Gateway
13:38

**Figure 3** - Configure username and password

# Step 4 - HTTP API User details page

After you click on the OK the details page of the HTTP User will appeare (Figure 4).


OZEKI 10 (192.168.0.113)
localhost:9515/SMS+Gateway/?a=connection_details&ConnectionID=john@localhost&refreshrequest=IDPURXEV
SMS Gateway - 192.168.0.113 - Trial version. Click here to purchase a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Configure
Help
Test
General
Advanced
Ok
Start
SMS Gateway
13:38

john

Connection

| | |
|---|---|
| Type: | HTTP User |
| Username: | john |
| Status: | Ready |
| Speed: | 0.00 MPM |

Folders

Inbox (0)
Outbox (0/0)
Sent (0)
Not sent (0)
Deleted (0)

Description

This is an HTTP API user account. In order to use this account, you must enable the HTTP API service. The HTTP API allows your customers to send messages to your system using HTTP GET and HTTP POST queries.

john details

HTTP user

Please provide the configuration settings for the HTTP service user account.

Login information

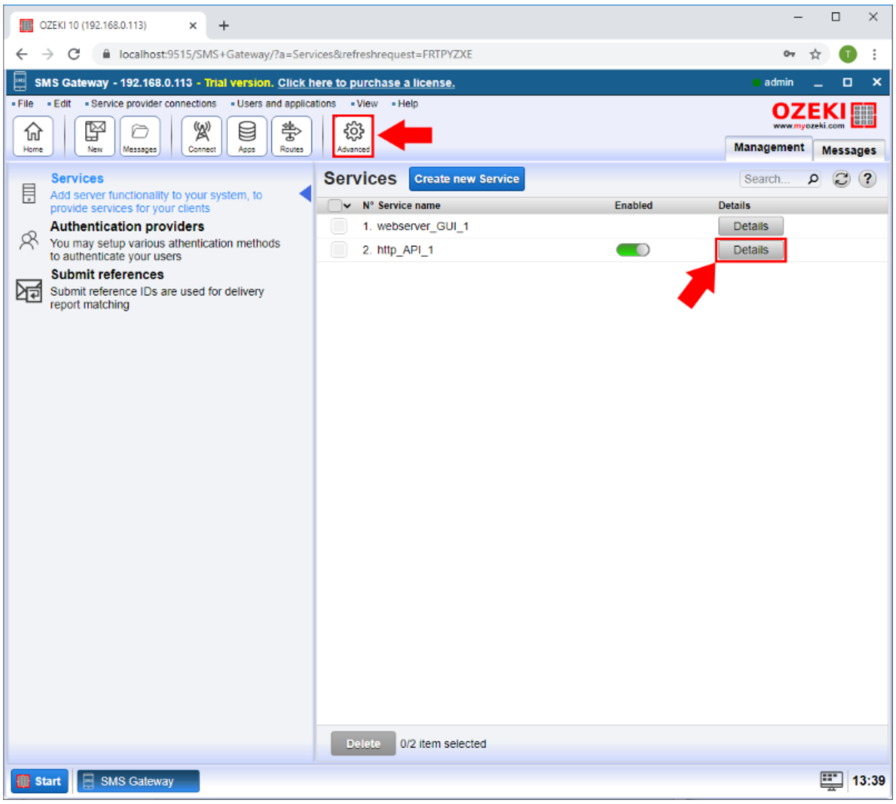
Please provide a username and a password for the HTTP API user. This information can be used to access the system through the HTTP API.

Username: john
Password: ••••••••••••••••

**Figure 4** - HTTP API User details page

# Step 5 - HTTP API Service

In the advanced menu you can find the HTTP API service for the user. Open the details page of the service as you can see on the Figure 5.



Services
Add server functionality to your system, to provide services for your clients

Authentication providers
You may setup various athentication methods to authenticate your users

Submit references
Submit reference IDs are used for delivery report matching

Services

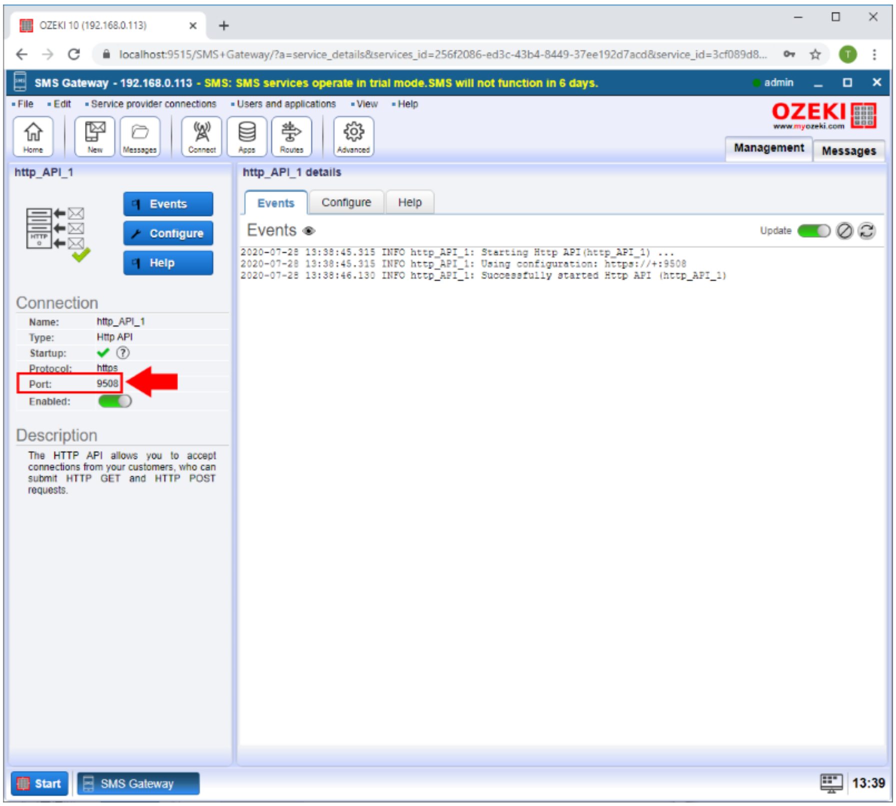
| N° Service name | Enabled | Details |
|---|---|---|
| 1. webserver_GUI_1 | | Details |
| 2. http_API_1 | ● | Details |

0/2 item selected

**Figure 5** - HTTP API Service

# Step 6 - HTTP Service events

The HTTP service Events tab shows the logs for the HTTP API and here you can see the port wher the servce is lisening for the requests (Figure 6).


OZEKI 10 (192.168.0.113)
localhost:9515/SMS+Gateway/?a=service_details&services_id=256f2086-ed3c-43b4-8449-37ee192d7acd&service_id=3cf089d8...
SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.SMS will not function in 6 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
OZEKI
www.myozeki.com
Management
Messages
http_API_1
Events
Configure
Help
http_API_1 details
Events
Configure
Help
Events
Update
Start
SMS Gateway
13:39

```
2020-07-28 13:38:45.315 INFO http_API_1: Starting Http API(http_API_1) ...
2020-07-28 13:38:45.315 INFO http_API_1: Using configuration: https://+:9508
2020-07-28 13:38:46.130 INFO http_API_1: Successfully started Http API (http_API_1)
```

Connection

| | |
|---|---|
| Name: | http_API_1 |
| Type: | Http API |
| Startup: | ✓ ? |
| Protocol: | https |
| Port: | 9508 |
| Enabled: | ● |

Description

The HTTP API allows you to accept connections from your customers, who can submit HTTP GET and HTTP POST requests.

**Figure 6 -** HTTP Service events

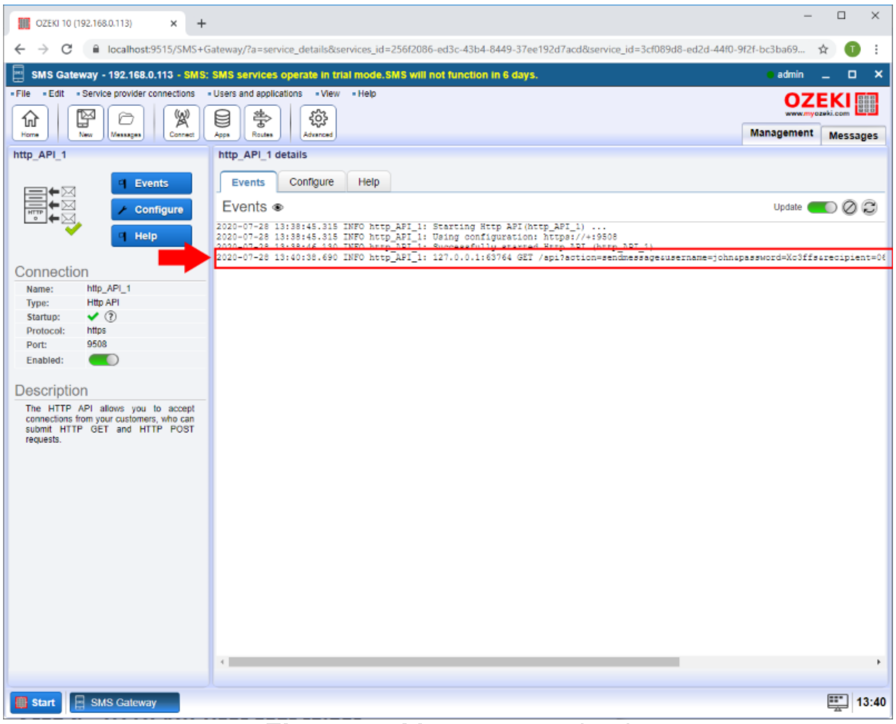
# Step 7 - Send an HTTP request form your application

The HTTP API service is capable of receiving HTTP requests from your application. This can be done by using the SMS API, that can initiate HTTP requests and send them to the HTTP API service that can process the HTTP requests and sends the SMS message. The link below contain example codes of how you can send SMS messages from applications written in different programming languages.

**HTTP SMS API examples**
How to send SMS from C#
How to send SMS from Java
How to send SMS from Python
How to send SMS from Ruby
How to send SMS from Perl
How to send SMS from PHP
How to send SMS from Chrome

# Step 8 - Message received

If a message is received by the HTTP API service you can see it in the logs (Figure 7).



http_API_1 details

Events

```
2020-07-28 13:38:45.315 INFO http_API_1: Starting Http API(http_API_1) ...
2020-07-28 13:38:45.315 INFO http_API_1: Using configuration: https://+:9508
2020-07-28 13:38:46.130 INFO http_API_1: Successfully started Http API (http_API_1)
2020-07-28 13:40:38.690 INFO http_API_1: 127.0.0.1:63764 GET /api?action=sendmessage&username=john&password=Xc3ffs&recipient=06
```

Connection

Name: http_API_1
Type: Http API
Protocol: https
Port: 9508

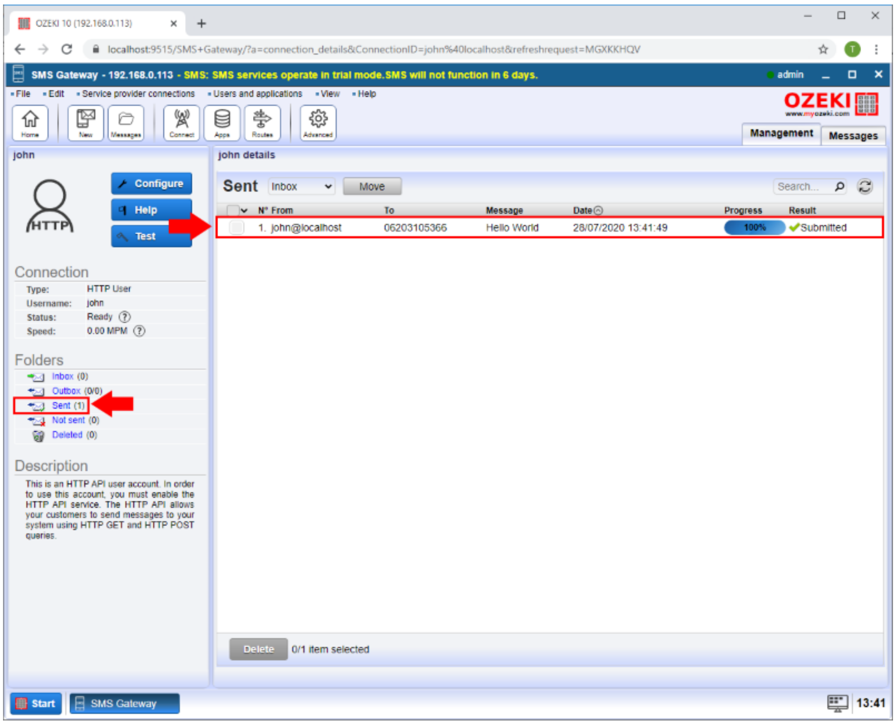
Description

The HTTP API allows you to accept connections from your customers, who can submit HTTP GET and HTTP POST requests.

**Figure 7** - Message received

# Step 9 - HTTP API User sent folder

Also you can find the message in the HTTP API user Sent folder (Figure 8).



john details

Sent

| N° | From | To | Message | Date | Progress | Result |
|---|---|---|---|---|---|---|
| 1. | john@localhost | 06203105366 | Hello World | 28/07/2020 13:41:49 | 100% | ✓Submitted |

Connection

Type: HTTP User
Username: john
Status: Ready
Speed: 0.00 MPM

Folders

Inbox (0)
Outbox (0/0)
Sent (1)
Not sent (0)
Deleted (0)

Description

This is an HTTP API user account. In order to use this account, you must enable the HTTP API service. The HTTP API allows your customers to send messages to your system using HTTP GET and HTTP POST queries.

0/1 item selected

**Figure 8** - HTTP API User sent folder

# HTTP API - Get Started

***The HTTP API allows you to send and receive SMS messages through your Ozeki SMS Gateway with the help of HTTP requests. Requests can be done using both the GET and POST HTTP methods and you can use http or https, depending on your configuration.***

For each HTTP request, you must include a set of mandatory parameters. These are: action, username and password. The action defines the command you wish to execute and the username and password identify you. To How to create an HTTP API user guide gives you information about how to create a username and password.

**Example HTTP request:**

```
https://127.0.0.1:9508/api?action=sendmessage&username=admin&password=abc123&
recipient=06203105366&messagedata=Hello+World
```

**Example HTTP response:**

```
StatusCode: 200, ReasonPhrase: 'OK'
```

You are advised to try this example with the HTTP API Tester which is a GUI interface for Ozeki SMS Gateway API testing. You can find the HTTP API Tester tutorial here.

## Mandatory parameters in every HTTP API request

| Parameter | Description | Possible values | Example | M/O* |
|---|---|---|---|---|
| action | The name of the action. | sendmessage,<br>receivemessage, | action=sendmessage | M |
| username | Specifies the username. The username and password parameters are used to authenticate the user. When you send an message it will be sent in the name of the authenticated user. The value must be URL encoced. | string value | username=admin | M |
| password | Specifies the password. The username and password parameters are used to authenticate the user. When you send an message it will be sent in the name of the authenticated user. The value must be URL encoced. | string value, minimum length: 6 characters | password=abc123 | M |

* M = Mandatory parameter, O = Optional parameter

# HTTP API - 'sendmessage' action

**This http api command can be used to send an SMS message via the SMS Gateway. The command can be used to send text SMS messages or other message types, such as binary SMS messages, operator logos, ringtones, WAP PUSH, etc... When you use this command, you must use an http api username and password, you must specify the recipient phone number, and the the message text.**

## Description

To send an SMS, use the following URL format:

https://127.0.0.1:9508/api?action=sendmessage&username=UUUUU&password=PPPPP&
recipient=NNNNN&messagetype=MMMMM&messagedata=DDDDD

127.0.0.1 is a local IP, so please replace it to the IP address or host name of the computer with the installed Ozeki SMS Gateway. (Note: 127.0.0.1 is a local loopback address that can be used when you are connecting to the SMS Gateway from the same computer.) **9508 is the default port of Ozeki SMS Gateway's HTTP API. This port number can be viewed and edited in Ozeki 10 SMS Gateway-s user interface, by clicking on the Advanced button in the toolbar.**

"UUUUU" and "PPPPP" should be replaced to the username and password of the user you have created in the SMS gateway.

Substitute "NNNNN" to the phone number you wish to send the SMS message to. You can use the local phone number format as well as international phone number formats (telephone numbers formatted according to the international number format always start with a '+' sign). If the international phone number format is used, note that you must substitute '%2B' for the '+' character, because of URL encoding rules.

Please substitute "MMMMM" for the message type. "SMS:TEXT" message type should be used for text messages.

The message data contains the message you would like to send. Place the message data instead of "DDDDD". The message data should be using UTF-8 characters and should be URL encoded.

Other parameters can also be added to the request.
**For a complete list of available parameters please take a look at the 'Request parameters' table below:**

## Example URL Request

https://127.0.0.1:9508/api?action=sendmessage&username=admin&password=abc123&
recipient=06203105366&messagetype=SMS:TEXT&messagedata=Hello+World

## Example Response

```
HTTP/1.1 200 OK
Content-Type: application/xml
Content-Length: 246

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE smsapi PUBLIC "-//OZEKI//DTD XML 1.0//EN" "http://www.ozekisms.com/DTD/smsapi.xml">
<response>
<action>sendmessage</action>
<data>
<acceptreport>
<statuscode>0</statuscode>
<statusmessage>Message accepted for delivery</statusmessage>
<messageid>ERFAV23D</messageid>
<recipient>06203105366</recipient>
</acceptreport>
```

```
        </data>
</response>
```

# HTTP request parameters

| Parameter | Description | Possible values | Example | M/O* |
|---|---|---|---|---|
| action | Specifies the HTTP API command | **sendmessage** | action=sendmessage | M |
| username | Specifies the username. The username and password parameters are used to authenticate the user. When you send an message it will be sent in the name of the authenticated user. The value must be urlencoced. | string value, maximum length is 16 characters | username=admin | M |
| password | Specifies the password. The username and password parameters are used to authenticate the user. When you send an message it will be sent in the name of the authenticated user. The value must be urlencoced. | string value, maximum length is 16 characters | password=abc123 | M |
| originator | Specifies the sender address. This information will be displayed on the mobile phone, that receives the message. This is the sender address. This can be a telephone number, a short code or an alphanumeric sender address. The telephone number must can be formatted in local number format (e.g.06201234567) or in international number format (e.g.+36201234567). If you use an alphanumeric sender address (e.g.ozeki), the characters must be encoded in UTF8 and the value must be urlencoded. | string value, maximum length is 16 characters | originator=%2B36201112222 | O |
| recipient | Specifies the recipient phone number. The message will be sent to this telephone number. The telephone number can be specified in local number format (e.g.06201234567), or in international number format (e.g.+36201234567).<br>More then one recipient addresses can be separated by a colon (e.g.: +36201234567,+36202222222) or semi-colon.<br>The value must be urlencoded. | string value, maximum length is 16 characters | recipient=%2B36201234567 | M |
| messagetype | Specifies the message type. The type of the SMS message data based on the Mobile Message Type Specification. For text messages the message data will be plain text, for other message types it will be an XML document. | SMS:TEXT<br>SMS:WAPPUSH<br>...<br>Possible values can be found in the Mobil Message Type Specification | messagetype=SMS:TEXT | O |
| messagedata | Specifies the text or the data of the SMS message. The value must be encoded in UTF8 and must be urlencoded. | string value, maximum length is 32768 characters | messagedata=Hello+World | M |
| _charset_ | Specifies the charset of the encoded data (if not specified utf- | utf-8<br>windows-1250 | _charset_=iso-8859-2 | O |

| | | | | |
|---|---|---|---|---|
| | 8 will be assumed). Newer browsers should set the value of _charset_ automatically. If your browser doesn't support this feature you can set is manually to: | iso-8859-1<br>iso-8859-2<br>...<br>(Supported character set values) | | |
| serviceprovider | Specifies the name of the GSM Modem or IP SMS service provider connection to use to send the message. More information about service provider connection selection is available in the HTTP SMS API - SMS routing guide.<br>The value must match the string specified on the configuration form of the service provider connection. | string value, maximum length is 16 characters | serviceprovider=Vodafone | O |
| sendondate | Specifies the date and time, when the message should be sent.<br>The value must use the following date format: YYYY-MM-DD hh:mm:ss. The value must be url encoded | date value in YYYY-MM-DD hh:mm:ss format | sendondate=2018-12-12+10%3A07%3A05 | O |
| responseformat | After the gateway submits the SMS message, the gateway will return a web page indicating that the message was submitted successfully. The content of the webpage is formatted according to the responseformat parameter. You can have html text response to make it easy for humans to read it or you can have xml format to make it easy for software to process the response. | xml (default)<br>html<br>urlencoded | responseformat=xml | O |
| continueurl | After the gateway submits the SMS message, the gateway will return a web page indicating that the message was submitted successfully. The content of the webpage is formatted according to the responseformat parameter. If the responseformat parameter is set to html, the webpage can contain a "Continue" link. If you specify the URL in this parameter the continue link will be displayed and it will point to the specified URL. The value of the URL must be urlencoded.<br>The URL you specify can contain keywords that will be replaced to state information corresponding to the submitted message. More information about possible keywords can be found in the "Submit URL keywords" guide. | string value, maximum length is 1024 characters | continueurl=www.ozekisms.com<br>or<br>continueurl=http%3A%2F%2Fwww.ozekisms.com%2Findex.php%3Fowpn%3D159<br>*Note: the second example contains an urlencoded URL.* | O |
| redirecturl | After the gateway submits the SMS message, by default the gateway will return a web page indicating that the message was submitted successfully. Optionally you can ask the gateway to automatically redirect the browser to an URL you | string value, maximum length is 1024 characters | redirecturl=www.ozekisms.com<br>or<br>redirecturl=http%3A%2F%2Fwww.ozekisms.com%2Findex.php%3Fowpn%3D159 | O |

| | | | | |
|---|---|---|---|---|
| | specify. If you specify an URL in the redirect parameter, the HTTP response returned by the SMS gateway will contain a redirect URL in the HTTP header. This will instruct the webbrowser to follow the link you have specified. The value of the URL must be urlencoded.<br>The URL you specify can contain keywords that will be replaced to state information corresponding to the submitted message. More information about possible keywords can be found in the "Submit URL keywords" guide. | | *Note: the second example contains an urlencoded URL.* | |
| reporturl | You can setup a webpage to process information about **"delivered to network"** and **"delivered to handset"** events. If you specify an URL in the reporturl parameter, your webpage will be called when these events happen. The value of the URL you specify in the reporturl parameter must be URL encoded.<br>The URL you specify can contain keywords that will be replaced to state information corresponding to the submitted message.<br>A good example on how to use the report URL option can be found in the how to send a scheduled sms and use the report url functionality guide. | string value, maximum length is 1024 characters | reporturl=http%3A%2F%2Fwww.ozekisms.com%2Fproc.php%3Freporttype%3D%24reporttype%26messageid%3D%24messageid<br>*Note: this is the urlencoded version of the following URL. Before this URL is called by the SMS gateway, the $reporttype and $messageid parameters will be replaced to the appropriate values:*<br>*http://www.ozekisms.com/proc.php?reporttype=$reporttype&messageid=$messageid*<br>*will be called as:*<br>*http://www.ozekisms.com/proc.php?reporttype=deliveredtonetwork&messageid=ERFAV23D*<br>*The list of keywords you can use in the report url are:*<br>*$reporttype*<br>*$messageid*<br>*$statuscode*<br>*$statusmessage*<br>*$fromstation*<br>*$fromconnection*<br>*$fromaddress*<br>*$tostation*<br>*$toconnection*<br>*$toaddress*<br>*$text*<br>*$createdate*<br>*$submitdate*<br>*$receiveddate* | O |
| messagecount | It specifies the exact number of messages you would like to send. If set, indexing is needed for the 'recipient', 'messagetype' and 'messagedata' parameters. The detailed 'messagecount' tutorial can be found here. | number (default value: 1) | messagecount=6 | O |
| maxresponse | This number specifies the maximum of messages about which you will receive a feedback. If you exceed this number, your messages will be sent out but you won't receive feedback about them. By default, this parameter is set to 500 messages. | number | maxresponse=1000 | O |
| vp | Specifies the **v**alidity **p**eriod for your message.<br>The value must use the following | date value in YYYY.MM.DD hh:mm:ss format | vp=2019.01.28.+10%3A07%3A05 | O |

| | | | | |
|---|---|---|---|---|
| | date format: YYYY.MM.DD hh:mm:ss. The value must be url encoded.<br>Read this tutorial for more information. | | | |

* M = Mandatory parameter, O = Optional parameter

# Response parameters

**(xml response format)**

| Parameter | Description | Possible values | Example |
|---|---|---|---|
| acceptreport | Contains the response for the submit request for a single recipient address. If many recipients were specified, an acceptreport will be included in the response for each recipient. The order of acceptreports will match the order of the recipient addresses. | | `<acceptreport>`<br>`   <statuscode>0</statuscode>`<br>`   <statusmessage>Message accepted`<br>`   for delivery</statusmessage>`<br>`   <messageid>ERFAV23D</messageid>`<br>`   <recipient>06203105366</recipient>`<br>`</acceptreport>` |
| acceptreport.statuscode | Contains an integer value to indicate success or failure. If the value is 0, it means the message was accepted for delivery. If the value is greater then 0 it means there was an error, the message was not accepted for delivery. | Integer value, greater then or equal to 0. Less the 32768. | <statuscode>0</statuscode> |
| acceptreport.statusmessage | Contains a textual representation of the status code. If the message was accepted the value will be "Message accepted for delivery". If the message was not accepted for delivery, you can find the error message in this field. | string value, maximum length is 1024 characters | <statusmessage>Message accepted for delivery</statusmessage> |
| acceptreport.messageid | Contains a message reference that can be used to track the message in the SMS gateway. This message reference is also used to identify delivered to network and delivered to handset reports or to query information about the message. | string value, maximum length is 16 characters | <messageid>ERFAV23D</messageid> |
| acceptreport.recipient | Contains the recipient address. | string value, maximum length is 16 characters | <recipient>06203105366</recipient> |

# HTTP API - URL template

**This guide gives you instructions on how to create an URL template to be used in the redirecturl, the continueurl and the reporturl parameters of the HTTP API sendmessage method.**

## Introduction

If the the redirecturl or the continueurl parameters are included in your sendmessage HTTP request, the SMS server will return a response containing the URL. If the URL specified contains certain keywords. The keywords will be replaced to the appropriate values corresponding to the submitted message. In other words an URL template is an URL that will be processed by the SMS gateway before it is used in the HTTP response. Processing means, that certain keywords will be replaced in the URL.

For example if you specify the following URL in the continueurl parameter:

http://www.myserver.com/index.asp?msgid=**$messageid**&statusmsg=**$statusmessage**

It will be modified into this when it is returned in the HTTP response:

http://www.myserver.com/index.asp?msgid=**AC32DEF**&statusmsg=**Message+accepted+for+delivery**
Note that the keywords $messageid and $statusmessage have been replaced. The message id of the submitted messages is substituted for the $messageid keyword in the URL and the status message of the submission is is substituted for the $statusmessage keywrod. Ozeki NG SMS Gateway replaces the keywords specified in the table bellow.

## How to use?

To pass an URL template in the redirecturl, the continueurl or in the reporturl parameter, the URL template must be urlencoded. This is necessary, because otherwise the HTTP syntax of the sendmessage method would break. To urlencode your URL template you can use the online url encoding tool (available at http://www.ozeki.hu/index.php?owpn=5679) or you can do the encoding manually.

The URL template we used in the example above looks like this before the encoding:

http://www.myserver.com/index.asp?msgid=**$messageid**&statusmsg=**$statusmessage**

After urlencogind it will look like this:

http%3A%2F%2Fwww.myserver.com%2Findex.asp%3Fmsgid%3D%24messageid%26statusmsg%3D%24statusmessage

If you want to include this URL template in your submit request, your request would look like this:

http://127.0.0.1:9501/api?action=sendmessage&username=admin&password=abc123&recipient=06203105366&messagetype=SMS:TEXT&messagedata=Hello+World&**redirecturl=http%3A%2F%2Fwww.myserver.com%2Findex.asp%3Fmsgid%3D%24messageid%26statusmsg%3D%24statusmessage**

## Keywords

URL templates can contain the following keywords

| **Keyword** | **Replaced to** | Can be used in the following parameter of the HTTP API sendmessage method |
|---|---|---|
| statuscode | The result code of the message submission | redirecturl<br>continueurl<br>reporturl |
| statusmessage | The status message of the message submission. (The | redirecturl |

| | | |
|---|---|---|
| | statuscode explained.) | continueurl<br>reporturl |
| messageid | The id of the submitted message | redirecturl<br>continueurl<br>reporturl |
| recipient | The recipient phone number of the submitted message | redirecturl<br>continueurl<br>reporturl |
| originator | The sender name or sender phone number of the submitted message | redirecturl<br>continueurl<br>reporturl |
| messagetype | The messagetype of the submitted message | redirecturl<br>continueurl<br>reporturl |
| messagedata | The messagedata of the submitted message | redirecturl<br>continueurl<br>reporturl |
| submitdate | The date and time value of the message submission event. The date format is YYYY-MM-DD hh:mm:ss. | reporturl |
| deliveredtonetworkdate | The date and time value of the message delivered to network event. The date format is YYYY-MM-DD hh:mm:ss. | reporturl |
| deliveredtohandsetdate | The date and time value of the message delivered to handset event. The date format is YYYY-MM-DD hh:mm:ss. | reporturl |
| status | The current status of the submitted message. Possible values:<br>deliveredtonetwork<br>deliveredtohandset<br>deliveryfailed<br>If the delivery fails, the statuscode and statusmessage parameters contain the errorcode and the errormessage returned by the service provider | reporturl |

If more then one messages are submitted with a single HTTP request, the URL template will contain the values corresponding to the last message in the list.

# HTTP API - Submitting multiple SMS over HTTP Post

To get better performance it is a good practice to send several messages with a single HTTP request. You can do this by adding the messagecount parameter to your request and by attaching a sequence number to the end of each HTTP api parameter that relates to the submitted messages.

**Example URL Request for HTTP GET**

https://127.0.0.1:9508/api?action=sendmessage&messagecount=2&username=admin&password=abc123&
recipient0=06203105366&messagetype0=SMS:TEXT&messagedata0=Hello+World&
recipient1=06202222222&messagetype1=SMS:TEXT&messagedata1=Second+Message

**Example Request for HTTP POST**

If you have a large number of messages, you should use HTTP Post to submit your request. You can do this by using the following HTML form:

```
<form action=https://127.0.0.1:9508/api method=post><br>
action: <input type=text name=action value=sendmessage><br>
messagecount: <input type=text name=messagecount value=2><br>
username: <input type=text name=username value=admin><br>
password: <input type=text name=password value=abc123><br>
recipient0: <input type=text name=recipient0 value=06203105366><br>
messagetype0: <input type=text name=messagetype0 value=SMS:TEXT><br>
messagedata0: <input type=text name=messagedata0 value='Hello World'><br>
recipient1: <input type=text name=recipient1 value=06202222222><br>
messagetype1: <input type=text name=messagetype1 value=SMS:TEXT><br>
messagedata1: <input type=text name=messagedata1 value='Second Message'><br>
<input type=submit value=OK>
</form>
```

This will produce the following HTTP POST request:

```
POST /api HTTP/1.1
Host: 127.0.0.1:9509
User-Agent: Mozilla/5.0 (Windows; U; Windows NT 5.1; en-US; rv:1.9.0.10)
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-us,en;q=0.5
Accept-Encoding: gzip,deflate
Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7
Keep-Alive: 300
Connection: keep-alive
Content-Type: application/x-www-form-urlencoded
Content-Length: 211

action=sendmessage&messagecount=2&username=admin&password=abc123&recipient0=06203105366&
messagetype0=SMS%3ATEXT&messagedata0=Hello+World&recipient1=06202222222&messagetype1=SMS%3ATEXT&
messagedata1=Second+Message
```

The response for this request will be:

```
<?xml version="1.0" encoding="utf-8"?>

<response>
  <action>sendmessage</action>
  <data>
    <acceptreport0>
      <statuscode0>0</statuscode0>
      <statusmessage0>Message accepted for delivery</statusmessage0>
      <messageid0>c4ce4d5e-b2d1-4399-bd2e-83dff97dfb19</messageid0>

      <originator0>+44777888</originator0>
      <recipient>06203105366</recipient>
      <messagetype0>SMS:TEXT</messagetype0>
      <messagedata0>Hello World</messagedata0>
      <serviceprovider0 />
    </acceptreport0>
    <acceptreport1>

      <statuscode1>0</statuscode1>
      <statusmessage1>Message accepted for delivery</statusmessage1>
      <messageid1>5ebae14b-8b58-4d99-979b-71e0bbd66132</messageid1>
      <originator1>+44777888</originator1>
      <recipient>06202222222</recipient>
      <messagetype1>SMS:TEXT</messagetype1>

      <messagedata1>Second Message</messagedata1>
      <serviceprovider1 />
    </acceptreport1>
  </data>
</response>
```

# HTTP API - Redirect browser

**This example posts an SMS message to the SMS gateway, then redirects the browser to an url. The url in the example is http://www.yahoo.com**

## Example request:

https://127.0.0.1:9508/api?action=sendmessage&username=admin&password=abc123&recipient=06203105366&messagetype=SMS:TEXT&messagedata=Hello+World&**redirecturl=http%3A%2F%2Fwww.yahoo.com**

## Example response:

```
HTTP/1.1 302 Found
Cache-Control: no-cache, must-revalidate
Pragma: no-cache
Content-Length: 440
Content-Type: text/xml
Last-Modified: Wed, 13 Feb 2008 23:19:47 GMT
Location: http://www.yahoo.com
Server: OzekiNG/3.0.1 Microsoft-HTTPAPI/1.0
Date: Wed, 13 Feb 2008 22:19:46 GMT

<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE response PUBLIC "-//OZEKI//DTD XML 1.0//EN" "http://192.168.91.10:9501/DTD/response.xml">
<response>
<action>sendmessage</action>
<data>
<acceptreport>
<statuscode>0</statuscode>
<statusmessage>Message accepted for delivery</statusmessage>
<messageid>ROTTTZFS</messageid>
<recipient>06203105366</recipient>
</acceptreport>
</data>
</response>
```

# HTTP API - Supported Character Sets

ASMO-708
big5
cp1025
cp866
cp875
csISO2022JP
DOS-720
DOS-862
EUC-CN
euc-jp
EUC-JP
euc-kr
GB18030
gb2312
hz-gb-2312
IBM00858
IBM00924
IBM01047
IBM01140
IBM01141
IBM01142
IBM01143
IBM01144
IBM01145
IBM01146
IBM01147
IBM01148
IBM01149
IBM037
IBM1026
IBM273
IBM277
IBM278
IBM280
IBM284
IBM285
IBM290
IBM297
IBM420
IBM423
IBM424
IBM437
IBM500
ibm737
ibm775
ibm850
ibm852
IBM855
ibm857
IBM860
ibm861
IBM863
IBM864
IBM865
ibm869
IBM870
IBM871
IBM880
IBM905
IBM-Thai
iso-2022-jp
iso-2022-kr
iso-8859-1
iso-8859-13
iso-8859-15
iso-8859-2
iso-8859-3
iso-8859-4
iso-8859-5

iso-8859-6
iso-8859-7
iso-8859-8
iso-8859-8-i
iso-8859-9
Johab
koi8-r
koi8-u
ks_c_5601-1987
macintosh
shift_jis
unicodeFFFE
us-ascii
utf-16
utf-32
utf-32BE
utf-7
utf-8
windows-1250
windows-1251
Windows-1252
windows-1253
windows-1254
windows-1255
windows-1256
windows-1257
windows-1258
windows-874
x-Chinese-CNS
x-Chinese-Eten
x-cp20001
x-cp20003
x-cp20004
x-cp20005
x-cp20261
x-cp20269
x-cp20936
x-cp20949
x-cp50227
x-EBCDIC-KoreanExtended
x-Europa
x-IA5
x-IA5-German
x-IA5-Norwegian
x-IA5-Swedish
x-iscii-as
x-iscii-be
x-iscii-de
x-iscii-gu
x-iscii-ka
x-iscii-ma
x-iscii-or
x-iscii-pa
x-iscii-ta
x-iscii-te
x-mac-arabic
x-mac-ce
x-mac-chinesesimp
x-mac-chinesetrad
x-mac-croatian
x-mac-cyrillic
x-mac-greek
x-mac-hebrew
x-mac-icelandic
x-mac-japanese
x-mac-korean
x-mac-romanian
x-mac-thai
x-mac-turkish
x-mac-ukrainian

# HTTP API - Set validity period for your messages

**This page explains briefly how you can specify the validity period for your messages. Please find the necessary parameter below this page.**

Ozeki NG SMS Gateway allows you to send SMS messages from various applications. If you use HTTP API for sending messages, you can set the validity period for your messages.

For specifying the validity period, please use the following parameter:

```
&vp=2011.07.26.+10%3A07%3A58
```

Please note that **2011.07.26.+10%3A07%3A58** is the Urlencoded form of **2011.07.26. 10:07:58**.

With the use of this parameter, you can send validity information to the service provider. Therefore, the service provider can set messages to be valid for the specified period.

Please check the following example on how to use the above parameter:

```
https://127.0.0.1:9509/api?action=sendmessage&username=admin&password=abc123
&recipient=06203105366&messagetype=SMS:TEXT &messagedata=Hello+World&vp=2011.07.26.+10%3A07%3A58
```

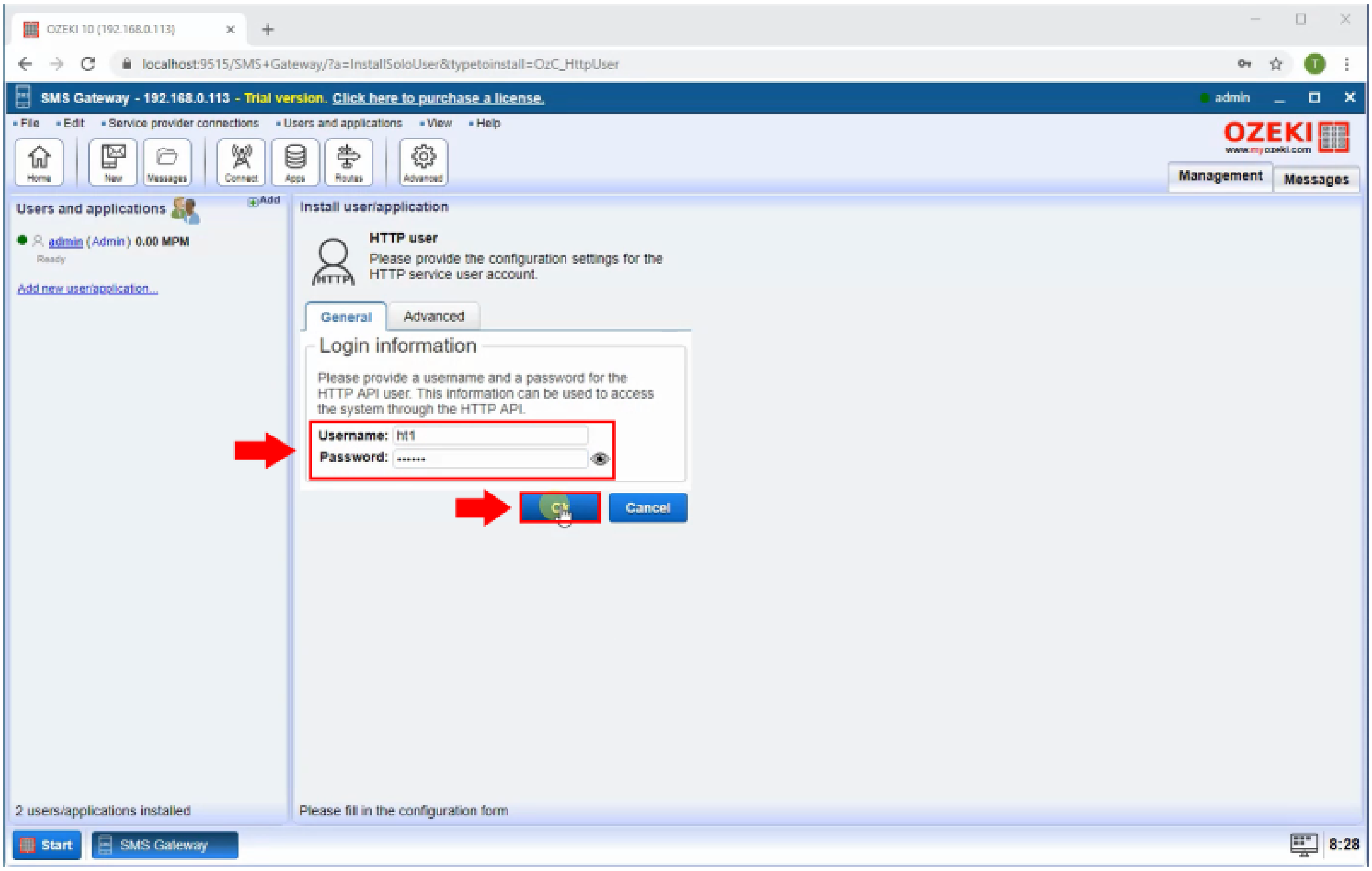
# How to send a scheduled SMS message using the HTTP API

***This guide helps you to understand how you can send scheduled SMS messages using HTTP API. With this solution, you can set the exact date when you would like to send the message. The document contains a video tutorial and a step by step guide where each step consists of a short description of the current step and a screenshot to demonstrate what you need to do. To complete this guide, you don't need to have any specific knowledge and it does not need more than ten minutes to complete. So, let's start right now!***

## Video tutorial

## Step 1 - Create HTTP user

In the first step, you need to create the HTTP user connection. For that, open the Apps menu from the toolbar, and here, in the Network users section, click on the Install button of the HTTP user. To start configuring a HTTP user please provide a username and a password for the HTTP API user as you can see in the Figure 1. This information can be used to access the system through the HTTP API.



localhost:9515/SMS+Gateway/?a=InstallSoloUser&typetoinstall=OzC_HttpUser

SMS Gateway - 192.168.0.113 - Trial version. Click here to purchase a license.

Install user/application

HTTP user

Please provide the configuration settings for the HTTP service user account.

Login information

Please provide a username and a password for the HTTP API user. This information can be used to access the system through the HTTP API.

Username: ht1

Password: ••••••

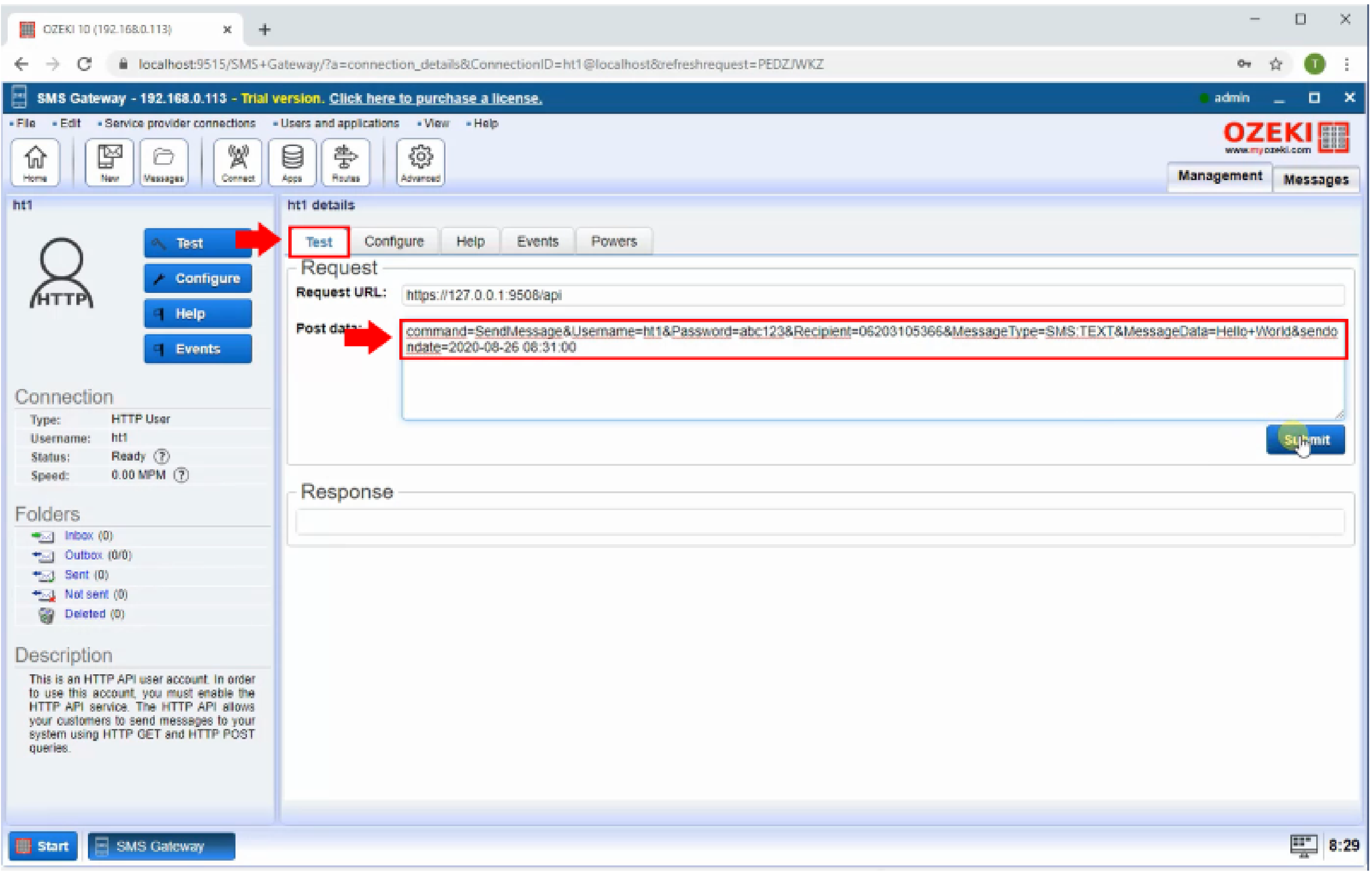
Please fill in the configuration form

Figure 1 - Create Ozeki SMS Gateway HTTP user

## Step 2 - HTTP request for send message

After you created the HTTP user connection, just open the connection to see the Details page of the connection. Here, you need to navigate to the Test tab, where you can send HTTP request to send a SMS message across the Ozeki SMS Gateway. To the request, add the sendondate parameter where you can define that when will be the message sent by the Ozeki SMS Gateway as Figure 2 demonstrates it. To initiate the HTTP request, just click on the Submit button. You can find this command here:

```
command=SendMessage&Username=HTTP_User_1&Password=qwe123&Recipient=06203105366&
MessageType=SMS:TEXT&MessageData=Hello+World&sendondate=2020-08-26 08:31:00
```



Request

Request URL: https://127.0.0.1:9508/api

Post data: command=SendMessage&Username=ht1&Password=abc123&Recipient=06203105366&MessageType=SMS:TEXT&MessageData=Hello+World&sendondate=2020-08-26 08:31:00

Response

Connection

Type: HTTP User
Username: ht1
Status: Ready
Speed: 0.00 MPM

Folders

Inbox (0)
Outbox (0/0)
Sent (0)
Not sent (0)
Deleted (0)

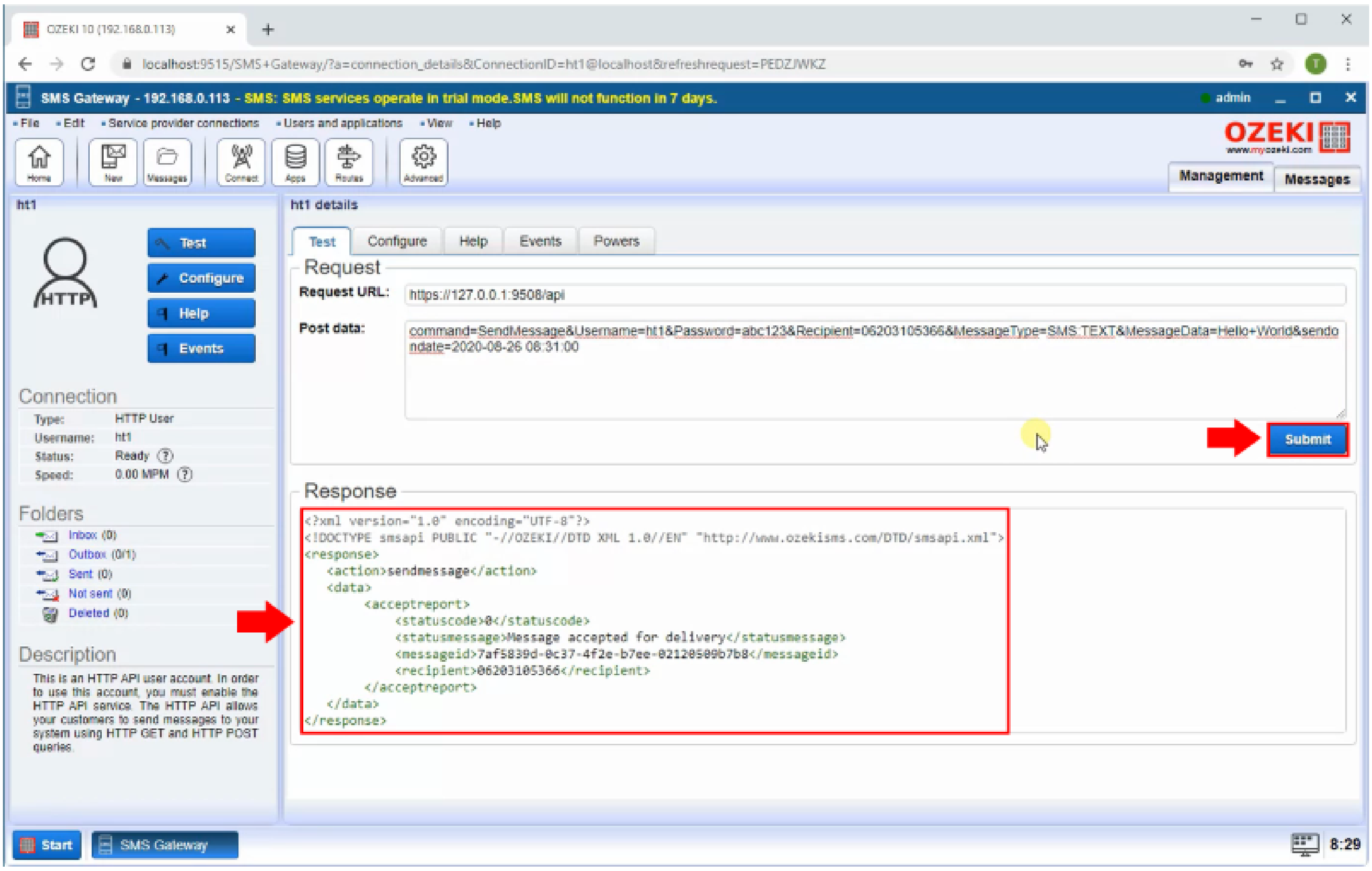
Description

This is an HTTP API user account. In order to use this account, you must enable the HTTP API service. The HTTP API allows your customers to send messages to your system using HTTP GET and HTTP POST queries.

Figure 2 - Send message HTTP request

# Step 3 - Message accepted by Ozeki SMS Gateway

After you clicked on the Submit button and you initiated the HTTP request, you will be able to see the repsonse message from the Ozeki SMS Gateway as the Figure 3 shows below. This response message contains the status code and the status message as well. This message should tell that the message was accepted for delivery. If the message is accepted it will be placed in the outbox folder.



Connection

Type: HTTP User
Username: ht1
Status: Ready
Speed: 0.00 MPM

Description

This is an HTTP API user account. In order to use this account, you must enable the HTTP API service. The HTTP API allows your customers to send messages to your system using HTTP GET and HTTP POST queries.

Request

Request URL: https://127.0.0.1:9508/api

Post data:

```
command=SendMessage&Username=ht1&Password=abc123&Recipient=06203105366&MessageType=SMS:TEXT&MessageData=Hello+World&sendondate=2020-08-26 08:31:00
```

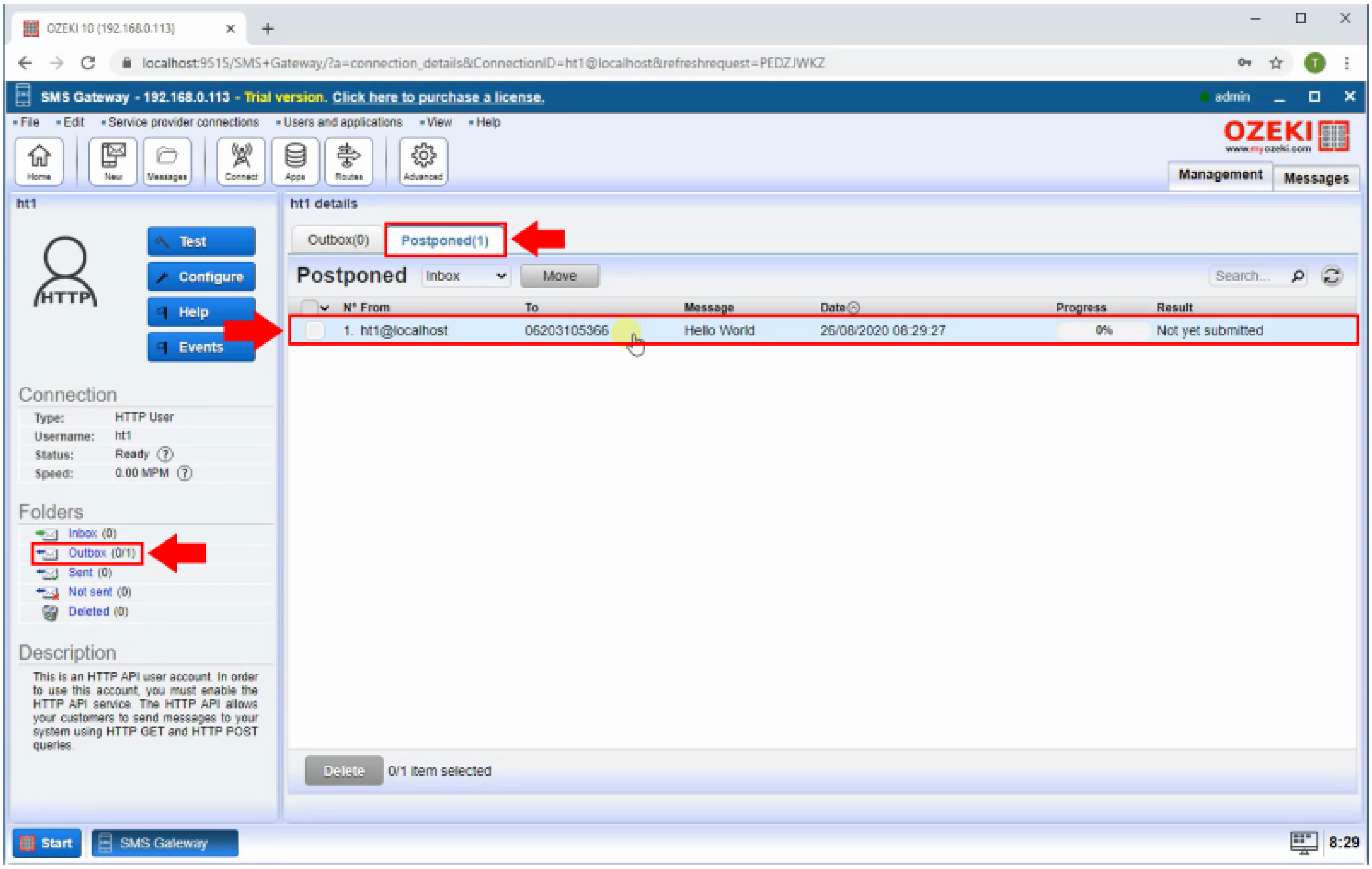
Response

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE smsapi PUBLIC "-//OZEKI//DTD XML 1.0//EN" "http://www.ozekisms.com/DTD/smsapi.xml">
<response>
   <action>sendmessage</action>
   <data>
        <acceptreport>
            <statuscode>0</statuscode>
            <statusmessage>Message accepted for delivery</statusmessage>
            <messageid>7af5839d-0c37-4f2e-b7ee-02120509b7b8</messageid>
            <recipient>06203105366</recipient>
        </acceptreport>
   </data>
</response>
```

Figure 3 - Message accepted

## Step 4 - Message in the outbox folder

The accepted message stored in the Outbox folder of the HTTP user connection. Before the date of the sending, it can be found in the list of the Postponed tab as you can see it in Figure 4. At this point, it waits to reach the sending date and be delivered.



| N° | From | To | Message | Date | Progress | Result |
|---|---|---|---|---|---|---|
| 1. | ht1@localhost | 06203105366 | Hello World | 26/08/2020 08:29:27 | 0% | Not yet submitted |

Connection

Type: HTTP User
Username: ht1
Status: Ready
Speed: 0.00 MPM

Folders

Inbox (0)
Outbox (0/1)
Sent (0)
Not sent (0)
Deleted (0)

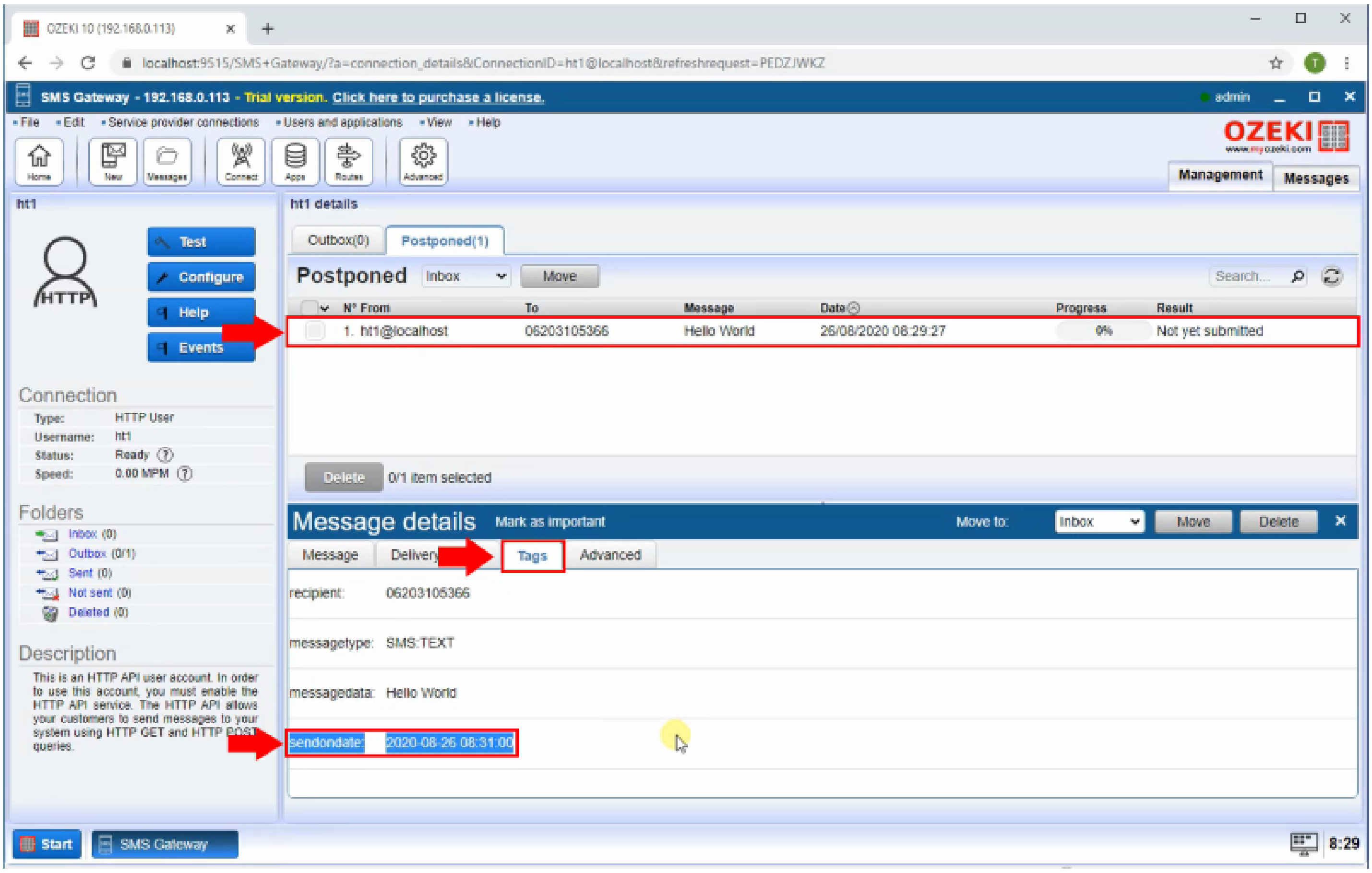
Description

This is an HTTP API user account. In order to use this account, you must enable the HTTP API service. The HTTP API allows your customers to send messages to your system using HTTP GET and HTTP POST queries.

Figure 4 - Message in outbox folder

# Step 5 - Message tags

In that list of postponed messages, you can click on the selected message to see the details of it. Here, you can select the Tags tab as Figure 5 shows that. This tab provides you the main tags that describes the details of the message. Here, you can see the sendondate tag, that shows the date when the message is about to be sent by the SMS Gateway.


OZEKI 10 (192.168.0.113)
localhost:9515/SMS+Gateway/?a=connection_details&ConnectionID=ht1@localhost&refreshrequest=PEDZJWKZ
SMS Gateway - 192.168.0.113 - Trial version. Click here to purchase a license.
File Edit Service provider connections Users and applications View Help
Home New Messages Connect Apps Routes Advanced
Management Messages
Test Configure Help Events
Outbox(0) Postponed(1)
Postponed Inbox Move
Delete 0/1 item selected
Message details Mark as important Move to: Inbox Move Delete
Message Delivery Tags Advanced
Start SMS Gateway 8:29

ht1

Connection
Type: HTTP User
Username: ht1
Status: Ready
Speed: 0.00 MPM

Folders
Inbox (0)
Outbox (0/1)
Sent (0)
Not sent (0)
Deleted (0)

Description
This is an HTTP API user account. In order to use this account, you must enable the HTTP API service. The HTTP API allows your customers to send messages to your system using HTTP GET and HTTP POST queries.

ht1 details

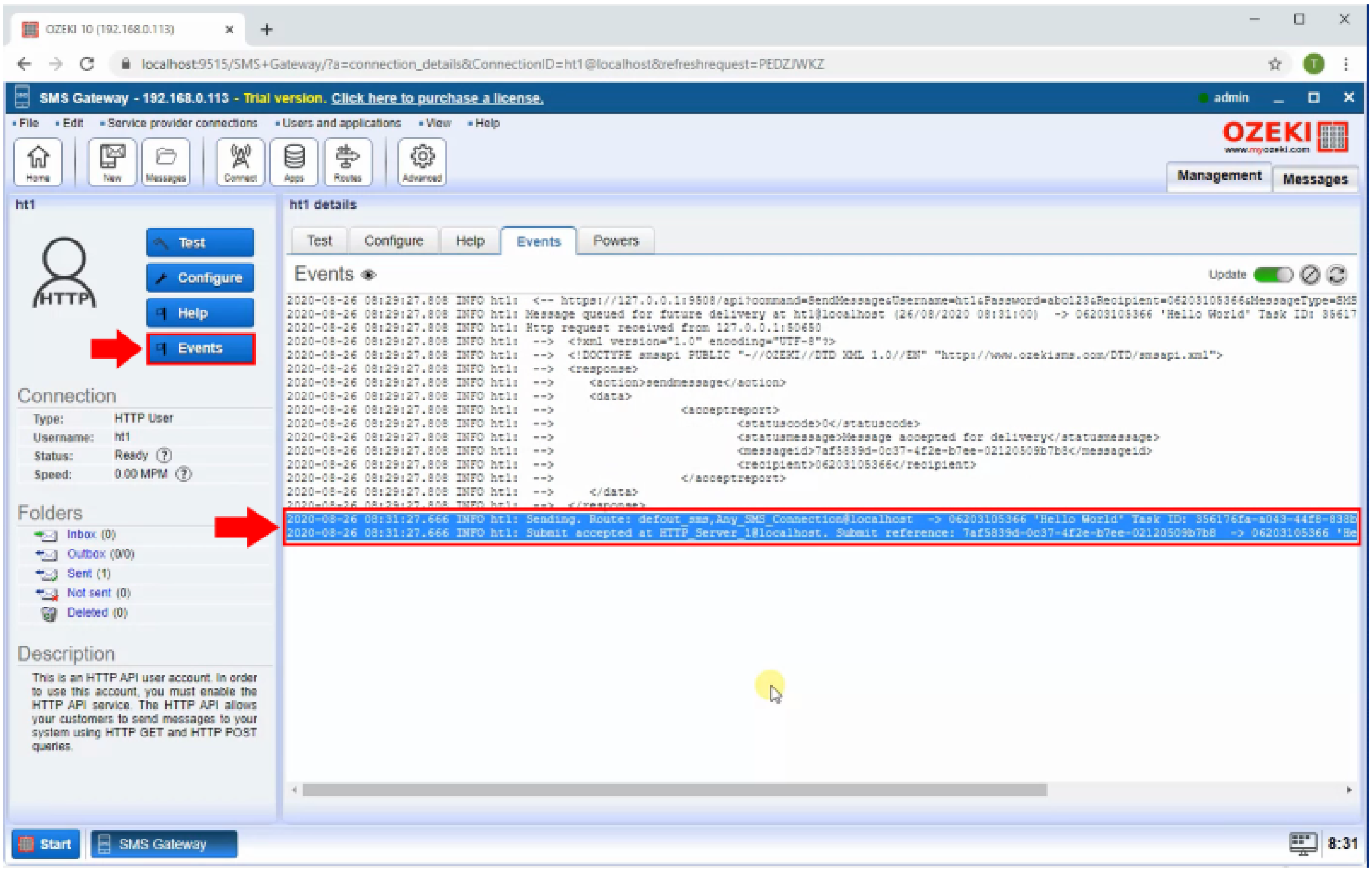
| N° | From | To | Message | Date | Progress | Result |
|---|---|---|---|---|---|---|
| 1. | ht1@localhost | 06203105366 | Hello World | 26/08/2020 08:29:27 | 0% | Not yet submitted |

recipient: 06203105366
messagetype: SMS:TEXT
messagedata: Hello World
sendondate: 2020-08-26 08:31:00

Figure 5 - Message tags

## Step 6 - Message sent on the selected date

As soon as the sending date arrives, the HTTP user connection sends the message to the specified recipient. If you would like to see if the delivery was successful, you can check the Events menu of the HTTP user connection. For that, select the Events menu like in Figure 6, and here, you will be able to see the events of the message. Every event contains a timestamp and a simple description about the event itself.



Connection

Type: HTTP User
Username: ht1
Status: Ready
Speed: 0.00 MPM

Folders

- Inbox (0)
- Outbox (0/0)
- Sent (1)
- Not sent (0)
- Deleted (0)

Description

This is an HTTP API user account. In order to use this account, you must enable the HTTP API service. The HTTP API allows your customers to send messages to your system using HTTP GET and HTTP POST queries.

```
2020-08-26 08:29:27.808 INFO ht1:  <-- https://127.0.0.1:9508/api?command=SendMessage&Username=ht1&Password=abc123&Recipient=06203105366&MessageType=SMS
2020-08-26 08:29:27.808 INFO ht1: Message queued for future delivery at ht1@localhost (26/08/2020 08:31:00)  -> 06203105366 'Hello World' Task ID: 35617
2020-08-26 08:29:27.808 INFO ht1: Http request received from 127.0.0.1:50650
2020-08-26 08:29:27.808 INFO ht1:  -->  <?xml version="1.0" encoding="UTF-8"?>
2020-08-26 08:29:27.808 INFO ht1:  -->  <!DOCTYPE smsapi PUBLIC "-//OZEKI//DTD XML 1.0//EN" "http://www.ozekisms.com/DTD/smsapi.xml">
2020-08-26 08:29:27.808 INFO ht1:  -->  <response>
2020-08-26 08:29:27.808 INFO ht1:  -->     <action>sendmessage</action>
2020-08-26 08:29:27.808 INFO ht1:  -->     <data>
2020-08-26 08:29:27.808 INFO ht1:  -->                 <acceptreport>
2020-08-26 08:29:27.808 INFO ht1:  -->                         <statuscode>0</statuscode>
2020-08-26 08:29:27.808 INFO ht1:  -->                         <statusmessage>Message accepted for delivery</statusmessage>
2020-08-26 08:29:27.808 INFO ht1:  -->                         <messageid>7af5839d-0c37-4f2e-b7ee-02120509b7b8</messageid>
2020-08-26 08:29:27.808 INFO ht1:  -->                         <recipient>06203105366</recipient>
2020-08-26 08:29:27.808 INFO ht1:  -->                 </acceptreport>
2020-08-26 08:29:27.808 INFO ht1:  -->     </data>
2020-08-26 08:29:27.808 INFO ht1:  -->  </response>
2020-08-26 08:31:27.666 INFO ht1: Sending. Route: defout_sms,Any_SMS_Connection@localhost  -> 06203105366 'Hello World' Task ID: 356176fa-a043-44f8-838b
2020-08-26 08:31:27.666 INFO ht1: Submit accepted at HTTP_Server_1@localhost. Submit reference: 7af5839d-0c37-4f2e-b7ee-02120509b7b8  -> 06203105366 'He
```

Figure 6 - Message sent on date

# HTTP API - 'receivemessage' action

**This action can be used to receive SMS messages from the SMS Gateway using HTTP polling. HTTP polling means that you download SMS messages from the inbox folder of the gateway like you would download a webpage. In this section you can get information about how you can use the receiveimessage action, what are the parameters, and you will be presented with an example HTTP request to download incoming messages.**

## Description

To download incoming text messages use the following URL format:

https://127.0.0.1:9508/api?action=receivemessage&username=UUUUU&password=PPPPP&folder=inbox&limit=2&afterdownload=delete

For 127.0.0.1, you should substitute the host name or the IP address of the computer your SMS gateway is installed on. The port number 9508 is the default HTTP port number of the Ozeki SMS gateway. It port number can be configured in the user interface of Ozeki 10 SMS Gateway, by clicking on the Advanced button in the toolbar.

The username and the password should be substituted for "UUUUU" and "PPPPP". The username and password identifies the user you have created in the SMS gateway. When you invoke the receivemessage action by calling the URL, you will download messages from the inbox of the user specified in the query with the username and the password.

The folder parameter of the message should be set to inbox.

The limit parameter specifies the number of messages that will be downloaded. It is recommended to download less then 1000 messages in a single request. You can download the remaining messages in subsequent requests.

The afterdownload parameter can be used to remove messages from the SMS gateway once they have been successfully downloaded.

## Example URL Request

https://127.0.0.1:9508/api?action=receivemessage&username=admin&password=abc123&folder=inbox&limit=2&afterdownload=delete

## Example Response

```
HTTP/1.1 200 OK
Content-Type: application/xml
Content-Length: 824

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE response PUBLIC "-//OZEKI//DTD XML 1.0//EN" "http://www.ozekisms.com/DTD/response.xml">
<response>
<action>receivemessage</action>
<data>
<message>
<messageid>ERFAV23D</messageid>
<originator>06301234567</originator>
<recipient>06201112222</recipient>
<messagetype>SMS:TEXT</messagetype>
<messagedata>Hello world</messagedata>
<senttime>2008-01-16 10:04:00</senttime>
<receivedtime>2008-01-18 20:58:04</receivedtime>
</message>
<message>
```

```
<messageid>GYT4455D</messageid>
<originator>06209994444</originator>
<recipient>06201112222</recipient>
<messagetype>SMS:TEXT</messagetype>
<messagedata>Second message</messagedata>
<senttime>2008-01-16 10:08:00</senttime>
<receivedtime>2008-01-18 20:58:05</receivedtime>
</message>
</data>
</response>
```

# Request parameters

| Parameter | Description | Possible values | Example | M/O* |
|---|---|---|---|---|
| action | Specifies the HTTP API command | receivemessage | action=receivemessage | M |
| username | Specifies the username. The username and password parameters are used to authenticate the user. Once the user is authenticated messages will be downloaded from his inbox. The value must be urlencoced. | string value, maximum length is 16 characters | username=admin | M |
| password | Specifies the password. The username and password parameters are used to authenticate the user. Once the user is authenticated messages will be downloaded from his inbox. The value must be urlencoced. | string value, maximum length is 16 characters | password=abc123 | M |
| folder | Specifies the name of the folder. By default messages will be downloaded from the inbox folder. You can use the receivemessage action to download messages from other folders as well. | inbox (default)<br>outbox<br>deliveredtonetwork<br>deliveredtohandset<br>deliveryfailed | folder=inbox | O |
| limit | Specifies the maximum number of messages to be downloded from the server within the request. You can download the remaining messages with subsequent requests. | integer value, the default limit is 1000 | limit=10 | O |
| afterdownload | Specifies the way messages should be handled after a successful download. Messages can be deleted from the SMS gateway, they can be marked as downloaded and they can be left untouched. If they are left untouched, the next time you initiate a download request they will be downloaded again. | delete (default)<br>mark<br>untouch | afterdownload=delete | O |
| responseformat | You can specify the format of the user's incoming messages. | xml, html, urlencoded | responseformat=xml | O |

* M = Mandatory parameter, O = Optional parameter

# Response parameters

**(xml response format)**

| Parameter | Description | Possible values | Example |
|---|---|---|---|
| message | Contains a single message. The data section can contain many messages. The number of messages in the message data can is less then or equal to the value of the limit parameter | | <message><br>    <messageid>ERFAV23D</messageid><br>    <originator>06301234567</originator><br>    <recipient>06201112222</recipient><br>    <messagetype>SMS:TEXT</messagetype><br>    <messagedata>Hello world</messagedata><br>    <senttime>2028-01-16 10:04:00</senttime><br>    <receivedtime>2028-01-18 20:58:04</receivedtime><br></message> |
| message.messageid | Contains a message reference that can be used to track the message in the SMS gateway. This message reference is also used to identify delivered to network and delivered | string value, maximum length is 16 characters | <messageid>ERFAV23D</messageid> |

| | | | |
|---|---|---|---|
| | to handset reports or to query information about the message. | | |
| message.originator | Contains the sender telephone number. This is the telephone number of the cellphone the sent the message. | string value, maximum length is 16 characters | <recipient>06301234567</recipient> |
| message.recipient | Contains the recipient address. This is the telephone number that was used to send the message to. | string value, maximum length is 16 characters | <recipient>06201112222</recipient> |
| message.messagetype | Contains the message type identifier specified in the Mobile Message Type Specification. In most cases this will be SMS:TEXT. | string value, maximum length is 1024 characters | <recipient>SMS:TEXT</recipient> |
| message.messagedata | Contains the message data. For text messages this is the text of the SMS messages. For other message types this is an XML structure specified in the Mobile Message Type Specification. The messagedata is encoded as UTF8. | string value | <messagedata>Hello world</messagedata> |
| message.senttime | Contains the timestamp that tells when the message was submitted from the sender phone. | Date time value in the following format: YYYY-MM-DD hh:mm:ss | <senttime>2028-01-16 10:04:00</senttime> |
| message.receivedtime | Contains the timestamp that tells when the message was received by the SMS gateway. | Date time value in the following format: YYYY-MM-DD hh:mm:ss | <senttime>2028-01-18 20:58:04</senttime> |

# HTTP API - URL Encoding

In order to pass parameters in URL-s, you have to replace certain characters to keep the syntax. This is called URL escaping or URL encoding. For example the space must be replace to the + sign. The following table lists some of the characters that should be replaced.

| | |
|---|---|
| space | + |
| ! | %21 |
| " | %22 |
| # | %23 |
| % | %25 |
| & | %26 |
| ' | %27 |
| * | %2A |
| + | %2B |
| , | %2C |
| / | %2F |
| : | %3A |
| < | %3C |
| = | %3D |
| > | %3E |
| ? | %3F |
| Euro (€) | %E2%82%AC |

## On-line URL encoding

If you would like to use UTF8 URL encoding, please use the following tool:

UTF8 URL encode (for Ozeki NG)

To use Windows 1250 URL encoding, please use the following tool:

Windows 1250 URL encode

# Using the HTTP API

***The HTTP API of the SMS Gateway allows you to send SMS messages to mobile phones from programming languages. The following list contains programming languages for which we provide example code. Open the programming language you are interested in and send your first SMS in a matter of minutes.***

**Send SMS form the browser**
Learn about how you can use your browser to sens SMS messages. With the browser, you are capable of composing an API command. This command will be processed by SMS Gateway and it sends the SMS message that you set up in the API command.
Learn about how to send SMS from your browser

**Send SMS form C#**
C# is a widely used programming language, mostly to develop desktop applications, web applications or web services. The following guide shows how you can build your own console application which allows you to send SMS messages to SMS Gateway by using HTTP requests.
View the C# SMS example code

**Send SMS form Java**
Java is a general-purpose programming language that is class-based and object-oriented. The program written in Java works on differenrt platforms like Windows, Max, Linux, Raspberry Pi, etc. This is a superb reason to see and learn how you can create a Java program that can send SMS messages with HTTP requests over Ozeki SMS Gateway.
View the Java SMS example code

**Send SMS form PHP**
PHP is a general-purpose scripting language that is especially suited to web development. The code written in PHP is usually processed on a web server. This simple PHP example demonstrates how you can send SMS from PHP using HTTP requests.
View the PHP SMS example code

**Send SMS form Perl**
Perl is a general-purpose programming language originally developed for text manipulation and now used for a wide range of tasks including system administration, web development, network programming. This guide contains a simple example written in Perl which shows the way how you can send SMS messages.
View the Perl SMS example code

**Send SMS form Python**
Python is a general purpose and high level programming language. You can use Python for developing desktop GUI applications, websites and web applications. The following Python program example demonstrate how you can establish HTTP requests and send SMS messages using them.
View the Python SMS example code

**Send SMS form Ruby**
Ruby is a dynamic, open source programming language with a focus on simplicity and productivity. Let's see an example which is capable of using HTTP requests to send SMS message via Ozeki SMS Gateway.
View the Ruby SMS example code

# How to send an SMS from the Browser

***After reading this lecture you will be able to send SMS messages to pre-setup HTTP API users using Ozeki 10 sms gateway. You will see how to compose the API command and how it works. Lastly you will see what message the browser returns if the request was successful and the error message.***

## Prerequisites

1. Ozeki 10 sms gateway downloaded and installed
2. A HTTP API user setup on Ozeki 10 sms gateway

## Steps to send an SMS in the browser

1. Open a browser window
2. Click into the URL textbox
3. Type in the sms command template
4. Find out the IP address of the sms gateway
5. Find out the port number of the HTTP API user
6. Refit the command to your likes
7. Hit enter to summit the sms.
8. Check the response

## Open a browser

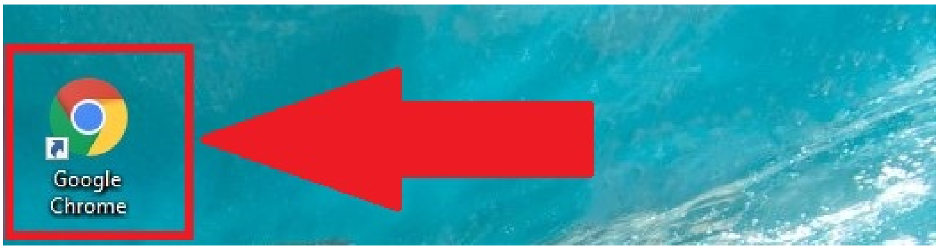


Figure 1 - Click on the browser icon

## Click in the URL textbox

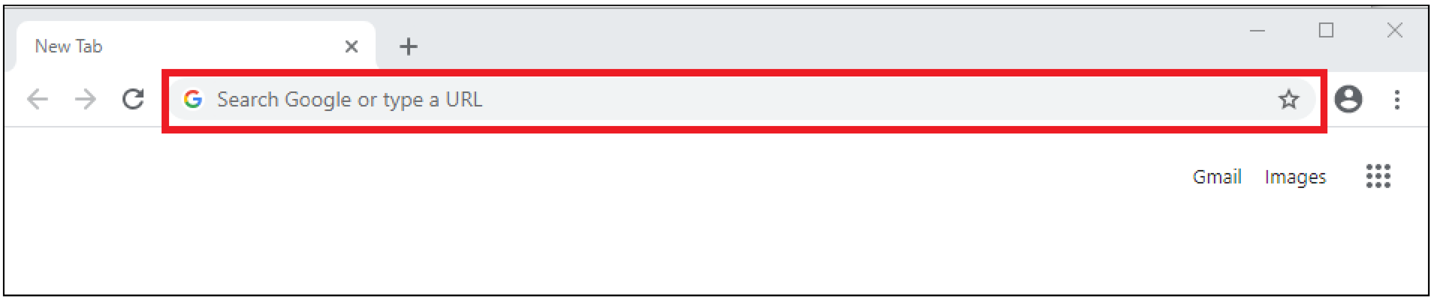
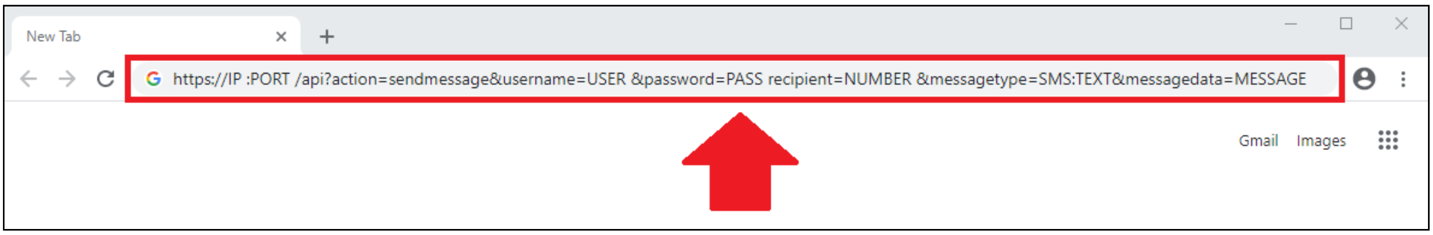


Figure 2 - URL textbox

## Type in the command template

Command template:
https://IP :PORT /api?action=sendmessage&username=USER &password=PASS recipient=NUMBER &messagetype=SMS:TEXT&messagedata=MESSAGE



https://IP :PORT /api?action=sendmessage&username=USER &password=PASS recipient=NUMBER &messagetype=SMS:TEXT&messagedata=MESSAGE

Figure 3 - URL textbox filled in

## Check your IP address.

**1)Press the win key and start typing cmd.**
**2)Click on the command shell icon.**
**3)Type ipconfig**

Figure 4 - Command shell with ipconfig

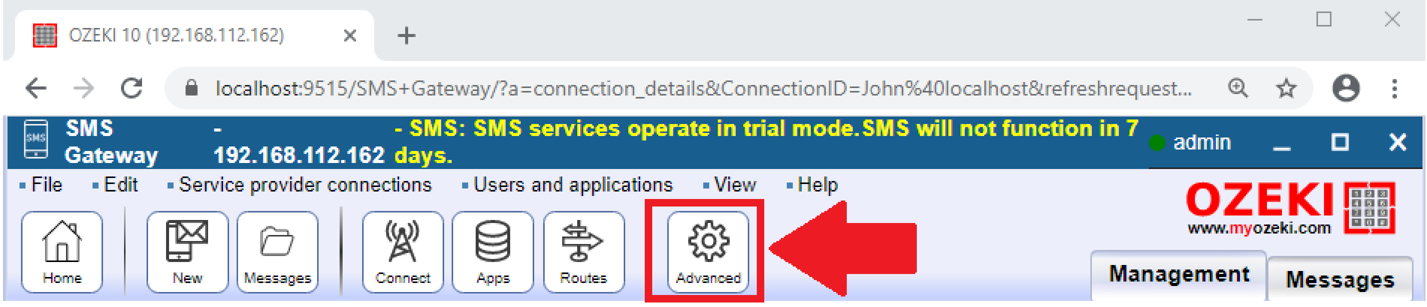
**4)Hit enter**
**5)If you use ethernet then go to the line with IPv4 Address and there you will find your IP. If you use wifi then go to the line with IPv4 Address and there you will find your IP.**

## Replace the API commands parameters to your case.

**Replace the IP highlighted in red to your computers IP address**

https://IP:PORT/api?action=sendmessage&username=USER&password=PASS
recipient=NUMBER&messagetype=SMS:TEXT&messagedata=MESSAGE

**Replace the PORT highlighted in orange to your HTTP API users port number**
1)Click on the advanced button on the top toolbar.

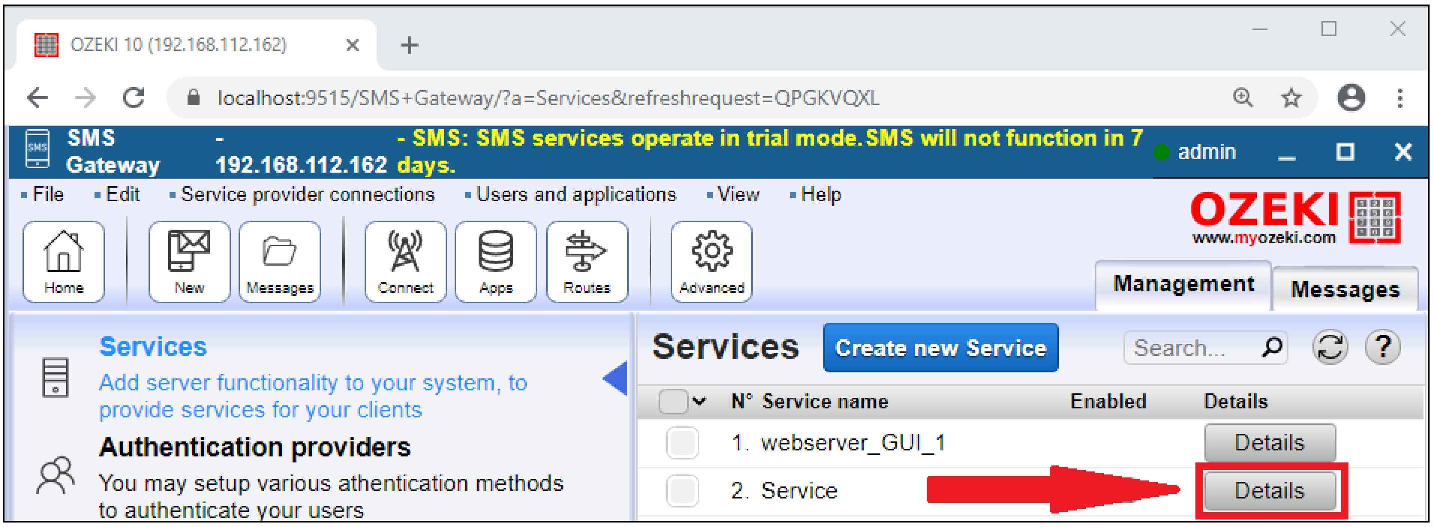


localhost:9515/SMS+Gateway/?a=connection_details&ConnectionID=John%40localhost&refreshrequest...

OZEKI 10 (192.168.112.162)
SMS Gateway - 192.168.112.162
- SMS: SMS services operate in trial mode.SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
OZEKI
www.myozeki.com
Management
Messages

Figure 1 - advanced button

2)Then click on the details button button.



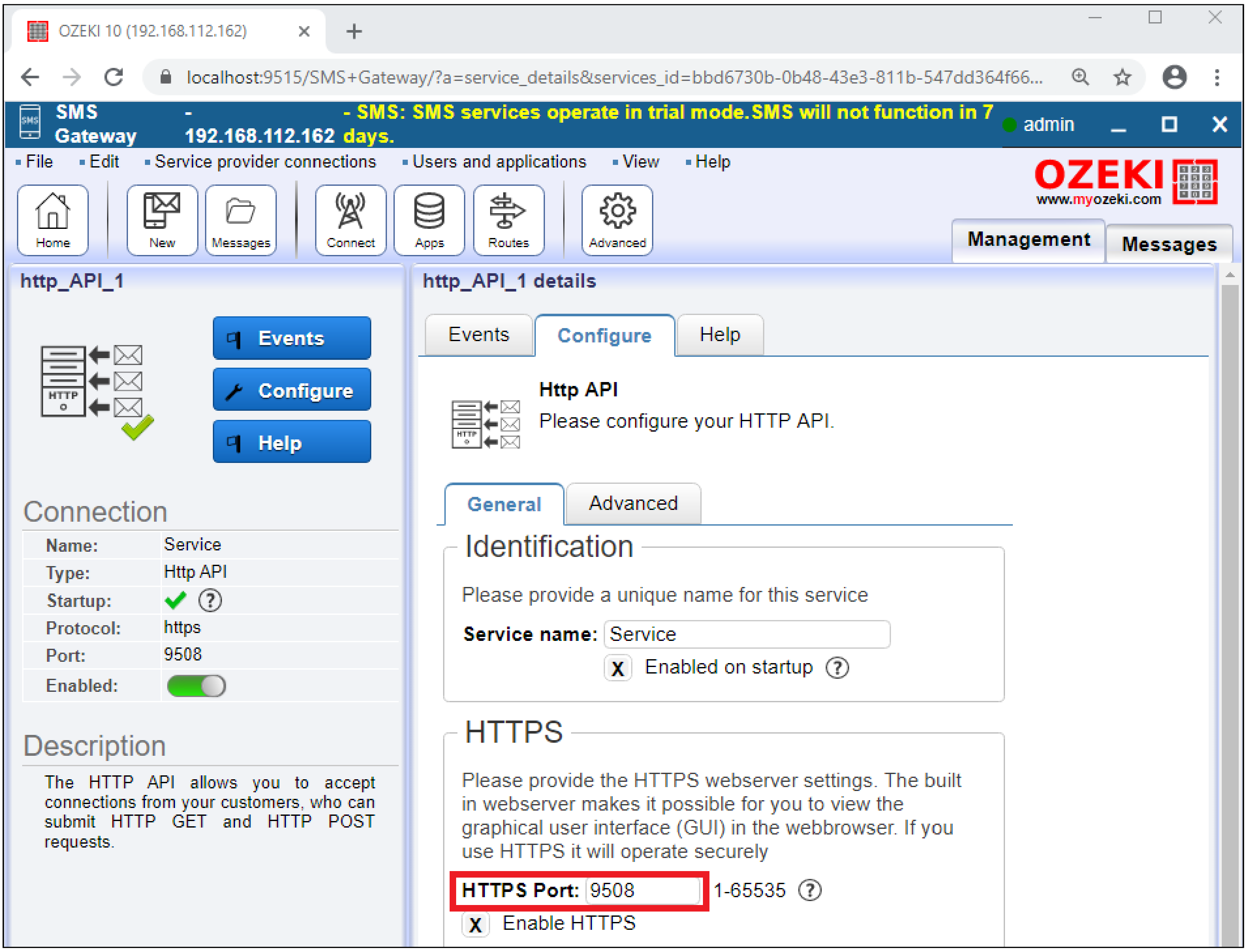
localhost:9515/SMS+Gateway/?a=Services&refreshrequest=QPGKVQXL

| N° | Service name | Enabled | Details |
|---|---|---|---|
| 1. | webserver_GUI_1 | | Details |
| 2. | Service | | Details |

OZEKI 10 (192.168.112.162)
SMS Gateway - 192.168.112.162
- SMS: SMS services operate in trial mode.SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
OZEKI
www.myozeki.com
Management
Messages
Services
Add server functionality to your system, to provide services for your clients
Authentication providers
You may setup various athentication methods to authenticate your users
Services
Create new Service
Search...

Figure 3 - details button

3)Finally in the HTTPS PORT input field you will find your port number.


OZEKI 10 (192.168.112.162)
localhost:9515/SMS+Gateway/?a=service_details&services_id=bbd6730b-0b48-43e3-811b-547dd364f66...
SMS Gateway - 192.168.112.162
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
OZEKI
www.myozeki.com
Management
Messages
Events
Configure
Help
Events
Configure
Help
General
Advanced

- SMS: SMS services operate in trial mode.SMS will not function in 7 days.

http_API_1

Connection

| | |
|---|---|
| Name: | Service |
| Type: | Http API |
| Startup: | ✓ ? |
| Protocol: | https |
| Port: | 9508 |
| Enabled: | ● |

Description

The HTTP API allows you to accept connections from your customers, who can submit HTTP GET and HTTP POST requests.

http_API_1 details

Http API

Please configure your HTTP API.

Identification

Please provide a unique name for this service

Service name: Service

☒ Enabled on startup ?

HTTPS

Please provide the HTTPS webserver settings. The built in webserver makes it possible for you to view the graphical user interface (GUI) in the webbrowser. If you use HTTPS it will operate securely

HTTPS Port: 9508 1-65535 ?

☒ Enable HTTPS

Figure 4 - HTTPS PORT input field

https://IP:PORT/api?action=sendmessage&username=USER&password=PASS
recipient=NUMBER&messagetype=SMS:TEXT&messagedata=MESSAGE

**Replace the USER highlighted in green to your HTTP API users name**

https://IP:PORT/api?action=sendmessage&username=USER&password=PASS
recipient=NUMBER&messagetype=SMS:TEXT&messagedata=MESSAGE

**Replace the PASS highlighted in purple to your HTTP API users password**

https://IP:PORT/api?action=sendmessage&username=USER&password=PASS
recipient=NUMBER&messagetype=SMS:TEXT&messagedata=MESSAGE

**Replace the NUMBER highlighted in blue to the phone's phone number you want to send the SMS to.**

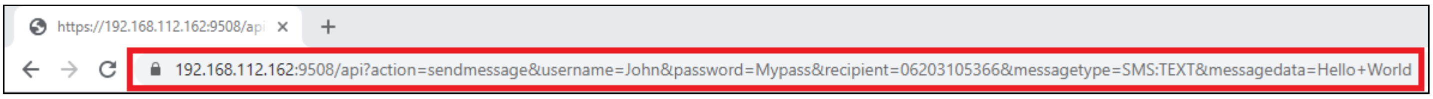
https://IP:PORT/api?action=sendmessage&username=USER&password=PASS
recipient=NUMBER&messagetype=SMS:TEXT&messagedata=MESSAGE

**Finally Replace the MESSAGE highlighted in turquoise to a text you want to send but substitute all spaces with + signs**

https://IP:PORT/api?action=sendmessage&username=USER&password=PASS
recipient=NUMBER&messagetype=SMS:TEXT&messagedata=MESSAGE

# Finish off.

**Now that you have fully filled in the parameters this is how your URL textbox should look.**



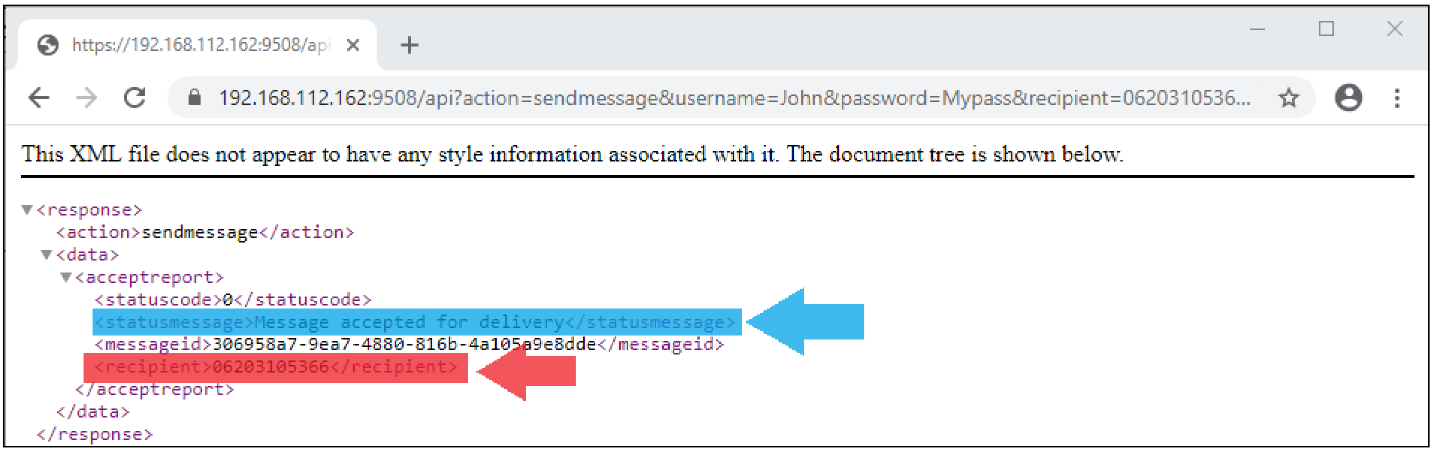
```
192.168.112.162:9508/api?action=sendmessage&username=John&password=Mypass&recipient=06203105366&messagetype=SMS:TEXT&messagedata=Hello+World
```

Figure 5 - URL textbox

Now press enter.

# Outcome.

This is how your browser should look like now.



```
192.168.112.162:9508/api?action=sendmessage&username=John&password=Mypass&recipient=0620310536...
```

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
▼<response>
   <action>sendmessage</action>
 ▼<data>
   ▼<acceptreport>
      <statuscode>0</statuscode>
      <statusmessage>Message accepted for delivery</statusmessage>
      <messageid>306958a7-9ea7-4880-816b-4a105a9e8dde</messageid>
      <recipient>06203105366</recipient>
    </acceptreport>
  </data>
</response>
```

Figure 6 - Browser

The line highlighted in blue tells you what happened to the message.
The line highlighted in red tells you who have you sent it to.

Message should also arrive at the phone.

# Error message.

Another scenario is when you did not fill in the parameters as required.
Then this is how browser should appear.

Figure 7 - Browser

The line in Red is what tells you what went wrong.

# Now lets look into how the command works.

**The first key part is when we write api:**
https://IP:PORT/api?action=sendmessage&username=USER&password=PASS
recipient=NUMBER&messagetype=SMS:TEXT&messagedata=MESSAGE
**Description:**
This is when we address the API.

**After this we addressed some values where the first one was the *action*:**
https://IP:PORT/api?action=sendmessage&username=USER&password=PASS
recipient=NUMBER&messagetype=SMS:TEXT&messagedata=MESSAGE
**Description:**
The ***action*** value describes the action we want to make but at the current moment has only one value which is sendmessage.

**The next value was the *username*:**
https://IP:PORT/api?action=sendmessage&username=USER&password=PASS
recipient=NUMBER&messagetype=SMS:TEXT&messagedata=MESSAGE
**Description:**

The ***username*** value tells what user to use to send the message and needs to be URL encoded. The max length is 16 characters.

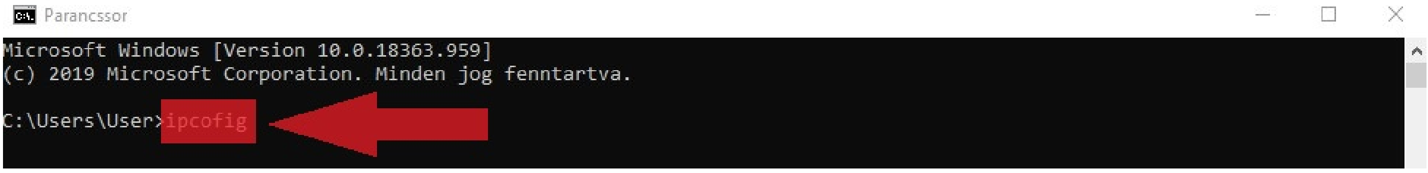
**After that came the *password*:**
https://IP:PORT/api?action=sendmessage&username=USER&password=PASS
recipient=NUMBER&messagetype=SMS:TEXT&messagedata=MESSAGE
**Description:**
The ***password*** value tells what password to use when using the user and needs to be URL encoded. The max length is 16 characters.

**The next was the *recipient*:**
https://IP:PORT/api?action=sendmessage&username=USER&password=PASS
recipient=NUMBER&messagetype=SMS:TEXT&messagedata=MESSAGE
**Description:**
The ***recipient*** value describes the name of the device we are sending to and needs to be URL encoded. The max length is 16 characters.

**The next value was the *messagetype*:**
https://IP:PORT/api?action=sendmessage&username=USER&password=PASS
recipient=NUMBER&messagetype=SMS:TEXT&messagedata=MESSAGE
**Description:**
The ***messagetype*** contains the type of message we want to send and needs to be URL encoded. To see the possible values click here.

**The next value was the *messagedata*:**
. https://IP:PORT/api?action=sendmessage&username=USER&password=PASS
recipient=NUMBER&messagetype=SMS:TEXT&messagedata=MESSAGE
**Description:**
The ***messagedata***this contains the data we need to send, must be encoded in UTF8 and needs to be URL encoded. the max length is 32768.

## See more properties we can add to our SMS API command and get a more in depth explanation.



```
Microsoft Windows [Version 10.0.18363.959]
(c) 2019 Microsoft Corporation. Minden jog fenntartva.

C:\Users\User>ipcofig
```

Figure 1 -

# How to send SMS from C#

***You can easily build a C# console application with that allows you to send a HTTP request to the Ozeki 10 SMS gateway. When you run this attached application, first, it will print a HTTP request on the console and after that send it out. If everything goes well the Ozeki 10 SMS gateway receives this request, and sends back a response. Our C# application receives this response and displaying on the console. From this response we can find out if the delivery was successful.***

## What is a C# SMS API?

The C# SMS API is a great tool to be able to send SMS message from any kind of C# based project or application by initiating HTTP requests and forwarding them to the SMS Gateway.

## Prerequisites

- Installed Visual Studio 2019 Community Edition
- Installed Ozeki 10 SMS Gateway
- A configured HTTP API User

## Send SMS from C#

1. Open https://localhost:9515 in your browser and log in
2. Open the SMS Gateway application
3. Create a new HTTP API User connection
4. Check the port number of the HTTP API service
5. Start Visual Studio
6. Create a new Console App project
7. Copy-Paste the example source code from this page to Program.cs
8. Run the Console App project

## C# SMS source code example

The following example C# source code is free to use, you can simply implement it into your project or you can modify the source code to use it for other projects or applications. If you would like to run this example code, you just need to copy-paste it into your Console App project and run the project.

```
using System;
using System.Net.Http;
using System.Text;
using System.Web;
namespace HttpApiTester
{
    class Program
    {
        static async System.Threading.Tasks.Task Main(string[] args)
        {
            var username = "john";
            var password = "Xc3ffs";
            var messagetype = "SMS:TEXT";
            var httpUrl = "https://127.0.0.1:9508/";
            var recipient = HttpUtility.UrlEncode("+36201324567", Encoding.UTF8);
            var messagedata = HttpUtility.UrlEncode("TestMessage", Encoding.UTF8);

            var sendString = $"{httpUrl}api?action=sendmessage&username=" +
                             $"{username}&password={password}" +
                             $"&recipient={recipient}&messagetype=" +
                             $"{messagetype}&messagedata={messagedata}";

            Console.WriteLine("Sending request: " + sendString);

            var handler = new HttpClientHandler();
            handler.ServerCertificateCustomValidationCallback =
                (sender, cert, chain, sslPolicyErrors) => { return true; };

            using var client = new HttpClient(handler);

            try
            {
                var response = await client.GetStringAsync(sendString);
                Console.WriteLine("Http response received: ");
                Console.WriteLine(response);

            } catch(Exception e)
            {
                Console.WriteLine(e.Message);
            }
        }
    }
}
```

## Step 1 - Open Visual Studio

The first step to create the application that can send SMS message is to open the Visual Studio. If you haven't downloaded Visual Studio yet, just follow the link in the Prerequisites section above to download the latest version. If you have the installed Visual Studio on your computer, all you need to do is to click on its icon (Figure 1) on the desktop to open Visual Studio.

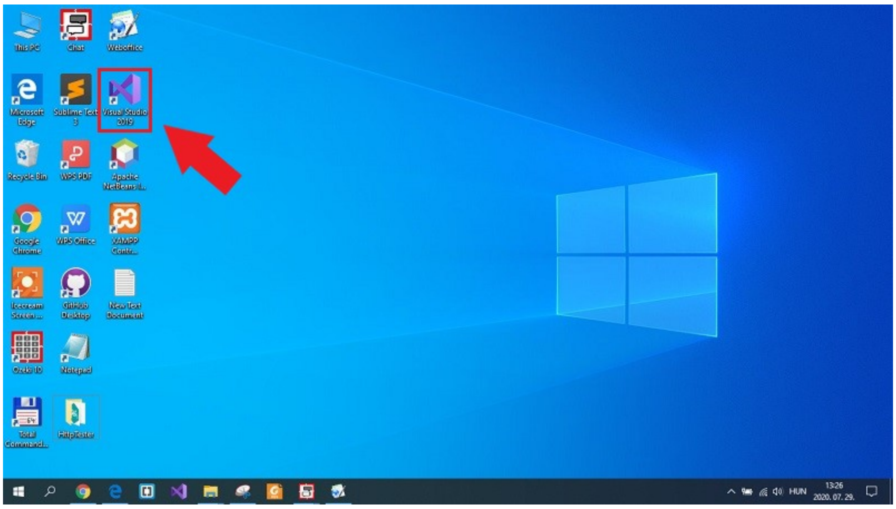


Figure 1 - Open Visual Studio 2019

## Step 2 - Create a new project

After you opened Visual Studio, the opening window will show up for you as you can see it in Figure 2. Here you can see your solutions, you can open an existing one, clone a repository or open a local folder. Now, to follow the guide, you need to click on the 'Create a new project' button to create the project for SMS sending.

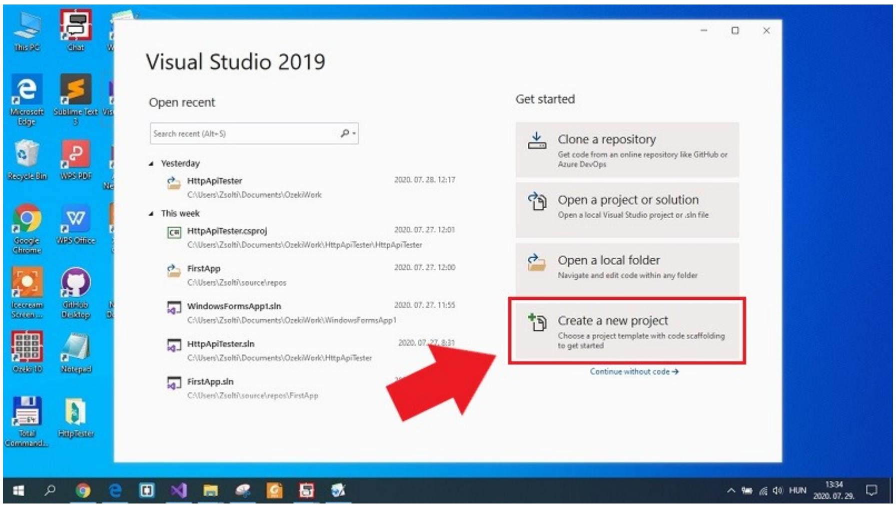
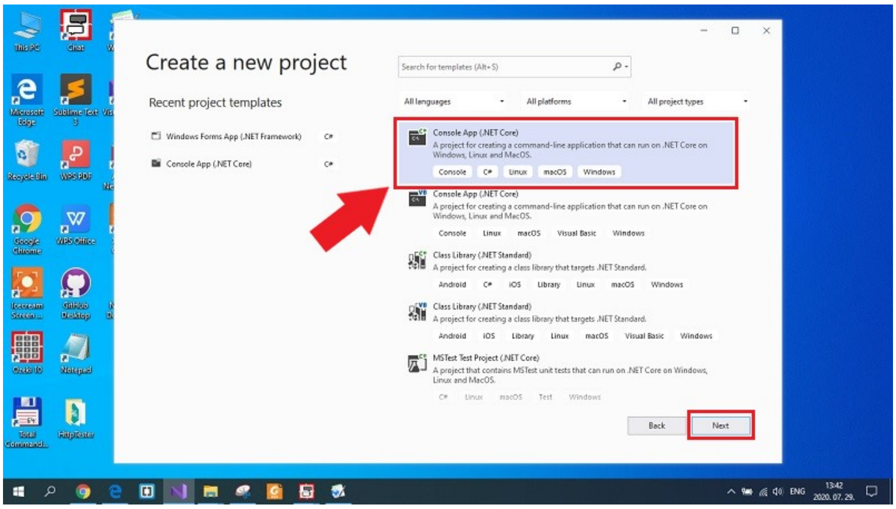

Visual Studio 2019
Open recent
Search recent (Alt+S)
Yesterday
HttpApiTester
C:\Users\Zsolti\Documents\OzekiWork
2020. 07. 28. 12:17
This week
HttpApiTester.csproj
C:\Users\Zsolti\Documents\OzekiWork\HttpApiTester\HttpApiTester
2020. 07. 27. 12:01
FirstApp
C:\Users\Zsolti\source\repos
2020. 07. 27. 12:00
WindowsFormsApp1.sln
C:\Users\Zsolti\Documents\OzekiWork\WindowsFormsApp1
2020. 07. 27. 11:55
HttpApiTester.sln
C:\Users\Zsolti\Documents\OzekiWork\HttpApiTester
2020. 07. 27. 8:31
FirstApp.sln
C:\Users\Zsolti\source\repos\FirstApp
Get started
Clone a repository
Get code from an online repository like GitHub or Azure DevOps
Open a project or solution
Open a local Visual Studio project or .sln file
Open a local folder
Navigate and edit code within any folder
Create a new project
Choose a project template with code scaffolding to get started
Continue without code →
HUN
13:34
2020. 07. 29.

Figure 2 - Creating new project

# Step 3 - Select Console App

The next window lists all the available types of projects that can be created in Visual Studio. This example requires you to create a simple Console App as Figure 3 demonstrates it, but you can use the SMS sending service in many other types of project. So, all you have to do here is to click on the Console App option from the list of avalable types of projects.



Create a new project

Recent project templates

Windows Forms App (.NET Framework) C#

Console App (.NET Core) C#

Console App (.NET Core)
A project for creating a command-line application that can run on .NET Core on Windows, Linux and MacOS.
Console C# Linux macOS Windows

Console App (.NET Core)
A project for creating a command-line application that can run on .NET Core on Windows, Linux and MacOS.
Console Linux macOS Visual Basic Windows

Class Library (.NET Standard)
A project for creating a class library that targets .NET Standard.
Android C# iOS Library Linux macOS Windows

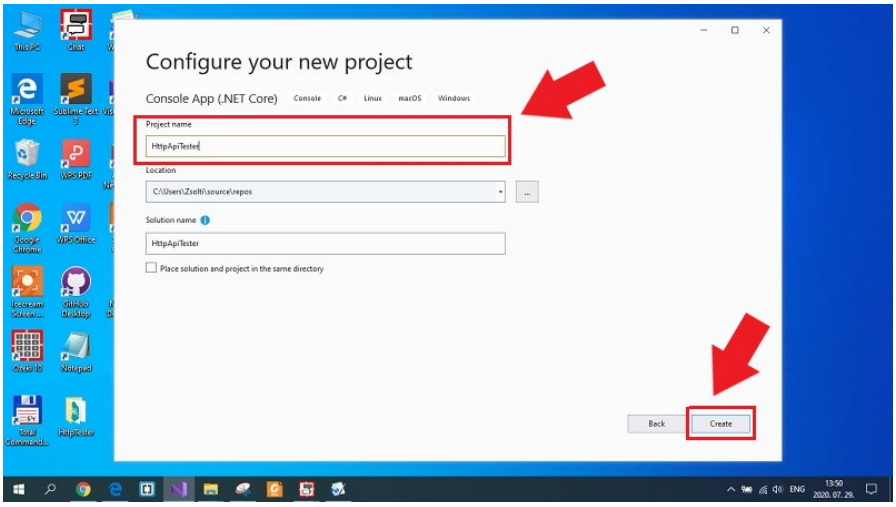
Class Library (.NET Standard)
A project for creating a class library that targets .NET Standard.
Android iOS Library Linux macOS Visual Basic Windows

MSTest Test Project (.NET Core)
A project that contains MSTest unit tests that can run on .NET Core on Windows, Linux and MacOS.
C# Linux macOS Test Windows

Figure 3 - Choosing Console Application

# Step 4 - Configure the project

Before crating the project, the final step is to configure it by specifying some details of the project. Here, you can give a name to the project and also set the location folder as well as you can see it in Figure 4. If you finished with the configuration, you can just click on 'Create' to create the Console App project.



Configure your new project

Console App (.NET Core) Console C# Linux macOS Windows

Project name: HttpApiTester

Location: C:\Users\Zsolti\source\repos

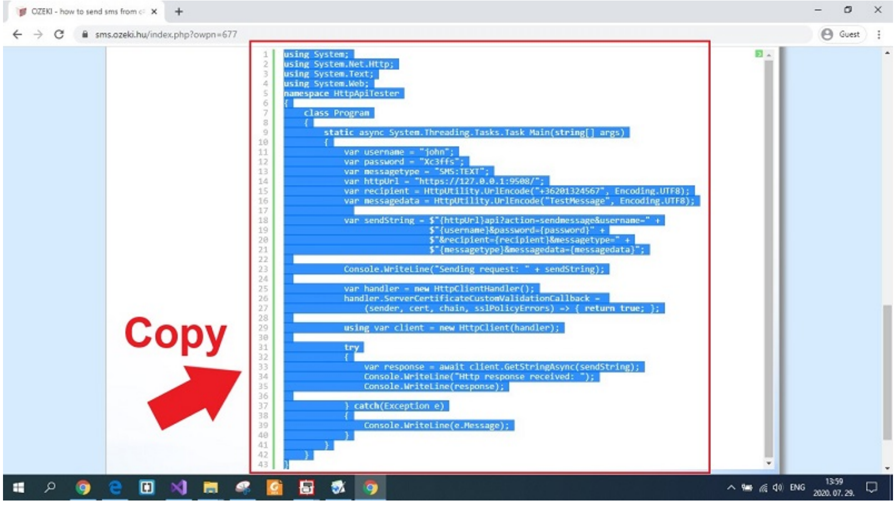
Solution name: HttpApiTester

□ Place solution and project in the same directory

Figure 4 - Setup Visual Studio project

# Step 5 - Copy the example code

After you created the Console App project, nowe you need to set up the example code that you need to execute. For that, scroll up on this page to find the example code section and mark out the whole source code as Figure 5 shows that. Then just press Ctrl+C on your keyboard to copy the source code.

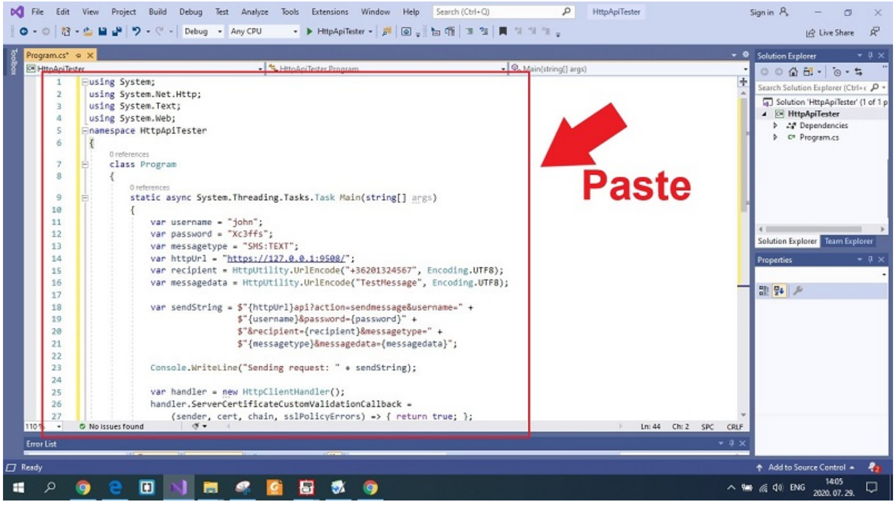

OZEKI - how to send sms from c
sms.ozeki.hu/index.php?owpn=677
Copy

```
using System;
using System.Net.Http;
using System.Text;
using System.Web;
namespace HttpApiTester
{
    class Program
    {
        static async System.Threading.Tasks.Task Main(string[] args)
        {
            var username = "john";
            var password = "Xc3ffs";
            var messagetype = "SMS:TEXT";
            var httpUrl = "https://127.0.0.1:9508/";
            var recipient = HttpUtility.UrlEncode("+36201324567", Encoding.UTF8);
            var messagedata = HttpUtility.UrlEncode("TestMessage", Encoding.UTF8);

            var sendString = $"{httpUrl}api?action=sendmessage&username=" +
                             $"{username}&password={password}" +
                             $"&recipient={recipient}&messagetype=" +
                             $"{messagetype}&messagedata={messagedata}";

            Console.WriteLine("Sending request: " + sendString);

            var handler = new HttpClientHandler();
            handler.ServerCertificateCustomValidationCallback =
                (sender, cert, chain, sslPolicyErrors) => { return true; };

            using var client = new HttpClient(handler);

            try
            {
                var response = await client.GetStringAsync(sendString);
                Console.WriteLine("Http response received: ");
                Console.WriteLine(response);

            } catch(Exception e)
            {
                Console.WriteLine(e.Message);
            }
        }
    }
}
```

Figure 5 - Copy code from website

## Step 6 - Paste the source code into your project

In the Visual Studio, the created project contains an initial Program.cs source file. At this point, this is the file that you have to use to execute the example program. Here, first, mark out the code in that file and delete it. After that, you Figure 6 demonstrates it, press Ctrl+V on your keyboard to paste the example code into your Program.cs file. Now, the project is ready to use.

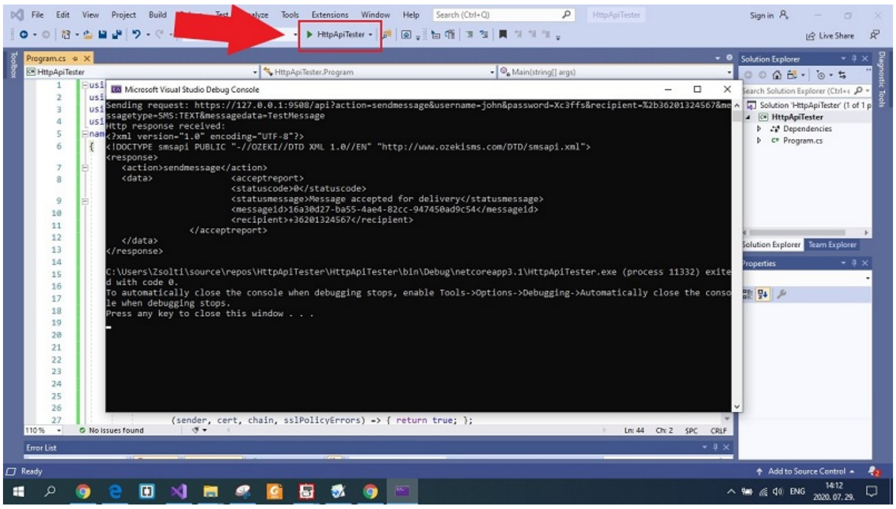


```
using System;
using System.Net.Http;
using System.Text;
using System.Web;
namespace HttpApiTester
{
    0 references
    class Program
    {
        0 references
        static async System.Threading.Tasks.Task Main(string[] args)
        {
            var username = "john";
            var password = "Xc3ffs";
            var messagetype = "SMS:TEXT";
            var httpUrl = "https://127.0.0.1:9508/";
            var recipient = HttpUtility.UrlEncode("+36201324567", Encoding.UTF8);
            var messagedata = HttpUtility.UrlEncode("TestMessage", Encoding.UTF8);

            var sendString = $"{httpUrl}api?action=sendmessage&username=" +
                             $"{username}&password={password}" +
                             $"&recipient={recipient}&messagetype=" +
                             $"{messagetype}&messagedata={messagedata}";

            Console.WriteLine("Sending request: " + sendString);

            var handler = new HttpClientHandler();
            handler.ServerCertificateCustomValidationCallback =
                (sender, cert, chain, sslPolicyErrors) => { return true; };
```

Figure 6 - Paste code into Visual Studio

## Step 7 - Execute the example project

The final step of this guide is to execute the example program. In Visual Studio, it is quite simple to run the project, you just need to click on the Run button as you can see it in Figure 7. After starting the program, it shows up a console window, that prints the HTTP request that the program sent to the SMS Gateway. It also prints the response from the SMS Gateway that shows if the delivery of the SMS was successful.

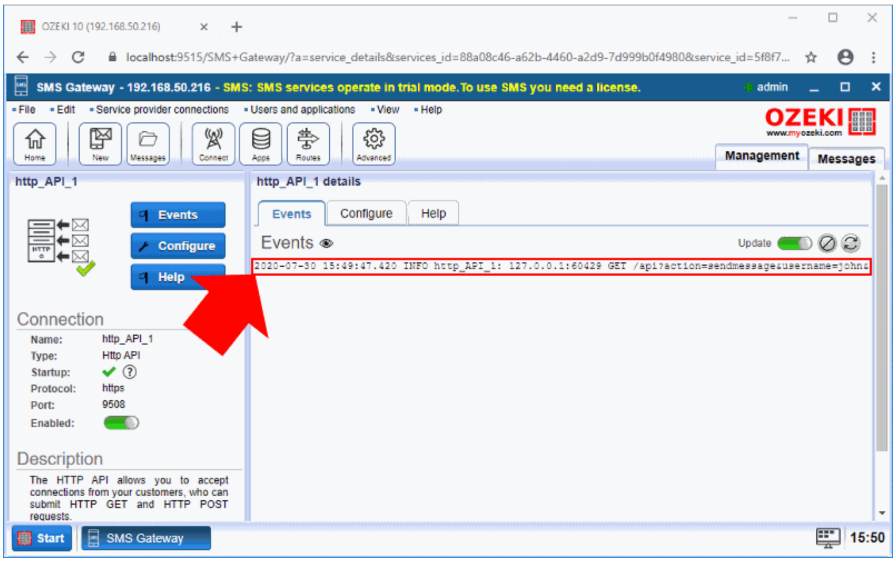


```
Sending request: https://127.0.0.1:9508/api?action=sendmessage&username=john&password=Xc3ffs&recipient=%2b36201324567&me
ssagetype=SMS:TEXT&messagedata=TestMessage
Http response received:
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE smsapi PUBLIC "-//OZEKI//DTD XML 1.0//EN" "http://www.ozekisms.com/DTD/smsapi.xml">
<response>
   <action>sendmessage</action>
   <data>                    <acceptreport>
                             <statuscode>0</statuscode>
                             <statusmessage>Message accepted for delivery</statusmessage>
                             <messageid>16a30d27-ba55-4ae4-82cc-947450ad9c54</messageid>
                             <recipient>+36201324567</recipient>
                     </acceptreport>
   </data>
</response>

C:\Users\Zsolti\source\repos\HttpApiTester\HttpApiTester\bin\Debug\netcoreapp3.1\HttpApiTester.exe (process 11332) exite
d with code 0.
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the conso
le when debugging stops.
Press any key to close this window . . .
```

Figure 7 - Build and run your code in Visual Studio

# Step 8 - Check the send result in the Ozeki log

In Ozeki 10 SMS Gateway, you can check easily what messages sent by your application, since the HTTP API service logs every event that occured during the time it is enabled. So, if you open the SMS Gateway, and select details of the HTTP API service, you will be able to see the events. As you can see it in Figure 8, the service logged an event, when the C# example program sent the HTTP request to the service.



SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode. To use SMS you need a license.
http_API_1
Events
Configure
Help
http_API_1 details
Events
Update

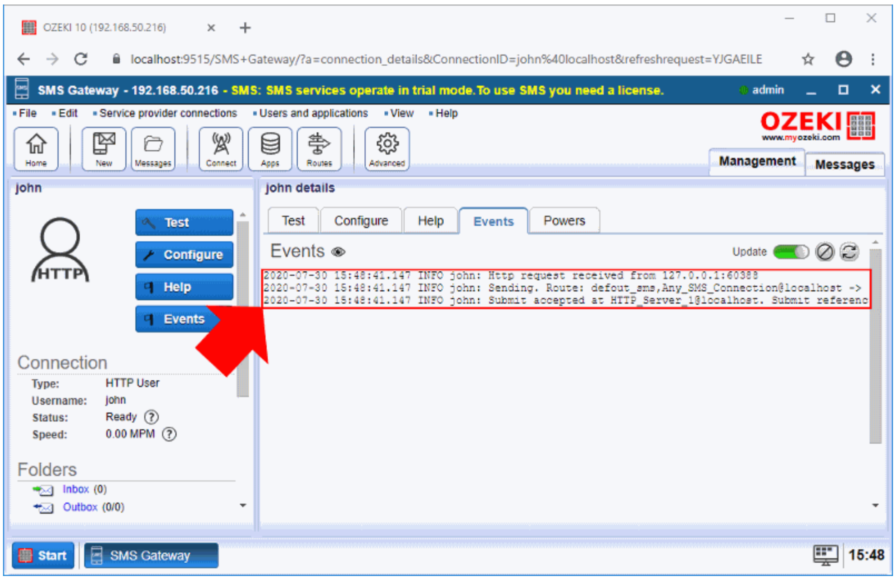
Connection
Name: http_API_1
Type: Http API
Startup: ✓
Protocol: https
Port: 9508
Enabled:

Description
The HTTP API allows you to accept connections from your customers, who can submit HTTP GET and HTTP POST requests.

```
2020-07-30 15:49:47.420 INFO http_API_1: 127.0.0.1:60429 GET /api?action=sendmessage&username=john&
```

Figure 8 - Check the logs of the HTTP API Service

The process of the message sending can be also viewed back by the events. For that, open the HTTP API User connection, that you had to configure before. Figure 9 demonstrates that how the connection handles the HTTP request and send the message to the recipient that you specified in your C# program.



Type: HTTP User
Username: john
Status: Ready
Speed: 0.00 MPM

```
2020-07-30 15:48:41.147 INFO john: Http request received from 127.0.0.1:60388
2020-07-30 15:48:41.147 INFO john: Sending. Route: default_sms,Any_SMS_Connection@localhost ->
2020-07-30 15:48:41.147 INFO john: Submit accepted at HTTP_Server_1@localhost. Submit referenc
```

Figure 9 - Check the logs of the HTTP API User

# Simulate incoming messages

# Receive SMS in C#

***The following guide is going to demonstrate how you can get the messages received by your SMS Gateway with a C# application. This C# example uses HTTP requests to ask for the messages from your inbox folder in SMS Gateway. The SMS Gateway collects all these received messages and sends them to the C# application as a response to the HTTP request. The guide also shows, how you can use the HTTP server connection to simulate incoming messages. It does take not more than five minutes to complete this guide, so let's start right now.***

## What is a C# SMS API?

The C# SMS API is a great tool to be able to send SMS message from any kind of C# based project or application by initiating HTTP requests and forwarding them to the SMS Gateway.

## Prerequisites

- Installed Visual Studio 2019 Community Edition
- Installed Ozeki 10 SMS Gateway
- A configured HTTP Server connection

## Receive SMS in C#

1. Open Visual Studio
2. Create a new Console App project
3. Copy the C# example code from this page
4. Paste the code into your C# application
5. Open Ozeki 10 SMS Gateway
6. Select the HTTP Server connection
7. Send test messages using the HTTP Server connection
8. Run your C# application to get all incoming messages

The example C# code below is capable of collecting all incoming messages using HTTP requests. This example code is free to use, you can modify and implement it into your own project application. If you wish to just test the solution, you need to follow the step by step instructions below or watch the video above which demonstrates how you can use this example C# code.

```
using System;
using System.Net.Http;
using System.Xml;
using System.Threading.Tasks;

namespace ReceiveSMS
{
    class Program
    {
        static async Task Main(string[] args)
        {
            var username = "john";
            var password = "Xc3ffs";
            var httpUrl = "https://127.0.0.1:9508/";
            var folder = "inbox";
            var limit = "3";

            var sendString = $"{httpUrl}api?action=receivemessage&username=" +
                 $"{username}&password={password}&folder={folder}&limit=" +
                 $"{limit}&afterdownload=delete";

            Console.WriteLine("Sending request: " + sendString + "\n");

            var handler = new HttpClientHandler();
            handler.ServerCertificateCustomValidationCallback =
                (sender, cert, chain, sslPolicyErrors) => { return true; };

            using var client = new HttpClient(handler);

            try
            {
                var response = await client.GetStringAsync(sendString);
                Console.WriteLine("Http response received: ");
                DisplayMessages(response);
            }
            catch (Exception e)
            {
                Console.WriteLine(e.Message);
            }
        }
        static void DisplayMessages(string response)
        {
            var xmlDoc = new XmlDocument();
            xmlDoc.LoadXml(response);
            var nodes = xmlDoc.SelectNodes("response/data/message");

            foreach (XmlNode node in nodes)
            {
                try
                {
                    var sender = node.SelectSingleNode("originator").InnerText;
                    var text = node.SelectSingleNode("messagedata").InnerText;
                    DisplayMessage(sender, text);
                }
                catch (System.NullReferenceException e)
                {
                    Console.WriteLine("The inbox is empty!");
                }
            }
        }
        static void DisplayMessage(string sender, string text)
        {
            Console.WriteLine($"{sender}: {text}");
        }

    }
}
```

## Step 1 - Open Visual Studio

The first step of the guide, to create a C# application where you can place the example code from this page. The best tool to create a C# application is the Visual Studio. You can download it from the Prerequisites section if you haven't got it yet. If Visual Studio is already installed on your computer, you just need to open it from your desktop as you can see it in Figure 1.

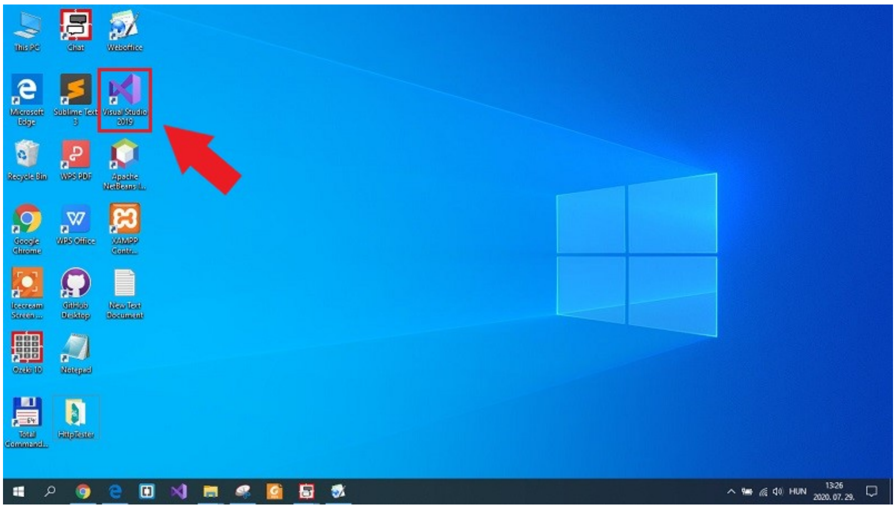


Figure 1 - Open Visual Studio 2019

## Step 2 - Create a new project

After you opened the Visual Studio, the welcome window is going to show up. You can open a recent project or create a completely new one from this window. As Figure 2 demonstrates, now you need to create a new project, so you just need to select this option from that window.

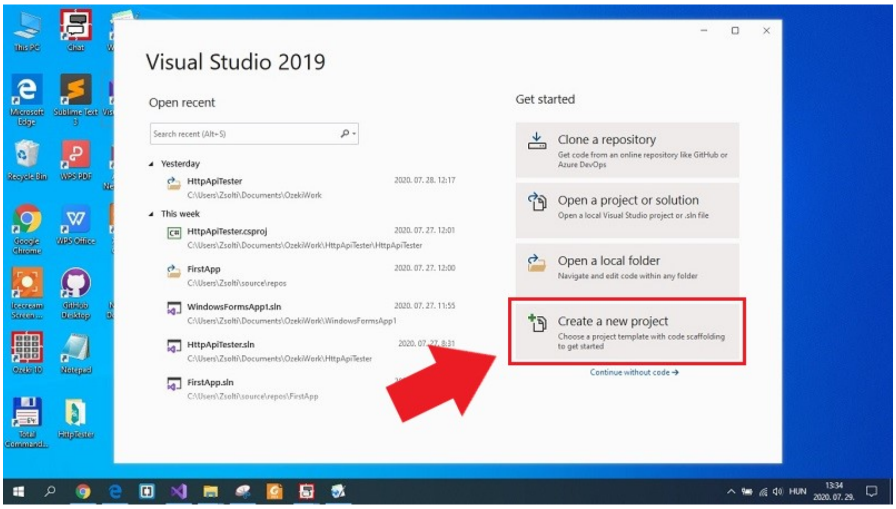

Visual Studio 2019
Open recent
Search recent (Alt+S)
Yesterday
HttpApiTester
C:\Users\Zsolti\Documents\OzekiWork
2020. 07. 28. 12:17
This week
HttpApiTester.csproj
C:\Users\Zsolti\Documents\OzekiWork\HttpApiTester\HttpApiTester
2020. 07. 27. 12:01
FirstApp
C:\Users\Zsolti\source\repos
2020. 07. 27. 12:00
WindowsFormsApp1.sln
C:\Users\Zsolti\Documents\OzekiWork\WindowsFormsApp1
2020. 07. 27. 11:55
HttpApiTester.sln
C:\Users\Zsolti\Documents\OzekiWork\HttpApiTester
FirstApp.sln
C:\Users\Zsolti\source\repos\FirstApp
Get started
Clone a repository
Get code from an online repository like GitHub or Azure DevOps
Open a project or solution
Open a local Visual Studio project or .sln file
Open a local folder
Navigate and edit code within any folder
Create a new project
Choose a project template with code scaffolding to get started
Continue without code →

Figure 2 - Create new project in Visual Studio

# Step 3 - Select Console App

After you decided to create a new project, the next window lists all the available options that you can select from if you wish to create a new C# application. To follow this guide, you just need to create a simple Console App, so as Figure 3 shows that, just select that option and click on 'Next'.

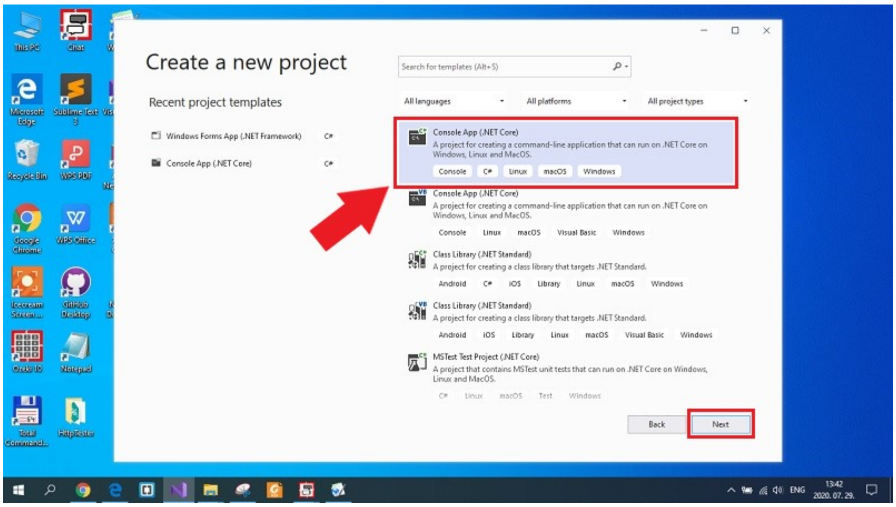
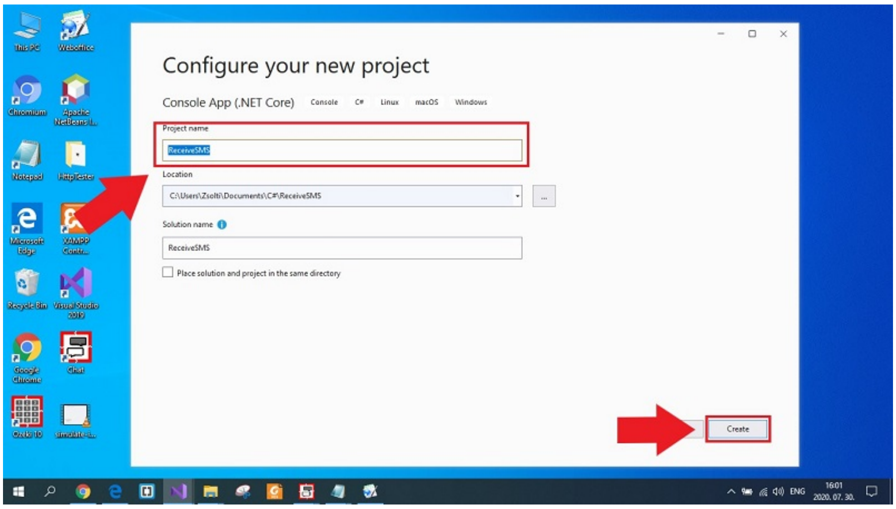

Create a new project
Search for templates (Alt+S)
Recent project templates
Windows Forms App (.NET Framework) C#
Console App (.NET Core) C#
All languages
All platforms
All project types
Console App (.NET Core)
A project for creating a command-line application that can run on .NET Core on Windows, Linux and MacOS.
Console C# Linux macOS Windows
Console App (.NET Core)
A project for creating a command-line application that can run on .NET Core on Windows, Linux and MacOS.
Console Linux macOS Visual Basic Windows
Class Library (.NET Standard)
A project for creating a class library that targets .NET Standard.
Android C# iOS Library Linux macOS Windows
Class Library (.NET Standard)
A project for creating a class library that targets .NET Standard.
Android iOS Library Linux macOS Visual Basic Windows
MSTest Test Project (.NET Core)
A project that contains MSTest unit tests that can run on .NET Core on Windows, Linux and MacOS.
C# Linux macOS Test Windows
Back
Next

Figure 3 - Select console App

# Step 4 - Configure your project

Before finally creating the Console App project, you need to do some configurations on the project. These are very simple configurations, you just need to give a name to your project as you can see it in Figure 4. At this window, you can also specify the location of the project files. If you finished with all the configurations, just click on 'Create' to create the Console App project.



Configure your new project

Console App (.NET Core)

Project name: ReceiveSMS

Location: C:\Users\Zsolti\Documents\C#\ReceiveSMS

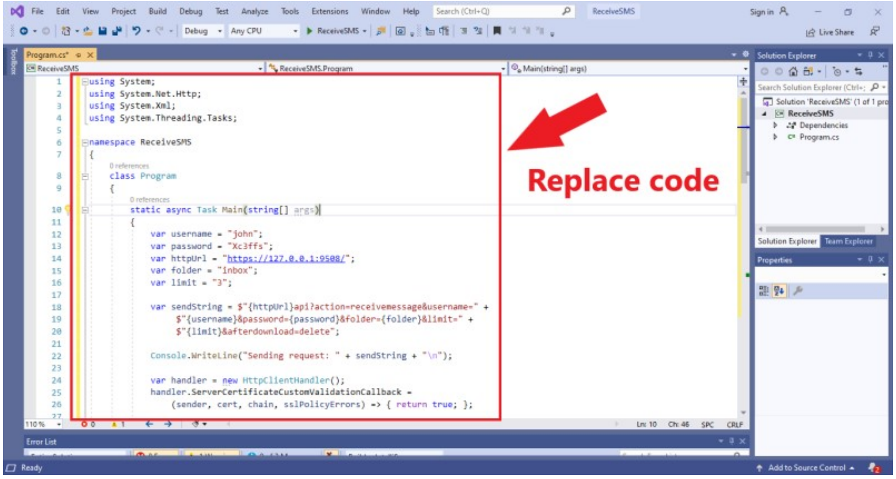
Solution name: ReceiveSMS

□ Place solution and project in the same directory

Figure 4 - Configure project name

## Step 5 - Place the example code into the project

The created project contains one C# file, which is Program.cs. This is the file, where you need to place the example code. For that, just scroll up to the example code, mark out the whole code, and press Ctrl+C on your keyboard. This operation just copied the code to your clipboard. After that, go to the Program.cs file, delete the code that you can find here, and like in Figure 5, press Ctrl+V to paste the code into that file.

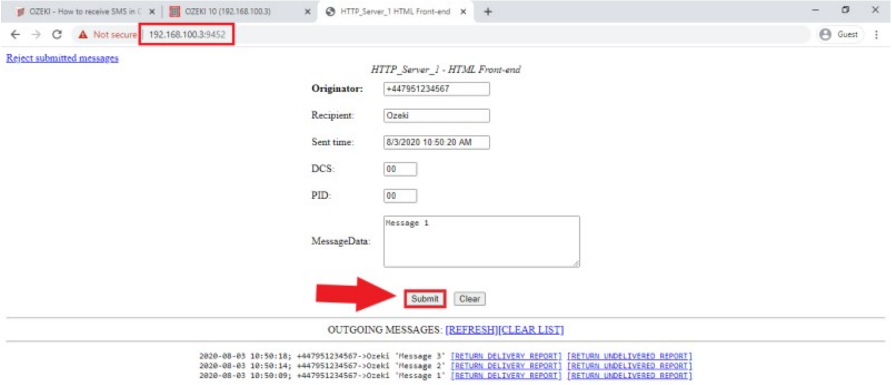


```
using System;
using System.Net.Http;
using System.Xml;
using System.Threading.Tasks;

namespace ReceiveSMS
{
    class Program
    {
        static async Task Main(string[] args)
        {
            var username = "john";
            var password = "Xc3ffs";
            var httpUrl = "https://127.0.0.1:9508/";
            var folder = "inbox";
            var limit = "3";

            var sendString = $"{httpUrl}api?action=receivemessage&username=" +
                $"{username}&password={password}&folder={folder}&limit=" +
                $"{limit}&afterdownload=delete";

            Console.WriteLine("Sending request: " + sendString + "\n");

            var handler = new HttpClientHandler();
            handler.ServerCertificateCustomValidationCallback =
                (sender, cert, chain, sslPolicyErrors) => { return true; };
```

Figure 5 - Replace code in Visual Studio

## Step 6 - Simlutate some incoming messages

The next thing that you need to do, is to wait for some incoming messages in your SMS Gateway. This can take some time, so you can choose the option to simulate some messages using the HTTP Server connection. So, as Figure 6 demonstrates that, just open the HTML form of the HTTP Server connection. This form provides the opportunity to create some messages. You just need to type 'Ozeki' as a recipient and write some text into the 'MessageData' field. To send the test message, just click on 'Submit'.



OZEKI - How to receive SMS in C
OZEKI 10 (192.168.100.3)
HTTP_Server_1 HTML Front-end
Not secure
192.168.100.3:9452
Guest
Reject submitted messages

*HTTP_Server_1 - HTML Front-end*

| | |
|---|---|
| **Originator:** | +447951234567 |
| Recipient: | Ozeki |
| Sent time: | 8/3/2020 10:50:20 AM |
| DCS: | 00 |
| PID: | 00 |
| MessageData: | Message 1 |

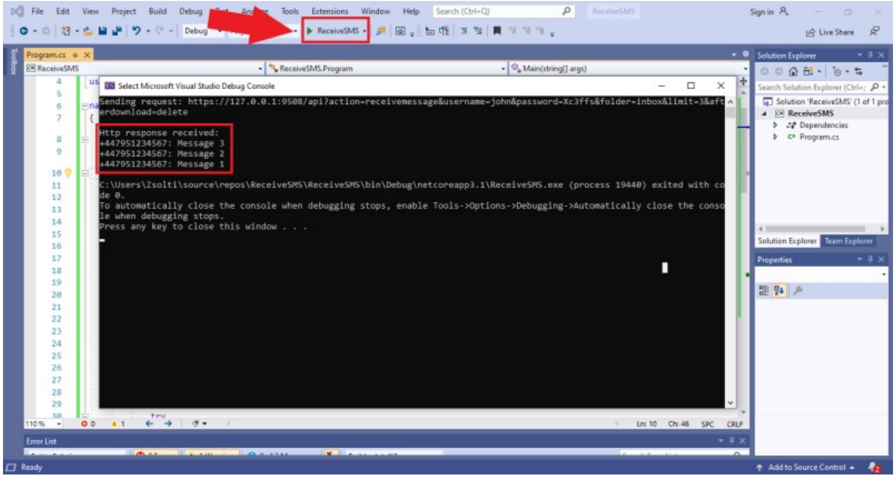
OUTGOING MESSAGES: [REFRESH][CLEAR LIST]

```
2020-08-03 10:50:18; +447951234567->Ozeki 'Message 3' [RETURN DELIVERY REPORT] [RETURN UNDELIVERED REPORT]
2020-08-03 10:50:14; +447951234567->Ozeki 'Message 2' [RETURN DELIVERY REPORT] [RETURN UNDELIVERED REPORT]
2020-08-03 10:50:09; +447951234567->Ozeki 'Message 1' [RETURN DELIVERY REPORT] [RETURN UNDELIVERED REPORT]
```

Figure 6 - Simulate some incoming SMS in Ozeki 10

# Step 7 - Run the example C# code

The last step that you need to perform is to run the C# code and get the received messages. to do that, just click on the 'Run' button in the Visual Studio as you can see it in Figure 7. By doing this, a console window shows up which prints the HTTP request that your application sent. It also prints the response message from SMS Gateway, which is a list of all messages that can be found in your inbox folder.



```
Sending request: https://127.0.0.1:9508/api?action=receivemessage&username=john&password=Xc3ffs&folder=inbox&limit=3&aft
erdownload=delete

Http response received:
+447951234567: Message 3
+447951234567: Message 2
+447951234567: Message 1

C:\Users\Zsolti\source\repos\ReceiveSMS\ReceiveSMS\bin\Debug\netcoreapp3.1\ReceiveSMS.exe (process 19440) exited with co
de 0.
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the conso
le when debugging stops.
Press any key to close this window . . .
```

Figure 7 - Run the C# code to receive SMS

# How to send SMS from Java

***The following example console application written in Java programming language is going to show you the way you can send HTTP requests to the SMS Gateway and get the response message as well. These HTTP requests can be used for sending SMS messages and you can easily configure the details of the SMS message by modifying the variables which contain all information regarding the SMS message.***

## What is a Java SMS API?

The Java SMS API enables you to implement SMS services into your Java Application. The API communicates with the SMS Gateway using HTTP requests which sends the message and retuns with a response message.

## Prerequisites

- Installed Apache NetBeans IDE 12.0
- Installed Ozeki 10 SMS Gateway
- A configured HTTP API User

## Send SMS from Java

1. Open Apache NetBeans IDE
2. Click on 'New project...'
3. Create a new Java Application
4. Name the project and the package
5. Create a new Java Class
6. Copy-Paste the example source code below
7. Run the Java Application
8. Select the main class if needed

## Java SMS source code example

The following example source code written in Java programming language is free to use, you can simply implement it into your project or you can modify the source code to use it for other projects or applications. If you would like to run this example code, you just need to create a new Java application with a single Java class and run the project as you can see it in the step instructions and the video above.

```
package tester;

import java.io.BufferedReader;
import java.io.InputStreamReader;
import java.net.HttpURLConnection;
import java.net.URL;
import java.net.URLEncoder;
import java.security.GeneralSecurityException;
import java.security.cert.X509Certificate;
import javax.net.ssl.HostnameVerifier;
import javax.net.ssl.HttpsURLConnection;
import javax.net.ssl.SSLContext;
import javax.net.ssl.SSLSession;
import javax.net.ssl.TrustManager;
import javax.net.ssl.X509TrustManager;

public class HttpApiTester {

    public static void main(String[] args) {

        TrustManager[] trustAllCerts = new TrustManager[]{
            new X509TrustManager() {
                @Override
                public java.security.cert.X509Certificate[] getAcceptedIssuers() {
                    return new X509Certificate[0];
                }

                @Override
                public void checkClientTrusted(
                        java.security.cert.X509Certificate[] certs, String authType) {
                }

                @Override
                public void checkServerTrusted(
                        java.security.cert.X509Certificate[] certs, String authType) {
                }
            }
        };

        try {
            SSLContext sc = SSLContext.getInstance("SSL");
            sc.init(null, trustAllCerts, new java.security.SecureRandom());
            HttpsURLConnection.setDefaultSSLSocketFactory(sc.getSocketFactory());

            HostnameVerifier allHostsValid = new HostnameVerifier() {
                @Override
                public boolean verify(String hostname, SSLSession session) {
                    return true;
                }
            };
            HttpsURLConnection.setDefaultHostnameVerifier(allHostsValid);

        } catch (GeneralSecurityException e) {
            System.out.println(e.getMessage());
        }

        try {
            StringBuilder sendString = new StringBuilder();
            String username = "john";
            String password = "Xc3ffs";
            String messagetype = "SMS:TEXT";
            String httpUrl = "https://127.0.0.1:9508/";
            String recipient = URLEncoder.encode("+36201324567", "UTF-8");
            String messagedata = URLEncoder.encode("TestMessage", "UTF-8");

            sendString.append(httpUrl).append("api?action=sendmessage").
                    append("&username=").append(username).append("&password=").
                    append(password).append("&recipient=").append(recipient).
                    append("&messagetype=").append(messagetype).append("&messagedata=").
                    append(messagedata);

            System.out.println("Sending request: " + sendString.toString());

            URL url = new URL(sendString.toString());
            HttpURLConnection con = (HttpURLConnection) url.openConnection();
            con.setRequestMethod("GET");

            BufferedReader br = null;
            System.out.println("Http response received: ");
            if (con.getResponseCode() == 200) {
                br = new BufferedReader(new InputStreamReader(con.getInputStream()));
                String strCurrentLine;
                while ((strCurrentLine = br.readLine()) != null) {
                    System.out.println(strCurrentLine);
                }
            } else {
                br = new BufferedReader(new InputStreamReader(con.getErrorStream()));
                String strCurrentLine;
                while ((strCurrentLine = br.readLine()) != null) {
                    System.out.println(strCurrentLine);
                }
            }

        } catch (Exception ex) {
            System.out.println(ex.getMessage());
        }
    }
}
```

## Step 1 - Open Apache NetBeans IDE

The first step is to create a Java application that can send SMS messages is to open Apache NetBeans IDE on your computer. Apache NetBeans provides editors, wizards, and templates to help you create applications in

Java, PHP and many other languages. If you haven't downloaded Apache NetBeans yet, you can download it from the Prerequisites section. After you installed it, you just have to open it from your desktop like in Figure 1.

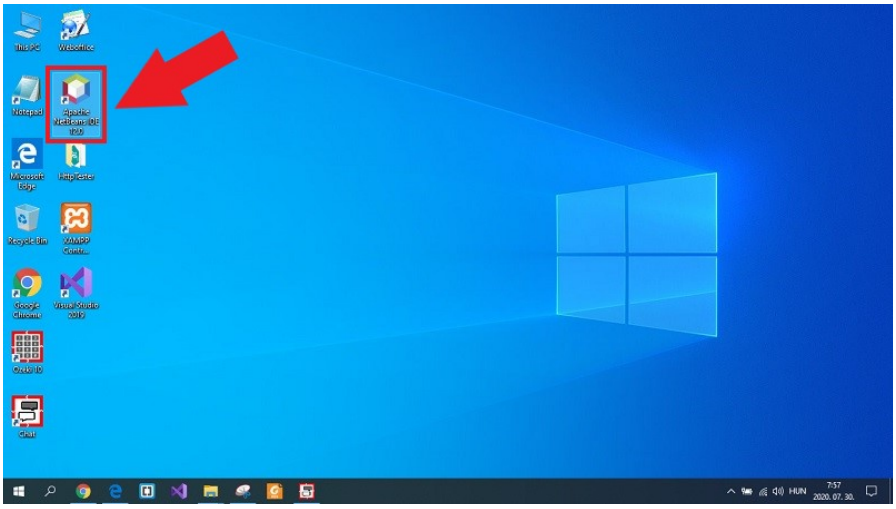


Figure 1 - Open Apache NetBeans IDE

## Step 2 - Create a new Java project

After you opened Apache NetBeans IDE, it opens up with the main window. Here, you can create the Java Application that needed to send SMS messages. For that, as you can see it in Figure 2, click on 'File' on the toolbar, and here, select the 'New project...' option.

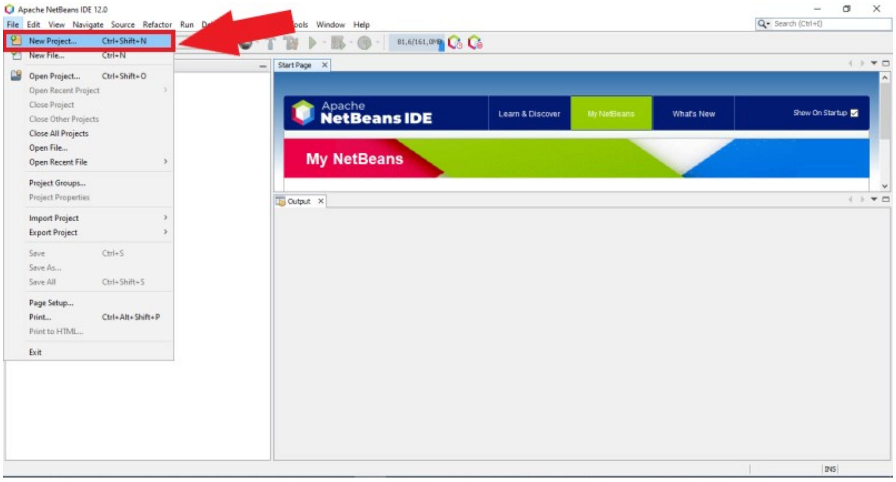
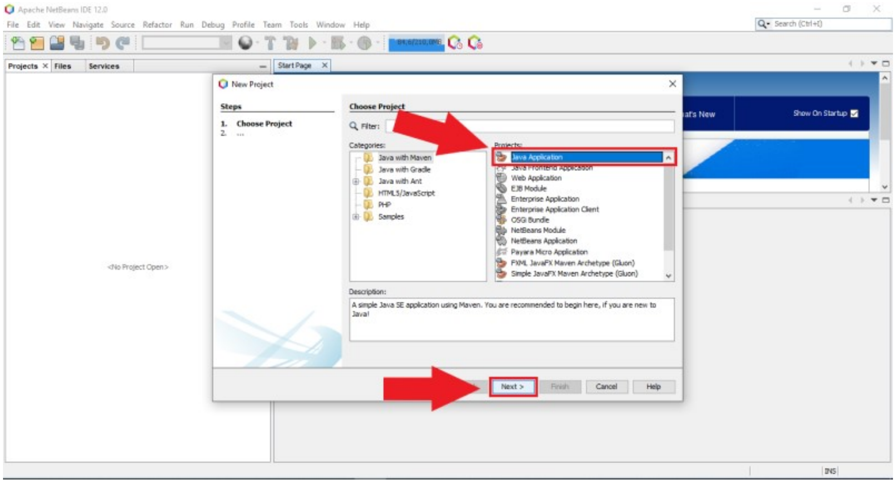


Figure 2 - Create new Java Project

## Step 3 - Select Java Application

Next, a new window pops up, that contains all types of projects that can be created in Apache NetBeans. Here, you can create the simple Java Application or you can select from more advanced types of projects. For this example you can use the simple Java Application, so select it from the list, and like in Figure 3 click Next.

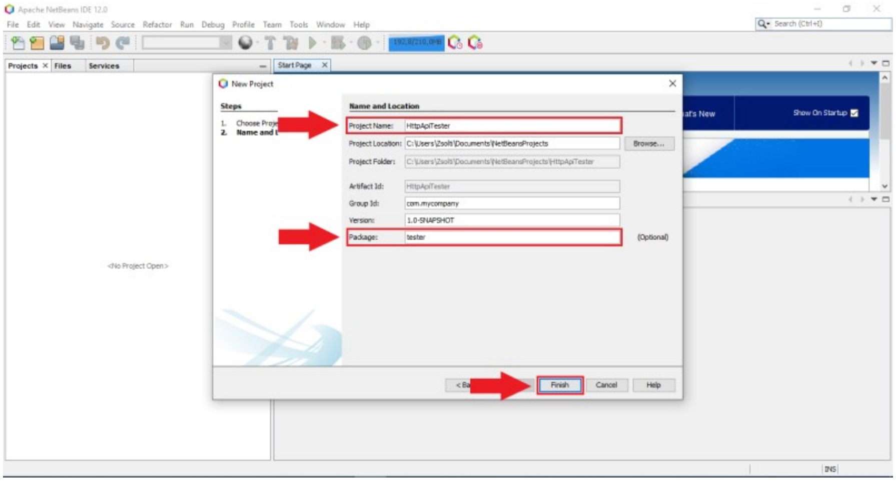


A simple Java SE application using Maven. You are recommended to begin here, if you are new to Java!

Figure 3 - Select java application project type

## Step 4 - Configure the Java Application

In the next step, you need to do some basic configuration for the Java Application. As you can see it in Figure 4, you can give a name for the project. You can also set the location of the project and lastly you can specify a name for the package as well. If you finished with the configuration, just click on Finish.



Project Name: HttpApiTester
Project Location: C:\Users\Zsolt\Documents\NetBeansProjects
Project Folder: C:\Users\Zsolt\Documents\NetBeansProjects\HttpApiTester
Artifact Id: HttpApiTester
Group Id: com.mycompany
Version: 1.0-SNAPSHOT
Package: tester

Figure 4 - Choose Project Name and Package

## Step 5 - Create a new Java Class

The create Java Application project is empty, so it does not contain any file with source code, so you need to create one. For that, select the package of your project, and click with the right button on your mouse. From the pop-up window, as Figure 5 shows that, select New, and after that, click on 'Java class...' to create a new Java class in your application.

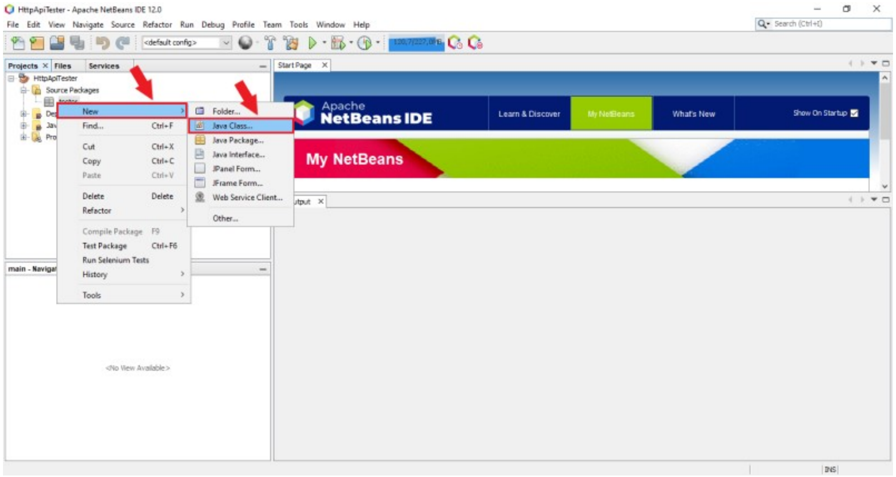
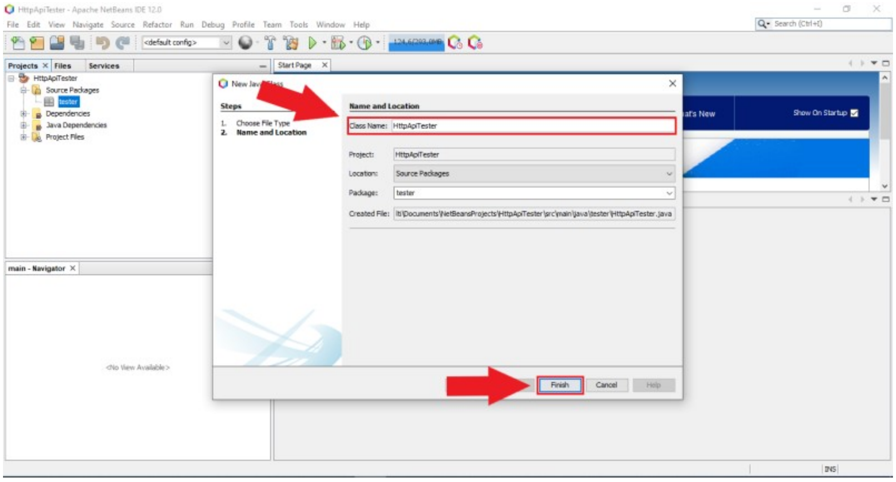


Figure 5 - Create new Class

# Step 6 - Configure the Java class

Before finishing the creation of the Java class, you need to specify a name for the class. That can be easily done as Figure 7 shows that. After you gave the right name for the Java class, just click on the Finish button to successfully create the Java class.



New Java Class

Steps

1. Choose File Type
2. Name and Location

Name and Location

Class Name: HttpApiTester

Project: HttpApiTester

Location: Source Packages

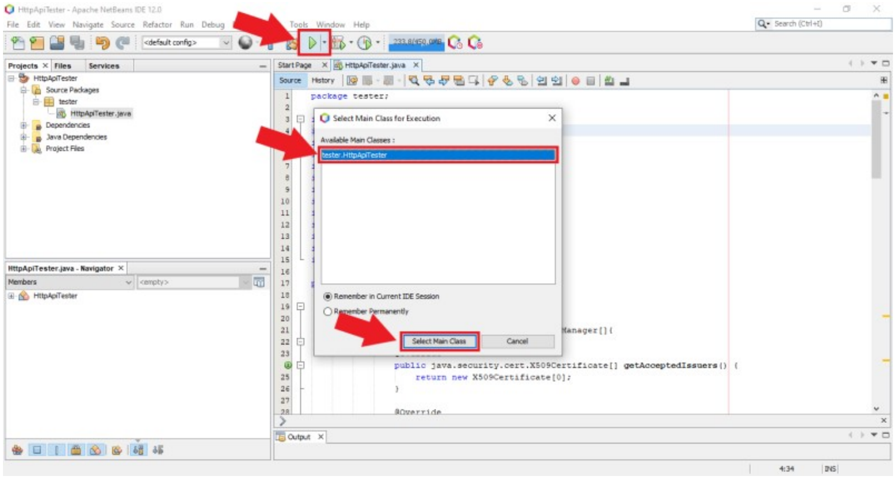
Package: tester

Created File: lti\Documents\NetBeansProjects\HttpApiTester\src\main\java\tester\HttpApiTester.java

Figure 6 - Give the new class a name

# Step 7 - Paste the source code

The next thing, that you need to do is to place the example code into your Java Application. For that, just scroll up to the example code copy the whole source code to your clipboard and place it into your newly created Java class. At this point, you can run the Java Application, so just click on the Run button on the toolbar. In the first run, as Figure 7 demonstrates that, you need to select the Main class for execution. So, just select the HttpApiTester class, and click on 'Select Main class'.

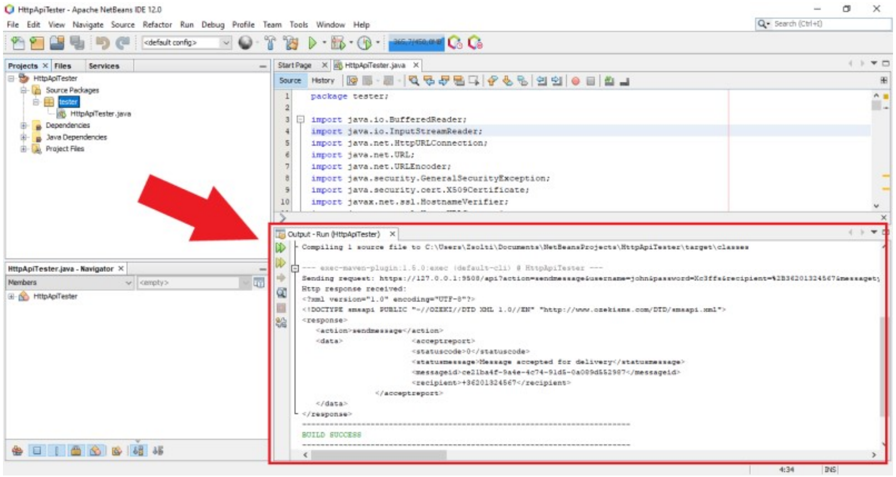


```
package tester;
Manager[] {
    public java.security.cert.X509Certificate[] getAcceptedIssuers() {
        return new X509Certificate[0];
    }
    @Override
```

Figure 7 - Paste Code from website, then select Main class

# Step 8 - See the result of the application

If you have done everything right till this point, the application starts, and you will be able to see the result in the console window as Figure 8 shows that. This window prints the HTTP request that was initiated and sent to the SMS Gateway and it also prints the response from the SMS Gateway which indicates that the delivery of the SMS message was successful or not.



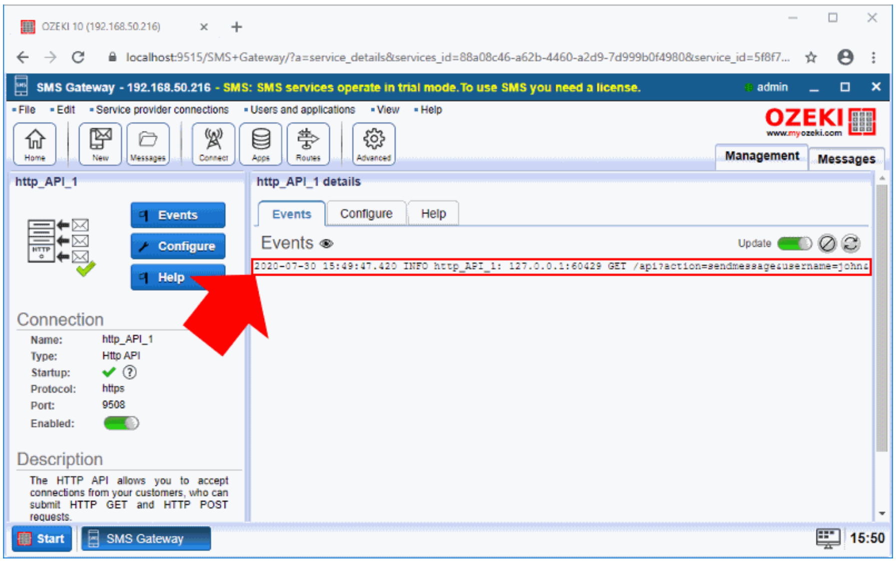
```
package tester;

import java.io.BufferedReader;
import java.io.InputStreamReader;
import java.net.HttpURLConnection;
import java.net.URL;
import java.net.URLEncoder;
import java.security.GeneralSecurityException;
import java.security.cert.X509Certificate;
import javax.net.ssl.HostnameVerifier;
```

```
Compiling 1 source file to C:\Users\Zsolti\Documents\NetBeansProjects\HttpApiTester\target\classes

--- exec-maven-plugin:1.5.0:exec (default-cli) @ HttpApiTester ---
Sending request: https://127.0.0.1:9508/api?action=sendmessage&username=john&password=Xc3ffs&recipient=%2B36201324567&messagety
Http response received:
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE smsapi PUBLIC "-//OZEKI//DTD XML 1.0//EN" "http://www.ozekisms.com/DTD/smsapi.xml">
<response>
   <action>sendmessage</action>
   <data>                  <acceptreport>
                           <statuscode>0</statuscode>
                           <statusmessage>Message accepted for delivery</statusmessage>
                           <messageid>ce21ba4f-9a4e-4c74-91d5-0a089d552987</messageid>
                           <recipient>+36201324567</recipient>
                    </acceptreport>
   </data>
</response>
------------------------------------------------------------------------
BUILD SUCCESS
------------------------------------------------------------------------
```

Figure 8 - The program starts and the result is displayed

# Step 9 - Check the send result in the Ozeki log

In Ozeki 10 SMS Gateway, you can follow what messages sent by your application, since the HTTP API service logs every event that occured during the time it is enabled. So, after you opened the SMS Gateway, and selected the details of the HTTP API service, you will be able to see the events. As Figure 9 shows that, the service logged an event, when the example Java Application sent the HTTP request to the service.


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=service_details&services_id=88a08c46-a62b-4460-a2d9-7d999b0f4980&service_id=5f8f7...
SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode. To use SMS you need a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
http_API_1
Events
Configure
Help
http_API_1 details
Events
Configure
Help
Events
Update
Start
SMS Gateway
15:50

Connection

Name: http_API_1
Type: Http API
Startup: ✓
Protocol: https
Port: 9508
Enabled:

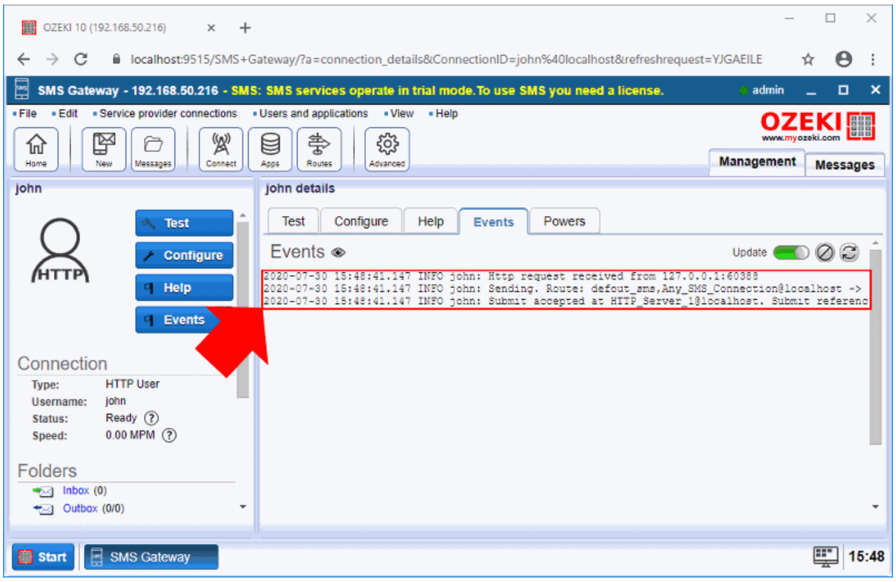
Description

The HTTP API allows you to accept connections from your customers, who can submit HTTP GET and HTTP POST requests.

```
2020-07-30 15:49:47.420 INFO http_API_1: 127.0.0.1:60429 GET /api?action=sendmessage&username=john&
```

Figure 9 - Check the logs of the HTTP API Service

The way that SMS Gateway processes the messages can be also viewed back by the events. For that, open the HTTP API User connection, that you had to configure before. Figure 10 demonstrates that how the connection handles the HTTP request and send the message to the recipient that you specified in your Java Application.


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=connection_details&ConnectionID=john%40localhost&refreshrequest=YJGAEILE
SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode. To use SMS you need a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
john
HTTP
Test
Configure
Help
Events
john details
Test
Configure
Help
Events
Powers
Events
Update
Start
SMS Gateway
15:48

Connection

Type: HTTP User
Username: john
Status: Ready
Speed: 0.00 MPM

Folders

Inbox (0)
Outbox (0/0)

```
2020-07-30 15:48:41.147 INFO john: Http request received from 127.0.0.1:60388
2020-07-30 15:48:41.147 INFO john: Sending. Route: defout_sms,Any_SMS_Connection@localhost ->
2020-07-30 15:48:41.147 INFO john: Submit accepted at HTTP_Server_1@localhost. Submit referenc
```

Figure 10 - Check the logs of the HTTP API User

# Receive SMS in Java

***The guide on this page is about to give you a brief introduction to how you can get the received messages from SMS Gateway to your Java application. This operation is demonstrated by a simple Java code that uses HTTP requests to collect the incoming messages from SMS Gateway. If you follow this guide you are going to learn how you can create an example Java application that gets the messages from the inbox folder and you will be able to see how to test the solution. So, let's get started.***

## What is a Java SMS API?

The Java SMS API enables you to implement SMS services into your Java Application. The API communicates with the SMS Gateway using HTTP requests which sends the message and retuns with a response message.

## Prerequisites

- Installed Apache NetBeans IDE 12.0
- Installed Ozeki 10 SMS Gateway
- A configured HTTP Server connection

## Receive SMS in Java

1. Open Apache NetBeans IDE
2. Create a new Java Application
3. Create a new Java Class
4. Copy-Paste the example source code below
5. Type 'https://localhost:9515' in your browser to open SMS Gateway
6. Select HTTP Server connection and open its HTML form
7. Send some test messages
8. Run the Java application to get the messages

The Java SMS example code below which can get the incoming messages from the SMS Gateway is free to use, you can modify it or use it in your project. If you would like to just test the solution, you need to follow the step by step instructions below or you can also watch the video above to learn how you can create the example Java application that can get the incoming messages from the SMS Gateway.

```java
package ozeki;

import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStreamReader;
import java.net.HttpURLConnection;
import java.net.URL;
import java.security.GeneralSecurityException;
import java.security.cert.X509Certificate;
import java.util.stream.Collectors;
import javax.net.ssl.HostnameVerifier;
import javax.net.ssl.HttpsURLConnection;
import javax.net.ssl.SSLContext;
import javax.net.ssl.SSLSession;
import javax.net.ssl.TrustManager;
import javax.net.ssl.X509TrustManager;
import javax.xml.parsers.*;
import org.xml.sax.InputSource;
import org.w3c.dom.*;
import java.io.*;

public class ReceiveSMS {

    public static void main(String[] args) {

        TrustManager[] trustAllCerts = new TrustManager[]{
            new X509TrustManager() {
                @Override
                public java.security.cert.X509Certificate[] getAcceptedIssuers() {
                    return new X509Certificate[0];
                }

                @Override
                public void checkClientTrusted(
                        java.security.cert.X509Certificate[] certs, String authType) {
                }

                @Override
                public void checkServerTrusted(
                        java.security.cert.X509Certificate[] certs, String authType) {
                }
            }
        };
        try {
            SSLContext sc = SSLContext.getInstance("SSL");
            sc.init(null, trustAllCerts, new java.security.SecureRandom());
            HttpsURLConnection.setDefaultSSLSocketFactory(sc.getSocketFactory());

            HostnameVerifier allHostsValid = new HostnameVerifier() {
                @Override
                public boolean verify(String hostname, SSLSession session) {
                    return true;
                }
            };
            HttpsURLConnection.setDefaultHostnameVerifier(allHostsValid);

        } catch (GeneralSecurityException e) {
            System.out.println(e.getMessage());
        }

        try {
            StringBuilder sendString = new StringBuilder();
            String username = "john";
            String password = "Xc3ffs";
            String httpUrl = "https://127.0.0.1:9508/";
            String folder = "inbox";
            String limit = "3";

            sendString.append(httpUrl).append("api?action=receivemessage&username=").
                    append(username).append("&password=").
                    append(password).append("&folder=").append(folder).
                    append("&limit=").append(limit).append("&afterdownload=delete");

            System.out.println("Sending request: " + sendString.toString());

            URL url = new URL(sendString.toString());
            HttpURLConnection con = (HttpURLConnection) url.openConnection();
            con.setRequestMethod("GET");

            BufferedReader br = null;
            System.out.println("Http response received: ");
            if (con.getResponseCode() == 200) {
                br = new BufferedReader(new InputStreamReader(con.getInputStream()));
                String response =  br.lines().collect(Collectors.joining("\n"));
                response = response.substring(response.indexOf('\n')+1);
                response = response.substring(response.indexOf('\n')+1);
                DisplayMessages(response);

            } else {
                br = new BufferedReader(new InputStreamReader(con.getErrorStream()));
                String response =  br.lines().collect(Collectors.joining("\n"));
                System.out.println(response);
            }

        } catch (IOException ex) {
            System.out.println(ex.getMessage());
        }
    }

    static void DisplayMessages(String response) {
        try{
            DocumentBuilderFactory dbf =
                DocumentBuilderFactory.newInstance();
            DocumentBuilder db = dbf.newDocumentBuilder();
            InputSource is = new InputSource();
            is.setCharacterStream(new StringReader(response));

            Document doc = db.parse(is);
            NodeList nodes = doc.getElementsByTagName("message");

            for (int i = 0; i < nodes.getLength(); i++) {
                Element element = (Element) nodes.item(i);

                NodeList originator = element.getElementsByTagName("originator");
                Element line = (Element) originator.item(0);
                String sender = getCharacterDataFromElement(line);

                NodeList messagedata = element.getElementsByTagName("messagedata");
                line = (Element) messagedata.item(0);
                String text = getCharacterDataFromElement(line);
                DisplayMessage(sender, text);
            }
        } catch (Exception ex) {
            System.out.println("The inbox is empty");
        }
    }
    public static void DisplayMessage(String sender, String text) {
        System.out.println(sender + ": "+ text);
    }

    public static String getCharacterDataFromElement(Element e) {
        Node child = e.getFirstChild();
        if (child instanceof CharacterData) {
           CharacterData cd = (CharacterData) child;
           return cd.getData();
        }
        return "?";
      }
}
```

## Step 1 - Open Apache NetBeans IDE

To be able to create a Java application, you need to have an application which capable of doing that. One of these applications is the Apache NetBeans IDE which can be downloaded from the PRerequisites section above. If you already have got this IDE, you just need to open it from your desktop as you can see it in Figure 1.

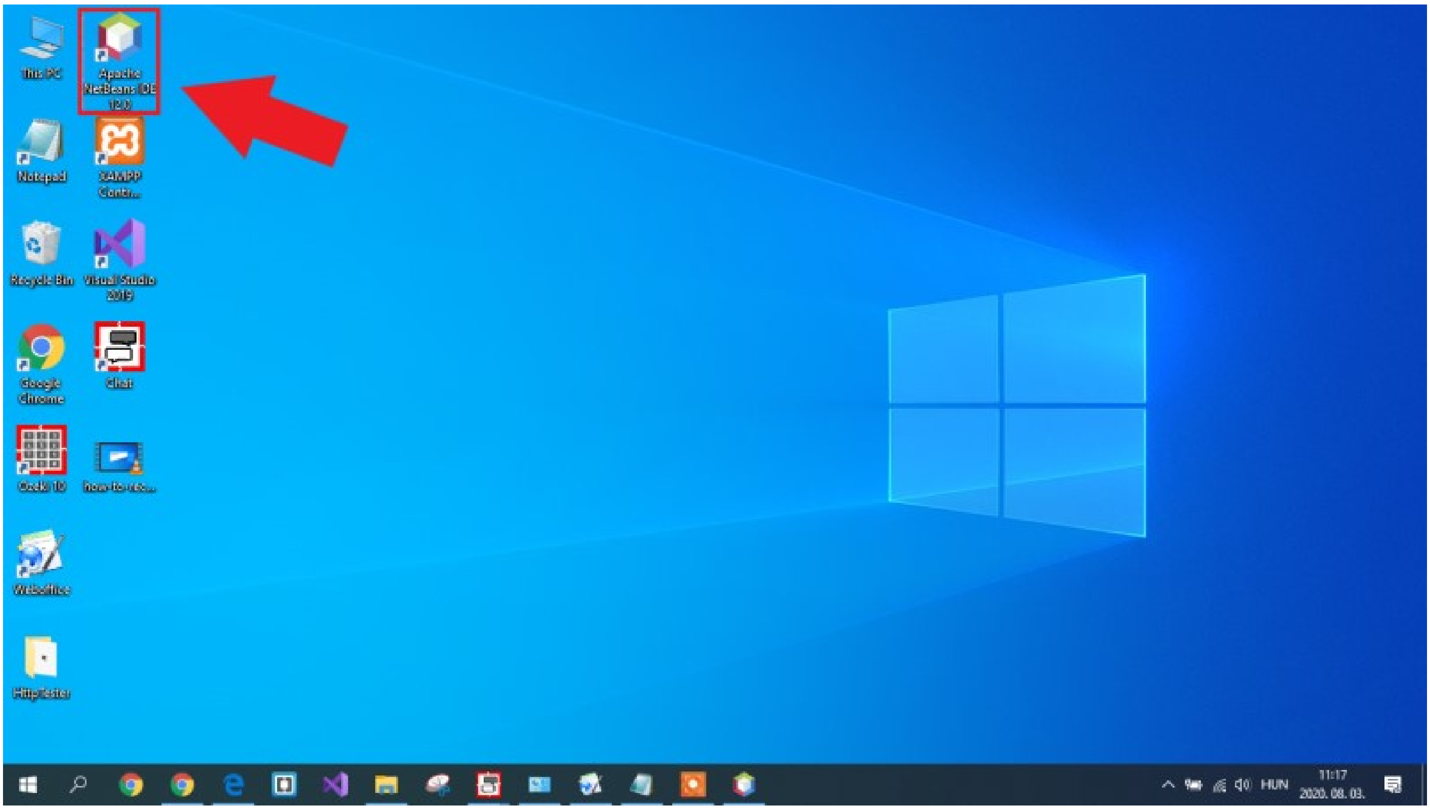
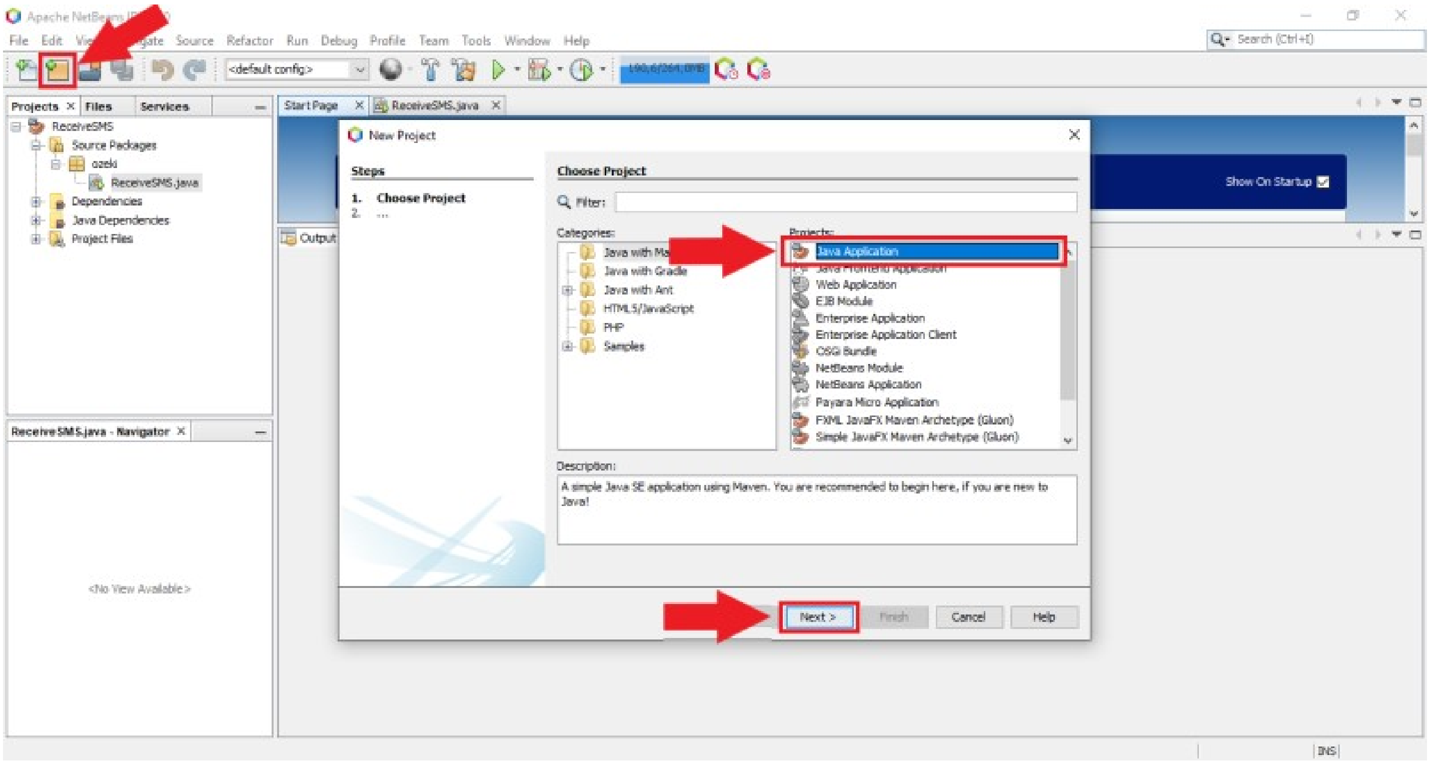


Figure 1 - Open Apache NetBeans

## Step 2 - Create a new Java application

After you opened Apache NetBeans IDE, the first thing that you need to do here is to create a Java application. For that, select the 'New project..' option from the toolbar. By doing this action, a window shows up, which contains all the available projects that can be created in Apache NetBeans. Here, select the Java application option like in Figure 2, and lastly, click on 'Next'.

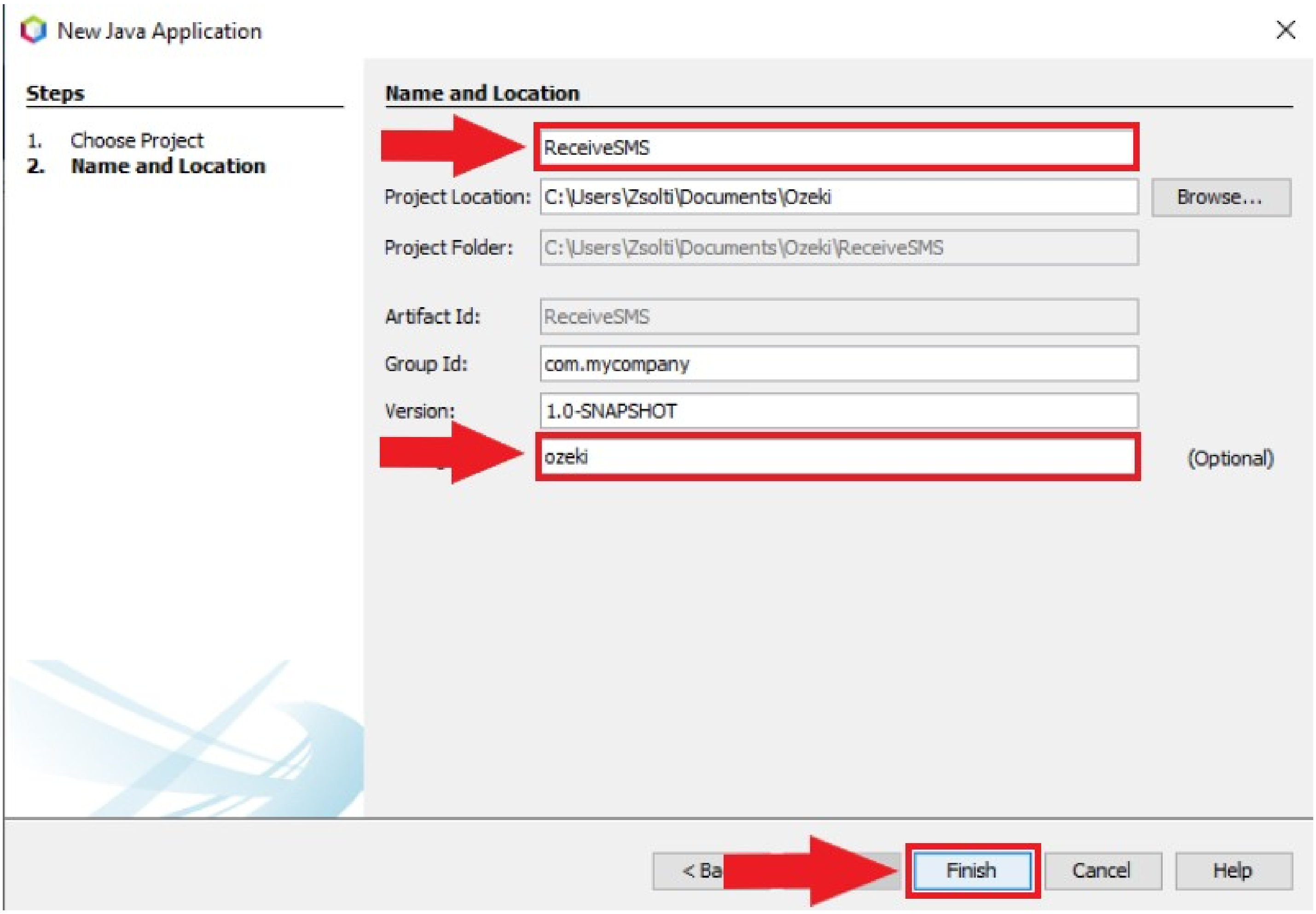


A simple Java SE application using Maven. You are recommended to begin here, if you are new to Java!

Figure 2 - Create new Java Application project

## Step 3 - Configure the Java application

Before creating the Java application, you need to do some configurations on it. Here in this menu, you need to specify a name for the project, you can also select a location for the project. The name of the package can be specified here as well. If you finished with all the configurations, just click on 'Finish' just like in Figure 3.



ReceiveSMS

Project Location: C:\Users\Zsolti\Documents\Ozeki

Project Folder: C:\Users\Zsolti\Documents\Ozeki\ReceiveSMS

Artifact Id: ReceiveSMS

Group Id: com.mycompany

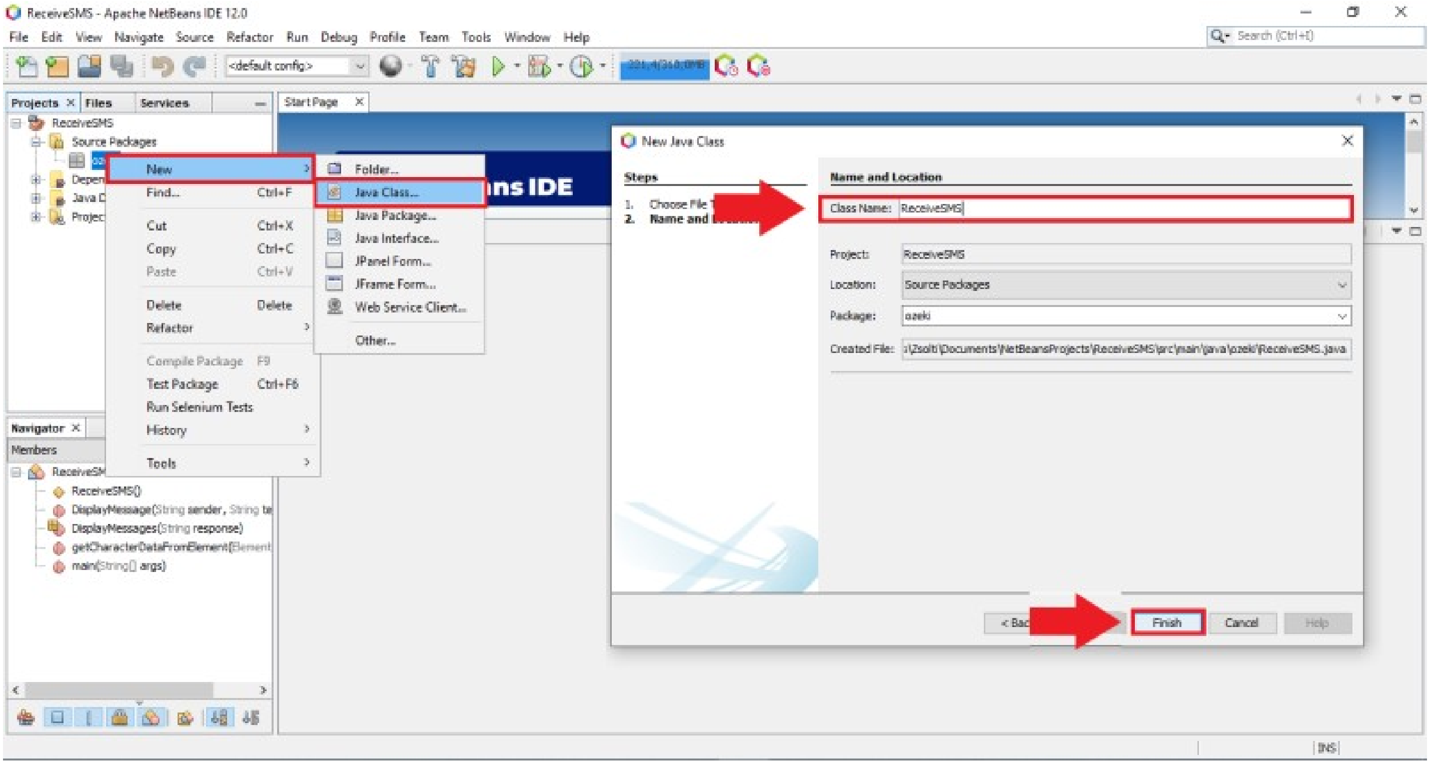
Version: 1.0-SNAPSHOT

ozeki

Figure 3 - Configure project and package name

## Step 4 - Create a Java class

After you created the Java application, you have to create a Java class since at this point it does not contain any class yet. For that, select the package with a right click, then 'New', and as Figure 4 shows that, click on 'Java class...'. Then, you need to specify a name for the Java class and if you finished, just click on 'Create'.



Class Name: ReceiveSMS

Project: ReceiveSMS

Location: Source Packages

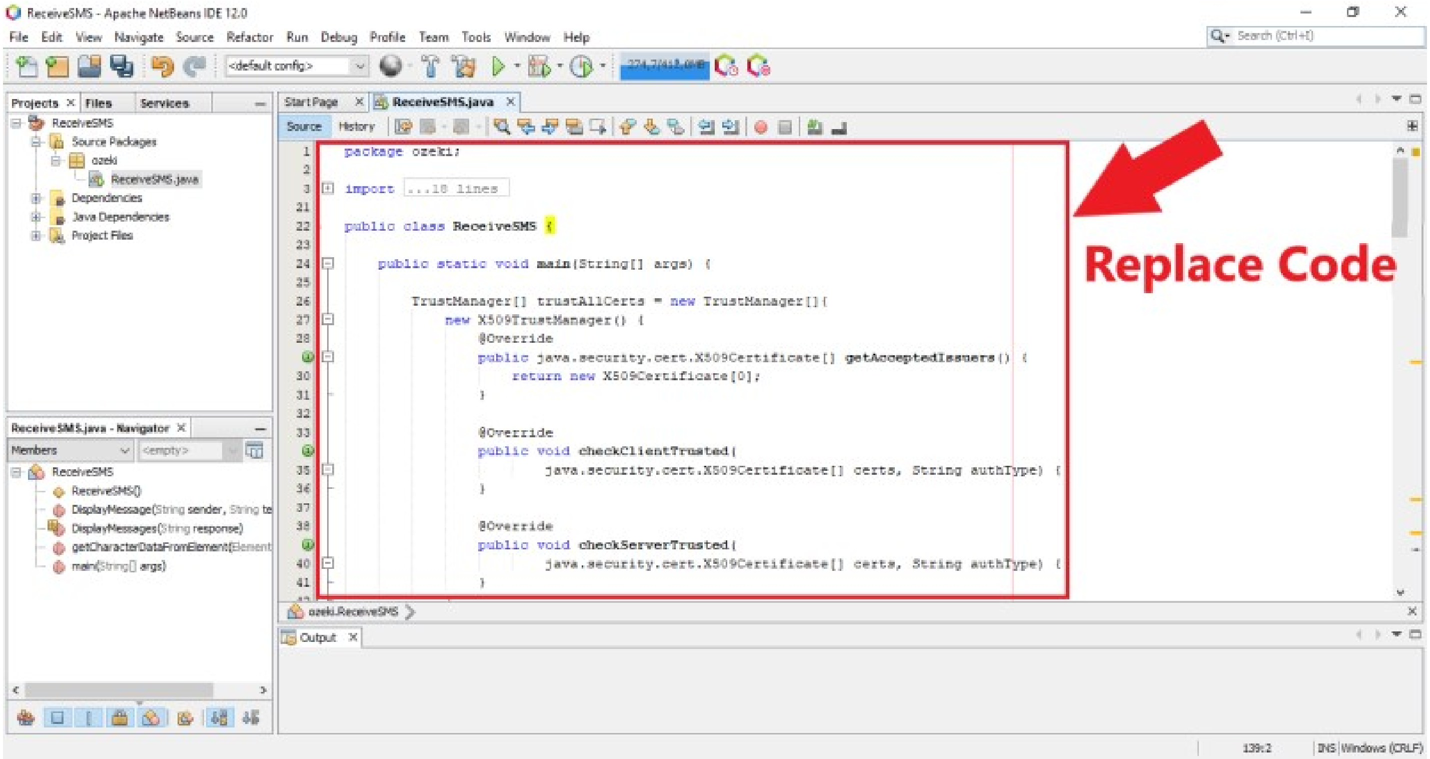
Package: ozeki

Created File: :\Zsolti\Documents\NetBeansProjects\ReceiveSMS\src\main\java\ozeki\ReceiveSMS.java

Figure 4 - Create new Java Class

## Step 5 - Replace the Java code from this page

At this point, you have the Java application that you need, but it does not contain the right code. To fix that, you need to replace the original code with the example code from this page. So, just go to the example code, and copy it to your clipboard using the Ctrl+C keyboard shortcut. Then, go to the created Java class, and first, delete the code that you can be found here. After that, press Ctrl+V to paste the example code to the Java class as you can see it in Figure 5.

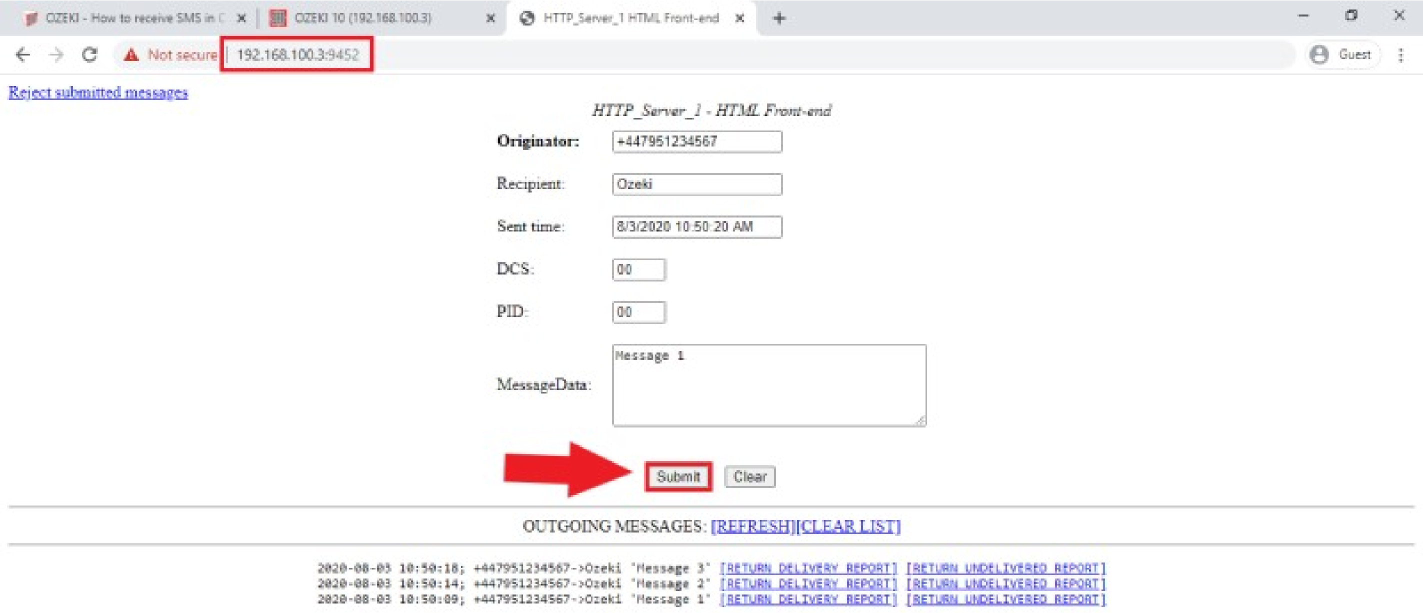


```
package ozeki;

import ...18 lines

public class ReceiveSMS {

    public static void main(String[] args) {

        TrustManager[] trustAllCerts = new TrustManager[]{
            new X509TrustManager() {
                @Override
                public java.security.cert.X509Certificate[] getAcceptedIssuers() {
                    return new X509Certificate[0];
                }

                @Override
                public void checkClientTrusted(
                        java.security.cert.X509Certificate[] certs, String authType) {
                }

                @Override
                public void checkServerTrusted(
                        java.security.cert.X509Certificate[] certs, String authType) {
                }
```

Figure 5 - Replace java code from the website

## Step 6 - Send some test messages

Before running the example Java application, you need to have some incoming messages that can be collected by the application. SMS Gateway provides you the opportunity to simulate the incoming messages so you can test your solution. For that, just open SMS Gateway and select the HTTP Server connection. If you don't have a HTTP Server connection, check how to create a HTTP Server connection. Here, open the HTML form of this connection, and like in Figure 6, send some test messages.



Reject submitted messages

HTTP_Server_1 - HTML Front-end

Originator: +447951234567

Recipient: Ozeki

Sent time: 8/3/2020 10:50:20 AM

DCS: 00

PID: 00

MessageData: Message 1

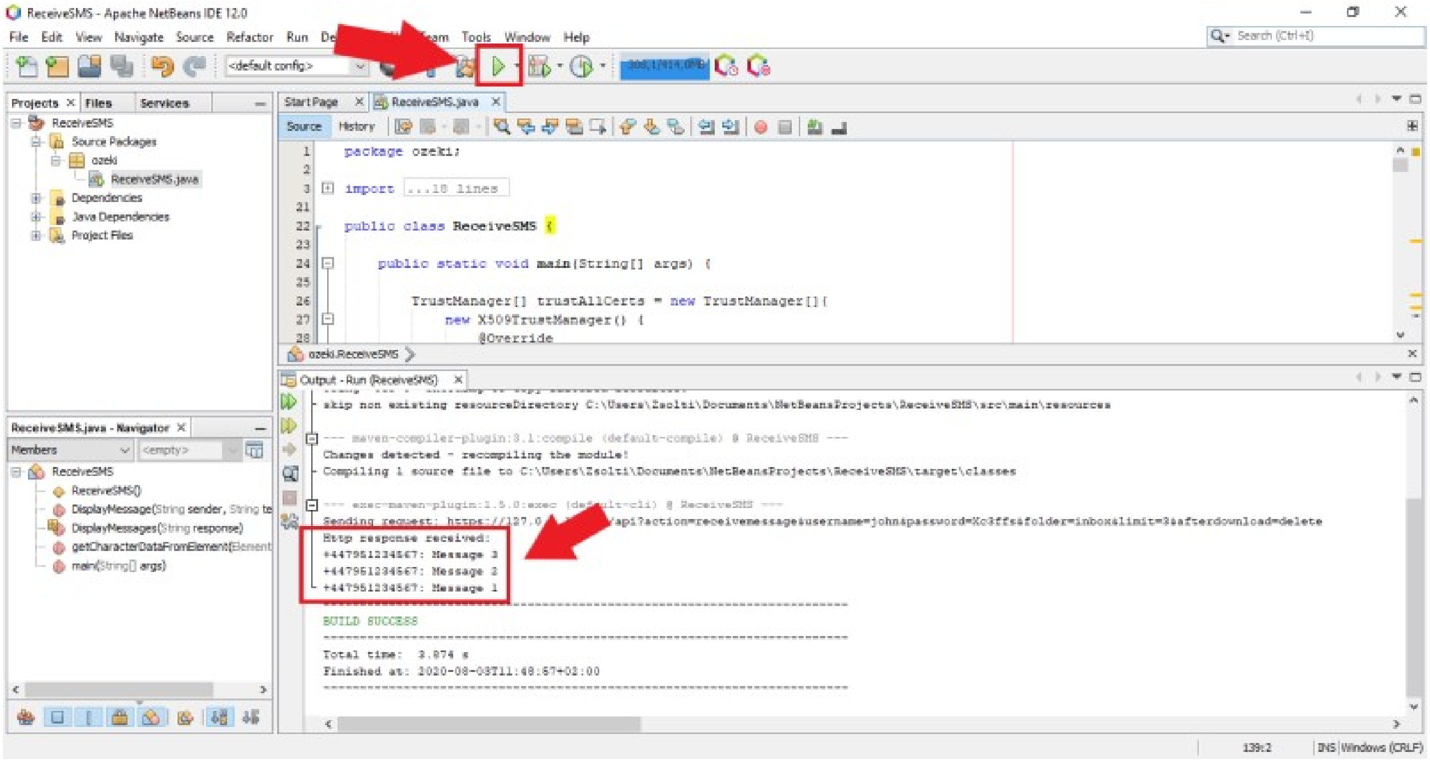
OUTGOING MESSAGES: [REFRESH][CLEAR LIST]

```
2020-08-03 10:50:18; +447951234567->Ozeki 'Message 3' [RETURN DELIVERY REPORT] [RETURN UNDELIVERED REPORT]
2020-08-03 10:50:14; +447951234567->Ozeki 'Message 2' [RETURN DELIVERY REPORT] [RETURN UNDELIVERED REPORT]
2020-08-03 10:50:09; +447951234567->Ozeki 'Message 1' [RETURN DELIVERY REPORT] [RETURN UNDELIVERED REPORT]
```

Figure 6 - Simulate some incoming SMS

# Step 7 - Run the Java application

The last step of this guide is to run the Java application. This is a simple operation, all you need to do is to click on the 'Run' button in Apache NetBeans like in Figure 7. This action runs the example Java code, which prints the HTTP request that was sent to the SMS Gateway, and then, it prints the response as well from the SMS Gateway which contains all the incoming messages.



```
package ozeki;

import ...18 lines

public class ReceiveSMS {

    public static void main(String[] args) {

        TrustManager[] trustAllCerts = new TrustManager[]{
            new X509TrustManager() {
                @Override
```

```
skip non existing resourceDirectory C:\Users\Zsolti\Documents\NetBeansProjects\ReceiveSMS\src\main\resources

--- maven-compiler-plugin:3.1:compile (default-compile) @ ReceiveSMS ---
Changes detected - recompiling the module!
Compiling 1 source file to C:\Users\Zsolti\Documents\NetBeansProjects\ReceiveSMS\target\classes

--- exec-maven-plugin:1.5.0:exec (default-cli) @ ReceiveSMS ---
Sending request: https://127.0.          /api?action=receivemessage&username=john&password=Xc3ffs&folder=inbox&limit=3&afterdownload=delete
Http response received:
+447951234567: Message 3
+447951234567: Message 2
+447951234567: Message 1
------------------------------------------------------------------------
BUILD SUCCESS
------------------------------------------------------------------------
Total time:  3.874 s
Finished at: 2020-08-03T11:48:57+02:00
------------------------------------------------------------------------
```

Figure 7 - Run the java code in NetBeans to receive SMS

# How to send SMS from PHP

***This page provides a simple PHP example code that allows you to send SMS messages using HTTP requests via the SMS Gateway. The PHP code embedded into an HTML document, so to run the code, you just need to create an HTML file, copy-paste the example code and open the file using your browser. The browser runs the whole HTML file with the PHP code as well, and you will be able to see the results of the SMS sending in the opened HTML page.***

## What is a PHP SMS API?

The PHP SMS API ensures you to send SMS messages from a web application of a website. The API uses HTTP requests to communicate with the SMS Gateway that delivers the message and sends a response back.

## Prerequisites

- Installed XAMPP
- Installed Ozeki 10 SMS Gateway
- A configured HTTP API User

## Send SMS from PHP

1. Open XAMPP Control Panel
2. Start Apache service
3. Open Sublime Text or other text editor application
4. Copy-Paste the PHP source code below
5. Save the file to the xampp/htmldocs folder
6. Name the file as index.php
7. Open a new tab in your browser
8. Type 'localhost' and hit Enter to run the PHP code

## PHP SMS source code example

The following example PHP source code below is free to use, you can simply implement it into your project or you can modify any segments of the source code to use it for other projects or applications. If you would like to run this example code itself, you just have to follow the instructions from the steps and the video above.

```
<!DOCTYPE html>
<html>
<head>
    <title>Http Request sending</title>
</head>
<body>
    <?php
        $username = "john";
        $password = "Xc3ffs";
        $messagetype = "SMS:TEXT";
        $httpUrl = "https://127.0.0.1:9508/";
        $recipient = urlencode("+36201324567");
        $messagedata = urlencode("TestMessage");

        $sendString = $httpUrl."api?action=sendmessage"."&username="
                     .$username."&password="
                     .$password."&recipient=".$recipient."&messagetype="
                     .$messagetype."&messagedata=".$messagedata;

        echo '<p><b> Sending html request:</b> '.$sendString.'</p>';
        $aContext = array(
            'http' => array(
                'method'  => 'GET',
            ),
            'ssl' => array(
                'verify_peer' => false,
                'verify_peer_name' => false,
            )
        );
        $cxContext = stream_context_create($aContext);
        $response = file_get_contents($sendString, true, $cxContext);

        echo '<p><b> Http response received :</b> </p>';
        echo '<xmp>' . $response. '</xmp>';
    ?>
</body>
</html>
```

# Step 1 - Open XAMPP Control Panel

The first step to run the PHP example code on your computer is to open XAMPP Control Panel. XAMPP is a free and open-source cross-platform web server solution stack, which contains a Apache HTTP Server, MariaDB database, and interpreters for script written in PHP and Perl programming languages. If you haven't downloaded this application yet, you can download it easily by following the link in the Prerequisites section. If you installed XAMPP already, you just have to open it by clicking on its icon on the desktop as you can see it in Figure 1.

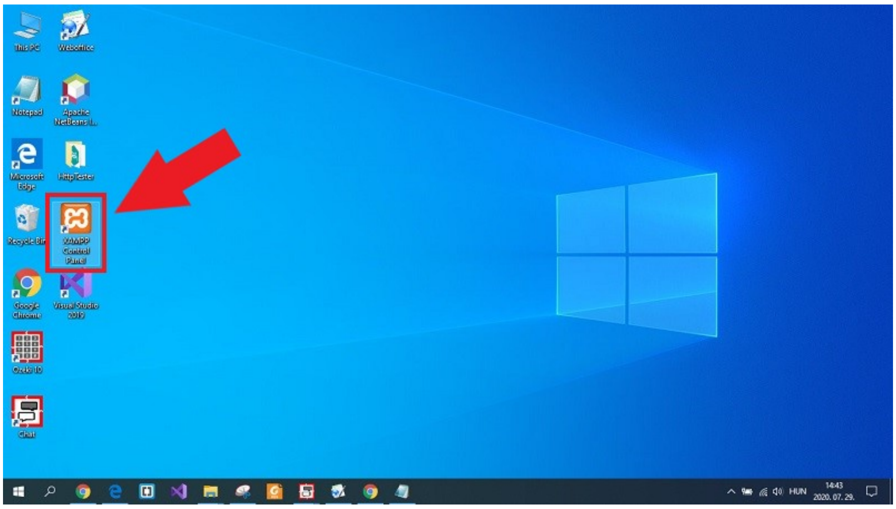
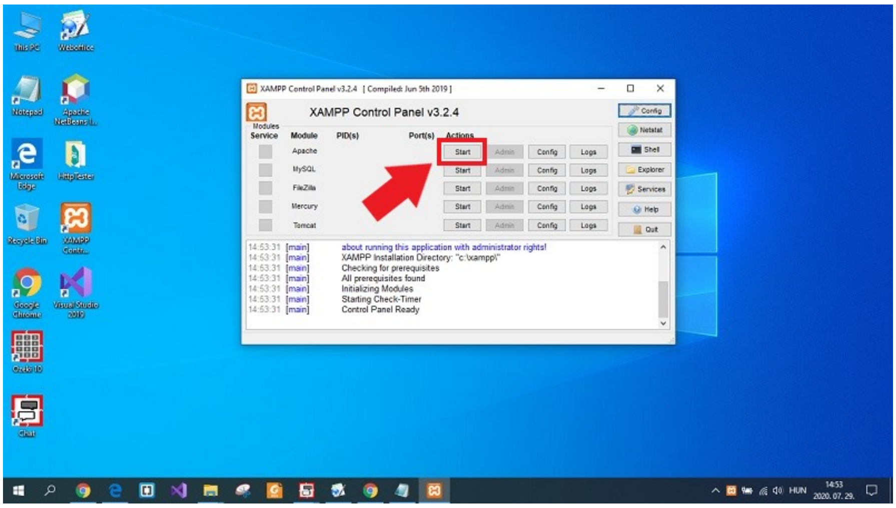


Figure 1 - Start XAMPP Control Panel

# Step 2 - Start Apache Server

After you opened XAMPP Control Panel, the main window of XAMPP shows up. Here, you will be able to see all the services that XAMPP provides for you. At this point, you need to start the Apache Server as Figure 2 demonstrates it. This service ensures you to run a local webserver where you can build and develop your web application or website. Now, this service provides the environment to run the PHP SMS source code.

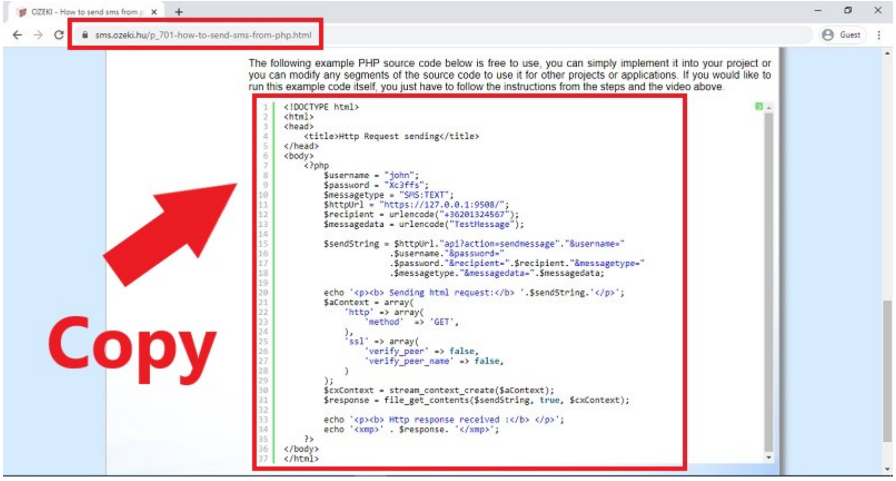


```
14:53:31  [main]    about running this application with administrator rights!
14:53:31  [main]    XAMPP Installation Directory: "c:\xampp\"
14:53:31  [main]    Checking for prerequisites
14:53:31  [main]    All prerequisites found
14:53:31  [main]    Initializing Modules
14:53:31  [main]    Starting Check-Timer
14:53:31  [main]    Control Panel Ready
```

Figure 2 - Starting Apache Server

## Step 3 - Copy the PHP SMS example source code

The next step after starting the Apache server is to get the example source code. Luckily, this is the right place to grab that PHP SMS source code. All you have to do here, is to scroll up to the 'PHP SMS source code example' section, and as you can see it in Figure 3, just mark out every line of the source code and press Ctrl+C on your keyboard to copy the source code to the clipboard.



OZEKI - How to send sms from p
sms.ozeki.hu/p_701-how-to-send-sms-from-php.html
Guest
Copy

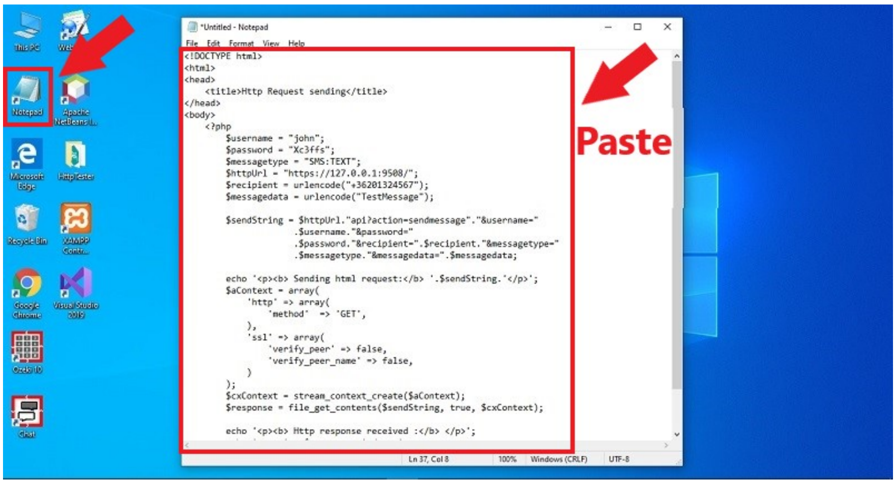
The following example PHP source code below is free to use, you can simply implement it into your project or you can modify any segments of the source code to use it for other projects or applications. If you would like to run this example code itself, you just have to follow the instructions from the steps and the video above.

```
<!DOCTYPE html>
<html>
<head>
    <title>Http Request sending</title>
</head>
<body>
    <?php
        $username = "john";
        $password = "Xc3ffs";
        $messagetype = "SMS:TEXT";
        $httpUrl = "https://127.0.0.1:9508/";
        $recipient = urlencode("+36201324567");
        $messagedata = urlencode("TestMessage");

        $sendString = $httpUrl."api?action=sendmessage"."&username="
                     .$username."&password="
                     .$password."&recipient=".$recipient."&messagetype="
                     .$messagetype."&messagedata=".$messagedata;

        echo '<p><b> Sending html request:</b> '.$sendString.'</p>';
        $aContext = array(
            'http' => array(
                'method'  => 'GET',
            ),
            'ssl' => array(
                'verify_peer' => false,
                'verify_peer_name' => false,
            )
        );
        $cxContext = stream_context_create($aContext);
        $response = file_get_contents($sendString, true, $cxContext);

        echo '<p><b> Http response received :</b> </p>';
        echo '<xmp>' . $response. '</xmp>';
    ?>
</body>
</html>
```

Figure 3 - Copy code from website

## Step 4 - Paste the source code to a text file

You need to place the copied source code into a simple text file. To do that, first, you have to open a text editor on your computer. That can be the simple Notepad application or you can use a more developed text editor. After you opened your text editor and created a new, empty text file, just press Ctrl+V on your keyboard. As Figure 4 shows that, the source code will be pasted into the text document.

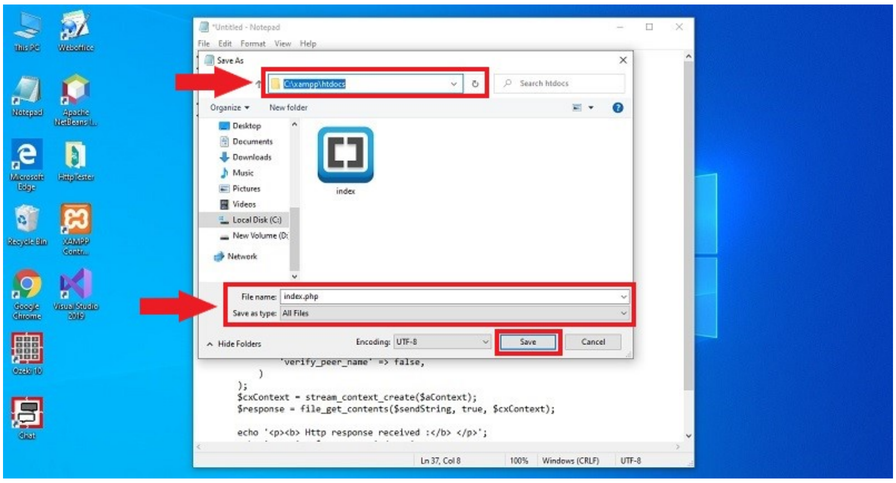


```
<!DOCTYPE html>
<html>
<head>
    <title>Http Request sending</title>
</head>
<body>
    <?php
        $username = "john";
        $password = "Xc3ffs";
        $messagetype = "SMS:TEXT";
        $httpUrl = "https://127.0.0.1:9508/";
        $recipient = urlencode("+36201324567");
        $messagedata = urlencode("TestMessage");

        $sendString = $httpUrl."api?action=sendmessage"."&username="
                     .$username."&password="
                     .$password."&recipient=".$recipient."&messagetype="
                     .$messagetype."&messagedata=".$messagedata;

        echo '<p><b> Sending html request:</b> '.$sendString.'</p>';
        $aContext = array(
            'http' => array(
                'method'  => 'GET',
            ),
            'ssl' => array(
                'verify_peer' => false,
                'verify_peer_name' => false,
            )
        );
        $cxContext = stream_context_create($aContext);
        $response = file_get_contents($sendString, true, $cxContext);

        echo '<p><b> Http response received :</b> </p>';
```

Figure 4 - Open Text Editor and Paste Code into

## Step 5 - Save the document as PHP file

At this point, you have got the document that you need, but it is not in the correct format. To correct that, you just need to save it with .php file extension. So, as Figure 5 shows that, save the text file into the xampp/htdocs folder. It is important to save the file into that folder, since the Apache server will be able to read the file from that folder. You also need to name the file as 'index.php'. That naming ensures that the file will be read automatically in case you load the local web service.



C:\xampp\htdocs

File name: index.php

Save as type: All Files

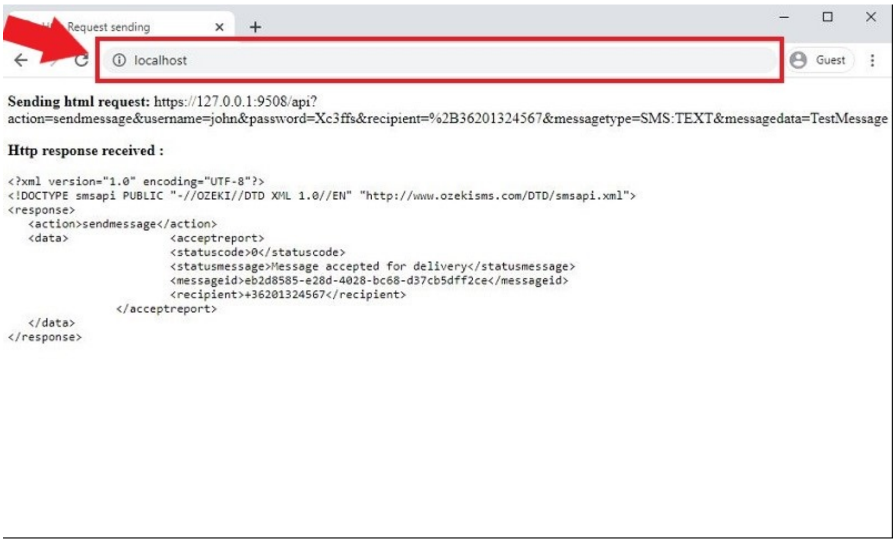
Encoding: UTF-8

```
            'verify_peer_name' => false,
        )
    );
    $cxContext = stream_context_create($aContext);
    $response = file_get_contents($sendString, true, $cxContext);

    echo '<p><b> Http response received :</b> </p>';
```

Figure 5 - Save the document as PHP file

# Step 6 - Open 'localhost' to run the example code

The last step of the guide is to open your localhost service. For that, just open your browser, type 'localhost' like in Figure 6, and hit Enter. By performing this action, the local web service loads the saved 'index.php' with the PHP SMS source code. The page that the service loaded prints the HTTP request and also prints the response message from the SMS Gateway that shows if the delivery of the SMS was successful.



**Sending html request:** https://127.0.0.1:9508/api?action=sendmessage&username=john&password=Xc3ffs&recipient=%2B36201324567&messagetype=SMS:TEXT&messagedata=TestMessage

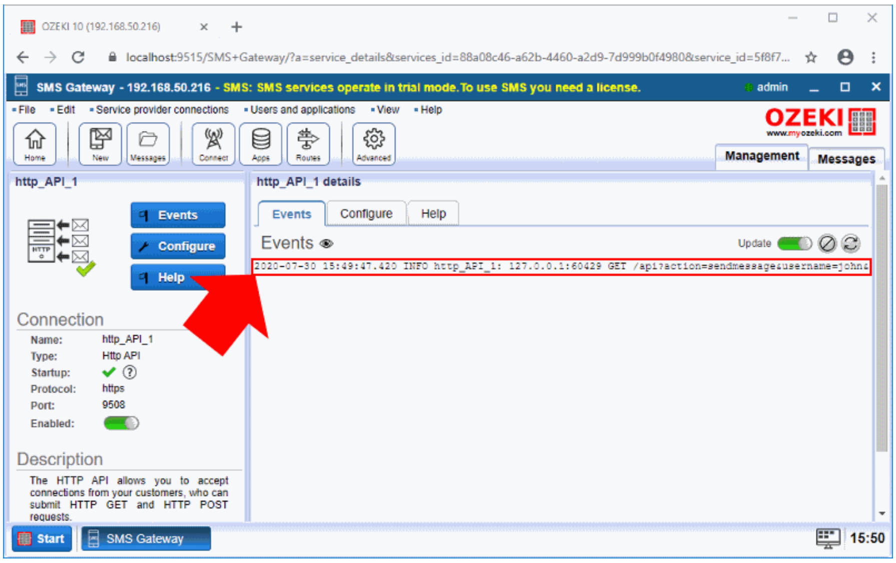
**Http response received :**

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE smsapi PUBLIC "-//OZEKI//DTD XML 1.0//EN" "http://www.ozekisms.com/DTD/smsapi.xml">
<response>
   <action>sendmessage</action>
   <data>                <acceptreport>
                         <statuscode>0</statuscode>
                         <statusmessage>Message accepted for delivery</statusmessage>
                         <messageid>eb2d8585-e28d-4028-bc68-d37cb5dff2ce</messageid>
                         <recipient>+36201324567</recipient>
                 </acceptreport>
   </data>
</response>
```

Figure 6 - Open localhost address in browser

# Step 7 - Check the send result in the Ozeki log

In Ozeki 10 SMS Gateway, you can check easily what messages sent by your application, since the HTTP API service logs every event that occured during the time it is enabled. So, if you open the SMS Gateway, and select details of the HTTP API service, you will be able to see the events. As you can see it in Figure 7, the service logged an event, when your web application sent the HTTP request to the service.



OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=service_details&services_id=88a08c46-a62b-4460-a2d9-7d999b0f4980&service_id=5f8f7...
SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode. To use SMS you need a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
http_API_1
Events
Configure
Help
http_API_1 details
Events
Configure
Help
Events
Update
Start
SMS Gateway
15:50

Connection

Name: http_API_1
Type: Http API
Startup: ✓
Protocol: https
Port: 9508
Enabled:

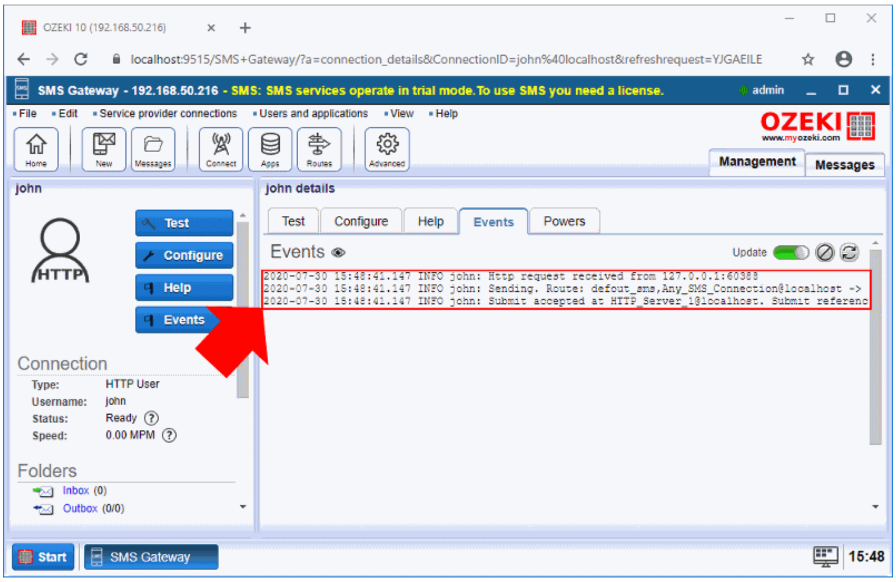
Description

The HTTP API allows you to accept connections from your customers, who can submit HTTP GET and HTTP POST requests.

```
2020-07-30 15:49:47.420 INFO http_API_1: 127.0.0.1:60429 GET /api?action=sendmessage&username=john&
```

Figure 7 - Check the logs of the HTTP API Service

The process of the message sending can be also viewed back by the events. To be able to see that, open the HTTP API User connection, that you had to configure before. Figure 8 shows that how the connection handles the HTTP request and send the message to the recipient that you specified in your PHP SMS example code.



OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=connection_details&ConnectionID=john%40localhost&refreshrequest=YJGAEILE
SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode. To use SMS you need a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
john
HTTP
Test
Configure
Help
Events
john details
Test
Configure
Help
Events
Powers
Events
Update
Start
SMS Gateway
15:48

Connection

Type: HTTP User
Username: john
Status: Ready
Speed: 0.00 MPM

Folders

Inbox (0)
Outbox (0/0)

```
2020-07-30 15:48:41.147 INFO john: Http request received from 127.0.0.1:60388
2020-07-30 15:48:41.147 INFO john: Sending. Route: defout_sms,Any_SMS_Connection@localhost ->
2020-07-30 15:48:41.147 INFO john: Submit accepted at HTTP_Server_1@localhost. Submit referenc
```

Figure 8 - Check the logs of the HTTP API User

# Receive SMS in PHP

***The following document is going to show how you can receive SMS messages with a web application or a website that uses PHP code. The PHP code initiates HTTP requests to the SMS Gateway to get the incoming messages. The code gets these messages as a response from the SMS Gateway, and then, you can use the messages in your web application.***

## What is a PHP SMS API?

The PHP SMS API ensures you to send SMS messages from a web application of a website. The API uses HTTP requests to communicate with the SMS Gateway that delivers the message and sends a response back.

## Prerequisites

- Installed XAMPP
- Installed Ozeki 10 SMS Gateway
- A configured HTTP API User

## Receive SMS in PHP

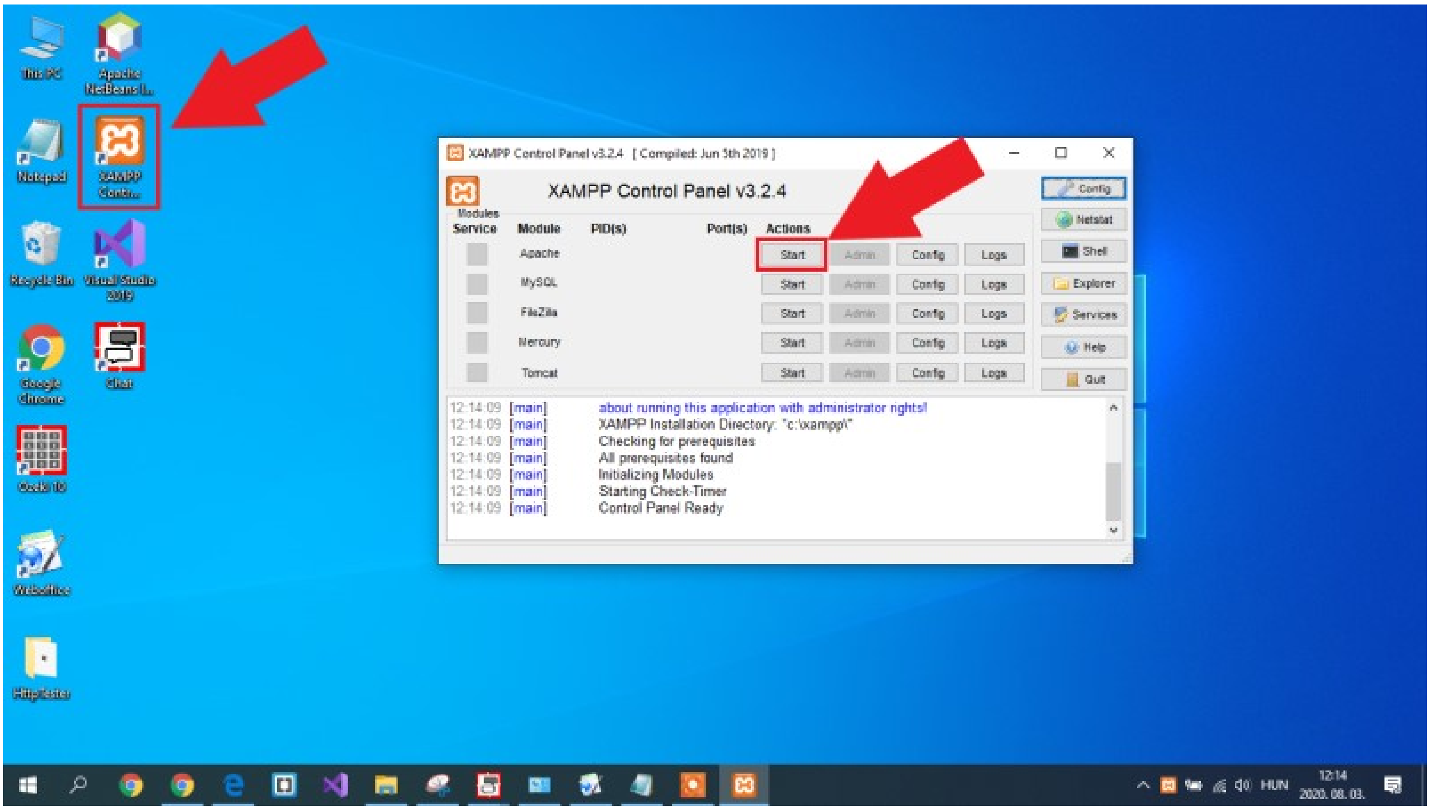
1. Open XAMPP Control Panel
2. Start Apache service
3. Open Notepad
4. Copy-Paste the PHP source code below
5. Save the file to the xampp/htmldocs folder as index.php
6. Open SMS Gateway and select HTTP Server connection
7. Send some test messages
8. Type 'localhost' in your browser to run the PHP code

The PHP example code below can get the received messages from the SMS Gateway. This code is free to use in your web application or website, and you can modify it if you want to. The step by step guide below and the video on this page shows how you can use the example code, if you would like to just run the PHP code on your computer and test the solution.

```
<!DOCTYPE html>
<html>
<head>
    <title>Http Request sending</title>
</head>
<body>
    <?php
        $username = "john";
        $password = "Xc3ffs";
        $httpUrl = "https://127.0.0.1:9508/";
        $folder = "inbox";
        $limit = "3";

        $sendString = $httpUrl."api?action=receivemessage&username="
                     .$username."&password=".$password."&folder="
                     .$folder."&limit=".$limit
                     ."&messagedata=&afterdownload=delete";

        echo '<p><b> Sending html request:</b> '.$sendString.'</p>';
        $aContext = array(
            'http' => array(
                'method'  => 'GET',
            ),
            'ssl' => array(
                'verify_peer' => false,
                'verify_peer_name' => false,
            )
        );
        $cxContext = stream_context_create($aContext);
        $response = file_get_contents($sendString, true, $cxContext);

        echo '<p><b> Http response received :</b> </p>';
        DisplayMessages($response);


        function DisplayMessages($response){
            $xml = simplexml_load_string($response);
            if($xml -> data -> message == "No more messages."){
                echo '<p><b>The inbox is empty</b></p>';
                return;
            }

            foreach ($xml -> data -> message as $value) {
                $sender  = $value -> originator;
                $text = $value -> messagedata;
                DisplayMessage($sender, $text);
            }
        }

        function DisplayMessage($sender, $text){
            echo '<p><b>'.$sender.': '.$text.'</b></p>';
        }
    ?>
</body>
</html>
```

## Step 1 - Open XAMPP Control Panel

The first step of the guide is to set up an Apache server on your computer which is capable of executing PHP codes. These localhost servers can ve set up easily using XAMPP. If you haven't got this application yet, check the Prerequisites section on this page to download it. If you have already installed it on your computer, you just need to open the XAMPP Control Panel as you can see it in Figure 1. Here, you need to start the Apache service.

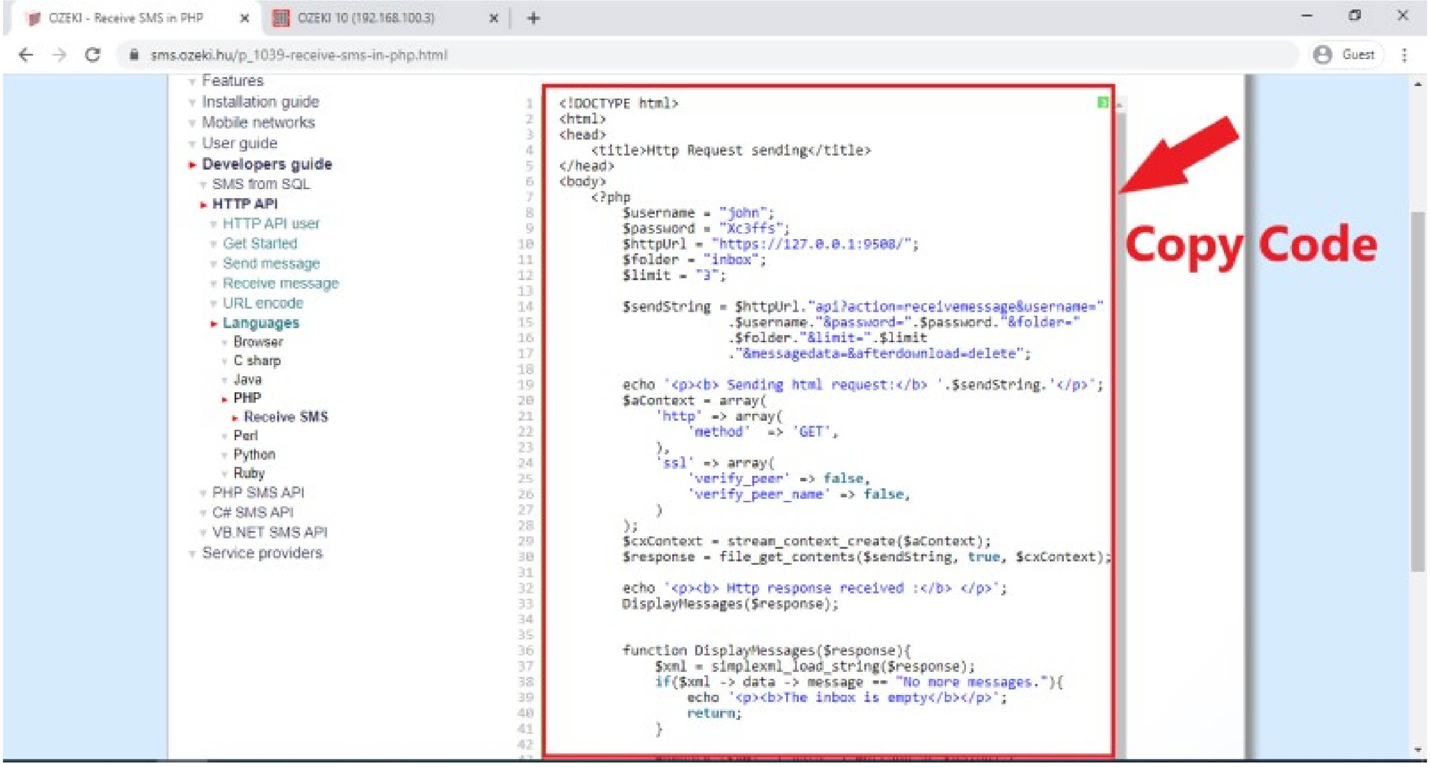


```
12:14:09  [main]   about running this application with administrator rights!
12:14:09  [main]   XAMPP Installation Directory: "c:\xampp\"
12:14:09  [main]   Checking for prerequisites
12:14:09  [main]   All prerequisites found
12:14:09  [main]   Initializing Modules
12:14:09  [main]   Starting Check-Timer
12:14:09  [main]   Control Panel Ready
```

Figure 1 - Open XAMPP control panel, then start Apache server

## Step 2 - Copy PHP code from this page

The next step of this guide is to get the code from this page to use it in your solution. For that, just go to the example code on this page, and mark out whole source code. Then, press Ctrl+C on your keyboard as you can see it in Figure 2 to copy the code to your clipboard.

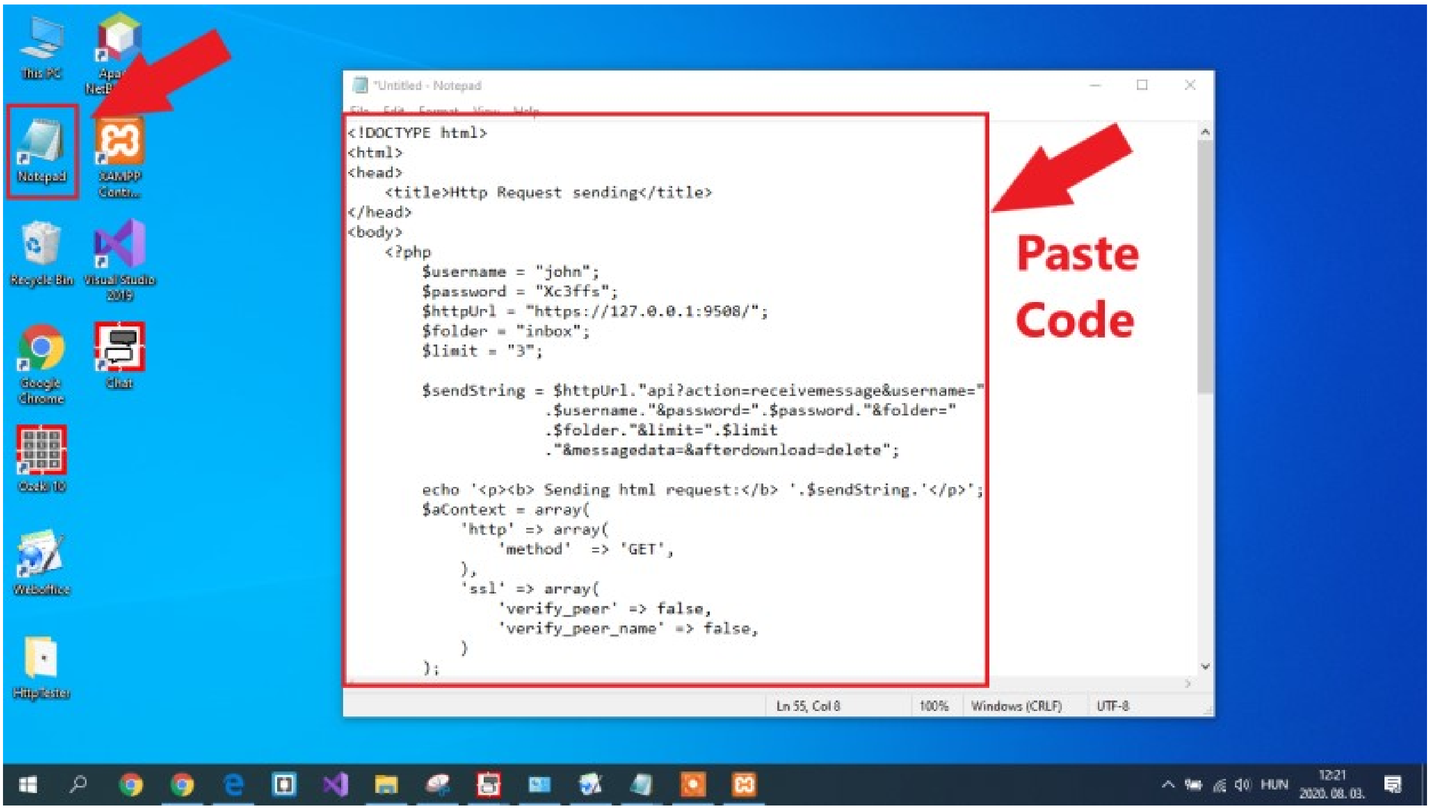


```
<!DOCTYPE html>
<html>
<head>
    <title>Http Request sending</title>
</head>
<body>
    <?php
        $username = "john";
        $password = "Xc3ffs";
        $httpUrl = "https://127.0.0.1:9508/";
        $folder = "inbox";
        $limit = "3";

        $sendString = $httpUrl."api?action=receivemessage&username="
                        .$username."&password=".$password."&folder="
                        .$folder."&limit=".$limit
                        ."&messagedata=&afterdownload=delete";

        echo '<p><b> Sending html request:</b> '.$sendString.'</p>';
        $aContext = array(
            'http' => array(
                'method'  => 'GET',
            ),
            'ssl' => array(
                'verify_peer' => false,
                'verify_peer_name' => false,
            )
        );
        $cxContext = stream_context_create($aContext);
        $response = file_get_contents($sendString, true, $cxContext);

        echo '<p><b> Http response received :</b> </p>';
        DisplayMessages($response);


        function DisplayMessages($response){
            $xml = simplexml_load_string($response);
            if($xml -> data -> message == "No more messages."){
                echo '<p><b>The inbox is empty</b></p>';
                return;
            }

```

Figure 2 - Copy php source code from the website

## Step 3 - Paste the code into a text document

To run the code on your local server, you need a file that contains the PHP code. So, open the Notepad application and create a new text file. Here, just press Ctrl+V on your keyboard to paste the source code into that text document as Figure 3 demonstrates it. Lastly, you need to save the file to the xampp/htdocs folder and name the file as 'index.php' to be able to run it from your local server.

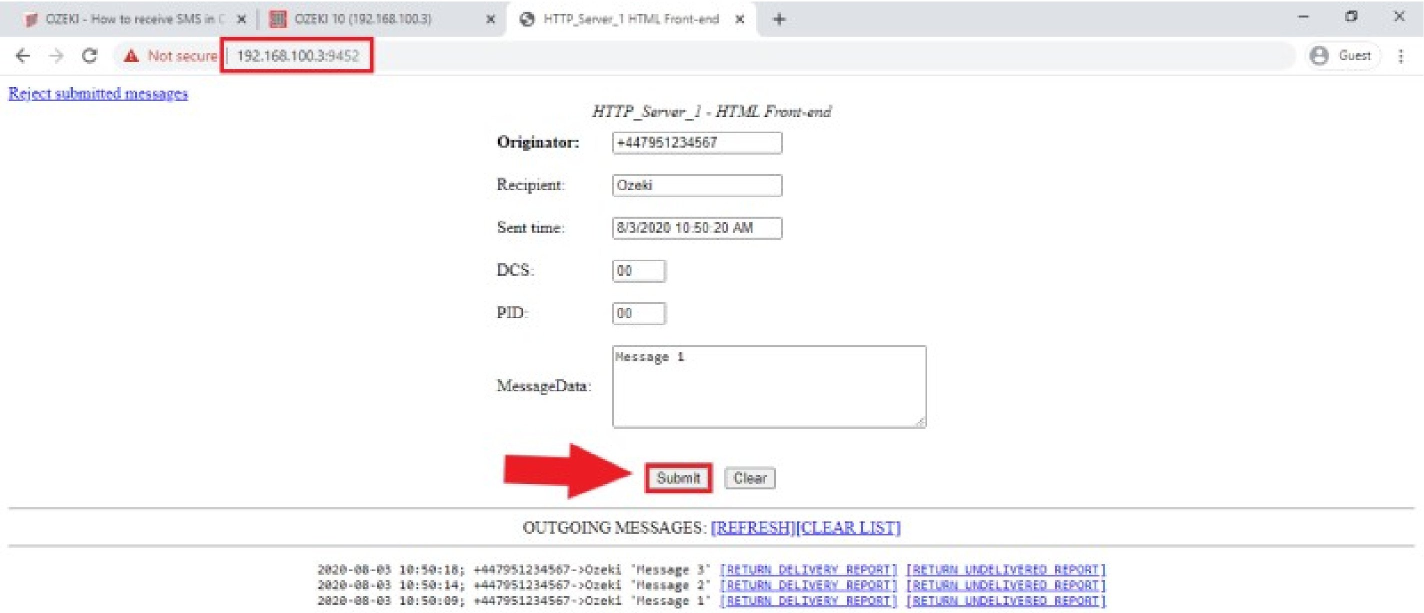


```
<!DOCTYPE html>
<html>
<head>
    <title>Http Request sending</title>
</head>
<body>
    <?php
        $username = "john";
        $password = "Xc3ffs";
        $httpUrl = "https://127.0.0.1:9508/";
        $folder = "inbox";
        $limit = "3";

        $sendString = $httpUrl."api?action=receivemessage&username="
                        .$username."&password=".$password."&folder="
                        .$folder."&limit=".$limit
                        ."&messagedata=&afterdownload=delete";

        echo '<p><b> Sending html request:</b> '.$sendString.'</p>';
        $aContext = array(
            'http' => array(
                'method'  => 'GET',
            ),
            'ssl' => array(
                'verify_peer' => false,
                'verify_peer_name' => false,
            )
        );
```

Figure 3 - Open Notepad and paste php code from the website

## Step 4 - Send some test messages

Now, your example code is up and ready to run, but first, you need to have some messages in your inbox folder. For testing purposes, you can use the HTTP Server connection to send some test messages to yourself and test the PHP example code. So, open the SMS Gateway and select the HTTP Server connection. Here, open the HTML form of the connection, and like in Figure 4, send some messages to the 'Ozeki' recipient.


OZEKI - How to receive SMS in C
OZEKI 10 (192.168.100.3)
HTTP_Server_1 HTML Front-end
Not secure
192.168.100.3:9452
Guest
Reject submitted messages
Submit
Clear

*HTTP_Server_1 - HTML Front-end*

**Originator:** +447951234567

Recipient: Ozeki

Sent time: 8/3/2020 10:50:20 AM

DCS: 00

PID: 00

MessageData: Message 1

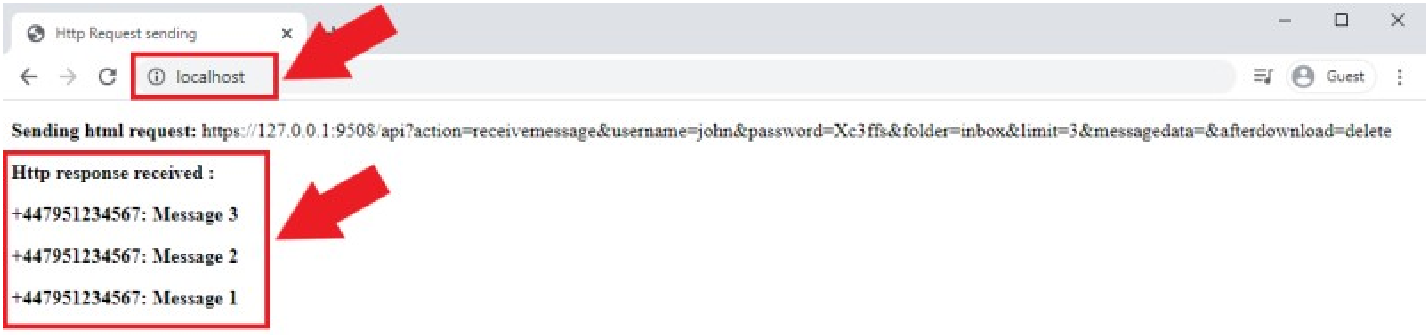
OUTGOING MESSAGES: [REFRESH][CLEAR LIST]

```
2020-08-03 10:50:18; +447951234567->Ozeki 'Message 3' [RETURN DELIVERY REPORT] [RETURN UNDELIVERED REPORT]
2020-08-03 10:50:14; +447951234567->Ozeki 'Message 2' [RETURN DELIVERY REPORT] [RETURN UNDELIVERED REPORT]
2020-08-03 10:50:09; +447951234567->Ozeki 'Message 1' [RETURN DELIVERY REPORT] [RETURN UNDELIVERED REPORT]
```

Figure 4 - Simulate some incoming SMS

## Step 5 - Run the PHP code

The last step is to test your solution and the PHP code. To do that, open your web browser, and type 'localhost' as an address as you can see it in Figure 5. If you press Enter, the PHP example code executes and you will be able to see the result in the browser. This shows the HTTP request that was sent to the SMS Gateway and the response message that lists all SMS received by the SMS Gateway with the phone number of the sender and the text of the message as well.



```
Sending html request: https://127.0.0.1:9508/api?action=receivemessage&username=john&password=Xc3ffs&folder=inbox&limit=3&messagedata=&afterdownload=delete

Http response received :

+447951234567: Message 3

+447951234567: Message 2

+447951234567: Message 1
```

Figure 5 - Run php code in browser

# How to send a scheduled SMS

***This guide gives information on how to send a scheduled SMS message, and how to get reports from Ozeki SMS Gateway when the message is submitted to the mobile network and when it is delivered to the recipient handset.***

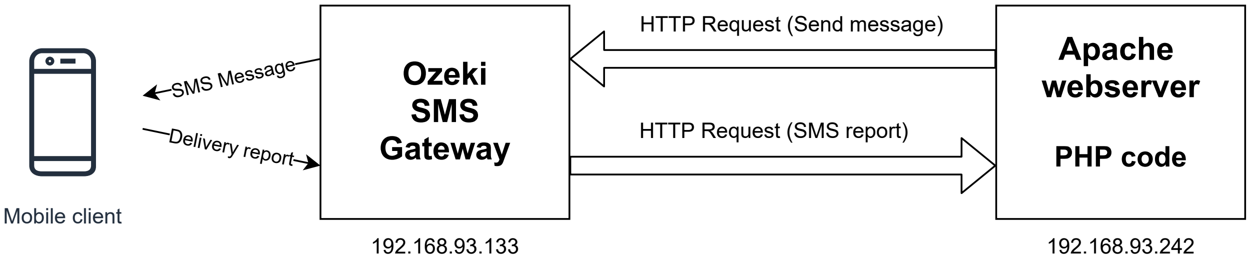
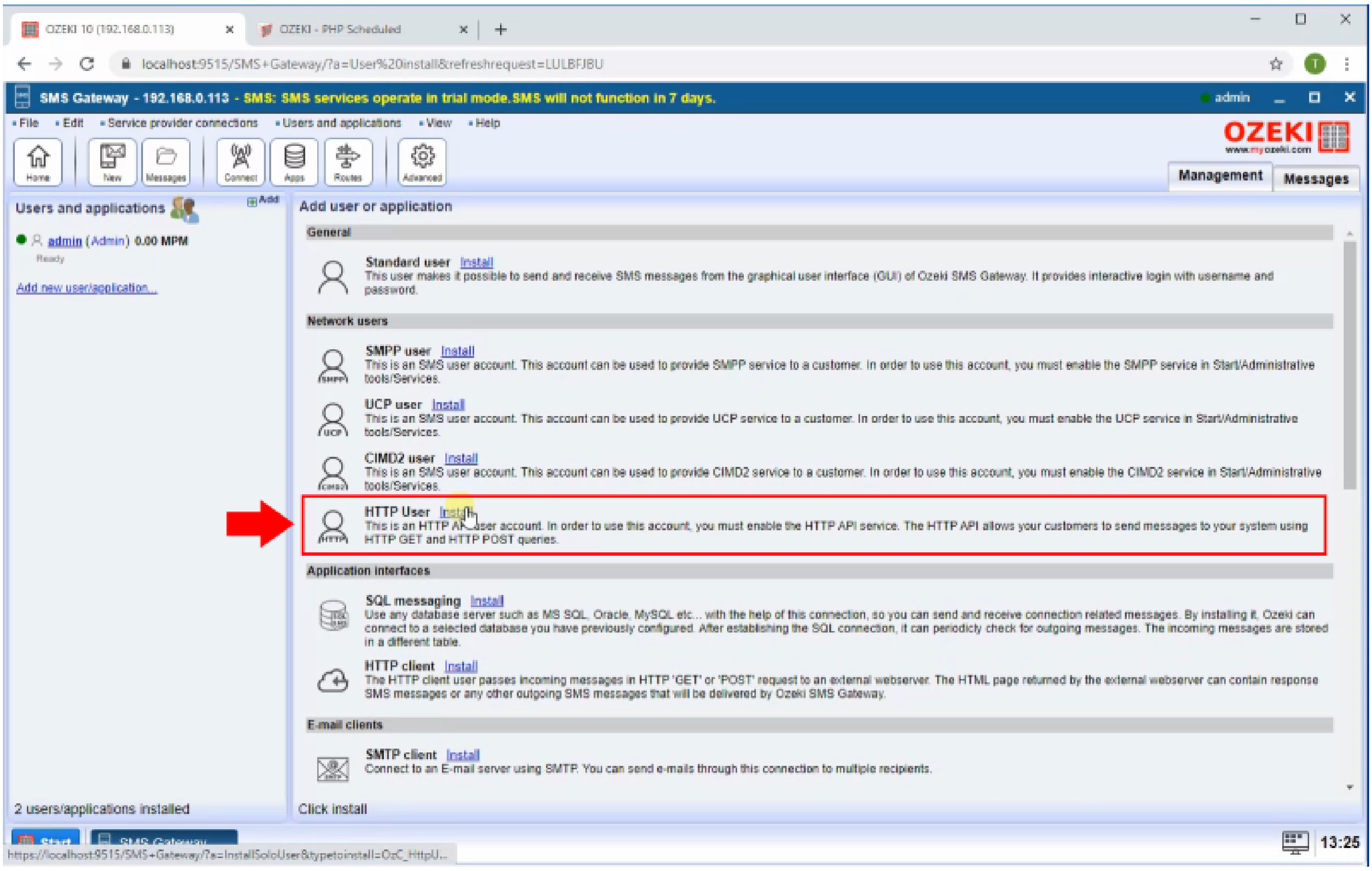


Figure 1 - Scheduled SMS from Apache webserver

## Step 1 - Install HTTP API User

You can simply install the HTTP API User on the Management console by clicking Add new user/application... in the Users/Applications panel. An interface will open consisting of two panels. The right side panel contains the users and applications you can install with a brief description next to them. Find the HTTP API User and click the blue 'install' button next to it (Figure 2).


OZEKI 10 (192.168.0.113)
OZEKI - PHP Scheduled
localhost:9515/SMS+Gateway/?a=User%20install&refreshrequest=LULBFJBU
SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
OZEKI
www.myozeki.com
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Users and applications
Add
admin (Admin) 0.00 MPM
Ready
Add new user/application...
Add user or application
General
Network users
Application interfaces
E-mail clients
2 users/applications installed
Click install
Start
SMS Gateway
13:25
https://localhost:9515/SMS+Gateway/?a=InstallSoloUser&typetoinstall=OzC_HttpU...

Standard user Install
This user makes it possible to send and receive SMS messages from the graphical user interface (GUI) of Ozeki SMS Gateway. It provides interactive login with username and password.

SMPP user Install
This is an SMS user account. This account can be used to provide SMPP service to a customer. In order to use this account, you must enable the SMPP service in Start/Administrative tools/Services.

UCP user Install
This is an SMS user account. This account can be used to provide UCP service to a customer. In order to use this account, you must enable the UCP service in Start/Administrative tools/Services.

CIMD2 user Install
This is an SMS user account. This account can be used to provide CIMD2 service to a customer. In order to use this account, you must enable the CIMD2 service in Start/Administrative tools/Services.

HTTP User Install
This is an HTTP API user account. In order to use this account, you must enable the HTTP API service. The HTTP API allows your customers to send messages to your system using HTTP GET and HTTP POST queries.

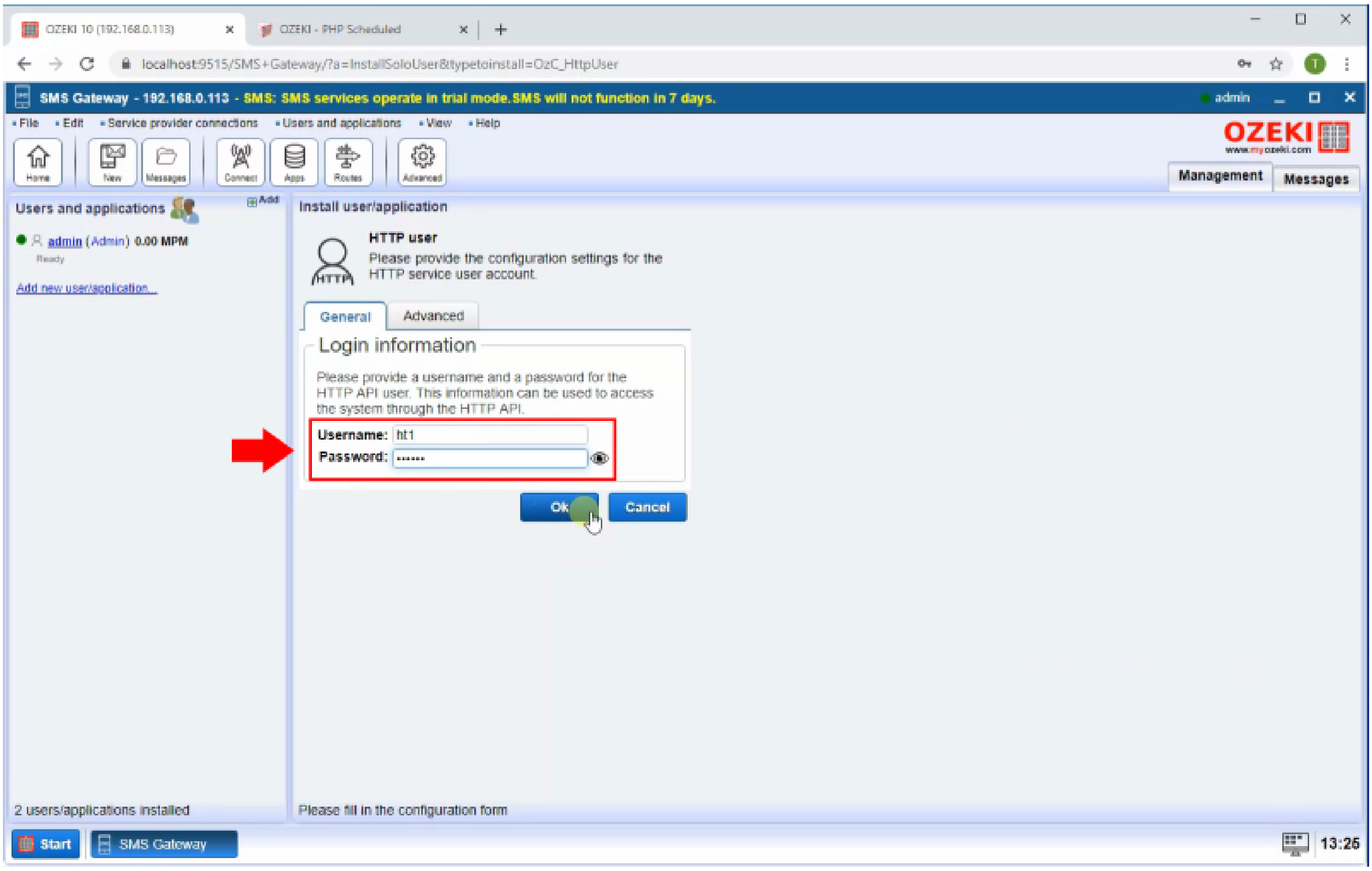
SQL messaging Install
Use any database server such as MS SQL, Oracle, MySQL etc... with the help of this connection, so you can send and receive connection related messages. By installing it, Ozeki can connect to a selected database you have previously configured. After establishing the SQL connection, it can periodicly check for outgoing messages. The incoming messages are stored in a different table.

HTTP client Install
The HTTP client user passes incoming messages in HTTP 'GET' or 'POST' request to an external webserver. The HTML page returned by the external webserver can contain response SMS messages or any other outgoing SMS messages that will be delivered by Ozeki SMS Gateway.

SMTP client Install
Connect to an E-mail server using SMTP. You can send e-mails through this connection to multiple recipients.

Figure 2 - Install HTTP API user

On the Name section provide the unique name for the HTTP API User and provide the username and password for the authentication (Figure 3).



Install user/application

HTTP user

Please provide the configuration settings for the HTTP service user account.

Login information

Please provide a username and a password for the HTTP API user. This information can be used to access the system through the HTTP API.

Username: ht1

Password: ••••••

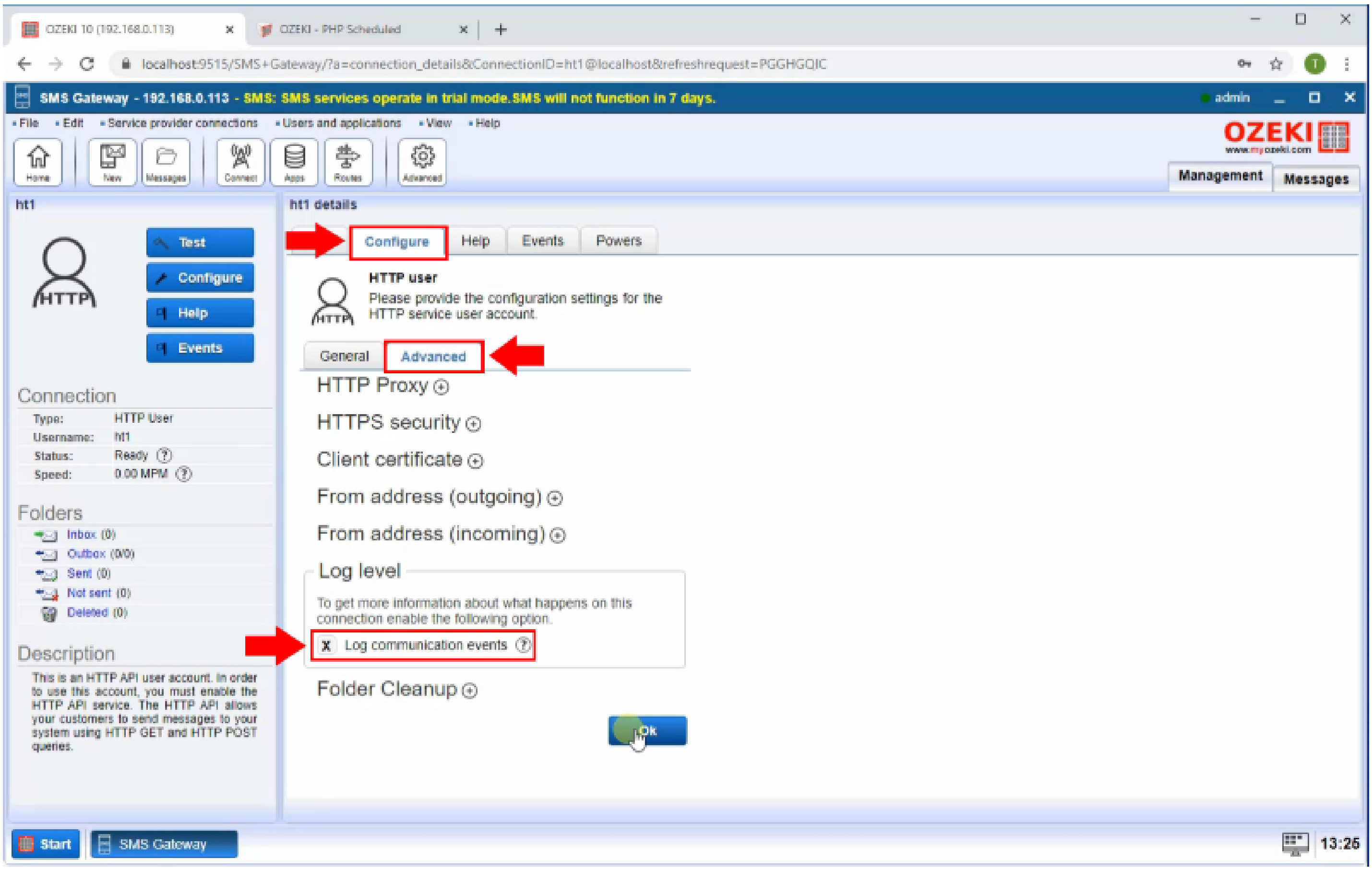
Please fill in the configuration form

SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode. SMS will not function in 7 days.
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Users and applications
admin (Admin) 0.00 MPM
Add new user/application...
General
Advanced
Ok
Cancel
2 users/applications installed
Start
SMS Gateway
13:25

Figure 3 - Define username and password

# Step 2 - Enable Log communication events

In order to able to see the HTTP communication in the HTTP API user Events tab you need to enable log communication events. To make this. open the HTTP user Configure tab. Under it in the Advanced tab Log level section check the Log communication events option as the Figure 4 shows.



HTTP user

Please provide the configuration settings for the HTTP service user account.

Connection

Type: HTTP User
Username: ht1
Status: Ready
Speed: 0.00 MPM

Folders

Inbox (0)
Outbox (0/0)
Sent (0)
Not sent (0)
Deleted (0)

Description

This is an HTTP API user account. In order to use this account, you must enable the HTTP API service. The HTTP API allows your customers to send messages to your system using HTTP GET and HTTP POST queries.

HTTP Proxy ⊕

HTTPS security ⊕

Client certificate ⊕

From address (outgoing) ⊕

From address (incoming) ⊕

Log level

To get more information about what happens on this connection enable the following option.

☒ Log communication events

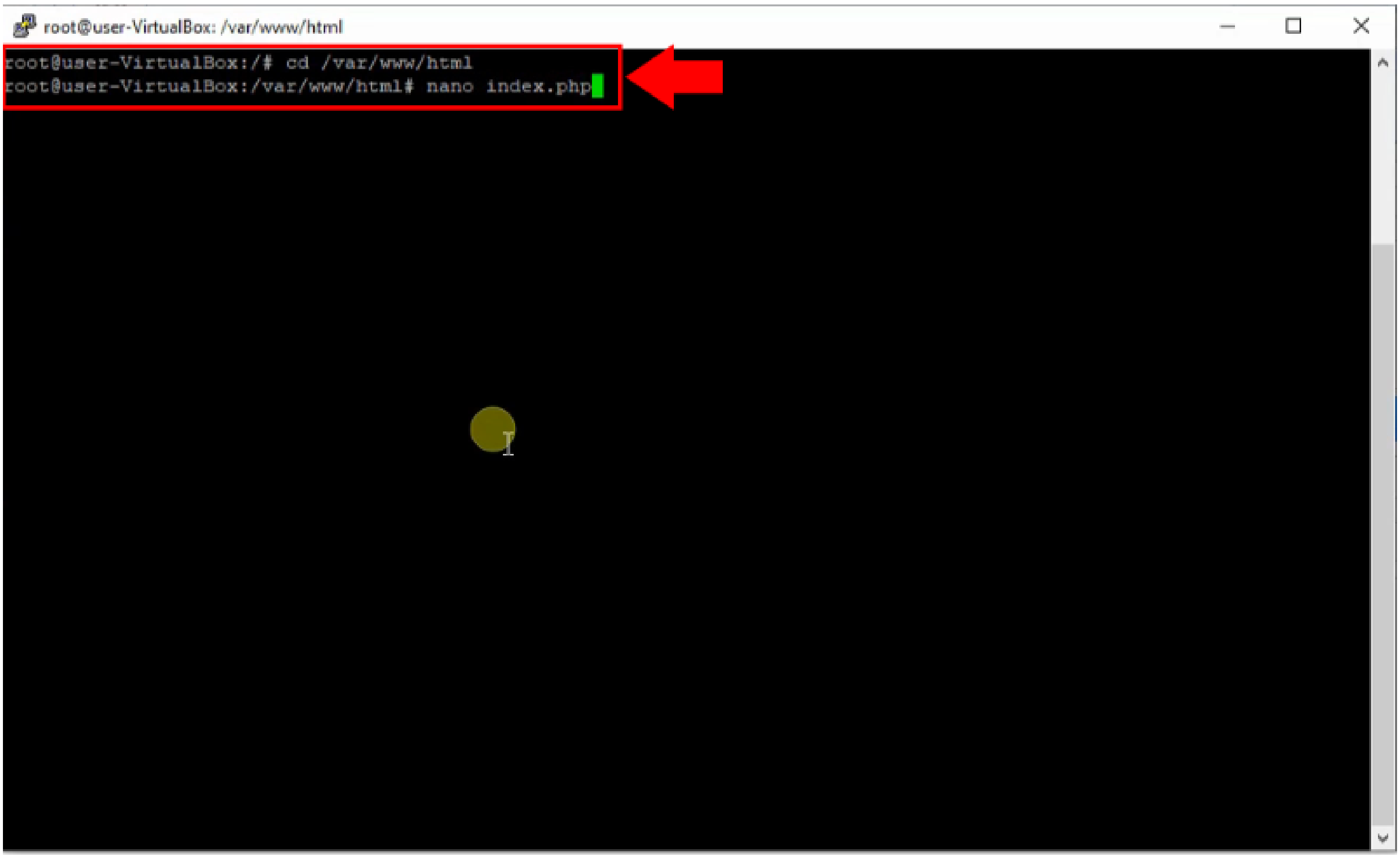
Folder Cleanup ⊕

Ok

Figure 4 - Enable log communication

# Step 3 - Create PHP Code for send message

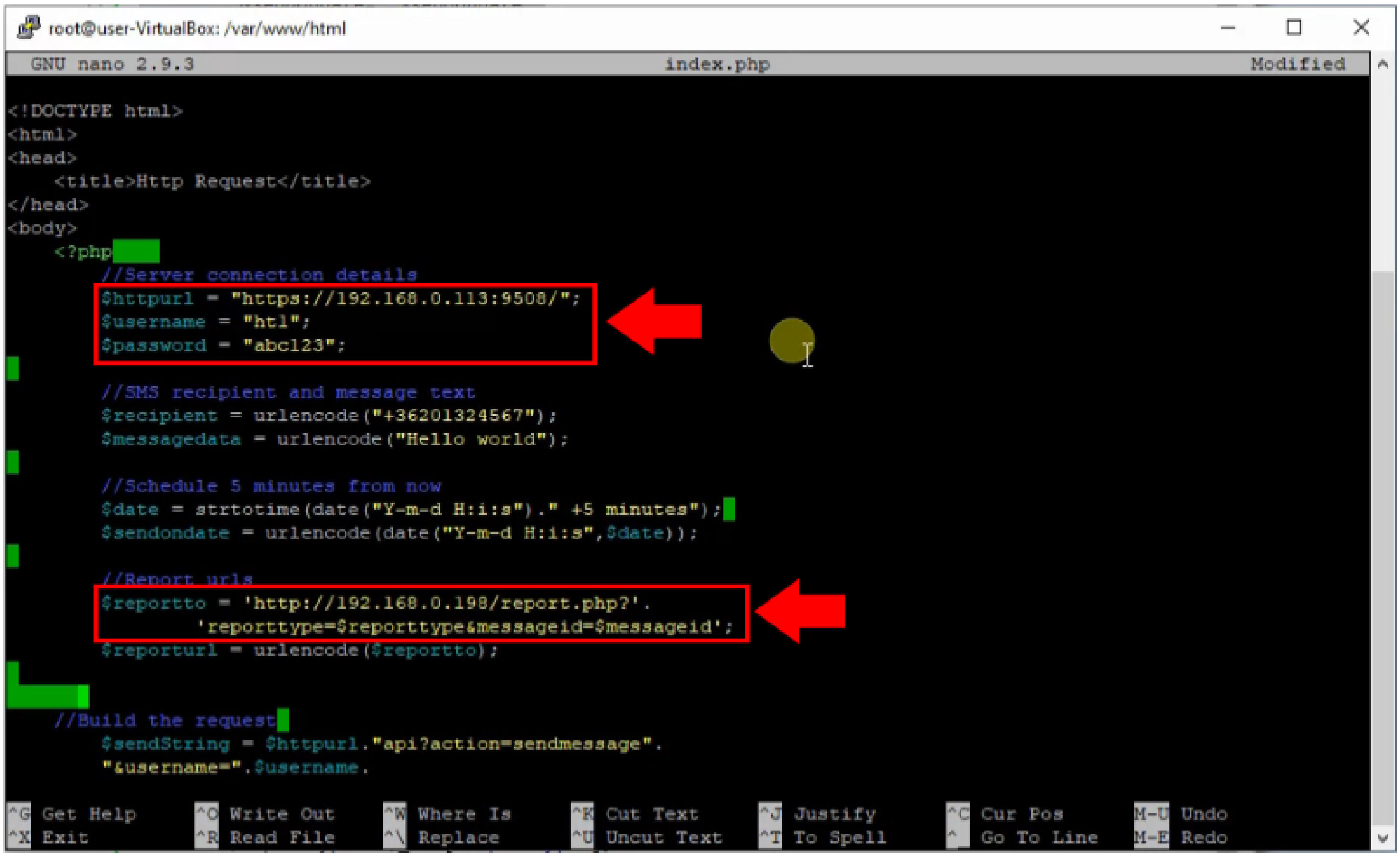
The next step is to create the php files in the Apache web server www root. Go to the /var/www/html folder and create the index.php file (Figure 5).



```
root@user-VirtualBox:/# cd /var/www/html
root@user-VirtualBox:/var/www/html# nano index.php
```

Figure 5 - Create index.php

After you pasted the php code modify the Server conncetion, change the URL, username and password to your SMS Gateway IP and HTTP API user username and password. Then modify the Report URL-s IP to the Apache webserver IP (Figure 6).



```
root@user-VirtualBox: /var/www/html
  GNU nano 2.9.3                         index.php                         Modified

<!DOCTYPE html>
<html>
<head>
    <title>Http Request</title>
</head>
<body>
    <?php
        //Server connection details
        $httpurl = "https://192.168.0.113:9508/";
        $username = "ht1";
        $password = "abc123";

        //SMS recipient and message text
        $recipient = urlencode("+36201324567");
        $messagedata = urlencode("Hello world");

        //Schedule 5 minutes from now
        $date = strtotime(date("Y-m-d H:i:s")." +5 minutes");
        $sendondate = urlencode(date("Y-m-d H:i:s",$date));

        //Report urls
        $reportto = 'http://192.168.0.198/report.php?'.
               'reporttype=$reporttype&messageid=$messageid';
        $reporturl = urlencode($reportto);

    //Build the request
        $sendString = $httpurl."api?action=sendmessage".
        "&username=".$username.

^G Get Help    ^O Write Out   ^W Where Is    ^K Cut Text    ^J Justify     ^C Cur Pos     M-U Undo
^X Exit        ^R Read File   ^\ Replace     ^U Uncut Text  ^T To Spell    ^_ Go To Line  M-E Redo
```

Figure 6 - Configure index.php details

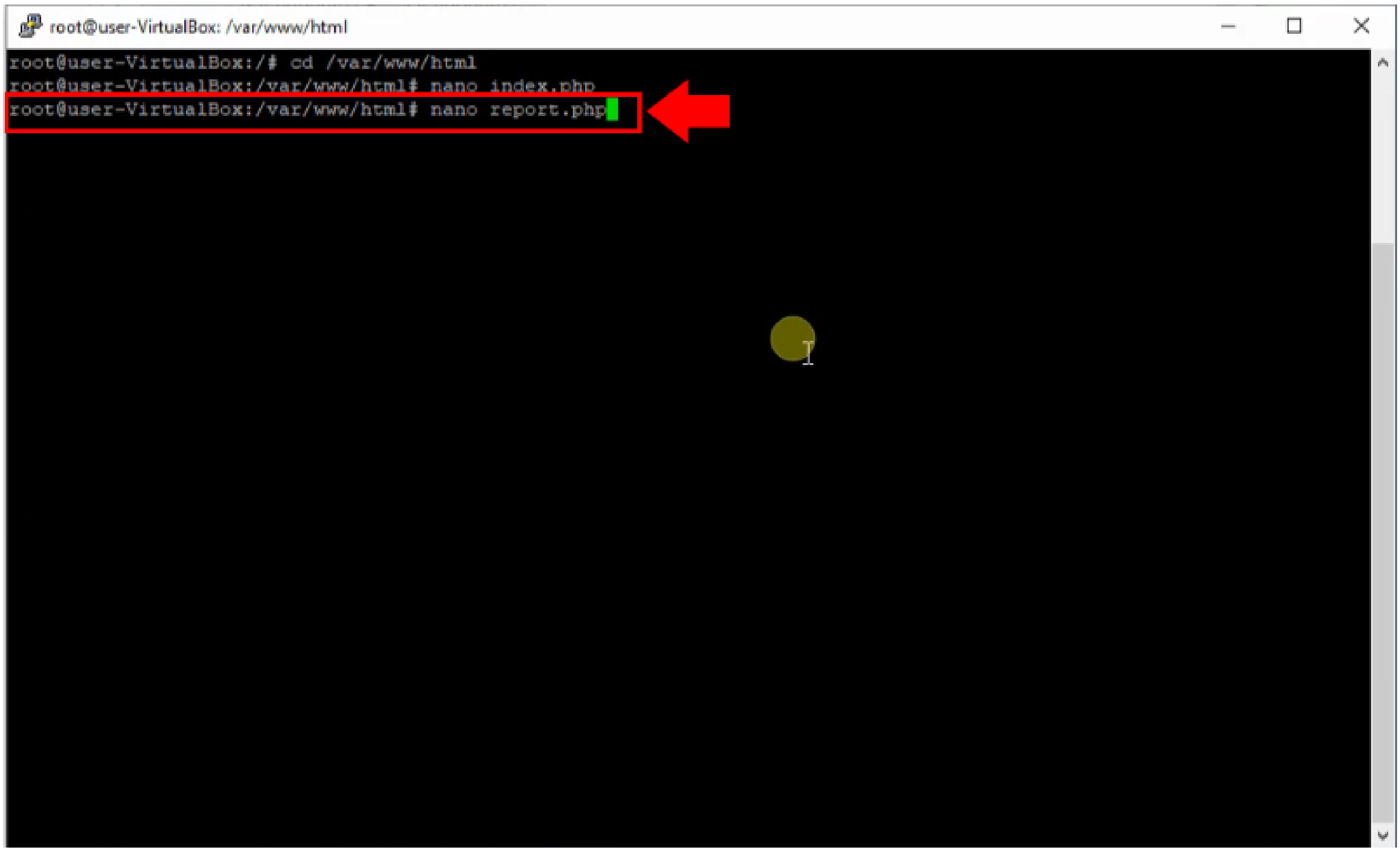
# Example code to submit a scheduled SMS

```
<!DOCTYPE html>
<html>
<head>
    <title>Http Request</title>
</head>
<body>
    <?php
        //Server connection details
        $httpurl = "https://192.168.93.133:9509/";
        $username = "ht1";
        $password = "qwe123";

        //SMS recipient and message text
        $recipient = urlencode("+36201324567");
        $messagedata = urlencode("Hello world");

        //Schedule 5 minutes from now
        $date = strtotime(date("Y-m-d H:i:s")." +5 minutes");
        $sendondate = urlencode(date("Y-m-d H:i:s",$date));

        //Report urls
        $reportto = 'http://192.168.93.242/report.php?'.
                'reporttype=$reporttype&messageid=$messageid';
        $reporturl = urlencode($reportto);


    //Build the request
        $sendString = $httpurl."api?action=sendmessage".
        "&username=".$username.
        "&password=".$password.
        "&recipient=".$recipient.
        "&recipient=".$recipient.
        "&sendondate=".$sendondate.
```

```
                "&reporturl=".$reporturl.
                "&messagedata=".$messagedata;

        echo '<p><b> Sending http request:</b><br> '.$sendString.'</p>';
        $aContext = array(
            'http' => array(
                'method'  => 'GET',
            ),
            'ssl' => array(
                'verify_peer' => false,
                'verify_peer_name' => false,
            )
        );
        $cxContext = stream_context_create($aContext);
        $response = file_get_contents($sendString, true, $cxContext);

        echo '<p><b> Http response received :</b> </p>';
        echo '<xmp>' . $response. '</xmp>';
    ?>
</body>
</html>
```

## Step 4 - Create PHP Code for message reports

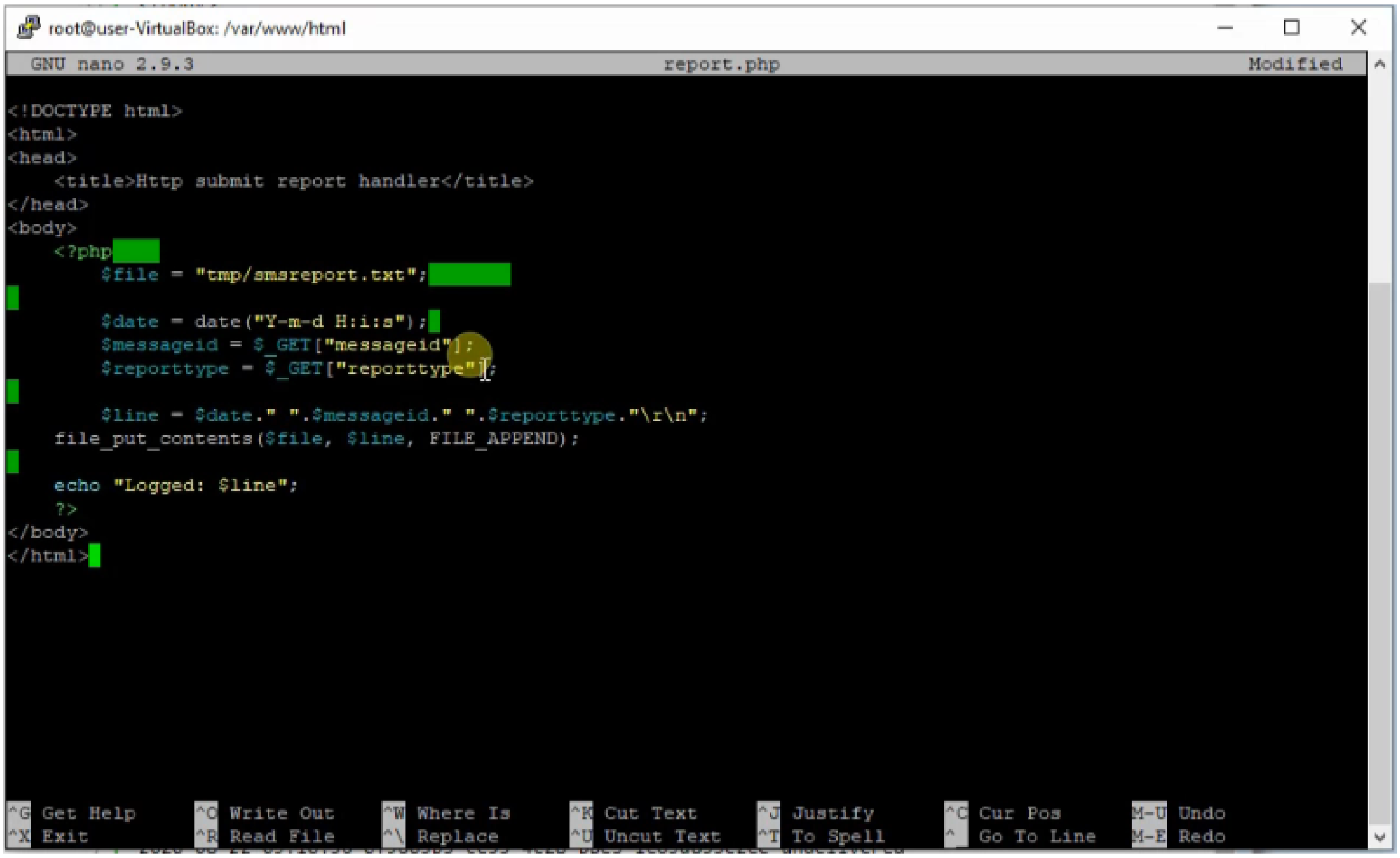
Now create the report php file in the Apache web server www root. In the /var/www/html folder create the report.php file as you can see in the Figure 7.



```
root@user-VirtualBox:/# cd /var/www/html
root@user-VirtualBox:/var/www/html# nano index.php
root@user-VirtualBox:/var/www/html# nano report.php
```

Figure 7 - Create report.php

In the report php file paste the below php code for the SMS report receiving. This code saves all the received SMS report into an smsreport.txt file under the tmp folder (Figure 8).



```
<!DOCTYPE html>
<html>
<head>
    <title>Http submit report handler</title>
</head>
<body>
    <?php
        $file = "tmp/smsreport.txt";

        $date = date("Y-m-d H:i:s");
        $messageid = $_GET["messageid"];
        $reporttype = $_GET["reporttype"];

        $line = $date." ".$messageid." ".$reporttype."\r\n";
    file_put_contents($file, $line, FILE_APPEND);

    echo "Logged: $line";
    ?>
</body>
</html>
```

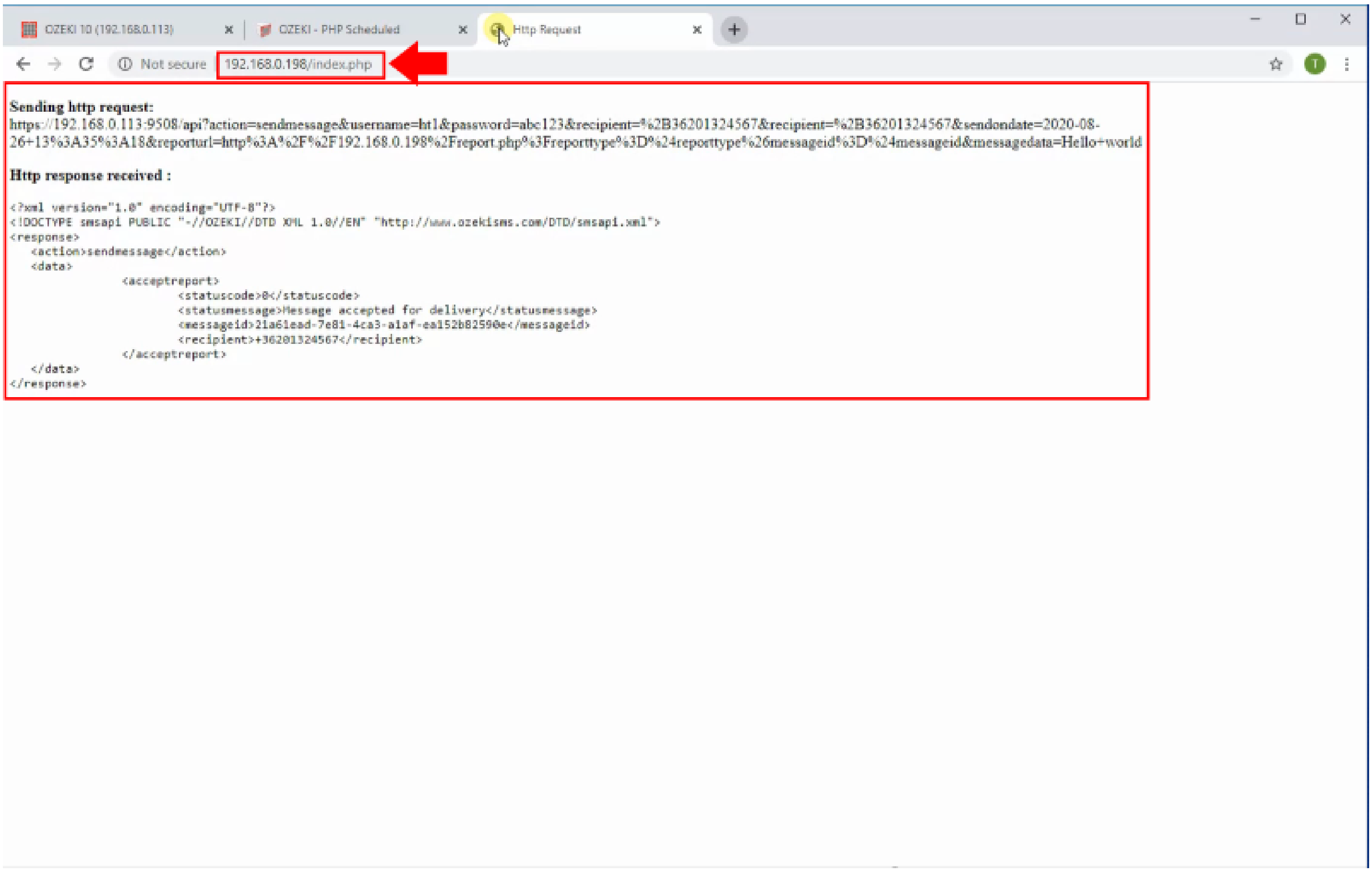
Figure 8 - Report.php details

## Example code to process incoming reports

```
<!DOCTYPE html>
<html>
<head>
    <title>Http submit report handler</title>
</head>
<body>
    <?php
        $file = "tmp/smsreport.txt";

        $date = date("Y-m-d H:i:s");
        $messageid = $_GET["messageid"];
        $reporttype = $_GET["reporttype"];

        $line = $date." ".$messageid." ".$reporttype."\r\n";
    file_put_contents($file, $line, FILE_APPEND);

    echo "Logged: $line";
    ?>
</body>
</html>
```

## Step 5 - Send SMS Message

After the PHP files are created in the Apache server you are able tu send SMS using it. In a browser open the index.php and you will see the sent http request and the SMS gateway's HTTP response (Figure 9).



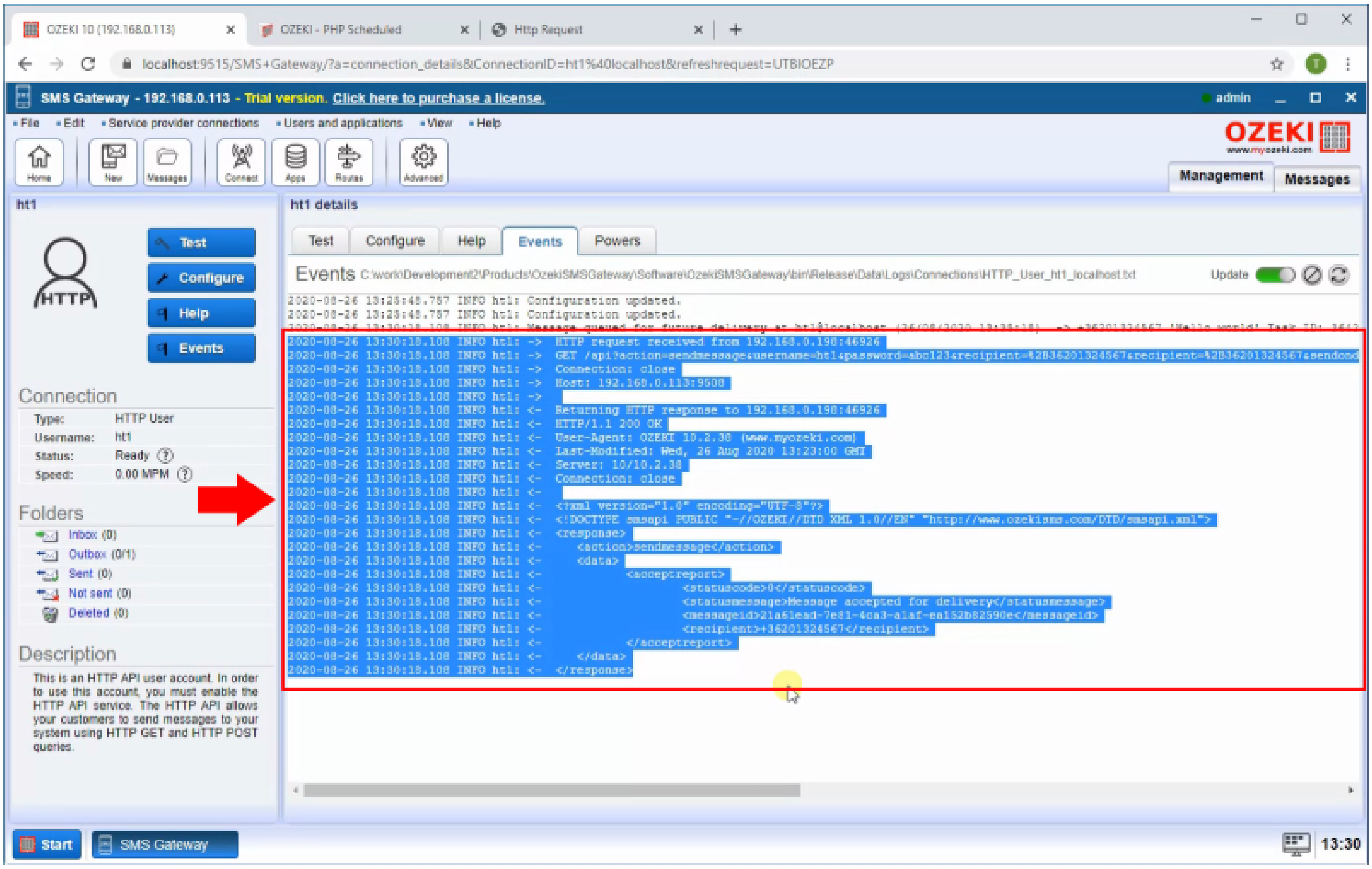
**Sending http request:**
https://192.168.0.113:9508/api?action=sendmessage&username=ht1&password=abc123&recipient=%2B36201324567&recipient=%2B36201324567&sendondate=2020-08-26+13%3A35%3A18&reporturl=http%3A%2F%2F192.168.0.198%2Freport.php%3Freporttype%3D%24reporttype%26messageid%3D%24messageid&messagedata=Hello+world

**Http response received :**

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE smsapi PUBLIC "-//OZEKI//DTD XML 1.0//EN" "http://www.ozekisms.com/DTD/smsapi.xml">
<response>
   <action>sendmessage</action>
   <data>
              <acceptreport>
                     <statuscode>0</statuscode>
                     <statusmessage>Message accepted for delivery</statusmessage>
                     <messageid>21a61ead-7e81-4ca3-a1af-ea152b82590e</messageid>
                     <recipient>+36201324567</recipient>
              </acceptreport>
   </data>
</response>
```

Figure 9 - Send SMS message using index.php

In the Ozeki SMS Gateway events tab you can also see the http communication (Figure 10).



Connection

Type: HTTP User
Username: ht1
Status: Ready
Speed: 0.00 MPM

Description

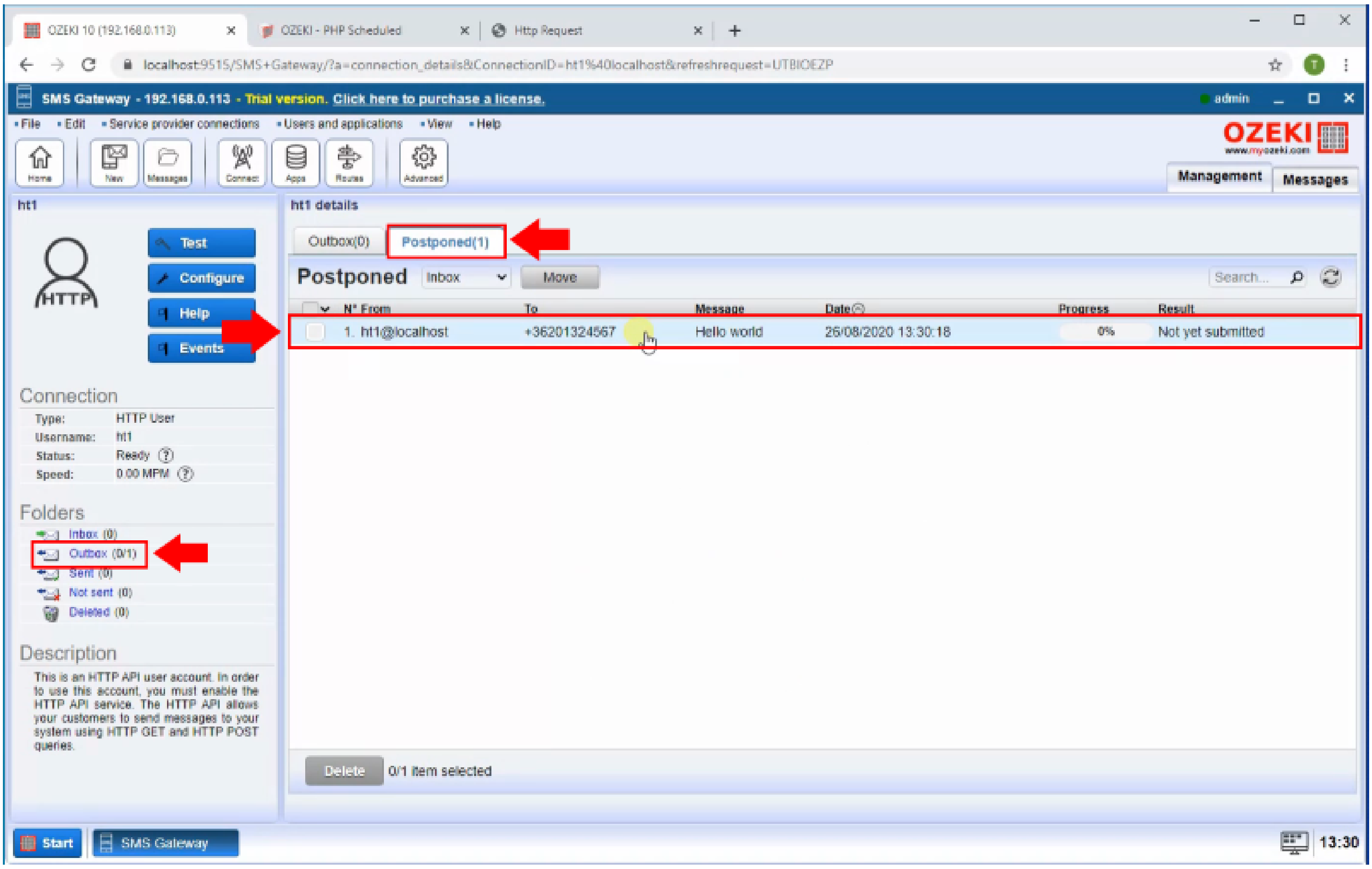
This is an HTTP API user account. In order to use this account, you must enable the HTTP API service. The HTTP API allows your customers to send messages to your system using HTTP GET and HTTP POST queries.

Events C:\work\Development2\Products\OzekiSMSGateway\Software\OzekiSMSGateway\bin\Release\Data\Logs\Connections\HTTP_User_ht1_localhost.txt

```
2020-08-26 13:28:48.787 INFO ht1: Configuration updated.
2020-08-26 13:28:48.787 INFO ht1: Configuration updated.
2020-08-26 13:30:18.108 INFO ht1: Message queued for future delivery at ht1@localhost (26/08/2020 13:35:18) -> +36201324567 'Hello world!' Task ID: 3642
2020-08-26 13:30:18.108 INFO ht1: ->  HTTP request received from 192.168.0.198:46926
2020-08-26 13:30:18.108 INFO ht1: ->  GET /api?action=sendmessage&username=ht1&password=abc123&recipient=%2B36201324567&recipient=%2B36201324567&sendond
2020-08-26 13:30:18.108 INFO ht1: ->  Connection: close
2020-08-26 13:30:18.108 INFO ht1: ->  Host: 192.168.0.113:9508
2020-08-26 13:30:18.108 INFO ht1: ->
2020-08-26 13:30:18.108 INFO ht1: <-  Returning HTTP response to 192.168.0.198:46926
2020-08-26 13:30:18.108 INFO ht1: <-  HTTP/1.1 200 OK
2020-08-26 13:30:18.108 INFO ht1: <-  User-Agent: OZEKI 10.2.38 (www.myozeki.com)
2020-08-26 13:30:18.108 INFO ht1: <-  Last-Modified: Wed, 26 Aug 2020 13:23:00 GMT
2020-08-26 13:30:18.108 INFO ht1: <-  Server: 10/10.2.38
2020-08-26 13:30:18.108 INFO ht1: <-  Connection: close
2020-08-26 13:30:18.108 INFO ht1: <-
2020-08-26 13:30:18.108 INFO ht1: <-  <?xml version="1.0" encoding="UTF-8"?>
2020-08-26 13:30:18.108 INFO ht1: <-  <!DOCTYPE smsapi PUBLIC "-//OZEKI//DTD XML 1.0//EN" "http://www.ozekisms.com/DTD/smsapi.xml">
2020-08-26 13:30:18.108 INFO ht1: <-  <response>
2020-08-26 13:30:18.108 INFO ht1: <-     <action>sendmessage</action>
2020-08-26 13:30:18.108 INFO ht1: <-     <data>
2020-08-26 13:30:18.108 INFO ht1: <-            <acceptreport>
2020-08-26 13:30:18.108 INFO ht1: <-                   <statuscode>0</statuscode>
2020-08-26 13:30:18.108 INFO ht1: <-                   <statusmessage>Message accepted for delivery</statusmessage>
2020-08-26 13:30:18.108 INFO ht1: <-                   <messageid>21a61ead-7e81-4ca3-a1af-ea152b82590e</messageid>
2020-08-26 13:30:18.108 INFO ht1: <-                   <recipient>+36201324567</recipient>
2020-08-26 13:30:18.108 INFO ht1: <-            </acceptreport>
2020-08-26 13:30:18.108 INFO ht1: <-     </data>
2020-08-26 13:30:18.108 INFO ht1: <-  </response>
```

Figure 10 - Message received by SMS Gateway

If you open the Ozeki SMS Gateway outbox folder you are able to see thet the message is stored in the Postponed folder because it will be send in the future (Figure 11).



SMS Gateway - 192.168.0.113 - Trial version. Click here to purchase a license.

localhost:9515/SMS+Gateway/?a=connection_details&ConnectionID=ht1%40localhost&refreshrequest=UTBIOEZP

ht1 details

Postponed

| N° | From | To | Message | Date | Progress | Result |
|---|---|---|---|---|---|---|
| 1. | ht1@localhost | +36201324567 | Hello world | 26/08/2020 13:30:18 | 0% | Not yet submitted |

0/1 item selected

Connection

Type: HTTP User
Username: ht1
Status: Ready
Speed: 0.00 MPM

Folders

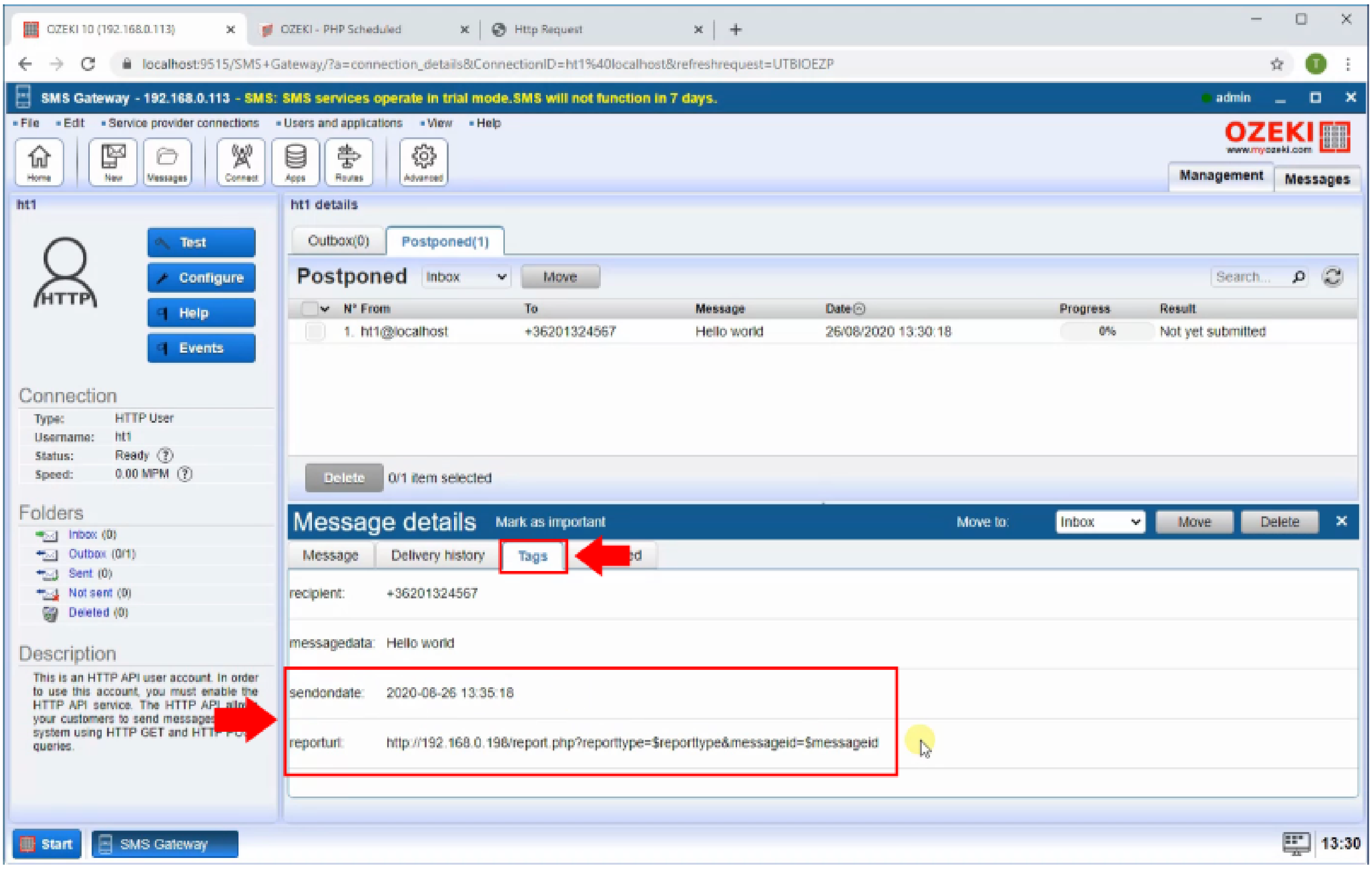
Inbox (0)
Outbox (0/1)
Sent (0)
Not sent (0)
Deleted (0)

Description

This is an HTTP API user account. In order to use this account, you must enable the HTTP API service. The HTTP API allows your customers to send messages to your system using HTTP GET and HTTP POST queries.

Figure 11 - Message in outbox

In the message details Tags tab you can check when will the SMS be send and the report URL what will call the Ozeki SMS Gateway if the SMS is sent.



SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.

localhost:9515/SMS+Gateway/?a=connection_details&ConnectionID=ht1%40localhost&refreshrequest=UTBIOEZP

ht1 details

Postponed

| N° | From | To | Message | Date | Progress | Result |
|---|---|---|---|---|---|---|
| 1. | ht1@localhost | +36201324567 | Hello world | 26/08/2020 13:30:18 | 0% | Not yet submitted |

0/1 item selected

Message details

recipient: +36201324567
messagedata: Hello world
sendondate: 2020-08-26 13:35:18
reporturl: http://192.168.0.198/report.php?reporttype=$reporttype&messageid=$messageid

Connection

Type: HTTP User
Username: ht1
Status: Ready
Speed: 0.00 MPM

Folders

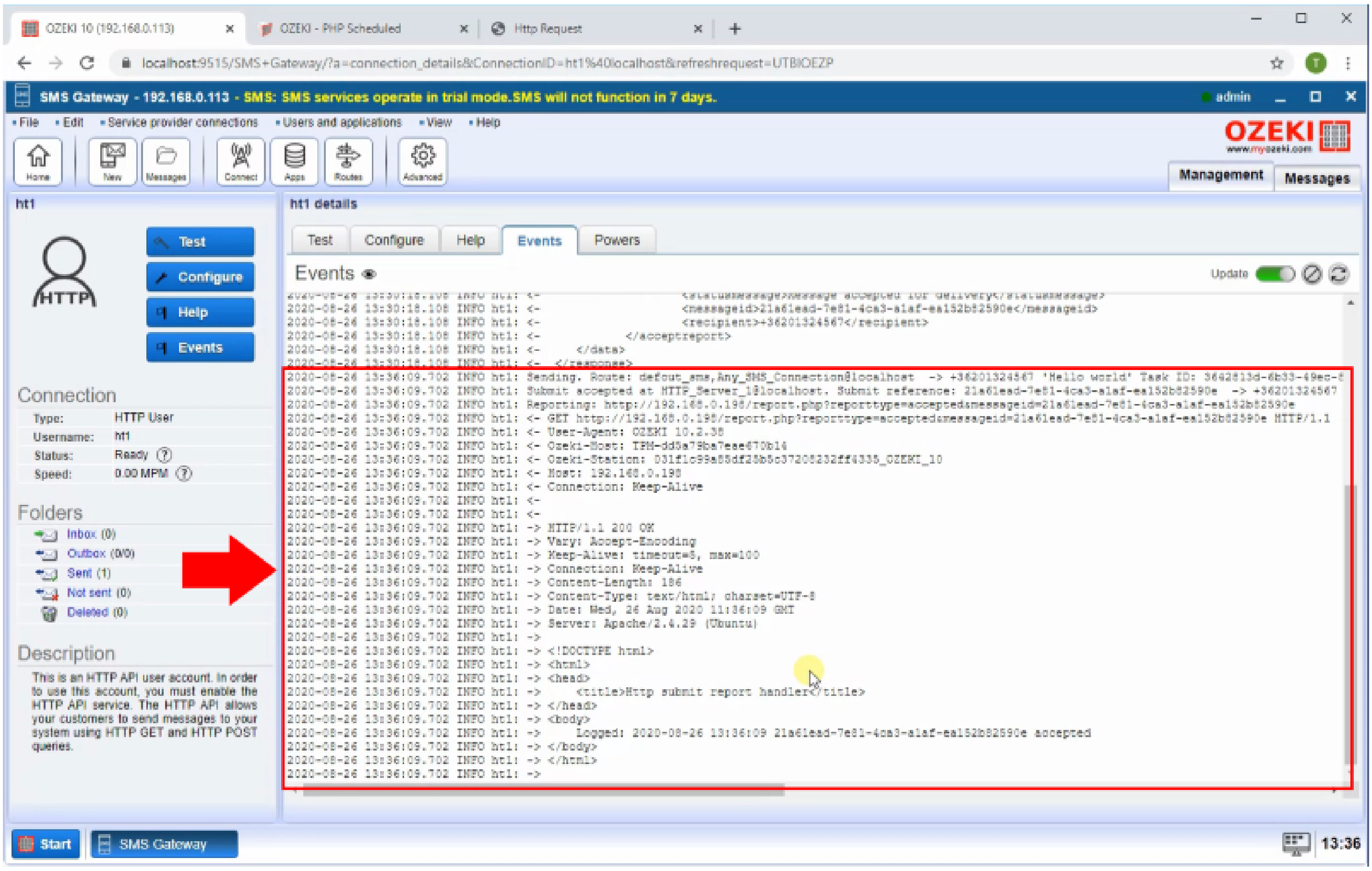
Inbox (0)
Outbox (0/1)
Sent (0)
Not sent (0)
Deleted (0)

Description

This is an HTTP API user account. In order to use this account, you must enable the HTTP API service. The HTTP API allows your customers to send messages to your system using HTTP GET and HTTP POST queries.

Figure 12 - Message tags

Finally you will see the Ozeki SMS Gateway is send the message on the send on date time as you can see it in the Figure 13.



Connection
Type: HTTP User
Username: ht1
Status: Ready
Speed: 0.00 MPM

Folders
Inbox (0)
Outbox (0/0)
Sent (1)
Not sent (0)
Deleted (0)

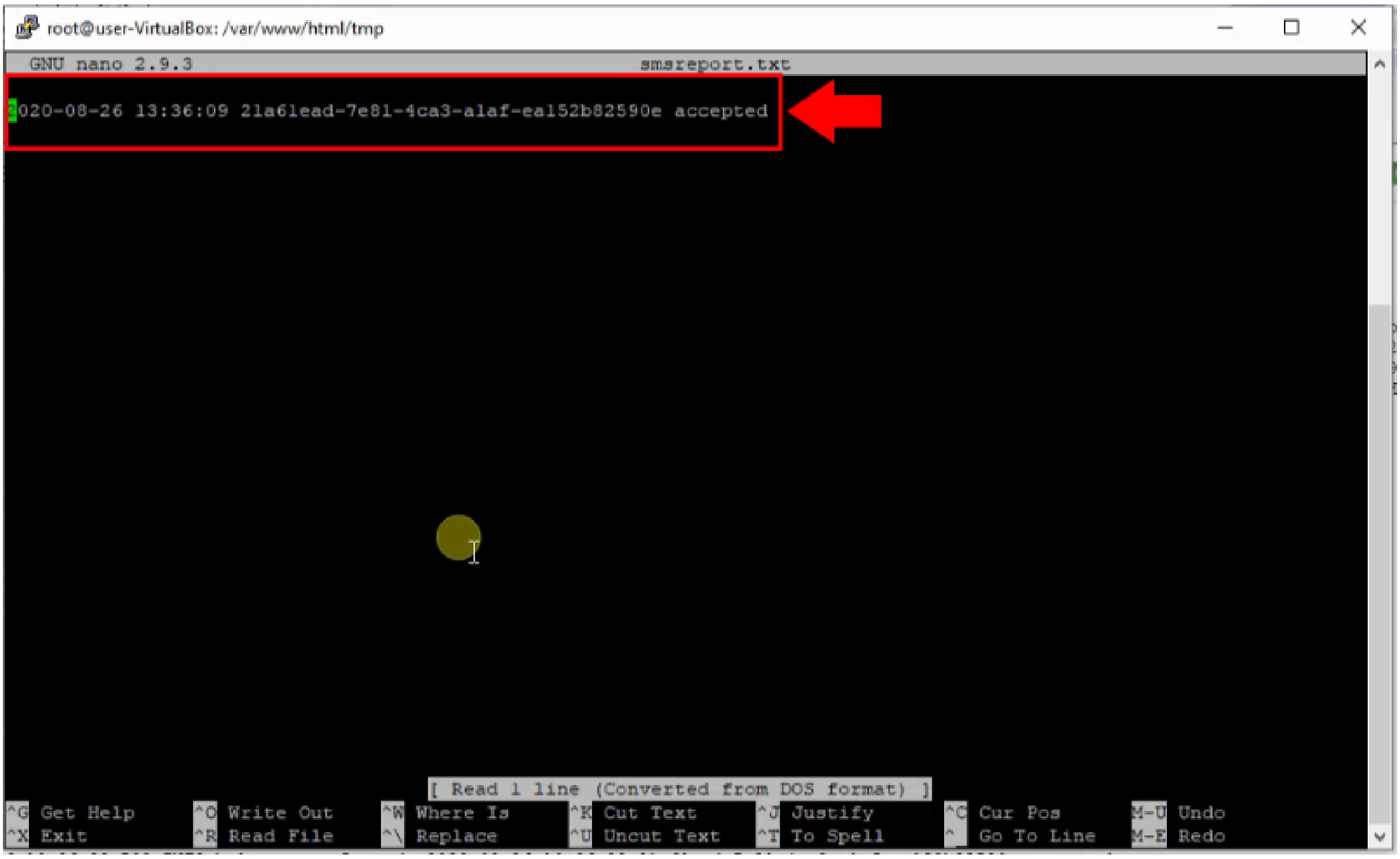
Description
This is an HTTP API user account. In order to use this account, you must enable the HTTP API service. The HTTP API allows your customers to send messages to your system using HTTP GET and HTTP POST queries.

```
2020-08-26 13:30:18.108 INFO ht1: <-                <statusmessage>Message accepted for delivery</statusmessage>
2020-08-26 13:30:18.108 INFO ht1: <-                <messageid>21a61ead-7e81-4ca3-a1af-ea152b82590e</messageid>
2020-08-26 13:30:18.108 INFO ht1: <-                <recipient>+36201324567</recipient>
2020-08-26 13:30:18.108 INFO ht1: <-            </acceptreport>
2020-08-26 13:30:18.108 INFO ht1: <-        </data>
2020-08-26 13:30:18.108 INFO ht1: <-    </response>
2020-08-26 13:36:09.702 INFO ht1: Sending. Route: defout_sms,Any_SMS_Connection@localhost  -> +36201324567 'Hello world' Task ID: 3642813d-6b33-49ec-8
2020-08-26 13:36:09.702 INFO ht1: Submit accepted at HTTP_Server_1@localhost. Submit reference: 21a61ead-7e81-4ca3-a1af-ea152b82590e  -> +36201324567
2020-08-26 13:36:09.702 INFO ht1: Reporting: http://192.168.0.198/report.php?reporttype=accepted&messageid=21a61ead-7e81-4ca3-a1af-ea152b82590e
2020-08-26 13:36:09.702 INFO ht1: <- GET http://192.168.0.198/report.php?reporttype=accepted&messageid=21a61ead-7e81-4ca3-a1af-ea152b82590e HTTP/1.1
2020-08-26 13:36:09.702 INFO ht1: <- User-Agent: OZEKI 10.2.38
2020-08-26 13:36:09.702 INFO ht1: <- Ozeki-Host: TPM-dd5a79ba7eae670b14
2020-08-26 13:36:09.702 INFO ht1: <- Ozeki-Station: 031f1c99a88df28b5c37208232ff4338_OZEKI_10
2020-08-26 13:36:09.702 INFO ht1: <- Host: 192.168.0.198
2020-08-26 13:36:09.702 INFO ht1: <- Connection: Keep-Alive
2020-08-26 13:36:09.702 INFO ht1: <-
2020-08-26 13:36:09.702 INFO ht1: <-
2020-08-26 13:36:09.702 INFO ht1: -> HTTP/1.1 200 OK
2020-08-26 13:36:09.702 INFO ht1: -> Vary: Accept-Encoding
2020-08-26 13:36:09.702 INFO ht1: -> Keep-Alive: timeout=5, max=100
2020-08-26 13:36:09.702 INFO ht1: -> Connection: Keep-Alive
2020-08-26 13:36:09.702 INFO ht1: -> Content-Length: 186
2020-08-26 13:36:09.702 INFO ht1: -> Content-Type: text/html; charset=UTF-8
2020-08-26 13:36:09.702 INFO ht1: -> Date: Wed, 26 Aug 2020 11:36:09 GMT
2020-08-26 13:36:09.702 INFO ht1: -> Server: Apache/2.4.29 (Ubuntu)
2020-08-26 13:36:09.702 INFO ht1: ->
2020-08-26 13:36:09.702 INFO ht1: -> <!DOCTYPE html>
2020-08-26 13:36:09.702 INFO ht1: -> <html>
2020-08-26 13:36:09.702 INFO ht1: -> <head>
2020-08-26 13:36:09.702 INFO ht1: ->     <title>Http submit report handler</title>
2020-08-26 13:36:09.702 INFO ht1: -> </head>
2020-08-26 13:36:09.702 INFO ht1: -> <body>
2020-08-26 13:36:09.702 INFO ht1: ->     Logged: 2020-08-26 13:36:09 21a61ead-7e81-4ca3-a1af-ea152b82590e accepted
2020-08-26 13:36:09.702 INFO ht1: -> </body>
2020-08-26 13:36:09.702 INFO ht1: -> </html>
2020-08-26 13:36:09.702 INFO ht1: ->
```

Figure 13 - Message sent

## Step 5 - Message report received

Then the SMS report is sent back to the Apach webserver and it store it in the smsreport.txt (Figure 14).



```
2020-08-26 13:36:09 21a61ead-7e81-4ca3-a1af-ea152b82590e accepted
```

```
[ Read 1 line (Converted from DOS format) ]
```

Figure 14 - Message report in smsreporter.txt

## Example report file

```
2020-08-22 09:06:27 9b457df4-7a38-4fef-b03f-a11d4cf73e70 accepted
2020-08-22 09:07:11 9b457df4-7a38-4fef-b03f-a11d4cf73e7 accepted
2020-08-22 09:07:54 617c6ec4-5844-4895-9a5c-af98d3ecace5 accepted
2020-08-22 09:10:33 617c6ec4-5844-4895-9a5c-af98d3ecace5 delivered
2020-08-22 09:10:40 9b457df4-7a38-4fef-b03f-a11d4cf73e70 delivered
2020-08-22 09:10:54 0f9603b3-ce53-4c28-bbe5-1c69d855e2ee accepted
2020-08-22 09:10:56 0f9603b3-ce53-4c28-bbe5-1c69d855e2ee undelivered
2020-08-22 09:11:27 f531af05-462d-4a74-8eb2-40ad29f06351 accepted
2020-08-22 09:11:32 f531af05-462d-4a74-8eb2-40ad29f06351 delivered
2020-08-22 09:13:34 7ad2c5e2-fa46-472f-adc1-fe2a28bbfa7c accepted
2020-08-22 09:13:45 7ad2c5e2-fa46-472f-adc1-fe2a28bbfa7c delivered
2020-08-22 09:14:35 0e2286bb-5a29-4d59-b636-ff42e2eec375 accepted
2020-08-22 09:17:35 4079feb1-2cf5-4d0f-92e6-4d97e086b918 accepted
2020-08-22 09:17:40 4079feb1-2cf5-4d0f-92e6-4d97e086b918 undelivered
```

# How to send SMS from Perl

***The example code below is about to demonstrate how you can use HTTP requests to send SMS messages from Perl via SMS Gateway. The Perl program first sets up all the important variables that will be used for sending the message. Then it creates the string for sending the message and lastly, it initiates the HTTP request. The response message will be also printed by the program.***

## What is a Perl SMS API?

The Perl SMS API ensures the ability to send SMS messages from a program written in Perl. It uses HTTP requests and SSL encrypted communication to forward the SMS sending request to the SMS Gateway.

## Prerequisites

- Installed Perl
- Installed Ozeki 10 SMS Gateway
- A configured HTTP API User

## Send SMS from Perl

1. Open Notepad
2. Create a new file
3. Copy-Paste the example code from this page
4. Save the file as HttpApiTester.pl
5. Open Command Prompt
6. Navigate to the folder where you saved the Perl file
7. Type 'perl HttpApitester.pl' in the Command Prompt
8. Press Enter to run the Perl file

## Perl SMS source code example

This example Perl source code below is free to use, you can simply implement it into your project or you can modify the source code to use it for other projects or applications. If you want to just run and test this example, all you have to do is to follow the instructions above or watch the tutorial video which demonstrates the way to create a Perl file for the source code and run it from the Command Prompt.

```perl
#!/usr/bin/perl
use strict;
use warnings;
use Encode;
require HTTP::Request;
require LWP::UserAgent;

sub main
{
    my $username = "john";
    my $password = "Xc3ffs";
    my $messagetype = "SMS:TEXT";
    my $httpUrl = "https://127.0.0.1:9508/";
    my $recipient = Encode::encode("utf8", "+36201324567");
    my $messagedata = Encode::encode("utf8", "TestMessage");

    my $sendString = $httpUrl."api?action=sendmessage"."&username="
                    .$username."&password="
                    .$password."&recipient=".$recipient."&messagetype="
                    .$messagetype."&messagedata=".$messagedata;

    print "Sending html request: ".$sendString."\n\n";

    my $request = HTTP::Request->new(GET => $sendString);
    my $ua = LWP::UserAgent->new (
            ssl_opts => { verify_hostname => 0 },
            );
    my $response = $ua->request($request);

    print "Http response received :\n";
    print $response->content;

}

# Call main.
main();
```

## Step 1 - Open a text editor application

To be able to run the example code above, first, you need to create a Perl file for that. To perform this operation you only need to have a text editor. You can download a custom text editor, but the default Notepad application installed on Windows can do the job as well. So, as Figure 1 shows that, just open the Notepad application from your desktop.

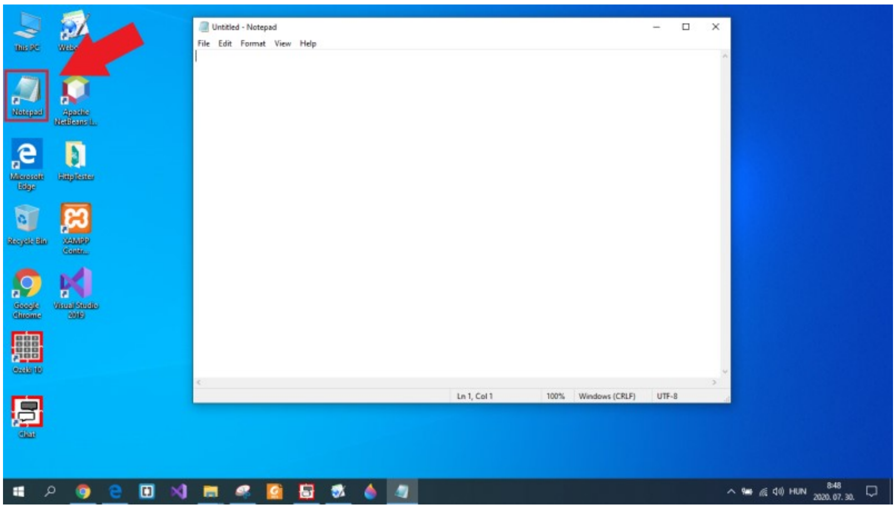
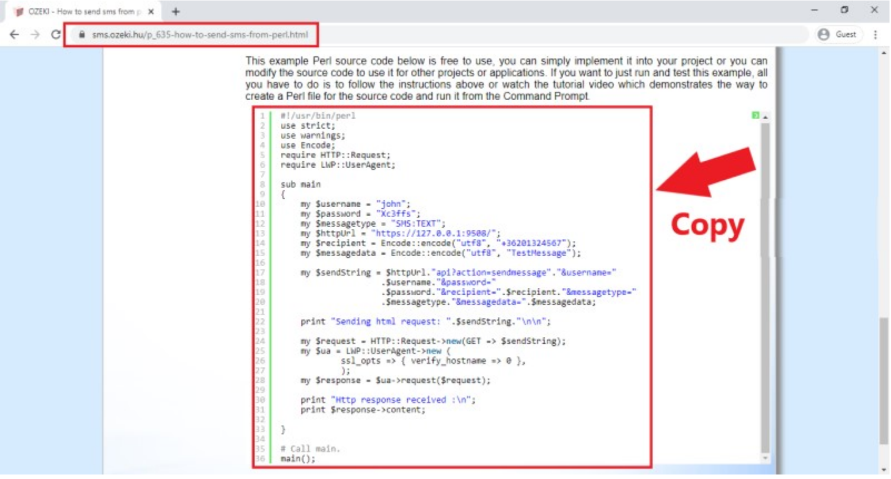


Figure 1 - Open a text editor

## Step 2 - Copy the source code

The next step is to get the source code. This is an easy task, since the code can be found on this page, so you just need to copy that. So, as you can see it in Figure 2, just go to the example code section on this page, mark out the whole source code and press Ctrl+C on your keyboard to copy the source code to the clipboard.


OZEKI - How to send sms from p
sms.ozeki.hu/p_635-how-to-send-sms-from-perl.html
Guest
Copy

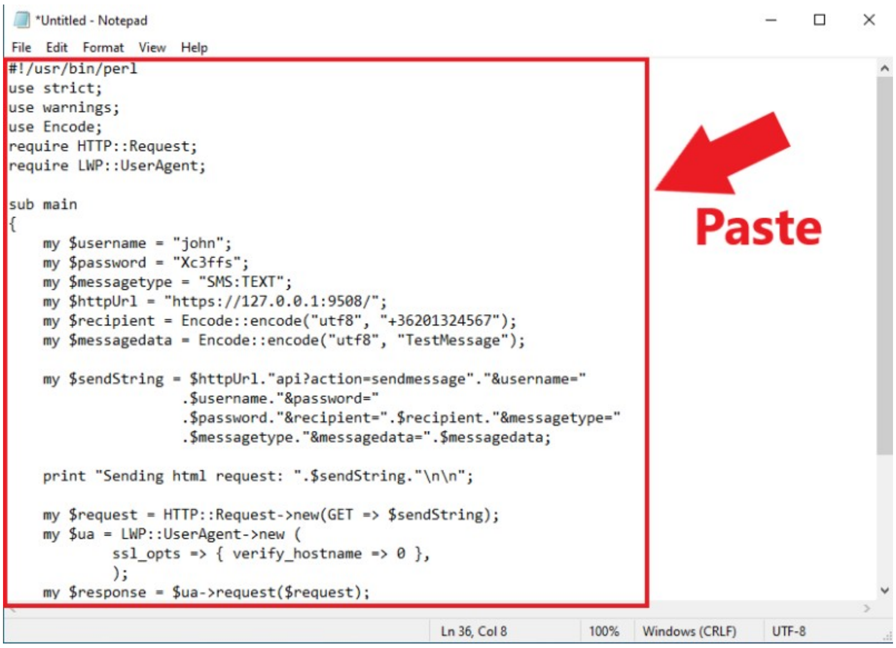
This example Perl source code below is free to use, you can simply implement it into your project or you can modify the source code to use it for other projects or applications. If you want to just run and test this example, all you have to do is to follow the instructions above or watch the tutorial video which demonstrates the way to create a Perl file for the source code and run it from the Command Prompt.

```
#!/usr/bin/perl
use strict;
use warnings;
use Encode;
require HTTP::Request;
require LWP::UserAgent;

sub main
{
    my $username = "john";
    my $password = "Xc3ffs";
    my $messagetype = "SMS:TEXT";
    my $httpUrl = "https://127.0.0.1:9508/";
    my $recipient = Encode::encode("utf8", "+36201324567");
    my $messagedata = Encode::encode("utf8", "TestMessage");

    my $sendString = $httpUrl."api?action=sendmessage"."&username="
                    .$username."&password="
                    .$password."&recipient=".$recipient."&messagetype="
                    .$messagetype."&messagedata=".$messagedata;

    print "Sending html request: ".$sendString."\n\n";

    my $request = HTTP::Request->new(GET => $sendString);
    my $ua = LWP::UserAgent->new (
            ssl_opts => { verify_hostname => 0 },
            );
    my $response = $ua->request($request);

    print "Http response received :\n";
    print $response->content;

}

# Call main.
main();
```

Figure 2 - Copy Code from the website

## Step 3 - Paste the code into your text file

Next, you need to place the copied source code into your empty text file. To do that, you just need to navigate to the text file and press Ctrl+V on your keyboard. By performing this action, the source code will be placed into your text file as Figure 3 demonstrates it.

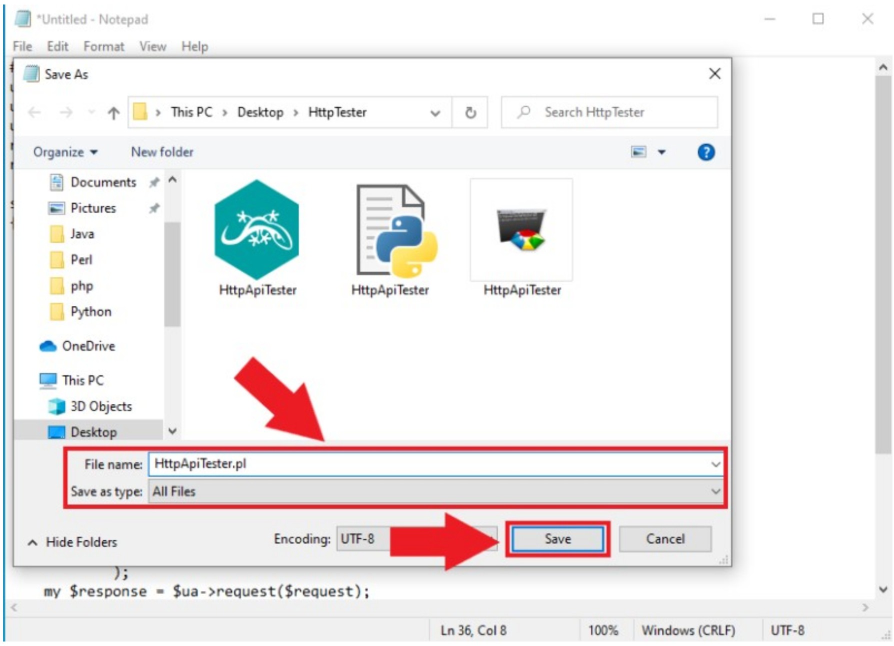


```
#!/usr/bin/perl
use strict;
use warnings;
use Encode;
require HTTP::Request;
require LWP::UserAgent;

sub main
{
    my $username = "john";
    my $password = "Xc3ffs";
    my $messagetype = "SMS:TEXT";
    my $httpUrl = "https://127.0.0.1:9508/";
    my $recipient = Encode::encode("utf8", "+36201324567");
    my $messagedata = Encode::encode("utf8", "TestMessage");

    my $sendString = $httpUrl."api?action=sendmessage"."&username="
                    .$username."&password="
                    .$password."&recipient=".$recipient."&messagetype="
                    .$messagetype."&messagedata=".$messagedata;

    print "Sending html request: ".$sendString."\n\n";

    my $request = HTTP::Request->new(GET => $sendString);
    my $ua = LWP::UserAgent->new (
            ssl_opts => { verify_hostname => 0 },
            );
    my $response = $ua->request($request);
```

Figure 3 - Paste the program code into the text editor

## Step 4 - Save the text as a Perl file

To create the Perl file, you just need to save the text file with the file extension of the Perl files. This extension is .pl. So, select the 'Save as' option in Notepad, and as you can see it in Figure 4, and after you gave a name for the file, append it with the '.pl' file extension and click on 'Save'. The operation just created the Perl file that you need to run the Perl SMS example.



File name: HttpApiTester.pl

Save as type: All Files

Encoding: UTF-8

```
            );
    my $response = $ua->request($request);
```

Figure 4 - Save the HttpApiTester Perl file

## Step 5 - Open Command Prompt

The last main step the guide is to run the Perl file you created before. In this case, you are going to run the example using the Command Prompt. You can easily open the Command Prompt by opening the File Explorer, and like in Figure 5, just type 'cmd' in the address bar and lastly, just hit Enter. If you navigate into the folder where you saved the Perl file in the File Explorer, the Command Prompt will use this location as default, so you can run the Perl file straightaway.

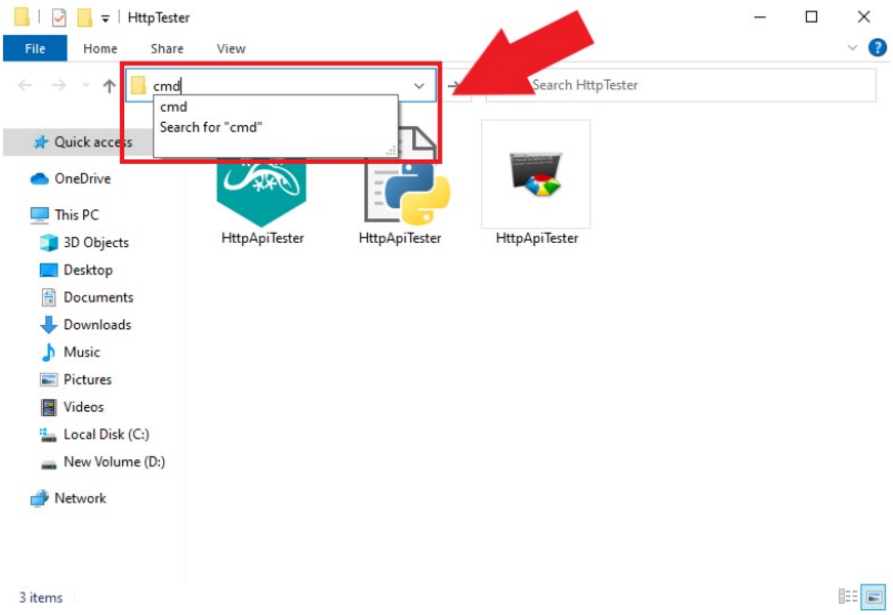
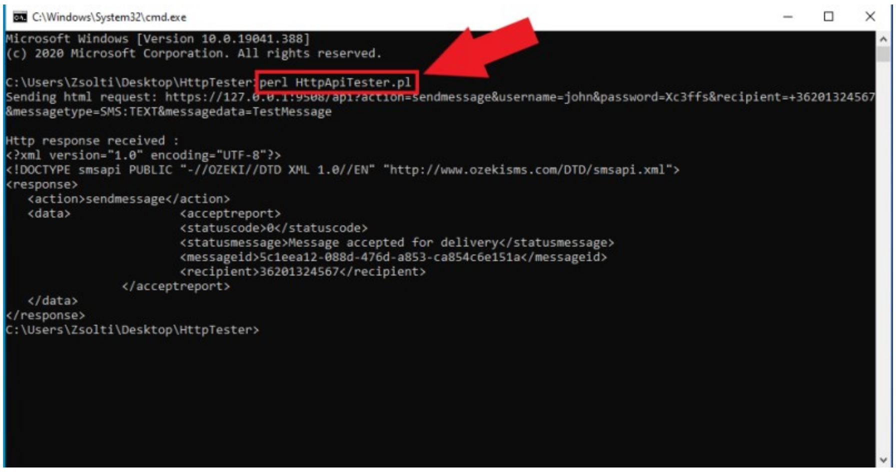


Figure 5 - Open Command Prompt

# Step 6 - Run the Perl file

The last thing that you need to perform, is to initiate the command, that runs the Perl file. This is a simple command, you just need to write 'perl *file name*.pl' in the Command Prompt as Figure 6 shows that. After you pressed Enter, the Perl file executes the commands within and sends the test message. The HTTP request and response from the SMS Gateway will be printed in the Command Prompt.

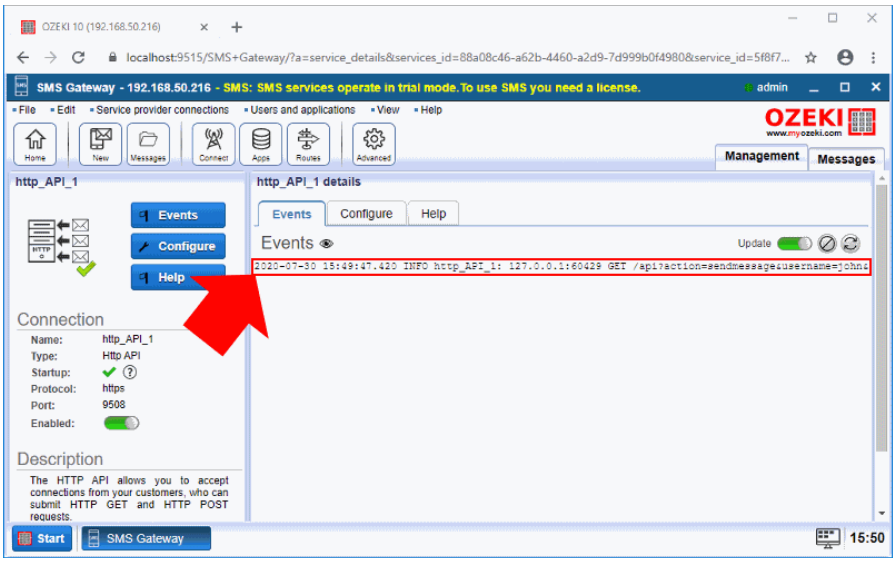


```
Microsoft Windows [Version 10.0.19041.388]
(c) 2020 Microsoft Corporation. All rights reserved.

C:\Users\Zsolti\Desktop\HttpTester>perl HttpApiTester.pl
Sending html request: https://127.0.0.1:9508/api?action=sendmessage&username=john&password=Xc3ffs&recipient=+36201324567
&messagetype=SMS:TEXT&messagedata=TestMessage

Http response received :
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE smsapi PUBLIC "-//OZEKI//DTD XML 1.0//EN" "http://www.ozekisms.com/DTD/smsapi.xml">
<response>
   <action>sendmessage</action>
   <data>                    <acceptreport>
                    <statuscode>0</statuscode>
                    <statusmessage>Message accepted for delivery</statusmessage>
                    <messageid>5c1eea12-088d-476d-a853-ca854c6e151a</messageid>
                    <recipient>36201324567</recipient>
            </acceptreport>
   </data>
</response>
C:\Users\Zsolti\Desktop\HttpTester>
```

Figure 6 - Run the HttpApiTester.pl script

# Step 7 - Check the send result in the Ozeki log

Ozeki 10 SMS Gateway ensures that to check what messages sent by your application, since the HTTP API service logs every event that occured during the time it is enabled. So, if you open the SMS Gateway, and select details of the HTTP API service, you will be able to see the events. As you can see it in Figure 7, the service logged an event, when the Perl program sent the HTTP request to the service.


SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode. To use SMS you need a license.

http_API_1

Connection

Name: http_API_1
Type: Http API
Startup: ✓
Protocol: https
Port: 9508
Enabled:

Description

The HTTP API allows you to accept connections from your customers, who can submit HTTP GET and HTTP POST requests.

http_API_1 details

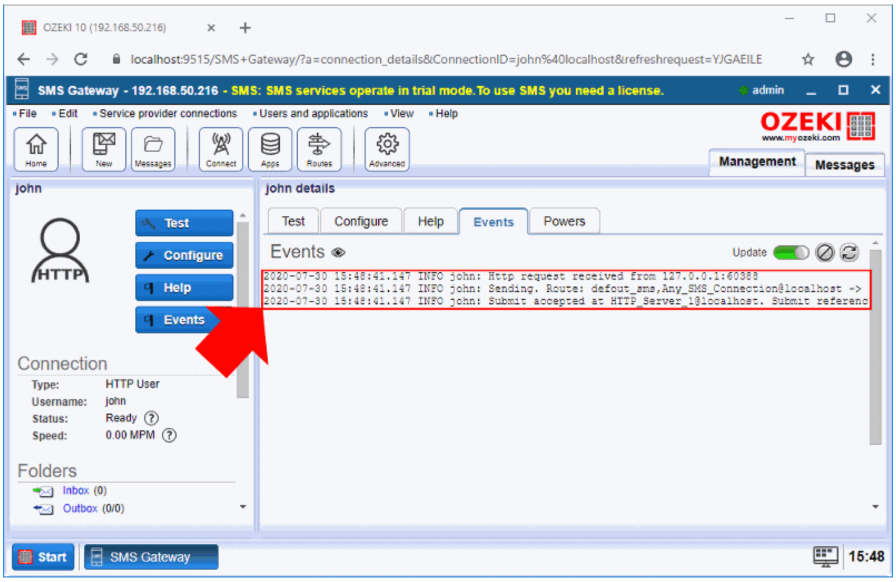
Events

```
2020-07-30 15:49:47.420 INFO http_API_1: 127.0.0.1:60429 GET /api?action=sendmessage&username=john&
```

OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=service_details&services_id=88a08c46-a62b-4460-a2d9-7d999b0f4980&service_id=5f8f7...
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
OZEKI
www.myozeki.com
Management
Messages
Events
Configure
Help
Update
Start
SMS Gateway
15:50

Figure 7 - Check the logs of the HTTP API Service

The events that occures during the sending of the message back be checked by the events as well. For that, open the HTTP API User connection, that you had to configure before. Figure 8 demonstrates that how the connection handles the HTTP request and send the message to the recipient that you specified in your Perl SMS exaple program.


SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode. To use SMS you need a license.

john

Connection

Type: HTTP User
Username: john
Status: Ready
Speed: 0.00 MPM

Folders

Inbox (0)
Outbox (0/0)

john details

Events

```
2020-07-30 15:48:41.147 INFO john: Http request received from 127.0.0.1:60388
2020-07-30 15:48:41.147 INFO john: Sending. Route: defout_sms,Any_SMS_Connection@localhost ->
2020-07-30 15:48:41.147 INFO john: Submit accepted at HTTP_Server_1@localhost. Submit referenc
```

Figure 8 - Check the logs of the HTTP API User

# Receive SMS in Perl

***On this page, you will be able to learn about how you can use your Perl application to receive SMS messages. This example Perl source is capable of collecting the incoming messages from the SMS Gateway and use them in the application. It performs this operation by using HTTP request. This guide shows how easy you can create a Perl application and run it on your computer. So, let's start right now.***

## What is a Perl SMS API?

The Perl SMS API ensures the ability to send SMS messages from a program written in Perl. It uses HTTP requests and SSL encrypted communication to forward the SMS sending request to the SMS Gateway.

## Prerequisites

- Installed Perl
- Installed Ozeki 10 SMS Gateway
- A configured HTTP Server connection

## Receive SMS in Perl

1. Open Notepad
2. Copy-Paste the example code from this page
3. Save the file as ReceiveSMS.pl
4. Open Ozeki SMS Gateway and select HTTP Server connection
5. Open the HTML form of the HTTP Server connection
6. Send some test messages
7. Open Command Prompt
8. Type 'perl ReceiveSMS.pl' in the Command Prompt and press Enter

This example code below is free to use in your own solution or application and you can modify it if you want to. The example code can collect the messages from the SMS Gateway and use it in the Perl application. If you would like to test the example just follow the step by step instructions below or check the video which clearly shows all steps that you need to perform to execute the Perl SMS example.

```perl
#!/usr/bin/perl
use strict;
use warnings;
use XML::Simple;
use Data::Dumper;
require HTTP::Request;
require LWP::UserAgent;

sub main
{
    my $username = "john";
    my $password = "Xc3ffs";
    my $httpUrl = "https://127.0.0.1:9508/";
    my $folder = "inbox";
    my $limit = "3";

    my $sendString = $httpUrl."api?action=receivemessage&username="
                    .$username."&password=".$password."&folder="
                    .$folder."&limit=".$limit
                    ."&messagedata=&afterdownload=delete";

    print "Sending html request: ".$sendString."\n\n";

    my $request = HTTP::Request->new(GET => $sendString);
    my $ua = LWP::UserAgent->new (
            ssl_opts => { verify_hostname => 0 },
            );
    my $response = $ua->request($request);
    print "Http response received :\n";

    DisplayMessages($response->content)
}
```

```

main();

sub DisplayMessages {
    my $data = XMLin(@_);
    my $sender = "";
    my $text = "";
    if($data->{data}->{message} eq 'No more messages.')   {
        print "The inbox is empty!";
        return;
    }

    if (ref($data->{data}->{message}) eq 'ARRAY') {
        foreach my $value (@{ $data->{data}->{message} }) {
            $sender = $value->{originator};
            $text =  $value->{messagedata};
            DisplayMessage($sender,$text);
        }
    }
    else {
        $sender = $data->{data}->{message}->{originator};
        $text = $data->{data}->{message}->{messagedata};
        DisplayMessage($sender,$text);
    }
}

sub DisplayMessage {
    print $_[0].": ".$_[1]."\n";
}
```

## Step 1 - Open Notepad

The first thing that you have to perform to complete this guide is to create the Perl file that you have to execute to see how it gets the SMS messages from the SMS Gateway. So, at this point, you need to open the Notepad application on your computer to be able to create the Perl file. For that, just like in Figure 1, click on its icon on the desktop.

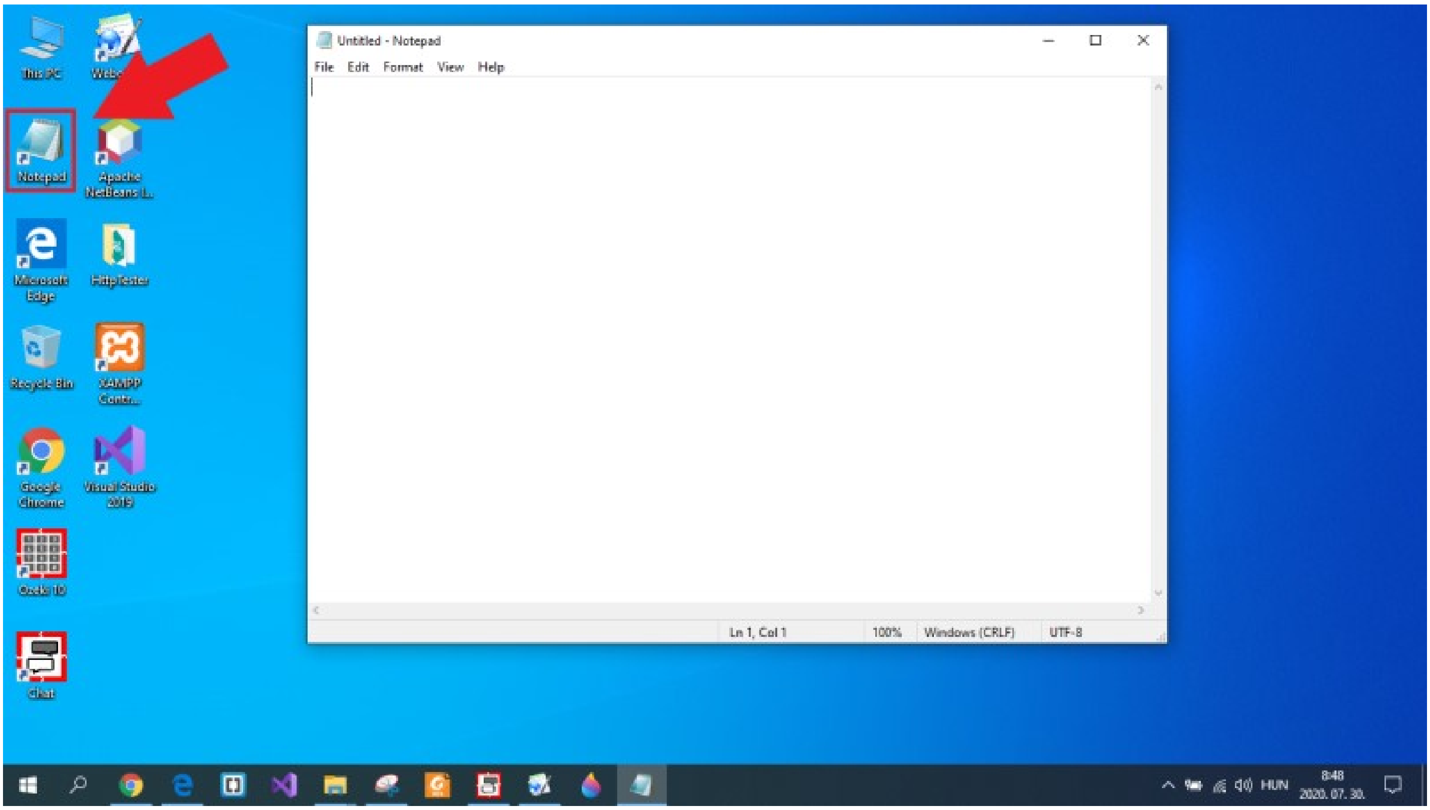
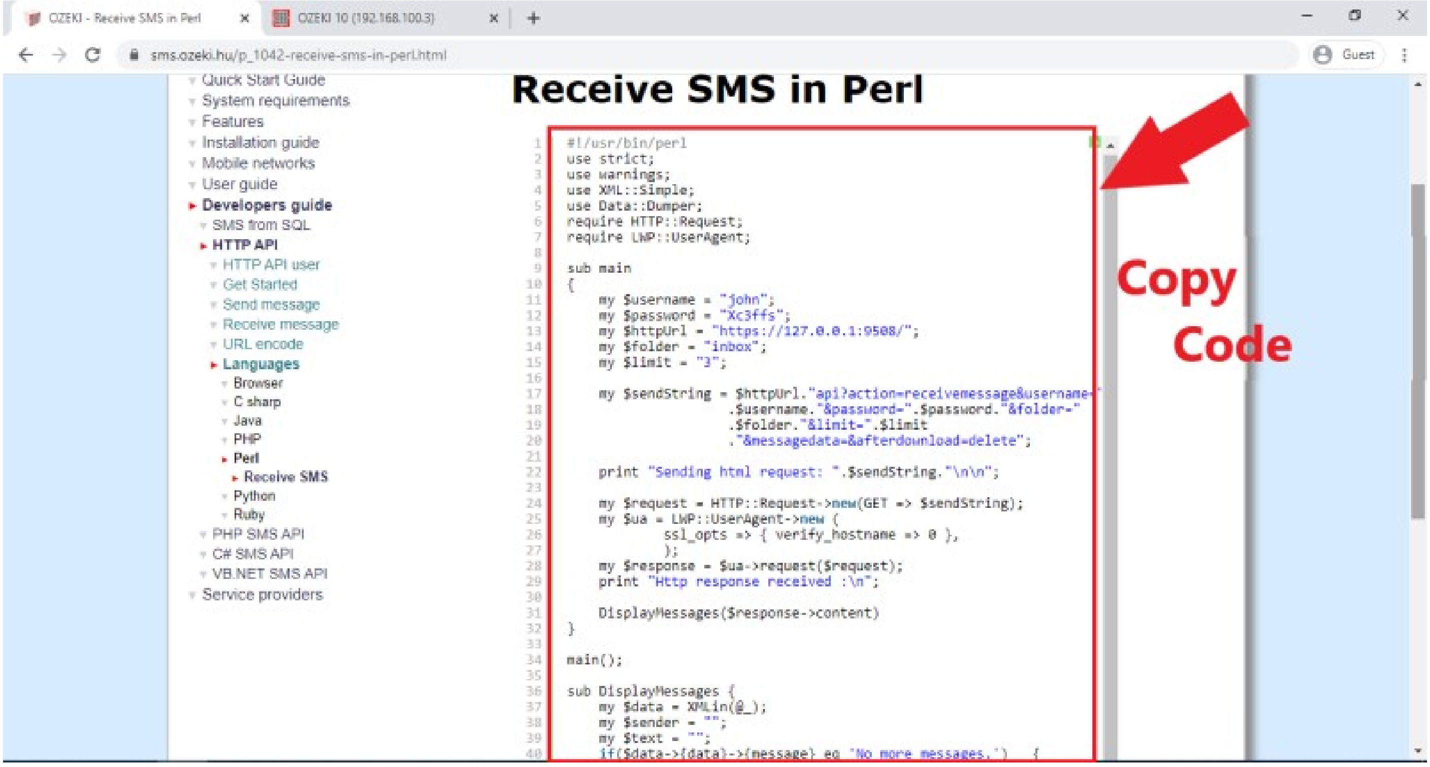


Figure 1 - Open notepad text editor

## Step 2 - Copy the Perl source code from this page

Next, you need to get the Perl source code from this page to use it on your computer. So, just scroll up to the example code section and mark out the whole source code. Then, as you can see it in Figure 2, you need to copy it to your clipboard. This can be done by pressing Ctrl+C on your keyboard.

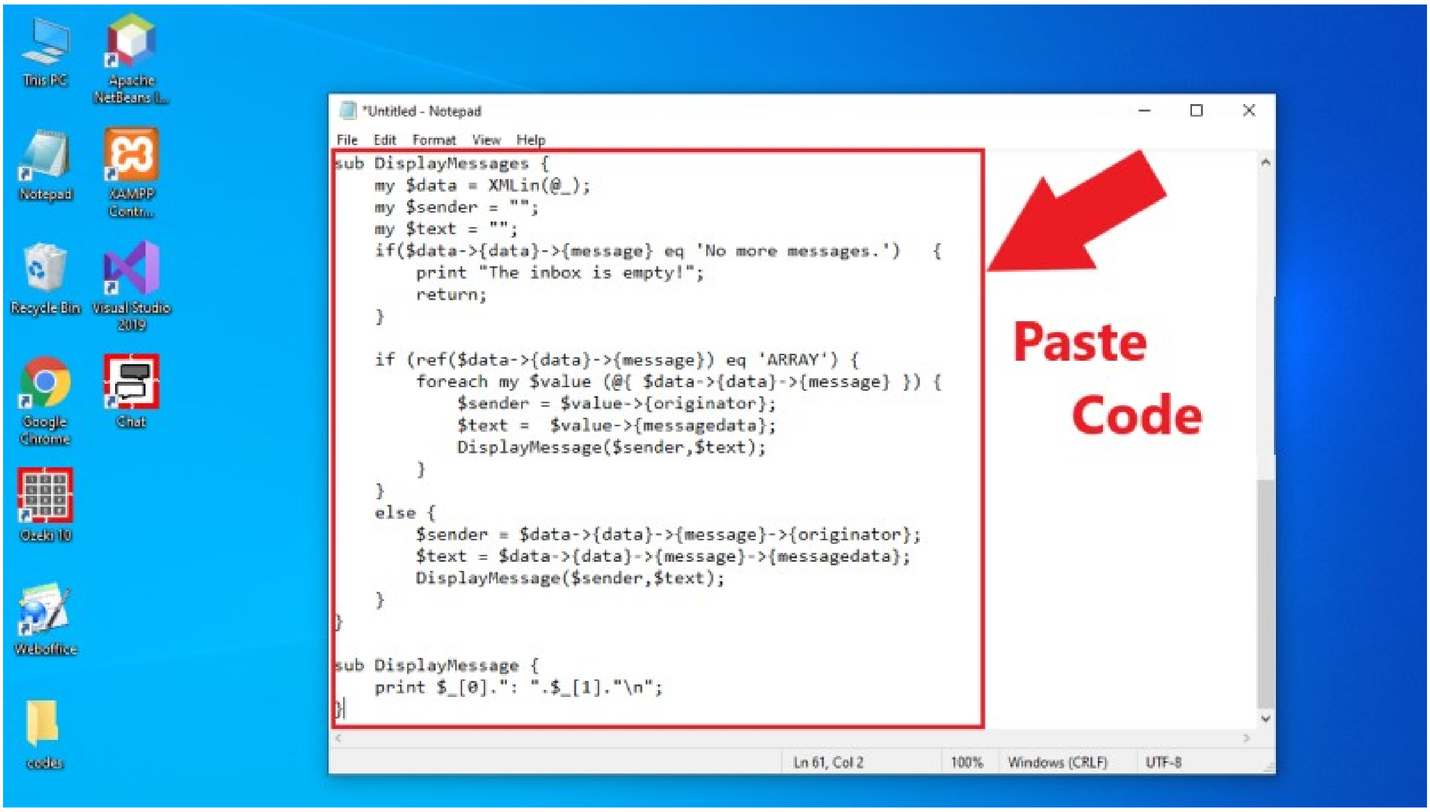


```perl
#!/usr/bin/perl
use strict;
use warnings;
use XML::Simple;
use Data::Dumper;
require HTTP::Request;
require LWP::UserAgent;

sub main
{
    my $username = "john";
    my $password = "Xc3ffs";
    my $httpUrl = "https://127.0.0.1:9508/";
    my $folder = "inbox";
    my $limit = "3";

    my $sendString = $httpUrl."api?action=receivemessage&username="
                    .$username."&password=".$password."&folder="
                    .$folder."&limit=".$limit
                    ."&messagedata=&afterdownload=delete";

    print "Sending html request: ".$sendString."\n\n";

    my $request = HTTP::Request->new(GET => $sendString);
    my $ua = LWP::UserAgent->new (
            ssl_opts => { verify_hostname => 0 },
            );
    my $response = $ua->request($request);
    print "Http response received :\n";

    DisplayMessages($response->content)
}

main();

sub DisplayMessages {
    my $data = XMLin(@_);
    my $sender = "";
    my $text = "";
    if($data->{data}->{message} eq 'No more messages.')   {
```

Figure 2 - Copy perl source code from the website

## Step 3 - Paste the code into the text file

After you copied the Perl source code from this page, you need to place it into a text document. To do this, just go to the Notepad application that you opened before, and as you can see it in Figure 3, just press Ctrl+V on your keyboard to paste the whole source code into the text document.

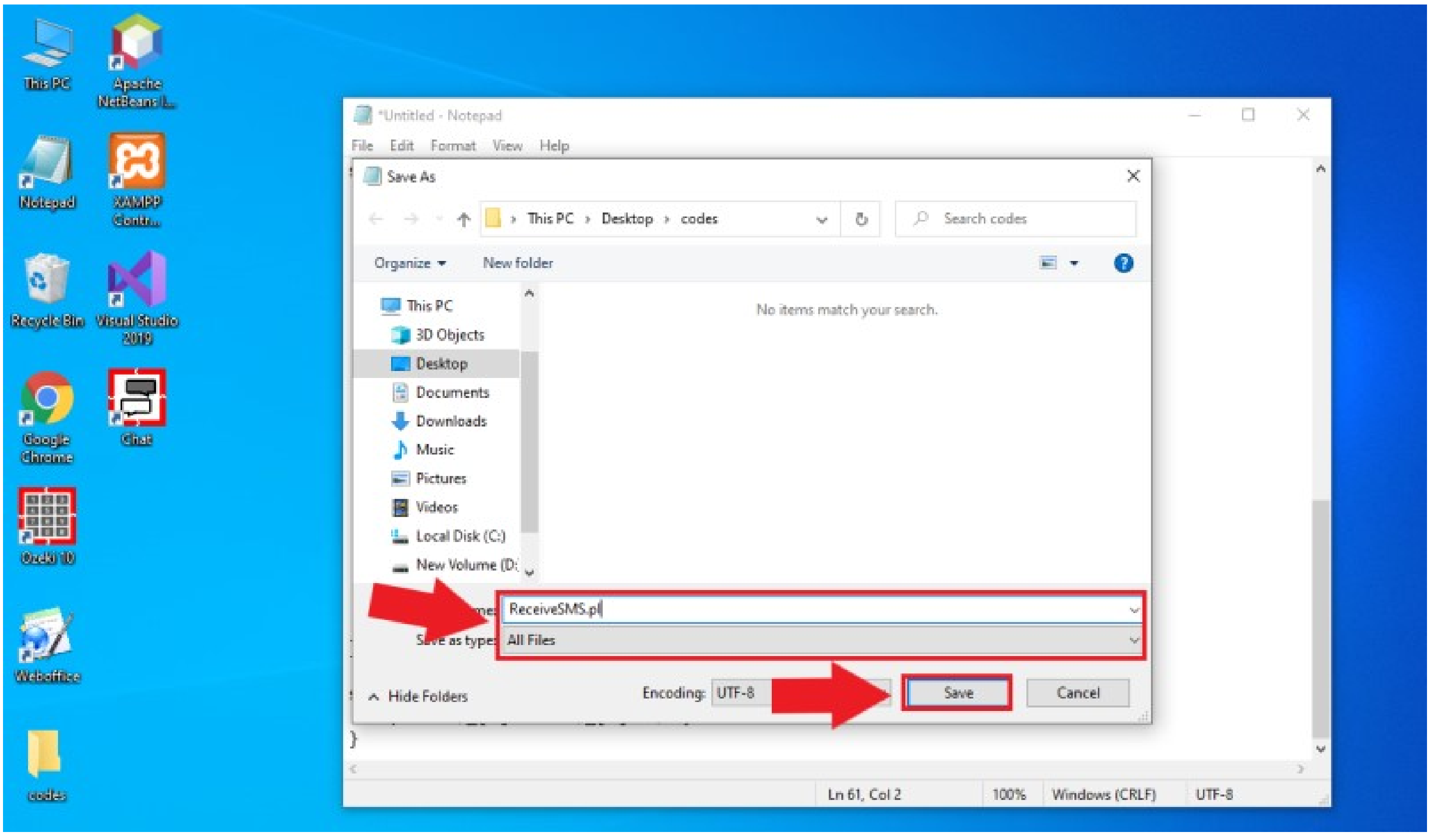


```perl
sub DisplayMessages {
    my $data = XMLin(@_);
    my $sender = "";
    my $text = "";
    if($data->{data}->{message} eq 'No more messages.')   {
        print "The inbox is empty!";
        return;
    }

    if (ref($data->{data}->{message}) eq 'ARRAY') {
        foreach my $value (@{ $data->{data}->{message} }) {
            $sender = $value->{originator};
            $text =  $value->{messagedata};
            DisplayMessage($sender,$text);
        }
    }
    else {
        $sender = $data->{data}->{message}->{originator};
        $text = $data->{data}->{message}->{messagedata};
        DisplayMessage($sender,$text);
    }
}

sub DisplayMessage {
    print $_[0].": ".$_[1]."\n";
}
```

Figure 3 - Paste perl source code to notepad

# Step 4 - Save the text as a Perl file

The source code is now placed into the text document, but it still hasn't got the right file format. But this is now a huge problem, you can solve it easily with Notepad. Just select the 'Save as' option, and here, after you gave a name to the file, use the '.pl' file extension as Figure 4 demonstrates it. If you click 'Save', the file will be saved in the right Perl file format that you need to execute later.

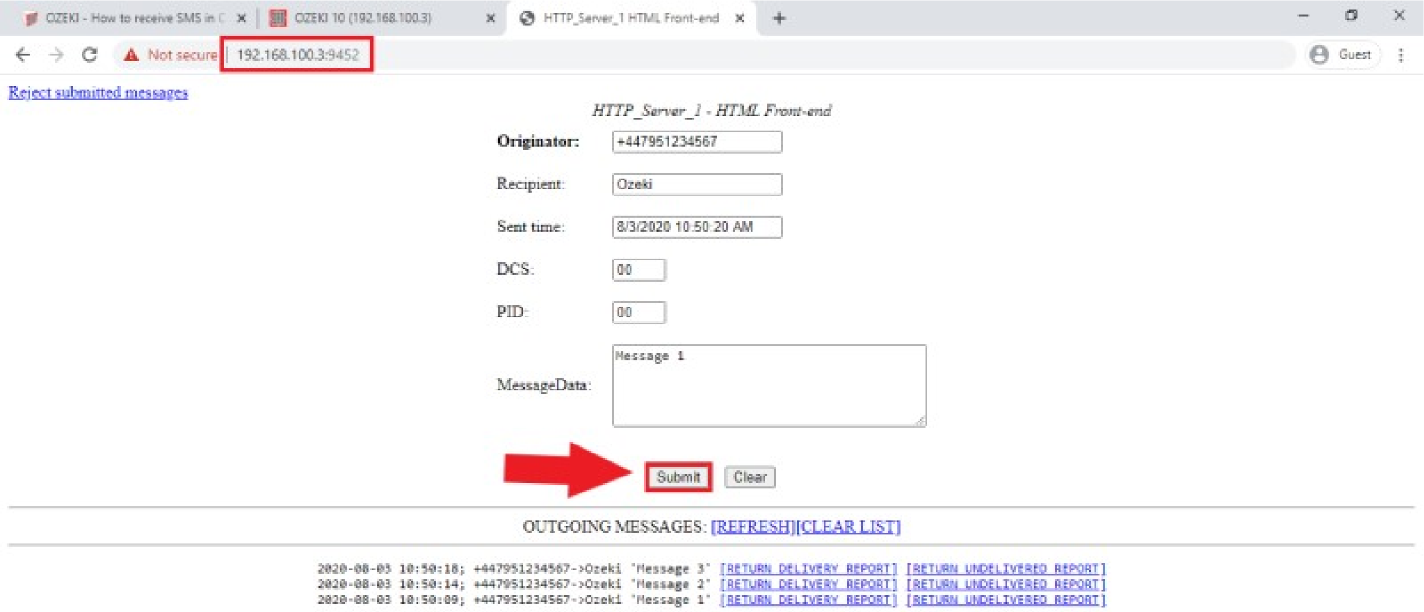


File name: ReceiveSMS.pl
Save as type: All Files
Encoding: UTF-8

Figure 4 - Save file as ReceiveSMS.pl

# Step 5 - Send some test messages

After you set up the Perl file, you need some message to your inbox folder. To finish the test quickly, you can simulate incoming messages with the HTTP Server connection in SMS Gateway. If you haven't configured this connection yet, here you can see how to configure a HTTP Server connection. So, open SMS Gateway, and select the HTTP Server connection. In the menu of the connection, open the HTML form (Figure 5). Here, you can write some messages to the SMS Gateway by setting the recipient to 'Ozeki'.


OZEKI - How to receive SMS in C
OZEKI 10 (192.168.100.3)
HTTP_Server_1 HTML Front-end
Not secure
192.168.100.3:9452
Guest
Reject submitted messages

*HTTP_Server_1 - HTML Front-end*

**Originator:** +447951234567

Recipient: Ozeki

Sent time: 8/3/2020 10:50:20 AM

DCS: 00

PID: 00

MessageData: Message 1

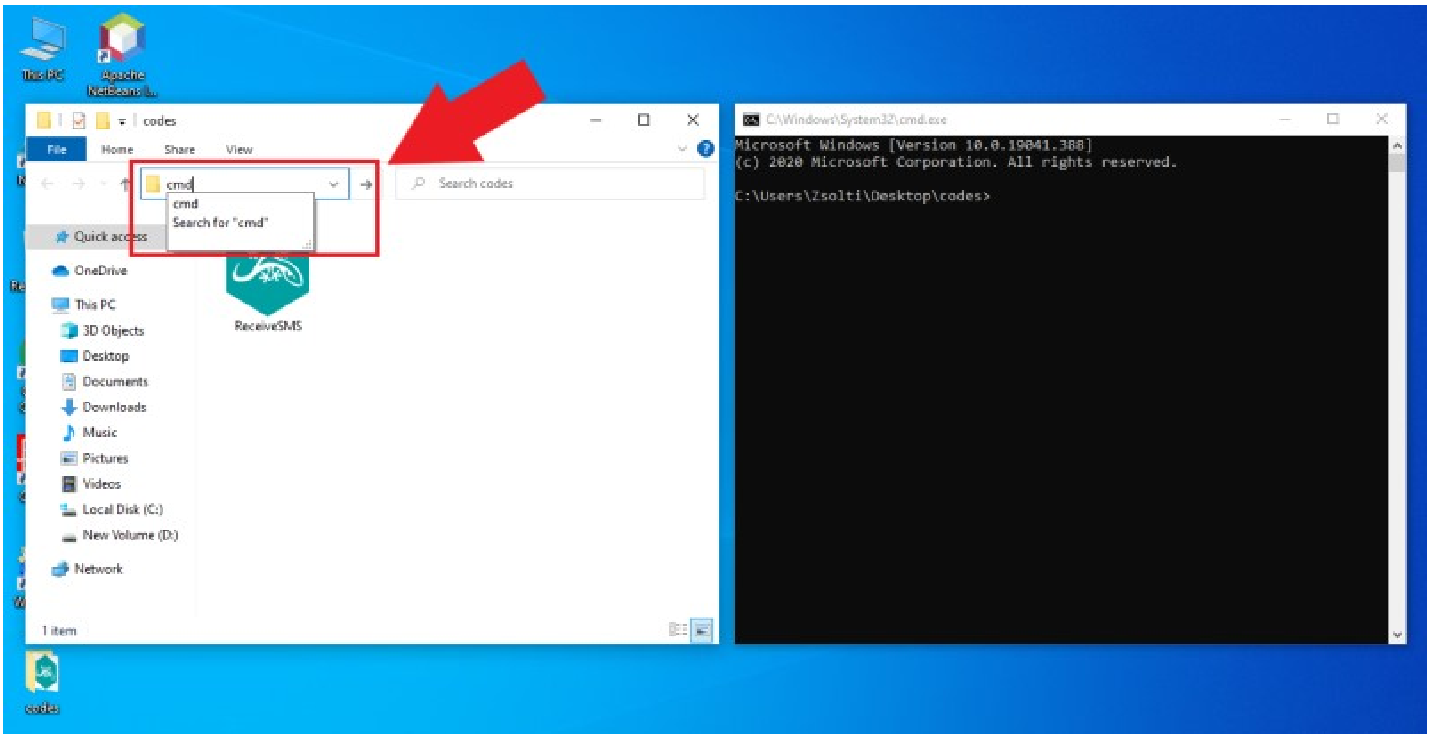
OUTGOING MESSAGES: [REFRESH][CLEAR LIST]

```
2020-08-03 10:50:18; +447951234567->Ozeki 'Message 3' [RETURN DELIVERY REPORT] [RETURN UNDELIVERED REPORT]
2020-08-03 10:50:14; +447951234567->Ozeki 'Message 2' [RETURN DELIVERY REPORT] [RETURN UNDELIVERED REPORT]
2020-08-03 10:50:09; +447951234567->Ozeki 'Message 1' [RETURN DELIVERY REPORT] [RETURN UNDELIVERED REPORT]
```

Figure 5 - Simulate some incoming SMS

## Step 6 - Open Command Prompt

To run the Perl code, you need to Command Prompt since it provides a simple way to execute the code. So, first open the File Explorer and navigate to the folder, where you saved your Perl file. Here, as you can see it in Figure 6, type 'cmd' into the File Explorer. If you hit Enter, and Command Prompt opens in a new window.



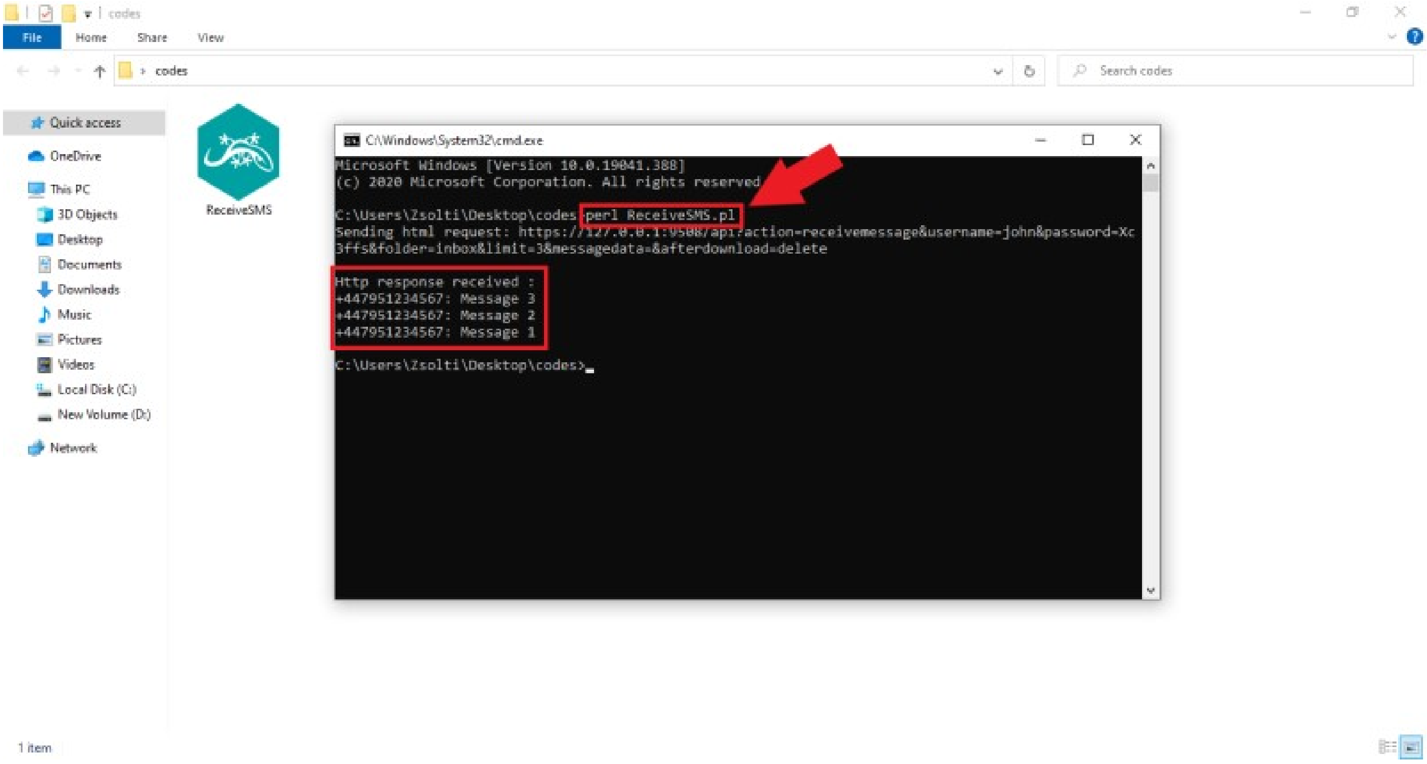
```
Microsoft Windows [Version 10.0.19041.388]
(c) 2020 Microsoft Corporation. All rights reserved.

C:\Users\Zsolti\Desktop\codes>
```

Figure 6 - Open Command Prompt

## Step 7 - Run the Perl SMS example

The lest thing that you have to do is to execute the Perl program. For that, you just need to execute a simple command in Command Prompt. The command is 'perl *filename*.pl' as Figure 7 shows that. Then, just press Enter to execute the example program. The program prints the HTTP request that was sent to the SMS

Gateway, and also prints the response message from the SMS Gateway. This contains all messages that was received by the SMS Gateway.



```
Microsoft Windows [Version 10.0.19041.388]
(c) 2020 Microsoft Corporation. All rights reserved.

C:\Users\Zsolti\Desktop\codes>perl ReceiveSMS.pl
Sending html request: https://127.0.0.1:9508/api?action=receivemessage&username=john&password=Xc
3ffs&folder=inbox&limit=3&messagedata=&afterdownload=delete

Http response received :
+447951234567: Message 3
+447951234567: Message 2
+447951234567: Message 1

C:\Users\Zsolti\Desktop\codes>
```

Figure 7 - Run perl code in command prompt

# How to send SMS from Python

***This page contains a simple Python script, which is going to demonstrate how easy you can send SMS messages from that script by initiating an HTTP request and forward it to the SMS Gateway which will further processes the request, sends the message and returns a response message to the Python script. As the example script shows below, all these operations only take about twenty lines of code which keeps the script simple.***

## What is a Python SMS API?

The Python SMS API creates the opportunity to send SMS messages from a Python script. By using this Python SMS API, your Python script can send HTTP requests to the SMS Gateway which sends the SMS message.

## Prerequisites

- Installed Python
- Installed Ozeki 10 SMS Gateway
- A configured HTTP API User

## Send SMS from Python

1. Open Notepad
2. Create a new empty file
3. Copy-Paste the example Python source code below
4. Save the file as HttpApiTester.py
5. Open Command Prompt
6. Navigate to the folder where you saved the Python file
7. Type 'python HttpApiTester.py' in the Command Prompt
8. Hit Enter to run the Python script

## Python SMS source code example

The following example source code written in Python programming language is free to use, you can simply implement it into your project or you can modify the source code to use it for other projects or applications. If you wish to run this example code unchanged, you just need to create a new Python file, copy-paste the example code below and save the Python file. Then, all you have to do is to run the Python file as the steps guide and the video demonstrates it above.

```python
import requests
import urllib.parse
import ssl

username = "john"
password = "Xc3ffs"
messagetype = "SMS:TEXT"
httpUrl = "https://127.0.0.1:9508/"
recipient = urllib.parse.quote("+36201324567")
messagedata = urllib.parse.quote("TestMessage")

sendString = (httpUrl + "api?action=sendmessage" + "&username="
    + username + "&password="
    + password + "&recipient=" + recipient + "&messagetype=" +
     messagetype + "&messagedata=" + messagedata)

print("Sending html request: " + sendString)
requests.packages.urllib3.disable_warnings()

response = requests.get(sendString, verify=False)
print("Http response received: ")
print(response.text)
```

# Step 1 - Open a text editor

To be able to execute the Python SMS example code above, first, you need to create a Python file which contains the source code. To perform this action, you only need a text editor. You can download a custom text editor if you wish, but the default Notepad application installed on Windows can do the job as well. So, as Figure 1 demonstrates that, just click on the icon of Notepad to open it.

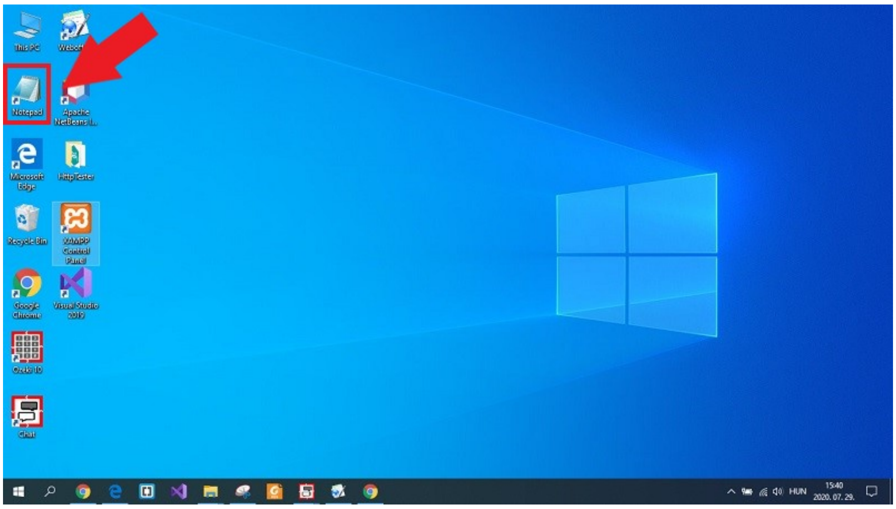
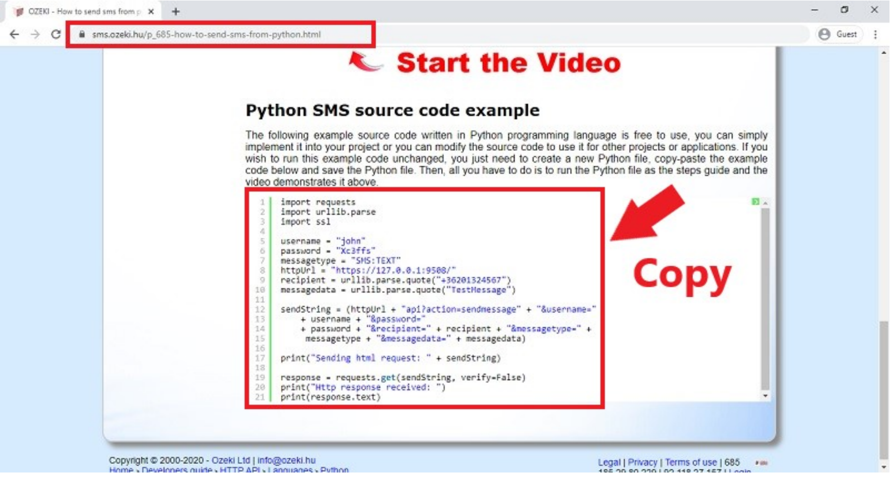


Figure 1 - Open a text editor

# Step 2 - Copy the source code from this page

The next step of the guide is to get the Python source code. This can be easily done, since the code can be found on this page, so you just need to copy that. So, as Figure 2 shows that, just go to the example code section on this page, mark out the whole source code and press Ctrl+C on your keyboard to copy the source code to the clipboard.


OZEKI - How to send sms from p
sms.ozeki.hu/p_685-how-to-send-sms-from-python.html
Guest
Start the Video
Copy

## Python SMS source code example

The following example source code written in Python programming language is free to use, you can simply implement it into your project or you can modify the source code to use it for other projects or applications. If you wish to run this example code unchanged, you just need to create a new Python file, copy-paste the example code below and save the Python file. Then, all you have to do is to run the Python file as the steps guide and the video demonstrates it above.

```
import requests
import urllib.parse
import ssl

username = "john"
password = "Xc3ffs"
messagetype = "SMS:TEXT"
httpUrl = "https://127.0.0.1:9508/"
recipient = urllib.parse.quote("+36201324567")
messagedata = urllib.parse.quote("TestMessage")

sendString = (httpUrl + "api?action=sendmessage" + "&username="
     + username + "&password="
     + password + "&recipient=" + recipient + "&messagetype=" +
      messagetype + "&messagedata=" + messagedata)

print("Sending html request: " + sendString)

response = requests.get(sendString, verify=False)
print("Http response received: ")
print(response.text)
```

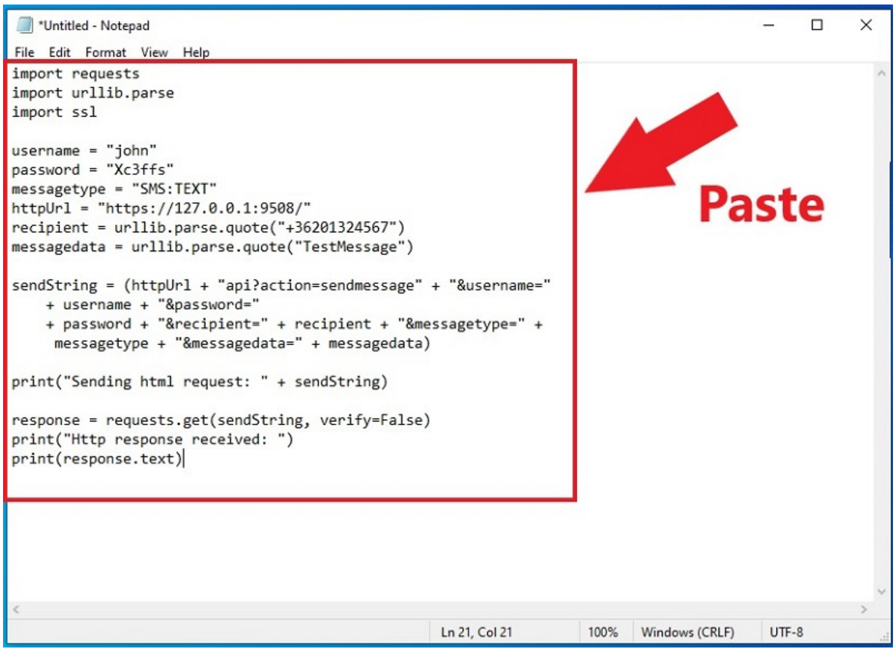
Copyright © 2000-2020 - Ozeki Ltd | info@ozeki.hu

Legal | Privacy | Terms of use | 685

Figure 2 - Copy program code from website

# Step 3 - Paste the code into the text file

After you copied the code from this page, you need to paste it into your empty text file. To do that, you just need to navigate to the text file and just press Ctrl+V on your keyboard. By performing this action, the Python code will be pasted into your text file as Figure 3 shows that.

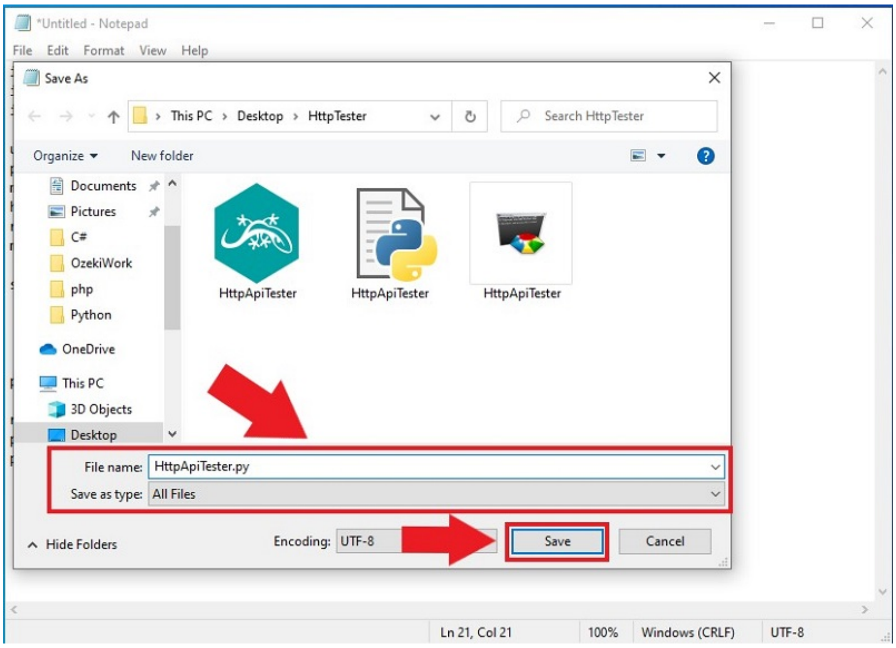


```
import requests
import urllib.parse
import ssl

username = "john"
password = "Xc3ffs"
messagetype = "SMS:TEXT"
httpUrl = "https://127.0.0.1:9508/"
recipient = urllib.parse.quote("+36201324567")
messagedata = urllib.parse.quote("TestMessage")

sendString = (httpUrl + "api?action=sendmessage" + "&username="
    + username + "&password="
    + password + "&recipient=" + recipient + "&messagetype=" +
     messagetype + "&messagedata=" + messagedata)

print("Sending html request: " + sendString)

response = requests.get(sendString, verify=False)
print("Http response received: ")
print(response.text)
```

Figure 3 - Paste the program code into the text editor

## Step 4 - Save the text file as Python file

To create the Python file, you just need to save the text file with the file extension of the Python files. This extension is .py. So, select the 'Save as' option in Notepad, and as you can see it in Figure 4, and after you gave a name for the file, append it with the '.py' file extension and click on 'Save'. The operation just created the Python file that you need to run the Python SMS example.



File name: HttpApiTester.py

Save as type: All Files

Encoding: UTF-8

Figure 4 - Save the HttpApiTester python file

## Step 5 - Open Command Prompt

The next step of this guide is to run the Python file you created before. In this case, you are going to run the example using the Command Prompt. You can easily open the Command Prompt by opening the File Explorer, and as you can see it in Figure 5, just type 'cmd' in the address bar and lastly, just press Enter. If you navigate into the folder where you saved the Python code in the File Explorer, the Command Prompt will use this location as default, so you just need to run the Python file.

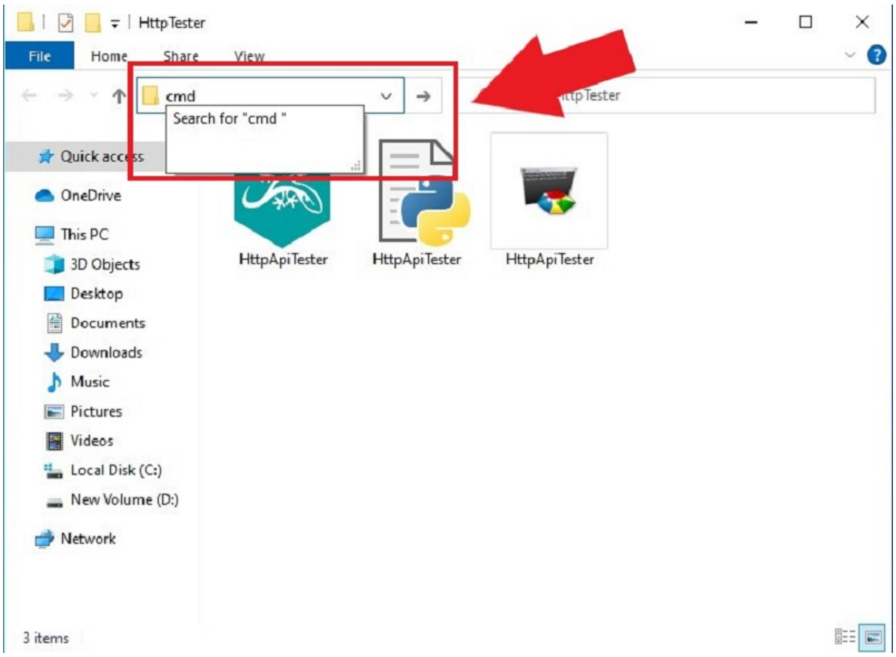
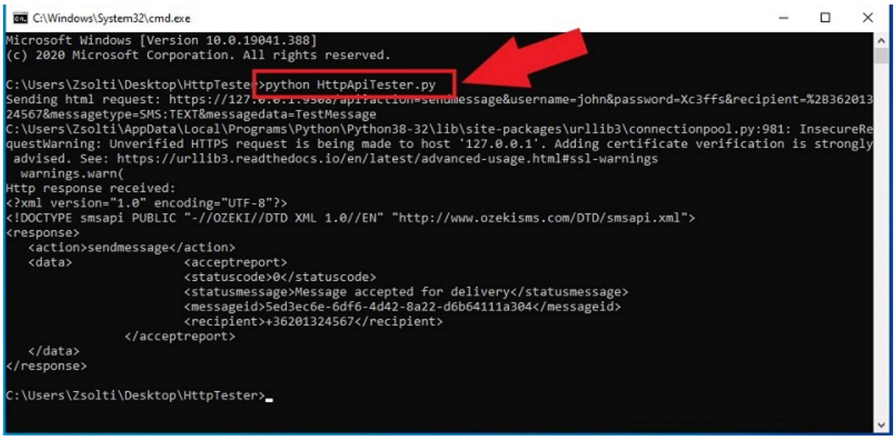


Figure 5 - Open Command Prompt

# Step 6 - Run the Python file

The last step that you need to do, is to initiate the command, that runs the Python script. This is a simple command, you just need to write 'python *file name*.py' in the Command Prompt as Figure 6 shows that. After you pressed Enter, the Python file executes the commands within and sends the test message. The HTTP request and response from the SMS Gateway will be printed in the Command Prompt.

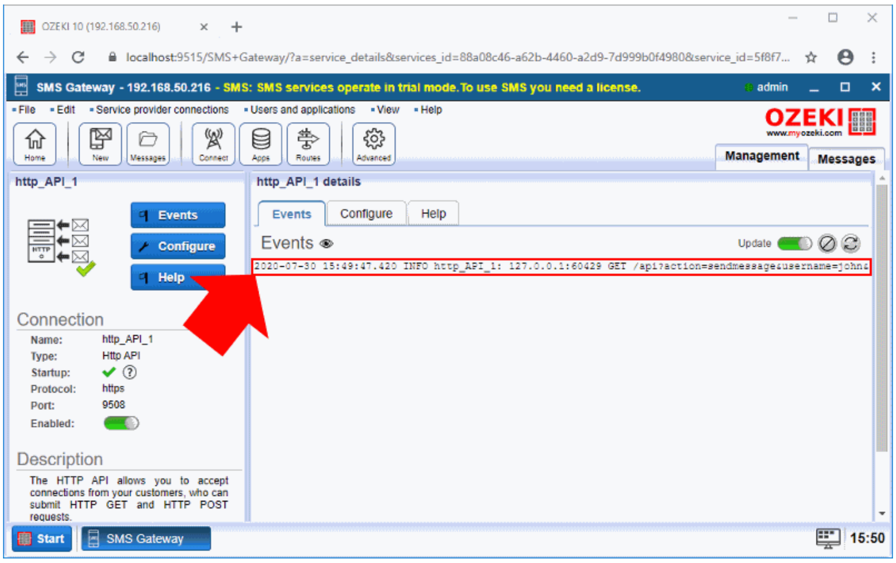


```
Microsoft Windows [Version 10.0.19041.388]
(c) 2020 Microsoft Corporation. All rights reserved.

C:\Users\Zsolti\Desktop\HttpTester>python HttpApiTester.py
Sending html request: https://127.0.0.1:9508/api?action=sendmessage&username=john&password=Xc3ffs&recipient=%2B362013
24567&messagetype=SMS:TEXT&messagedata=TestMessage
C:\Users\Zsolti\AppData\Local\Programs\Python\Python38-32\lib\site-packages\urllib3\connectionpool.py:981: InsecureRe
questWarning: Unverified HTTPS request is being made to host '127.0.0.1'. Adding certificate verification is strongly
 advised. See: https://urllib3.readthedocs.io/en/latest/advanced-usage.html#ssl-warnings
  warnings.warn(
Http response received:
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE smsapi PUBLIC "-//OZEKI//DTD XML 1.0//EN" "http://www.ozekisms.com/DTD/smsapi.xml">
<response>
   <action>sendmessage</action>
   <data>                    <acceptreport>
                             <statuscode>0</statuscode>
                             <statusmessage>Message accepted for delivery</statusmessage>
                             <messageid>5ed3ec6e-6df6-4d42-8a22-d6b64111a304</messageid>
                             <recipient>+36201324567</recipient>
                    </acceptreport>
   </data>
</response>

C:\Users\Zsolti\Desktop\HttpTester>_
```

Figure 6 - Run the Python file in cmd

# Step 7 - Check the send result in the Ozeki log

In Ozeki 10 SMS Gateway, you can check easily what messages sent by your application, since the HTTP API service logs every event that occured during the time it is enabled. So, if you open the SMS Gateway, and select details of the HTTP API service, you will be able to see the events. As you can see it in Figure 7, the service logged an event, when the Python SMS example program sent the HTTP request to the service.



OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=service_details&services_id=88a08c46-a62b-4460-a2d9-7d999b0f4980&service_id=5f8f7...
SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode. To use SMS you need a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
http_API_1
Events
Configure
Help
http_API_1 details
Events
Configure
Help
Events
Update
Start
SMS Gateway
15:50

Connection

Name: http_API_1
Type: Http API
Startup: ✓
Protocol: https
Port: 9508
Enabled:

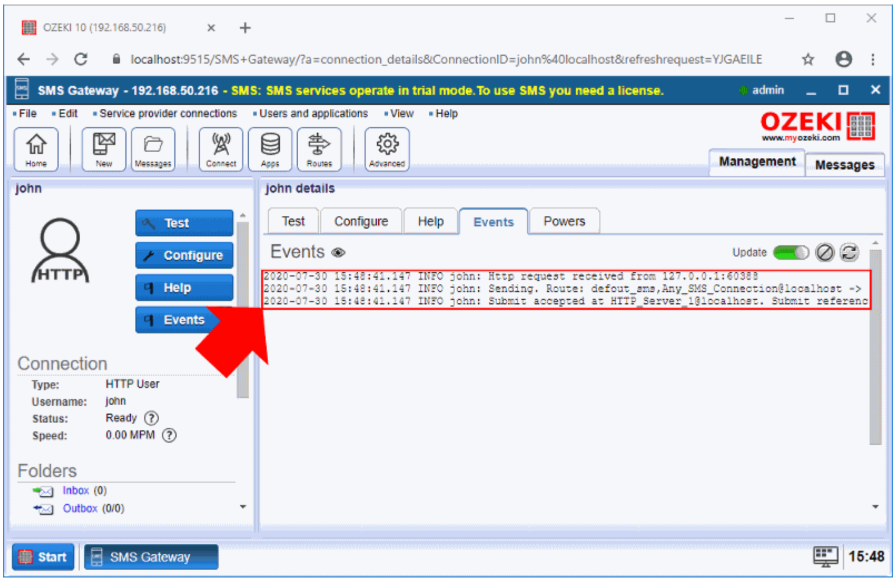
Description

The HTTP API allows you to accept connections from your customers, who can submit HTTP GET and HTTP POST requests.

```
2020-07-30 15:49:47.420 INFO http_API_1: 127.0.0.1:60429 GET /api?action=sendmessage&username=john&
```

Figure 7 - Check the logs of the HTTP API Service

The process of the message sending can be also viewed back by the events. For that, open the HTTP API User connection, that you had to configure before. In Figure 8, you can see that how the connection handles the HTTP request and send the message to the recipient that you specified in your Python program.



OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=connection_details&ConnectionID=john%40localhost&refreshrequest=YJGAEILE
SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode. To use SMS you need a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
john
HTTP
Test
Configure
Help
Events
john details
Test
Configure
Help
Events
Powers
Events
Update
Folders
Inbox (0)
Outbox (0/0)
Start
SMS Gateway
15:48

Connection

Type: HTTP User
Username: john
Status: Ready
Speed: 0.00 MPM

```
2020-07-30 15:48:41.147 INFO john: Http request received from 127.0.0.1:60388
2020-07-30 15:48:41.147 INFO john: Sending. Route: defout_sms,Any_SMS_Connection@localhost ->
2020-07-30 15:48:41.147 INFO john: Submit accepted at HTTP_Server_1@localhost. Submit referenc
```

Figure 8 - Check the logs of the HTTP API User

# Receive SMS in Python

***This document provides you some useful information about how you can get the messages received by the SMS Gateway using your Python application. The Python SMS example shows how it collects the messages from your inbox folder via HTTP Requests and how it makes it available for further usage. The page contains a video tutorial and a step by step guide which takes not more than five minutes to complete. Let's start right now!***

## What is a Python SMS API?

The Python SMS API creates the opportunity to send SMS messages from a Python script. By using this Python SMS API, your Python script can send HTTP requests to the SMS Gateway which sends the SMS message.

## Prerequisites

- Installed Python
- Installed Ozeki 10 SMS Gateway
- A configured HTTP Server connection

## Receive SMS in Python

1. Open Notepad application
2. Copy-Paste the example code from this page
3. Save the file as ReceiveSMS.py
4. Open Ozeki SMS Gateway and select HTTP Server connection
5. Open the HTML form of the HTTP Server connection
6. Send some test messages
7. Open Command Prompt
8. Type 'python ReceiveSMS.py' in the Command Prompt and press Enter

This example Python SMS source code below is free to use in your projects and you can modify it as well to fit into your solution. The code initiates a HTTP request to get the messages from the SMS Gateway. To test the example just follow the instructions below or watch the video tutorial which clearly demonstrates all steps that have to be taken to run the Python SMS script on your computer.

```python
import requests
import ssl
import xml.etree.ElementTree as ET

def main():
    username = "john"
    password = "Xc3ffs"
    httpUrl = "https://127.0.0.1:9508/"
    folder = "inbox";
    limit = "3";

    sendString = (httpUrl + "api?action=receivemessage&username="
        + username + "&password="
        + password + "&folder=" + folder + "&limit="
        + limit + "&afterdownload=delete")

    print("Sending html request: " + sendString + "\n")
    requests.packages.urllib3.disable_warnings()

    response = requests.get(sendString, verify=False)
    print("Http response received: ")
    DisplayMessages(response.text)

def DisplayMessages(response):
    root = ET.fromstring(response)
    if root.findall('data/message/*') == []:
        print('The inbox is empty')
        return

    for child in root.findall('data/message'):
        sender = child.find('originator').text
```

```
            text = child.find('messagedata').text
            DisplayMessage(sender, text)

def DisplayMessage(sender, text):
    print(sender + ": " + text)

if __name__=="__main__":
   main()
```

# Step 1 - Open Notepad

The first step of this guide is to open Notepad on your computer. This application is needed to create the Python file that you will execute at the end of the guide. So, as Figure 1 demonstrates, just click on the icon of the Notepad application on the desktop to open it.

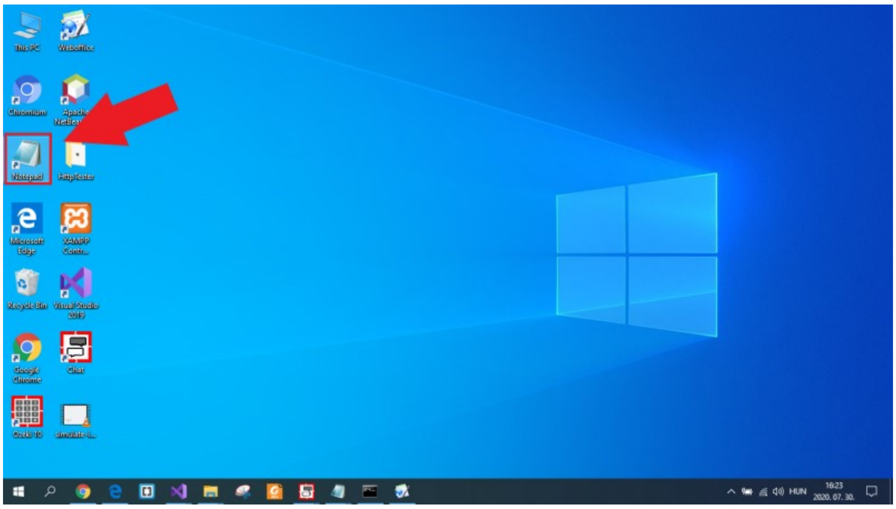
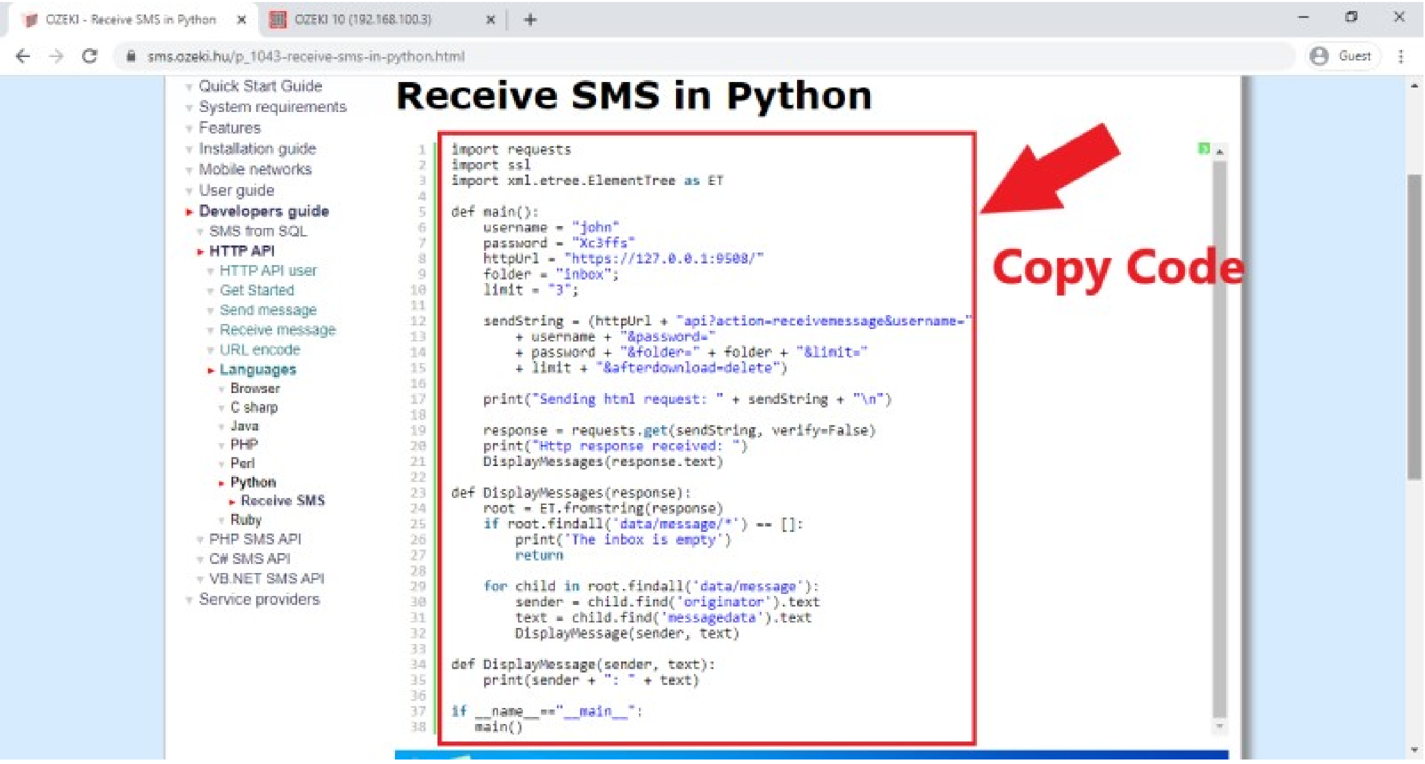


Figure 1 - Open notepad

# Step 2 - Copy the Python code

Next, you need to get the Python example script from this page. This is an easy task to do since you need to just copy it. So, to do that, just go to the example code on this page, mark out the whole source code, and like in Figure 2, press Ctrl+C on your keyboard. This action copies the source code to your clipboard and makes it ready to place it into a text file.



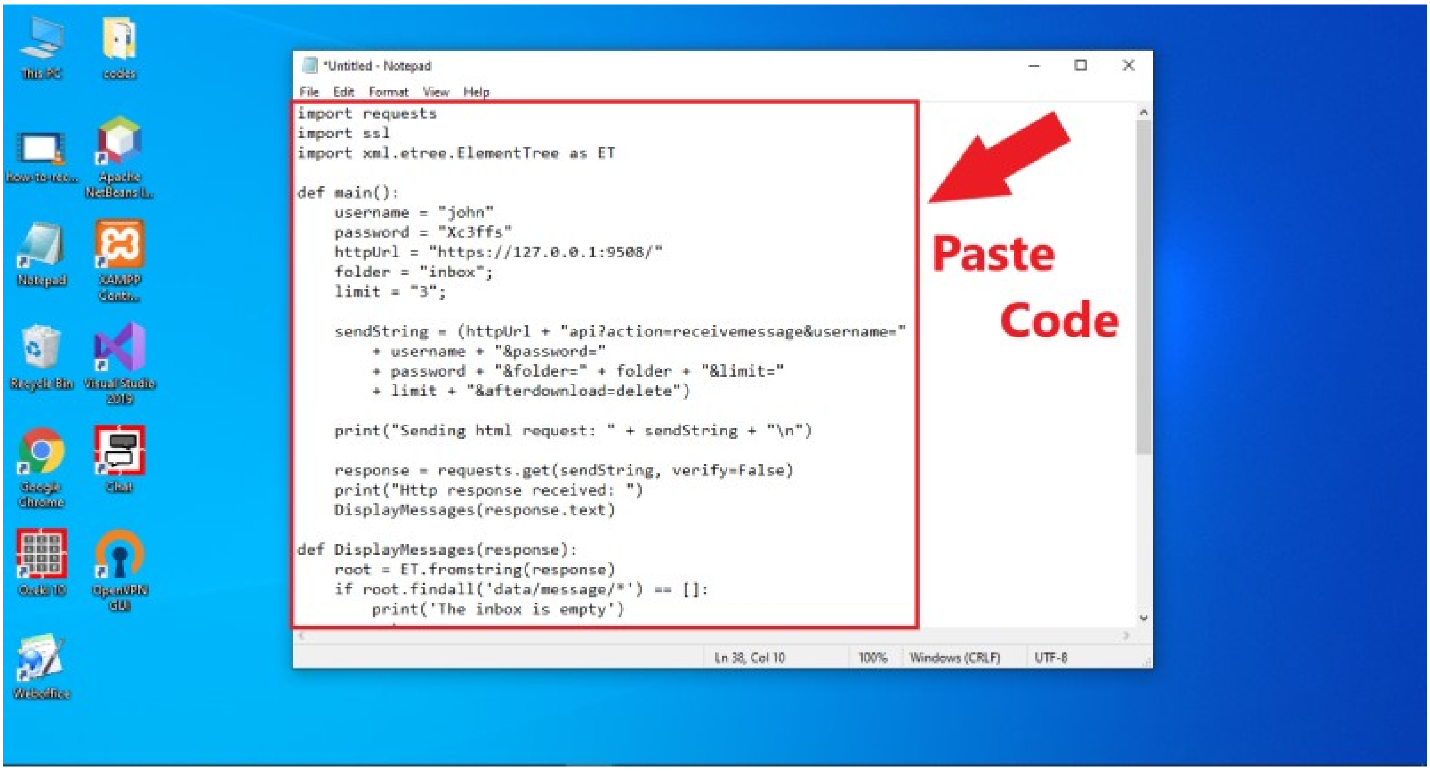
Receive SMS in Python

```
import requests
import ssl
import xml.etree.ElementTree as ET

def main():
    username = "john"
    password = "Xc3ffs"
    httpUrl = "https://127.0.0.1:9508/"
    folder = "inbox";
    limit = "3";

    sendString = (httpUrl + "api?action=receivemessage&username="
        + username + "&password="
        + password + "&folder=" + folder + "&limit="
        + limit + "&afterdownload=delete")

    print("Sending html request: " + sendString + "\n")

    response = requests.get(sendString, verify=False)
    print("Http response received: ")
    DisplayMessages(response.text)

def DisplayMessages(response):
    root = ET.fromstring(response)
    if root.findall('data/message/*') == []:
        print('The inbox is empty')
        return

    for child in root.findall('data/message'):
        sender = child.find('originator').text
        text = child.find('messagedata').text
        DisplayMessage(sender, text)

def DisplayMessage(sender, text):
    print(sender + ": " + text)

if __name__=="__main__":
   main()
```

Figure 2 - Copy source code from the website

## Step 3 - Paste the code into a text file

Next, you need to paste the Python source code into the text document that you opened in Step 1. This action can be performed by using the Ctrl+V shortcut which pastes the source code into the text file from the clipboard. If you have done it correctly, the Python code will be in the text file as Figure 3 shows that.

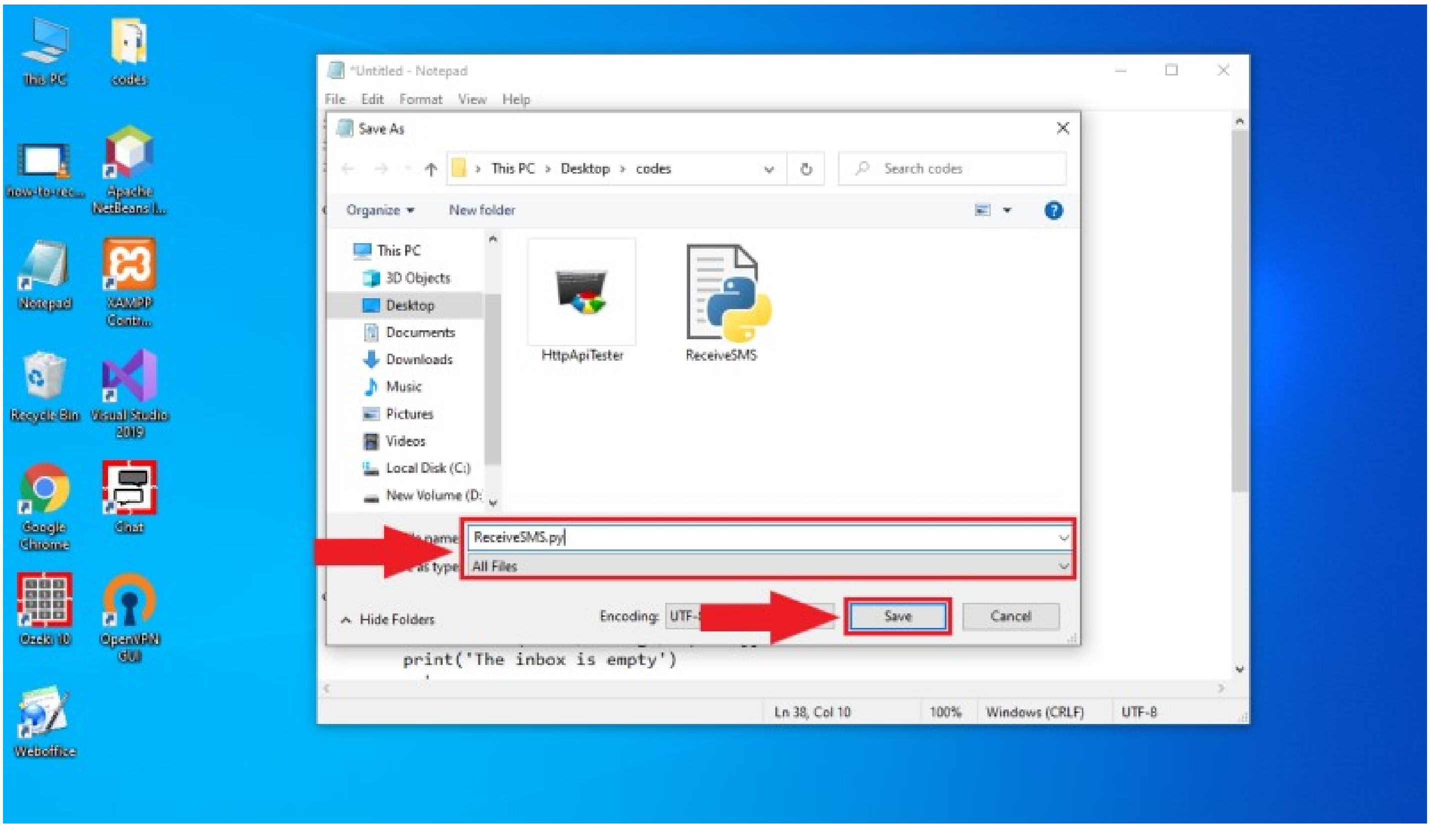


```
import requests
import ssl
import xml.etree.ElementTree as ET

def main():
    username = "john"
    password = "Xc3ffs"
    httpUrl = "https://127.0.0.1:9508/"
    folder = "inbox";
    limit = "3";

    sendString = (httpUrl + "api?action=receivemessage&username="
        + username + "&password="
        + password + "&folder=" + folder + "&limit="
        + limit + "&afterdownload=delete")

    print("Sending html request: " + sendString + "\n")

    response = requests.get(sendString, verify=False)
    print("Http response received: ")
    DisplayMessages(response.text)

def DisplayMessages(response):
    root = ET.fromstring(response)
    if root.findall('data/message/*') == []:
        print('The inbox is empty')
```

Figure 3 - Paste source code to notepad

## Step 4 - Save the text as a Python file

At this point, you have got a text file that contains the source code, but it is not in the right file format yet. Notepad can save files in any format, so you can solve this problem quite easily. So, select the 'Save as' option in Notepad, and save the file with the '.py' file extension. This procedure is demonstrated in Figure 4.



File name: ReceiveSMS.py

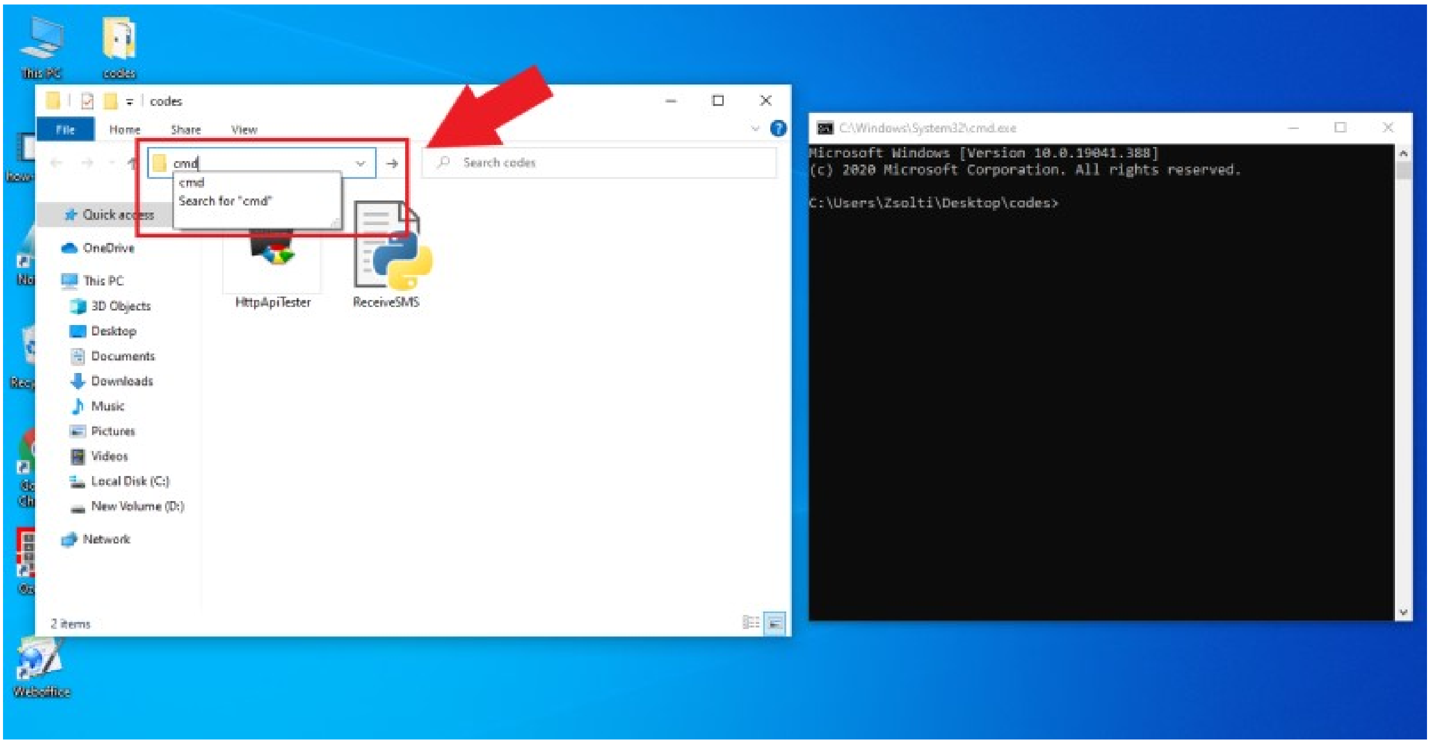
Save as type: All Files

```
print('The inbox is empty')
```

Figure 4 - Save python file as ReceiveSMS.py

## Step 5 - Open Command Prompt

Now, the Python script is ready to use, but you need an environment where you can execute the script. For that purpose, you can simply use the Command Prompt. To open it, just navigate to the folder where you saved the Python file with the File Explorer, and like in Figure 5, type 'cmd' into the address bar of the File Explorer. If you press Enter, the Command Prompt shows up in a new window.

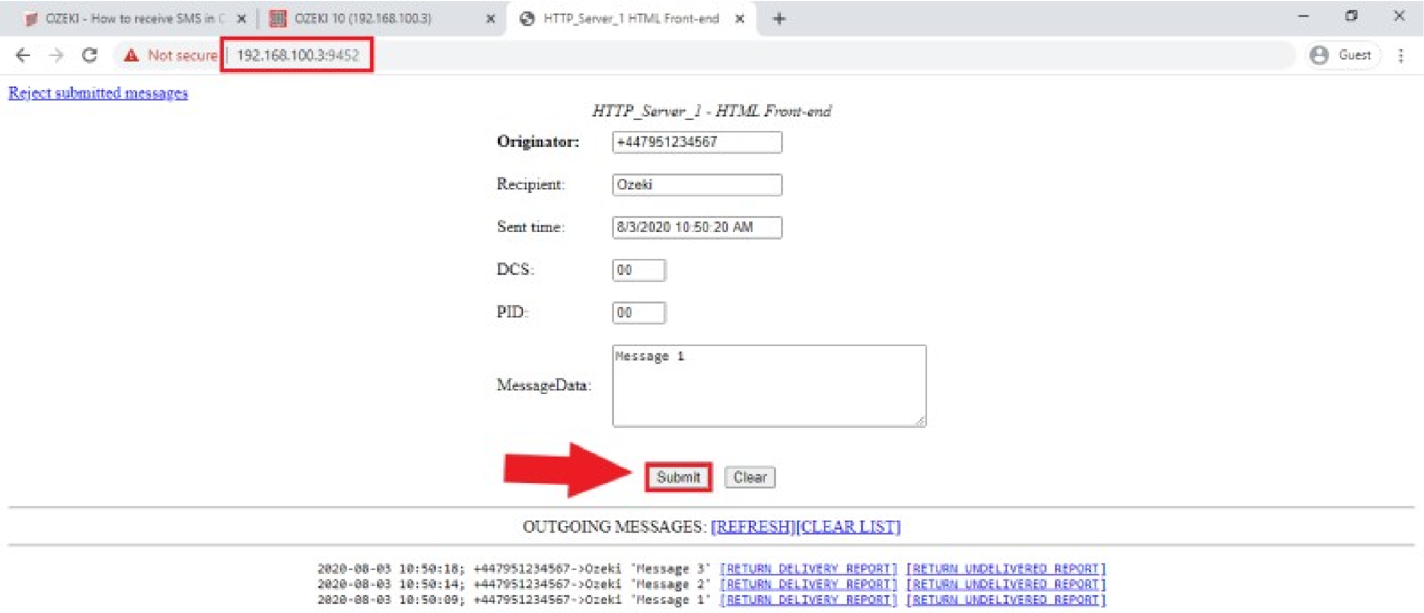


```
Microsoft Windows [Version 10.0.19041.388]
(c) 2020 Microsoft Corporation. All rights reserved.

C:\Users\Zsolti\Desktop\codes>
```

Figure 5 - Open command promt

## Step 6 - Send some test messages

After you set up the Python file, you need some message to your inbox folder. To finish the test quickly, you can simulate incoming messages with the HTTP Server connection in SMS Gateway. If you haven't configured this connection yet, here you can see how to configure a HTTP Server connection. So, open the SMS Gateway, and select the HTTP Server connection. In the menu of the connection, open the HTML form (Figure 6). Here, you can write some messages to the SMS Gateway by setting the recipient to 'Ozeki'.


OZEKI - How to receive SMS in C
OZEKI 10 (192.168.100.3)
HTTP_Server_1 HTML Front-end
Not secure
192.168.100.3:9452
Guest
Reject submitted messages

HTTP_Server_1 - HTML Front-end

Originator: +447951234567

Recipient: Ozeki

Sent time: 8/3/2020 10:50:20 AM

DCS: 00

PID: 00

MessageData: Message 1

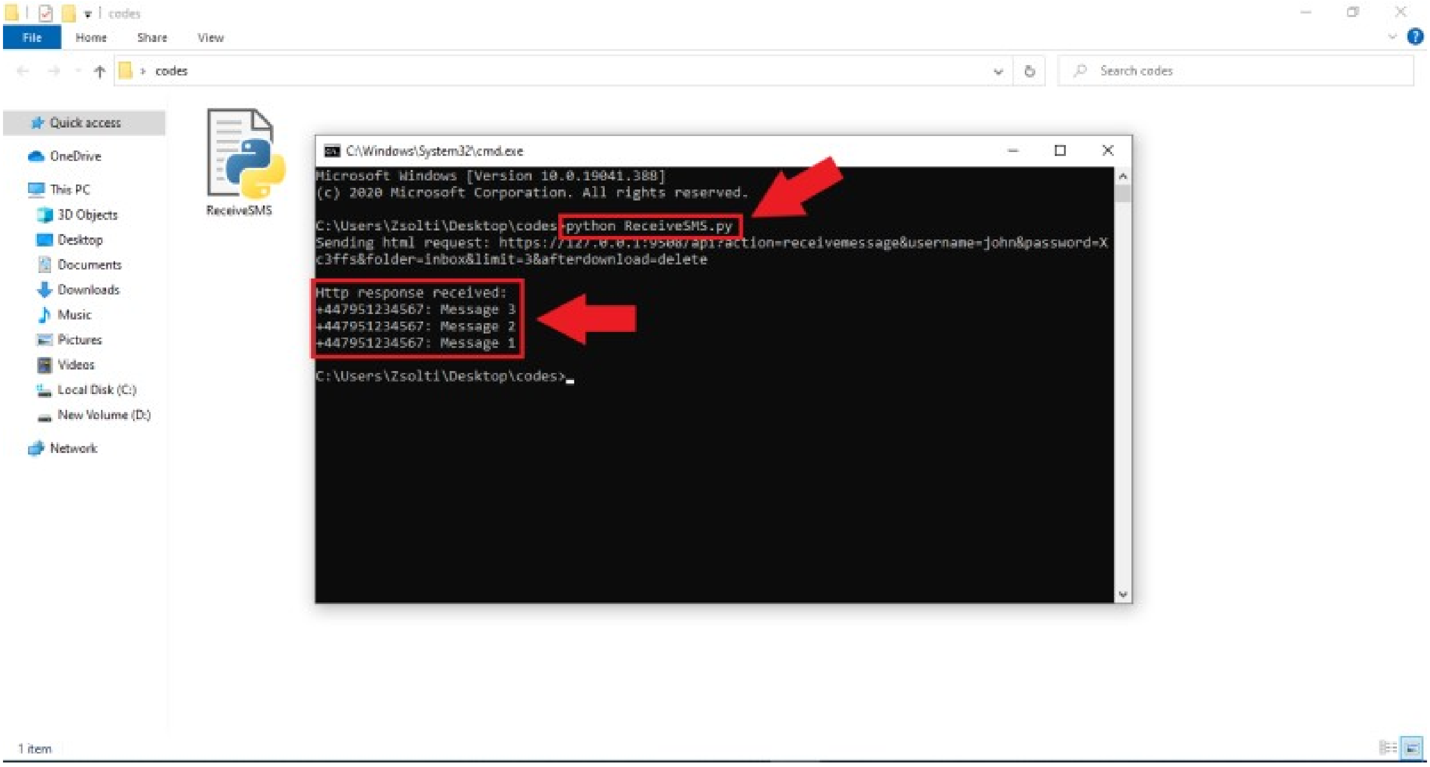
OUTGOING MESSAGES: [REFRESH][CLEAR LIST]

```
2020-08-03 10:50:18; +447951234567->Ozeki 'Message 3' [RETURN DELIVERY REPORT] [RETURN UNDELIVERED REPORT]
2020-08-03 10:50:14; +447951234567->Ozeki 'Message 2' [RETURN DELIVERY REPORT] [RETURN UNDELIVERED REPORT]
2020-08-03 10:50:09; +447951234567->Ozeki 'Message 1' [RETURN DELIVERY REPORT] [RETURN UNDELIVERED REPORT]
```

Figure 6 - Simulate some incoming SMS

## Step 7 - Run the Python script

The last thing that you need to do to complete this guide is to run the Python script in the Command Prompt. For that, just type 'python *filename*.py' into the Command Prompt as you can see it in Figure 7, and hit Enter. The execution of the Python SMS example prints the HTTP request that was initiated by the Python program. Then, you will be able to see the messages from the SMS Gateway as well.



```
Microsoft Windows [Version 10.0.19041.388]
(c) 2020 Microsoft Corporation. All rights reserved.

C:\Users\Zsolti\Desktop\codes>python ReceiveSMS.py
Sending html request: https://127.0.0.1:9508/api?action=receivemessage&username=john&password=X
c3ffs&folder=inbox&limit=3&afterdownload=delete

Http response received:
+447951234567: Message 3
+447951234567: Message 2
+447951234567: Message 1

C:\Users\Zsolti\Desktop\codes>_
```

Figure 7 - Run python code in command prompt

# How to send SMS from Ruby

***Let's see how you can use Ruby to send SMS messages with HTTP requests. The example code demonstrates that you only need a couple of lines of code to be able to set up the details of the message and send an HTTP request. Then, the SMS Gateway receives the request and delivers the SMS to the recipient. Lastly, the SMS Gateway sends the response back to the Ruby program which prints that response to the console.***

## What is the Ruby SMS API?

The Ruby SMS API allows you to send SMS messages from your Ruby program. The Ruby SMS API initiates HTTP requests and sends them to the SMS Gateway that delivers the message to the recipient.

## Prerequisites

- Installed Ruby
- Installed Ozeki 10 SMS Gateway
- A configured HTTP API User

## Send SMS from Ruby

1. Open Notepad
2. Create a new file
3. Copy-Paste the example code below
4. Save the file as HttpApiTester.rb
5. Open Command Prompt
6. Navigate to the folder where you saved the Ruby file
7. Type 'ruby HttpApiTester.rb' in the Command Prompt
8. Press Enter to execute the Ruby program

## Ruby SMS source code example

This simple example source code below, written in Ruby programming language is free to use, you can simply implement it into your project or you can modify the source code to use it for other projects or applications. To run the example code and see how to send HTTP requests from Ruby, you just need to create a new Ruby file and place this example code into that file. To execute the Ruby program, just run it from the Command Prompt. These operations stated and demonstrated in the steps and the video sections.

```ruby
require 'uri'
require 'net/http'

username = "john"
password = "Xc3ffs"
messagetype = "SMS:TEXT"
httpUrl = "https://127.0.0.1:9508/"
recipient = URI.encode_www_form_component("+36201324567")
messagedata = URI.encode_www_form_component("TestMessage")

sendString = "#{httpUrl}api?action=sendmessage&username=#{username}&password="\
             "#{password}&recipient=#{recipient}&messagetype="\
             "#{messagetype}&messagedata=#{messagedata}"

puts("Sending http request #{sendString}")

OpenSSL::SSL.send(:remove_const, :VERIFY_PEER)
OpenSSL::SSL.const_set(:VERIFY_PEER, OpenSSL::SSL::VERIFY_NONE)
url = URI(sendString)
response = Net::HTTP.get(url)
puts("Http response received:")
puts(response)
```

# Step 1 - Open a text editor

The first step is to execute the Ruby SMS example code above is to create a Ruby file which contains the source code. To perform this operation, you just need a text editor. You can download a custom text editor to do this, but the default Notepad application installed on Windows can do the job as well. So, as you can see it in Figure 1, just click on the icon of Notepad to open it.

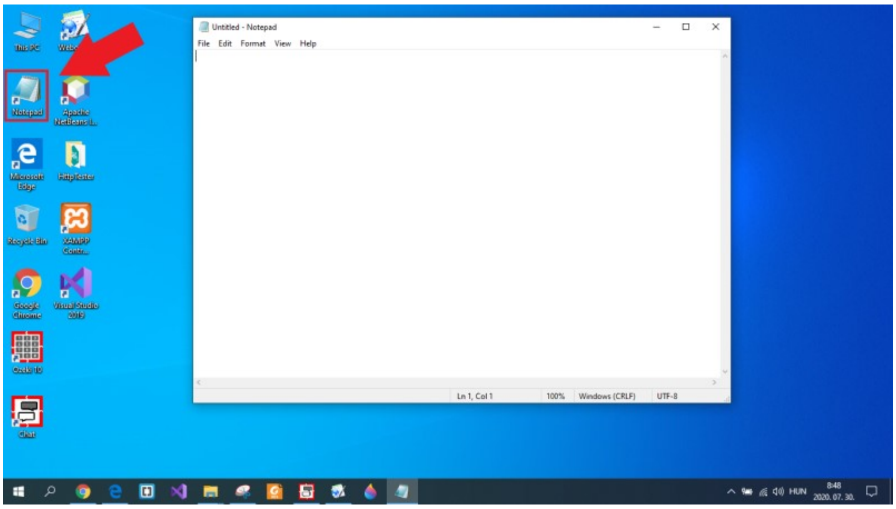
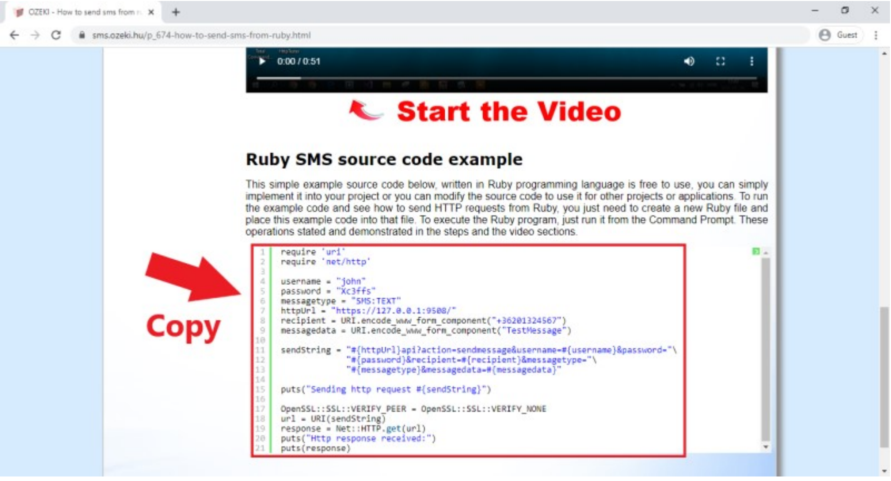


Figure 1 - Open a text editor

# Step 2 - Copy the Ruby source code

The next step of the guide is to get the Ruby source code. This can be easily done, since the code can be found on this page, so you just need to copy that. So, as Figure 2 shows that, just go to the example code section, mark out the whole source code and press Ctrl+C on your keyboard to copy the Ruby code to the clipboard.



OZEKI - How to send sms from r...
sms.ozeki.hu/p_674-how-to-send-sms-from-ruby.html
Guest
0:00 / 0:51
Start the Video
Copy

**Ruby SMS source code example**

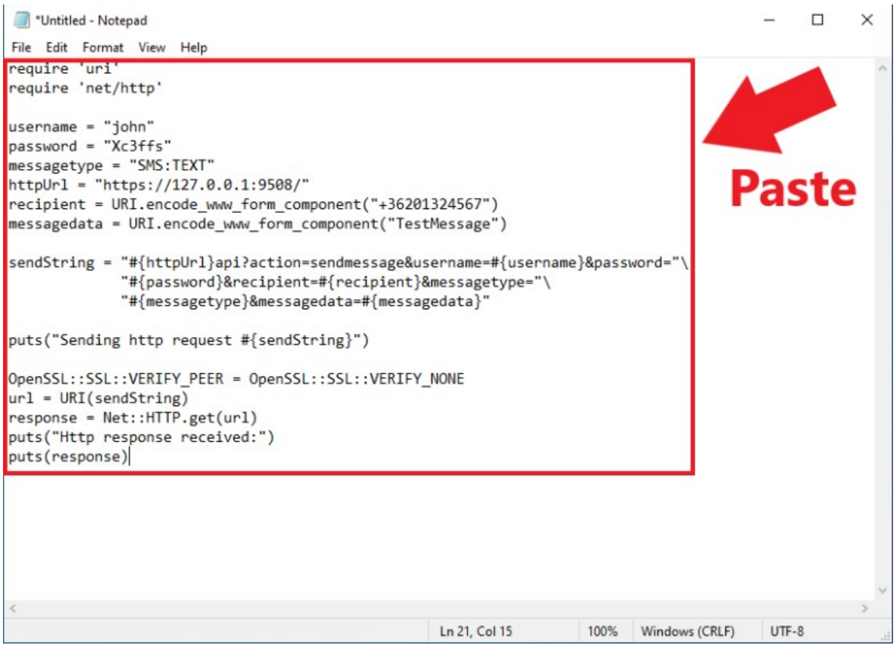
This simple example source code below, written in Ruby programming language is free to use, you can simply implement it into your project or you can modify the source code to use it for other projects or applications. To run the example code and see how to send HTTP requests from Ruby, you just need to create a new Ruby file and place this example code into that file. To execute the Ruby program, just run it from the Command Prompt. These operations stated and demonstrated in the steps and the video sections.

```
require 'uri'
require 'net/http'

username = "john"
password = "Xc3ffs"
messagetype = "SMS:TEXT"
httpUrl = "https://127.0.0.1:9508/"
recipient = URI.encode_www_form_component("+36201324567")
messagedata = URI.encode_www_form_component("TestMessage")

sendString = "#{httpUrl}api?action=sendmessage&username=#{username}&password="\
             "#{password}&recipient=#{recipient}&messagetype="\
             "#{messagetype}&messagedata=#{messagedata}"

puts("Sending http request #{sendString}")

OpenSSL::SSL::VERIFY_PEER = OpenSSL::SSL::VERIFY_NONE
url = URI(sendString)
response = Net::HTTP.get(url)
puts("Http response received:")
puts(response)
```

Figure 2 - Copy source code from the website

# Step 3 - Paste the Ruby code into your text editor

After you copied the Ruby code from this page, you need to paste it into your empty text file. To do that, you just need to navigate to the text file and just press Ctrl+V on your keyboard. By performing this action, the Ruby code will be pasted into your text file as Figure 3 shows that.

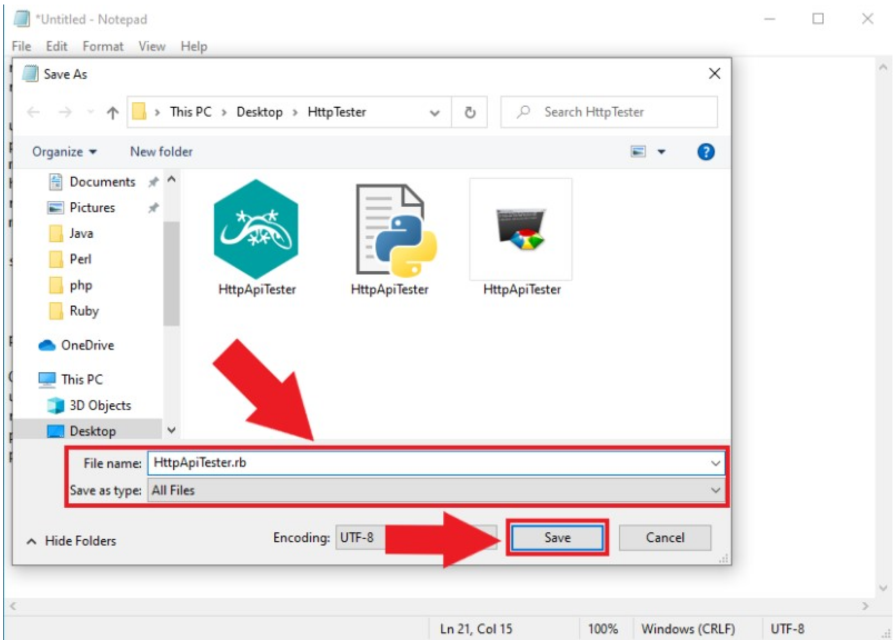


```
require 'uri'
require 'net/http'

username = "john"
password = "Xc3ffs"
messagetype = "SMS:TEXT"
httpUrl = "https://127.0.0.1:9508/"
recipient = URI.encode_www_form_component("+36201324567")
messagedata = URI.encode_www_form_component("TestMessage")

sendString = "#{httpUrl}api?action=sendmessage&username=#{username}&password="\
             "#{password}&recipient=#{recipient}&messagetype="\
             "#{messagetype}&messagedata=#{messagedata}"

puts("Sending http request #{sendString}")

OpenSSL::SSL::VERIFY_PEER = OpenSSL::SSL::VERIFY_NONE
url = URI(sendString)
response = Net::HTTP.get(url)
puts("Http response received:")
puts(response)
```

Figure 3 - Paste the program code into the text editor

# Step 4 - Save the Ruby file

After you placed the source code into the text file, the next step that you need to do, is to save the file using the file extension of Ruby which is '.rb'. So, just select the 'Save as' option in Notepad, name the file and add the '.rb' file extension as you can see it in Figure 4, and lastly, click on Save.



File name: HttpApiTester.rb
Save as type: All Files
Encoding: UTF-8

Figure 4 - Save the HttpApiTester Ruby file

# Step 5 - Open Command Prompt

The next step is to run the created Ruby file. This task can be easily done by using the Command Prompt. You can open this byx opening the File Explorer, navigating to the folder, where you saved the Ruby file, and typing 'cmd' in the address bar of the File Explorer like in Figure 5. If you press Enter at this point, the Command Prompt is going to show up and will be ready to use.

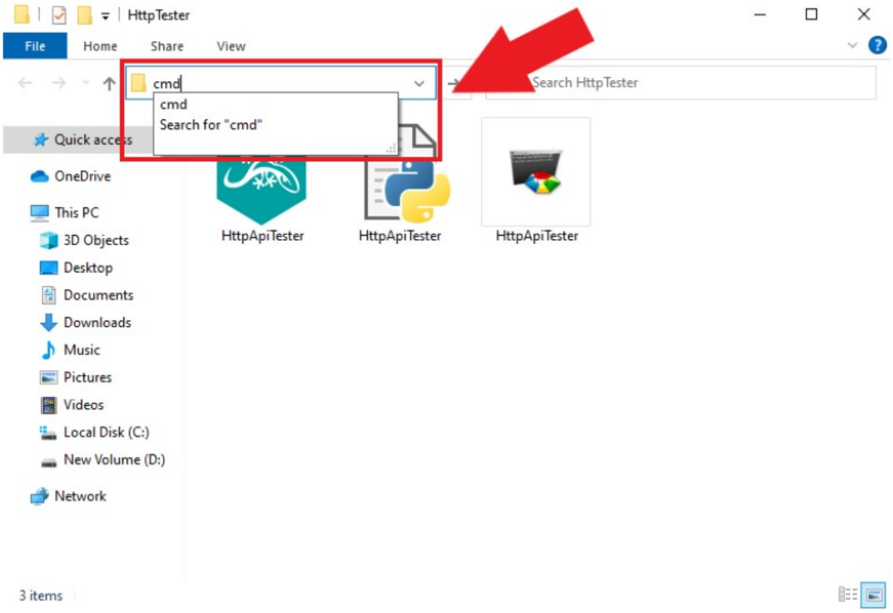
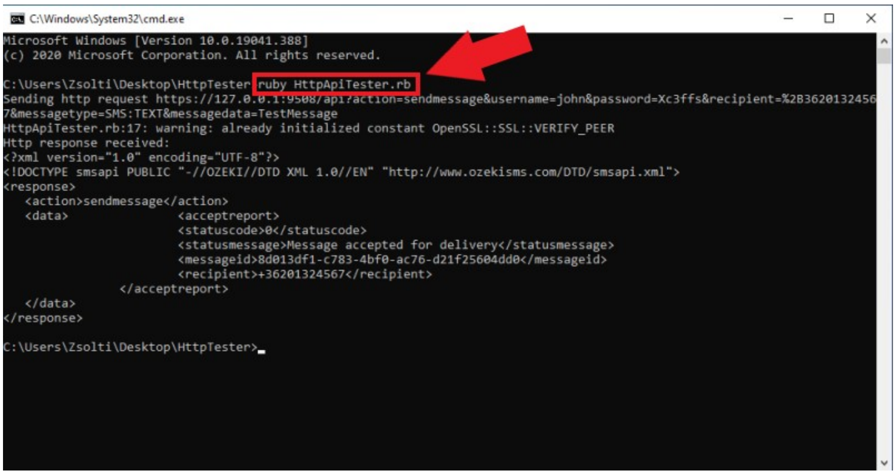


Figure 5 - Open Command Prompt

## Step 6 - Run the Ruby file

In the Command Prompt, you only need to execute one simple command to be able to run the Ruby SMS example. The command needs to be 'ruby *file name*.rb'. After you typed that command, just press Enter, and the command executes the Ruby program. The example program prints the HTTP request and the response message from the SMS Gateway to the Command Prompt.

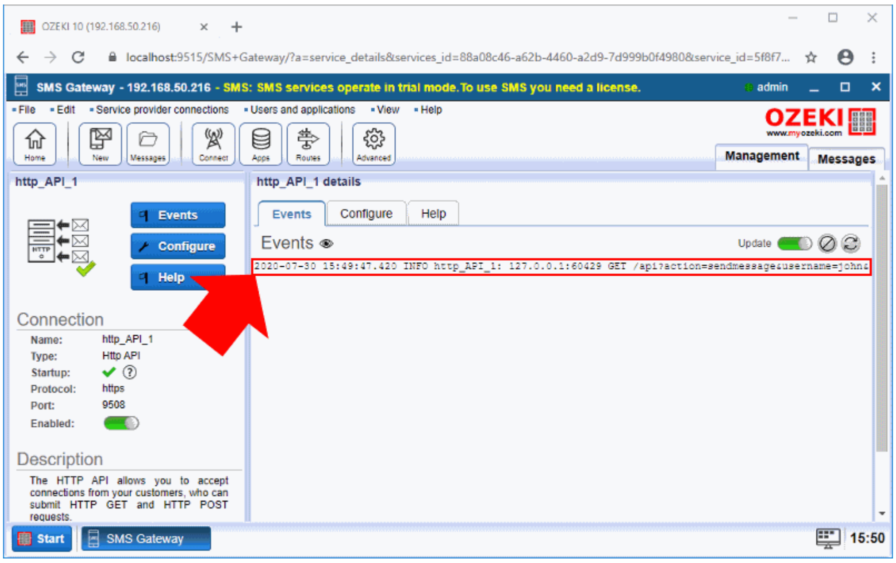


```
Microsoft Windows [Version 10.0.19041.388]
(c) 2020 Microsoft Corporation. All rights reserved.

C:\Users\Zsolti\Desktop\HttpTester ruby HttpApiTester.rb
Sending http request https://127.0.0.1:9508/api?action=sendmessage&username=john&password=Xc3ffs&recipient=%2B3620132456
7&messagetype=SMS:TEXT&messagedata=TestMessage
HttpApiTester.rb:17: warning: already initialized constant OpenSSL::SSL::VERIFY_PEER
Http response received:
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE smsapi PUBLIC "-//OZEKI//DTD XML 1.0//EN" "http://www.ozekisms.com/DTD/smsapi.xml">
<response>
   <action>sendmessage</action>
   <data>                    <acceptreport>
                             <statuscode>0</statuscode>
                             <statusmessage>Message accepted for delivery</statusmessage>
                             <messageid>8d013df1-c783-4bf0-ac76-d21f25604dd0</messageid>
                             <recipient>+36201324567</recipient>
                     </acceptreport>
   </data>
</response>

C:\Users\Zsolti\Desktop\HttpTester>_
```

Figure 6 - Run HttpApiTester.rb file in cmd

## Step 7 - Check the send result in the Ozeki log

In Ozeki 10 SMS Gateway, you can check easily what messages sent by your application, since the HTTP API service logs every event that occured during the time it is enabled. So, if you open the SMS Gateway, and select details of the HTTP API service, you will be able to see the events. As Figure 7 shows that, the service logged an event, when the Ruby example program sent the HTTP request to the service.



localhost:9515/SMS+Gateway/?a=service_details&services_id=88a08c46-a62b-4460-a2d9-7d999b0f4980&service_id=5f8f7...

SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode. To use SMS you need a license.

http_API_1 details

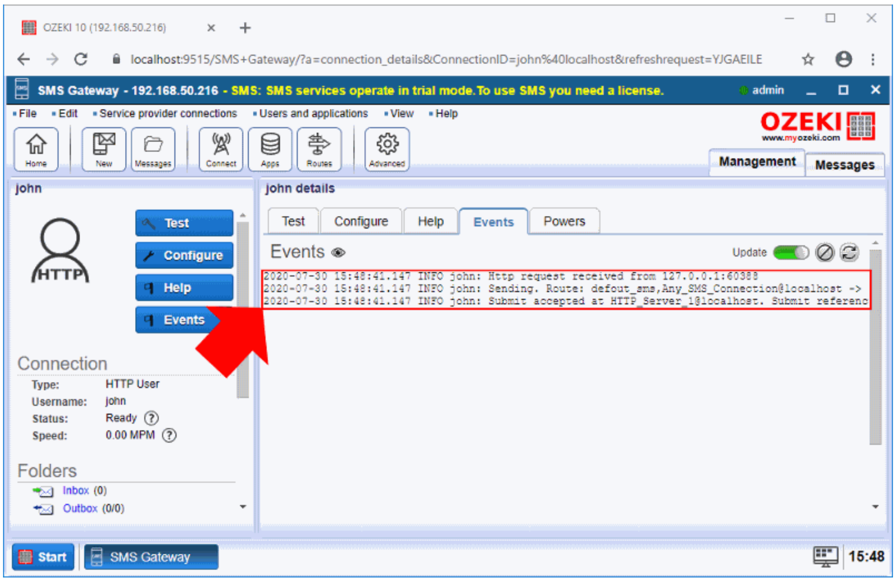
```
2020-07-30 15:49:47.420 INFO http_API_1: 127.0.0.1:60429 GET /api?action=sendmessage&username=john&
```

Connection
Name: http_API_1
Type: Http API
Protocol: https
Port: 9508

Description
The HTTP API allows you to accept connections from your customers, who can submit HTTP GET and HTTP POST requests.

Figure 7 - Check the logs of the HTTP API Service

The process of the message sending can be also viewed back by the events. For that, open the HTTP API User connection, that you had to configure before. Figure 8 demonstrates that how the connection handles the HTTP request and send the message to the recipient that you specified in your Ruby SMS example program.



localhost:9515/SMS+Gateway/?a=connection_details&ConnectionID=john%40localhost&refreshrequest=YJGAEILE

SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode. To use SMS you need a license.

john details

```
2020-07-30 15:48:41.147 INFO john: Http request received from 127.0.0.1:60388
2020-07-30 15:48:41.147 INFO john: Sending. Route: defout_sms,Any_SMS_Connection@localhost ->
2020-07-30 15:48:41.147 INFO john: Submit accepted at HTTP_Server_1@localhost. Submit referenc
```

Connection
Type: HTTP User
Username: john
Status: Ready
Speed: 0.00 MPM

Folders
Inbox (0)
Outbox (0/0)

Figure 8 - Check the logs of the HTTP API User

# Receive SMS in Ruby

***The guide on this page provides you the opportunity to learn how you can use a Ruby program to get the received SMS messages from the SMS Gateway. This can be done by using HTTP requests that collect the received SMS messages. The guide contains a video tutorial and a step by step guide which helps you to learn the way you can receive SMS messages with Ruby.***

## What is the Ruby SMS API?

The Ruby SMS API allows you to send SMS messages from your Ruby program. The Ruby SMS API initiates HTTP requests and sends them to the SMS Gateway that delivers the message to the recipient.

## Prerequisites

- Installed Ruby
- Installed Ozeki 10 SMS Gateway
- A configured HTTP Server connection

## Receive SMS in Ruby

1. Open Notepad application
2. Copy-Paste the example code from this page
3. Save the file as ReceiveSMS.rb
4. Open Ozeki SMS Gateway and choose HTTP Server connection
5. Open the HTML form of the HTTP Server connection
6. Send some test messages
7. Open Command Prompt
8. Type 'ruby ReceiveSMS.rb' in the Command Prompt and press Enter

You can freely use the Ruby example SMS code in your projects and you also allowed to modify it as well. If you wish to just run this example code to see how it works, just check the step by step guide below or the video tutorial on this page to see that it takes to execute the Ruby SMS example code on your computer.

```
require 'net/http'
require 'rexml/document'
include REXML

def DisplayMessages(response)
    xmldoc = Document.new(response)

    xmldoc.elements.each('response/data/message') do |value|
        begin
            sender = value.elements['originator'].text
            text = value.elements['messagedata'].text
            DisplayMessage(sender,text)
        rescue
            puts("The inbox is empty.")
        end
    end
end

def DisplayMessage(sender, text)
    puts("#{sender}: #{text}")
end

username = "john"
password = "Xc3ffs"
httpUrl = "https://127.0.0.1:9508/"
folder = "inbox";
limit = "3";

sendString = "#{httpUrl}api?action=receivemessage&username="\
             "#{username}&password="\
             "#{password}&folder=#{folder}&limit="\
             "#{limit}&afterdownload=delete"

```

```
puts("Sending http request #{sendString}")

OpenSSL::SSL.send(:remove_const, :VERIFY_PEER)
OpenSSL::SSL.const_set(:VERIFY_PEER, OpenSSL::SSL::VERIFY_NONE)

url = URI(sendString)
response = Net::HTTP.get(url)
puts("Http response received:")
DisplayMessages(response)
```

## Step 1 - Open Notepad

In this first step, you need to open a text editor application that can handle Ruby files. For that purpose, you can use the default Notepad application that can be found on all computers using Windows operating system. So, as Figure 1 shows that, just open it from your desktop.

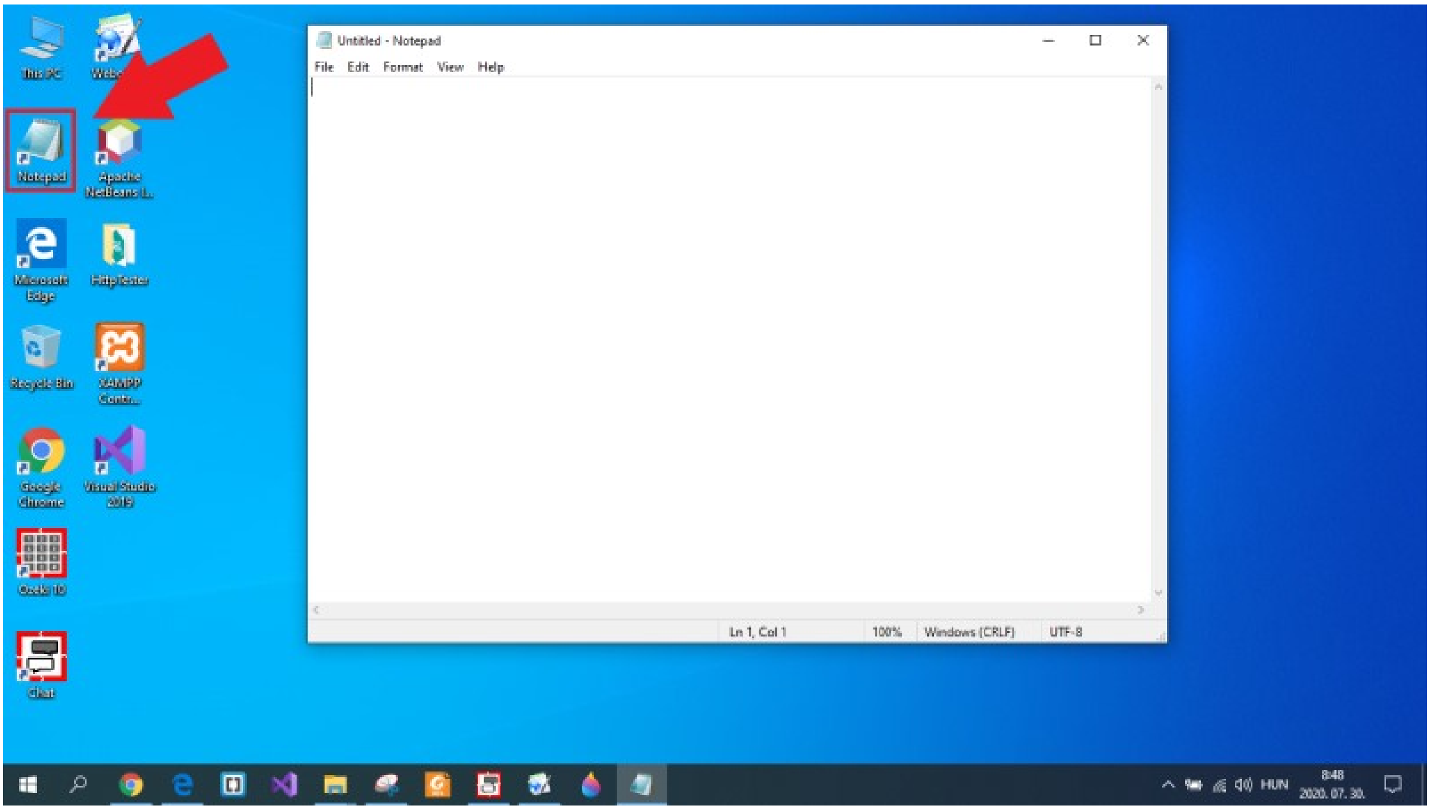
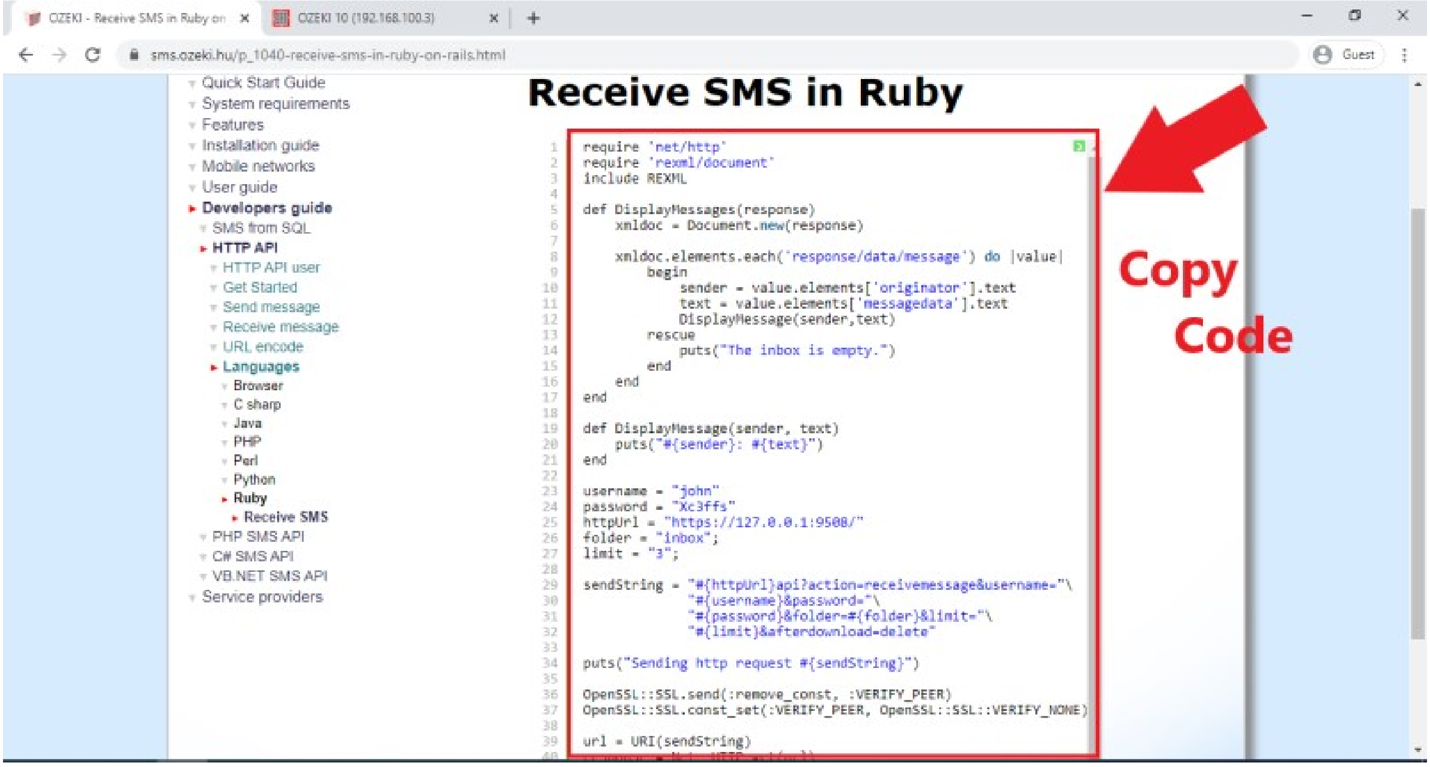


Figure 1 - Open notepad text editor

## Step 2 - Copy the Ruby code

Next, you need to copy the Ruby example code from this page. To do that, just scroll up to the example code section of this page, and mark out the whole example Ruby code. Next, as you can see it in Figure 2, press Ctrl+C on your keyboard to copy the Ruby code to the clipboard.

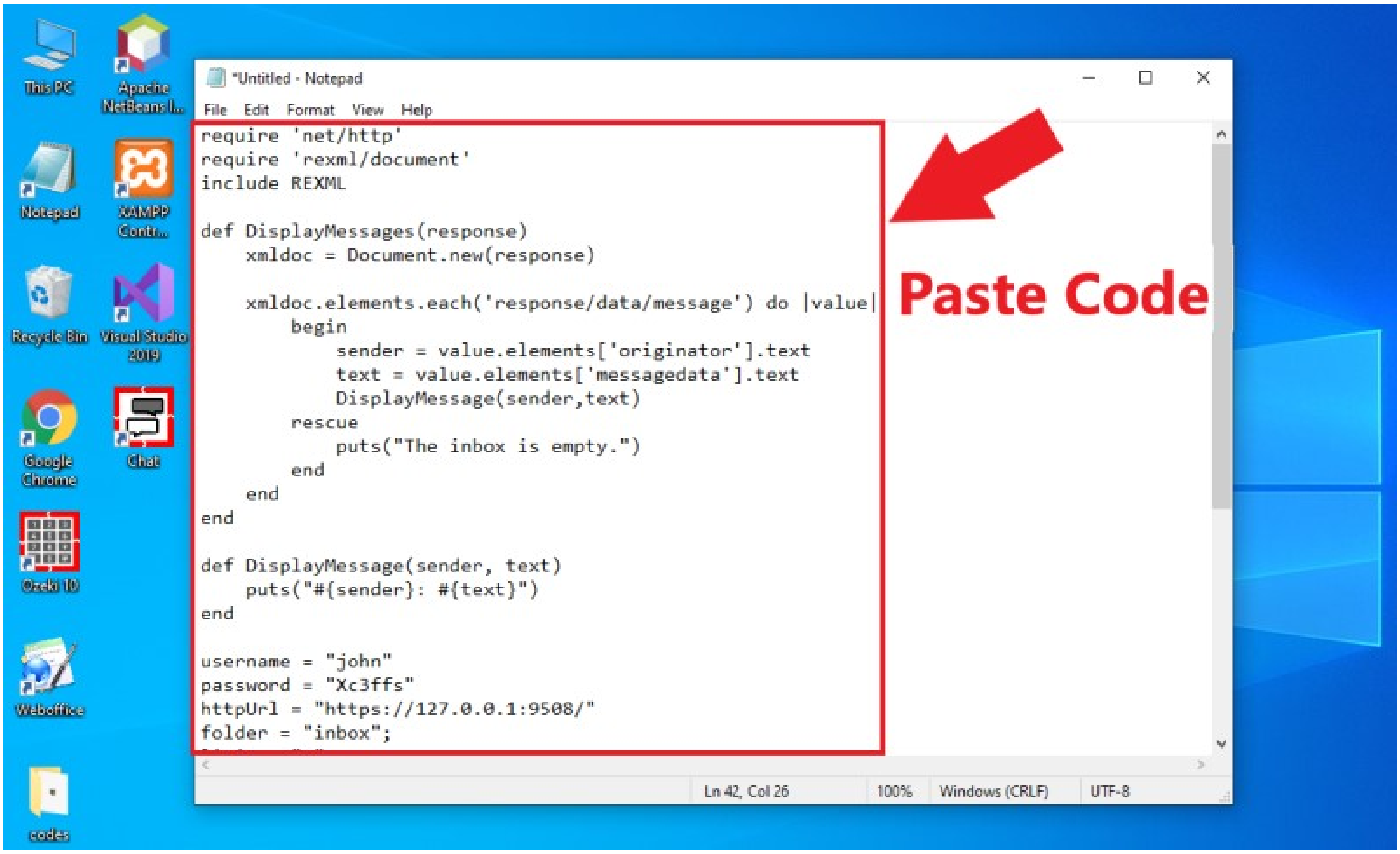


```
require 'net/http'
require 'rexml/document'
include REXML

def DisplayMessages(response)
    xmldoc = Document.new(response)

    xmldoc.elements.each('response/data/message') do |value|
        begin
            sender = value.elements['originator'].text
            text = value.elements['messagedata'].text
            DisplayMessage(sender,text)
        rescue
            puts("The inbox is empty.")
        end
    end
end

def DisplayMessage(sender, text)
    puts("#{sender}: #{text}")
end

username = "john"
password = "Xc3ffs"
httpUrl = "https://127.0.0.1:9508/"
folder = "inbox";
limit = "3";

sendString = "#{httpUrl}api?action=receivemessage&username="\
             "#{username}&password="\
             "#{password}&folder=#{folder}&limit="\
             "#{limit}&afterdownload=delete"

puts("Sending http request #{sendString}")

OpenSSL::SSL.send(:remove_const, :VERIFY_PEER)
OpenSSL::SSL.const_set(:VERIFY_PEER, OpenSSL::SSL::VERIFY_NONE)

url = URI(sendString)
```

Figure 2 - Copy ruby source code from the website

## Step 3 - Paste the code into the text file

After you copied the Ruby source code from this page, you need to place it into a text document. To do this, just go to the Notepad application that you opened before, and as you can see it in Figure 3, just press Ctrl+V on your keyboard to paste the whole Ruby source code into the text document.

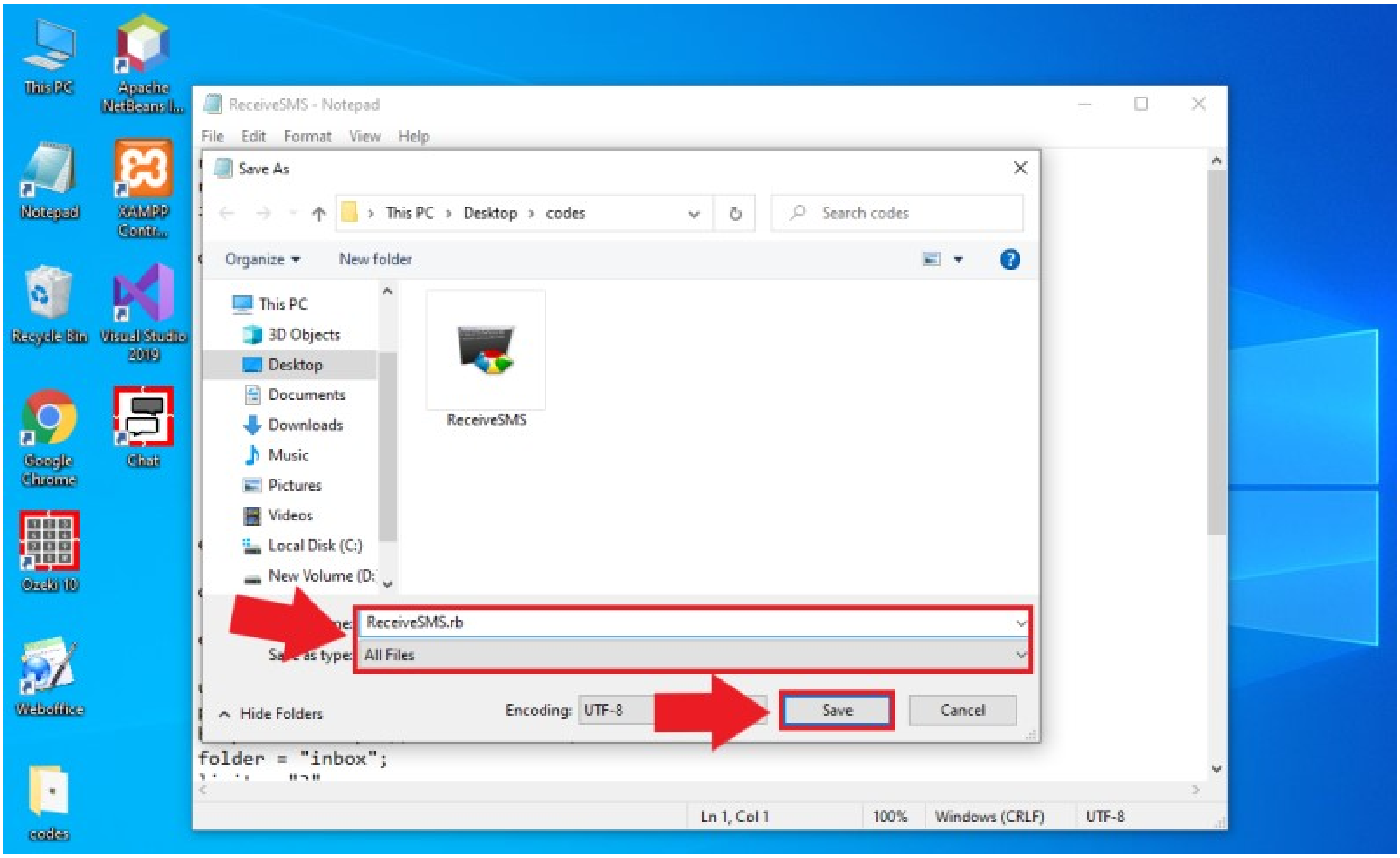


```
require 'net/http'
require 'rexml/document'
include REXML

def DisplayMessages(response)
    xmldoc = Document.new(response)

    xmldoc.elements.each('response/data/message') do |value|
        begin
            sender = value.elements['originator'].text
            text = value.elements['messagedata'].text
            DisplayMessage(sender,text)
        rescue
            puts("The inbox is empty.")
        end
    end
end

def DisplayMessage(sender, text)
    puts("#{sender}: #{text}")
end

username = "john"
password = "Xc3ffs"
httpUrl = "https://127.0.0.1:9508/"
folder = "inbox";
```

Figure 3 - Paste source code to notepad

## Step 4 - Save the text as a Ruby file

At this point, you have got a text file that contains the source code, but it is not in the right, Ruby file format yet. Notepad can save files in any format, so you can solve this problem quite easily. So, select the 'Save as' option in Notepad, and save the file with the '.rb' file extension. This procedure is demonstrated in Figure 4.



File name: ReceiveSMS.rb

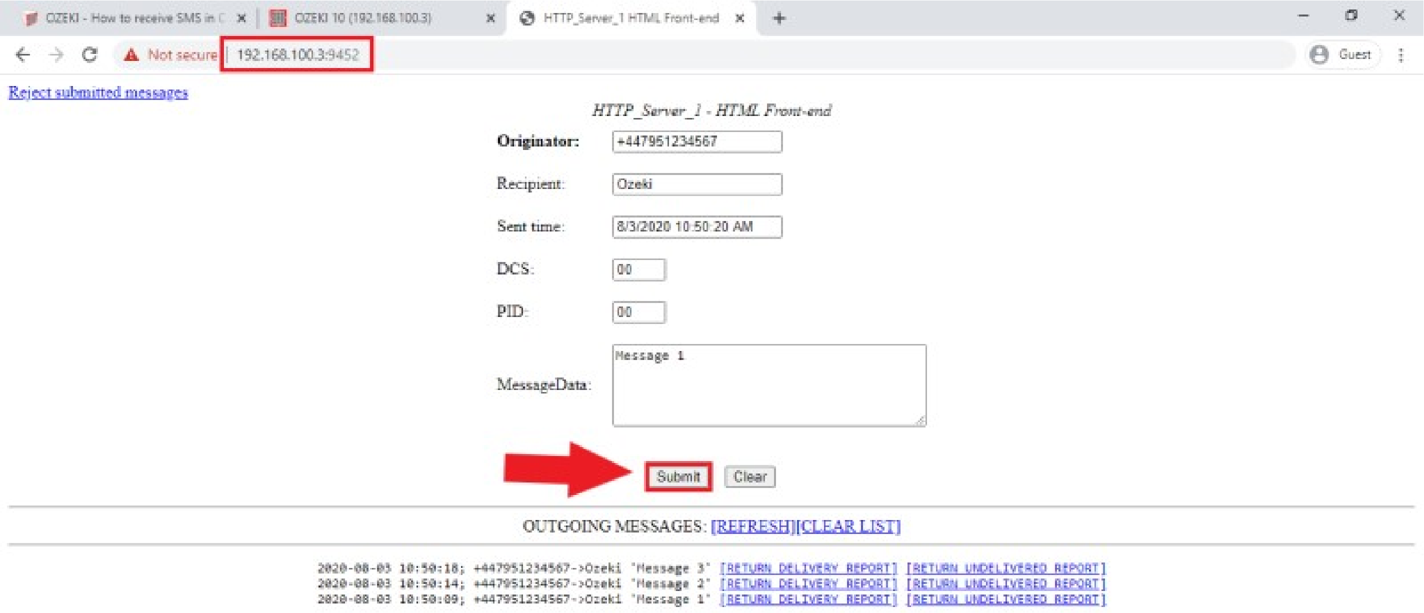
Save as type: All Files

```
folder = "inbox";
```

Figure 4 - Save ruby file as ReceiveSMS.rb

# Step 5 - Send some test messages

Now, your example code is ready to run, but first, you need to have some messages in your inbox folder. For testing purposes, you can use the HTTP Server connection to send some test messages to yourself and test the PHP example code. So, open the SMS Gateway and select the HTTP Server connection. Here, open the HTML form of the connection, and like in Figure 5, send some messages to the 'Ozeki' recipient.


OZEKI - How to receive SMS in C
OZEKI 10 (192.168.100.3)
HTTP_Server_1 HTML Front-end
Not secure
192.168.100.3:9452
Guest
Reject submitted messages

HTTP_Server_1 - HTML Front-end

Originator: +447951234567

Recipient: Ozeki

Sent time: 8/3/2020 10:50:20 AM

DCS: 00

PID: 00

MessageData: Message 1

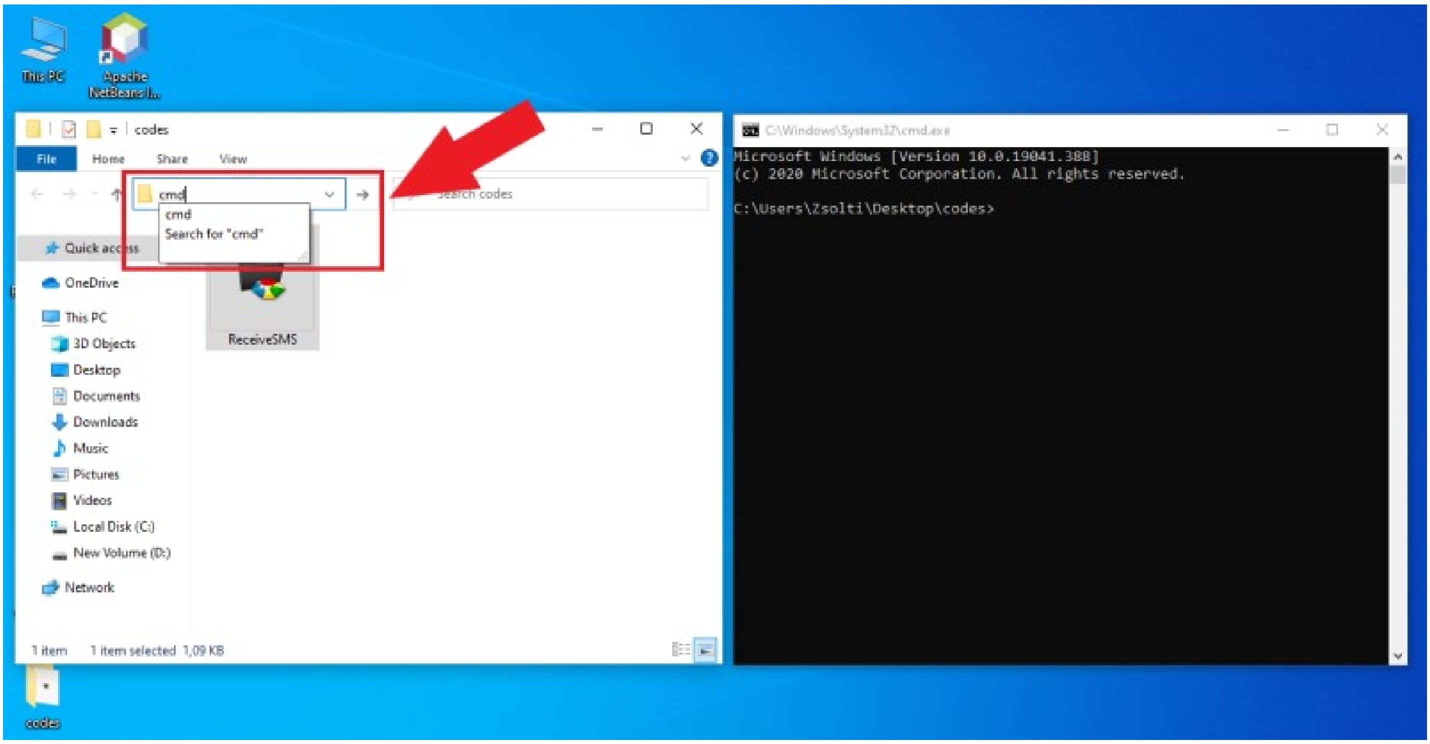
OUTGOING MESSAGES: [REFRESH][CLEAR LIST]

```
2020-08-03 10:50:18; +447951234567->Ozeki 'Message 3' [RETURN DELIVERY REPORT] [RETURN UNDELIVERED REPORT]
2020-08-03 10:50:14; +447951234567->Ozeki 'Message 2' [RETURN DELIVERY REPORT] [RETURN UNDELIVERED REPORT]
2020-08-03 10:50:09; +447951234567->Ozeki 'Message 1' [RETURN DELIVERY REPORT] [RETURN UNDELIVERED REPORT]
```

Figure 5 - Simulate some incoming SMS

# Step 6 - Open Command Prompt

The main step of the guide is to execute the Ruby program. To do that, you need to open the Command Prompt on your computer. So, first navigate to the folder, where you saved the Ruby file using the File Explorer. Here, as you can see it in Figure 6, type 'cmd' in the File Explorer, and if you press Enter, the Command Prompt opens on your computer.

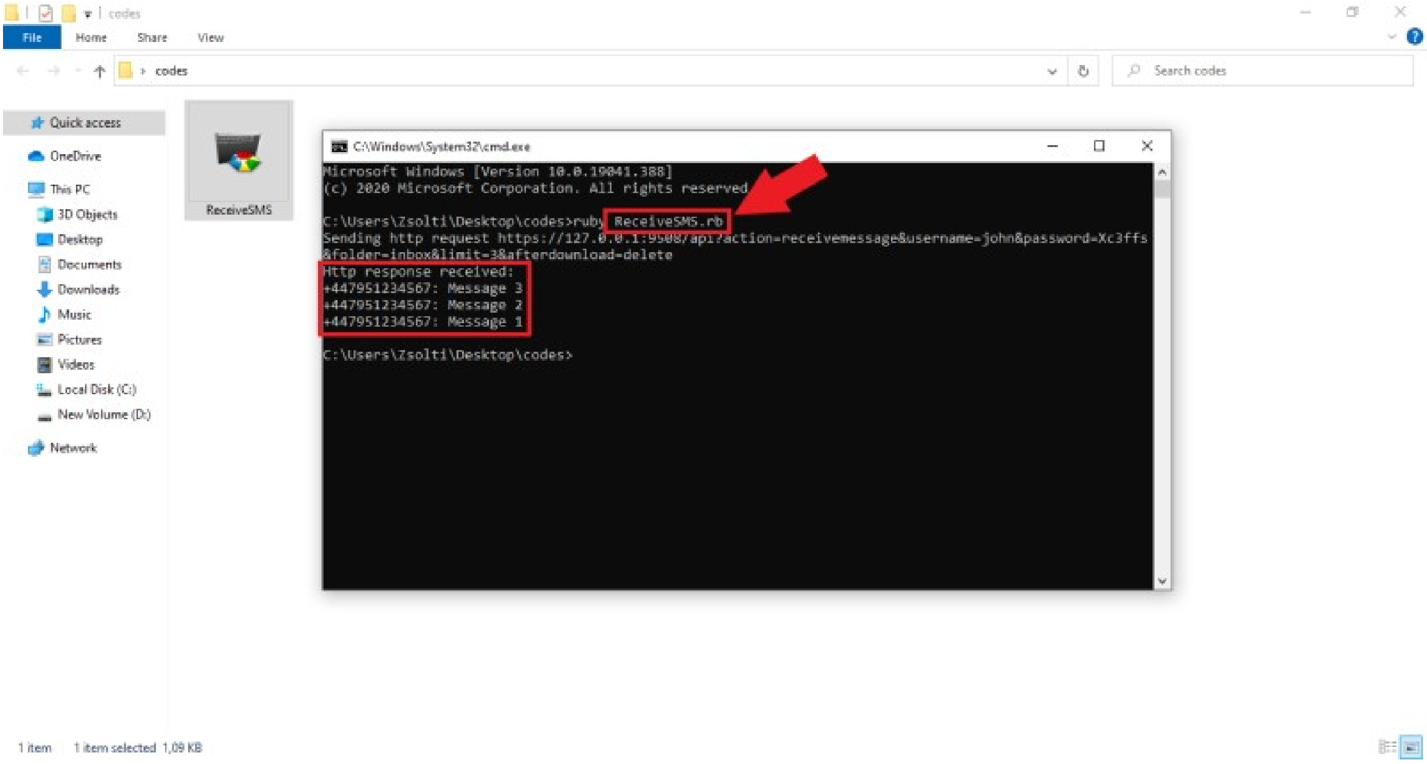


```
Microsoft Windows [Version 10.0.19041.388]
(c) 2020 Microsoft Corporation. All rights reserved.

C:\Users\Zsolti\Desktop\codes>
```

Figure 6 - Open command prompt

# Step 7 - Execute the Ruby program

In the Command Prompt, you just need to perform one line of command to execute the Ruby program. So, just type 'ruby *filename*.rb' into the Command Prompt and press Enter. By doing this, the Ruby example will be executed and you will be able to see the results in the console window (Figure 7). Here, you can see the HTTP request and the response to that request as well which contains all received messages.



```
Microsoft Windows [Version 10.0.19041.388]
(c) 2020 Microsoft Corporation. All rights reserved

C:\Users\Zsolti\Desktop\codes>ruby ReceiveSMS.rb
Sending http request https://127.0.0.1:9508/api?action=receivemessage&username=john&password=Xc3ffs
&folder=inbox&limit=3&afterdownload=delete
Http response received:
+447951234567: Message 3
+447951234567: Message 2
+447951234567: Message 1

C:\Users\Zsolti\Desktop\codes>
```

Figure 7 - Run ruby code in command prompt

# PHP SMS API

**You can easily build a website for SMS messaging with the Ozeki PHP API. This tutorial shows you how to use Ozeki SMS Gateway's PHP API. It will take you less then an hour to integrate the solution that can send and receive messages from a MySQL database.**

In most cases there is a database server behind PHP enabled websites and the implementation is done by PHP developers. **If you are already using a database server for your website, you should consider an SQL SMS Gateway PHP solution**.

## How to send / receive SMS messages from PHP through a database (🎥)

Please **create two database tables in a database for sending and receiving messages**. The default sender table in our example is ozekimessageout, while the default receiver table is ozekimessagein.

**If Ozeki SMS Gateway receives a message** then a new record will be inserted in the incoming message table. Your PHP application should easily find every available message record.

**When sending an SMS message** you should insert a new record into the outgoing message table, which is called ozekimessageout in this tutorial. It should be periodically checked by Ozeki SMS Gateway with a simple SQL SELECT statement to find messages that should be sent.

You can **set up a two way SMS communication** channel with this simple strategy (**Figure 1**).

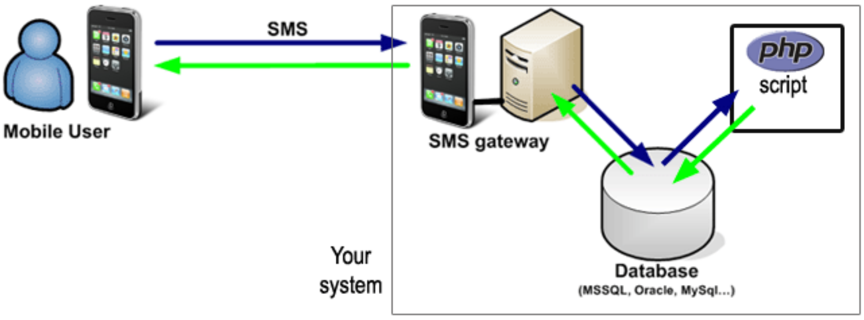


**Figure 1** - Ozeki SMS Gateway's Database + PHP SMS solution (Two way communication method)

Please **check out the video to see how to send SMS messages from a website.**
Check these sites for relevant information:
How to set up a Database User in Ozeki SMS Gateway
How to connect the Database User to your MySQL database
(you can use other database solutions as well)

## How to send/receive SMS messages from PHP through the HTTP API (🎥)

You can send or receive SMS messages by directly communicating with Ozeki SMS Gateway's HTTP API (**Figure 2**). This method does not need any database server connection.

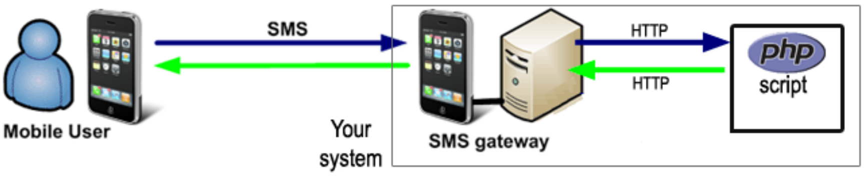


**Figure 2** - Ozeki SMS Gateway's HTTP API + PHP SMS solution (Two way communication method)

This solution has a great advantage, which is speed. The two way communication is direct and fast. It can also generate automatic SMS message responses as well.

**Here you can find a video and detailed guide for the HTTP API+PHP solution.**
This technique uses the 'sendmessage' action of the HTTP API.

# Use your website to send SMS through MySQL

**You can manage databases with SQL statements. You can send SQL statements through PHP. This makes you possible to manipulate database tables. For example PHP can add table records of SMS messages to send, but it also can read records of incoming messages.**

Download: MySQL_PHP_Example.zip

The PHP solution is intended for web developers with basic knowledge in PHP and SQL. You can download the PHP source code, so you can do this step-by-step tutorial containing basic instruction how to implement the solution.

## Why is SMS Gateway with a database and PHP a useful combo

This is a useful PHP solution for

- adding SMS functions to your website.
- adding SMS functions to your corporate intranet.
- creating automated SMS notifications.
- increasing website security by adding SMS login.

## Requirements

You will need to host an Ozeki SMS Gateway, a webserver and a MySQL server. You can host these function from the same computer or from two machines as you can see in the table below:

| | |
|---|---|
| PHP host: | Operating system: Linux or Windows<br>Webserver (Apache or IIS)<br>PHP<br>MySQL Server |
| Ozeki SMS Gateway host: | Operating system: Windows or Linux<br>.NET framework (if you use Windows) or Mono (if you use Linux)<br>Ozeki SMS Gateway |

Ozeki SMS Gateway can be obtained by opening the download page:
**Download Ozeki SMS Gateway!**

## How does it work

Before you start using this PHP application, **you should install Ozeki SMS Gateway and a MySQL Server on your computer**. You can use Windows or Linux. It is important to choose a hardware or software solution for SMS messaging. As a hardware solution a mobile phone or GSM modem is advised and for a software solution you can use your IP SMS service provider over the internet.

The PHP example you can find in this tutorial is capable to read records or insert new table records. This is very useful when reading incoming messages or writing outgoing messages. If you would like to see how the solution works, please look at **Figure 1**.

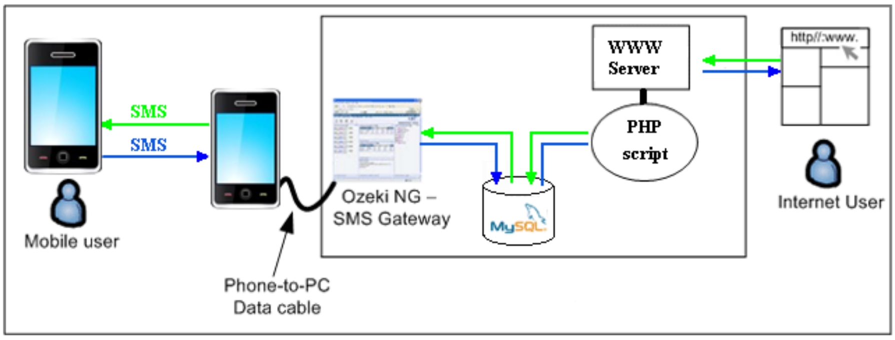


**Figure 1** - SMS messaging between Ozeki SMS Gateway, a MySQL database and PHP

You can see the 'Internet User' in the diagram, who will actually send and receive SMS messages through a browser. The internet user is connected to a PHP enabled web server, which is capable to manage databases on the SQL Server and these databases belong to Ozeki SMS Gateway. In the final step Ozeki SMS Gateway sends the message to a mobile phone through the GSM network.

The internet user can access the internet from anywhere. The only thing that matters is if they know the IP address or the URL of the computer running the PHP server and if they are authorized to log into the MySQL database, so they can insert the desired SMS message record which will be selected later on by Ozeki SMS Gateway to be sent out to the recipeint's cell phone by using the GSM network.

This workflow also works in the opposite direction. Where Ozeki SMS Gateway receives the SMS message from the GSM network and saves it in the MySQL database, so the PHP server can read it and show it on the internet user's screen.

## How to create a MySQL database

```
CREATE DATABASE ozekisms;

USE ozekisms;

CREATE TABLE ozekimessagein
(
id int(11) NOT NULL auto_increment,
sender varchar(30) default NULL,
receiver varchar(30) default NULL,
msg varchar(1024) default NULL,
senttime varchar(100) default NULL,
receivedtime varchar(100) default NULL,
operator varchar(100),
msgtype varchar(160) default NULL,
PRIMARY KEY (id)
);

CREATE TABLE ozekimessageout
(
id int(11) NOT NULL auto_increment,
sender varchar(30) default NULL,
receiver varchar(30) default NULL,
msg varchar(1024) default NULL,
senttime varchar(100) default NULL,
receivedtime varchar(100) default NULL,
status varchar(20) default NULL,
msgtype varchar(160) default NULL,
operator varchar(100),
PRIMARY KEY (id)
);

GRANT insert,update,select,delete on ozekisms.* to sqluser@localhost
IDENTIFIED BY 'abc123';
```

**Figure 2** - Creating tables in the structure required by Ozeki SMS Gateway

## How to create a Database User in Ozeki SMS Gateway

**Step 1:** Please open Ozeki 10 in a webbrowser
**Step 2:** Select SMS Gateway application from Ozeki 10's integrated desktop
**Step 3:** Create the Database User in SMS Gateway by following the instructions.

# How to use the code

Please use the PHP code you have downloaded and follow these steps:

**Step 1:** First unpack the downloaded zip file.

**Step 2:** Copy *sqlsmshandling.php*, *sqlsmshandling_inoutmessages.php* and *sqlsmshandling_functions.php* into the web server's main directory.

**Step 3:** Rewrite the SQL server's IP address, username and password in *sqlsmshandling_functions.php*.

**Step 4:** If Ozeki SMS Gateway is not running, please start it.

**Step 5:** Enter your PHP server's IP address in your web browser: *http://127.0.0.1/sqlsmshandling.php*
(You should change the IP of the PHP server or leave it on 127.0.0.1 if the PHP server and your webbrowser run on the same machine.)

**Step 6:** Please fill the HTML form and press 'Send'.

**Step 7:** If everything went fine, Ozeki SMS Gateway will show the message status in the Database User's log.

# Examine the PHP script

Here you can find out more details of the downloadable sample script.

The script must know the user credentials to your MySQL database. The main role of the script is to insert the new message in your outgoing message table. This table is called 'ozekimessageout' in the current example.

Ozeki SMS Gateway's Database User periodically checks the table for new records and if the status of the message is 'Send', it will try to send it. In case of success the Database User will change the message's status to 'Sent'.

This is the structure of the source code:

**sqlsmshandling.php:**

It contains message builder ASP elements. For example labels, text boxes and the 'Send' button. The 'Send' button calls the '*connectToDatabase()*', '*insertMessage(...)*' and '*closeConnection()*' functions from *sqlsmshandling_functions.php*. *sqlsmshandling.php* handles not just SMS messages, but exceptions as well. For example empty fields.

**sqlsmshandling_functions.php:**

This PHP file contains all the internal functions used by the 3 file.

-*connectToDatabase()*: Connects to a database of the MySQL server.
-*closeConnection()*: Closes the connection, which was created in *connectToDatabase()*.
-*insertMessage(recipient, messageType, messageText)*: Runs INSERT statement on the database. This statement inserts the message record into the outgoing message table.
-*showOutgoingMessagesInTable()*: Selects message from the outgoing messages SQL table and builds a HTML table from them in your webbrowser.
-*showIncomingMessagesInTable()*: Selects message from the incoming messages SQL table and builds a HTML table from them in your webbrowser.

**sqlsmshandling_inoutmessages.php:**

It builds HTML tables. It uses *connectToDatabase()*, *showOutgoingMessagesInTable()*, *showIncomingMessagesInTable()* and *closeConnection()* function for re-generate tables. These functions are available in *sqlsmshandling_functions.php*.

**The detailed description of the process in Figure 1 above:**

**<u>Step 1:</u> Create input form**

*sqlsmshandling.php* creates a form (**Figure 3**) to request SMS data from the user. The form consists of Labels and Textboxes, plus a 'Send' button. The internet user fills the recipient and message fields and clicks 'Send'.

**sqlsmshandling.php**

```
...
<form action="" method="post">
    <table border="0" align="center">
        <tr>
            <td colspan="2" align="center">
                <font style="font-weight: bold; font-size: 16px;">Compose message</font>
                <br /><br />
            </td>
        </tr>
        <tr>
            <td valign="top">Recipient: </td>
            <td>
                <textarea name="textAreaRecipient" cols="40" rows="2">...</textarea>
            </td>
        </tr>
        <tr>
            <td valign="top">Message text: </td>
            <td>
                <textarea name="textAreaMessage" cols="40" rows="10">...</textarea>
            </td>
        </tr>
        <tr>
            <td colspan="2" align="center">
                <input type="submit" value="Send">
            </td>
        </tr>
        <tr><td colspan='2' align='center'>
        ...
        </td></tr>
    </table>
</form>
...
```

**Figure 3** - Builds HTML form

**If** the internet user clicks 'Send' the following script runs:

```
...

<?php

    if (isset($_POST["textAreaRecipient"]) && $_POST["textAreaRecipient"] == "")
    {
        echo "Recipient field mustn't be empty!";
    }
    else if (isset($_POST["textAreaRecipient"]) && $_POST["textAreaRecipient"] != "")
    {
    try
    {
        connectToDatabase();
        if (insertMessage($_POST["textAreaRecipient"],"SMS:TEXT",$_POST["textAreaMessage"]))
        {
            echo "Insert was successful!";
        }
        closeConnection();
    }
    catch (Exception $exc)
    {
        echo "Error: " . $exc->getMessage();
    }
}
?>

...
```

**Figure 4** - It logs in and inserts the message to the database table by calling insertMessage(...) function

**<u>Step 2:</u> Processing data in the HTML textboxes**

If both textboxes are filled, the data will be processed and inserted in the SQL database's outgoing message table. The **insertMessage(...) function** (**Figure 5**) can achieve this.

Keep in mind that the message record is going to be inserted in the MySQL database by the script which has previously logged in (**Figure 4 above**).

**sqlsmshandling_functions.php**

```
...
function insertMessage ($recipient, $messageType, $messageText)
{
    $query = "insert into ozekimessageout (receiver,msgtype,msg,status) ";
    $query .= "values ('" . $recipient . "',
        '" . $messageType . "', '" . $messageText . "', 'send');";
    $result = mysql_query($query);
    if (!$result)
    {
        echo (mysql_error() . "<br>");
        return false;
    }

    return true;
}
...
```

**Figure 5** - insertMessage(...) function

## Step 3: Creating outgoing and incoming message tables

If you press F5 or click refresh in your browser, the outgoing and incoming tables will be updated by reading information from the MySQL database. As an example you can see showOutgoingMessagesInTable() below (**Figure 6**).

**sqlsmshandling_functions.php**

```
function showOutgoingMessagesInTable()
{
    $query = "select id,sender,receiver,senttime,receivedtime,operator,status,msgtype,
                msg from ozekimessageout;";
    $result = mysql_query($query);
    if (!$result)
    {
        echo (mysql_error() . "<br>");
        return false;
    }

    try
    {
        echo "<table border='1'>";
        echo "<tr><td>ID</td><td>Sender</td><td>Receiver</td>
        <td>Sent time</td><td>Received time</td><td>Operator</td>";
        echo "<td>Status</td><td>
        Message type</td><td>Message text</td></tr>";
        while ($row = mysql_fetch_assoc($result))
        {
            echo "<tr>";

            echo "<td>" . $row["id"] . "</td>";
            echo "<td>" . $row["sender"] . "</td>";
            echo "<td>" . $row["receiver"] . "</td>";
            echo "<td>" . $row["senttime"] . "</td>";
            echo "<td>" . $row["receivedtime"] . "</td>";
            echo "<td>" . $row["operator"] . "</td>";
            echo "<td>" . $row["status"] . "</td>";
            echo "<td>" . $row["msgtype"] . "</td>";
            echo "<td>" . $row["msg"] . "</td>";

            echo "</tr>";
        }
        echo "</table>";
        mysql_free_result($result);
    }
    catch (Exception $exc)
    {
        echo (mysql_error() . "<br>");
        return false;
    }

    return true;
}
...
```

**Figure 6** - showOutgoingMessagesInTable() function: It generates a HTML table from outgoing messages

# Frequently asked questions

**Question:** Can I send another type of message than 'SMS:TEXT' ?
**Answer:** Yes. For example, a Wap push message when calling the insertMessage(...) function (**Figure 7**).

```
insertMessage ($_POST["textAreaRecipient"], "SMS:WAPPUSH", $_POST["textAreaMessage"])
```

On the form the following should be written in the 'Message text' textbox:

```
<si>
<indication href="http://target_address" action="signal-high">
text of description
</indication>
</si>
```

**Figure 7** - SMS:TEXT changed to SMS:WAPPUSH
The 'action' parameter of the 'indication' tag can be one of the following:
signal-high, signal-medium, signal-low, signal-none, signal-delete.

**Question:** Can the PHP enabled HTTP server run on a different computer from the MySQL server and Ozeki SMS Gateway?
**Answer:** Yes, it can. In PHP script please set the IP address and actual login credentials to your MySQL database.

**Question:** Can I show the recipient the phone number of the sender?
**Answer:** Yes, you can. Please create textbox for the sender's phone number on modify the insertMessage(...) function (**Figure 8**). Keep in mind that this only works if you have an IP SMS service provider connection.

```
function insertMessage($sender, $recipient, $messageType, $messageText)
{
...
$query = "insert into ozekimessageout (sender,receiver,msgtype,msg,status) ";
$query .= "values ('".$sender."','".$recipient."',".
"'".$messageType."','".$messageText."','send');";
$result = mysql_query($query);
...
}
```

**Figure 8** - Modified insertMessage(...) function

# Send SMS from MySQL with PHP

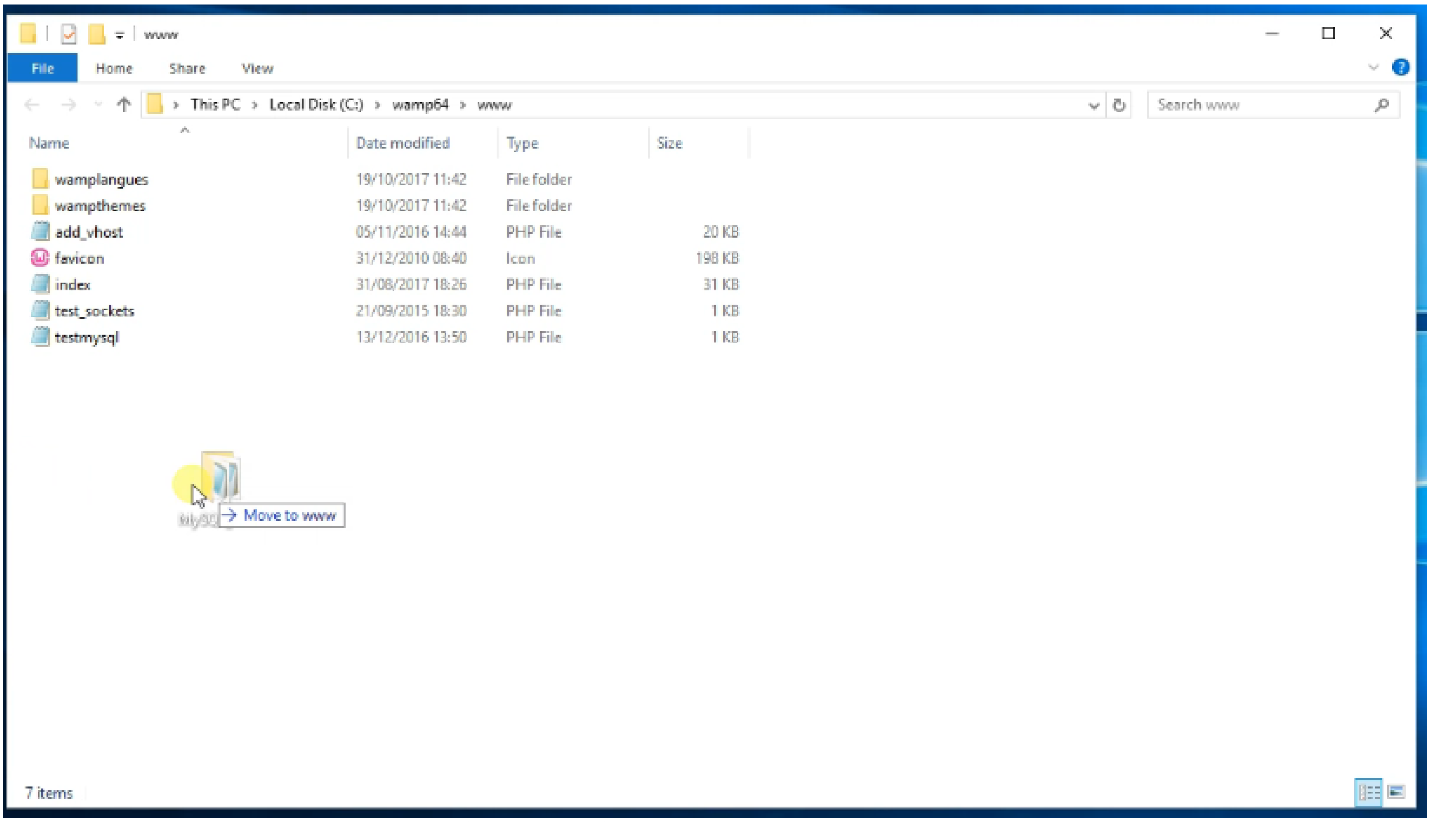
***You will see how a random person can use their internet browser to send SMS. The trick is that this PHP script can log into your MySQL database server and add the outgoing SMS to a table. Ozeki SMS Gateway can connect to your MySQL database as well and read the table for the SMS to send.***

**Video content**

1. Download PHP File
2. Copy PHP File to webserver
3. Modify PHP File
4. Open webpage
5. Send test message

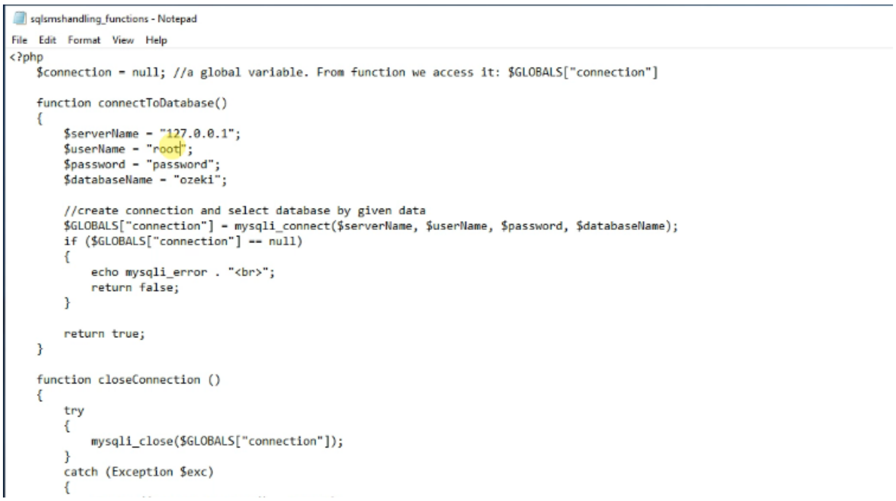
---

If you scroll down you can find screenshots that describe how to implement the **MySQL database + PHP** solution. Keep in mind that this PHP script can SELECT messages from tables as well, so you can visualize outgoing and incoming messages simultaneously in the same webbrowser.



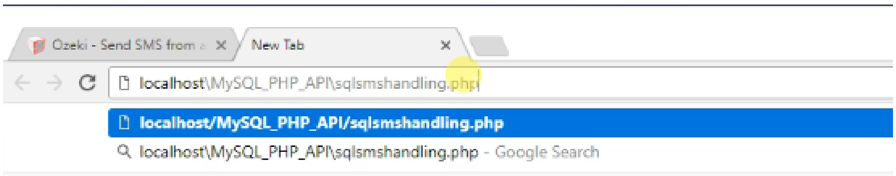
| Name | Date modified | Type | Size |
|---|---|---|---|
| wamplangues | 19/10/2017 11:42 | File folder | |
| wampthemes | 19/10/2017 11:42 | File folder | |
| add_vhost | 05/11/2016 14:44 | PHP File | 20 KB |
| favicon | 31/12/2010 08:40 | Icon | 198 KB |
| index | 31/08/2017 18:26 | PHP File | 31 KB |
| test_sockets | 21/09/2015 18:30 | PHP File | 1 KB |
| testmysql | 13/12/2016 13:50 | PHP File | 1 KB |

Figure 2 - Copy PHP files to the webserver's WWW directory


sqlsmshandling_functions - Notepad
File Edit Format View Help

```
<?php
    $connection = null; //a global variable. From function we access it: $GLOBALS["connection"]

    function connectToDatabase()
    {
        $serverName = "127.0.0.1";
        $userName = "root";
        $password = "password";
        $databaseName = "ozeki";

        //create connection and select database by given data
        $GLOBALS["connection"] = mysqli_connect($serverName, $userName, $password, $databaseName);
        if ($GLOBALS["connection"] == null)
        {
            echo mysqli_error . "<br>";
            return false;
        }

        return true;
    }

    function closeConnection ()
    {
        try
        {
            mysqli_close($GLOBALS["connection"]);
        }
        catch (Exception $exc)
        {
```

Figure 3 - Modify sqlsmshandling_functions.php

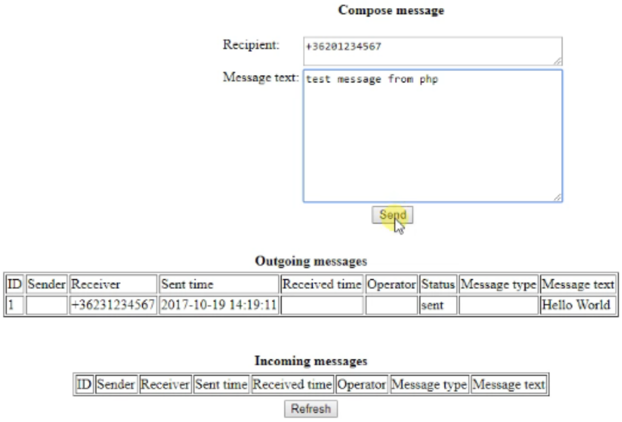


localhost\MySQL_PHP_API\sqlsmshandling.php

localhost/MySQL_PHP_API/sqlsmshandling.php

localhost\MySQL_PHP_API\sqlsmshandling.php - Google Search

Figure 4 - Open the created webpage


**Compose message**

Recipient: +36201234567

Message text: test message from php

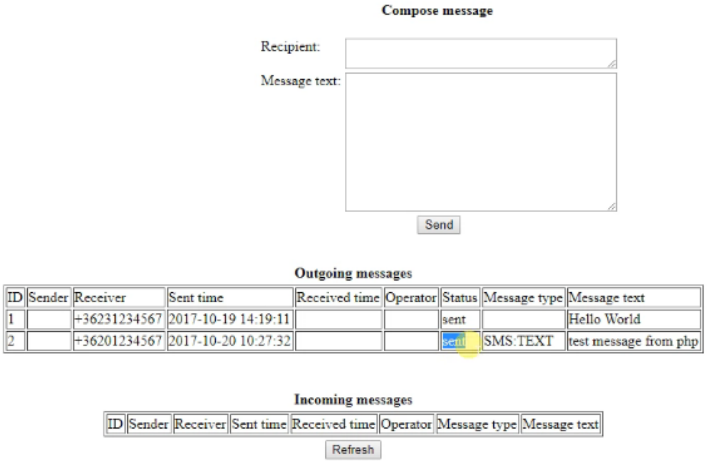
**Outgoing messages**

| ID | Sender | Receiver | Sent time | Received time | Operator | Status | Message type | Message text |
|---|---|---|---|---|---|---|---|---|
| 1 | | +36231234567 | 2017-10-19 14:19:11 | | | sent | | Hello World |

**Incoming messages**

| ID | Sender | Receiver | Sent time | Received time | Operator | Message type | Message text |
|---|---|---|---|---|---|---|---|

Figure 5 - Send test message


**Compose message**

Recipient:

Message text:

**Outgoing messages**

| ID | Sender | Receiver | Sent time | Received time | Operator | Status | Message type | Message text |
|---|---|---|---|---|---|---|---|---|
| 1 | | +36231234567 | 2017-10-19 14:19:11 | | | sent | | Hello World |
| 2 | | +36201234567 | 2017-10-20 10:27:32 | | | sent | SMS:TEXT | test message from php |

**Incoming messages**

| ID | Sender | Receiver | Sent time | Received time | Operator | Message type | Message text |
|---|---|---|---|---|---|---|---|

Figure 6 - Message has been sent

# Use your website to send SMS

**You can add SMS functionality to your website. This method presents how to use a PHP script to communicate with Ozeki SMS Gateway's HTTP API. First you need to enable the PHP scripting support on your webserver.**

Download: sendsms.zip

The presented example shows how message sending can work from your website by clicking 'Send'. The idea uses a HTML form. The user can fill this HTML form and click 'Send'. The PHP script processes the two variables in the textboxes called 'Recipient' and 'Message'. It also sends the generated HTTP request URL to the HTTP API of Ozeki SMS Gateway (**Figure 1**). Your webserver and Ozeki SMS Gateway can be on two different machines or on the same machine as well.

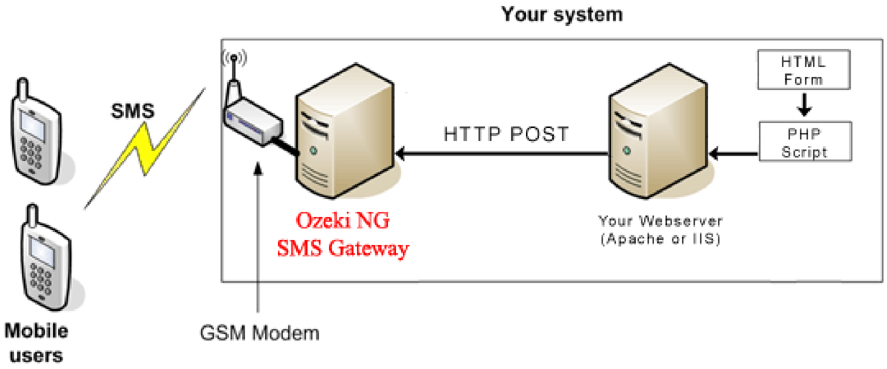


**Figure 1** - How an SMS can be sent from your website

You should install Ozeki SMS Gateway in the first place and check if you can manually send SMS messages from it. Then please download the HTML+PHP script to create and use the SMS sending form.

Ozeki SMS Gateway can be obtained by opening the download page:
**Download Ozeki SMS Gateway!**

## Create the HTML Form for sms sending

Please save sendsms.html to get started with this solution. You should save it in the 'WWW' directory of your webserver. What does it contain? It cointains the HTML form with a recipient and message textbox, plus a 'Send' button (**Figure 2**).

```
<html>
 <body>
   <h1>My SMS form</h1>
   <form method=post action='<b>sendsms.php</b>'>
   <table border=0>
   <tr>
     <td>Recipient</td>
     <td><input type='text' name='<b>recipient</b>'></td>
   </tr>
   <tr>
     <td>Message</td>
     <td><textarea rows=4 cols=40 name='<b>message</b>'></textarea></td>
   </tr>
   <tr>
     <td> </td>
     <td><input type=submit name=submit value=Send></td>
   </tr>
   </table>
   </form>
 </body>
</html>
```

**Figure 2** - The source of *\WWW\sendsms.html

If you have successfully saved this file in your webserver's directory, you can open it in any webbrowser (**Figure 3**). You can add any telephone number to the recipient field (e.g. +4407776134588) and write your message text in the next textbox.

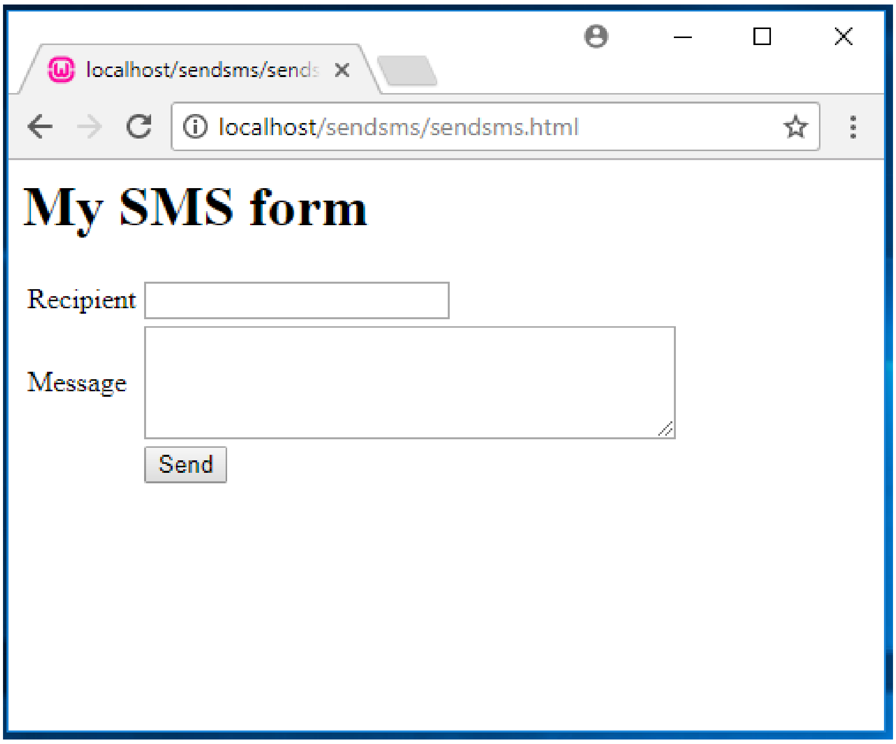


**Figure 3** - How the SMS form should look like

The form tag's action attribute points to sendsms.php. Which means that the data entered on the form will be forwarded to sendsms.php. The data will be sent to the HTTP API if 'Send' is pressed.

## How to prepare your PHP SMS script

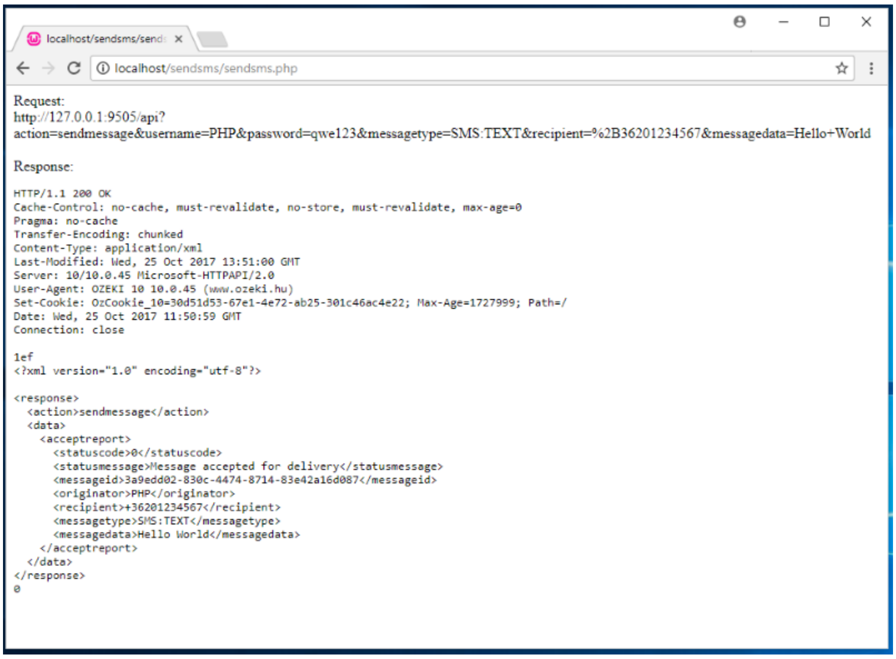
'sendsms.php' is the target for the recipient and message fields. This PHP script can build an URL for Ozeki SMS Gateway's HTTP API. It will invoke the URL if 'Send' is pressed. Do no forget to **configure $ozeki_user, $ozeki_password and $ozeki_url variables**. In case Ozeki SMS Gateway and your webserver runs on the same machine, you can leave $ozeki_url untouched. Otherwise you should change the IP to the IP address of Ozeki SMS Gateway's machine. The newest version of Ozeki SMS Gateway uses port 9505, while 9501 belongs to the older one.

```
?php

########################################################
# Login information for the SMS Gateway
########################################################

$ozeki_user = "admin";
$ozeki_password = "abc123";
$ozeki_url = "http://127.0.0.1:9505/api?";

########################################################
# Functions used to send the SMS message
########################################################
function httpRequest($url){
    $pattern = "/http...([0-9a-zA-Z-.]*).([0-9]*).(.*)/";
    preg_match($pattern,$url,$args);
    $in = "";
    $fp = fsockopen("$args[1]", $args[2], $errno, $errstr, 30);
    if (!$fp) {
       return("$errstr ($errno)");
    } else {
        $out = "GET /$args[3] HTTP/1.1\r\n";
        $out .= "Host: $args[1]:$args[2]\r\n";
        $out .= "User-agent: Ozeki PHP client\r\n";
        $out .= "Accept: */*\r\n";
        $out .= "Connection: Close\r\n\r\n";

        fwrite($fp, $out);
```

```php
        while (!feof($fp)) {
           $in.=fgets($fp, 128);
        }
    }
    fclose($fp);
    return($in);
}



function ozekiSend($phone, $msg, $debug=false){
      global $ozeki_user,$ozeki_password,$ozeki_url;

      $url= 'action=sendmessage';
      $url.= '&username='.$ozeki_user;
      $url.= '&password='.$ozeki_password;
      $url.= '&messagetype=SMS:TEXT';
      $url.= '&recipient='.urlencode($phone);
      $url.= '&messagedata='.urlencode($msg);

      $urltouse =  $ozeki_url.$url;
      if ($debug) { echo "Request: <br>$urltouse<br><br>"; }

      //Open the URL to send the message
      $response = httpRequest($urltouse);
      if ($debug) {
           echo "Response: <br><pre>".
           str_replace(array("<",">"),array("&lt;","&gt;"),$response).
           "</pre><br>"; }

      return($response);
}

#########################################################
# GET data from sendsms.html
#########################################################

$phonenum = $_POST['recipient'];
$message = $_POST['message'];
$debug = true;

ozekiSend($phonenum,$message,$debug);

?>
```

**Figure 4** - The source code of the *\WWW\sendsms.php script

After both sendsms.html and sendsms.php are copied and modified at *\WWW\, an SMS message can be submitted by the gateway. After it gets accepted for delivery, the SMS Gateway will return the message reference number in the response (**Figure 5**). Messages are tracked in Ozeki SMS Gateway by their reference number.



```
Request:
http://127.0.0.1:9505/api?
action=sendmessage&username=PHP&password=qwe123&messagetype=SMS:TEXT&recipient=%2B36201234567&messagedata=Hello+World

Response:

HTTP/1.1 200 OK
Cache-Control: no-cache, must-revalidate, no-store, must-revalidate, max-age=0
Pragma: no-cache
Transfer-Encoding: chunked
Content-Type: application/xml
Last-Modified: Wed, 25 Oct 2017 13:51:00 GMT
Server: 10/10.0.45 Microsoft-HTTPAPI/2.0
User-Agent: OZEKI 10 10.0.45 (www.ozeki.hu)
Set-Cookie: OzCookie_10=30d51d53-67e1-4e72-ab25-301c46ac4e22; Max-Age=1727999; Path=/
Date: Wed, 25 Oct 2017 11:50:59 GMT
Connection: close

1ef
<?xml version="1.0" encoding="utf-8"?>

<response>
  <action>sendmessage</action>
  <data>
    <acceptreport>
      <statuscode>0</statuscode>
      <statusmessage>Message accepted for delivery</statusmessage>
      <messageid>3a9edd02-830c-4474-8714-83e42a16d087</messageid>
      <originator>PHP</originator>
      <recipient>+36201234567</recipient>
      <messagetype>SMS:TEXT</messagetype>
      <messagedata>Hello World</messagedata>
    </acceptreport>
  </data>
</response>
0
```

**Figure 5** - Response XML from Ozeki SMS Gateway

# Send SMS from HTTP API with PHP

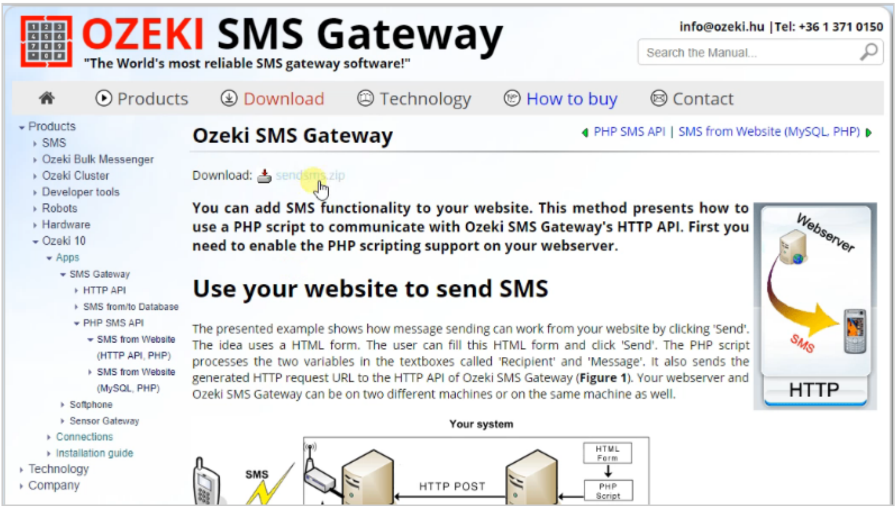
**You will see how a random person can use their internet browser to send SMS. The trick is that this PHP script can make 'sendmessage' HTTP requests to Ozeki SMS Gateway's HTTP API. The 'sendmessage' request can send any SMS from SMS Gateway.**

### Video content

1. Download PHP File
2. Copy PHP File to webserver
3. Create HTTP Server user
4. Modify PHP File
5. Open webpage
6. Send test message

---

If you scroll down you can find screenshots that describe how to implement the **HTTP API + PHP** solution.



Ozeki SMS Gateway

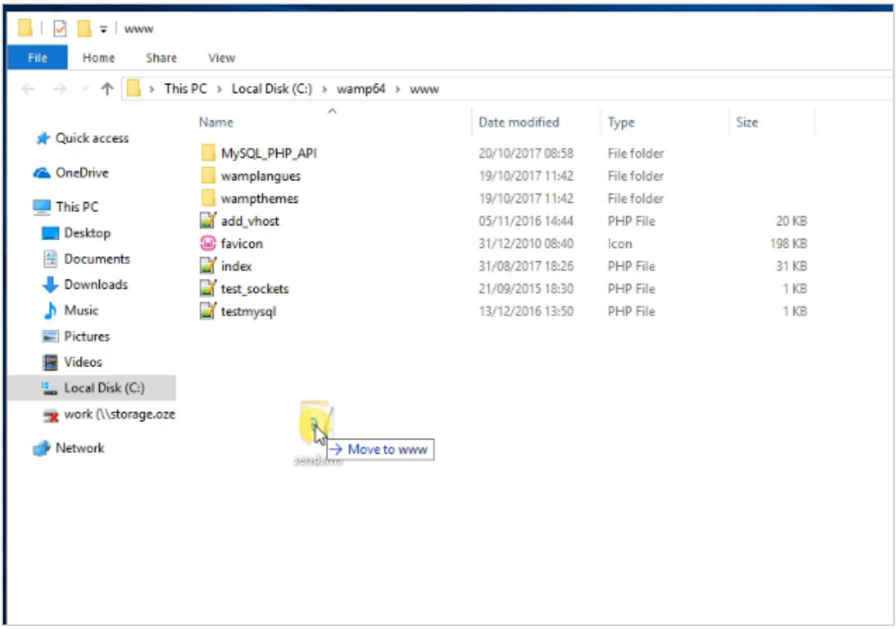
Download: sendsms.zip

**You can add SMS functionality to your website. This method presents how to use a PHP script to communicate with Ozeki SMS Gateway's HTTP API. First you need to enable the PHP scripting support on your webserver.**

Use your website to send SMS

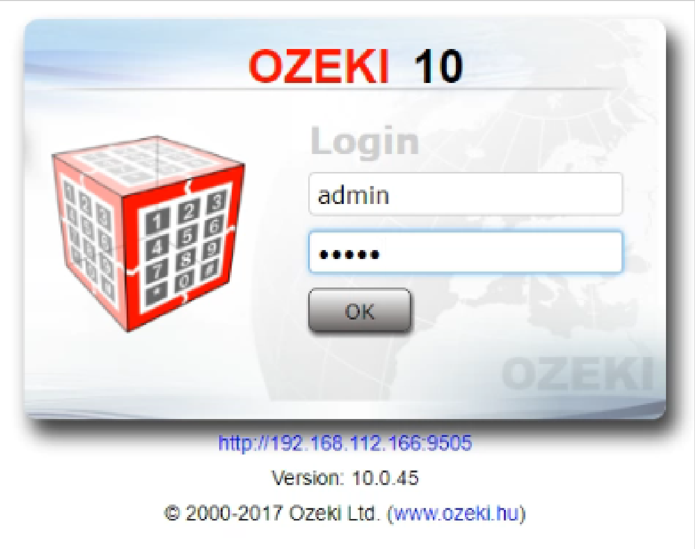
The presented example shows how message sending can work from your website by clicking 'Send'. The idea uses a HTML form. The user can fill this HTML form and click 'Send'. The PHP script processes the two variables in the textboxes called 'Recipient' and 'Message'. It also sends the generated HTTP request URL to the HTTP API of Ozeki SMS Gateway (**Figure 1**). Your webserver and Ozeki SMS Gateway can be on two different machines or on the same machine as well.

**Figure 1 -** Download PHP/HTML files



| Name | Date modified | Type | Size |
|---|---|---|---|
| MySQL_PHP_API | 20/10/2017 08:58 | File folder | |
| wamplangues | 19/10/2017 11:42 | File folder | |
| wampthemes | 19/10/2017 11:42 | File folder | |
| add_vhost | 05/11/2016 14:44 | PHP File | 20 KB |
| favicon | 31/12/2010 08:40 | Icon | 198 KB |
| index | 31/08/2017 18:26 | PHP File | 31 KB |
| test_sockets | 21/09/2015 18:30 | PHP File | 1 KB |
| testmysql | 13/12/2016 13:50 | PHP File | 1 KB |

**Figure 2 -** Copy the downloaded PHP and HTML files to the webserver's WWW directory



http://192.168.112.166:9505

Version: 10.0.45

© 2000-2017 Ozeki Ltd. (www.ozeki.hu)

**Figure 3** - Login to Ozeki 10

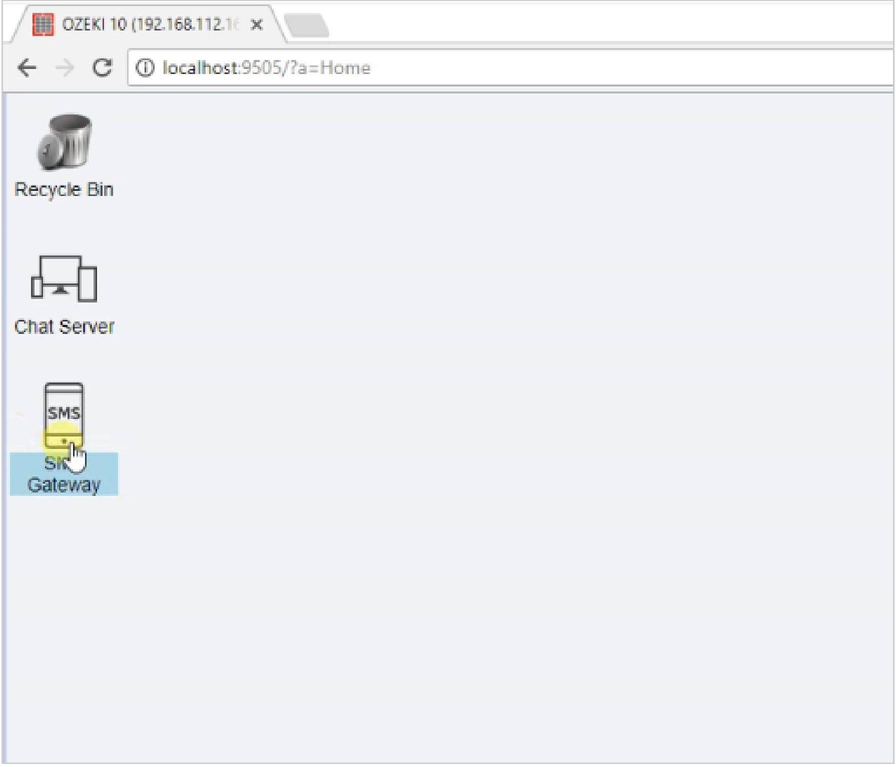
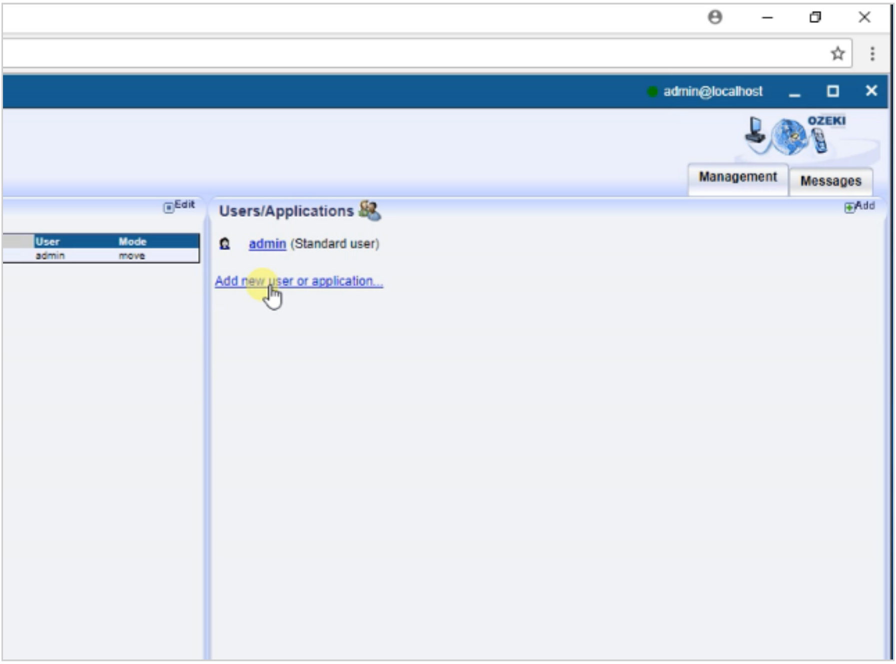


**Figure 4** - Open the SMS Gateway applicaton

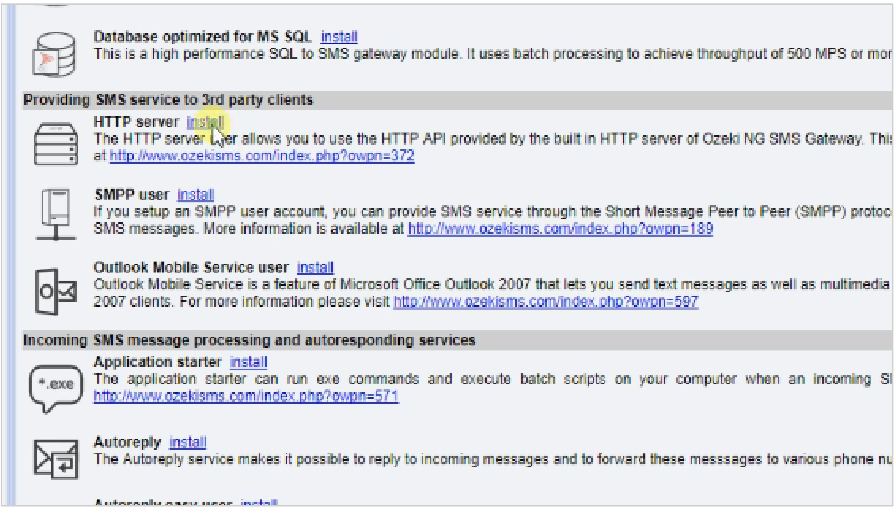


Users/Applications

admin (Standard user)

Add new user or application...

**Figure 5 -** Click on 'Add new user or application...'


Database optimized for MS SQL install
This is a high performance SQL to SMS gateway module. It uses batch processing to achieve throughput of 500 MPS or mor

Providing SMS service to 3rd party clients

HTTP server install
The HTTP server user allows you to use the HTTP API provided by the built in HTTP server of Ozeki NG SMS Gateway. Thi
at http://www.ozekisms.com/index.php?owpn=372

SMPP user install
If you setup an SMPP user account, you can provide SMS service through the Short Message Peer to Peer (SMPP) protoc
SMS messages. More information is available at http://www.ozekisms.com/index.php?owpn=189

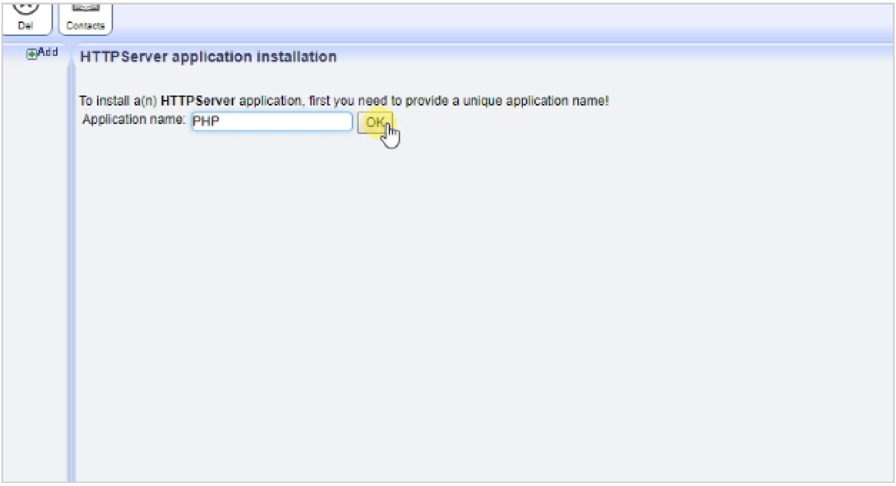
Outlook Mobile Service user install
Outlook Mobile Service is a feature of Microsoft Office Outlook 2007 that lets you send text messages as well as multimedia
2007 clients. For more information please visit http://www.ozekisms.com/index.php?owpn=597

Incoming SMS message processing and autoresponding services

Application starter install
The application starter can run exe commands and execute batch scripts on your computer when an incoming S
http://www.ozekisms.com/index.php?owpn=571

Autoreply install
The Autoreply service makes it possible to reply to incoming messages and to forward these messsages to various phone nu

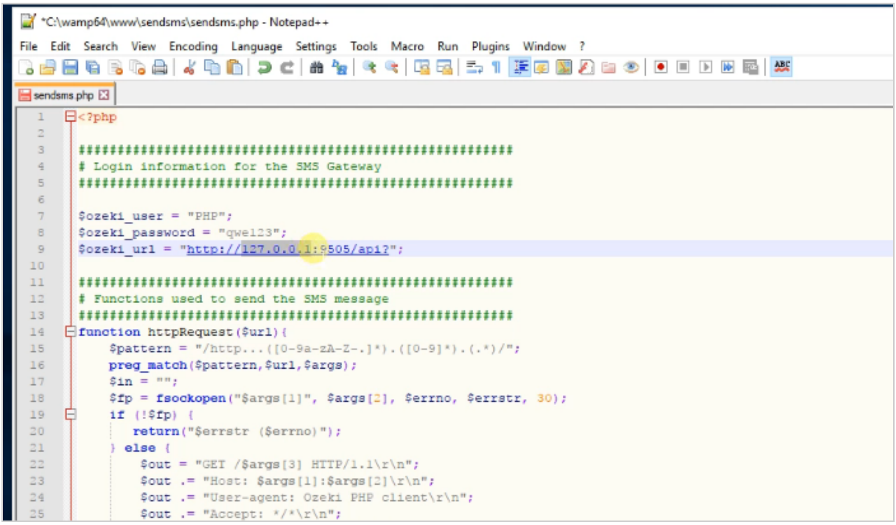
**Figure 6 -** Install 'HTTP server' user



HTTPServer application installation

To install a(n) HTTPServer application, first you need to provide a unique application name!
Application name: PHP OK

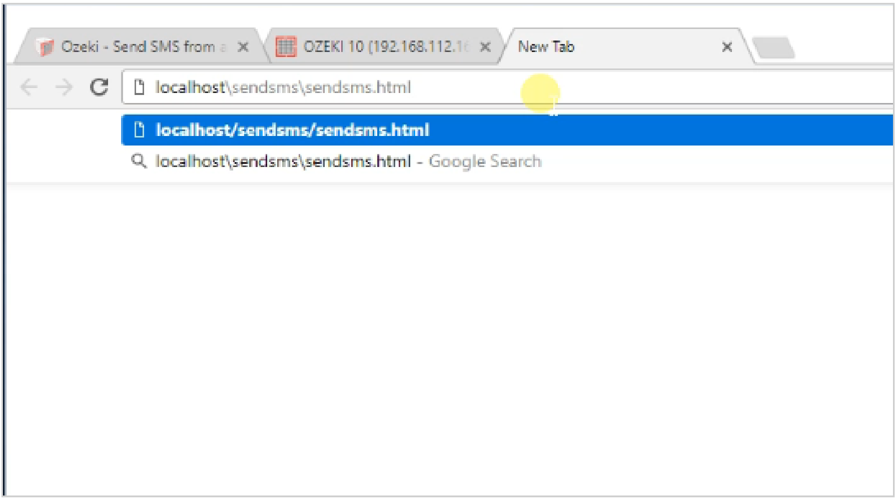
**Figure 7 -** Create the name of the User



*C:\wamp64\www\sendsms\sendsms.php - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
sendsms.php

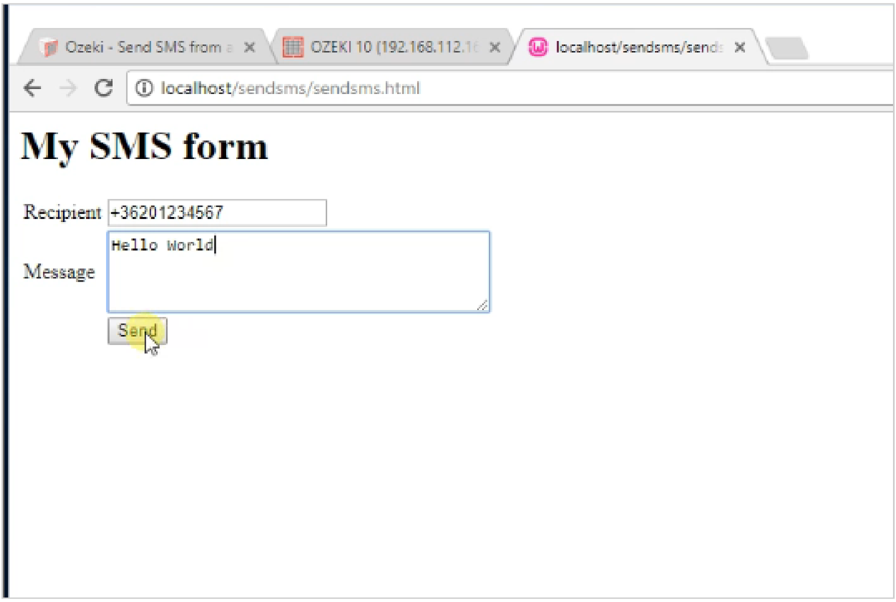
```php
<?php

########################################################
# Login information for the SMS Gateway
########################################################

$ozeki_user = "PHP";
$ozeki_password = "qwe123";
$ozeki_url = "http://127.0.0.1:9505/api?";

########################################################
# Functions used to send the SMS message
########################################################
function httpRequest($url){
    $pattern = "/http...([0-9a-zA-Z-.]*).([0-9]*).(.*)/";
    preg_match($pattern,$url,$args);
    $in = "";
    $fp = fsockopen("$args[1]", $args[2], $errno, $errstr, 30);
    if (!$fp) {
       return("$errstr ($errno)");
    } else {
        $out = "GET /$args[3] HTTP/1.1\r\n";
        $out .= "Host: $args[1]:$args[2]\r\n";
        $out .= "User-agent: Ozeki PHP client\r\n";
        $out .= "Accept: */*\r\n";
```

**Figure 8** - Modify sendsms.php



Ozeki - Send SMS from
OZEKI 10 (192.168.112.1
New Tab
localhost\sendsms\sendsms.html
localhost/sendsms/sendsms.html
localhost\sendsms\sendsms.html - Google Search

**Figure 9** - Open the created webpage

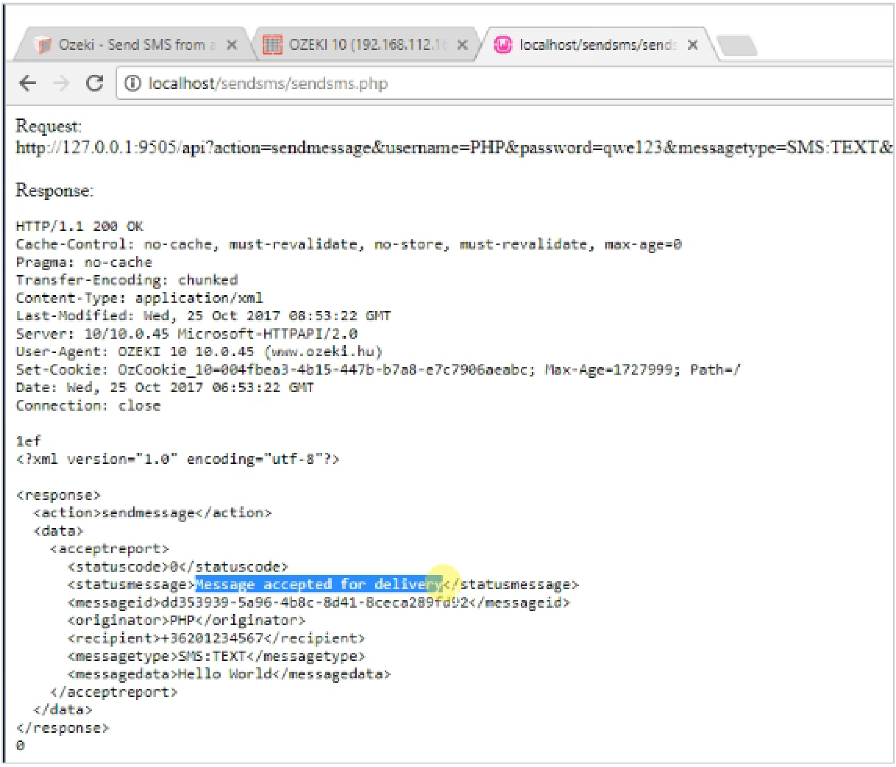


My SMS form

Recipient +36201234567

Message Hello World

**Figure 10** - Send test message



Request:
http://127.0.0.1:9505/api?action=sendmessage&username=PHP&password=qwe123&messagetype=SMS:TEXT&

Response:

```
HTTP/1.1 200 OK
Cache-Control: no-cache, must-revalidate, no-store, must-revalidate, max-age=0
Pragma: no-cache
Transfer-Encoding: chunked
Content-Type: application/xml
Last-Modified: Wed, 25 Oct 2017 08:53:22 GMT
Server: 10/10.0.45 Microsoft-HTTPAPI/2.0
User-Agent: OZEKI 10 10.0.45 (www.ozeki.hu)
Set-Cookie: OzCookie_10=004fbea3-4b15-447b-b7a8-e7c7906aeabc; Max-Age=1727999; Path=/
Date: Wed, 25 Oct 2017 06:53:22 GMT
Connection: close

1ef
<?xml version="1.0" encoding="utf-8"?>

<response>
  <action>sendmessage</action>
  <data>
    <acceptreport>
      <statuscode>0</statuscode>
      <statusmessage>Message accepted for delivery</statusmessage>
      <messageid>dd353939-5a96-4b8c-8d41-8ceca289fd92</messageid>
      <originator>PHP</originator>
      <recipient>+36201234567</recipient>
      <messagetype>SMS:TEXT</messagetype>
      <messagedata>Hello World</messagedata>
    </acceptreport>
  </data>
</response>
0
```

**Figure 11** - Message has been sent

# Use your C# application for SMS messaging

**Your C# application can connect to a Standard User or any other user created in Ozeki SMS Gateway. You can easily prepare your C# codes to send or receive SMS messages through SMS Gateway. Below you can download source codes for two C#.NET SMS demo projects which demonstrates how easy it is to connect C# with Ozeki SMS Gateway.**

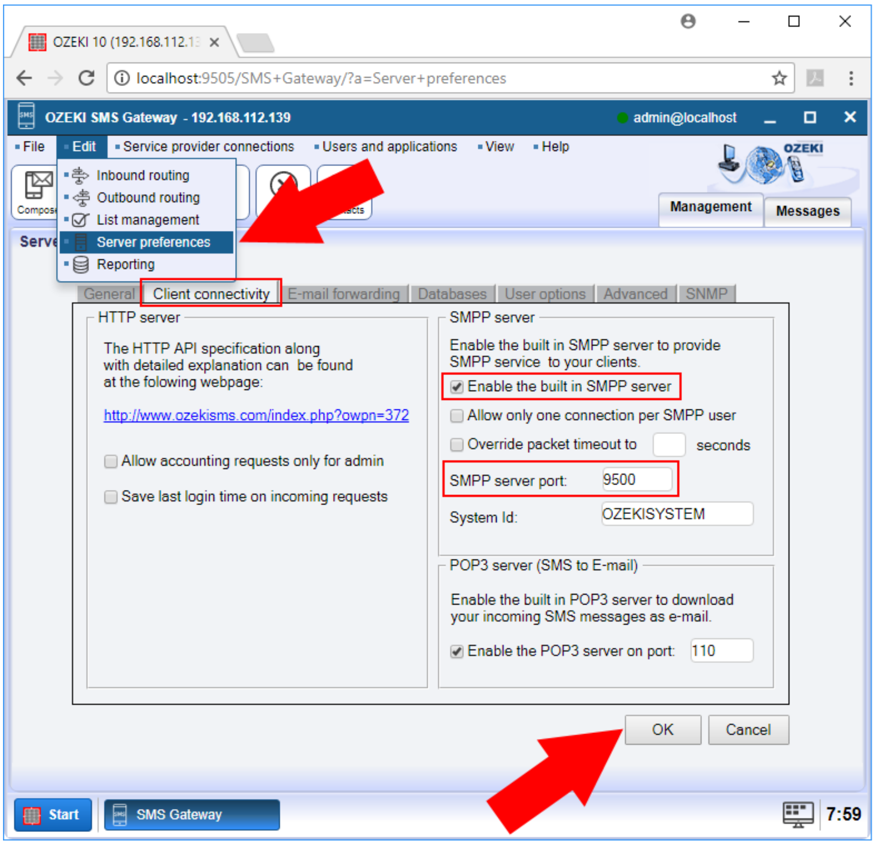
This demo projects can only work if Ozeki SMS Gateway is connected to the GSM network through a GSM modem or an IP SMS connection, such as SMPP, CIMD2 or UCP/EMI.

**Unzip** **CsNETDll_v3.zip** and **open** the **...\CsNETDll_v3\SMSDemo** directory to **run SMSDemo.sln** in your C# IDE.

## Introduction

The C# SMS API is capable to connect locally or remotely over LAN or internet to Ozeki SMS Gateway. It is a classic example how to use the .NET TCP/IP networking, so you can make two applications work together. Thanks to the advantages of the IP connectivity, it creates a client-server model, where the C# API is the client while Ozeki SMS Gateway's built in SMPP server is the server.

Please **activate the SMPP server from 'Edit/Server preferences'** and by clicking on the **'Client connectivity' tabpage**. Now you can **'Enable the built in SMPP server'** (**Figure 1**). *The default port number is 9500.* **In the examples below, you will be able to connect to the port number you have provided.**



HTTP server

The HTTP API specification along with detailed explanation can be found at the folowing webpage:

http://www.ozekisms.com/index.php?owpn=372

□ Allow accounting requests only for admin

□ Save last login time on incoming requests

SMPP server

Enable the built in SMPP server to provide SMPP service to your clients.

☑ Enable the built in SMPP server

□ Allow only one connection per SMPP user

□ Override packet timeout to seconds

SMPP server port: 9500

System Id: OZEKISYSTEM

POP3 server (SMS to E-mail)

Enable the built in POP3 server to download your incoming SMS messages as e-mail.

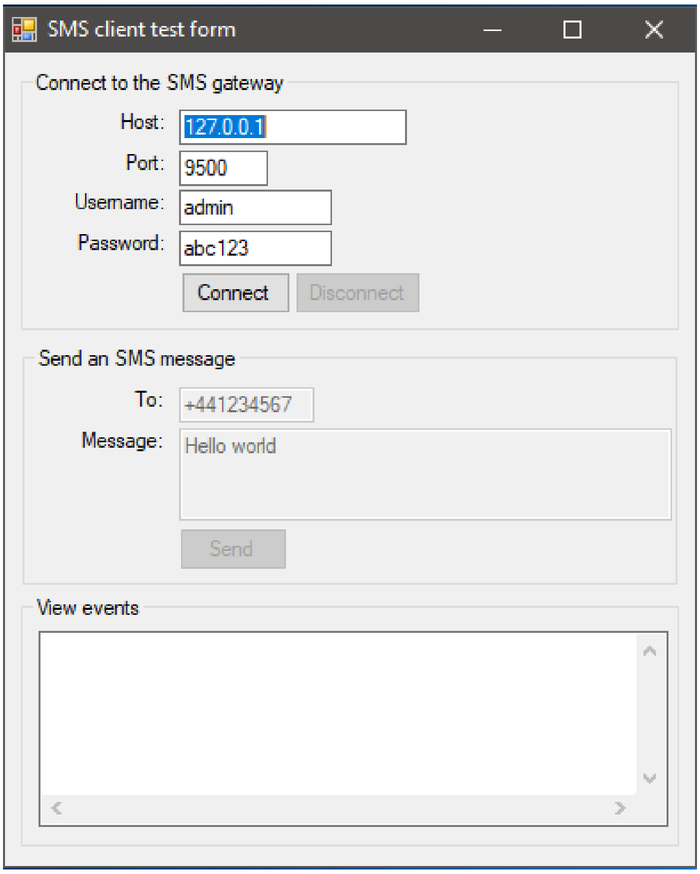
☑ Enable the POP3 server on port: 110

**Figure 1** - Enable the built in SMPP server and click 'OK'

## Examples

**The API comes with 2 examples, which you can download above** in a single zip.

The **first example project is 'SMSDemo'**. Through a working GUI you can connect to a Standard User of Ozeki SMS Gateway for SMS messaging (**Figure 2**).



Host: 127.0.0.1
Port: 9500
Username: admin
Password: abc123
To: +441234567
Message: Hello world

**Figure 2** - GUI of 'SMSDemo' example project

The **second example project is called 'SMSDemoConsole'**, with it you can send or receive SMS messages through a simple console, which shows the deliver status as well.

You can control 'SMSDemoConsole' by modifying the programcode. The parameters below help you to connect to any user in Ozeki SMS Gateway and send an SMS message using the 'sendMessage' method (**Figure 3**). You can send multiple SMS messages as well by multiplying the 'sendMessage' method.

```
mySMSClient.Username = "admin";
mySMSClient.Password = "abc123";
mySMSClient.Host = "127.0.0.1";
mySMSClient.Port = 9500;

mySMSClient.Connected = true;

mySMSClient.sendMessage("+44987654", "TEST", "vp=" + DateTime.Now + "&ttt=werwerwe rewwe34232 1");

Console.ReadKey(); //Press any key
mySMSClient.Connected = false;
```

**Figure 3** - The most important code fragment in the 'SMSDemoConsole' example. It is contained in 'Program.cs'.

Press any key to close the console.

'sendMessage' method's Validity Period parameter:
If you look back to Figure 3, above, you can see the "vp=" parameter, which represents the validity period of each message. *The validity period is the time limit of the message to be stored by the service provider in case the recipient's phone is turned off.* For example you can use the following parameter value:

```
"vp=" + 2011.07.26. + 10%3A07%3A58
```

**Figure 4** - Validity period parameter

**2018.07.26.+10%3A07%3A58** is basicly **2018.07.26. 10:07:58** in an URL encoded form.

**Tip:** Download, install and use Microsoft Visual Studio to test these two examples.

# Solutions to solve with the C# API

C# SMS API can send different type of SMS message types. For example WAP Push, Flash SMS, Ringtones, VCard, VCalendar, etc. The C# API can also subscribe to notification events, so it can help you track SMS messages. This way you can know when the message was delivered to the network if it was delivered at all and you can also know when it was received by the recipient's phone.

You can also receive SMS messages to the C# API. This helps you create great applications. For example SMS voting, stock market data querying or SMS quiz games. **It is strongly recommended to read the full documentation of the C# API on the following page:** C# SMS API Reference

Here is an alternative solution through SQL:

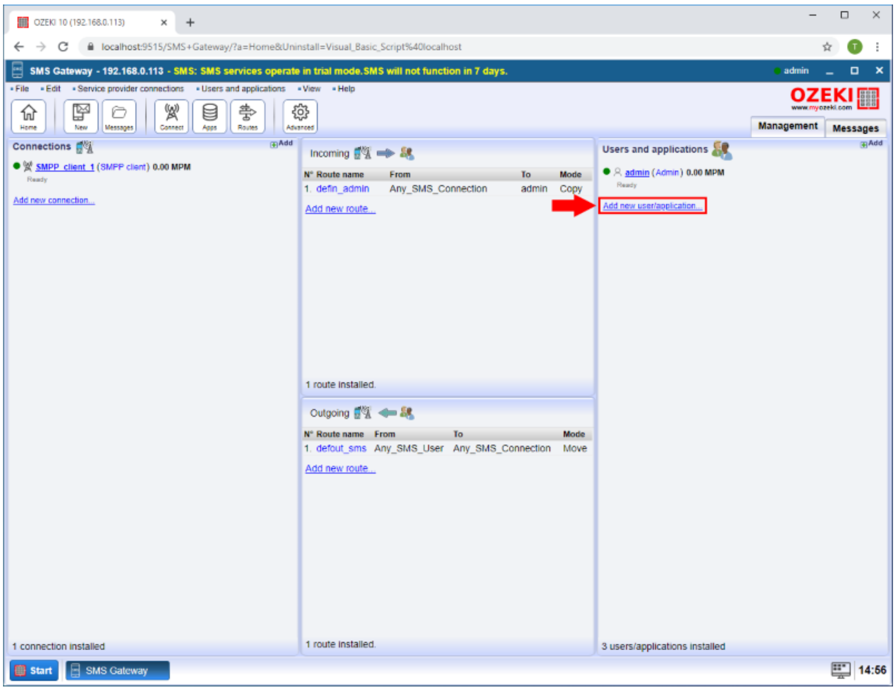
Connecting your C# API to a Database User through your SQL database.

# C#.Net SMS Script

***In the Ozeki SMS Gateway you can write a script, that allows you to run C# .NET code when an SMS message arrives. This script can implement any logic to do various tasks with the incoming messages, and optionally send response SMS messages.***

## Step 1 - Add new user/application...

You can simply install the C# script User on the Management console by clicking Add new user/application... in the Users/Applications panel (Figure 1).



Incoming

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defin_admin | Any_SMS_Connection | admin | Copy |

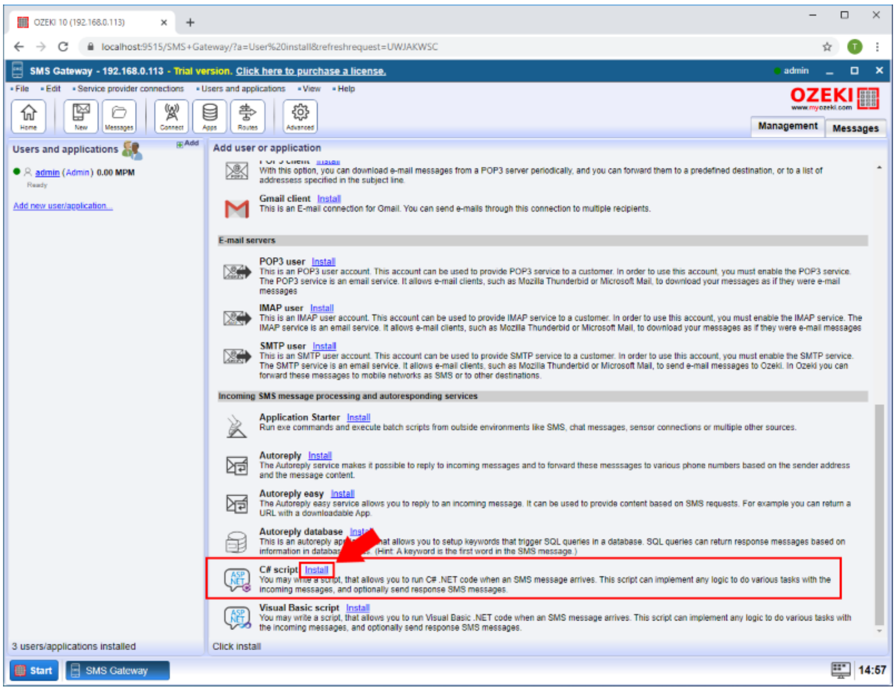
Outgoing

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move |

**Figure 1** - Add new user/application...

## Step 2 - Add C# script

An interface will open consisting of two panels. The left side panel contains the already installed users and applications. The right side panel contains the users and applications you can install with a brief description next to them. Search the C# script User and click the blue 'install' button next to it (Figure 2).



OZEKI 10 (192.168.0.113)
localhost:9515/SMS+Gateway/?a=User%20install&refreshrequest=UWJAKWSC
SMS Gateway - 192.168.0.113 - Trial version. Click here to purchase a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Users and applications
Add
admin (Admin) 0.00 MPM
Ready
Add new user/application...
3 users/applications installed
Start
SMS Gateway
14:57

Add user or application

POP3 client Install
With this option, you can download e-mail messages from a POP3 server periodically, and you can forward them to a predefined destination, or to a list of addresses specified in the subject line.

Gmail client Install
This is an E-mail connection for Gmail. You can send e-mails through this connection to multiple recipients.

E-mail servers

POP3 user Install
This is an POP3 user account. This account can be used to provide POP3 service to a customer. In order to use this account, you must enable the POP3 service. The POP3 service is an email service. It allows e-mail clients, such as Mozilla Thunderbid or Microsoft Mail, to download your messages as if they were e-mail messages

IMAP user Install
This is an IMAP user account. This account can be used to provide IMAP service to a customer. In order to use this account, you must enable the IMAP service. The IMAP service is an email service. It allows e-mail clients, such as Mozilla Thunderbid or Microsoft Mail, to download your messages as if they were e-mail messages

SMTP user Install
This is an SMTP user account. This account can be used to provide SMTP service to a customer. In order to use this account, you must enable the SMTP service. The SMTP service is an email service. It allows e-mail clients, such as Mozilla Thunderbid or Microsoft Mail, to send e-mail messages to Ozeki. In Ozeki you can forward these messages to mobile networks as SMS or to other destinations.

Incoming SMS message processing and autoresponding services

Application Starter Install
Run exe commands and execute batch scripts from outside environments like SMS, chat messages, sensor connections or multiple other sources.

Autoreply Install
The Autoreply service makes it possible to reply to incoming messages and to forward these messsages to various phone numbers based on the sender address and the message content.

Autoreply easy Install
The Autoreply easy service allows you to reply to an incoming message. It can be used to provide content based on SMS requests. For example you can return a URL with a downloadable App.

Autoreply database Install
This is an autoreply ap hat allows you to setup keywords that trigger SQL queries in a database. SQL queries can return response messages based on information in databas . (Hint: A keyword is the first word in the SMS message.)

C# script Install
You may write a script, that allows you to run C# .NET code when an SMS message arrives. This script can implement any logic to do various tasks with the incoming messages, and optionally send response SMS messages.

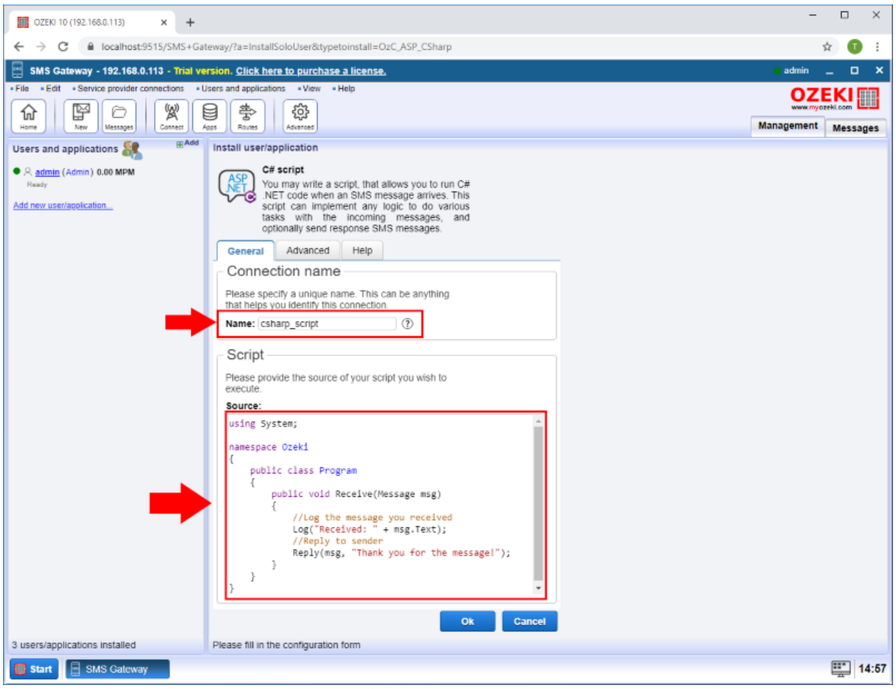
Visual Basic script Install
You may write a script, that allows you to run Visual Basic .NET code when an SMS message arrives. This script can implement any logic to do various tasks with the incoming messages, and optionally send response SMS messages.

Click install

**Figure 2** - Add C# script

## Step 3 - Provide the script

The 'Configuration' panel has a 'General' tab which contains the basic settings. First please provide a unique Name. In the Script section of tabpage contains the source of the script you wish to execute (Figure 3).



OZEKI 10 (192.168.0.113)
localhost:9515/SMS+Gateway/?a=InstallSoloUser&typetoinstall=OzC_ASP_CSharp
SMS Gateway - 192.168.0.113 - Trial version. Click here to purchase a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Users and applications
Add
admin (Admin) 0.00 MPM
Ready
Add new user/application...
3 users/applications installed
General
Advanced
Help
Ok
Cancel
Start
SMS Gateway
14:57

Install user/application

C# script
You may write a script, that allows you to run C# .NET code when an SMS message arrives. This script can implement any logic to do various tasks with the incoming messages, and optionally send response SMS messages.

Connection name

Please specify a unique name. This can be anything that helps you identify this connection.

Name: csharp_script

Script

Please provide the source of your script you wish to execute.

Source:

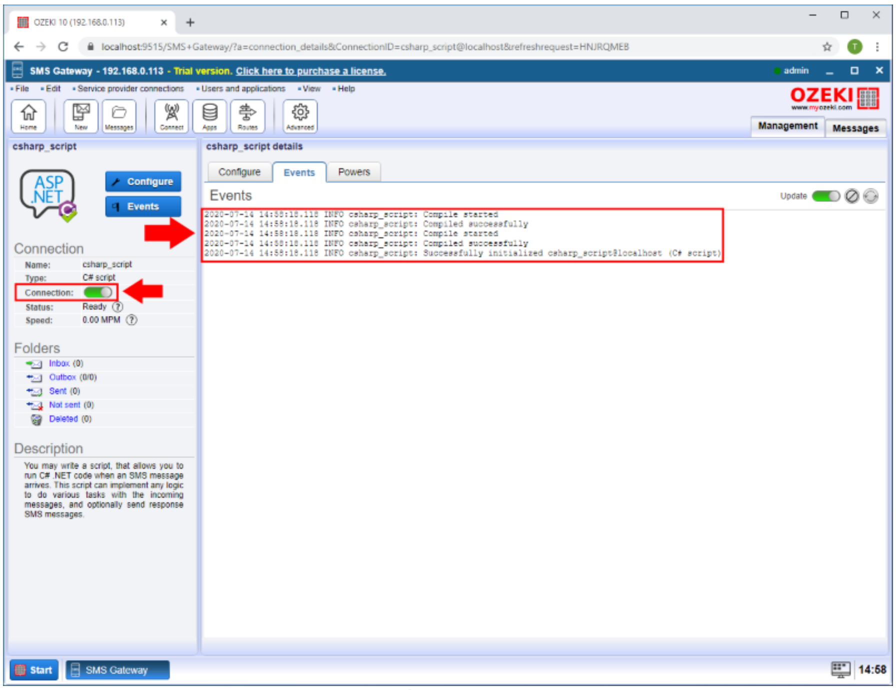
```
using System;

namespace Ozeki
{
    public class Program
    {
        public void Receive(Message msg)
        {
            //Log the message you received
            Log("Received: " + msg.Text);
            //Reply to sender
            Reply(msg, "Thank you for the message!");
        }
    }
}
```

Please fill in the configuration form

**Figure 3** - Provide the script

## Step 4 - Enable user

Please enable te user with the Connection switch and in the Events tab view that the Script is compiled and the user is initialized successfully (Figure 4).



```
2020-07-14 14:58:18.118 INFO csharp_script: Compile started
2020-07-14 14:58:18.118 INFO csharp_script: Compiled successfully
2020-07-14 14:58:18.118 INFO csharp_script: Compile started
2020-07-14 14:58:18.118 INFO csharp_script: Compiled successfully
2020-07-14 14:58:18.118 INFO csharp_script: Successfully initialized csharp_script@localhost (C# script)
```

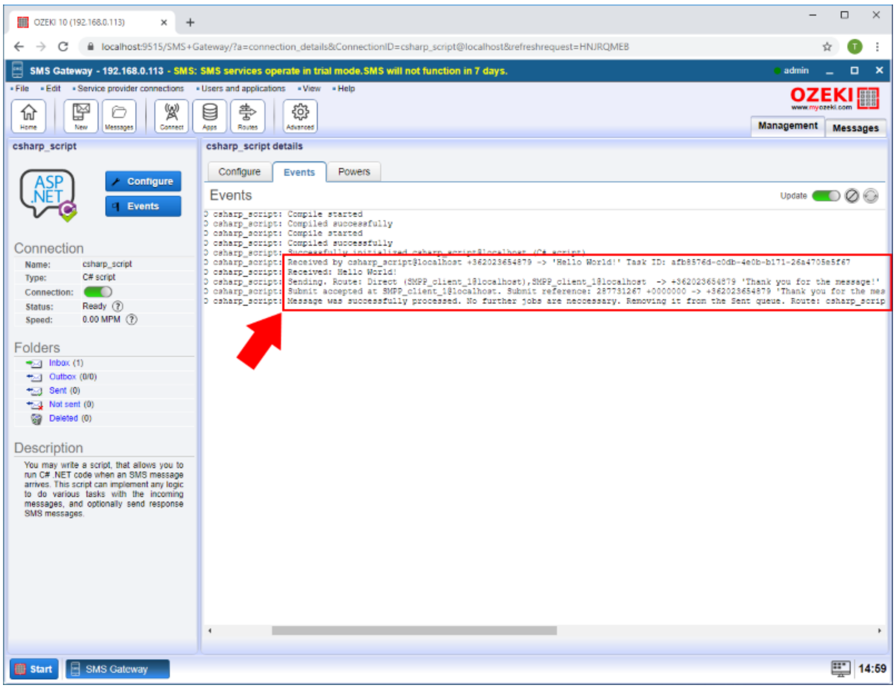
Name: csharp_script
Type: C# script
Status: Ready
Speed: 0.00 MPM

You may write a script, that allows you to run C# .NET code when an SMS message arrives. This script can implement any logic to do various tasks with the incoming messages, and optionally send response SMS messages.

**Figure 4** - Enable user

# Step 5 - Message received

Finally you can see if a message is received by this user the script will run and as this example works the response SMS message is sent to the Original sender (Figure 5).



```
) csharp_script: Compile started
) csharp_script: Compiled successfully
) csharp_script: Compile started
) csharp_script: Compiled successfully
) csharp_script: Successfully initialized csharp_script@localhost (C# script)
) csharp_script: Received by csharp_script@localhost +362023654879 -> 'Hello World!' Task ID: afb8576d-c0db-4e0b-b171-26a4705e5f67
) csharp_script: Received: Hello World!
) csharp_script: Sending. Route: Direct (SMPP_client_1@localhost),SMPP_client_1@localhost  -> +362023654879 'Thank you for the message!'
) csharp_script: Submit accepted at SMPP_client_1@localhost. Submit reference: 287731267 +0000000 -> +362023654879 'Thank you for the mes
) csharp_script: Message was successfully processed. No further jobs are neccessary. Removing it from the Sent queue. Route: csharp_scrip
```

**Figure 5** - Message received

# ASP C# programming guide

Create a class named 'Program' in the 'Ozeki' namespace. You may also include the necessary using directives.

```
using System;

namespace Ozeki
{
public class Program
{
                }
              }
```

Implement the 'Receive' function inside the 'Program' class. This function will be called when your connection receives a message. The parameter of the 'Receive' function will be an object with the type of 'Message'.

```
        using System;

       namespace Ozeki
              {
      public class Program
                {
  public void Receive(Message msg)
                  {
                  }
                }
              }
```

# Send a reply message

When receiving a message, you may send a reply to notify your sender of successful delivery. To achieve this, you may use the built-in 'Reply' function.

```
        using System;

       namespace Ozeki
              {
      public class Program
                {
  public void Receive(Message msg)
                  {
Reply(msg, "Thank you for the message!");
                  }
                }
              }
```

# Message forwarding

You may also forward your received message to an other connection by using the 'Send' function.

```
        using System;

       namespace Ozeki
              {
      public class Program
                {
  public void Receive(Message msg)
                  {
   Send("admin@localhost", msg.Text);
                  }
                }
              }
```

# Compose your message

Composing a message is easier, than you think. You just need to use the built-in 'Message' type to create a new message object. In the following example, when a message was received, we will create and send a new message to the admin connection.

```
        using System;

       namespace Ozeki
              {
      public class Program
```

```
{
public void Receive(Message msg)
{
var message = new Message();
message.Text = "Hello World!";
message.ToAddress = "+4412345678910";
message.ToConnection = "admin@localhost";
message.FromAddress = "+448888999910";

Send(message);
}
}
}
```

# Logging

Using log messages will make debugging your script much easier. By calling the 'Log' function you may implement proper logging in your script if needed.

```
using System;

namespace Ozeki
{
public class Program
{
public void Receive(Message msg)
{
Log("Message received: " + msg.Text);
}
}
}
```

# C# SMS API reference

**'SMSClient' is the namespace of the C# SMS API's class. Add the following line to include it: 'using SMSClient;'. On this page you can find useful properties, methods and events. By modifying the properties you can log into Ozeki SMS Gateway to send and receive SMS.**

## Constructor

'mySMSClient' will be the name of the *C# instance* used in this reference guide.

```
ozSMSClient mySMSClient = new ozSMSClient();
```

## Properties

- **Username:** The name of the SMS Gateway user you would like to use for login. *Default value: admin*
- **Password:** The password belonging to the username. *Default value: abc123*
- **Host:** The IP address of your Ozeki SMS Gateway. *Default value: 127.0.0.1*
- **Port:** The port number of Ozeki SMS Gateway's SMPP server. *Default value: 9500*
- **Connected:** You can build up a connection by changing this bool value to true. *Default value: false*
- **KeepalivePeriod:** The time to wait in seconds between pinging the SMPP server. *Default value: 60*
- **SocketTimeout:** The time span to spend on waiting for the login PDU (in seconds). *Default value: 10*

**Example:**

```
mySMSClient.Username = "admin";
mySMSClient.Password = "abc123";
mySMSClient.Host = "127.0.0.1";
mySMSClient.Port = 9500;

mySMSClient.Connected = true;
```

## Methods

| | Method name | Description |
|---|---|---|
| ◆ | **sendMessage** | You can easily send SMS messages with this method. The recipient's phone number and the text message are mandatory parameters. You can see the parameters of 'sendMessage' by checking out this page. |

**sendMessage example:**

```
mySMSClient.sendMessage("+3620123456","Hello world",
        "param1=value1&"+
        "vp=2018.07.26.+10%3A07%3A58&"+
        "param3=value3");
```

| | Name | Description |
|---|---|---|
| ◆ | **sendMessageEx** | It contains extra parameters compared to 'sendMessage', although both are capable to send SMS messages. All parameters are mandatory, but you can use "" or NULL at the following parameters: sender, serviceprovider and optionalParameterList. |

**sendMessageEx example:**

```
Dictionary<string, string> HelloWorld_parameters = new Dictionary<string, string>();

HelloWorld_parameters.Add("param1", "value1");
HelloWorld_parameters.Add("vp", "2018.07.26.+10%3A07%3A58");
HelloWorld_parameters.Add("param3", "value3");

mySMSClient.sendMessageEx("+3670654321", "+3620123456", "Hello world",
        "SMS:TEXT", "SMPP0", HelloWorld_parameters);
```

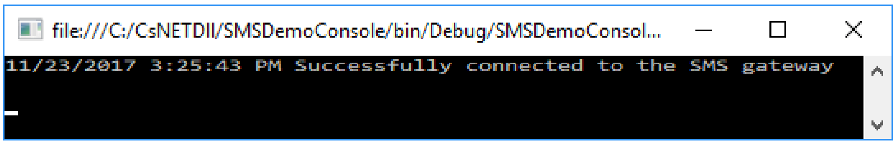
## Events

'mySMSClient' is an example created with the constructor.

**Events connected to mySMSClient**

```
mySMSClient.OnClientConnected += new SimpleEventHandler
                                                (mySMSClient_OnClientConnected);
mySMSClient.OnClientDisconnected += new SimpleEventHandler
                                                        (mySMSClient_OnClientDisconnected);
mySMSClient.OnClientConnectionError += new ErrorEventHandler
                                                        (mySMSClient_OnClientConnectionError);
mySMSClient.OnMessageAcceptedForDelivery += new DeliveryEventHandler
                                                        (mySMSClient_OnMessageAcceptedForDelivery);
mySMSClient.OnMessageDeliveredToNetwork += new DeliveryEventHandler
                                                        (mySMSClient_OnMessageDeliveredToNetwork);
mySMSClient.OnMessageDeliveredToHandset += new DeliveryEventHandler
                                                        (mySMSClient_OnMessageDeliveredToHandset);
mySMSClient.OnMessageDeliveryError += new DeliveryErrorEventHandler
                                                        (mySMSClient_OnMessageDeliveryError);
mySMSClient.OnMessageReceived += new DeliveryEventHandler
                                                (mySMSClient_OnMessageReceived);
```

**Events can be shown on the console** of your C# application (**Figure 1**).



file:///C:/CsNETDll/SMSDemoConsole/bin/Debug/SMSDemoConsol...

```
11/23/2017 3:25:43 PM Successfully connected to the SMS gateway
```

**Figure 1** - An 'OnClientConnected' event

Now look at the detailed description of each event on the table below:

| | Event name | Description |
|---|---|---|
| ◆ | **OnClientConnected** | This event is invoked by your C# application if the connection to the SMPP server of Ozeki SMS Gateway was successful. |
| ◆ | **OnClientDisconnected** | It is invoked if your C# application was disconnected from the SMPP server of Ozeki SMS Gateway. |
| ◆ | **OnClientConnectionError** | It is invoked by your C# application if the connection to the SMPP server of Ozeki SMS Gateway was unsuccessful. |
| ◆ | **OnMessageAcceptedForDelivery** | It is invoked if the message is accepted for delivery by Ozeki SMS Gateway through a user. The SMS Gateway user must be connected to your C# application and the user must have enough SMS credits as well. |
| ◆ | **onMessageDeliveredToNetwork** | It is invoked if the GSM network provider accepts the SMS message for delivery. |
| ◆ | **onMessageDeliveredToHandset** | It is invoked exactly when the SMS message reaches the recipient's mobile phone. The phone can be out of coverage or can also be turned off, so this event might happen long after the 'onMessageDeliveredToNetwork' event. |
| ◆ | **OnMessageDeliveryError** | Message delivery error can happen in 3 different situations:<br>1. There is no recipient with the selected phone number.<br>2. There is no money on the sender's account.<br>3. It has been stored on the GSM server for more time then the validity period. |
| ◆ | **onMessageReceived** | **Your Ozeki SMS Gateway can receive SMS messages too.** In this case an 'onMessageReceived' event will be invoked for your C# application. This happens exactly when the message is received by a user of Ozeki SMS Gateway. This user must be connected to the C# SMS API. |

# sendMessage method

**SMS messages can be sent with the 'sendMessage' method. Before using it, please make sure you are connected to Ozeki SMS Gateway's SMPP server by modifying the properties in the programcode. The recipient's phone number and the text message are mandatory parameters. You can see the parameters of 'sendMessage' in the table below.**

'SMS:TEXT' is the type of these SMS message. To change the type, please use 'sendMessageEx' instead.

| | Parameter name | Description |
|---|---|---|
| | string **receiver**<br>(mandatory) | The recipient's phone number can be provided as an international format, which means it must start with a plus sign, continued by the country code. But it can also be provided in a local number format. |
| | string **messagedata**<br>(mandatory) | The message text to be sent. It can contain unicode characters too (Japanese, Chinese, Indian, Arabic, Hebrew, Greek etc.) Sending unicode messages is not cost efficient, since it can contain 70, while a GSM 7bit SMS can contain 160 characters. |
| | string **optionalParameters** | You can add any optional parameters as you wish in the following form:<br>param1=value1&vp=2018.07.26.+10%3A07%3A58&param3=value3 |

**Example:**

```
mySMSClient.sendMessage("+3620123456","Hello world");
```

**Example** with optional parameters**:**

```
mySMSClient.sendMessage("+3620123456","Hello world",
        "param1=value1&"+
        "vp=2018.07.26.+10%3A07%3A58&"+
        "param3=value3");
```

# sendMessageEx method

**'sendMessageEx' method contains extra parameters compared to 'sendMessage', although both are capable to send SMS messages. Modify the appropriated properties to connect the code to Ozeki SMS Gateway's SMPP server. All parameters are mandatory, but you can use "" or NULL at the following parameters: sender, serviceprovider and optionalParameterList.**

| | Parameter name | Description |
|---|---|---|
| | string **sender**<br>("" if not used) | The sender's phone number can be provided as an international format, which means it must start with a plus sign, continued by the country code. But it can also be provided in a local number format. |
| | string **receiver**<br>(mandatory) | The recipient's phone number can be provided as an international format, which means it must start with a plus sign, continued by the country code. But it can also be provided in a local number format. |
| | string **messagedata**<br>(mandatory) | The message text to be sent. It can contain unicode characters too (Japanese, Chinese, Indian, Arabic, Hebrew, Greek etc.) Sending unicode messages is not cost efficient, since it can contain 70, while a GSM 7bit SMS can contain 160 characters. |
| | string **messagetype**<br>(mandatory) | It is the type of the SMS message you wish to send. Here are a few examples: SMS:TEXT, SMS:WAPPUSH, SMS:VCARD, SMS:VCalendar etc. |
| | string **serviceprovider**<br>("" if not used) | This is the name of the SMPP service provider, which will send the SMS message towards the recipient's cell phone. |
| | Dictionary<string,string><br>**optionalParameters**<br>(NULL if not used) | You can add as many optional parameters as you wish by adding them to a dictionary. See example below. |

**Example:**

```
Dictionary<string, string> HelloWorld_parameters = new Dictionary<string, string>();

HelloWorld_parameters.Add("param1", "value1");
HelloWorld_parameters.Add("vp", "2018.07.26.+10%3A07%3A58");
HelloWorld_parameters.Add("param3", "value3");

mySMSClient.sendMessageEx("+3670654321", "+3620123456", "Hello world",
        "SMS:TEXT", "SMPP0", HelloWorld_parameters);
```

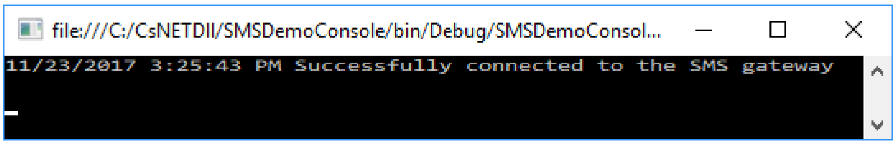
# Events

**The following events can be connected to any SMSClient instance by using an event handler. You can find great examples how to do this in the zip file you can download on the last page.**

Events connected to 'mySMSClient' (example instance)

```
mySMSClient.OnClientConnected += new SimpleEventHandler
                                                (mySMSClient_OnClientConnected);
mySMSClient.OnClientDisconnected += new SimpleEventHandler
                                                        (mySMSClient_OnClientDisconnected);
mySMSClient.OnClientConnectionError += new ErrorEventHandler
                                                        (mySMSClient_OnClientConnectionError);
mySMSClient.OnMessageAcceptedForDelivery += new DeliveryEventHandler
                                                        (mySMSClient_OnMessageAcceptedForDelivery);
mySMSClient.OnMessageDeliveredToNetwork += new DeliveryEventHandler
                                                        (mySMSClient_OnMessageDeliveredToNetwork);
mySMSClient.OnMessageDeliveredToHandset += new DeliveryEventHandler
                                                        (mySMSClient_OnMessageDeliveredToHandset);
mySMSClient.OnMessageDeliveryError += new DeliveryErrorEventHandler
                                                        (mySMSClient_OnMessageDeliveryError);
mySMSClient.OnMessageReceived += new DeliveryEventHandler
                                                (mySMSClient_OnMessageReceived);
```

**Events can be shown on the console** of your C# application (**Figure 1**).



file:///C:/CsNETDll/SMSDemoConsole/bin/Debug/SMSDemoConsol...

```
11/23/2017 3:25:43 PM Successfully connected to the SMS gateway
```

**Figure 1** - An 'OnClientConnected' event

Now look at the detailed description of each event on the table below:

| | Event name | Description |
|---|---|---|
| ◆ | **OnClientConnected** | This event is invoked by your C# application if the connection to the SMPP server of Ozeki SMS Gateway was successful. |
| ◆ | **OnClientDisconnected** | It is invoked if your C# application was disconnected from the SMPP server of Ozeki SMS Gateway. |
| ◆ | **OnClientConnectionError** | It is invoked by your C# application if the connection to the SMPP server of Ozeki SMS Gateway was unsuccessful. |
| ◆ | **OnMessageAcceptedForDelivery** | It is invoked if the message is accepted for delivery by Ozeki SMS Gateway through a user. The SMS Gateway user must be connected to your C# application and the user must have enough SMS credits as well. |
| ◆ | **onMessageDeliveredToNetwork** | It is invoked if the GSM network provider accepts the SMS message for delivery. |
| ◆ | **onMessageDeliveredToHandset** | It is invoked exactly when the SMS message reaches the recipient's mobile phone. The phone can be out of coverage or can also be turned off, so this event might happen long after the 'onMessageDeliveredToNetwork' event. |
| ◆ | **OnMessageDeliveryError** | Message delivery error can happen in 3 different situations:<br>1. There is no recipient with the selected phone number.<br>2. There is no money on the sender's account.<br>3. It has been stored on the GSM server for more time then the validity period. |
| ◆ | **onMessageReceived** | **Your Ozeki SMS Gateway can receive SMS messages too.** In this case an 'onMessageReceived' event will be invoked for your C# application. This happens exactly when the message is received by a user of Ozeki SMS Gateway. This user must be connected to the C# SMS API. |

# OnClientConnected event

**It is invoked by your C# application if the connection to the SMPP server of Ozeki SMS Gateway was successful. You can do this by providing the correct instance properties both for reaching the SMPP server and the SMS Gateway user.**

It uses the 'SimpleEventHandler', which has the following event args delegated to it:

```
public class EventArgs
{
    public static readonly EventArgs Empty;

    public EventArgs();
}

public delegate void SimpleEventHandler(object sender, EventArgs e);
```

You can publish the event by using the 'SimpleEventHandler':

```
public event SimpleEventHandler OnClientConnected;
```

You can create an event thread with the 'OnClientConnected' event:

```
ThreadStart starter = delegate { OnClientConnected(this, new EventArgs()); };
new Thread(starter).Start();
```

*All of these codes were copied from the demo project.*

<table>
<tr><th></th><th>Method plus Parameters</th><th>Description</th></tr>
<tr><td></td><td>

void **OnClientConnected** (this, EventArgs e);

</td><td>

This event is invoked by your C# application if the connection to Ozeki SMS Gateway's SMPP server was successful. Here you can see how to create an event handler and attach it to an instance 'ozSMSClient'.

**Parameters**

- **this:** The current instance of the class ('ozSMSClient.cs'), which is called 'mySMSClient' in the current example.
- **e:** It is empty. It contains no arguments. (scroll up to see more)

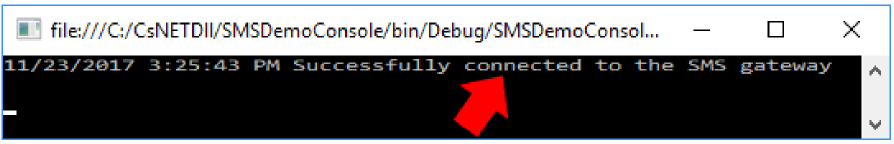
**How to use it?**
**Example is in the SMSDemoConsole project of CsNETDll_v3.zip:**

**Step 1:** Create an event handler

```
static void mySMSClient_OnClientConnected
(object sender, EventArgs e)
{
WriteEvent(DateTime.Now.ToString() +
" Successfully connected to the SMS gateway " + "\r\n");
}
```

**Step 2:** Attach event handler to the created instance

```
mySMSClient.OnClientConnected += new SimpleEventHandler
                (mySMSClient_OnClientConnected);
```


file:///C:/CsNETDll/SMSDemoConsole/bin/Debug/SMSDemoConsol...

```
11/23/2017 3:25:43 PM Successfully connected to the SMS gateway
```

**Figure 1 -** The event handler writes on the console by using example code

</td></tr>
</table>

# OnClientDisconnected event

**It is invoked if your C# application was disconnected from the SMPP server of Ozeki SMS Gateway. This is the case if Ozeki 10 service stops or the built-in SMPP server goes offline.**

It uses the 'SimpleEventHandler', which has the following event args delegated to it:

```
public class EventArgs
{
    public static readonly EventArgs Empty;

    public EventArgs();
}

public delegate void SimpleEventHandler(object sender, EventArgs e);
```

You can publish the event by using the 'SimpleEventHandler':

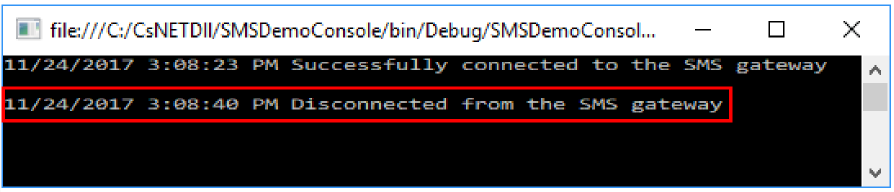
```
public event SimpleEventHandler OnClientDisconnected;
```

'OnClientDisconnected' event is called after losing connection:

```
private void doOnClientDisconnected()
{
    fConnected = false;
    if (OnClientDisconnected != null)
    {
        try
        {
        OnClientDisconnected(this, new EventArgs());
        }
        catch { }
    }
}
```

*All of these codes were copied from the demo project.*

<table>
<tr><th></th><th>Method plus Parameters</th><th>Description</th></tr>
<tr><td></td><td>void <b>OnClientDisconnected</b><br>(this, EventArgs e);</td><td>This event is invoked if your C# application was disconnected from the SMPP server of Ozeki SMS Gateway. Here you can see how to create an event handler and attach it to an instance of 'ozSMSClient'.<br><br><b>Parameters</b><br>• <b>this:</b> The current instance of the class ('ozSMSClient.cs'), which is called 'mySMSClient' in the current example.<br>• <b>e:</b> It is empty. It contains no arguments.<br>(scroll up to see more)<br><br><b>How to use it?</b><br><b>Example is in the SMSDemoConsole project of CsNETDll_v3.zip:</b><br><br><b>Step 1:</b> Create an event handler<pre>
static void mySMSClient_OnClientDisconnected
(object sender, EventArgs e)
{
WriteEvent(DateTime.Now.ToString() +
" Disconnected from the SMS gateway " + "\r\n");
}
</pre><b>Step 2:</b> Attach event handler to the created instance<pre>
mySMSClient.OnClientDisconnected += new SimpleEventHandler
                (mySMSClient_OnClientDisconnected);
</pre></td></tr>
</table>


file:///C:/CsNETDII/SMSDemoConsole/bin/Debug/SMSDemoConsol...

```
11/24/2017 3:08:23 PM Successfully connected to the SMS gateway
11/24/2017 3:08:40 PM Disconnected from the SMS gateway
```

**Figure 1 -** The event handler writes on the console by using example code

# OnClientConnectionError event

**It is invoked by your C# application if the connection to the SMPP server of Ozeki SMS Gateway was unsuccessful. This can be the case if you have provided wrong instance properties for reaching the SMPP server or the SMS Gateway user. Remember to check the Ozeki 10 service and the built-in SMPP server to make sure they are running.**

It uses the 'ErrorEventHandler', which has the following event args delegated to it:

```
public class ErrorEventArgs : EventArgs // is empty
{
    public int ErrorCode;
    public string ErrorMessage;
}

public delegate void ErrorEventHandler(object sender, ErrorEventArgs e);
```

You can publish the event by using the 'ErrorEventHandler':

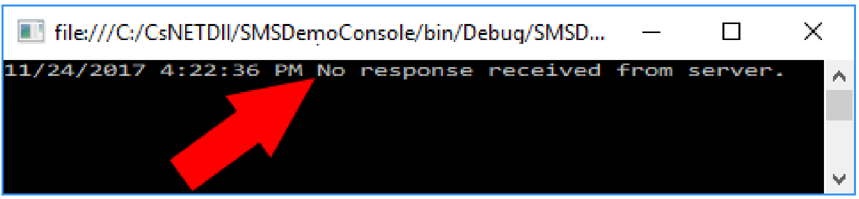
```
public event ErrorEventHandler OnClientConnectionError;
```

You can create an event thread with the 'OnClientConnectionError' event:

```
ErrorEventArgs ea = new ErrorEventArgs();
ea.ErrorCode = lastErrorCode;
ea.ErrorMessage = lastErrorMessage;

ThreadStart starter = delegate { OnClientConnectionError(this, ea); };
new Thread(starter).Start();
```

*All of these codes were copied from the demo project.*

| | Method plus Parameters | Description |
|---|---|---|
| | void **OnClientConnectionError** (this, ErrorEventArgs ea); | This event is invoked by your C# application if the connection to the SMPP server of Ozeki SMS Gateway was unsuccessful. Here you can see how to create an event handler and attach it to an instance of 'ozSMSClient'.<br>**Parameters**<br>- **this:** The current instance of the class ('ozSMSClient.cs'), which is called 'mySMSClient' in the current example.<br>- **ea:** It contains fields from the ErrorEventArgs class (scroll up to see more)<br>**How to use it?**<br>**Example is in the SMSDemoConsole project of CsNETDll_v3.zip:**<br>**Step 1:** Create an event handler<br>`static void mySMSClient_OnClientConnectionError`<br>`(object sender, ErrorEventArgs e)`<br>`{`<br>`WriteEvent(DateTime.Now.ToString() + " " +`<br>`e.ErrorMessage + "\r\n");`<br>`}`<br>**Step 2:** Attach event handler to the created instance<br>`mySMSClient.OnClientConnectionError += new ErrorEventHandler`<br>`(mySMSClient_OnClientConnectionError);` |

| | | |
|---|---|---|
| | | <br>file:///C:/CsNETDII/SMSDemoConsole/bin/Debug/SMSD...<br>`11/24/2017 4:22:36 PM No response received from server.`<br>**Figure 1** - The event handler writes on the console by using example code |

# OnMessageAcceptedForDelivery event

**It is invoked if the text message can be delivered by Ozeki SMS Gateway's user. The user must be connected to your C# application. SMS Gateway checks if the message is accepted for delivery. It also checks if the user has enough SMS credits in their SMS Gateway account.**

It uses the 'DeliveryEventHandler', which has the following event args delegated to it:

```
public class DeliveryEventArgs : EventArgs //is empty
{
    public string Messageid;
    public string Senderaddress;
    public string Receiver;
    public string Messagedata;
    public string Messagetype;
    public string Serviceprovider;
    public DateTime Sentdate;
    public DateTime Receiveddate;
}

public delegate void DeliveryEventHandler(object sender, DeliveryEventArgs e);
```

You can publish the event by using the 'DeliveryEventHandler':

```
public event DeliveryEventHandler OnMessageAcceptedForDelivery;
```

You can create an event thread with the 'OnMessageAcceptedForDelivery' event:

```
DeliveryEventArgs ea = new DeliveryEventArgs();
ea.Messageid = sms.messageID;
ea.Receiver = sms.receiver;
ea.Senderaddress = sms.sender;
ea.Sentdate = sms.sentdate;
ea.Receiveddate = sms.receiveddate;
ea.Messagedata = sms.messagedata;
ea.Messagetype = sms.messagetype;
ea.Serviceprovider = sms.serviceprovider;

ThreadStart starter = delegate { OnMessageAcceptedForDelivery(this, ea); };
new Thread(starter).Start();
```

*All of these codes were copied from the demo project.*

| | Method plus Parameters | Description |
|---|---|---|
| | void<br>**OnMessageAcceptedForDelivery**<br>(this, DeliveryEventArgs ea); | It is invoked if the message is accepted for delivery by Ozeki SMS Gateway through a user. The SMS Gateway user must be connected to your C# application. Here you can see how to create an event handler and attach it to an instance of 'ozSMSClient'.<br>**Parameters**<br>• **this:** The current instance of the class ('ozSMSClient.cs'), which is called 'mySMSClient' in the current example.<br>• **ea:** It contains fields from the DeliveryEventArgs class (scroll up to see more)<br>**How to use it?**<br>**Example is in the SMSDemoConsole project of CsNETDll_v3.zip:**<br>**Step 1:** Create an event handler<br>`static void mySMSClient_OnMessageAcceptedForDelivery`<br>`(object sender, DeliveryEventArgs e)`<br>`{`<br>`WriteEvent(DateTime.Now.ToString() + " " +`<br>`"Message accepted for delivery. ID: " + e.Messageid + "\r\n");`<br>`}` |

|  |  |  |
|---|---|---|
|  |  | **Step 2:** Attach event handler to the created instance<br>`mySMSClient.OnMessageAcceptedForDelivery += new DeliveryEventHandler`<br>`                (mySMSClient_OnMessageAcceptedForDelivery);` |

# onMessageDeliveredToNetwork event

**It is invoked if the GSM network provider accepts the SMS message for delivery. More SMS messages can be queued up in the outbox for delivery and the delivery time can depend on the GSM service provider's capacity.**

A GSM service provider is a hardware or service used for sending SMS messages, which can be a GSM modem connected to your PC or any IP SMS service provider you prefer.

It uses the 'DeliveryEventHandler', which has the following event args delegated to it:

```
public class DeliveryEventArgs : EventArgs //is empty
{
    public string Messageid;
    public string Senderaddress;
    public string Receiver;
    public string Messagedata;
    public string Messagetype;
    public string Serviceprovider;
    public DateTime Sentdate;
    public DateTime Receiveddate;
}

public delegate void DeliveryEventHandler(object sender, DeliveryEventArgs e);
```

You can publish the event by using the 'DeliveryEventHandler':

```
public event DeliveryEventHandler OnMessageDeliveredToNetwork;
```

You can create an event thread with the 'OnMessageDeliveredToNetwork' event:

```
DeliveryEventArgs ea = new DeliveryEventArgs();
ea.Messageid = sms.messageID;
ea.Receiver = sms.receiver;
ea.Senderaddress = sms.sender;
ea.Sentdate = sms.sentdate;
ea.Receiveddate = sms.receiveddate;
ea.Messagedata = sms.messagedata;
ea.Messagetype = sms.messagetype;
ea.Serviceprovider = sms.serviceprovider;

ThreadStart starter = delegate { OnMessageDeliveredToNetwork(this, ea); };
new Thread(starter).Start();
```

*All of these codes were copied from the demo project.*

| | Method plus Parameters | Description |
|---|---|---|
| | void<br>**onMessageDeliveredToNetwork**<br>(this, DeliveryEventArgs ea); | This event will be invoked if the GSM network provider accepts the SMS message for delivery. Here you can see how to create an event handler and attach it to an instance of 'ozSMSClient'.<br>**Parameters**<br>• **this:** The current instance of the class ('ozSMSClient.cs'), which is called 'mySMSClient' in the current example.<br>• **ea:** It contains fields from the DeliveryEventArgs class (scroll up to see more)<br>**How to use it?**<br>**Example is in the SMSDemoConsole project of CsNETDll_v3.zip:**<br>**Step 1:** Create an event handler<br>`static void mySMSClient_OnMessageDeliveredToNetwork`<br>`(object sender, DeliveryEventArgs e)`<br>`{`<br>`WriteEvent(DateTime.Now.ToString() + " " +` |

```
"Message delivered to network. ID: " + e.Messageid + "\r\n");
}
```

**Step 2:** Attach event handler to the created instance

```
mySMSClient.OnMessageDeliveredToNetwork += new DeliveryEventHandler
                (mySMSClient_OnMessageDeliveredToNetwork);
```

# onMessageDeliveredToHandset event

**It is invoked exactly when the SMS message reaches the recipient's mobile phone. This event happens some time after the 'onMessageDeliveredToNetwork' event and is sent by the GSM network provider. The SMS will only arrive when the recipient turns on the mobile phone or gets coverage.**

It uses the 'DeliveryEventHandler', which has the following event args delegated to it:

```
public class DeliveryEventArgs : EventArgs //is empty
{
    public string Messageid;
    public string Senderaddress;
    public string Receiver;
    public string Messagedata;
    public string Messagetype;
    public string Serviceprovider;
    public DateTime Sentdate;
    public DateTime Receiveddate;
}

public delegate void DeliveryEventHandler(object sender, DeliveryEventArgs e);
```

You can publish the event by using the 'DeliveryEventHandler':

```
public event DeliveryEventHandler OnMessageDeliveredToHandset;
```

You can create an event thread with the 'OnMessageDeliveredToHandset' event:

```
DeliveryEventArgs ea = new DeliveryEventArgs();
ea.Messageid = sms.messageID;
ea.Receiver = sms.receiver;
ea.Senderaddress = sms.sender;
ea.Sentdate = sms.sentdate;
ea.Receiveddate = sms.receiveddate;
ea.Messagedata = sms.messagedata;
ea.Messagetype = sms.messagetype;
ea.Serviceprovider = sms.serviceprovider;

ThreadStart starter = delegate { OnMessageDeliveredToHandset(this, ea); };
new Thread(starter).Start();
```

*All of these codes were copied from the demo project.*

| | Method plus Parameters | Description |
|---|---|---|
| | void<br>**OnMessageDeliveredToHandset**<br>(this, DeliveryEventArgs ea); | This event will be invoked exactly when the SMS message reaches the recipient's mobile phone. Here you can see how to create an event handler and attach it to an instance of 'ozSMSClient'.<br>**Parameters**<br>• **this:** The current instance of the class ('ozSMSClient.cs'), which is called 'mySMSClient' in the current example.<br>• **ea:** It contains fields from the DeliveryEventArgs class (scroll up to see more)<br>**How to use it?**<br>**Example is in the SMSDemoConsole project of CsNETDII_v3.zip:**<br>**Step 1:** Create an event handler<br>`static void mySMSClient_OnMessageDeliveredToHandset`<br>`(object sender, DeliveryEventArgs e)`<br>`{`<br>`WriteEvent(DateTime.Now.ToString() + " " +`<br>`"Message delivered to handset. ID: " + e.Messageid + "\r\n");`<br>`}` |

**Step 2:** Attach event handler to the created instance

```
mySMSClient.OnMessageDeliveredToHandset += new DeliveryEventHandler
                (mySMSClient_OnMessageDeliveredToHandset);
```

# OnMessageDeliveryError event

**Message delivery error can happen in 3 different situations:**
**1. There is no recipient with the selected phone number.**
**2. There is no money on the sender's account.**
**3. It has been stored on the GSM server for more time then the validity period.**

It uses the 'DeliveryErrorEventHandler', which has the following event args delegated to it:

```
public class DeliveryErrorEventArgs : DeliveryEventArgs
{
    public int ErrorCode;
    public string ErrorMessage;
    //Plus it contains the args in DeliveryEventArgs:
    //public string Messageid;
    //public string Senderaddress;
    //public string Receiver;
    //public string Messagedata;
    //public string Messagetype;
    //public string Serviceprovider;
    //public DateTime Sentdate;
    //public DateTime Receiveddate;
}

public delegate void DeliveryErrorEventHandler(object sender, DeliveryErrorEventArgs e);
```

You can publish the event by using the 'DeliveryErrorEventHandler':

```
public event DeliveryErrorEventHandler OnMessageDeliveryError;
```

You can create an event thread with the 'OnMessageDeliveryError' event:

```
DeliveryErrorEventArgs ea = new DeliveryErrorEventArgs();
ea.Messageid = sms.messageID;
ea.Receiver = sms.receiver;
ea.Senderaddress = sms.sender;
ea.Sentdate = sms.sentdate;
ea.Receiveddate = sms.receiveddate;
ea.Messagedata = sms.messagedata;
ea.Messagetype = sms.messagetype;
ea.Serviceprovider = sms.serviceprovider;
ea.ErrorCode = sms.errorcode;
ea.ErrorMessage = sms.errormessage;

ThreadStart starter = delegate { OnMessageDeliveryError(this, ea); };
new Thread(starter).Start();
```

*All of these codes were copied from the demo project.*

| | Method plus Parameters | Description |
|---|---|---|
| | void **OnMessageDeliveryError** (this, DeliveryErrorEventArgs ea); | This event will be invoked when the SMS message cannot be delivered. Here you can see how to create an event handler and attach it to an instance of 'ozSMSClient'.<br>**Parameters**<br>• **this:** The current instance of the class ('ozSMSClient.cs'), which is called 'mySMSClient' in the current example.<br>• **ea:** It contains fields from the DeliveryErrorEventArgs class (scroll up to see more)<br>**How to use it?**<br>**Example is in the SMSDemoConsole project of CsNETDll_v3.zip:**<br>**Step 1:** Create an event handler<br>`static void mySMSClient_OnMessageDeliveryError` |

```
(object sender, DeliveryErrorEventArgs e)
{
WriteEvent(DateTime.Now.ToString() + " " +
"Message could not be delivered. ID: " + e.Messageid +
" Error message: " + e.ErrorMessage + "\r\n");
}
```

**Step 2:** Attach event handler to the created instance

```
mySMSClient.OnMessageDeliveryError += new DeliveryErrorEventHandler
                (mySMSClient_OnMessageDeliveryError);
```

# OnMessageReceived event

**Your Ozeki SMS Gateway can receive SMS messages too. In this case an 'onMessageReceived' event will be invoked for your C# application. This happens exactly when the message is received by a user of Ozeki SMS Gateway. This user must be connected to the C# SMS API.**

It uses the 'DeliveryEventHandler', which has the following event args delegated to it:

```
public class DeliveryEventArgs : EventArgs //is empty
{
    public string Messageid;
    public string Senderaddress;
    public string Receiver;
    public string Messagedata;
    public string Messagetype;
    public string Serviceprovider;
    public DateTime Sentdate;
    public DateTime Receiveddate;
}

public delegate void DeliveryEventHandler(object sender, DeliveryEventArgs e);
```

You can publish the event by using the 'DeliveryEventHandler':

```
public event DeliveryEventHandler OnMessageReceived;
```

You can create an event thread with the 'OnMessageReceived' event:

```
DeliveryEventArgs ea = new DeliveryEventArgs();
ea.Messageid = sms.messageID;
ea.Receiver = sms.receiver;
ea.Senderaddress = sms.sender;
ea.Sentdate = sms.sentdate;
ea.Receiveddate = sms.receiveddate;
ea.Messagedata = sms.messagedata;
ea.Messagetype = sms.messagetype;
ea.Serviceprovider = sms.serviceprovider;

ThreadStart starter = delegate { OnMessageReceived(this, ea); };
new Thread(starter).Start();
```

*All of these codes were copied from the demo project.*

| | Method plus Parameters | Description |
|---|---|---|
| | void **OnMessageReceived**<br>(this, DeliveryEventArgs ea); | This event will be invoked exactly when an SMS message is received by Ozeki SMS Gateway's user. The user must be connected to the C# SMS API. Here you can see how to create an event handler and attach it to an instance of 'ozSMSClient'.<br>**Parameters**<br>• **this:** The current instance of the class ('ozSMSClient.cs'), which is called 'mySMSClient' in the current example.<br>• **ea:** It contains fields from the DeliveryEventArgs class (scroll up to see more)<br>**How to use it?**<br>**Example is in the SMSDemoConsole project of CsNETDll_v3.zip:**<br>**Step 1:** Create an event handler<br>`static void mySMSClient_OnMessageReceived`<br>`(object sender, DeliveryEventArgs e)`<br>`{`<br>`WriteEvent(DateTime.Now.ToString() + " " +`<br>`"Message received. Sender address: " + e.Senderaddress +` |

```
" Message text: " + e.Messagedata + "\r\n");
}
```

**Step 2:** Attach event handler to the created instance

```
mySMSClient.OnMessageReceived += new DeliveryEventHandler
                (mySMSClient_OnMessageReceived);
```

# Sending SMS through Microsoft SQL Server

Download: Sending_SMS_through_MS_SQL_Server.zip

**See how to send SMS by inserting rows into a Microsoft SQL database through a C#.NET application. This technology is intended for developers with basic knowledge in C#.NET and SQL. The downloadable source code helps you get started.**

In the following chapters you can find the required prerequisites and a detailed explanation on how to use the code. A helpful workflow diagram shows you the basic connection between the C# application's user and the recipient's phone.

**The code is useful if who would like to**

- include SMS functionality to your C# application.
- integrate automated SMS notification.
- secure your products by adding SMS login.

### Prerequisites

The software requirements of the system is listed on the following table. Please download and install Ozeki SMS Gateway with .NET framework 4.5 and a Microsoft Visual Studio to run your code.

| | |
|---|---|
| Operating System: | Windows Vista, Windows 7, Windows Server 2008 R2, Windows Server 2012 R2, Windows Server 2016, Windows 8 or Windows 10 |
| Basic software requirements: | .NET Framework 4.5<br>Ozeki SMS Gateway<br>Microsoft SQL Server 2005 Express Edition or newer Microsoft SQL Server versions |
| Development platform: | Microsoft Visual Studio |

## How does it work

First you need to install Ozeki SMS Gateway and create a Database User in the SMS Gateway. Connect the user to your MSSQL database. The SMS messages to send will be SELECT-ed from the outgoing messages SQL table, which is called 'ozekimessageout' as default.

Then you should install a C#.NET environment. Your C# application can connect to your MSSQL database and insert SMS messages to the SQL table of outgoing messages.

Make sure your Ozeki SMS Gateway is connected to the GSM network through a GSM modem or any IP SMS service provider over the internet. For example SMPP, CIMD2 or UCP/EMI are very popular service provider connections.

**See the workflow of the C# through MSSQL connection on Figure 1.**

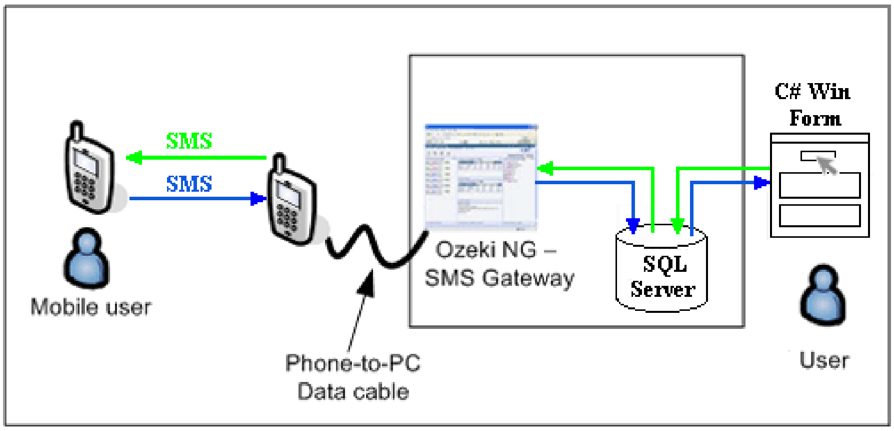


**Figure 1** - Message flow from a cellphone to your C# application and vice versa

So basicly the **Microsoft SQL Server can be reached from both your C# application and Ozeki SMS Gateway**. All incoming and outgoing messages are stored on the MSSQL Server for further usage.

By adding the appropriate codes to your C# application, you can connect it to your SQL database to make it able to INSERT new rows into the outgoing message table and read incoming messages.

Ozeki SMS Gateway's Database User is capable to read the outgoing message table to send SMS messages and INSERT new rows into the incoming message table for other users to read. For example the C# application user can read it.

# How to set up your MSSQL Server

**Step 1:** Install Microsoft SQL Server 2005 Express Edition
**Step 2:** Start Microsoft SQL Server 2005 Express Edition and log in.
**Step 3:** Turn on server authentication, so it would always require database user and password pair. (Short help tutorial)
**Step 4:** Create database and grant select, insert, update, delete permissions to a user. (Short help tutorial)

# Add a Database User to Ozeki SMS Gateway

**Step 1:** Start Ozeki 10 browser GUI (Picture help)
**Step 2:** Start Ozeki SMS Gateway from Ozeki 10's desktop.
**Step 3:** Install a Database User and set the connection string. (Short help tutorial)

# Use the downloaded C# code

**Step 1:** Download the zip file and unpack it.
**Step 2:** Set SQL database connection data in 'DatabaseHandling.cs'. (Short help tutorial)
**Step 3:** Start Ozeki 10 service if it is not running. (Short help tutorial)
**Step 4:** Build and run the project in Microsoft Visual Studio
**Step 5:** A window will pop up. Click the **Compose message** button.
Fill the recipient and message text fields and click 'Send'. (Short help tutorial)

# How does the example code work

The downloadable code on the top of the page can insert messages in the 'ozekimessageout' table. Ozeki SMS Gateway's Database User periodically checks 'ozekimessageout' to find new message rows and send them to the recipient's phone. Each message row has a status attribute. After sending the SMS message, the status will change from 'Send' to 'Sent'.

Do not forget to check if your Microsoft SQL server is up and running and modify the server connection details in 'DatabaseHandling.cs', which you can find in the example project.

### C# classes of the example code

**MainForm.cs (Figure 2):**
This class contains the first window that opens up. As you can see on Figure 2, it can show the content of two tables. This content is read by a SELECT statement from 'ozekimessageout' and 'ozekimessagein'. It also contains two 'Refresh' buttons and a 'Compose message' button as well. Press it to use the next class, 'ComposeMessageForm.cs'.

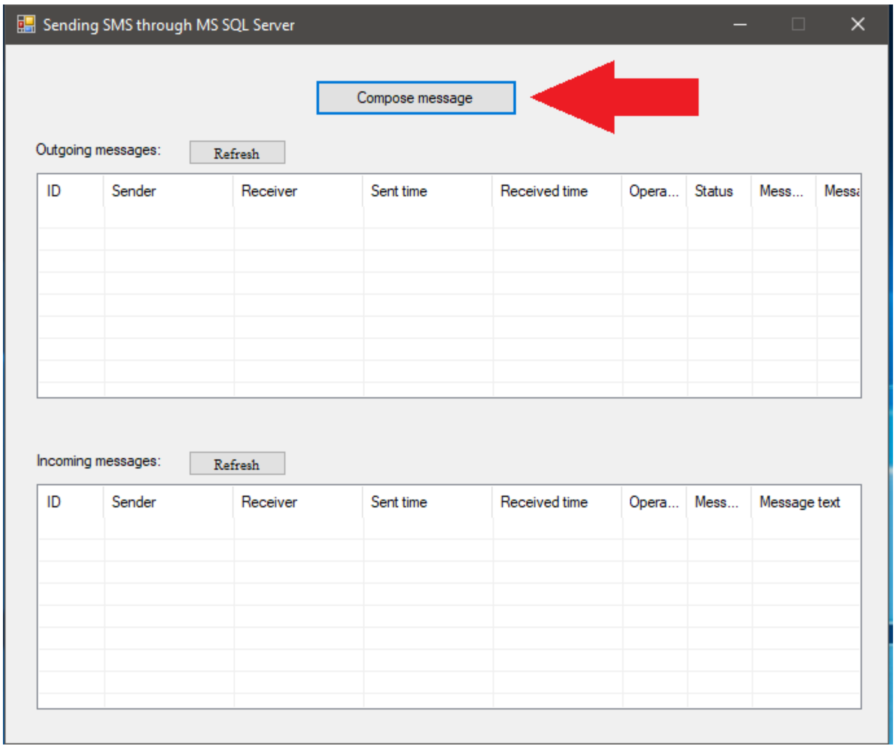
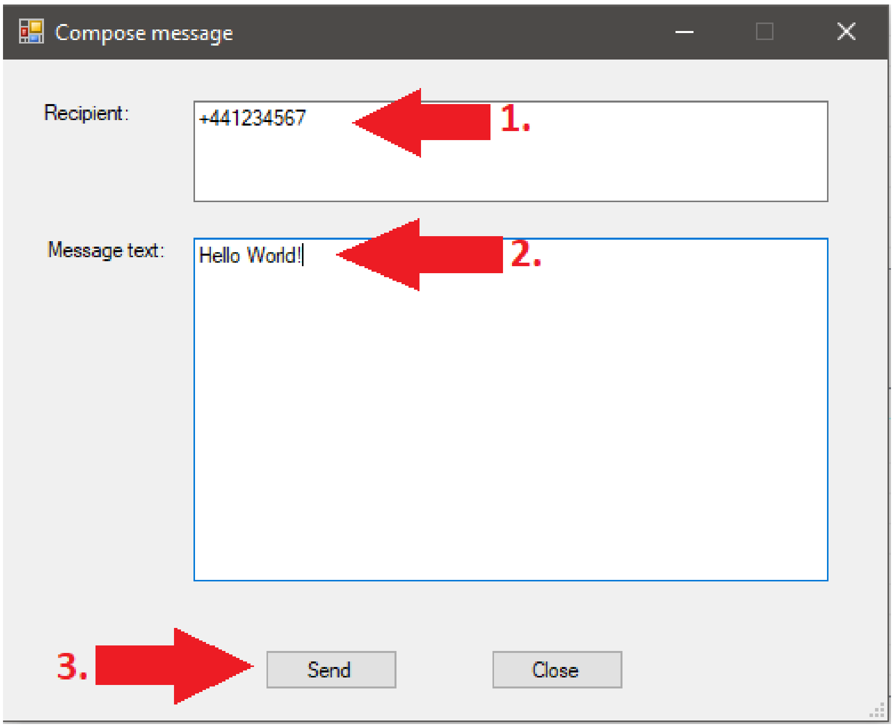


**Figure 2** - The GUI generated from 'MainForm.cs'

**ComposeMessageForm.cs (Figure 3):**
This is the next page where you can fill the necessary data fields to create a new message.

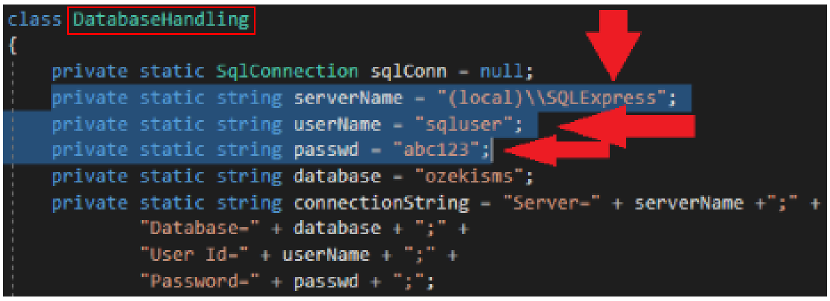


Recipient: +441234567

Message text: Hello World!

**Figure 3** - The GUI generated from 'ComposeMessageForm.cs'

**DatabaseHandling.cs:**
This is where you can set the database connection (**Figure 4**) and refresh the tables of 'MainForm.cs'.



```
class DatabaseHandling
{
    private static SqlConnection sqlConn = null;
    private static string serverName = "(local)\\SQLExpress";
    private static string userName = "sqluser";
    private static string passwd = "abc123";
    private static string database = "ozekisms";
    private static string connectionString = "Server=" + serverName +";" +
            "Database=" + database + ";" +
            "User Id=" + userName + ";" +
            "Password=" + passwd + ";";
```

**Figure 4** - Database connection settings in 'DatabaseHandling.cs'

**How does the message sending code work**

In 'ComposeMessageForm.cs' you can fill the two fields with the recipient's address and message text. By pressing 'Send', you can start the following code:

**ComposeMessageForm.cs**

```
...
private void buttonSend_Click(object sender, EventArgs e)
{
    CheckAndSendMessage();
}

private void CheckAndSendMessage()
{
    if (textBoxRecipient.Text == "")
    {
        MessageBox.Show("Recipient field mustn't be empty!",
                    "Incorrect field value");
        return;
    }

    string errorMsg = "";
    DatabaseHandling.insertMessage(textBoxRecipient.Text,
                textBoxMessageText.Text, out errorMsg);

    MessageBox.Show(errorMsg, "Result of inserting message");
}
...
```

**Figure 5** - Checks if the recipient's field is empty

The CheckAndSendMessage() method is called by 'buttonSend_Click(...)'. The method starts by checking the recipient's field and runs insertMessage(...) from 'DatabaseHandling.cs' if the recipient's field is NOT empty (**Figure 5**). insertMessage(...) can insert a new row to the 'ozekimessageout' table (**Figure 6**) (The default value of the msgtype attribute is SMS:TEXT). *Ozeki SMS Gateway will read the rows from the SQL server as you can see on the workflow diagram above.*

**DatabaseHandling.cs**

```
...
public static void insertMessage(string receiver, string messageText,
            out string errorMsg)
{
    Connect(out errorMsg);
    if (errorMsg != "")
       return;

    try
    {
       SqlCommand sqlComm = sqlConn.CreateCommand();
       sqlComm.CommandText = "insert into ozekimessageout " +
                    "(msgtype,receiver,msg,status) " +
                    "values ('SMS:TEXT','" + receiver + "','" +
                    messageText + "','send');";
       if (sqlComm.ExecuteNonQuery() == 0)
       {
            errorMsg = "Insert was UNsuccessful!";
       }
       else
       {
            errorMsg = "Insert was successful!";
       }
    }
    catch (Exception e)
    {
        errorMsg = e.Message;
    }

    CloseConnection();
}
...
```

**Figure 6** - INSERT's message into your SQL database

You can use other message types than 'SMS:TEXT' (e.g. 'SMS:WAPPUSH', 'SMS:VCARD') as you can see on **Figure 7**.

```
...
sqlComm.CommandText = "insert into ozekimessageout " +
                      "(msgtype,receiver,msg,status) " +
                      "values ('SMS:WAPPUSH','" + receiver + "','" +
                      messageText + "','send');";
...
```

**Figure 7** - Message type changed from 'SMS:TEXT' to 'SMS:WAPPUSH'

## Frequently asked questions

**Question:** Can this C# example run on a different computer than Ozeki SMS Gateway's or MSSQL server's machine?
**Answer:** Yes, it can. Please modify the IP address in 'DatabaseHandling.cs' to your MSSQL server's IP address.

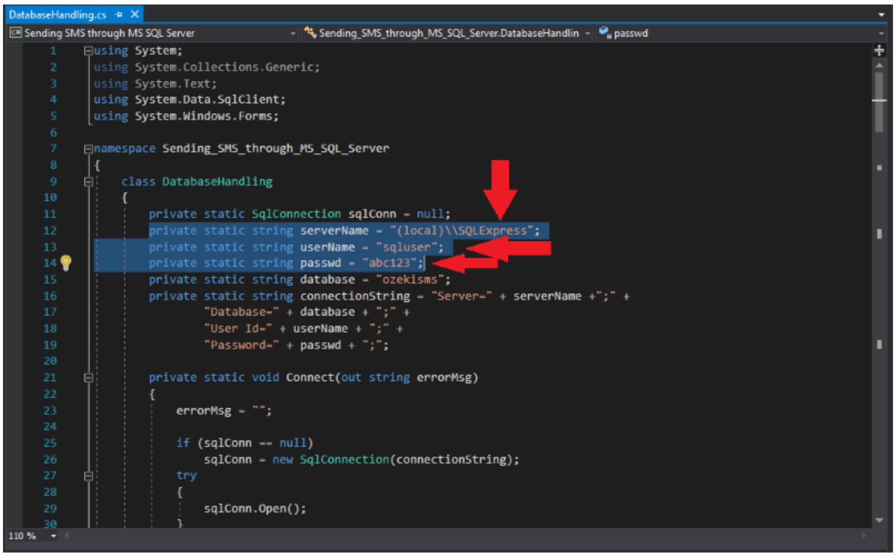
**Question:** Can I change the sender's phone number?
**Answer:** Yes. Please INSERT the sender's number into the message row as well (**Figure 8**).
It only works if you have an IP SMS connection.

```
...
string SenderNumber = "+449876543";
sqlComm.CommandText = "insert into ozekimessageout " +
                      "(msgtype,sender,receiver,msg,status) " +
                      "values ('SMS:TEXT','" + SenderNumber + "','" + receiver + "','" +
                      messageText + "','send');";
...
```

**Figure 8** - Modified INSERT INTO statement with an additional 'sender' attritube

# Connect to your Microsoft SQL server

**Open the downloaded project and provide the MS SQL server connection details to 'DatabaseHandling.cs'. These details contain the server name, username, password and database name. These 4 fields are included in the connection string.**


DatabaseHandling.cs
Sending SMS through MS SQL Server
Sending_SMS_through_MS_SQL_Server.DatabaseHandlin
passwd
110 %

```
using System;
using System.Collections.Generic;
using System.Text;
using System.Data.SqlClient;
using System.Windows.Forms;

namespace Sending_SMS_through_MS_SQL_Server
{
    class DatabaseHandling
    {
        private static SqlConnection sqlConn = null;
        private static string serverName = "(local)\\SQLExpress";
        private static string userName = "sqluser";
        private static string passwd = "abc123";
        private static string database = "ozekisms";
        private static string connectionString = "Server=" + serverName +";" +
                "Database=" + database + ";" +
                "User Id=" + userName + ";" +
                "Password=" + passwd + ";";

        private static void Connect(out string errorMsg)
        {
            errorMsg = "";

            if (sqlConn == null)
                sqlConn = new SqlConnection(connectionString);
            try
            {
                sqlConn.Open();
            }
```

**Figure 1** - Setting connection information in 'DatabaseHandling.cs'

The MS SQL connection string is built up from the provided connection details (**Figure 2**). The 'serverName' is the domain name or IP address in most cases. Keep in mind that the database user must have rights to use the database.

```
private static SqlConnection sqlConn = null;
private static string serverName = "(local)\\SQLExpress";
private static string userName = "sqluser";
private static string passwd = "abc123";
private static string database = "ozekisms";
private static string connectionString = "Server=" + serverName +";" +
                "Database=" + database + ";" +
                "User Id=" + userName + ";" +
                "Password=" + passwd + ";";
```

**Figure 2** - Shows how the connection string is built up in 'DatabaseHandling.cs'

# Run example project

**You can use Microsoft Visual Studio to start the C#.NET application. The downloaded example project is called 'Sending SMS through MS SQL Server'. Please run it.**

On the opening window, see the outgoing and incoming messages and click 'Compose message' (**Figure 1**), so you can write your own message. The outgoing and incoming messages can be refreshed from the database.

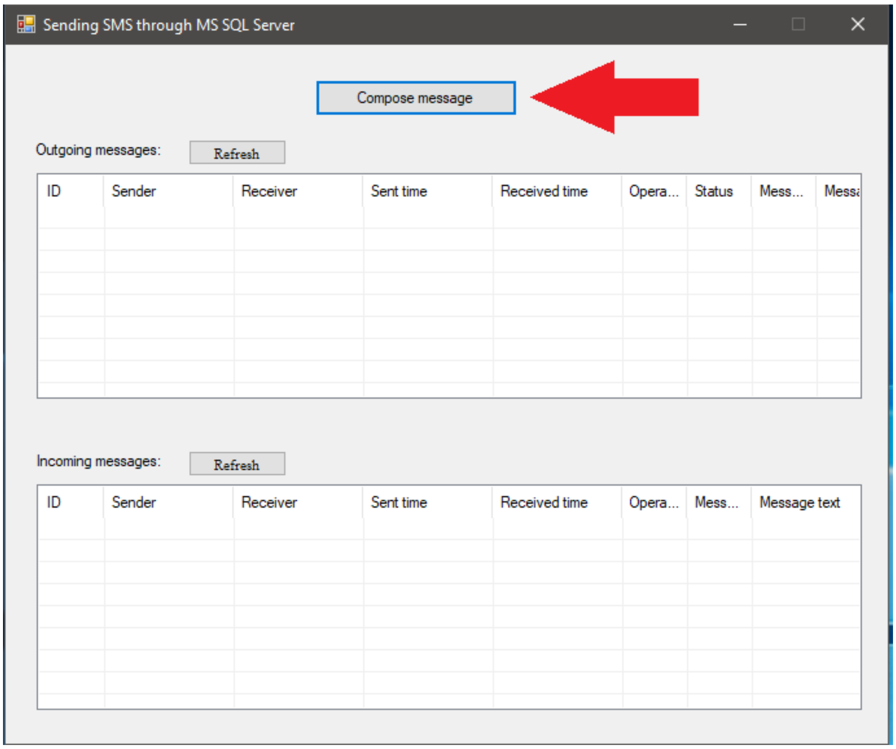
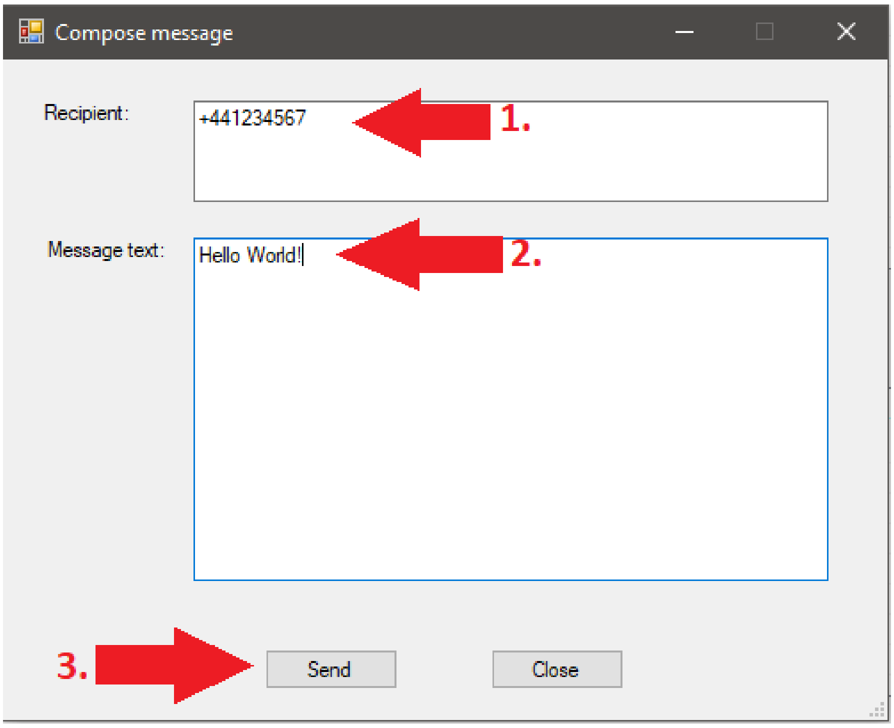


**Figure 1** - Run project and click 'Compose message'

Write message and click 'Send'. Do not forget to provide the recipient (**Figure 2**).



Recipient: +441234567

Message text: Hello World!

**Figure 2** - Write message and click 'Send'

# Use your VB.NET application for SMS messaging

**See how to send and receive SMS messages by using your Visual Basic.NET applications. It is a great choice to communicate with your clients or employees with simple text messages. Choose which is the best method for your system by downloading a database and a HTTP API example.**

In the 1st example the database server is between your SMS Gateway and VB.NET application. In the 2nd example the HTTP API of Ozeki SMS Gateway can be directly reached.

## Introduction

You can implement SMS functionalities into your Visual Basic.NET program, so it can send SMS messages through Ozeki SMS Gateway. By using a database server (MySQL, MSSQL, Oracle) you can send or receive SMS messages with SQL statements like INSERT or SELECT, but you can also use 'sendmessage' request of the HTTP API to post your message to the mobile network. These two options guarantee speed, reliability and safety to offer a professional and reliable solution. In the following paragraphs you can read some possibilities of the above mentioned benefits. You can find links to two detailed setup guides that include example source codes and detailed explanations.

## How to send and receive SMS in Visual Basic.NET using SQL

In this example your Visual Basic.NET application and the Ozeki SMS Gateway share a database server. To send an SMS message the Visual Basic.NET application must insert a database record into the database using an SQL request. Ozeki SMS gateway will read this database record and will send the SMS. Messages can be received the similar way. If an SMS comes in, the SMS gateway will insert it into the database. The VB application can pick it up from their through a standard database connection (**Figure 1**).

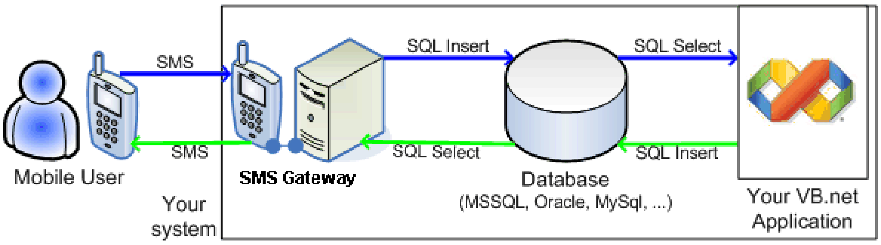


**Figure 1 -** How to send / receive SMS from VB.NET through SQL

Read more: How to send and receive SMS messages from Visual Basic.NET through a database server

## How to send and receive SMS in Visual Basic.NET using HTTP

This example explains how to use HTTP request in VB.NET to submit text-messages to mobile phones. After implementing this example, your Visual Basic.NET application will be able to deliver text messages to the cellphones of mobile users through Ozeki SMS Gateway, by calling it's HTTP interface. Ozeki SMS Gateway will be responsible for converting these HTTP requests to SMS messages and for delivering them to the recipient handsets (**Figure 2**).

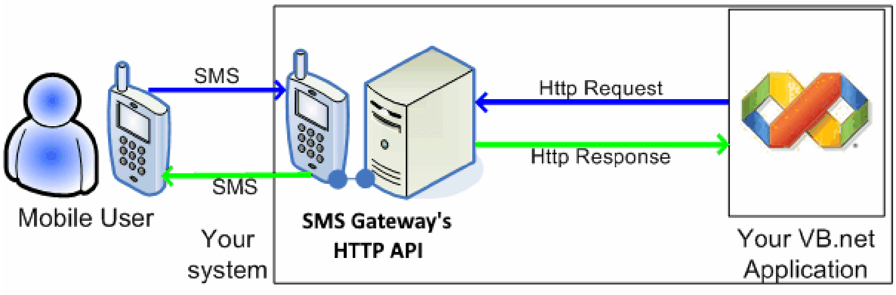


**Figure 2 -** How to send SMS from VB.NET using HTTP requests

Read more: Step by step guide that explains how to use HTTP to send SMS from VB.NET.

## Benefits

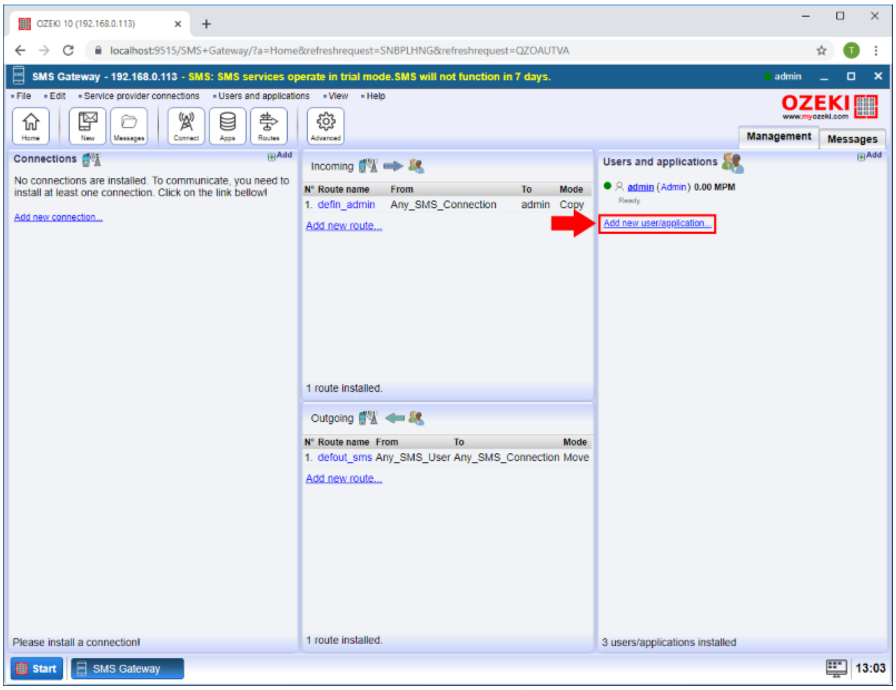
By implementing the above configurations with Ozeki SMS Gateway, several benefits will open for you. You will have a very reliable notification system, that can be configured to notify the intended person directly. You will be able the react instantly to incoming SMS messages and you will be able to track message delivery status using delivery reports. The above configurations can be used independently of mobile network operators. You can configure Ozeki SMS Gateway to use GSM modems or IP SMS connections to send your SMS messages. If one provider fails, or the provided throughput is not sufficient, you can switch to another provider. You can use HTTPS or secure database connection, thus you can build a secure notification system. By using this next generation, you will be able to provide a professional service for your mobile users. Your work will be more efficient, faster and nonetheless more accurate.

# VB.Net SMS Script

***In Ozeki SMS Gateway You can write a script, that allows you to run Visual Basic .NET code when an SMS message arrives. This script can implement any logic to do various tasks with the incoming messages, and optionally send response SMS messages.***

## Step 1 - Add new user/application...

You can simply install the Visual Basic script User on the 'Management' console by clicking 'Add new user/application...' in the 'Users/Applications' panel (Figure 1).



No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defin_admin | Any_SMS_Connection | admin | Copy |

Add new route...

1 route installed.

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move |

Add new route...

1 route installed.

admin (Admin) 0.00 MPM
Ready

Add new user/application...

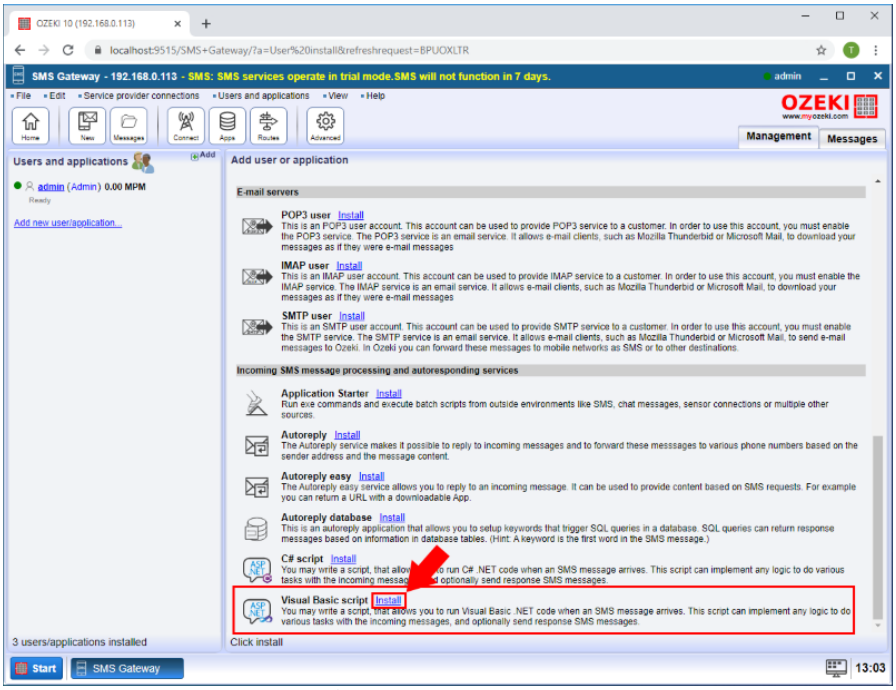
Please install a connection!

3 users/applications installed

**Figure 1** - Add new user/application...

## Step 2 - Add Visual Basic script

An interface will open consisting of two panels. The left side panel contains the already installed users and applications. The right side panel contains the users and applications you can install with a brief description next to them. Search the Visual Basic script User and click the blue 'install' button next to it (Figure 2).



OZEKI 10 (192.168.0.113)
localhost:9515/SMS+Gateway/?a=User%20install&refreshrequest=BPUOXLTR
SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
OZEKI
www.myozeki.com
Management
Messages
Users and applications
Add
admin (Admin) 0.00 MPM
Ready
Add new user/application...
3 users/applications installed
Start
SMS Gateway
13:03

Add user or application

**E-mail servers**

**POP3 user** Install
This is an POP3 user account. This account can be used to provide POP3 service to a customer. In order to use this account, you must enable the POP3 service. The POP3 service is an email service. It allows e-mail clients, such as Mozilla Thunderbid or Microsoft Mail, to download your messages as if they were e-mail messages

**IMAP user** Install
This is an IMAP user account. This account can be used to provide IMAP service to a customer. In order to use this account, you must enable the IMAP service. The IMAP service is an email service. It allows e-mail clients, such as Mozilla Thunderbid or Microsoft Mail, to download your messages as if they were e-mail messages

**SMTP user** Install
This is an SMTP user account. This account can be used to provide SMTP service to a customer. In order to use this account, you must enable the SMTP service. The SMTP service is an email service. It allows e-mail clients, such as Mozilla Thunderbid or Microsoft Mail, to send e-mail messages to Ozeki. In Ozeki you can forward these messages to mobile networks as SMS or to other destinations.

**Incoming SMS message processing and autoresponding services**

**Application Starter** Install
Run exe commands and execute batch scripts from outside environments like SMS, chat messages, sensor connections or multiple other sources.

**Autoreply** Install
The Autoreply service makes it possible to reply to incoming messages and to forward these messsages to various phone numbers based on the sender address and the message content.

**Autoreply easy** Install
The Autoreply easy service allows you to reply to an incoming message. It can be used to provide content based on SMS requests. For example you can return a URL with a downloadable App.

**Autoreply database** Install
This is an autoreply application that allows you to setup keywords that trigger SQL queries in a database. SQL queries can return response messages based on information in database tables. (Hint: A keyword is the first word in the SMS message.)

**C# script** Install
You may write a script, that allow o run C# .NET code when an SMS message arrives. This script can implement any logic to do various tasks with the incoming messag d optionally send response SMS messages.

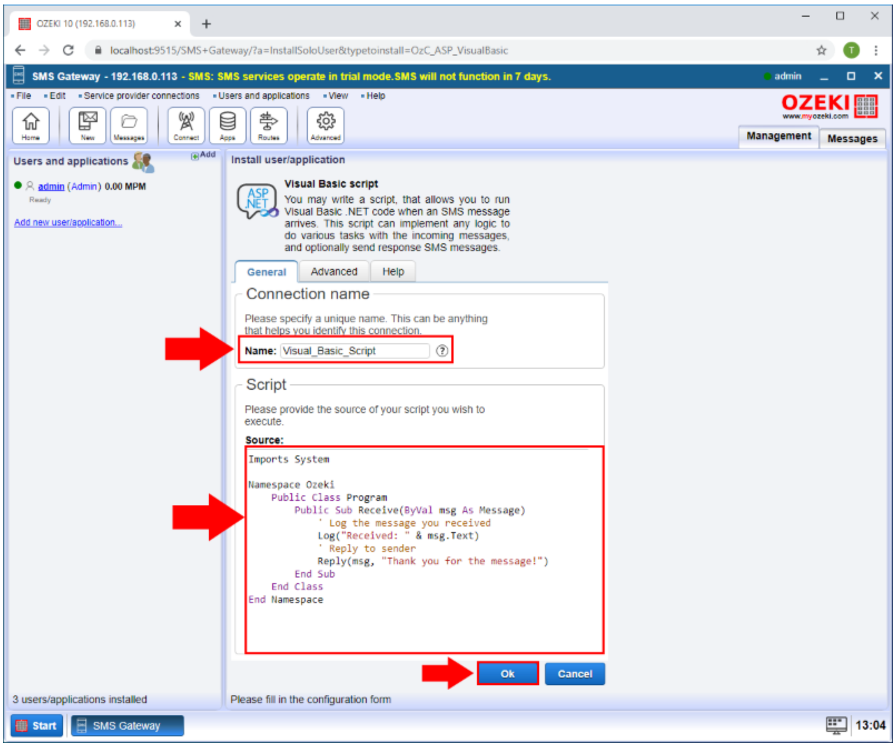
**Visual Basic script** Install
You may write a script, that allows you to run Visual Basic .NET code when an SMS message arrives. This script can implement any logic to do various tasks with the incoming messages, and optionally send response SMS messages.

Click install

**Figure 2** - Add Visual Basic script

## Step 3 - Provide the script

The 'Configuration' panel has a 'General' tab which contains the basic settings. First please provide a unique Name. In the Script section of tabpage contains the source of the script you wish to execute (Figure 3).



OZEKI 10 (192.168.0.113)
localhost:9515/SMS+Gateway/?a=InstallSoloUser&typetoinstall=OzC_ASP_VisualBasic
SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
OZEKI
www.myozeki.com
Management
Messages
Users and applications
Add
admin (Admin) 0.00 MPM
Ready
Add new user/application...
3 users/applications installed
Start
SMS Gateway
13:04
General
Advanced
Help
Ok
Cancel

Install user/application

**Visual Basic script**
You may write a script, that allows you to run Visual Basic .NET code when an SMS message arrives. This script can implement any logic to do various tasks with the incoming messages, and optionally send response SMS messages.

Connection name

Please specify a unique name. This can be anything that helps you identify this connection.

**Name:** Visual_Basic_Script

Script

Please provide the source of your script you wish to execute.

**Source:**

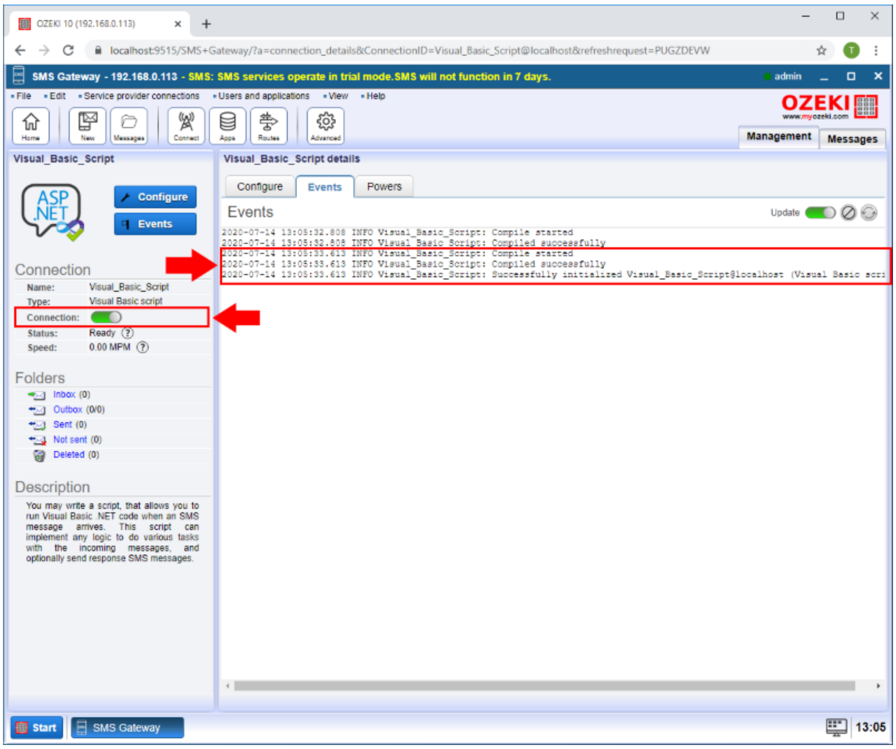
```
Imports System

Namespace Ozeki
    Public Class Program
        Public Sub Receive(ByVal msg As Message)
            ' Log the message you received
            Log("Received: " & msg.Text)
            ' Reply to sender
            Reply(msg, "Thank you for the message!")
        End Sub
    End Class
End Namespace
```

Please fill in the configuration form

**Figure 3** - Provide the script

## Step 4 - Enable user

Please enable te user with the Connection switch and in the Events tab view that the Script is compiled and the user is initialized successfully (Figure 4).



Name: Visual_Basic_Script
Type: Visual Basic script
Status: Ready
Speed: 0.00 MPM

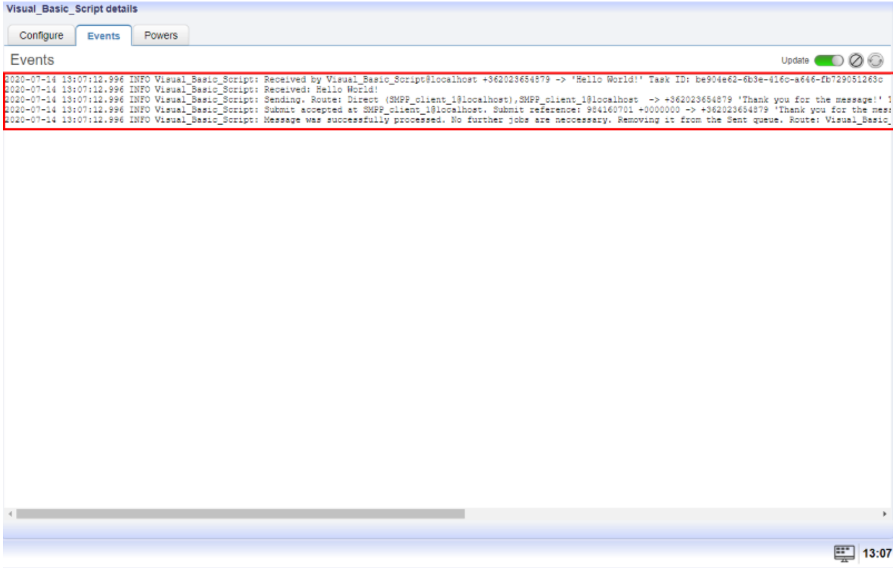
You may write a script, that allows you to run Visual Basic .NET code when an SMS message arrives. This script can implement any logic to do various tasks with the incoming messages, and optionally send response SMS messages.

```
2020-07-14 13:05:32.808 INFO Visual_Basic_Script: Compile started
2020-07-14 13:05:32.808 INFO Visual_Basic_Script: Compiled successfully
2020-07-14 13:05:33.613 INFO Visual_Basic_Script: Compile started
2020-07-14 13:05:33.613 INFO Visual_Basic_Script: Compiled successfully
2020-07-14 13:05:33.613 INFO Visual_Basic_Script: Successfully initialized Visual_Basic_Script@localhost (Visual Basic scri
```

**Figure 4** - Enable user

## Step 5 - Message received

Finally you can see if a message is received by this user the script will run and as this example works the response SMS message is sent to the Original sender (Figure 5).



```
2020-07-14 13:07:12.996 INFO Visual_Basic_Script: Received by Visual_Basic_Script@localhost +362023654879 -> 'Hello World!' Task ID: be904e62-6b3e-416c-a646-fb729051263c
2020-07-14 13:07:12.996 INFO Visual_Basic_Script: Received: Hello World!
2020-07-14 13:07:12.996 INFO Visual_Basic_Script: Sending. Route: Direct (SMPP_client_1@localhost),SMPP_client_1@localhost  -> +362023654879 'Thank you for the message!' 1
2020-07-14 13:07:12.996 INFO Visual_Basic_Script: Submit accepted at SMPP_client_1@localhost. Submit reference: 984160701 +0000000 -> +362023654879 'Thank you for the mes:
2020-07-14 13:07:12.996 INFO Visual_Basic_Script: Message was successfully processed. No further jobs are neccessary. Removing it from the Sent queue. Route: Visual_Basic_
```

**Figure 5** - Message received

## Visual Basic programming guide

Create a class named 'Program' in the 'Ozeki' namespace. You may also include the necessary using directives.

```
Imports System

Namespace Ozeki
Public Class Program
End Class
End Namespace
```

Implement the 'Receive' function inside the 'Program' class. This function will be called when your connection receives a message. The parameter of the 'Receive' function will be an object with the type of 'Message'.

```
         Imports System

        Namespace Ozeki
        Public Class Program
Public Sub Receive(ByVal msg As Message)
                End Sub
             End Class
         End Namespace
```

# Send a reply message

When receiving a message, you may send a reply to notify your sender of successful delivery. To achieve this, you may use the built-in 'Reply' function.

```
         Imports System

        Namespace Ozeki
        Public Class Program
Public Sub Receive(ByVal msg As Message)
  Reply(msg, "Thank you for the message!")
                End Sub
             End Class
         End Namespace
```

# Message forwarding

You may also forward your received message to an other connection by using the 'Send' function.

```
         Imports System

        Namespace Ozeki
        Public Class Program
Public Sub Receive(ByVal msg As Message)
     Send("admin@localhost", msg.Text)
                End Sub
             End Class
         End Namespace
```

# Compose your message

Composing a message is easier, than you think. You just need to use the built-in 'Message' type to create a new message object. In the following example, when a message was received, we will create and send a new message to the admin connection.

```
         Imports System

        Namespace Ozeki
        Public Class Program
Public Sub Receive(ByVal msg As Message)
        Dim message = New Message()
       message.Text = "Hello World!"
    message.ToAddress = "+4412345678910"
  message.ToConnection = "admin@localhost"
   message.FromAddress = "+448888999910"
               Send(message)
                End Sub
             End Class
End Namespace
```

# Logging

Using log messages will make debugging your script much easier. By calling the 'Log' function you may implement proper logging in your script if needed.

```
         Imports System

        Namespace Ozeki
        Public Class Program
Public Sub Receive(ByVal msg As Message)
    Log("Message received: " & msg.Text)
                End Sub
             End Class
         End Namespace
```

# VB.NET database SMS example

**See how to add SMS functionality to your VB.NET application. For the solution you will need a database server that stores sent and received messages. MS SQL, MS SQL Express, MySQL, Access, Oracle is accepted. The source code is provided. You can download and edit it.**

Download: vb.net-sms-example-sql.zip (70 Kb)

## Introduction

The solution consists of 3 different parts (**Figure 1**). **Ozeki SMS Gateway**, the **SQL database** and your **VB.NET application**. Check if Ozeki SMS Gateway is connected to the SQL database. You can find plenty of connection tutorials on the

**Figure 1 -** Send and receive SMS messages with VB.NET

The figure shows exactly what was described in the last paragraph. You can send SMS messages by inserting a new record by using the VB.NET application. Check the SQL table to see if new records were placed into it. These messages should be automatically sent. Check if the VB.net application works with the database both by reading or inserting records. Reading records are important since this is the way to read incoming messages.

## Prerequisites

The main software of the operation is Ozeki SMS Gateway which you can get to your Ozeki 10 application. This software can connect your computer to the mobile network as it can connect your PC to any of the following database server:

- send SMS from **Microsoft SQL Express**
- send SMS from **Access**
- send SMS from **MySQL**
- send SMS from **Oracle**
- send SMS from **Postgres**
- send SMS from **SAP SQL Anywhere**

You will also need a C# or Visual Basic development IDE, such as Microsoft Visual Studio. If you haven't done yet, please download example project. To sum it up, here is a full list of the required software:

- Ozeki SMS Gateway
- Database server (MS SQL, SQL Expres, Access, MySQL, Oracle, Postgres, SAP SQL Anywhere, etc)
- Microsot Visual Studio
- vb.net-sms-example-sql.zip (70 Kb)

## How to Install and Configure VB.NET

Use the following steps to create a working solution. In this example you will see a Microsoft SQL Express solution with database examples ready to copy-paste. Please download and install Ozeki 10 before going forward with these steps. Manually test if you can send and receive SMS message with Ozeki 10. If the test was successful, you will be able to send and receive SMS messages from the graphical user interface of the VB.net application. Check if it was sent by using the event logs of Ozeki 10.

If you are not using SQL Express, please copy the table creation code from one of these SQL server options.

### Step 1 - Create database user and tables

Please install Microsoft SQL Express and Microsoft Visual Studio so you can follow these steps. Create the database tables on your Microsoft SQL Express database server by copy-pasting **code 1**.

```
create database ozeki
GO

use ozeki
GO
```

```

CREATE TABLE ozekimessagein (
 id int IDENTITY (1,1),
 sender varchar(30),
 receiver varchar(30),
 msg varchar(160),
 senttime varchar(100),
 receivedtime varchar(100),
 operator varchar(30),
 msgtype varchar(30),
 reference varchar(30),
);

CREATE TABLE ozekimessageout (
 id int IDENTITY (1,1),
 sender varchar(30),
 receiver varchar(30),
 msg varchar(160),
 senttime varchar(100),
 receivedtime varchar(100),
 operator varchar(100),
 msgtype varchar(30),
 reference varchar(30),
 status varchar(30),
 errormsg varchar(250)
);
GO

sp_addLogin 'ozekiuser', 'ozekipass'
GO

sp_addsrvrolemember 'ozekiuser', 'sysadmin'
GO
```

**Code 1** - The code that creates the table structure

## Step 2 - Connect Ozeki SMS Gateway to your database

Now it is time to create a Database user in Ozeki 10. After the user has been created select 'SQL server' and copy-paste the following connection string (**Code 2**). Do not forget to use your own Database name, user ID and password.

```
Server=.\SQLEXPRESS;Database=ozeki;UID=ozekiuser;PWD=ozekipass;
```

**Code 2** - Connection string. *Change parameters if necessary*

## Step 3 - Modify the VB.NET code to connect with Ozeki 10

Open example project in Visual Studio and edit source code if necessary. In **code 3** you should provide the same connection parameters used above in **code 2**. The VB.NET parameters are added to 'myConnection' connection string (**Code 3**).

```
Dim dbUsername As String = "ozekiuser"
Dim dbPassword As String = "ozekipass"
Dim database As String = "ozeki"

Dim myConnection As New SqlConnection("Server=.\SQLEXPRESS;User ID=" _
& dbUsername _
& ";password=" _
& dbPassword _
& ";Database=" _
& database _
& ";Persist Security Info=True")
```

**Code 3** - Check if the connection parameters are correct

## Step 4 - Insert SMS record into database

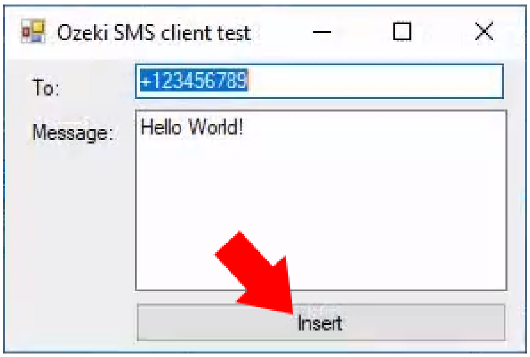
The following code inserts textbox content from the VB.NET example GUI to the database (**Code 4**).

```
Dim mySqlQuery As String = "INSERT INTO ozekimessageout (receiver,msg,status) " _
```

```
& "VALUES ('" & tbReceiver.Text & "', '" & tbMessage.Text & "', 'send');"

Dim mySqlCommand As New SqlCommand(mySqlQuery, myConnection)
```

**Code 4** - Inserts new SMS message record into the database

This code can only execute if the VB.NET application can connect to the database. The insertable values can be provided in the application GUI (**Figure 2**). After typing the recipient's number and message text, click 'Insert' and Ozeki 10's Database User will forward the inserted message to the recipient after finding the record in the database.



Ozeki SMS client test

To: +123456789

Message: Hello World!

**Figure 2 -** The user GUI of this VB.NET example

Finally the code sequence on the bottom of the class will run (**Code 5**). These 3 methods will be called. They will connect to the database and insert the message record of the SMS to send and close the connection.

```
myConnection.Open()

mySqlCommand.ExecuteNonQuery()

myConnection.Close()
```

**Code 5** - Inserts message record

# Full VB.NET example code

The full code you can see below (**Code 6**) is built up from segments described above (Code 3 - 5). You can freely use and modify the example code as you wish. The method can drop an exception if it cannot INSERT the SMS message to send.

```
Imports System
Imports System.Data
Imports System.Data.SqlClient

Public Class Form1

Private Sub bSend_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
    Handles bSend.Click
    Try
    Dim dbUsername As String = "ozekiuser"
    Dim dbPassword As String = "ozekipass"
    Dim database As String = "ozeki"

    Dim myConnection As New SqlConnection("Server=.\SQLEXPRESS;User ID=" _
        & dbUsername _
        & ";password=" _
        & dbPassword _
        & ";Database=" _
        & database _
        & ";Persist Security Info=True")


    Dim mySqlQuery As String = "INSERT INTO ozekimessageout (receiver,msg,status) " _
    & "VALUES ('" & tbReceiver.Text & "', '" & tbMessage.Text & "', 'send');"

    Dim mySqlCommand As New SqlCommand(mySqlQuery, myConnection)

    myConnection.Open()

    mySqlCommand.ExecuteNonQuery()

    myConnection.Close()


    Catch ex As Exception
        MessageBox.Show(ex.Message)
    End Try
End Sub
End Class
```

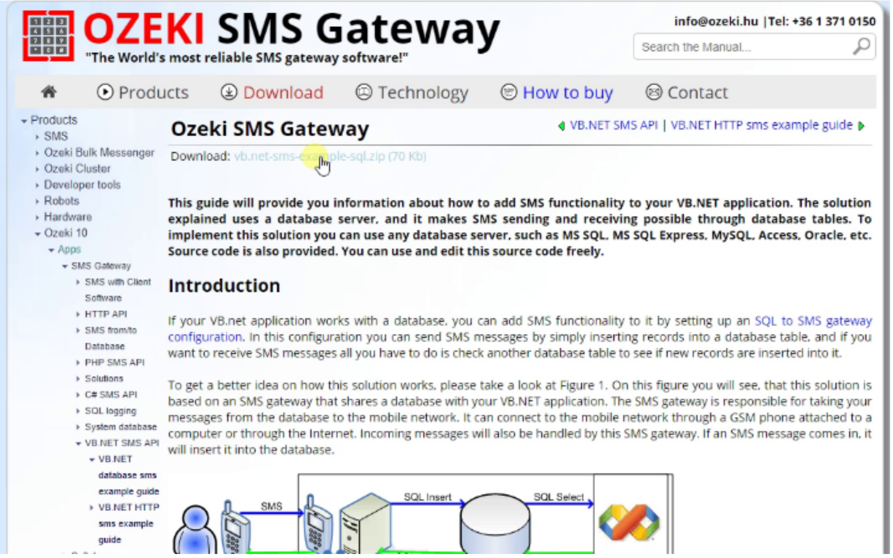
**Code 6** - The full example code

With this solution you won't have to worry about queing outgoing messages, since the queue will be handled by Ozeki 10's SMS Gateway application. It will also ad a timestamp and modify the message status register attribute to show successful or unsuccessful delivery towards the recipient.

# Incoming messages

They will be received and stored on the database server as well. Although the VB.NET application doesn't need to be online when receiving a message, since it can view the 'ozekimessagein' table anytime it goes online by running a SELECT statement over the table.

# Send SMS From VB.NET using Database



Ozeki SMS Gateway

Download: vb.net-sms-example-sql.zip (70 Kb)

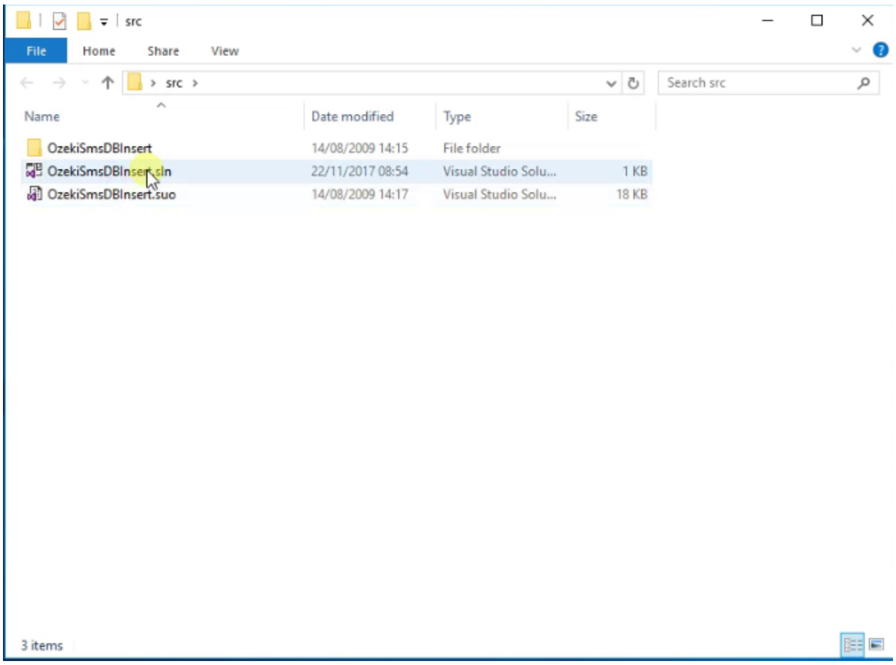
**This guide will provide you information about how to add SMS functionality to your VB.NET application. The solution explained uses a database server, and it makes SMS sending and receiving possible through database tables. To implement this solution you can use any database server, such as MS SQL, MS SQL Express, MySQL, Access, Oracle, etc. Source code is also provided. You can use and edit this source code freely.**

Introduction

If your VB.net application works with a database, you can add SMS functionality to it by setting up an SQL to SMS gateway configuration. In this configuration you can send SMS messages by simply inserting records into a database table, and if you want to receive SMS messages all you have to do is check another database table to see if new records are inserted into it.

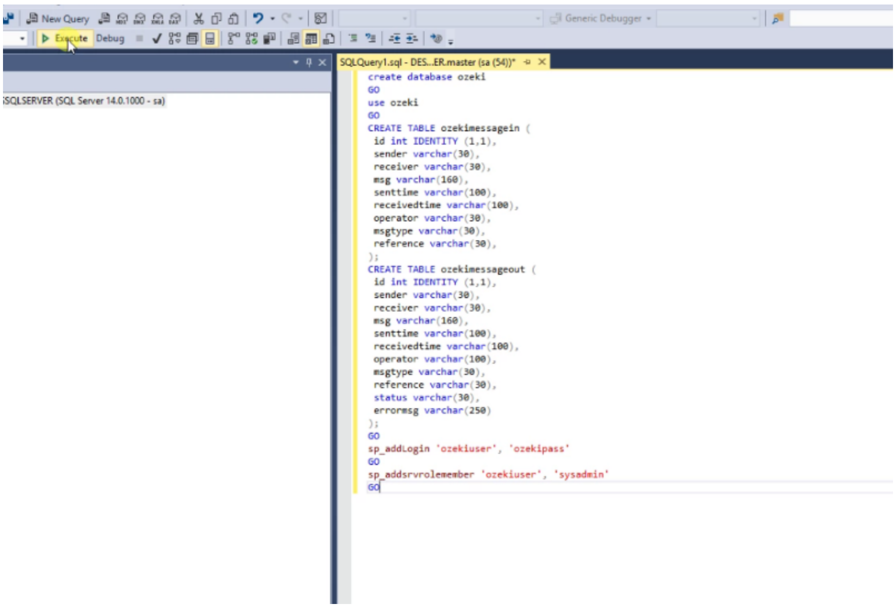
To get a better idea on how this solution works, please take a look at Figure 1. On this figure you will see, that this solution is based on an SMS gateway that shares a database with your VB.NET application. The SMS gateway is responsible for taking your messages from the database to the mobile network. It can connect to the mobile network through a GSM phone attached to a computer or through the Internet. Incoming messages will also be handled by this SMS gateway. If an SMS message comes in, it will insert it into the database.

**Figure 1** - Download example project



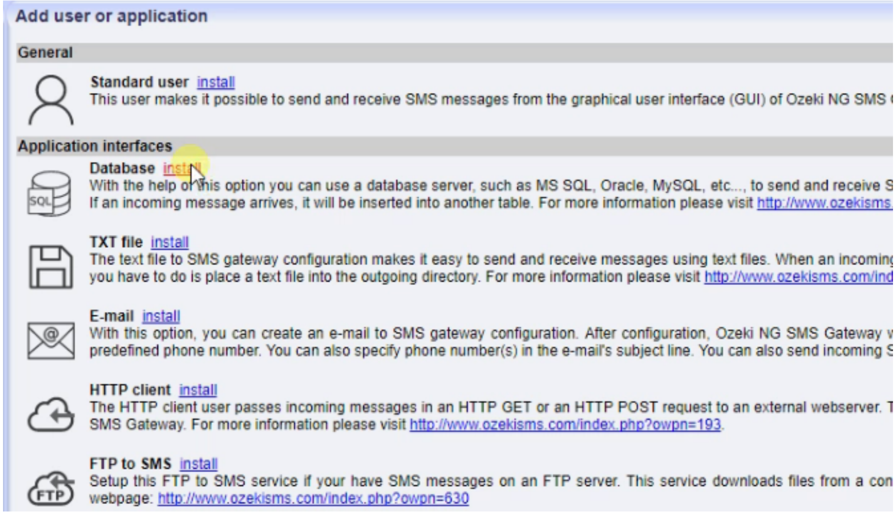
| Name | Date modified | Type | Size |
|---|---|---|---|
| OzekiSmsDBInsert | 14/08/2009 14:15 | File folder | |
| OzekiSmsDBInsert.sln | 22/11/2017 08:54 | Visual Studio Solu... | 1 KB |
| OzekiSmsDBInsert.suo | 14/08/2009 14:17 | Visual Studio Solu... | 18 KB |

**Figure 2** - Open the project file


New Query
Execute
Debug
Generic Debugger
SQLSERVER (SQL Server 14.0.1000 - sa)
SQLQuery1.sql - DES...ER.master (sa (54))*

```
create database ozeki
GO
use ozeki
GO
CREATE TABLE ozekimessagein (
 id int IDENTITY (1,1),
 sender varchar(30),
 receiver varchar(30),
 msg varchar(160),
 senttime varchar(100),
 receivedtime varchar(100),
 operator varchar(30),
 msgtype varchar(30),
 reference varchar(30),
);
CREATE TABLE ozekimessageout (
 id int IDENTITY (1,1),
 sender varchar(30),
 receiver varchar(30),
 msg varchar(160),
 senttime varchar(100),
 receivedtime varchar(100),
 operator varchar(100),
 msgtype varchar(30),
 reference varchar(30),
 status varchar(30),
 errormsg varchar(250)
);
GO
sp_addLogin 'ozekiuser', 'ozekipass'
GO
sp_addsrvrolemember 'ozekiuser', 'sysadmin'
GO
```

**Figure 3** - Create the database layout in MSSQL Server



Add user or application

General

Standard user install
This user makes it possible to send and receive SMS messages from the graphical user interface (GUI) of Ozeki NG SMS

Application interfaces

Database install
With the help of this option you can use a database server, such as MS SQL, Oracle, MySQL, etc..., to send and receive S
If an incoming message arrives, it will be inserted into another table. For more information please visit http://www.ozekisms

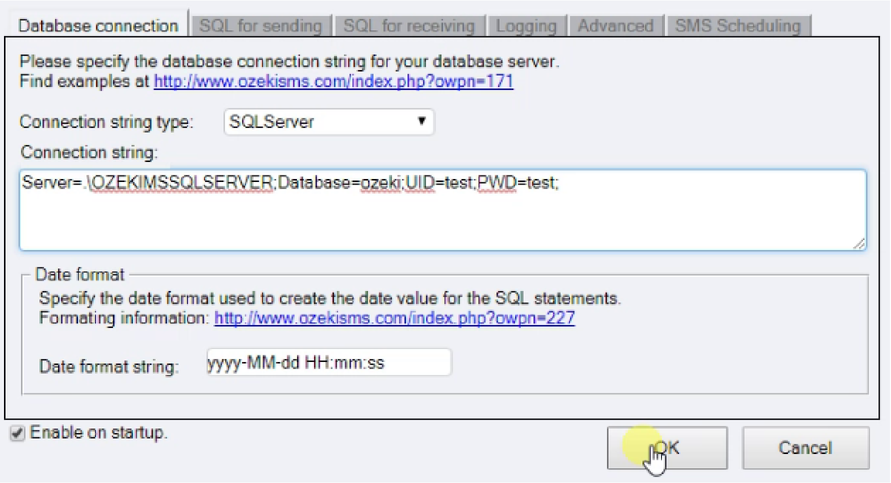
TXT file install
The text file to SMS gateway configuration makes it easy to send and receive messages using text files. When an incoming
you have to do is place a text file into the outgoing directory. For more information please visit http://www.ozekisms.com/ind

E-mail install
With this option, you can create an e-mail to SMS gateway configuration. After configuration, Ozeki NG SMS Gateway w
predefined phone number. You can also specify phone number(s) in the e-mail's subject line. You can also send incoming S

HTTP client install
The HTTP client user passes incoming messages in an HTTP GET or an HTTP POST request to an external webserver. T
SMS Gateway. For more information please visit http://www.ozekisms.com/index.php?owpn=193.

FTP to SMS install
Setup this FTP to SMS service if your have SMS messages on an FTP server. This service downloads files from a con
webpage: http://www.ozekisms.com/index.php?owpn=630

**Figure 4** - Create database user in Ozeki SMS Gateway



Please specify the database connection string for your database server.
Find examples at http://www.ozekisms.com/index.php?owpn=171

Connection string type: SQLServer

Connection string:

Server=.\OZEKIMSSQLSERVER;Database=ozeki;UID=test;PWD=test;

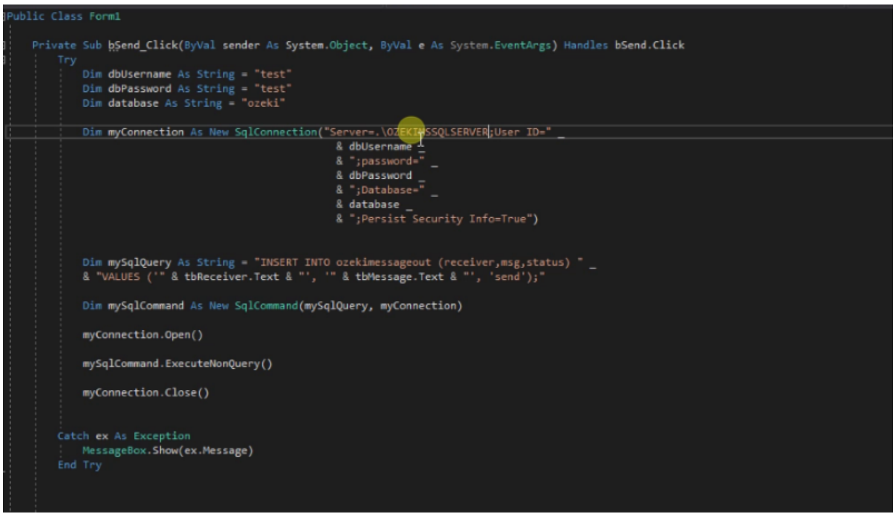
Date format

Specify the date format used to create the date value for the SQL statements.
Formating information: http://www.ozekisms.com/index.php?owpn=227

Date format string: yyyy-MM-dd HH:mm:ss

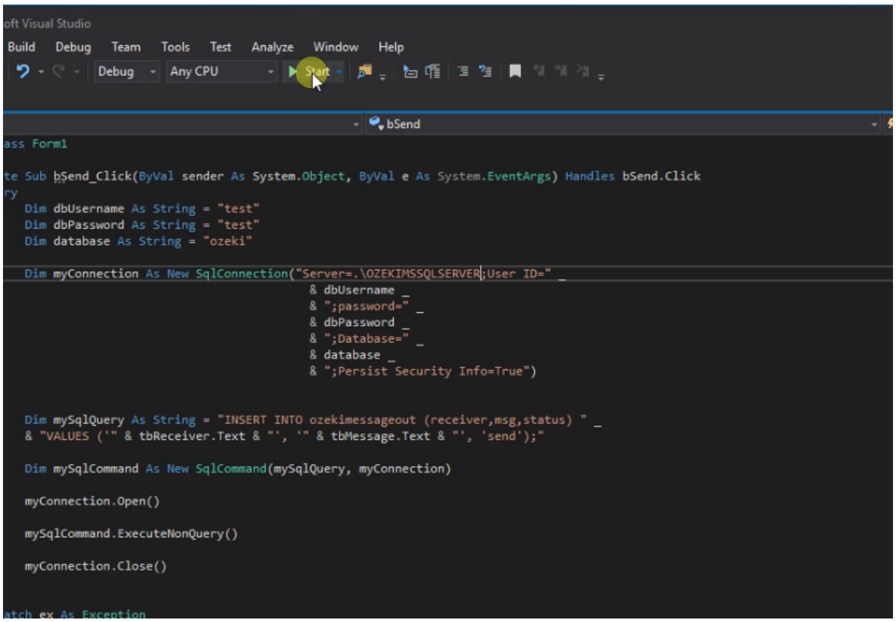
☑ Enable on startup.

**Figure 5** - Connect to the database



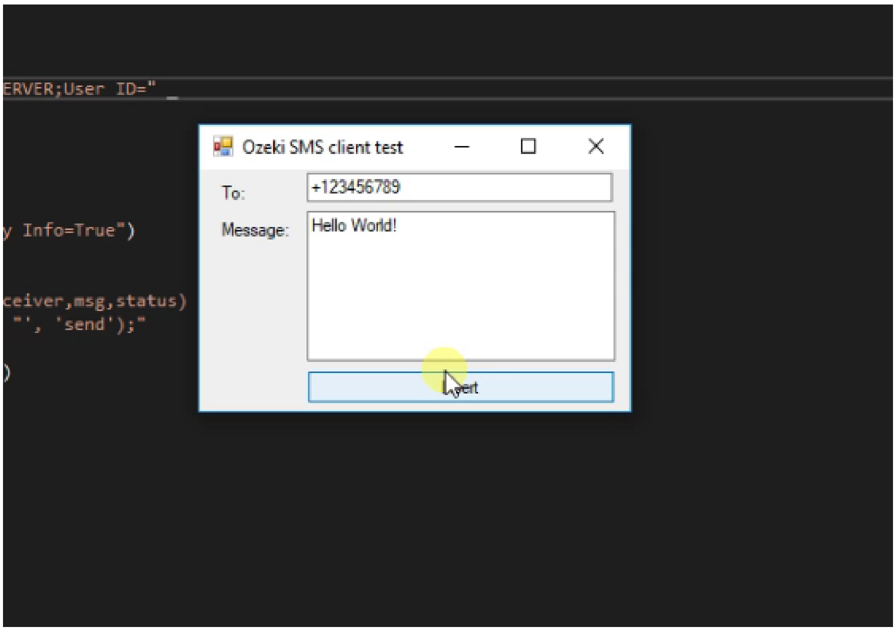
```
Public Class Form1

    Private Sub bSend_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles bSend.Click
        Try
            Dim dbUsername As String = "test"
            Dim dbPassword As String = "test"
            Dim database As String = "ozeki"

            Dim myConnection As New SqlConnection("Server=.\OZEKIMSSQLSERVER;User ID=" _
                                                  & dbUsername _
                                                  & ";password=" _
                                                  & dbPassword _
                                                  & ";Database=" _
                                                  & database _
                                                  & ";Persist Security Info=True")


            Dim mySqlQuery As String = "INSERT INTO ozekimessageout (receiver,msg,status) " _
            & "VALUES ('" & tbReceiver.Text & "', '" & tbMessage.Text & "', 'send');"

            Dim mySqlCommand As New SqlCommand(mySqlQuery, myConnection)

            myConnection.Open()

            mySqlCommand.ExecuteNonQuery()

            myConnection.Close()


        Catch ex As Exception
            MessageBox.Show(ex.Message)
        End Try
```

**Figure 6** - Modify the login creditals in the source code



```
ass Form1

te Sub bSend_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles bSend.Click
ry
   Dim dbUsername As String = "test"
   Dim dbPassword As String = "test"
   Dim database As String = "ozeki"

   Dim myConnection As New SqlConnection("Server=.\OZEKIMSSQLSERVER;User ID=" _
                                         & dbUsername _
                                         & ";password=" _
                                         & dbPassword _
                                         & ";Database=" _
                                         & database _
                                         & ";Persist Security Info=True")


   Dim mySqlQuery As String = "INSERT INTO ozekimessageout (receiver,msg,status) " _
   & "VALUES ('" & tbReceiver.Text & "', '" & tbMessage.Text & "', 'send');"

   Dim mySqlCommand As New SqlCommand(mySqlQuery, myConnection)

   myConnection.Open()

   mySqlCommand.ExecuteNonQuery()

   myConnection.Close()


atch ex As Exception
```

**Figure 7 -** Start the project



```
ERVER;User ID=" _

y Info=True")

ceiver,msg,status)
 "', 'send');"

)
```

To: +123456789

Message: Hello World!

**Figure 8 -** Send the Test SMS

# VB.NET HTTP SMS example

***See how to add SMS functionality to your VB.NET application. For the solution you will only need an Ozeki 10 and a Microsoft Visual Studio. The communication will work through the HTTP protocol. The source code is provided. You can download and edit it.***

Download: send-sms-vb.net-http-sms-example.zip (106 Kb)

## Introduction

VB.NET applications can be flexibly developed for creating dynamic web pages and standard applications. Developers are usually required to add mobile messaging to their applications, so users can send SMS messages to any recipient. Usually the most simplest and convenient way to add a SMS messaging to your system is by using a HTTP SMS gateway, such as Ozeki 10's SMS Gateway application and post messages to this gateway using HTTP requests (**Figure 1**).

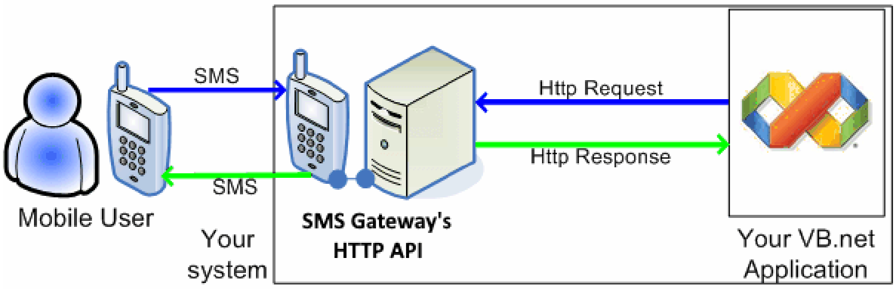


**Figure 1 -** Send and receive SMS messages with HTTP requests

Besides sending SMS messages, you can receive them too with VB.NET. You can do this in two ways. The easiest way is to create a HTTP Client User in Ozeki SMS Gateway, which can forward incoming messages to your VB.NET programcodes. This works if you are able to process HTTP requests. The other way is to periodically download incoming messages. You can download text messages with delivery reports from Ozeki SMS Gateway.

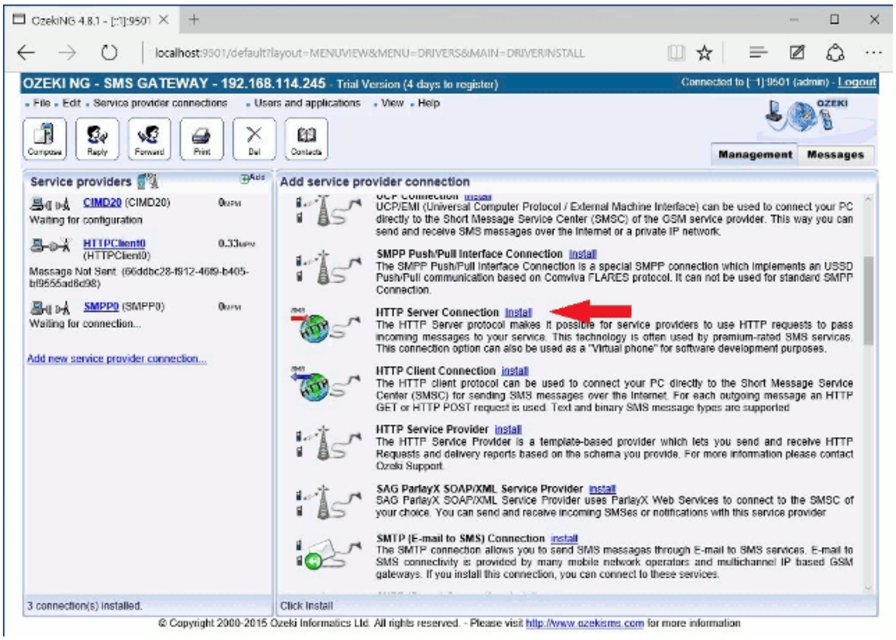
## Prerequisites

There are only two prerequisites you should download. Ozeki 10's SMS Gateway application and Microsoft Visual Studio for coding VB.NET. Both are the two ends of the HTTP communication. Besides Microsoft Visual Studio you can use any other IDE that let's you develop in VB.NET. If you scroll bellow you can see the VB.NET example codes for the HTTP application. Here you can find the software checklist and the example program:

- Download Ozeki SMS Gateway
- Microsoft Visual Studio
- send-sms-vb.net-http-sms-example.zip

### Set up Ozeki 10 and run the example code

You can easily download and install Ozeki 10. Check the SMS quick start guide to easily connect it to the mobile network. You will also need to create a HTTP Server Connection in Ozeki 10. Click 'Add new user or application...' and look for the HTTP Server Connection in the list. Click the blue 'Install' button next to it (**Figure 2**).



OzekiNG 4.8.1 - [::1]:9501
localhost:9501/default?layout=MENUVIEW&MENU=DRIVERS&MAIN=DRIVERINSTALL
OZEKI NG - SMS GATEWAY - 192.168.114.245 - Trial Version (4 days to register)
Connected to [::1]:9501 (admin) - Logout
File
Edit
Service provider connections
Users and applications
View
Help
Compose
Reply
Forward
Print
Del
Contacts
Management
Messages
Service providers
Add
Add service provider connection

CIMD20 (CIMD20) 0UPM
Waiting for configuration

HTTPClient0 (HTTPClient0) 0.33UPM
Message Not Sent. (66ddbc28-f812-4689-b405-bf9555ad8d98)

SMPP0 (SMPP0) 0UPM
Waiting for connection...

Add new service provider connection...

UCP Connection Install
UCP/EMI (Universal Computer Protocol / External Machine Interface) can be used to connect your PC directly to the Short Message Service Center (SMSC) of the GSM service provider. This way you can send and receive SMS messages over the Internet or a private IP network.

SMPP Push/Pull Interface Connection Install
The SMPP Push/Pull Interface Connection is a special SMPP connection which implements an USSD Push/Pull communication based on Comviva FLARES protocol. It can not be used for standard SMPP Connection.

HTTP Server Connection Install
The HTTP Server protocol makes it possible for service providers to use HTTP requests to pass incoming messages to your service. This technology is often used by premium-rated SMS services. This connection option can also be used as a "Virtual phone" for software development purposes.

HTTP Client Connection install
The HTTP client protocol can be used to connect your PC directly to the Short Message Service Center (SMSC) for sending SMS messages over the Internet. For each outgoing message an HTTP GET or HTTP POST request is used. Text and binary SMS message types are supported

HTTP Service Provider install
The HTTP Service Provider is a template-based provider which lets you send and receive HTTP Requests and delivery reports based on the schema you provide. For more information please contact Ozeki Support.

SAG ParlayX SOAP/XML Service Provider install
SAG ParlayX SOAP/XML Service Provider uses ParlayX Web Services to connect to the SMSC of your choice. You can send and receive incoming SMSes or notifications with this service provider

SMTP (E-mail to SMS) Connection install
The SMTP connection allows you to send SMS messages through E-mail to SMS services. E-mail to SMS connectivity is provided by many mobile network operators and multichannel IP based GSM gateways. If you install this connection, you can connect to these services.

3 connection(s) installed.

Click Install

© Copyright 2000-2015 Ozeki Informatics Ltd. All rights reserved. - Please visit http://www.ozekisms.com for more information

**Figure 2** - Installing a HTTP Server Connection

You will need to provide the username and password. Do not forget these login credentials. You will need to provide them in VB.NET by replacing the example strings in the source code.

# Step 1 - Set up the connection parameters in the VB.NET example code

Open the VB.NET example project and rewrite the necessary parameters (**Code 1**).

```
Dim request As HttpWebRequest

Dim response As HttpWebResponse = Nothing
Dim url As String
Dim username As String = "admin"
Dim password As String = "abc123"
Dim host As String = "http://127.0.0.1:9501"
Dim originator As String = "06201234567"
```

**Code 1** - HTTP parameters

The 'username' and 'password' strings are the login credentials belonging to the HTTP Server Connection. If Ozeki 10 and your VB.NET application is running on different machines then it is required to rewrite the 'host' parameter to the IP address and port number of the Ozek 10 machine. The 'originator' is the phone number used as the sender.

# Step 2 - Compose URL from parameters

**Code 2** composes a HTTP request from the parameters provided in **Code 1**. This URL will be used to post your SMS message to Ozeki 10's SMS Gateway application. All parameters specified in the Ozeki HTTP SMS API documentation can be contained in the URL. The values must be URL encoded, so special characters can not break the HTTP specification.

```
url = host + "/api?action=sendmessage&" _
& "username=" & HttpUtility.UrlEncode(username) _
& "&password=" + HttpUtility.UrlEncode(password) _
& "&recipient=" + HttpUtility.UrlEncode(tbReceiver.Text) _
& "&messagetype=SMS:TEXT" _
& "&messagedata=" + HttpUtility.UrlEncode(tbMessage.Text) _
& "&originator=" + HttpUtility.UrlEncode(originator) _
& "&serviceprovider=" _
& "&responseformat=html"
```

**Code 2** - HTTP request URL created from the parameters

# Step 3 - Submit URL to Ozeki 10

**Code 3** contains the last three lines of this VB.NET example. It sends the HTTP request and shows the response in a pop-up window. The WebRequest.Create("...") built in VB.NET method sends the URL to Ozeki 10, while GetResponse() method collects the response. It will appear in a pop-up window generated with the MessageBox.Show("...") method.

```
request = DirectCast(WebRequest.Create(url), HttpWebRequest)

response = DirectCast(request.GetResponse(), HttpWebResponse)

MessageBox.Show("Response: " & response.StatusDescription)
```

**Code 3** - Send HTTP request and show response in pop-up window

It is advised to walk through the HTTP API actions and the corresponding parameters to control Ozeki 10 through the HTTP API. This way you can create request URLs for your needs.

# Full VB.NET example code

The full code you can see below (**Code 4**) is built up from segments described above (Code 1 - 3). You can freely use and modify the example code as you wish. An exception can be dropped if it is a problem with the HTTP communication.

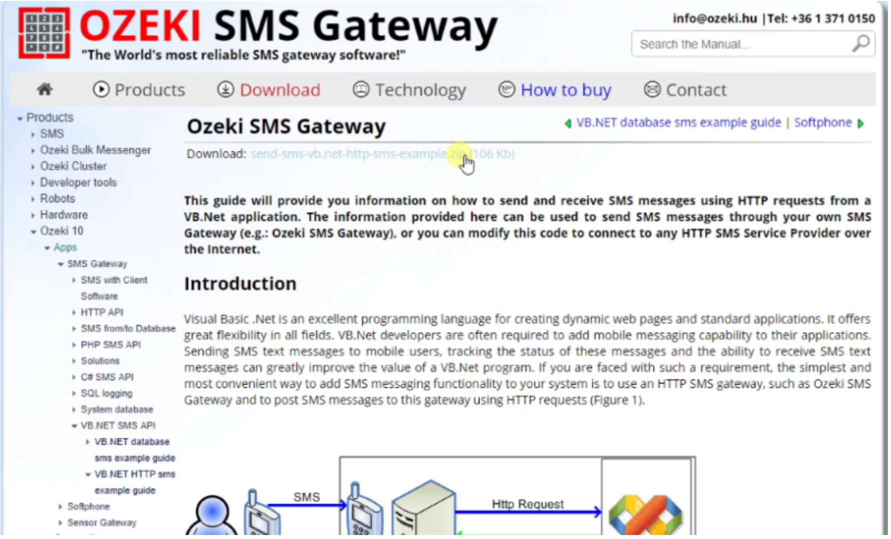
```
Imports System
Imports System.IO
Imports System.Net
Imports System.Text
Imports System.Web

Public Class fMain

Private Sub bSend_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles bSend.Click
    Dim request As HttpWebRequest
    Dim response As HttpWebResponse = Nothing
    Dim url As String
    Dim username As String
    Dim password As String
    Dim host As String
    Dim originator As String

    Try

        host = "http://127.0.0.1:9501"
        originator = "06201234567"
        username = "admin"
        password = "abc123"

        url = host + "/api?action=sendmessage&" _
                 & "username=" & HttpUtility.UrlEncode(username) _
                 & "&password=" + HttpUtility.UrlEncode(password) _
                 & "&recipient=" + HttpUtility.UrlEncode(tbReceiver.Text) _
                 & "&messagetype=SMS:TEXT" _
                 & "&messagedata=" + HttpUtility.UrlEncode(tbMessage.Text) _
                 & "&originator=" + HttpUtility.UrlEncode(originator) _
                 & "&serviceprovider=GSMModem1" _
                 & "&responseformat=html"

        request = DirectCast(WebRequest.Create(url), HttpWebRequest)

        response = DirectCast(request.GetResponse(), HttpWebResponse)

        MessageBox.Show("Response: " & response.StatusDescription)

    Catch ex As Exception
        MessageBox.Show(ex.Message)

    End Try
End Sub
End Class
```

**Code 4** - The full example code

# Send SMS From VB.NET using HTTP API

## Step 1 - Download the VB.NET example code

You can download send-sms-vb.net-http-sms-example.zip (106 Kb)



### Ozeki SMS Gateway

Download: send-sms-vb.net-http-sms-example.zip (106 Kb)

**This guide will provide you information on how to send and receive SMS messages using HTTP requests from a VB.Net application. The information provided here can be used to send SMS messages through your own SMS Gateway (e.g.: Ozeki SMS Gateway), or you can modify this code to connect to any HTTP SMS Service Provider over the Internet.**

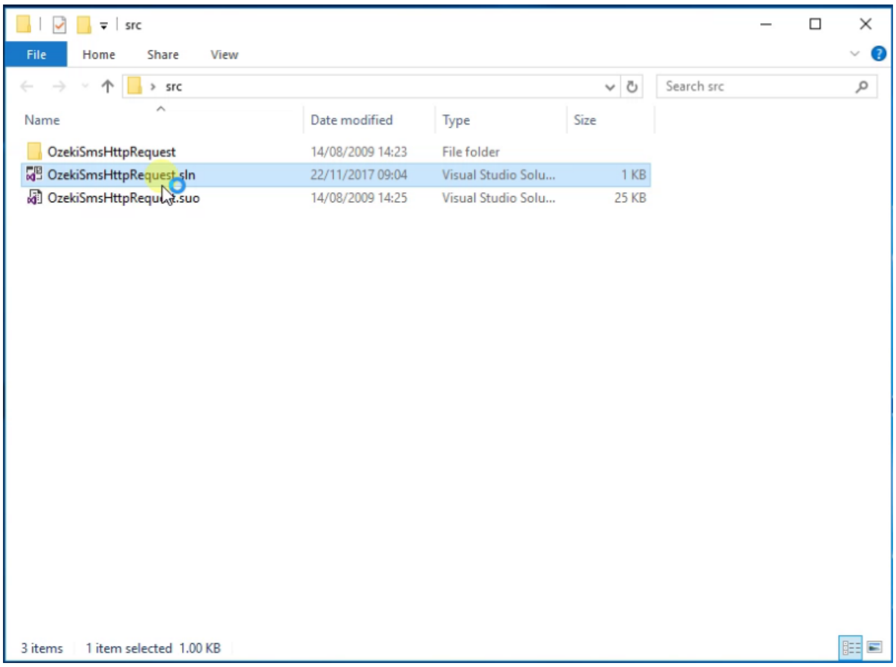
### Introduction

Visual Basic .Net is an excellent programming language for creating dynamic web pages and standard applications. It offers great flexibility in all fields. VB.Net developers are often required to add mobile messaging capability to their applications. Sending SMS text messages to mobile users, tracking the status of these messages and the ability to receive SMS text messages can greatly improve the value of a VB.Net program. If you are faced with such a requirement, the simplest and most convenient way to add SMS messaging functionality to your system is to use an HTTP SMS gateway, such as Ozeki SMS Gateway and to post SMS messages to this gateway using HTTP requests (Figure 1).

**Figure 1** - Download example project

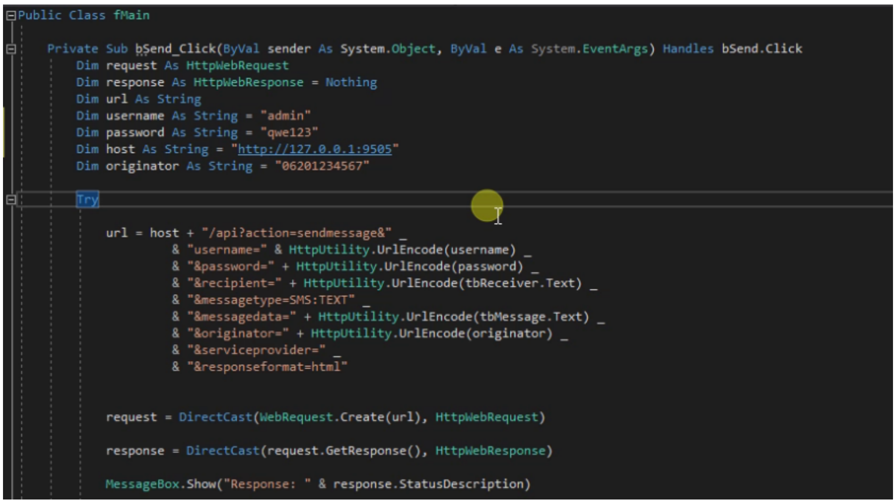
## Step 2 - Open the VB.NET project file

Click on the OzekiSMSHttpReuest.sln file in the src directory.



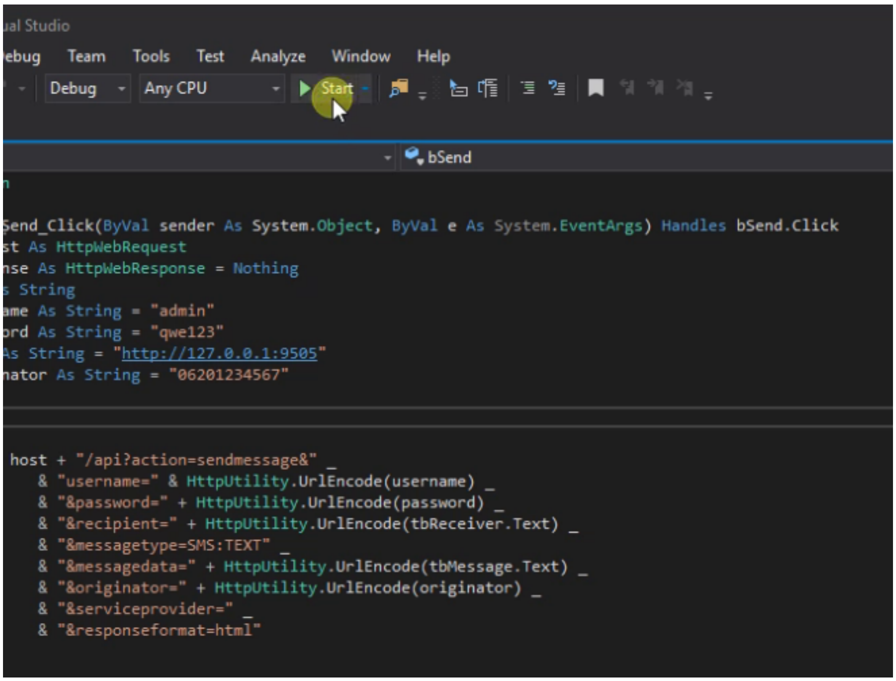
| Name | Date modified | Type | Size |
|---|---|---|---|
| OzekiSmsHttpRequest | 14/08/2009 14:23 | File folder | |
| OzekiSmsHttpRequest.sln | 22/11/2017 09:04 | Visual Studio Solu... | 1 KB |
| OzekiSmsHttpRequest.suo | 14/08/2009 14:25 | Visual Studio Solu... | 25 KB |

**Figure 2** - Open the project file



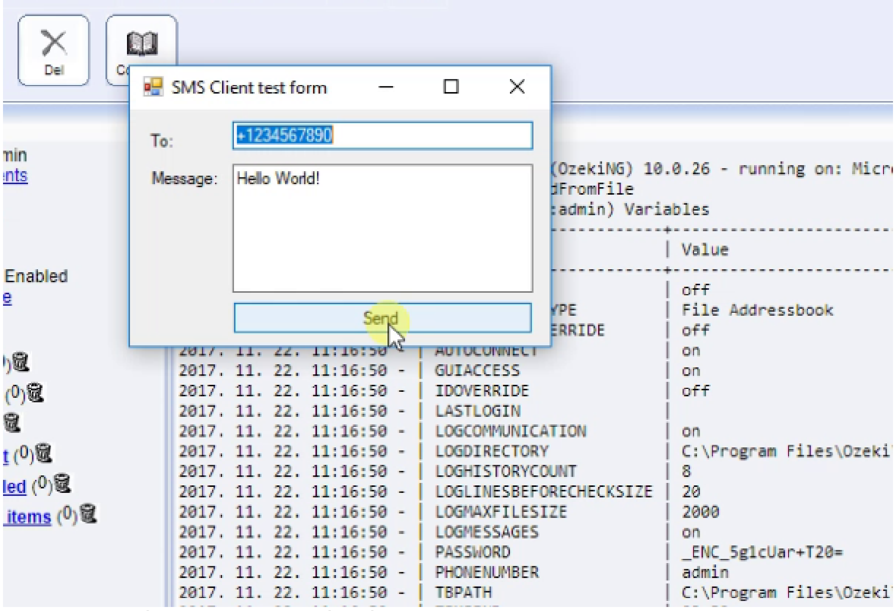
```
Public Class fMain

    Private Sub bSend_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles bSend.Click
        Dim request As HttpWebRequest
        Dim response As HttpWebResponse = Nothing
        Dim url As String
        Dim username As String = "admin"
        Dim password As String = "qwe123"
        Dim host As String = "http://127.0.0.1:9505"
        Dim originator As String = "06201234567"

        Try

            url = host + "/api?action=sendmessage&" _
                     & "username=" & HttpUtility.UrlEncode(username) _
                     & "&password=" + HttpUtility.UrlEncode(password) _
                     & "&recipient=" + HttpUtility.UrlEncode(tbReceiver.Text) _
                     & "&messagetype=SMS:TEXT" _
                     & "&messagedata=" + HttpUtility.UrlEncode(tbMessage.Text) _
                     & "&originator=" + HttpUtility.UrlEncode(originator) _
                     & "&serviceprovider=" _
                     & "&responseformat=html"


            request = DirectCast(WebRequest.Create(url), HttpWebRequest)

            response = DirectCast(request.GetResponse(), HttpWebResponse)

            MessageBox.Show("Response: " & response.StatusDescription)
```

**Figure 3 -** Modify the login creditals in the source code



```
end_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles bSend.Click
st As HttpWebRequest
nse As HttpWebResponse = Nothing
s String
ame As String = "admin"
ord As String = "qwe123"
As String = "http://127.0.0.1:9505"
nator As String = "06201234567"


 host + "/api?action=sendmessage&" _
    & "username=" & HttpUtility.UrlEncode(username) _
    & "&password=" + HttpUtility.UrlEncode(password) _
    & "&recipient=" + HttpUtility.UrlEncode(tbReceiver.Text) _
    & "&messagetype=SMS:TEXT" _
    & "&messagedata=" + HttpUtility.UrlEncode(tbMessage.Text) _
    & "&originator=" + HttpUtility.UrlEncode(originator) _
    & "&serviceprovider=" _
    & "&responseformat=html"
```

**Figure 4 -** Start the project



To: +1234567890

Message: Hello World!

```
(OzekiNG) 10.0.26 - running on: Micr
dFromFile
:admin) Variables
-------------+---------------------
             | Value
-------------+---------------------
             | off
YPE          | File Addressbook
ERRIDE       | off
2017. 11. 22. 11:16:50 - | AUTOCONNECT             | on
2017. 11. 22. 11:16:50 - | GUIACCESS               | on
2017. 11. 22. 11:16:50 - | IDOVERRIDE              | off
2017. 11. 22. 11:16:50 - | LASTLOGIN               |
2017. 11. 22. 11:16:50 - | LOGCOMMUNICATION        | on
2017. 11. 22. 11:16:50 - | LOGDIRECTORY            | C:\Program Files\Ozeki
2017. 11. 22. 11:16:50 - | LOGHISTORYCOUNT         | 8
2017. 11. 22. 11:16:50 - | LOGLINESBEFORECHECKSIZE | 20
2017. 11. 22. 11:16:50 - | LOGMAXFILESIZE          | 2000
2017. 11. 22. 11:16:50 - | LOGMESSAGES             | on
2017. 11. 22. 11:16:50 - | PASSWORD                | _ENC_5g1cUar+T20=
2017. 11. 22. 11:16:50 - | PHONENUMBER             | admin
2017. 11. 22. 11:16:50 - | TBPATH                  | C:\Program Files\Ozeki
```

**Figure 5 -** Send the Test SMS

# Service providers

**This guide gives you information about how to setup an SMPP service provider system. You will learn how to configure an SMPP SMS service, how to make it secure using SSL/TLS and certificates, and you will learn about SQL reporting, that allows you to keep track of the traffic going through your system.**

### SMPP Client

An SMPP Client connection is a connection you use to connect your system to Mobile Network Operators (MNOs) or independendt SMS Service Providers over the Internet. This is the connection you need to send your SMS messages to the mobile network. This connection uses the SMPP protocol which was invented for delivering short messages. An SMPP client connection can operate over a secure network link (TLS/TCP/IP), or over an standard unencrypted link (TCP/IP).

Learn More

### How to setup an SMPP Server

An SMPP Server connection is something you want to setup if you wish to provide SMS service to your customers. If you are a mobile network operator, or an independent SMS service provider, this is the connection you need to configure. You will have to setup the connection, than you have to create user accounts (login name and password), that you can distribute to your customers.

Learn More

### How to setup SQL reporting

SQL reporting uses a database server to record all messages going through the system. You can use MS SQL, Oracle, MySQL and any other database system that provides connectivity on Windows. When you configure SQL reporting the Ozeki SMS Gateway software will insert a record when a message passes through the system, and it will update this record with time stamps, delivery referencdes and status information as the message data changes.

Learn More

### How to setup SMS routing

When you operate an SMS service, sooner or later you will connect to multiple mobile networks. You will use various IP SMS protocols, such as SMPP, CIMD2 or UCP, you may use HTTP Client connections and you will probably setup a backup wireless link to make sure your system operates when the network connection fails. The SMS routing capability of Ozeki 10 SMS gateway allows you to control how these mobile network connections are used.

Learn More

### Custom SMS sender IDs

If you provide an SMS service you want to have control over the Sender ID of outgoing messages. For example you may want to assign a phone number to each user in your system, or you may want to setup a pool of phone numbers and instruct the system to pick a sender ID randomly from this pool for each outgoing SMS message. Ozeki SMS Gateway gives you total control over SMS Sender IDs.

Learn More

### Detailed logs

If you provide an SMS service it is important to be able to lookup what happens on a certain connection or what happens with a certain SMS messages. Ozeki 10 SMS gateway offers you detailed logs for connections and messages. For connections you can log detailed protocol communication, for SMS messages you will see which connection is used for message submission and when deleviry reports arrive.

Learn More

### Performance tuning

The floowing scetion provides you some really useful information about how you can tune the performance of your SMPP service. With these performance settings, you can set a limit of the rate for the incoming messages or you can set a speed limit at passing messages to a certain SMS service provider.

Learn More

### User authentication

Here, you can learn about how to give your users right to access SMS Gateway. By using the Authentication Provider connections in SMS Gateway you are allowed to connect to a database or a webserver where you can get the list of users and use them for authentication purposes in SMS Gateway.

Learn More

# How to setup an OZX service

# SMPP client connection

***This guide is about SMPP client connection setup. It defines what an SMPP connection is, what does SMPP stand for and how you can use an SMPP client connection to connect your Ozeki SMS gateway system directly to the Short Message Service Center (SMSC) of a mobile network operator over the Internet.***

## What is an SMPP connection?

An SMPP client is a software that allows you to connect to an SMS service provider on the Internet. An SMPP client, such as Ozeki SMS Gateway, uses the SMPP protocol to send and receive SMS text messages.

## What does SMPP stand for?

SMPP stands for Short Message Peer-to-Peer Protocol. This is an industry standard protocol designed to deliver SMS messages over TCP/IP connections through the Internet. This protocol is implemented by Ozeki SMS Gateway.

## How to connect an SMPP connection

The following short video shows the steps you need to create to setup an SMPP client connection in Ozeki 10 SMS Gateway. The video starts with the login form, and takes you all the way to sending your first SMS test message over the newly created SMPP connection.

## Connection steps

- Open https://localhost:9515 in your browser
- Login using your username and password
- Click on "Add new connection"
- Select "SMPP client"
- Enter the smpp host name and port
- Enter the telephone number
- Click ok and send a test message

## Detailed setup instructions

Setting up an SMS connection in Ozeki SMS Gateway is a relatively simple procedure. You need to login to the SMS gateway using a web browser as administrator, and you need to perform a few simple steps. For configuration we recommend to login using the administrator account. The administrator account username is "admin", and the password is the one you provided during install.

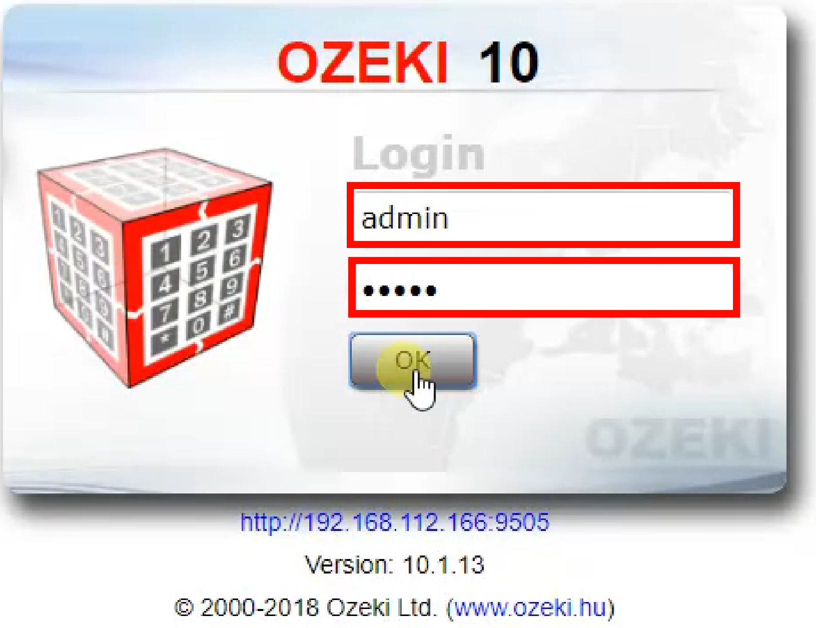
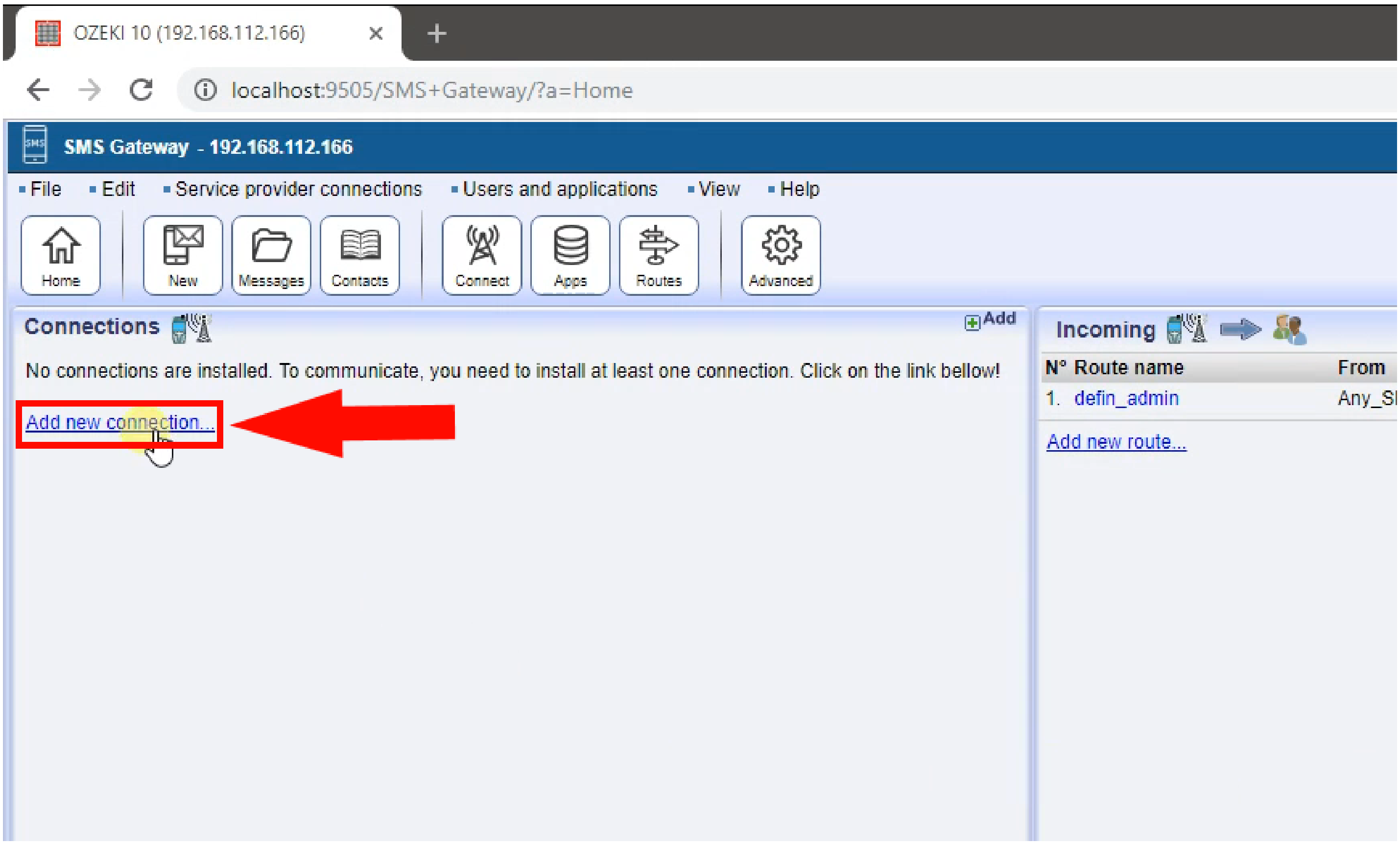

OZEKI 10
Login
admin
•••••
OK
OZEKI
http://192.168.112.166:9505
Version: 10.1.13
© 2000-2018 Ozeki Ltd. (www.ozeki.hu)

Figure 1 - Logging in to Ozeki 10 for SMPP client connection setup

# How to create a new SMPP connection

To create a new SMPP connection after login, you need to click on the '**Add new connection**' link in the management console of the Ozeki 10 SMS Gateway app. This will bring up a list of available protocols. You will have to select SMPP client from the list. Note, that an **SMPP client** connection is used if you wish to connect your SMS gateway to an SMS service provider over the Internet. (If you wish to provide an SMS service, and you want your customers to connect to your SMS gateway over SMPP, you need to setup an SMPP user account and you need to configure an SMPP service.)

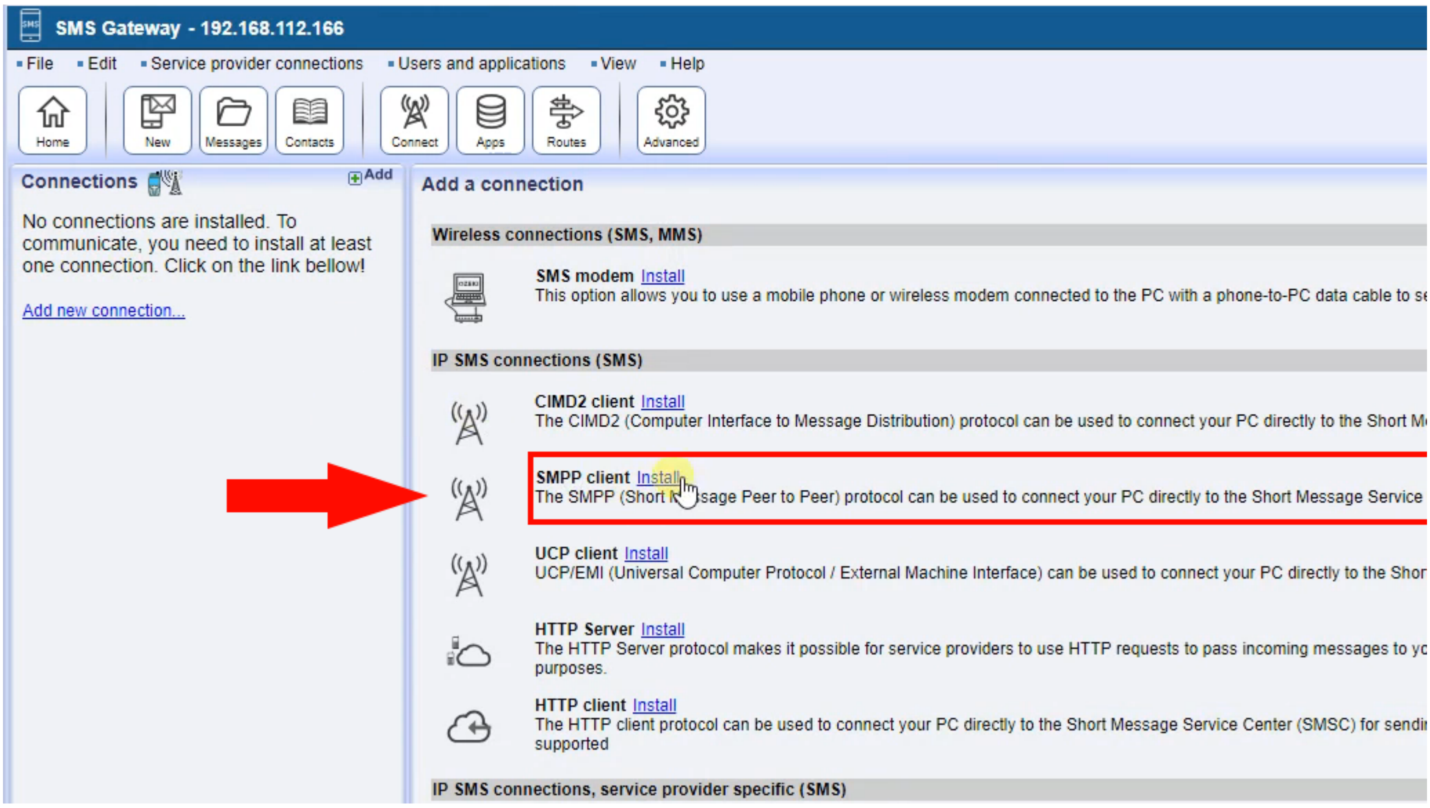


No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

Figure 2 - Create a new SMPP connection

Select the '**SMPP Client Connection**' from the list and click on install next to it.



No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

Wireless connections (SMS, MMS)

SMS modem Install
This option allows you to use a mobile phone or wireless modem connected to the PC with a phone-to-PC data cable to se

IP SMS connections (SMS)

CIMD2 client Install
The CIMD2 (Computer Interface to Message Distribution) protocol can be used to connect your PC directly to the Short M

SMPP client Install
The SMPP (Short M sage Peer to Peer) protocol can be used to connect your PC directly to the Short Message Service

UCP client Install
UCP/EMI (Universal Computer Protocol / External Machine Interface) can be used to connect your PC directly to the Shor

HTTP Server Install
The HTTP Server protocol makes it possible for service providers to use HTTP requests to pass incoming messages to yo purposes.

HTTP client Install
The HTTP client protocol can be used to connect your PC directly to the Short Message Service Center (SMSC) for sendi supported

IP SMS connections, service provider specific (SMS)

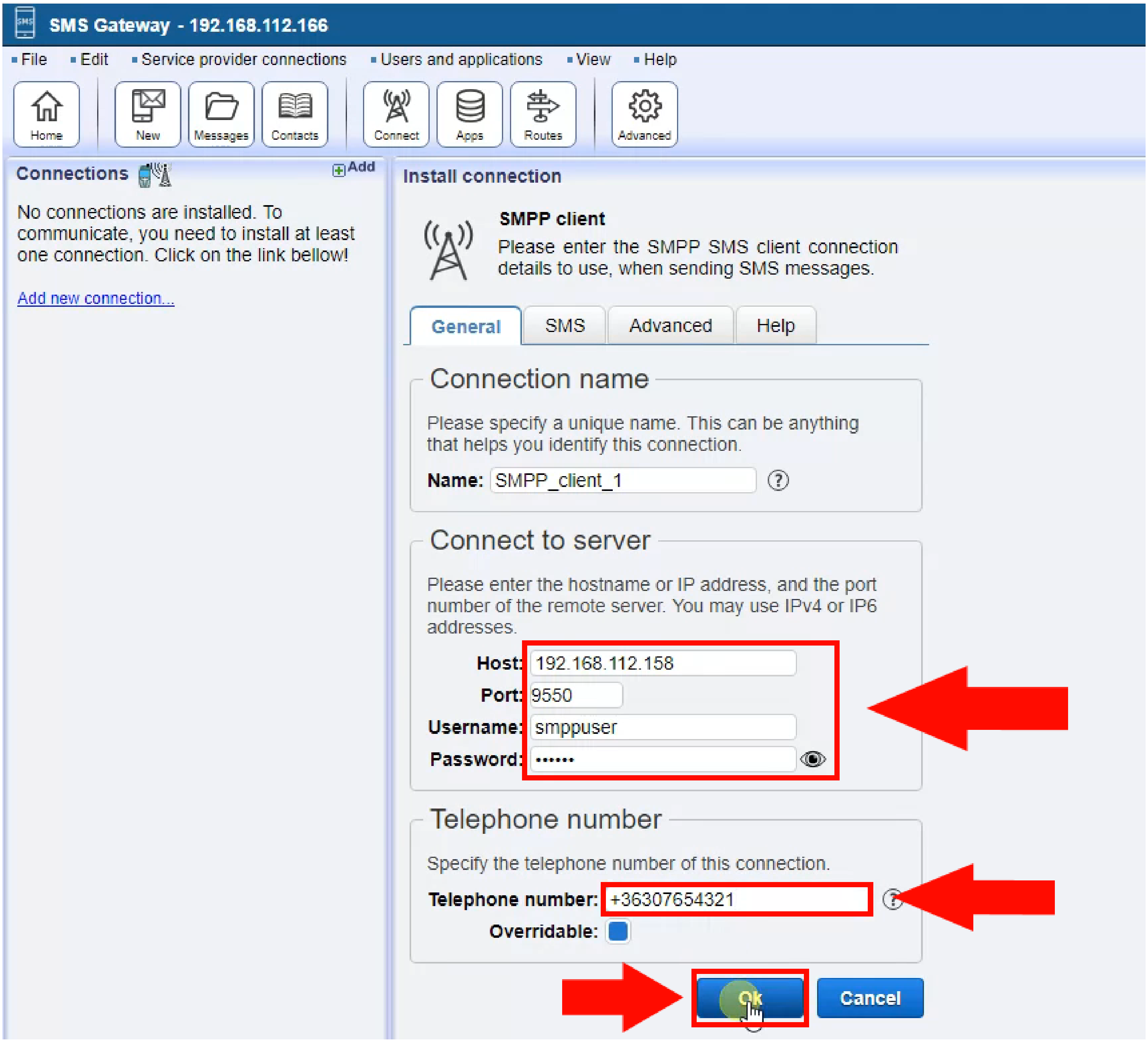
Figure 3 - Installing the SMPP connection

# Configure your SMPP connection

In order to configure your SMPP connection, you need to provide the host name and port number of the SMPP service, your SMPP credentials and your must specify telephone number associated with this connection. There could be more than one phone numbers associated with this connection. In this case provide the first one and check the overridable checkbox. If this checkbox is checked, you will be able to use all phone numbers as sender IDs.

**SMPP connection configuration steps:**

□ Select the General tab on the SMPP connection form
□ Give a name to this SMPP connection
□ Enter the SMPP hostname
□ Enter the SMPP port number
□ Provide your SMPP username
□ Enter your password
□ Assign a telephone number to this SMPP connection
□ Click OK



No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

**SMPP client**

Please enter the SMPP SMS client connection details to use, when sending SMS messages.

Connection name

Please specify a unique name. This can be anything that helps you identify this connection.

**Name:** SMPP_client_1

Connect to server

Please enter the hostname or IP address, and the port number of the remote server. You may use IPv4 or IP6 addresses.

**Host:** 192.168.112.158
**Port:** 9550
**Username:** smppuser
**Password:** ••••••

Telephone number

Specify the telephone number of this connection.

**Telephone number:** +36307654321
**Overridable:** ☑

Figure 4 - Providing SMPP Client credentials

It is important to mention that the SMPP hostname and port plus the username and password are provided by your SMS service provider. For example if you contact Vodafone, and ask for an SMPP SMS service, you will sign a contract with them, and often an attachment of this contract will contain the connection following information. If you subscribe to an SMS service on-line, you will likely find this information in the on-line control panel of your SMS service provider. Of course you can always ask your SMS service provider in e-mail and ask what are the SMPP server connection details.

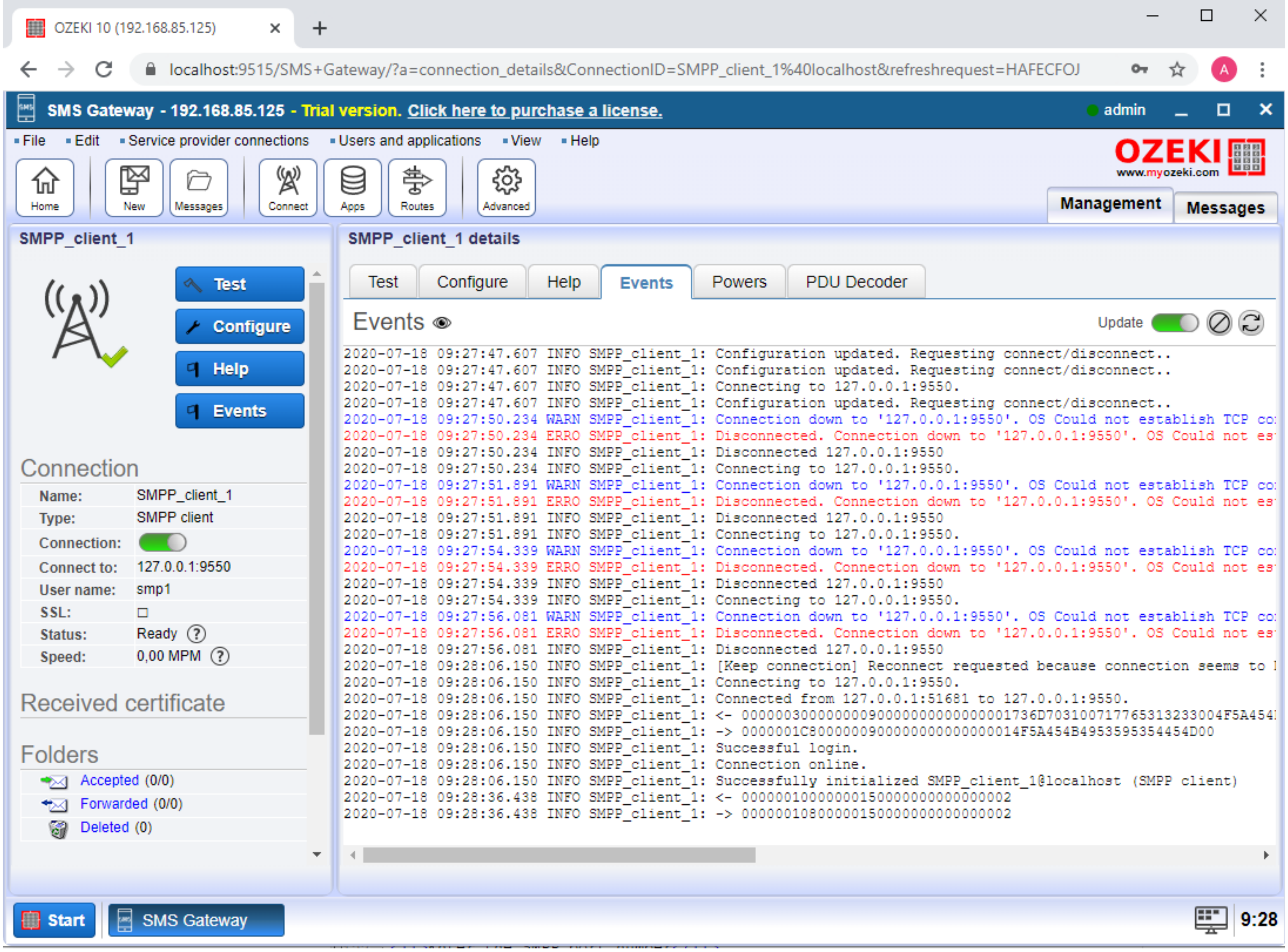
# Check the SMPP log

Once the SMPP connection is configured, you should check the SMPP connection log to see if the system connected properly to your SMS service provider. If the system connected properly you will see the "Connetion online" log entry. The SMS connection logs can be found in the following directory in Windows: C:\Program Files\Ozeki\Data\Logs\Connections\. In this Event tab of the SMPP connection's detalis page, you can see a preview of the log. This page contains the last 100 log entries. Note that the SMPP log files are rotated to save disk space.

**How to view the SMPP connection log:**

- Open the SMPP connection's details page
- Click on the Events tab
- Click on the eye icon next to the Events title
- Copy the file name next to the Events title
- Open notepad

- □ Paste the filename into the File/Open dialog in notepad
- □ Click OK to open the SMPP log file
- □ Use F3 to find the date you are interested in
- □ Search for "ERRO" to find errors in the log


OZEKI 10 (192.168.85.125)
localhost:9515/SMS+Gateway/?a=connection_details&ConnectionID=SMPP_client_1%40localhost&refreshrequest=HAFECFOJ
SMS Gateway - 192.168.85.125 - Trial version. Click here to purchase a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
OZEKI
www.myozeki.com
Management
Messages
Test
Configure
Help
Events
Powers
PDU Decoder
Update
Start
SMS Gateway
9:28

SMPP_client_1

Connection

Name: SMPP_client_1
Type: SMPP client
Connection:
Connect to: 127.0.0.1:9550
User name: smp1
SSL: □
Status: Ready
Speed: 0,00 MPM

Received certificate

Folders

Accepted (0/0)
Forwarded (0/0)
Deleted (0)

SMPP_client_1 details

Events

```
2020-07-18 09:27:47.607 INFO SMPP_client_1: Configuration updated. Requesting connect/disconnect..
2020-07-18 09:27:47.607 INFO SMPP_client_1: Configuration updated. Requesting connect/disconnect..
2020-07-18 09:27:47.607 INFO SMPP_client_1: Connecting to 127.0.0.1:9550.
2020-07-18 09:27:47.607 INFO SMPP_client_1: Configuration updated. Requesting connect/disconnect..
2020-07-18 09:27:50.234 WARN SMPP_client_1: Connection down to '127.0.0.1:9550'. OS Could not establish TCP co
2020-07-18 09:27:50.234 ERRO SMPP_client_1: Disconnected. Connection down to '127.0.0.1:9550'. OS Could not es
2020-07-18 09:27:50.234 INFO SMPP_client_1: Disconnected 127.0.0.1:9550
2020-07-18 09:27:50.234 INFO SMPP_client_1: Connecting to 127.0.0.1:9550.
2020-07-18 09:27:51.891 WARN SMPP_client_1: Connection down to '127.0.0.1:9550'. OS Could not establish TCP co
2020-07-18 09:27:51.891 ERRO SMPP_client_1: Disconnected. Connection down to '127.0.0.1:9550'. OS Could not es
2020-07-18 09:27:51.891 INFO SMPP_client_1: Disconnected 127.0.0.1:9550
2020-07-18 09:27:51.891 INFO SMPP_client_1: Connecting to 127.0.0.1:9550.
2020-07-18 09:27:54.339 WARN SMPP_client_1: Connection down to '127.0.0.1:9550'. OS Could not establish TCP co
2020-07-18 09:27:54.339 ERRO SMPP_client_1: Disconnected. Connection down to '127.0.0.1:9550'. OS Could not es
2020-07-18 09:27:54.339 INFO SMPP_client_1: Disconnected 127.0.0.1:9550
2020-07-18 09:27:54.339 INFO SMPP_client_1: Connecting to 127.0.0.1:9550.
2020-07-18 09:27:56.081 WARN SMPP_client_1: Connection down to '127.0.0.1:9550'. OS Could not establish TCP co
2020-07-18 09:27:56.081 ERRO SMPP_client_1: Disconnected. Connection down to '127.0.0.1:9550'. OS Could not es
2020-07-18 09:27:56.081 INFO SMPP_client_1: Disconnected 127.0.0.1:9550
2020-07-18 09:28:06.150 INFO SMPP_client_1: [Keep connection] Reconnect requested because connection seems to
2020-07-18 09:28:06.150 INFO SMPP_client_1: Connecting to 127.0.0.1:9550.
2020-07-18 09:28:06.150 INFO SMPP_client_1: Connected from 127.0.0.1:51681 to 127.0.0.1:9550.
2020-07-18 09:28:06.150 INFO SMPP_client_1: <- 0000003000000009000000000000001736D7031007177653132333004F5A454
2020-07-18 09:28:06.150 INFO SMPP_client_1: -> 0000001C800000090000000000000014F5A454B4953595354454D00
2020-07-18 09:28:06.150 INFO SMPP_client_1: Successful login.
2020-07-18 09:28:06.150 INFO SMPP_client_1: Connection online.
2020-07-18 09:28:06.150 INFO SMPP_client_1: Successfully initialized SMPP_client_1@localhost (SMPP client)
2020-07-18 09:28:36.438 INFO SMPP_client_1: <- 00000010000000150000000000000002
2020-07-18 09:28:36.438 INFO SMPP_client_1: -> 00000010800000150000000000000002
```

Figure 5 - View the SMPP connection log

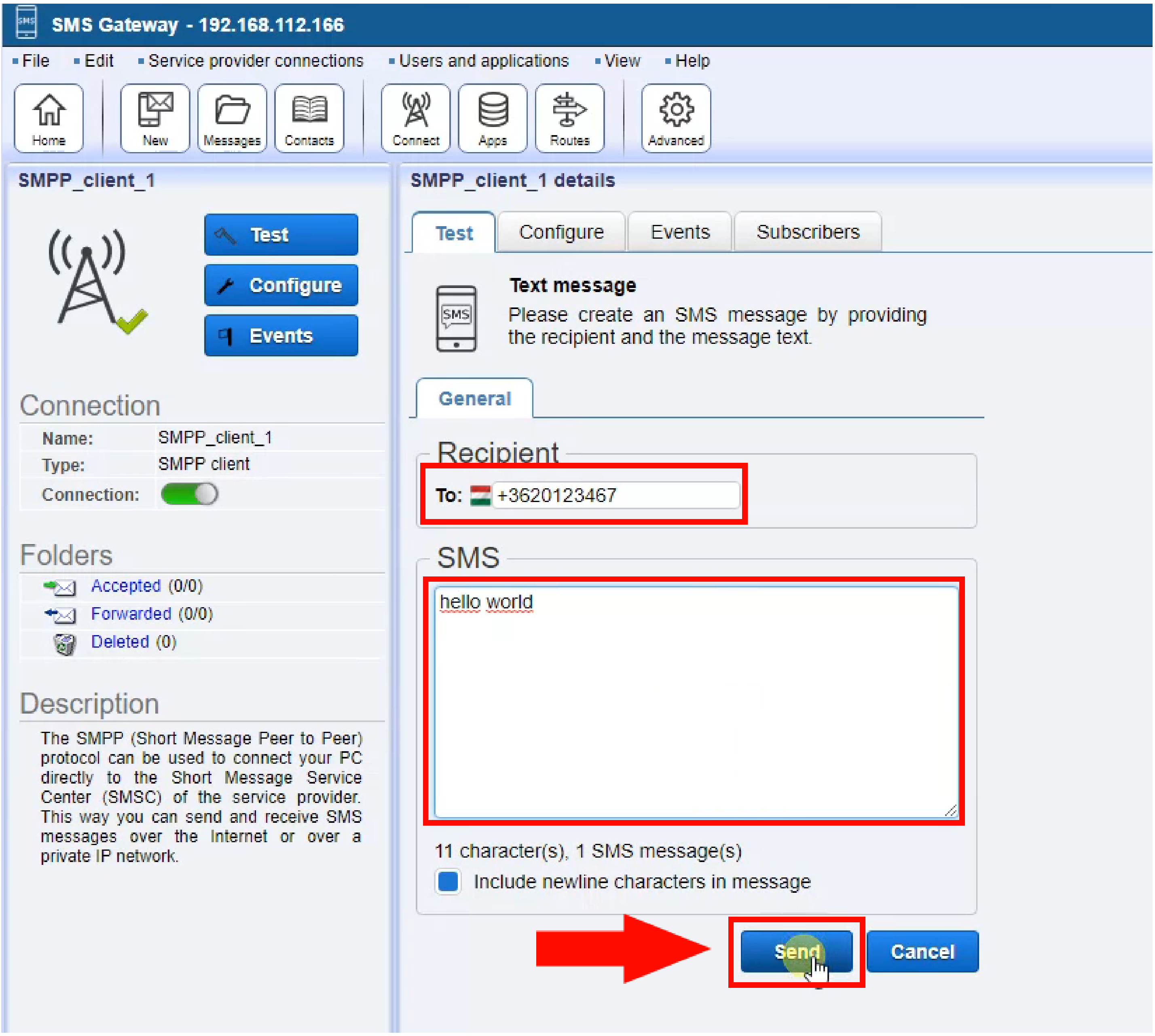
# Send a test SMS message

Once your connection is connected, you can check to see if it is ready to deliver SMS messages by sending a test SMS. To send a test SMS message, you need to open the Test tab, and you need to enter the phone number and message text. It is recommended to provide the phone number in international format. This means the phone number should start with a plus sign followed by a country code. If your local phone number starts with a 0, it is likely that you will need to drop the 0 prefix. For example if your UK phone number 07958663698, you would send the test SMS to +447958663698.

**How to send a test SMPP message:**

- □ Navigate to the SMPP connection's details page
- □ Select the Test tab
- □ Enter the recipient phone number
- □ Make sure the phone number is in international format
- □ Enter the message text
- □ Make sure the message text is less than 160 characters
- □ Click on the Send button
- □ Check the SMPP logs

You might ask why should the message text be less than 160 characters. This is because GSM system was designed to send 160 character long text messages. If a message is longer, it will be split into multiple message segments, and will be delivered in more than one SMS message.



SMPP_client_1

SMPP_client_1 details

Connection

Name: SMPP_client_1
Type: SMPP client
Connection:

Folders

Accepted (0/0)
Forwarded (0/0)
Deleted (0)

Description

The SMPP (Short Message Peer to Peer) protocol can be used to connect your PC directly to the Short Message Service Center (SMSC) of the service provider. This way you can send and receive SMS messages over the Internet or over a private IP network.

**Text message**
Please create an SMS message by providing the recipient and the message text.

Recipient

To: +3620123467

SMS

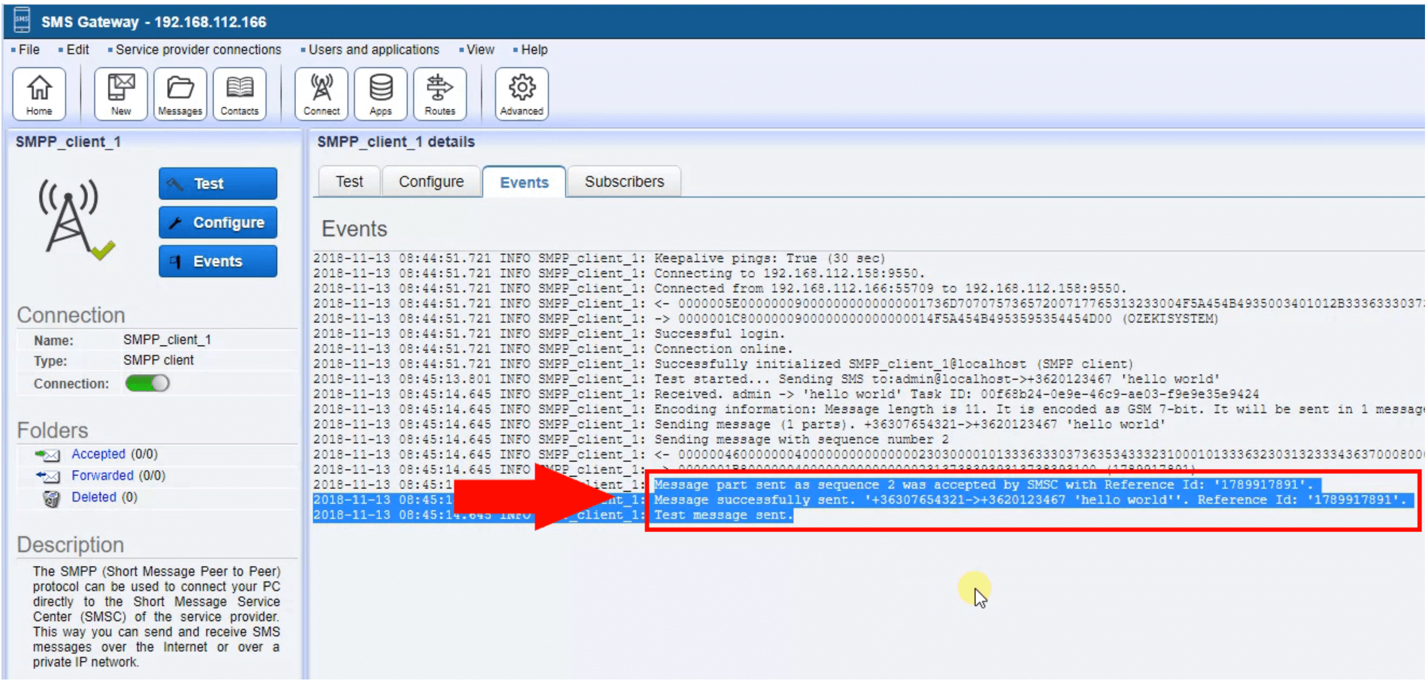
hello world

11 character(s), 1 SMS message(s)

■ Include newline characters in message

Figure 6 - Sending a test message

After the message is successfully sent, you should check the SMPP logs. The SMPP logs will reveal the low level SMPP messages, that are used to pass the message content to the SMS service provider. These messages are also called an SMPP PDU (SMPP protocol data unit). If there is a problem with message submission, your SMS service provider will ask for the SMPP logs. In this case, you need to send the SMPP PDUs to them. For every SMS submission, two SMPP PDUs will appear in the log. One is the SMPP SUBMIT_SM pdu, which passes the message to the SMS service provider's over, and the SMPP SSUBMIT_SM_RESP, which returns a reference ID, that can ce used for tracking a message, and for finding the delivery reports returned to you when the message is delivered to the recipient handset.



Connection

Name: SMPP_client_1
Type: SMPP client
Connection:

The SMPP (Short Message Peer to Peer) protocol can be used to connect your PC directly to the Short Message Service Center (SMSC) of the service provider. This way you can send and receive SMS messages over the Internet or over a private IP network.

```
2018-11-13 08:44:51.721 INFO SMPP_client_1: Keepalive pings: True (30 sec)
2018-11-13 08:44:51.721 INFO SMPP_client_1: Connecting to 192.168.112.158:9550.
2018-11-13 08:44:51.721 INFO SMPP_client_1: Connected from 192.168.112.166:55709 to 192.168.112.158:9550.
2018-11-13 08:44:51.721 INFO SMPP_client_1: <- 0000005E00000009000000000000001736D70707573657200717765313233004F5A454B4935003401012B33363330373
2018-11-13 08:44:51.721 INFO SMPP_client_1: -> 0000001C80000009000000000000000014F5A454B4953595354454D00 (OZEKISYSTEM)
2018-11-13 08:44:51.721 INFO SMPP_client_1: Successful login.
2018-11-13 08:44:51.721 INFO SMPP_client_1: Connection online.
2018-11-13 08:44:51.721 INFO SMPP_client_1: Successfully initialized SMPP_client_1@localhost (SMPP client)
2018-11-13 08:45:13.801 INFO SMPP_client_1: Test started... Sending SMS to:admin@localhost->+3620123467 'hello world'
2018-11-13 08:45:14.645 INFO SMPP_client_1: Received. admin -> 'hello world' Task ID: 00f68b24-0e9e-46c9-ae03-f9e9e35e9424
2018-11-13 08:45:14.645 INFO SMPP_client_1: Encoding information: Message length is 11. It is encoded as GSM 7-bit. It will be sent in 1 messag
2018-11-13 08:45:14.645 INFO SMPP_client_1: Sending message (1 parts). +36307654321->+3620123467 'hello world'
2018-11-13 08:45:14.645 INFO SMPP_client_1: Sending message with sequence number 2
2018-11-13 08:45:14.645 INFO SMPP_client_1: <- 0000004600000004000000000000000230300001013336333037363534333231000101333632303132333436370008000
2018-11-13 08:45:14.645 INFO SMPP_client_1: -> 0000001B8000000400000000000000023137383930313738393100 (1789917891)
2018-11-13 08:45:1            client_1: Message part sent as sequence 2 was accepted by SMSC with Reference Id: '1789917891'.
2018-11-13 08:45:1              nt_1: Message successfully sent. '+36307654321->+3620123467 'hello world''. Reference Id: '1789917891'.
2018-11-13 08:45:14.645 INFO      _client_1: Test message sent.
```

Figure 7 - SMPP SMS submit result in the logs

# SMPP protocol specification

The SMPP protocol specification can be used to understand each SMPP PDU you see in the logs. You will see, that there are several operations in this protocol specifications. For example there is an operation for keeping the SMPP connection alive using keepalive messages, and there are multiple operations for submitting and receiving SMS messages and SMS delivery reports.

**SMPP protocol specification:**

Download: smpp-protocol-specification-v3.4.pdf

# SMPP protocol versions

The SMPP standard is an evolving protocol. The first widely adopted version was v3.3. Currently the most common version you will find is v3.4, but there is also a newer version v5.0 which is rarely used in SMS services.

SMPP 3.3 the oldest version supports GSM SMS messages only. It generates an immediate response for each message sent. In most cases this version is not supported over SSL connections. The problem with SMPP 3.3 is that it requires two SMPP links: an SMPP transmitter and an SMPP receiver link to the SMS service provider. When you setup an SMPP v3.3 link in Ozeki SMS Gateway, you will have to create two SMPP connections, and configure on as SMPP transmitter and the other as SMPP receiver.

The SMPP 3.4 protocol is similar to SMPP 3.3, but it has a strong advantage: it allows you to send and receive SMS messages over a single TCP/IP link. In Ozeki SMS gateway you will only have to setup on SMPP client connection if you wish to use this protocol. SMPP v3.4 also adds optional Tag-Length-Value (TLV) parameters, to the SMS message, which allows the user to work with non-GSM SMS technologies, such as SMS messaging in CDMA networks.

SMPP 5.0 is the latest version of SMPP. It extends v3.4 by adding support for cell broadcasting, smart flow control. Not many SMS service providers use this protocol. We recommend you to setup SMPP v3.4 connections in Ozeki SMS gateway.

# How to setup an SMPP service

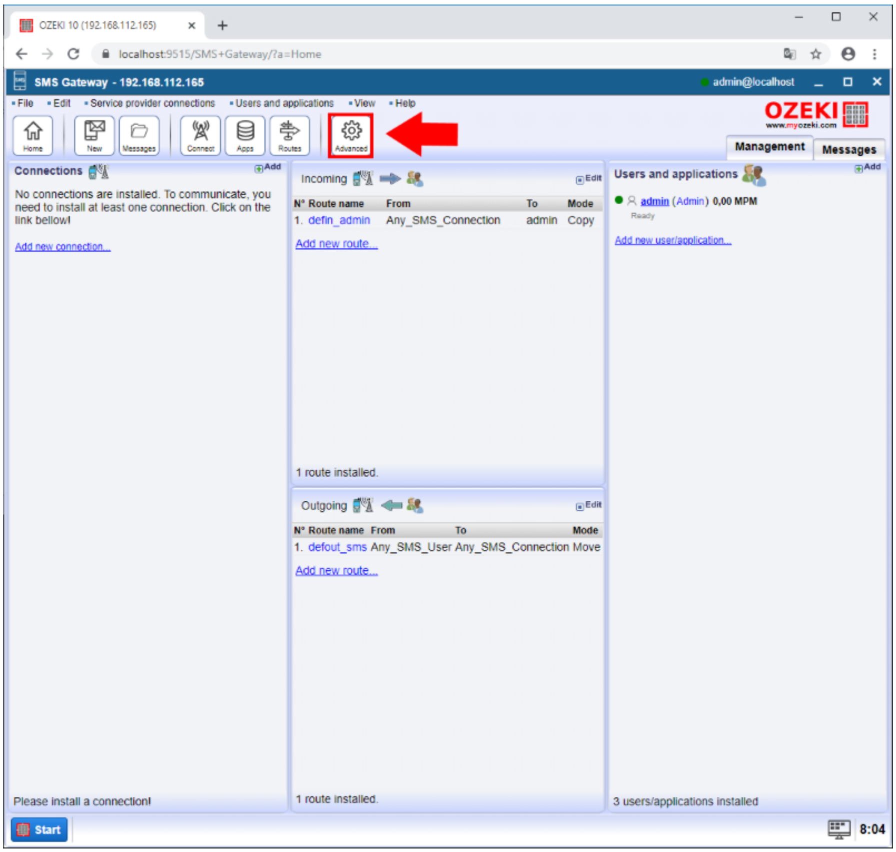
***If you wish to provide SMS service to your customers, the best protocol to use is the Short Message Peer to Peer protocol (SMPP), which is the most widely used IP SMS protocol in the world. Ozeki 10 SMS gateawy has a built in SMPP server, that allows yout to provide SMPP SMS Service. In this guide you will learn how to enable the SMPP SMS service and how you can create SMPP user account, to allow your users (or customers) to connect to your system.***

## Step 1.) Install Ozeki SMS gateway

To provide an SMPP SMS service first you need to install Ozeki SMS Gateway. Depending on your configuration, we recommend you to use a computer with sufficient hardware capacity, to make sure you system will operate well. Check out the Ozeki SMS Gateway prerequisites page for more information about the recommended hardware configuration.

## Step 2.) Open Advanced menu

You can find Ozeki 10 services in the Advanced menu. To open it please click the **"Advanced"** button on the main page.



No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

Please install a connection!

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defin_admin | Any_SMS_Connection | admin | Copy |

Add new route...

1 route installed.

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move |

Add new route...

1 route installed.

admin (Admin) 0,00 MPM
Ready

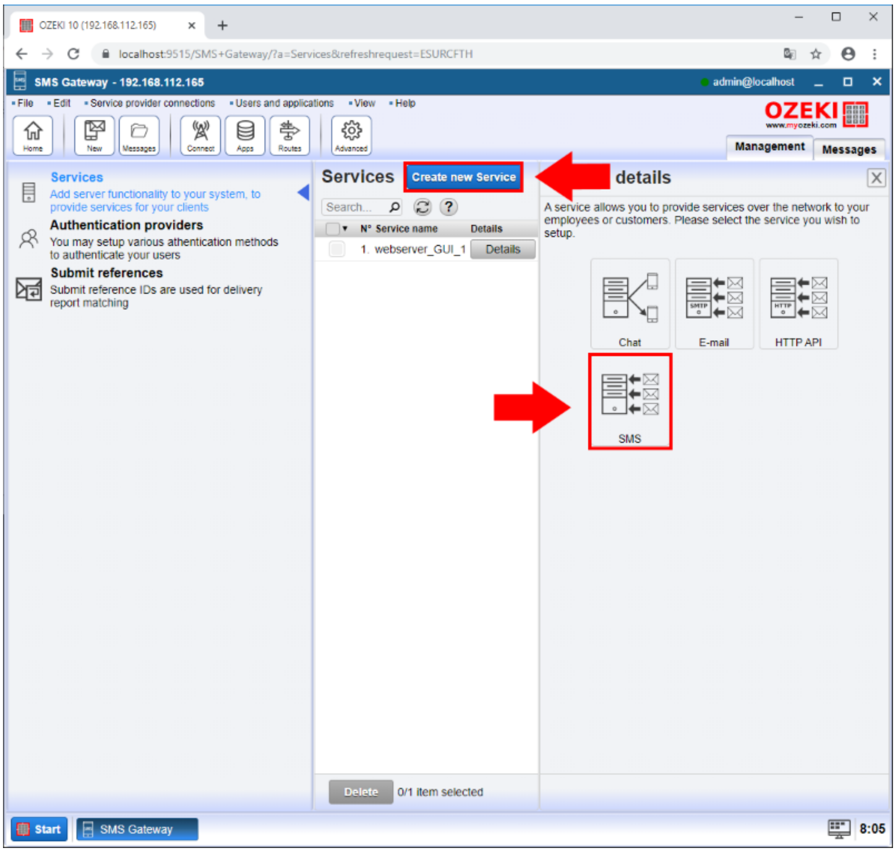
Add new user/application...

3 users/applications installed

**Figure 1** - Click on Advanced button.

## Step 3.) Create new service

In the Advanced menu click on the **"Create new Service"** button and then select SMS service.



Services

Add server functionality to your system, to provide services for your clients

Authentication providers

You may setup various athentication methods to authenticate your users

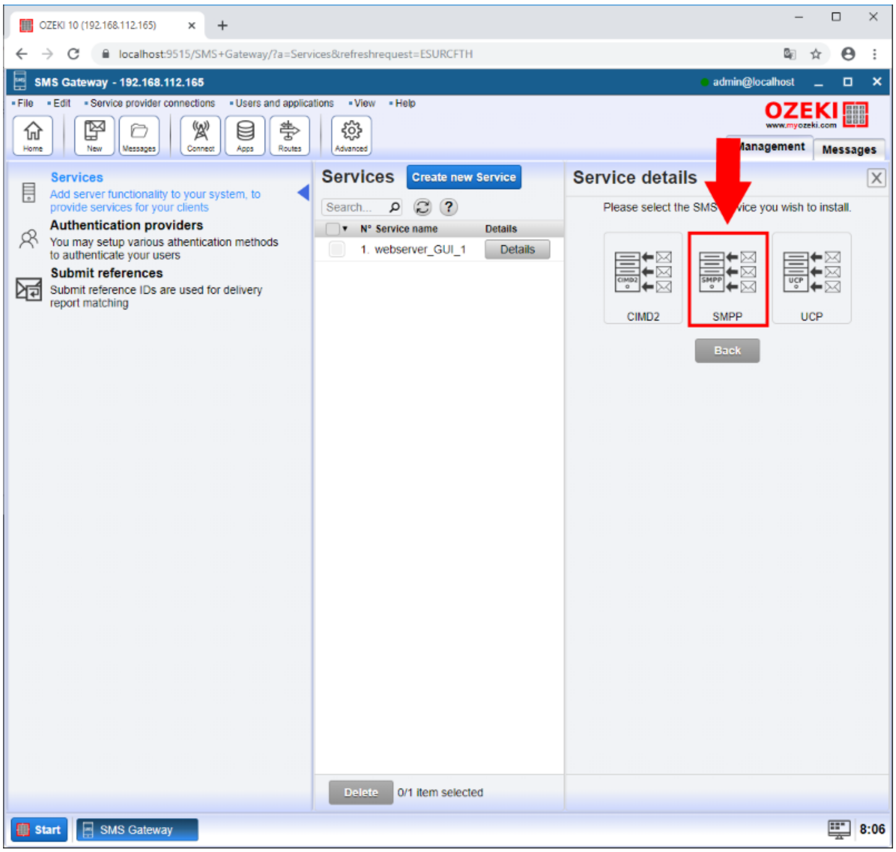
Submit references

Submit reference IDs are used for delivery report matching

A service allows you to provide services over the network to your employees or customers. Please select the service you wish to setup.

**Figure 2** - Create SMS service

After it select SMPP service.



Services

Add server functionality to your system, to provide services for your clients

Authentication providers

You may setup various athentication methods to authenticate your users

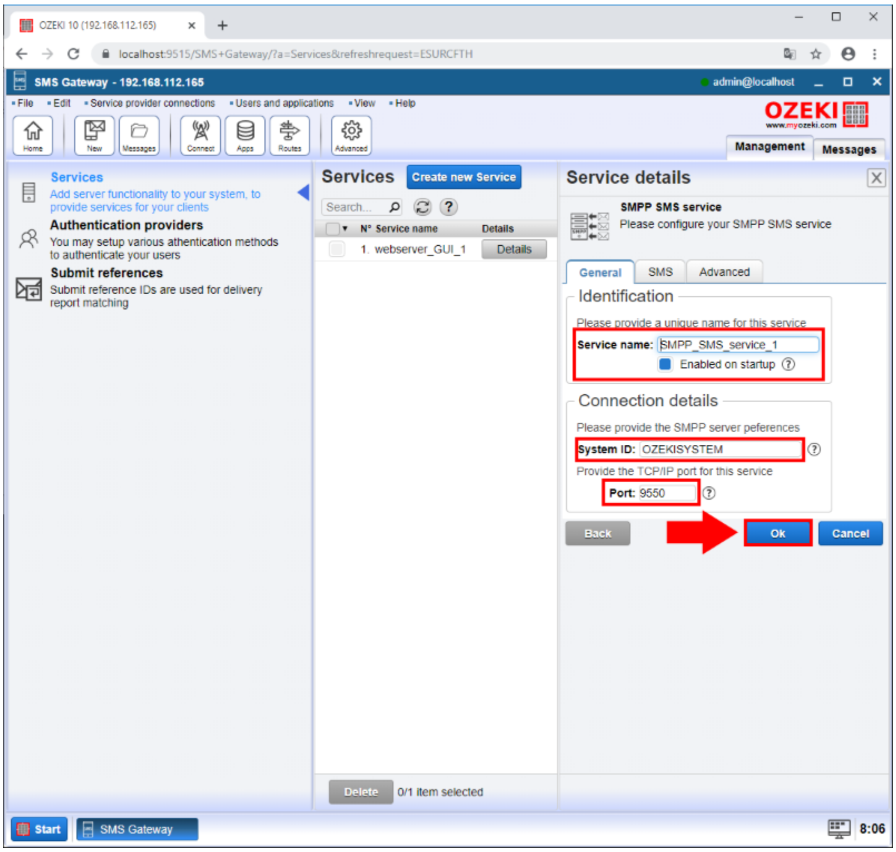
Submit references

Submit reference IDs are used for delivery report matching

Please select the SMS service you wish to install.

**Figure 3** - Select SMPP service

Please provide a unique name a system ID and a port for this service.



Please provide a unique name for this service

Service name: SMPP_SMS_service_1

☑ Enabled on startup

Please provide the SMPP server peferences

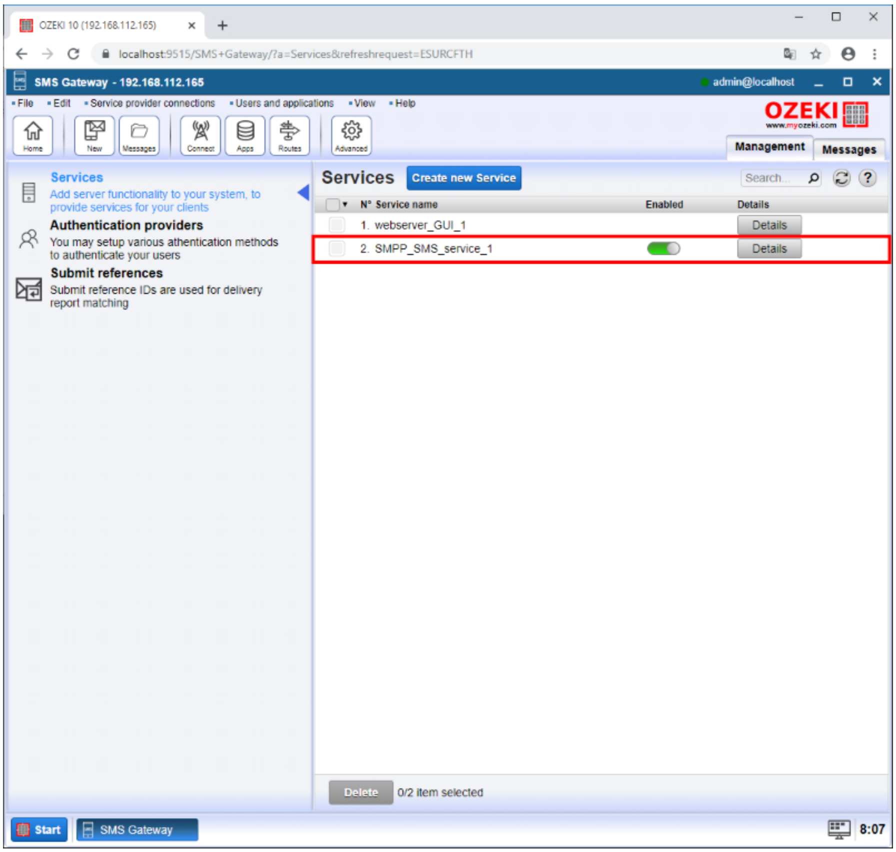
System ID: OZEKISYSTEM

Provide the TCP/IP port for this service

Port: 9550

**Figure 4 -** Provide Service informations

You can see the new service created.

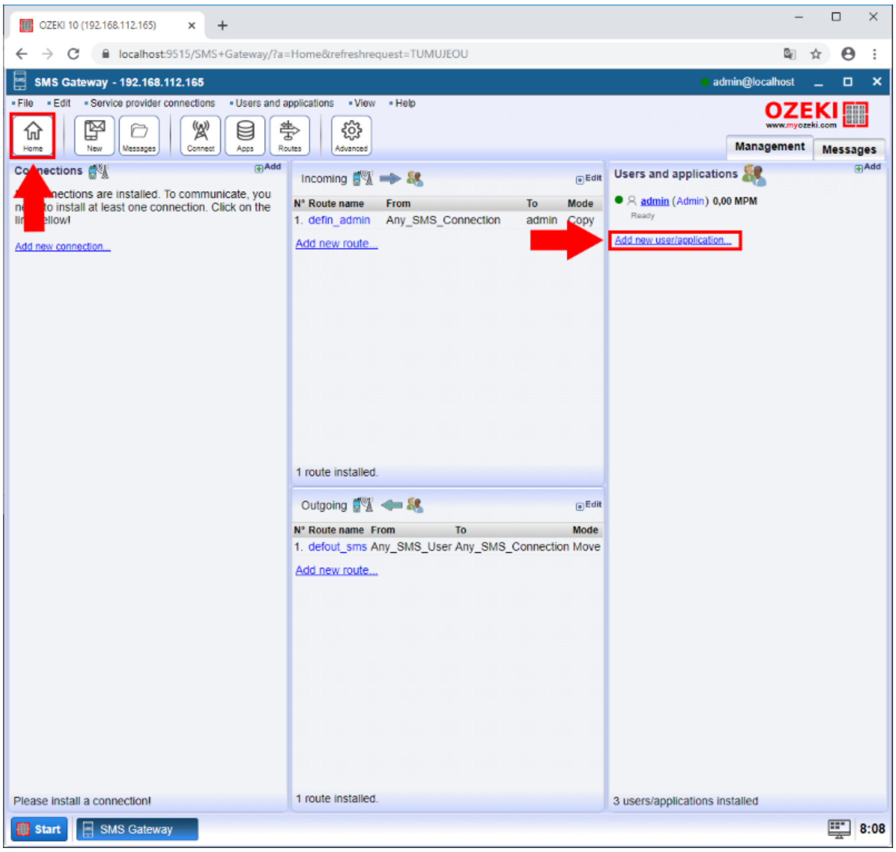


| N° | Service name | Enabled | Details |
|---|---|---|---|
| 1. | webserver_GUI_1 | | Details |
| 2. | SMPP_SMS_service_1 | ● | Details |

**Figure 5 -** SMPP service created

# Step 4.) Create new SMPP user

To start installing and configuring an SMPP user in the Home page click the Add new user/application.



Connections

No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

Please install a connection!

Incoming

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defin_admin | Any_SMS_Connection | admin | Copy |

Add new route...

1 route installed.

Outgoing

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move |

Add new route...

1 route installed.

Users and applications

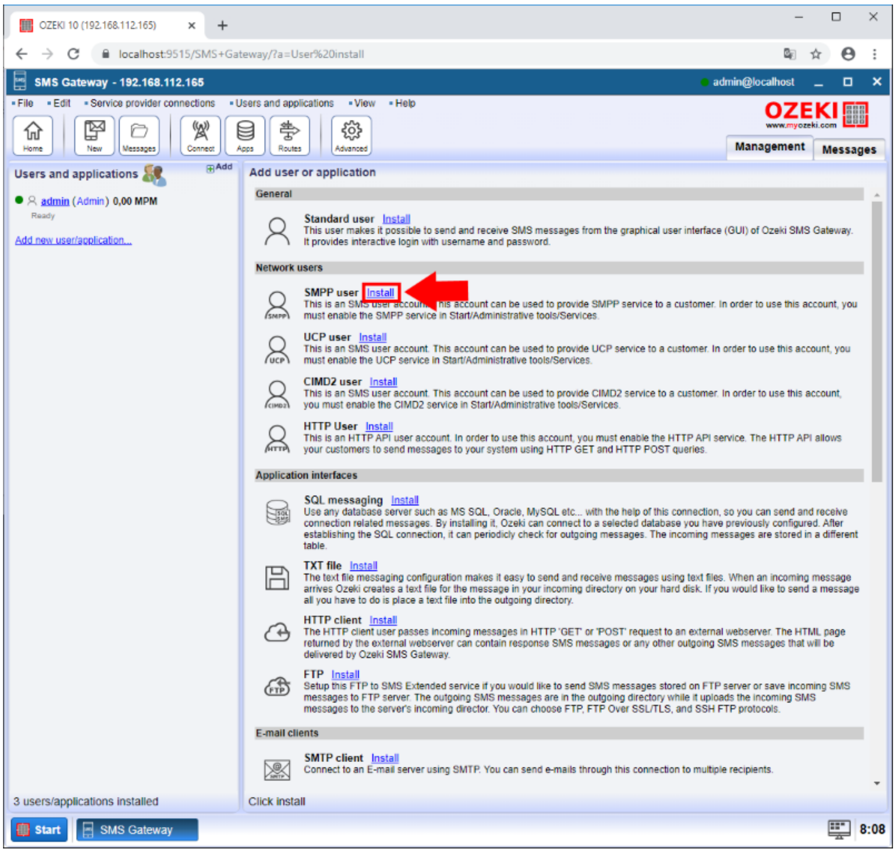
● admin (Admin) 0,00 MPM
Ready

Add new user/application...

3 users/applications installed

**Figure 6** - Add new user

Then click the Install button next to the SMPP user in the list.



Users and applications

● admin (Admin) 0,00 MPM
Ready

Add new user/application...

3 users/applications installed

Add user or application

General

Standard user Install
This user makes it possible to send and receive SMS messages from the graphical user interface (GUI) of Ozeki SMS Gateway. It provides interactive login with username and password.

Network users

SMPP user Install
This is an SMS user account. This account can be used to provide SMPP service to a customer. In order to use this account, you must enable the SMPP service in Start/Administrative tools/Services.

UCP user Install
This is an SMS user account. This account can be used to provide UCP service to a customer. In order to use this account, you must enable the UCP service in Start/Administrative tools/Services.

CIMD2 user Install
This is an SMS user account. This account can be used to provide CIMD2 service to a customer. In order to use this account, you must enable the CIMD2 service in Start/Administrative tools/Services.

HTTP User Install
This is an HTTP API user account. In order to use this account, you must enable the HTTP API service. The HTTP API allows your customers to send messages to your system using HTTP GET and HTTP POST queries.

Application interfaces

SQL messaging Install
Use any database server such as MS SQL, Oracle, MySQL etc... with the help of this connection, so you can send and receive connection related messages. By installing it, Ozeki can connect to a selected database you have previously configured. After establishing the SQL connection, it can periodicly check for outgoing messages. The incoming messages are stored in a different table.

TXT file Install
The text file messaging configuration makes it easy to send and receive messages using text files. When an incoming message arrives Ozeki creates a text file for the message in your incoming directory on your hard disk. If you would like to send a message all you have to do is place a text file into the outgoing directory.

HTTP client Install
The HTTP client user passes incoming messages in HTTP 'GET' or 'POST' request to an external webserver. The HTML page returned by the external webserver can contain response SMS messages or any other outgoing SMS messages that will be delivered by Ozeki SMS Gateway.

FTP Install
Setup this FTP to SMS Extended service if you would like to send SMS messages stored on FTP server or save incoming SMS messages to FTP server. The outgoing SMS messages are in the outgoing directory while it uploads the incoming SMS messages to the server's incoming director. You can choose FTP, FTP Over SSL/TLS, and SSH FTP protocols.

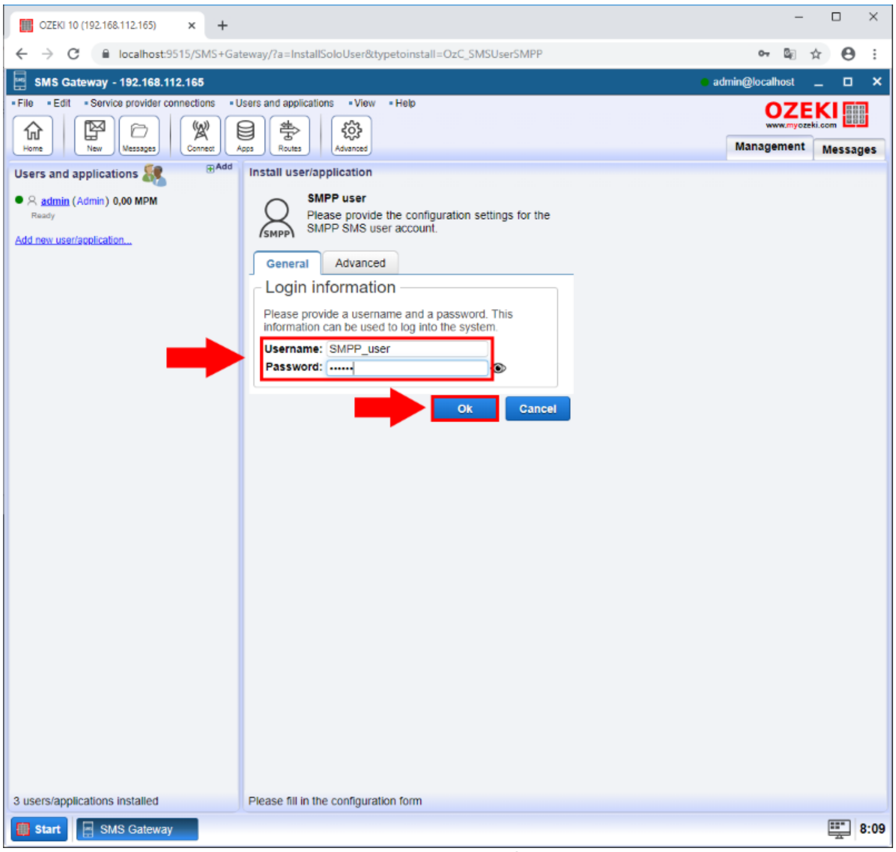
E-mail clients

SMTP client Install
Connect to an E-mail server using SMTP. You can send e-mails through this connection to multiple recipients.

Click install

**Figure 7** - Install SMPP user

Provide the Username and the Password for the new SMPP user.



Install user/application

SMPP user

Please provide the configuration settings for the SMPP SMS user account.

Login information

Please provide a username and a password. This information can be used to log into the system.

Username: SMPP_user

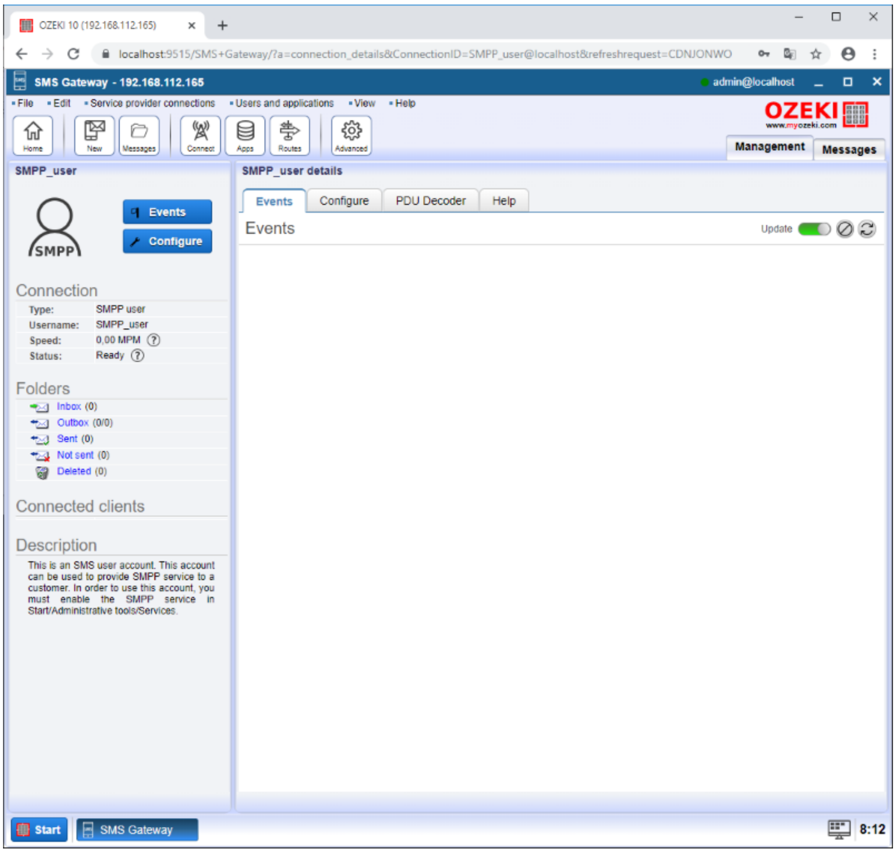
Password: ••••••

3 users/applications installed

Please fill in the configuration form

**Figure 8 -** Login information

Then you click on OK you will see the User's Events page.



Connection

Type: SMPP user

Username: SMPP_user

Speed: 0,00 MPM

Status: Ready

Folders

Inbox (0)

Outbox (0/0)

Sent (0)

Not sent (0)

Deleted (0)

Connected clients

Description

This is an SMS user account. This account can be used to provide SMPP service to a customer. In order to use this account, you must enable the SMPP service in Start/Administrative tools/Services.

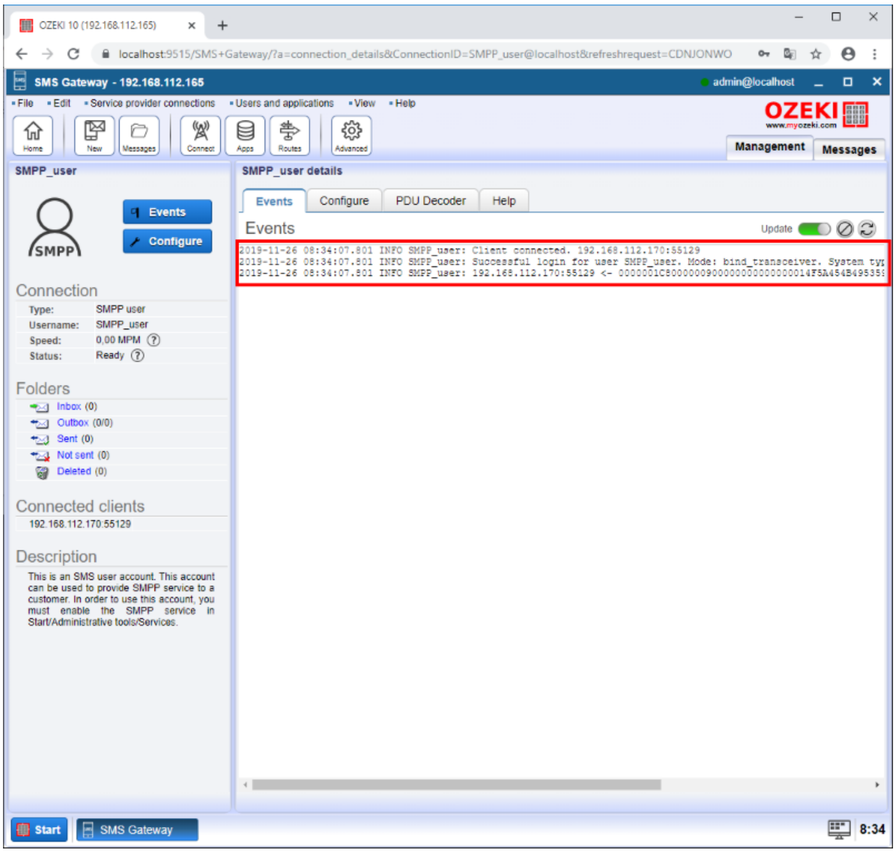
**Figure 9 -** SMPP user created

# Step 5.) Setup an SMPP client on a different computer

To receive SMS via SMPP service connection You need an SMPP Client on a different computer. Here You can find detailed instructions on how to install and configure an SMPP Client connection for SMS messaging using the Ozeki 10 SMS Gateway software.

## Step 6.) Check the logs about client connections

In the User's Events tab you can see the successful login.



Connection

Type: SMPP user
Username: SMPP_user
Speed: 0.00 MPM
Status: Ready

Folders

Inbox (0)
Outbox (0/0)
Sent (0)
Not sent (0)
Deleted (0)

Connected clients

192.168.112.170:55129

Description

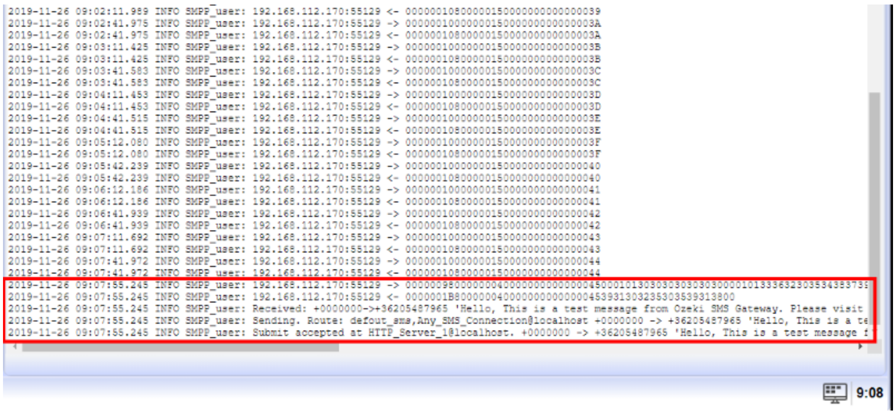
This is an SMS user account. This account can be used to provide SMPP service to a customer. In order to use this account, you must enable the SMPP service in Start/Administrative tools/Services.

```
2019-11-26 08:34:07.801 INFO SMPP_user: Client connected. 192.168.112.170:55129
2019-11-26 08:34:07.801 INFO SMPP_user: Successful login for user SMPP_user. Mode: bind_transceiver. System typ
2019-11-26 08:34:07.801 INFO SMPP_user: 192.168.112.170:55129 <- 0000001C80000009000000000000000014F5A454B49535
```

**Figure 10** - SMPP Client connected

## Step 7.) Check the logs about an SMS being received

If everything fine the message successfully received.



```
2019-11-26 09:02:11.989 INFO SMPP_user: 192.168.112.170:55129 <- 00000010800000150000000000000039
2019-11-26 09:02:41.975 INFO SMPP_user: 192.168.112.170:55129 -> 0000001000000015000000000000003A
2019-11-26 09:02:41.975 INFO SMPP_user: 192.168.112.170:55129 <- 0000001080000015000000000000003A
2019-11-26 09:03:11.425 INFO SMPP_user: 192.168.112.170:55129 -> 0000001000000015000000000000003B
2019-11-26 09:03:11.425 INFO SMPP_user: 192.168.112.170:55129 <- 0000001080000015000000000000003B
2019-11-26 09:03:41.583 INFO SMPP_user: 192.168.112.170:55129 -> 0000001000000015000000000000003C
2019-11-26 09:03:41.583 INFO SMPP_user: 192.168.112.170:55129 <- 0000001080000015000000000000003C
2019-11-26 09:04:11.453 INFO SMPP_user: 192.168.112.170:55129 -> 0000001000000015000000000000003D
2019-11-26 09:04:11.453 INFO SMPP_user: 192.168.112.170:55129 <- 0000001080000015000000000000003D
2019-11-26 09:04:41.515 INFO SMPP_user: 192.168.112.170:55129 -> 0000001000000015000000000000003E
2019-11-26 09:04:41.515 INFO SMPP_user: 192.168.112.170:55129 <- 0000001080000015000000000000003E
2019-11-26 09:05:12.080 INFO SMPP_user: 192.168.112.170:55129 -> 0000001000000015000000000000003F
2019-11-26 09:05:12.080 INFO SMPP_user: 192.168.112.170:55129 <- 0000001080000015000000000000003F
2019-11-26 09:05:42.239 INFO SMPP_user: 192.168.112.170:55129 -> 00000010000000150000000000000040
2019-11-26 09:05:42.239 INFO SMPP_user: 192.168.112.170:55129 <- 00000010800000150000000000000040
2019-11-26 09:06:12.186 INFO SMPP_user: 192.168.112.170:55129 -> 00000010000000150000000000000041
2019-11-26 09:06:12.186 INFO SMPP_user: 192.168.112.170:55129 <- 00000010800000150000000000000041
2019-11-26 09:06:41.939 INFO SMPP_user: 192.168.112.170:55129 -> 00000010000000150000000000000042
2019-11-26 09:06:41.939 INFO SMPP_user: 192.168.112.170:55129 <- 00000010800000150000000000000042
2019-11-26 09:07:11.692 INFO SMPP_user: 192.168.112.170:55129 -> 00000010000000150000000000000043
2019-11-26 09:07:11.692 INFO SMPP_user: 192.168.112.170:55129 <- 00000010800000150000000000000043
2019-11-26 09:07:41.972 INFO SMPP_user: 192.168.112.170:55129 -> 00000010000000150000000000000044
2019-11-26 09:07:41.972 INFO SMPP_user: 192.168.112.170:55129 <- 00000010800000150000000000000044
2019-11-26 09:07:55.245 INFO SMPP_user: 192.168.112.170:55129 -> 0000009800000004000000000000004500010130303030303030000101333632303534383739
2019-11-26 09:07:55.245 INFO SMPP_user: 192.168.112.170:55129 <- 0000001B80000004000000000000004539313032353035393138 00
2019-11-26 09:07:55.245 INFO SMPP_user: Received: +0000000->+36205487965 'Hello, This is a test message from Ozeki SMS Gateway. Please visit
2019-11-26 09:07:55.245 INFO SMPP_user: Sending. Route: defout_sms,Any_SMS_Connection@localhost +0000000 -> +36205487965 'Hello, This is a te
2019-11-26 09:07:55.245 INFO SMPP_user: Submit accepted at HTTP_Server_1@localhost. +0000000 -> +36205487965 'Hello, This is a test message f
```

**Figure 11** - SMS received

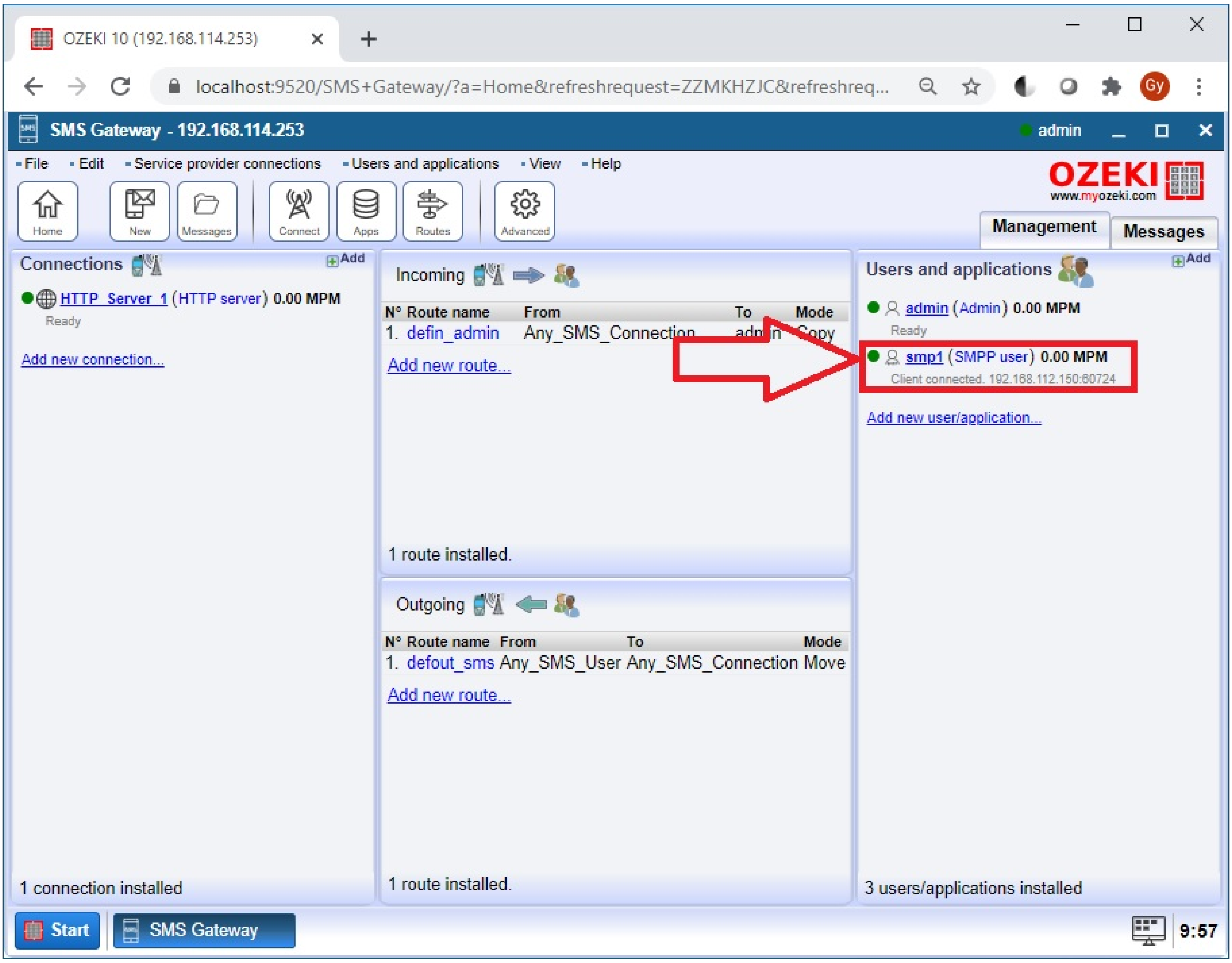
# How to track an SMPP SMS

***Ozeki SMS Gateway provides several logs to find out what happened to a certain SMS that went through the system. If you provide an SMPP service, sometimes you will get a request from your customer asking about an SMS. This guide gives you information on how to find out what happened to a single SMS.***

## Find the SMS sent by the customer

To find the customer's SMS, first open the SMPP user account of the customer. Next select the event log tab, so you can see the communication between your system and your customer's system. If you don't see the message in the logs, you might want to open the log file with notepad. The logfile can be found at:

**C:\Program Files\Ozeki\Data\Logs\Connections\SMPP_user_smp1_localhost.txt**



OZEKI 10 (192.168.114.253)

localhost:9520/SMS+Gateway/?a=Home&refreshrequest=ZZMKHZJC&refreshreq...

SMS Gateway - 192.168.114.253

Connections

● HTTP_Server_1 (HTTP server) 0.00 MPM
Ready

Add new connection...

1 connection installed

Incoming

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defin_admin | Any_SMS_Connection | admin | Copy |

Add new route...

1 route installed.

Outgoing

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move |

Add new route...

1 route installed.

Users and applications

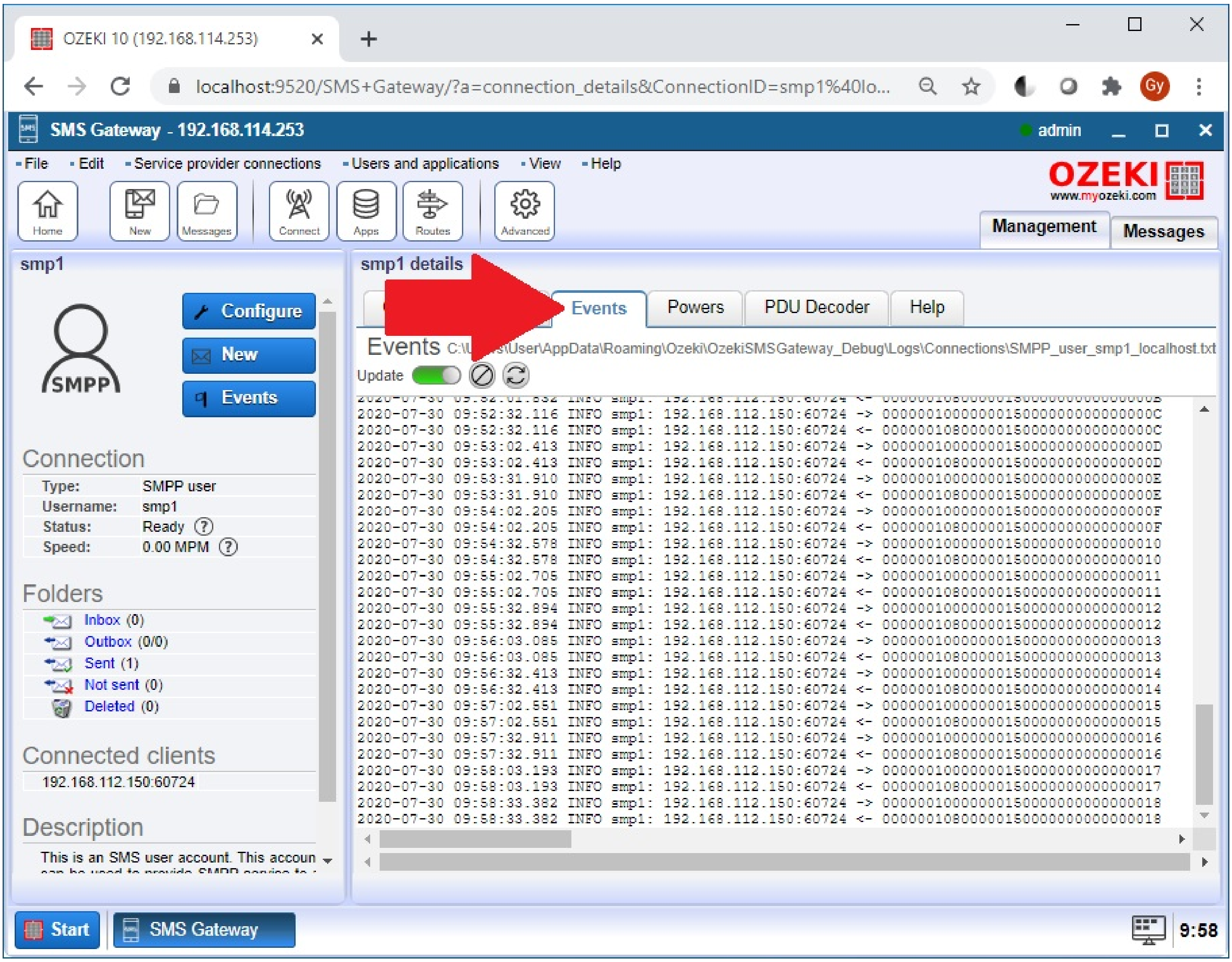
● admin (Admin) 0.00 MPM
Ready

● smp1 (SMPP user) 0.00 MPM
Client connected. 192.168.112.150:60724

Add new user/application...

3 users/applications installed

Figure 1 - Open the SMPP user account



Events C:\U...\User\AppData\Roaming\Ozeki\OzekiSMSGateway_Debug\Logs\Connections\SMPP_user_smp1_localhost.txt

```
2020-07-30 09:52:32.116 INFO smp1: 192.168.112.150:60724 -> 0000001000000015000000000000000C
2020-07-30 09:52:32.116 INFO smp1: 192.168.112.150:60724 <- 0000001080000015000000000000000C
2020-07-30 09:53:02.413 INFO smp1: 192.168.112.150:60724 -> 0000001000000015000000000000000D
2020-07-30 09:53:02.413 INFO smp1: 192.168.112.150:60724 <- 0000001080000015000000000000000D
2020-07-30 09:53:31.910 INFO smp1: 192.168.112.150:60724 -> 0000001000000015000000000000000E
2020-07-30 09:53:31.910 INFO smp1: 192.168.112.150:60724 <- 0000001080000015000000000000000E
2020-07-30 09:54:02.205 INFO smp1: 192.168.112.150:60724 -> 0000001000000015000000000000000F
2020-07-30 09:54:02.205 INFO smp1: 192.168.112.150:60724 <- 0000001080000015000000000000000F
2020-07-30 09:54:32.578 INFO smp1: 192.168.112.150:60724 -> 00000010000000150000000000000010
2020-07-30 09:54:32.578 INFO smp1: 192.168.112.150:60724 <- 00000010800000150000000000000010
2020-07-30 09:55:02.705 INFO smp1: 192.168.112.150:60724 -> 00000010000000150000000000000011
2020-07-30 09:55:02.705 INFO smp1: 192.168.112.150:60724 <- 00000010800000150000000000000011
2020-07-30 09:55:32.894 INFO smp1: 192.168.112.150:60724 -> 00000010000000150000000000000012
2020-07-30 09:55:32.894 INFO smp1: 192.168.112.150:60724 <- 00000010800000150000000000000012
2020-07-30 09:56:03.085 INFO smp1: 192.168.112.150:60724 -> 00000010000000150000000000000013
2020-07-30 09:56:03.085 INFO smp1: 192.168.112.150:60724 <- 00000010800000150000000000000013
2020-07-30 09:56:32.413 INFO smp1: 192.168.112.150:60724 -> 00000010000000150000000000000014
2020-07-30 09:56:32.413 INFO smp1: 192.168.112.150:60724 <- 00000010800000150000000000000014
2020-07-30 09:57:02.551 INFO smp1: 192.168.112.150:60724 -> 00000010000000150000000000000015
2020-07-30 09:57:02.551 INFO smp1: 192.168.112.150:60724 <- 00000010800000150000000000000015
2020-07-30 09:57:32.911 INFO smp1: 192.168.112.150:60724 -> 00000010000000150000000000000016
2020-07-30 09:57:32.911 INFO smp1: 192.168.112.150:60724 <- 00000010800000150000000000000016
2020-07-30 09:58:03.193 INFO smp1: 192.168.112.150:60724 -> 00000010000000150000000000000017
2020-07-30 09:58:03.193 INFO smp1: 192.168.112.150:60724 <- 00000010800000150000000000000017
2020-07-30 09:58:33.382 INFO smp1: 192.168.112.150:60724 -> 00000010000000150000000000000018
2020-07-30 09:58:33.382 INFO smp1: 192.168.112.150:60724 <- 00000010800000150000000000000018
```

Connection

| | |
|---|---|
| Type: | SMPP user |
| Username: | smp1 |
| Status: | Ready |
| Speed: | 0.00 MPM |

Folders

- Inbox (0)
- Outbox (0/0)
- Sent (1)
- Not sent (0)
- Deleted (0)

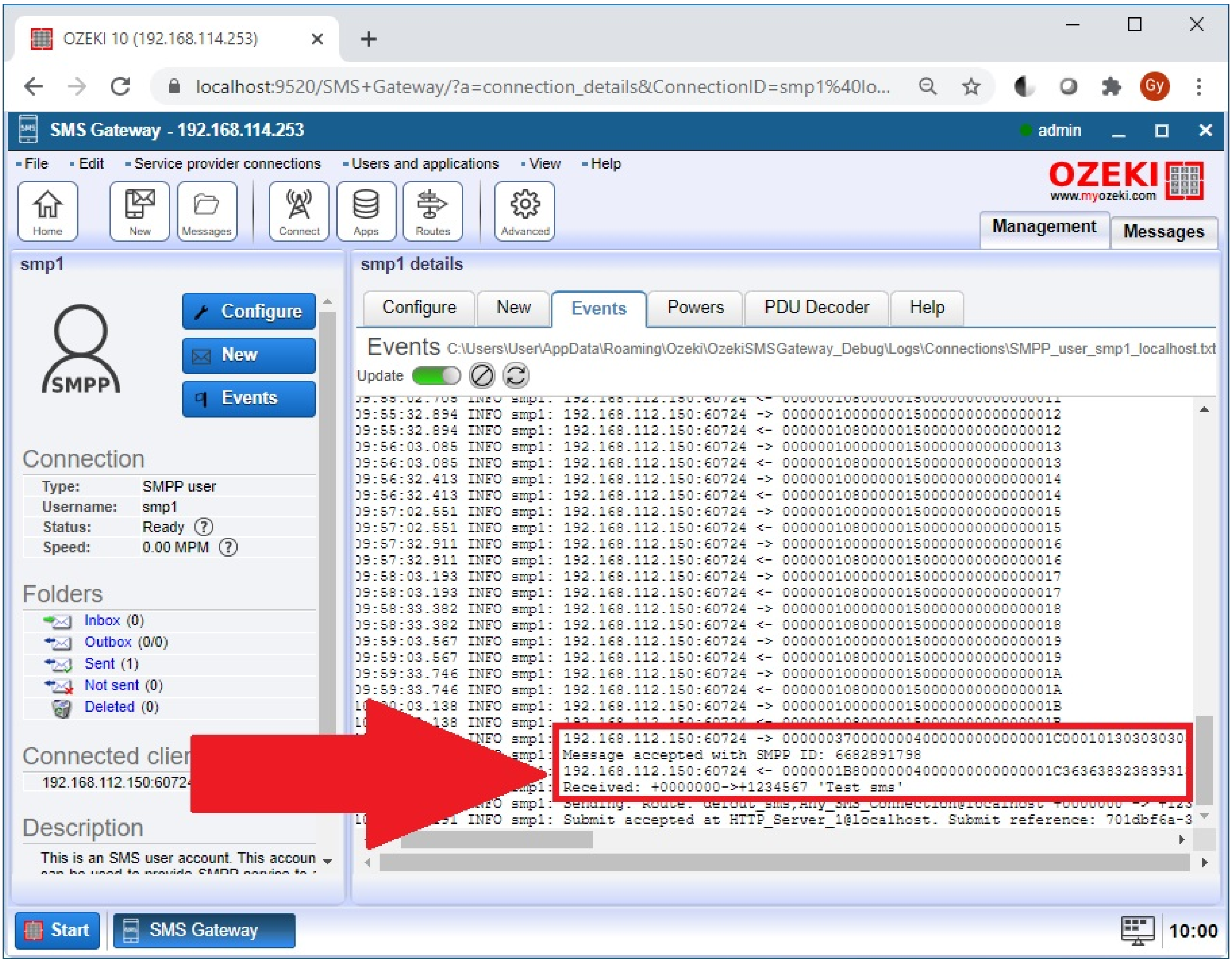
Connected clients

192.168.112.150:60724

Description

This is an SMS user account. This accoun

Figure 2 - Select the events tab



Connection

| | |
|---|---|
| Type: | SMPP user |
| Username: | smp1 |
| Status: | Ready |
| Speed: | 0.00 MPM |

Folders

- Inbox (0)
- Outbox (0/0)
- Sent (1)
- Not sent (0)
- Deleted (0)

Connected clier

192.168.112.150:6072

Description

This is an SMS user account. This accoun

Events C:\Users\User\AppData\Roaming\Ozeki\OzekiSMSGateway_Debug\Logs\Connections\SMPP_user_smp1_localhost.txt

```
09:55:32.894 INFO smp1: 192.168.112.150:60724 -> 00000010000000150000000000000012
09:55:32.894 INFO smp1: 192.168.112.150:60724 <- 00000010800000150000000000000012
09:56:03.085 INFO smp1: 192.168.112.150:60724 -> 00000010000000150000000000000013
09:56:03.085 INFO smp1: 192.168.112.150:60724 <- 00000010800000150000000000000013
09:56:32.413 INFO smp1: 192.168.112.150:60724 -> 00000010000000150000000000000014
09:56:32.413 INFO smp1: 192.168.112.150:60724 <- 00000010800000150000000000000014
09:57:02.551 INFO smp1: 192.168.112.150:60724 -> 00000010000000150000000000000015
09:57:02.551 INFO smp1: 192.168.112.150:60724 <- 00000010800000150000000000000015
09:57:32.911 INFO smp1: 192.168.112.150:60724 -> 00000010000000150000000000000016
09:57:32.911 INFO smp1: 192.168.112.150:60724 <- 00000010800000150000000000000016
09:58:03.193 INFO smp1: 192.168.112.150:60724 -> 00000010000000150000000000000017
09:58:03.193 INFO smp1: 192.168.112.150:60724 <- 00000010800000150000000000000017
09:58:33.382 INFO smp1: 192.168.112.150:60724 -> 00000010000000150000000000000018
09:58:33.382 INFO smp1: 192.168.112.150:60724 <- 00000010800000150000000000000018
09:59:03.567 INFO smp1: 192.168.112.150:60724 -> 00000010000000150000000000000019
09:59:03.567 INFO smp1: 192.168.112.150:60724 <- 00000010800000150000000000000019
09:59:33.746 INFO smp1: 192.168.112.150:60724 -> 0000001000000015000000000000001A
09:59:33.746 INFO smp1: 192.168.112.150:60724 <- 0000001080000015000000000000001A
10:00:03.138 INFO smp1: 192.168.112.150:60724 -> 0000001000000015000000000000001B
10:00:03.138 INFO smp1: 192.168.112.150:60724 <- 0000001080000015000000000000001B
              INFO smp1: 192.168.112.150:60724 -> 000000370000000400000000000000001C00010130303030
                   smp1: Message accepted with SMPP ID: 6682891798
                         192.168.112.150:60724 <- 0000001B8000000400000000000000001C3636383238393
                   smp1: Received: +0000000->+1234567 'Test sms'
                   smp1: Sending. Route: defout_sms,Any_SMS_Connection@localhost +0000000 -> +123
10:       31 INFO smp1: Submit accepted at HTTP_Server_1@localhost. Submit reference: 701dbf6a-3
```

Figure 3 - Find the log entry corresponding to the message.

## Submit SM log

This log entry usually contains 5 lines of code. The first line is the submit request sent by the customers system, then you see our response to this request, then we log the routing and delivery events corresponding to the message.

```
INFO smp1: 192.168.112.150:60724 ->
000000370000000400000000000000001C000101303030303030300001013132333435363700000000100000100000008546
INFO smp1: Message accepted with SMPP ID: 6682891798
INFO smp1: 192.168.112.150:60724 <- 0000001B8000000400000000000000001C36363832383931373938 00
INFO smp1: Received: +0000000->+1234567 'Test sms'
INFO smp1: Sending. Route: defout_sms,Any_SMS_Connection@localhost +0000000 -> +1234567 'Test sms' Task ID:
1326c0f0-e8fd-4ddd-97d2-68ff9401b112
INFO smp1: Submit accepted at HTTP_Server_1@localhost. Submit reference: 701dbf6a-30a4-4bd9-8409-
848fd68ce1a3 +0000000 -> +1234567 'Test sms' Task ID: 1326c0f0-e8fd-4ddd-97d2-68ff9401b112
```

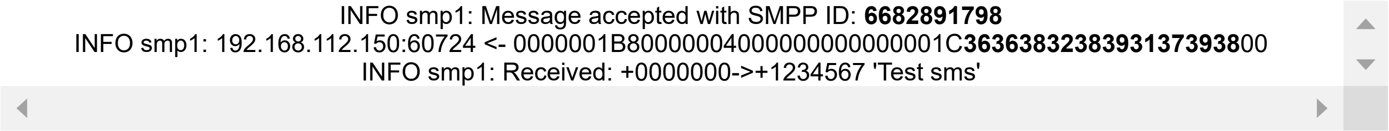
## Submit SM log / Submit request

The first line of the above log is the data the system received from your customer. Your customer submitted his SMS message using the SMPP SUBMIT_SM PDU request. Here is the byte data represented in HEX format:

```
INFO smp1: 192.168.112.150:60724 ->
000000370000000400000000000000001C000101303030303030300001013132333435363700000000100000100000008546
```

# Submit SM log / Submit Response

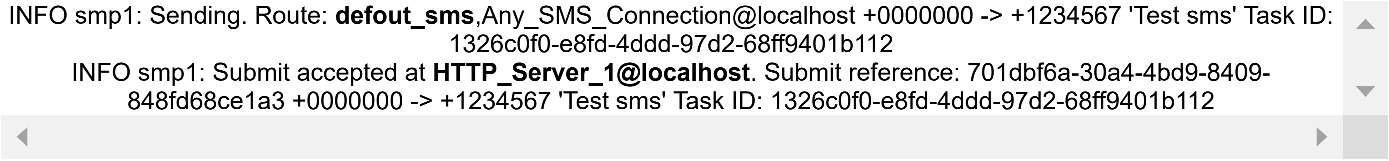
The next three lines in the log are related to the response. Your system assigns an SMPP ID to the message. This is ID is **6682891798** in our case. This ID will be used to reference this message when a delivery report comes in. Then it sends a response to your customer in the form of a SUBMIT_SM_RESP PDU. This PDU contains the assigned ID. Your customer can store this ID for later reference.



```
INFO smp1: Message accepted with SMPP ID: 6682891798
INFO smp1: 192.168.112.150:60724 <- 0000001B80000004000000000000001C3636383238393137393800
INFO smp1: Received: +0000000->+1234567 'Test sms'
```

# Submit SM log / Routing log

The next two lines are related to message routing. The system gives you information on which route was used to forward the message to the mobile network. After routing completes the system will also log what happened to the message at the destination connection. In our case you will see that the **default_sms** route was used, and the message was sent to the mobile network through the **HTTP_Server_1@localhost** connection.

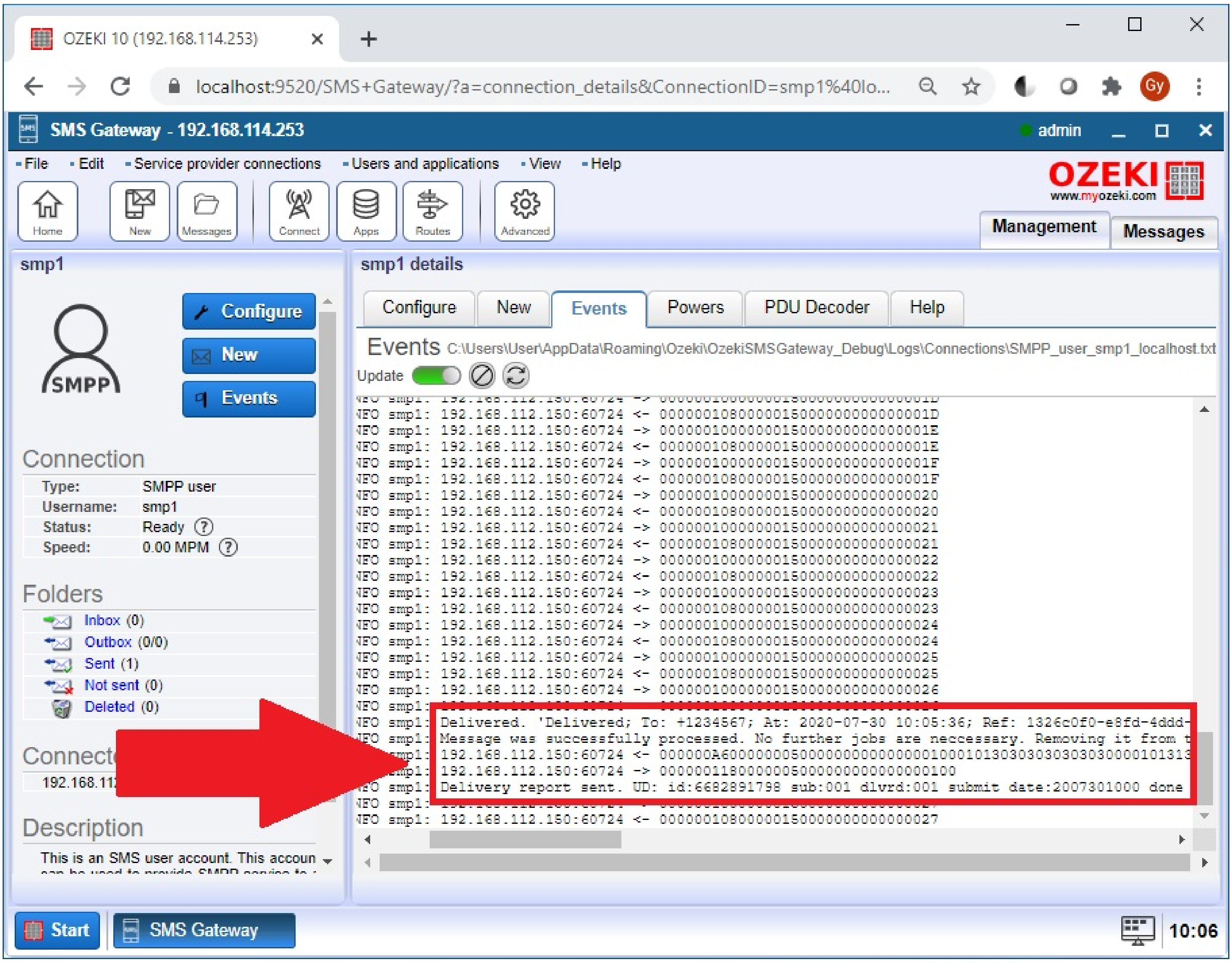


```
INFO smp1: Sending. Route: defout_sms,Any_SMS_Connection@localhost +0000000 -> +1234567 'Test sms' Task ID:
1326c0f0-e8fd-4ddd-97d2-68ff9401b112
INFO smp1: Submit accepted at HTTP_Server_1@localhost. Submit reference: 701dbf6a-30a4-4bd9-8409-
848fd68ce1a3 +0000000 -> +1234567 'Test sms' Task ID: 1326c0f0-e8fd-4ddd-97d2-68ff9401b112
```

If you want more detailed information on what happened to the message, you might want to open the mobile network connection's log and see the delivery events corresponding to the message in that log file. In this case you would open the log of the **HTTP_Server_1@localhost** connection.

# Find the SMPP delivery report

After some minutes, when the mobile network delivery the SMS to the recipient's phone, a delivery report will be returned to your system. Your system will forward this delivery report to the customer using an SMPP_DELIVER_SM request. This delivery report will contain the original SMPP ID of the message. In our case it will be: **6682891798**. To find the corresponing delivery report log in your log file, search for this ID.



Connection

Type: SMPP user
Username: smp1
Status: Ready
Speed: 0.00 MPM

Folders

Inbox (0)
Outbox (0/0)
Sent (1)
Not sent (0)
Deleted (0)

Connect
192.168.11

Description

This is an SMS user account. This accoun

Events C:\Users\User\AppData\Roaming\Ozeki\OzekiSMSGateway_Debug\Logs\Connections\SMPP_user_smp1_localhost.txt

```
NFO smp1: 192.168.112.150:60724 <- 0000001080000015000000000000001D
NFO smp1: 192.168.112.150:60724 -> 0000001000000015000000000000001E
NFO smp1: 192.168.112.150:60724 <- 0000001080000015000000000000001E
NFO smp1: 192.168.112.150:60724 -> 0000001000000015000000000000001F
NFO smp1: 192.168.112.150:60724 <- 0000001080000015000000000000001F
NFO smp1: 192.168.112.150:60724 -> 00000010000000150000000000000020
NFO smp1: 192.168.112.150:60724 <- 00000010800000150000000000000020
NFO smp1: 192.168.112.150:60724 -> 00000010000000150000000000000021
NFO smp1: 192.168.112.150:60724 <- 00000010800000150000000000000021
NFO smp1: 192.168.112.150:60724 -> 00000010000000150000000000000022
NFO smp1: 192.168.112.150:60724 <- 00000010800000150000000000000022
NFO smp1: 192.168.112.150:60724 -> 00000010000000150000000000000023
NFO smp1: 192.168.112.150:60724 <- 00000010800000150000000000000023
NFO smp1: 192.168.112.150:60724 -> 00000010000000150000000000000024
NFO smp1: 192.168.112.150:60724 <- 00000010800000150000000000000024
NFO smp1: 192.168.112.150:60724 -> 00000010000000150000000000000025
NFO smp1: 192.168.112.150:60724 <- 00000010800000150000000000000025
NFO smp1: 192.168.112.150:60724 -> 00000010000000150000000000000026
NFO smp1:
NFO smp1: Delivered. 'Delivered; To: +1234567; At: 2020-07-30 10:05:36; Ref: 1326c0f0-e8fd-4ddd-
NFO smp1: Message was successfully processed. No further jobs are neccessary. Removing it from t
    mp1: 192.168.112.150:60724 <- 000000A6000000050000000000000000100010130303030303030000101313
    mp1: 192.168.112.150:60724 -> 00000011800000050000000000000000100
NFO smp1: Delivery report sent. UD: id:6682891798 sub:001 dlvrd:001 submit date:2007301000 done
NFO smp1:
NFO smp1: 192.168.112.150:60724 <- 00000010800000150000000000000027
```

Figure 1 - Delivery report log

# Delivery report log

The correspoding delivery log in this case contains 5 log entries. The first log entry prints information for you that states, that the message was delivered. The next line gives you information about which inbound route was used to forward the incoming delivery report to this user's account. The next two lines contain the communication between your system and the customer's system. You will see, that your system sends the SMPP Deliver_SM PDU to the customer, and the customer returns a response to acknowledge this request.

```
2020-07-30 10:05:36.674 INFO smp1: Delivered. 'Delivered; To: +1234567; At: 2020-07-30 10:05:36; Ref: 1326c0f0-e8fd-4ddd-97d2-68ff9401b112; Successful delivery at 30/07/2020 10:05:36' +0000000 -> +1234567 'Test sms' Task ID: 1326c0f0-e8fd-4ddd-97d2-68ff9401b112
2020-07-30 10:05:36.674 INFO smp1: Message was successfully processed. No further jobs are neccessary. Removing it from the Sent queue. Route: smp1@localhost->HTTP_Server_1@localhost (Move). Message: +0000000->+1234567 'Test sms' Task ID: 1326c0f0-e8fd-4ddd-97d2-68ff9401b112
2020-07-30 10:05:36.674 INFO smp1: 192.168.112.150:60724 <- 000000A6000000050000000000000001000101303030303030300001013132333435363700040000000000000030077696
2020-07-30 10:05:36.674 INFO smp1: 192.168.112.150:60724 -> 00000011800000050000000000000000100
2020-07-30 10:05:36.674 INFO smp1: Delivery report sent. UD: id:6682891798 sub:001 dlvrd:001 submit date:2007301000 done date:2007301005 stat:DELIVRD err:000 text:Delivery success.
```

# How to secure your SMPP server using SSL

***When a secure TCP/IP connection is used for SMPP connections, the network traffic between your system and the SMPP clients will be encrypted. This will prevent unwanted parties access to the data that goes through the network. The two communicating endpoints will also verify each other using certificates. This verification procedure prevents man in the middle attacks. This guide explains how you can create an SSL certificate using the security app of Ozeki 10, and how you can use this SSL certificate to setup a secure SMPP service, that operates using SSL/TLS.***

## Create an SSL certificate

You can create an SSL certificate by opening the Security app of Ozeki 10 from the **Ozeki 10 Start** menu.

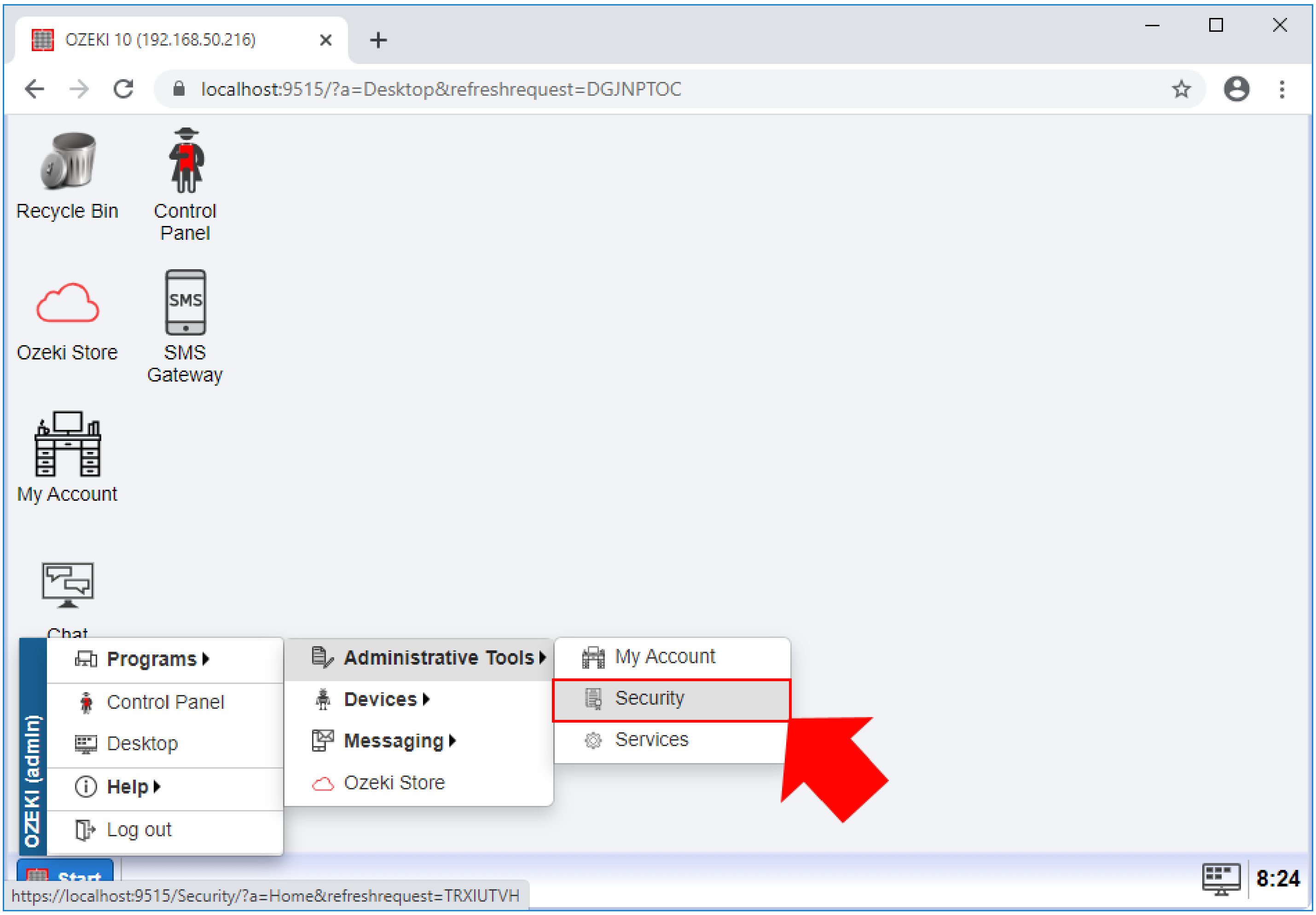
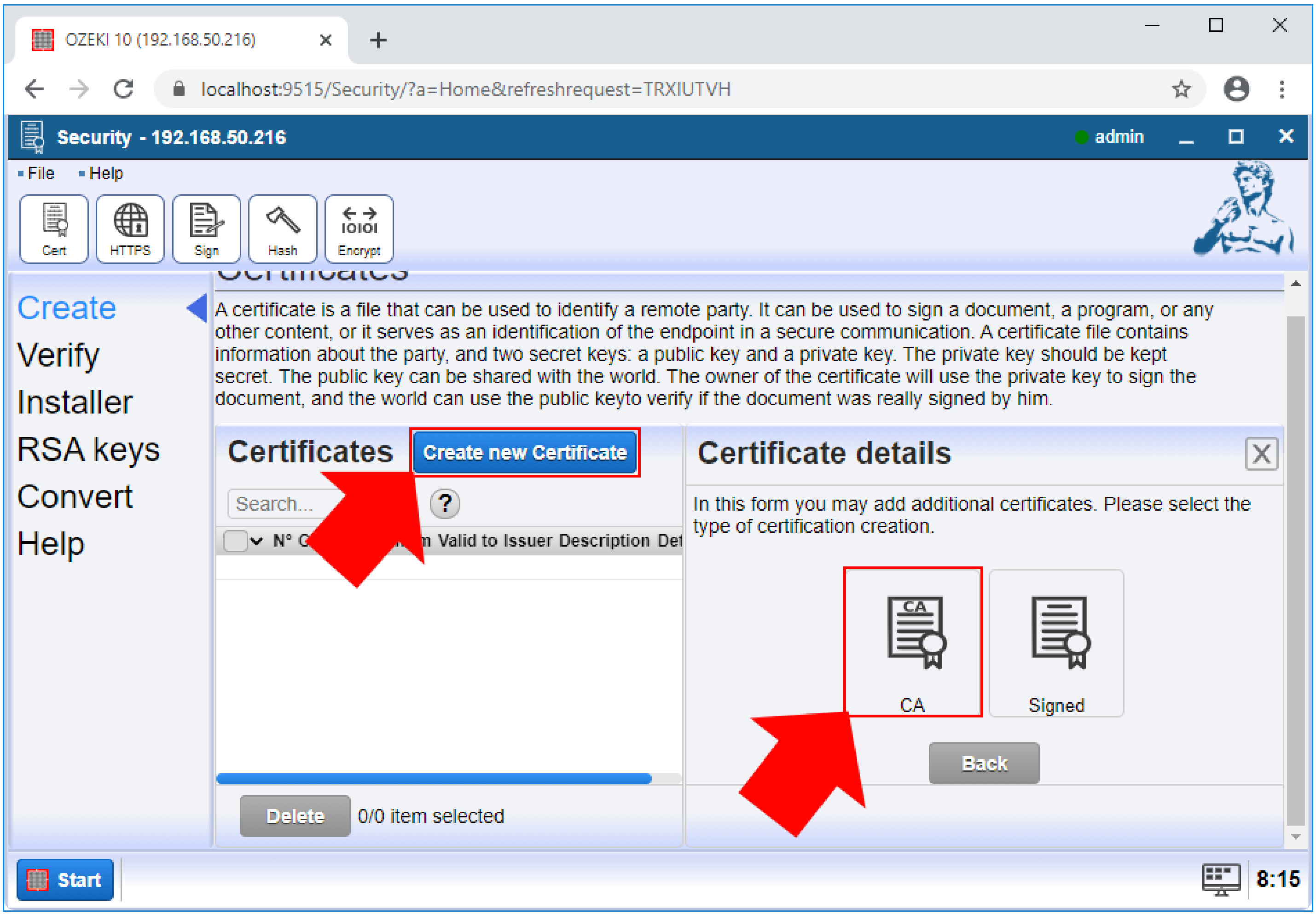


Figure 1 - Open Security app

## Create an SSL CA certificate

The term CA certificate stands for "Certificate Authority" certificate. A Certificate Athority is the top level organization that can issue certificates and can sign these certificates using it's secret private key. With Ozeki 10 you can become your own Certificate Authority, and you can issue certificates to your customers. You will than be able to verify incoming SSL connections and see if they use the certificate you have assigned to them. You can create your CA certificate in the secuirty app.



A certificate is a file that can be used to identify a remote party. It can be used to sign a document, a program, or any other content, or it serves as an identification of the endpoint in a secure communication. A certificate file contains information about the party, and two secret keys: a public key and a private key. The private key should be kept secret. The public key can be shared with the world. The owner of the certificate will use the private key to sign the document, and the world can use the public keyto verify if the document was really signed by him.

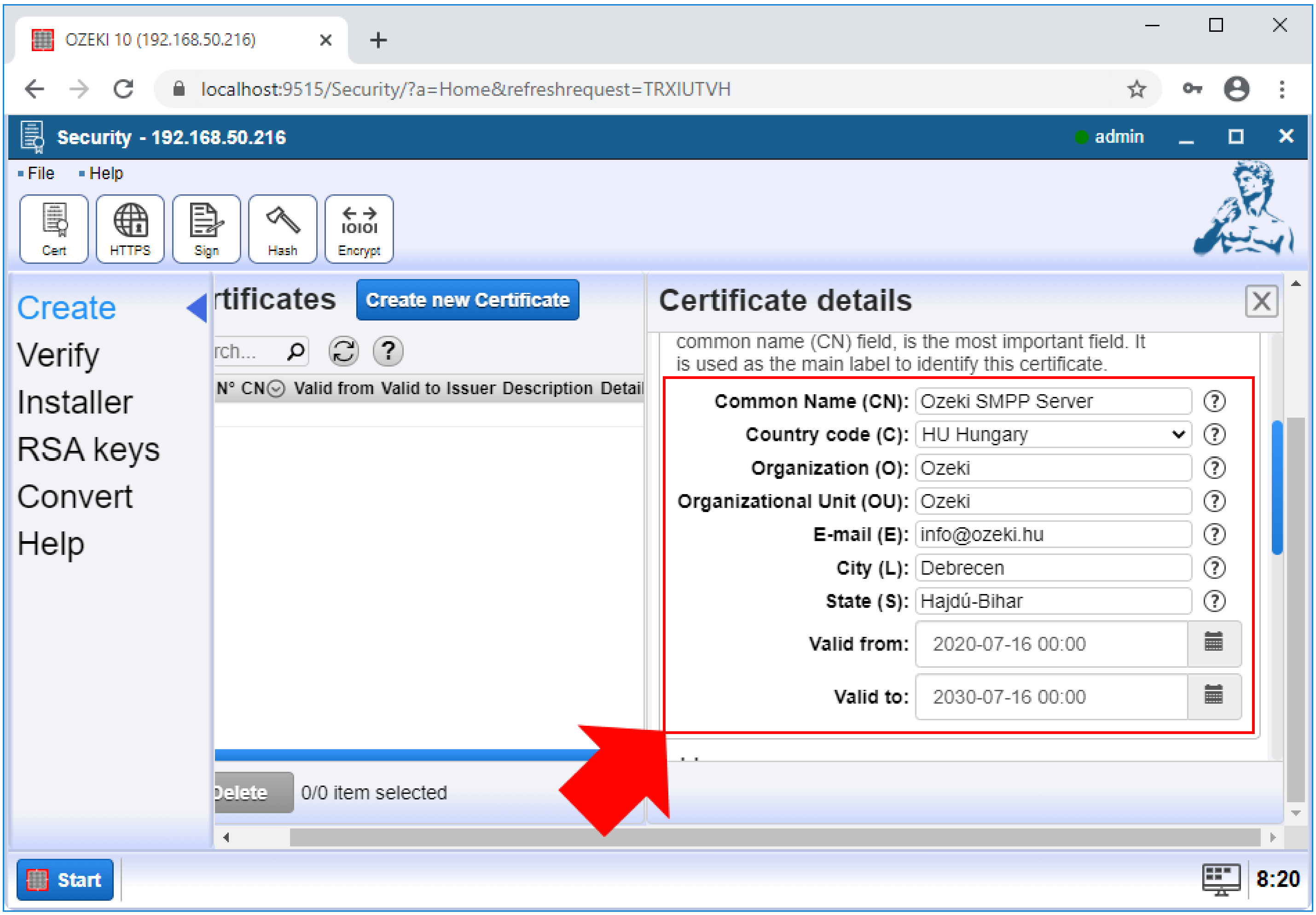
Certificate details

In this form you may add additional certificates. Please select the type of certification creation.

0/0 item selected

Figure 2 - Create CA certificate

Specify the certificate details and provide the password for the created certificate. Then click OK.

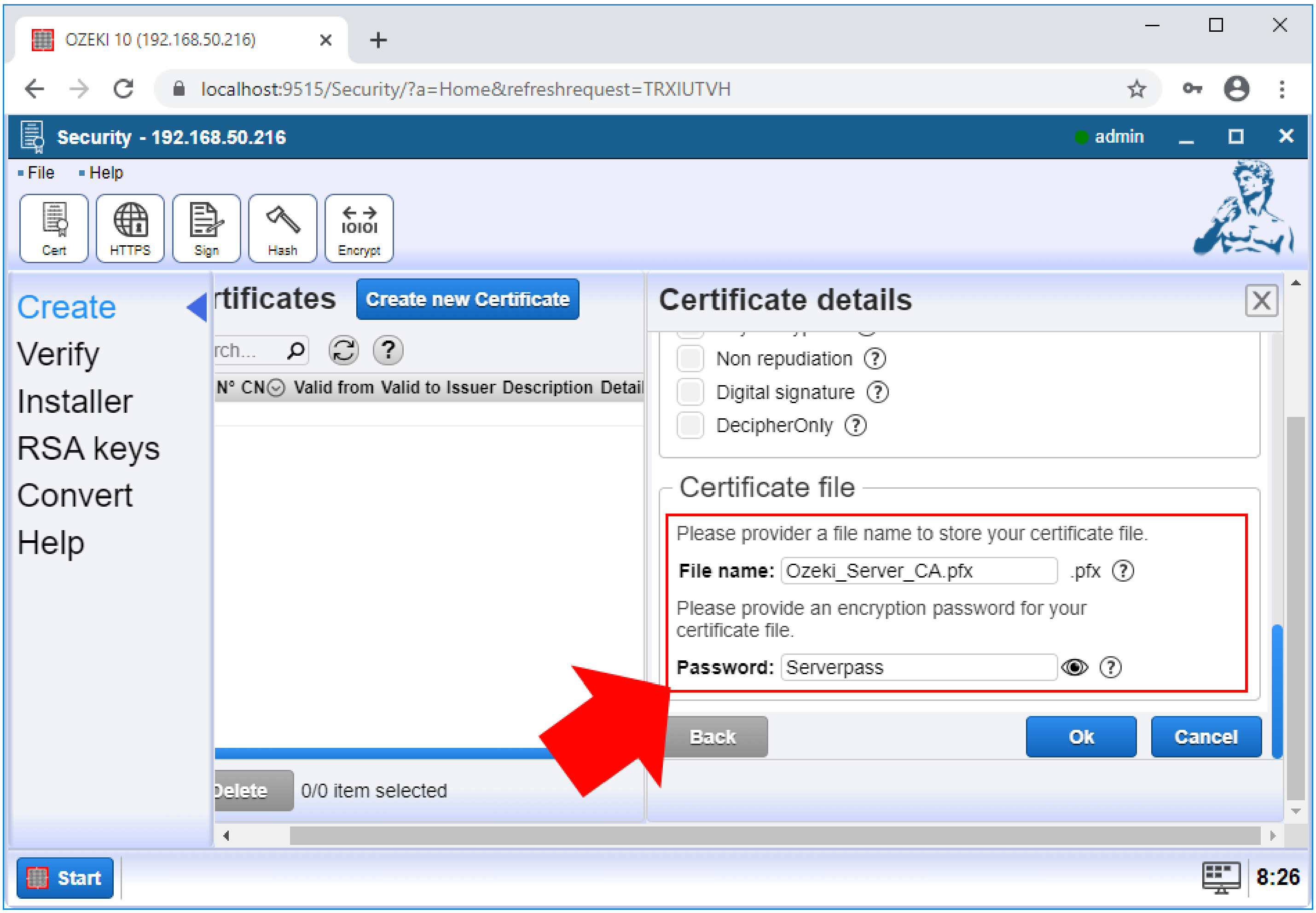

OZEKI 10 (192.168.50.216)
localhost:9515/Security/?a=Home&refreshrequest=TRXIUTVH
Security - 192.168.50.216
admin
File
Help
Cert
HTTPS
Sign
Hash
Encrypt
Create
Verify
Installer
RSA keys
Convert
Help
rtificates
Create new Certificate
rch...
N° CN Valid from Valid to Issuer Description Detail
Delete
0/0 item selected
Start
8:20

Certificate details

common name (CN) field, is the most important field. It is used as the main label to identify this certificate.

| | |
|---|---|
| Common Name (CN): | Ozeki SMPP Server |
| Country code (C): | HU Hungary |
| Organization (O): | Ozeki |
| Organizational Unit (OU): | Ozeki |
| E-mail (E): | info@ozeki.hu |
| City (L): | Debrecen |
| State (S): | Hajdú-Bihar |
| Valid from: | 2020-07-16 00:00 |
| Valid to: | 2030-07-16 00:00 |

Figure 3 - Provide certificate details



Certificate details

□ Non repudiation
□ Digital signature
□ DecipherOnly

Certificate file

Please provider a file name to store your certificate file.

File name: Ozeki_Server_CA.pfx .pfx

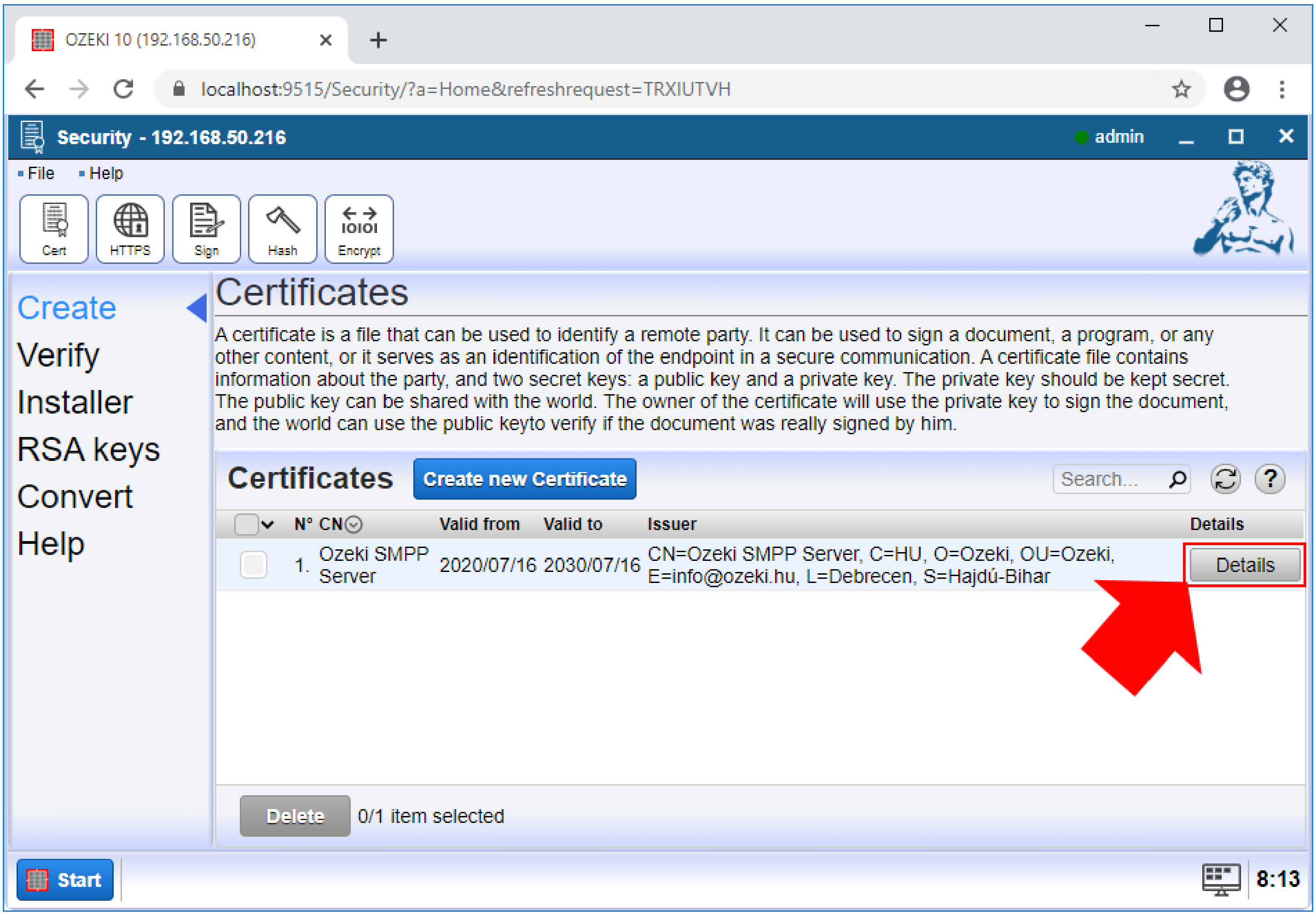
Please provide an encryption password for your certificate file.

Password: Serverpass

Figure 4 - Provide certificate file details

# Download the created SSL certificate

You can download the created certificate from the certificate list of the Ozeki 10 security app.

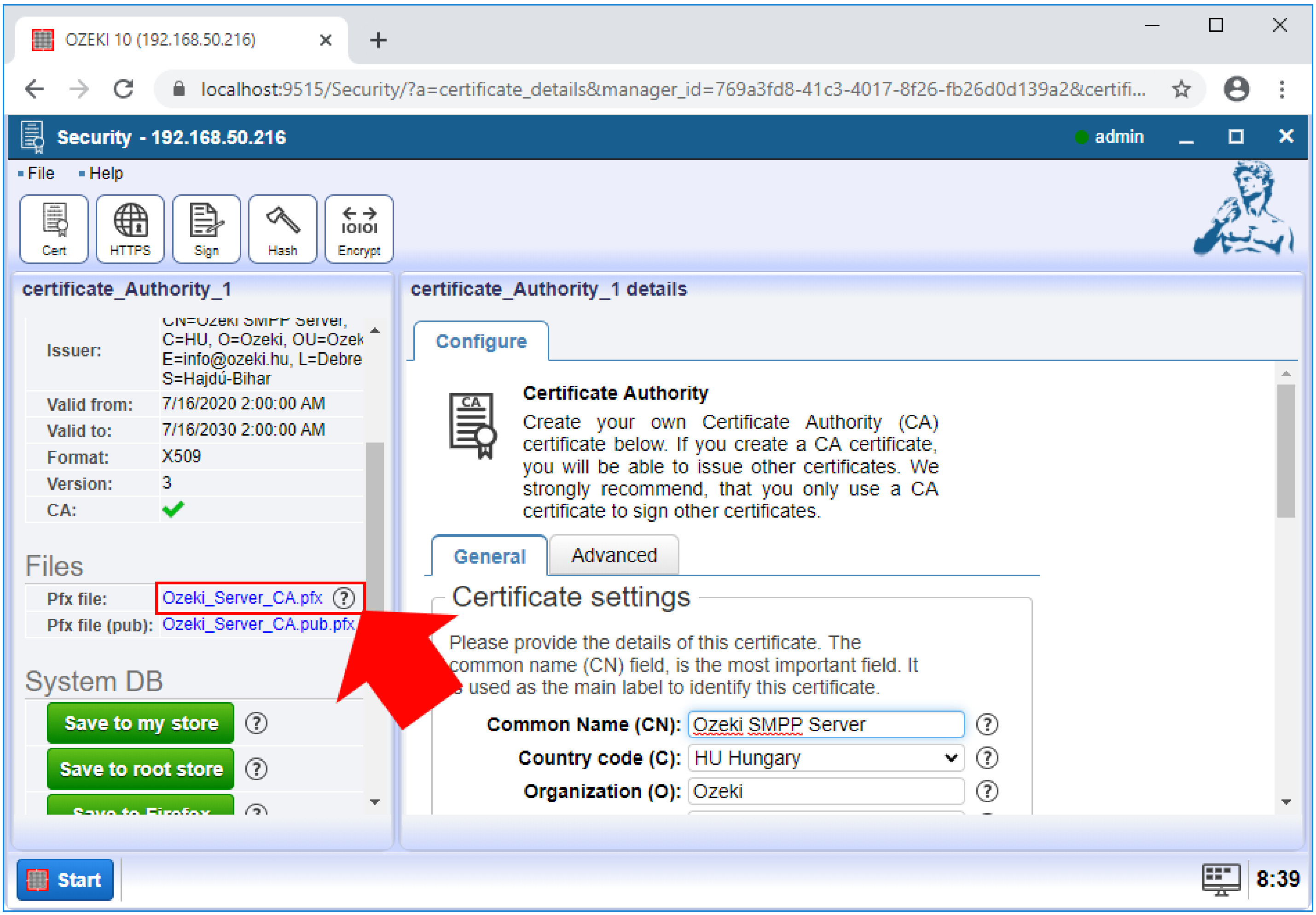


A certificate is a file that can be used to identify a remote party. It can be used to sign a document, a program, or any other content, or it serves as an identification of the endpoint in a secure communication. A certificate file contains information about the party, and two secret keys: a public key and a private key. The private key should be kept secret. The public key can be shared with the world. The owner of the certificate will use the private key to sign the document, and the world can use the public keyto verify if the document was really signed by him.

| N° | CN | Valid from | Valid to | Issuer | Details |
|---|---|---|---|---|---|
| 1. | Ozeki SMPP Server | 2020/07/16 | 2030/07/16 | CN=Ozeki SMPP Server, C=HU, O=Ozeki, OU=Ozeki, E=info@ozeki.hu, L=Debrecen, S=Hajdú-Bihar | Details |

0/1 item selected

Figure 5 - Select details of certificate



certificate_Authority_1

Issuer: CN=Ozeki SMPP Server, C=HU, O=Ozeki, OU=Ozek E=info@ozeki.hu, L=Debre S=Hajdú-Bihar

Valid from: 7/16/2020 2:00:00 AM

Valid to: 7/16/2030 2:00:00 AM

Format: X509

Version: 3

CA: ✓

Files

Pfx file: Ozeki_Server_CA.pfx

Pfx file (pub): Ozeki_Server_CA.pub.pfx

System DB

certificate_Authority_1 details

Certificate Authority

Create your own Certificate Authority (CA) certificate below. If you create a CA certificate, you will be able to issue other certificates. We strongly recommend, that you only use a CA certificate to sign other certificates.

Certificate settings

Please provide the details of this certificate. The common name (CN) field, is the most important field. It s used as the main label to identify this certificate.

Common Name (CN): Ozeki SMPP Server

Country code (C): HU Hungary

Organization (O): Ozeki

OZEKI 10 (192.168.50.216)
localhost:9515/Security/?a=certificate_details&manager_id=769a3fd8-41c3-4017-8f26-fb26d0d139a2&certifi...
Security - 192.168.50.216
admin
File
Help
Cert
HTTPS
Sign
Hash
Encrypt
Save to my store
Save to root store
Save to Firefox
Configure
General
Advanced
Start
8:39

Figure 6 - Download certificate

# Install the created SSL certificate into your SMPP server

To use the created SSL certificate to secure your SMPP connection, it must be installed. To do this open Advanced page form the management screen of the SMS gateway.

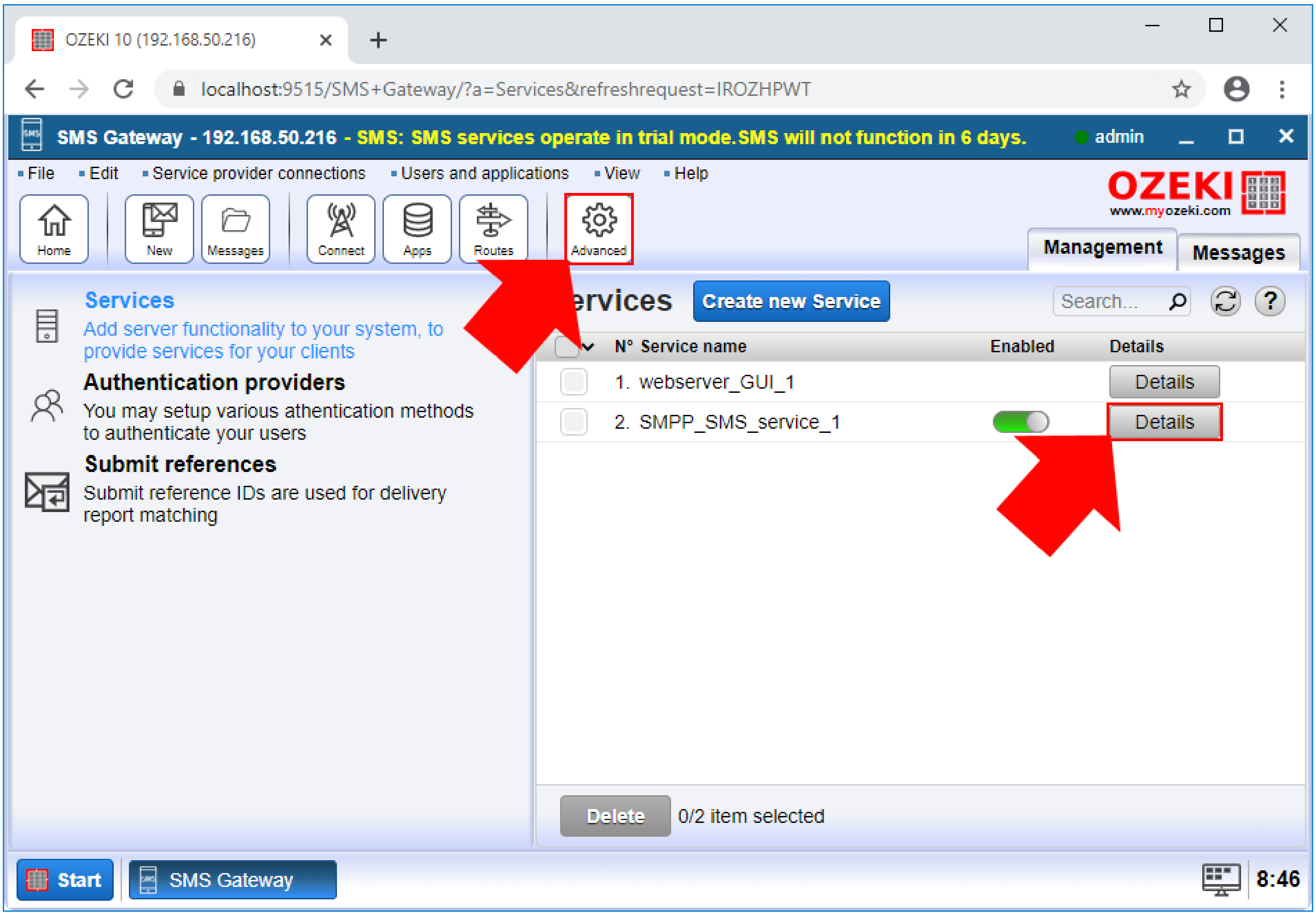
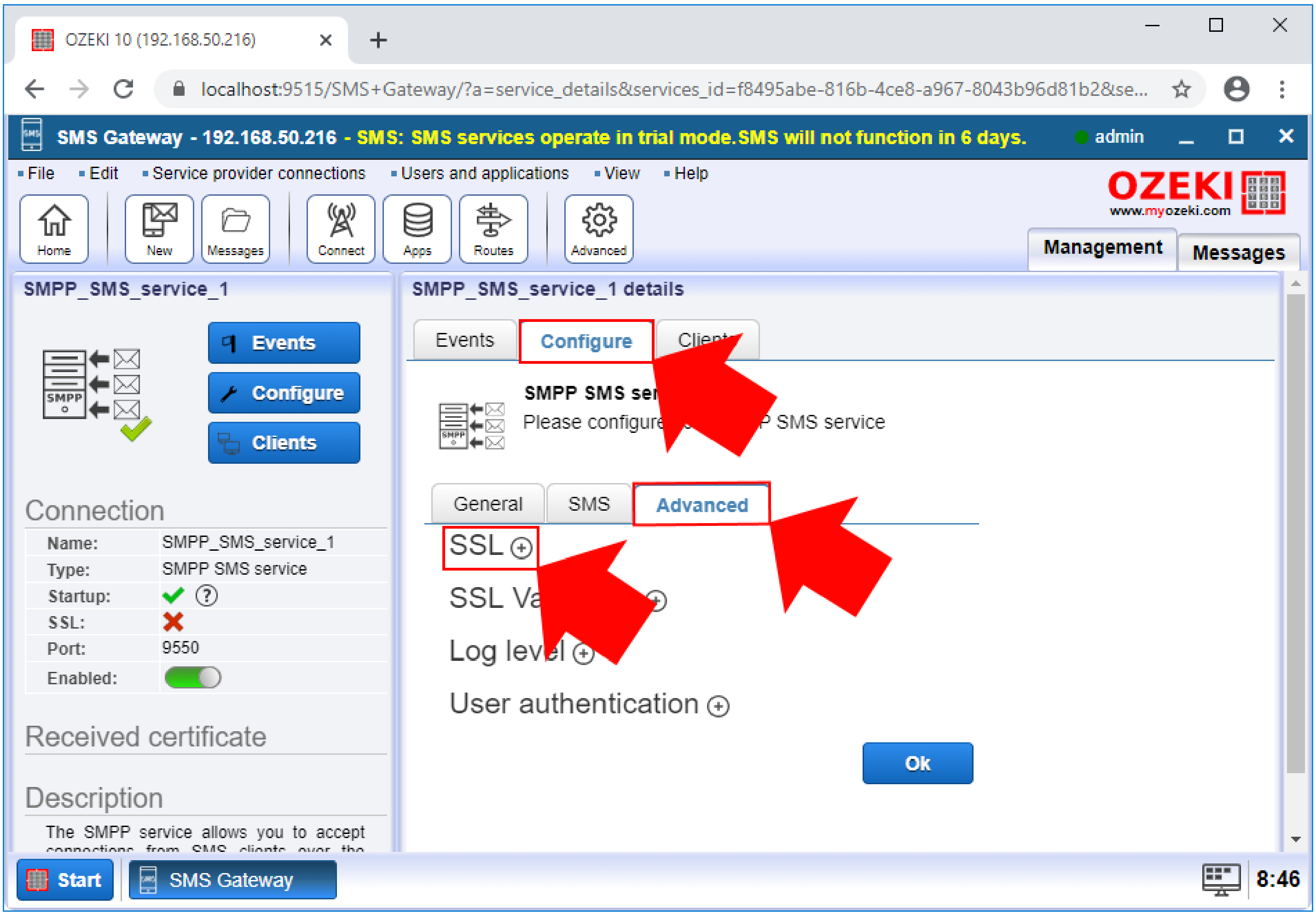

OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=Services&refreshrequest=IROZHPWT
SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode.SMS will not function in 6 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
OZEKI
www.myozeki.com
Management
Messages
Services
Add server functionality to your system, to provide services for your clients
Authentication providers
You may setup various athentication methods to authenticate your users
Submit references
Submit reference IDs are used for delivery report matching
ervices
Create new Service
Search...
N° Service name
Enabled
Details
1. webserver_GUI_1
Details
2. SMPP_SMS_service_1
Details
Delete
0/2 item selected
Start
SMS Gateway
8:46

Figure 7 - Open Advanced menu

Open the SMPP Service's security option.


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=service_details&services_id=f8495abe-816b-4ce8-a967-8043b96d81b2&se...
SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode.SMS will not function in 6 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
SMPP_SMS_service_1
Events
Configure
Clients
SMPP_SMS_service_1 details
Events
Configure
Clients
General
SMS
Advanced
SSL
SSL Va
Log level
User authentication
Ok
Start
SMS Gateway
8:46

SMPP SMS ser

Please configure ... P SMS service

Connection

| | |
|---|---|
| Name: | SMPP_SMS_service_1 |
| Type: | SMPP SMS service |
| Startup: | ✓ |
| SSL: | ✗ |
| Port: | 9550 |
| Enabled: | ● |

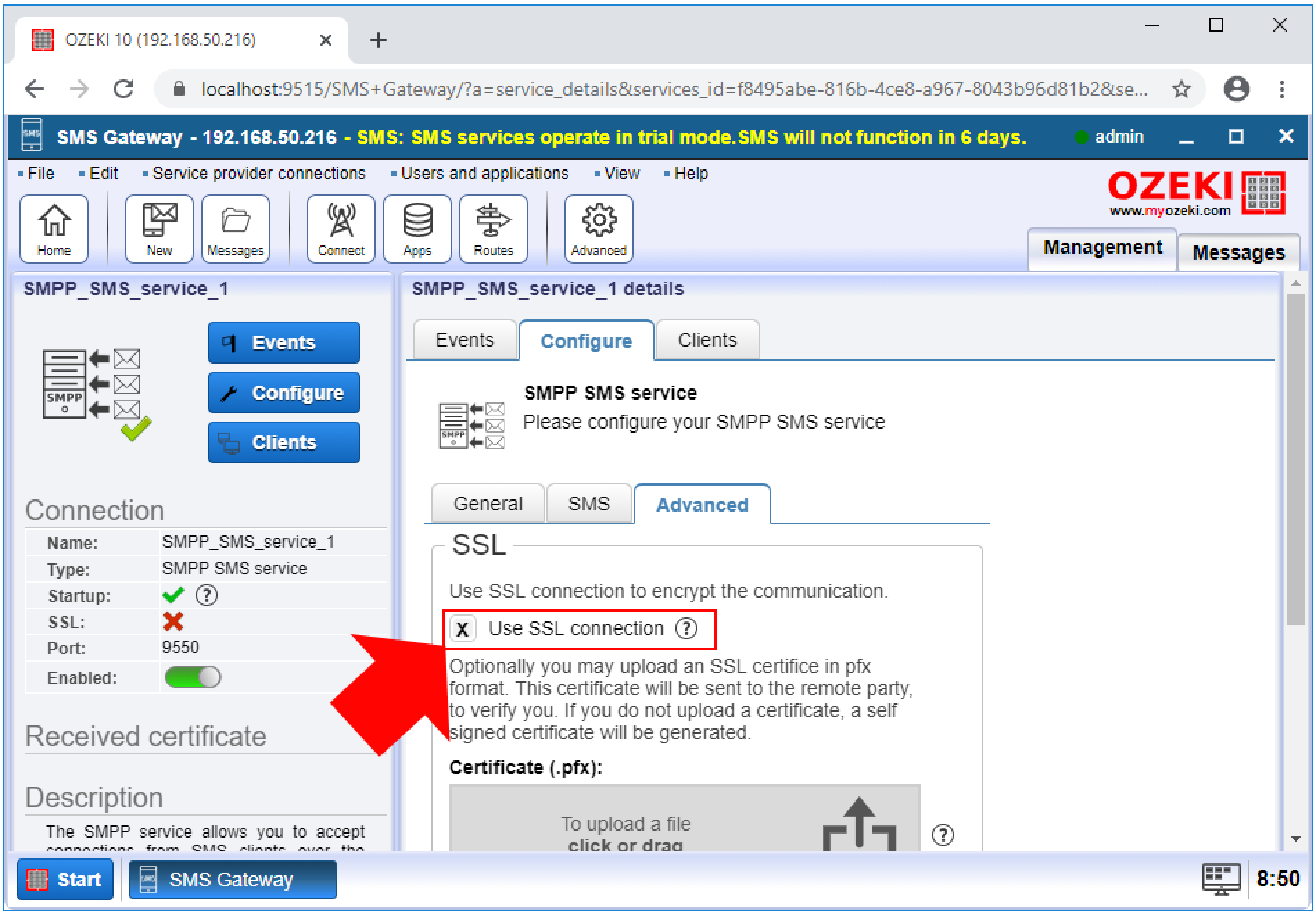
Received certificate

Description

The SMPP service allows you to accept connections from SMS clients over the

Figure 8 - Open security option

Enable SSL connection



SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode.SMS will not function in 6 days. admin

SMPP_SMS_service_1

Connection

| | |
|---|---|
| Name: | SMPP_SMS_service_1 |
| Type: | SMPP SMS service |
| Startup: | ✓ ? |
| SSL: | ✗ |
| Port: | 9550 |
| Enabled: | ● |

Received certificate

Description

The SMPP service allows you to accept connections from SMS clients over the

SMPP_SMS_service_1 details

SMPP SMS service

Please configure your SMPP SMS service

SSL

Use SSL connection to encrypt the communication.

☒ Use SSL connection ?

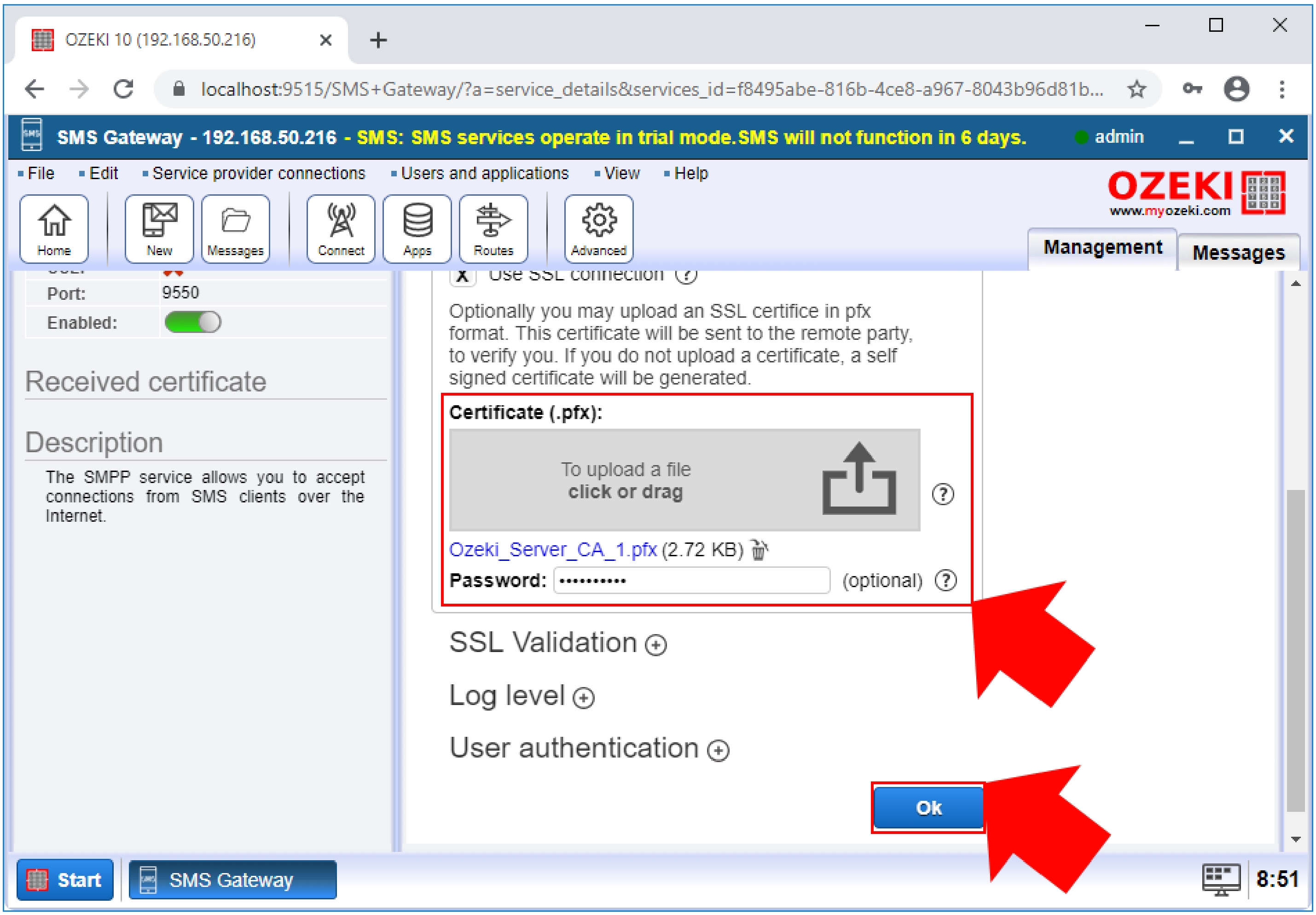
Optionally you may upload an SSL certifice in pfx format. This certificate will be sent to the remote party, to verify you. If you do not upload a certificate, a self signed certificate will be generated.

Certificate (.pfx):

To upload a file click or drag ?

Figure 9 - Enable SSL connection

Upload the certificate and provide the certificate's password.



OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=service_details&services_id=f8495abe-816b-4ce8-a967-8043b96d81b...
SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode.SMS will not function in 6 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Start
SMS Gateway
8:51

Port: 9550

Enabled:

Received certificate

Description

The SMPP service allows you to accept connections from SMS clients over the Internet.

☒ Use SSL connection

Optionally you may upload an SSL certificate in pfx format. This certificate will be sent to the remote party, to verify you. If you do not upload a certificate, a self signed certificate will be generated.

Certificate (.pfx):

To upload a file
click or drag

Ozeki_Server_CA_1.pfx (2.72 KB)

Password: •••••••••• (optional)

SSL Validation

Log level

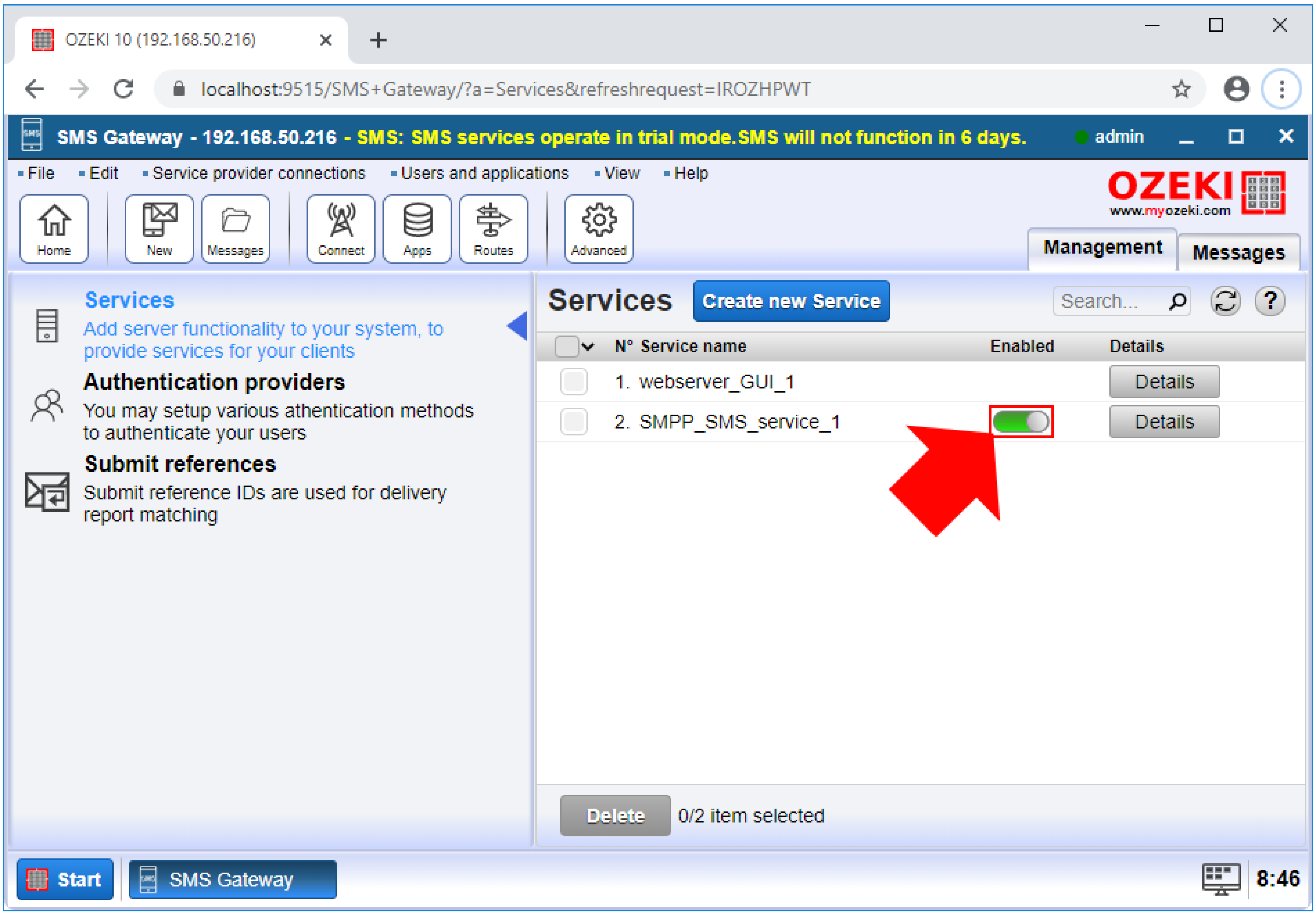
User authentication

Ok

Figure 10 - Upload certificate

## Restart your SMPP server to let the changes take effect

Enable SMPP service.


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=Services&refreshrequest=IROZHPWT
SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode.SMS will not function in 6 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Start
SMS Gateway
8:46

**Services**
Add server functionality to your system, to provide services for your clients

**Authentication providers**
You may setup various athentication methods to authenticate your users

**Submit references**
Submit reference IDs are used for delivery report matching

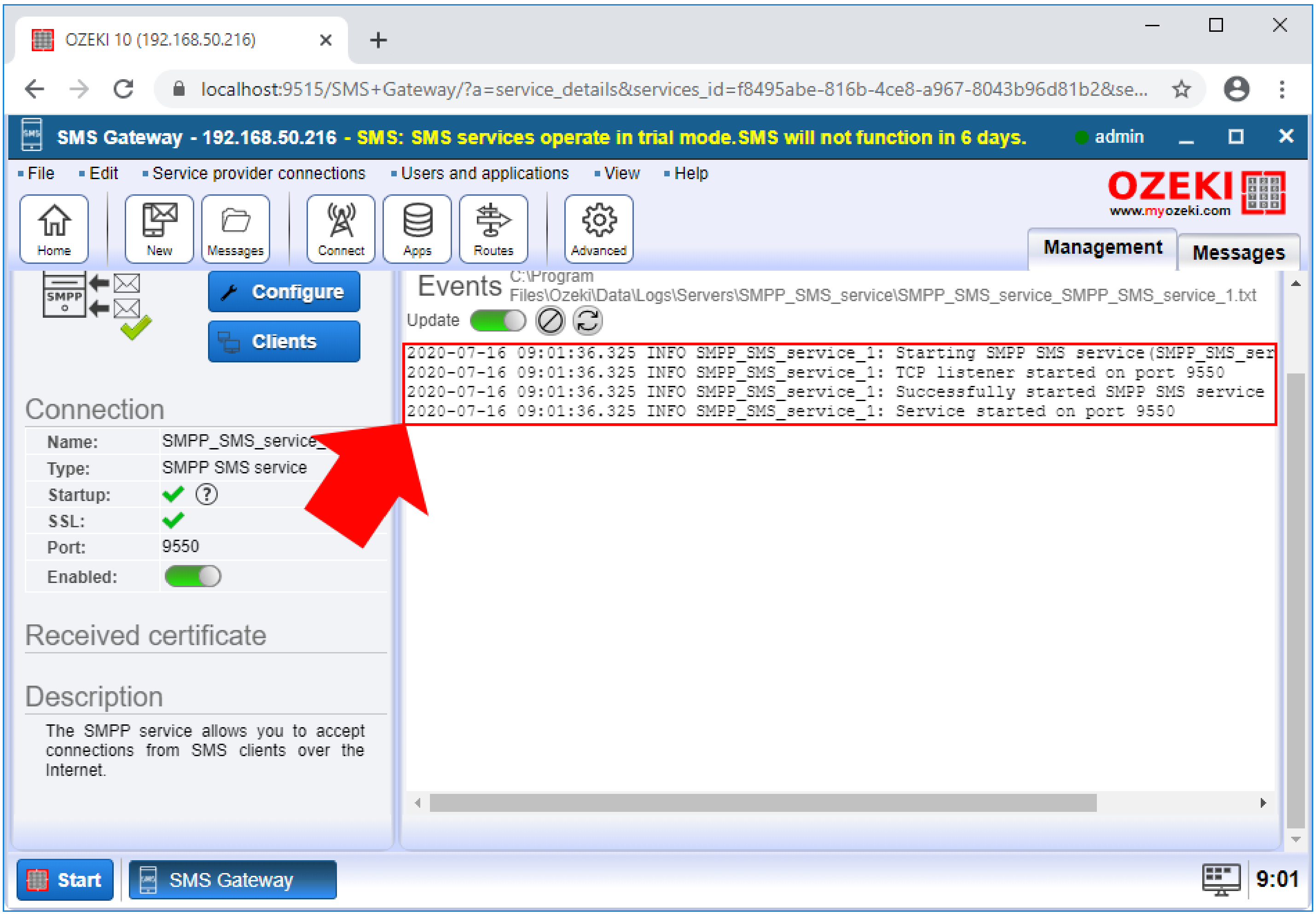
Services — Create new Service

| N° | Service name | Enabled | Details |
|---|---|---|---|
| 1. | webserver_GUI_1 | | Details |
| 2. | SMPP_SMS_service_1 | ● | Details |

Delete 0/2 item selected

Figure 11 - Enable service

Service successfully started.



OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=service_details&services_id=f8495abe-816b-4ce8-a967-8043b96d81b2&se...
SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode.SMS will not function in 6 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
OZEKI
www.myozeki.com
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
SMPP
Configure
Clients
Update
Start
SMS Gateway
9:01

Connection

Name: SMPP_SMS_service_

Type: SMPP SMS service

Startup: ✓ ?

SSL: ✓

Port: 9550

Enabled:

Received certificate

Description

The SMPP service allows you to accept connections from SMS clients over the Internet.

Events C:\Program Files\Ozeki\Data\Logs\Servers\SMPP_SMS_service\SMPP_SMS_service_SMPP_SMS_service_1.txt

```
2020-07-16 09:01:36.325 INFO SMPP_SMS_service_1: Starting SMPP SMS service(SMPP_SMS_ser
2020-07-16 09:01:36.325 INFO SMPP_SMS_service_1: TCP listener started on port 9550
2020-07-16 09:01:36.325 INFO SMPP_SMS_service_1: Successfully started SMPP SMS service
2020-07-16 09:01:36.325 INFO SMPP_SMS_service_1: Service started on port 9550
```

Figure 12 - Service started

# How to setup an SSL connection with CA verification

**This domumentation provides detailed instructions on how to configure SSL connection with CA verification SMS messaging using the Ozeki 10 - SMS Gateway software.**

## I. Create certificates

Open the **Security** applicaton from the start menu.

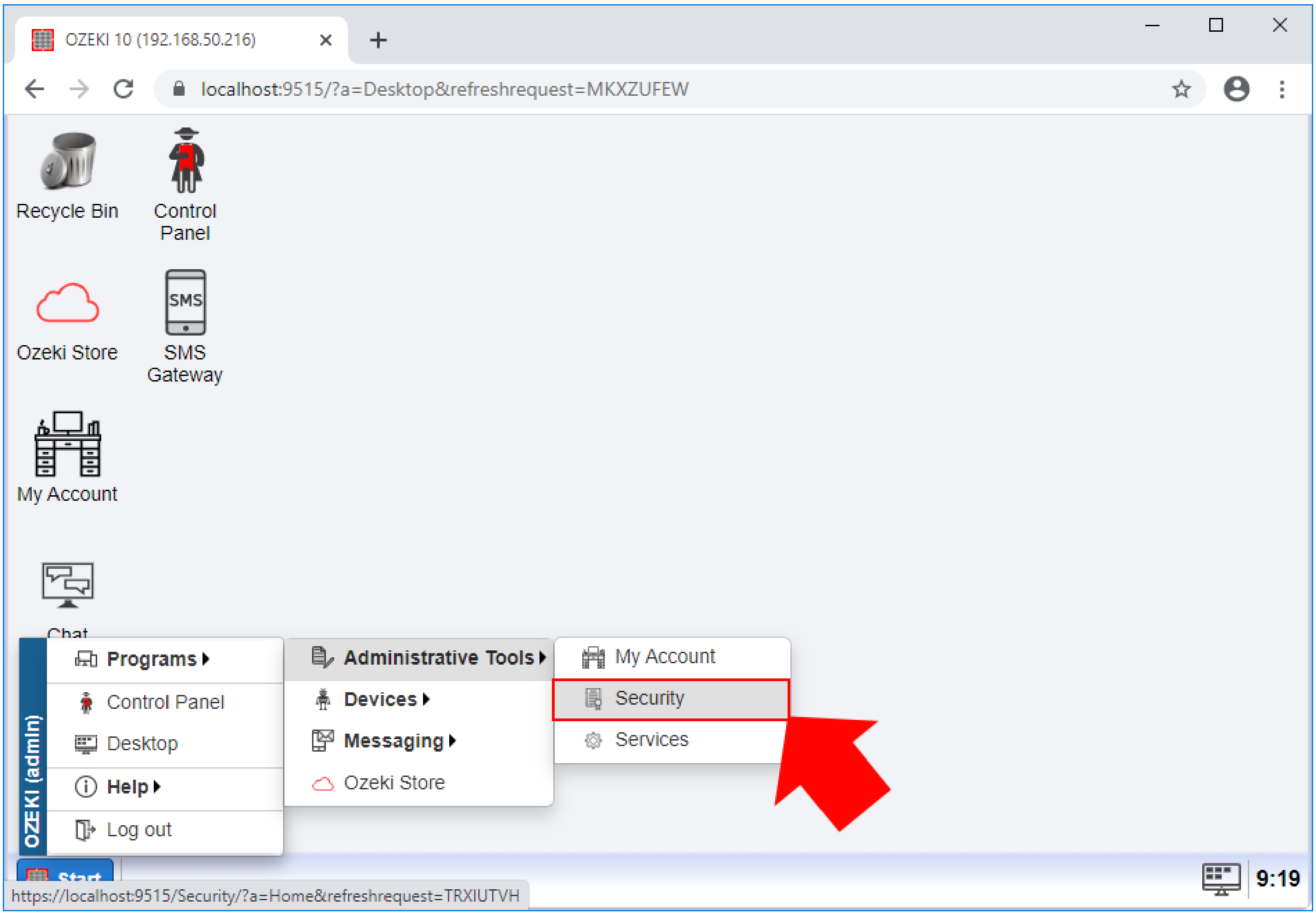
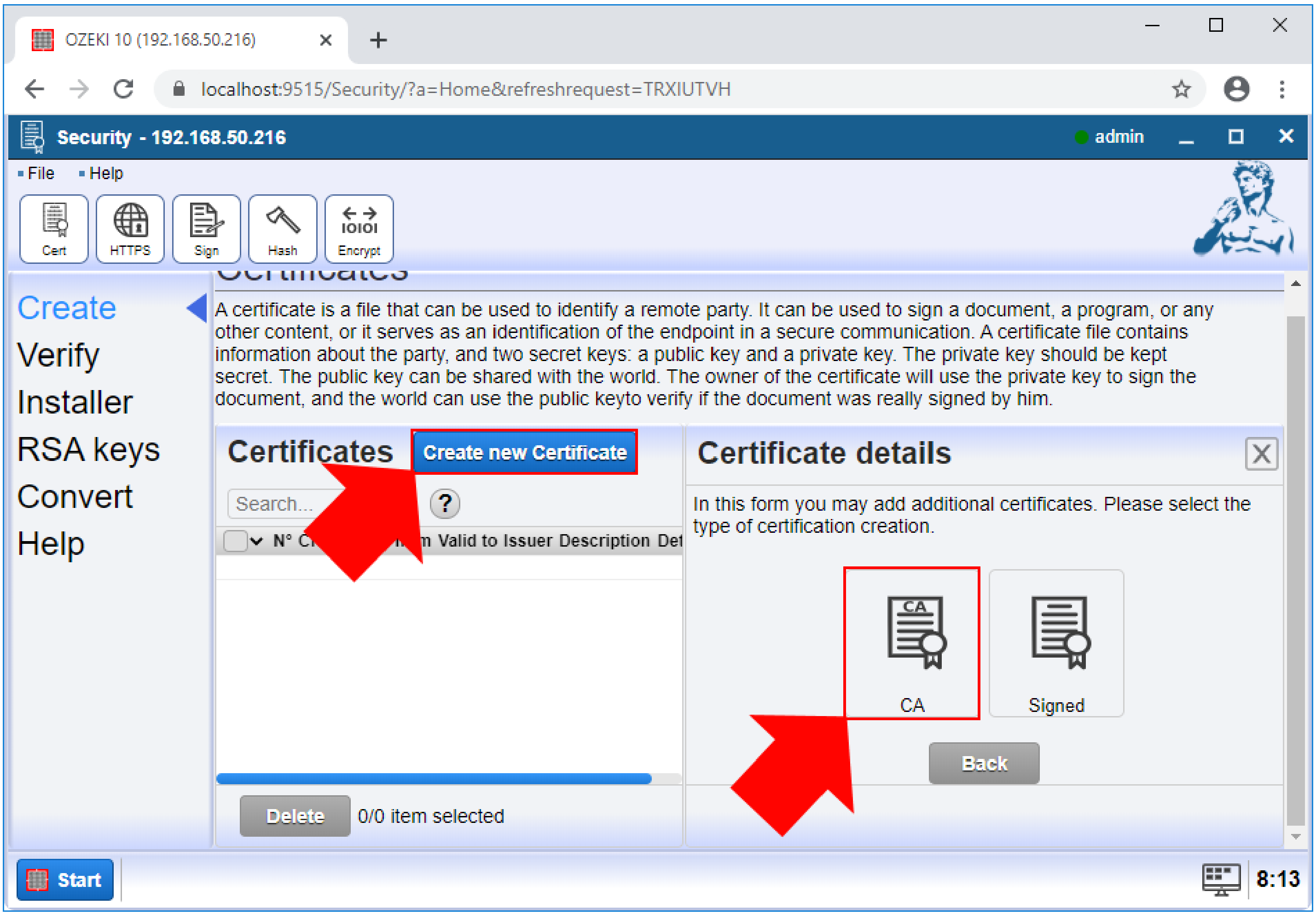


Figure 1 - Open Security app

To create a CA cerificate please click on **Create new Certificate** button and then **Create CA**.

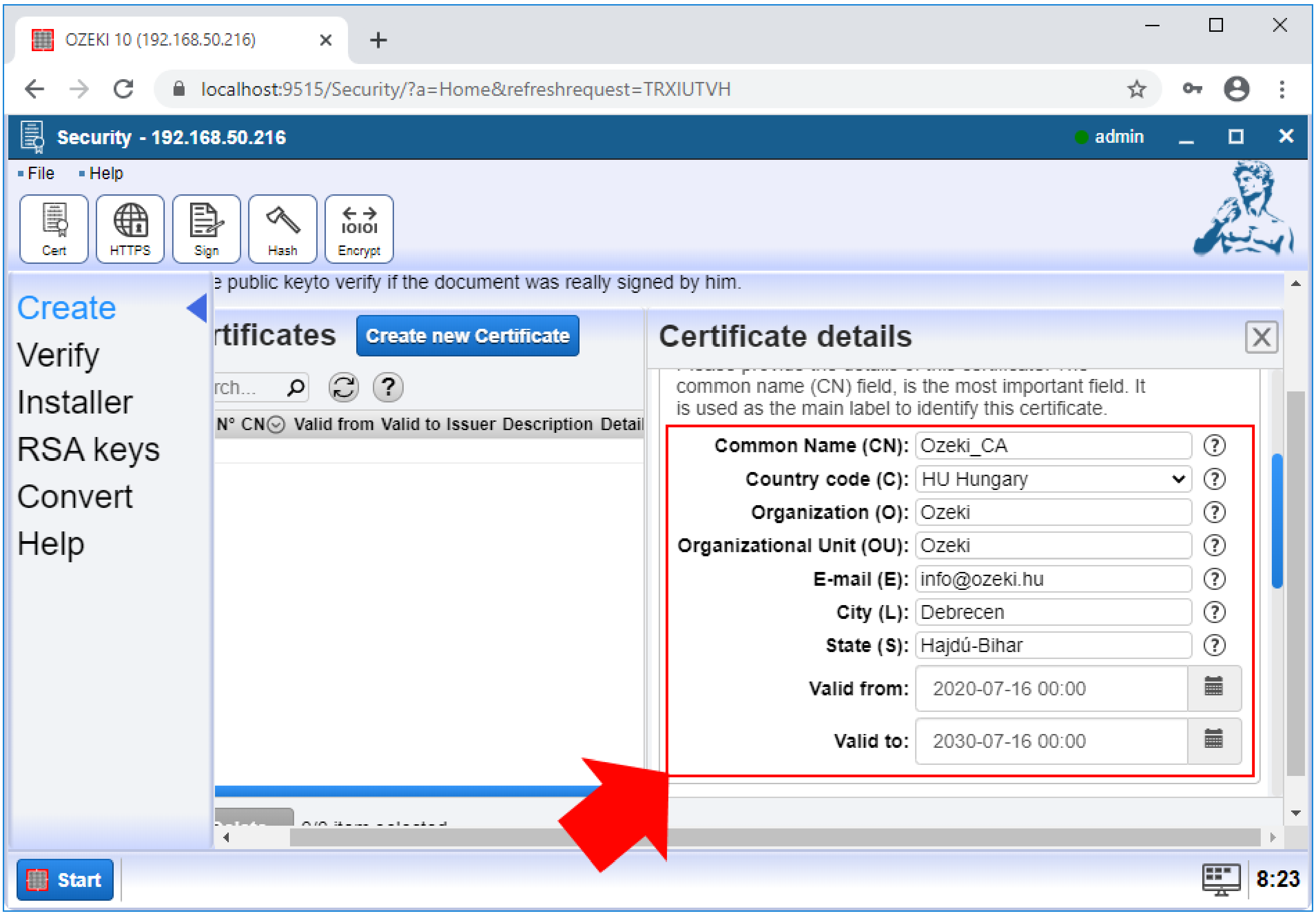


A certificate is a file that can be used to identify a remote party. It can be used to sign a document, a program, or any other content, or it serves as an identification of the endpoint in a secure communication. A certificate file contains information about the party, and two secret keys: a public key and a private key. The private key should be kept secret. The public key can be shared with the world. The owner of the certificate will use the private key to sign the document, and the world can use the public keyto verify if the document was really signed by him.

In this form you may add additional certificates. Please select the type of certification creation.

Figure 2 - Create CA certificate

On the general tab provide the details of the certificate.



OZEKI 10 (192.168.50.216)
localhost:9515/Security/?a=Home&refreshrequest=TRXIUTVH
Security - 192.168.50.216
admin
File
Help
Cert
HTTPS
Sign
Hash
Encrypt
Create
Verify
Installer
RSA keys
Convert
Help
rtificates
Create new Certificate
rch...
N° CN Valid from Valid to Issuer Description Detail
Start
8:23

e public keyto verify if the document was really signed by him.

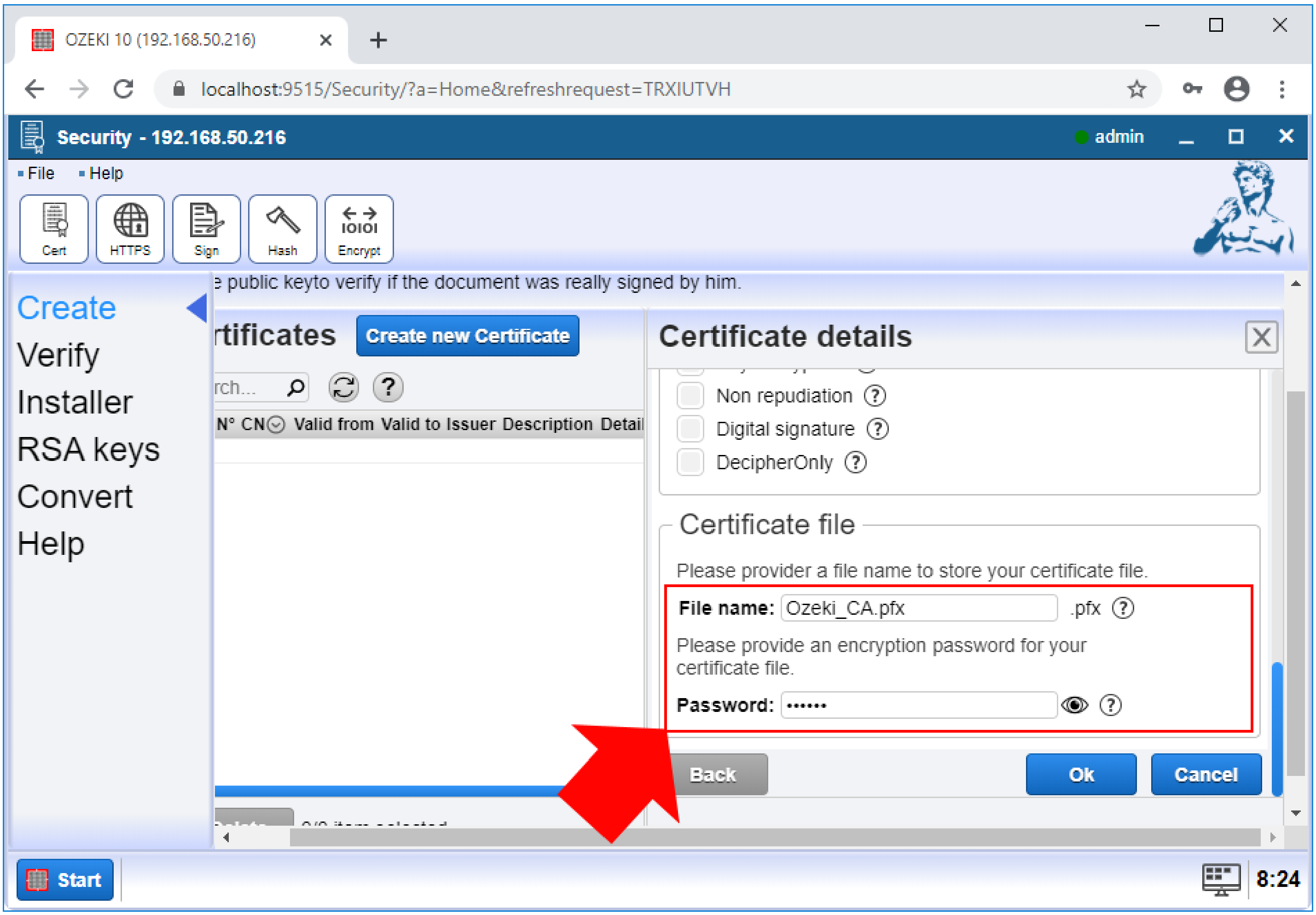
Certificate details

common name (CN) field, is the most important field. It is used as the main label to identify this certificate.

| | |
|---|---|
| Common Name (CN): | Ozeki_CA |
| Country code (C): | HU Hungary |
| Organization (O): | Ozeki |
| Organizational Unit (OU): | Ozeki |
| E-mail (E): | info@ozeki.hu |
| City (L): | Debrecen |
| State (S): | Hajdú-Bihar |
| Valid from: | 2020-07-16 00:00 |
| Valid to: | 2030-07-16 00:00 |

Figure 3 - Provide CA certificate details

Select the **usage** and specify the **file name** and the **password**.


OZEKI 10 (192.168.50.216)
localhost:9515/Security/?a=Home&refreshrequest=TRXIUTVH
Security - 192.168.50.216
admin
File
Help
Cert
HTTPS
Sign
Hash
Encrypt
Create
Verify
Installer
RSA keys
Convert
Help
e public keyto verify if the document was really signed by him.
rtificates
Create new Certificate
rch...
N° CN Valid from Valid to Issuer Description Detail
Start
8:24

Certificate details

□ Non repudiation ?
□ Digital signature ?
□ DecipherOnly ?

Certificate file

Please provider a file name to store your certificate file.

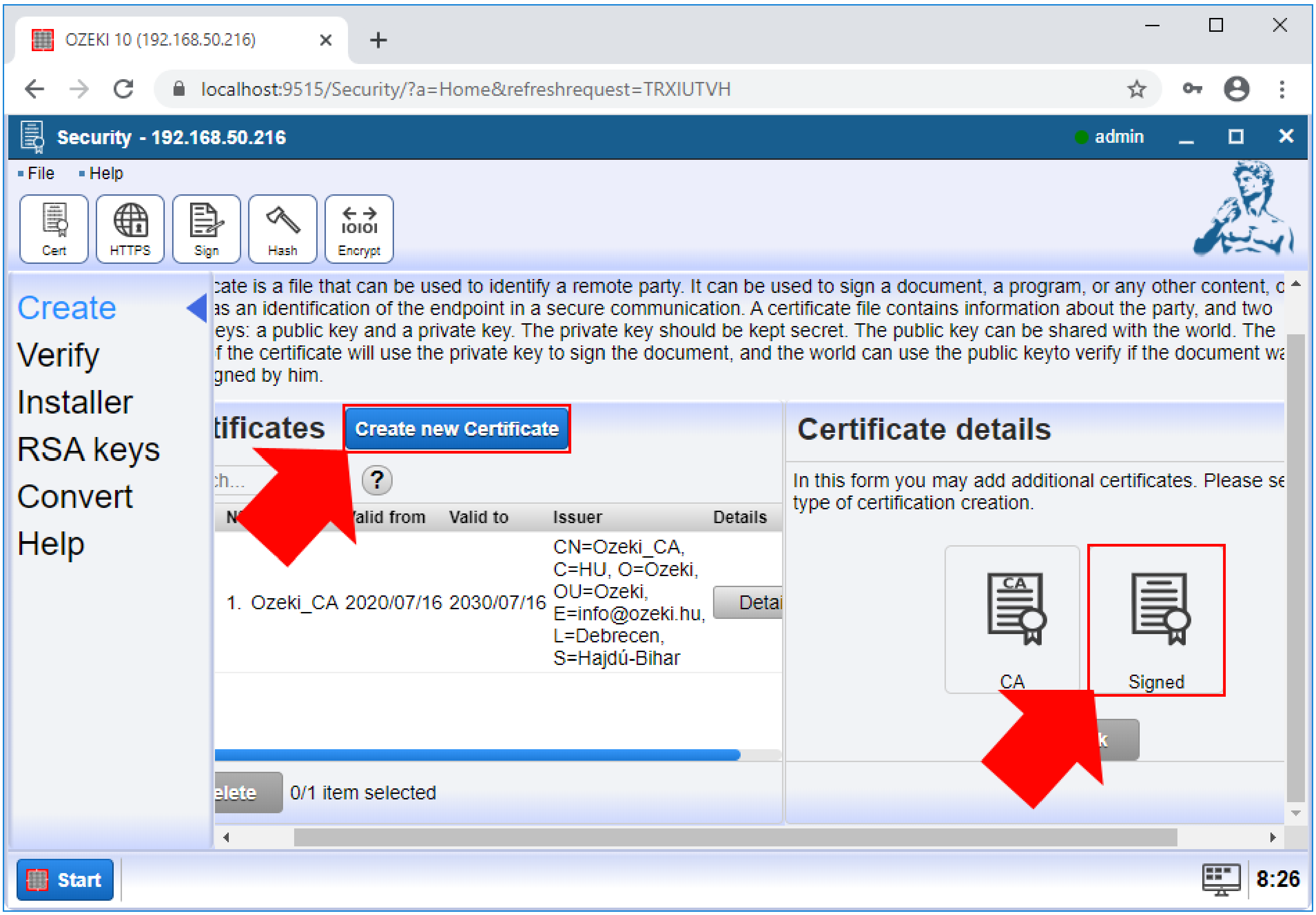
File name: Ozeki_CA.pfx .pfx ?

Please provide an encryption password for your certificate file.

Password: •••••• ?

Figure 4 - Provide usage, file name and password

After it you can create the signed certificates. Click on **Create new Certificate** button and then **Create Signed**.



cate is a file that can be used to identify a remote party. It can be used to sign a document, a program, or any other content, o as an identification of the endpoint in a secure communication. A certificate file contains information about the party, and two eys: a public key and a private key. The private key should be kept secret. The public key can be shared with the world. The f the certificate will use the private key to sign the document, and the world can use the public keyto verify if the document wa gned by him.

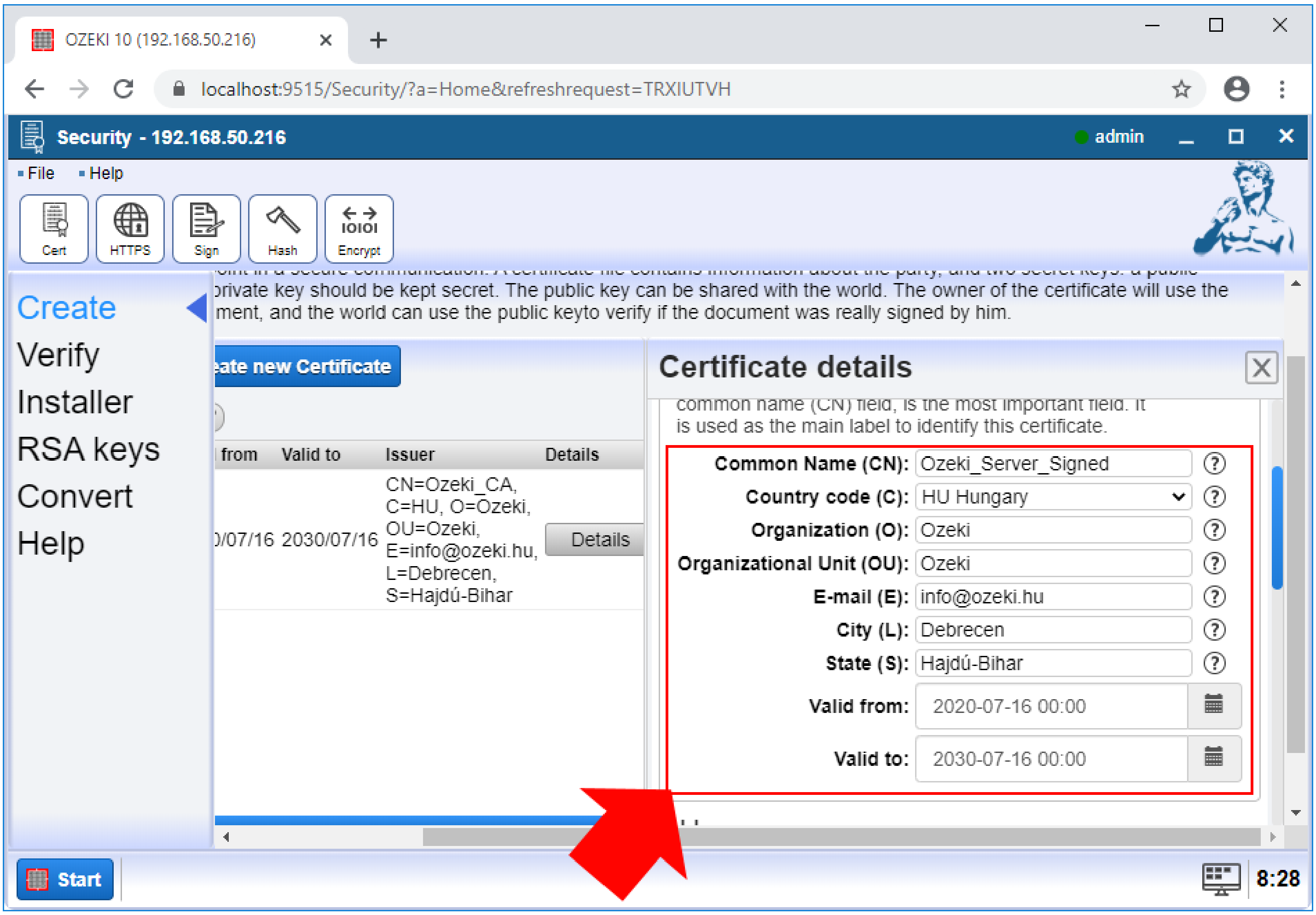
| N | /alid from | Valid to | Issuer | Details |
|---|---|---|---|---|
| 1. Ozeki_CA | 2020/07/16 | 2030/07/16 | CN=Ozeki_CA,<br>C=HU, O=Ozeki,<br>OU=Ozeki,<br>E=info@ozeki.hu,<br>L=Debrecen,<br>S=Hajdú-Bihar | Detai |

elete 0/1 item selected

In this form you may add additional certificates. Please se type of certification creation.

Figure 5 - Create signed certificate for the server

Then provide the details of the server's certificate.



private key should be kept secret. The public key can be shared with the world. The owner of the certificate will use the
ment, and the world can use the public keyto verify if the document was really signed by him.

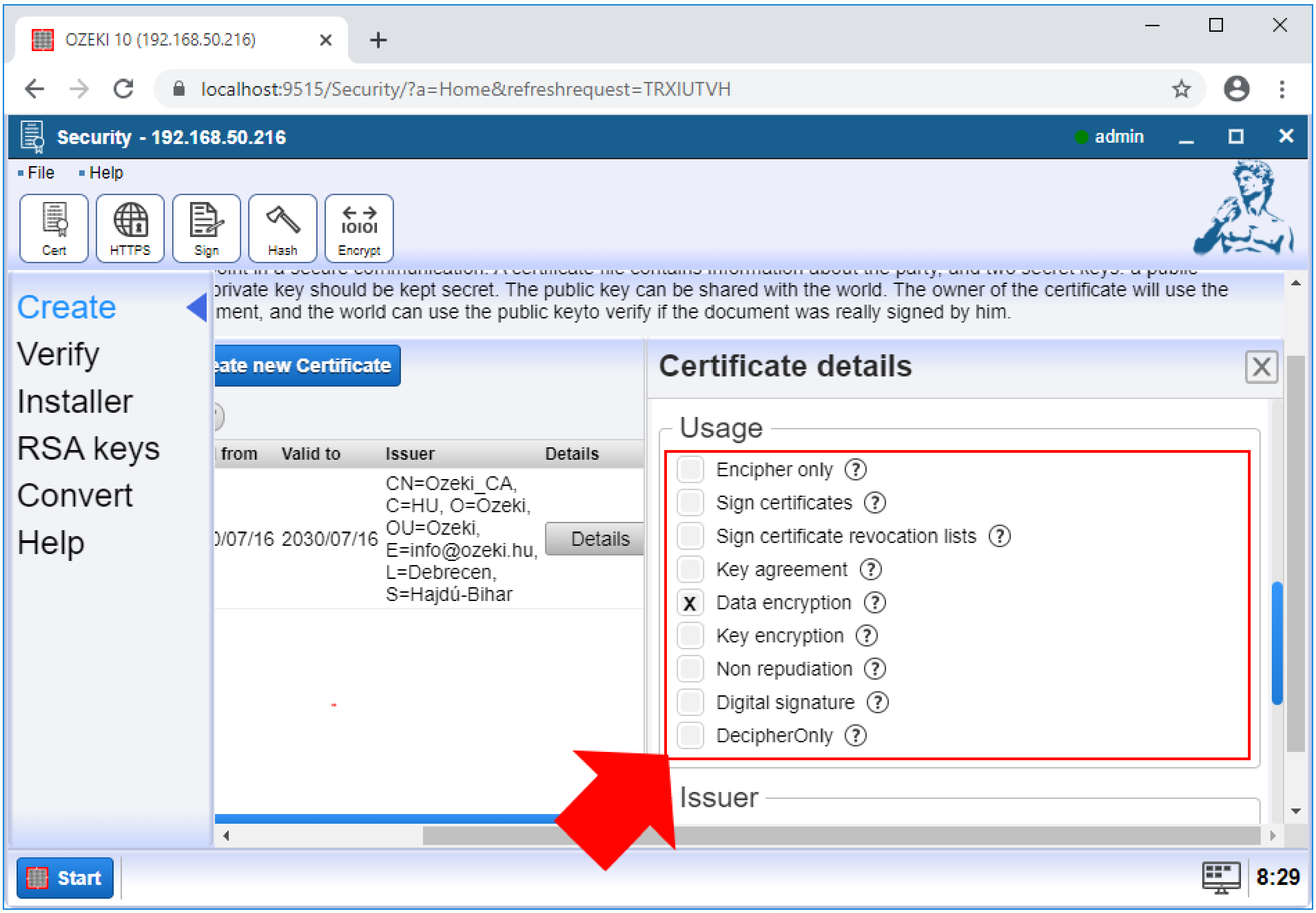
| from | Valid to | Issuer | Details |
|---|---|---|---|
| )/07/16 | 2030/07/16 | CN=Ozeki_CA,<br>C=HU, O=Ozeki,<br>OU=Ozeki,<br>E=info@ozeki.hu,<br>L=Debrecen,<br>S=Hajdú-Bihar | Details |

Certificate details

common name (CN) field, is the most important field. It is used as the main label to identify this certificate.

| | |
|---|---|
| Common Name (CN): | Ozeki_Server_Signed |
| Country code (C): | HU Hungary |
| Organization (O): | Ozeki |
| Organizational Unit (OU): | Ozeki |
| E-mail (E): | info@ozeki.hu |
| City (L): | Debrecen |
| State (S): | Hajdú-Bihar |
| Valid from: | 2020-07-16 00:00 |
| Valid to: | 2030-07-16 00:00 |

Figure 6 - Provide server's signed certificate details



private key should be kept secret. The public key can be shared with the world. The owner of the certificate will use the
ment, and the world can use the public keyto verify if the document was really signed by him.

| from | Valid to | Issuer | Details |
|---|---|---|---|
| )/07/16 | 2030/07/16 | CN=Ozeki_CA,<br>C=HU, O=Ozeki,<br>OU=Ozeki,<br>E=info@ozeki.hu,<br>L=Debrecen,<br>S=Hajdú-Bihar | Details |

Certificate details

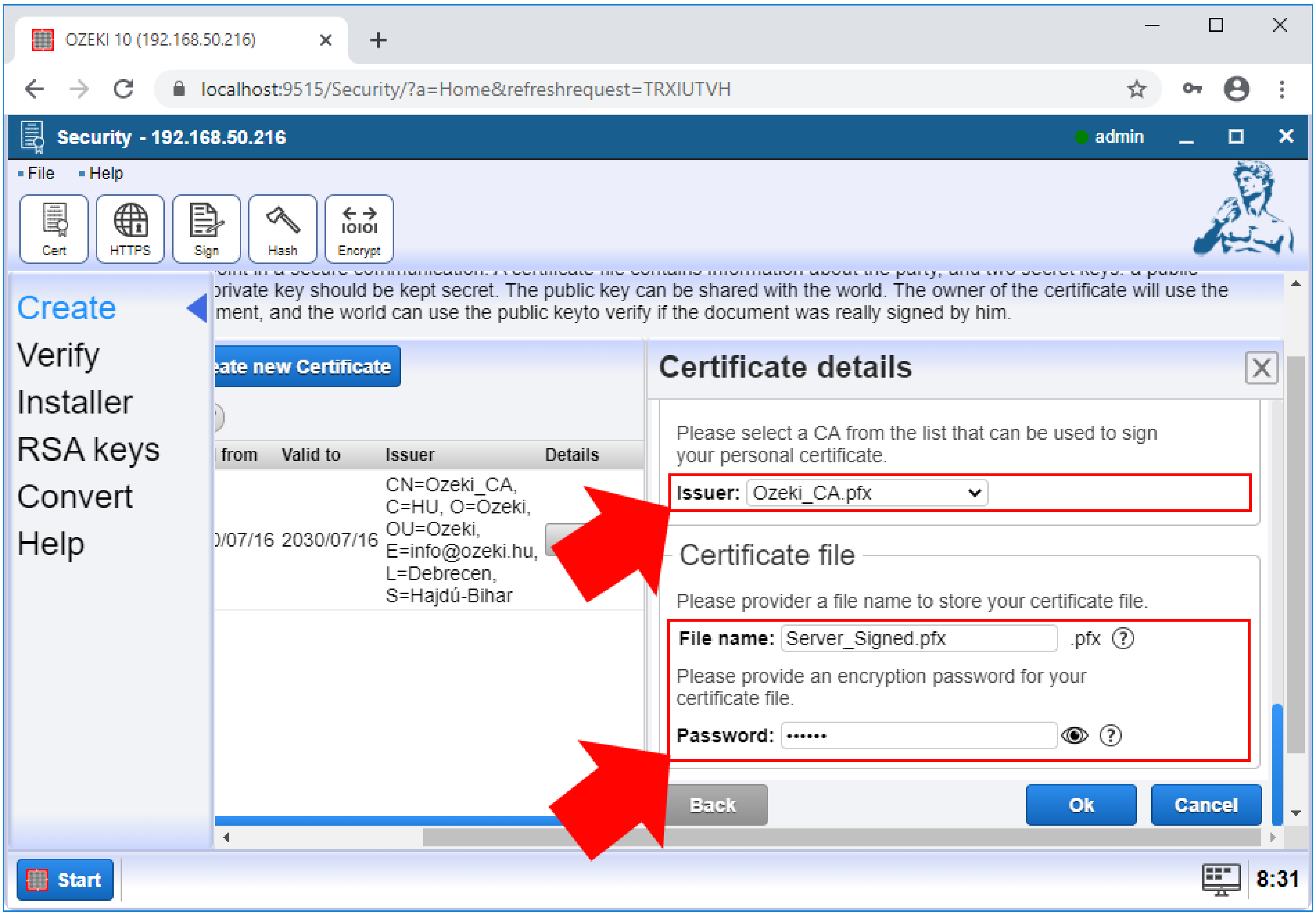
Usage

- ☐ Encipher only
- ☐ Sign certificates
- ☐ Sign certificate revocation lists
- ☐ Key agreement
- ☒ Data encryption
- ☐ Key encryption
- ☐ Non repudiation
- ☐ Digital signature
- ☐ DecipherOnly

Issuer

Figure 7 - Provide server's signed certificate usage details

Select the **issuer**, the **file name** and **password**.



private key should be kept secret. The public key can be shared with the world. The owner of the certificate will use the
ment, and the world can use the public keyto verify if the document was really signed by him.

| from | Valid to | Issuer | Details |
|---|---|---|---|
| )/07/16 | 2030/07/16 | CN=Ozeki_CA,<br>C=HU, O=Ozeki,<br>OU=Ozeki,<br>E=info@ozeki.hu,<br>L=Debrecen,<br>S=Hajdú-Bihar | |

Certificate details

Please select a CA from the list that can be used to sign your personal certificate.

**Issuer:** Ozeki_CA.pfx

Certificate file

Please provider a file name to store your certificate file.

**File name:** Server_Signed.pfx .pfx

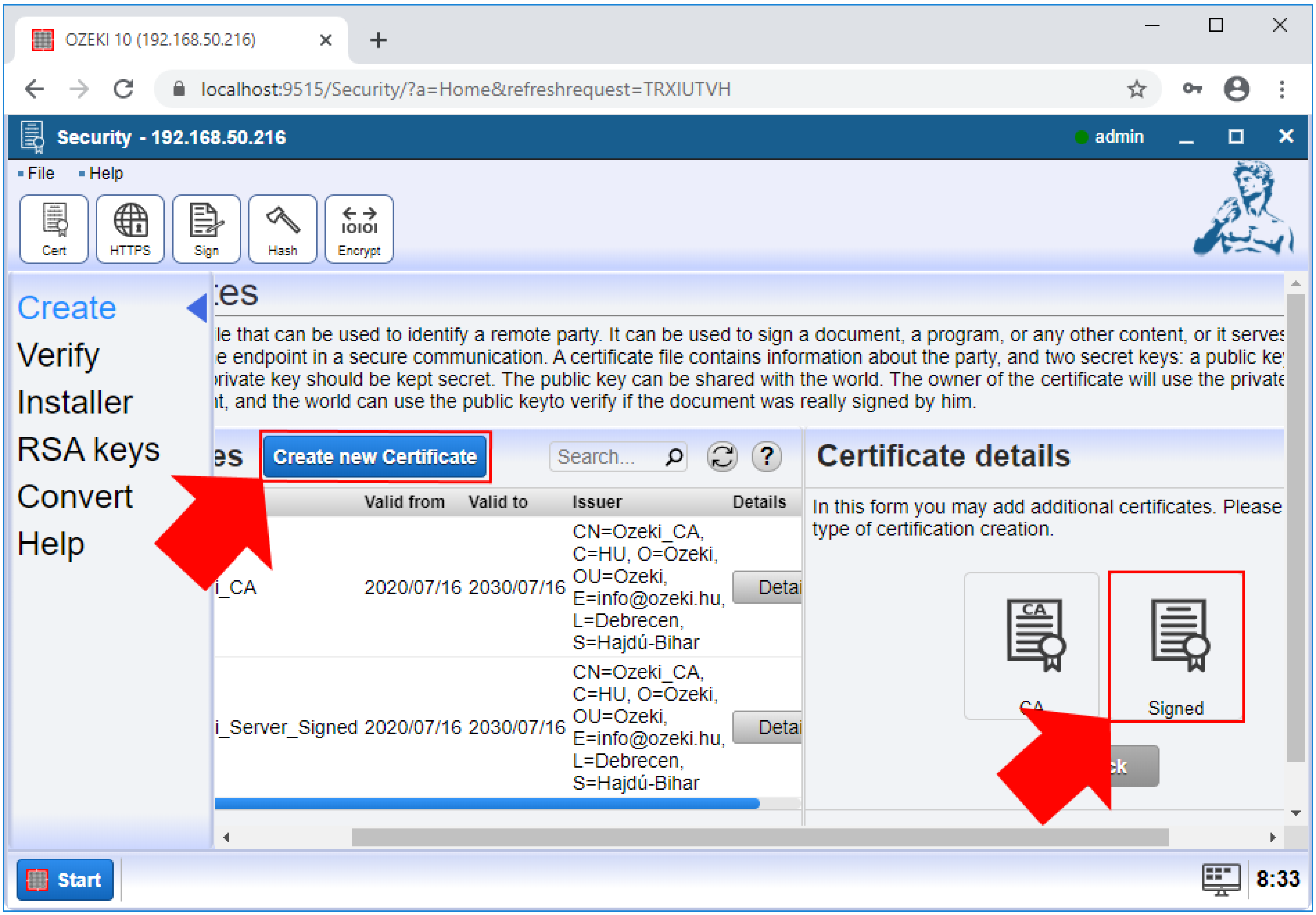
Please provide an encryption password for your certificate file.

**Password:** ••••••

Back  Ok  Cancel

Figure 8 - Provide usage, issuer, file name and password

Finally create the client's signed certificate. Click on **Create new Certificate** button and then **Create Signed**.



tes

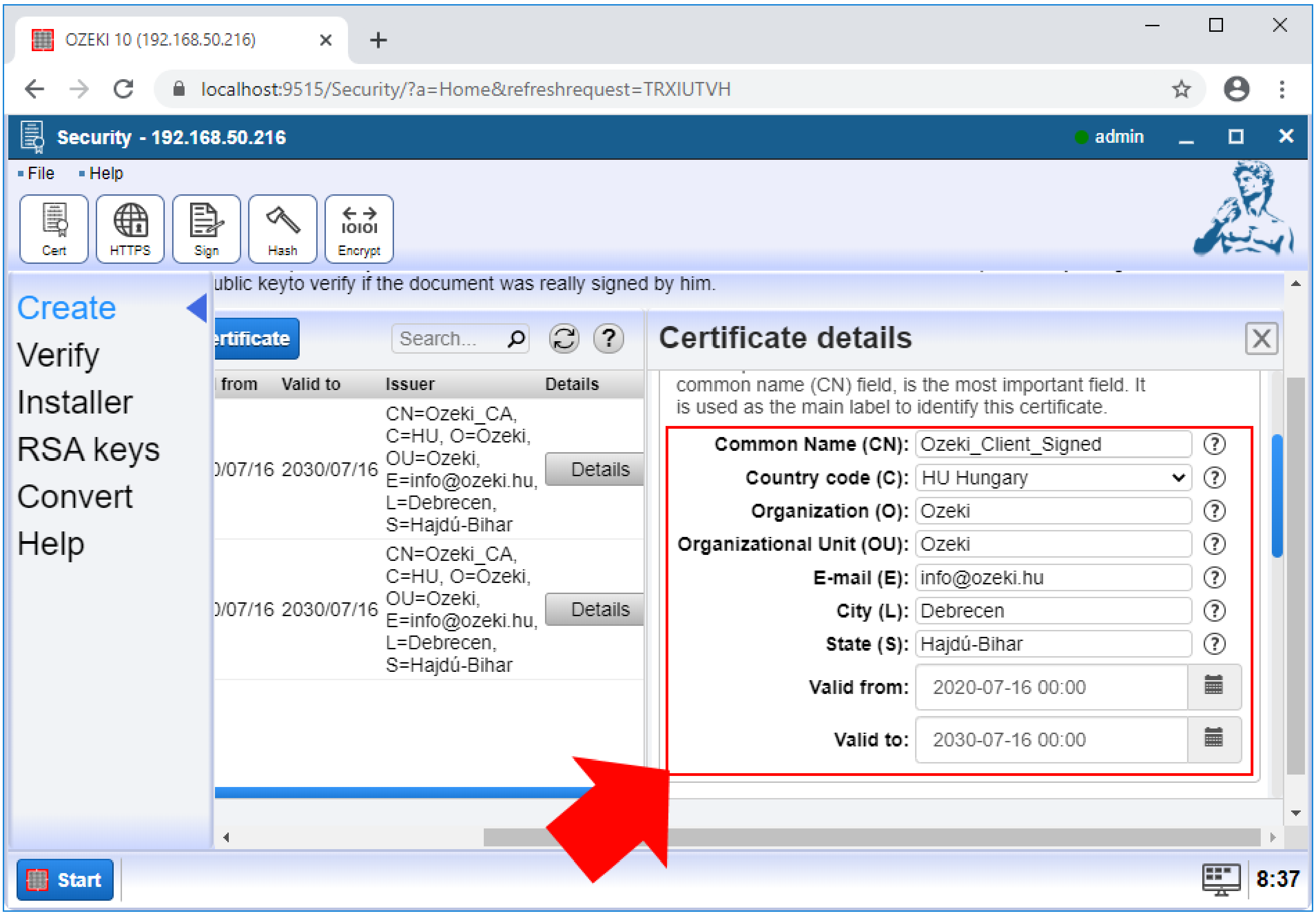
le that can be used to identify a remote party. It can be used to sign a document, a program, or any other content, or it serves e endpoint in a secure communication. A certificate file contains information about the party, and two secret keys: a public ke rivate key should be kept secret. The public key can be shared with the world. The owner of the certificate will use the private it, and the world can use the public keyto verify if the document was really signed by him.

| | Valid from | Valid to | Issuer | Details |
|---|---|---|---|---|
| i_CA | 2020/07/16 | 2030/07/16 | CN=Ozeki_CA,<br>C=HU, O=Ozeki,<br>OU=Ozeki,<br>E=info@ozeki.hu,<br>L=Debrecen,<br>S=Hajdú-Bihar | Detai |
| i_Server_Signed | 2020/07/16 | 2030/07/16 | CN=Ozeki_CA,<br>C=HU, O=Ozeki,<br>OU=Ozeki,<br>E=info@ozeki.hu,<br>L=Debrecen,<br>S=Hajdú-Bihar | Detai |

In this form you may add additional certificates. Please type of certification creation.

Figure 9 - Create signed certificate for the client

Provide the details of the client's certificate.



common name (CN) field, is the most important field. It is used as the main label to identify this certificate.

Common Name (CN): Ozeki_Client_Signed

Country code (C): HU Hungary

Organization (O): Ozeki

Organizational Unit (OU): Ozeki

E-mail (E): info@ozeki.hu

City (L): Debrecen

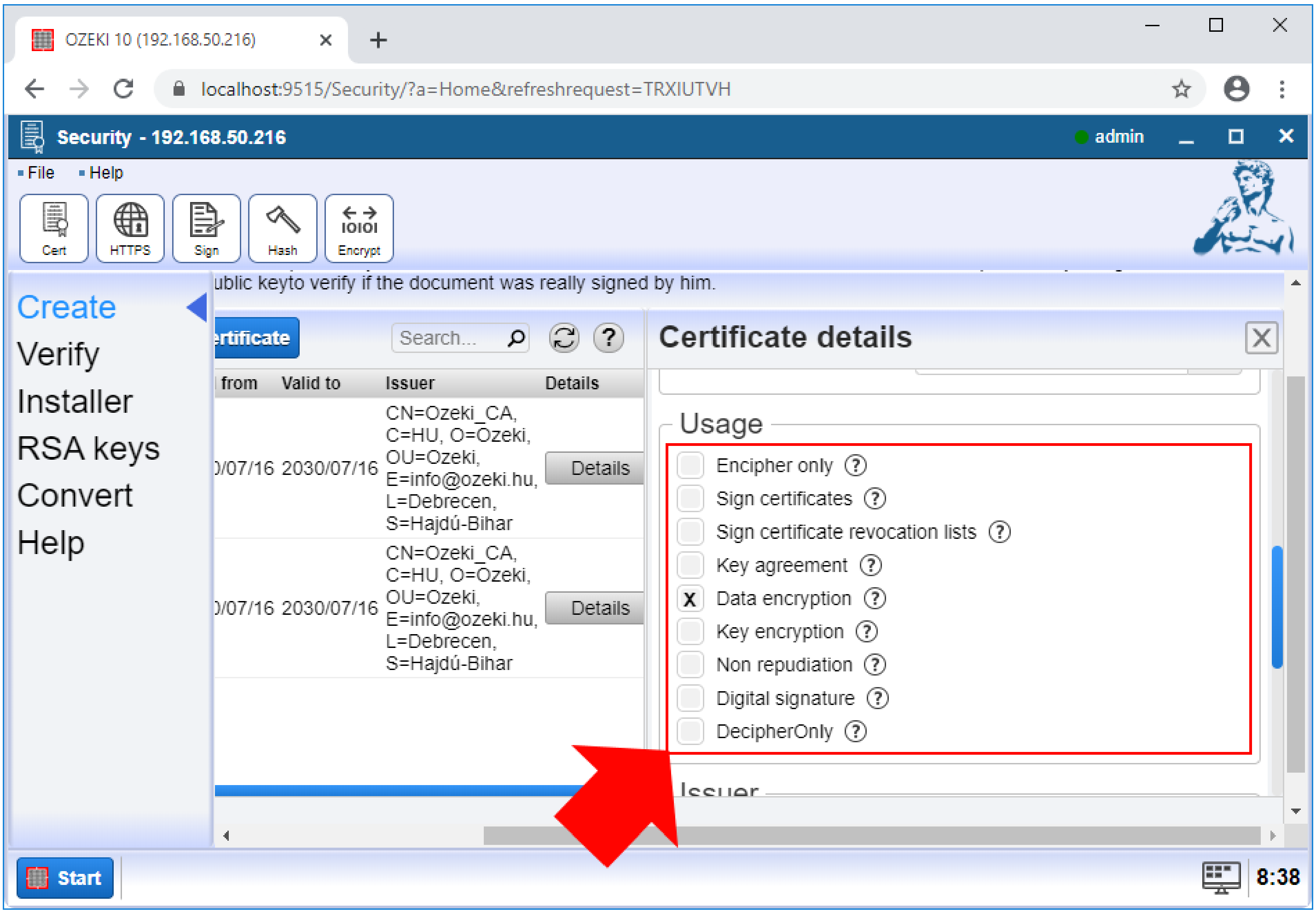
State (S): Hajdú-Bihar

Valid from: 2020-07-16 00:00

Valid to: 2030-07-16 00:00

OZEKI 10 (192.168.50.216)
localhost:9515/Security/?a=Home&refreshrequest=TRXIUTVH
Security - 192.168.50.216
admin
File
Help
Cert
HTTPS
Sign
Hash
Encrypt
Create
Verify
Installer
RSA keys
Convert
Help
ublic keyto verify if the document was really signed by him.
rtificate
Search...
Certificate details
Start
8:37

Figure 10 - Provide client's signed certificate details



Usage

☐ Encipher only

☐ Sign certificates

☐ Sign certificate revocation lists

☐ Key agreement

☒ Data encryption

☐ Key encryption

☐ Non repudiation

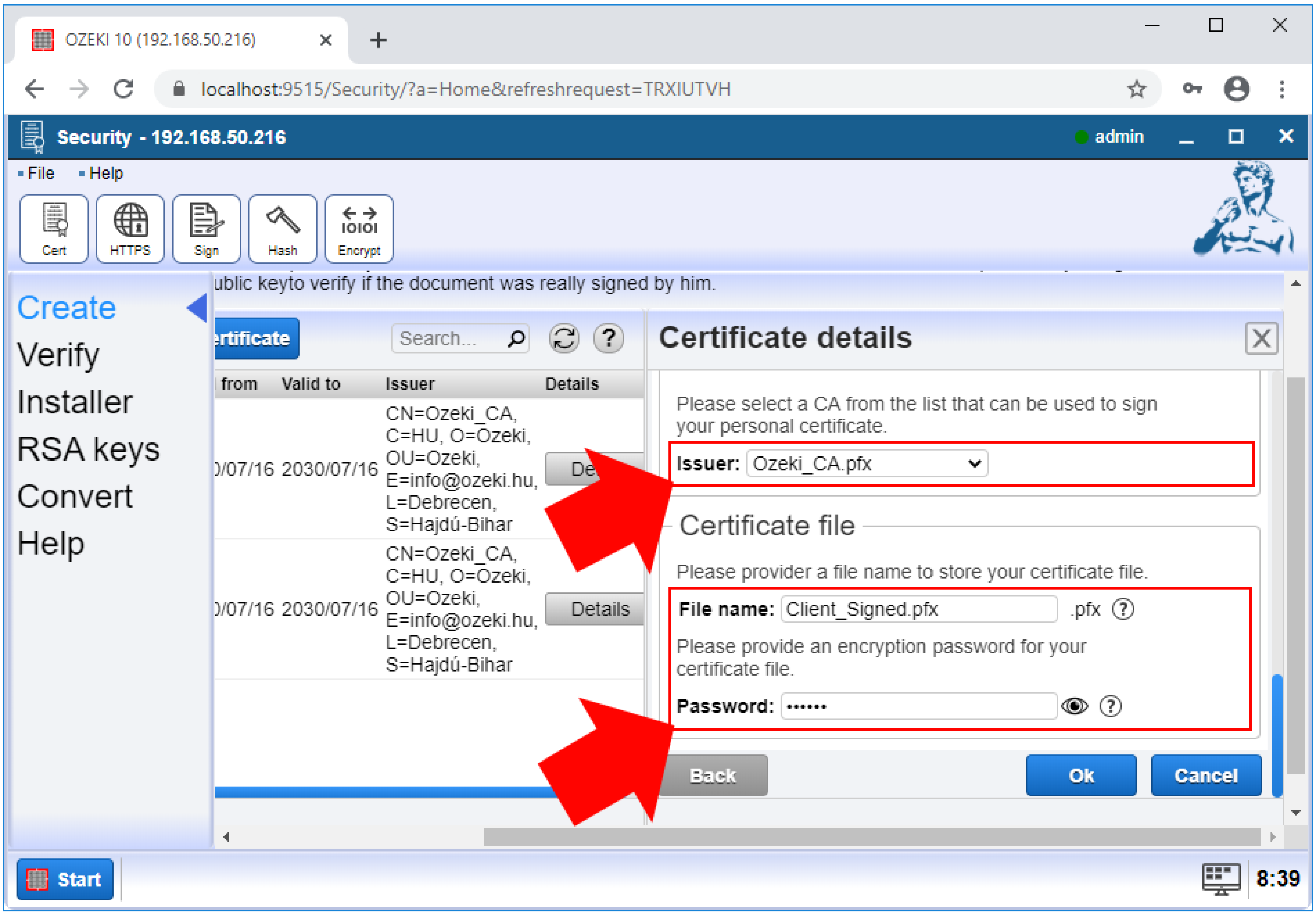
☐ Digital signature

☐ DecipherOnly

OZEKI 10 (192.168.50.216)
localhost:9515/Security/?a=Home&refreshrequest=TRXIUTVH
Security - 192.168.50.216
admin
Certificate details
Issuer
Start
8:38

Figure 11 - Provide client's signed certificate usage details

Select the **issuer**, the **file name** and **password**.



ublic keyto verify if the document was really signed by him.

| from | Valid to | Issuer | Details |
|---|---|---|---|
| )/07/16 | 2030/07/16 | CN=Ozeki_CA, C=HU, O=Ozeki, OU=Ozeki, E=info@ozeki.hu, L=Debrecen, S=Hajdú-Bihar | De |
| )/07/16 | 2030/07/16 | CN=Ozeki_CA, C=HU, O=Ozeki, OU=Ozeki, E=info@ozeki.hu, L=Debrecen, S=Hajdú-Bihar | Details |

Certificate details

Please select a CA from the list that can be used to sign your personal certificate.

Issuer: Ozeki_CA.pfx

Certificate file

Please provider a file name to store your certificate file.

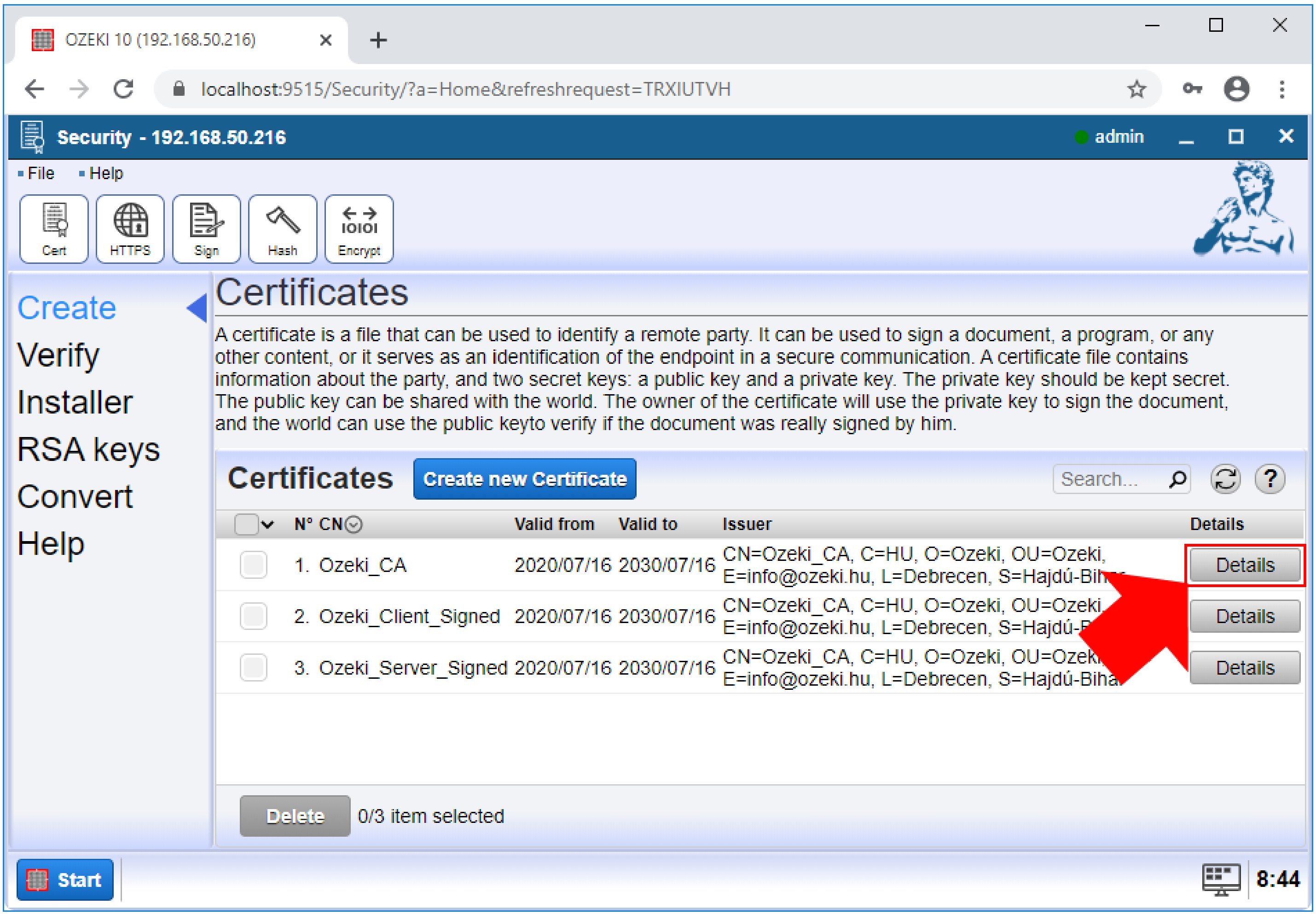
File name: Client_Signed.pfx .pfx

Please provide an encryption password for your certificate file.

Password: ••••••

Figure 12 - Provide usage, issuer, file name and password

If the certificates have been created you need to download them. Open the CA certificate **Details** page.



## Certificates

A certificate is a file that can be used to identify a remote party. It can be used to sign a document, a program, or any other content, or it serves as an identification of the endpoint in a secure communication. A certificate file contains information about the party, and two secret keys: a public key and a private key. The private key should be kept secret. The public key can be shared with the world. The owner of the certificate will use the private key to sign the document, and the world can use the public keyto verify if the document was really signed by him.

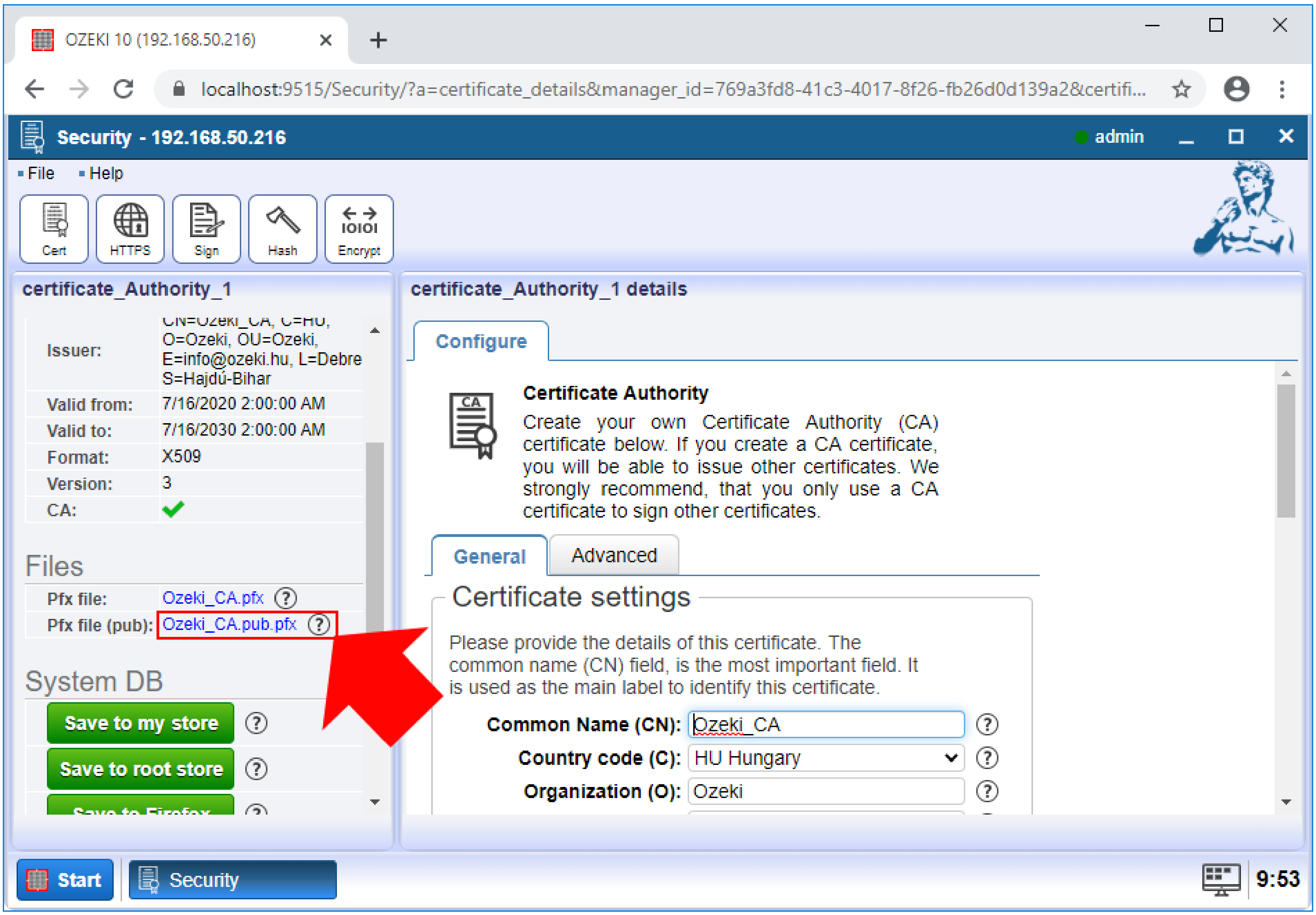
### Certificates

| | N° | CN | Valid from | Valid to | Issuer | Details |
|---|---|---|---|---|---|---|
| □ | 1. | Ozeki_CA | 2020/07/16 | 2030/07/16 | CN=Ozeki_CA, C=HU, O=Ozeki, OU=Ozeki, E=info@ozeki.hu, L=Debrecen, S=Hajdú-Bih | Details |
| □ | 2. | Ozeki_Client_Signed | 2020/07/16 | 2030/07/16 | CN=Ozeki_CA, C=HU, O=Ozeki, OU=Ozeki, E=info@ozeki.hu, L=Debrecen, S=Hajdú-B | Details |
| □ | 3. | Ozeki_Server_Signed | 2020/07/16 | 2030/07/16 | CN=Ozeki_CA, C=HU, O=Ozeki, OU=Ozeki, E=info@ozeki.hu, L=Debrecen, S=Hajdú-Biha | Details |

0/3 item selected

Figure 13 - Open CA certificate details page

And **download** the **Public** CA certificate.


OZEKI 10 (192.168.50.216)
localhost:9515/Security/?a=certificate_details&manager_id=769a3fd8-41c3-4017-8f26-fb26d0d139a2&certifi...
Security - 192.168.50.216
admin
File
Help
Cert
HTTPS
Sign
Hash
Encrypt
Save to my store
Save to root store
Save to Firefox
Configure
General
Advanced
Start
Security
9:53

certificate_Authority_1

| | |
|---|---|
| Issuer: | CN=Ozeki_CA, C=HU, O=Ozeki, OU=Ozeki, E=info@ozeki.hu, L=Debre S=Hajdú-Bihar |
| Valid from: | 7/16/2020 2:00:00 AM |
| Valid to: | 7/16/2030 2:00:00 AM |
| Format: | X509 |
| Version: | 3 |
| CA: | ✓ |

Files

| | |
|---|---|
| Pfx file: | Ozeki_CA.pfx |
| Pfx file (pub): | Ozeki_CA.pub.pfx |

System DB

certificate_Authority_1 details

**Certificate Authority**

Create your own Certificate Authority (CA) certificate below. If you create a CA certificate, you will be able to issue other certificates. We strongly recommend, that you only use a CA certificate to sign other certificates.

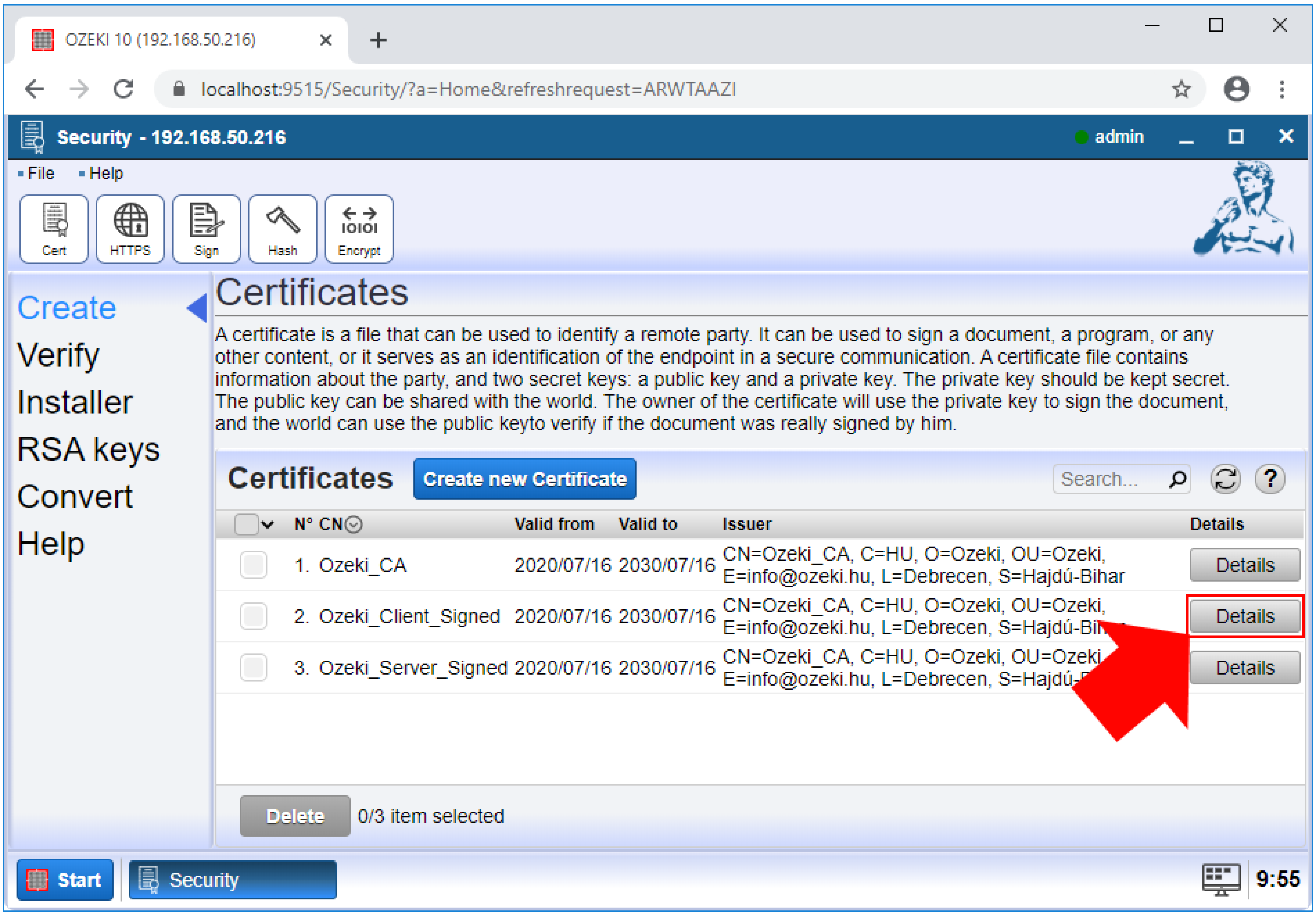
Certificate settings

Please provide the details of this certificate. The common name (CN) field, is the most important field. It is used as the main label to identify this certificate.

| | |
|---|---|
| Common Name (CN): | Ozeki_CA |
| Country code (C): | HU Hungary |
| Organization (O): | Ozeki |

Figure 14 - Download Public CA certificate

Then open client certificate **Details** page.



Certificates

A certificate is a file that can be used to identify a remote party. It can be used to sign a document, a program, or any other content, or it serves as an identification of the endpoint in a secure communication. A certificate file contains information about the party, and two secret keys: a public key and a private key. The private key should be kept secret. The public key can be shared with the world. The owner of the certificate will use the private key to sign the document, and the world can use the public keyto verify if the document was really signed by him.

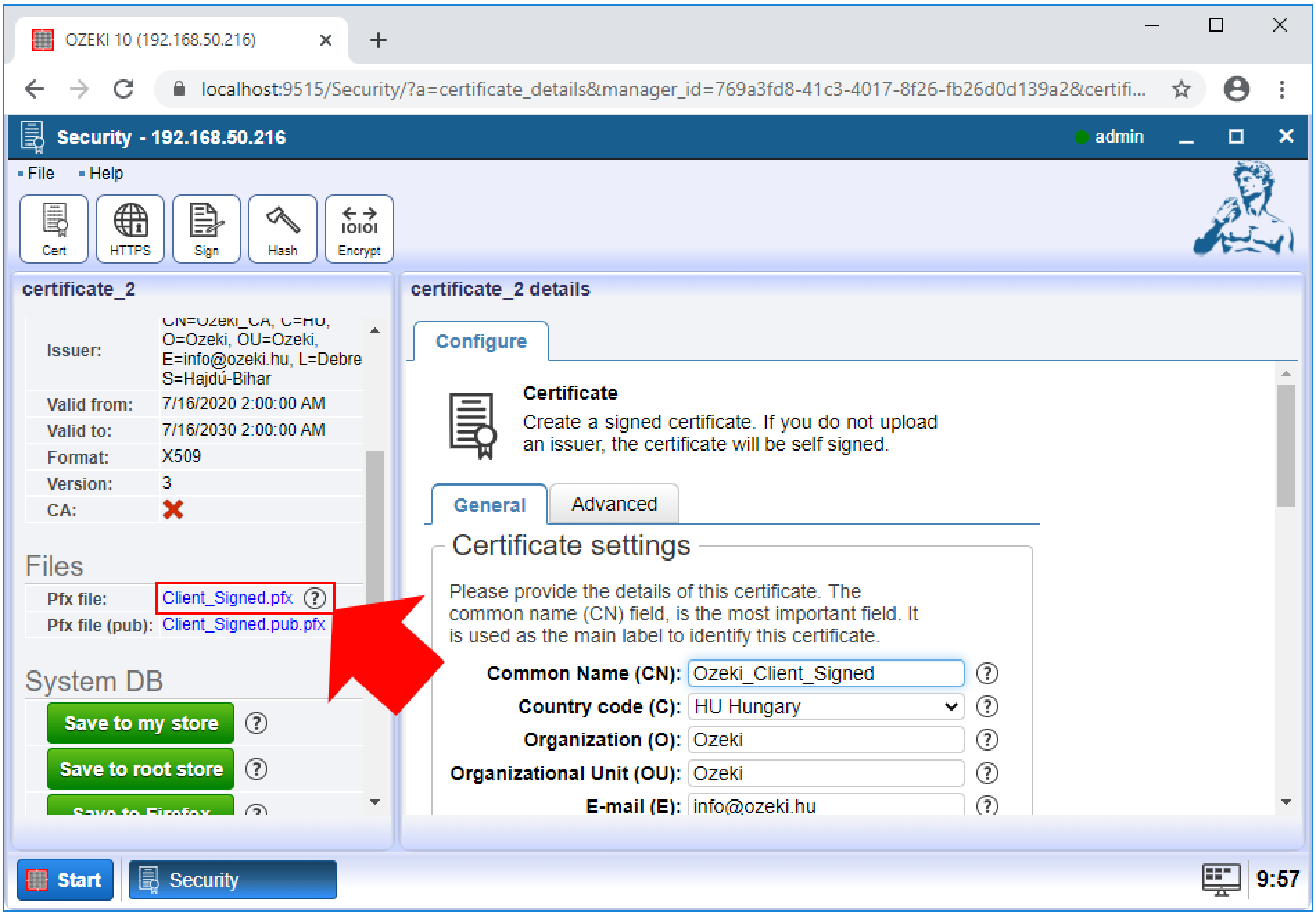
Certificates

| □ | N° | CN | Valid from | Valid to | Issuer | Details |
|---|---|---|---|---|---|---|
| □ | 1. | Ozeki_CA | 2020/07/16 | 2030/07/16 | CN=Ozeki_CA, C=HU, O=Ozeki, OU=Ozeki, E=info@ozeki.hu, L=Debrecen, S=Hajdú-Bihar | Details |
| □ | 2. | Ozeki_Client_Signed | 2020/07/16 | 2030/07/16 | CN=Ozeki_CA, C=HU, O=Ozeki, OU=Ozeki, E=info@ozeki.hu, L=Debrecen, S=Hajdú-Bi | Details |
| □ | 3. | Ozeki_Server_Signed | 2020/07/16 | 2030/07/16 | CN=Ozeki_CA, C=HU, O=Ozeki, OU=Ozeki E=info@ozeki.hu, L=Debrecen, S=Hajdú- | Details |

0/3 item selected

Figure 15 - Open client certificate details page

After it, **download** client's certificate.



OZEKI 10 (192.168.50.216)
localhost:9515/Security/?a=certificate_details&manager_id=769a3fd8-41c3-4017-8f26-fb26d0d139a2&certifi...
Security - 192.168.50.216
admin
File
Help
Cert
HTTPS
Sign
Hash
Encrypt
certificate_2
certificate_2 details
Configure
General
Advanced
Start
Security
9:57

| | |
|---|---|
| Issuer: | CN=Ozeki_CA, C=HU, O=Ozeki, OU=Ozeki, E=info@ozeki.hu, L=Debre S=Hajdú-Bihar |
| Valid from: | 7/16/2020 2:00:00 AM |
| Valid to: | 7/16/2030 2:00:00 AM |
| Format: | X509 |
| Version: | 3 |
| CA: | ✗ |

Files

| | |
|---|---|
| Pfx file: | Client_Signed.pfx |
| Pfx file (pub): | Client_Signed.pub.pfx |

System DB

Save to my store

Save to root store

Save to Firefox

**Certificate**

Create a signed certificate. If you do not upload an issuer, the certificate will be self signed.

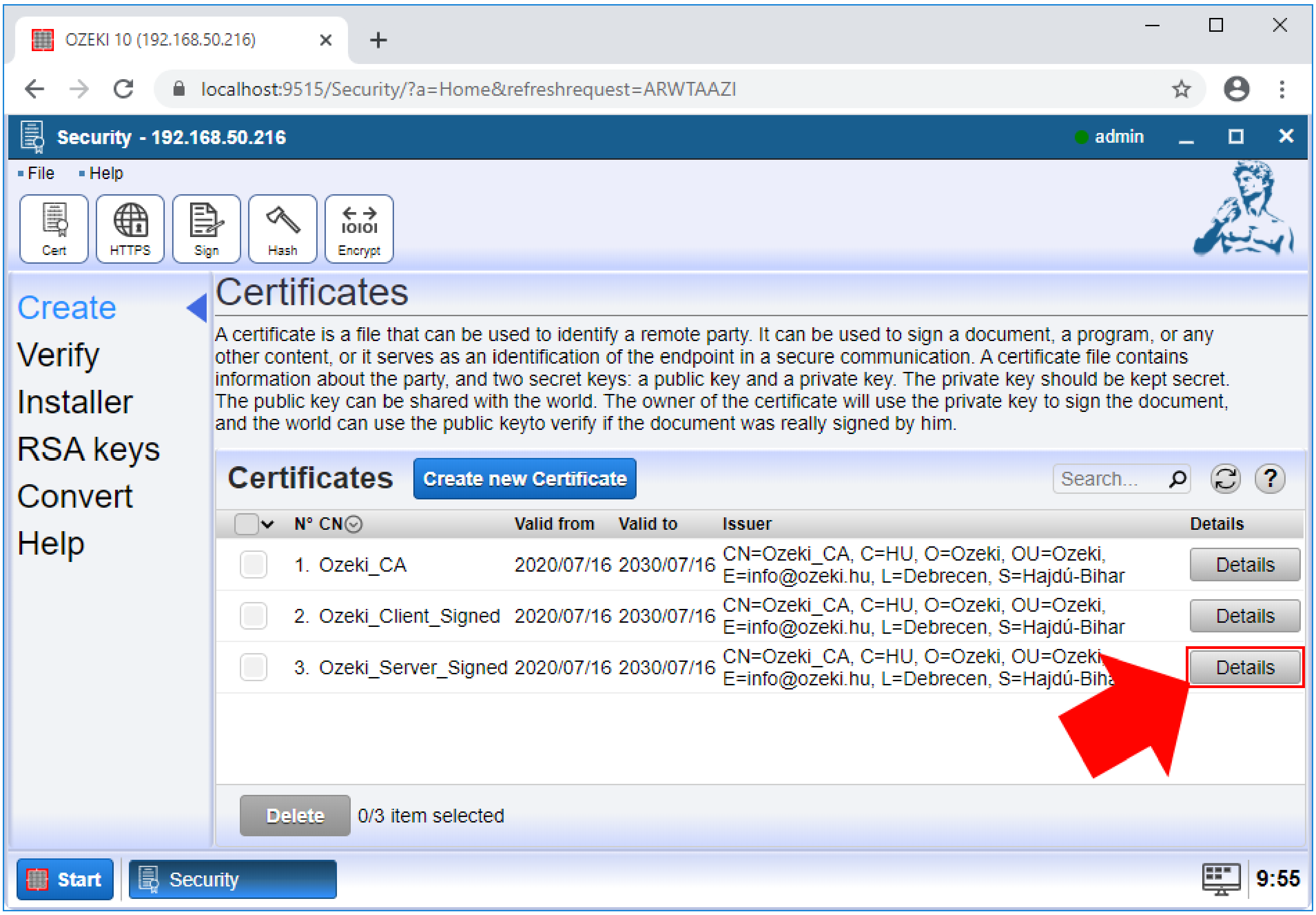
Certificate settings

Please provide the details of this certificate. The common name (CN) field, is the most important field. It is used as the main label to identify this certificate.

| | |
|---|---|
| Common Name (CN): | Ozeki_Client_Signed |
| Country code (C): | HU Hungary |
| Organization (O): | Ozeki |
| Organizational Unit (OU): | Ozeki |
| E-mail (E): | info@ozeki.hu |

Figure 16 - Download client's certificate

And finally open server certificate **Details** page.



OZEKI 10 (192.168.50.216)
localhost:9515/Security/?a=Home&refreshrequest=ARWTAAZI
Security - 192.168.50.216
admin
File
Help
Cert
HTTPS
Sign
Hash
Encrypt
Create
Verify
Installer
RSA keys
Convert
Help

Certificates

A certificate is a file that can be used to identify a remote party. It can be used to sign a document, a program, or any other content, or it serves as an identification of the endpoint in a secure communication. A certificate file contains information about the party, and two secret keys: a public key and a private key. The private key should be kept secret. The public key can be shared with the world. The owner of the certificate will use the private key to sign the document, and the world can use the public keyto verify if the document was really signed by him.

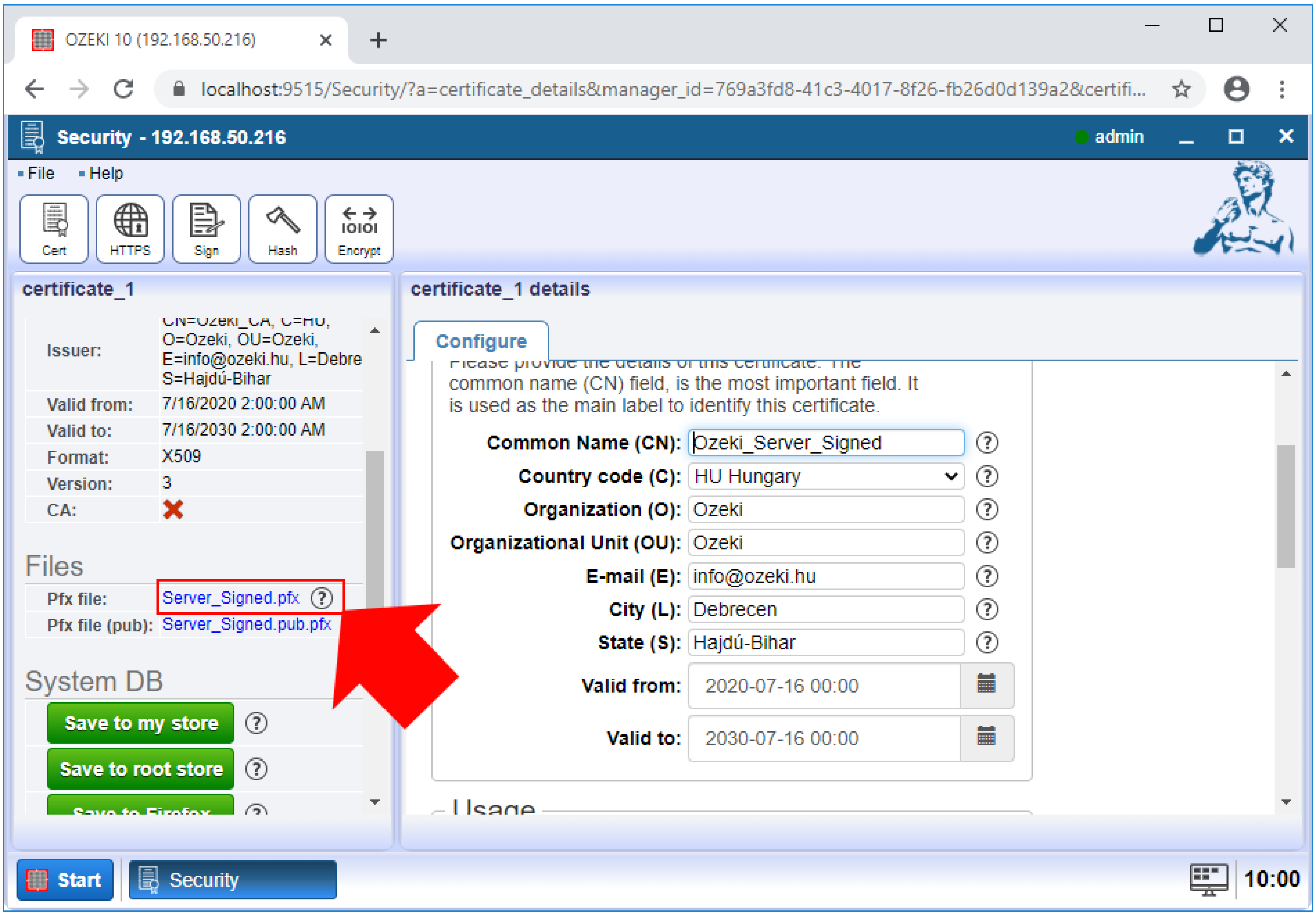
Certificates

| | N° | CN | Valid from | Valid to | Issuer | Details |
|---|---|---|---|---|---|---|
| □ | 1. | Ozeki_CA | 2020/07/16 | 2030/07/16 | CN=Ozeki_CA, C=HU, O=Ozeki, OU=Ozeki, E=info@ozeki.hu, L=Debrecen, S=Hajdú-Bihar | Details |
| □ | 2. | Ozeki_Client_Signed | 2020/07/16 | 2030/07/16 | CN=Ozeki_CA, C=HU, O=Ozeki, OU=Ozeki, E=info@ozeki.hu, L=Debrecen, S=Hajdú-Bihar | Details |
| □ | 3. | Ozeki_Server_Signed | 2020/07/16 | 2030/07/16 | CN=Ozeki_CA, C=HU, O=Ozeki, OU=Ozeki, E=info@ozeki.hu, L=Debrecen, S=Hajdú-Biha | Details |

0/3 item selected

Figure 17 - Open server certificate details page

Thereafter **download** servers's certificate.


OZEKI 10 (192.168.50.216)
localhost:9515/Security/?a=certificate_details&manager_id=769a3fd8-41c3-4017-8f26-fb26d0d139a2&certifi...
Security - 192.168.50.216
admin
File
Help
Cert
HTTPS
Sign
Hash
Encrypt
Configure
Save to my store
Save to root store
Save to Firefox
Usage
Start
Security
10:00

certificate_1

Issuer: CN=Ozeki_CA, C=HU, O=Ozeki, OU=Ozeki, E=info@ozeki.hu, L=Debre S=Hajdú-Bihar

Valid from: 7/16/2020 2:00:00 AM

Valid to: 7/16/2030 2:00:00 AM

Format: X509

Version: 3

CA: ✗

Files

Pfx file: Server_Signed.pfx

Pfx file (pub): Server_Signed.pub.pfx

System DB

certificate_1 details

Please provide the details of this certificate. The common name (CN) field, is the most important field. It is used as the main label to identify this certificate.

Common Name (CN): Ozeki_Server_Signed

Country code (C): HU Hungary

Organization (O): Ozeki

Organizational Unit (OU): Ozeki

E-mail (E): info@ozeki.hu

City (L): Debrecen

State (S): Hajdú-Bihar

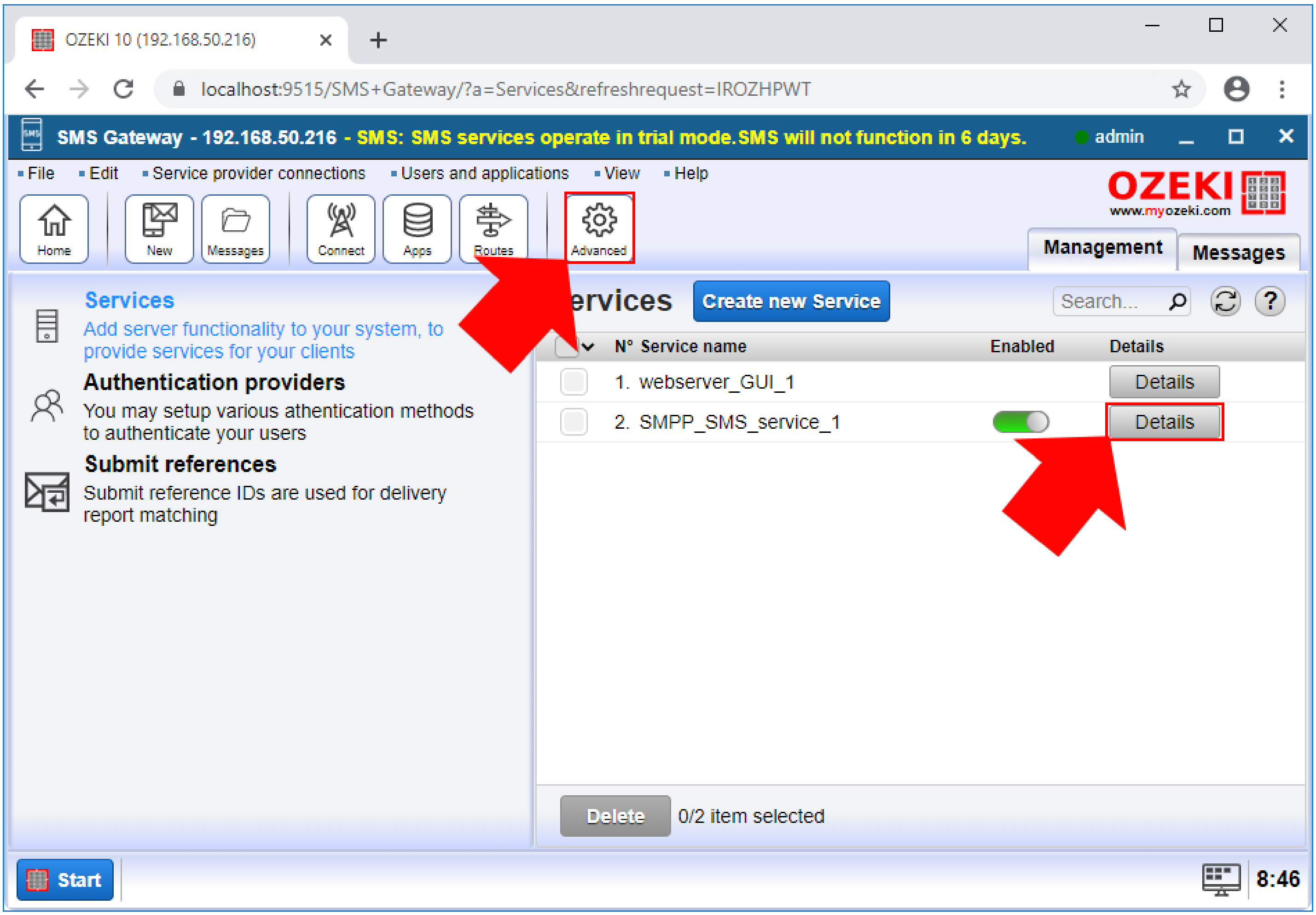
Valid from: 2020-07-16 00:00

Valid to: 2030-07-16 00:00

Figure 18 - Download servers's certificate

## II. Configure SSL connection for the server

On the server side open the **Advanced** menu,and select the **Details** page.


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=Services&refreshrequest=IROZHPWT
SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode.SMS will not function in 6 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Create new Service
Search...
Delete
Start
8:46

**Services**
Add server functionality to your system, to provide services for your clients

**Authentication providers**
You may setup various athentication methods to authenticate your users

**Submit references**
Submit reference IDs are used for delivery report matching

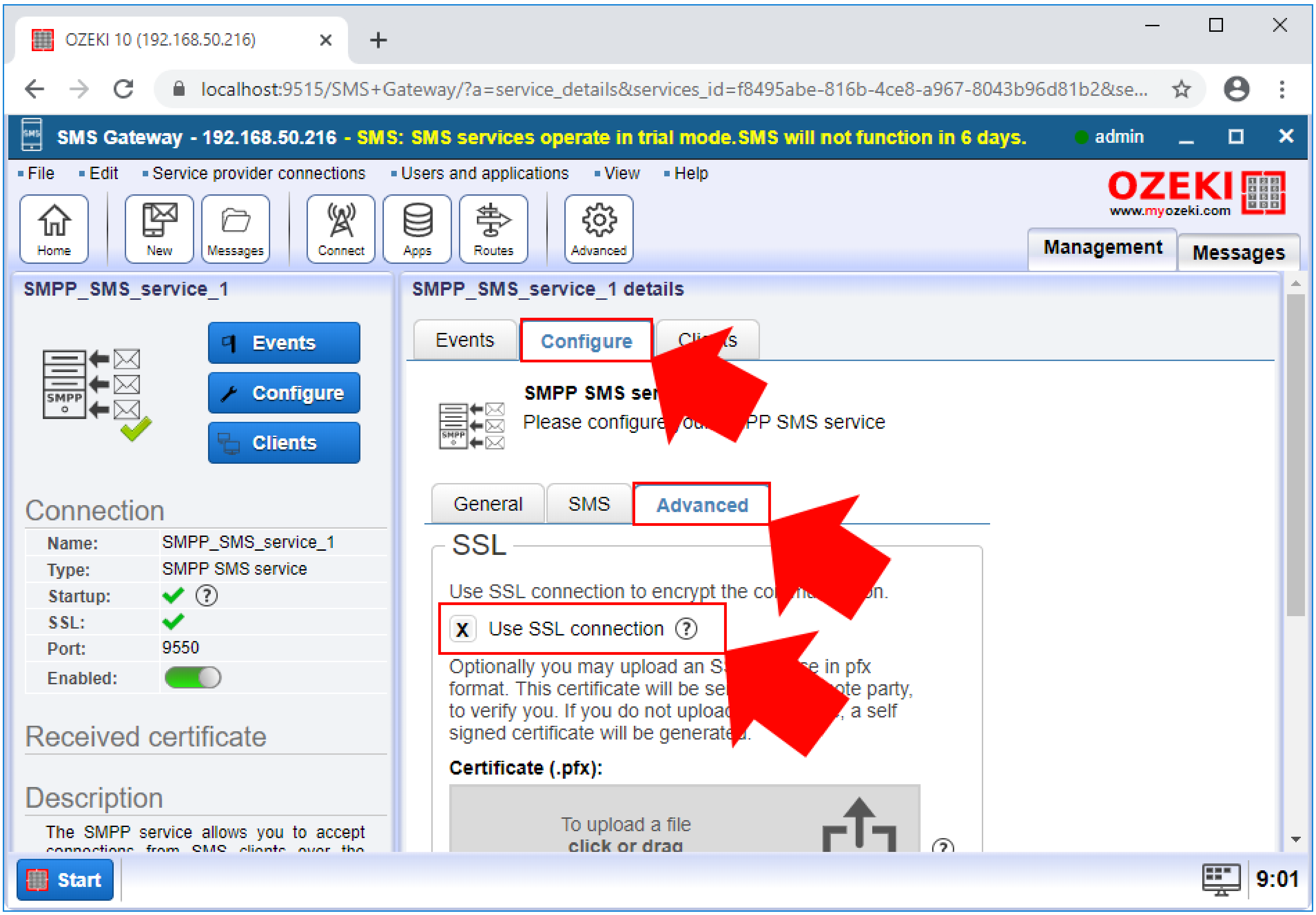
ervices

| N° | Service name | Enabled | Details |
|---|---|---|---|
| 1. | webserver_GUI_1 | | Details |
| 2. | SMPP_SMS_service_1 | ● | Details |

0/2 item selected

Figure 19 - Open Advanced menu on the server

Here, go to the **Configure** menu to modify the SSL settings.


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=service_details&services_id=f8495abe-816b-4ce8-a967-8043b96d81b2&se...
SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode.SMS will not function in 6 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
OZEKI
www.myozeki.com
Management
Messages
SMPP_SMS_service_1
Events
Configure
Clients
SMPP_SMS_service_1 details
Events
Configure
Cli ts
General
SMS
Advanced
Certificate (.pfx):
To upload a file
click or drag
Start
9:01

Connection

| | |
|---|---|
| Name: | SMPP_SMS_service_1 |
| Type: | SMPP SMS service |
| Startup: | ✓ ? |
| SSL: | ✓ |
| Port: | 9550 |
| Enabled: | ● |

Received certificate

Description

The SMPP service allows you to accept connections from SMS clients over the

SMPP SMS ser

Please configure you PP SMS service

SSL

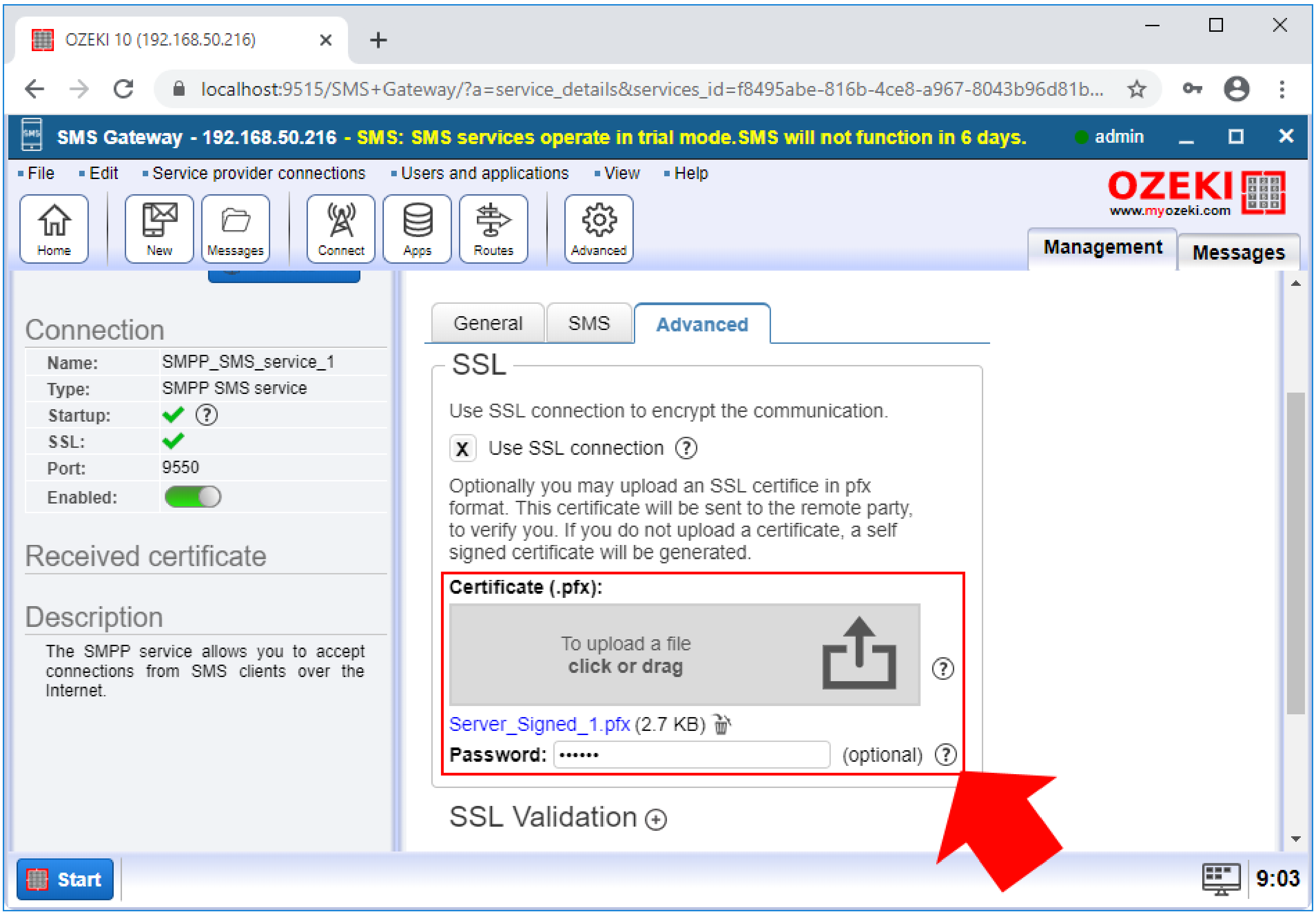
Use SSL connection to encrypt the co n.

☒ Use SSL connection ?

Optionally you may upload an S e in pfx format. This certificate will be se ote party, to verify you. If you do not uploa , a self signed certificate will be generate .

Figure 20 - Open service details page

To enable SSL connection, in the **Configure** menu select the **Advanced** tab and open **SSL**. Under SSL section check the "**Use SSL connection option**". **Upload** the server signed certificate and provid the **password** for it.


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=service_details&services_id=f8495abe-816b-4ce8-a967-8043b96d81b...
SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode.SMS will not function in 6 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
General
SMS
Advanced
Start
9:03

Connection

| | |
|---|---|
| Name: | SMPP_SMS_service_1 |
| Type: | SMPP SMS service |
| Startup: | ✓ |
| SSL: | ✓ |
| Port: | 9550 |
| Enabled: | ● |

Received certificate

Description

The SMPP service allows you to accept connections from SMS clients over the Internet.

SSL

Use SSL connection to encrypt the communication.

☒ Use SSL connection

Optionally you may upload an SSL certifice in pfx format. This certificate will be sent to the remote party, to verify you. If you do not upload a certificate, a self signed certificate will be generated.

Certificate (.pfx):

To upload a file
click or drag

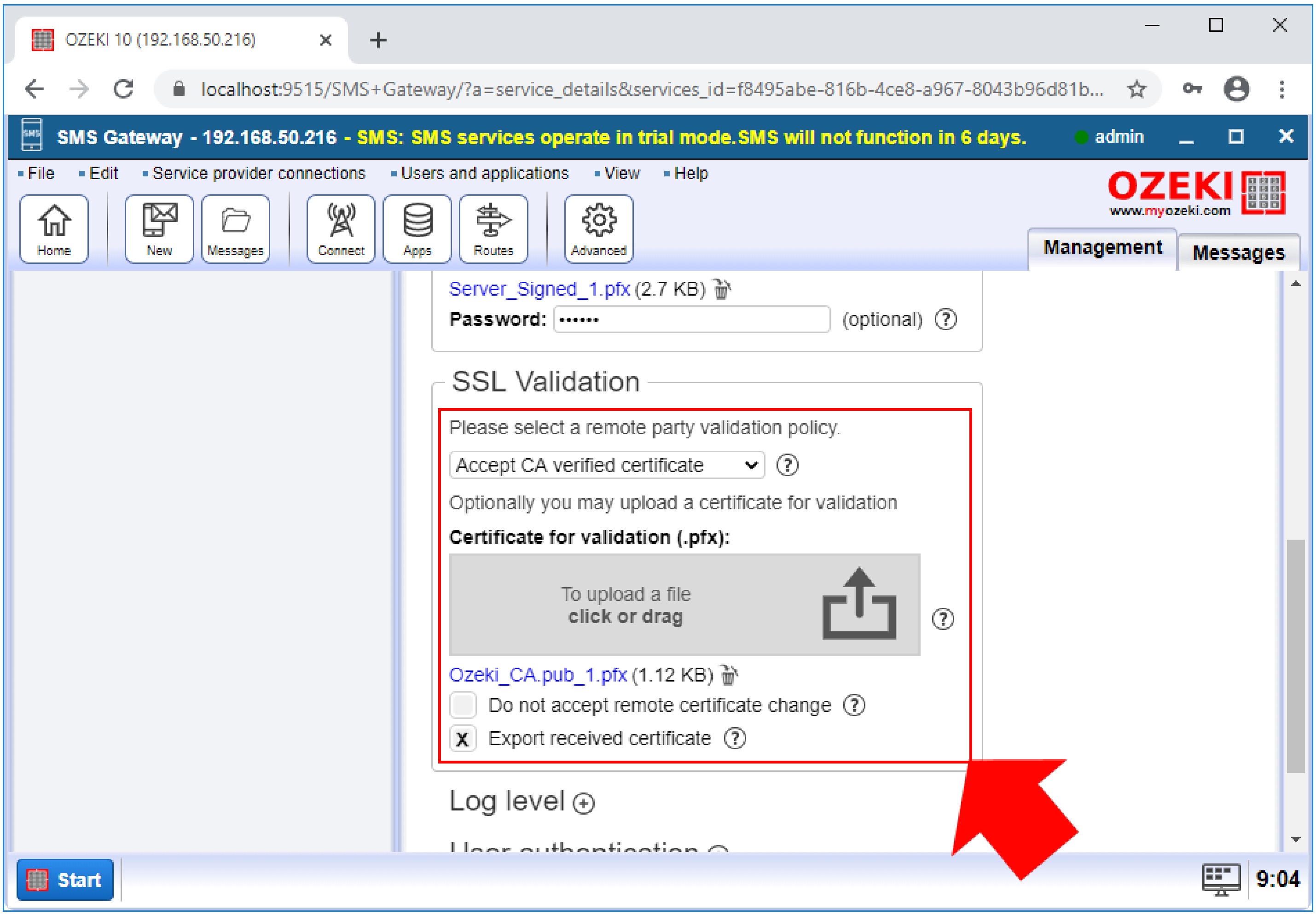
Server_Signed_1.pfx (2.7 KB)

Password: •••••• (optional)

SSL Validation

Figure 21 - Enable SSL connection for the service

For CA validation open the **SSL Validation** section. Select the "**Accept CA verified certificate**" form the list and **upload** the public CA certificate.


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=service_details&services_id=f8495abe-816b-4ce8-a967-8043b96d81b...
SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode.SMS will not function in 6 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Start
9:04

Server_Signed_1.pfx (2.7 KB)

Password: •••••• (optional)

SSL Validation

Please select a remote party validation policy.

Accept CA verified certificate

Optionally you may upload a certificate for validation

**Certificate for validation (.pfx):**

To upload a file
**click or drag**

Ozeki_CA.pub_1.pfx (1.12 KB)

☐ Do not accept remote certificate change

☒ Export received certificate

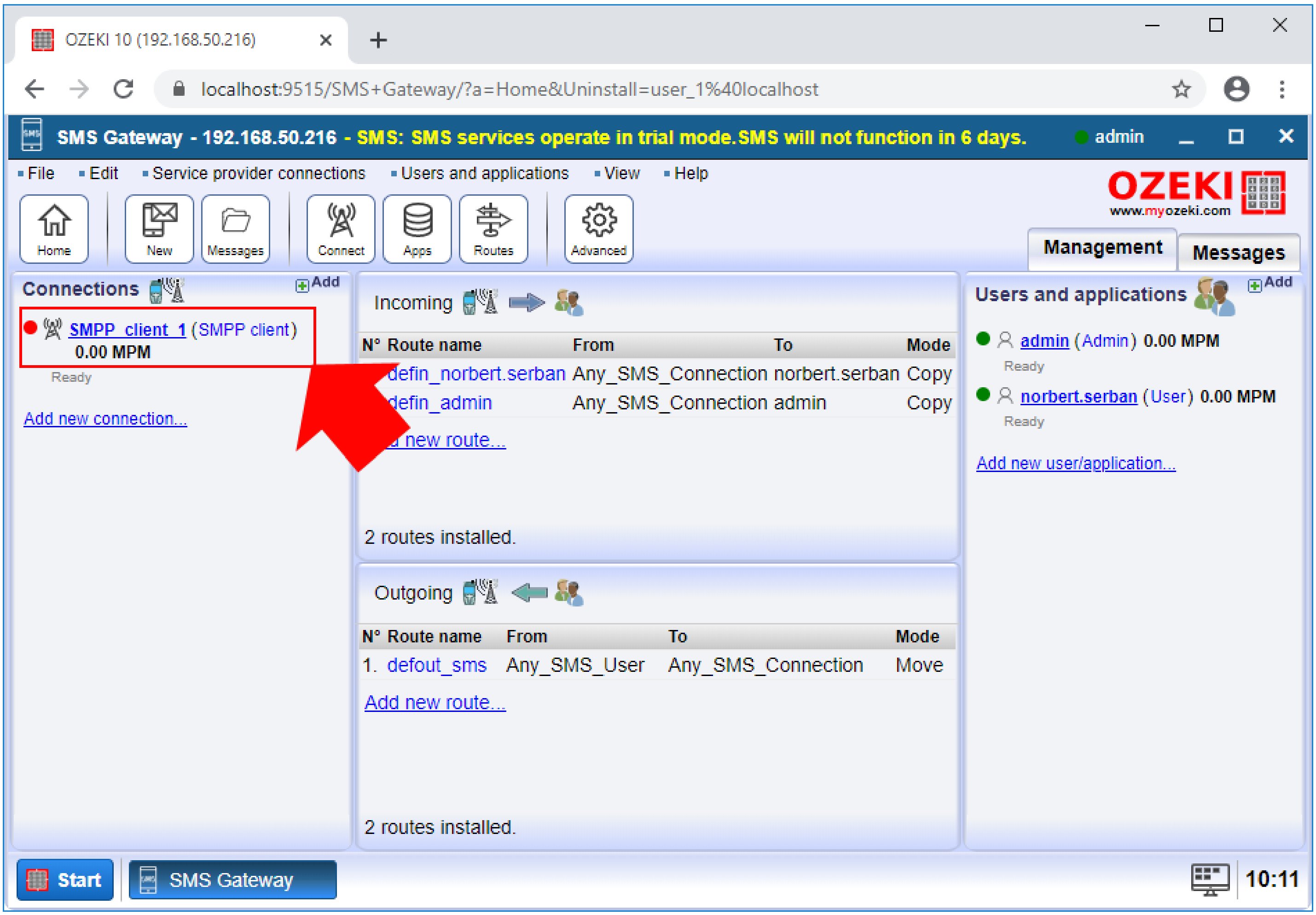
Log level

User authentication

Figure 22 - Configure CA validation

# III. Configure SSL connection for the client

For client side configuration first of all you need to send the client's certificate to your client in a secure way. After it, open the client connection in the main page.


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=Home&Uninstall=user_1%40localhost
SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode.SMS will not function in 6 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
OZEKI
www.myozeki.com
Management
Messages
Start
SMS Gateway
10:11

Connections Add

SMPP_client_1 (SMPP client) 0.00 MPM
Ready

Add new connection...

Incoming

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| | defin_norbert.serban | Any_SMS_Connection | norbert.serban | Copy |
| | defin_admin | Any_SMS_Connection | admin | Copy |

d new route...

2 routes installed.

Outgoing

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move |

Add new route...

2 routes installed.

Users and applications Add

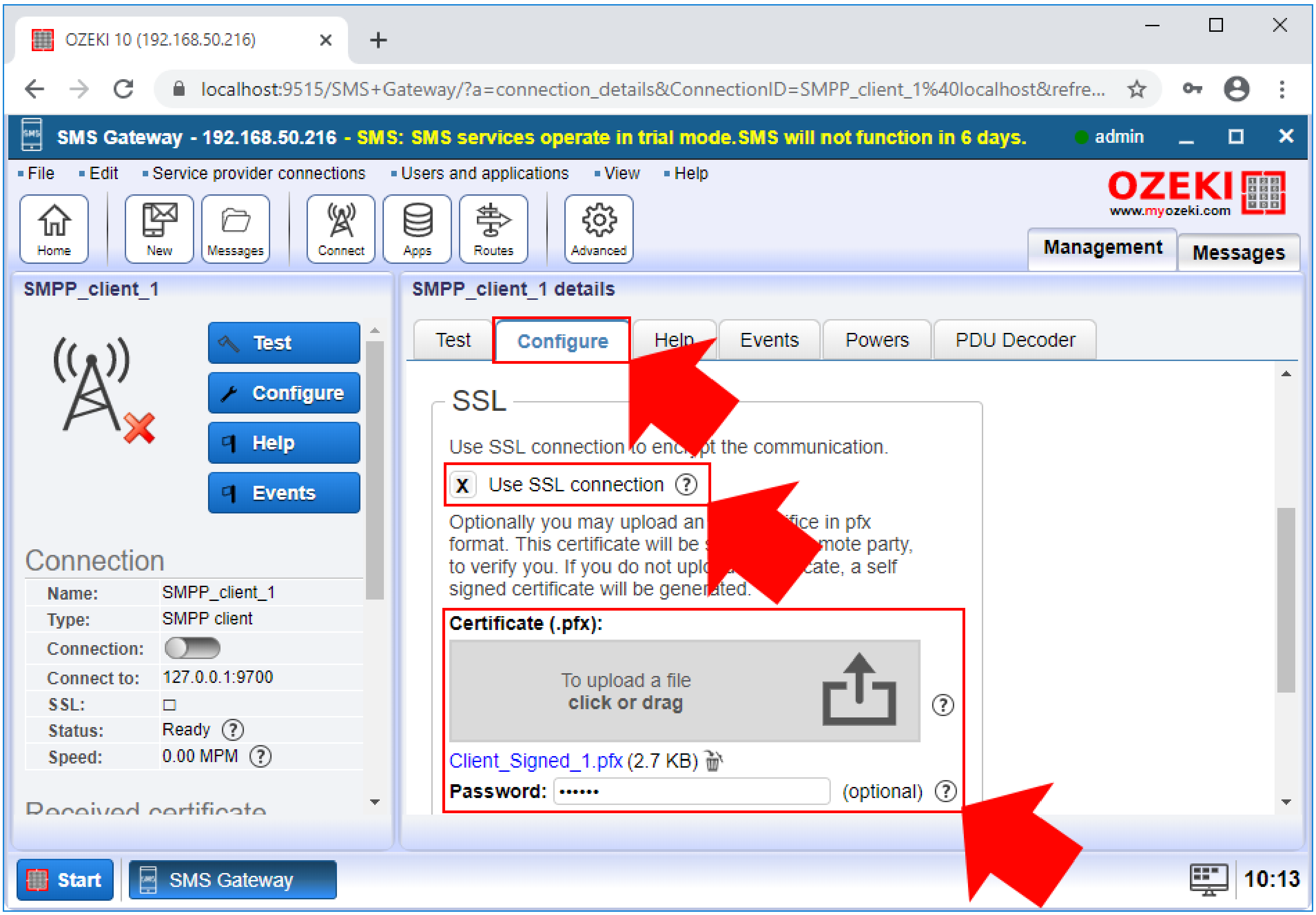
admin (Admin) 0.00 MPM
Ready

norbert.serban (User) 0.00 MPM
Ready

Add new user/application...

Figure 23 - Open client connection

SSL connection configuration is same as the server side. In the **Configure** menu select the **Advanced** tab and open **SSL**. Under SSL section check the "**Use SSL connection option**". **Upload** the server signed certificate and provid the **password** for it.


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=connection_details&ConnectionID=SMPP_client_1%40localhost&refre...
SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode.SMS will not function in 6 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
OZEKI
www.myozeki.com
Management
Messages
SMPP_client_1
Test
Configure
Help
Events
SMPP_client_1 details
Test
Configure
Help
Events
Powers
PDU Decoder
Start
SMS Gateway
10:13

Connection

Name: SMPP_client_1
Type: SMPP client
Connection:
Connect to: 127.0.0.1:9700
SSL: □
Status: Ready
Speed: 0.00 MPM

Received certificate

SSL

Use SSL connection to encrypt the communication.

☒ Use SSL connection

Optionally you may upload an certificate in pfx format. This certificate will be sent to the remote party, to verify you. If you do not upload a certificate, a self signed certificate will be generated.

Certificate (.pfx):

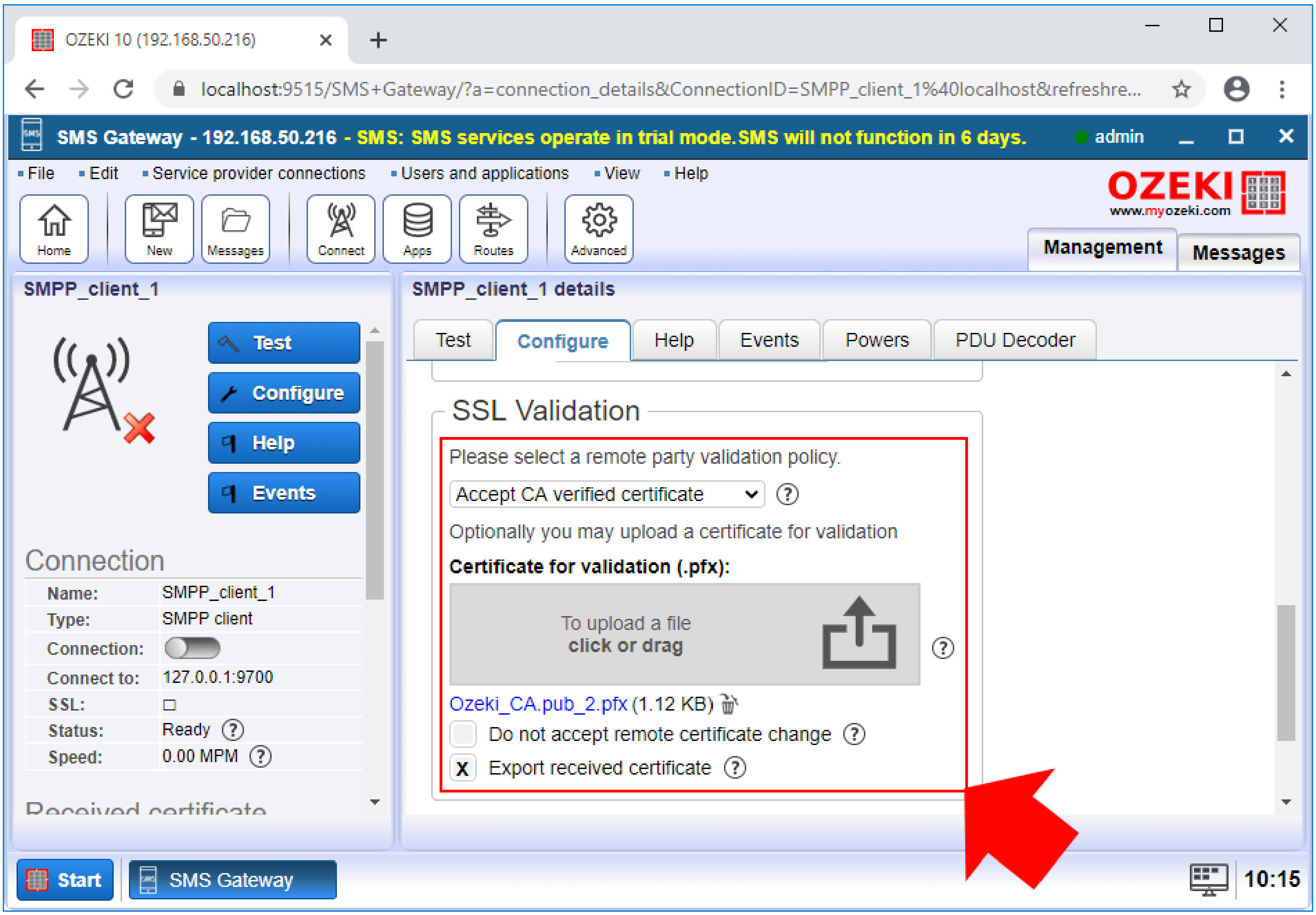
To upload a file
**click or drag**

Client_Signed_1.pfx (2.7 KB)

**Password:** •••••• (optional)

Figure 24 - Enable SSL connection for the client

For CA validation open the **SSL Validation** section. Select the "**Accept CA verified certificate**" form the list and **upload** the public CA certificate.



OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=connection_details&ConnectionID=SMPP_client_1%40localhost&refreshre...
SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode.SMS will not function in 6 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
SMPP_client_1
Test
Configure
Help
Events
SMPP_client_1 details
Test
Configure
Help
Events
Powers
PDU Decoder
Start
SMS Gateway
10:15

SSL Validation

Please select a remote party validation policy.

Accept CA verified certificate

Optionally you may upload a certificate for validation

**Certificate for validation (.pfx):**

To upload a file
**click or drag**

Ozeki_CA.pub_2.pfx (1.12 KB)

□ Do not accept remote certificate change

☒ Export received certificate

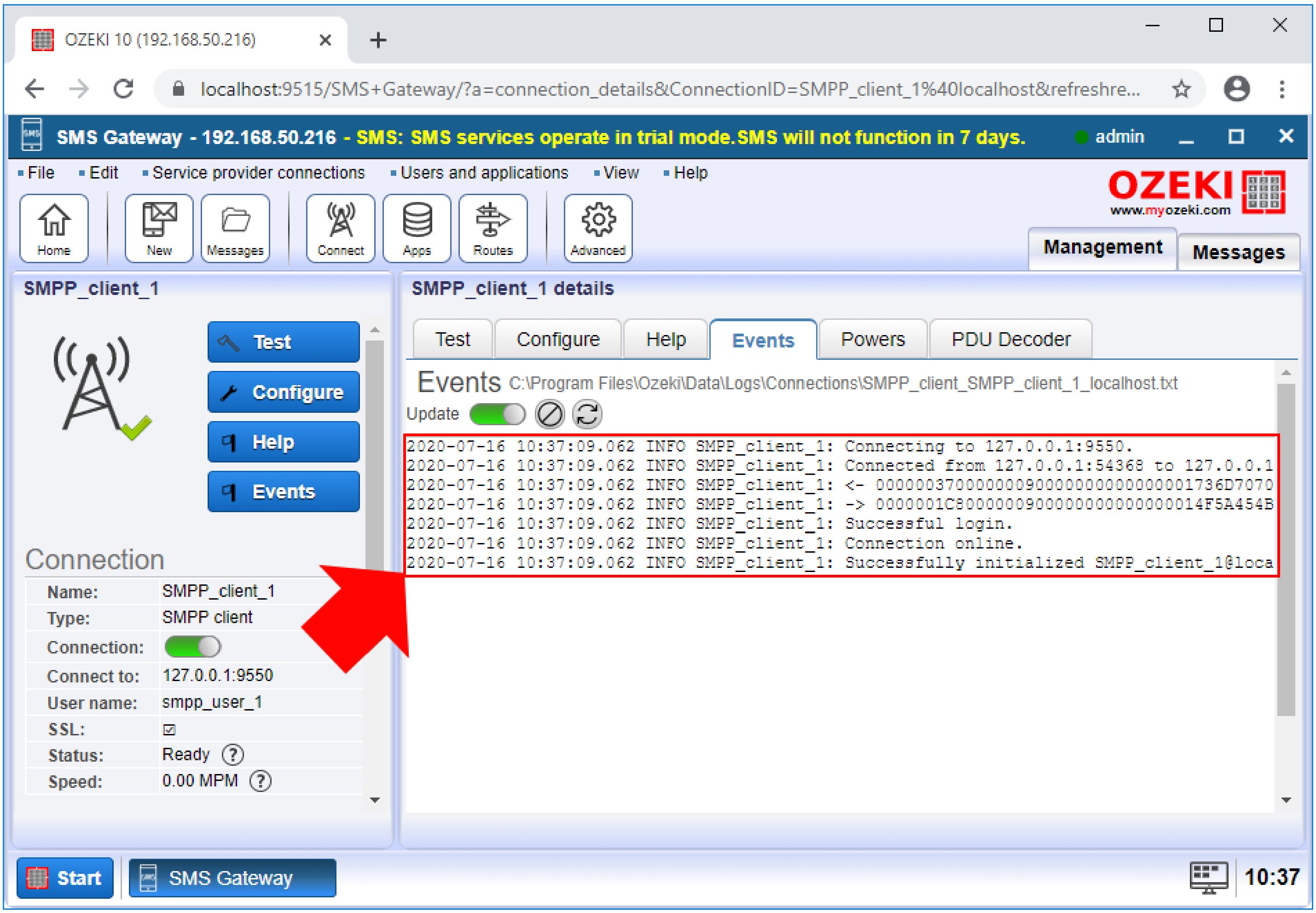
Connection

| | |
|---|---|
| Name: | SMPP_client_1 |
| Type: | SMPP client |
| Connection: | |
| Connect to: | 127.0.0.1:9700 |
| SSL: | □ |
| Status: | Ready |
| Speed: | 0.00 MPM |

Received certificate

Figure 25 - Configure CA validation

Now you can see on the client side the secure connection is succesfully initialzed.


OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=connection_details&ConnectionID=SMPP_client_1%40localhost&refreshre...
SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
OZEKI
www.myozeki.com
Management
Messages
SMPP_client_1
Test
Configure
Help
Events
SMPP_client_1 details
Test
Configure
Help
Events
Powers
PDU Decoder
Update
Start
SMS Gateway
10:37

Events C:\Program Files\Ozeki\Data\Logs\Connections\SMPP_client_SMPP_client_1_localhost.txt

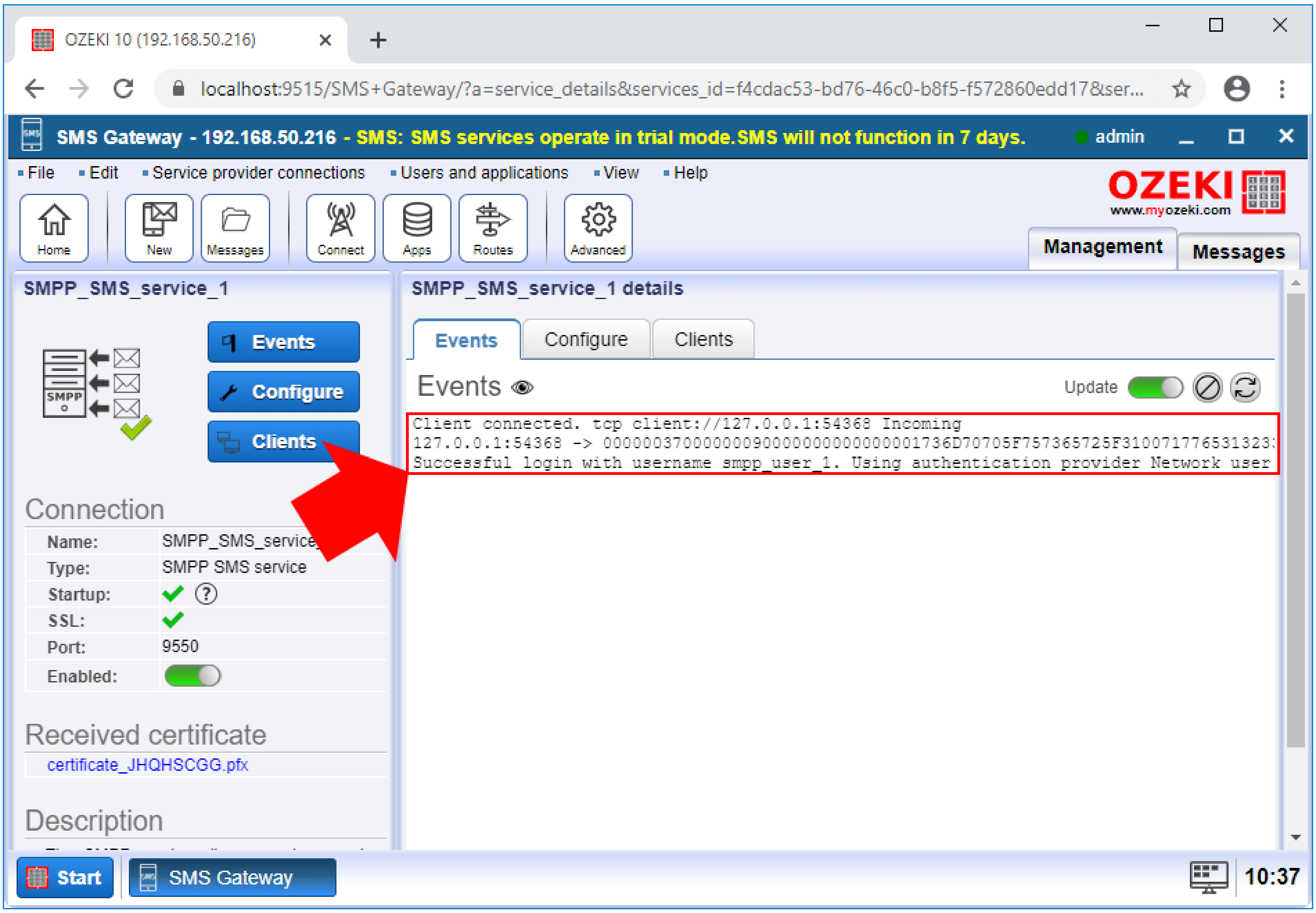
```
2020-07-16 10:37:09.062 INFO SMPP_client_1: Connecting to 127.0.0.1:9550.
2020-07-16 10:37:09.062 INFO SMPP_client_1: Connected from 127.0.0.1:54368 to 127.0.0.1
2020-07-16 10:37:09.062 INFO SMPP_client_1: <- 00000037000000090000000000000001736D7070
2020-07-16 10:37:09.062 INFO SMPP_client_1: -> 0000001C80000009000000000000000014F5A454B
2020-07-16 10:37:09.062 INFO SMPP_client_1: Successful login.
2020-07-16 10:37:09.062 INFO SMPP_client_1: Connection online.
2020-07-16 10:37:09.062 INFO SMPP_client_1: Successfully initialized SMPP_client_1@loca
```

Connection

Name: SMPP_client_1
Type: SMPP client
Connection:
Connect to: 127.0.0.1:9550
User name: smpp_user_1
SSL: ☑
Status: Ready
Speed: 0.00 MPM

Figure 26 - Client successfully connected

And the server side connected too.



OZEKI 10 (192.168.50.216)
localhost:9515/SMS+Gateway/?a=service_details&services_id=f4cdac53-bd76-46c0-b8f5-f572860edd17&ser...
SMS Gateway - 192.168.50.216 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
OZEKI
www.myozeki.com
Management
Messages
SMPP_SMS_service_1
Events
Configure
Clients
SMPP_SMS_service_1 details
Events
Configure
Clients
Events
Update
Start
SMS Gateway
10:37

Connection

| | |
|---|---|
| Name: | SMPP_SMS_service |
| Type: | SMPP SMS service |
| Startup: | ✓ ? |
| SSL: | ✓ |
| Port: | 9550 |
| Enabled: | ● |

Received certificate

certificate_JHQHSCGG.pfx

Description

```
Client connected. tcp client://127.0.0.1:54368 Incoming
127.0.0.1:54368 -> 00000037000000090000000000000001736D70705F757365725F3100717765313233
Successful login with username smpp_user_1. Using authentication provider Network user
```

Figure 27 - Server successfully connected

# SMPP user authentication

***This page lists the available options that you can use to authenticate your users or employees. By managing the authentication process, you can give these users right to access SMS Gateway. You can use an SQL database, LDAP server or HTTP to manage the rights, and give permissions to your users.***

**SQL user authentication**

The SQL user authentication feature ensures the Database Authentication Provider feature which can be used to store user credentials in your SQL database. By this solution, you can compare the username and the hashed password in case of an login attempt with the database.

Check how to use SQL user authentication in SMS Gateway

**LDAP user authentication**

By checking this page, you can learn about how to authenticate the users registered in Microsoft Domain. Ozeki SMS Gateway provides the authentication of these users by using LDAP protocol that can reach the user database in Windows Server and check if the user is allowed to log in.

See how you can use LDAP protocol to authenticate users

**HTTP user authentication**

The HTTP user authentication option helps you to manage permissions with the HTTP Authentication Provider. This authentication service uses a HTTP webserver which will check the username and password and allow the user to log in.

Learn about how to use HTTP user authentication

# How to authenticate SMPP users with a database

***The Database Authentication Provider can be your own SQL database storing user credentials. You are probably storing your customer or employee base who have registered at your company. You can give these people rights to access Ozeki SMS Gateway. The authentication is done with SQL queries. When there is a login attempt Ozeki SMS Gateway compares the username and hashed password pairs with your SQL records.***

## Step 1 - Create Database Connection in Ozeki SMS Gateway

The first step is to open the Control Panel application in Ozeki SMS Gateway. So, just navigate to the desktop of Ozeki SMS Gateway, and here, as you can see it in Figure 1, just open Control Panel by clicking on its icon.

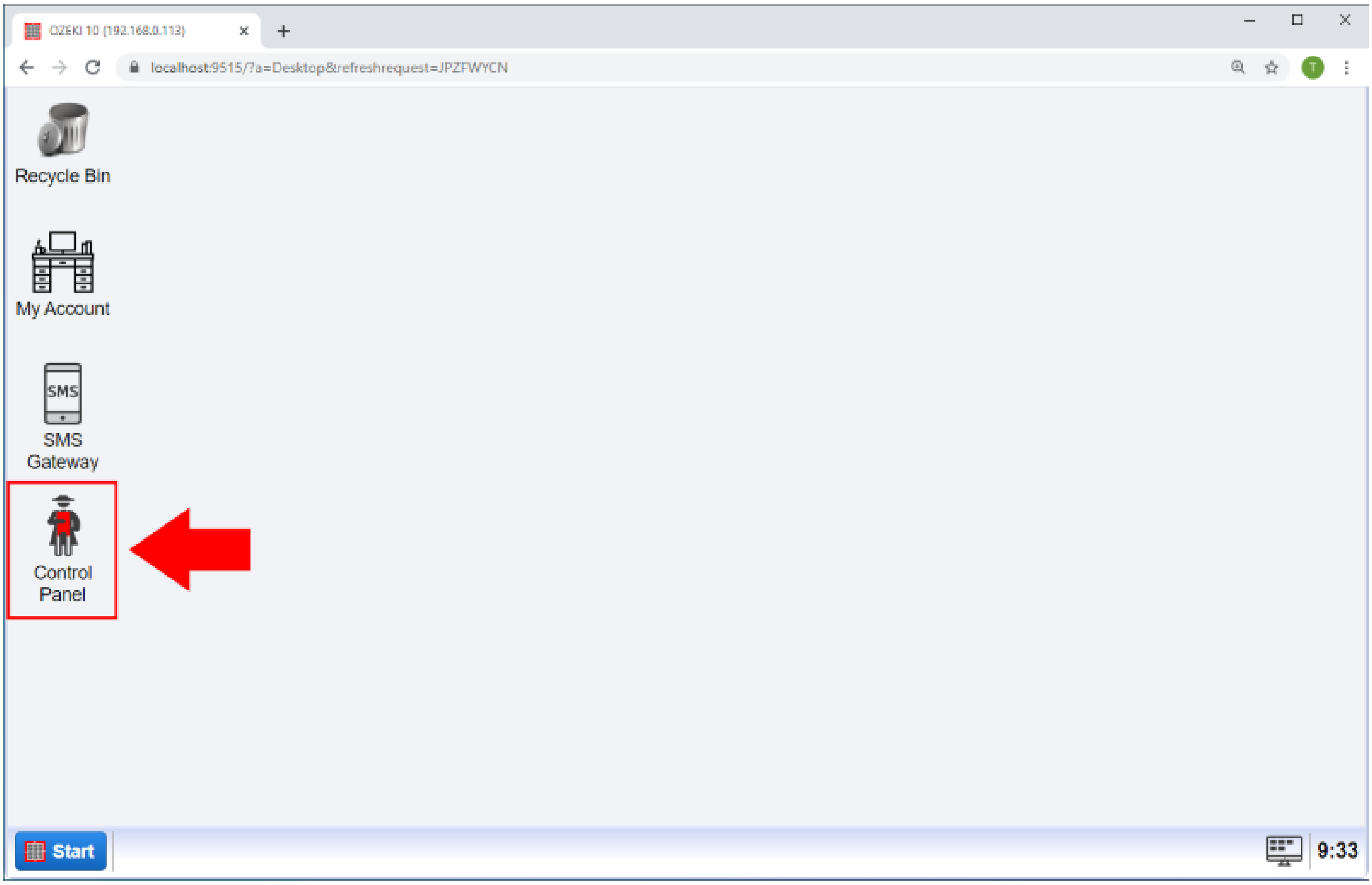


Figure 1 - Open Control Panel

Now you should be on the Control Panel main page where you can create, modify or delete connections. Click on the blue Create new connection button and select Application from the box appearing on the right side of the screen (Figure 2).

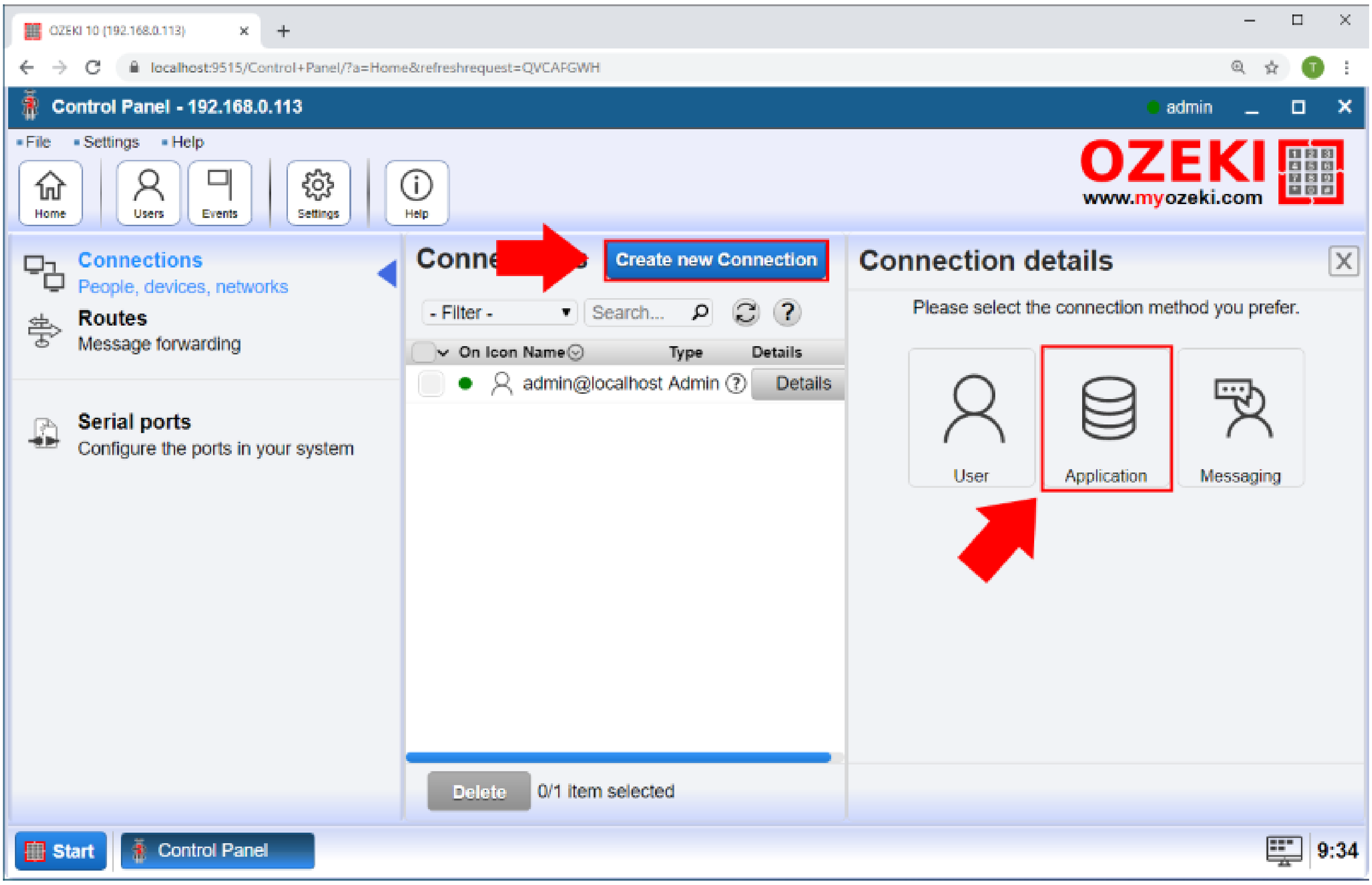


Figure 2 - Create new application connection

In this example we will use MySQL Database to store the user details for the Authentication Provider. So select the MySQL connection type as the Figure 3-5 shows.

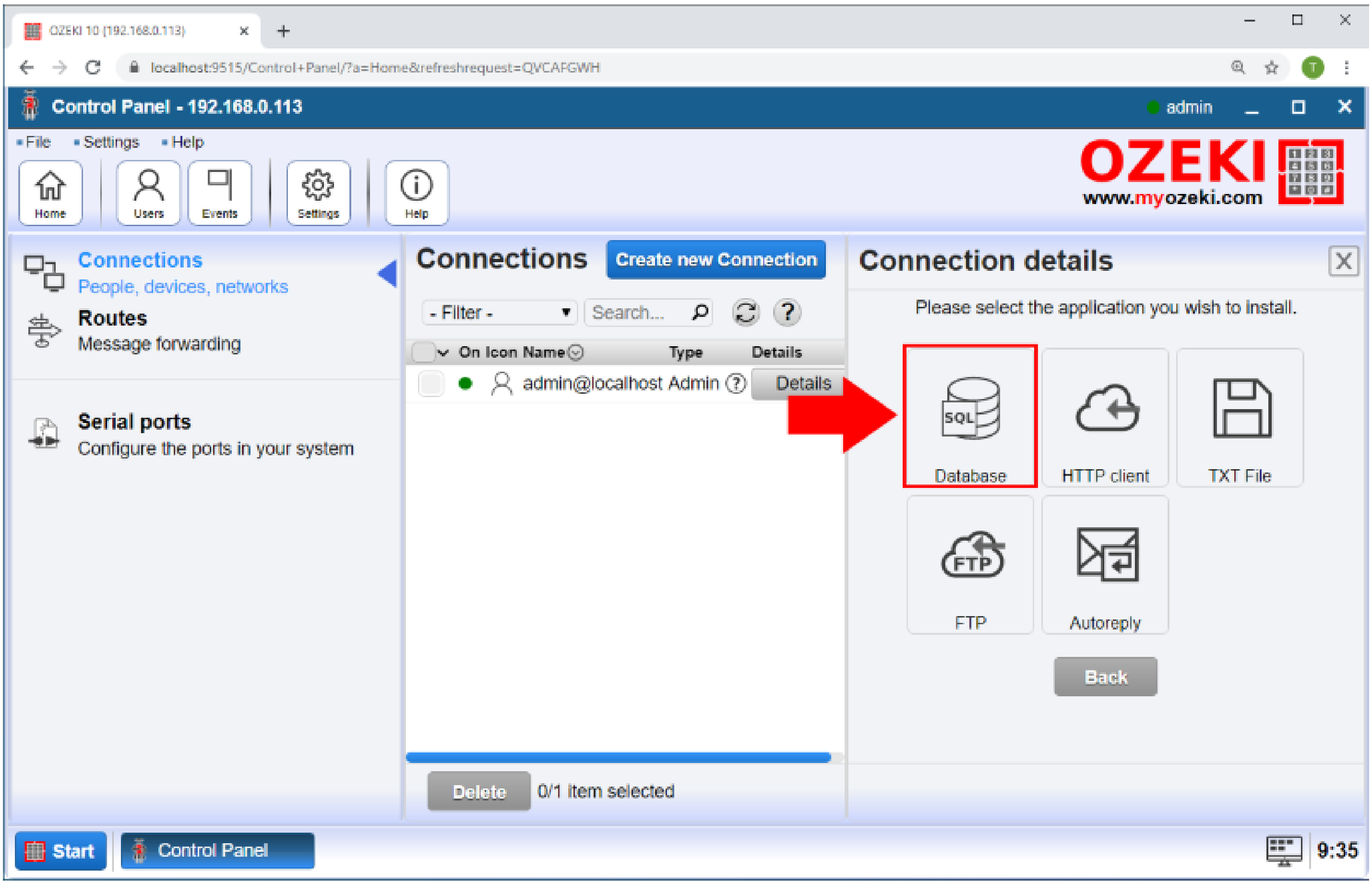


Figure 3 - Create database connection

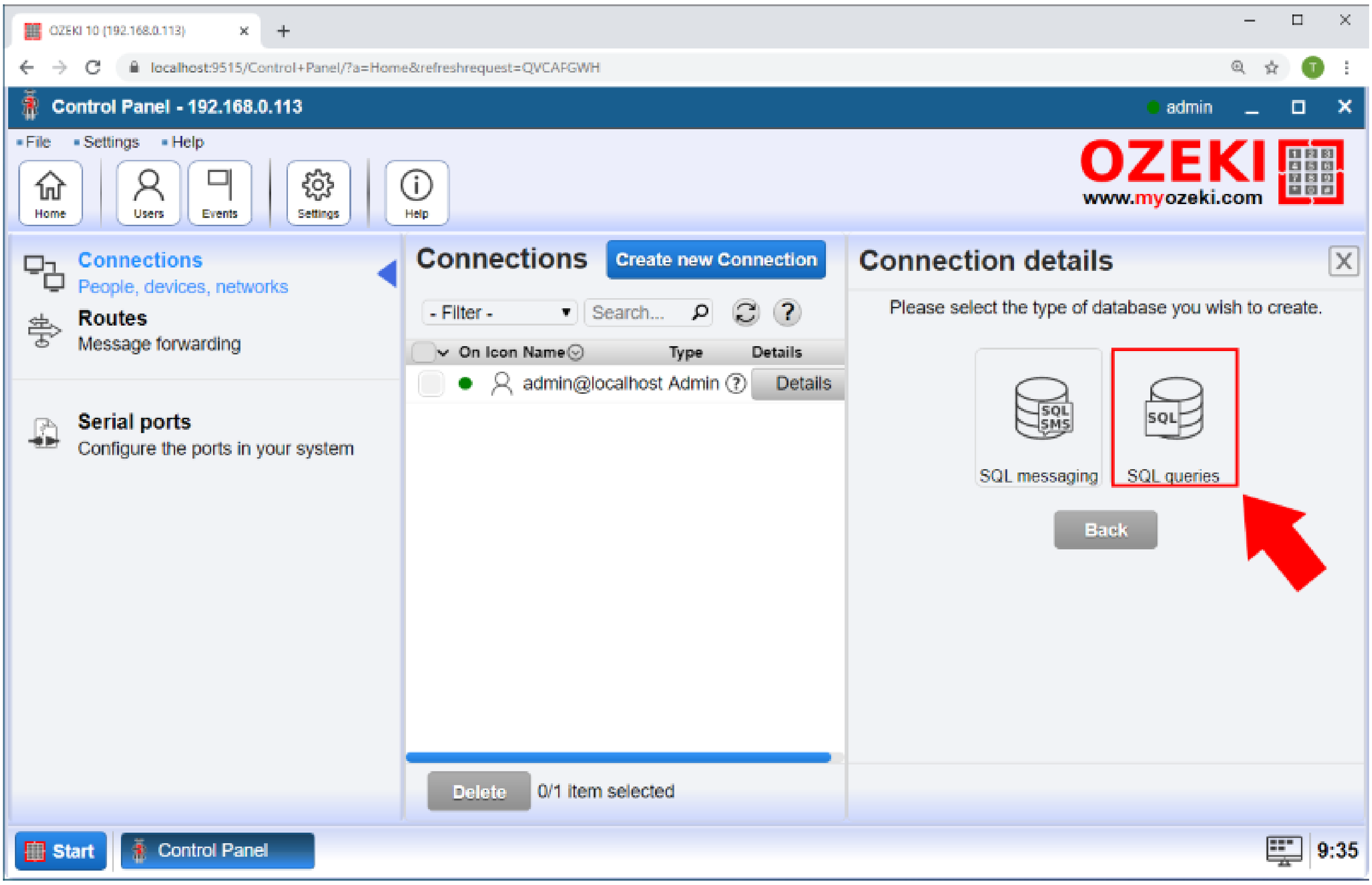


Figure 4 - SQL queries connection

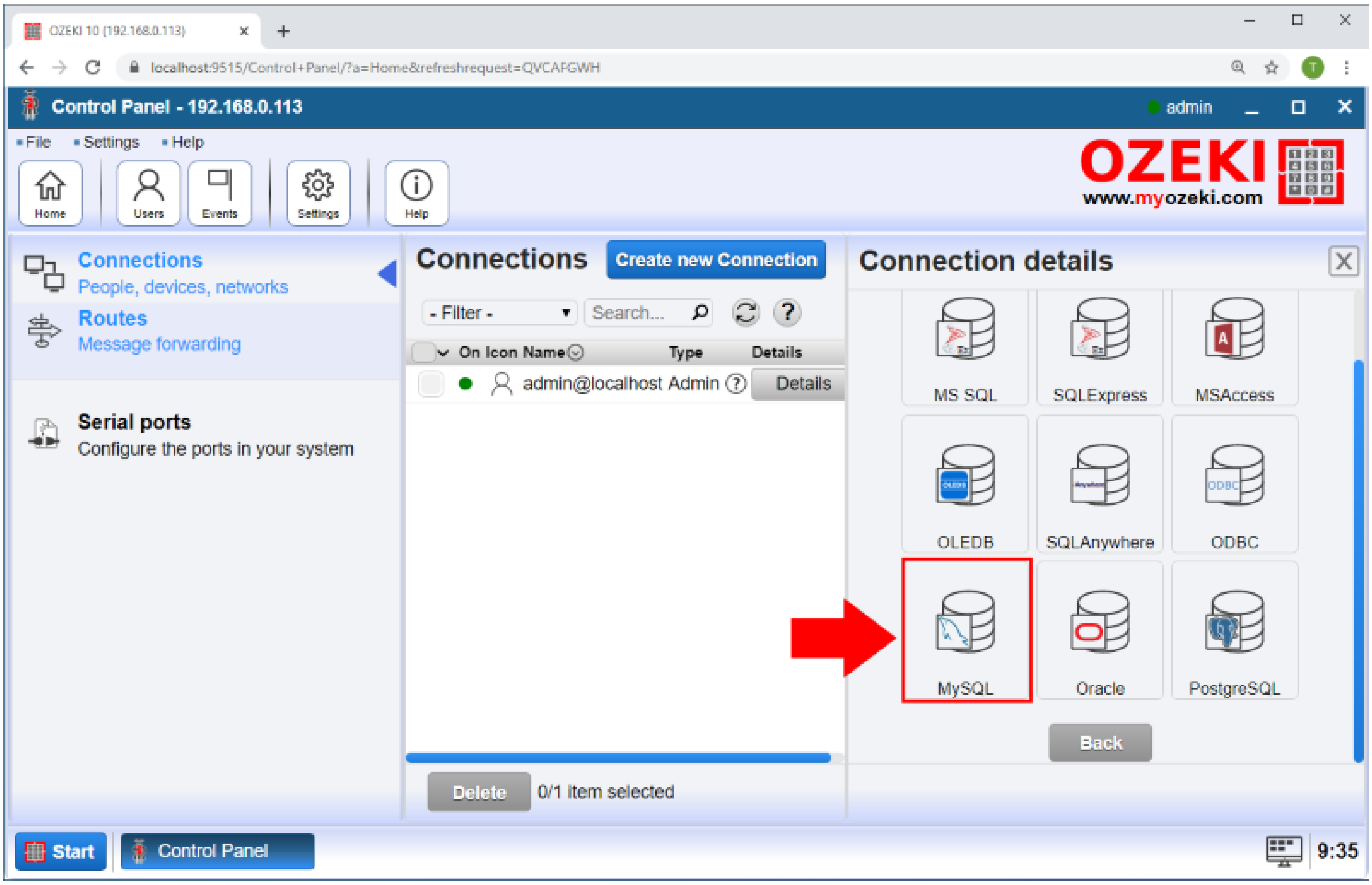
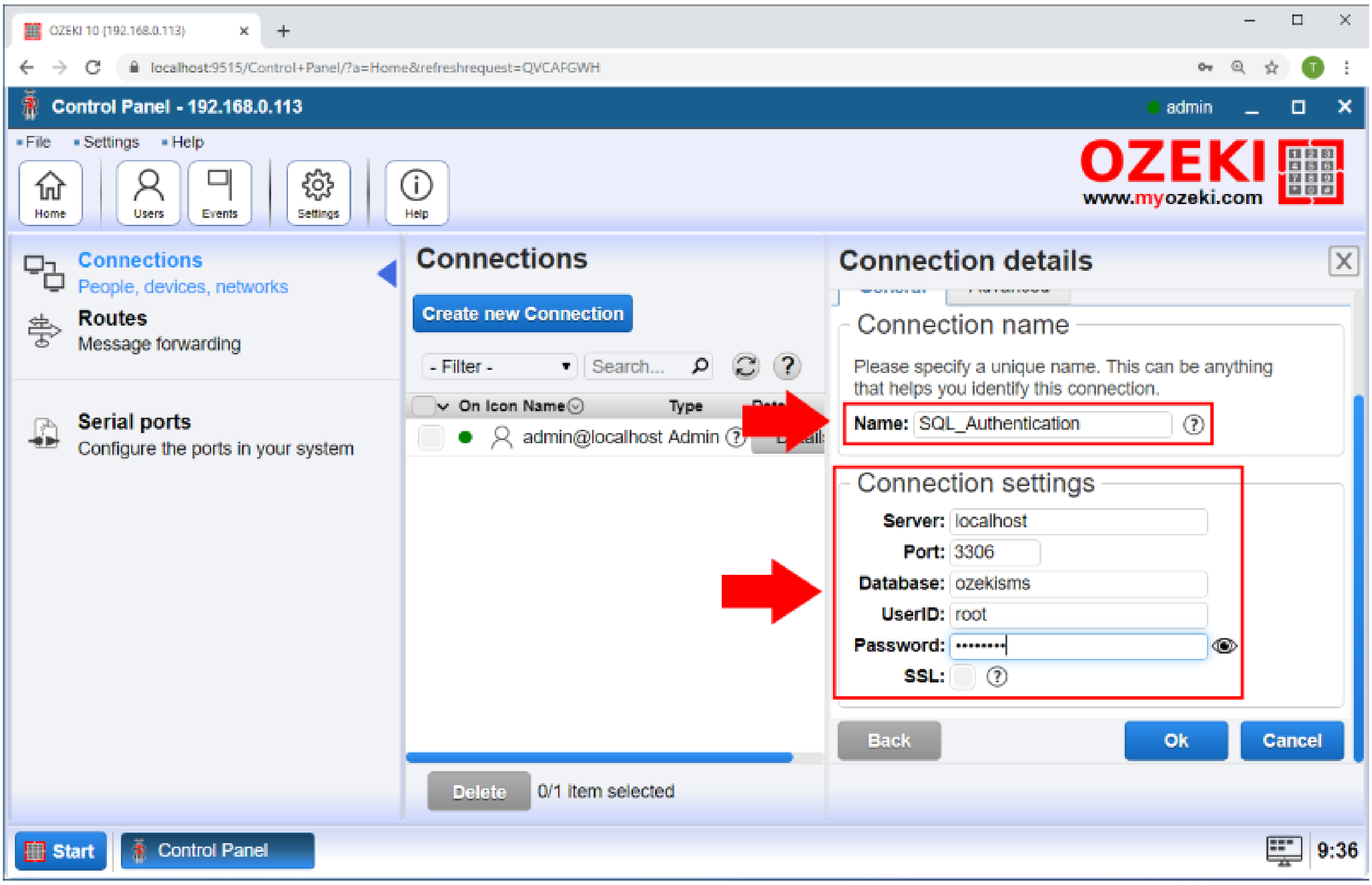


Figure 5 - MySQL Connection

Then please fill out the following form with connection details. Make sure to provide the SQL server IP and port number. The default SQL port number is usually 3306. You also need to provide the name of the database where your users are. Finally provide the SQL username and password and click 'OK' (Figure 6).



Connection details

Connection name

Please specify a unique name. This can be anything that helps you identify this connection.

Name: SQL_Authentication

Connection settings

Server: localhost

Port: 3306

Database: ozekisms

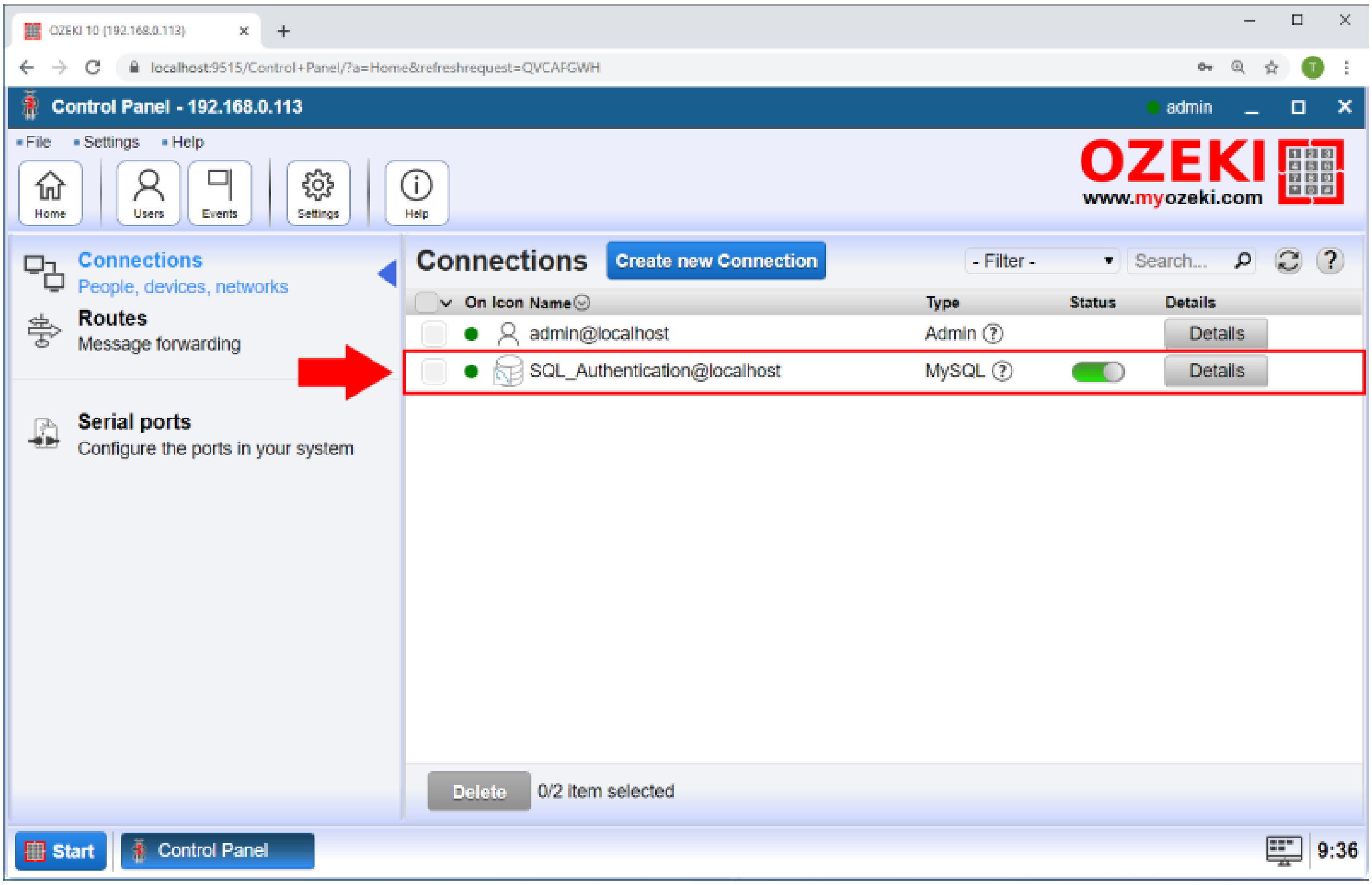
UserID: root

Password: ••••••••

SSL: □

Figure 6 - MySQL Connection details

Finally the connection was successful as you can see it on Figure 7.



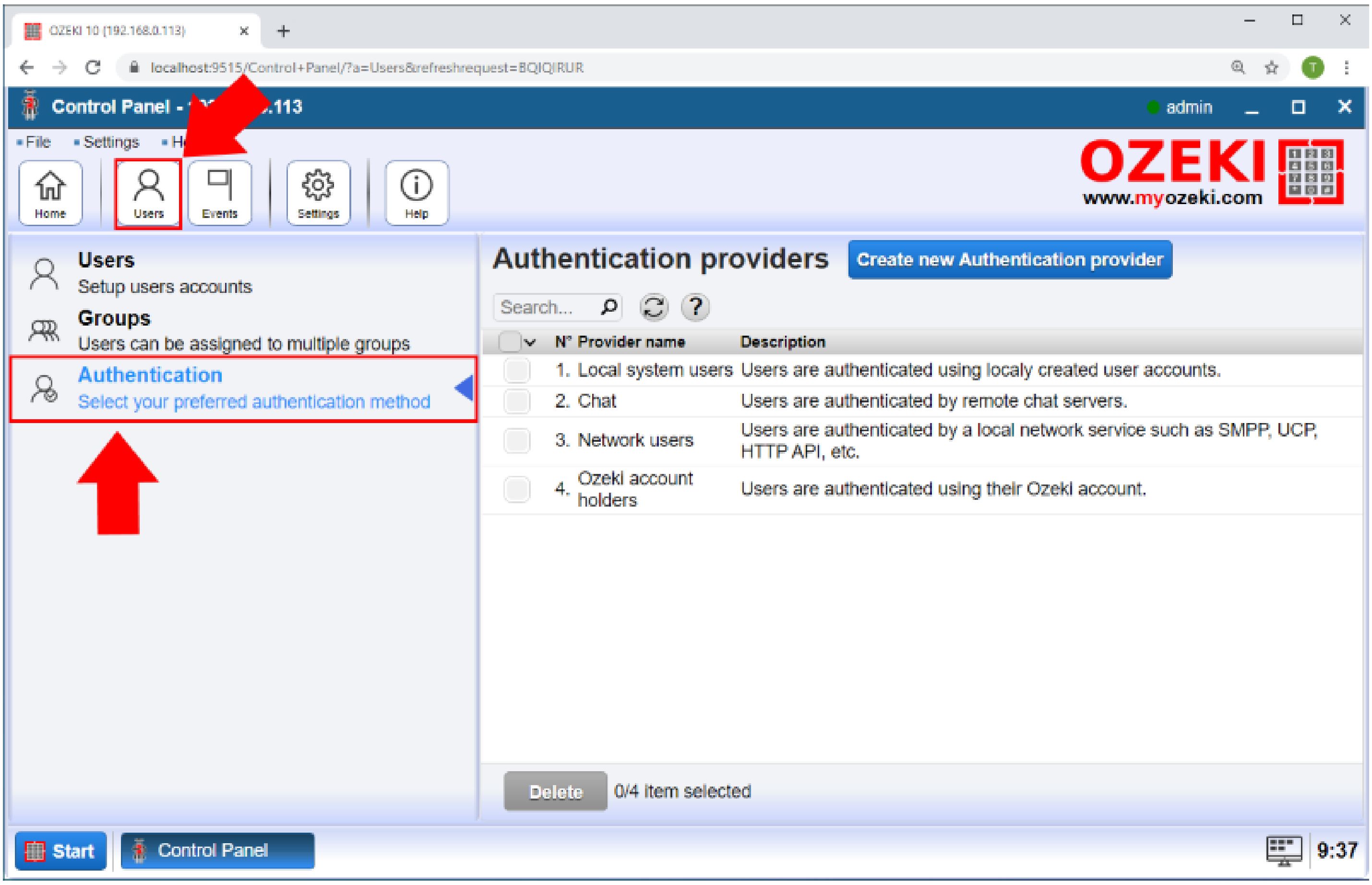
| On | Icon | Name | Type | Status | Details |
|---|---|---|---|---|---|
| ● | | admin@localhost | Admin | | Details |
| ● | | SQL_Authentication@localhost | MySQL | | Details |

0/2 item selected

Figure 7 - Database connected

# Step 2 - Connect Database Authentication Provider to the SQL database

In this step we will set a SELECT query on the SQL table. This SELECT query will be called when a login attempt occurs. It checks the user table for users. Jump to the authentication provider page by clicking Authentication providers in the Users menu (Figure 8).



Users
Setup users accounts

Groups
Users can be assigned to multiple groups

Authentication
Select your preferred authentication method

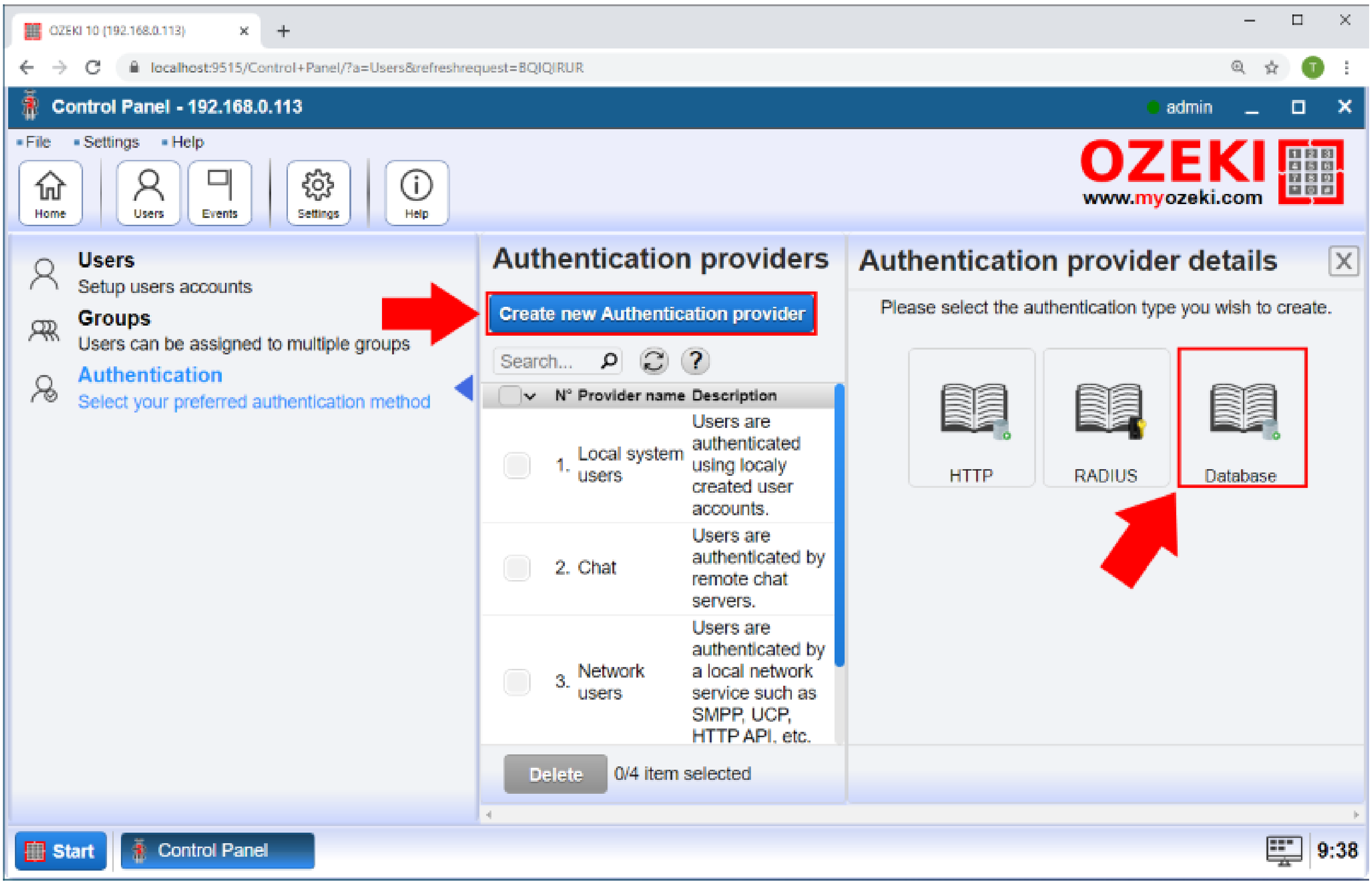
## Authentication providers

| N° | Provider name | Description |
|---|---|---|
| 1. | Local system users | Users are authenticated using localy created user accounts. |
| 2. | Chat | Users are authenticated by remote chat servers. |
| 3. | Network users | Users are authenticated by a local network service such as SMPP, UCP, HTTP API, etc. |
| 4. | Ozeki account holders | Users are authenticated using their Ozeki account. |

0/4 item selected

Figure 8 - Open Authentication providers menu

On the Authentication provider page you can create, modify or delete authentication providers. Create a new authentication provider by clicking the blue Create new authentication provider button. A new box will open on the right side of the page. In this box you can select between authentication providers. Please choose Database authentication provider (Figure 9).



Authentication providers

| N° | Provider name | Description |
|---|---|---|
| 1. | Local system users | Users are authenticated using localy created user accounts. |
| 2. | Chat | Users are authenticated by remote chat servers. |
| 3. | Network users | Users are authenticated by a local network service such as SMPP, UCP, HTTP API, etc. |

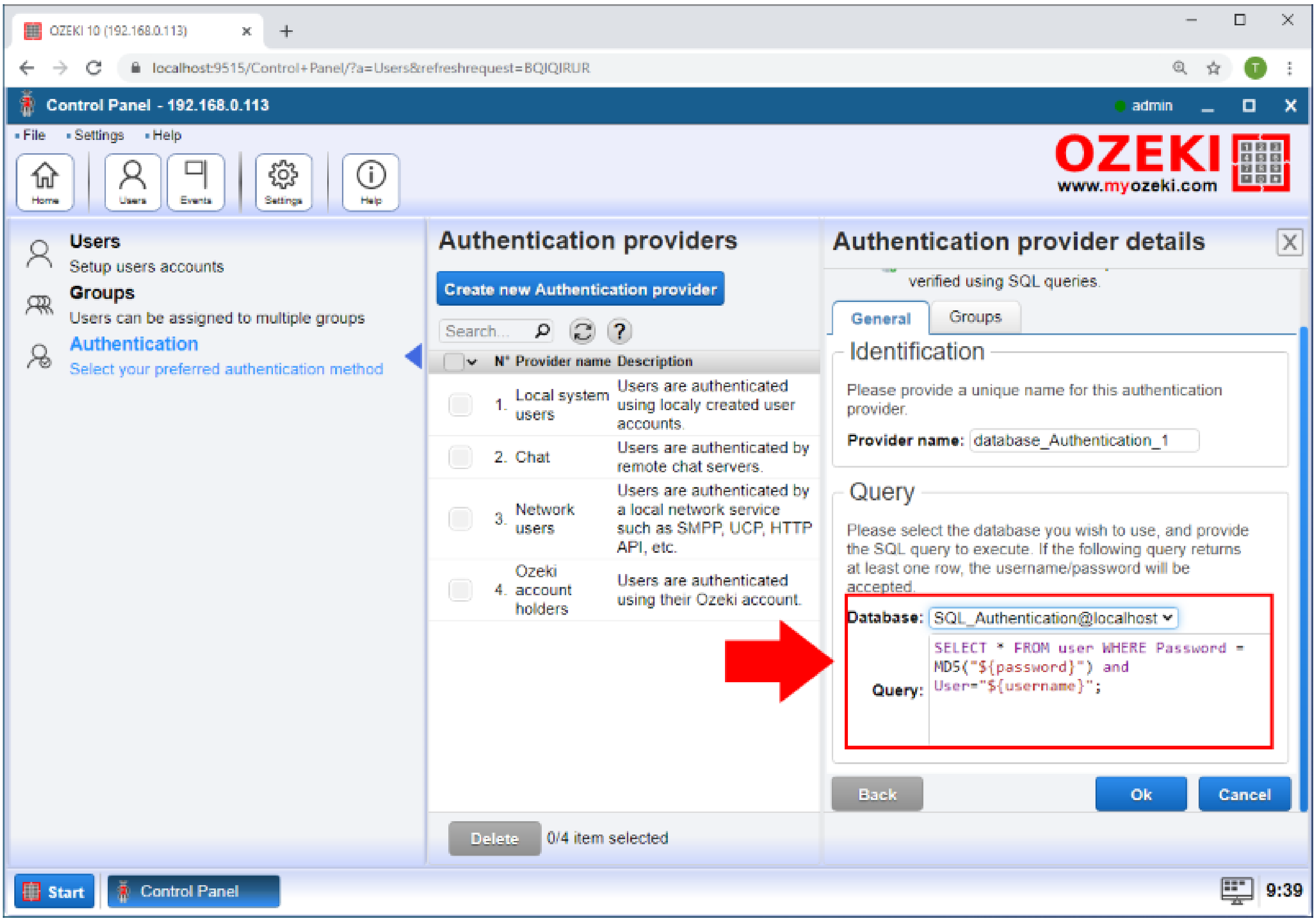
Authentication provider details

Please select the authentication type you wish to create.

Figure 9 - Create new database provider

A form will be available in the box. You should fill out the form to configure the database authentication provider. In the most important field you need to write the SQL query as you can see it in Figure 10. In STEP 4 we will set up an SQL database that is compatible with this SQL query.

```sql
SELECT * FROM user WHERE
Password = MD5("${password}") /* Hashes password and compares the hash. */
and User="${username}";
```



| N° | Provider name | Description |
|---|---|---|
| 1. | Local system users | Users are authenticated using localy created user accounts. |
| 2. | Chat | Users are authenticated by remote chat servers. |
| 3. | Network users | Users are authenticated by a local network service such as SMPP, UCP, HTTP API, etc. |
| 4. | Ozeki account holders | Users are authenticated using their Ozeki account. |

verified using SQL queries.

Identification

Please provide a unique name for this authentication provider.

**Provider name:** database_Authentication_1

Query

Please select the database you wish to use, and provide the SQL query to execute. If the following query returns at least one row, the username/password will be accepted.

**Database:** SQL_Authentication@localhost

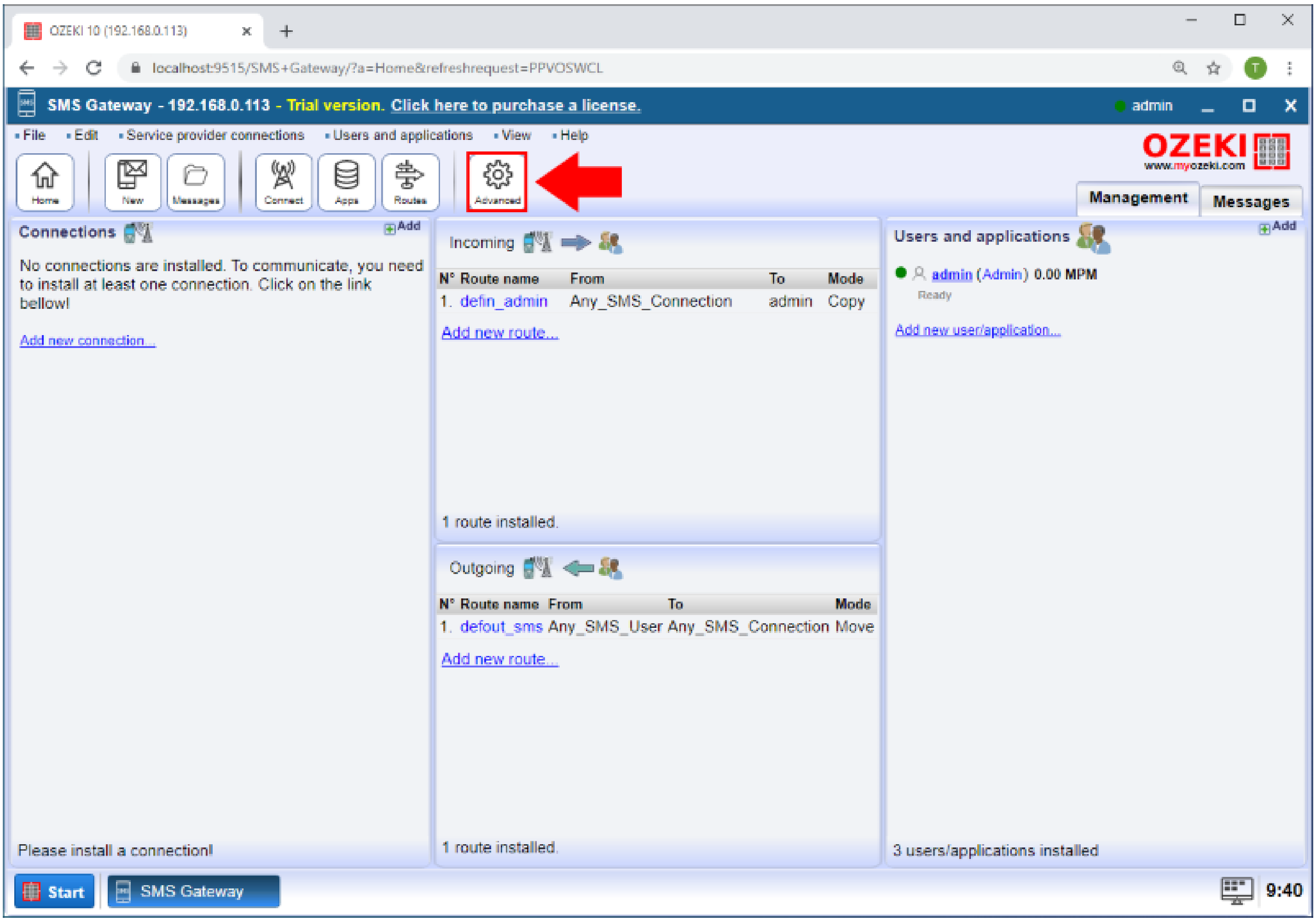
**Query:**

```
SELECT * FROM user WHERE Password =
MD5("${password}") and
User="${username}";
```

Figure 10 - Provide SQL query

## Step 3 - Create SMPP Service with SQL Authentication

The next step is to open the SMS Gateway application. So, just navigate to the desktop of, and here just open SMS Gateway by clicking on its icon. Here syu can find the Advanced menu of the SMS Gateway. To open it please click the "Advanced" button on the main page.



No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

Please install a connection!

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defin_admin | Any_SMS_Connection | admin | Copy |

Add new route...

1 route installed.

Outgoing

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move |

Add new route...

1 route installed.

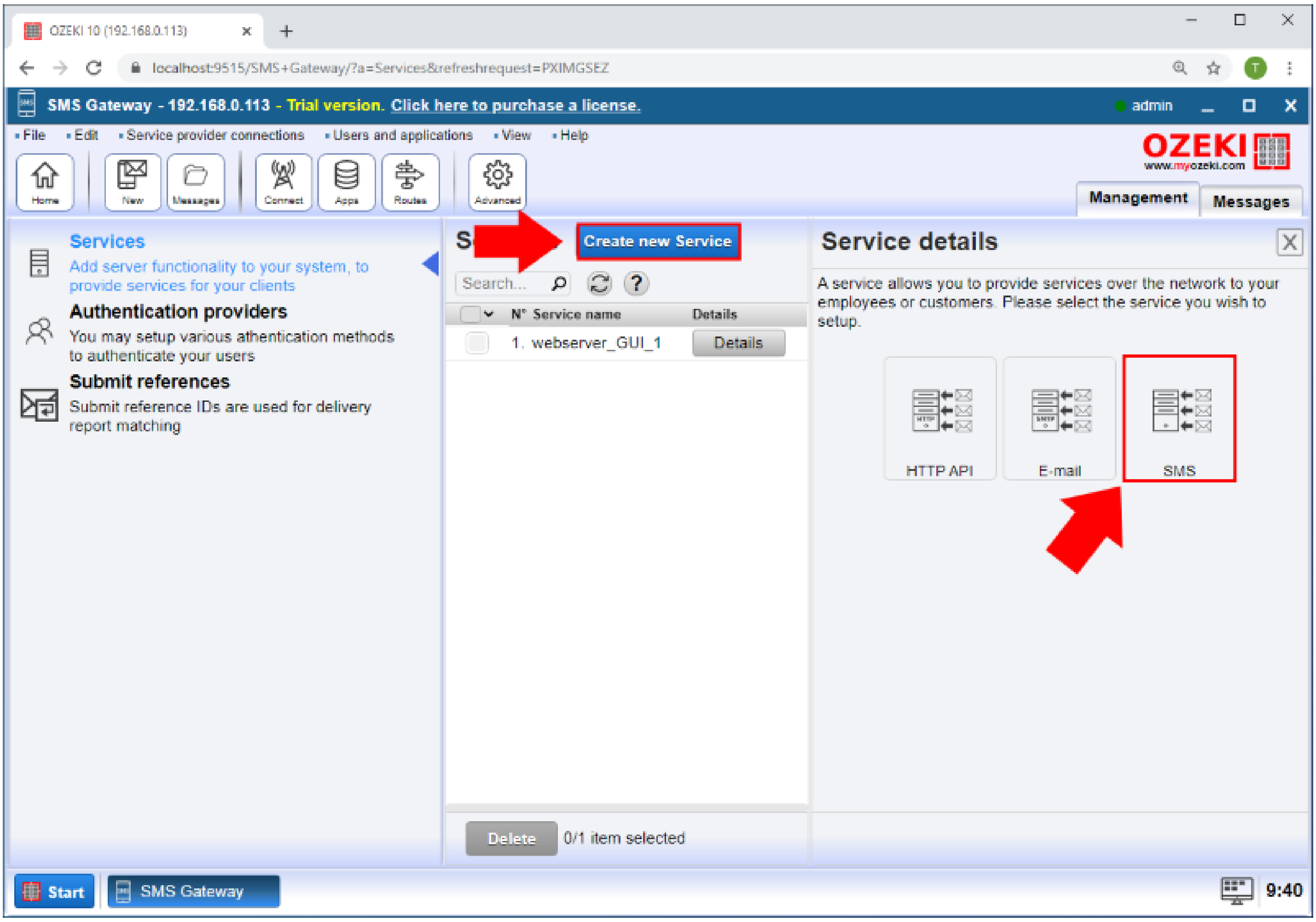
● admin (Admin) 0.00 MPM
Ready

Add new user/application...

3 users/applications installed

Figure 11 - Open SMS Gateway advanced menu

On the Advanced page you can create, modify or delete SMS Services. Create a new SMS Services by clicking the blue Create new Service button. A new box will open on the right side of the page. In this box you can select between SMS Services as you can see in the Figure 12. And then please choose SMPP Service (Figure 13).



SMS Gateway - 192.168.0.113 - Trial version. Click here to purchase a license.

**Services**
Add server functionality to your system, to provide services for your clients

**Authentication providers**
You may setup various athentication methods to authenticate your users

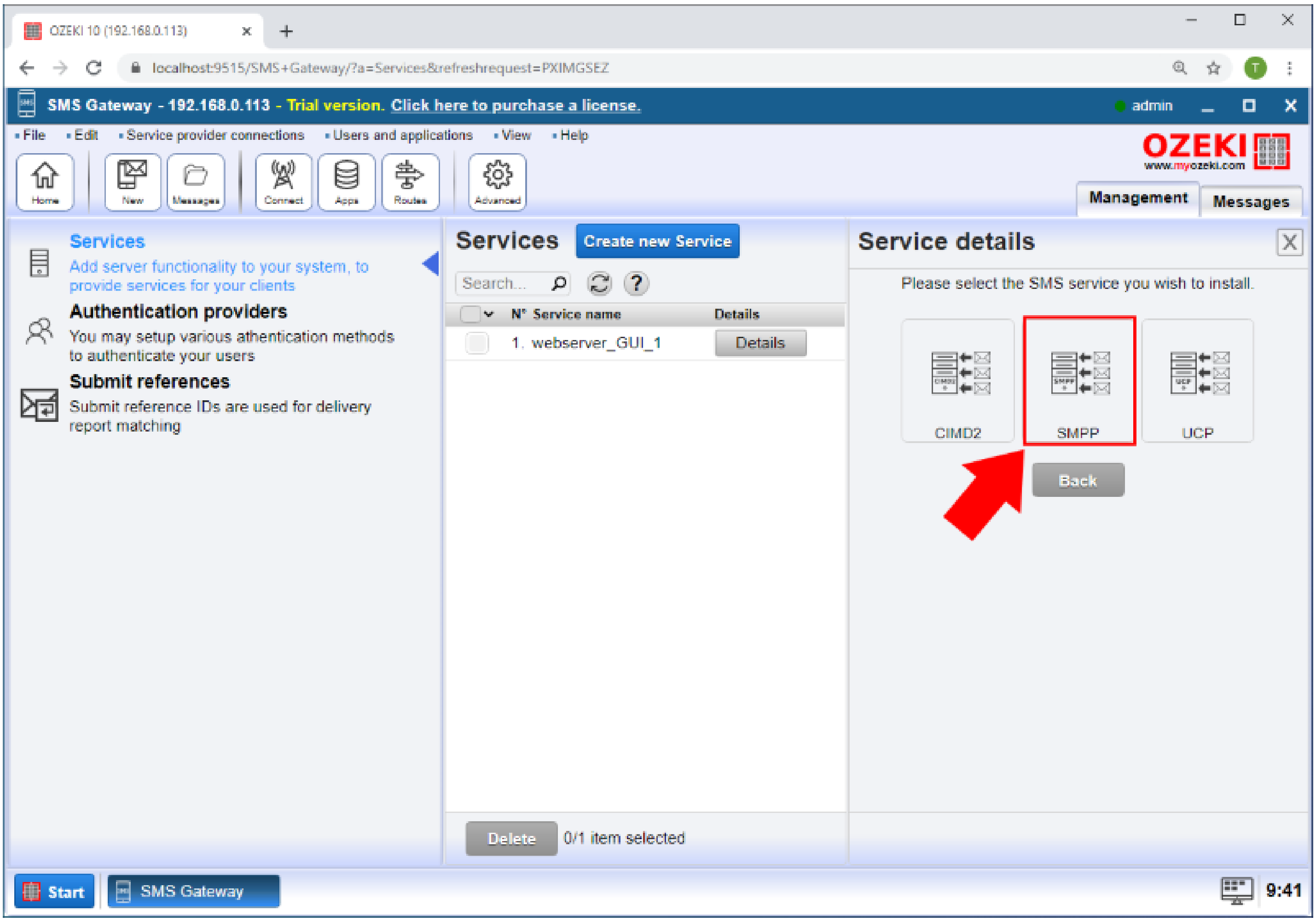
**Submit references**
Submit reference IDs are used for delivery report matching

| N° | Service name | Details |
|---|---|---|
| 1. | webserver_GUI_1 | Details |

0/1 item selected

**Service details**
A service allows you to provide services over the network to your employees or customers. Please select the service you wish to setup.

Figure 12 - Create new SMS Service



SMS Gateway - 192.168.0.113 - Trial version. Click here to purchase a license.

**Services**
Add server functionality to your system, to provide services for your clients

**Authentication providers**
You may setup various athentication methods to authenticate your users

**Submit references**
Submit reference IDs are used for delivery report matching

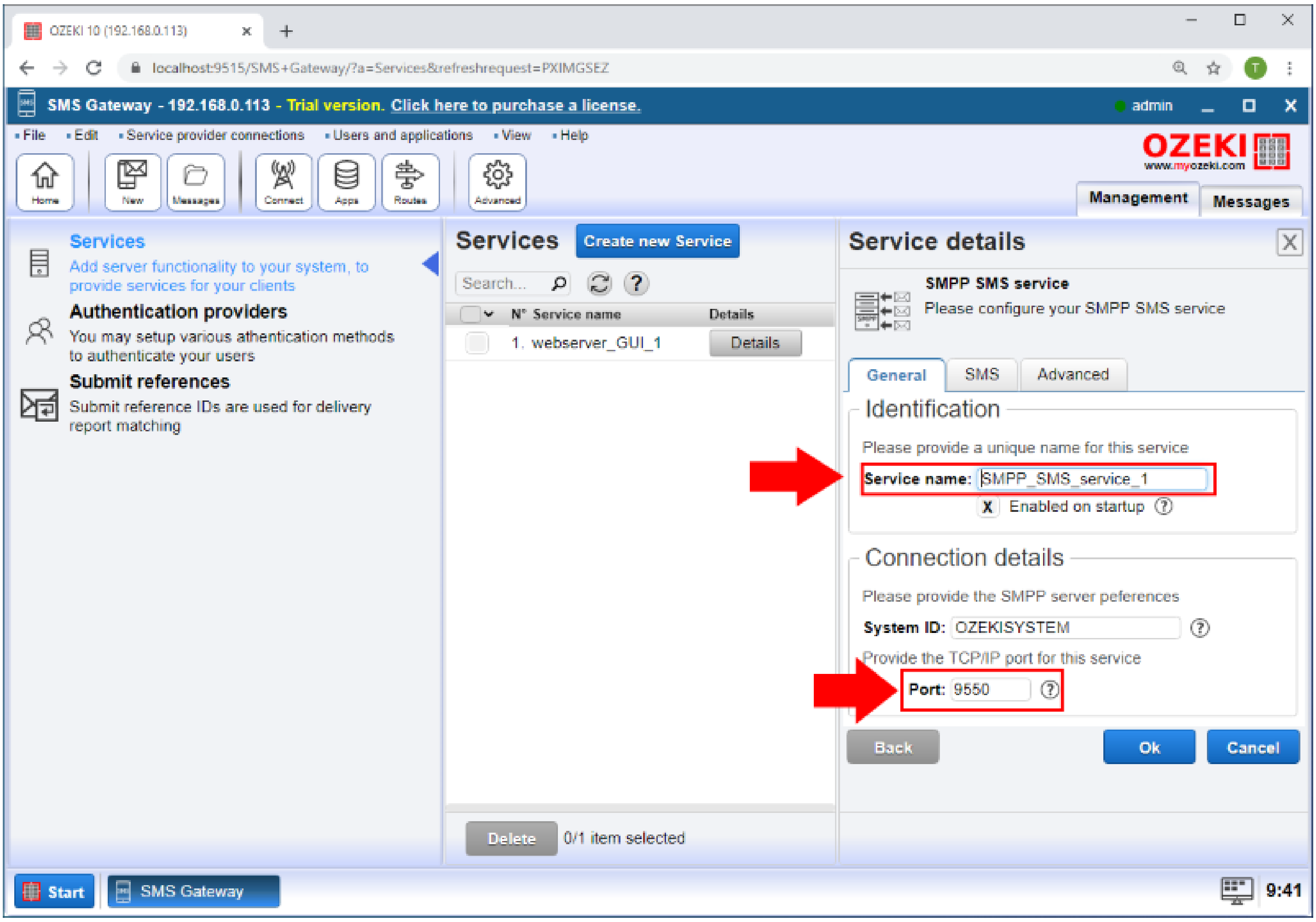
| N° | Service name | Details |
|---|---|---|
| 1. | webserver_GUI_1 | Details |

0/1 item selected

**Service details**
Please select the SMS service you wish to install.

Figure 13 - Create SMPP Service

A form will be available in the box. You should fill out the form to configure the SMPP Service. Please provide a unique name and a port for this service as you can see it in Figure 14.


OZEKI 10 (192.168.0.113)
localhost:9515/SMS+Gateway/?a=Services&refreshrequest=PXIMGSEZ
SMS Gateway - 192.168.0.113 - Trial version. Click here to purchase a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
OZEKI
www.myozeki.com
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Services
Add server functionality to your system, to provide services for your clients
Authentication providers
You may setup various athentication methods to authenticate your users
Submit references
Submit reference IDs are used for delivery report matching
Services
Create new Service
Search...
N° Service name
Details
1. webserver_GUI_1
Details
Delete
0/1 item selected
Service details
SMPP SMS service
Please configure your SMPP SMS service
General
SMS
Advanced
Back
Ok
Cancel
Start
SMS Gateway
9:41

Identification

Please provide a unique name for this service

Service name: SMPP_SMS_service_1

☒ Enabled on startup

Connection details

Please provide the SMPP server peferences

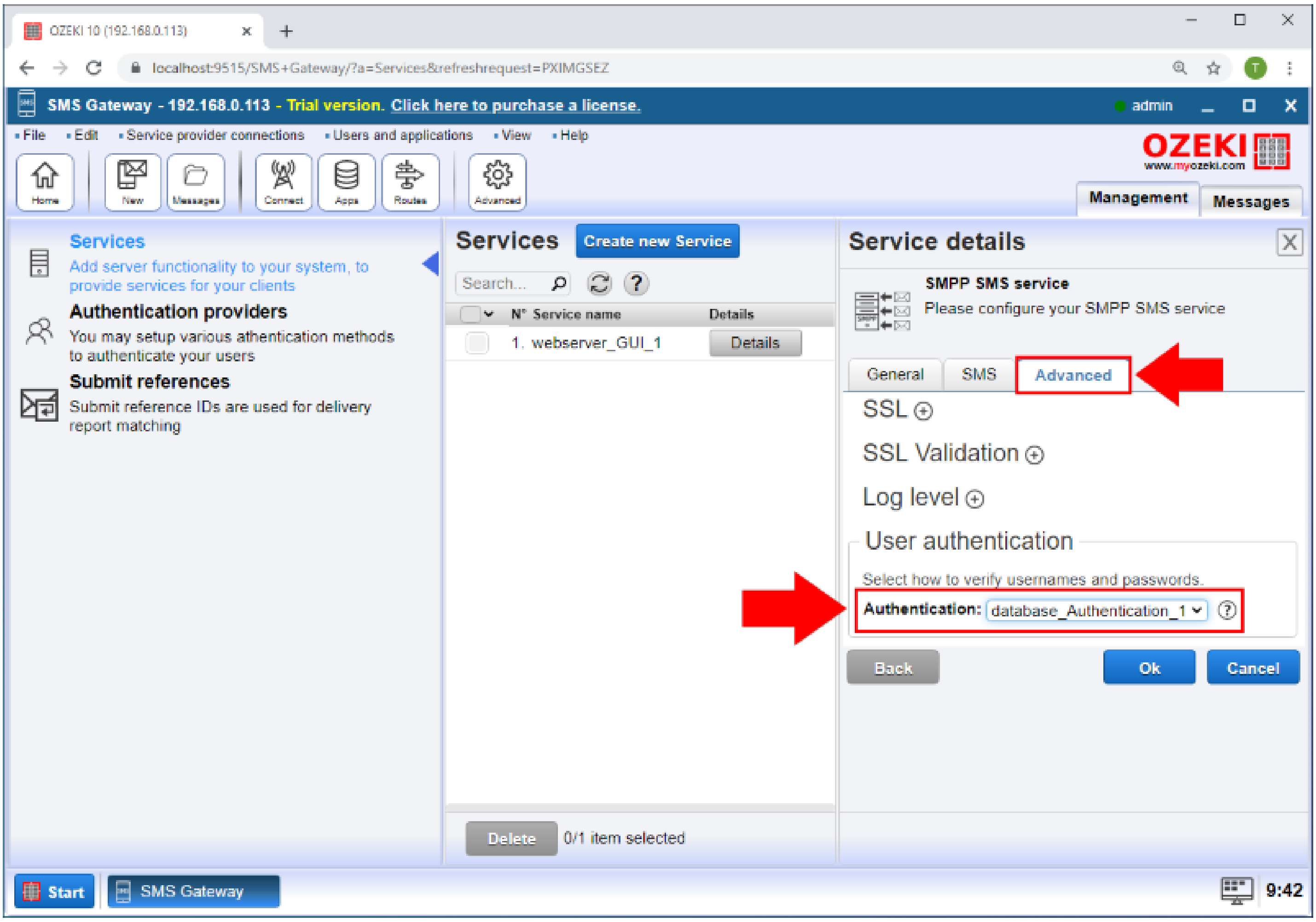
System ID: OZEKISYSTEM

Provide the TCP/IP port for this service

Port: 9550

Figure 14 - SMPP Service details

Then on the Advanced tab of the SMPP Service set the Database authentication provider in the User Authentication section as the Figure 15 shows.



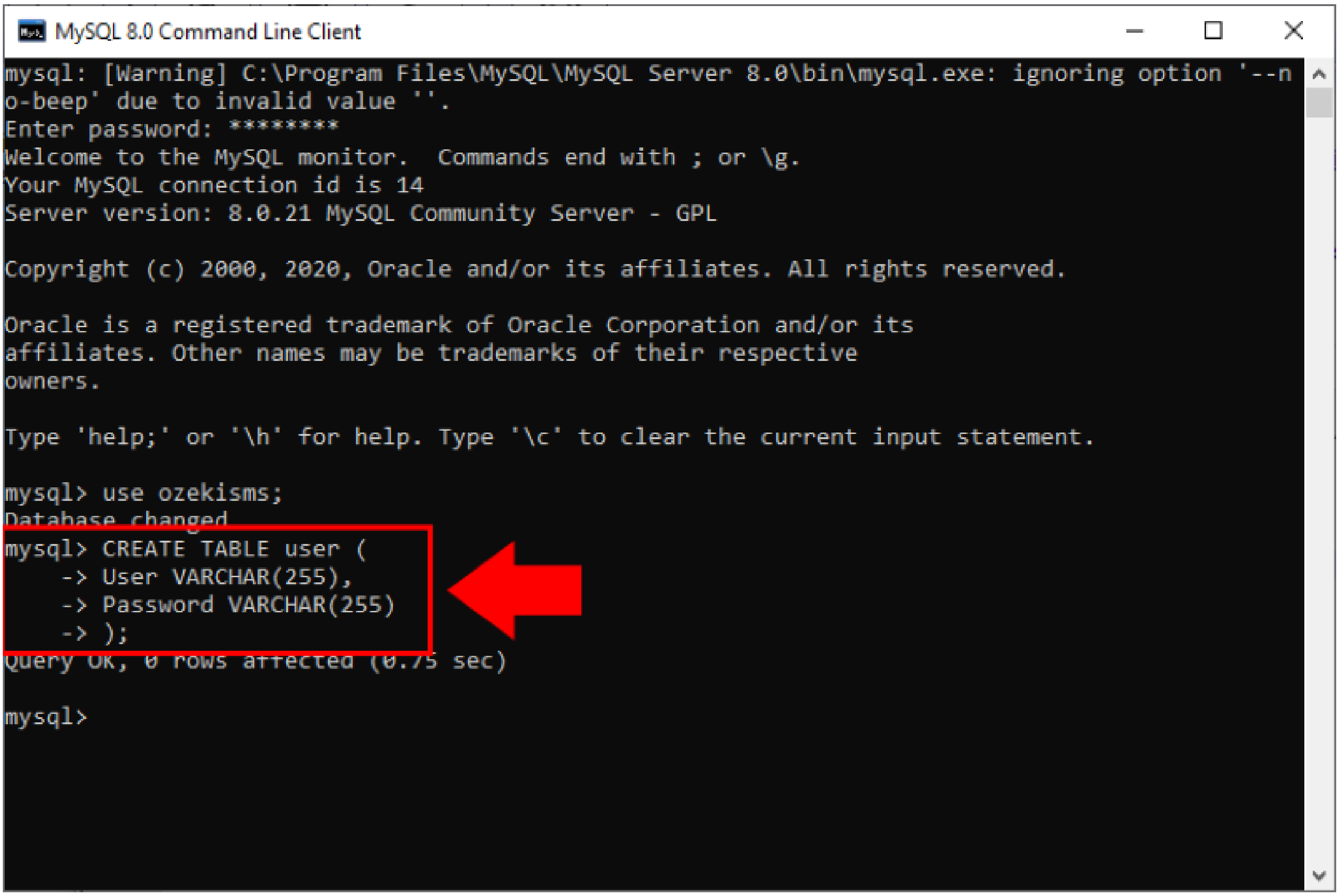
OZEKI 10 (192.168.0.113)
localhost:9515/SMS+Gateway/?a=Services&refreshrequest=PXIMGSEZ
SMS Gateway - 192.168.0.113 - Trial version. Click here to purchase a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Services
Add server functionality to your system, to provide services for your clients
Authentication providers
You may setup various athentication methods to authenticate your users
Submit references
Submit reference IDs are used for delivery report matching
Services
Create new Service
Search...
N° Service name
Details
1. webserver_GUI_1
Details
Delete
0/1 item selected
Service details
SMPP SMS service
Please configure your SMPP SMS service
General
SMS
Advanced
SSL
SSL Validation
Log level
User authentication
Select how to verify usernames and passwords.
Authentication: database_Authentication_1
Back
Ok
Cancel
Start
SMS Gateway
9:42

Figure 15 - Set Authentication provider

# Step 4 - Create SQL database table for users

To use the Database Authentication Provider you will need to have at least one SQL table that contains login information of the users. The table must have at least 2 columns. 1 column is for the username and the other one is for the password hash of the user's password. Now you will need to create a table in the database:

```
CREATE TABLE user (
    User VARCHAR(255),
    Password VARCHAR(255)
    );
```

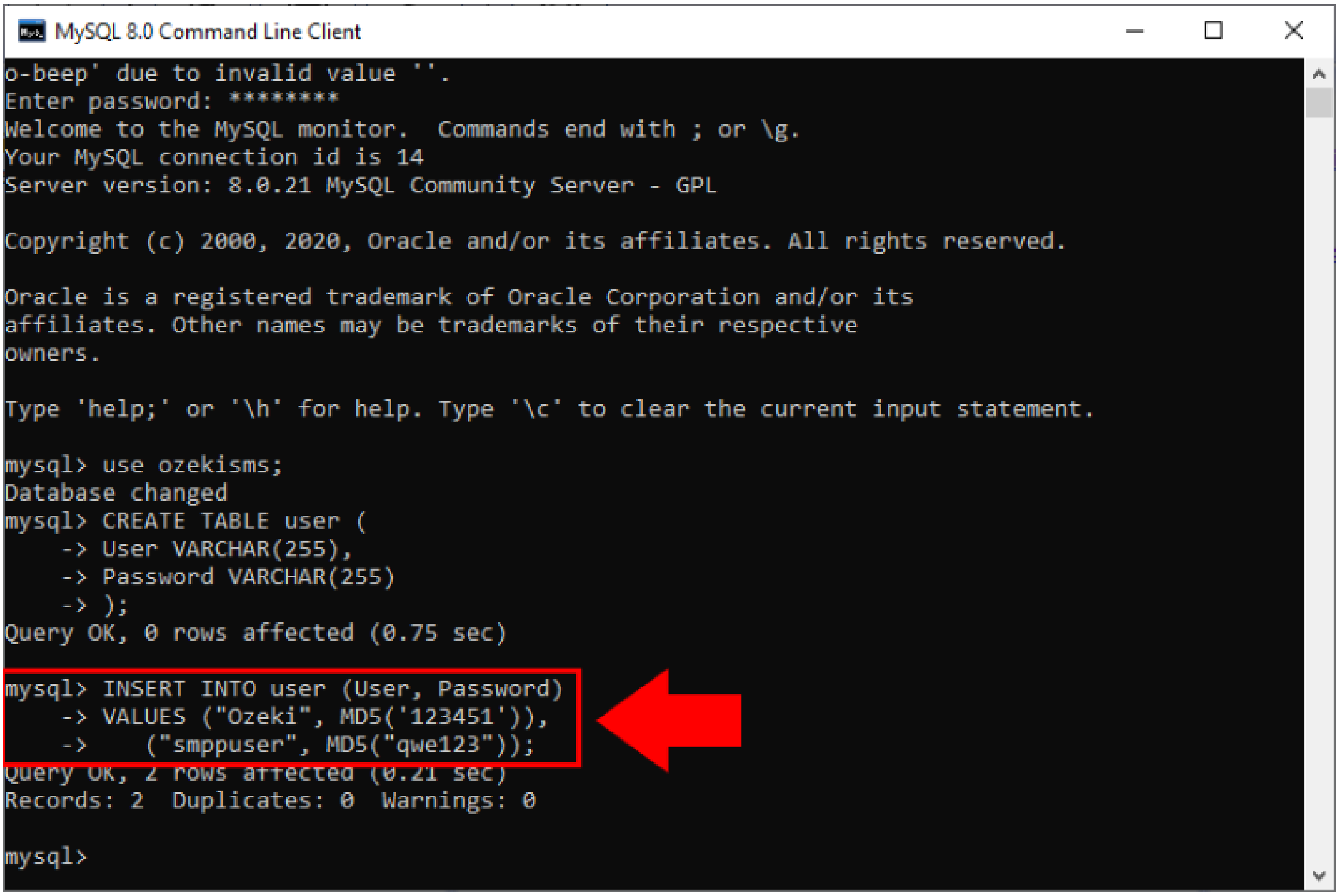


```
mysql: [Warning] C:\Program Files\MySQL\MySQL Server 8.0\bin\mysql.exe: ignoring option '--n
o-beep' due to invalid value ''.
Enter password: ********
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 14
Server version: 8.0.21 MySQL Community Server - GPL

Copyright (c) 2000, 2020, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> use ozekisms;
Database changed
mysql> CREATE TABLE user (
    -> User VARCHAR(255),
    -> Password VARCHAR(255)
    -> );
Query OK, 0 rows affected (0.75 sec)

mysql>
```

Figure 16 - Create User tabel in database

Add user credentials to your table. You can use this table later on for user authentication. Ozeki SMS Gateway will search users in this table. Do not forget to hash the passwords as you can see below:

```
INSERT INTO user (User, Password)
VALUES ("Ozeki", MD5('123451')),
       ("smppuser", MD5("qwe123"));
```

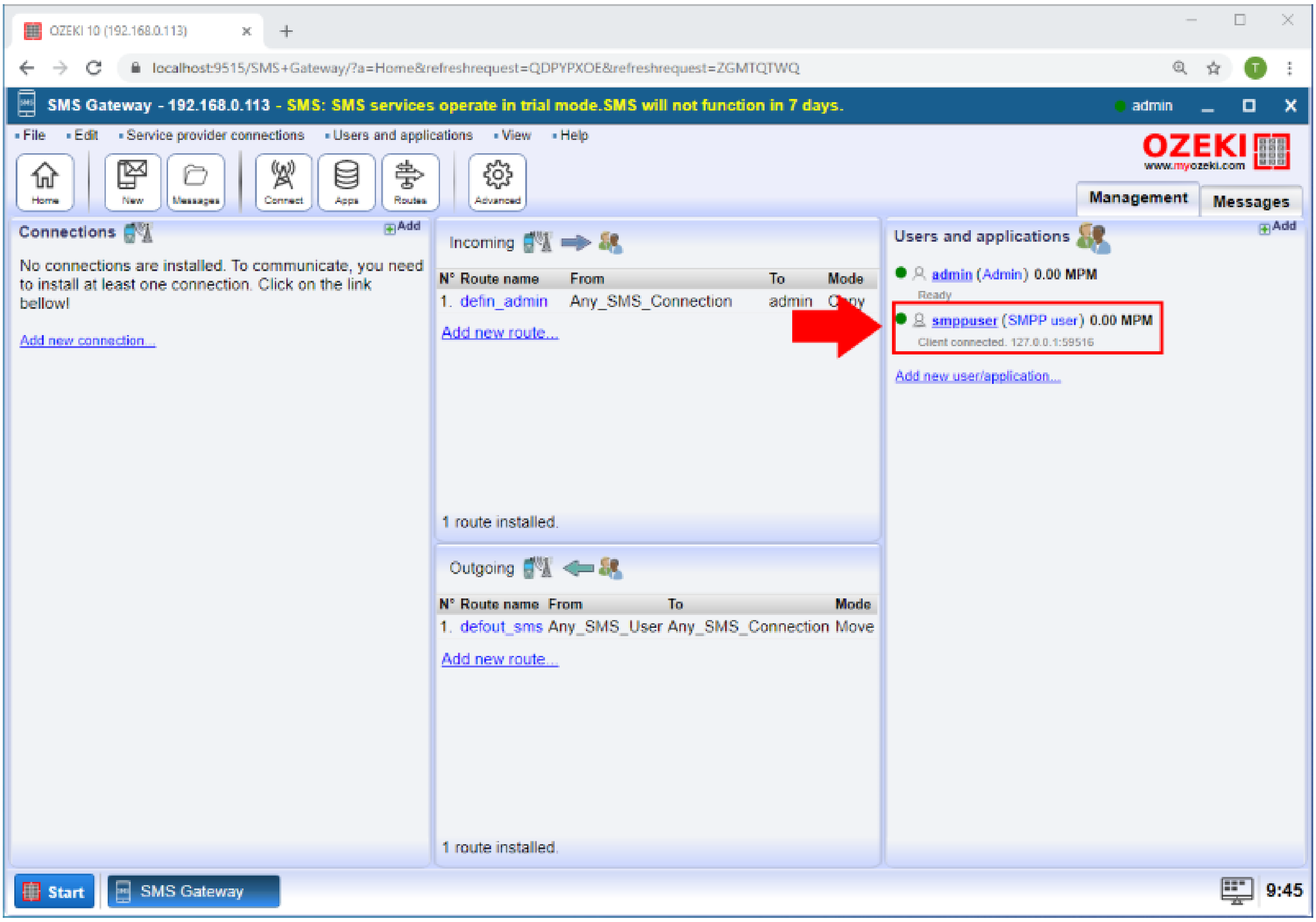


MySQL 8.0 Command Line Client

```
o-beep' due to invalid value ''.
Enter password: ********
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 14
Server version: 8.0.21 MySQL Community Server - GPL

Copyright (c) 2000, 2020, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> use ozekisms;
Database changed
mysql> CREATE TABLE user (
    -> User VARCHAR(255),
    -> Password VARCHAR(255)
    -> );
Query OK, 0 rows affected (0.75 sec)

mysql> INSERT INTO user (User, Password)
    -> VALUES ("Ozeki", MD5('123451')),
    ->    ("smppuser", MD5("qwe123"));
Query OK, 2 rows affected (0.21 sec)
Records: 2  Duplicates: 0  Warnings: 0

mysql>
```

Figure 17 - Create users in database

Finally if an SMPP Client is connecting with the username and password you created in the database, you will see that the User is appear in the SMS Gateway Users and applications section (Figure 18).



OZEKI 10 (192.168.0.113)
localhost:9515/SMS+Gateway/?a=Home&refreshrequest=QDPYPXOE&refreshrequest=ZGMTQTWQ
SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Start
SMS Gateway
9:45

Connections

Add

No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

Incoming

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defin_admin | Any_SMS_Connection | admin | Copy |

Add new route...

1 route installed.

Outgoing

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move |

Add new route...

1 route installed.

Users and applications

Add

admin (Admin) 0.00 MPM
Ready

smppuser (SMPP user) 0.00 MPM
Client connected. 127.0.0.1:59516

Add new user/application...

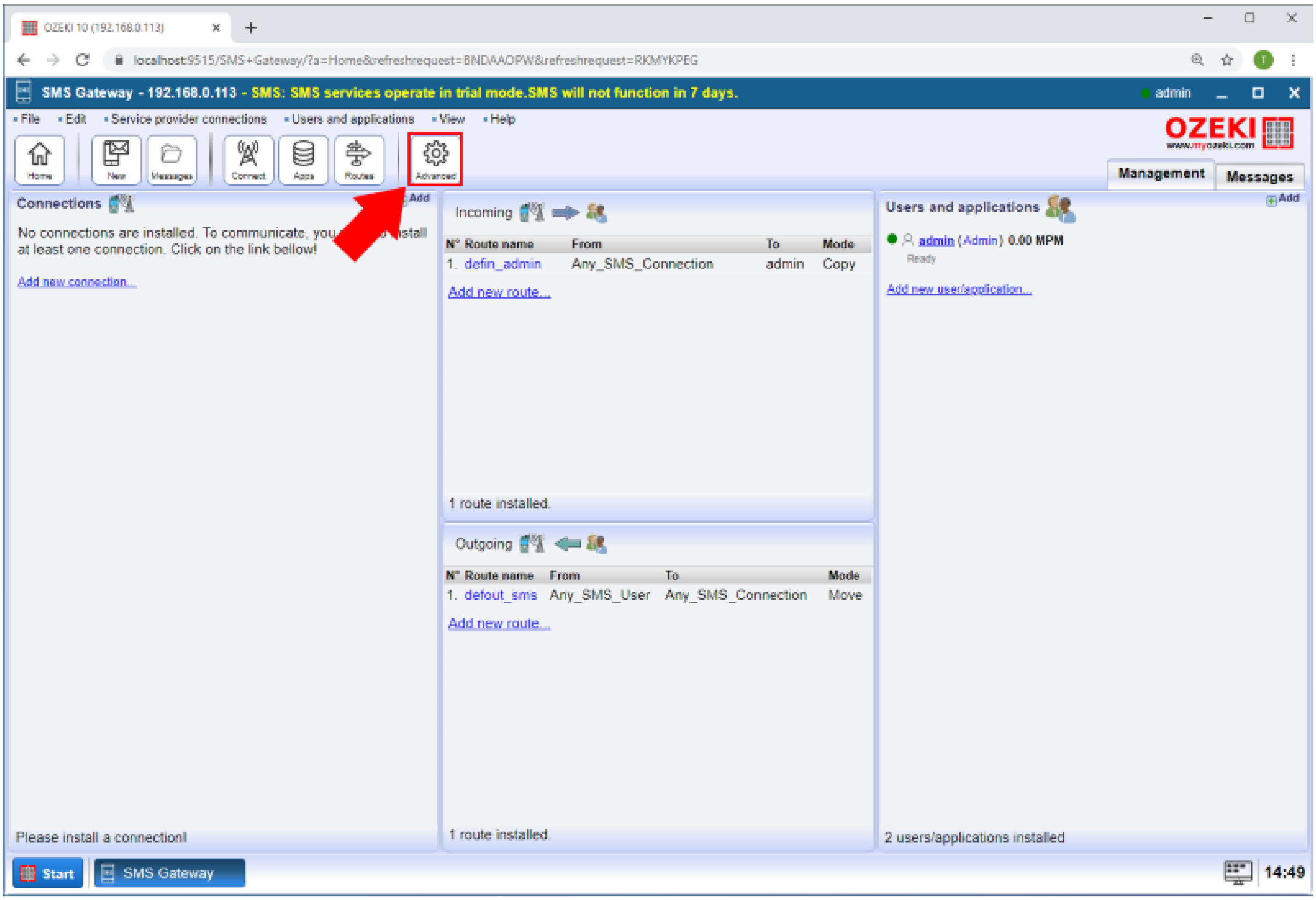
Figure 18 - SMPP user connected

# How to authenticate SMPP users with LDAP

*This document is going to demonstrate how you can connect to Microsoft Active Directory using the LDAP (Lightweight Directory Access Protocol) protocol and authenticate the users or the employees from the user database of Microsoft Domain. The guide contains a step by step that shows you how to set up and connect to the Active Directory, how to set up the authentication in SMS Gateway and lastly, how to establish the authentication in your SMPP service.*

## Step 1 - Create new HTTP Authentication Provide

First step is to open the SMS Gateway application. So, just navigate to the desktop of, and here just open SMS Gateway by clicking on its icon. Here syu can find the Advanced menu of the SMS Gateway. To open it please click the "Advanced" button on the main page.



SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.

No connections are installed. To communicate, you ... install at least one connection. Click on the link bellow!

Add new connection...

Please install a connection!

Incoming

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defin_admin | Any_SMS_Connection | admin | Copy |

Add new route...

1 route installed.

Outgoing

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move |

Add new route...

1 route installed.

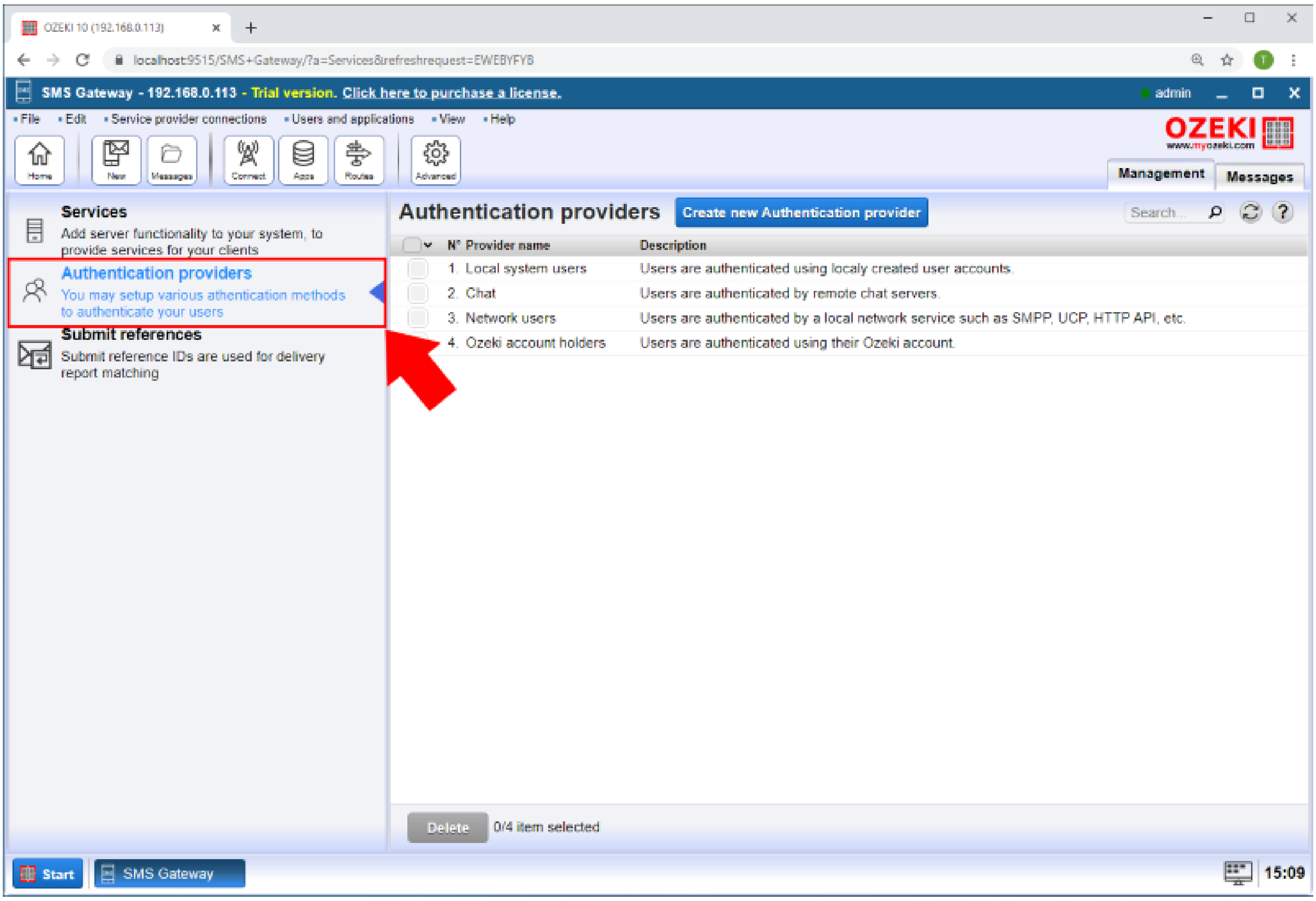
admin (Admin) 0.00 MPM
Ready

Add new user/application...

2 users/applications installed

Figure 1 - Open SMS Gateway Advanced menu

You will find yourself on the list of authentication providers. On the Authentication provider page you can create, modify or delete authentication providers. Create a new authentication provider by clicking the blue Create new authentication provider button. A new box will open on the right side of the page. In this box you can select between authentication providers. Please choose LDAP authentication provider as the Figure 2 and Figure 3 shows.



SMS Gateway - 192.168.0.113 - Trial version. Click here to purchase a license.

**Services**
Add server functionality to your system, to provide services for your clients

**Authentication providers**
You may setup various athentication methods to authenticate your users

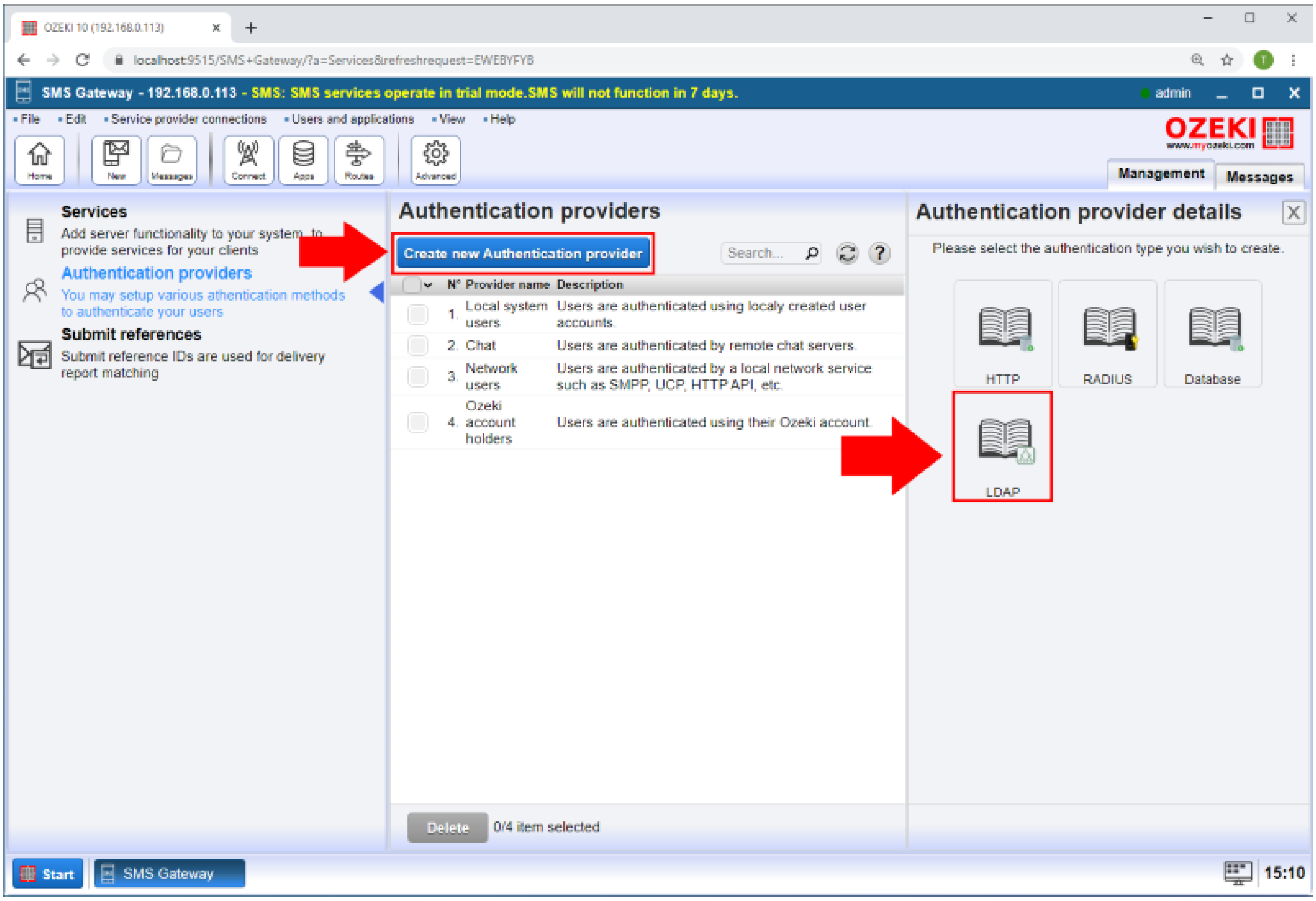
**Submit references**
Submit reference IDs are used for delivery report matching

## Authentication providers

| N° | Provider name | Description |
|---|---|---|
| 1. | Local system users | Users are authenticated using localy created user accounts. |
| 2. | Chat | Users are authenticated by remote chat servers. |
| 3. | Network users | Users are authenticated by a local network service such as SMPP, UCP, HTTP API, etc. |
| 4. | Ozeki account holders | Users are authenticated using their Ozeki account. |

0/4 item selected

Figure 2 - Authentication providers



SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.

**Services**
Add server functionality to your system, to provide services for your clients

**Authentication providers**
You may setup various athentication methods to authenticate your users

**Submit references**
Submit reference IDs are used for delivery report matching

## Authentication providers

| N° | Provider name | Description |
|---|---|---|
| 1. | Local system users | Users are authenticated using localy created user accounts |
| 2. | Chat | Users are authenticated by remote chat servers. |
| 3. | Network users | Users are authenticated by a local network service such as SMPP, UCP, HTTP API, etc. |
| 4. | Ozeki account holders | Users are authenticated using their Ozeki account. |

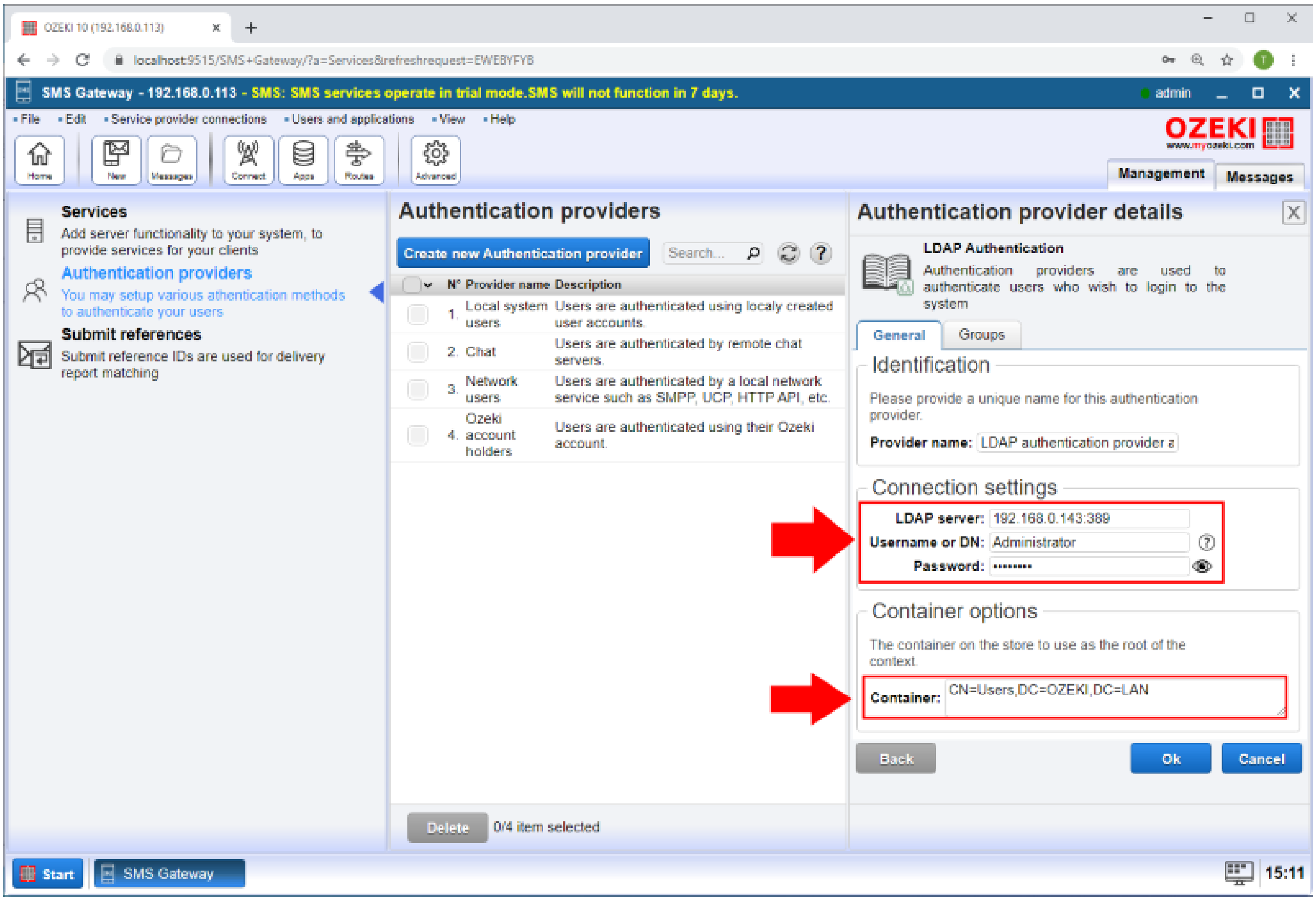
0/4 item selected

## Authentication provider details

Please select the authentication type you wish to create.

Figure 3 - LDAP provider

Please fill the LDAP Authentication Provider form (Figure 4) depending on your LDAP server IP and username/password pair. You also need to provide the correct Search DN. The LDAP Authentication provider will issue the search requests in this DN. E.g. "CN=Users,DC=OZEKI,DC=LAN" where LAN is the root node, OZEKI is an organization under the root node, and Users is an organizational unit of OZEKI organization.


OZEKI 10 (192.168.0.113)
localhost:9515/SMS+Gateway/?a=Services&refreshrequest=EWEBYFYB
SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
OZEKI
www.myozeki.com
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Start
SMS Gateway
15:11

**Services**

Add server functionality to your system, to provide services for your clients

**Authentication providers**

You may setup various athentication methods to authenticate your users

**Submit references**

Submit reference IDs are used for delivery report matching

**Authentication providers**

Create new Authentication provider | Search...

| N° | Provider name | Description |
|---|---|---|
| 1. | Local system users | Users are authenticated using localy created user accounts. |
| 2. | Chat | Users are authenticated by remote chat servers. |
| 3. | Network users | Users are authenticated by a local network service such as SMPP, UCP, HTTP API, etc. |
| 4. | Ozeki account holders | Users are authenticated using their Ozeki account. |

Delete | 0/4 item selected

**Authentication provider details**

**LDAP Authentication**

Authentication providers are used to authenticate users who wish to login to the system

General | Groups

**Identification**

Please provide a unique name for this authentication provider.

Provider name: LDAP authentication provider a

**Connection settings**

LDAP server: 192.168.0.143:389

Username or DN: Administrator

Password: ••••••••

**Container options**

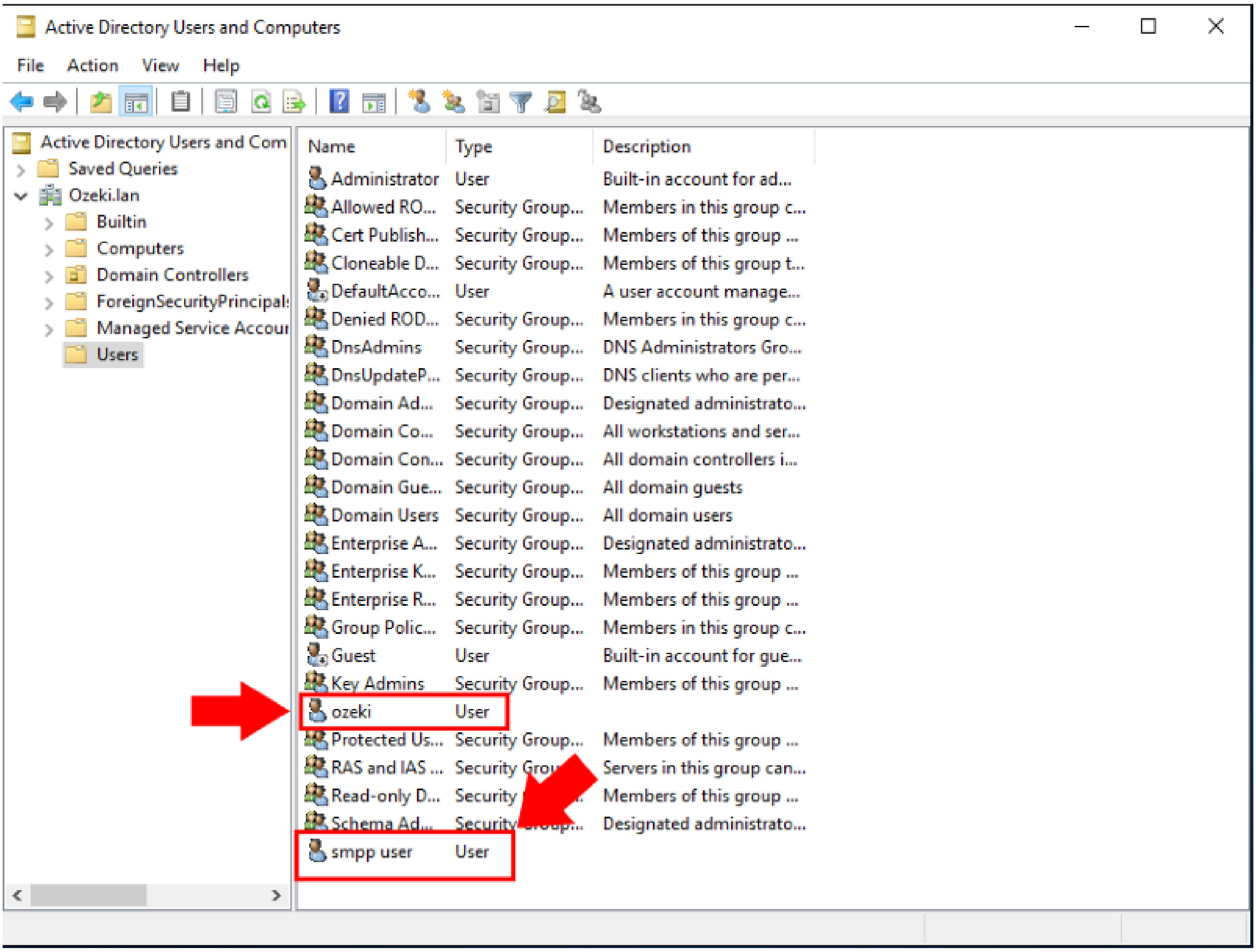
The container on the store to use as the root of the context.

Container: CN=Users,DC=OZEKI,DC=LAN

Back | Ok | Cancel

Figure 4 - Authentication provider details

If you wish to use the Active directory of Windows Server, you can find the Users under the Active Directory Users and Computers menu. Here you can create users to log in to the Ozeki SMS Gateway later.



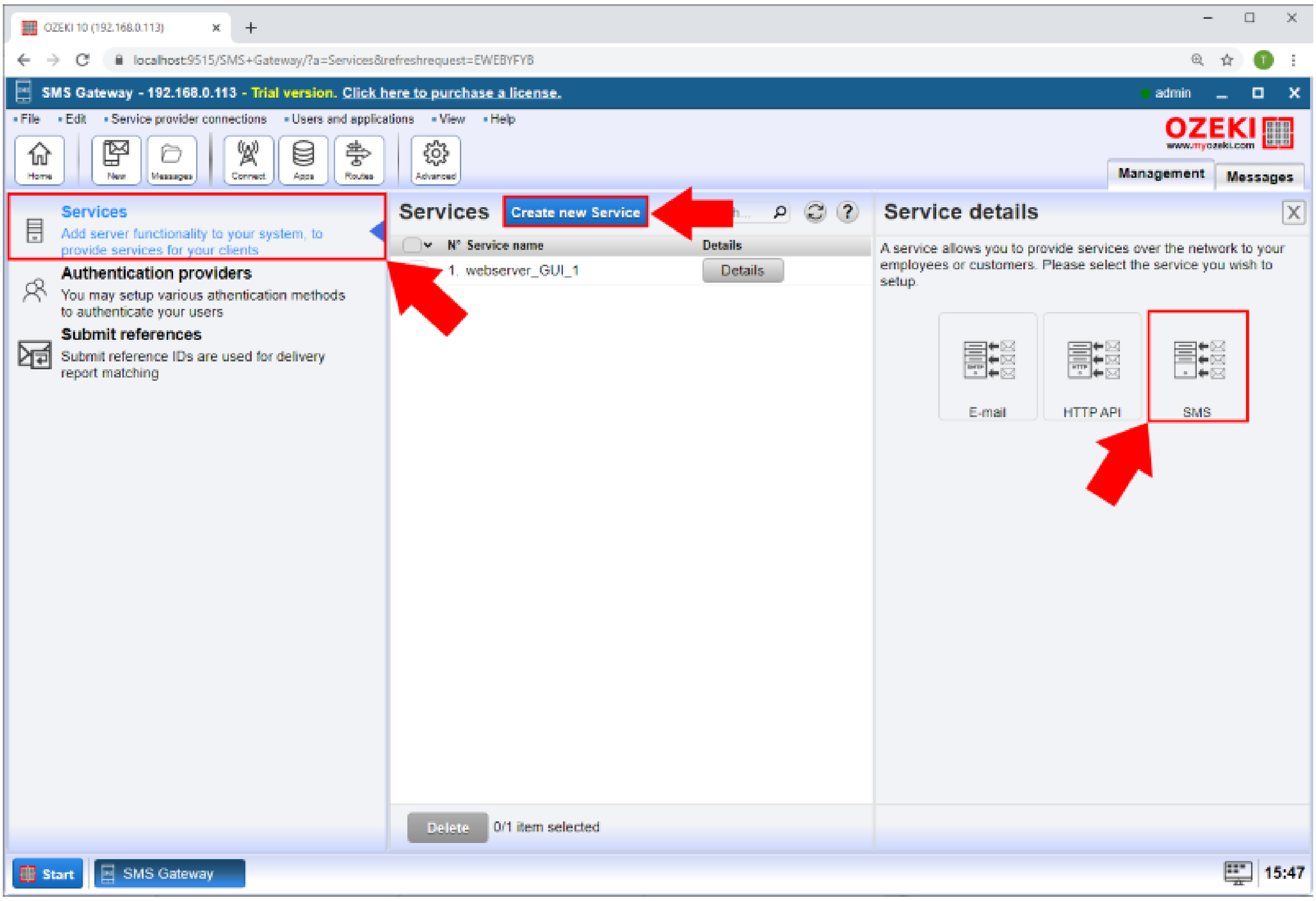
Active Directory Users and Com
Saved Queries
Ozeki.lan
Builtin
Computers
Domain Controllers
ForeignSecurityPrincipal:
Managed Service Accour
Users

| Name | Type | Description |
|---|---|---|
| Administrator | User | Built-in account for ad... |
| Allowed RO... | Security Group... | Members in this group c... |
| Cert Publish... | Security Group... | Members of this group ... |
| Cloneable D... | Security Group... | Members of this group t... |
| DefaultAcco... | User | A user account manage... |
| Denied ROD... | Security Group... | Members in this group c... |
| DnsAdmins | Security Group... | DNS Administrators Gro... |
| DnsUpdateP... | Security Group... | DNS clients who are per... |
| Domain Ad... | Security Group... | Designated administrato... |
| Domain Co... | Security Group... | All workstations and ser... |
| Domain Con... | Security Group... | All domain controllers i... |
| Domain Gue... | Security Group... | All domain guests |
| Domain Users | Security Group... | All domain users |
| Enterprise A... | Security Group... | Designated administrato... |
| Enterprise K... | Security Group... | Members of this group ... |
| Enterprise R... | Security Group... | Members of this group ... |
| Group Polic... | Security Group... | Members in this group c... |
| Guest | User | Built-in account for gue... |
| Key Admins | Security Group... | Members of this group ... |
| ozeki | User | |
| Protected Us... | Security Group... | Members of this group ... |
| RAS and IAS ... | Security Grou | Servers in this group can... |
| Read-only D... | Security | Members of this group ... |
| Schema Ad... | Security ...roup... | Designated administrato... |
| smpp user | User | |

Figure 5 - Active Directory users

## Step 2 - Create SMPP Service with LDAP Authentication

On the Advanced page Services section you can create, modify or delete SMS Services. Create a new SMS Services by clicking the blue Create new Service button. A new box will open on the right side of the page. In this box you can select between SMS Services as you can see in the Figure 6. And then please choose SMPP Service (Figure 7).



**Services**

Add server functionality to your system, to provide services for your clients

**Authentication providers**

You may setup various athentication methods to authenticate your users

**Submit references**

Submit reference IDs are used for delivery report matching

**Services**

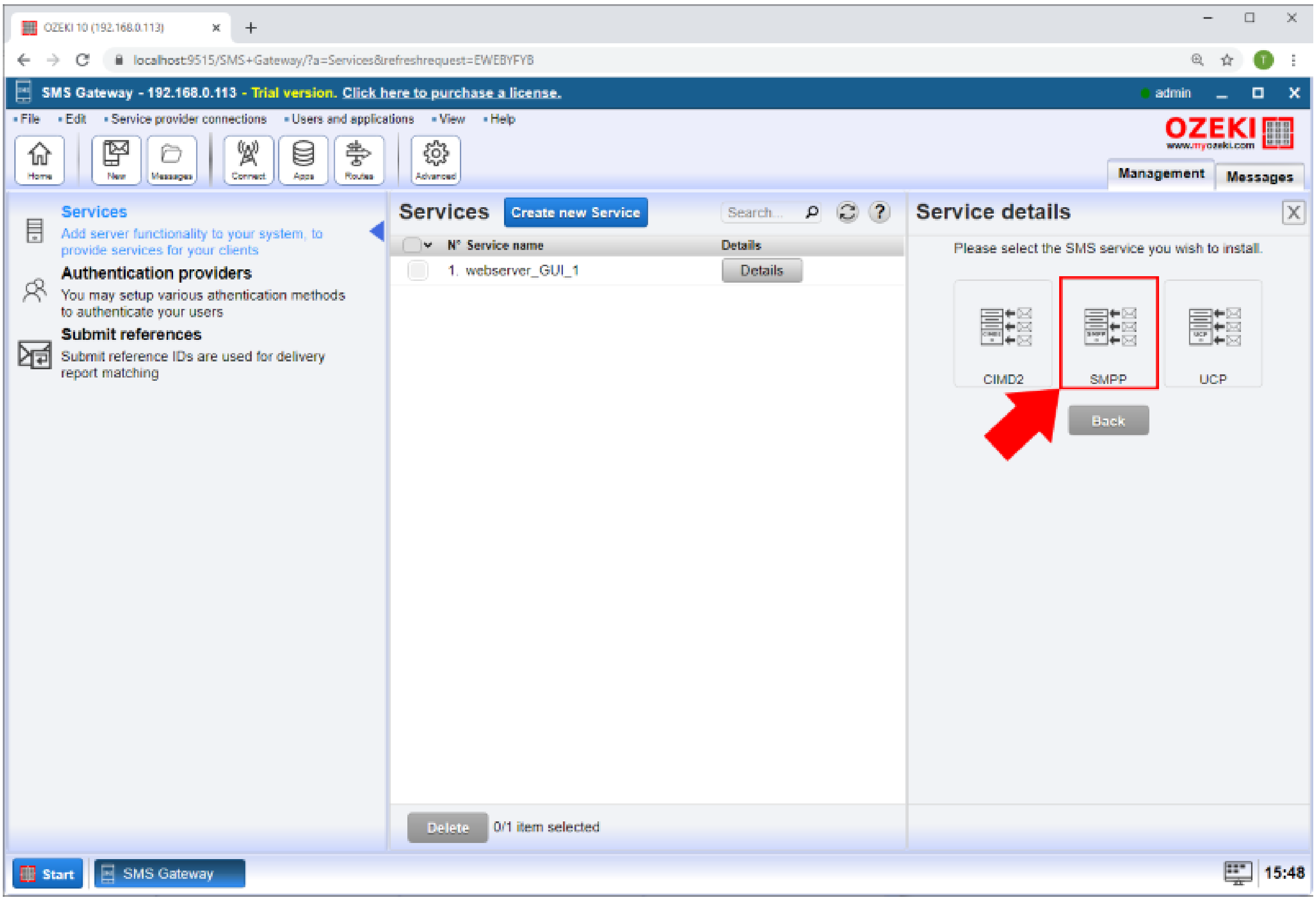
1. webserver_GUI_1 Details

0/1 item selected

**Service details**

A service allows you to provide services over the network to your employees or customers. Please select the service you wish to setup.

Figure 6 - New SMS service



**Services**

Add server functionality to your system, to provide services for your clients

**Authentication providers**

You may setup various athentication methods to authenticate your users

**Submit references**

Submit reference IDs are used for delivery report matching

**Services**

1. webserver_GUI_1 Details

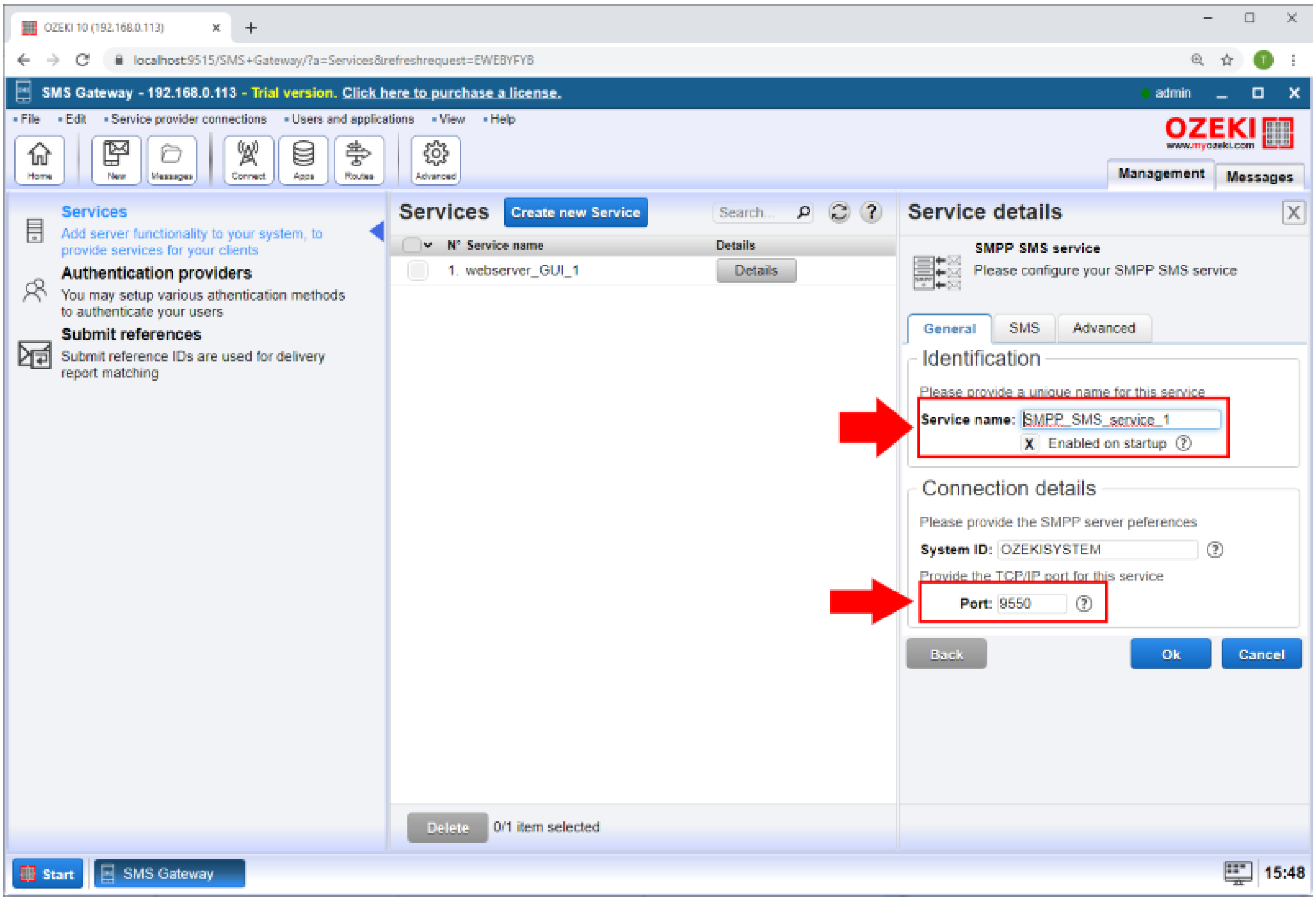
0/1 item selected

**Service details**

Please select the SMS service you wish to install.

Figure 7 - New SMPP service

A form will be available in the box. You should fill out the form to configure the SMPP Service. Please provide a unique name and a port for this service as you can see it in Figure 8.



Service details

SMPP SMS service

Please configure your SMPP SMS service

Identification

Please provide a unique name for this service

Service name: SMPP_SMS_service_1

☒ Enabled on startup

Connection details

Please provide the SMPP server peferences

System ID: OZEKISYSTEM

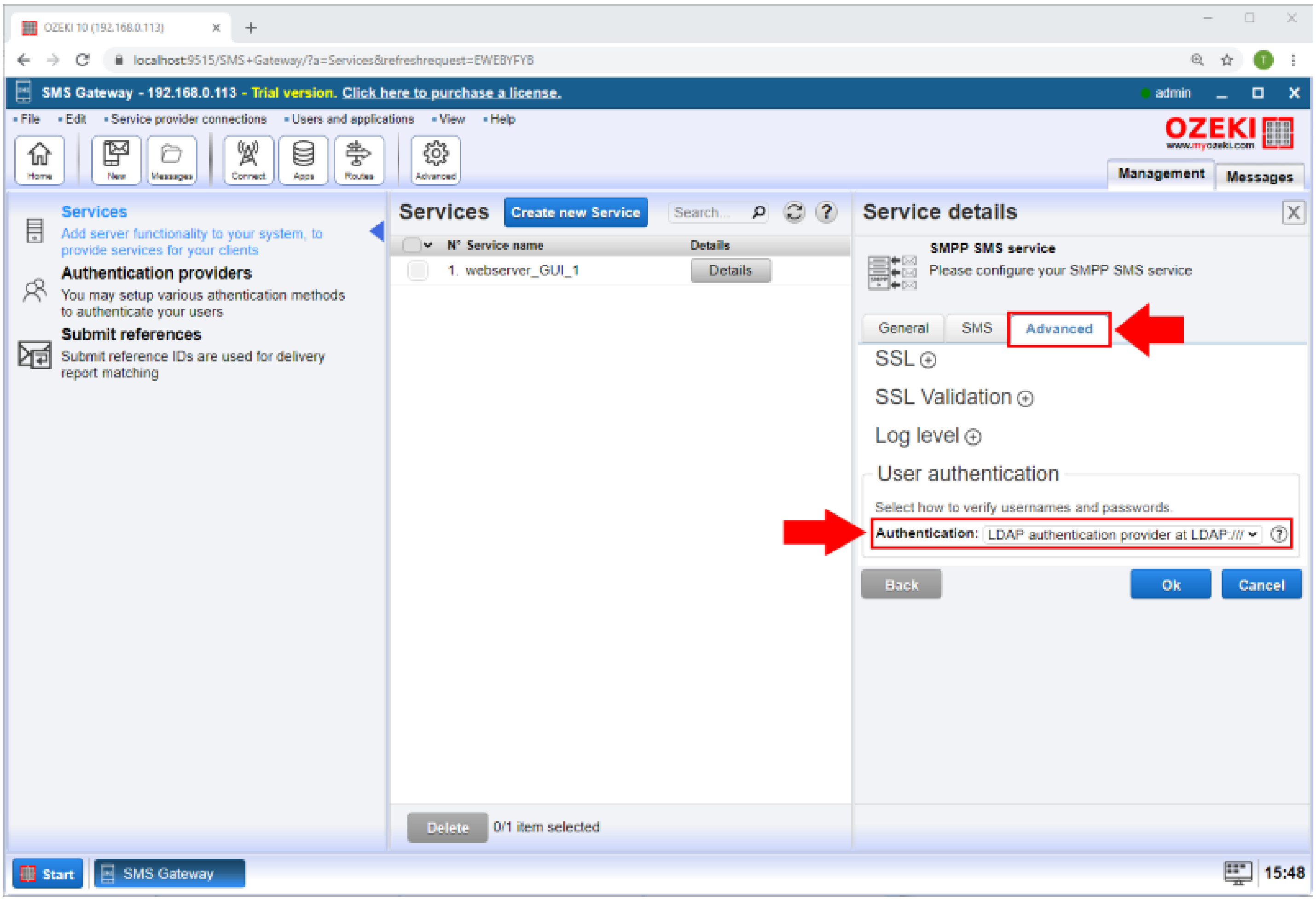
Provide the TCP/IP port for this service

Port: 9550

OZEKI 10 (192.168.0.113)
localhost:9515/SMS+Gateway/?a=Services&refreshrequest=EWEBYFYB
SMS Gateway - 192.168.0.113 - Trial version. Click here to purchase a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Services
Add server functionality to your system, to provide services for your clients
Authentication providers
You may setup various athentication methods to authenticate your users
Submit references
Submit reference IDs are used for delivery report matching
Services
Create new Service
Search...
N° Service name
Details
1. webserver_GUI_1
Details
General
SMS
Advanced
Back
Ok
Cancel
Delete
0/1 item selected
Start
SMS Gateway
15:48

Figure 8 - SMPP service details

Then on the Advanced tab of the SMPP Service set the LDAP authentication provider in the User Authentication section as the Figure 9 shows.



SMS Gateway - 192.168.0.113 - Trial version. Click here to purchase a license.
File
Edit
Service provider connections
Users and applications
View
Help
Management
Messages
Services
Add server functionality to your system, to provide services for your clients
Authentication providers
You may setup various athentication methods to authenticate your users
Submit references
Submit reference IDs are used for delivery report matching
Create new Service
Delete
Back
Ok
Cancel
Start
SMS Gateway
15:48

Services

| N° Service name | Details |
|---|---|
| 1. webserver_GUI_1 | Details |

0/1 item selected

Service details

SMPP SMS service

Please configure your SMPP SMS service

General | SMS | Advanced

SSL

SSL Validation

Log level

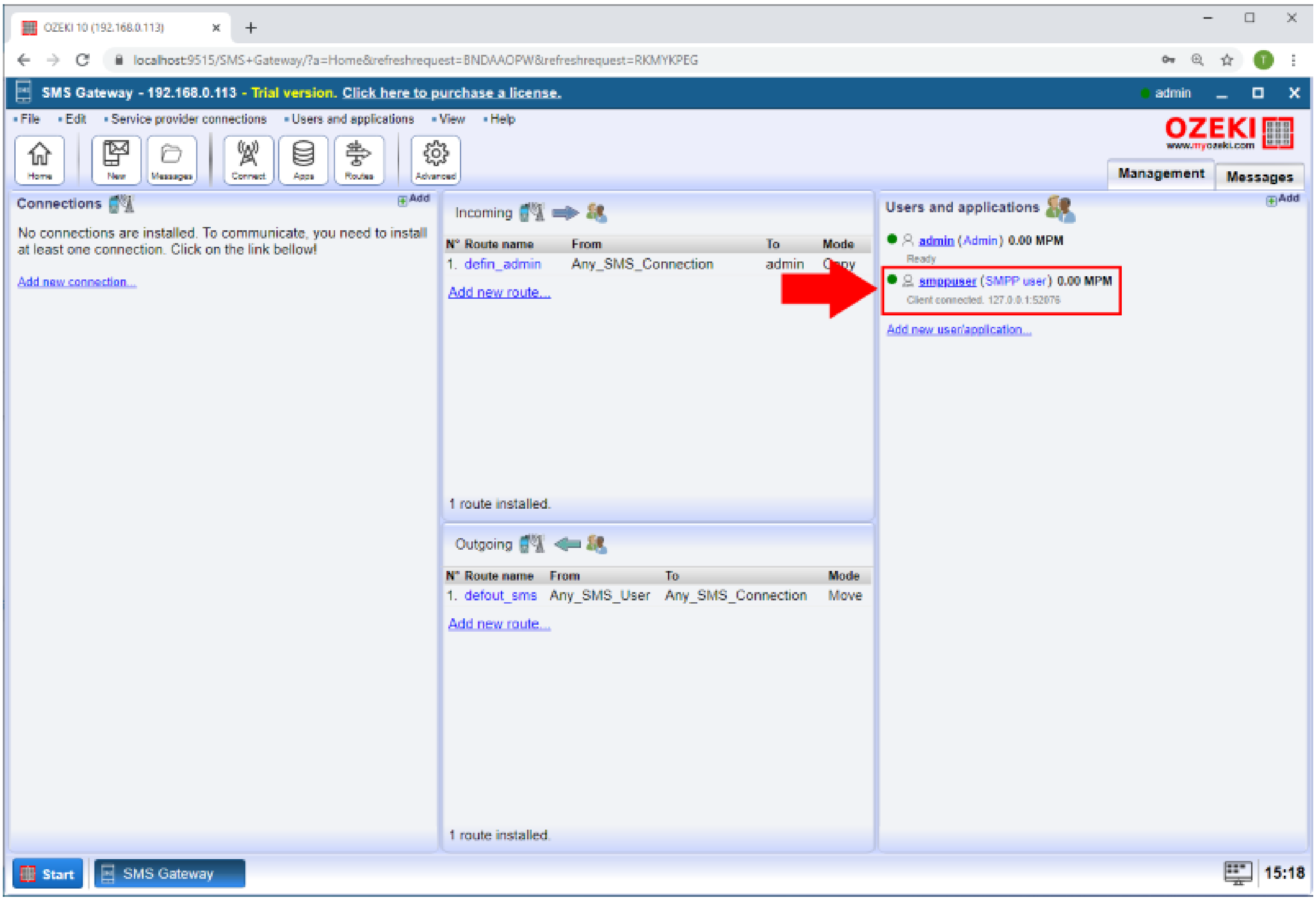
User authentication

Select how to verify usernames and passwords.

Authentication: LDAP authentication provider at LDAP:///

Figure 9 - LDAP authentication provider

Finally if an SMPP Client is connecting with the username and password you created in the database, you will see that the User is appear in the SMS Gateway Users and applications section (Figure 10).


OZEKI 10 (192.168.0.113)
localhost:9515/SMS+Gateway/?a=Home&refreshrequest=BNDAAOPW&refreshrequest=RKMYKPEG
SMS Gateway - 192.168.0.113 - Trial version. Click here to purchase a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
OZEKI
www.myozeki.com
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Add
Start
SMS Gateway
15:18

Connections

No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

Incoming

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defin_admin | Any_SMS_Connection | admin | Copy |

Add new route...

1 route installed.

Outgoing

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move |

Add new route...

1 route installed.

Users and applications

● admin (Admin) 0.00 MPM
Ready

● smppuser (SMPP user) 0.00 MPM
Client connected. 127.0.0.1:52076

Add new user/application...

Figure 10 - SMPP user connected

# How to authenticate SMPP users with HTTP

***With a HTTP Authentication Provider you can give your userbase login rights to Ozeki SMS Gateway. During the installation process you will be asked to provide details for the user's account. These details contain the username and password which you will need to use for your first login. You will also need a HTTP webserver which will approve the user credentials during the HTTP authentication. In this tutorial you will see example PHP codes for your webserver.***

## Step 1 - Create new HTTP Authentication Provider

The first step is to open the Control Panel application in Ozeki SMS Gateway. So, just navigate to the desktop of Ozeki SMS Gateway, and here, as you can see it in Figure 1, just open Control Panel by clicking on its icon.

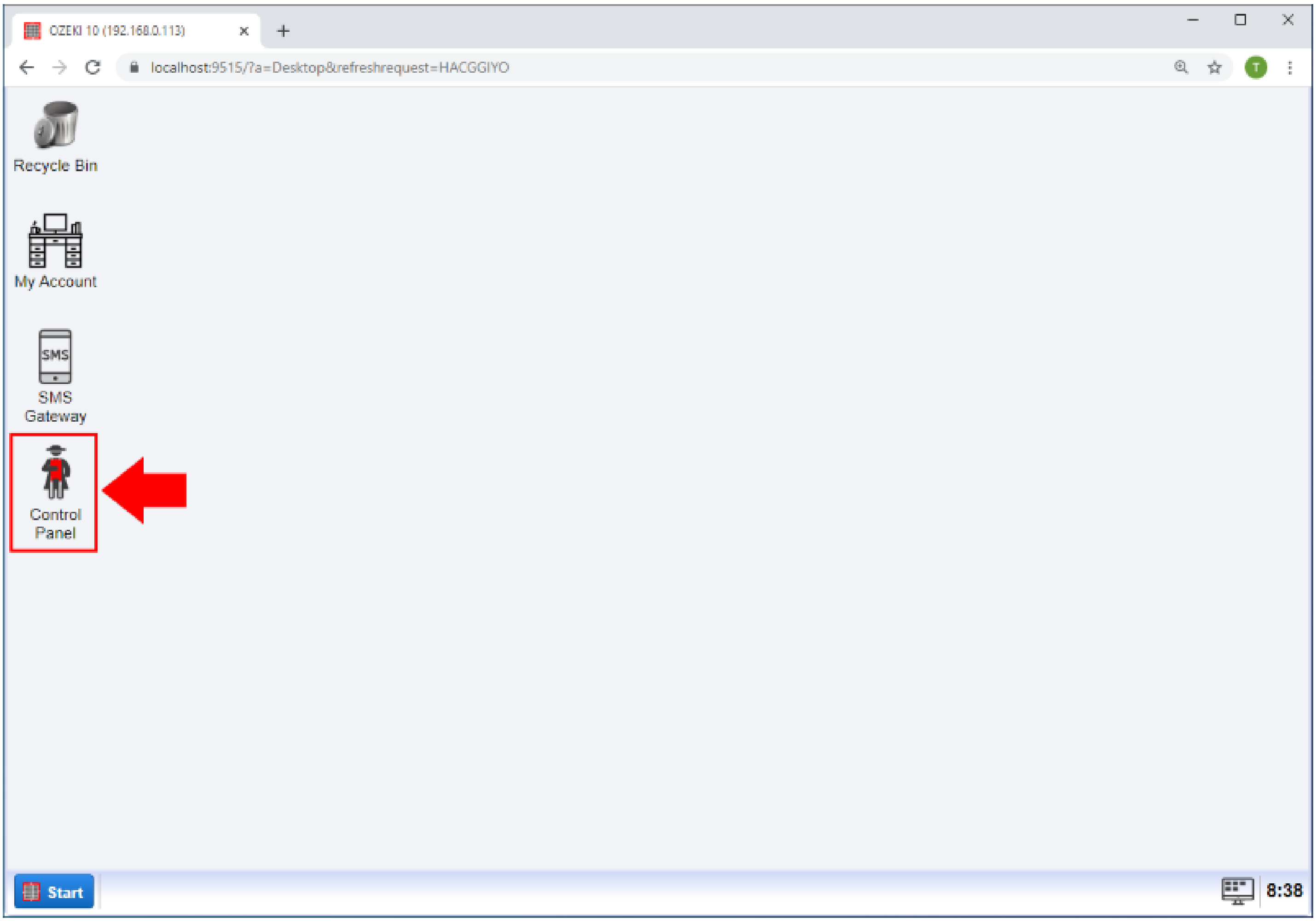


Figure 1 - Open Control Panel

Now you should be on the Control Panel main page where you can create, modify or delete connections. Click on the Users button and you will find yourself on the User accounts menu (Figure 2).

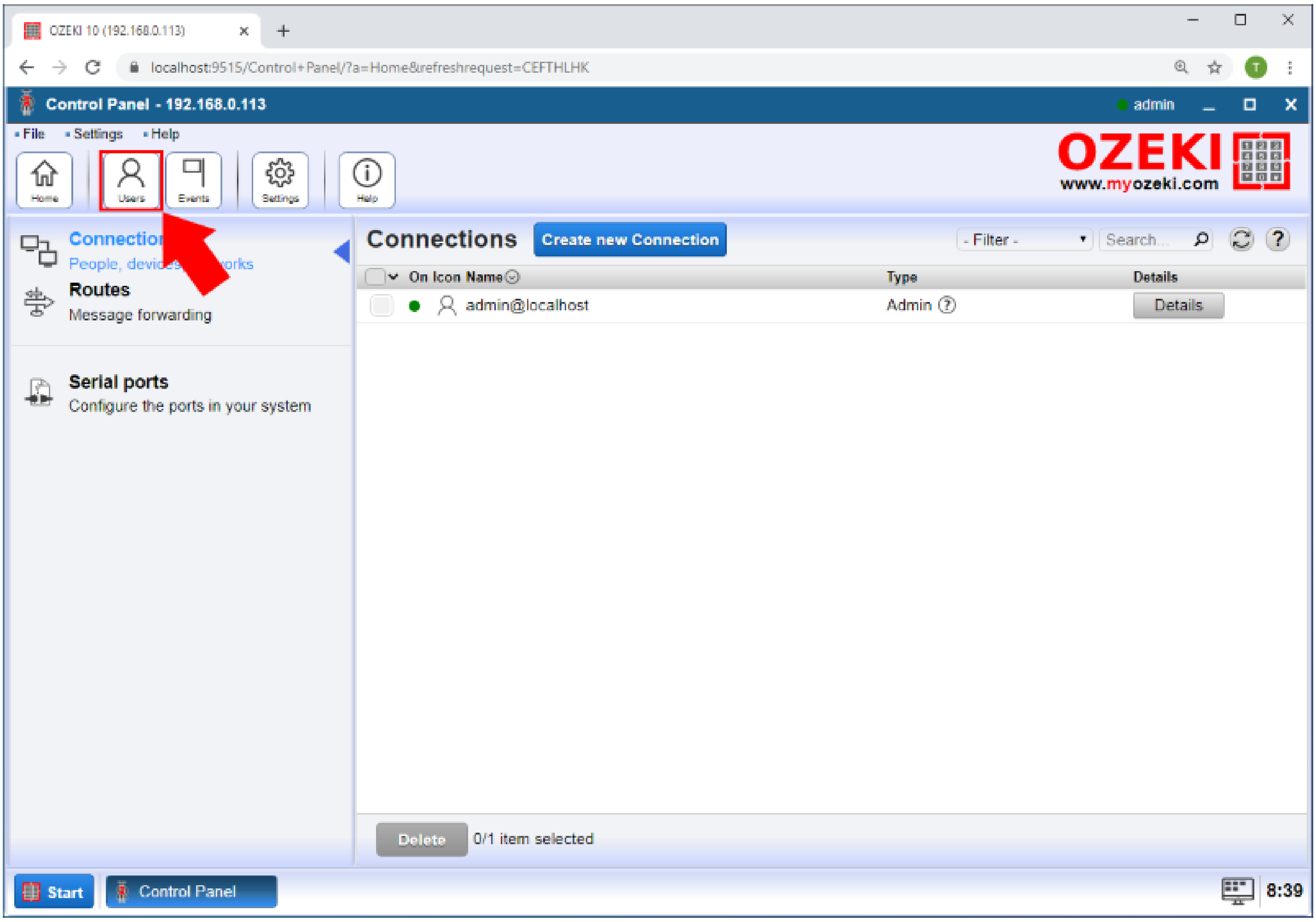
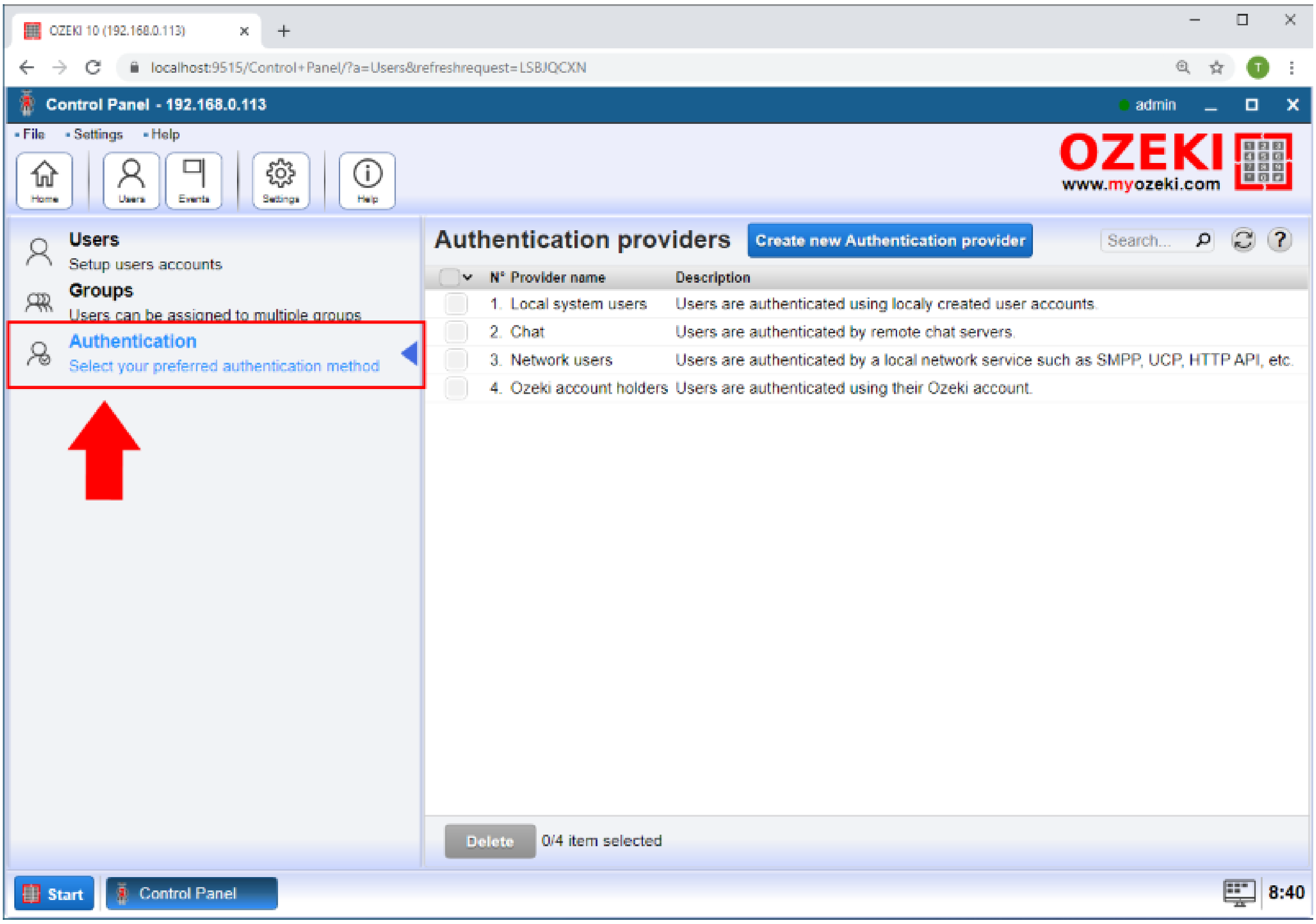


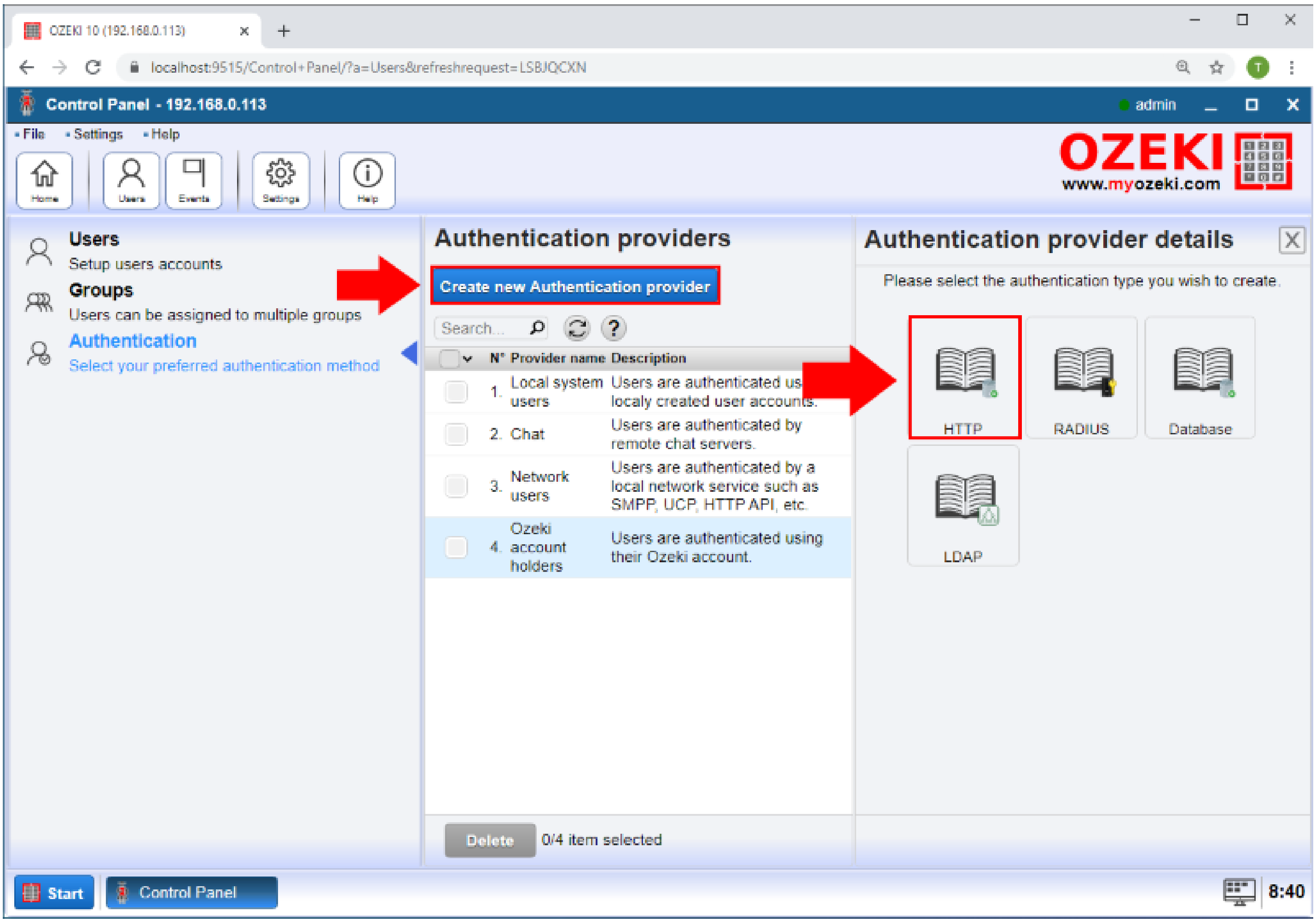
Figure 2 - Open Users menu

You will find yourself on the list of authentication providers. On the Authentication provider page you can create, modify or delete authentication providers. Create a new authentication provider by clicking the blue Create new authentication provider button. A new box will open on the right side of the page. In this box you can select between authentication providers. Please choose HTTP authentication provider as the Figure 3 and Figure 4 shows.



| N° | Provider name | Description |
|---|---|---|
| 1. | Local system users | Users are authenticated using localy created user accounts. |
| 2. | Chat | Users are authenticated by remote chat servers. |
| 3. | Network users | Users are authenticated by a local network service such as SMPP, UCP, HTTP API, etc. |
| 4. | Ozeki account holders | Users are authenticated using their Ozeki account. |

Figure 3 - Authentication providers menu


OZEKI 10 (192.168.0.113)
localhost:9515/Control+Panel/?a=Users&refreshrequest=LSBJQCXN
Control Panel - 192.168.0.113
admin
File
Settings
Help
Home
Users
Events
Settings
Help
OZEKI
www.myozeki.com
Users
Setup users accounts
Groups
Users can be assigned to multiple groups
Authentication
Select your preferred authentication method
Authentication providers
Create new Authentication provider
Search...
Authentication provider details
HTTP
RADIUS
Database
LDAP
Delete
0/4 item selected
Start
Control Panel
8:40

Please select the authentication type you wish to create.

| N° | Provider name | Description |
|---|---|---|
| 1. | Local system users | Users are authenticated us localy created user accounts. |
| 2. | Chat | Users are authenticated by remote chat servers. |
| 3. | Network users | Users are authenticated by a local network service such as SMPP, UCP, HTTP API, etc. |
| 4. | Ozeki account holders | Users are authenticated using their Ozeki account. |

Figure 4 - Create new HTTP provider

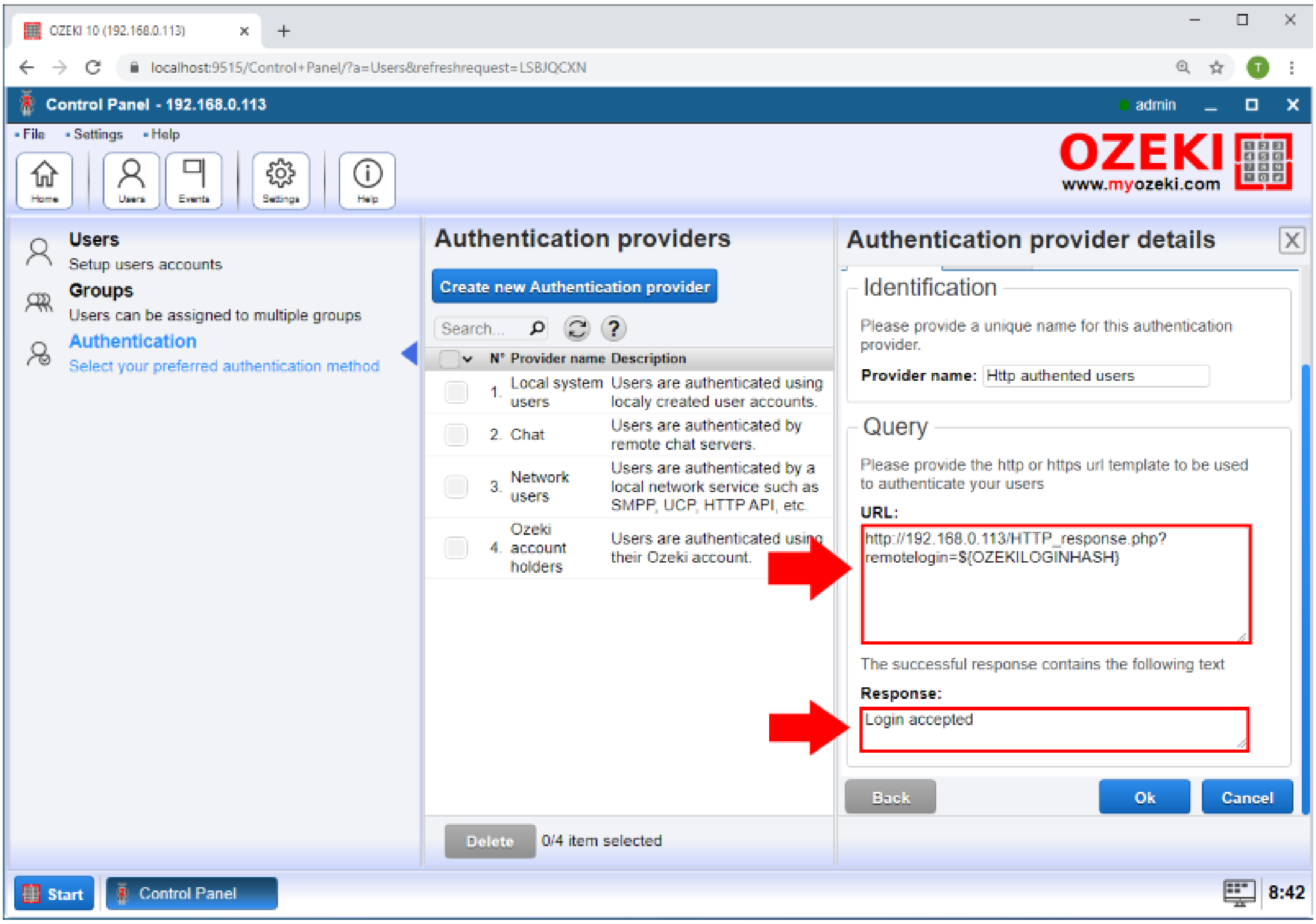
Please fill the HTTP Authentication Provider form (Figure 5) depending on your HTTP server IP and PHP file name. It is essential to set OZEKILOGINHASH as a HTTP parameter. You also need to provide the correct HTTP response Ozeki SMS Gateway should wait for.

The example HTTP request set for Ozeki SMS Gateway logins:

```
http://192.168.0.113/HTTP_response.php?remotelogin=${OZEKILOGINHASH}
```

The example HTTP response Ozeki SMS Gateway is waiting for:

```
Login accepted
```



Users
Setup users accounts
Groups
Users can be assigned to multiple groups
Authentication
Select your preferred authentication method

Authentication providers

Create new Authentication provider

Search...

| N° | Provider name | Description |
|---|---|---|
| 1. | Local system users | Users are authenticated using localy created user accounts. |
| 2. | Chat | Users are authenticated by remote chat servers. |
| 3. | Network users | Users are authenticated by a local network service such as SMPP, UCP, HTTP API, etc. |
| 4. | Ozeki account holders | Users are authenticated using their Ozeki account. |

Delete 0/4 item selected

Authentication provider details

Identification

Please provide a unique name for this authentication provider.

Provider name: Http authented users

Query

Please provide the http or https url template to be used to authenticate your users

URL: http://192.168.0.113/HTTP_response.php?remotelogin=${OZEKILOGINHASH}

The successful response contains the following text

Response: Login accepted

Back Ok Cancel

Figure 5 - Authentication provider details

It is time to write your PHP script and fill out the HTTP authentication form in Ozeki SMS Gateway. If you are experimenting on the field we advise to install an Apache which is a HTTP server. Please make sure that port 80 is free for your HTTP server. We have named the example file 'HTTP_response.php'.

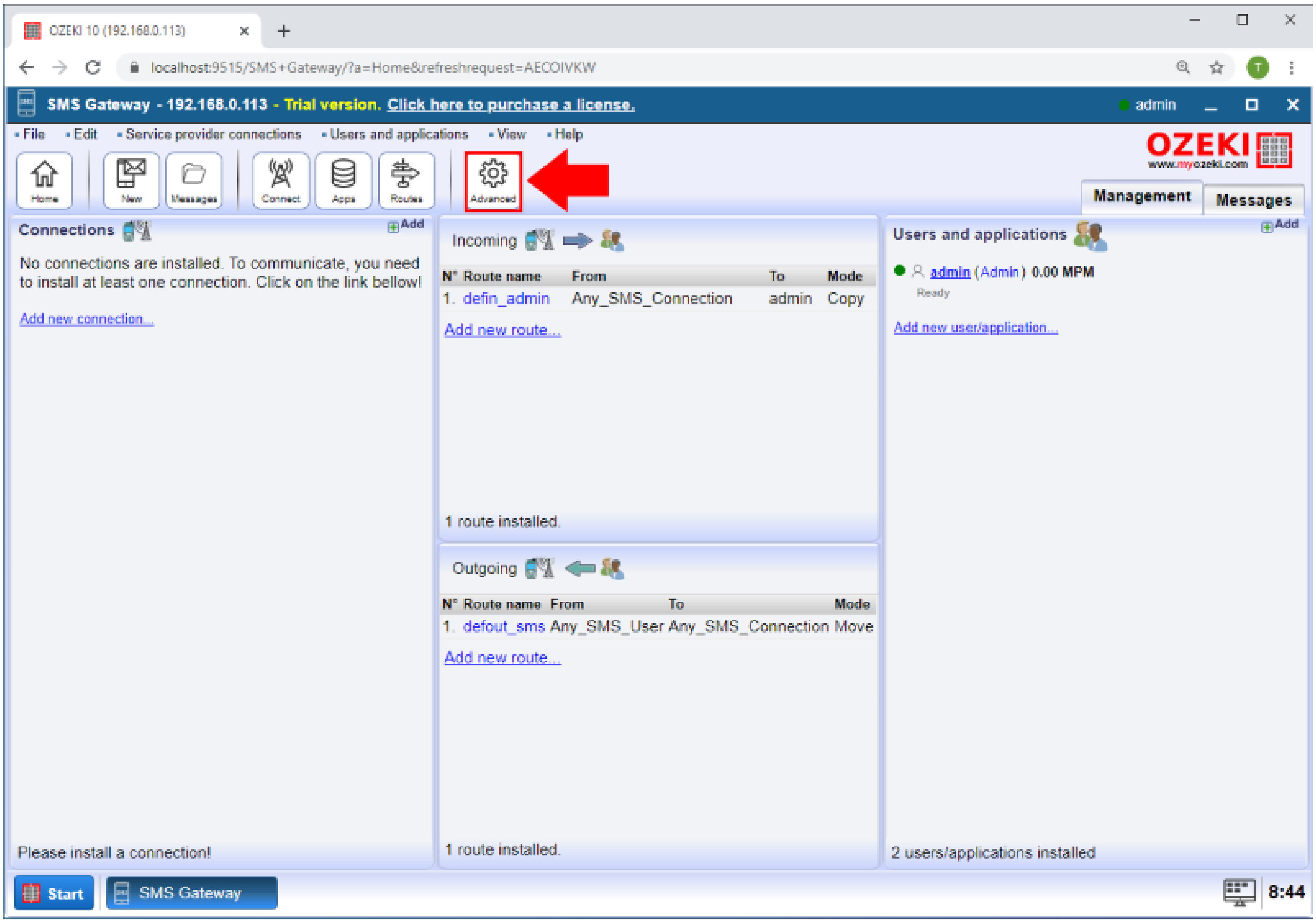
Here you can see the PHP example code:

```php
<?php
$remotelogin = ''; //the HTTP parameter of the OZEKILOGINHASH
$array = array(
    "Ozeki" => "462ac6d8aadf8b8f909d859c4aa696fa", //passw 12345
    "smppuser" => "296adaf6fd0612bfefacd1055f03c31c" //passw qwe123
);
if(isset($_REQUEST['remotelogin'])) $remotelogin = $_REQUEST['remotelogin'];

if($remotelogin == $array["Ozeki"] || $remotelogin == $array["smppuser"])
    echo 'Login accepted';
else
    echo 'Login refused';
?>
```

OZEKILOGINHASH contains an encrypted username and password pair. Comparing the OZEKILOGINHASH with your OZEKILOGINHASH-ed user list can help authenticate logins.

## Step 2 - Create SMPP Service with HTTP Authentication

The next step is to open the SMS Gateway application. So, just navigate to the desktop of, and here just open SMS Gateway by clicking on its icon. Here syu can find the Advanced menu of the SMS Gateway. To open it please click the "Advanced" button on the main page.



No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

Please install a connection!

Incoming

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defin_admin | Any_SMS_Connection | admin | Copy |

Add new route...

1 route installed.

Outgoing

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move |

Add new route...

1 route installed.

Users and applications

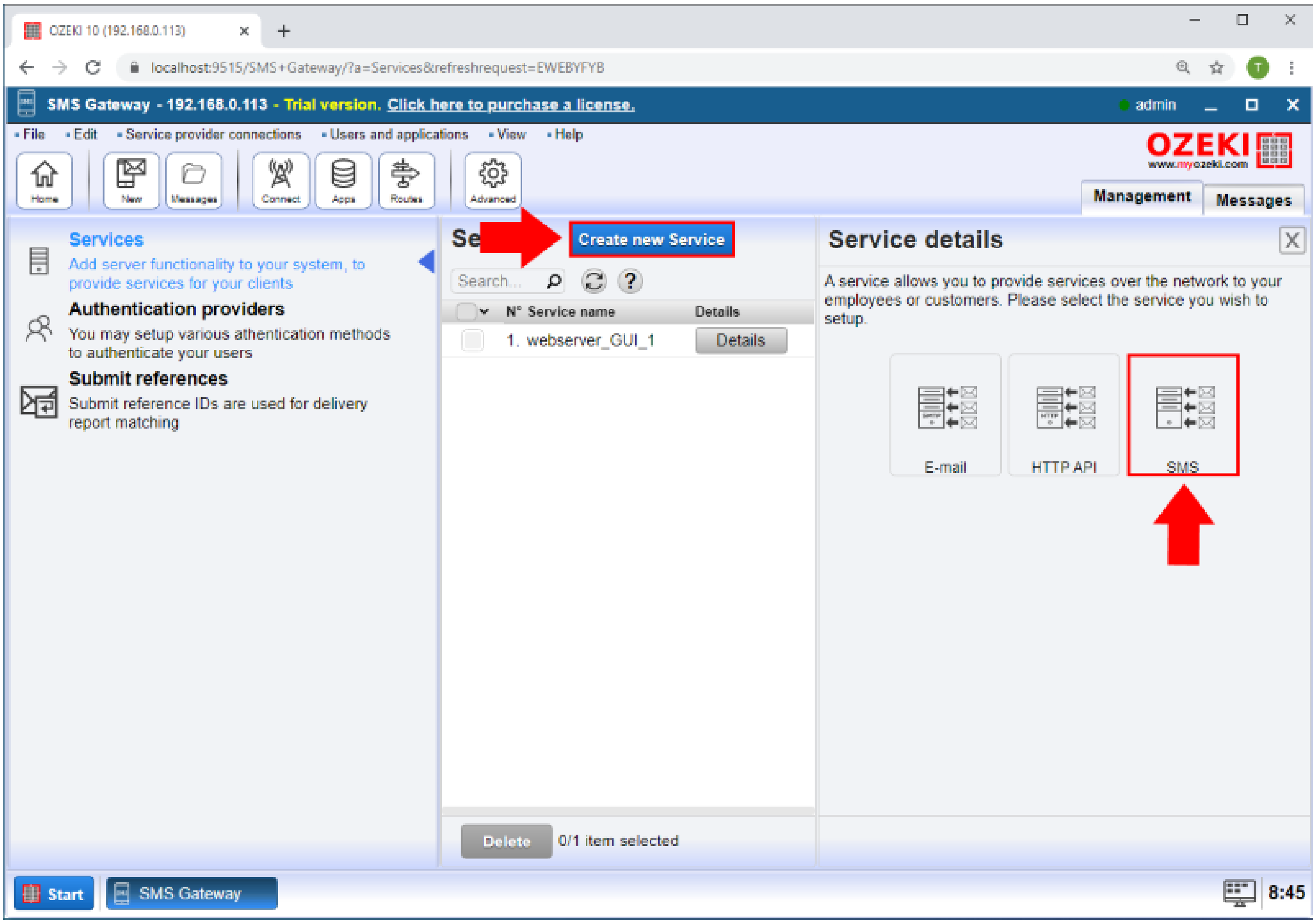
● admin (Admin) 0.00 MPM
Ready

Add new user/application...

2 users/applications installed

Figure 6 - Open SMS Gateway advanced menu

On the Advanced page you can create, modify or delete SMS Services. Create a new SMS Services by clicking the blue Create new Service button. A new box will open on the right side of the page. In this box you can select between SMS Services as you can see in the Figure 7. And then please choose SMPP Service (Figure 6).



Services

Add server functionality to your system, to provide services for your clients

Authentication providers

You may setup various athentication methods to authenticate your users

Submit references

Submit reference IDs are used for delivery report matching

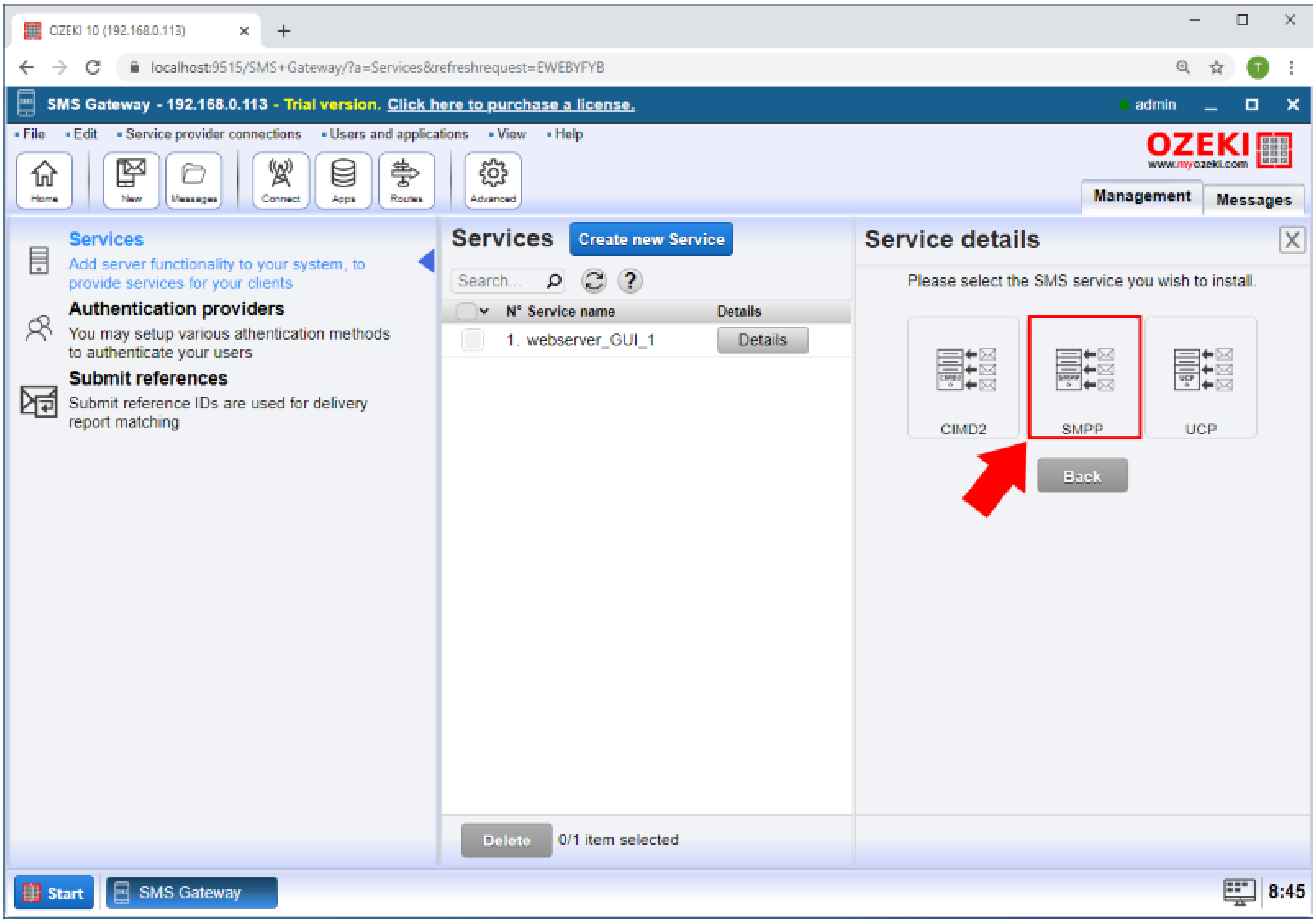
Service details

A service allows you to provide services over the network to your employees or customers. Please select the service you wish to setup.

1. webserver_GUI_1 Details

0/1 item selected

Figure 7 - Create new SMS Service



Services

Add server functionality to your system, to provide services for your clients

Authentication providers

You may setup various athentication methods to authenticate your users

Submit references

Submit reference IDs are used for delivery report matching

Services

Service details

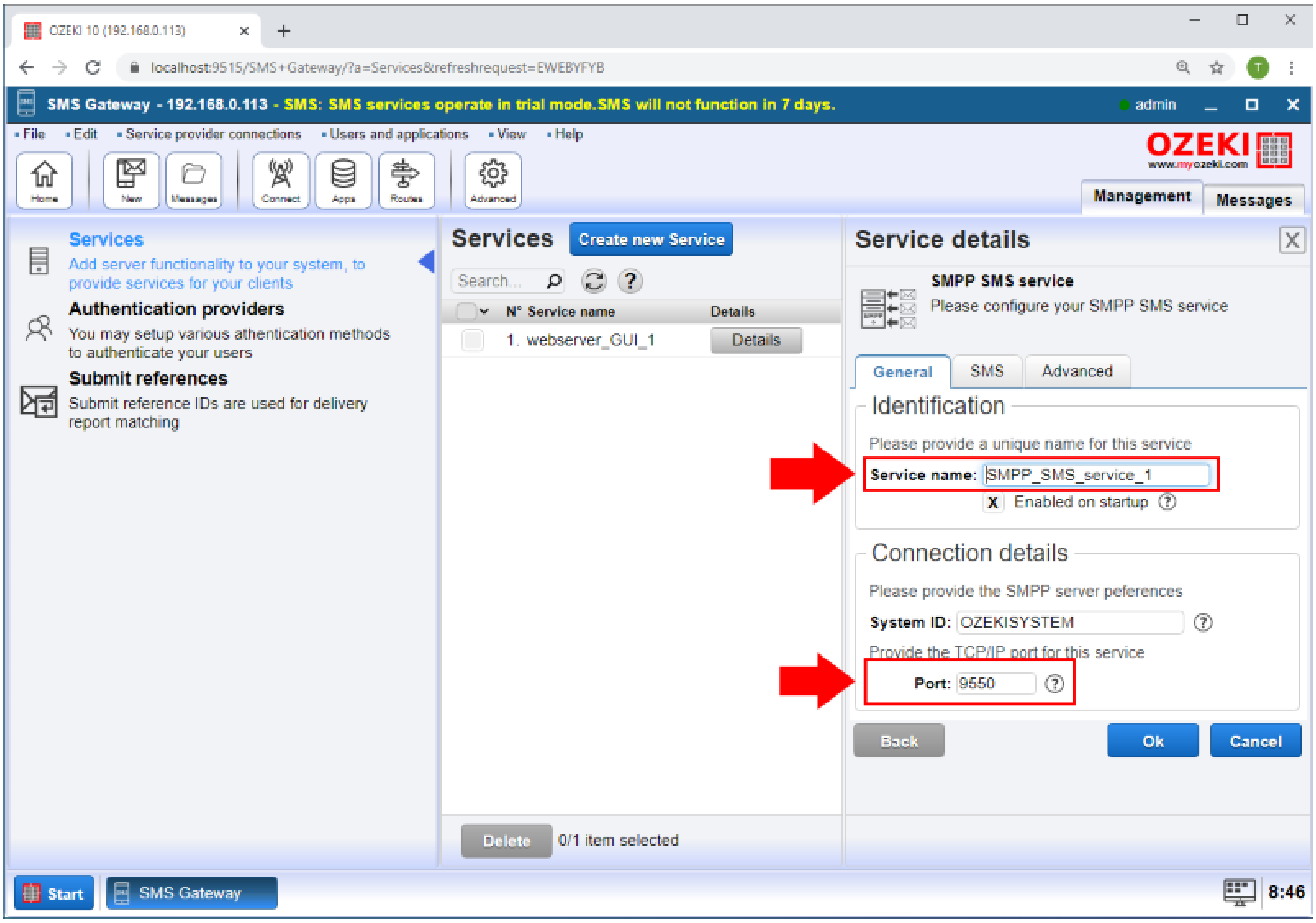
Please select the SMS service you wish to install.

1. webserver_GUI_1 Details

0/1 item selected

Figure 8 - Create SMPP Service

A form will be available in the box. You should fill out the form to configure the SMPP Service. Please provide a unique name and a port for this service as you can see it in Figure 9.



SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.

Service details

SMPP SMS service

Please configure your SMPP SMS service

Identification

Please provide a unique name for this service

Service name: SMPP_SMS_service_1

☒ Enabled on startup

Connection details

Please provide the SMPP server peferences

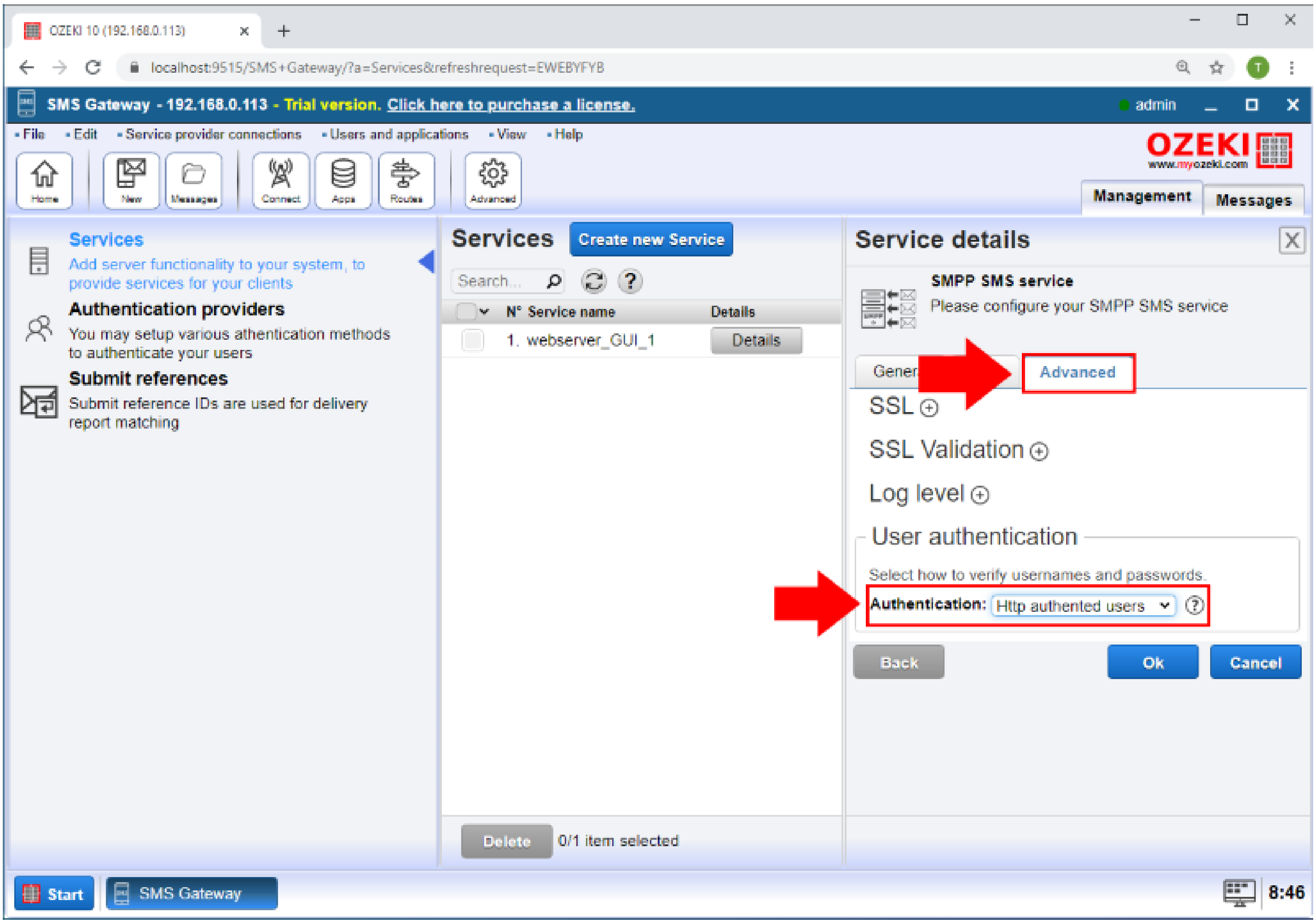
System ID: OZEKISYSTEM

Provide the TCP/IP port for this service

Port: 9550

Figure 9 - SMPP Service details

Then on the Advanced tab of the SMPP Service set the HTTP authentication provider in the User Authentication section as the Figure 10 shows.

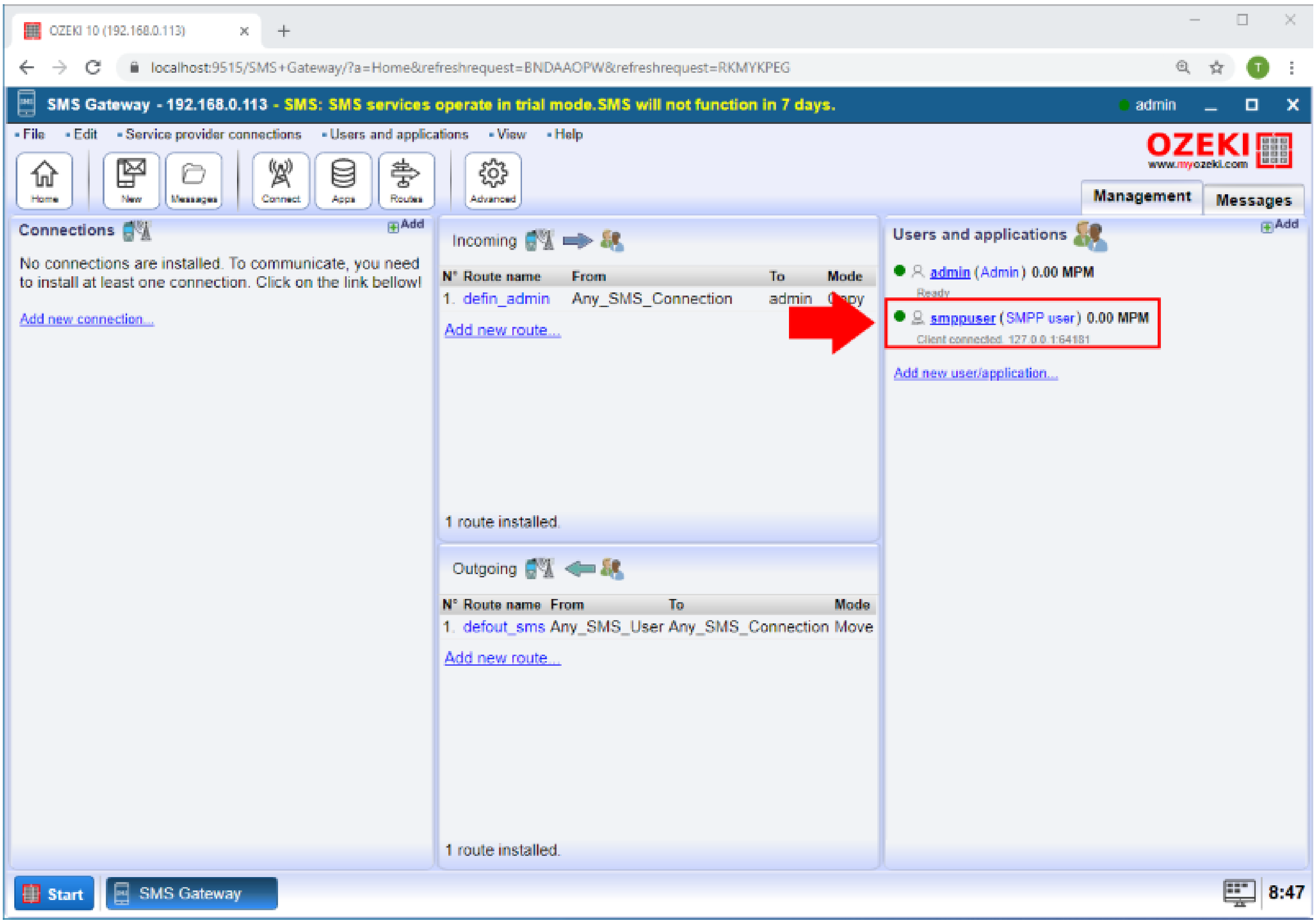


Select how to verify usernames and passwords.

Authentication: Http authented users

Figure 10 - Set Authentication provider

Finally if an SMPP Client is connecting with the username and password you created in the database, you will see that the User is appear in the SMS Gateway Users and applications section (Figure 11).



SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.

Connections

No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

Incoming

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defin_admin | Any_SMS_Connection | admin | Copy |

Add new route...

1 route installed.

Outgoing

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move |

Add new route...

1 route installed.

Users and applications

● admin (Admin) 0.00 MPM
Ready

● smppuser (SMPP user) 0.00 MPM
Client connected. 127.0.0.1:64181

Add new user/application...

Figure 11 - SMPP user connected

# How to use a database to save all SMS messages

***This guide explains how to setup a Microsoft SQL Database server to save all SMS messages going through the system. This setup makes it possible for you to create reports, write bills or to calculate the cost of operating your SMS gateway system.***

Firstly open the Reporting from the Edit menu in the navigation bar (Figure 1).

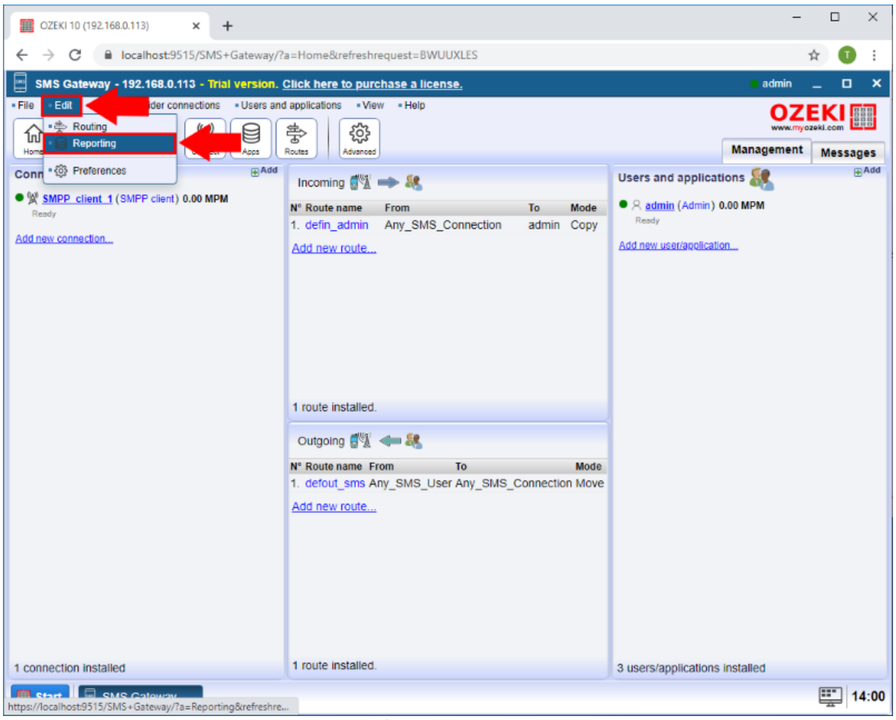


**Figure 1** - Open Reporting menu

In the Reporting menu click on the Create new SMS Reporter button and than select MS SQL type database (Figure 2).

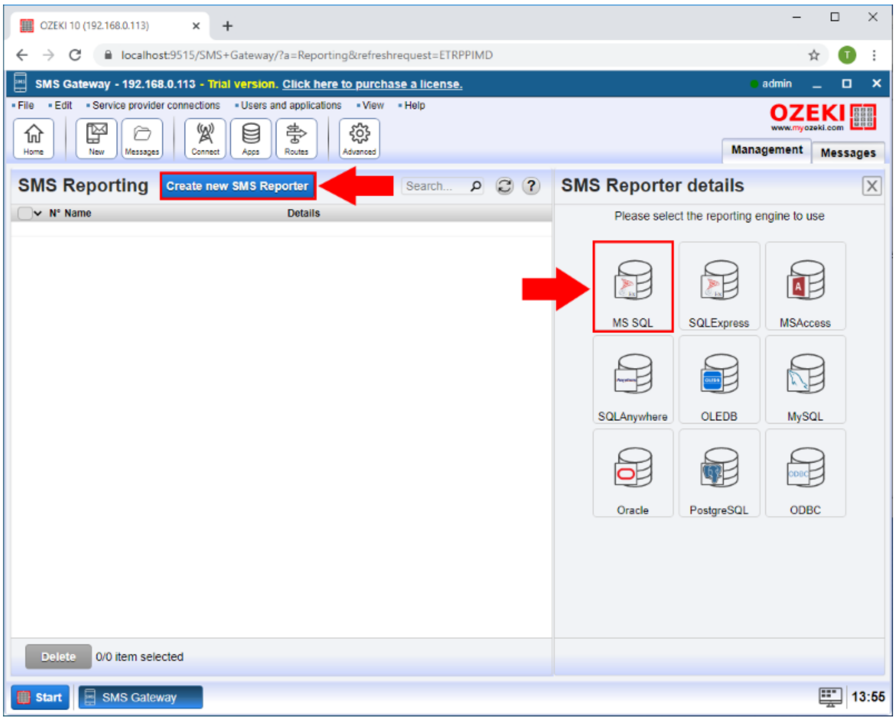
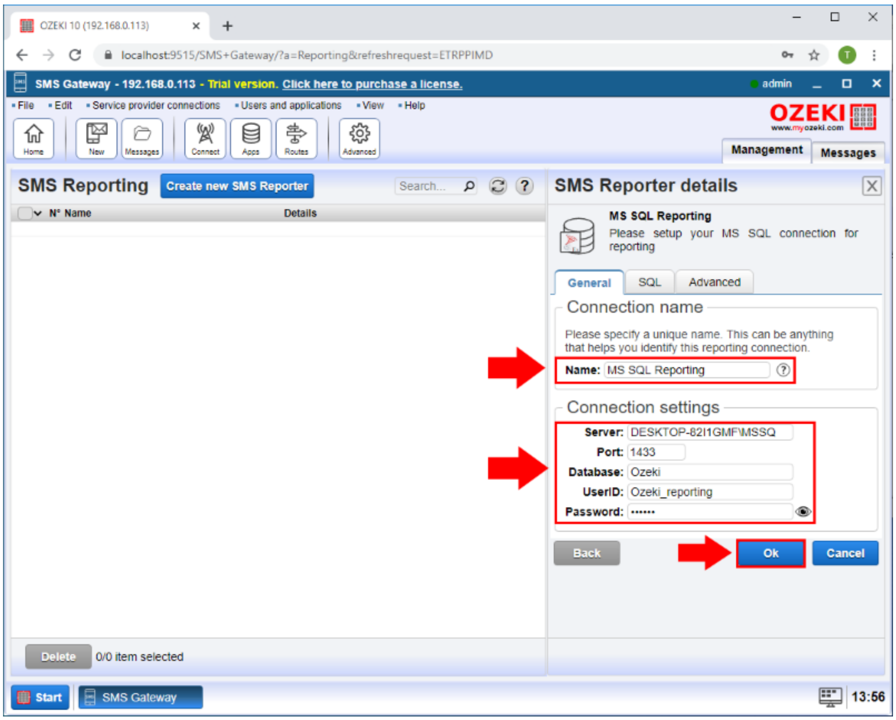


**Figure 2 -** Create new MSSQL reporter

Now enter the connection details like Server, Port, Database, UserID and Password. And enter the Connection name. Finally click on the OK button (Figure 3).



SMS Reporter details

MS SQL Reporting

Please setup your MS SQL connection for reporting

Connection name

Please specify a unique name. This can be anything that helps you identify this reporting connection.

Name: MS SQL Reporting

Connection settings

Server: DESKTOP-82I1GMF\MSSQ

Port: 1433

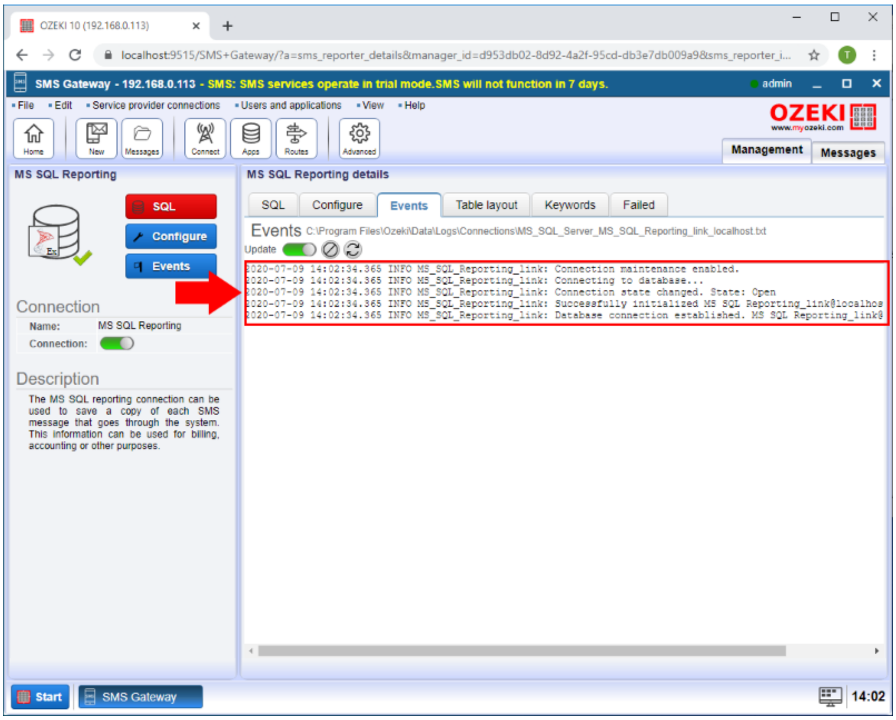
Database: Ozeki

UserID: Ozeki_reporting

Password: ••••••

**Figure 3 -** Specify connection details

In the Events tab you can see the logging of the latest server events. As you can see on the Figure 4 the database connection is established.



MS SQL Reporting details

Events C:\Program Files\Ozeki\Data\Logs\Connections\MS_SQL_Server_MS_SQL_Reporting_link_localhost.txt

```
2020-07-09 14:02:34.365 INFO MS_SQL_Reporting_link: Connection maintenance enabled.
2020-07-09 14:02:34.365 INFO MS_SQL_Reporting_link: Connecting to database...
2020-07-09 14:02:34.365 INFO MS_SQL_Reporting_link: Connection state changed. State: Open
2020-07-09 14:02:34.365 INFO MS_SQL_Reporting_link: Successfully initialized MS SQL Reporting_link@localhos
2020-07-09 14:02:34.365 INFO MS_SQL_Reporting_link: Database connection established. MS SQL Reporting_link@
```

Connection

Name: MS SQL Reporting

Connection: ●

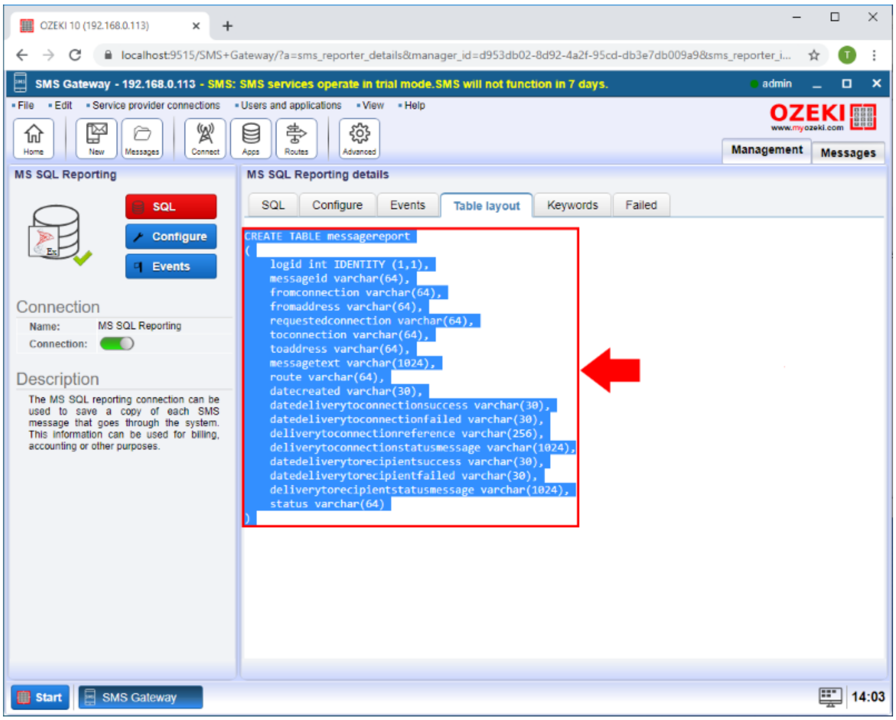
Description

The MS SQL reporting connection can be used to save a copy of each SMS message that goes through the system. This information can be used for billing, accounting or other purposes.

OZEKI 10 (192.168.0.113)
localhost:9515/SMS+Gateway/?a=sms_reporter_details&manager_id=d953db02-8d92-4a2f-95cd-db3e7db009a9&sms_reporter_i...
SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
OZEKI
Management
Messages
MS SQL Reporting
SQL
Configure
Events
SQL
Configure
Events
Table layout
Keywords
Failed
Update
Start
SMS Gateway
14:02

**Figure 4 -** Database connection established

On the Table layout tab you can find the corresponding query to ctrete the table for the reporting. Copy it from here (Figure 5).



Connection

Name: MS SQL Reporting

Connection:

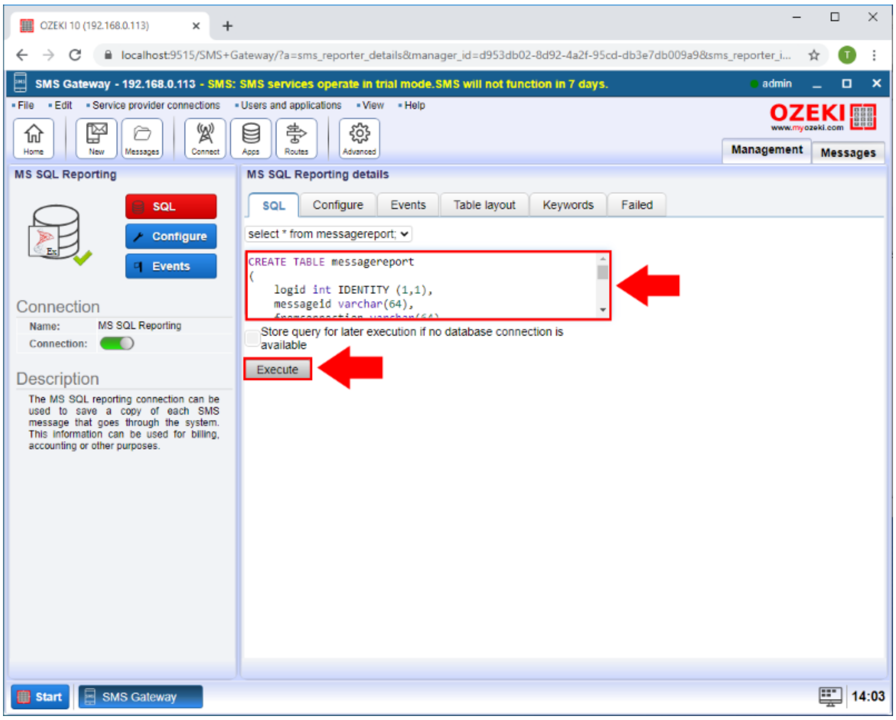
Description

The MS SQL reporting connection can be used to save a copy of each SMS message that goes through the system. This information can be used for billing, accounting or other purposes.

```
CREATE TABLE messagereport
(
    logid int IDENTITY (1,1),
    messageid varchar(64),
    fromconnection varchar(64),
    fromaddress varchar(64),
    requestedconnection varchar(64),
    toconnection varchar(64),
    toaddress varchar(64),
    messagetext varchar(1024),
    route varchar(64),
    datecreated varchar(30),
    datedeliverytoconnectionsuccess varchar(30),
    datedeliverytoconnectionfailed varchar(30),
    deliverytoconnectionreference varchar(256),
    deliverytoconnectionstatusmessage varchar(1024),
    datedeliverytorecipientsuccess varchar(30),
    datedeliverytorecipientfailed varchar(30),
    deliverytorecipientstatusmessage varchar(1024),
    status varchar(64)
)
```

**Figure 5** - Copy table layout

On the SQL tab you can execute SQL queries in the database connection. Paste here the create table query and execute it (Figure 6).



Connection

Name: MS SQL Reporting

Connection:

Description

The MS SQL reporting connection can be used to save a copy of each SMS message that goes through the system. This information can be used for billing, accounting or other purposes.

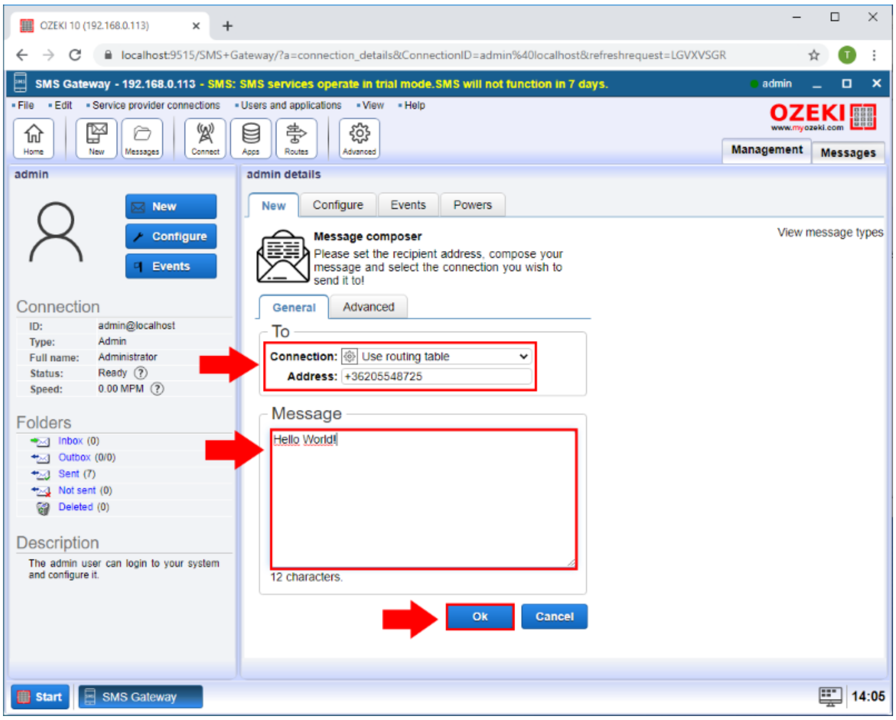
select * from messagereport;

```
CREATE TABLE messagereport
(
    logid int IDENTITY (1,1),
    messageid varchar(64),
```

□ Store query for later execution if no database connection is available

**Figure 6** - Execute create table query

From the Admin user you can send test SMS message. Provide the recipient address, the message and click on the OK button (Figure 7).



Message composer

Please set the recipient address, compose your message and select the connection you wish to send it to!

Connection: Use routing table

Address: +36205548725

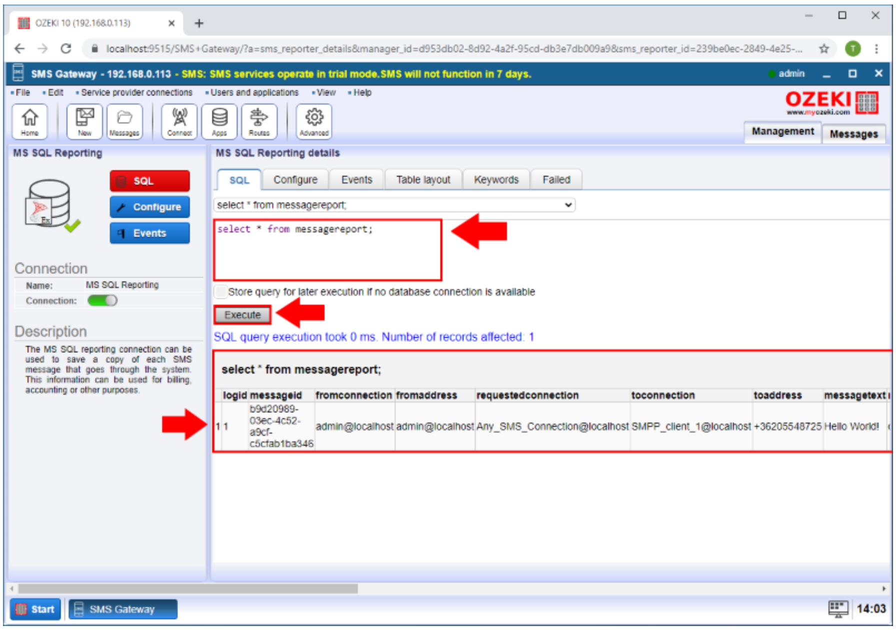
Message: Hello World!

12 characters.

SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
admin details
New
Configure
Events
Powers
General
Advanced
To
Ok
Cancel
View message types

**Figure 7 -** Send test message

Finally in the MS SQL Reporter you can query the messages from the database and you will se there is a new message in the table (Figure 8).



```
select * from messagereport;
```

□ Store query for later execution if no database connection is available

SQL query execution took 0 ms. Number of records affected: 1

select * from messagereport;

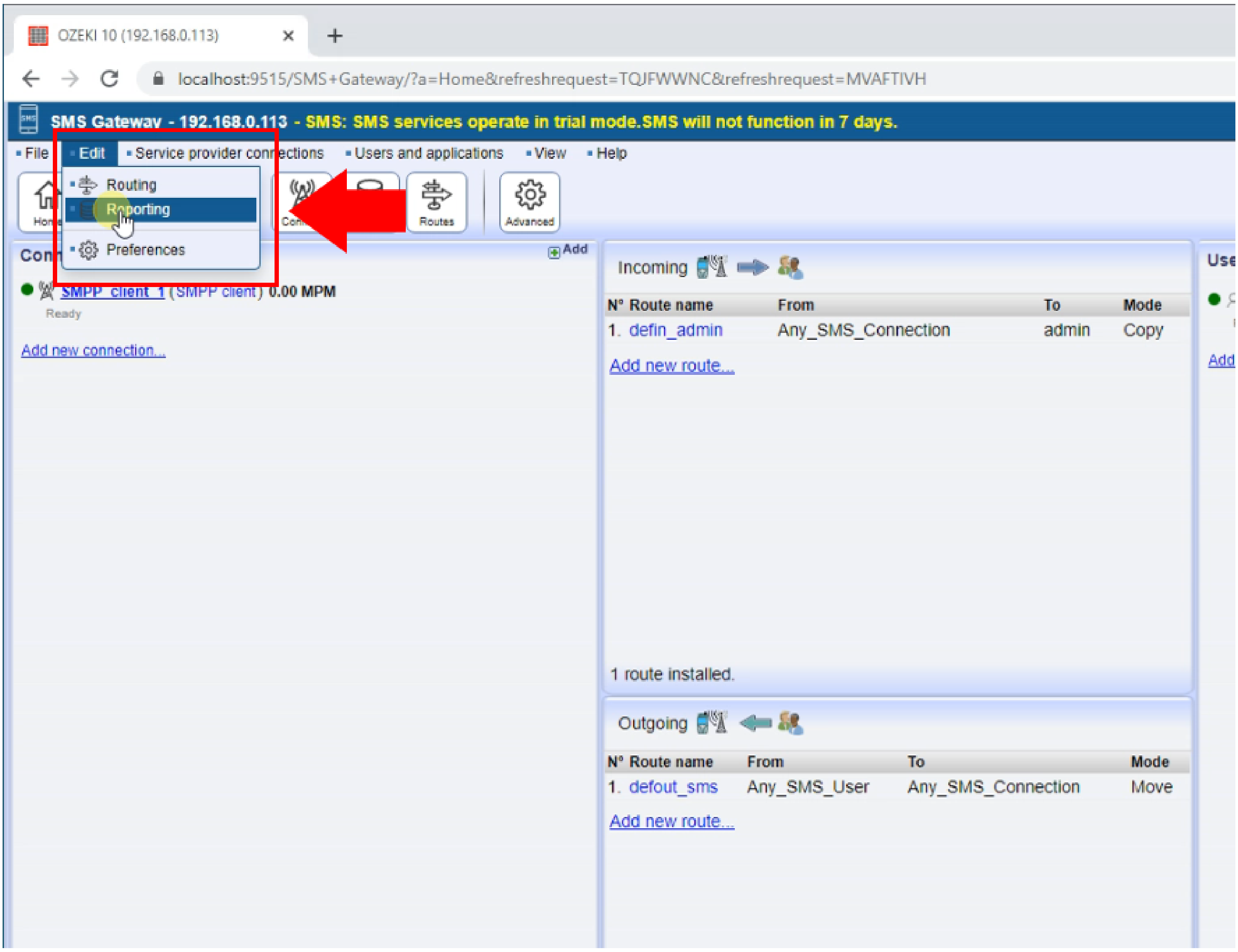
| | logid | messageid | fromconnection | fromaddress | requestedconnection | toconnection | toaddress | messagetext |
|---|---|---|---|---|---|---|---|---|
| 1 | 1 | b9d20989-03ec-4c52-a9cf-c5cfab1ba346 | admin@localhost | admin@localhost | Any_SMS_Connection@localhost | SMPP_client_1@localhost | +36205548725 | Hello World! |

**Figure 8 -** Query messages from database

# How to save all SMS messages into a CSV

***This guide gives you the steps to take to save all SMS messages going through your system into a CSV text file. This file can be later loaded into Microsoft Excel, WPS Office or any other spreadsheep application.***

The reporting system can save messages going through the system into different sources, such as databases, txt files, etc. One of the most popular choice is using CSV files to save a log of each message. You are able to open reporting system from the Edit menu Reporting option as you can see it on the Figure 1.


OZEKI 10 (192.168.0.113)
localhost:9515/SMS+Gateway/?a=Home&refreshrequest=TQJFWWNC&refreshrequest=MVAFTIVH
SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
File
Edit
Service provider connections
Users and applications
View
Help
Routing
Reporting
Preferences
Routes
Advanced
Add
SMPP_client_1 (SMPP client) 0.00 MPM
Ready
Add new connection...
Incoming
Outgoing
Use
Add

Incoming

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defin_admin | Any_SMS_Connection | admin | Copy |

Add new route...

1 route installed.

Outgoing

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move |

Add new route...

Figure 1 - Open reporting menu

In the Reporting menu click on the Create new SMS Reporter button and than select Csv reporting type (Figure 2).

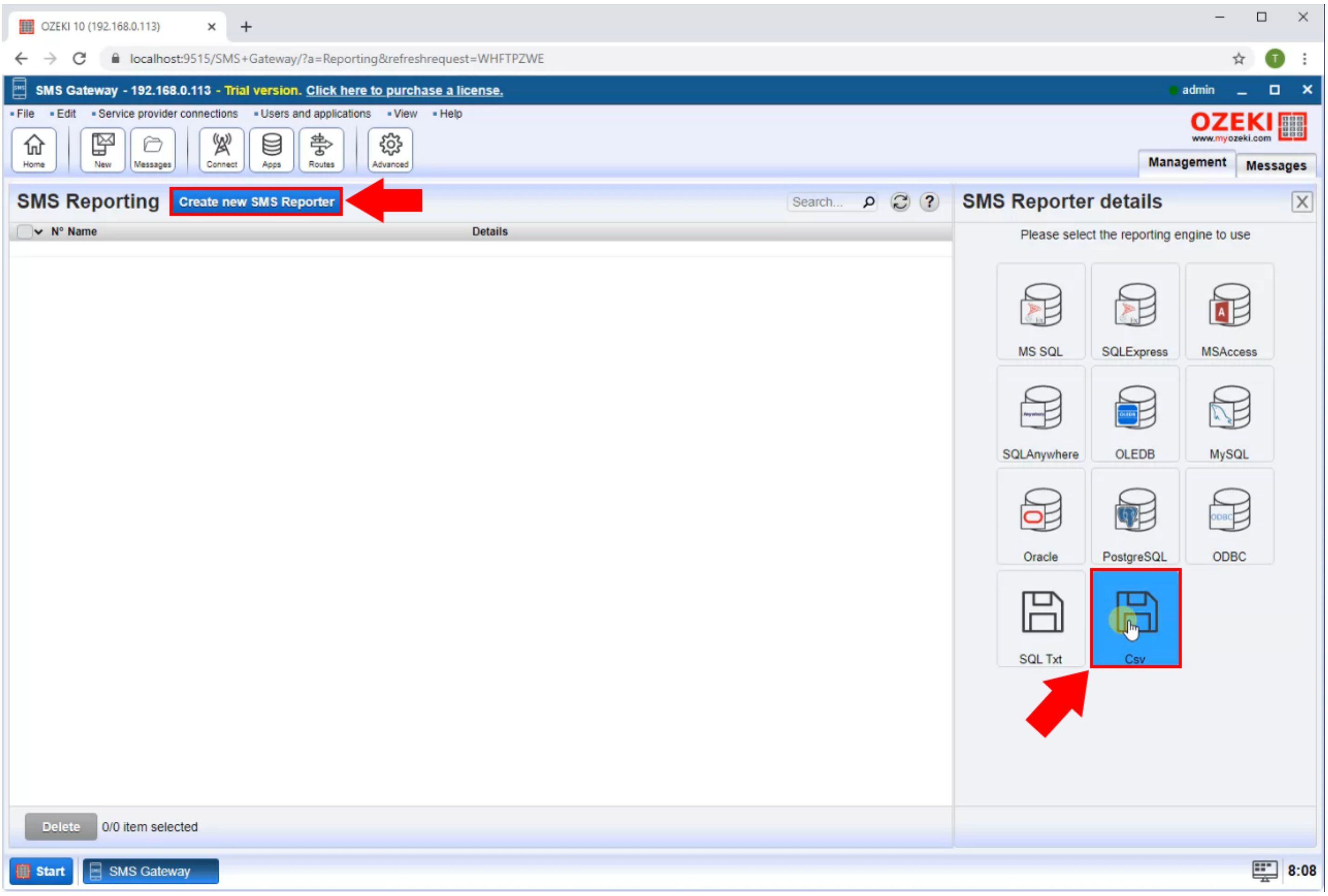
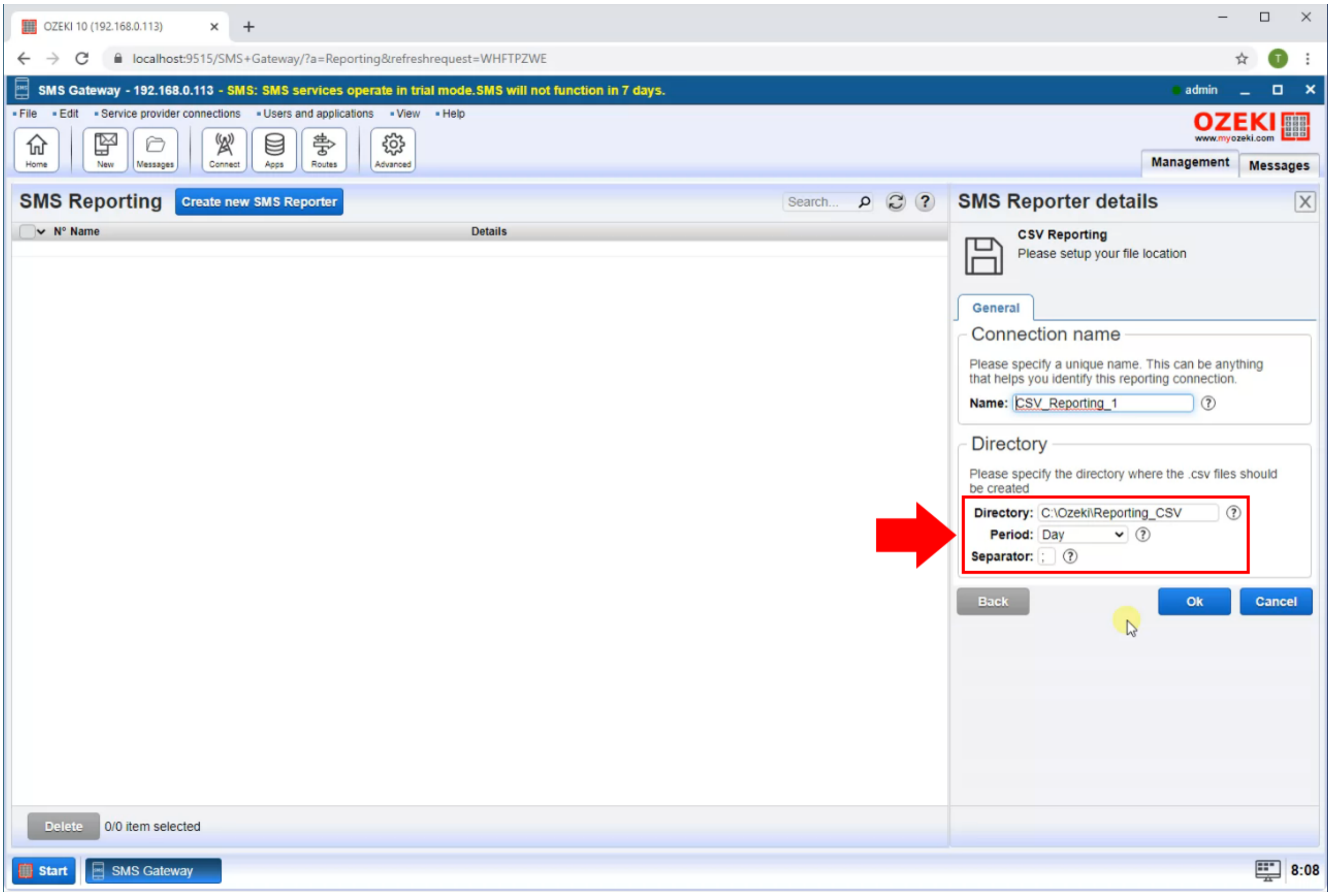


Figure 2 - Add Csv reporter

The details page of the Csv reporter allows you to configure the file location and when will be created an independent .csv file for the selected period by the SMS Gateway. The separator character used to separate items in the list (Figure 3).



SMS Reporter details

**CSV Reporting**
Please setup your file location

General

Connection name

Please specify a unique name. This can be anything that helps you identify this reporting connection.

Name: CSV_Reporting_1

Directory

Please specify the directory where the .csv files should be created

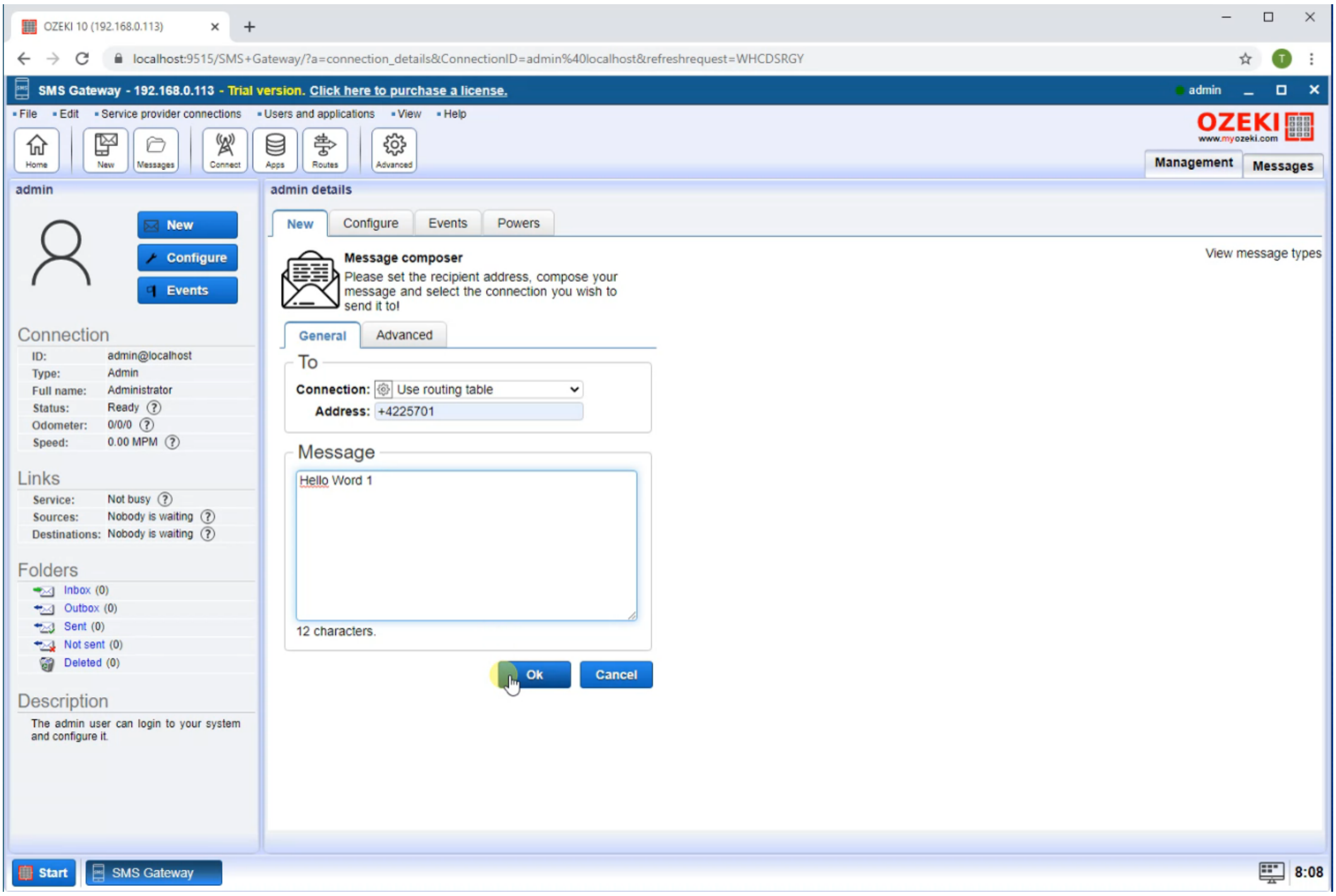
Directory: C:\Ozeki\Reporting_CSV

Period: Day

Separator: ;

Figure 3 - Configure Csv reporter

From the Admin user you can send SMS message. Provide the recipient address, the message and click on the OK button (Figure 4).


OZEKI 10 (192.168.0.113)
localhost:9515/SMS+Gateway/?a=connection_details&ConnectionID=admin%40localhost&refreshrequest=WHCDSRGY
SMS Gateway - 192.168.0.113 - Trial version. Click here to purchase a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
New
Configure
Events
Start
SMS Gateway
8:08

admin

Connection

| | |
|---|---|
| ID: | admin@localhost |
| Type: | Admin |
| Full name: | Administrator |
| Status: | Ready |
| Odometer: | 0/0/0 |
| Speed: | 0.00 MPM |

Links

| | |
|---|---|
| Service: | Not busy |
| Sources: | Nobody is waiting |
| Destinations: | Nobody is waiting |

Folders

- Inbox (0)
- Outbox (0)
- Sent (0)
- Not sent (0)
- Deleted (0)

Description

The admin user can login to your system and configure it.

admin details

New
Configure
Events
Powers
View message types

**Message composer**
Please set the recipient address, compose your message and select the connection you wish to send it to!

To

Connection: Use routing table

Address: +4225701

Message

Hello Word 1

12 characters.

Figure 4 - Send Messages

Now in the folder what you set in the Csv reporter details page you can see the .csv text file is created. The default folder is C:\Ozeki (Figure 5).

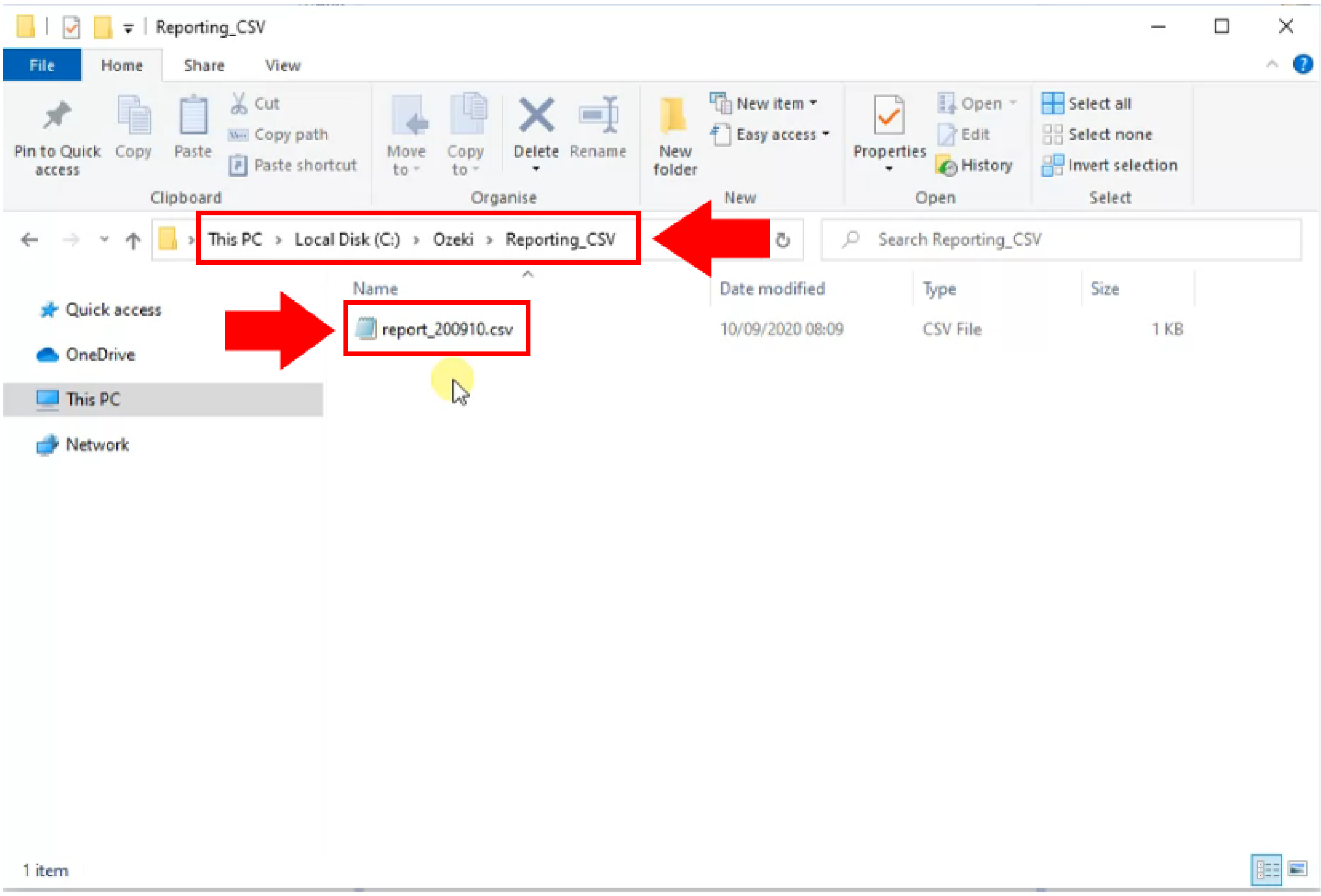
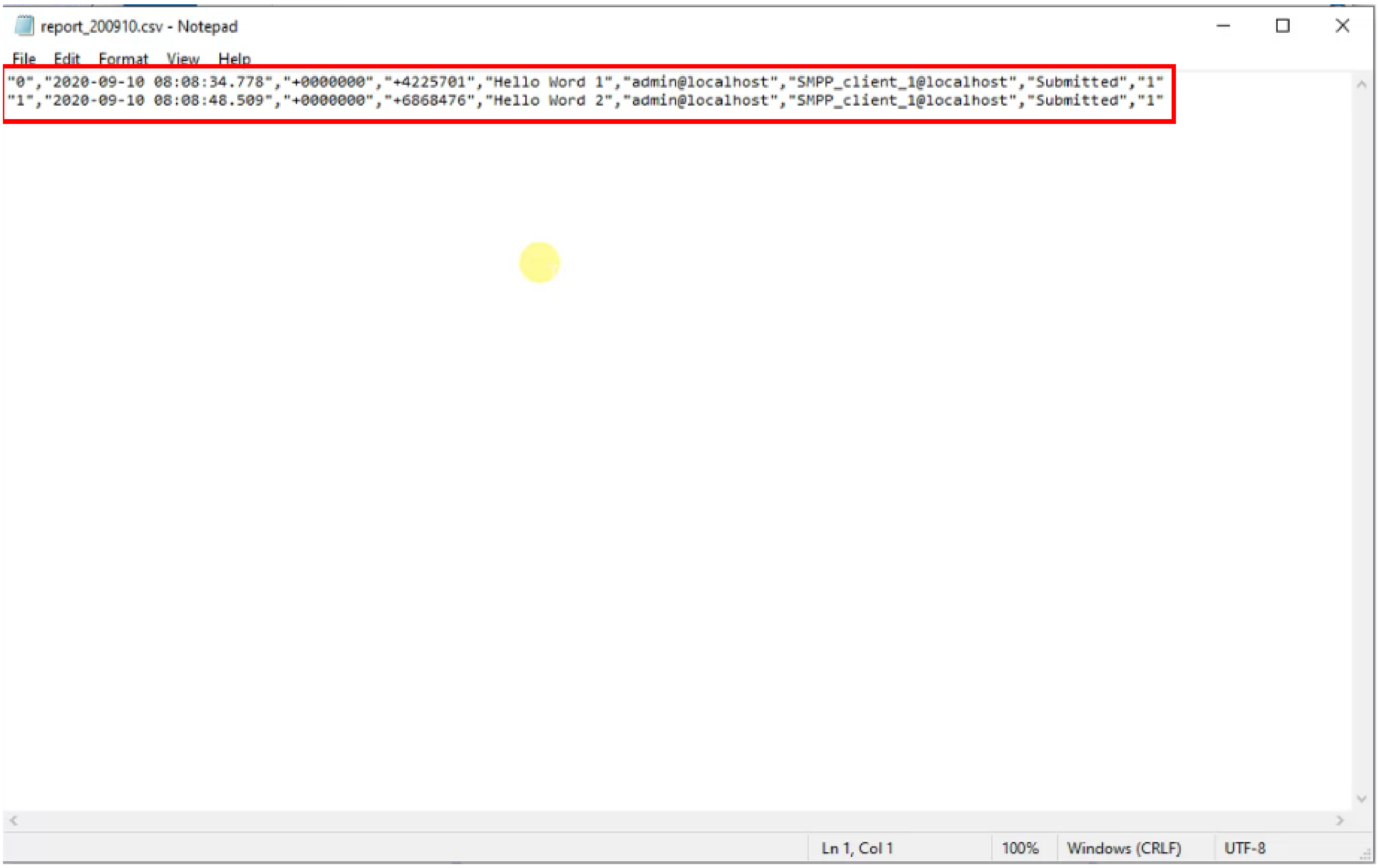


Figure 5 - Csv file in folder

Finally if you are open the .csv text file you can see that all the sent and received sms messages are stored as the Figure 6 shows. This file can be later loaded into Microsoft Excel, WPS Office or any other spreadsheep application.



```
"0","2020-09-10 08:08:34.778","+0000000","+4225701","Hello Word 1","admin@localhost","SMPP_client_1@localhost","Submitted","1"
"1","2020-09-10 08:08:48.509","+0000000","+6868476","Hello Word 2","admin@localhost","SMPP_client_1@localhost","Submitted","1"
```

Figure 6 - Csv file content

# How to save all SMS messages into an SQL Text file

***For high performance SMS systems, where message speeds go above 100 MPS (Messages Per Second), the database servers often cannot keep up with the load put onto them by the standard database reporting feature of Ozeki 10. They simply cannot execute several hundreds of thousands of SQL queries per second. To overcome this difficulty, Ozeki SMS Gateway provies SQL Text file reporting. This feature records each SQL command used for database reporting into a TXT file. You can use these txt files to do batch inserts into your database, to have a log of all messages. This guide explains how to setup this feature.***

The reporting system can save messages going through the system into different sources, such as databases, txt files, etc. One of the most popular choice is using SQL text files to save a log of each message. You are able to open reporting system from the Edit menu Reporting option as you can see it on the Figure 1.

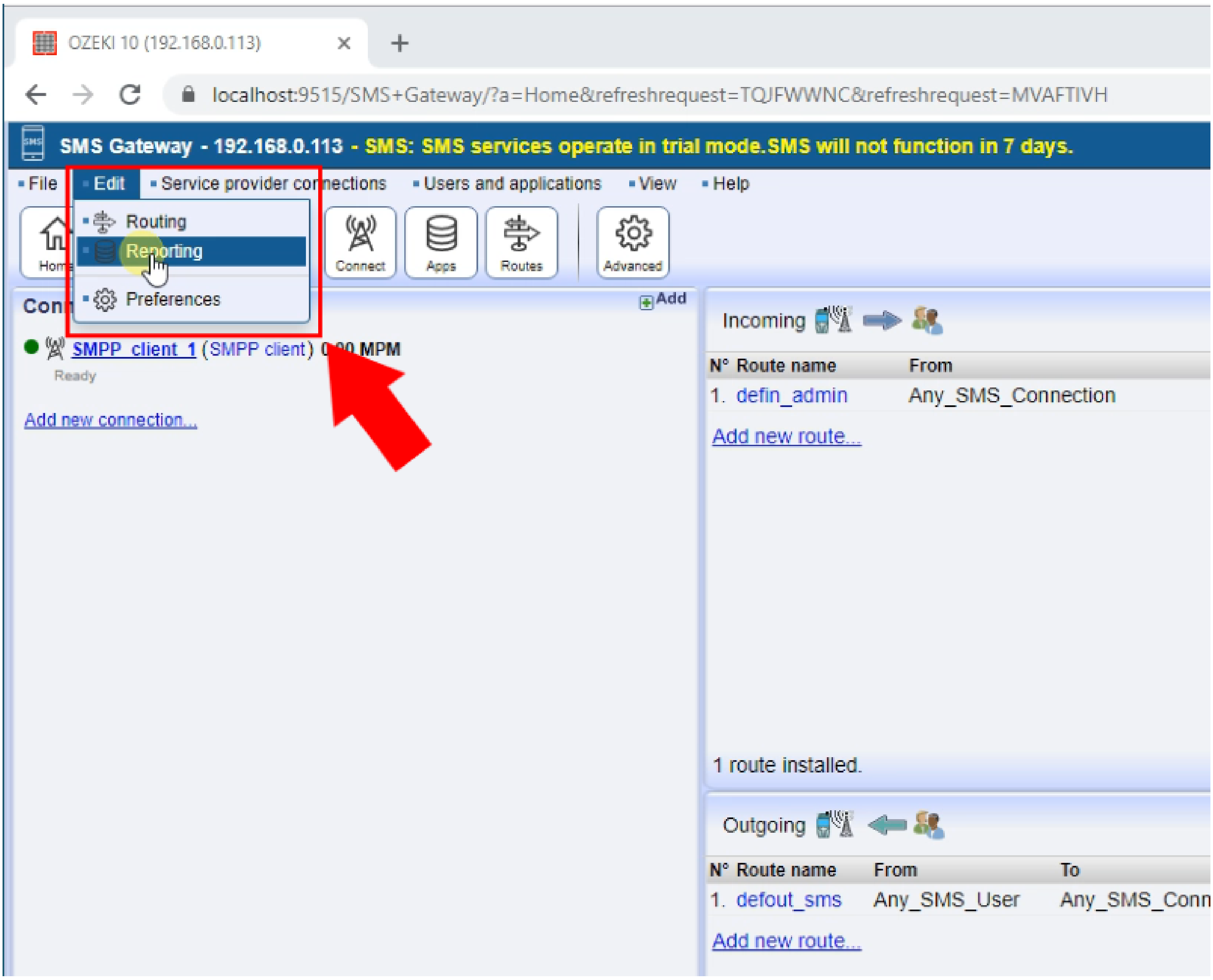

OZEKI 10 (192.168.0.113)
localhost:9515/SMS+Gateway/?a=Home&refreshrequest=TQJFWWNC&refreshrequest=MVAFTIVH
SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
File
Edit
Service provider connections
Users and applications
View
Help
Routing
Reporting
Preferences
Home
Connect
Apps
Routes
Advanced
Con
Add
SMPP_client_1 (SMPP client) 0.00 MPM
Ready
Add new connection...
Incoming
N° Route name From
1. defin_admin Any_SMS_Connection
Add new route...
1 route installed.
Outgoing
N° Route name From To
1. defout_sms Any_SMS_User Any_SMS_Conn
Add new route...

Figure 1 - Open reporting menu

In the Reporting menu click on the Create new SMS Reporter button and than select SQL Txt reporting type (Figure 2).

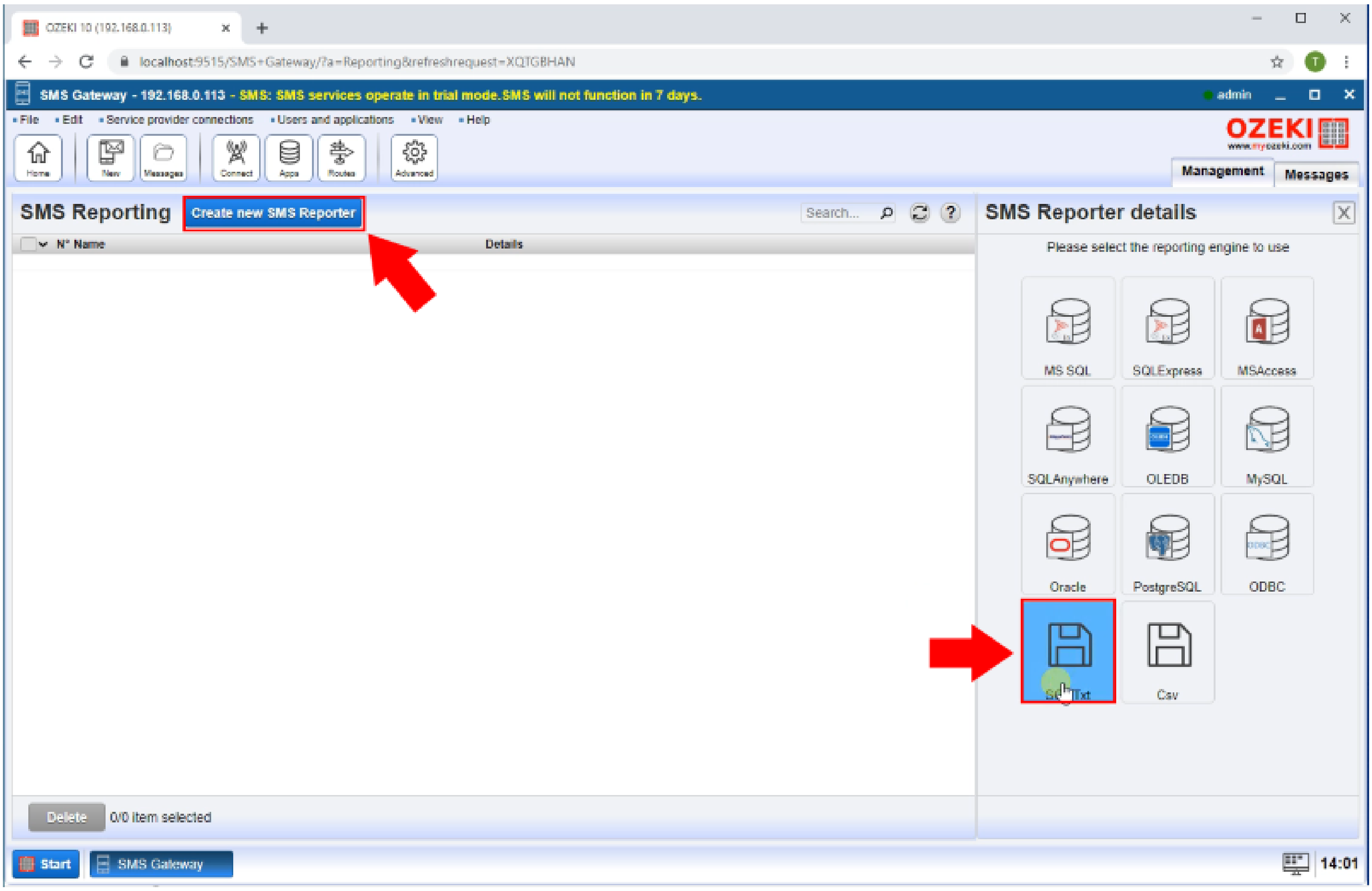
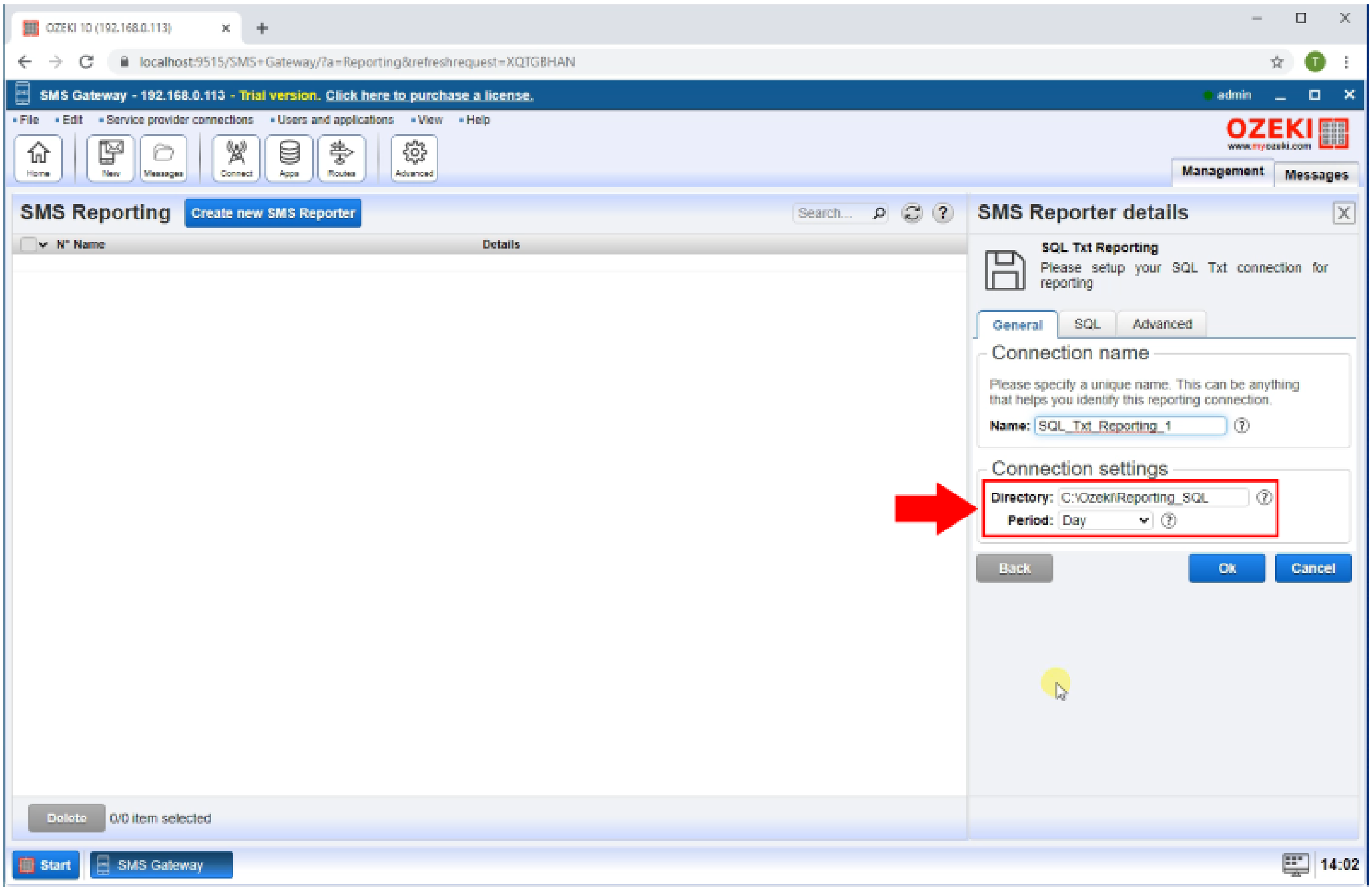


Figure 2 - Add SQL Txt reporter

The details page of the SQL Txt reporter allows you to configure the file location and when will be created an independent .sql file for the selected period by the SMS Gateway (Figure 3).


SMS Gateway - 192.168.0.113 - Trial version. Click here to purchase a license.
SMS Reporting
Create new SMS Reporter
SMS Reporter details
SQL Txt Reporting
Please setup your SQL Txt connection for reporting
General
SQL
Advanced
Back
Ok
Cancel

Connection name

Please specify a unique name. This can be anything that helps you identify this reporting connection.

Name: SQL_Txt_Reporting_1

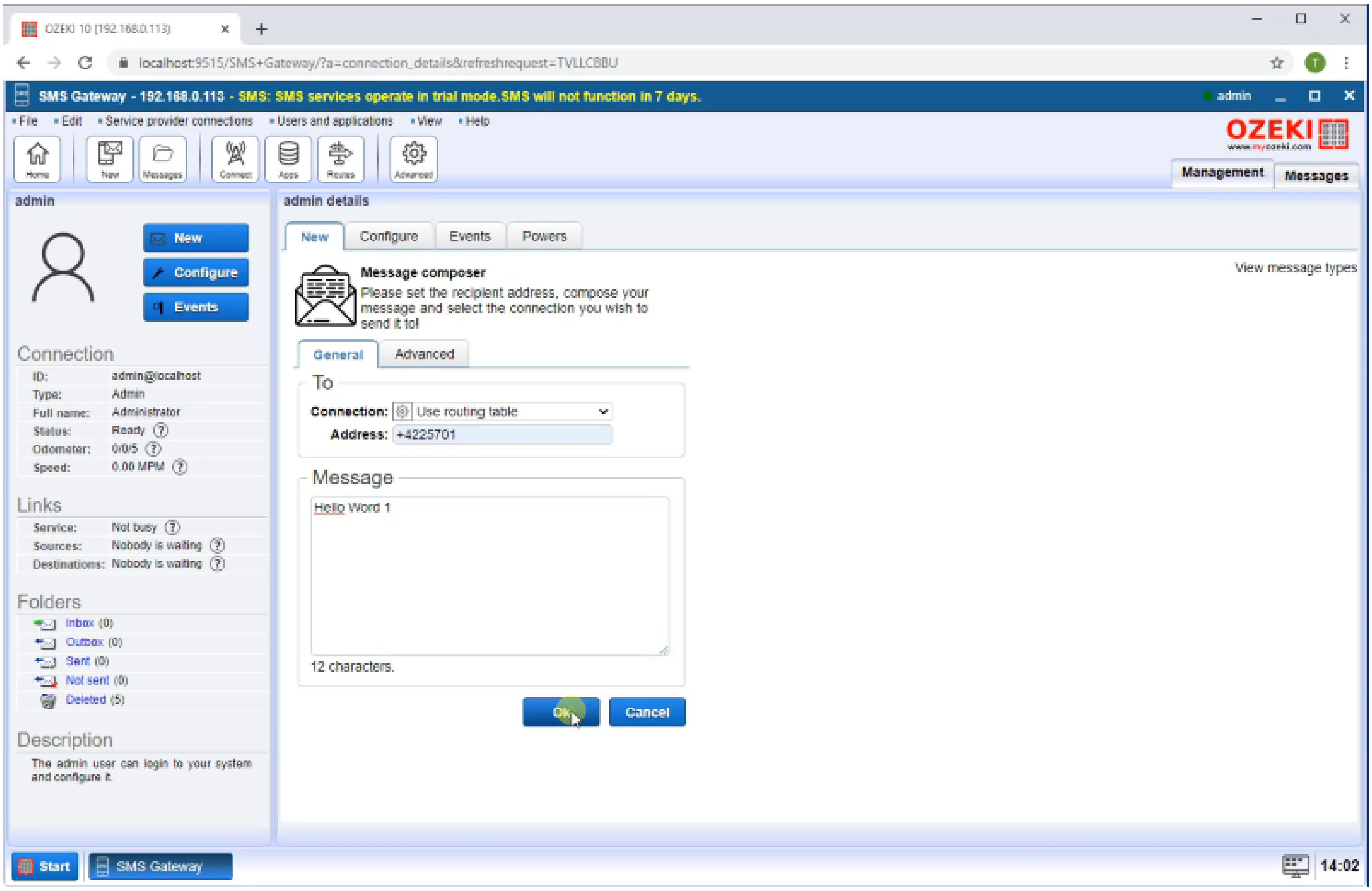
Connection settings

Directory: C:\Ozeki\Reporting_SQL

Period: Day

Figure 3 - Configure SQL Txt reporter

From the Admin user you can send SMS message. Provide the recipient address, the message and click on the OK button (Figure 4).



localhost:9515/SMS+Gateway/?a=connection_details&refreshrequest=TVLLCBBU

SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.

admin

Connection

ID: admin@localhost
Type: Admin
Full name: Administrator
Status: Ready
Odometer: 0/0/5
Speed: 0.00 MPM

Links

Service: Not busy
Sources: Nobody is waiting
Destinations: Nobody is waiting

Folders

Inbox (0)
Outbox (0)
Sent (0)
Not sent (0)
Deleted (5)

Description

The admin user can login to your system and configure it.

admin details

Message composer

Please set the recipient address, compose your message and select the connection you wish to send it to!

To

Connection: Use routing table
Address: +4225701

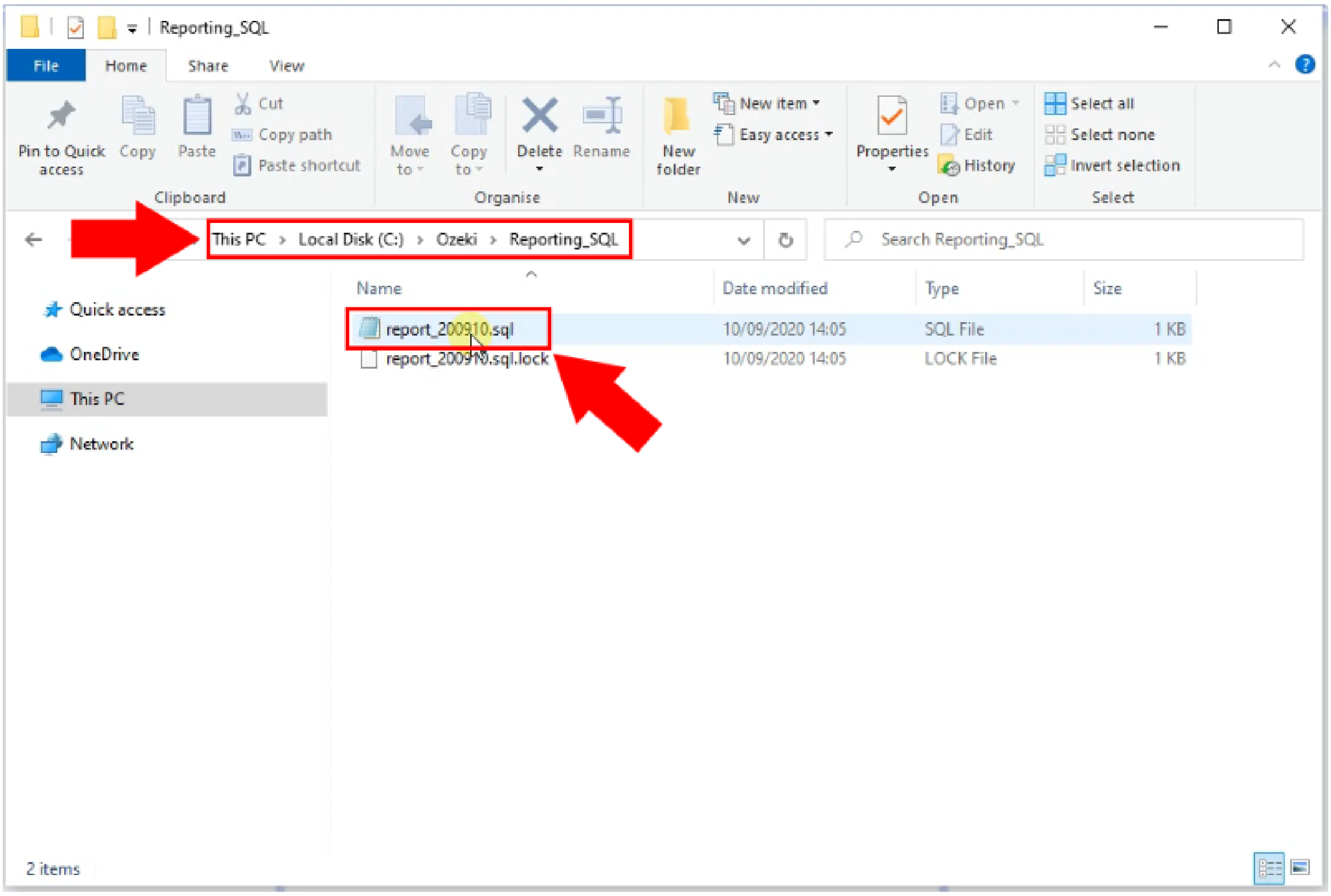
Message

Hello Word 1

12 characters.

Figure 4 - Send Messages

Now in the folder what you set in the SQL Txt reporter details page you can see the .sql text file is created. The default folder is C:\Ozeki (Figure 5).



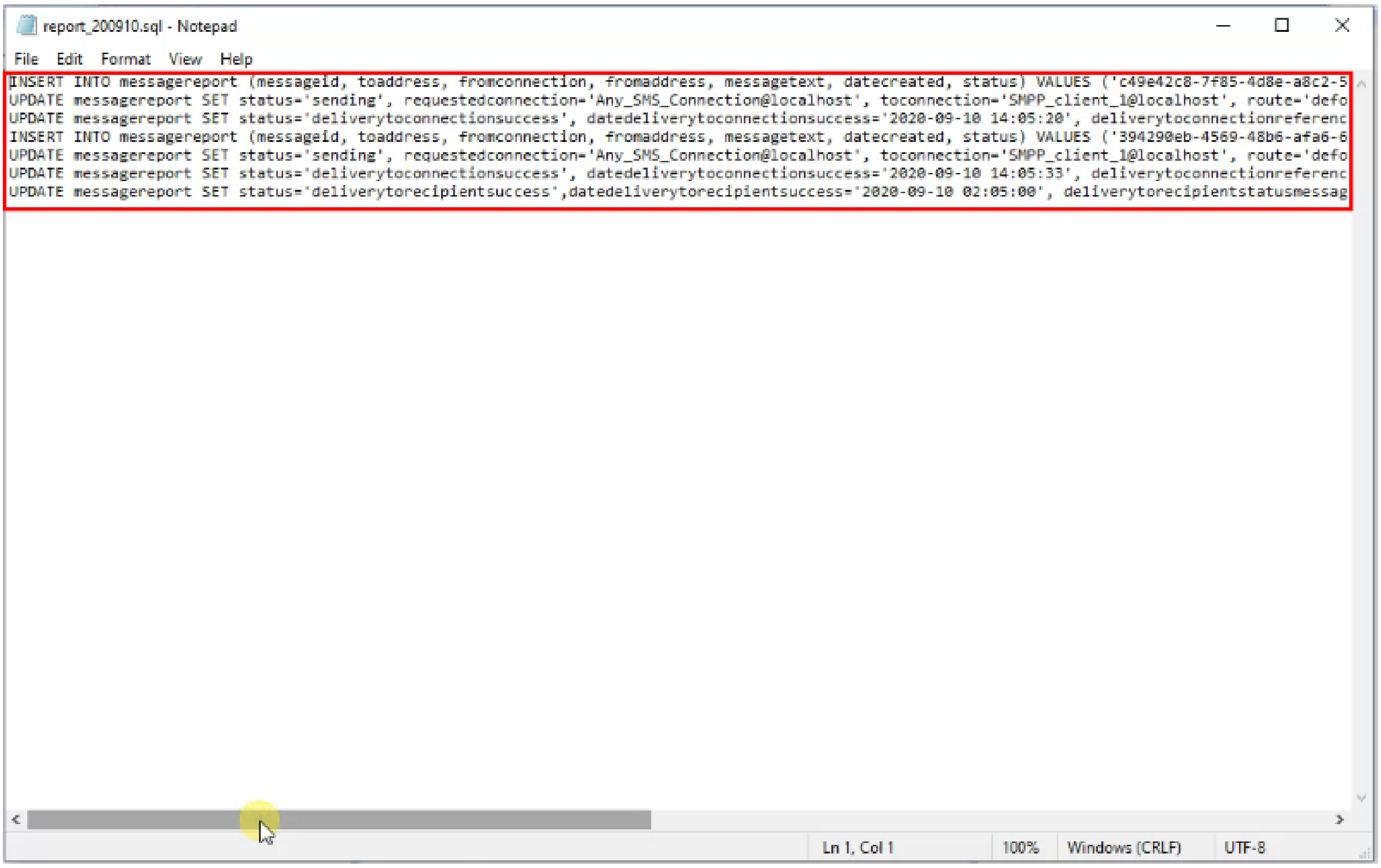
This PC > Local Disk (C:) > Ozeki > Reporting_SQL

| Name | Date modified | Type | Size |
|---|---|---|---|
| report_200910.sql | 10/09/2020 14:05 | SQL File | 1 KB |
| report_200910.sql.lock | 10/09/2020 14:05 | LOCK File | 1 KB |

2 items

Figure 5 - Sql file in folder

Finally if you are open the .sql text file you can see that all the sent and received sms messages are stored as the Figure 6 shows. You can use these .sql files to do batch inserts into your database, to have a log of all messages.



report_200910.sql - Notepad

```
INSERT INTO messagereport (messageid, toaddress, fromconnection, fromaddress, messagetext, datecreated, status) VALUES ('c49e42c8-7f85-4d8e-a8c2-5
UPDATE messagereport SET status='sending', requestedconnection='Any_SMS_Connection@localhost', toconnection='SMPP_client_1@localhost', route='defo
UPDATE messagereport SET status='deliverytoconnectionsuccess', datedeliverytoconnectionsuccess='2020-09-10 14:05:20', deliverytoconnectionreferenc
INSERT INTO messagereport (messageid, toaddress, fromconnection, fromaddress, messagetext, datecreated, status) VALUES ('394290eb-4569-48b6-afa6-6
UPDATE messagereport SET status='sending', requestedconnection='Any_SMS_Connection@localhost', toconnection='SMPP_client_1@localhost', route='defo
UPDATE messagereport SET status='deliverytoconnectionsuccess', datedeliverytoconnectionsuccess='2020-09-10 14:05:33', deliverytoconnectionreferenc
UPDATE messagereport SET status='deliverytorecipientsuccess',datedeliverytorecipientsuccess='2020-09-10 02:05:00', deliverytorecipientstatusmessag
```

Ln 1, Col 1 | 100% | Windows (CRLF) | UTF-8

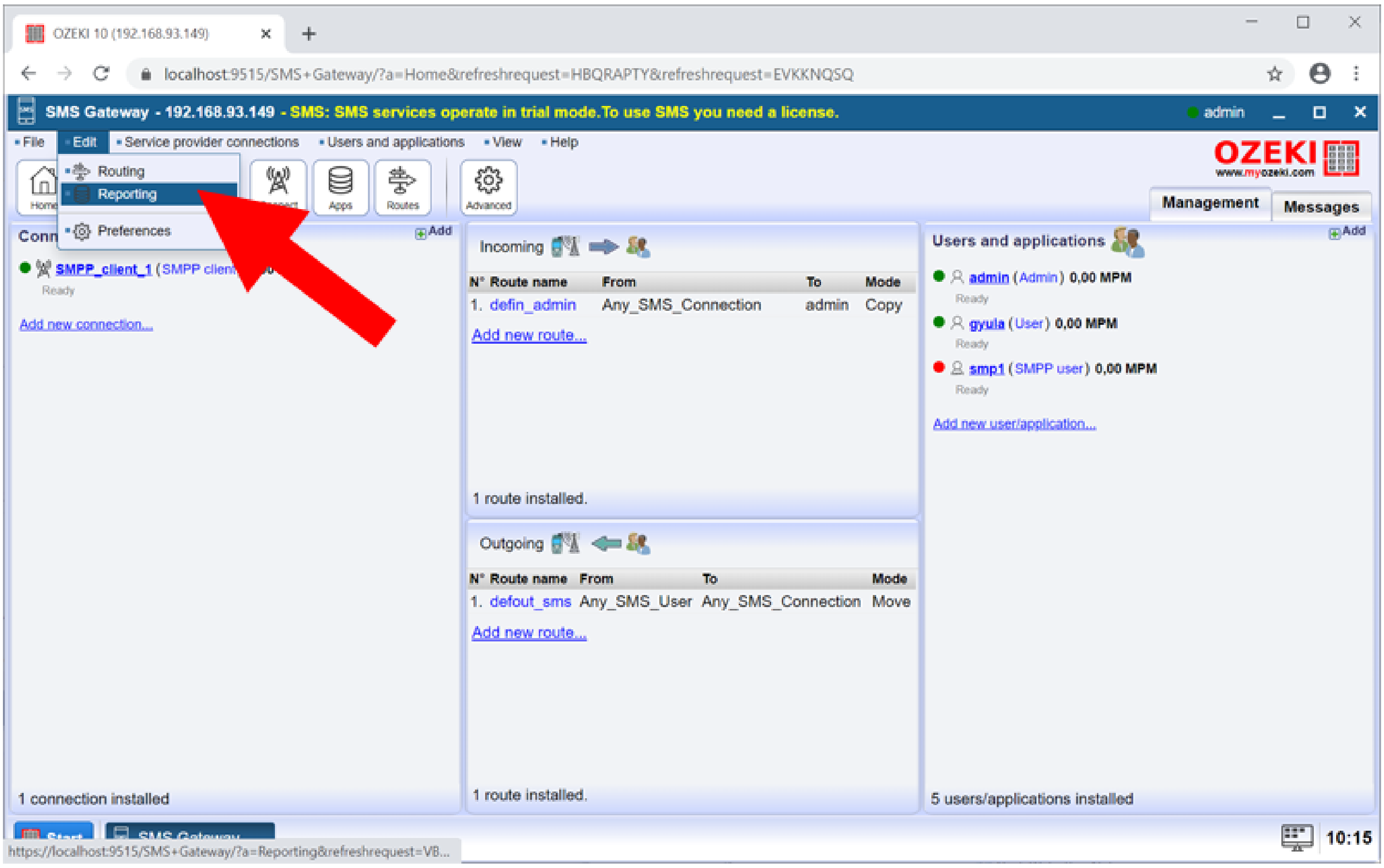
Figure 6 - Sql file content

# How to find out which SQL queries were not executed

***When you use database reporting, to maintain a log of each message going through the system, you might want to check if all SQL commands were executed properly on your database. Ozeki 10's SQL reporting feature provides a user interface to see which records are waiting to be executed, and which ones failed. This document gives you information on how monitor the execution of SQL queries in your reporting system.***

## Step 1 - Open the reporting system

The reporting system can save messages going through the system into different sources, such as databases, txt files, etc. The most popular choice is using an SQL database server, such as Microsoft SQL server of MySQL to save a log of each message. Database servers are great, because they can deal with a great amount of data, which is a must for high capacity systems.


OZEKI 10 (192.168.93.149)
localhost:9515/SMS+Gateway/?a=Home&refreshrequest=HBQRAPTY&refreshrequest=EVKKNQSQ
SMS Gateway - 192.168.93.149 - SMS: SMS services operate in trial mode.To use SMS you need a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Routing
Reporting
Preferences
Home
Apps
Routes
Advanced
OZEKI
www.myozeki.com
Management
Messages
Add
Start
SMS Gateway
10:15
https://localhost:9515/SMS+Gateway/?a=Reporting&refreshrequest=VB...

Conn

SMPP_client_1 (SMPP client
Ready

Add new connection...

1 connection installed

Incoming

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defin_admin | Any_SMS_Connection | admin | Copy |

Add new route...

1 route installed.

Outgoing

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move |

Add new route...

1 route installed.

Users and applications

admin (Admin) 0,00 MPM
Ready

gyula (User) 0,00 MPM
Ready

smp1 (SMPP user) 0,00 MPM
Ready

Add new user/application...

5 users/applications installed

Figure 1 - Open SQL reporting

## Step 2 - Open the details page of the database link

The details page of the SQL database link allows you to configure the database connection, to view events and to check out the SQL commands waiting to be executed and the ones that could not be executed.

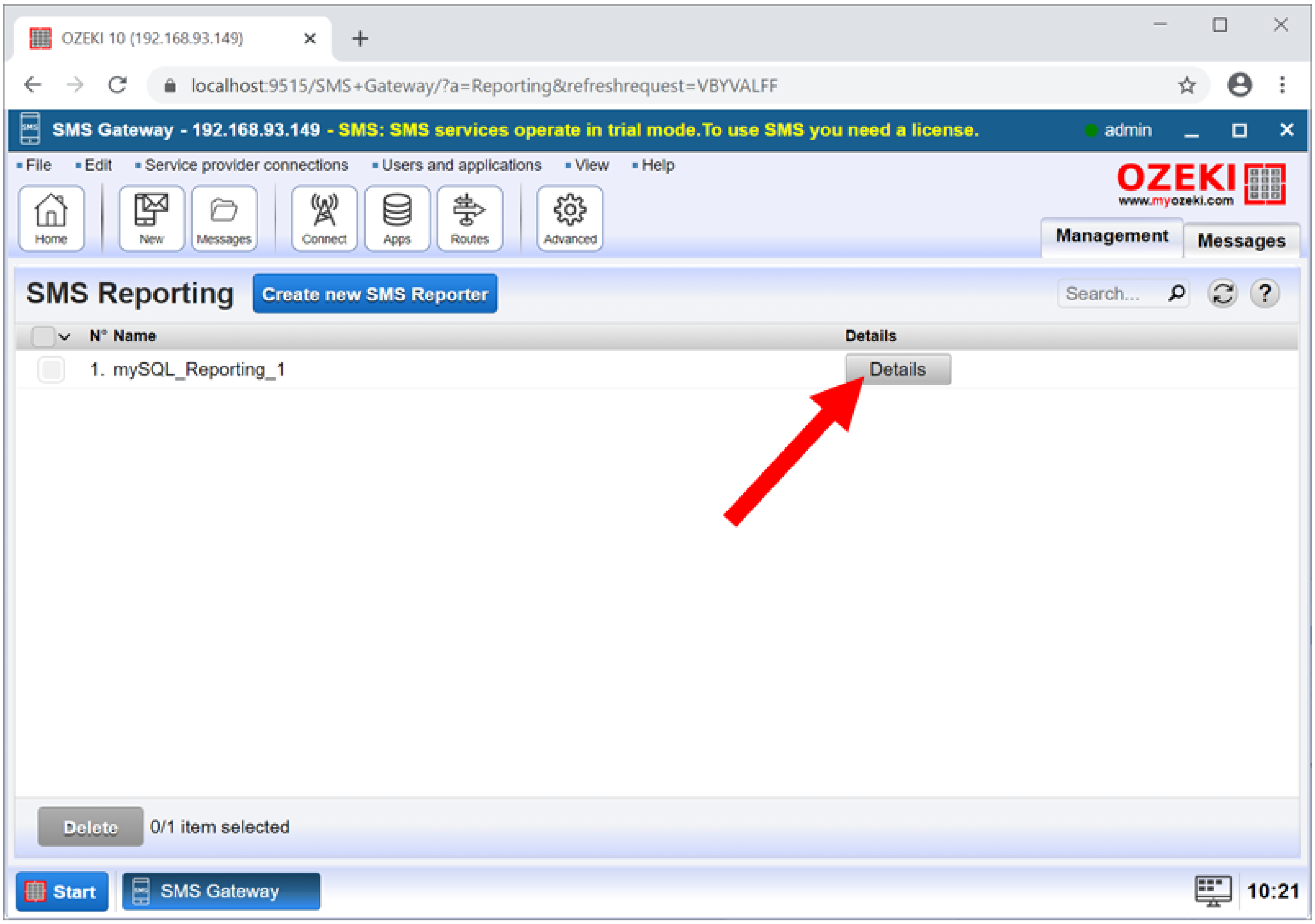
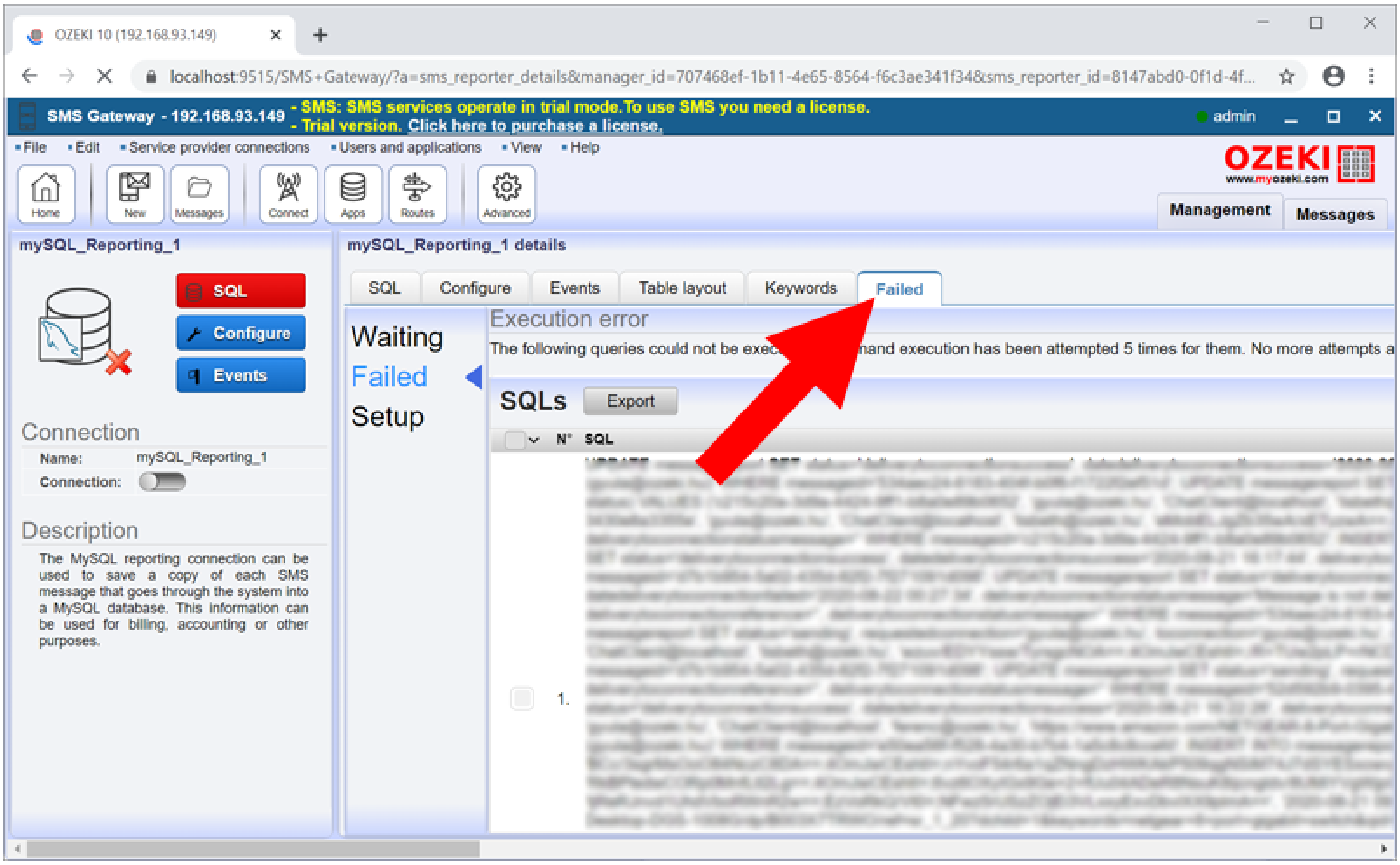


Figure 2 - Click details

# Step 3 - Select the "Failed" tab

The Failed tab is where you can see the list of SQL commands waiting to be executed and the list of commands that were not executed. You might notice, that multiple commands are grouped together. This is called batch processing. Most database servers are not able to handle several hundreds of transactions per second. By grouping multiple INSERT and UPDATE statements into a single SQL request, the performance can be greatly improved. If such a batch request fails the whole request is saved in the FAILED list.



- SMS: SMS services operate in trial mode.To use SMS you need a license.
- Trial version. Click here to purchase a license.

Name: mySQL_Reporting_1

Connection:

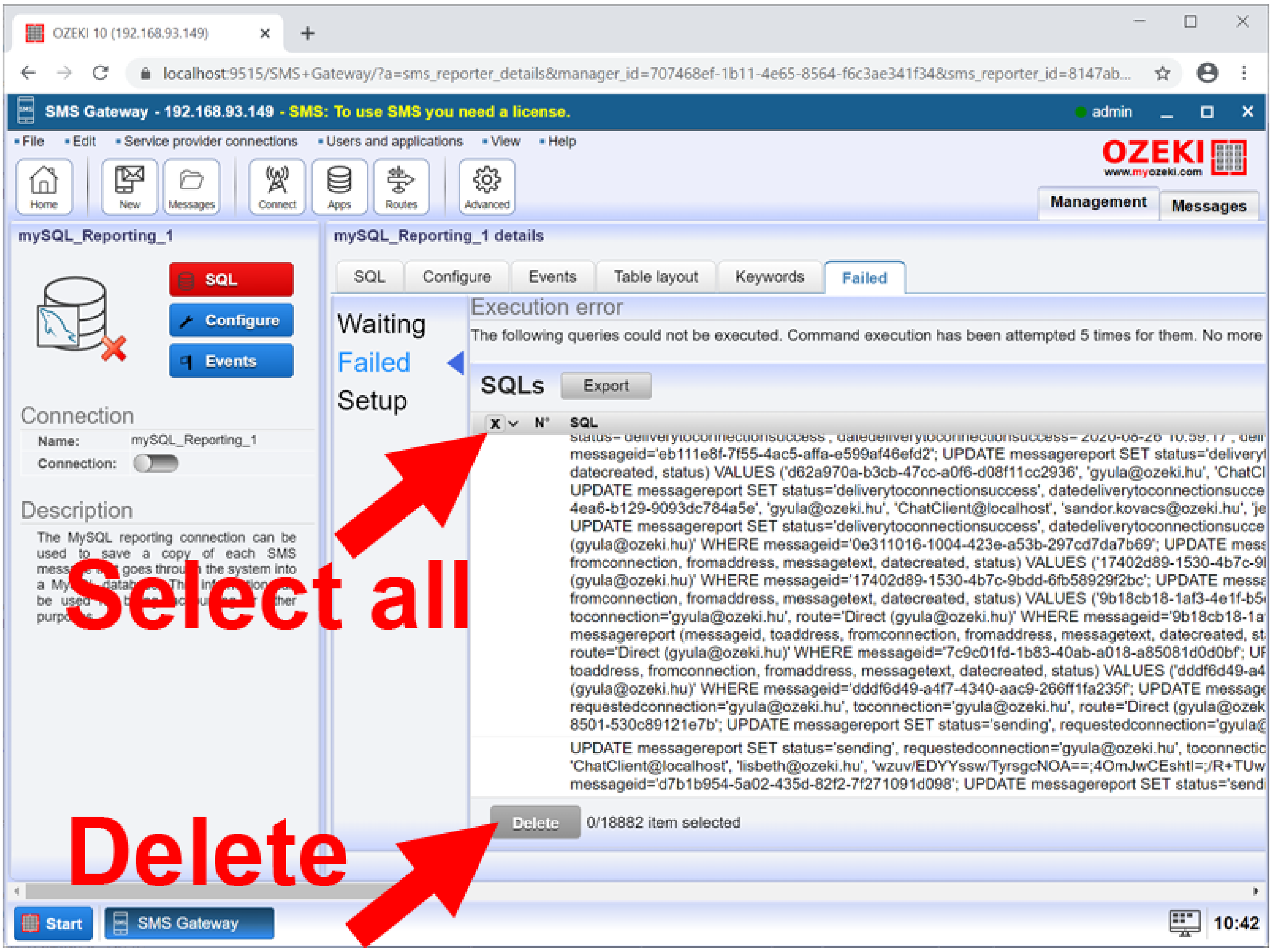
The MySQL reporting connection can be used to save a copy of each SMS message that goes through the system into a MySQL database. This information can be used for billing, accounting or other purposes.

The following queries could not be exec... and execution has been attempted 5 times for them. No more attempts a

Figure 3 - Select the "Failed" tab

## Step 4 - Delete the "Failed" queries

On this page you may export the failed queries into an .sql text file, which you can use for later execution on your database. You may also want to delete these queries. This can be useful, if you made a mistake when you created the original SQL templates and there is no chance for these queries to be executed.



Connection

Name: mySQL_Reporting_1

Connection:

Description

The MySQL reporting connection can be used to save a copy of each SMS message that goes through the system into a MySQL database. This information can be used ... other purposes.

Execution error

The following queries could not be executed. Command execution has been attempted 5 times for them. No more

```
status='deliverytoconnectionsuccess', datedeliverytoconnectionsuccess='2020-08-26 10:59:17', deli
messageid='eb111e8f-7f55-4ac5-affa-e599af46efd2'; UPDATE messagereport SET status='deliveryt
datecreated, status) VALUES ('d62a970a-b3cb-47cc-a0f6-d08f11cc2936', 'gyula@ozeki.hu', 'ChatCl
UPDATE messagereport SET status='deliverytoconnectionsuccess', datedeliverytoconnectionsucce
4ea6-b129-9093dc784a5e', 'gyula@ozeki.hu', 'ChatClient@localhost', 'sandor.kovacs@ozeki.hu', 'je
UPDATE messagereport SET status='deliverytoconnectionsuccess', datedeliverytoconnectionsucce
(gyula@ozeki.hu)' WHERE messageid='0e311016-1004-423e-a53b-297cd7da7b69'; UPDATE mess
fromconnection, fromaddress, messagetext, datecreated, status) VALUES ('17402d89-1530-4b7c-9l
(gyula@ozeki.hu)' WHERE messageid='17402d89-1530-4b7c-9bdd-6fb58929f2bc'; UPDATE messa
fromconnection, fromaddress, messagetext, datecreated, status) VALUES ('9b18cb18-1af3-4e1f-b5
toconnection='gyula@ozeki.hu', route='Direct (gyula@ozeki.hu)' WHERE messageid='9b18cb18-1a
messagereport (messageid, toaddress, fromconnection, fromaddress, messagetext, datecreated, st
route='Direct (gyula@ozeki.hu)' WHERE messageid='7c9c01fd-1b83-40ab-a018-a85081d0d0bf'; UF
toaddress, fromconnection, fromaddress, messagetext, datecreated, status) VALUES ('dddf6d49-a4
(gyula@ozeki.hu)' WHERE messageid='dddf6d49-a4f7-4340-aac9-266ff1fa235f'; UPDATE message
requestedconnection='gyula@ozeki.hu', toconnection='gyula@ozeki.hu', route='Direct (gyula@ozek
8501-530c89121e7b'; UPDATE messagereport SET status='sending', requestedconnection='gyula@

UPDATE messagereport SET status='sending', requestedconnection='gyula@ozeki.hu', toconnectic
'ChatClient@localhost', 'lisbeth@ozeki.hu', 'wzuv/EDYYssw/TyrsgcNOA==;4OmJwCEshtl=;/R+TUw
messageid='d7b1b954-5a02-435d-82f2-7f271091d098'; UPDATE messagereport SET status='sendi
```

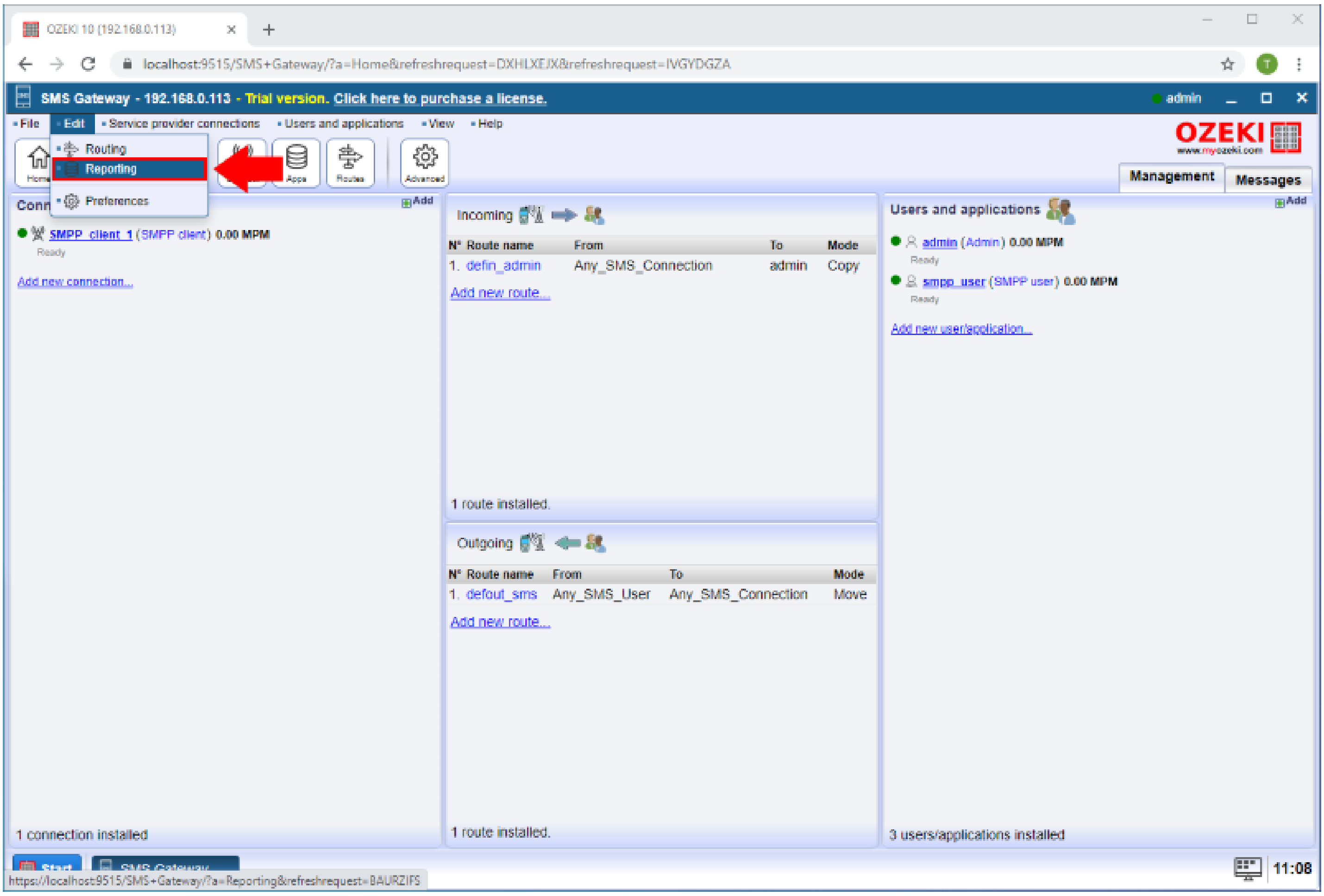
0/18882 item selected

Figure 1 -

# High performance database reporting settings for MS SQL

***This guide explains how to setup a high performance database reporting for Microsoft SQL Database server to save all SMS messages going through the system. This setup makes it possible for you to create reports, write bills or to calculate the cost of operating your SMS gateway system.***

The reporting system can save messages going through the system into different sources, such as databases, txt files, etc. One of the most popular choice is using MS SQL server to save a log of each message. You are able to open reporting system from the Edit menu Reporting option as you can see it on the Figure 1.



OZEKI 10 (192.168.0.113)
localhost:9515/SMS+Gateway/?a=Home&refreshrequest=DXHLXEJX&refreshrequest=IVGYDGZA
SMS Gateway - 192.168.0.113 - Trial version. Click here to purchase a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Routing
Reporting
Preferences
Apps
Routes
Advanced
OZEKI
www.myozeki.com
Management
Messages
Conn
Add
SMPP_client_1 (SMPP client) 0.00 MPM
Ready
Add new connection...
1 connection installed
Incoming
Outgoing
Add new route...
1 route installed.
Users and applications
admin (Admin) 0.00 MPM
Ready
smpp_user (SMPP user) 0.00 MPM
Ready
Add new user/application...
3 users/applications installed
Start
SMS Gateway
https://localhost:9515/SMS+Gateway/?a=Reporting&refreshrequest=BAURZIFS
11:08

Incoming

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defin_admin | Any_SMS_Connection | admin | Copy |

Outgoing

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defaut_sms | Any_SMS_User | Any_SMS_Connection | Move |

Figure 1 - Open Reporting menu

In the Reporting menu click on the Create new SMS Reporter button and than select MS SQL Fast type database (Figure 2).

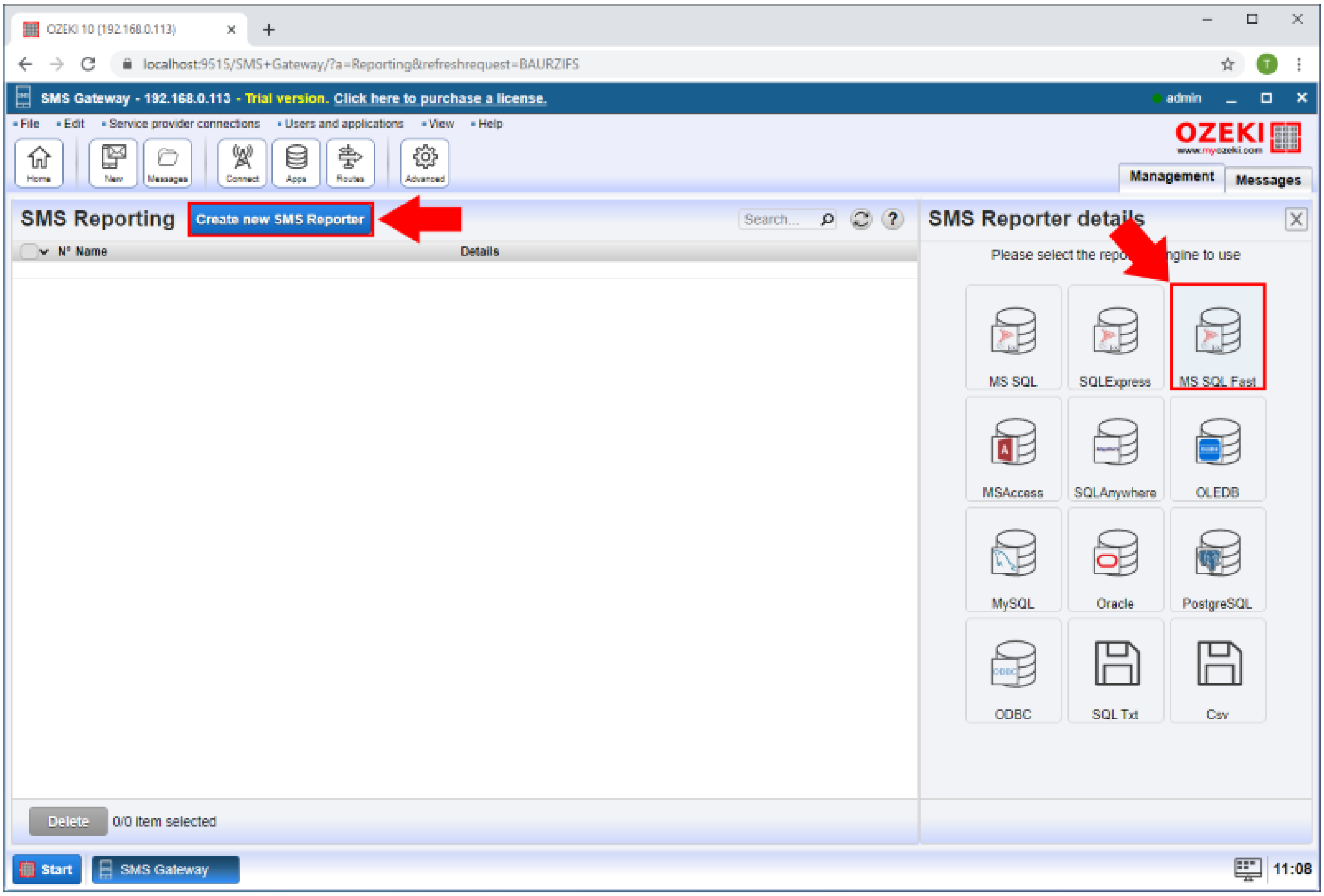
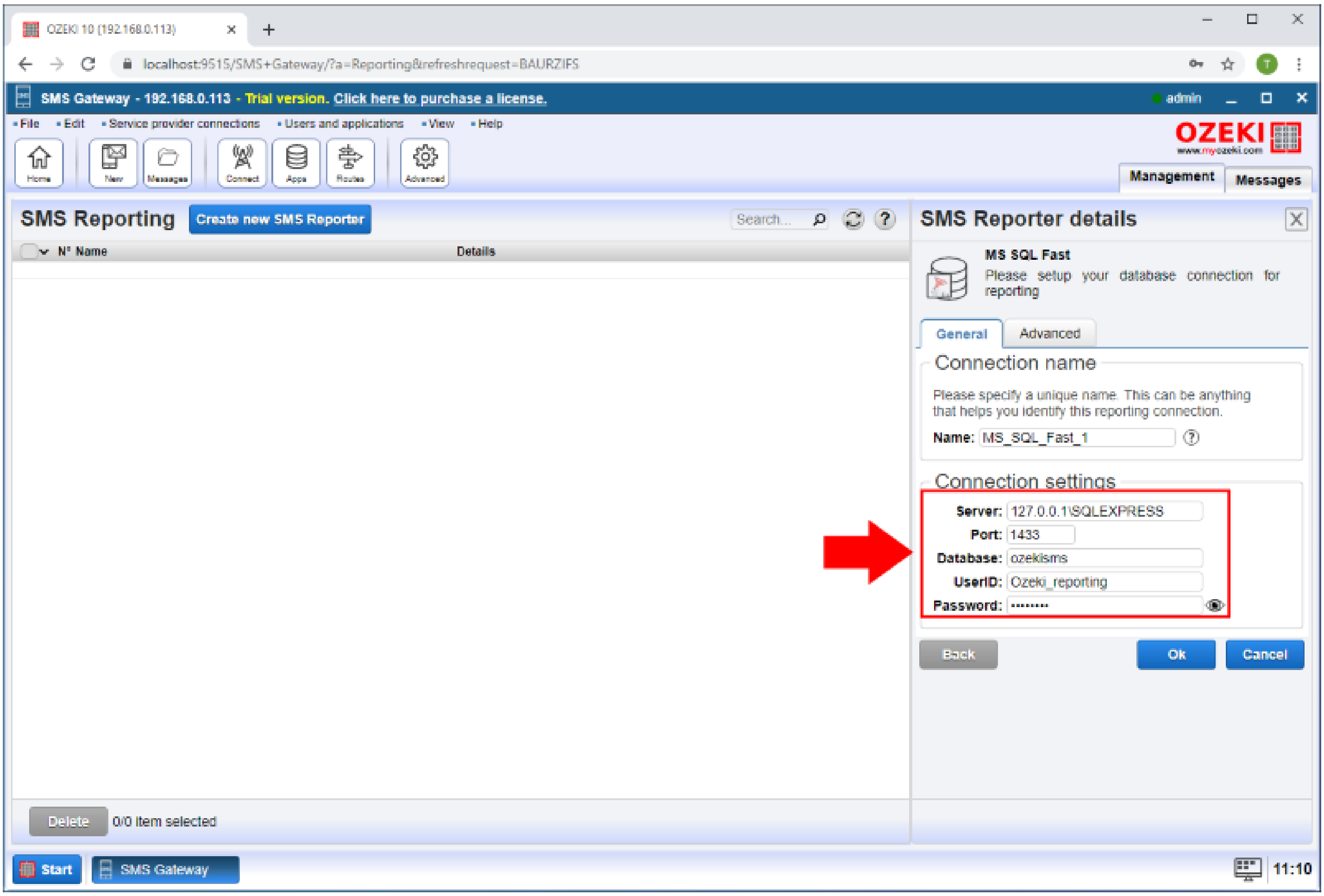


Figure 2 - Create new MS SQL Fast reporter

Now enter the connection details like Server, Port, Database, UserID and Password in the details page of the MS SQL Fast reporter. And enter the Connection name. Finally click on the OK button (Figure 3).



**MS SQL Fast**

Please setup your database connection for reporting

Connection name

Please specify a unique name. This can be anything that helps you identify this reporting connection.

Name: MS_SQL_Fast_1

Connection settings

Server: 127.0.0.1\SQLEXPRESS

Port: 1433

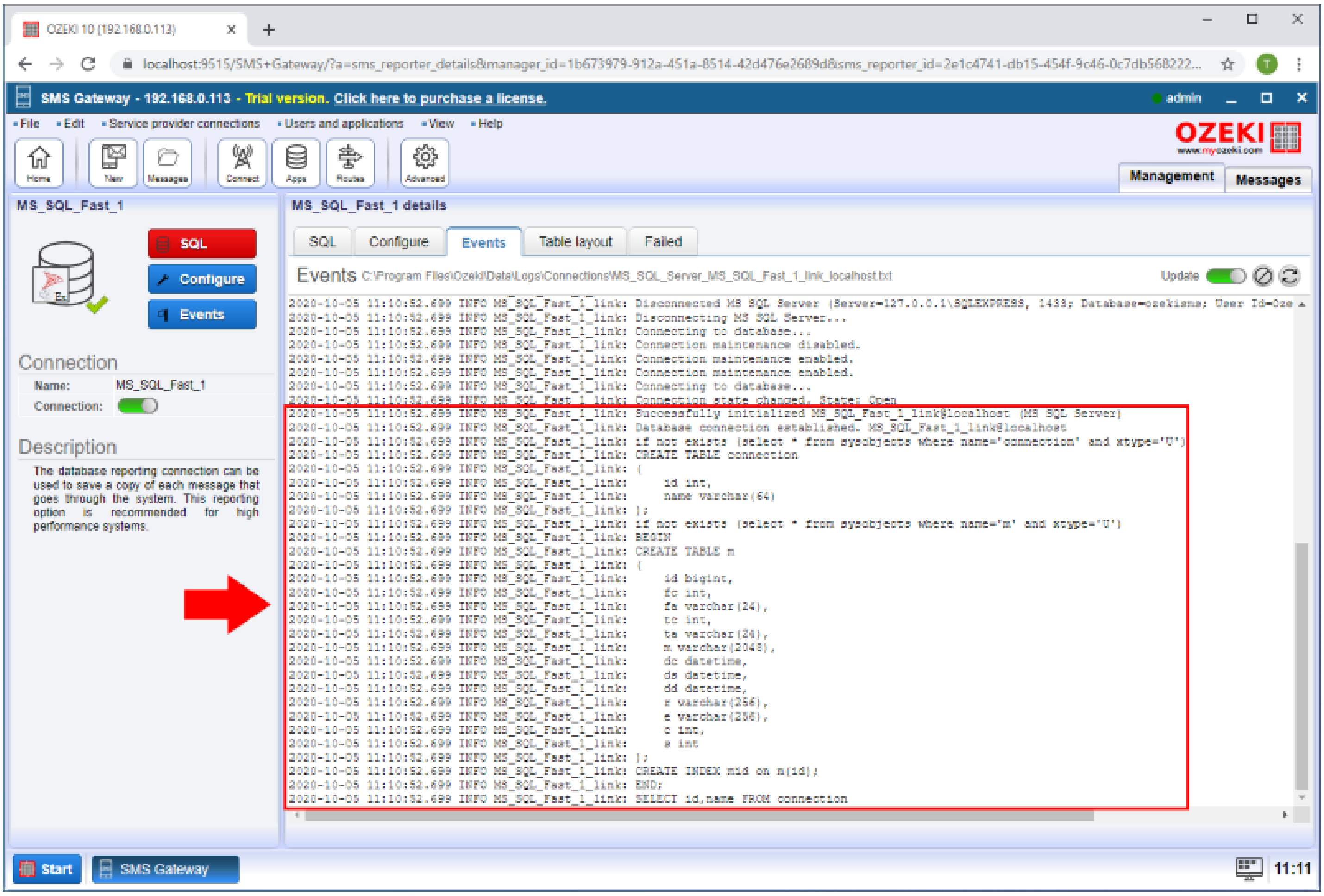
Database: ozekisms

UserID: Ozeki_reporting

Password: ••••••••

Figure 3 - Specify connection details

In the Events tab you can see the logging of the latest server events. As you can see on the Figure 4 the database connection is established.



Events C:\Program Files\Ozeki\Data\Logs\Connections\MS_SQL_Server_MS_SQL_Fast_1_link_localhost.txt

Connection

Name: MS_SQL_Fast_1

Connection:

Description

The database reporting connection can be used to save a copy of each message that goes through the system. This reporting option is recommended for high performance systems.

```
2020-10-05 11:10:52.699 INFO MS_SQL_Fast_1_link: Disconnected MS SQL Server (Server=127.0.0.1\SQLEXPRESS, 1433; Database=ozekisms; User Id=Oze
2020-10-05 11:10:52.699 INFO MS_SQL_Fast_1_link: Disconnecting MS SQL Server...
2020-10-05 11:10:52.699 INFO MS_SQL_Fast_1_link: Connecting to database...
2020-10-05 11:10:52.699 INFO MS_SQL_Fast_1_link: Connection maintenance disabled.
2020-10-05 11:10:52.699 INFO MS_SQL_Fast_1_link: Connection maintenance enabled.
2020-10-05 11:10:52.699 INFO MS_SQL_Fast_1_link: Connection maintenance enabled.
2020-10-05 11:10:52.699 INFO MS_SQL_Fast_1_link: Connecting to database...
2020-10-05 11:10:52.699 INFO MS_SQL_Fast_1_link: Connection state changed. State: Open
2020-10-05 11:10:52.699 INFO MS_SQL_Fast_1_link: Successfully initialized MS_SQL_Fast_1_link@localhost (MS SQL Server)
2020-10-05 11:10:52.699 INFO MS_SQL_Fast_1_link: Database connection established. MS_SQL_Fast_1_link@localhost
2020-10-05 11:10:52.699 INFO MS_SQL_Fast_1_link: if not exists (select * from sysobjects where name='connection' and xtype='U')
2020-10-05 11:10:52.699 INFO MS_SQL_Fast_1_link: CREATE TABLE connection
2020-10-05 11:10:52.699 INFO MS_SQL_Fast_1_link: (
2020-10-05 11:10:52.699 INFO MS_SQL_Fast_1_link:     id int,
2020-10-05 11:10:52.699 INFO MS_SQL_Fast_1_link:     name varchar(64)
2020-10-05 11:10:52.699 INFO MS_SQL_Fast_1_link: );
2020-10-05 11:10:52.699 INFO MS_SQL_Fast_1_link: if not exists (select * from sysobjects where name='m' and xtype='U')
2020-10-05 11:10:52.699 INFO MS_SQL_Fast_1_link: BEGIN
2020-10-05 11:10:52.699 INFO MS_SQL_Fast_1_link: CREATE TABLE m
2020-10-05 11:10:52.699 INFO MS_SQL_Fast_1_link: (
2020-10-05 11:10:52.699 INFO MS_SQL_Fast_1_link:     id bigint,
2020-10-05 11:10:52.699 INFO MS_SQL_Fast_1_link:     fc int,
2020-10-05 11:10:52.699 INFO MS_SQL_Fast_1_link:     fa varchar(24),
2020-10-05 11:10:52.699 INFO MS_SQL_Fast_1_link:     tc int,
2020-10-05 11:10:52.699 INFO MS_SQL_Fast_1_link:     ta varchar(24),
2020-10-05 11:10:52.699 INFO MS_SQL_Fast_1_link:     m varchar(2048),
2020-10-05 11:10:52.699 INFO MS_SQL_Fast_1_link:     dc datetime,
2020-10-05 11:10:52.699 INFO MS_SQL_Fast_1_link:     ds datetime,
2020-10-05 11:10:52.699 INFO MS_SQL_Fast_1_link:     dd datetime,
2020-10-05 11:10:52.699 INFO MS_SQL_Fast_1_link:     r varchar(256),
2020-10-05 11:10:52.699 INFO MS_SQL_Fast_1_link:     e varchar(256),
2020-10-05 11:10:52.699 INFO MS_SQL_Fast_1_link:     c int,
2020-10-05 11:10:52.699 INFO MS_SQL_Fast_1_link:     s int
2020-10-05 11:10:52.699 INFO MS_SQL_Fast_1_link: );
2020-10-05 11:10:52.699 INFO MS_SQL_Fast_1_link: CREATE INDEX mid on m(id);
2020-10-05 11:10:52.699 INFO MS_SQL_Fast_1_link: END;
2020-10-05 11:10:52.699 INFO MS_SQL_Fast_1_link: SELECT id,name FROM connection
```

Figure 4 - Database connection established

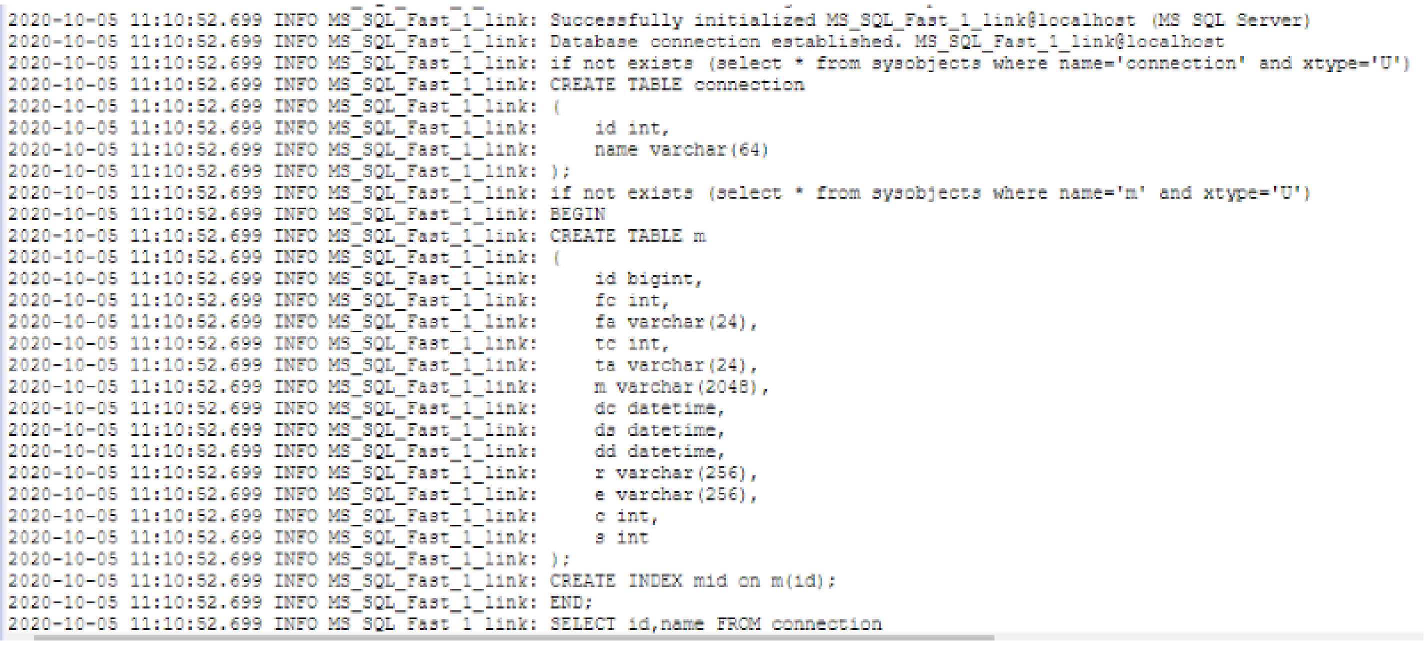
As Figure 5 shows on the events tab, you can see that the SMS Gateway automatically creates the appropriate database tables. Table "m" stores messages and "c" stores contacts.

In the "m" table you can see the below columns:

- fc: from connection
- fa: from address
- tc: to connection
- ta: to address
- m: message
- dc: created date
- ds: submitted date
- dd: delivered date
- r: submit reference
- e: error message
- c: pdu count
- s: status code

Status code shoud be the follow:

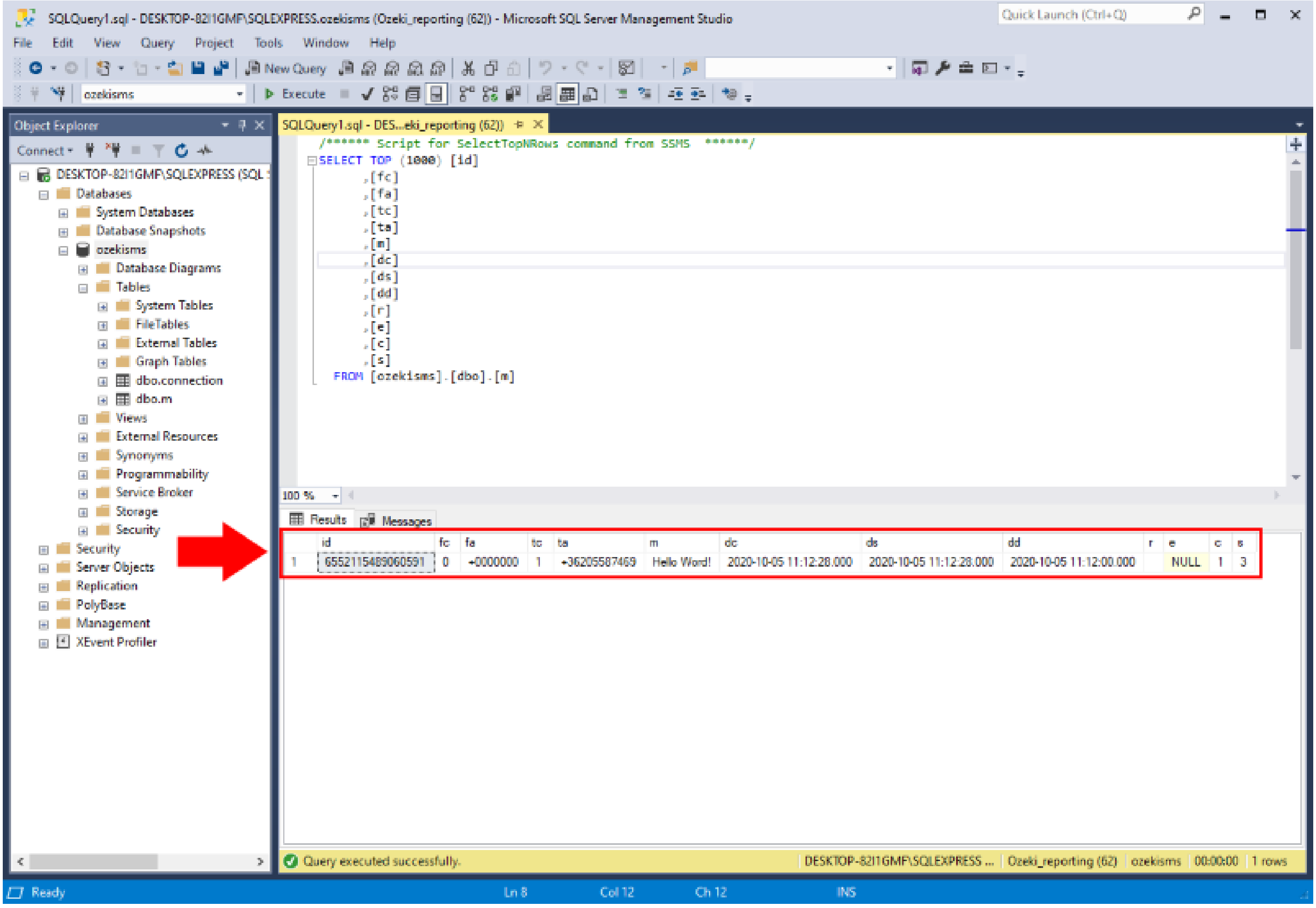
- 1 - submit success
- 2 - submit failed
- 3 - delivered
- 4 - delivery failed



```
2020-10-05 11:10:52.699 INFO MS_SQL_Fast_1_link: Successfully initialized MS_SQL_Fast_1_link@localhost (MS SQL Server)
2020-10-05 11:10:52.699 INFO MS_SQL_Fast_1_link: Database connection established. MS_SQL_Fast_1_link@localhost
2020-10-05 11:10:52.699 INFO MS_SQL_Fast_1_link: if not exists (select * from sysobjects where name='connection' and xtype='U')
2020-10-05 11:10:52.699 INFO MS_SQL_Fast_1_link: CREATE TABLE connection
2020-10-05 11:10:52.699 INFO MS_SQL_Fast_1_link: (
2020-10-05 11:10:52.699 INFO MS_SQL_Fast_1_link:     id int,
2020-10-05 11:10:52.699 INFO MS_SQL_Fast_1_link:     name varchar(64)
2020-10-05 11:10:52.699 INFO MS_SQL_Fast_1_link: );
2020-10-05 11:10:52.699 INFO MS_SQL_Fast_1_link: if not exists (select * from sysobjects where name='m' and xtype='U')
2020-10-05 11:10:52.699 INFO MS_SQL_Fast_1_link: BEGIN
2020-10-05 11:10:52.699 INFO MS_SQL_Fast_1_link: CREATE TABLE m
2020-10-05 11:10:52.699 INFO MS_SQL_Fast_1_link: (
2020-10-05 11:10:52.699 INFO MS_SQL_Fast_1_link:     id bigint,
2020-10-05 11:10:52.699 INFO MS_SQL_Fast_1_link:     fc int,
2020-10-05 11:10:52.699 INFO MS_SQL_Fast_1_link:     fa varchar(24),
2020-10-05 11:10:52.699 INFO MS_SQL_Fast_1_link:     tc int,
2020-10-05 11:10:52.699 INFO MS_SQL_Fast_1_link:     ta varchar(24),
2020-10-05 11:10:52.699 INFO MS_SQL_Fast_1_link:     m varchar(2048),
2020-10-05 11:10:52.699 INFO MS_SQL_Fast_1_link:     dc datetime,
2020-10-05 11:10:52.699 INFO MS_SQL_Fast_1_link:     ds datetime,
2020-10-05 11:10:52.699 INFO MS_SQL_Fast_1_link:     dd datetime,
2020-10-05 11:10:52.699 INFO MS_SQL_Fast_1_link:     r varchar(256),
2020-10-05 11:10:52.699 INFO MS_SQL_Fast_1_link:     e varchar(256),
2020-10-05 11:10:52.699 INFO MS_SQL_Fast_1_link:     c int,
2020-10-05 11:10:52.699 INFO MS_SQL_Fast_1_link:     s int
2020-10-05 11:10:52.699 INFO MS_SQL_Fast_1_link: );
2020-10-05 11:10:52.699 INFO MS_SQL_Fast_1_link: CREATE INDEX mid on m(id);
2020-10-05 11:10:52.699 INFO MS_SQL_Fast_1_link: END;
2020-10-05 11:10:52.699 INFO MS_SQL_Fast_1_link: SELECT id,name FROM connection
```

Figure 5 - Tables created

Finally in the MS SQL Server you will see there is a new message in the table as you can see in the Figure 6.



```
/****** Script for SelectTopNRows command from SSMS  ******/
SELECT TOP (1000) [id]
      ,[fc]
      ,[fa]
      ,[tc]
      ,[ta]
      ,[m]
      ,[dc]
      ,[ds]
      ,[dd]
      ,[r]
      ,[e]
      ,[c]
      ,[s]
  FROM [ozekisms].[dbo].[m]
```

| | id | fc | fa | tc | ta | m | dc | ds | dd | r | e | c | s |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 6552115489060591 | 0 | +0000000 | 1 | +36205587469 | Hello Word! | 2020-10-05 11:12:28.000 | 2020-10-05 11:12:28.000 | 2020-10-05 11:12:00.000 | | NULL | 1 | 3 |

Figure 6 - Message inserted into datbase

# Powershell script

***The following list on this page is about to show you the choices available for you to store SMS reports in a database. This solution is quite useful in case you enabled SQL Text Logging in SMS Gateway. By executing a Power shell script or setting up the Task Scheduler you don't have to insert the SMS reports one by one, these solutions insert the reports altogether and save a lot of time for you.***

**MSSQL Script**

If you follow the link to this document, you can learn about how to execute an MSSQL script on your computer to be able to insert the SMS reports into a database. This solution is great if you enabled SQL text logging for high performance configurations, so you can use this script to execute the SQL queries and insert the reports into your database.

Learn about how you can use an MSSQL script to store SMS reports

**MySQL Script**

The reports of the incoming and outgoing SMS messages can be stored in a single database. This operation can be executed with a MySQL script that you can find if you follow this link to the document. You can learn about how to save the script to your computer, what file extension you need to use, and how to execute the script using the Power Shell.

Check how you can execute MySQL script to store your SMS reports

**Task Scheduler**

Check the following page and see, how you can set up the Windows Task Scheduler to be able to execute a Power Shell script in a specified time period. This solution helps you to insert the newly generated SMS reports into your database which contains the reports of the incoming and outgoing SMS messages and keep this database up to date.

See how to set up Task Scheduler to run a Power Shell script.

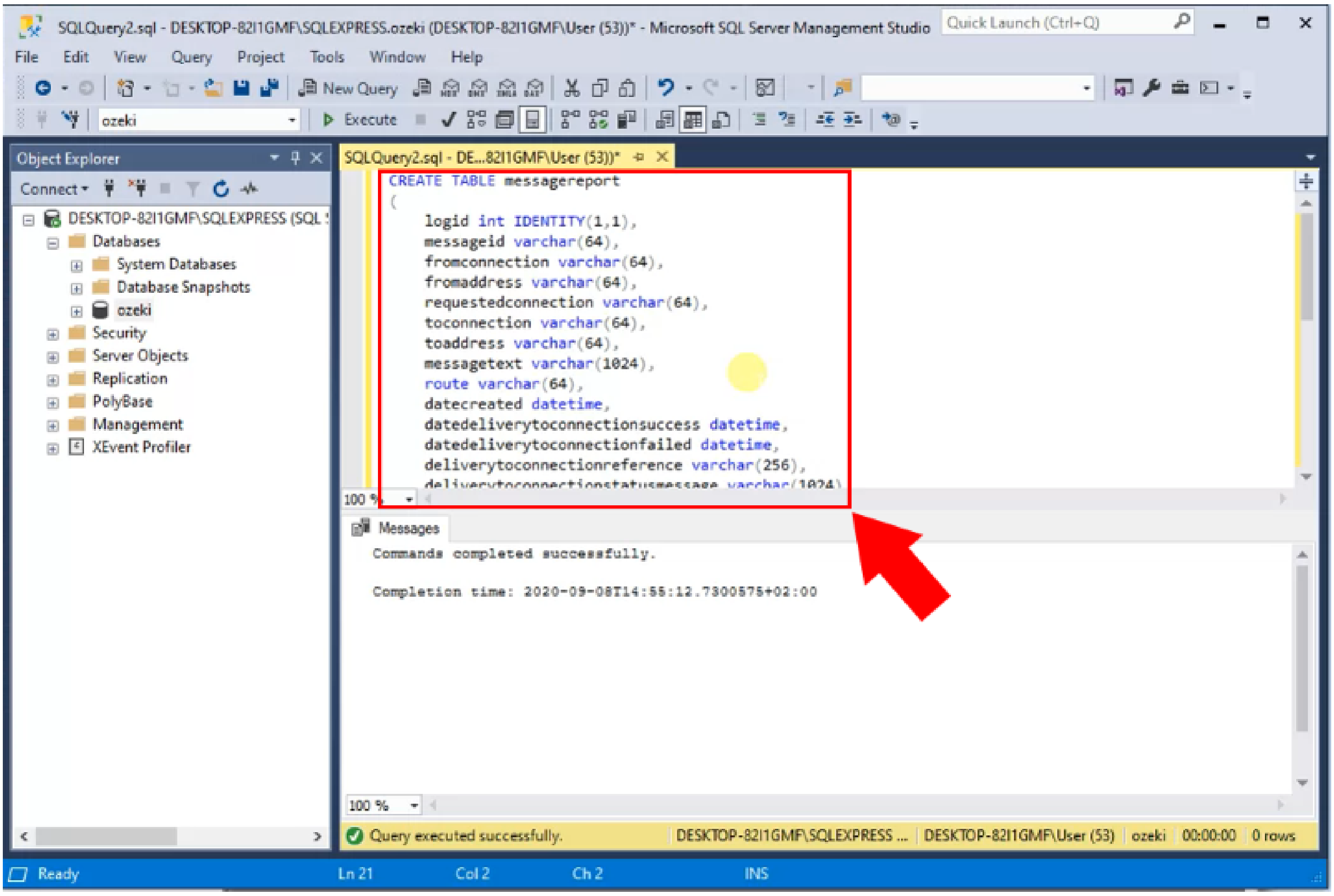
# Powershell script MSSQL

***This script can be used to automatically import sql files older then 20 minutes into the a MSSQL database. It is helpful if you enable SQL Text logging for high performance configurations. In order to execute this script you have to save it with the .ps1 extension, and you must enable power shell script execution for windows in the group policy editor (or you must execute the script as Administrator).***

## Create the following MSSQL table

```
CREATE TABLE messagereport
(
    logid int IDENTITY(1,1),
    messageid varchar(64),
    fromconnection varchar(64),
    fromaddress varchar(64),
    requestedconnection varchar(64),
    toconnection varchar(64),
    toaddress varchar(64),
    messagetext varchar(1024),
    route varchar(64),
    datecreated datetime,
    datedeliverytoconnectionsuccess datetime,
    datedeliverytoconnectionfailed datetime,
    deliverytoconnectionreference varchar(256),
    deliverytoconnectionstatusmessage varchar(1024),
    datedeliverytorecipientsuccess datetime,
    datedeliverytorecipientfailed datetime,
    deliverytorecipientstatusmessage varchar(1024),
    status varchar(64)
)
```

The first step is to create the messagereport table in the database. All outgoing and incoming messages will be stored in this table and updated according to their status. If you have already created this table you can skip this step. Copy the create table script above and execute it in your database as you can see it on the Figure 1.



```
CREATE TABLE messagereport
(
    logid int IDENTITY(1,1),
    messageid varchar(64),
    fromconnection varchar(64),
    fromaddress varchar(64),
    requestedconnection varchar(64),
    toconnection varchar(64),
    toaddress varchar(64),
    messagetext varchar(1024),
    route varchar(64),
    datecreated datetime,
    datedeliverytoconnectionsuccess datetime,
    datedeliverytoconnectionfailed datetime,
    deliverytoconnectionreference varchar(256),
    deliverytoconnectionstatusmessage varchar(1024)
```

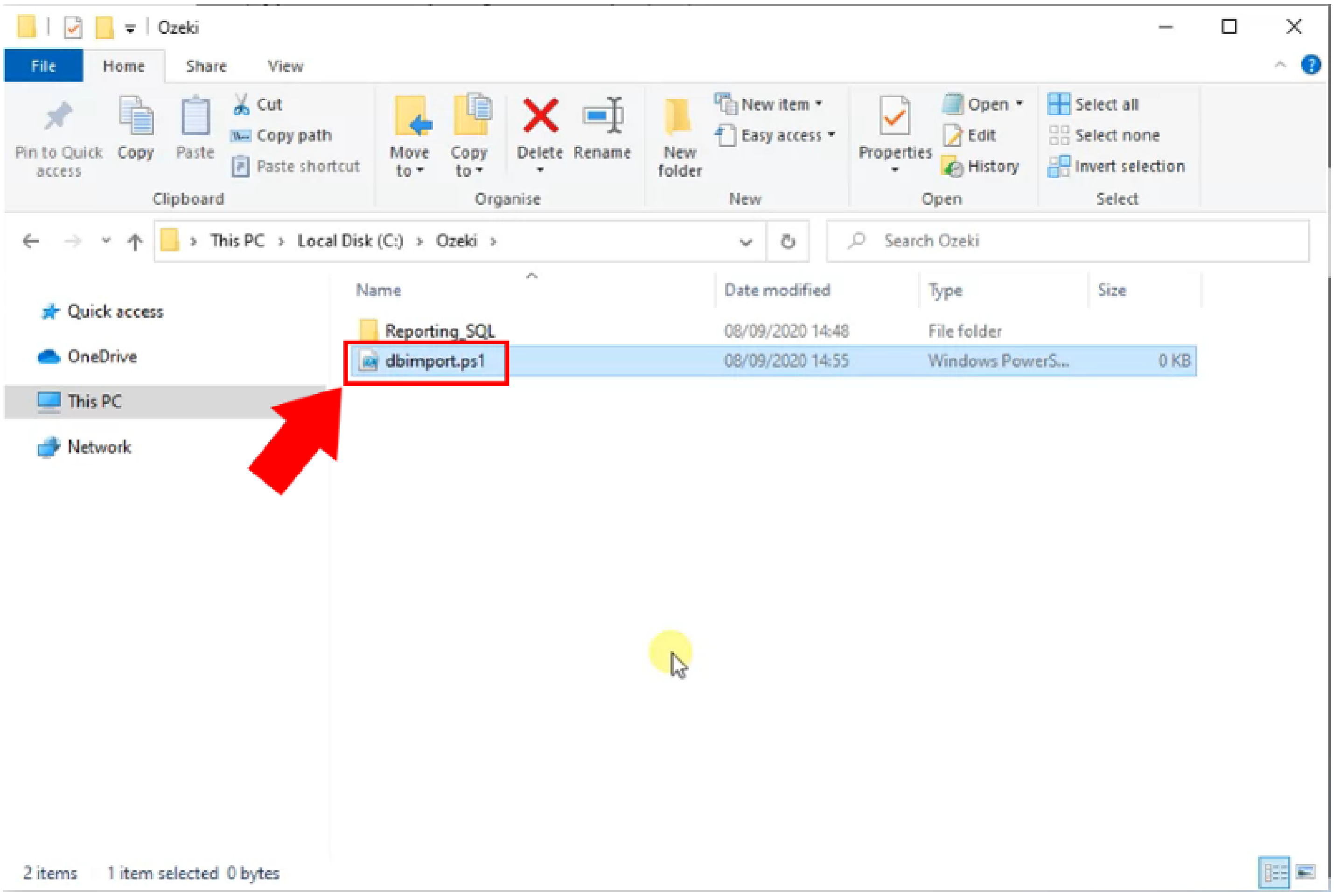
```
Commands completed successfully.

Completion time: 2020-09-08T14:55:12.7300575+02:00
```

Query executed successfully. | DESKTOP-82I1GMF\SQLEXPRESS ... | DESKTOP-82I1GMF\User (53) | ozeki | 00:00:00 | 0 rows

Figure 1 - Create messagereport table

# Save the following powershell script as "dbimport.ps1"

The next step is to create the "dbimport.ps1" file in the C:\Ozeki folder as shown in Figure 2. This file will contain the script that periodically runs SQL queries in the database.



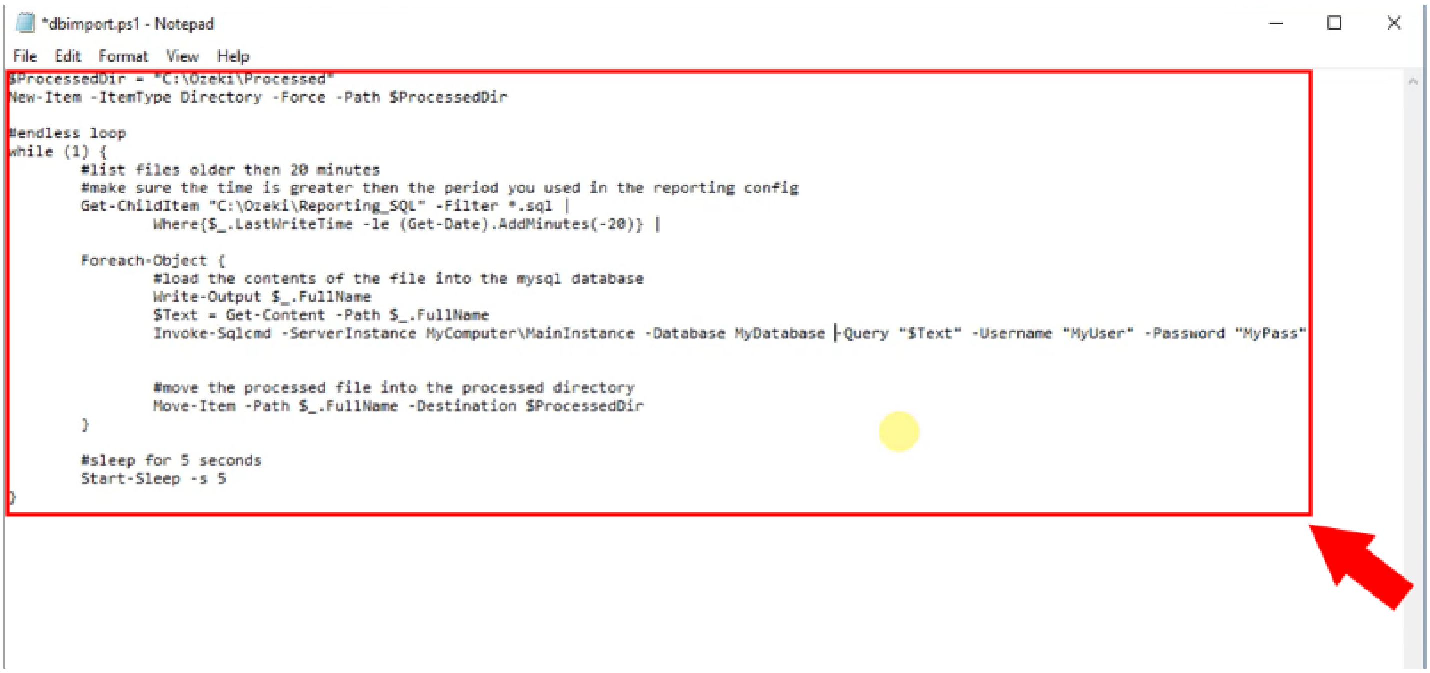
| Name | Date modified | Type | Size |
|---|---|---|---|
| Reporting_SQL | 08/09/2020 14:48 | File folder | |
| dbimport.ps1 | 08/09/2020 14:55 | Windows PowerS... | 0 KB |

2 items 1 item selected 0 bytes

Figure 2 - Create the script file

```
$ProcessedDir = "C:\Ozeki\Processed"
New-Item -ItemType Directory -Force -Path $ProcessedDir

#endless loop
while (1) {
#list files older then 20 minutes
#make sure the time is greater then the period you used in the reporting config
Get-ChildItem "C:\Ozeki\Reporting_SQL" -Filter *.sql |
Where{$_.LastWriteTime -le (Get-Date).AddMinutes(-20)} |

Foreach-Object {
#load the contents of the file into the mssql database
Write-Output $_.FullName
$Text = Get-Content -Path $_.FullName
Invoke-Sqlcmd -ServerInstance MyComputer\MainInstance -Database MyDatabase
-Query "$Text" -Username "MyUser" -Password "MyPass"


#move the processed file into the processed directory
Move-Item -Path $_.FullName -Destination $ProcessedDir
}

#sleep for 5 seconds
Start-Sleep -s 5
}
```

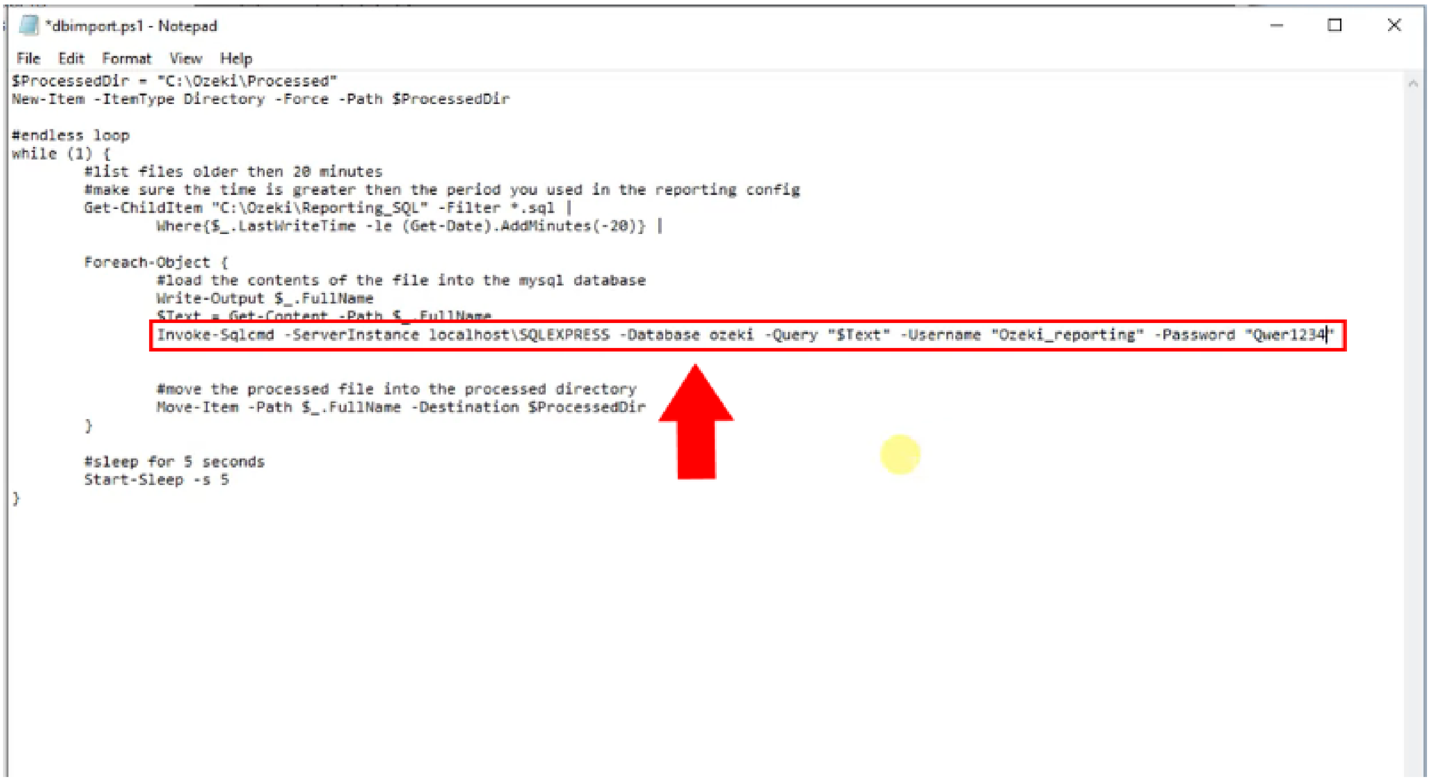
Copy the power shell script above and paste it in the "dbimport.ps1" file as you can see it on the Figure 3.



```
$ProcessedDir = "C:\Ozeki\Processed"
New-Item -ItemType Directory -Force -Path $ProcessedDir

#endless loop
while (1) {
	#list files older then 20 minutes
	#make sure the time is greater then the period you used in the reporting config
	Get-ChildItem "C:\Ozeki\Reporting_SQL" -Filter *.sql |
		Where{$_.LastWriteTime -le (Get-Date).AddMinutes(-20)} |

	Foreach-Object {
		#load the contents of the file into the mysql database
		Write-Output $_.FullName
		$Text = Get-Content -Path $_.FullName
		Invoke-Sqlcmd -ServerInstance MyComputer\MainInstance -Database MyDatabase -Query "$Text" -Username "MyUser" -Password "MyPass"


		#move the processed file into the processed directory
		Move-Item -Path $_.FullName -Destination $ProcessedDir
	}

	#sleep for 5 seconds
	Start-Sleep -s 5
}
```

Figure 3 - Paste the script into the file

Finally, modify the script according to your own MSSQL database. In the Server instance, define the computer and instance name of the MSSQL database. Enter the user name and password you want to use, and replace "Mydatabase" with the name of your database (Figure 4).



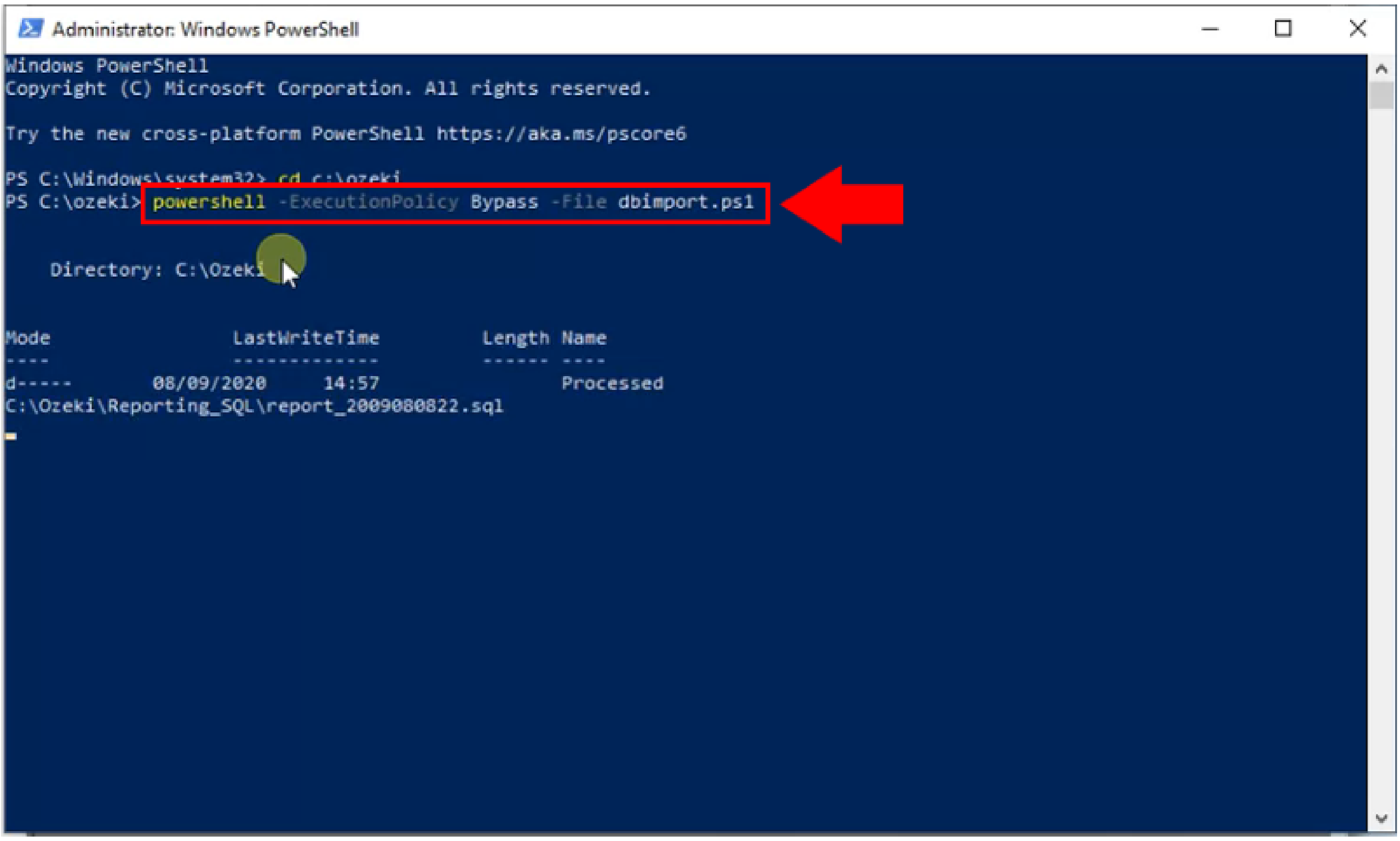
```
$ProcessedDir = "C:\Ozeki\Processed"
New-Item -ItemType Directory -Force -Path $ProcessedDir

#endless loop
while (1) {
	#list files older then 20 minutes
	#make sure the time is greater then the period you used in the reporting config
	Get-ChildItem "C:\Ozeki\Reporting_SQL" -Filter *.sql |
		Where{$_.LastWriteTime -le (Get-Date).AddMinutes(-20)} |

	Foreach-Object {
		#load the contents of the file into the mysql database
		Write-Output $_.FullName
		$Text = Get-Content -Path $_.FullName
		Invoke-Sqlcmd -ServerInstance localhost\SQLEXPRESS -Database ozeki -Query "$Text" -Username "Ozeki_reporting" -Password "Qwer1234"


		#move the processed file into the processed directory
		Move-Item -Path $_.FullName -Destination $ProcessedDir
	}

	#sleep for 5 seconds
	Start-Sleep -s 5
}
```

Figure 4 - Change MySQL command

# Execute the script as administrator

```
powershell -ExecutionPolicy Bypass -File dbimport.ps1
```

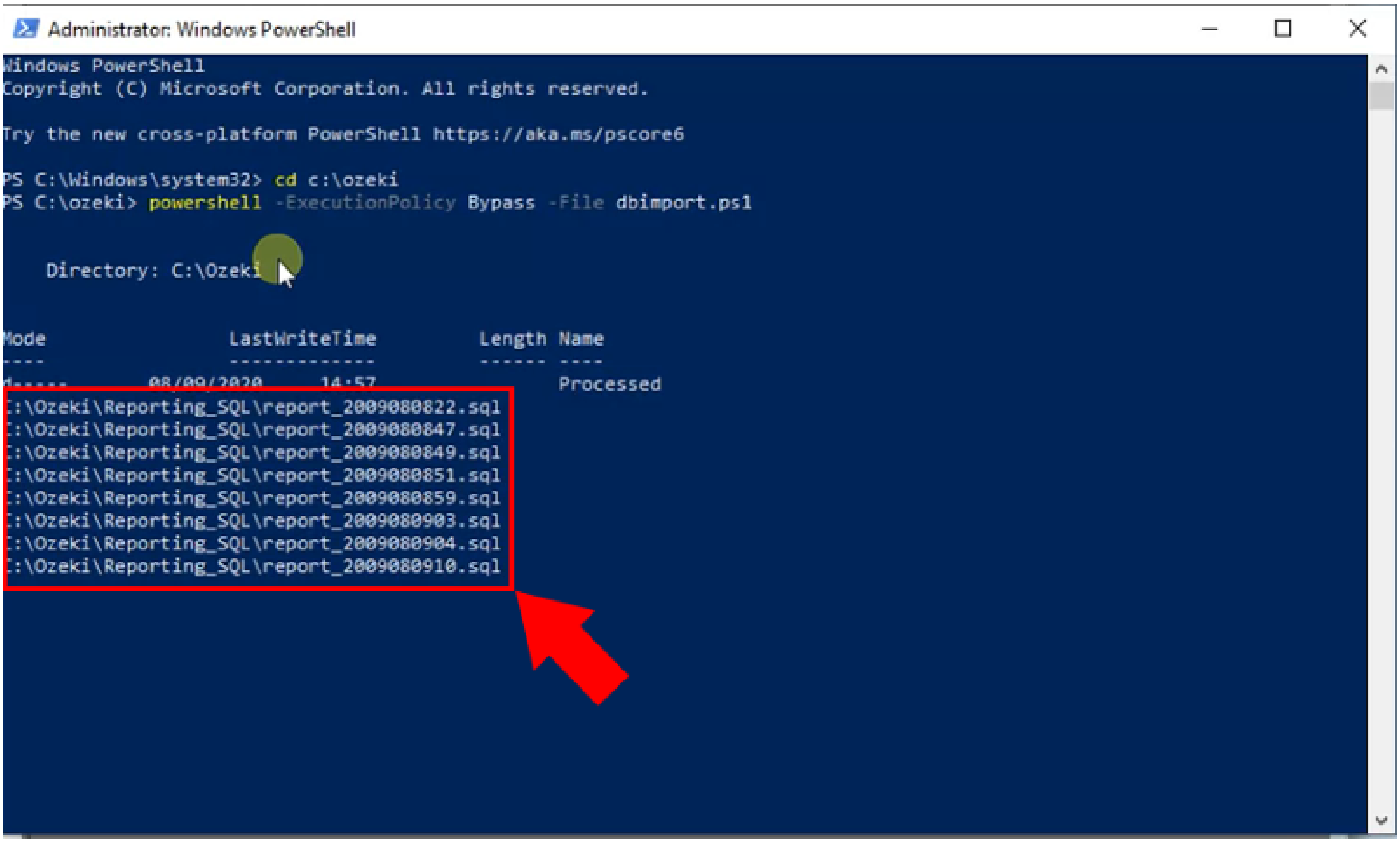
Now execute the script in the power shell by running the above command as you can see it in the Figure 5.



```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Windows\system32> cd c:\ozeki
PS C:\ozeki> powershell -ExecutionPolicy Bypass -File dbimport.ps1


    Directory: C:\Ozeki


Mode                LastWriteTime         Length Name
----                -------------         ------ ----
d-----       08/09/2020     14:57                Processed
C:\Ozeki\Reporting_SQL\report_2009080822.sql
```

Figure 5 - Run powershell script

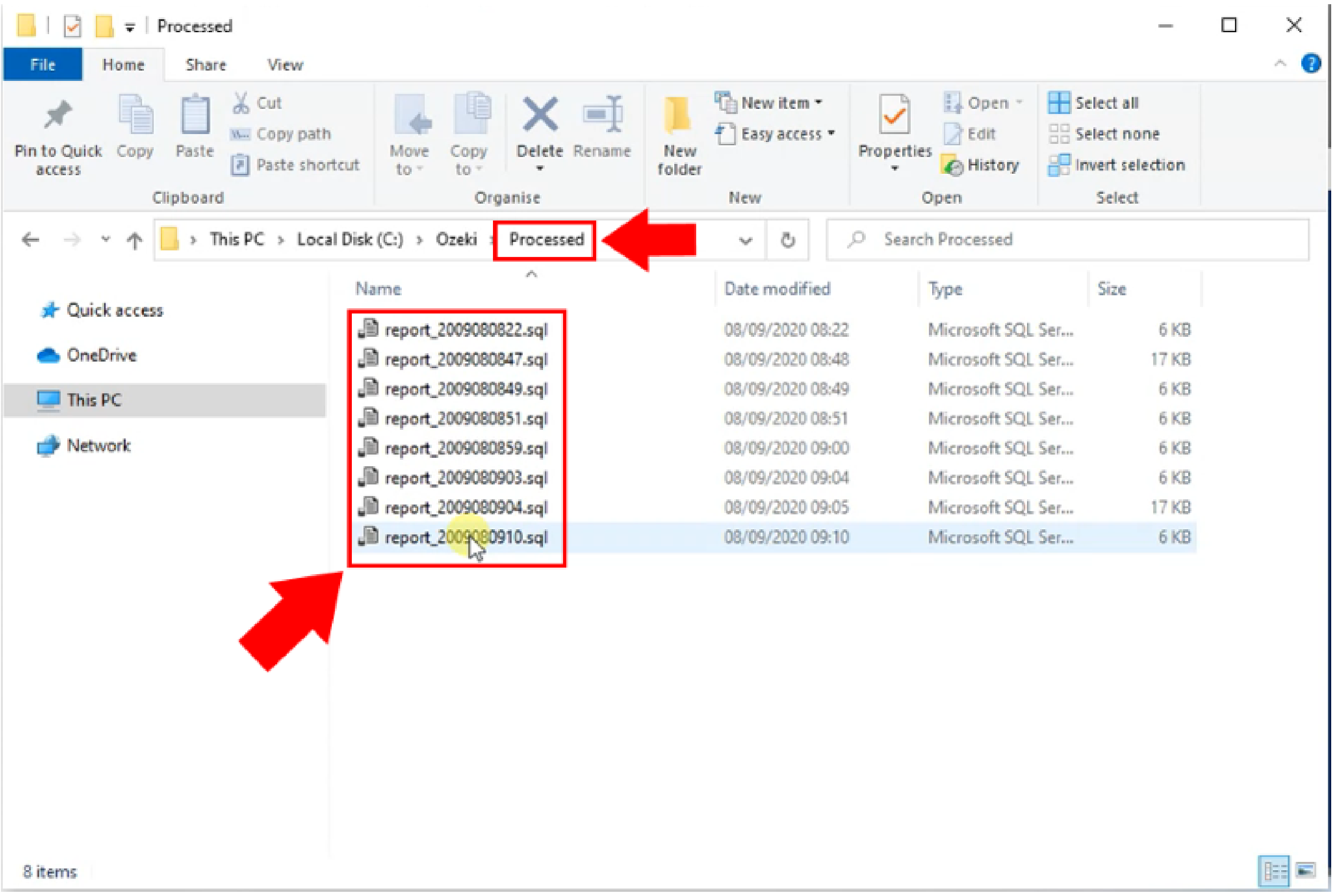
You can see that the script execute all the SQL files what is older than 20 minutes in the 'C:\Ozeki\Reporting_SQL' folder one by one (Figure 6).



```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Windows\system32> cd c:\ozeki
PS C:\ozeki> powershell -ExecutionPolicy Bypass -File dbimport.ps1


    Directory: C:\Ozeki


Mode                LastWriteTime         Length Name
----                -------------         ------ ----
d-----       08/09/2020     14:57                Processed
C:\Ozeki\Reporting_SQL\report_2009080822.sql
C:\Ozeki\Reporting_SQL\report_2009080847.sql
C:\Ozeki\Reporting_SQL\report_2009080849.sql
C:\Ozeki\Reporting_SQL\report_2009080851.sql
C:\Ozeki\Reporting_SQL\report_2009080859.sql
C:\Ozeki\Reporting_SQL\report_2009080903.sql
C:\Ozeki\Reporting_SQL\report_2009080904.sql
C:\Ozeki\Reporting_SQL\report_2009080910.sql
```

Figure 6 - Script execute SQL queries

Every SQL file that the script has run is moded to the Processed folder so that no queries are lost (Figure 7). From here, you can delete them manually if you no longer need them.



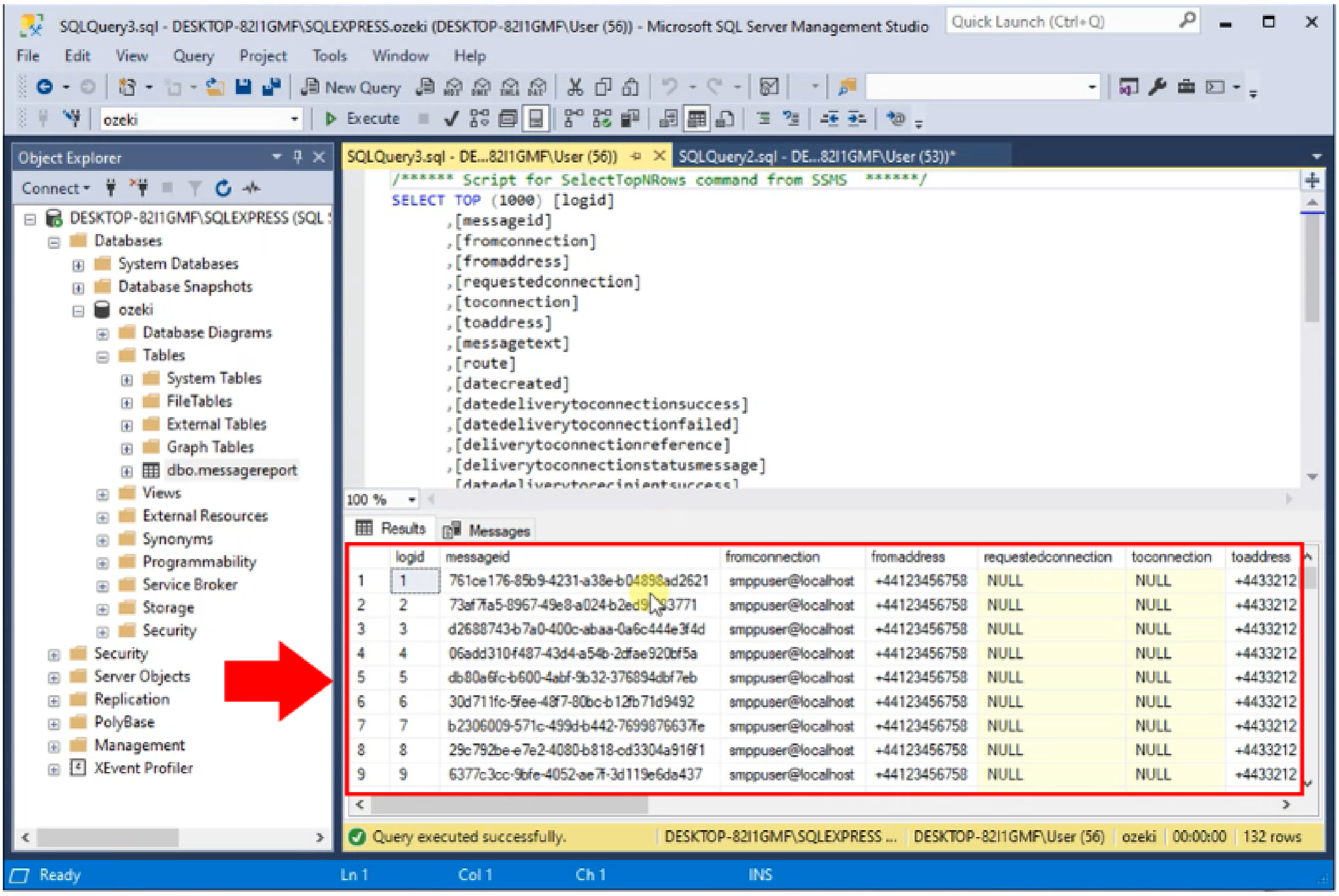
This PC > Local Disk (C:) > Ozeki > Processed

| Name | Date modified | Type | Size |
|---|---|---|---|
| report_2009080822.sql | 08/09/2020 08:22 | Microsoft SQL Ser... | 6 KB |
| report_2009080847.sql | 08/09/2020 08:48 | Microsoft SQL Ser... | 17 KB |
| report_2009080849.sql | 08/09/2020 08:49 | Microsoft SQL Ser... | 6 KB |
| report_2009080851.sql | 08/09/2020 08:51 | Microsoft SQL Ser... | 6 KB |
| report_2009080859.sql | 08/09/2020 09:00 | Microsoft SQL Ser... | 6 KB |
| report_2009080903.sql | 08/09/2020 09:04 | Microsoft SQL Ser... | 6 KB |
| report_2009080904.sql | 08/09/2020 09:05 | Microsoft SQL Ser... | 17 KB |
| report_2009080910.sql | 08/09/2020 09:10 | Microsoft SQL Ser... | 6 KB |

8 items

Figure 7 - SQL files moved to processed folder

Finally, if you look in the database you can see that all the messages have been inserted in the messsagereport table as the Figure 8 shows.



SQLQuery3.sql - DESKTOP-82I1GMF\SQLEXPRESS.ozeki (DESKTOP-82I1GMF\User (56)) - Microsoft SQL Server Management Studio

```
/****** Script for SelectTopNRows command from SSMS  ******/
SELECT TOP (1000) [logid]
      ,[messageid]
      ,[fromconnection]
      ,[fromaddress]
      ,[requestedconnection]
      ,[toconnection]
      ,[toaddress]
      ,[messagetext]
      ,[route]
      ,[datecreated]
      ,[datedeliverytoconnectionsuccess]
      ,[datedeliverytoconnectionfailed]
      ,[deliverytoconnectionreference]
      ,[deliverytoconnectionstatusmessage]
      ,[datedeliverytorecipientsuccess]
```

| | logid | messageid | fromconnection | fromaddress | requestedconnection | toconnection | toaddress |
|---|---|---|---|---|---|---|---|
| 1 | 1 | 761ce176-85b9-4231-a38e-b04898ad2621 | smppuser@localhost | +44123456758 | NULL | NULL | +4433212 |
| 2 | 2 | 73af7fa5-8967-49e8-a024-b2ed9[illegible]3771 | smppuser@localhost | +44123456758 | NULL | NULL | +4433212 |
| 3 | 3 | d2688743-b7a0-400c-abaa-0a6c444e3f4d | smppuser@localhost | +44123456758 | NULL | NULL | +4433212 |
| 4 | 4 | 06add310f487-43d4-a54b-2dfae920bf5a | smppuser@localhost | +44123456758 | NULL | NULL | +4433212 |
| 5 | 5 | db80a6fc-b600-4abf-9b32-376894dbf7eb | smppuser@localhost | +44123456758 | NULL | NULL | +4433212 |
| 6 | 6 | 30d711fc-5fee-48f7-80bc-b12fb71d9492 | smppuser@localhost | +44123456758 | NULL | NULL | +4433212 |
| 7 | 7 | b2306009-571c-499d-b442-7699879637fe | smppuser@localhost | +44123456758 | NULL | NULL | +4433212 |
| 8 | 8 | 29c792be-e7e2-4080-b818-cd3304a916f1 | smppuser@localhost | +44123456758 | NULL | NULL | +4433212 |
| 9 | 9 | 6377c3cc-9bfe-4052-ae7f-3d119e6da437 | smppuser@localhost | +44123456758 | NULL | NULL | +4433212 |

Query executed successfully. | DESKTOP-82I1GMF\SQLEXPRESS ... | DESKTOP-82I1GMF\User (56) | ozeki | 00:00:00 | 132 rows

Figure 8 - Messages inserted into database

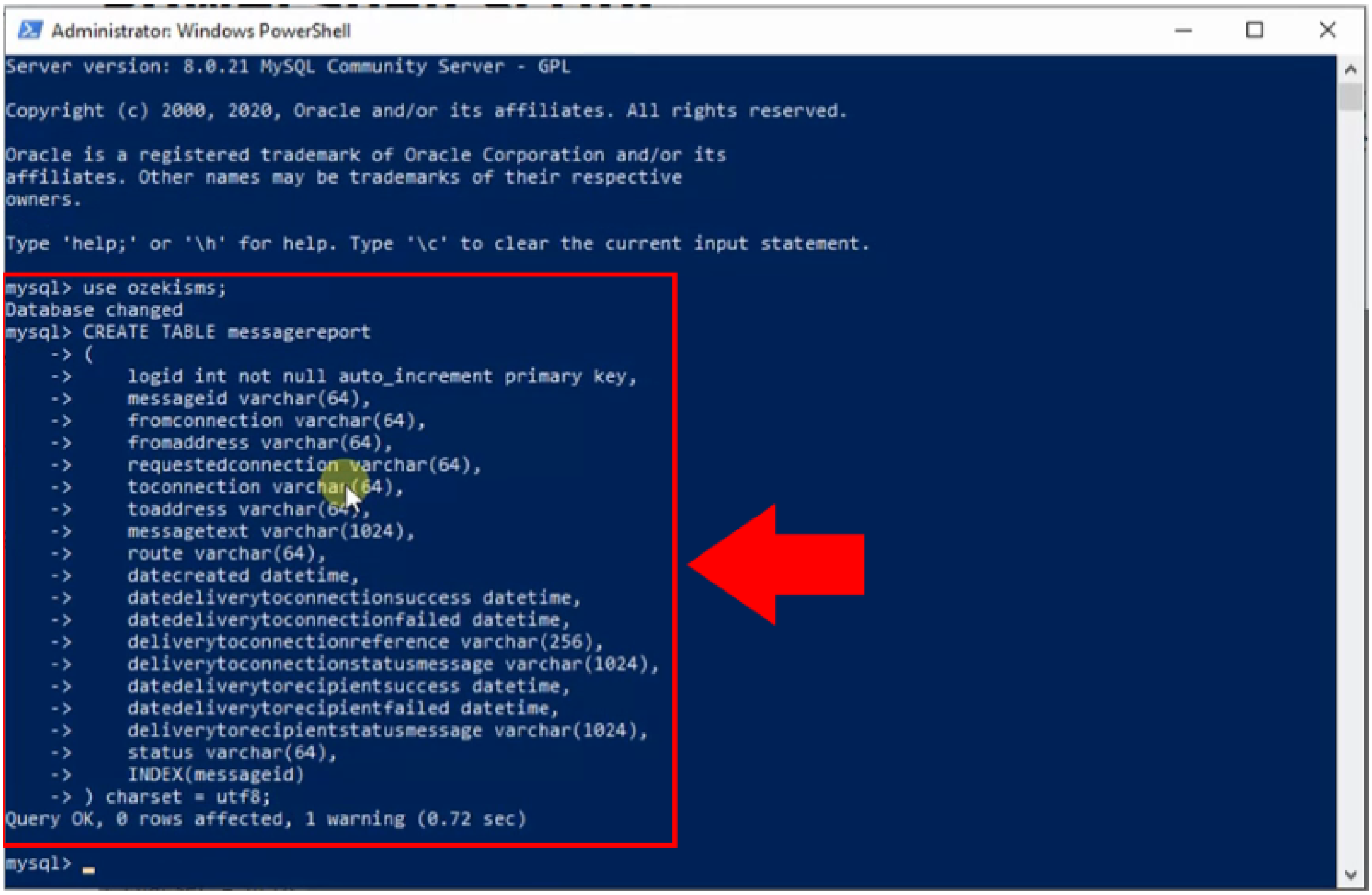
# PowerShell script

***This script can be used to automatically import sql files older then 20 minutes into the a MySQL database. It is helpful if you enable SQL Text logging for high performance configurations. In order to execute this script you have to save it with the .ps1 extension, and you must enable power shell script execution for windows in the group policy editor (or you must execute the script as Administrator).***

## Create the following MySQL table

```
CREATE TABLE messagereport
(
logid int not null auto_increment primary key,
messageid varchar(64),
fromconnection varchar(64),
fromaddress varchar(64),
requestedconnection varchar(64),
toconnection varchar(64),
toaddress varchar(64),
messagetext varchar(1024),
route varchar(64),
datecreated datetime,
datedeliverytoconnectionsuccess datetime,
datedeliverytoconnectionfailed datetime,
deliverytoconnectionreference varchar(256),
deliverytoconnectionstatusmessage varchar(1024),
datedeliverytorecipientsuccess datetime,
datedeliverytorecipientfailed datetime,
deliverytorecipientstatusmessage varchar(1024),
status varchar(64),
INDEX(messageid)
) charset = utf8;
```

The first step is to create the messagereport table in the database. All outgoing and incoming messages will be stored in this table and updated according to their status. If you have already created this table you can skip this step. Copy the create table script above and execute it in your database as you can see it on the Figure 1.

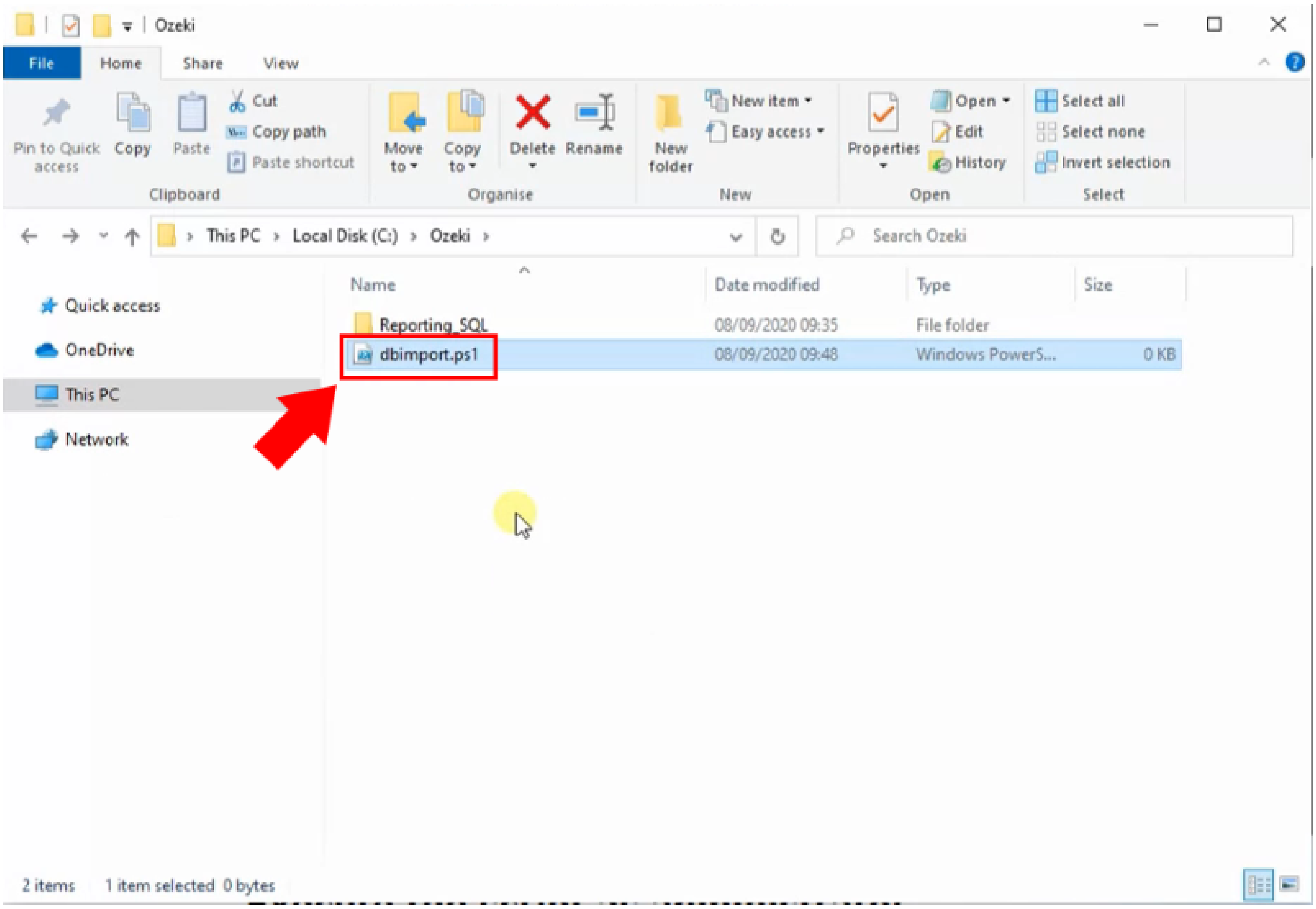


```
Server version: 8.0.21 MySQL Community Server - GPL

Copyright (c) 2000, 2020, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> use ozekisms;
Database changed
mysql> CREATE TABLE messagereport
    -> (
    ->     logid int not null auto_increment primary key,
    ->     messageid varchar(64),
    ->     fromconnection varchar(64),
    ->     fromaddress varchar(64),
    ->     requestedconnection varchar(64),
    ->     toconnection varchar(64),
    ->     toaddress varchar(64),
    ->     messagetext varchar(1024),
    ->     route varchar(64),
    ->     datecreated datetime,
    ->     datedeliverytoconnectionsuccess datetime,
    ->     datedeliverytoconnectionfailed datetime,
    ->     deliverytoconnectionreference varchar(256),
    ->     deliverytoconnectionstatusmessage varchar(1024),
    ->     datedeliverytorecipientsuccess datetime,
    ->     datedeliverytorecipientfailed datetime,
    ->     deliverytorecipientstatusmessage varchar(1024),
    ->     status varchar(64),
    ->     INDEX(messageid)
    -> ) charset = utf8;
Query OK, 0 rows affected, 1 warning (0.72 sec)

mysql> _
```

Figure 1 - Create messagereport table

## Save the following powershell script as "dbimport.ps1"

The next step is to create the "dbimport.ps1" file in the C:\Ozeki folder as shown in Figure 2. This file will contain the script that periodically runs SQL queries in the database.

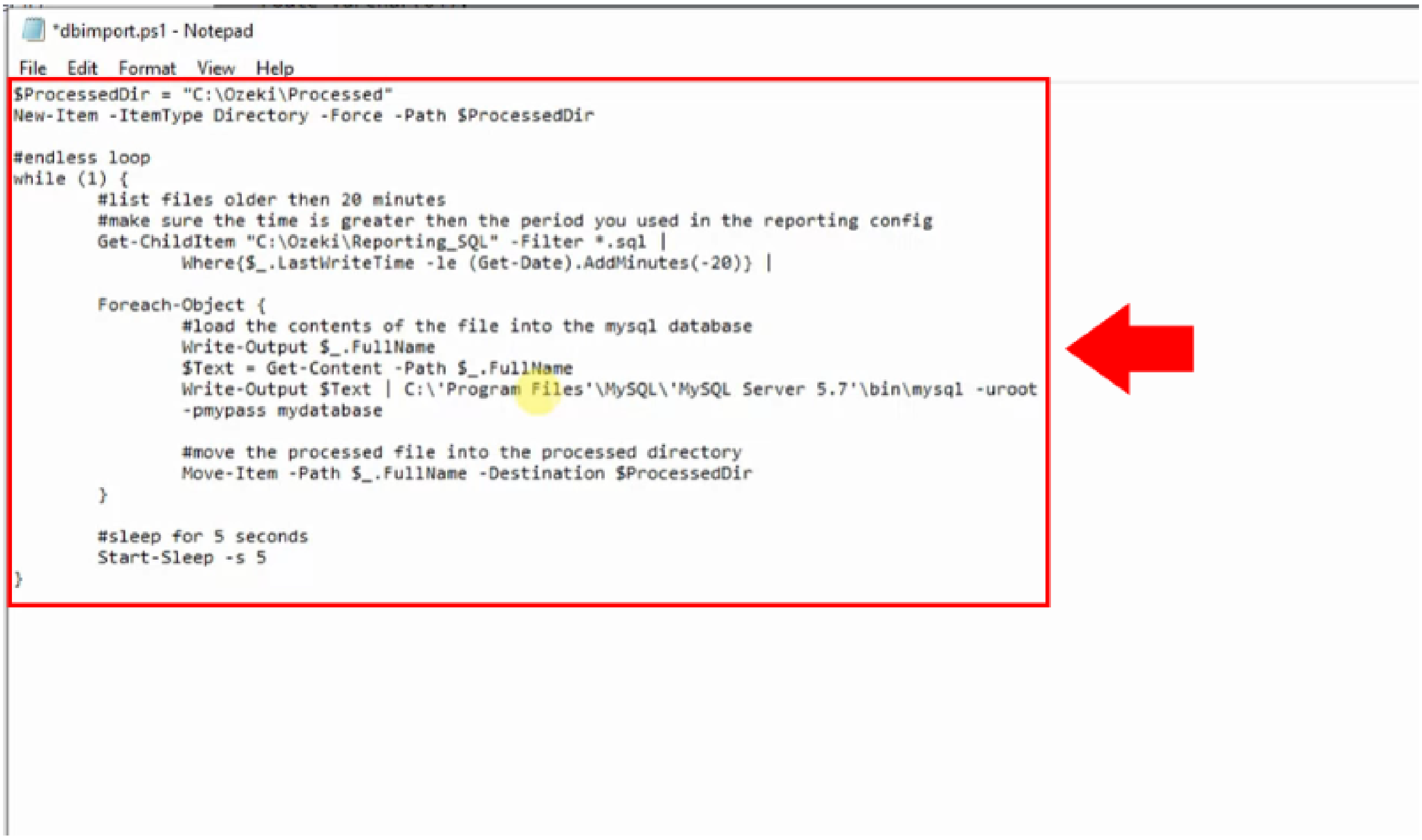


| Name | Date modified | Type | Size |
|---|---|---|---|
| Reporting_SQL | 08/09/2020 09:35 | File folder | |
| dbimport.ps1 | 08/09/2020 09:48 | Windows PowerS... | 0 KB |

Figure 2 - Create the script file

```
$ProcessedDir = "C:\Ozeki\Processed"
New-Item -ItemType Directory -Force -Path $ProcessedDir

#endless loop
while (1) {
#list files older then 20 minutes
#make sure the time is greater then the period you used in the reporting config
Get-ChildItem "C:\Ozeki\Reporting_SQL" -Filter *.sql |
Where{$_.LastWriteTime -le (Get-Date).AddMinutes(-20)} |

Foreach-Object {
#load the contents of the file into the mysql database
Write-Output $_.FullName
$Text = Get-Content -Path $_.FullName
Write-Output $Text | C:\'Program Files'\MySQL\'MySQL Server 5.7'\bin\mysql -uroot
-pmypass mydatabase

#move the processed file into the processed directory
Move-Item -Path $_.FullName -Destination $ProcessedDir
}

#sleep for 5 seconds
Start-Sleep -s 5
}
```

Copy the power shell script above and paste it in the "dbimport.ps1" file as you can see it on the Figure 3.

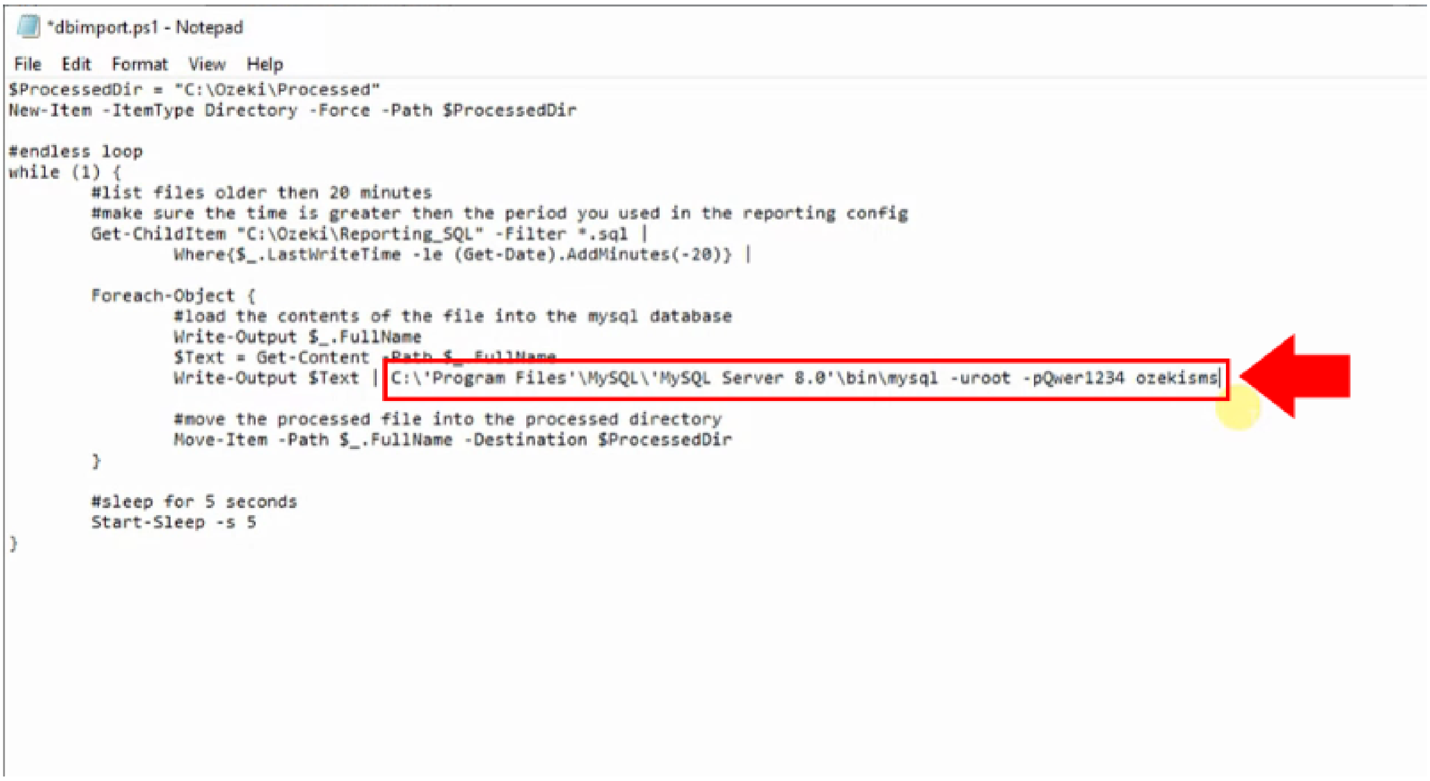


*dbimport.ps1 - Notepad

```
$ProcessedDir = "C:\Ozeki\Processed"
New-Item -ItemType Directory -Force -Path $ProcessedDir

#endless loop
while (1) {
        #list files older then 20 minutes
        #make sure the time is greater then the period you used in the reporting config
        Get-ChildItem "C:\Ozeki\Reporting_SQL" -Filter *.sql |
                Where{$_.LastWriteTime -le (Get-Date).AddMinutes(-20)} |

        Foreach-Object {
                #load the contents of the file into the mysql database
                Write-Output $_.FullName
                $Text = Get-Content -Path $_.FullName
                Write-Output $Text | C:\'Program Files'\MySQL\'MySQL Server 5.7'\bin\mysql -uroot
                -pmypass mydatabase

                #move the processed file into the processed directory
                Move-Item -Path $_.FullName -Destination $ProcessedDir
        }

        #sleep for 5 seconds
        Start-Sleep -s 5
}
```

Figure 3 - Paste the script into the file

Finally, modify the script according to your own MySQL database. In the path, change the version number of the MySQL database. Enter the user name and password you want to use, and replace "mydatabase" with the name of your database (Figure 4).



*dbimport.ps1 - Notepad

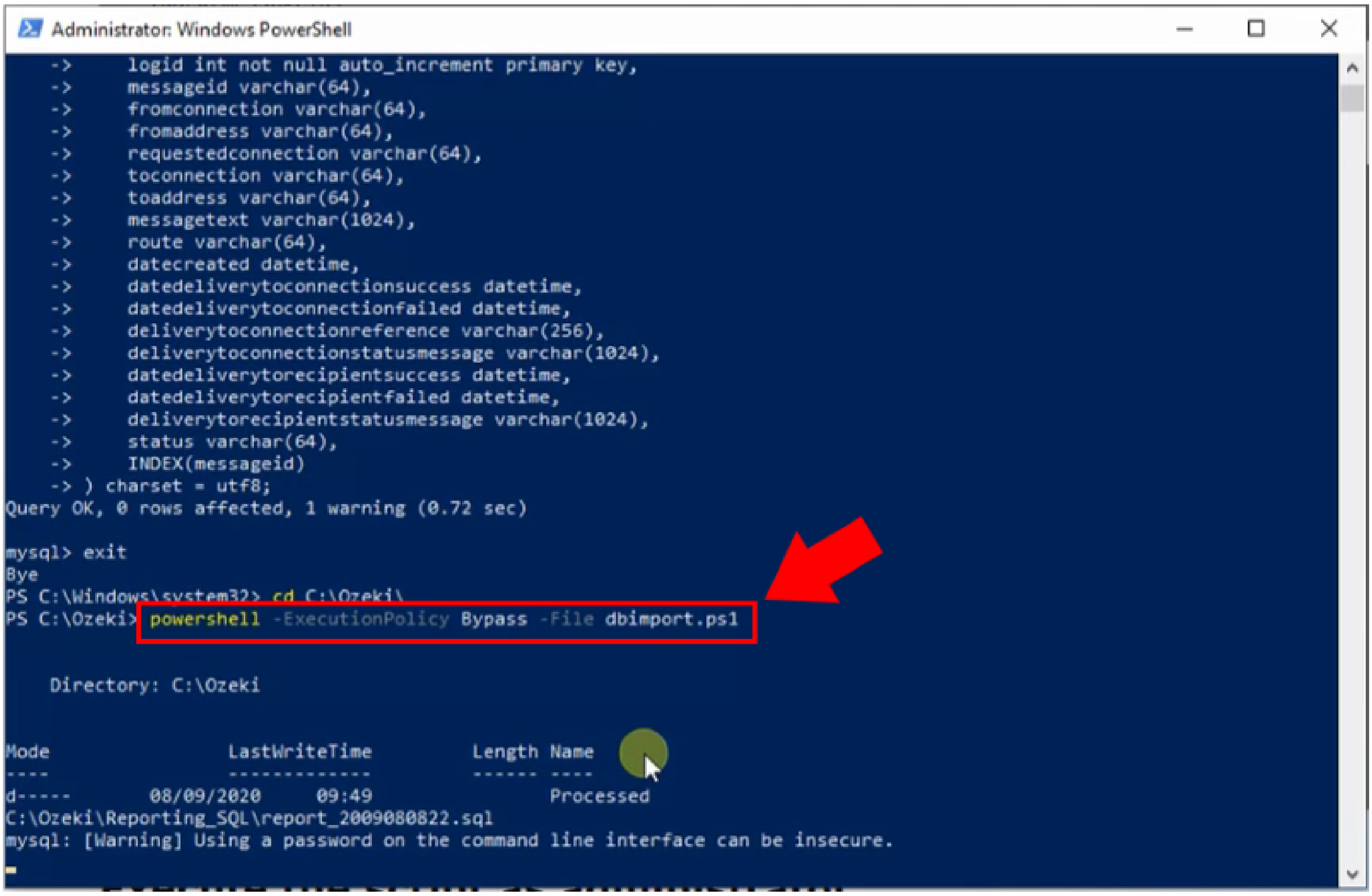
```
$ProcessedDir = "C:\Ozeki\Processed"
New-Item -ItemType Directory -Force -Path $ProcessedDir

#endless loop
while (1) {
        #list files older then 20 minutes
        #make sure the time is greater then the period you used in the reporting config
        Get-ChildItem "C:\Ozeki\Reporting_SQL" -Filter *.sql |
                Where{$_.LastWriteTime -le (Get-Date).AddMinutes(-20)} |

        Foreach-Object {
                #load the contents of the file into the mysql database
                Write-Output $_.FullName
                $Text = Get-Content -Path $_.FullName
                Write-Output $Text | C:\'Program Files'\MySQL\'MySQL Server 8.0'\bin\mysql -uroot -pQwer1234 ozekisms

                #move the processed file into the processed directory
                Move-Item -Path $_.FullName -Destination $ProcessedDir
        }

        #sleep for 5 seconds
        Start-Sleep -s 5
}
```

Figure 4 - Change MySQL command

# Execute the script as administrator

```
powershell -ExecutionPolicy Bypass -File dbimport.ps1
```

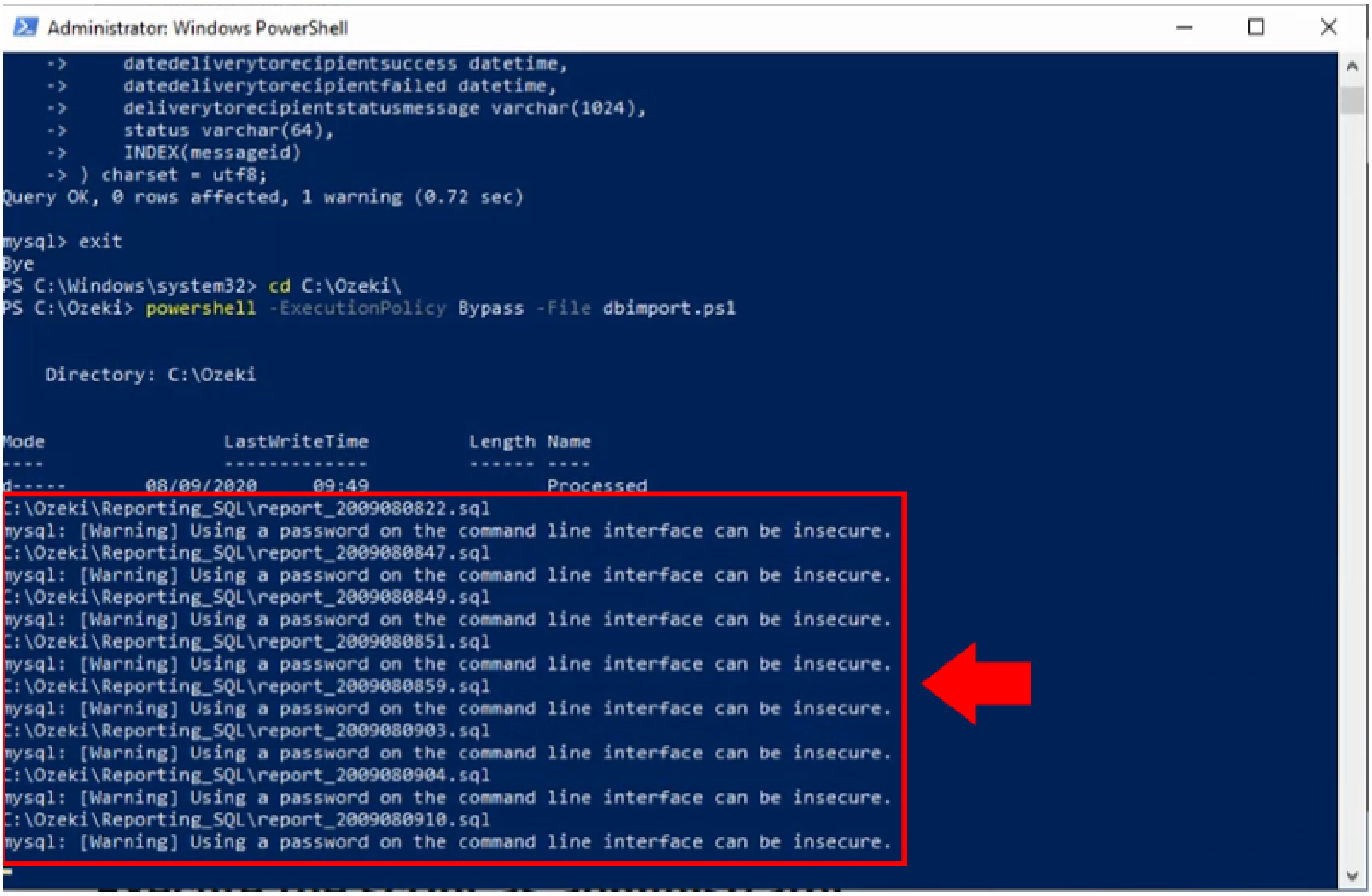
Now execute the script in the power shell by running the above command as you can see it in the Figure 5.



```
    ->     logid int not null auto_increment primary key,
    ->     messageid varchar(64),
    ->     fromconnection varchar(64),
    ->     fromaddress varchar(64),
    ->     requestedconnection varchar(64),
    ->     toconnection varchar(64),
    ->     toaddress varchar(64),
    ->     messagetext varchar(1024),
    ->     route varchar(64),
    ->     datecreated datetime,
    ->     datedeliverytoconnectionsuccess datetime,
    ->     datedeliverytoconnectionfailed datetime,
    ->     deliverytoconnectionreference varchar(256),
    ->     deliverytoconnectionstatusmessage varchar(1024),
    ->     datedeliverytorecipientsuccess datetime,
    ->     datedeliverytorecipientfailed datetime,
    ->     deliverytorecipientstatusmessage varchar(1024),
    ->     status varchar(64),
    ->     INDEX(messageid)
    -> ) charset = utf8;
Query OK, 0 rows affected, 1 warning (0.72 sec)

mysql> exit
Bye
PS C:\Windows\system32> cd C:\Ozeki\
PS C:\Ozeki> powershell -ExecutionPolicy Bypass -File dbimport.ps1


    Directory: C:\Ozeki


Mode                LastWriteTime         Length Name
----                -------------         ------ ----
d-----       08/09/2020     09:49                Processed
C:\Ozeki\Reporting_SQL\report_2009080822.sql
mysql: [Warning] Using a password on the command line interface can be insecure.
```

Figure 5 - Run powershell script

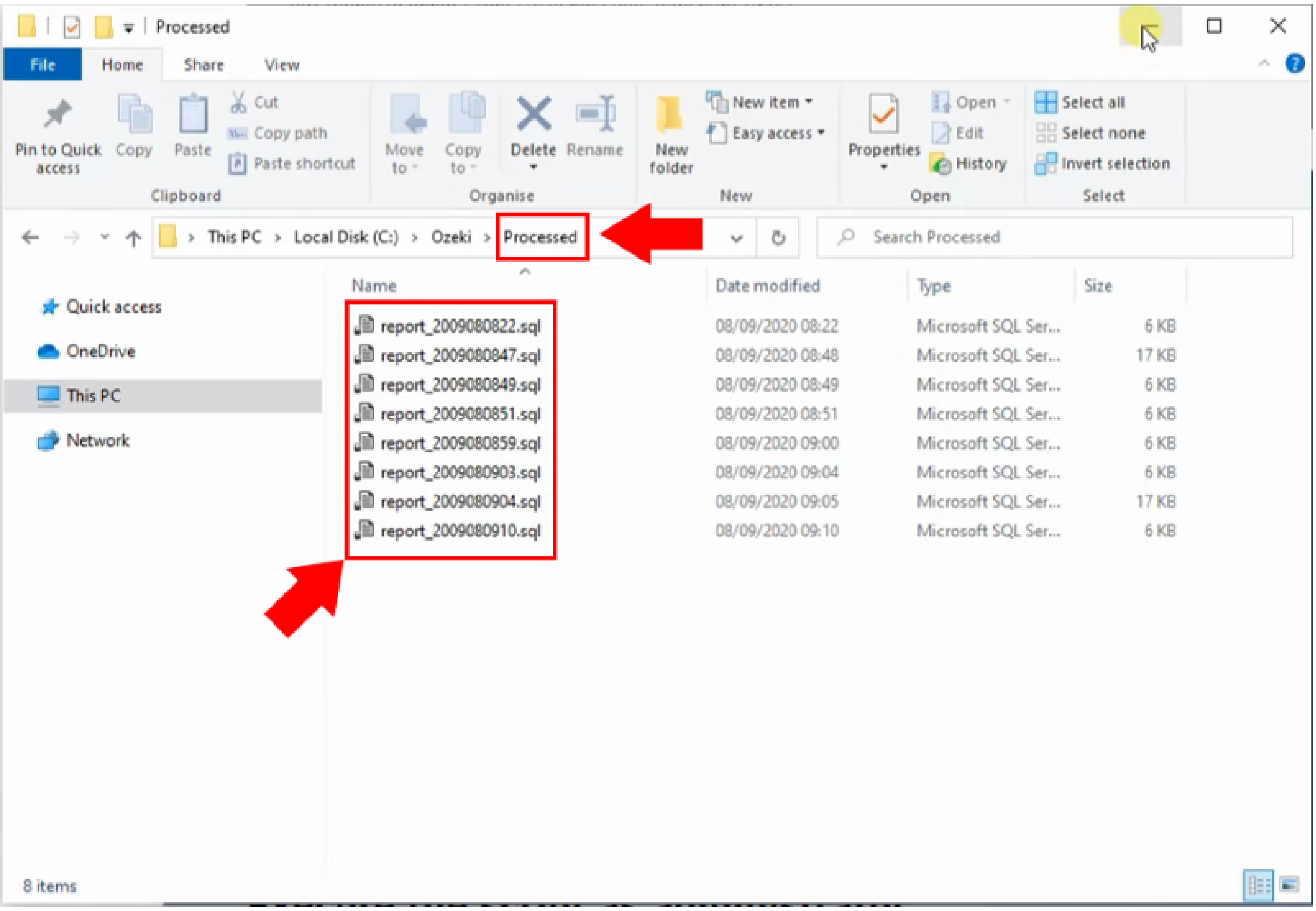
You can see that the script execute all the SQL files what is older than 20 minutes in the 'C:\Ozeki\Reporting_SQL' folder one by one (Figure 6).



```
    ->     datedeliverytorecipientsuccess datetime,
    ->     datedeliverytorecipientfailed datetime,
    ->     deliverytorecipientstatusmessage varchar(1024),
    ->     status varchar(64),
    ->     INDEX(messageid)
    -> ) charset = utf8;
Query OK, 0 rows affected, 1 warning (0.72 sec)

mysql> exit
Bye
PS C:\Windows\system32> cd C:\Ozeki\
PS C:\Ozeki> powershell -ExecutionPolicy Bypass -File dbimport.ps1


    Directory: C:\Ozeki


Mode                LastWriteTime         Length Name
----                -------------         ------ ----
d-----       08/09/2020     09:49                Processed
C:\Ozeki\Reporting_SQL\report_2009080822.sql
mysql: [Warning] Using a password on the command line interface can be insecure.
C:\Ozeki\Reporting_SQL\report_2009080847.sql
mysql: [Warning] Using a password on the command line interface can be insecure.
C:\Ozeki\Reporting_SQL\report_2009080849.sql
mysql: [Warning] Using a password on the command line interface can be insecure.
C:\Ozeki\Reporting_SQL\report_2009080851.sql
mysql: [Warning] Using a password on the command line interface can be insecure.
C:\Ozeki\Reporting_SQL\report_2009080859.sql
mysql: [Warning] Using a password on the command line interface can be insecure.
C:\Ozeki\Reporting_SQL\report_2009080903.sql
mysql: [Warning] Using a password on the command line interface can be insecure.
C:\Ozeki\Reporting_SQL\report_2009080904.sql
mysql: [Warning] Using a password on the command line interface can be insecure.
C:\Ozeki\Reporting_SQL\report_2009080910.sql
mysql: [Warning] Using a password on the command line interface can be insecure.
```

Figure 6 - Script execute SQL queries

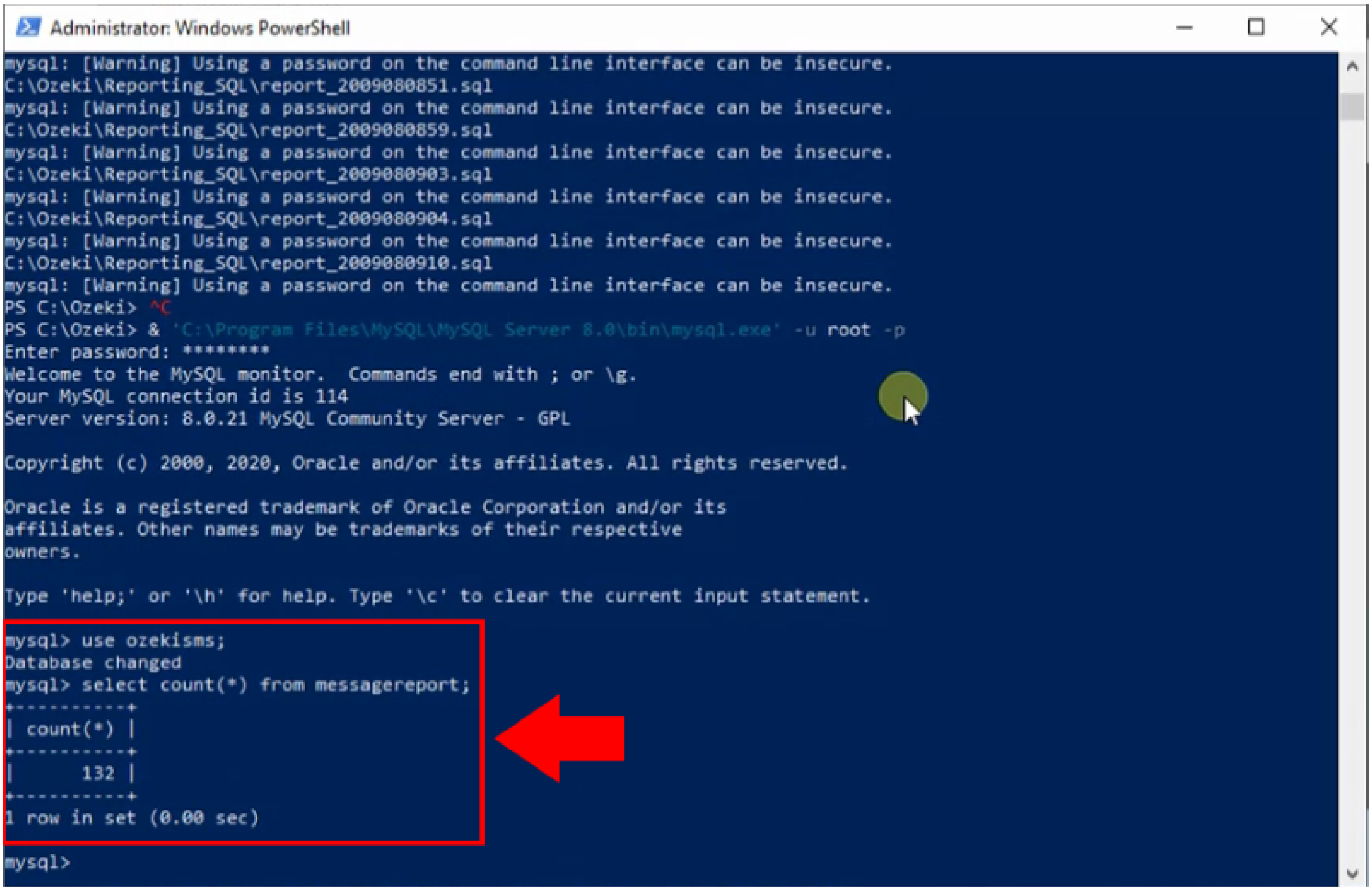
Every SQL file that the script has run is moded to the Processed folder so that no queries are lost (Figure 7). From here, you can delete them manually if you no longer need them.



| Name | Date modified | Type | Size |
|---|---|---|---|
| report_2009080822.sql | 08/09/2020 08:22 | Microsoft SQL Ser... | 6 KB |
| report_2009080847.sql | 08/09/2020 08:48 | Microsoft SQL Ser... | 17 KB |
| report_2009080849.sql | 08/09/2020 08:49 | Microsoft SQL Ser... | 6 KB |
| report_2009080851.sql | 08/09/2020 08:51 | Microsoft SQL Ser... | 6 KB |
| report_2009080859.sql | 08/09/2020 09:00 | Microsoft SQL Ser... | 6 KB |
| report_2009080903.sql | 08/09/2020 09:04 | Microsoft SQL Ser... | 6 KB |
| report_2009080904.sql | 08/09/2020 09:05 | Microsoft SQL Ser... | 17 KB |
| report_2009080910.sql | 08/09/2020 09:10 | Microsoft SQL Ser... | 6 KB |

Figure 7 - SQL files moved to processed folder

Finally, if you look in the database you can see that all the messages have been inserted in the messsagereport table as the Figure 8 shows.



```
mysql: [Warning] Using a password on the command line interface can be insecure.
C:\Ozeki\Reporting_SQL\report_2009080851.sql
mysql: [Warning] Using a password on the command line interface can be insecure.
C:\Ozeki\Reporting_SQL\report_2009080859.sql
mysql: [Warning] Using a password on the command line interface can be insecure.
C:\Ozeki\Reporting_SQL\report_2009080903.sql
mysql: [Warning] Using a password on the command line interface can be insecure.
C:\Ozeki\Reporting_SQL\report_2009080904.sql
mysql: [Warning] Using a password on the command line interface can be insecure.
C:\Ozeki\Reporting_SQL\report_2009080910.sql
mysql: [Warning] Using a password on the command line interface can be insecure.
PS C:\Ozeki> ^C
PS C:\Ozeki> & 'C:\Program Files\MySQL\MySQL Server 8.0\bin\mysql.exe' -u root -p
Enter password: ********
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 114
Server version: 8.0.21 MySQL Community Server - GPL

Copyright (c) 2000, 2020, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> use ozekisms;
Database changed
mysql> select count(*) from messagereport;
+----------+
| count(*) |
+----------+
|      132 |
+----------+
1 row in set (0.00 sec)

mysql>
```

Figure 8 - Messages inserted into database

# How to execute a power script with Windows Task Scheduler

***This guide shows you how to set up Windows Task Scheduler in order to run a power script with it every 10 minutes. It is helpful if you enable SQL Text logging for high performance configurations.***

Task Scheduler allows you to automate many tasks on Windows 10. With this tool, you can start applications, run commands, and execute scripts at a particular day and time, or you can also trigger tasks when a specific event occurs. To create a task using advanced settings using the Task Scheduler open the windows start menu and search for Task Scheduler, click the top result to open the experience (Figure 1).

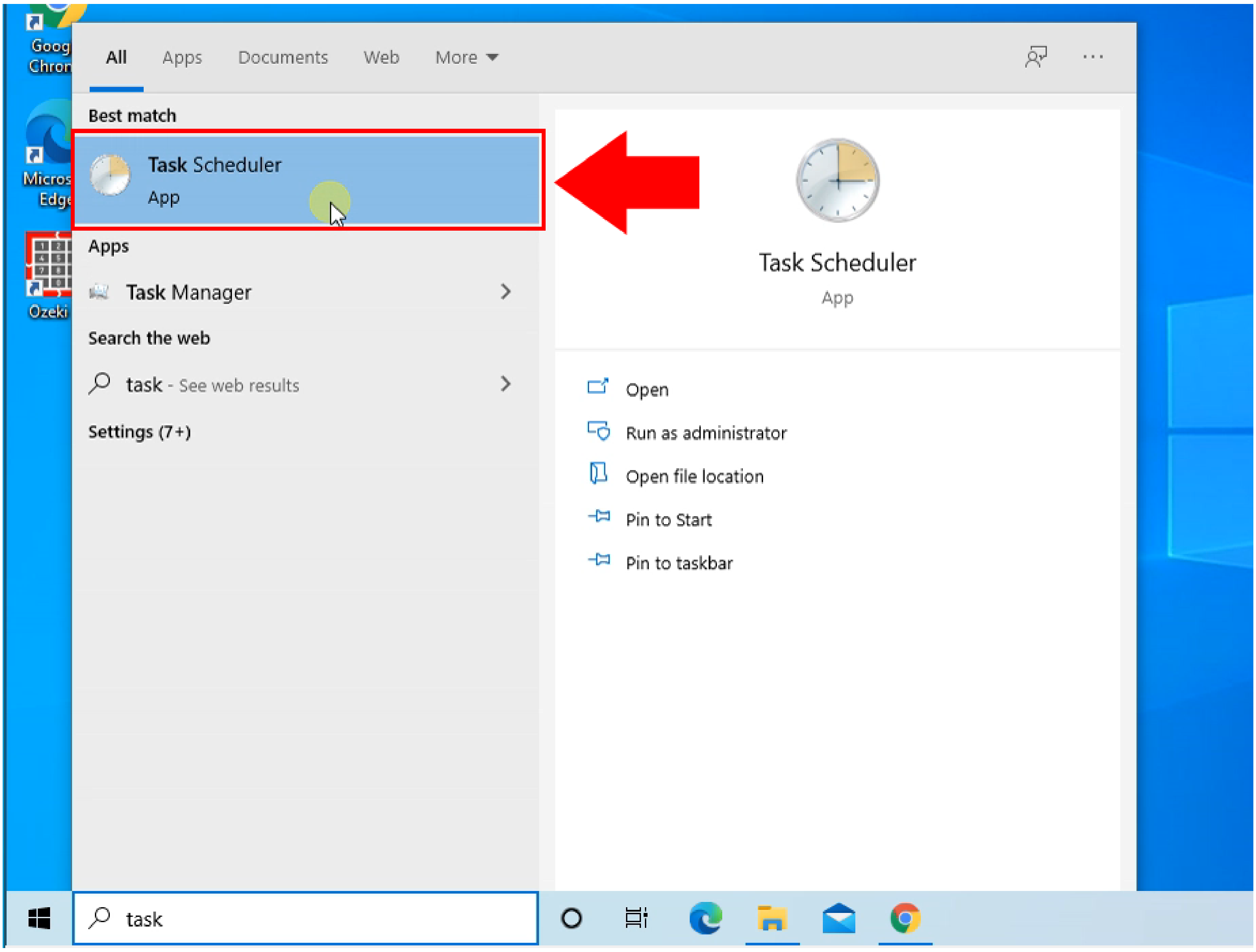
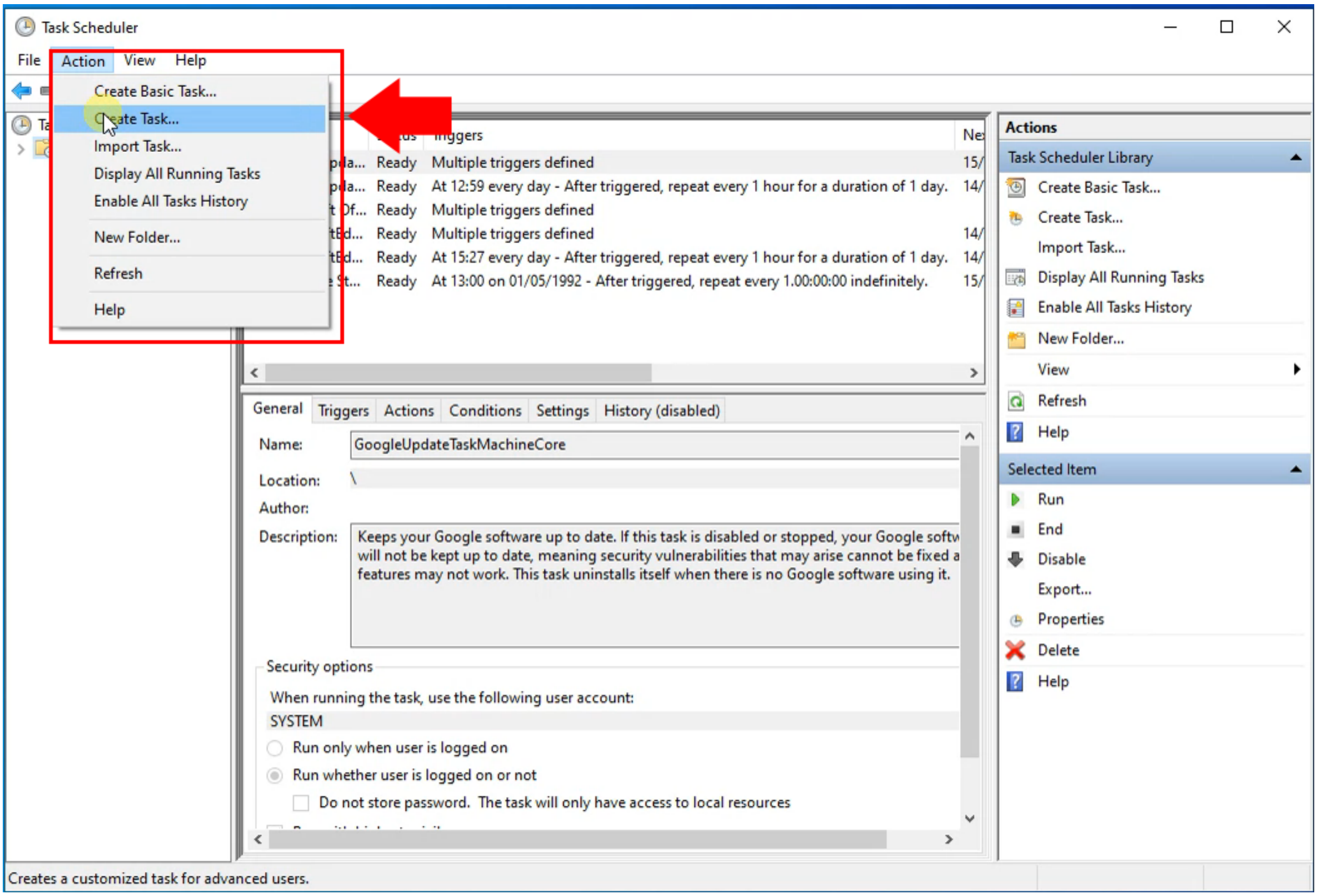


Figure 1 - Open Task Scheduler

Expand the "Task Scheduler Library" branch, and click the Action menu. Then select the Create Task option as you can see it in the Figure 2.



| | Status | Triggers | Ne |
|---|---|---|---|
| pda... | Ready | Multiple triggers defined | 15/ |
| pda... | Ready | At 12:59 every day - After triggered, repeat every 1 hour for a duration of 1 day. | 14/ |
| t Of... | Ready | Multiple triggers defined | |
| tEd... | Ready | Multiple triggers defined | 14/ |
| tEd... | Ready | At 15:27 every day - After triggered, repeat every 1 hour for a duration of 1 day. | 14/ |
| t St... | Ready | At 13:00 on 01/05/1992 - After triggered, repeat every 1.00:00:00 indefinitely. | 15/ |

Name: GoogleUpdateTaskMachineCore

Location: \

Author:

Description: Keeps your Google software up to date. If this task is disabled or stopped, your Google softw will not be kept up to date, meaning security vulnerabilities that may arise cannot be fixed a features may not work. This task uninstalls itself when there is no Google software using it.

Security options

When running the task, use the following user account:

SYSTEM

○ Run only when user is logged on

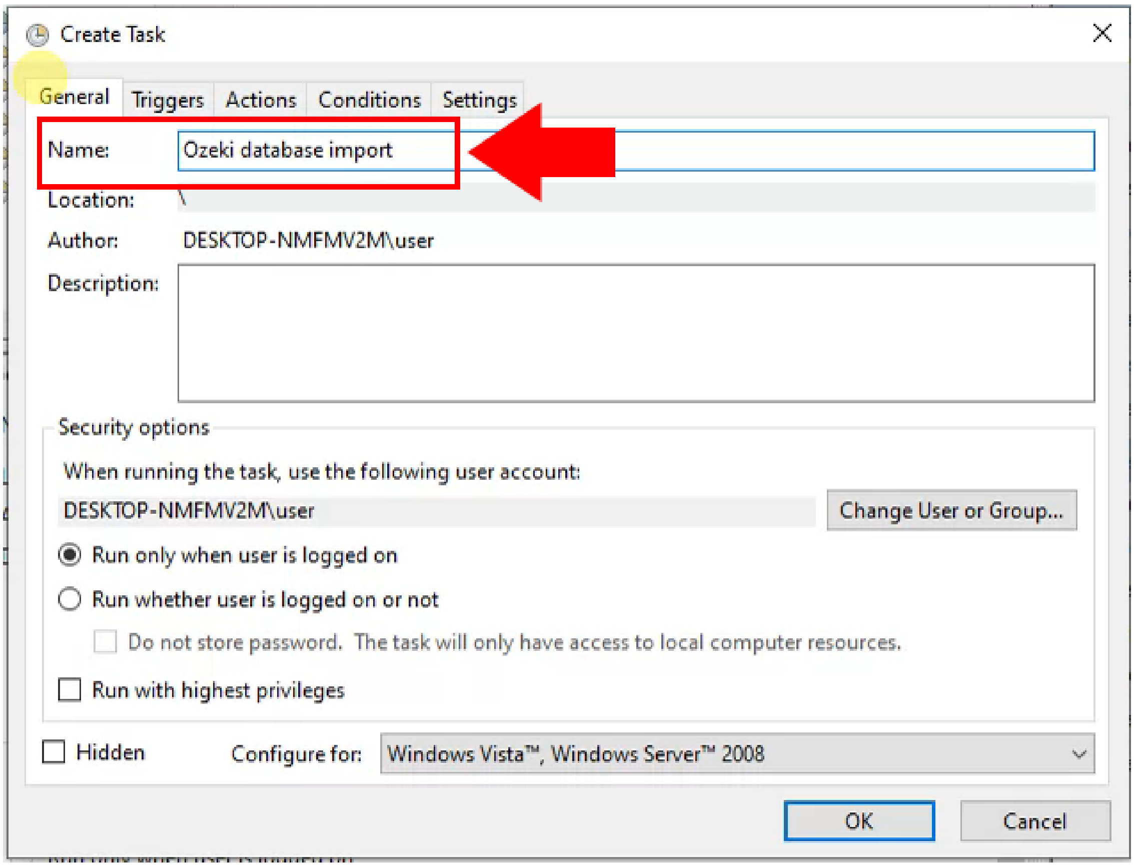
● Run whether user is logged on or not

□ Do not store password. The task will only have access to local resources

Creates a customized task for advanced users.

Figure 2 - Create new task

In the "Name" field, type a short descriptive name for the task. For example, Ozeki database import.



Name: Ozeki database import

Location: \

Author: DESKTOP-NMFMV2M\user

Description:

Security options

When running the task, use the following user account:

DESKTOP-NMFMV2M\user

● Run only when user is logged on

○ Run whether user is logged on or not

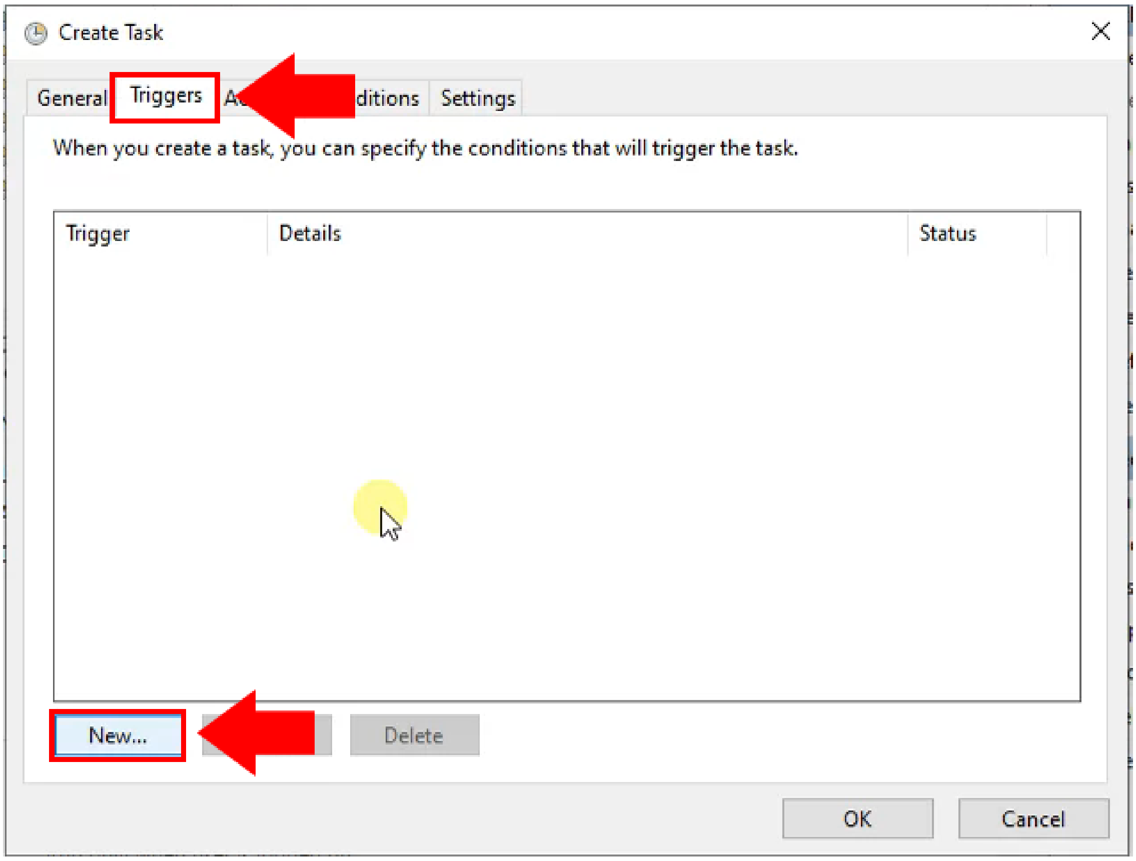
□ Do not store password. The task will only have access to local computer resources.

□ Run with highest privileges

□ Hidden    Configure for: Windows Vista™, Windows Server™ 2008

Figure 3 - Define task name

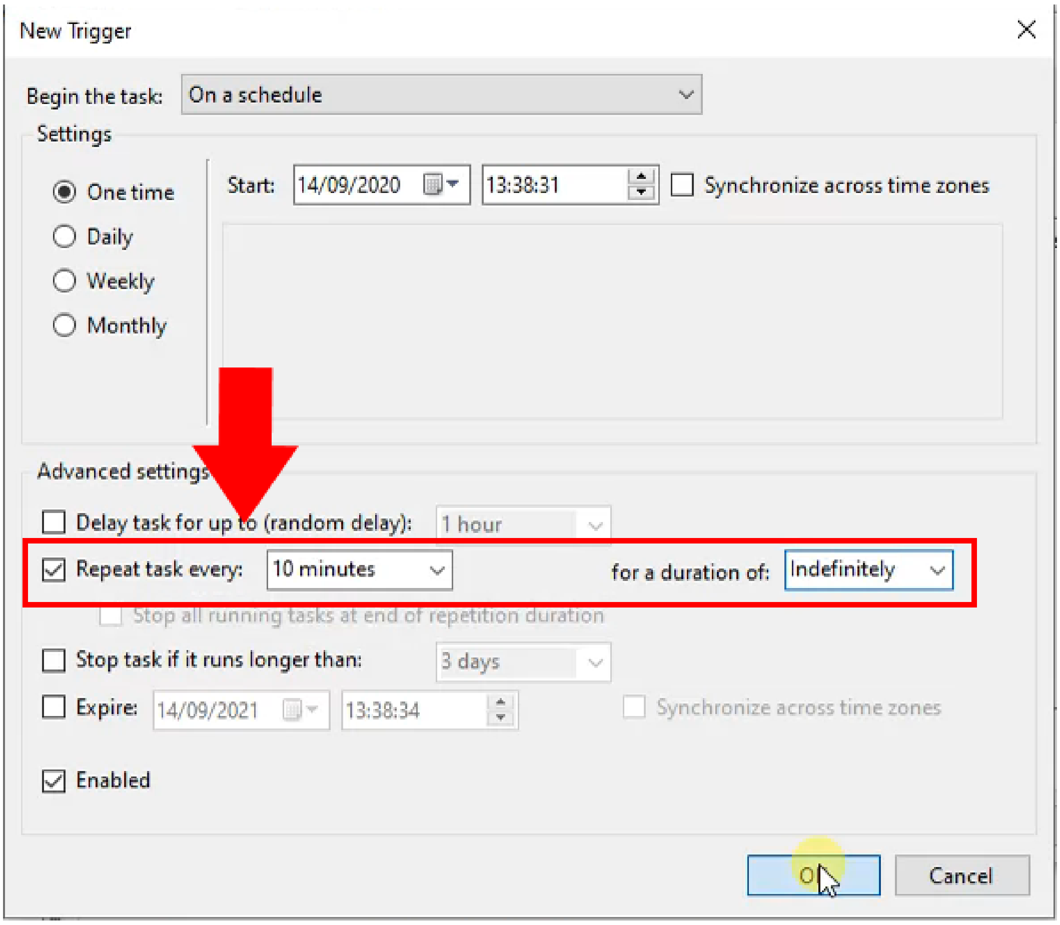
Now select the Trigger tab and click the New button. When you create a task, you can specify the conditions that will trigger the task (Figure 4).



When you create a task, you can specify the conditions that will trigger the task.

Figure 4 - Add new trigger

In the new trigger page Advanced section check the "Repeat task every:" option. Here select the 10 minutes option and the "for a duration of:" will be indefinitely as the Figure 5 shows.



Begin the task: On a schedule

● One time ○ Daily ○ Weekly ○ Monthly

Start: 14/09/2020 13:38:31 □ Synchronize across time zones

□ Delay task for up to (random delay): 1 hour

☑ Repeat task every: 10 minutes for a duration of: Indefinitely

□ Stop all running tasks at end of repetition duration

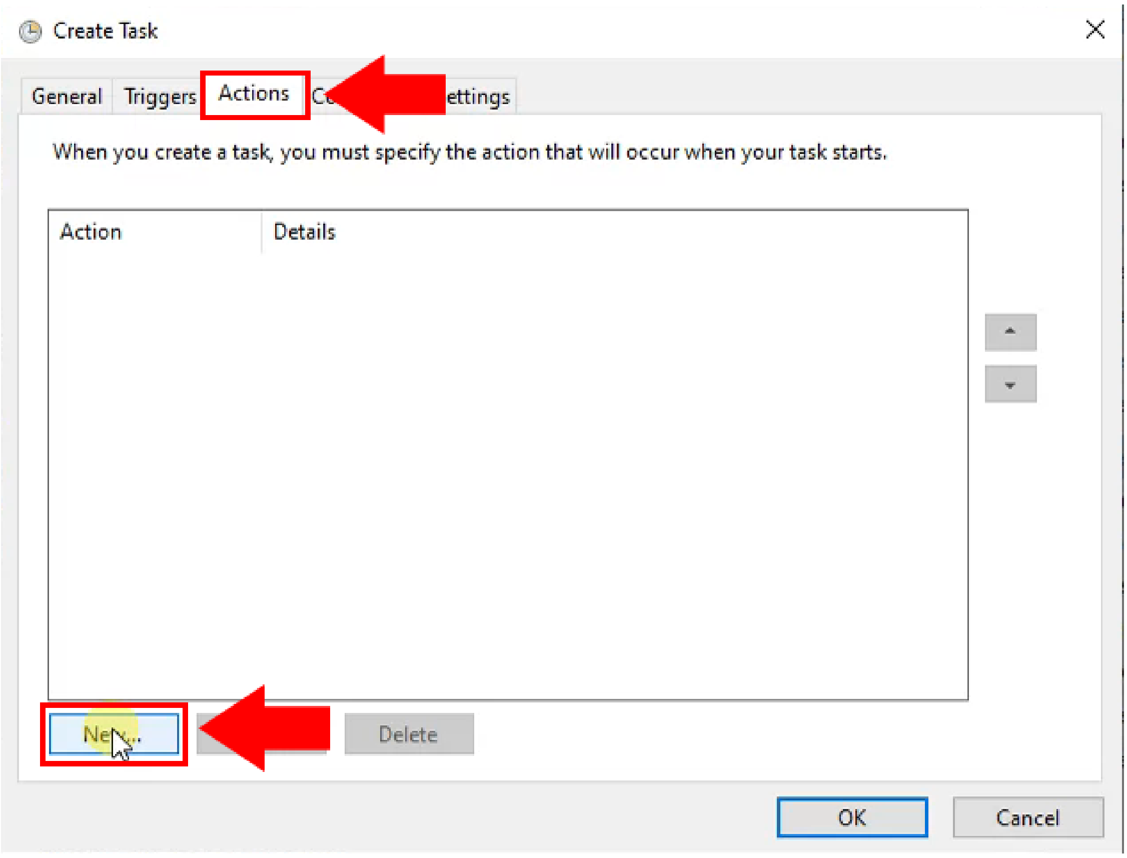
□ Stop task if it runs longer than: 3 days

□ Expire: 14/09/2021 13:38:34 □ Synchronize across time zones

☑ Enabled

Figure 5 - Define task running time

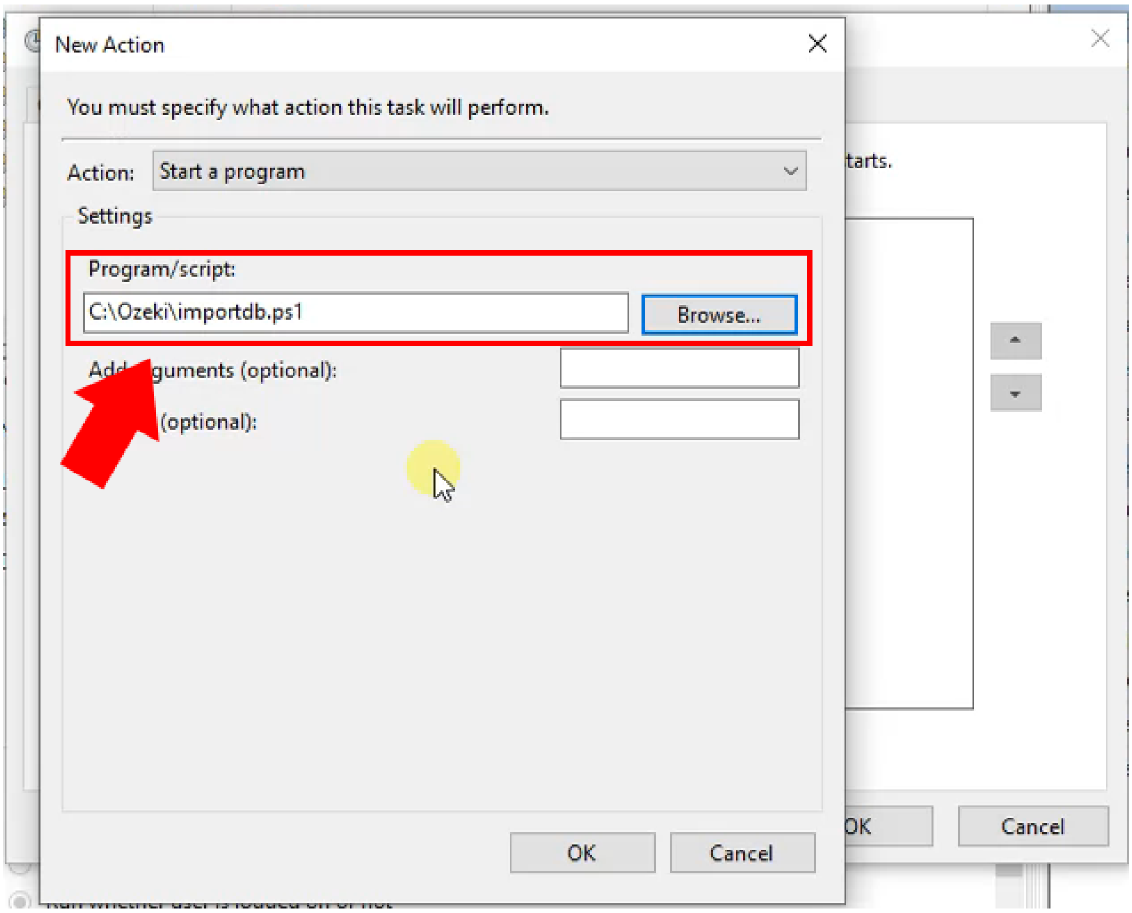
Now back in the Create Task page select the Action tab and click the New button. When you create a task, you must specify the action that will occur when your task starts (Figure 6).



When you create a task, you must specify the action that will occur when your task starts.

Figure 6 - Add new action

In the New Action page under the "Settings" section, in the "Program/script" field, specify the path for the Ozeki import database power shell script. In this example it is located in C:\Ozeki as you can see in the Figure 7.



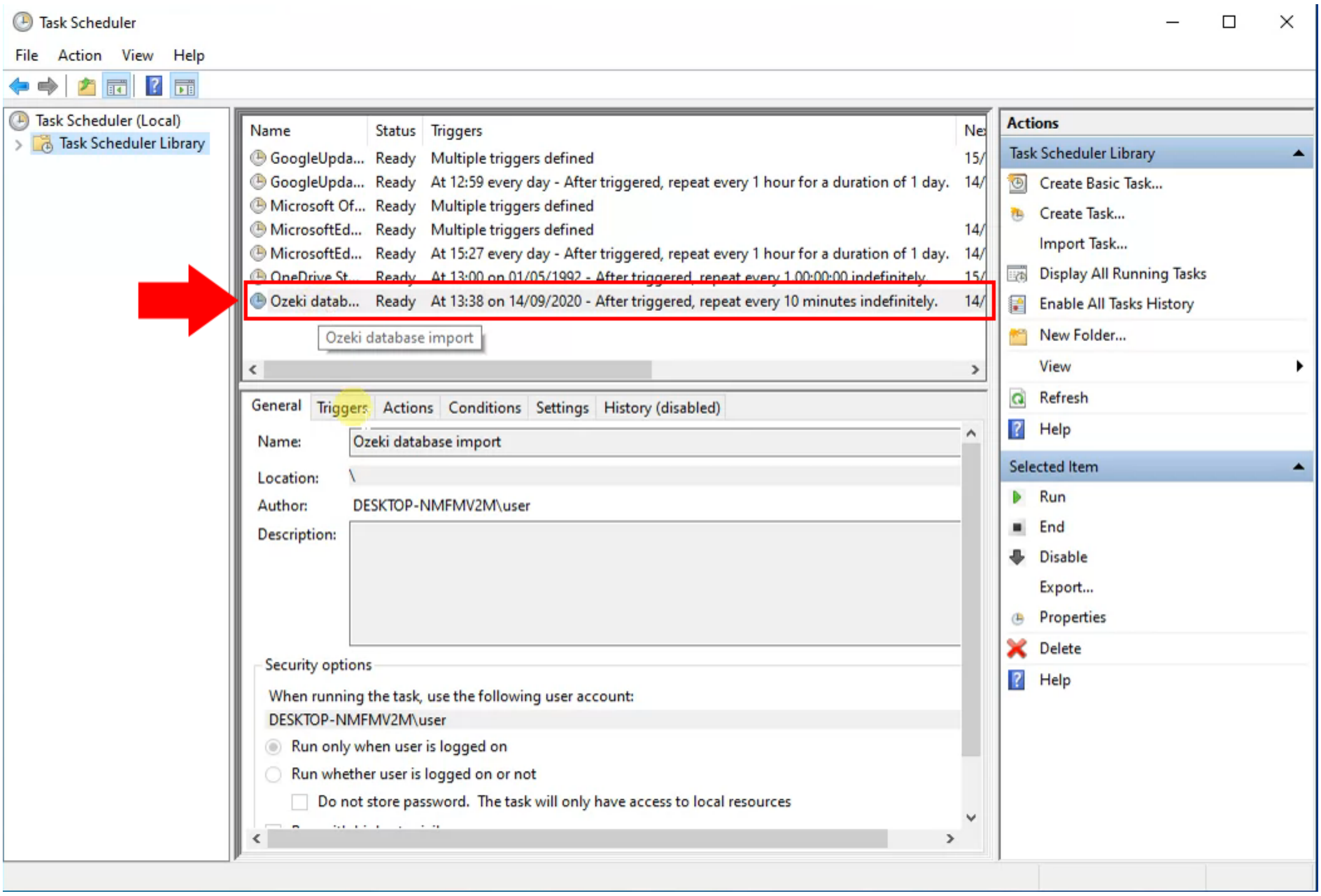
You must specify what action this task will perform.

Action: Start a program

Program/script: C:\Ozeki\importdb.ps1

Figure 7 - Browse import script

Finally in the Task Scheduler Library you will see the newly created scheduled task (Figure 8).



| Name | Status | Triggers | Ne |
|---|---|---|---|
| GoogleUpda... | Ready | Multiple triggers defined | 15/ |
| GoogleUpda... | Ready | At 12:59 every day - After triggered, repeat every 1 hour for a duration of 1 day. | 14/ |
| Microsoft Of... | Ready | Multiple triggers defined | |
| MicrosoftEd... | Ready | Multiple triggers defined | 14/ |
| MicrosoftEd... | Ready | At 15:27 every day - After triggered, repeat every 1 hour for a duration of 1 day. | 14/ |
| OneDrive St... | Ready | At 13:00 on 01/05/1992 - After triggered, repeat every 1.00:00:00 indefinitely. | 15/ |
| Ozeki datab... | Ready | At 13:38 on 14/09/2020 - After triggered, repeat every 10 minutes indefinitely. | 14/ |

Ozeki database import

Name: Ozeki database import

Location: \

Author: DESKTOP-NMFMV2M\user

Description:

Security options

When running the task, use the following user account:

DESKTOP-NMFMV2M\user

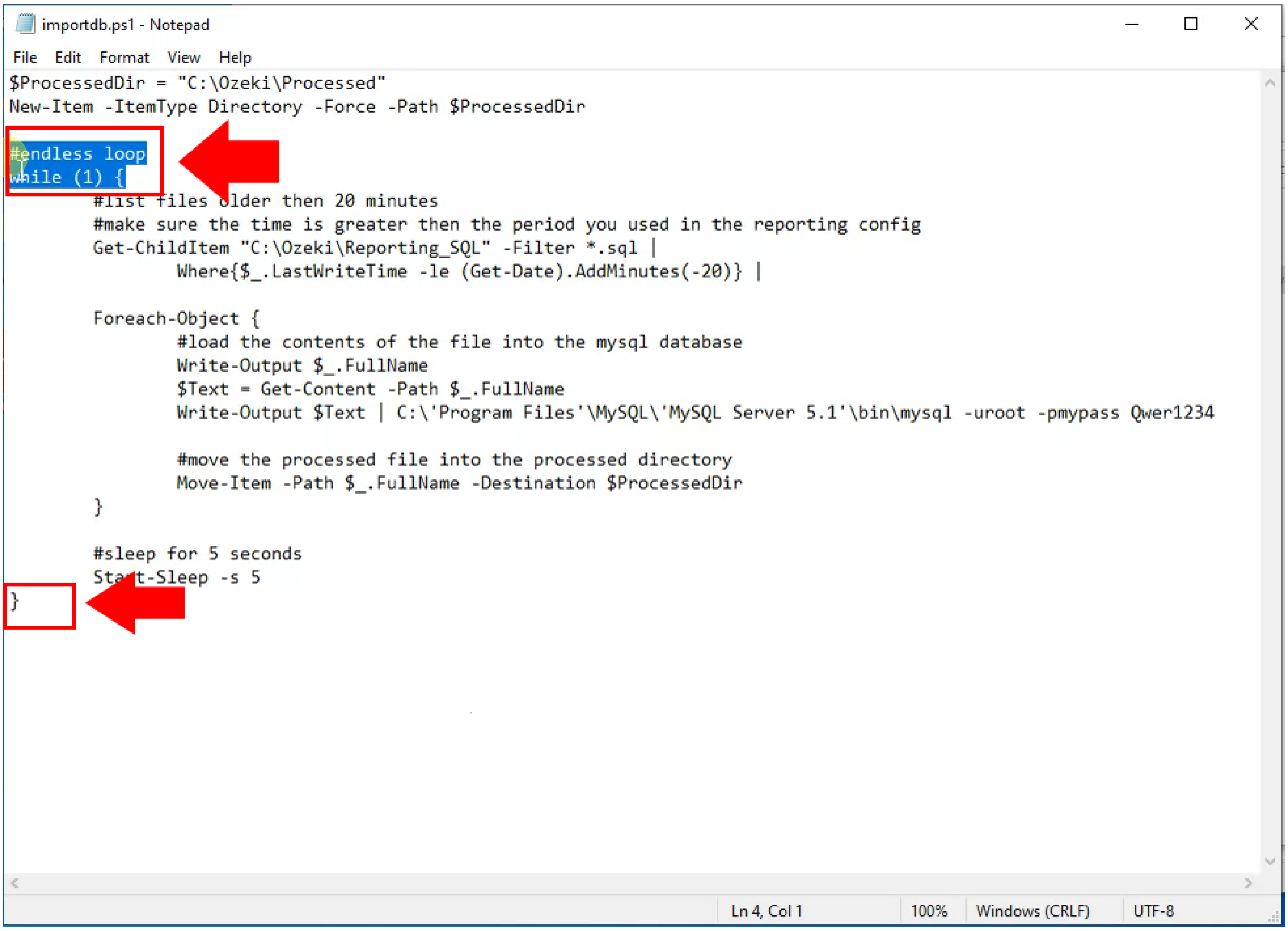
● Run only when user is logged on

○ Run whether user is logged on or not

□ Do not store password. The task will only have access to local resources

Figure 8 - Scheduled task created

The last but very important step is to remove the endless loop from the script so delete that section form the importdb.ps1 file as you can see in the Figure 9.



```
$ProcessedDir = "C:\Ozeki\Processed"
New-Item -ItemType Directory -Force -Path $ProcessedDir

#endless loop
while (1) {
        #list files older then 20 minutes
        #make sure the time is greater then the period you used in the reporting config
        Get-ChildItem "C:\Ozeki\Reporting_SQL" -Filter *.sql |
                Where{$_.LastWriteTime -le (Get-Date).AddMinutes(-20)} |

        Foreach-Object {
                #load the contents of the file into the mysql database
                Write-Output $_.FullName
                $Text = Get-Content -Path $_.FullName
                Write-Output $Text | C:\'Program Files'\MySQL\'MySQL Server 5.1'\bin\mysql -uroot -pmypass Qwer1234

                #move the processed file into the processed directory
                Move-Item -Path $_.FullName -Destination $ProcessedDir
        }

        #sleep for 5 seconds
        Start-Sleep -s 5
}
```

Figure 9 - Modify script

# How to tune the message throughput of your SMS gateway

***If you operate an SMPP service, you will often be faced with the request to limit message throughput. You may want to limit the incoming message rate, and you might have to limit the speed at which you pass messages to a certain SMS service provider. The following documents give information on how to tune message throughtput.***

**How to set an incomign speed limit for an SMPP user**

This document shows you all the configuration steps that you have to perform in the SMPP User connection to set a new Speed limit. With this manipulation you are able to slow down your message processing in that case if your Service provider is can not handle the sending speed that your license allows.

Learn More

**How to limit outgoing speed on an SMPP service provider connection**

This document shows you all the configuration steps that you have to perform in the SMPP Client connection to set a new Speed limit. With this manipulation you are able to slow down your message sending in that case if your Service provider is can not handle the sending speed that your license allows.

Learn More

**How to test the performance of your system**

If you follow the link to this document, you will be able to learn about how you can test the performacne of your SMS Gateway. To perform this test, the SMS Gateway provides you tester connection where you can set up the test environment and send a given number of SMS messages and see your MPS (Message per Sensond) which gives you indication about the performance of your system.

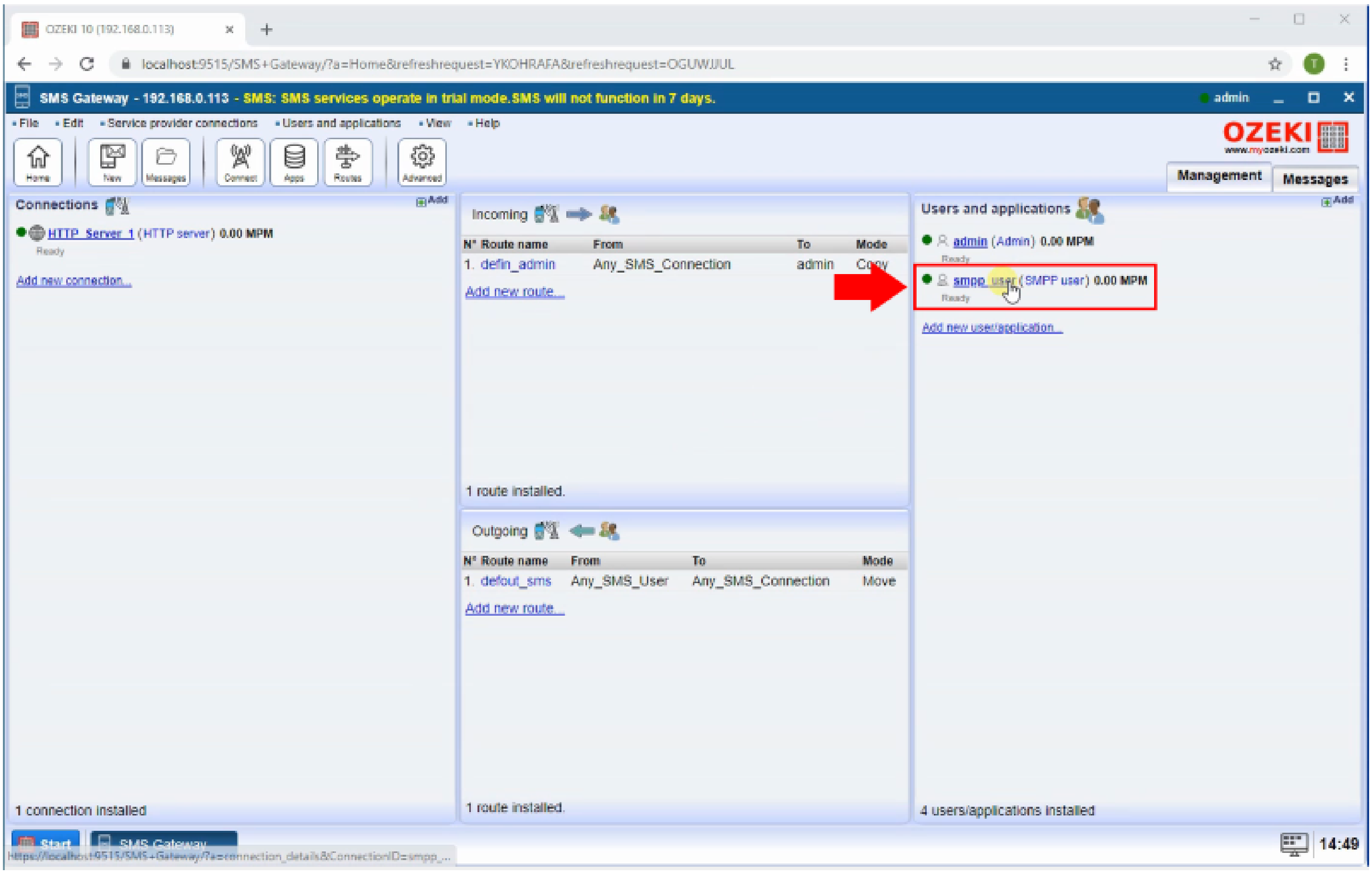
Learn More

# How to set a speed limit for incoming SMPP links

***This section is about to show you how easy you can set a speed limit for SMPP User. The document shows you all the configuration steps that you have to perform in the SMPP User connection to set a new Speed limit. With this manipulation you are able to slow down your message processing in that case if your Service provider is can not handle the sending speed that your license allows.***

## Step 1 - Open SMPP User

In order to limit you sending speed limit on the SMPP User connection your first step is to open it's configuration page. To make this click on the SMPP User connection in the SMS Gateway management console right side as the Figure 1 shows.



SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.

Connections

HTTP_Server_1 (HTTP server) 0.00 MPM
Ready

Add new connection...

1 connection installed

Incoming

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defin_admin | Any_SMS_Connection | admin | Copy |

Add new route...

1 route installed.

Outgoing

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defaul_sms | Any_SMS_User | Any_SMS_Connection | Move |

Add new route...

1 route installed.

Users and applications

admin (Admin) 0.00 MPM
Ready

smpp_user (SMPP user) 0.00 MPM
Ready

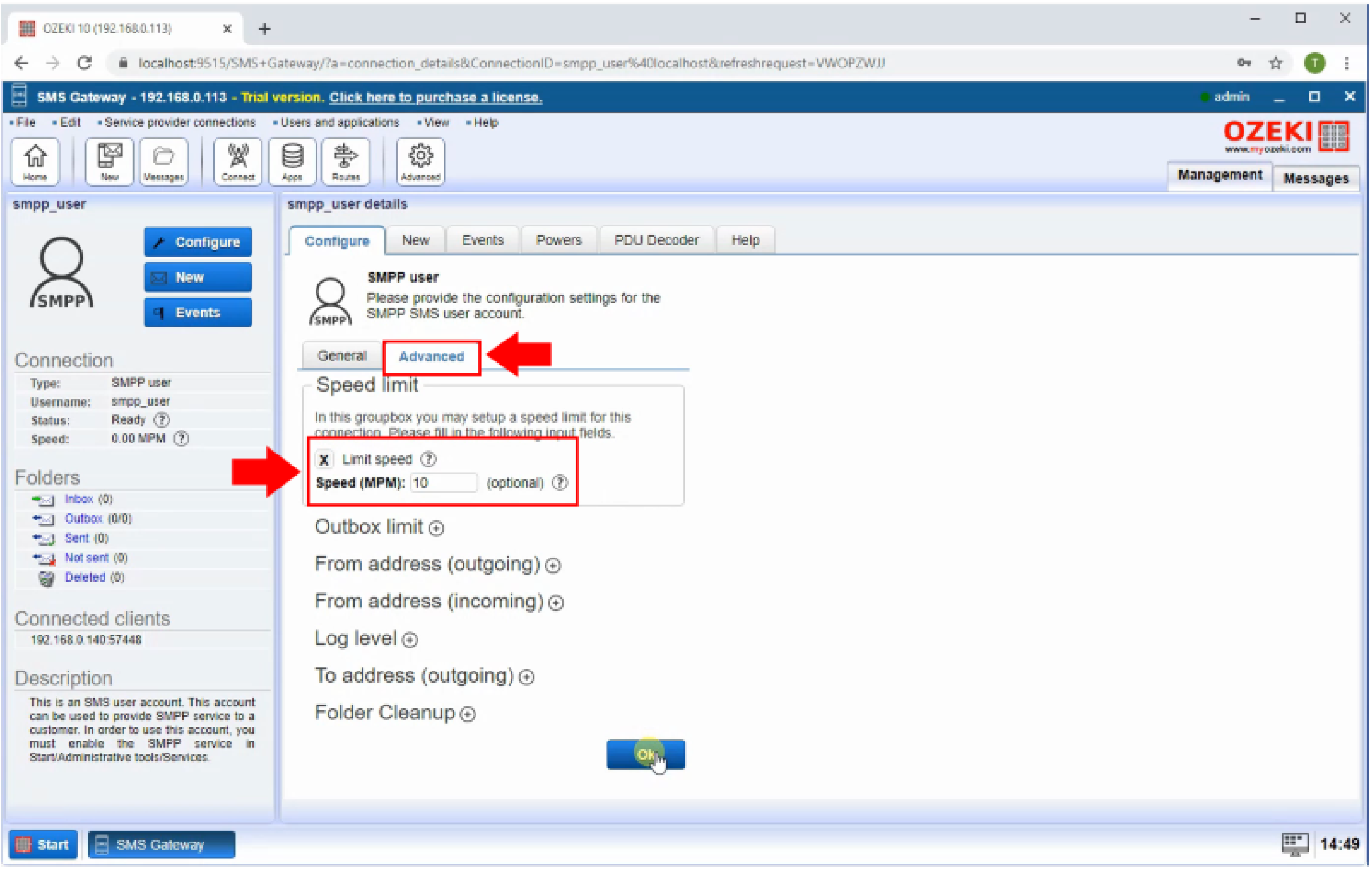
Add new user/application...

4 users/applications installed

Figure 1 - Open SMPP user

## Step 2 - Modify SMPP User speed limit

I the connection's Configuration tab select the Advanced tab and in the speed section you are able to set the maximum number of messages that can be processed per minute as you can see in the Figure 2.



SMPP user

Please provide the configuration settings for the SMPP SMS user account.

Speed limit

In this groupbox you may setup a speed limit for this connection. Please fill in the following input fields.

☒ Limit speed

Speed (MPM): 10 (optional)

Connection

Type: SMPP user
Username: smpp_user
Status: Ready
Speed: 0.00 MPM

Folders

Inbox (0)
Outbox (0/0)
Sent (0)
Not sent (0)
Deleted (0)

Connected clients

192.168.0.140:57448

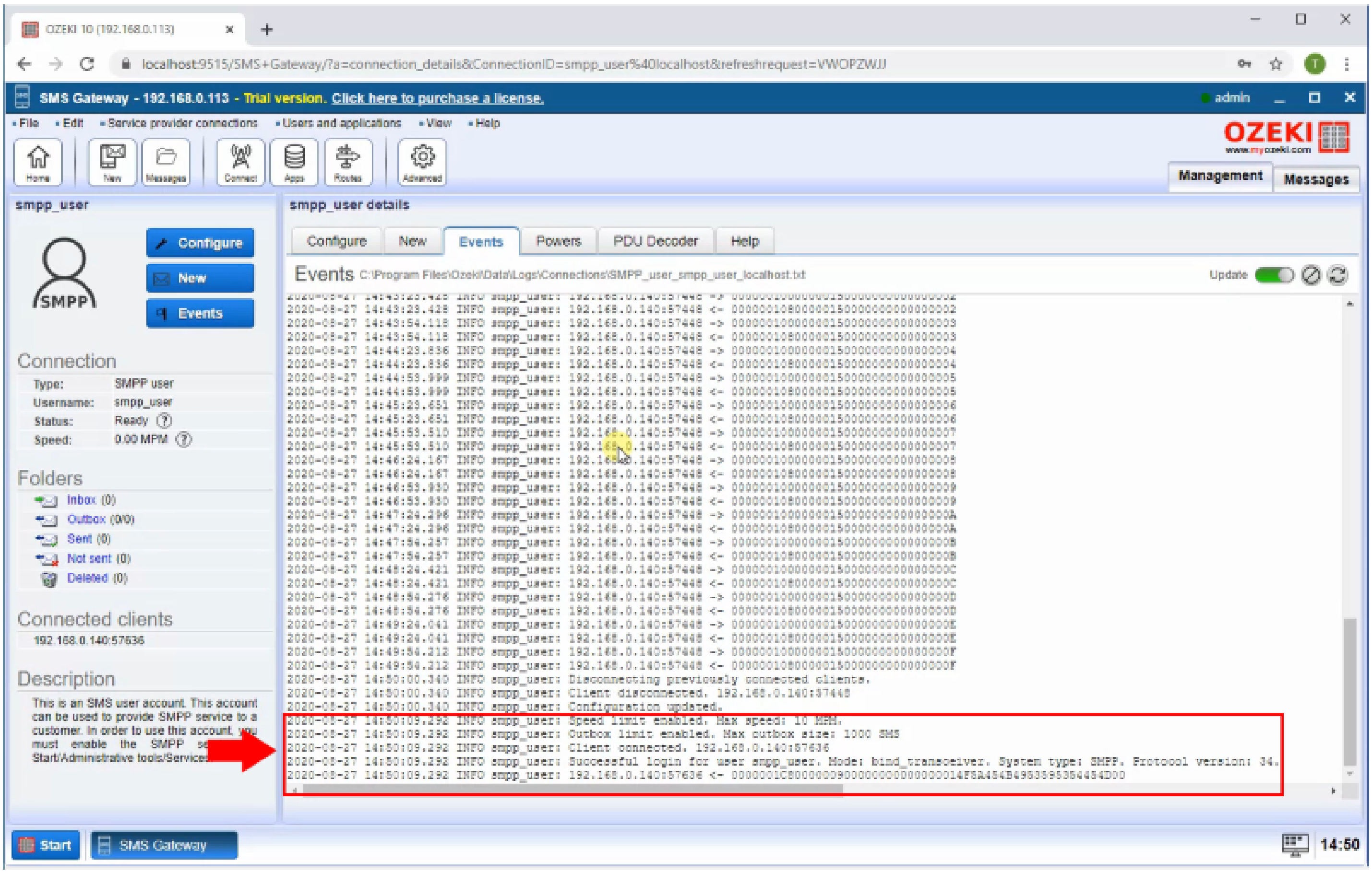
Description

This is an SMS user account. This account can be used to provide SMPP service to a customer. In order to use this account, you must enable the SMPP service in Start/Administrative tools/Services.

Figure 2 - Modify speed limit

## Step 3 - SMPP Client connected

After you configured the SMPP user you will see in the event log that the SMPP Client is connected to the SMS Gateway (Figure 3).



Connection

Type: SMPP user
Username: smpp_user
Status: Ready
Speed: 0.00 MPM

Folders

Inbox (0)
Outbox (0/0)
Sent (0)
Not sent (0)
Deleted (0)

Connected clients

192.168.0.140:57636

Description

This is an SMS user account. This account can be used to provide SMPP service to a customer. In order to use this account, you must enable the SMPP se... Start/Administrative tools/Services

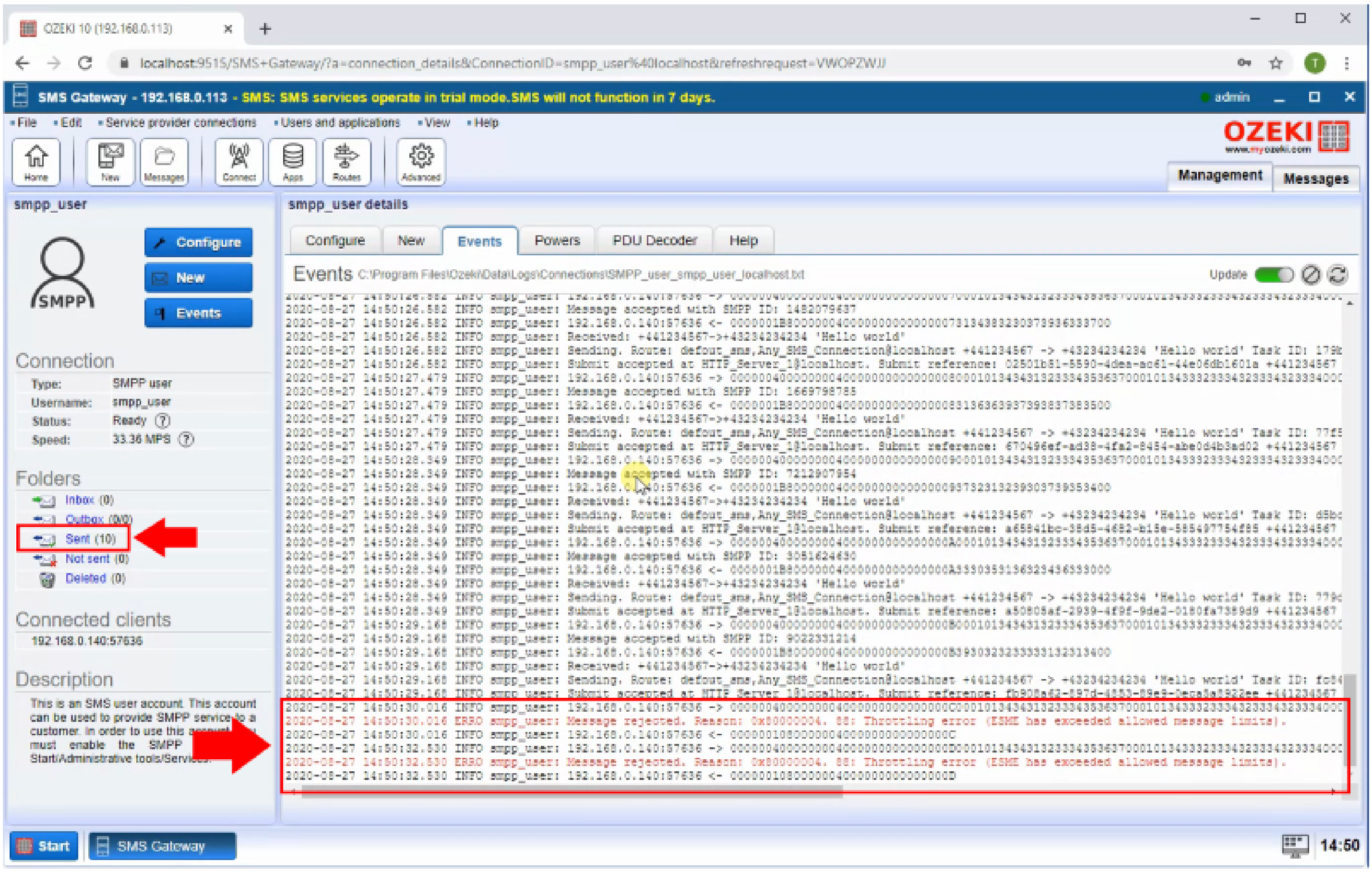
Events C:\Program Files\Ozeki\Data\Logs\Connections\SMPP_user_smpp_user_localhost.txt

```
2020-08-27 14:43:23.428 INFO smpp_user: 192.168.0.140:57448 -> 00000010000000150000000000000002
2020-08-27 14:43:23.428 INFO smpp_user: 192.168.0.140:57448 <- 00000010800000150000000000000002
2020-08-27 14:43:54.118 INFO smpp_user: 192.168.0.140:57448 -> 00000010000000150000000000000003
2020-08-27 14:43:54.118 INFO smpp_user: 192.168.0.140:57448 <- 00000010800000150000000000000003
2020-08-27 14:44:23.836 INFO smpp_user: 192.168.0.140:57448 -> 00000010000000150000000000000004
2020-08-27 14:44:23.836 INFO smpp_user: 192.168.0.140:57448 <- 00000010800000150000000000000004
2020-08-27 14:44:53.999 INFO smpp_user: 192.168.0.140:57448 -> 00000010000000150000000000000005
2020-08-27 14:44:53.999 INFO smpp_user: 192.168.0.140:57448 <- 00000010800000150000000000000005
2020-08-27 14:45:23.651 INFO smpp_user: 192.168.0.140:57448 -> 00000010000000150000000000000006
2020-08-27 14:45:23.651 INFO smpp_user: 192.168.0.140:57448 <- 00000010800000150000000000000006
2020-08-27 14:45:53.510 INFO smpp_user: 192.168.0.140:57448 -> 00000010000000150000000000000007
2020-08-27 14:45:53.510 INFO smpp_user: 192.168.0.140:57448 <- 00000010800000150000000000000007
2020-08-27 14:46:24.167 INFO smpp_user: 192.168.0.140:57448 -> 00000010000000150000000000000008
2020-08-27 14:46:24.167 INFO smpp_user: 192.168.0.140:57448 <- 00000010800000150000000000000008
2020-08-27 14:46:53.930 INFO smpp_user: 192.168.0.140:57448 -> 00000010000000150000000000000009
2020-08-27 14:46:53.930 INFO smpp_user: 192.168.0.140:57448 <- 00000010800000150000000000000009
2020-08-27 14:47:24.296 INFO smpp_user: 192.168.0.140:57448 -> 0000001000000015000000000000000A
2020-08-27 14:47:24.296 INFO smpp_user: 192.168.0.140:57448 <- 0000001080000015000000000000000A
2020-08-27 14:47:54.257 INFO smpp_user: 192.168.0.140:57448 -> 0000001000000015000000000000000B
2020-08-27 14:47:54.257 INFO smpp_user: 192.168.0.140:57448 <- 0000001080000015000000000000000B
2020-08-27 14:48:24.421 INFO smpp_user: 192.168.0.140:57448 -> 0000001000000015000000000000000C
2020-08-27 14:48:24.421 INFO smpp_user: 192.168.0.140:57448 <- 0000001080000015000000000000000C
2020-08-27 14:48:54.276 INFO smpp_user: 192.168.0.140:57448 -> 0000001000000015000000000000000D
2020-08-27 14:48:54.276 INFO smpp_user: 192.168.0.140:57448 <- 0000001080000015000000000000000D
2020-08-27 14:49:24.041 INFO smpp_user: 192.168.0.140:57448 -> 0000001000000015000000000000000E
2020-08-27 14:49:24.041 INFO smpp_user: 192.168.0.140:57448 <- 0000001080000015000000000000000E
2020-08-27 14:49:54.212 INFO smpp_user: 192.168.0.140:57448 -> 0000001000000015000000000000000F
2020-08-27 14:49:54.212 INFO smpp_user: 192.168.0.140:57448 <- 0000001080000015000000000000000F
2020-08-27 14:50:00.340 INFO smpp_user: Disconnecting previously connected clients.
2020-08-27 14:50:00.340 INFO smpp_user: Client disconnected. 192.168.0.140:57448
2020-08-27 14:50:00.340 INFO smpp_user: Configuration updated.
2020-08-27 14:50:09.292 INFO smpp_user: Speed limit enabled. Max speed: 10 MPM.
2020-08-27 14:50:09.292 INFO smpp_user: Outbox limit enabled. Max outbox size: 1000 SMS
2020-08-27 14:50:09.292 INFO smpp_user: Client connected. 192.168.0.140:57636
2020-08-27 14:50:09.292 INFO smpp_user: Successful login for user smpp_user. Mode: bind_transceiver. System type: SMPP. Protocol version: 34.
2020-08-27 14:50:09.292 INFO smpp_user: 192.168.0.140:57636 <- 0000001C8000000900000000000000014F5A454B4953535354454D00
```

Figure 3 - Client connected

## Step 4 - Throttling error when speed limit is exceeded

When the received messages number are exceeded the speed limit the SMS Gateway send back a Throttling error to the client. From the Throttling error, the Client will know that it needs to reduce the sending speed or wait for additional messages to be sent.



Connection

Type: SMPP user
Username: smpp_user
Status: Ready
Speed: 33.36 MPS

Folders

Inbox (0)
Outbox (0/0)
Sent (10)
Not sent (0)
Deleted (0)

Connected clients

192.168.0.140:57636

Description

This is an SMS user account. This account can be used to provide SMPP service to a customer. In order to use this account must enable the SMPP Start/Administrative tools/Servi

Events C:\Program Files\Ozeki\Data\Logs\Connections\SMPP_user_smpp_user_localhost.txt

```
2020-08-27 14:50:26.582 INFO smpp_user: 192.168.0.140:57636 -> 00000040000000040000000000000007000101343431323334353637000101343332333432333343233340
2020-08-27 14:50:26.582 INFO smpp_user: Message accepted with SMPP ID: 1482079637
2020-08-27 14:50:26.582 INFO smpp_user: 192.168.0.140:57636 <- 0000001B800000040000000000000007313438323037393633370
2020-08-27 14:50:26.582 INFO smpp_user: Received: +441234567->+43234234234 'Hello world'
2020-08-27 14:50:26.582 INFO smpp_user: Sending. Route: defout_sms,Any_SMS_Connection@localhost +441234567 -> +43234234234 'Hello world' Task ID: 179b
2020-08-27 14:50:26.582 INFO smpp_user: Submit accepted at HTTP_Server_1@localhost. Submit reference: 02501b51-5590-4dea-ac61-44e06db1601a +441234567
2020-08-27 14:50:27.479 INFO smpp_user: 192.168.0.140:57636 -> 000000400000000400000000000000080001013434313233343536370001013433323334323333432333400C
2020-08-27 14:50:27.479 INFO smpp_user: Message accepted with SMPP ID: 1669798785
2020-08-27 14:50:27.479 INFO smpp_user: 192.168.0.140:57636 <- 0000001B800000040000000000000008313636393739383738353500
2020-08-27 14:50:27.479 INFO smpp_user: Received: +441234567->+43234234234 'Hello world'
2020-08-27 14:50:27.479 INFO smpp_user: Sending. Route: defout_sms,Any_SMS_Connection@localhost +441234567 -> +43234234234 'Hello world' Task ID: 77f5
2020-08-27 14:50:27.479 INFO smpp_user: Submit accepted at HTTP_Server_1@localhost. Submit reference: 670496ef-ad38-4fa2-8454-abe0d4b3ad02 +441234567
2020-08-27 14:50:28.349 INFO smpp_user: 192.168.0.140:57636 -> 000000400000000400000000000000090001013434313233343536370001013433323334323333432333400C
2020-08-27 14:50:28.349 INFO smpp_user: Message accepted with SMPP ID: 7212907954
2020-08-27 14:50:28.349 INFO smpp_user: 192.168.0.140:57636 <- 0000001B800000040000000000000009373231323930373935340 0
2020-08-27 14:50:28.349 INFO smpp_user: Received: +441234567->+43234234234 'Hello world'
2020-08-27 14:50:28.349 INFO smpp_user: Sending. Route: defout_sms,Any_SMS_Connection@localhost +441234567 -> +43234234234 'Hello world' Task ID: d5bc
2020-08-27 14:50:28.349 INFO smpp_user: Submit accepted at HTTP_Server_1@localhost. Submit reference: a65841bc-38d5-4682-b15e-585497754f85 +441234567
2020-08-27 14:50:28.349 INFO smpp_user: 192.168.0.140:57636 -> 00000040000000040000000000000000A0001013434313233343536370001013433323334323333432333400C
2020-08-27 14:50:28.349 INFO smpp_user: Message accepted with SMPP ID: 3051624630
2020-08-27 14:50:28.349 INFO smpp_user: 192.168.0.140:57636 <- 0000001B80000004000000000000000A33303531363234363330 00
2020-08-27 14:50:28.349 INFO smpp_user: Received: +441234567->+43234234234 'Hello world'
2020-08-27 14:50:28.349 INFO smpp_user: Sending. Route: defout_sms,Any_SMS_Connection@localhost +441234567 -> +43234234234 'Hello world' Task ID: 779c
2020-08-27 14:50:28.349 INFO smpp_user: Submit accepted at HTTP_Server_1@localhost. Submit reference: a50805af-2939-4f9f-9de2-0180fa7389d9 +441234567
2020-08-27 14:50:29.168 INFO smpp_user: 192.168.0.140:57636 -> 0000004000000004000000000000000B0001013434313233343536370001013433323334323333432333400C
2020-08-27 14:50:29.168 INFO smpp_user: Message accepted with SMPP ID: 9022331214
2020-08-27 14:50:29.168 INFO smpp_user: 192.168.0.140:57636 <- 0000001B80000004000000000000000B39303232333331323134 00
2020-08-27 14:50:29.168 INFO smpp_user: Received: +441234567->+43234234234 'Hello world'
2020-08-27 14:50:29.168 INFO smpp_user: Sending. Route: defout_sms,Any_SMS_Connection@localhost +441234567 -> +43234234234 'Hello world' Task ID: fc84
2020-08-27 14:50:29.168 INFO smpp_user: Submit accepted at HTTP_Server_1@localhost. Submit reference: fb908a42-897d-4853-89e9-0eca8a8922ee +441234567
2020-08-27 14:50:30.016 INFO smpp_user: 192.168.0.140:57636 -> 0000004000000004000000000000000C0001013434313233343536370001013433323334323333432333400C
2020-08-27 14:50:30.016 ERRO smpp_user: Message rejected. Reason: 0x80000004. 88: Throttling error (ESME has exceeded allowed message limits).
2020-08-27 14:50:30.016 INFO smpp_user: 192.168.0.140:57636 <- 00000010800000040000000000000000C
2020-08-27 14:50:32.530 INFO smpp_user: 192.168.0.140:57636 -> 0000004000000004000000000000000D0001013434313233343536370001013433323334323333432333400C
2020-08-27 14:50:32.530 ERRO smpp_user: Message rejected. Reason: 0x80000004. 88: Throttling error (ESME has exceeded allowed message limits).
2020-08-27 14:50:32.530 INFO smpp_user: 192.168.0.140:57636 <- 00000010800000040000000000000000D
```

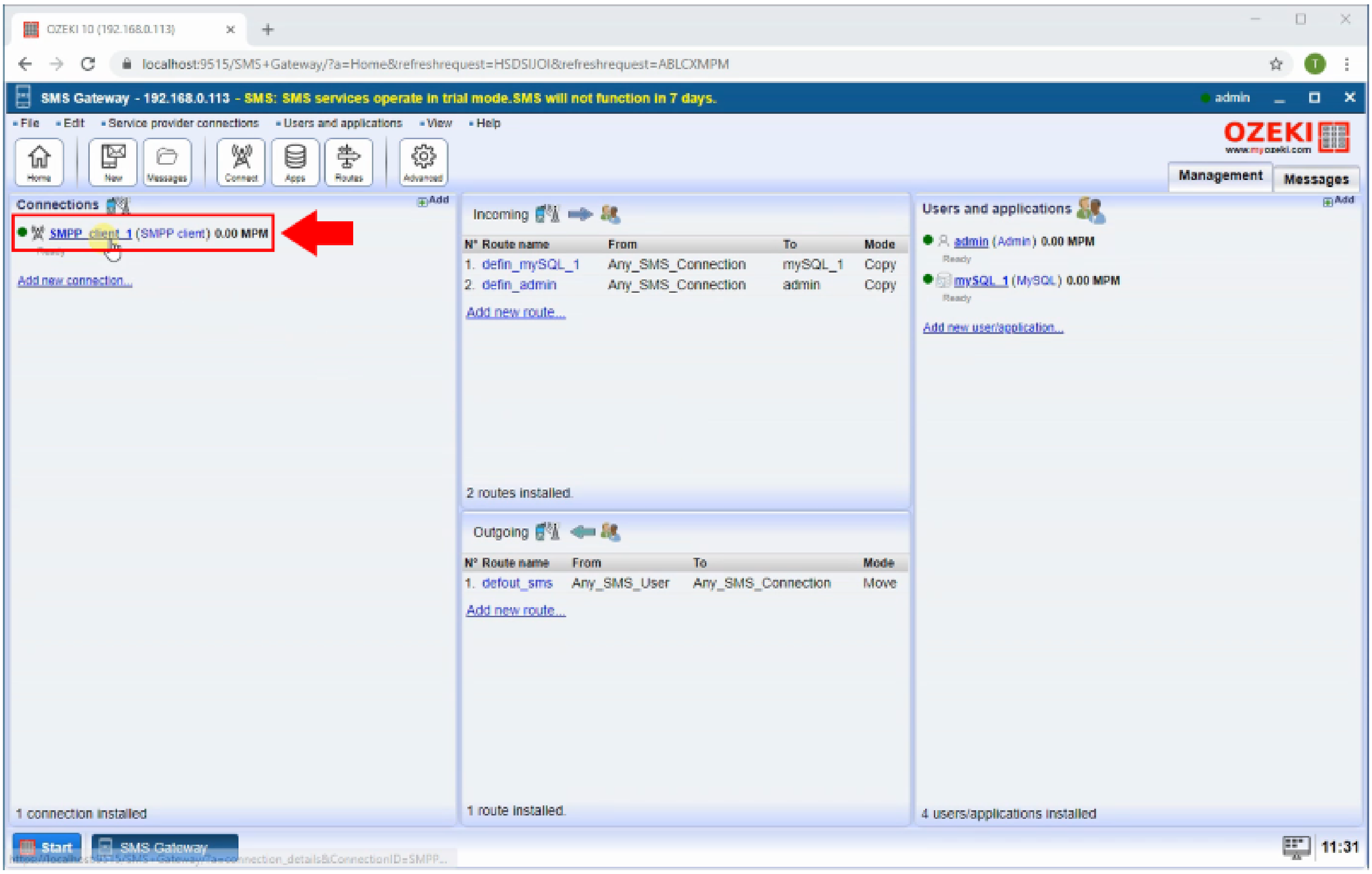
Figure 4 - Throttling error

# How to set a speed limit for outgoing SMPP links

***This section is about to show you how easy you can set a speed limit for SMPP Client connection. The document shows you all the configuration steps that you have to perform in the SMPP Client connection to set a new Speed limit. With this manipulation you are able to slow down your message sending in that case if your Service provider is can not handle the sending speed that your license allows.***

## Step 1 - Open SMPP Client connection

In order to limit you sending speed limit on the SMPP client connection your first step is to open the SMPP client connection Configuration page. To make this click on the SMPP client connection in the SMS Gateway management console left side as the Figure 1 shows.



localhost:9515/SMS+Gateway/?a=Home&refreshrequest=HSDSIJOI&refreshrequest=ABLCXMPM

SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.

Connections

SMPP_client_1 (SMPP client) 0.00 MPM

Ready

Add new connection...

1 connection installed

Incoming

| N° Route name | From | To | Mode |
|---|---|---|---|
| 1. defin_mySQL_1 | Any_SMS_Connection | mySQL_1 | Copy |
| 2. defin_admin | Any_SMS_Connection | admin | Copy |

Add new route...

2 routes installed.

Outgoing

| N° Route name | From | To | Mode |
|---|---|---|---|
| 1. defout_sms | Any_SMS_User | Any_SMS_Connection | Move |

Add new route...

1 route installed.

Users and applications

admin (Admin) 0.00 MPM

Ready

mySQL_1 (MySQL) 0.00 MPM

Ready

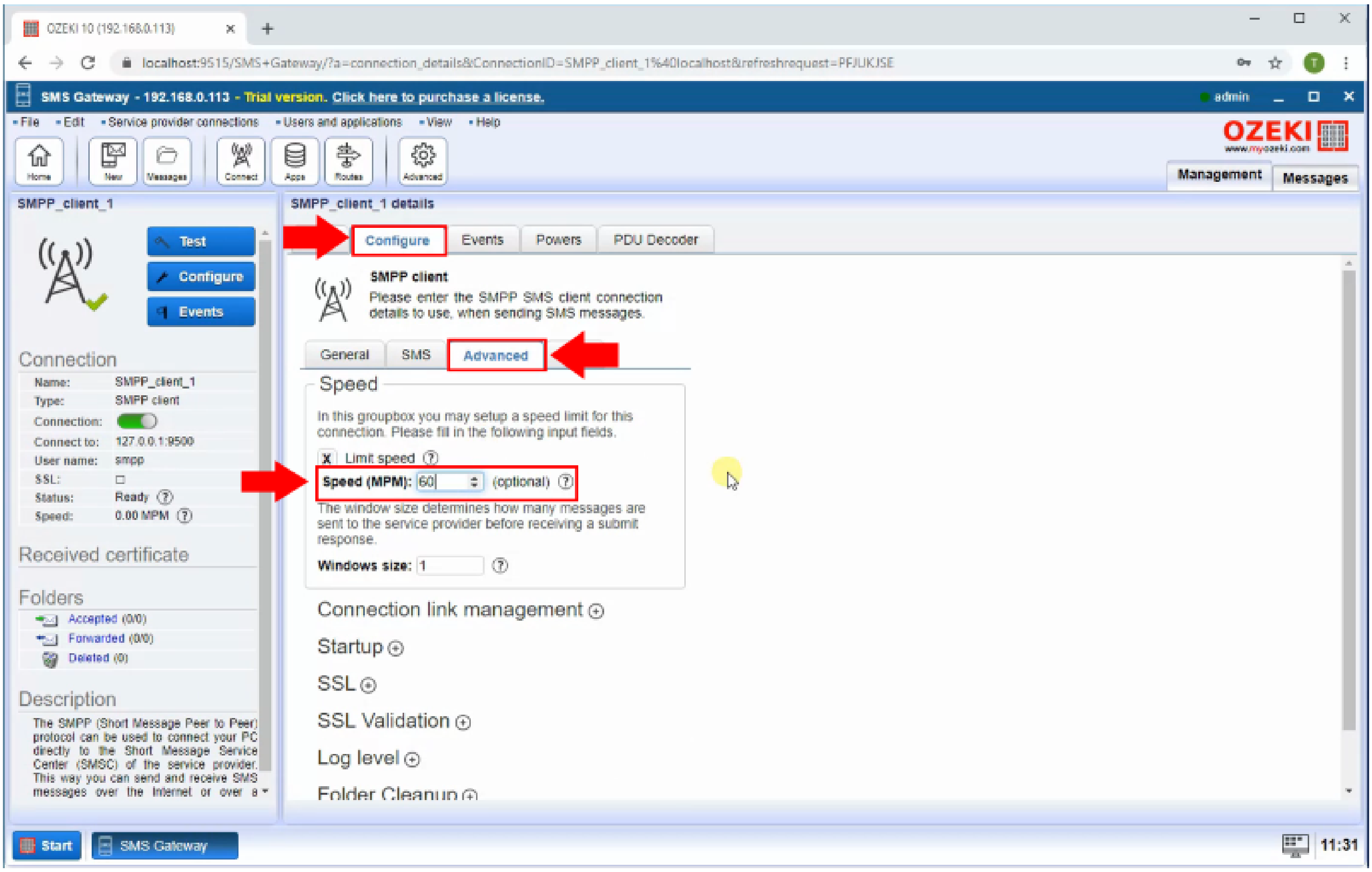
Add new user/application...

4 users/applications installed

Figure 1 - Open SMPP client connection

## Step 2 - Modify sending speed limit

I the connection's Configuration tab select the Advanced tab and in the speed section you are able to set the maximum number of messages that can be sent per minute as you can see in the Figure 2.



Connection

Name: SMPP_client_1

Type: SMPP client

Connection: ●

Connect to: 127.0.0.1:9500

User name: smpp

SSL: □

Status: Ready

Speed: 0.00 MPM

Received certificate

Folders

Accepted (0/0)

Forwarded (0/0)

Deleted (0)

Description

The SMPP (Short Message Peer to Peer) protocol can be used to connect your PC directly to the Short Message Service Center (SMSC) of the service provider. This way you can send and receive SMS messages over the Internet or over a

SMPP client

Please enter the SMPP SMS client connection details to use, when sending SMS messages.

Speed

In this groupbox you may setup a speed limit for this connection. Please fill in the following input fields.

☒ Limit speed

Speed (MPM): 60 (optional)

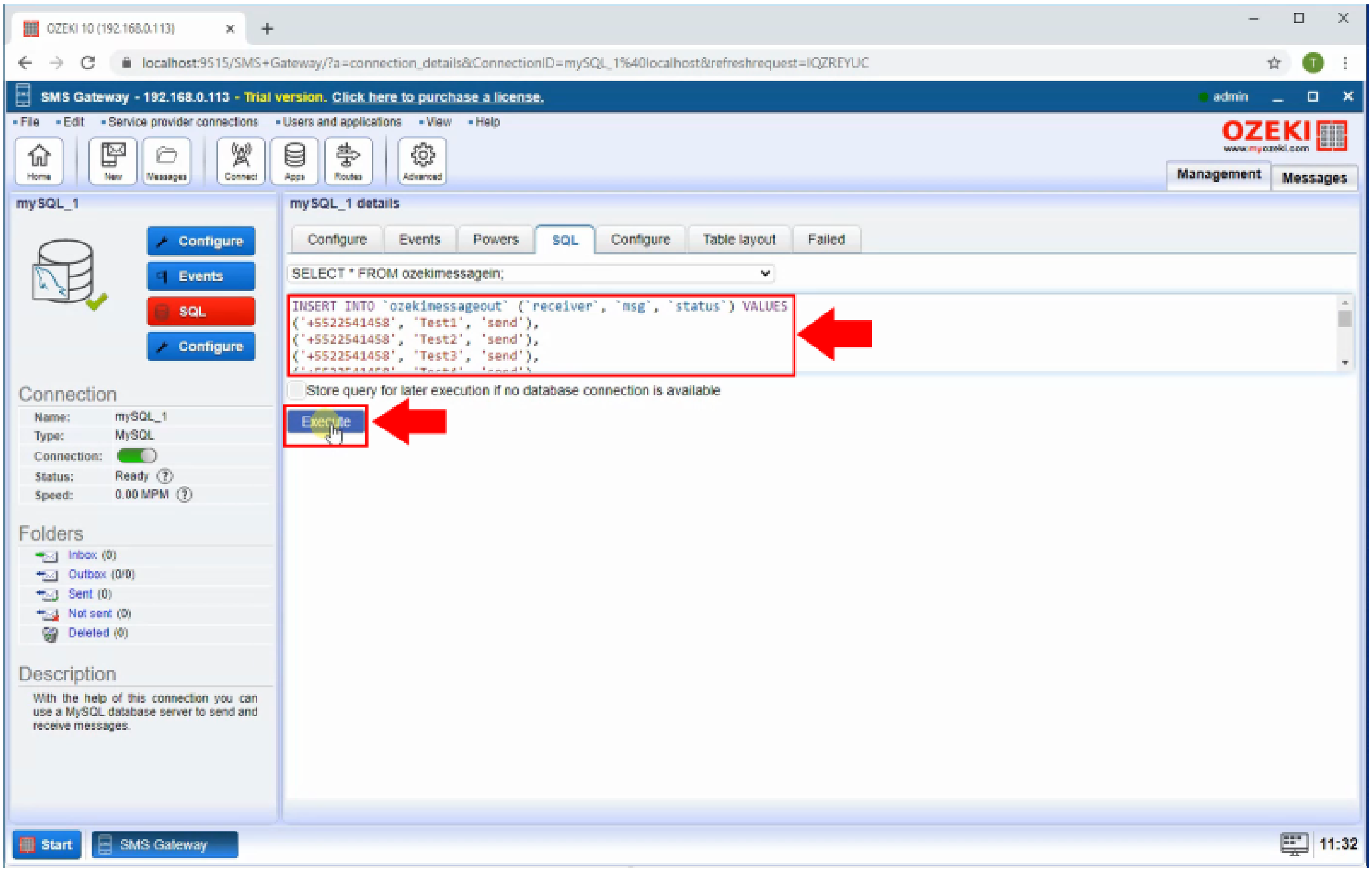
The window size determines how many messages are sent to the service provider before receiving a submit response.

Windows size: 1

Figure 2 - Modify sending speed limit

# Step 3 - Send message to the SMS Gateway

After you set the speed limit to test it send messages to the SMS Gateway. In this example, we use a database connection. With one query, we send 60 messages to the SMS Gateway (Figure 3).



SELECT * FROM ozekimessagein;

```
INSERT INTO `ozekimessageout` (`receiver`, `msg`, `status`) VALUES
('+5522541458', 'Test1', 'send'),
('+5522541458', 'Test2', 'send'),
('+5522541458', 'Test3', 'send'),
```

□ Store query for later execution if no database connection is available

Connection

Name: mySQL_1
Type: MySQL
Connection:
Status: Ready
Speed: 0.00 MPM

Folders

Inbox (0)
Outbox (0/0)
Sent (0)
Not sent (0)
Deleted (0)

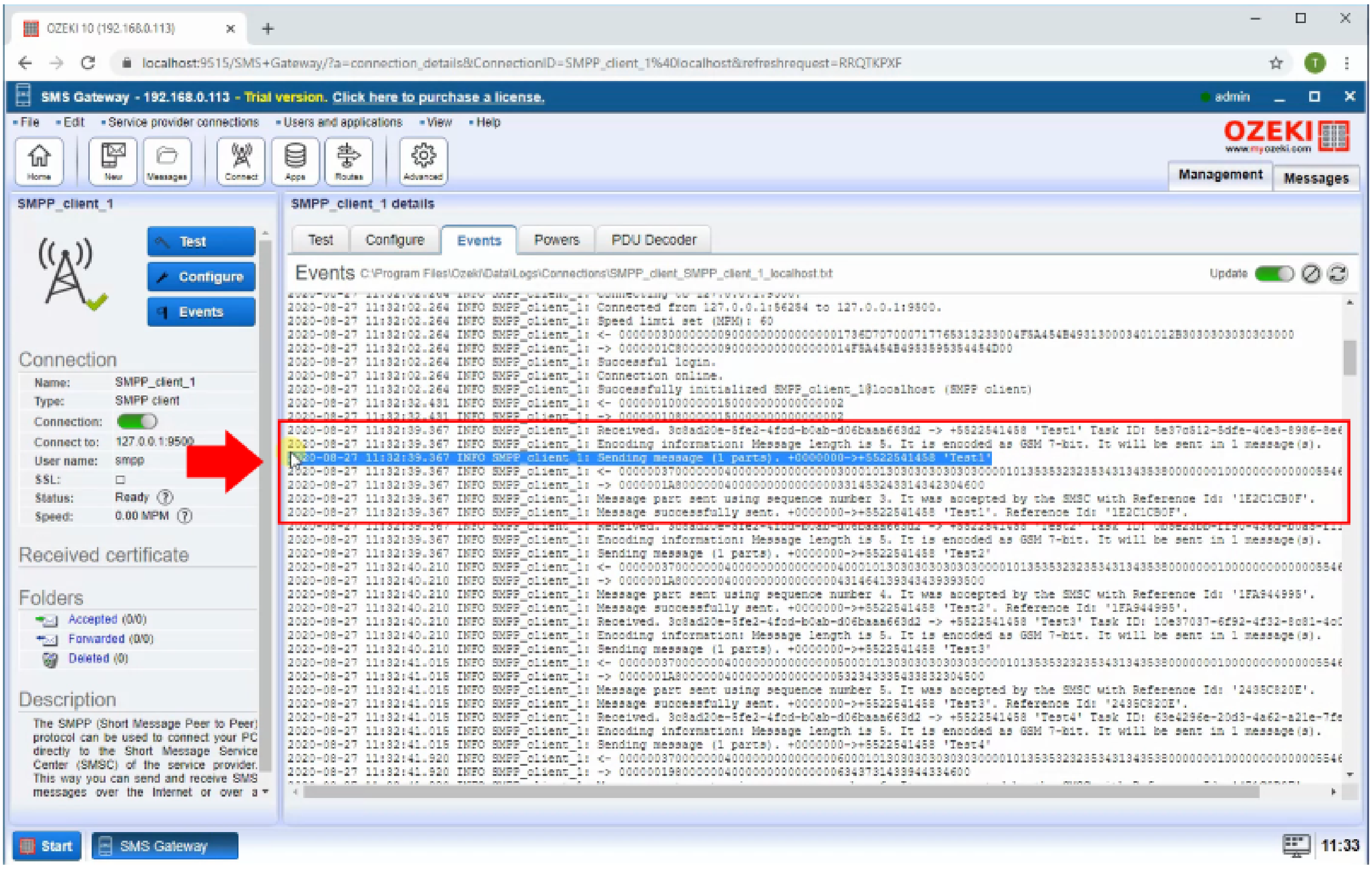
Description

With the help of this connection you can use a MySQL database server to send and receive messages.

Figure 3 - Send message to the SMS gateway

# Step 4 - Message time stamps

Finally you will see that the messages are sent according the speed limit. In this example the time between the first and last message is one minute because the speed limit is 60 Messages Per Minute. You can check the messagees in the SMPP connections Event tab az you can see in the Fifure 4 and Figure 5.


OZEKI 10 (192.168.0.113)
localhost:9515/SMS+Gateway/?a=connection_details&ConnectionID=SMPP_client_1%40localhost&refreshrequest=RRQTKPXF
SMS Gateway - 192.168.0.113 - Trial version. Click here to purchase a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
OZEKI
Management
Messages
Test
Configure
Events
Powers
PDU Decoder
Update
Start
SMS Gateway
11:33

SMPP_client_1

SMPP_client_1 details

Connection

Name: SMPP_client_1
Type: SMPP client
Connection:
Connect to: 127.0.0.1:9500
User name: smpp
SSL: □
Status: Ready
Speed: 0.00 MPM

Received certificate

Folders

Accepted (0/0)
Forwarded (0/0)
Deleted (0)

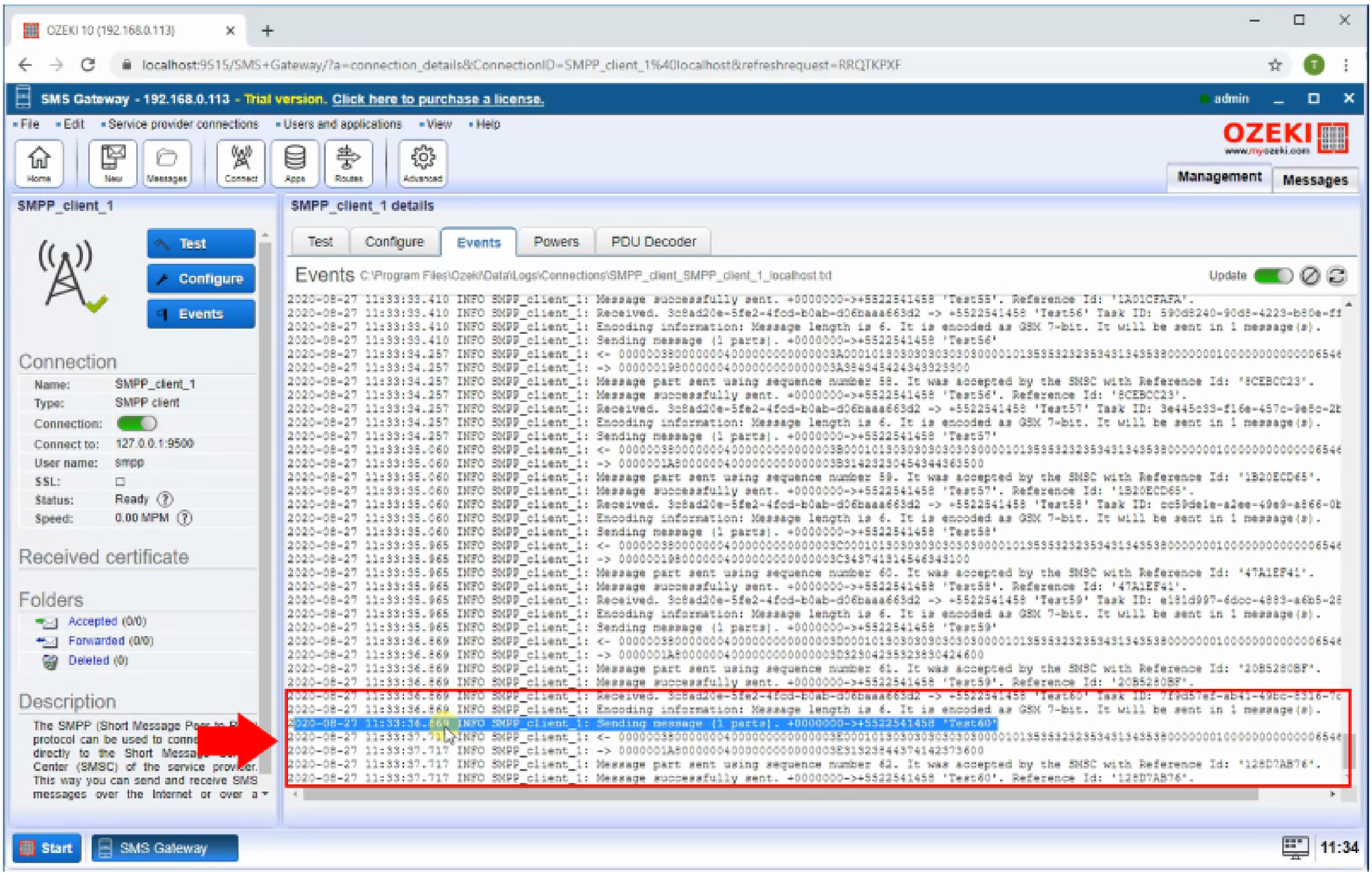
Description

The SMPP (Short Message Peer to Peer) protocol can be used to connect your PC directly to the Short Message Service Center (SMSC) of the service provider. This way you can send and receive SMS messages over the Internet or over a

Events C:\Program Files\Ozeki\Data\Logs\Connections\SMPP_client_SMPP_client_1_localhost.txt

```
2020-08-27 11:32:02.264 INFO SMPP_client_1: Connecting to 127.0.0.1:9500.
2020-08-27 11:32:02.264 INFO SMPP_client_1: Connected from 127.0.0.1:56284 to 127.0.0.1:9500.
2020-08-27 11:32:02.264 INFO SMPP_client_1: Speed limti set (MPM): 60
2020-08-27 11:32:02.264 INFO SMPP_client_1: <- 000000300000000900000000000000017360707000717765313233004F5A454B49313000340101 2B30303030303030000
2020-08-27 11:32:02.264 INFO SMPP_client_1: -> 0000001C80000009000000000000000014F5A454B49535953454454D00
2020-08-27 11:32:02.264 INFO SMPP_client_1: Successful login.
2020-08-27 11:32:02.264 INFO SMPP_client_1: Connection online.
2020-08-27 11:32:02.264 INFO SMPP_client_1: Successfully initialized SMPP_client_1@localhost (SMPP client)
2020-08-27 11:32:32.431 INFO SMPP_client_1: <- 00000010000000150000000000000002
2020-08-27 11:32:32.431 INFO SMPP_client_1: -> 00000010800000150000000000000002
2020-08-27 11:32:39.367 INFO SMPP_client_1: Received. 3c8ad20e-5fe2-4fcd-b0ab-d06baaa663d2 -> +5522541458 'Test1' Task ID: 5e37c512-5dfe-40e3-8986-8e6
2020-08-27 11:32:39.367 INFO SMPP_client_1: Encoding information: Message length is 5. It is encoded as GSM 7-bit. It will be sent in 1 message(s).
2020-08-27 11:32:39.367 INFO SMPP_client_1: Sending message (1 parts). +0000000->+5522541458 'Test1'
2020-08-27 11:32:39.367 INFO SMPP_client_1: <- 00000037000000040000000000000003000101303030303030300000101353532323534313435380000000010000000000005546
2020-08-27 11:32:39.367 INFO SMPP_client_1: -> 0000001A800000040000000000000003314532433143423046 00
2020-08-27 11:32:39.367 INFO SMPP_client_1: Message part sent using sequence number 3. It was accepted by the SMSC with Reference Id: '1E2C1CB0F'.
2020-08-27 11:32:39.367 INFO SMPP_client_1: Message successfully sent. +0000000->+5522541458 'Test1'. Reference Id: '1E2C1CB0F'.
2020-08-27 11:32:39.367 INFO SMPP_client_1: Received. 3c8ad20e-5fe2-4fcd-b0ab-d06baaa663d2 -> +5522541458 'Test2' Task ID: 0b8e23bb-1190-436d-b0a8-1
2020-08-27 11:32:39.367 INFO SMPP_client_1: Encoding information: Message length is 5. It is encoded as GSM 7-bit. It will be sent in 1 message(s).
2020-08-27 11:32:39.367 INFO SMPP_client_1: Sending message (1 parts). +0000000->+5522541458 'Test2'
2020-08-27 11:32:40.210 INFO SMPP_client_1: <- 00000037000000040000000000000004000101303030303030300000101353532323534313435380000000010000000000005546
2020-08-27 11:32:40.210 INFO SMPP_client_1: -> 0000001A80000004000000000000000431464139343439393500
2020-08-27 11:32:40.210 INFO SMPP_client_1: Message part sent using sequence number 4. It was accepted by the SMSC with Reference Id: '1FA944995'.
2020-08-27 11:32:40.210 INFO SMPP_client_1: Message successfully sent. +0000000->+5522541458 'Test2'. Reference Id: '1FA944995'.
2020-08-27 11:32:40.210 INFO SMPP_client_1: Received. 3c8ad20e-5fe2-4fcd-b0ab-d06baaa663d2 -> +5522541458 'Test3' Task ID: 10e37037-6f92-4f32-8c81-4c0
2020-08-27 11:32:40.210 INFO SMPP_client_1: Encoding information: Message length is 5. It is encoded as GSM 7-bit. It will be sent in 1 message(s).
2020-08-27 11:32:40.210 INFO SMPP_client_1: Sending message (1 parts). +0000000->+5522541458 'Test3'
2020-08-27 11:32:41.015 INFO SMPP_client_1: <- 00000037000000040000000000000005000101303030303030300000101353532323534313435380000000010000000000005546
2020-08-27 11:32:41.015 INFO SMPP_client_1: -> 0000001A80000004000000000000000532343335433832304500
2020-08-27 11:32:41.015 INFO SMPP_client_1: Message part sent using sequence number 5. It was accepted by the SMSC with Reference Id: '2435C820E'.
2020-08-27 11:32:41.015 INFO SMPP_client_1: Message successfully sent. +0000000->+5522541458 'Test3'. Reference Id: '2435C820E'.
2020-08-27 11:32:41.015 INFO SMPP_client_1: Received. 3c8ad20e-5fe2-4fcd-b0ab-d06baaa663d2 -> +5522541458 'Test4' Task ID: 63e4296e-20d3-4a62-a21e-7fe
2020-08-27 11:32:41.015 INFO SMPP_client_1: Encoding information: Message length is 5. It is encoded as GSM 7-bit. It will be sent in 1 message(s).
2020-08-27 11:32:41.015 INFO SMPP_client_1: Sending message (1 parts). +0000000->+5522541458 'Test4'
2020-08-27 11:32:41.920 INFO SMPP_client_1: <- 00000037000000040000000000000006000101303030303030300000101353532323534313435380000000010000000000005546
2020-08-27 11:32:41.920 INFO SMPP_client_1: -> 000000198000000400000000000000063437314339443346 00
```

Figure 4 - First message sent time stamp


OZEKI 10 (192.168.0.113)
localhost:9515/SMS+Gateway/?a=connection_details&ConnectionID=SMPP_client_1%40localhost&refreshrequest=RRQTKPXF
SMS Gateway - 192.168.0.113 - Trial version. Click here to purchase a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
OZEKI
Management
Messages
Test
Configure
Events
Powers
PDU Decoder
Update
Start
SMS Gateway
11:34

SMPP_client_1

SMPP_client_1 details

Connection

Name: SMPP_client_1
Type: SMPP client
Connection:
Connect to: 127.0.0.1:9500
User name: smpp
SSL: □
Status: Ready
Speed: 0.00 MPM

Received certificate

Folders

Accepted (0/0)
Forwarded (0/0)
Deleted (0)

Description

The SMPP (Short Message Peer to Peer) protocol can be used to connect your PC directly to the Short Message Service Center (SMSC) of the service provider. This way you can send and receive SMS messages over the Internet or over a

Events C:\Program Files\Ozeki\Data\Logs\Connections\SMPP_client_SMPP_client_1_localhost.txt

```
2020-08-27 11:33:33.410 INFO SMPP_client_1: Message successfully sent. +0000000->+5522541458 'Test55'. Reference Id: '1A01CFAFA'.
2020-08-27 11:33:33.410 INFO SMPP_client_1: Received. 3c8ad20e-5fe2-4fcd-b0ab-d06baaa663d2 -> +5522541458 'Test56' Task ID: 590d8240-90d5-4223-b60e-f1
2020-08-27 11:33:33.410 INFO SMPP_client_1: Encoding information: Message length is 6. It is encoded as GSM 7-bit. It will be sent in 1 message(s).
2020-08-27 11:33:33.410 INFO SMPP_client_1: Sending message (1 parts). +0000000->+5522541458 'Test56'
2020-08-27 11:33:34.257 INFO SMPP_client_1: <- 0000003800000004000000000000003A000101303030303030300000101353532323534313435380000000010000000000006546
2020-08-27 11:33:34.257 INFO SMPP_client_1: -> 000000198000000400000000000003A38434542434332333 00
2020-08-27 11:33:34.257 INFO SMPP_client_1: Message part sent using sequence number 58. It was accepted by the SMSC with Reference Id: '8CEBCC23'.
2020-08-27 11:33:34.257 INFO SMPP_client_1: Message successfully sent. +0000000->+5522541458 'Test56'. Reference Id: '8CEBCC23'.
2020-08-27 11:33:34.257 INFO SMPP_client_1: Received. 3c8ad20e-5fe2-4fcd-b0ab-d06baaa663d2 -> +5522541458 'Test57' Task ID: 3e445c33-f16e-457c-9e5c-2b
2020-08-27 11:33:34.257 INFO SMPP_client_1: Encoding information: Message length is 6. It is encoded as GSM 7-bit. It will be sent in 1 message(s).
2020-08-27 11:33:34.257 INFO SMPP_client_1: Sending message (1 parts). +0000000->+5522541458 'Test57'
2020-08-27 11:33:35.060 INFO SMPP_client_1: <- 0000003800000004000000000000003B000101303030303030300000101353532323534313435380000000010000000000006546
2020-08-27 11:33:35.060 INFO SMPP_client_1: -> 0000001A8000000400000000000003B31423230454344363500
2020-08-27 11:33:35.060 INFO SMPP_client_1: Message part sent using sequence number 59. It was accepted by the SMSC with Reference Id: '1B20ECD65'.
2020-08-27 11:33:35.060 INFO SMPP_client_1: Message successfully sent. +0000000->+5522541458 'Test57'. Reference Id: '1B20ECD65'.
2020-08-27 11:33:35.060 INFO SMPP_client_1: Received. 3c8ad20e-5fe2-4fcd-b0ab-d06baaa663d2 -> +5522541458 'Test58' Task ID: cc59de1e-a2ee-49e9-a566-0b
2020-08-27 11:33:35.060 INFO SMPP_client_1: Encoding information: Message length is 6. It is encoded as GSM 7-bit. It will be sent in 1 message(s).
2020-08-27 11:33:35.060 INFO SMPP_client_1: Sending message (1 parts). +0000000->+5522541458 'Test58'
2020-08-27 11:33:35.965 INFO SMPP_client_1: <- 0000003800000004000000000000003C000101303030303030300000101353532323534313435380000000010000000000006546
2020-08-27 11:33:35.965 INFO SMPP_client_1: -> 000000198000000400000000000003C34374131454634310 0
2020-08-27 11:33:35.965 INFO SMPP_client_1: Message part sent using sequence number 60. It was accepted by the SMSC with Reference Id: '47A1EF41'.
2020-08-27 11:33:35.965 INFO SMPP_client_1: Message successfully sent. +0000000->+5522541458 'Test58'. Reference Id: '47A1EF41'.
2020-08-27 11:33:35.965 INFO SMPP_client_1: Received. 3c8ad20e-5fe2-4fcd-b0ab-d06baaa663d2 -> +5522541458 'Test59' Task ID: e181d997-6dcc-4883-a6b5-28
2020-08-27 11:33:35.965 INFO SMPP_client_1: Encoding information: Message length is 6. It is encoded as GSM 7-bit. It will be sent in 1 message(s).
2020-08-27 11:33:35.965 INFO SMPP_client_1: Sending message (1 parts). +0000000->+5522541458 'Test59'
2020-08-27 11:33:36.869 INFO SMPP_client_1: <- 0000003800000004000000000000003D000101303030303030300000101353532323534313435380000000010000000000006546
2020-08-27 11:33:36.869 INFO SMPP_client_1: -> 0000001A8000000400000000000003D32304235323830424600
2020-08-27 11:33:36.869 INFO SMPP_client_1: Message part sent using sequence number 61. It was accepted by the SMSC with Reference Id: '20B5280BF'.
2020-08-27 11:33:36.869 INFO SMPP_client_1: Message successfully sent. +0000000->+5522541458 'Test59'. Reference Id: '20B5280BF'.
2020-08-27 11:33:36.869 INFO SMPP_client_1: Received. 3c8ad20e-5fe2-4fcd-b0ab-d06baaa663d2 -> +5522541458 'Test60' Task ID: 7f9d57ef-ab41-49bc-8316-7c
2020-08-27 11:33:36.869 INFO SMPP_client_1: Encoding information: Message length is 6. It is encoded as GSM 7-bit. It will be sent in 1 message(s).
2020-08-27 11:33:36.869 INFO SMPP_client_1: Sending message (1 parts). +0000000->+5522541458 'Test60'
2020-08-27 11:33:37.717 INFO SMPP_client_1: <- 0000003800000004000000000000003E000101303030303030300000101353532323534313435380000000010000000000006546
2020-08-27 11:33:37.717 INFO SMPP_client_1: -> 0000001A8000000400000000000003E31323844374142373600
2020-08-27 11:33:37.717 INFO SMPP_client_1: Message part sent using sequence number 62. It was accepted by the SMSC with Reference Id: '128D7AB76'.
2020-08-27 11:33:37.717 INFO SMPP_client_1: Message successfully sent. +0000000->+5522541458 'Test60'. Reference Id: '128D7AB76'.
```

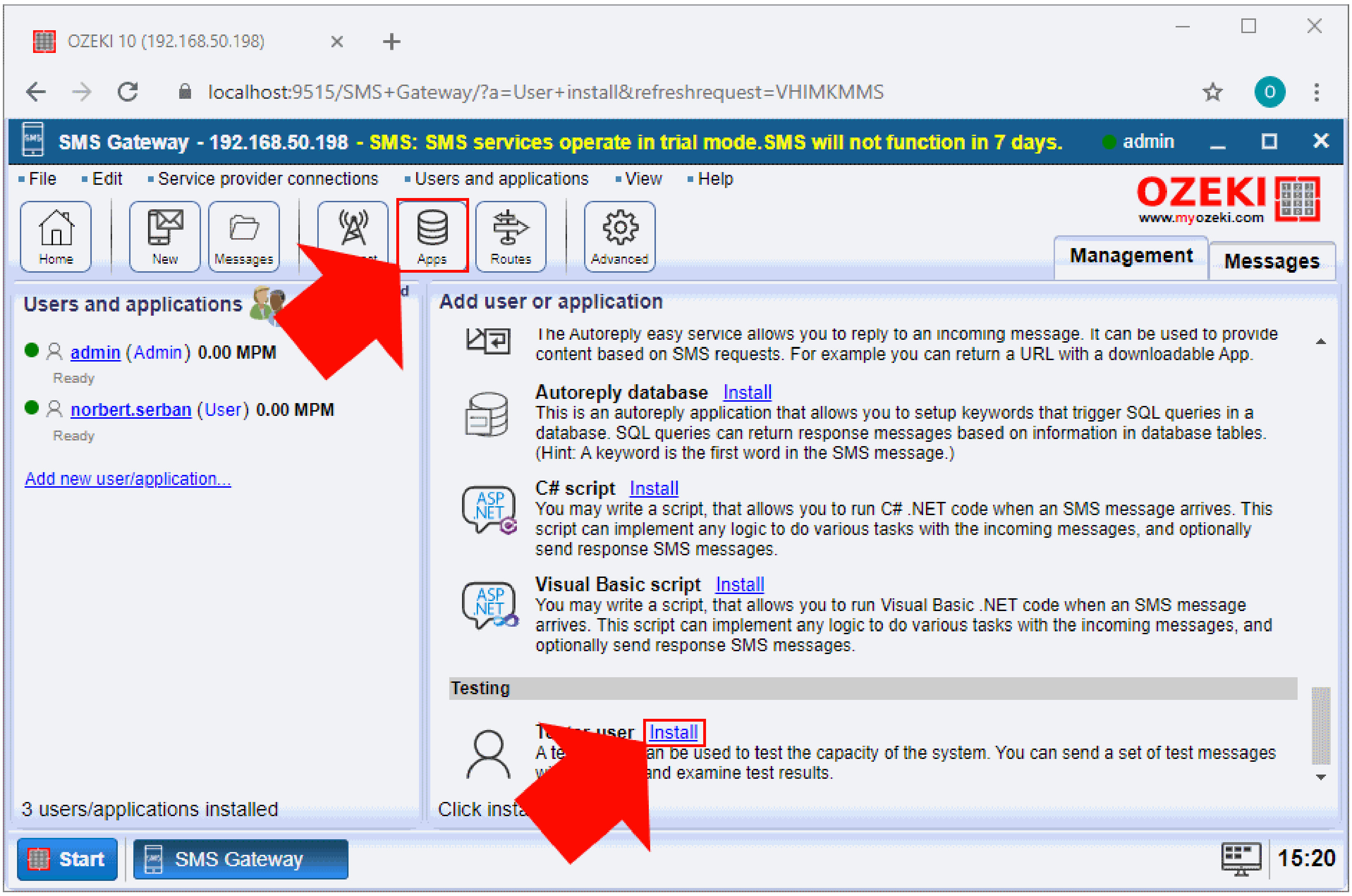
Figure 5 - Last message sent time stamp

# System performance testing

***This guide is about system performance testing. It gives you the steps to take to be able to determine the bandwidth of your connections. The guide consists of a few steps, in which you will install a tester user and a tester connection. It will show how you can send a lot of messages (tens of thousands) through the system and how you can see the test result.***

## Step 1 - Install Tester user connection

To test the performance of your system, the SMS Gateway provides you tester connections to be able to check the bandwidth of your connections. These tester connections can be installed the same as any other connection. First, you need to establish a tester user connection. For that, click on the Apps menu, and like in Figure 1, scroll down to the Testing section and click on the Install button of the Tester user connection.



OZEKI 10 (192.168.50.198)
localhost:9515/SMS+Gateway/?a=User+install&refreshrequest=VHIMKMMS
SMS Gateway - 192.168.50.198 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
OZEKI
www.myozeki.com
Home
New
Messages
Apps
Routes
Advanced
Management
Messages

Users and applications

admin (Admin) 0.00 MPM
Ready

norbert.serban (User) 0.00 MPM
Ready

Add new user/application...

3 users/applications installed

Add user or application

The Autoreply easy service allows you to reply to an incoming message. It can be used to provide content based on SMS requests. For example you can return a URL with a downloadable App.

**Autoreply database** Install
This is an autoreply application that allows you to setup keywords that trigger SQL queries in a database. SQL queries can return response messages based on information in database tables. (Hint: A keyword is the first word in the SMS message.)

**C# script** Install
You may write a script, that allows you to run C# .NET code when an SMS message arrives. This script can implement any logic to do various tasks with the incoming messages, and optionally send response SMS messages.

**Visual Basic script** Install
You may write a script, that allows you to run Visual Basic .NET code when an SMS message arrives. This script can implement any logic to do various tasks with the incoming messages, and optionally send response SMS messages.

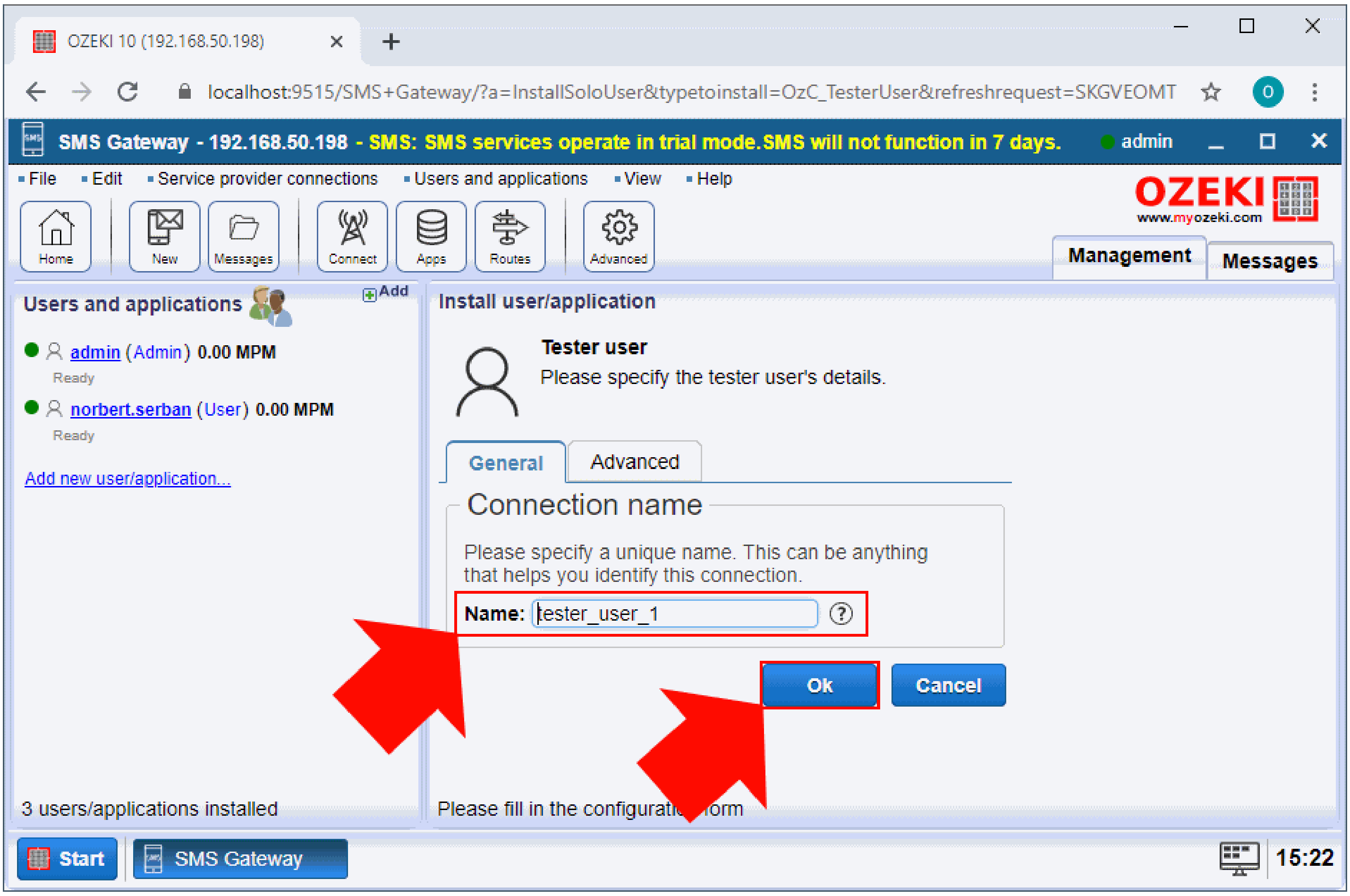
Testing

**Te... user** Install
A te... an be used to test the capacity of the system. You can send a set of test messages w... and examine test results.

Click insta...

Figure 1 - Select Tester user connection

Before you can create the Tester user connection, the configuration menu of the connection shows up for you. This connection does not need any further configuration, all you need to do is to specify a name for the connection as you can see it in Figure 2. If you have done that, just click on OK to create the tester connection.


OZEKI 10 (192.168.50.198)
localhost:9515/SMS+Gateway/?a=InstallSoloUser&typetoinstall=OzC_TesterUser&refreshrequest=SKGVEOMT
SMS Gateway - 192.168.50.198 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Users and applications
Add
admin (Admin) 0.00 MPM
Ready
norbert.serban (User) 0.00 MPM
Ready
Add new user/application...
3 users/applications installed
Install user/application
General
Advanced
Ok
Cancel
Please fill in the configuration form
Start
SMS Gateway
15:22

Tester user

Please specify the tester user's details.

Connection name

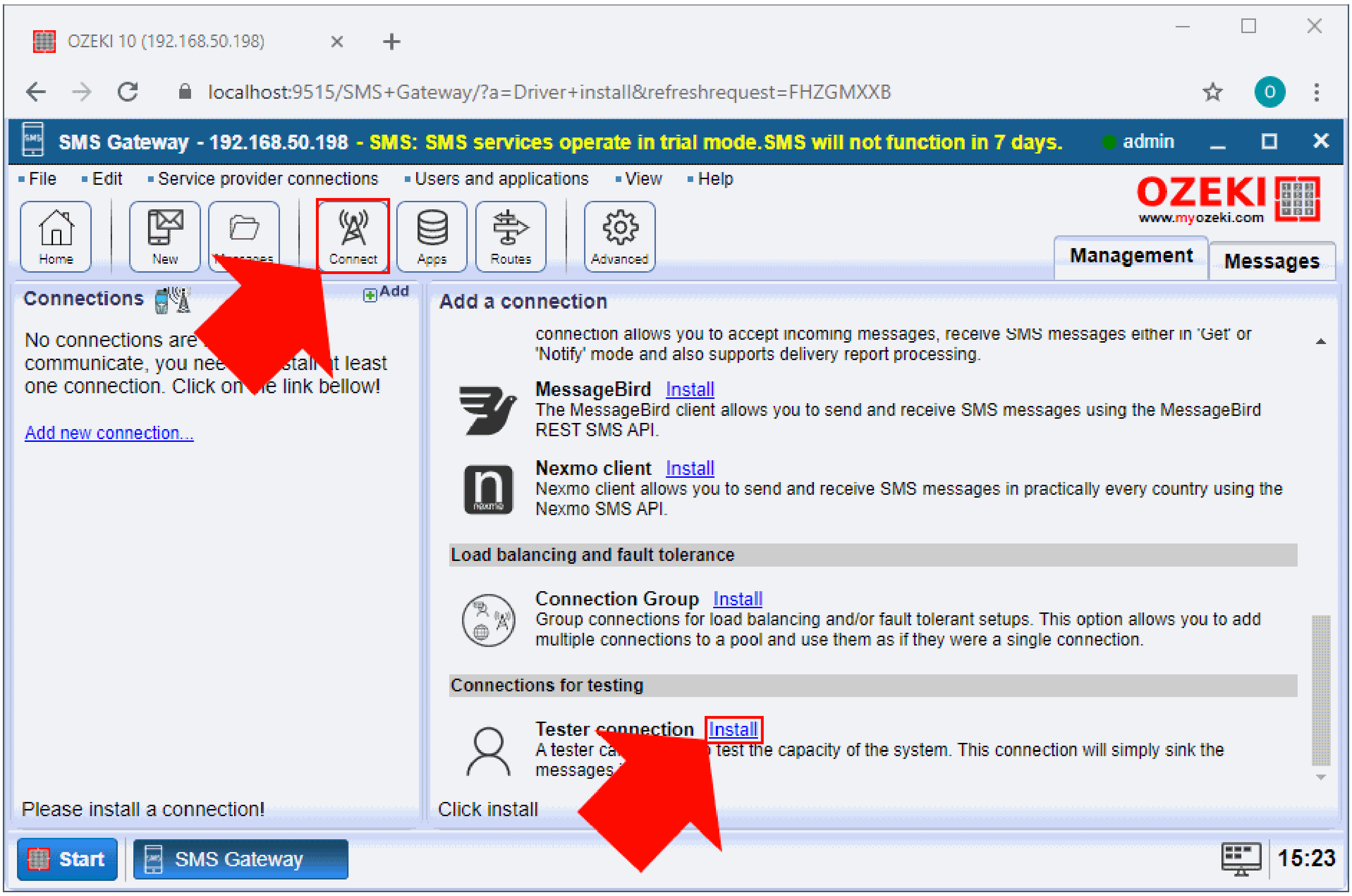
Please specify a unique name. This can be anything that helps you identify this connection.

Name: tester_user_1

Figure 2 - Configure Tester user connection

## Step 2 - Install Tester service provider connection

The next connection that you need to install for the performance test is the Tester service provider connection. This connection will be used as a test service provider, where the Tester user connection connects to. To create this connection, select the Connect menu from the toolbar, and from the Connections for testing section (Figure 3), click on the Install button of the Tester connection.


OZEKI 10 (192.168.50.198)
localhost:9515/SMS+Gateway/?a=Driver+install&refreshrequest=FHZGMXXB
SMS Gateway - 192.168.50.198 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Connections
Add
Add a connection
Start
SMS Gateway
15:23

No connections are ... communicate, you nee... stall ...t least one connection. Click on ...e link bellow!

Add new connection...

connection allows you to accept incoming messages, receive SMS messages either in 'Get' or 'Notify' mode and also supports delivery report processing.

MessageBird Install
The MessageBird client allows you to send and receive SMS messages using the MessageBird REST SMS API.

Nexmo client Install
Nexmo client allows you to send and receive SMS messages in practically every country using the Nexmo SMS API.

Load balancing and fault tolerance

Connection Group Install
Group connections for load balancing and/or fault tolerant setups. This option allows you to add multiple connections to a pool and use them as if they were a single connection.

Connections for testing

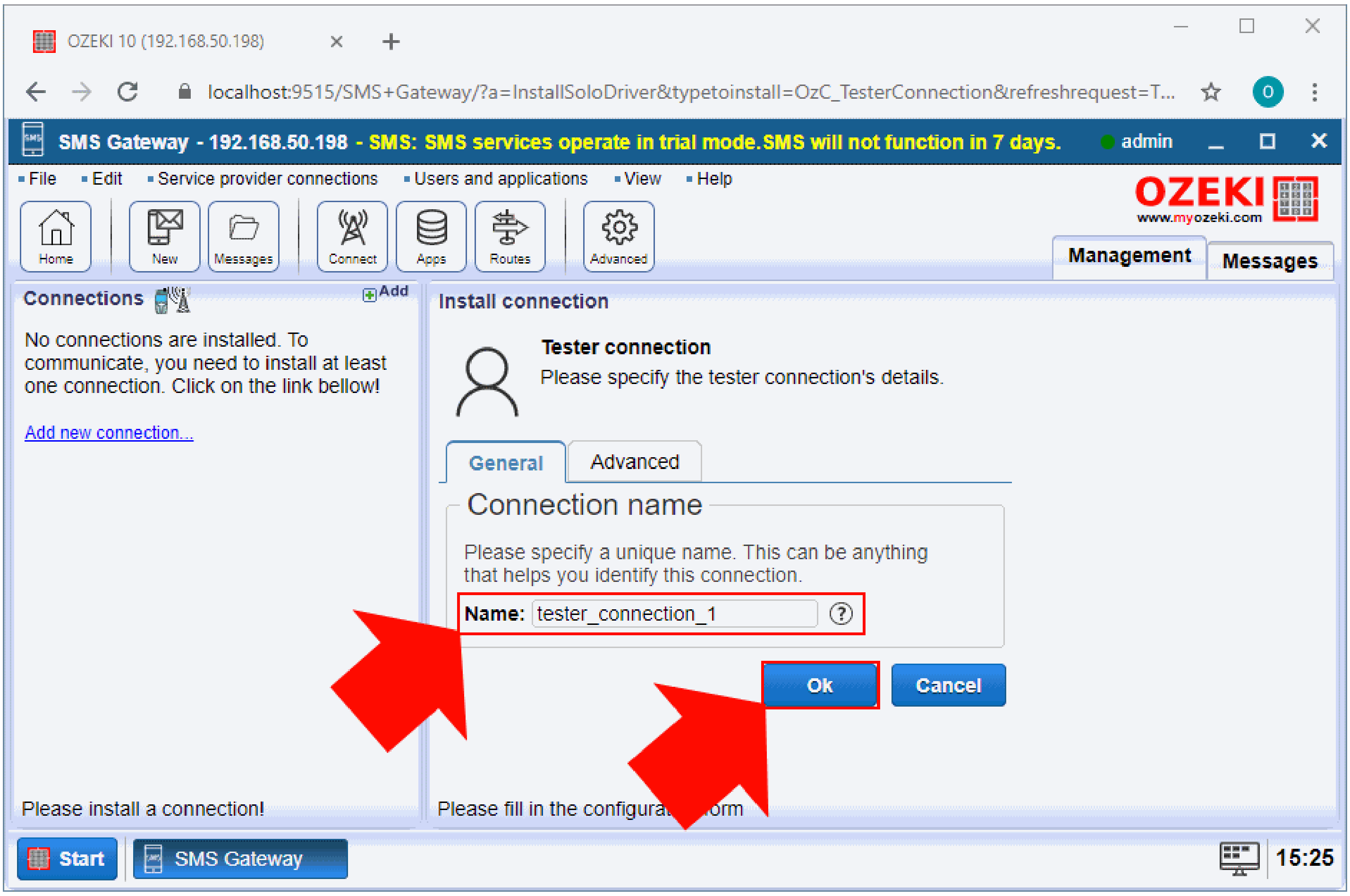
Tester connection Install
A tester ca... test the capacity of the system. This connection will simply sink the messages...

Please install a connection!

Click install

Figure 3 - Select Tester service provider connection

The next window that shows up in the SMS Gateway is the configuration menu of the Tester service provider connection. Here, you need to enter a name for that tester connection as Figure 4 demonstrates it. Then, you just need to click on OK to create the Tester service provider connection.


OZEKI 10 (192.168.50.198)
localhost:9515/SMS+Gateway/?a=InstallSoloDriver&typetoinstall=OzC_TesterConnection&refreshrequest=T...
SMS Gateway - 192.168.50.198 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Connections
Add
Install connection
General
Advanced
Ok
Cancel
Start
SMS Gateway
15:25

No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

Tester connection
Please specify the tester connection's details.

Connection name

Please specify a unique name. This can be anything that helps you identify this connection.

Name: tester_connection_1

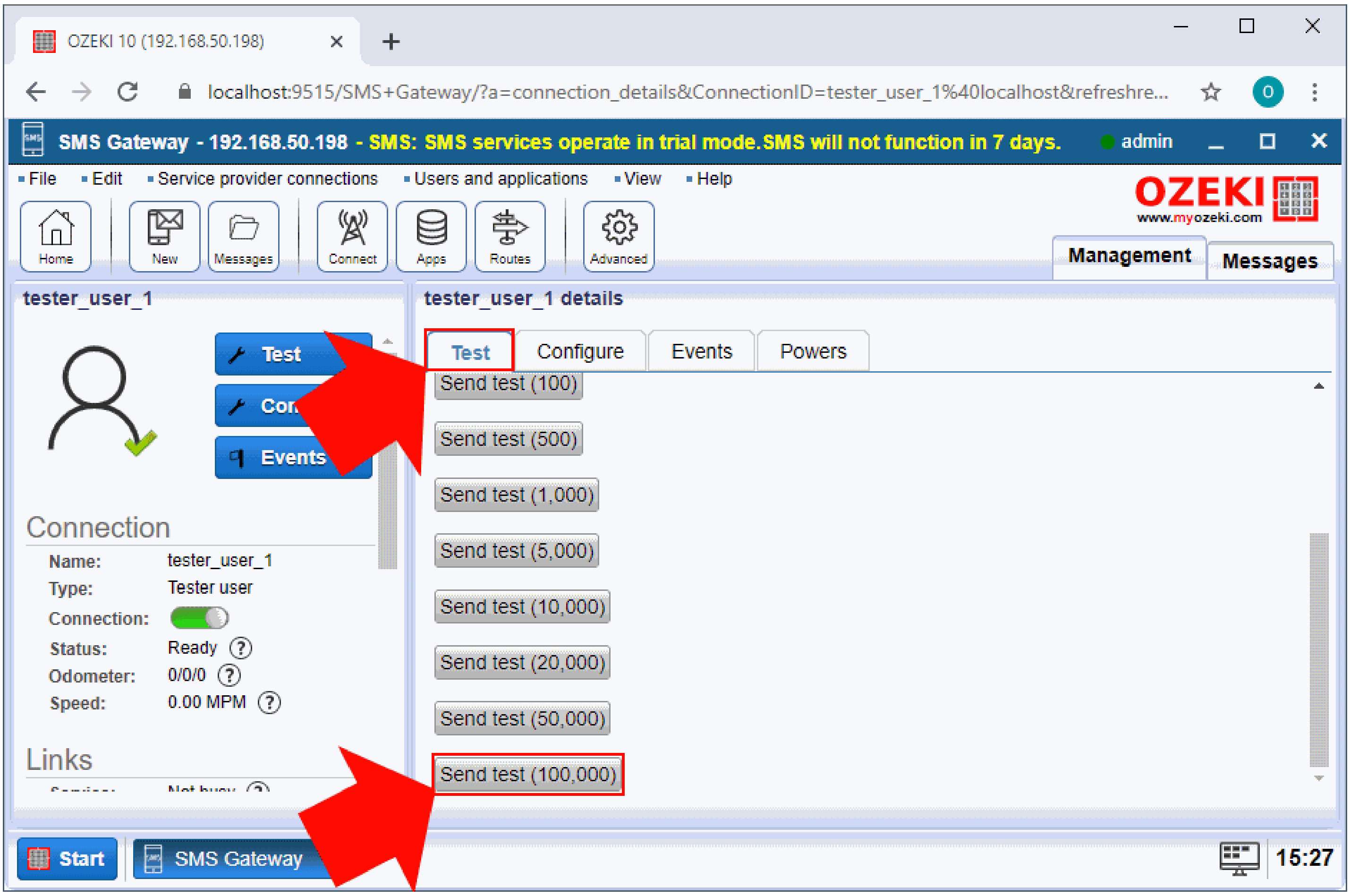
Please install a connection!

Please fill in the configura... orm

Figure 4 - Configure Tester service provider connection

## Step 3 - Start system performance test

At this point, you installed both of the required tester connections and your system is ready to be tested. For that, first, open the Tester user connection in the main menu of the SMS Gateway. Here, in the Test tab of the connection, you can select the amount messages the test is going to send. This allows you to test your system only with one single message or initiate 100 000 messages at once. So, as you can see it in Figure 5, select the test with 100 000 messages by clicking on the 'Send test (100,000)' button.

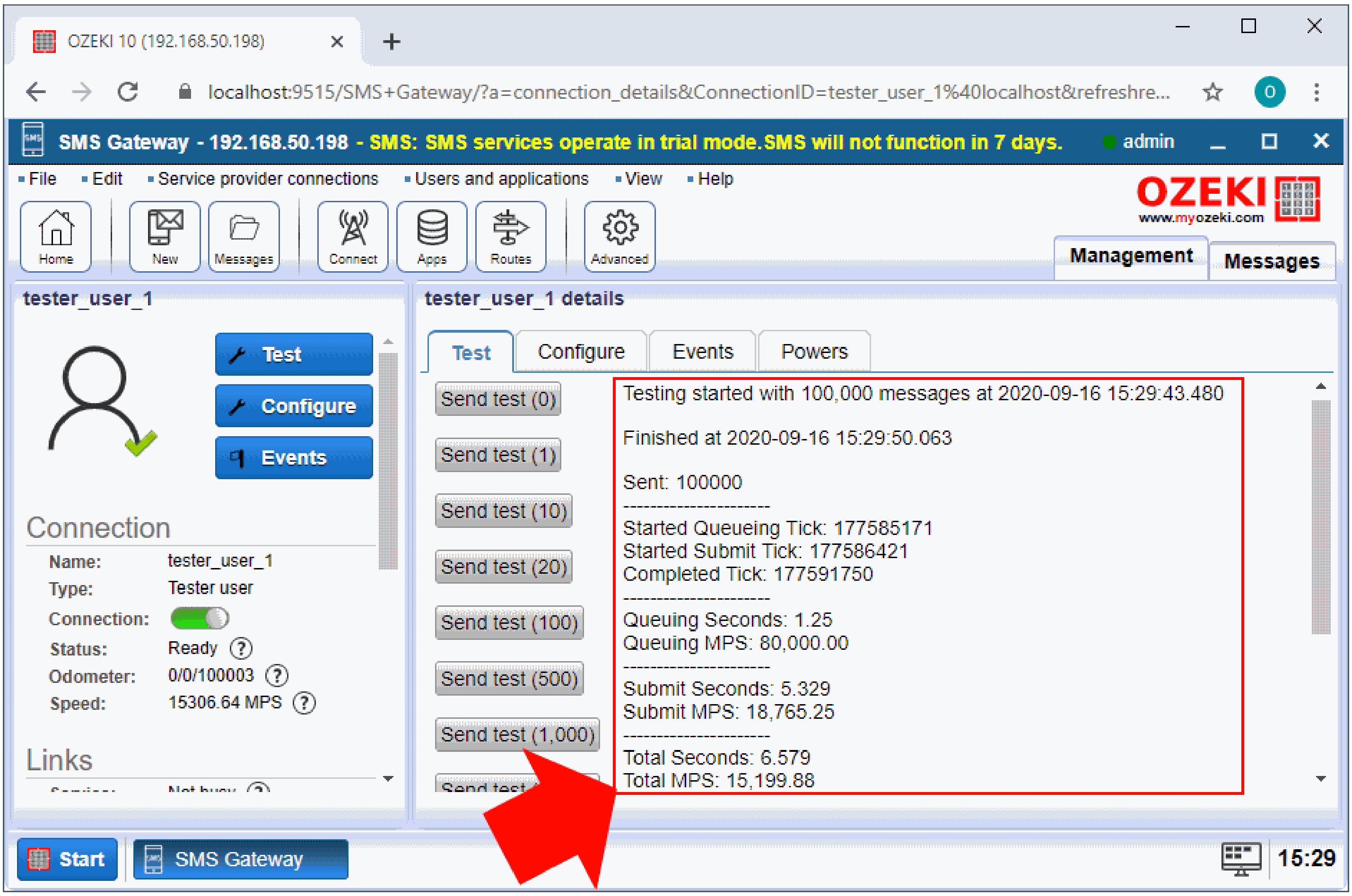


Name: tester_user_1
Type: Tester user
Connection:
Status: Ready
Odometer: 0/0/0
Speed: 0.00 MPM

Figure 5 - Select performance test

## Step 4 - Check test results

After you clicked on the test button, the connection starts the test of your system. As soon as the test finishes, you will be able to see the results in the Test menu. As Figure 6 demonstrates that, the main information about the test appears in the Test tab. Here, the most important detail is the Total MPS in the last row. This shows the number of messages that the connection is able to send in one second.



Connection

Name: tester_user_1
Type: Tester user
Connection:
Status: Ready
Odometer: 0/0/100003
Speed: 15306.64 MPS

Links

```
Testing started with 100,000 messages at 2020-09-16 15:29:43.480

Finished at 2020-09-16 15:29:50.063

Sent: 100000
----------------------
Started Queueing Tick: 177585171
Started Submit Tick: 177586421
Completed Tick: 177591750
----------------------
Queuing Seconds: 1.25
Queuing MPS: 80,000.00
----------------------
Submit Seconds: 5.329
Submit MPS: 18,765.25
----------------------
Total Seconds: 6.579
Total MPS: 15,199.88
```

Figure 6 - See test results in Events menu

# Further information

You might notice that for a test of 100 000, the number of messages sent were actually 100 003. This is caused by the fact, that the system sends 3 more messages to initiate the test system: the first message tells the system how many test messages to generate. The second starts the speed measurement timer, the third stops the speed measurement timer.

# Connecting to multiple SMS networks

***In this guide you will learn about connecting your SMS system to multiple mobile networks for sending SMS messages. This is useful to achive redundancy and to optimize for costs. You will learn, how to manage SMS routing to select the mobile network you wish to route your message to. The guide does not detail the setup of each individual connection, as there are separate guides for that. The focus is on working with multiple connections, and the ability to control what happens to messages going through your system. You will learn how to select a destination for your messages and how to modify the message content or the message sender and recipient addresses on the fly.***

## Terms and definitions

Let's get started by defining the most important definitions needed to understand multiple mobile network connections and SMS routing.

**What is an SMS message:** An SMS message is a sort text or binary message that goes through your system. It carries 140 bytes of useful data. It can carry 160 gsm (latin) characters, which are 7 bit long, 70 unicode characters which are 16 bit long, or 140 bytes of binary data. Long text messages (longer then 160 characters) are split into multiple SMS in order to carry the long text over the network. This technology is called segmentation and reassembly (SAR). The sender mobile phone segments the long text and sends it over the network using multiple SMS messages, and the recipient waits for all message segments to arrive and assembles them. When you are working with SMS routing you route standalone short messages (or message segments of multipart SM). When a multipart message goes through the system, each part is routed individualy.

**What is an SMS connection:** A connection is an interface for an SMS message to either come into the system or to go out from the system. For example you can setup an SMPP client "connection" to connect to a mobile network operator. In this case you can send messages and receive messages through this connection to and from the mobile network. Another example would be a database. You can setup a database with two database tables to send and receive SMS messages. In this case you would add a database "connection" to your system.

**What is an SMS route:** A route takes a message from one connection and passes it to another connection. For example you can setup a route to take a message from the database connection and to pass it to an SMPP client connection. Routes can also be used to modify a message on the fly (e.g. replace the message text to something else or to change the sender or recipient phone numbers.

**What is an SMS routing table:** A routing table contains a list of routes. When an SMS message comes in from a connection, the routing table is used. The system checks the first route and if it matches the sms message, it will use it to select the destination connection. If the incoming sms does not match the first route, it tries to use the second route, then the third and so on. Routing stops when a route matches the incoming sms.

## Overview

Since the Ozeki SMS software offers many different connections it can be used as an SMS router. It will work at high performance and it can be used to control your messages. In order to setup Ozeki as an SMS router, you need to learn how to setup connection and how to setup SMS routing. Figure 1 gives you an idea on what kind of SMS connections you may setup.

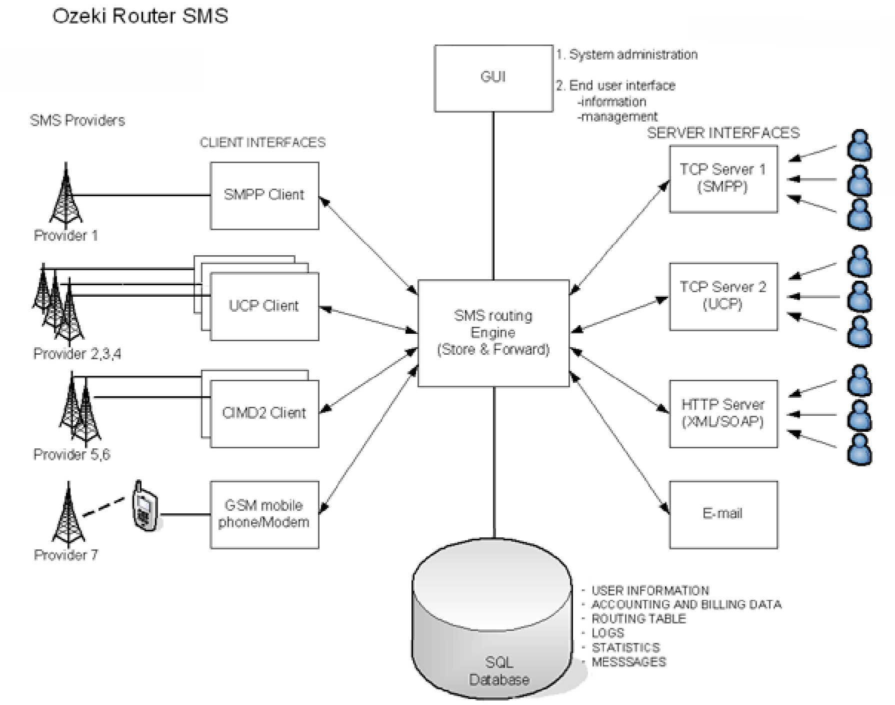
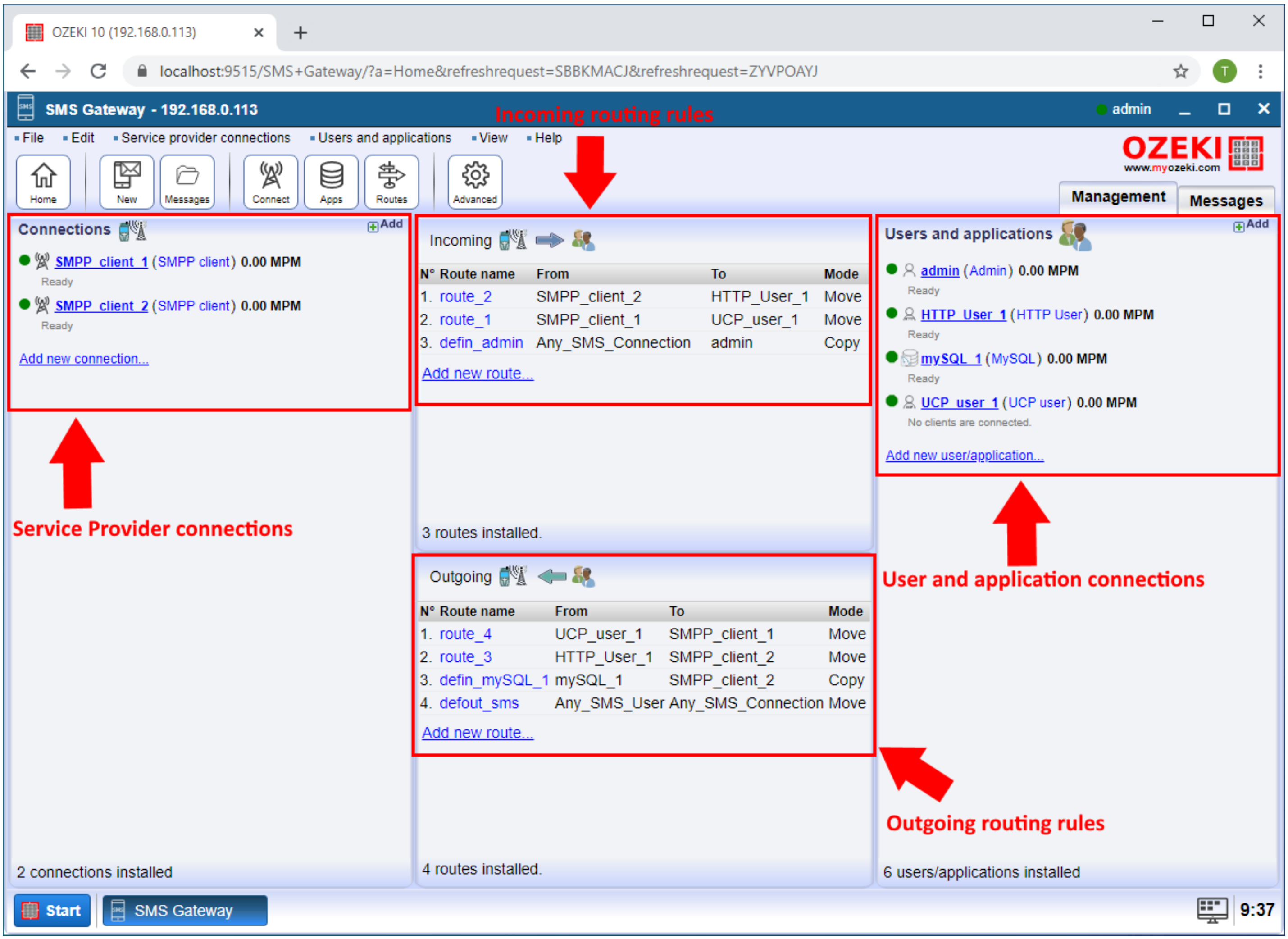


Figure 1 - SMS connections

# Step 1 - Connection and routes in the SMS Gateway

On the main page of the Ozeki SMS gateway, you can see the Service provider connections, the User/Application connections, outbound and inbound routing rules.



Incoming routing rules

**Connections** Add

- ● SMPP_client_1 (SMPP client) 0.00 MPM
  Ready
- ● SMPP_client_2 (SMPP client) 0.00 MPM
  Ready

Add new connection...

2 connections installed

Service Provider connections

Incoming

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | route_2 | SMPP_client_2 | HTTP_User_1 | Move |
| 2. | route_1 | SMPP_client_1 | UCP_user_1 | Move |
| 3. | defin_admin | Any_SMS_Connection | admin | Copy |

Add new route...

3 routes installed.

Outgoing

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | route_4 | UCP_user_1 | SMPP_client_1 | Move |
| 2. | route_3 | HTTP_User_1 | SMPP_client_2 | Move |
| 3. | defin_mySQL_1 | mySQL_1 | SMPP_client_2 | Copy |
| 4. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move |

Add new route...

4 routes installed.

**Users and applications** Add

- ● admin (Admin) 0.00 MPM
  Ready
- ● HTTP_User_1 (HTTP User) 0.00 MPM
  Ready
- ● mySQL_1 (MySQL) 0.00 MPM
  Ready
- ● UCP_user_1 (UCP user) 0.00 MPM
  No clients are connected.

Add new user/application...

6 users/applications installed

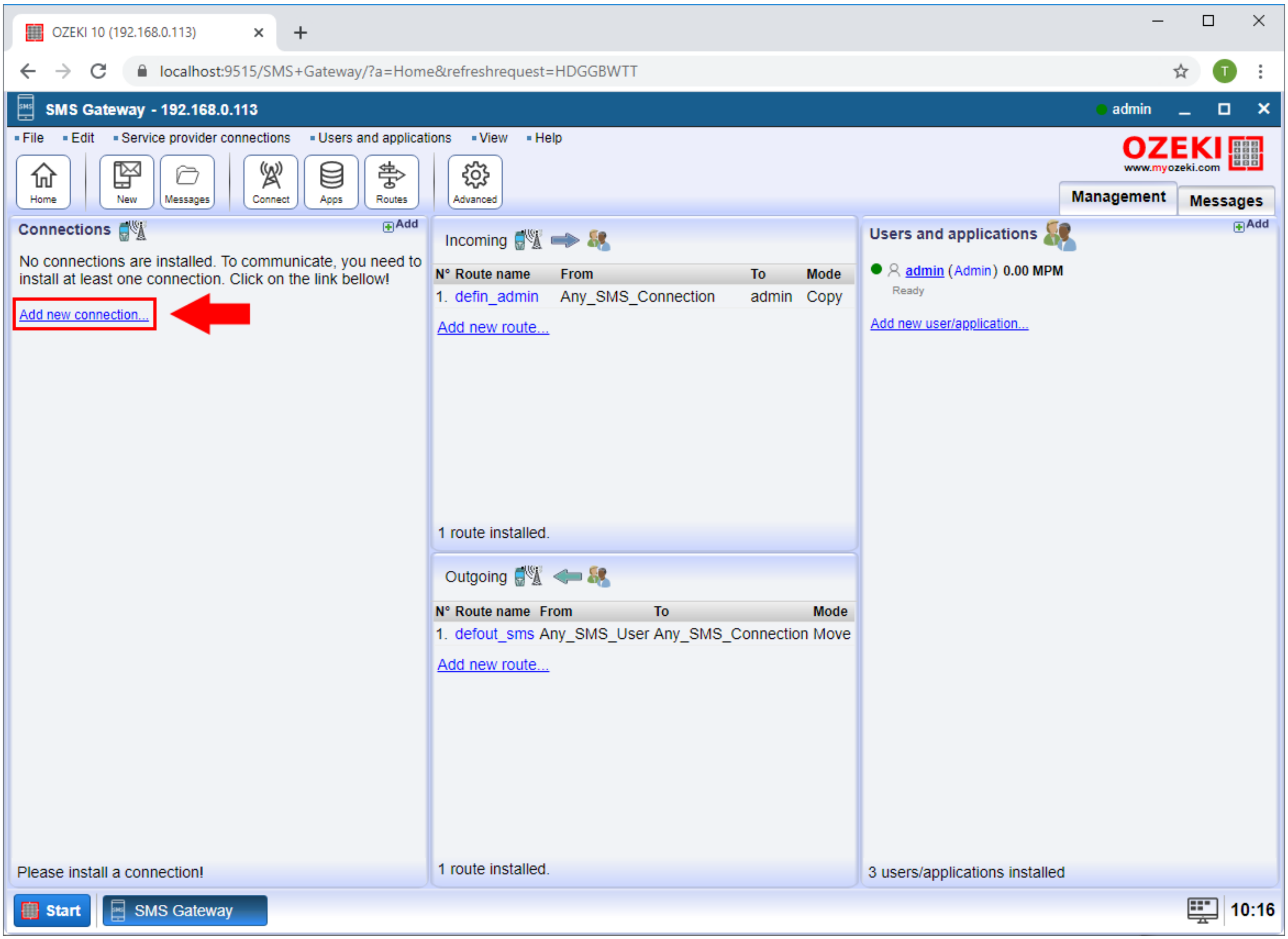
User and application connections

Outgoing routing rules

Figure 2 - SMS connections and routes on the Ozeki GUI

# Step 2 - Add a new SMS connection to the mobile metwork

You can add a service provier connection by clicking Add new connection in the Connections section. Then select the type of connection that suits for you from the list and click Install to add the connection.



Connections

No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

Please install a connection!

Incoming

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defin_admin | Any_SMS_Connection | admin | Copy |

Add new route...

1 route installed.

Outgoing

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move |

Add new route...

1 route installed.

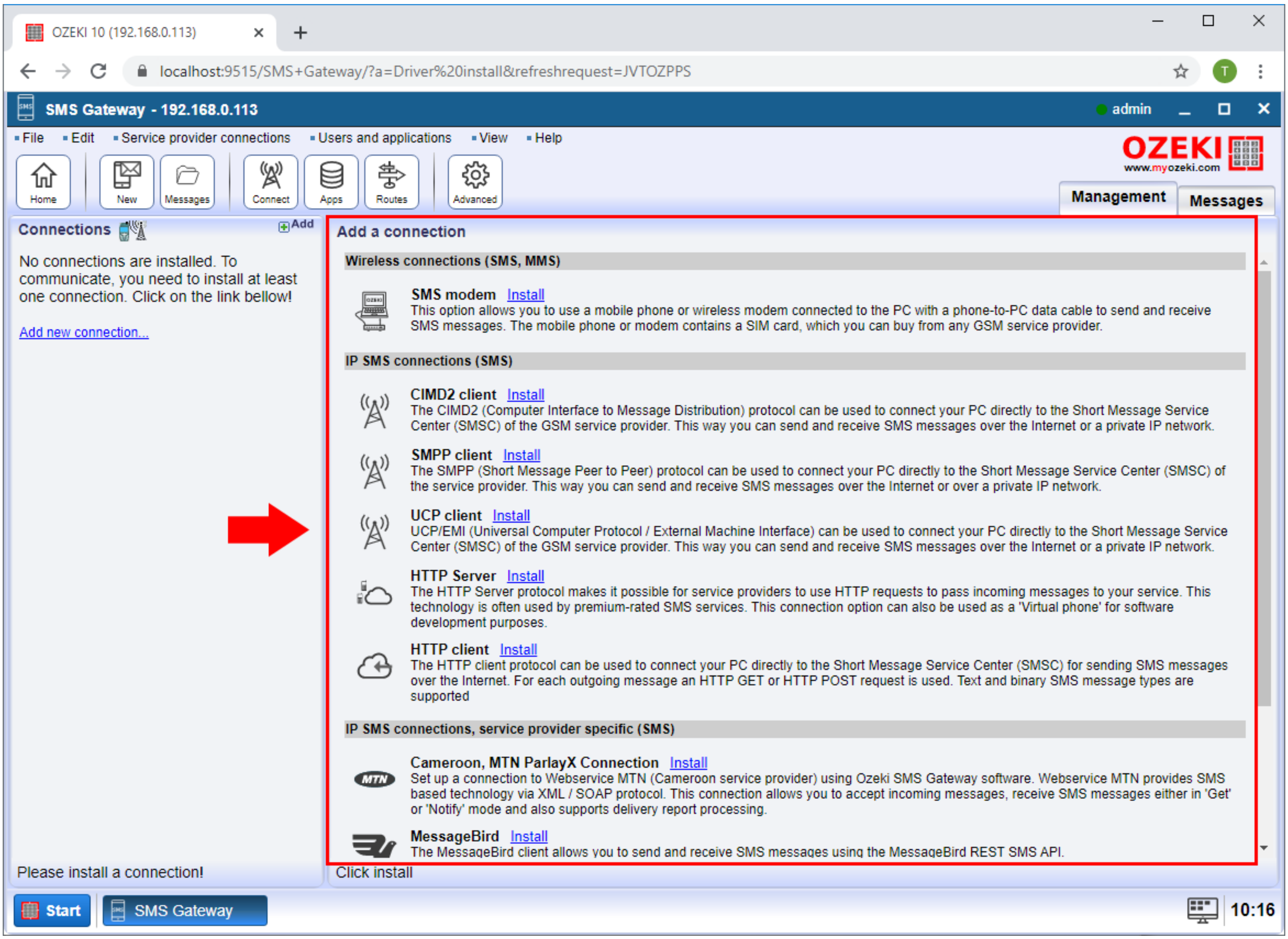
Users and applications

● admin (Admin) 0.00 MPM
Ready

Add new user/application...

3 users/applications installed

Figure 3 - Add a new SMS connection to the mobile network



No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

Please install a connection!

Add a connection

Wireless connections (SMS, MMS)

**SMS modem** Install
This option allows you to use a mobile phone or wireless modem connected to the PC with a phone-to-PC data cable to send and receive SMS messages. The mobile phone or modem contains a SIM card, which you can buy from any GSM service provider.

IP SMS connections (SMS)

**CIMD2 client** Install
The CIMD2 (Computer Interface to Message Distribution) protocol can be used to connect your PC directly to the Short Message Service Center (SMSC) of the GSM service provider. This way you can send and receive SMS messages over the Internet or a private IP network.

**SMPP client** Install
The SMPP (Short Message Peer to Peer) protocol can be used to connect your PC directly to the Short Message Service Center (SMSC) of the service provider. This way you can send and receive SMS messages over the Internet or over a private IP network.

**UCP client** Install
UCP/EMI (Universal Computer Protocol / External Machine Interface) can be used to connect your PC directly to the Short Message Service Center (SMSC) of the GSM service provider. This way you can send and receive SMS messages over the Internet or a private IP network.

**HTTP Server** Install
The HTTP Server protocol makes it possible for service providers to use HTTP requests to pass incoming messages to your service. This technology is often used by premium-rated SMS services. This connection option can also be used as a 'Virtual phone' for software development purposes.

**HTTP client** Install
The HTTP client protocol can be used to connect your PC directly to the Short Message Service Center (SMSC) for sending SMS messages over the Internet. For each outgoing message an HTTP GET or HTTP POST request is used. Text and binary SMS message types are supported

IP SMS connections, service provider specific (SMS)

**Cameroon, MTN ParlayX Connection** Install
Set up a connection to Webservice MTN (Cameroon service provider) using Ozeki SMS Gateway software. Webservice MTN provides SMS based technology via XML / SOAP protocol. This connection allows you to accept incoming messages, receive SMS messages either in 'Get' or 'Notify' mode and also supports delivery report processing.

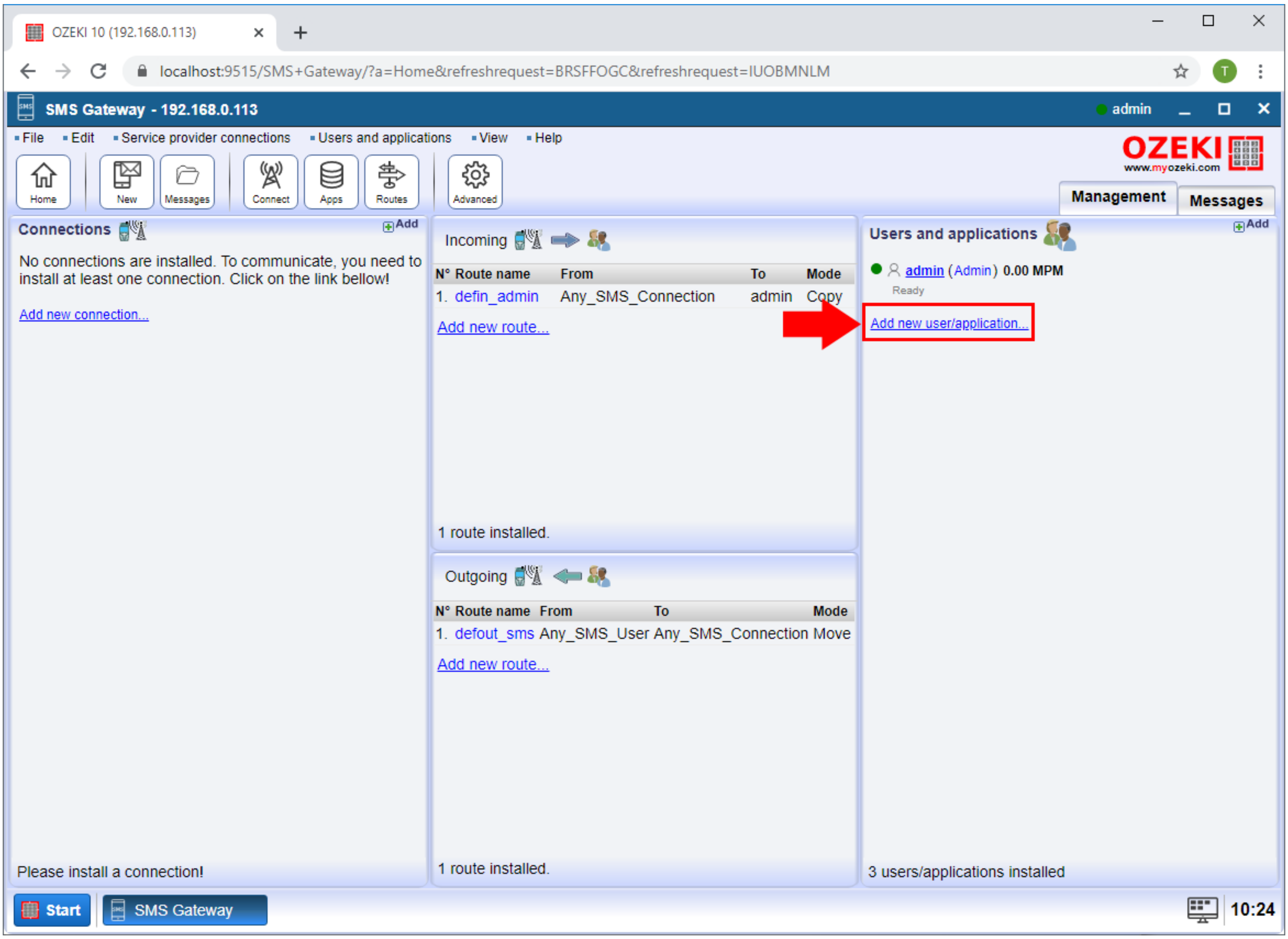
**MessageBird** Install
The MessageBird client allows you to send and receive SMS messages using the MessageBird REST SMS API.

Click install

Figure 4 - Add a new SMS connection to the mobile network

# Step 3 - Add a new SMS user or an SMS application

You can add a User or Application connection by clicking Add new connection in the Users and applications section. Then select the type of connection that suits for you from the list and click Install to add the connection.



No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

Please install a connection!

Incoming

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defin_admin | Any_SMS_Connection | admin | Copy |

Add new route...

1 route installed.

Outgoing

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move |

Add new route...

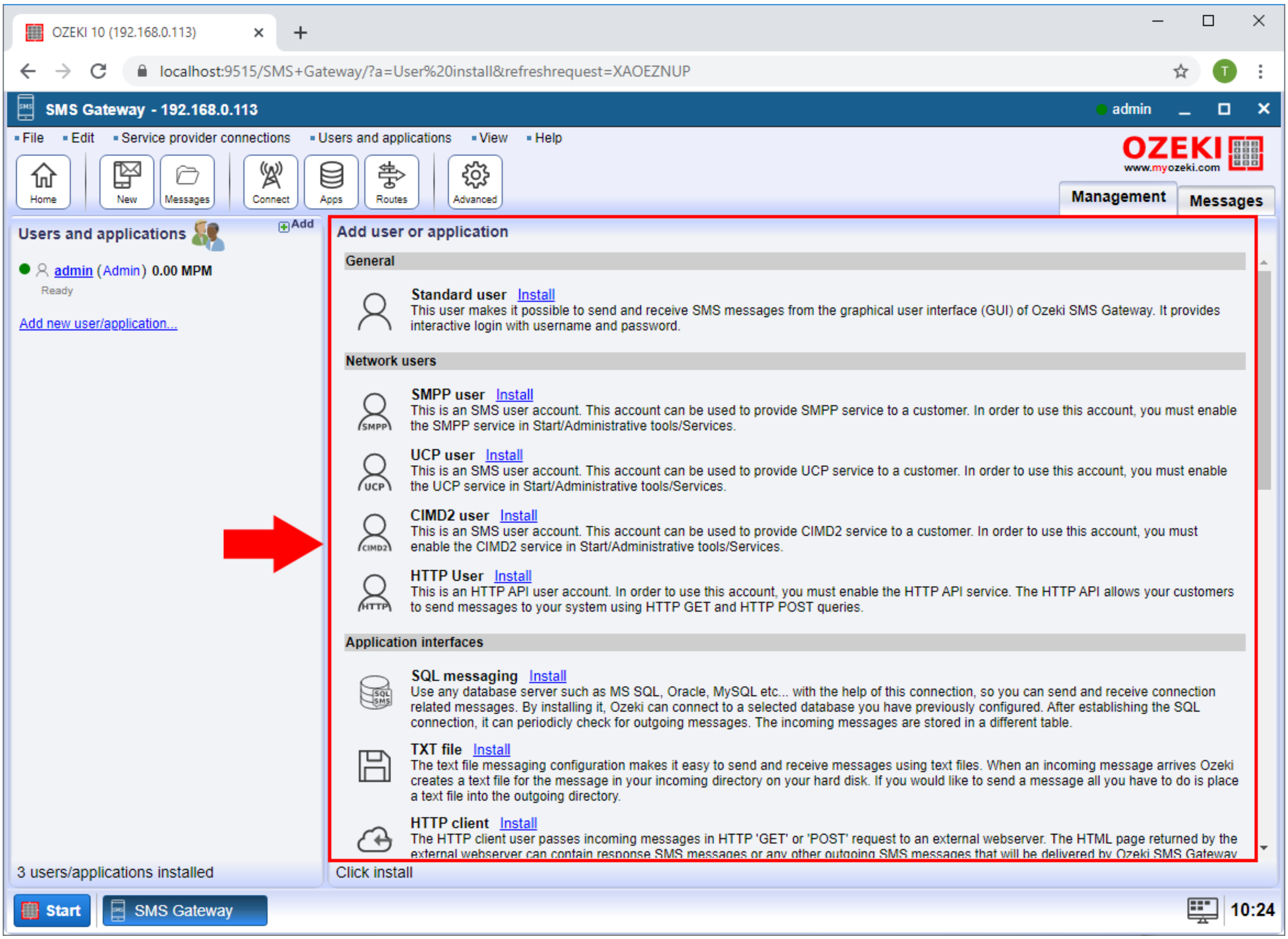
1 route installed.

● admin (Admin) 0.00 MPM
Ready

Add new user/application...

3 users/applications installed

Figure 5 - Add a new SMS application



Add user or application

General

**Standard user** Install
This user makes it possible to send and receive SMS messages from the graphical user interface (GUI) of Ozeki SMS Gateway. It provides interactive login with username and password.

Network users

**SMPP user** Install
This is an SMS user account. This account can be used to provide SMPP service to a customer. In order to use this account, you must enable the SMPP service in Start/Administrative tools/Services.

**UCP user** Install
This is an SMS user account. This account can be used to provide UCP service to a customer. In order to use this account, you must enable the UCP service in Start/Administrative tools/Services.

**CIMD2 user** Install
This is an SMS user account. This account can be used to provide CIMD2 service to a customer. In order to use this account, you must enable the CIMD2 service in Start/Administrative tools/Services.

**HTTP User** Install
This is an HTTP API user account. In order to use this account, you must enable the HTTP API service. The HTTP API allows your customers to send messages to your system using HTTP GET and HTTP POST queries.

Application interfaces

**SQL messaging** Install
Use any database server such as MS SQL, Oracle, MySQL etc... with the help of this connection, so you can send and receive connection related messages. By installing it, Ozeki can connect to a selected database you have previously configured. After establishing the SQL connection, it can periodicly check for outgoing messages. The incoming messages are stored in a different table.

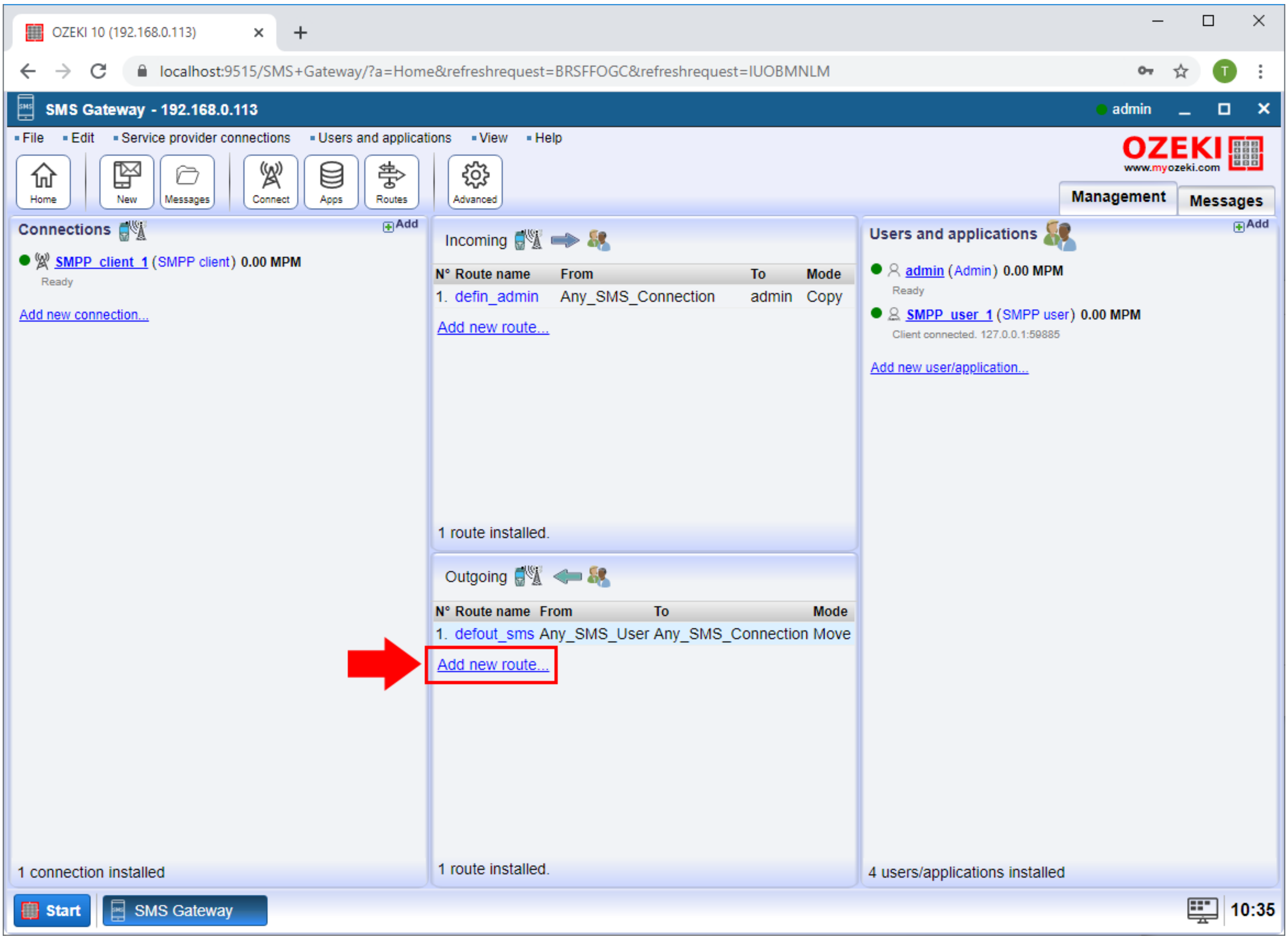
**TXT file** Install
The text file messaging configuration makes it easy to send and receive messages using text files. When an incoming message arrives Ozeki creates a text file for the message in your incoming directory on your hard disk. If you would like to send a message all you have to do is place a text file into the outgoing directory.

**HTTP client** Install
The HTTP client user passes incoming messages in HTTP 'GET' or 'POST' request to an external webserver. The HTML page returned by the external webserver can contain response SMS messages or any other outgoing SMS messages that will be delivered by Ozeki SMS Gateway

Figure 6 - Add a new SMS application

# Step 4 - Create an outbound SMS route

You can add an outbound routing rule by clicking add new route in the Outbound section. Then in the Settings, for the From connection, select the User connection from which you are sending the SMS, and for the To connection, select the Service provider connection through which you want to send the SMS to the service provider.



Connections

● SMPP_client_1 (SMPP client) 0.00 MPM
Ready

Add new connection...

1 connection installed

Incoming

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defin_admin | Any_SMS_Connection | admin | Copy |

Add new route...

1 route installed.

Outgoing

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move |

Add new route...

1 route installed.

Users and applications

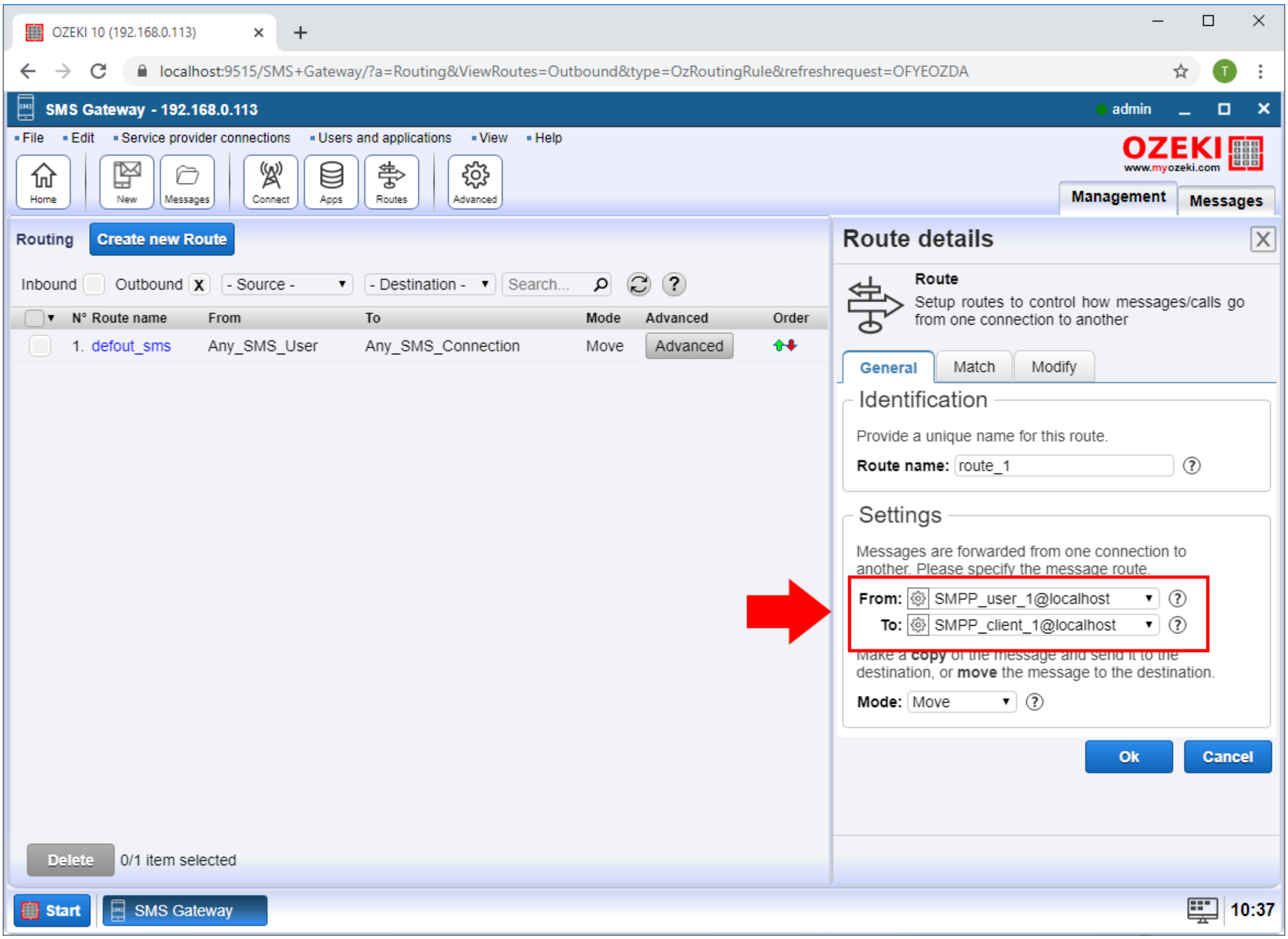
● admin (Admin) 0.00 MPM
Ready

● SMPP_user_1 (SMPP user) 0.00 MPM
Client connected. 127.0.0.1:59885

Add new user/application...

4 users/applications installed

Figure 7 - Create an outbound SMS route



localhost:9515/SMS+Gateway/?a=Routing&ViewRoutes=Outbound&type=OzRoutingRule&refreshrequest=OFYEOZDA

| N° | Route name | From | To | Mode | Advanced | Order |
|---|---|---|---|---|---|---|
| 1. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move | Advanced | |

0/1 item selected

**Route details**

**Route**
Setup routes to control how messages/calls go from one connection to another

**Identification**

Provide a unique name for this route.

**Route name:** route_1

**Settings**

Messages are forwarded from one connection to another. Please specify the message route.

**From:** SMPP_user_1@localhost

**To:** SMPP_client_1@localhost

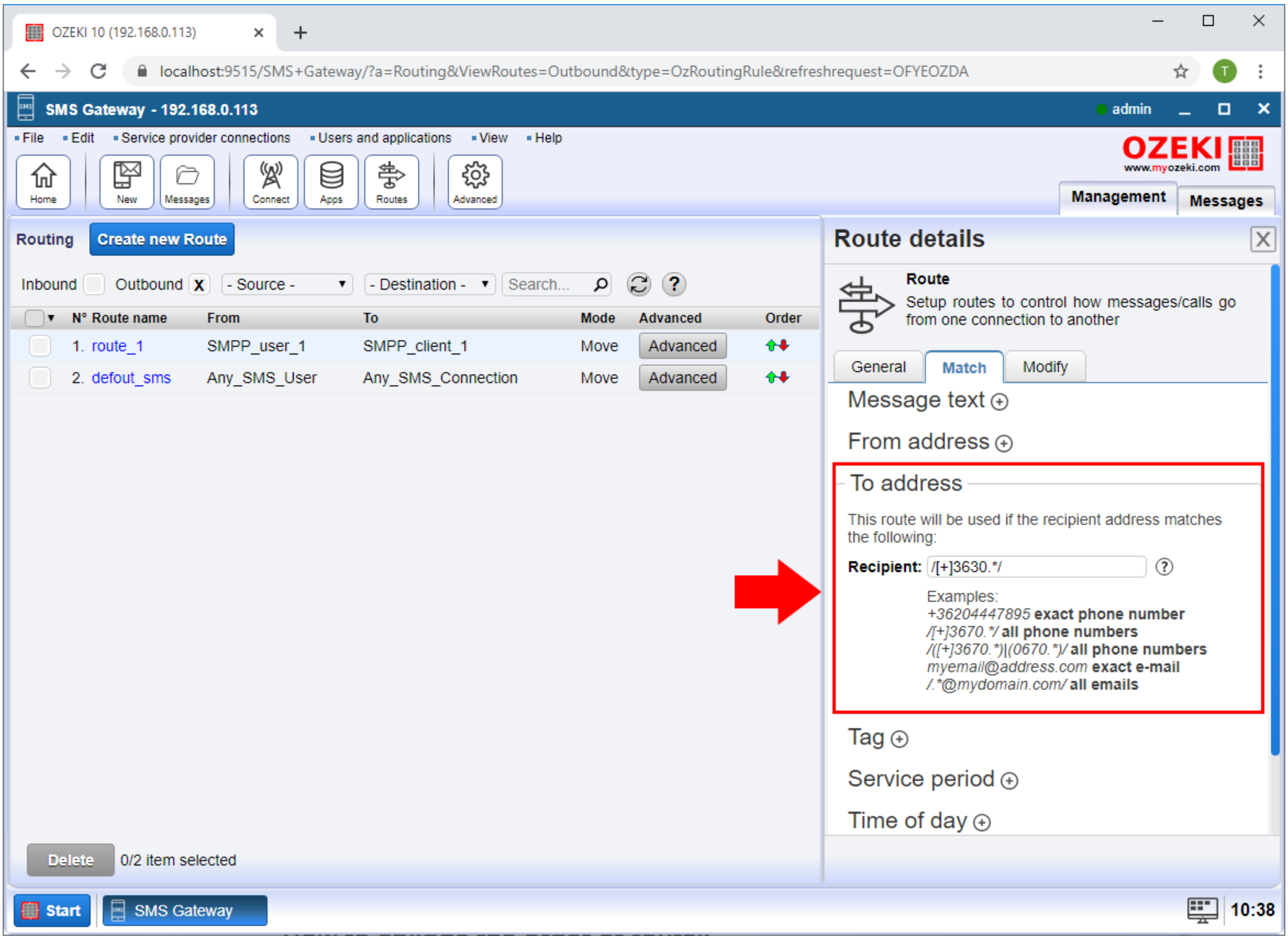
Make a **copy** of the message and send it to the destination, or **move** the message to the destination.

**Mode:** Move

Figure 8 - Create an outbound SMS route

## Step 5 - How to use phone number prefix for routing

In the SMS Gateway, during routing, you have the option of selecting which service provider connection to send the message to based on the prefix of the recipient's phone number. On the Match tab, in the To address field, you can enter the phone number prefix you want to use in the given routing rule.



| N° | Route name | From | To | Mode | Advanced | Order |
|---|---|---|---|---|---|---|
| 1. | route_1 | SMPP_user_1 | SMPP_client_1 | Move | Advanced | |
| 2. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move | Advanced | |

0/2 item selected

**Route details**

**Route**
Setup routes to control how messages/calls go from one connection to another

To address

This route will be used if the recipient address matches the following:

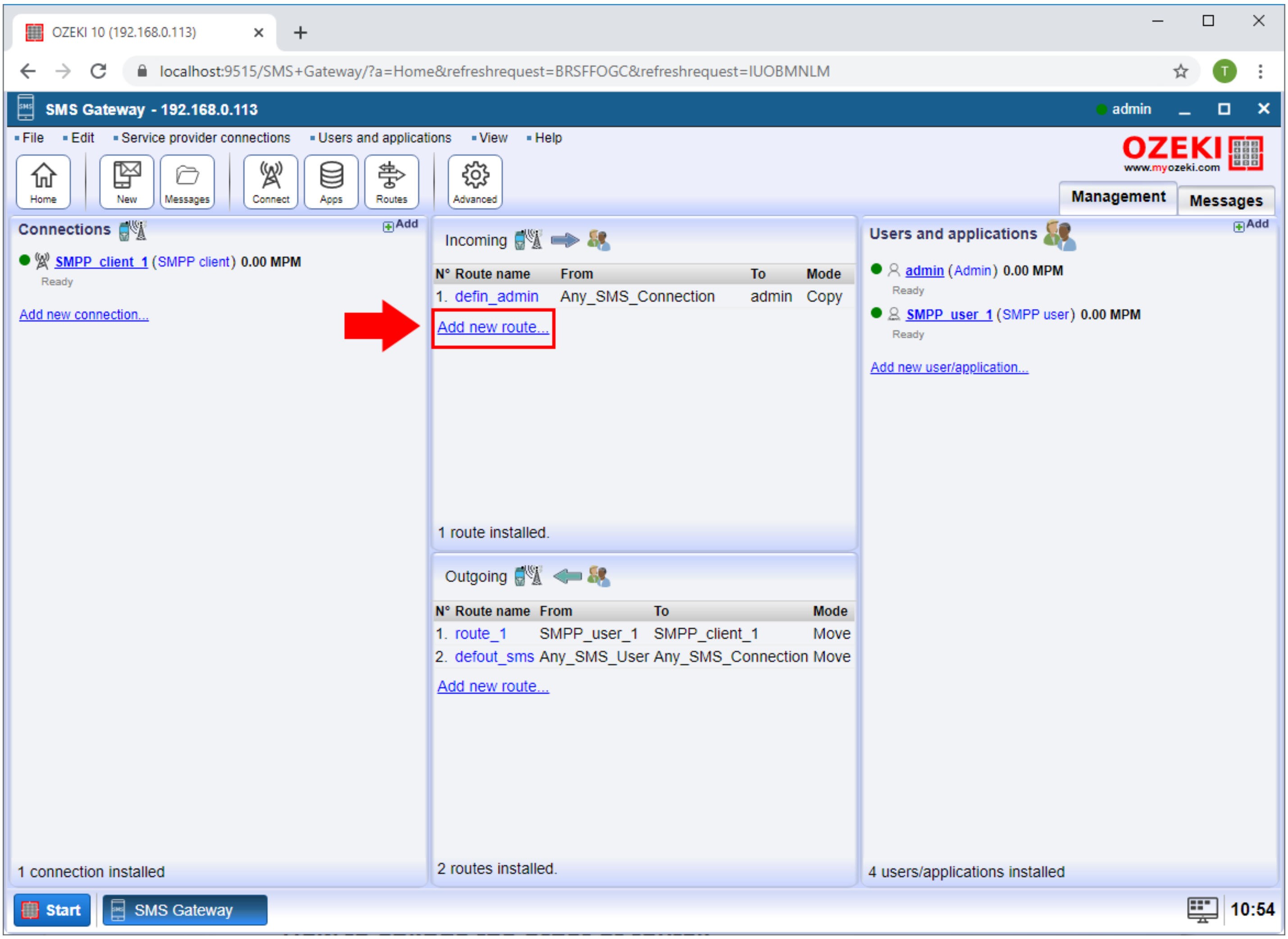
**Recipient:** /[+]3630.*/

Examples:
*+36204447895* **exact phone number**
*/[+]3670.*/* **all phone numbers**
*/([+]3670.*)|(0670.*)/* **all phone numbers**
*myemail@address.com* **exact e-mail**
*/.*@mydomain.com/* **all emails**

Figure 9 - Route based on phone number prefix

## Step 6 - Learn how to create an inbound SMS route

You can add an inbound routing rule by clicking add new route in the Inbound section. Then in the Settings, for the From connection, select the Service provider connection from teh sms is arrived, and for the To connection, select the User connection where you want to forward the SMS.



Connections

Add

● SMPP_client_1 (SMPP client) 0.00 MPM
Ready

Add new connection...

1 connection installed

Incoming

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defin_admin | Any_SMS_Connection | admin | Copy |

Add new route...

1 route installed.

Outgoing

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | route_1 | SMPP_user_1 | SMPP_client_1 | Move |
| 2. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move |

Add new route...

2 routes installed.

Users and applications

Add

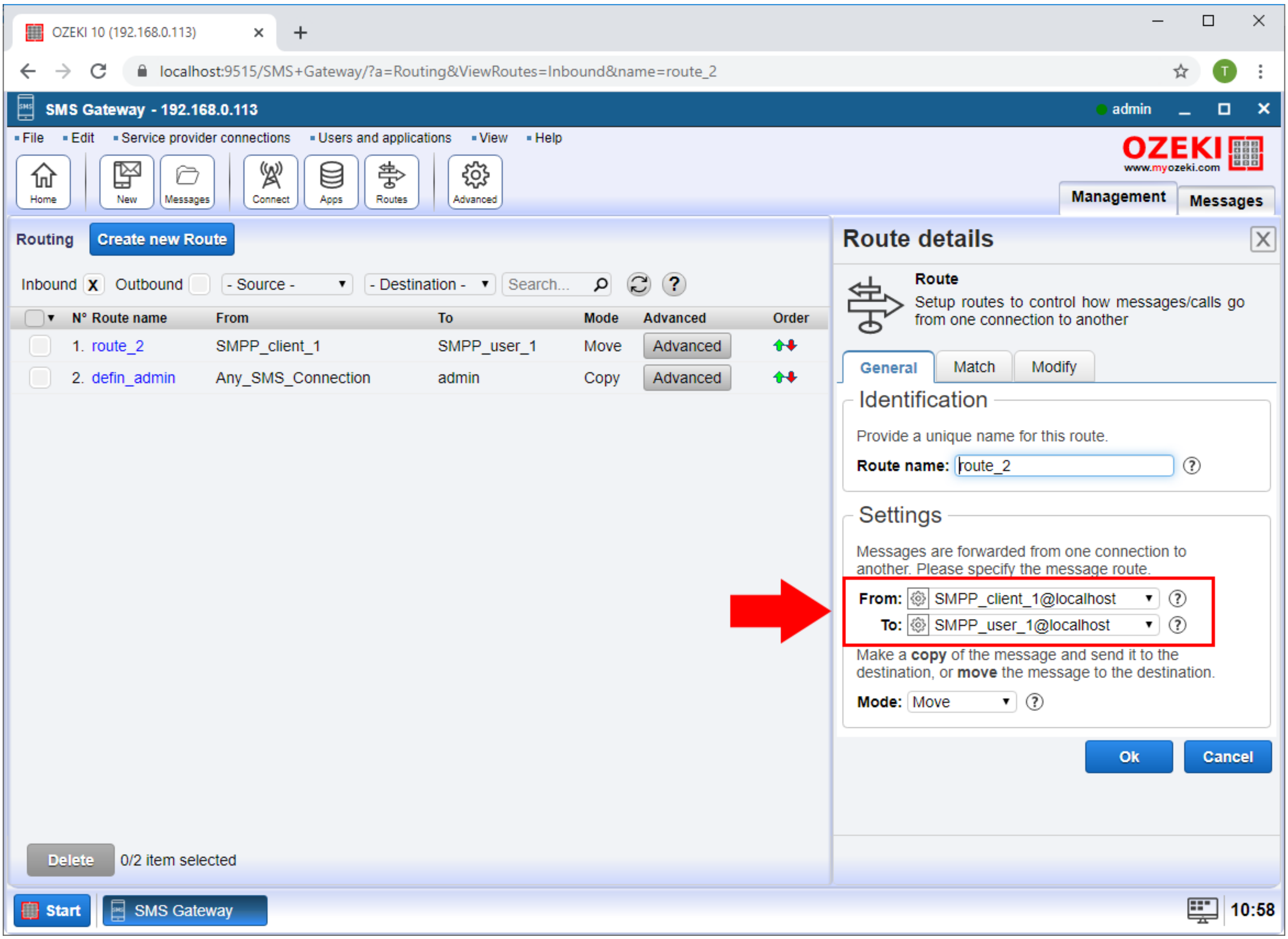
● admin (Admin) 0.00 MPM
Ready

● SMPP_user_1 (SMPP user) 0.00 MPM
Ready

Add new user/application...

4 users/applications installed

Figure 10 - Create an inbound SMS route



| N° | Route name | From | To | Mode | Advanced |
|---|---|---|---|---|---|
| 1. | route_2 | SMPP_client_1 | SMPP_user_1 | Move | Advanced |
| 2. | defin_admin | Any_SMS_Connection | admin | Copy | Advanced |

**Route details**

**Route**
Setup routes to control how messages/calls go from one connection to another

General | Match | Modify

Identification

Provide a unique name for this route.

**Route name:** route_2

Settings

Messages are forwarded from one connection to another. Please specify the message route.

**From:** SMPP_client_1@localhost

**To:** SMPP_user_1@localhost

Make a **copy** of the message and send it to the destination, or **move** the message to the destination.

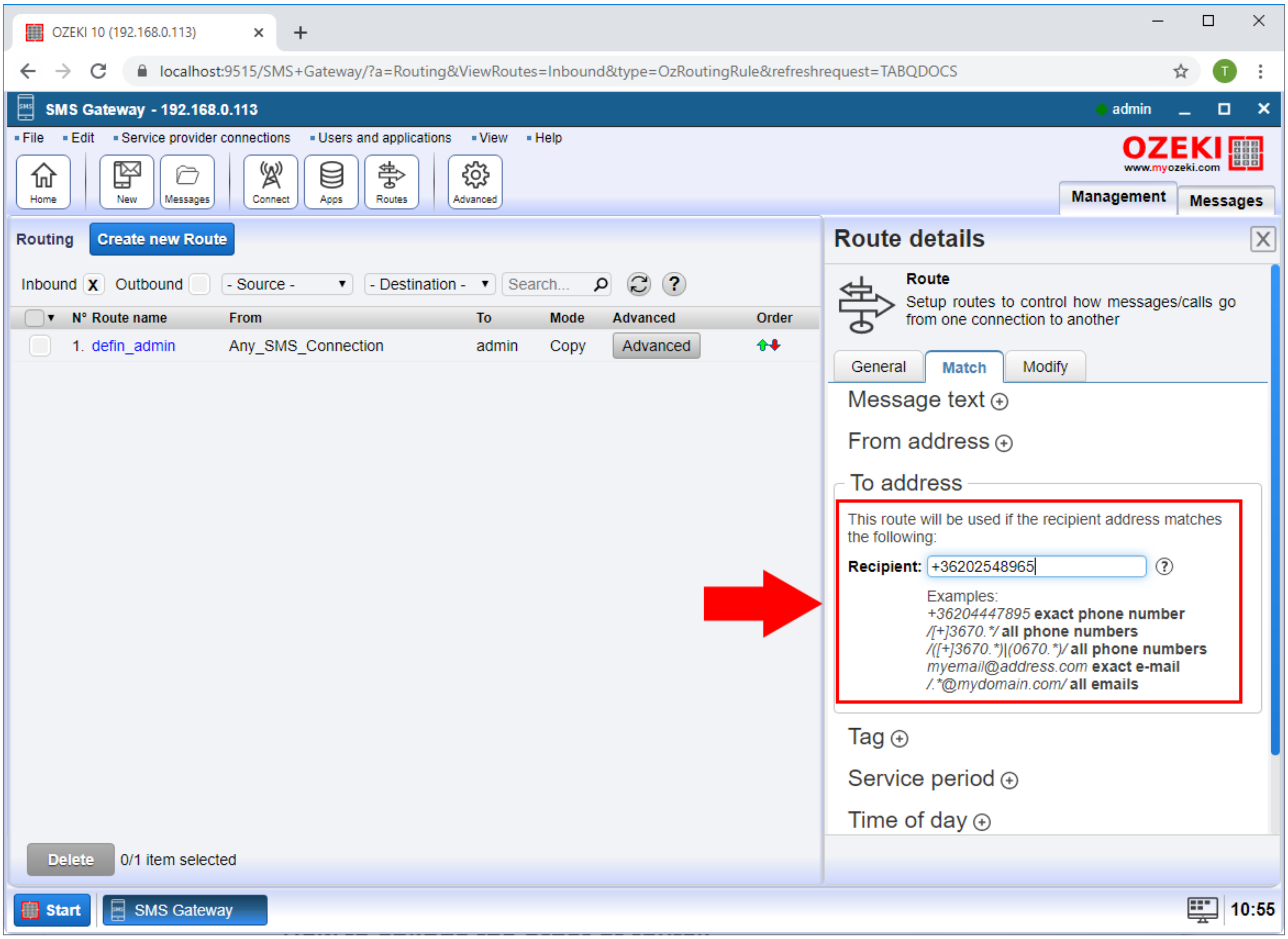
**Mode:** Move

Ok Cancel

Figure 11 - Create an inbound SMS route

# Step 7 - Learn how to route incoming SMS messages by recipient phone number

In the SMS Gateway, during routing, you have the option to select which user connection the message will be forwarded to based on the recipient's phone number. On the Match tab, in the To address field, you can enter the phone number you want to use in the given routing rule.



| N° | Route name | From | To | Mode | Advanced | Order |
|---|---|---|---|---|---|---|
| 1. | defin_admin | Any_SMS_Connection | admin | Copy | Advanced | |

0/1 item selected

Route details

Route
Setup routes to control how messages/calls go from one connection to another

General | Match | Modify

Message text ⊕

From address ⊕

To address

This route will be used if the recipient address matches the following:

Recipient: +36202548965

Examples:
*+36204447895* **exact phone number**
*/[+]3670.*/* **all phone numbers**
*/([+]3670.*)|(0670.*)/* **all phone numbers**
*myemail@address.com* **exact e-mail**
*/.*@mydomain.com/* **all emails**

Tag ⊕

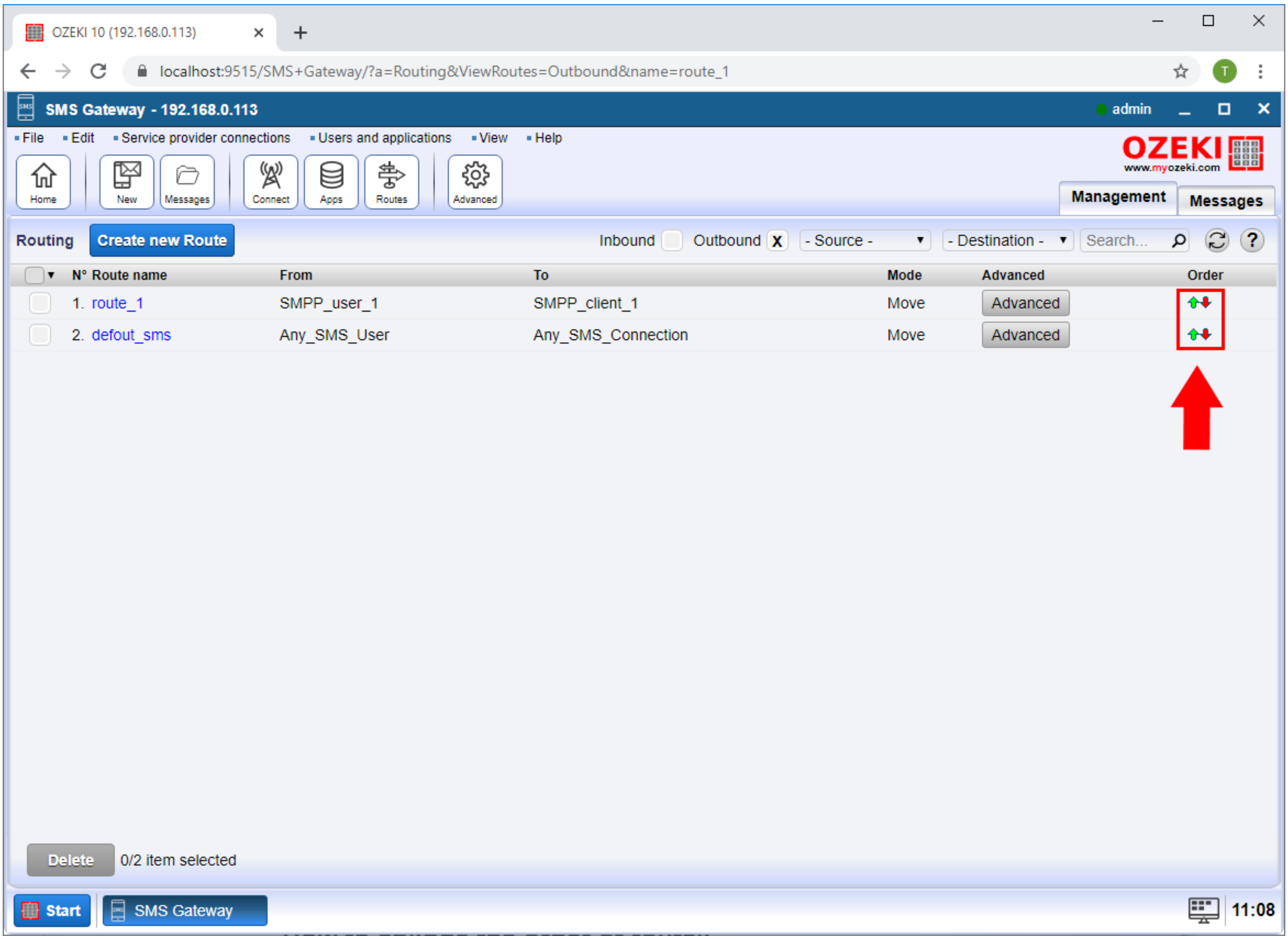
Service period ⊕

Time of day ⊕

Figure 12 - Route based on recipient phone number

# Step 8 - Learn how to change the order of routes

The order of the routing rules determines the order in which the Ozeki SMS gateway scans them. In the routing table, you can use the arrows in the order section to change the order of the rules.



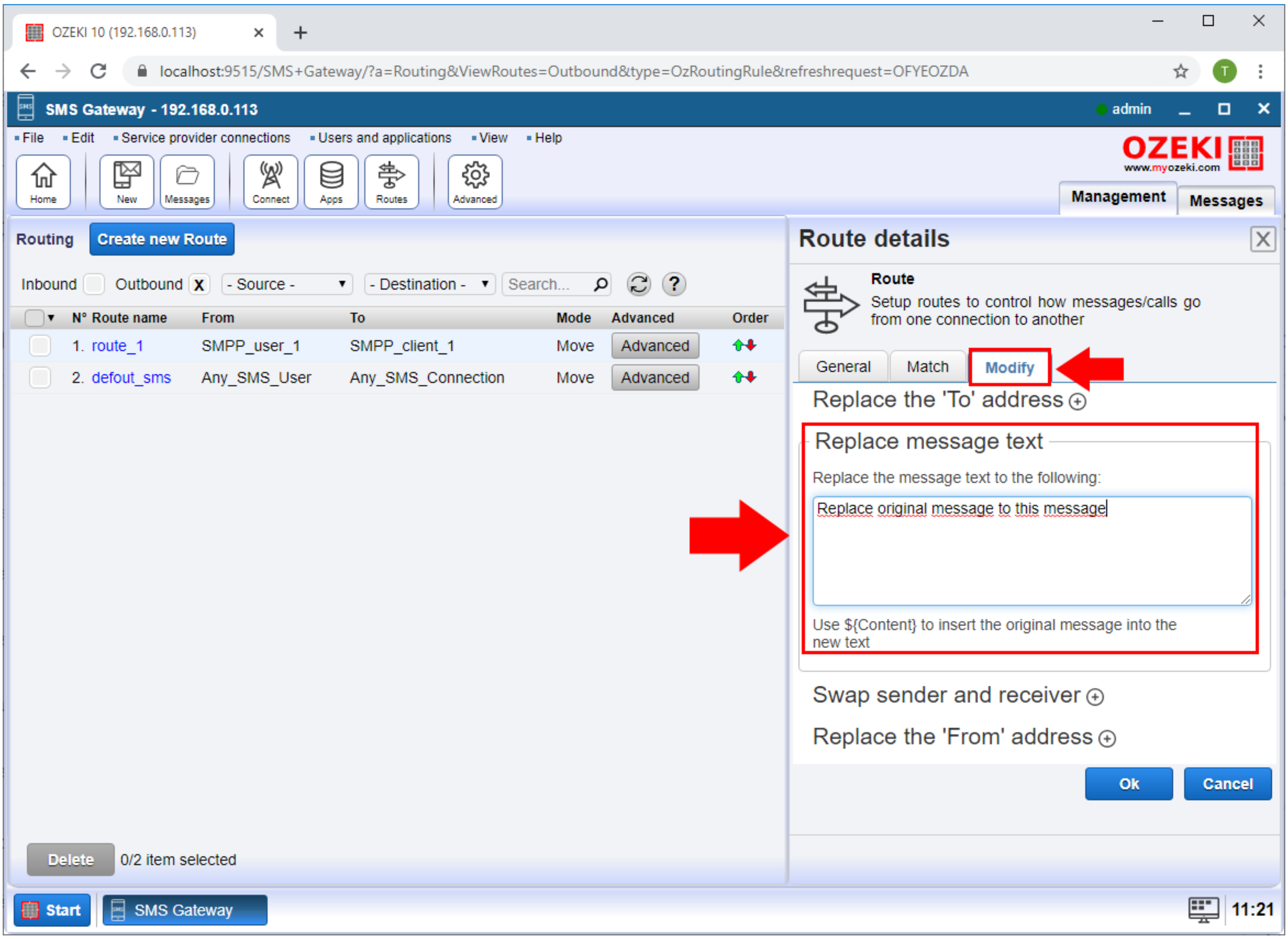
| N° | Route name | From | To | Mode | Advanced | Order |
|---|---|---|---|---|---|---|
| 1. | route_1 | SMPP_user_1 | SMPP_client_1 | Move | Advanced | ↑↓ |
| 2. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move | Advanced | ↑↓ |

0/2 item selected

Figure 13 - Change the order of routes

## Step 9 - Optionally modify the SMS message text

When sending an SMS, you have the option to modify the sent message during routing. You can do this in the Replace message Text field on the modify tab of the routing rule.



| N° | Route name | From | To | Mode | Advanced |
|---|---|---|---|---|---|
| 1. | route_1 | SMPP_user_1 | SMPP_client_1 | Move | Advanced |
| 2. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move | Advanced |

Route
Setup routes to control how messages/calls go from one connection to another

Replace the 'To' address ⊕

Replace message text

Replace the message text to the following:

Replace original message to this message

Use ${Content} to insert the original message into the new text

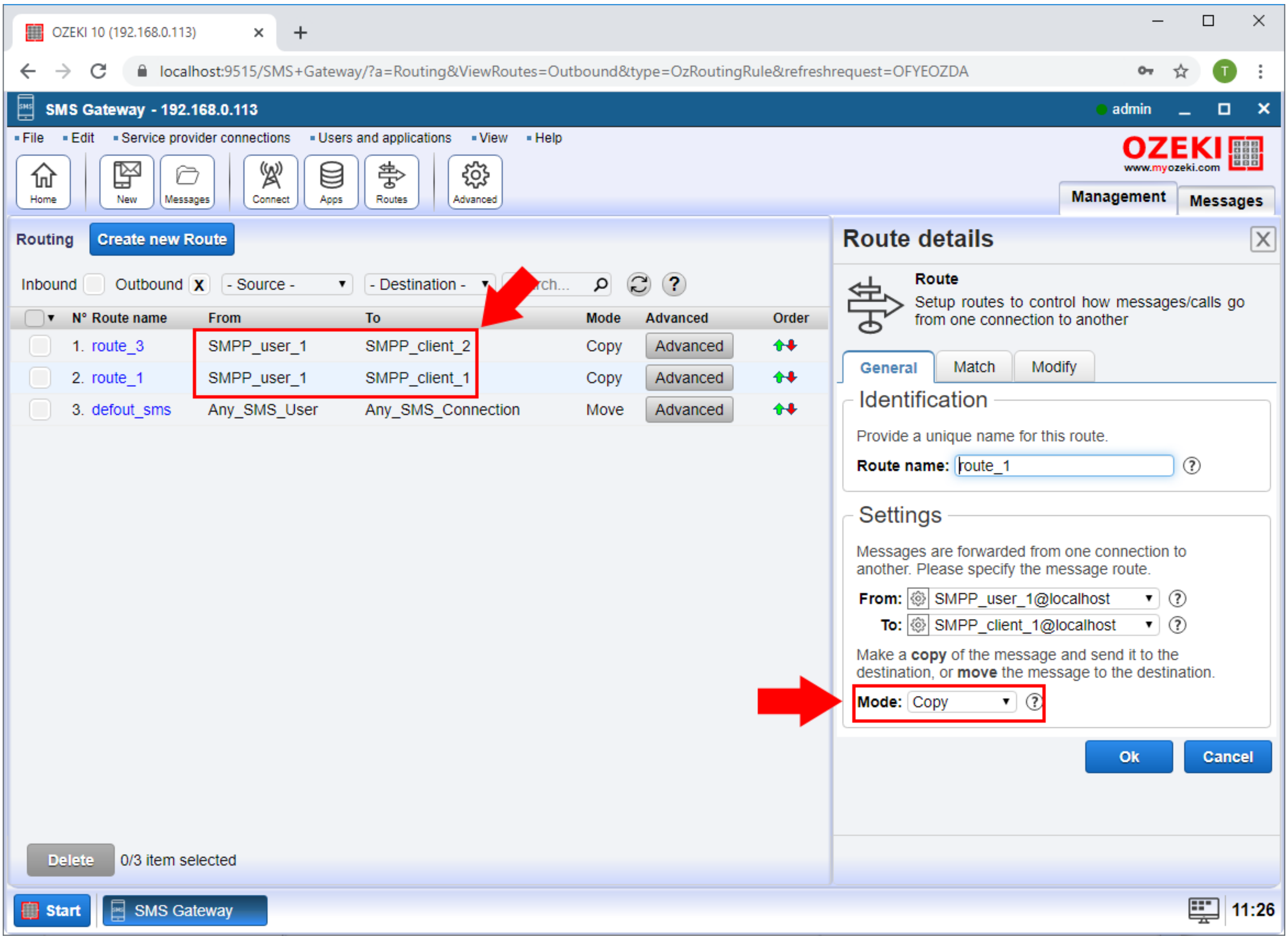
Swap sender and receiver ⊕

Replace the 'From' address ⊕

Figure 14 - Modify the SMS message text

## Step 10 - How to create a message copy

To send a message over two connections with the Ozeki SMS gateway, all you have to do is create two routing rules. For both, the From connection should be the user whom you want to send the message and the To connection should be the two service providers which you want to send the SMS. Finally set Mode to Copy for both rules.



| N° | Route name | From | To | Mode | Advanced | Order |
|---|---|---|---|---|---|---|
| 1. | route_3 | SMPP_user_1 | SMPP_client_2 | Copy | Advanced | |
| 2. | route_1 | SMPP_user_1 | SMPP_client_1 | Copy | Advanced | |
| 3. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move | Advanced | |

Delete 0/3 item selected

**Route details**

**Route**
Setup routes to control how messages/calls go from one connection to another

General | Match | Modify

**Identification**

Provide a unique name for this route.

**Route name:** route_1

**Settings**

Messages are forwarded from one connection to another. Please specify the message route.

**From:** SMPP_user_1@localhost

**To:** SMPP_client_1@localhost

Make a **copy** of the message and send it to the destination, or **move** the message to the destination.

**Mode:** Copy

Ok Cancel

Figure 15 - Send a message to two connections

# How to copy an SMS

***The following guide is going to demonstrate how you can copy an SMS message from one user to another user. This is a useful tool if you would like to route the delivered message to multiple users. The guide shows how to create standard users in SMS Gateway and how to route the incoming messages to users to copy the message. This document contains a video tutorial and a step by step guide which makes the whole procedure much easier to complete since it does not require any further specific knowledge. This guide takes about ten minutes to complete. So, let's get started!***

## What does SMS stand for?

SMS stands for Short Message Service. SMS is one of the oldest texting technology since it was invented in the 1980s and defined in 1985. It is still the most widespread and frequently used technology.

## What is an HTTP server?

HTTP server is basically a web server that is capable to communicate using the HTTP protocol. Its main objective is to process and deliver data to the users by using the HTTP protocol.

## Connection steps

1. Open Ozeki 10 GUI by typing 'https://localhost:9515' into your borwser
2. Open the SMS Gateway application
3. Create two Standard user connections
4. Create a HTTP Server connection
5. Configure the route of the first Standard user connection
6. Configure the route of the second Standard user connection
7. Send test message using the HTTP Server connection
8. Check the inbox folder of the Standard user connections

## Video tutorial

## Step 1 - Create a standard user connection

The first step of this guide is to create two standard users which makes it possible to copy the message to each user connection that you made. So, now let's create the first of the standard users. To do that, the first step is to open the SMS Gateway. For that, open Ozeki 10 GUI by typing 'https://localhost:9515' into your browser. Then, as in Figure 1, open the SMS Gateway application from the Ozeki 10 desktop.

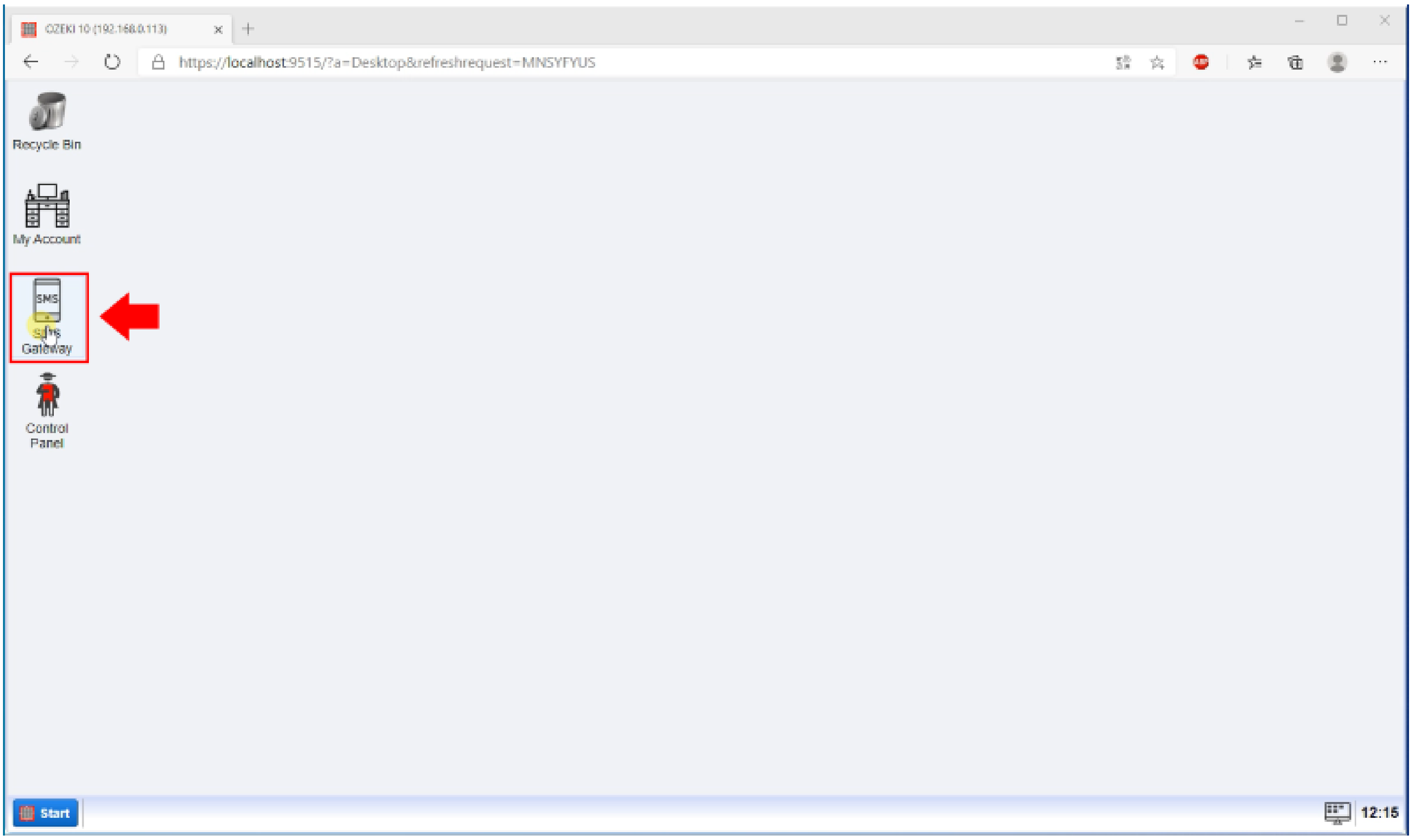
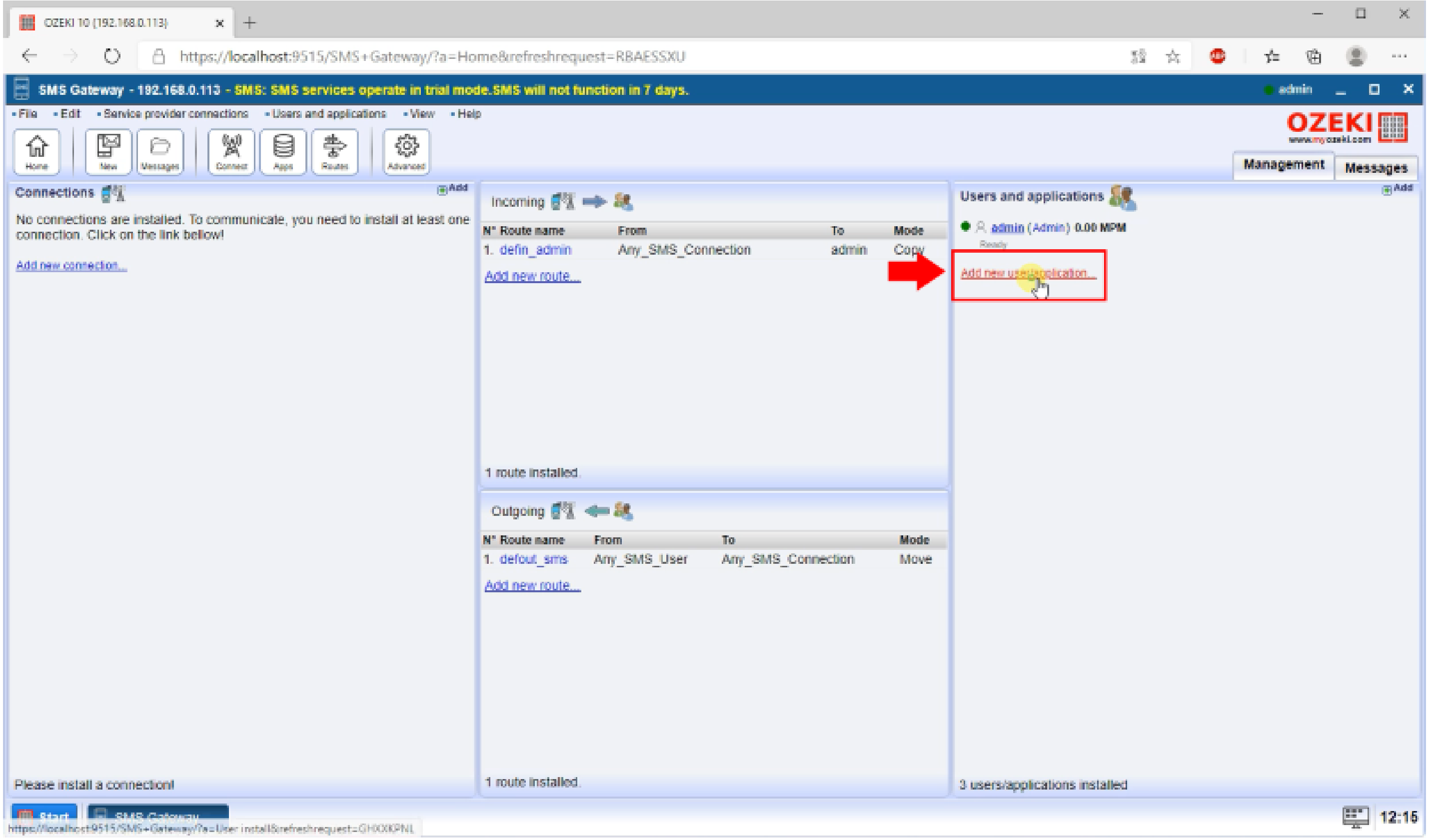


Figure 1 - Open SMS Gateway application

After you opened the SMS Gateway application, you will be able to see the main menu of the SMS Gateway. This main menu contains three main sections: Connections, Routes and Users and Applications. Now, at this point, you need to create a user in the SMS Gateway. To perform this operation, just click on 'Add new user/application...' as you can see it in Figure 2.



No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

Please install a connection!

| N° Route name | From | To | Mode |
|---|---|---|---|
| 1. defin_admin | Any_SMS_Connection | admin | Copy |

Add new route...

1 route installed.

| N° Route name | From | To | Mode |
|---|---|---|---|
| 1. defout_sms | Any_SMS_User | Any_SMS_Connection | Move |

Add new route...

1 route installed.

admin (Admin) 0.00 MPM

Ready

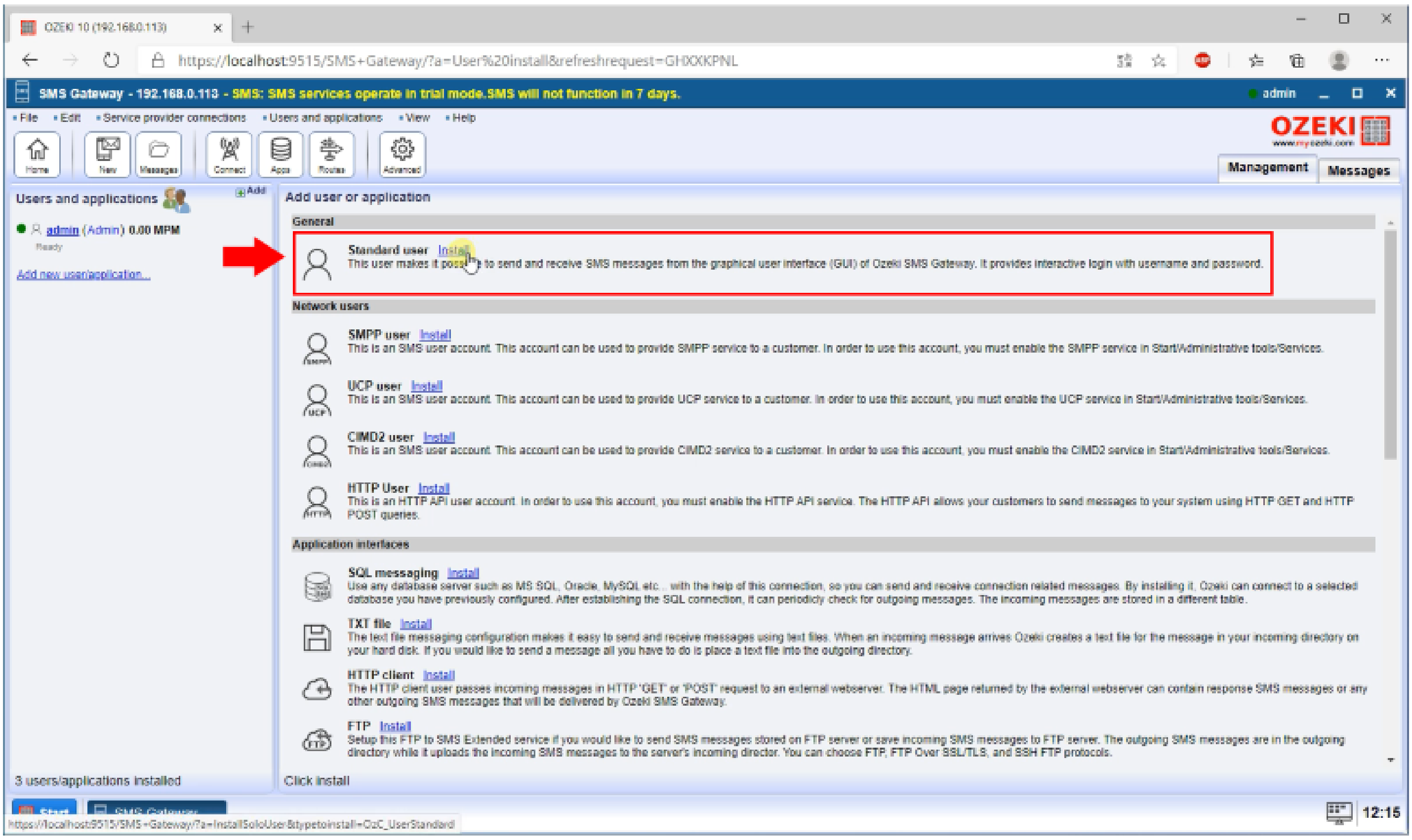
Add new user/application...

3 users/applications installed

https://localhost:9515/SMS+Gateway/?a=User install&refreshrequest=GHXXKPNL

Figure 2 - Add new user or application

The next menu that shows up, lists all the available users and applications that can be created in SMS Gateway. Every option has got a short description that helps you to decide which application is needed in your case. To follow this guide, you just need to have a simple Standard user, so as Figure 3 shows that, just click on the 'Install' button of the Standard user connection.


https://localhost:9515/SMS+Gateway/?a=User%20install&refreshrequest=GHXXKPNL

SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.

Add user or application

General

Standard user Install
This user makes it possible to send and receive SMS messages from the graphical user interface (GUI) of Ozeki SMS Gateway. It provides interactive login with username and password.

Network users

SMPP user Install
This is an SMS user account. This account can be used to provide SMPP service to a customer. In order to use this account, you must enable the SMPP service in Start/Administrative tools/Services.

UCP user Install
This is an SMS user account. This account can be used to provide UCP service to a customer. In order to use this account, you must enable the UCP service in Start/Administrative tools/Services.

CIMD2 user Install
This is an SMS user account. This account can be used to provide CIMD2 service to a customer. In order to use this account, you must enable the CIMD2 service in Start/Administrative tools/Services.

HTTP User Install
This is an HTTP API user account. In order to use this account, you must enable the HTTP API service. The HTTP API allows your customers to send messages to your system using HTTP GET and HTTP POST queries.

Application interfaces

SQL messaging Install
Use any database server such as MS SQL, Oracle, MySQL etc... with the help of this connection, so you can send and receive connection related messages. By installing it, Ozeki can connect to a selected database you have previously configured. After establishing the SQL connection, it can periodicly check for outgoing messages. The incoming messages are stored in a different table.

TXT file Install
The text file messaging configuration makes it easy to send and receive messages using text files. When an incoming message arrives Ozeki creates a text file for the message in your incoming directory on your hard disk. If you would like to send a message all you have to do is place a text file into the outgoing directory.

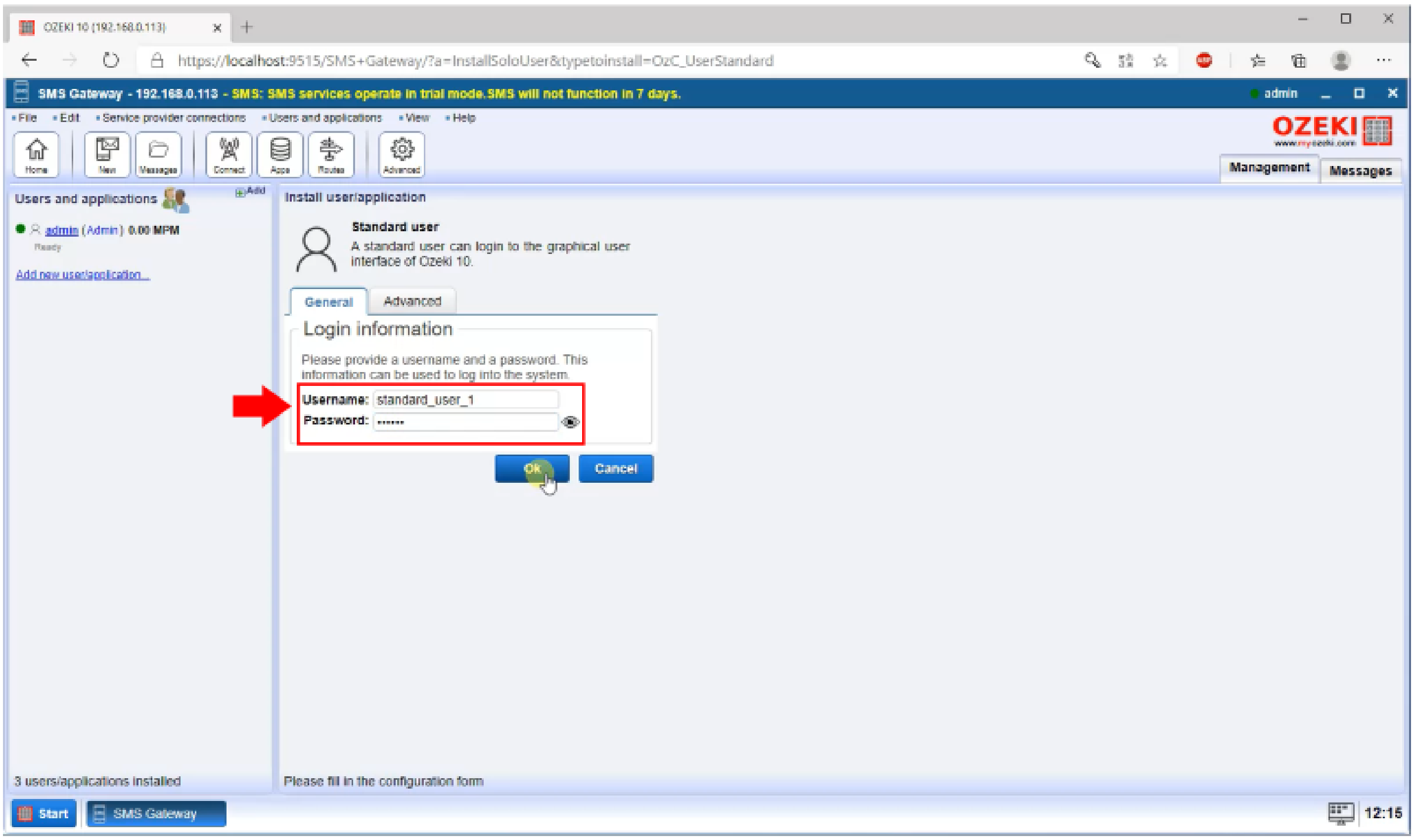
HTTP client Install
The HTTP client user passes incoming messages in HTTP 'GET' or 'POST' request to an external webserver. The HTML page returned by the external webserver can contain response SMS messages or any other outgoing SMS messages that will be delivered by Ozeki SMS Gateway.

FTP Install
Setup this FTP to SMS Extended service if you would like to send SMS messages stored on FTP server or save incoming SMS messages to FTP server. The outgoing SMS messages are in the outgoing directory while it uploads the incoming SMS messages to the server's incoming director. You can choose FTP, FTP Over SSL/TLS, and SSH FTP protocols.

Users and applications
admin (Admin) 0.00 MPM
Add new user/application...
3 users/applications installed
Click install

Figure 3 - Install standard user

Before finishing the creation of the Standard user connection, you have to specify some details for the connection. These details in this case are a username and a password for the connection. So, all you need to do is to provide these details for the connection like in Figure 4 and finally, click on Ok to create the first of the two Standard user connections that needed to complete this guide.


https://localhost:9515/SMS+Gateway/?a=InstallSoloUser&typetoinstall=OzC_UserStandard

SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.

Install user/application

Standard user
A standard user can login to the graphical user interface of Ozeki 10.

Login information

Please provide a username and a password. This information can be used to log into the system.

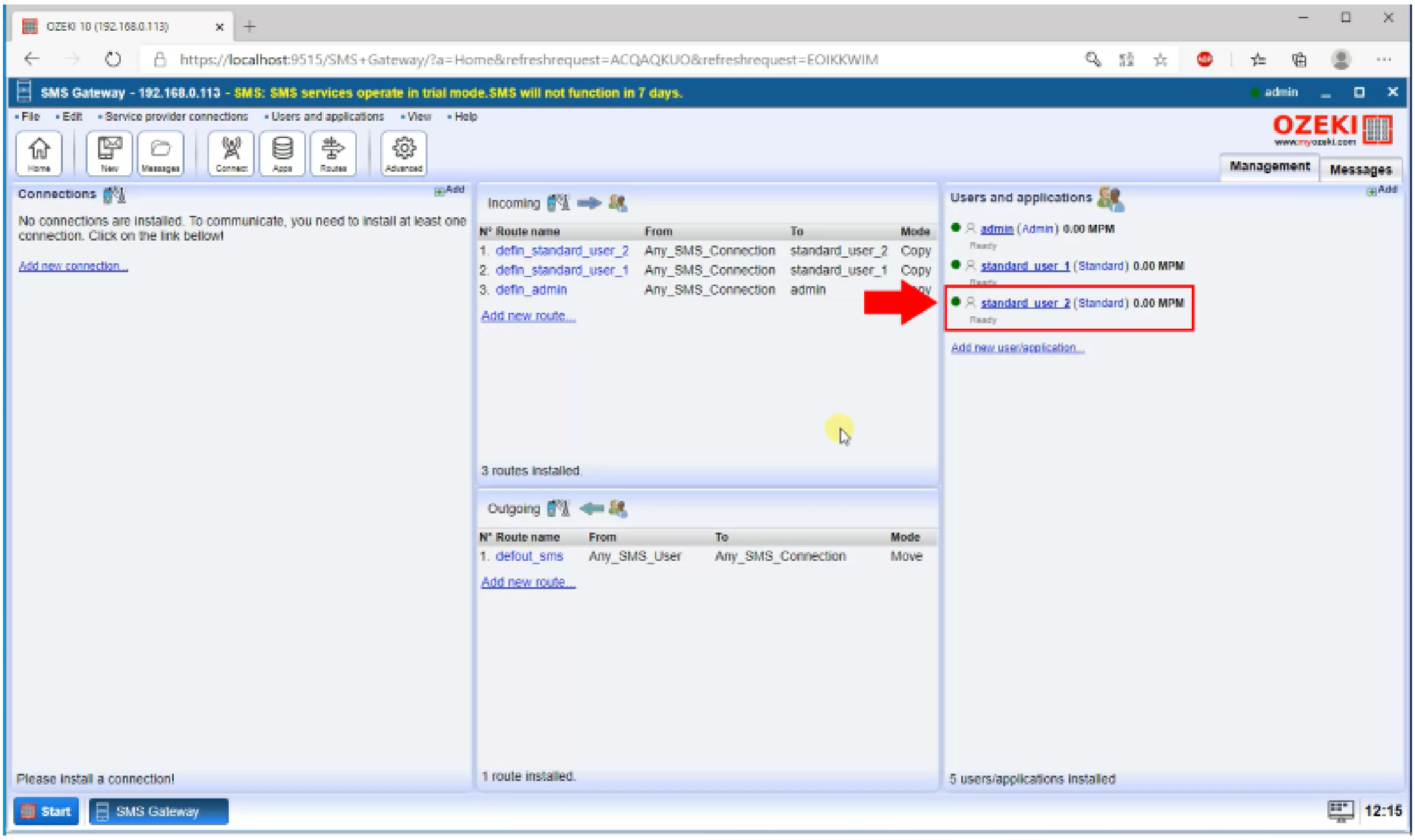
Username: standard_user_1
Password: ••••••

Users and applications
admin (Admin) 0.00 MPM
Add new user/application...
3 users/applications installed
Please fill in the configuration form

Figure 4 - Define username and password

# Step 2 - Create the second standard user connection

To see the effects of how Ozeki 10 SMS Gateway copies the SMS messages for multiple connections, you need to create a second Standard user connection. To be able to do this, you just need to follow the instructions of Step 1, since it needs to be created the same way. The only thing that have to be different is the username and the password of the connection. After you created the connection, it shows up as you can see it in Figure 5.



SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.

No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

Please install a connection!

Incoming

| N° Route name | From | To | Mode |
|---|---|---|---|
| 1. defin_standard_user_2 | Any_SMS_Connection | standard_user_2 | Copy |
| 2. defin_standard_user_1 | Any_SMS_Connection | standard_user_1 | Copy |
| 3. defin_admin | Any_SMS_Connection | admin | Copy |

Add new route...

3 routes installed.

Outgoing

| N° Route name | From | To | Mode |
|---|---|---|---|
| 1. defout_sms | Any_SMS_User | Any_SMS_Connection | Move |

Add new route...

1 route installed.

- admin (Admin) 0.00 MPM Ready
- standard_user_1 (Standard) 0.00 MPM Ready
- standard_user_2 (Standard) 0.00 MPM Ready

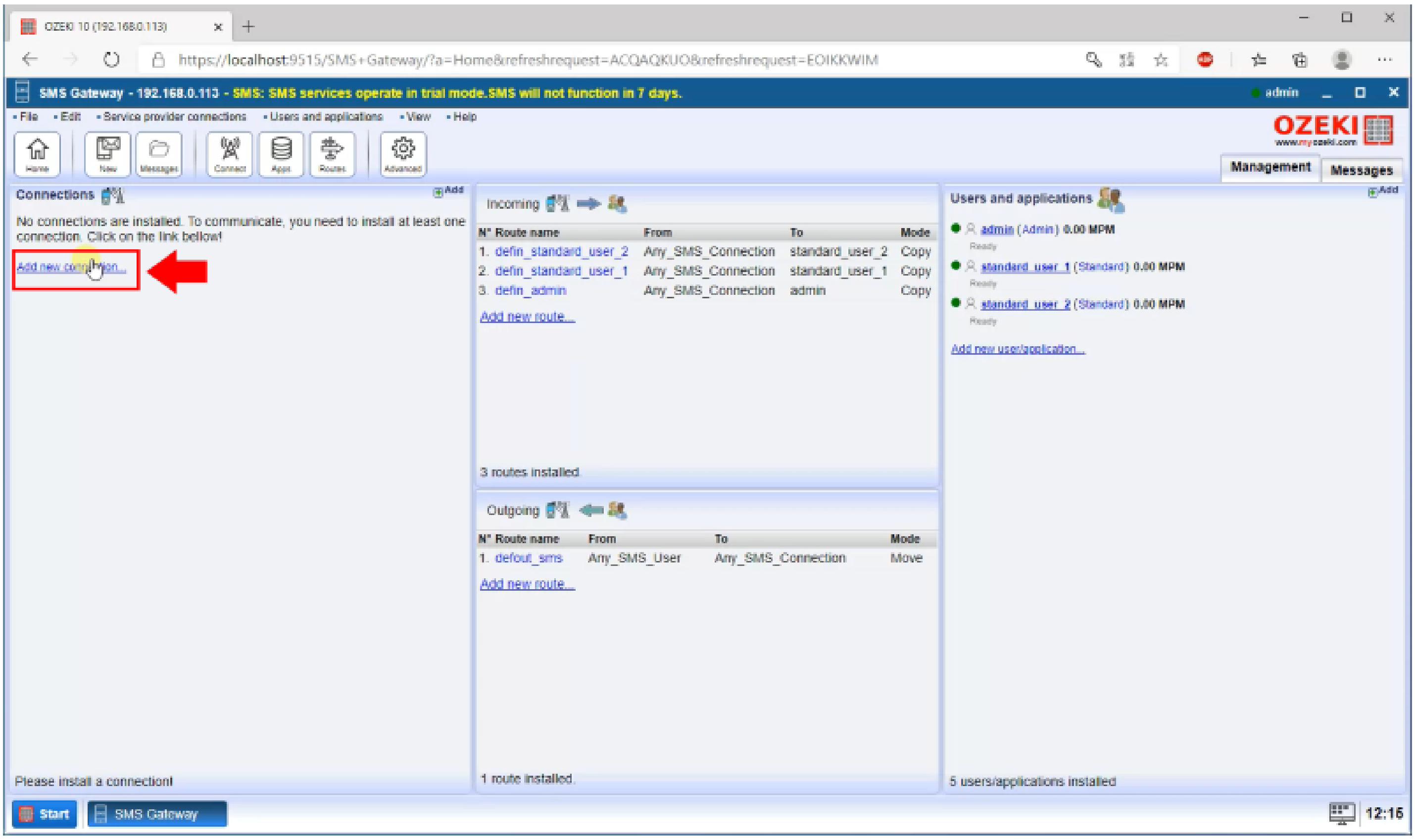
Add new user/application...

5 users/applications installed

Figure 5 - Create the second Standard user connection

# Step 3 - Create HTTP Server connection

The next step of the guide that you need to complete is to create an HTTP Server connection that is capable of sending SMS messages. This way, you will be able to see how the SMS messages are routed to the Standard user connections. To create a new HTTP Server connection, first, select the 'Add new connection...' option from the main menu of the SMS Gateway as Figure 6 demonstrates that.


OZEKI 10 (192.168.0.113)
https://localhost:9515/SMS+Gateway/?a=Home&refreshrequest=ACQAQKUO&refreshrequest=EOIKKWIM
SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
OZEKI
Management
Messages
Add
Start
SMS Gateway
12:16

Connections

No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

Please install a connection!

Incoming

| N° Route name | From | To | Mode |
|---|---|---|---|
| 1. defin_standard_user_2 | Any_SMS_Connection | standard_user_2 | Copy |
| 2. defin_standard_user_1 | Any_SMS_Connection | standard_user_1 | Copy |
| 3. defin_admin | Any_SMS_Connection | admin | Copy |

Add new route...

3 routes installed

Outgoing

| N° Route name | From | To | Mode |
|---|---|---|---|
| 1. defout_sms | Any_SMS_User | Any_SMS_Connection | Move |

Add new route...

1 route installed.

Users and applications

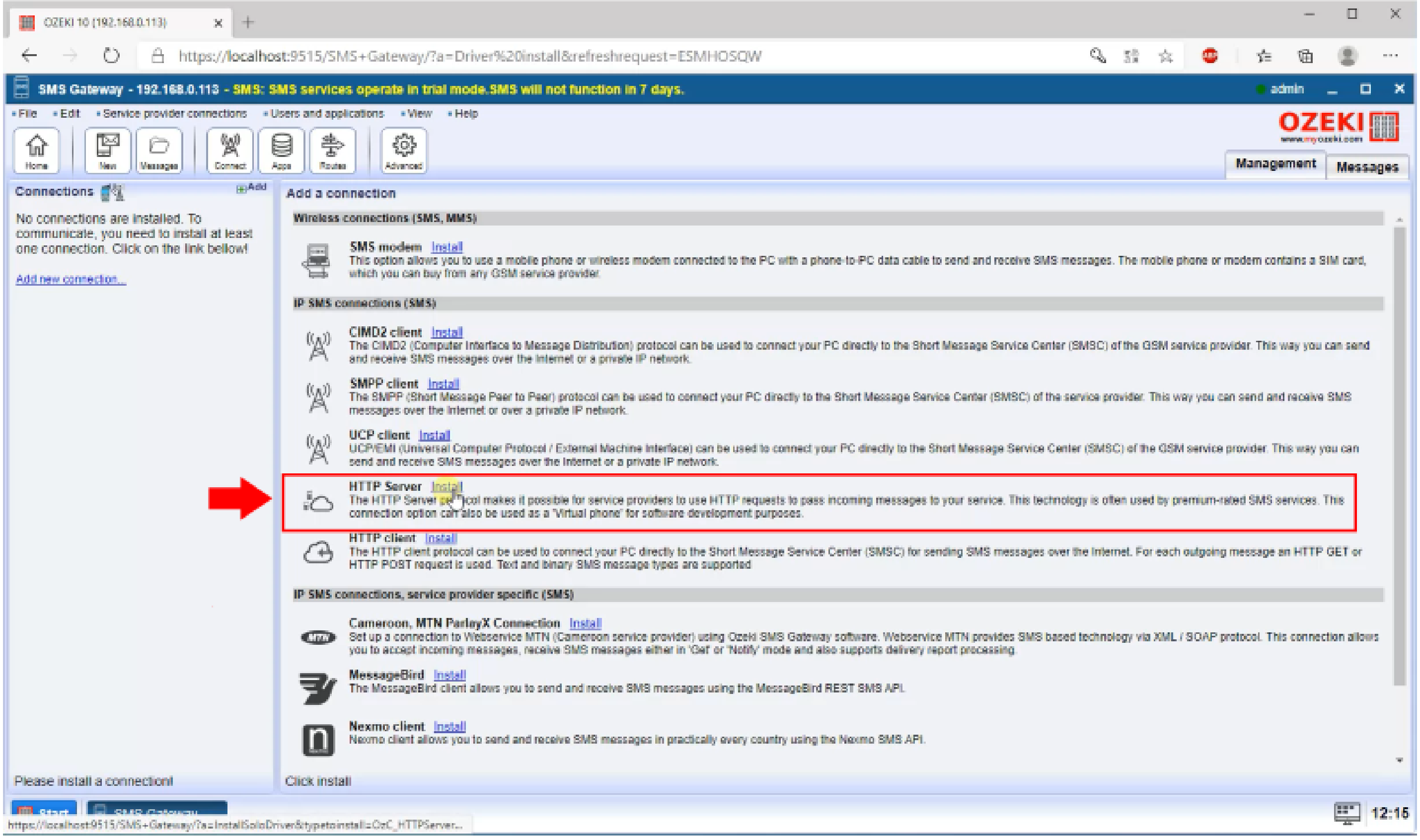
- admin (Admin) 0.00 MPM Ready
- standard_user_1 (Standard) 0.00 MPM Ready
- standard_user_2 (Standard) 0.00 MPM Ready

Add new user/application...

5 users/applications installed

Figure 6 - Add new connection

Next, the selection menu of the connection is going to show up, where you can choose from a lot of available connections with some description and choose which one suits your solution the most. Now, you need to choose the HTTP Server connection, so as you can see it in Figure 7, click on the 'Install' button of that connection the move to the installation menu, where you can specify some details of the connection.


OZEKI 10 (192.168.0.113)
https://localhost:9515/SMS+Gateway/?a=Driver%20install&refreshrequest=ESMHOSQW
SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
OZEKI
Management
Messages
Add
Start
SMS Gateway
12:15
https://localhost:9515/SMS+Gateway/?a=InstallSoloDriver&typetoinstall=OzC_HTTPServer...

Connections

No connections are installed. To communicate, you need to install at least one connection. Click on the link bellow!

Add new connection...

Please install a connection!

Add a connection

Wireless connections (SMS, MMS)

SMS modem Install
This option allows you to use a mobile phone or wireless modem connected to the PC with a phone-to-PC data cable to send and receive SMS messages. The mobile phone or modem contains a SIM card, which you can buy from any GSM service provider.

IP SMS connections (SMS)

CIMD2 client Install
The CIMD2 (Computer Interface to Message Distribution) protocol can be used to connect your PC directly to the Short Message Service Center (SMSC) of the GSM service provider. This way you can send and receive SMS messages over the Internet or a private IP network.

SMPP client Install
The SMPP (Short Message Peer to Peer) protocol can be used to connect your PC directly to the Short Message Service Center (SMSC) of the service provider. This way you can send and receive SMS messages over the Internet or over a private IP network.

UCP client Install
UCP/EMI (Universal Computer Protocol / External Machine Interface) can be used to connect your PC directly to the Short Message Service Center (SMSC) of the GSM service provider. This way you can send and receive SMS messages over the Internet or a private IP network.

HTTP Server Install
The HTTP Server protocol makes it possible for service providers to use HTTP requests to pass incoming messages to your service. This technology is often used by premium-rated SMS services. This connection option can also be used as a 'Virtual phone' for software development purposes.

HTTP client Install
The HTTP client protocol can be used to connect your PC directly to the Short Message Service Center (SMSC) for sending SMS messages over the Internet. For each outgoing message an HTTP GET or HTTP POST request is used. Text and binary SMS message types are supported

IP SMS connections, service provider specific (SMS)

Cameroon, MTN ParlayX Connection Install
Set up a connection to Webservice MTN (Cameroon service provider) using Ozeki SMS Gateway software. Webservice MTN provides SMS based technology via XML / SOAP protocol. This connection allows you to accept incoming messages, receive SMS messages either in 'Get' or 'Notify' mode and also supports delivery report processing.

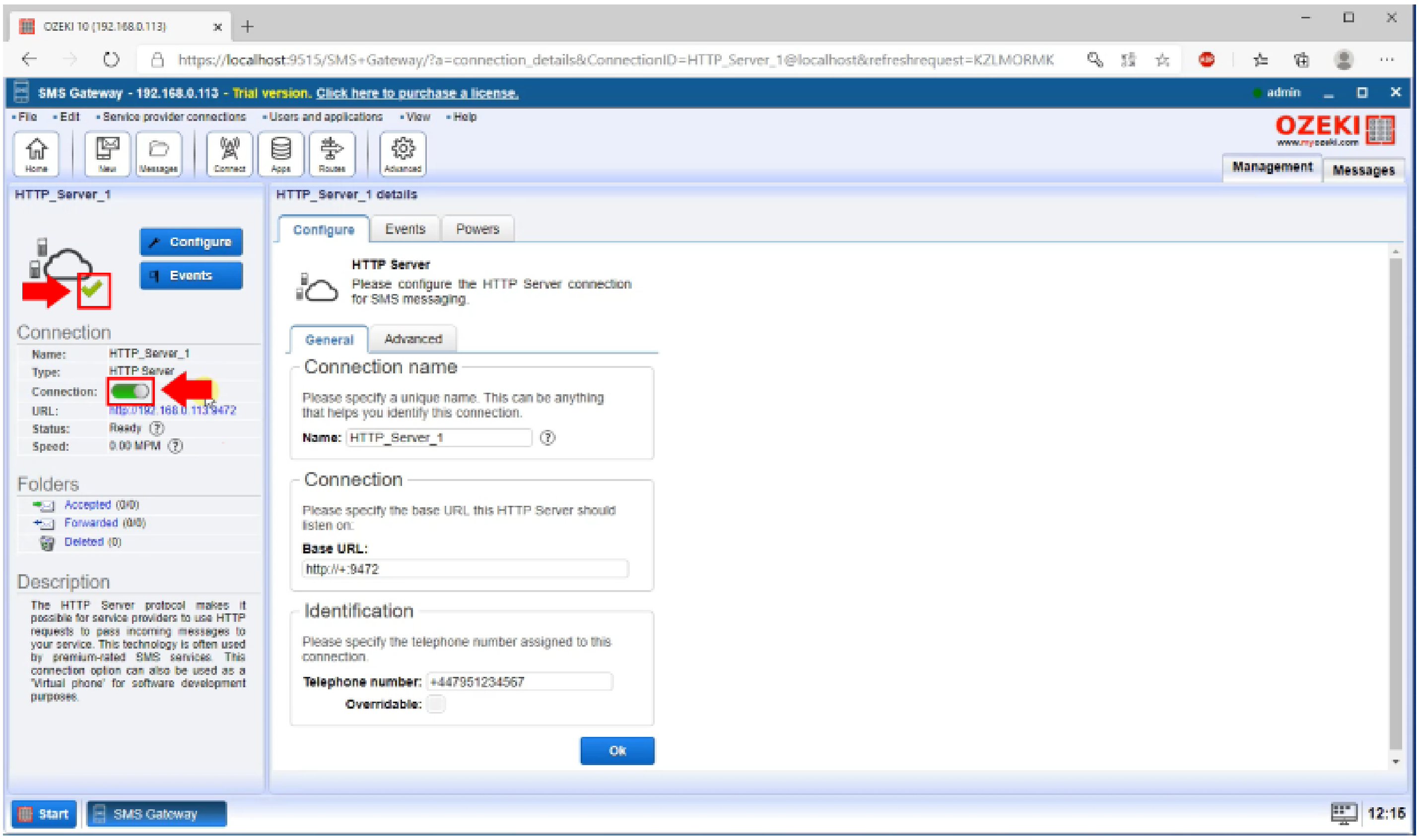
MessageBird Install
The MessageBird client allows you to send and receive SMS messages using the MessageBird REST SMS API.

Nexmo client Install
Nexmo client allows you to send and receive SMS messages in practically every country using the Nexmo SMS API.

Click install

Figure 7 - Install HTTP server

Finally, before you can create the HTTP Server connection, you have to provide the main details of the connection. The first thing that you have to specify here is the name of the connection. Next, you can choose the port, where the HTTP Server is going to run. You can modify this port, you can just leave it as the default port number. Lastly, specify a telephone number for this connection which makes it easier to identify. After you created the connection, you need to enable it by turning on the toggle like in Figure 8. If it is successfully enabled you will be able to see the green tick on the connection.


OZEKI 10 (192.168.0.113)
https://localhost:9515/SMS+Gateway/?a=connection_details&ConnectionID=HTTP_Server_1@localhost&refreshrequest=KZLMORMK
SMS Gateway - 192.168.0.113 - Trial version. Click here to purchase a license.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
OZEKI
www.myozeki.com
Management
Messages
Configure
Events
Powers
General
Advanced
Ok
Start
SMS Gateway
12:15

HTTP_Server_1

Connection

Name: HTTP_Server_1
Type: HTTP Server
Connection:
URL: http://192.168.0.113:9472
Status: Ready
Speed: 0.00 MPM

Folders

Accepted (0/0)
Forwarded (0/0)
Deleted (0)

Description

The HTTP Server protocol makes it possible for service providers to use HTTP requests to pass incoming messages to your service. This technology is often used by premium-rated SMS services. This connection option can also be used as a 'Virtual phone' for software development purposes.

HTTP_Server_1 details

HTTP Server
Please configure the HTTP Server connection for SMS messaging.

Connection name

Please specify a unique name. This can be anything that helps you identify this connection.

Name: HTTP_Server_1

Connection

Please specify the base URL this HTTP Server should listen on.

Base URL:
http://+:9472

Identification

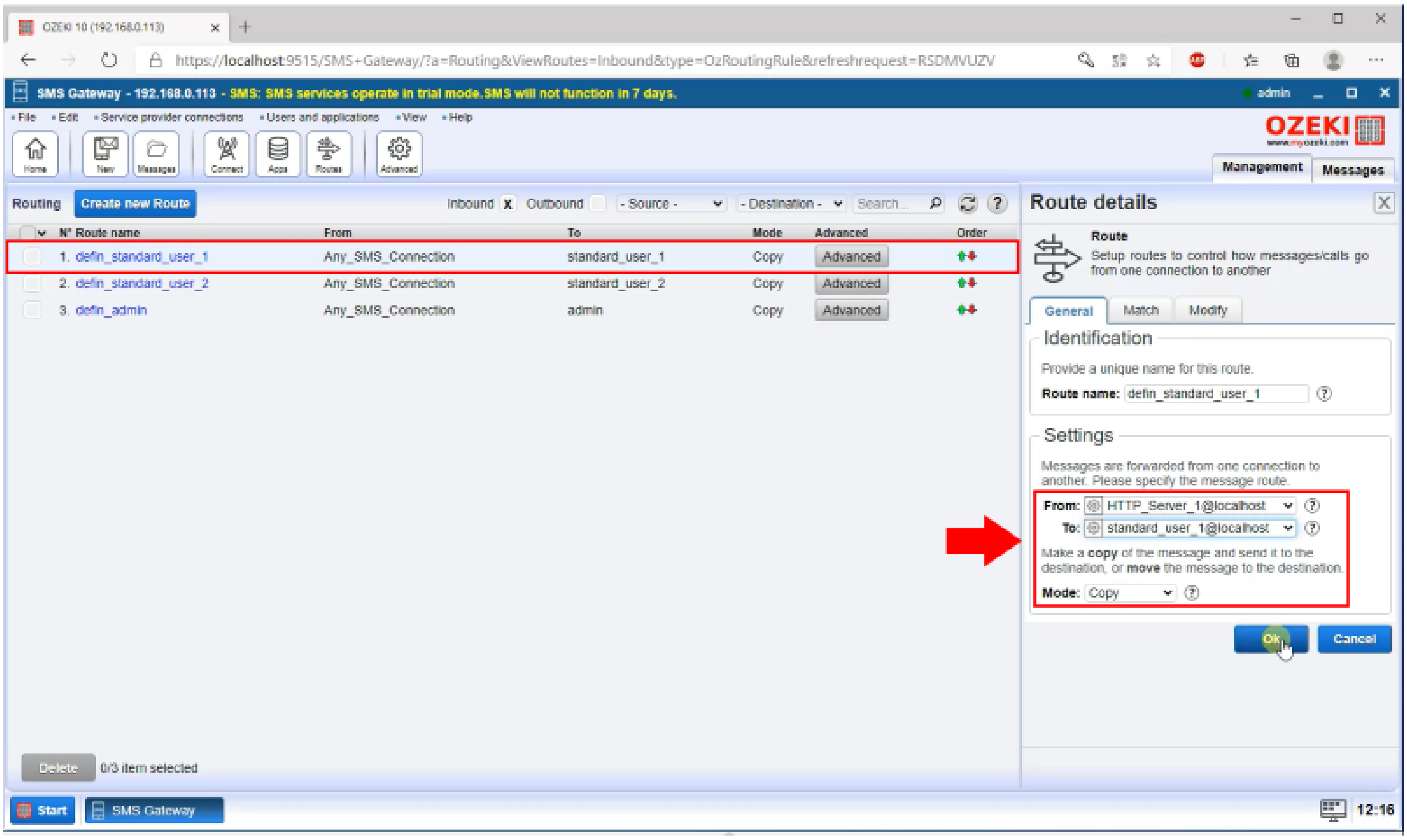
Please specify the telephone number assigned to this connection.

Telephone number: +447951234567
Overridable: □

Figure 8 - Enable HTTP server connection

# Step 4 - Create a route to the first Standard user

After you created all connections that needed, next, you should set up the routes that copy the messages to the user connections. Since the creation of each user connections also created a default route for them, you don't have to create a completely new route, you can just modify the existing one. So, just select the route of the first Standard user connection. Here, you need to modify the 'From' field by setting the HTTP Server connection for that like in Figure 9 and for the 'Mode', select the Copy option. If you have done these settings, just click on OK.



| N° Route name | From | To | Mode | Advanced | Order |
|---|---|---|---|---|---|
| 1. defin_standard_user_1 | Any_SMS_Connection | standard_user_1 | Copy | Advanced | |
| 2. defin_standard_user_2 | Any_SMS_Connection | standard_user_2 | Copy | Advanced | |
| 3. defin_admin | Any_SMS_Connection | admin | Copy | Advanced | |

Route details

Route
Setup routes to control how messages/calls go from one connection to another

General | Match | Modify

Identification

Provide a unique name for this route.

Route name: defin_standard_user_1

Settings

Messages are forwarded from one connection to another. Please specify the message route.

From: HTTP_Server_1@localhost

To: standard_user_1@localhost

Make a **copy** of the message and send it to the destination, or **move** the message to the destination.

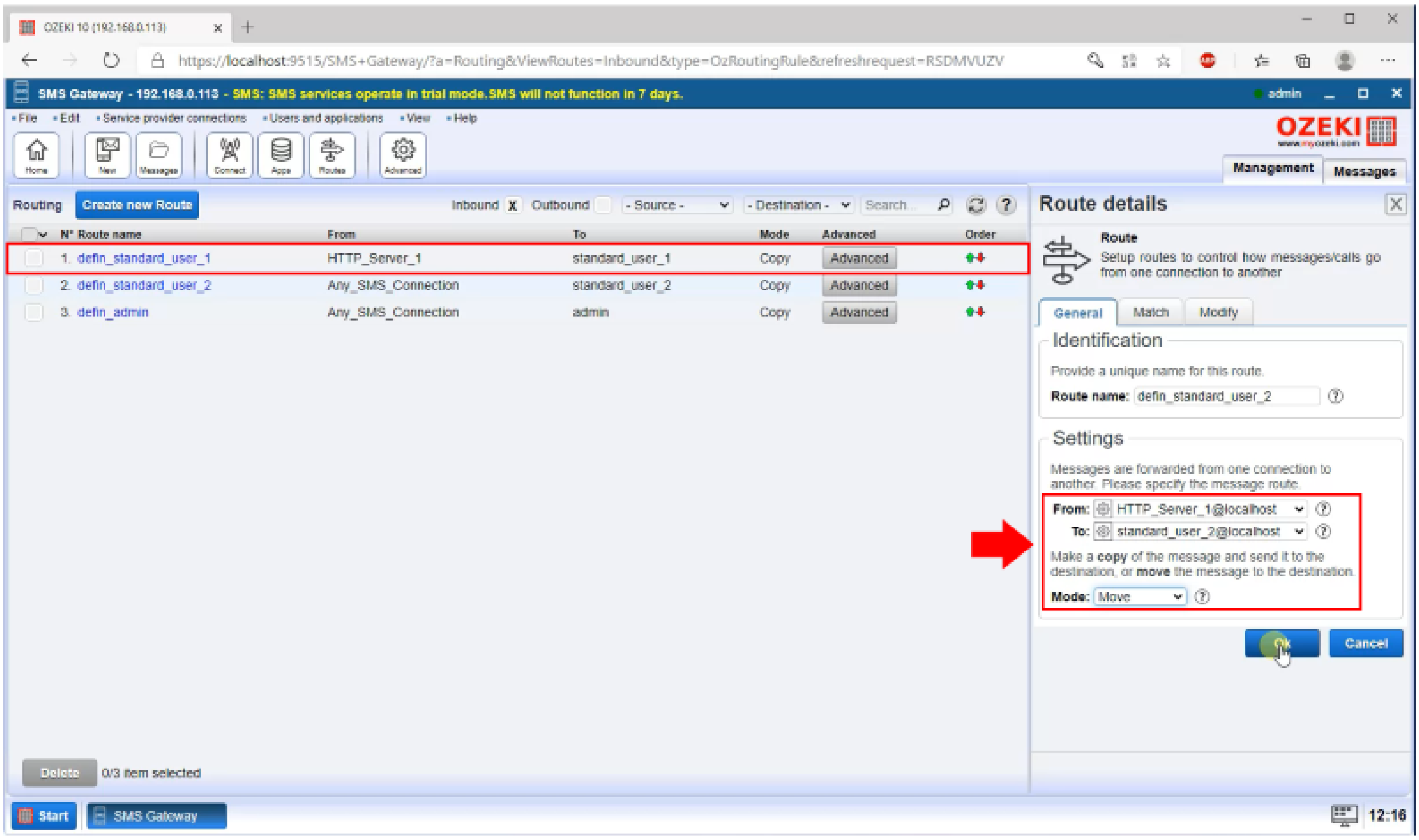
Mode: Copy

Ok | Cancel

Figure 9 - Modify the first route

# Step 5 - Create a route to the second Standard user

The route of the second Standard user connection needs to be set up properly as well. So, as you did with the first user connection, select the route of the second Standard user connection from the list of the routes. Here, you need to select the HTTP Server connection for the 'From' field, and as you can see it in Figure 10, select the 'Move' option as the Mode. Lastly, just click OK to save the modifications.



| N° Route name | From | To | Mode | Advanced | Order |
|---|---|---|---|---|---|
| 1. defin_standard_user_1 | HTTP_Server_1 | standard_user_1 | Copy | Advanced | |
| 2. defin_standard_user_2 | Any_SMS_Connection | standard_user_2 | Copy | Advanced | |
| 3. defin_admin | Any_SMS_Connection | admin | Copy | Advanced | |

Route details

Route
Setup routes to control how messages/calls go from one connection to another

General | Match | Modify

Identification

Provide a unique name for this route.

Route name: defin_standard_user_2

Settings

Messages are forwarded from one connection to another. Please specify the message route.

From: HTTP_Server_1@localhost

To: standard_user_2@localhost

Make a **copy** of the message and send it to the destination, or **move** the message to the destination.

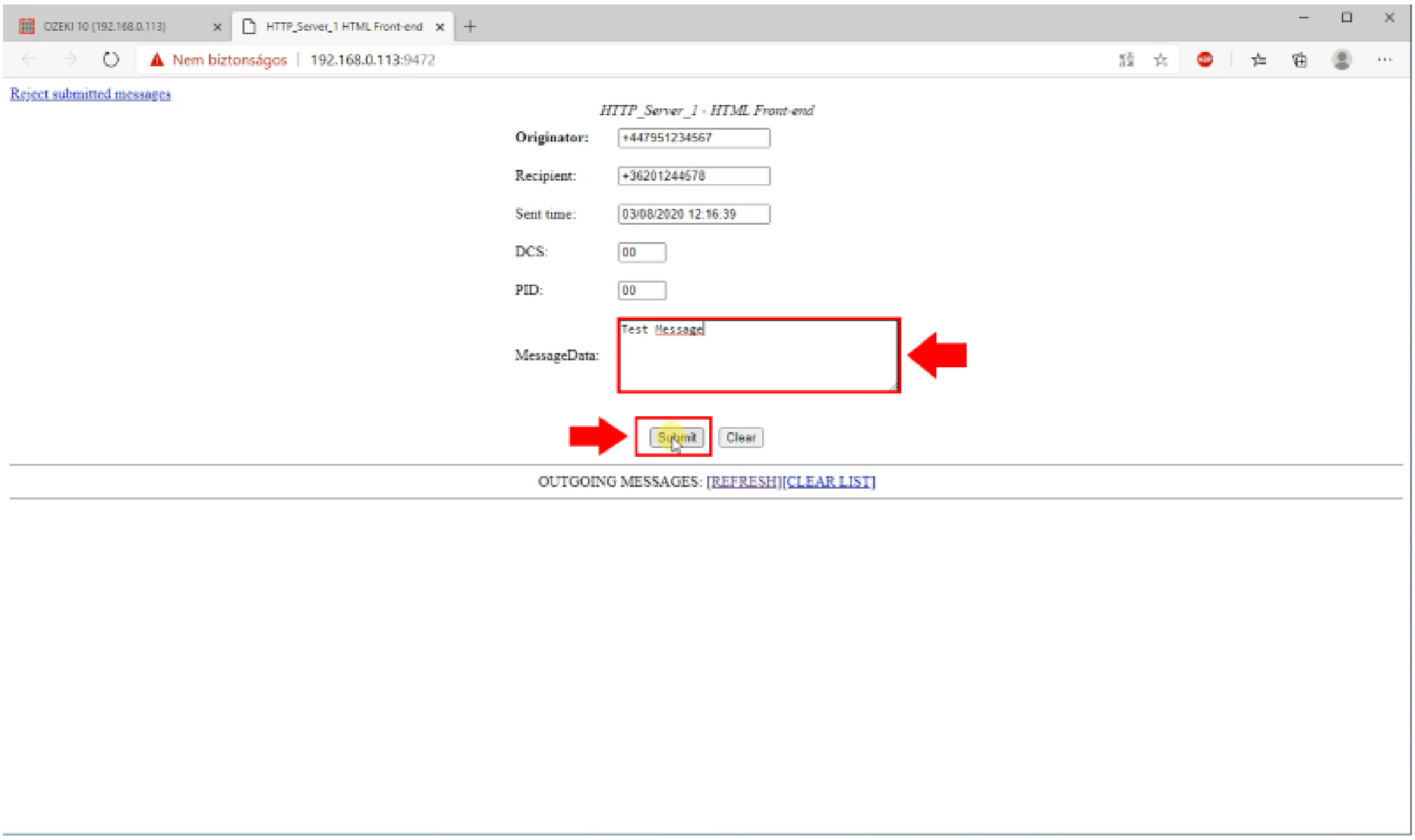
Mode: Move

Ok | Cancel

Figure 10 - Modify the second route

# Step 6 - Simulate incoming SMS

At this point, every connection and route set up for the incoming messages. Now, you just need to receive some. To test the solution, you can use the HTTP Server connection to initiate some message and see how it will be routed between the Standard users. For that, open the menu of the HTTP Server connection, and here, click on its URL on the left side. By doing this action, the HTML form of the connection shows up (Figure 11). All you need to do here to send a test message is to set up a recipient phone number and write the text of the message. If you finished, just click on 'Submit'.


OZEKI 10 (192.168.0.113)
HTTP_Server_1 HTML Front-end
Nem biztonságos | 192.168.0.113:9472
Reject submitted messages

*HTTP_Server_1 - HTML Front-end*

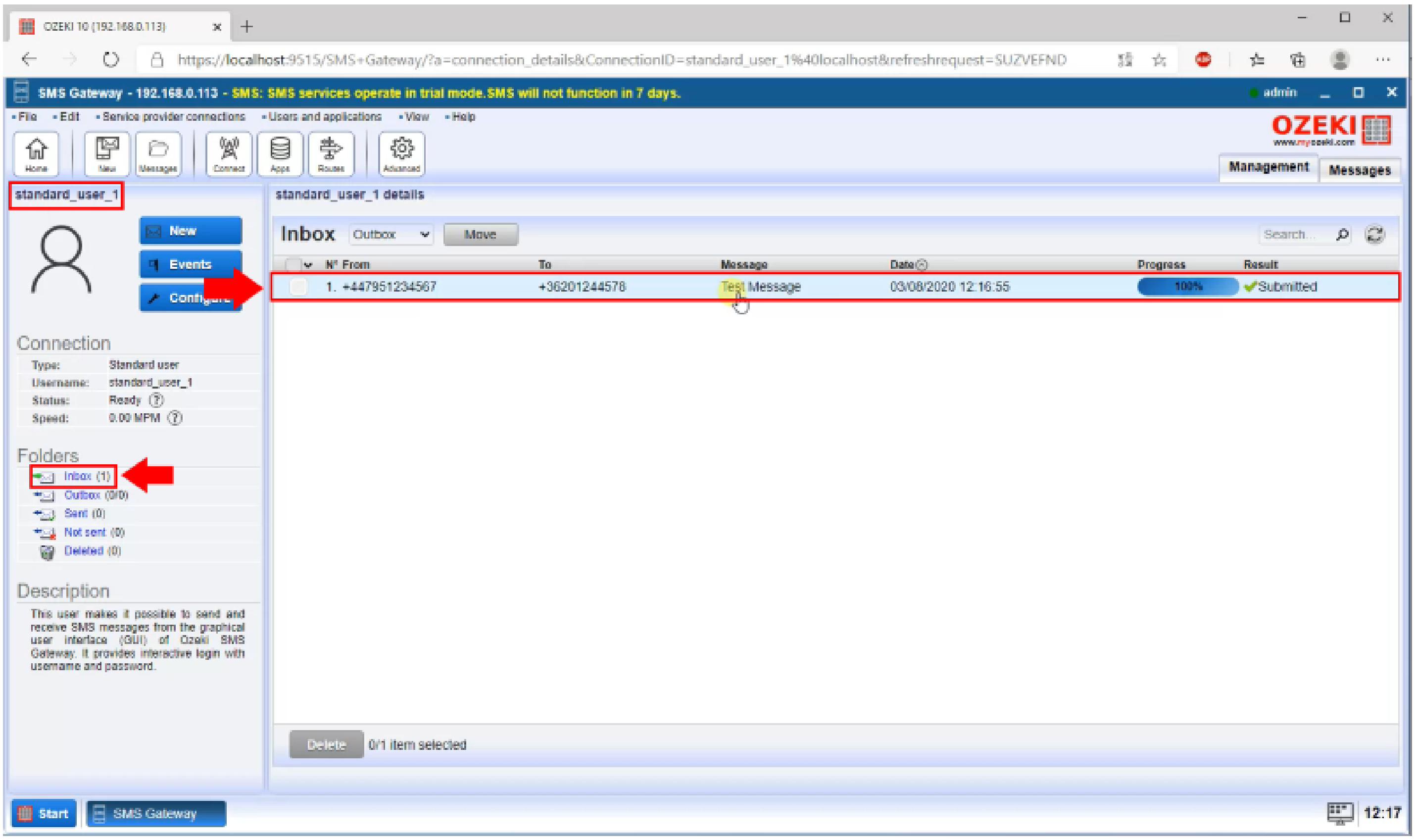
| | |
|---|---|
| **Originator:** | +447951234567 |
| Recipient: | +36201244678 |
| Sent time: | 03/08/2020 12:16:39 |
| DCS: | 00 |
| PID: | 00 |
| MessageData: | Test Message |

Submit
Clear
OUTGOING MESSAGES: [REFRESH][CLEAR LIST]

Figure 11 - Simulate an incoming sms

# Step 7 - Check the inbox folder of the first Standard user

Now, it's time to check the inbox folder of the Standard users to make sure the route works properly and copied the messages from the HTTP Server connection. So, to do this, just select the first Standard user from the main menu and click on the Inbox folder. Here, you will be able to see like in Figure 12, that the message sent by the HTTP server connection landed in the inbox folder of the Standard user connection as well. This means the route copied the message successfully.



SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.

https://localhost:9515/SMS+Gateway/?a=connection_details&ConnectionID=standard_user_1%40localhost&refreshrequest=SUZVEFND

standard_user_1 details

Inbox

| N° | From | To | Message | Date | Progress | Result |
|---|---|---|---|---|---|---|
| 1. | +447951234567 | +36201244578 | Test Message | 03/08/2020 12:16:55 | 100% | ✓Submitted |

Connection

| | |
|---|---|
| Type: | Standard user |
| Username: | standard_user_1 |
| Status: | Ready |
| Speed: | 0.00 MPM |

Folders

- Inbox (1)
- Outbox (0/0)
- Sent (0)
- Not sent (0)
- Deleted (0)

Description

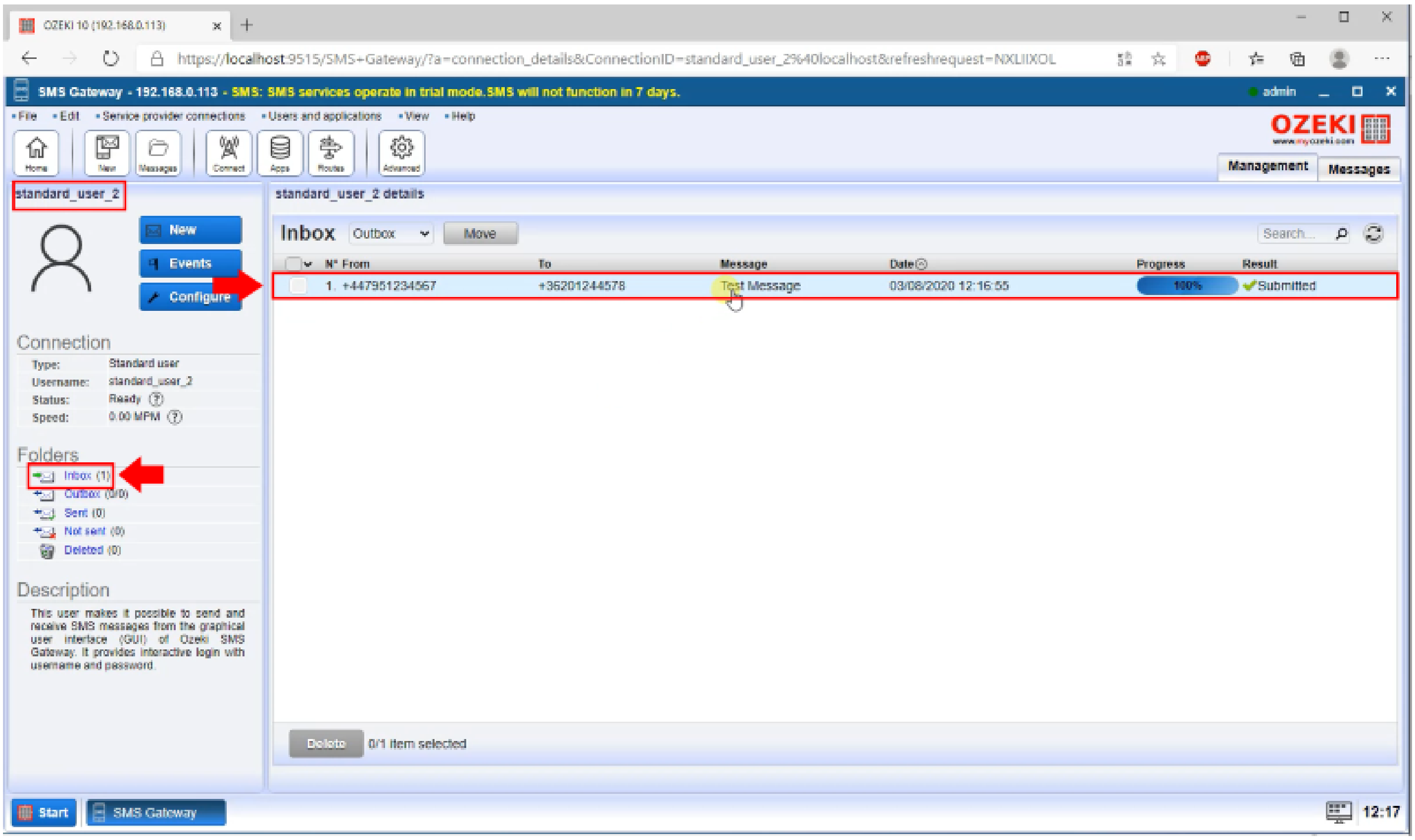
This user makes it possible to send and receive SMS messages from the graphical user interface (GUI) of Ozeki SMS Gateway. It provides interactive login with username and password.

0/1 item selected

Figure 12 - SMS in the first user inbox

# Step 8 - Check the inbox folder of the second Standard user

The last step of the guide is to check the inbox folder of the second Standard user connection as well. For that, just select the connection from the main menu, and as Figure 13 demonstrates that, click on the Inbox folder. Here, you should see the same message that was in the inbox folder of the first Standard user connection. If it is the case, that means the second route did its job successfully to move the message to the second Standard user.



Connection

| | |
|---|---|
| Type: | Standard user |
| Username: | standard_user_2 |
| Status: | Ready |
| Speed: | 0.00 MPM |

| N° | From | To | Message | Date | Progress | Result |
|---|---|---|---|---|---|---|
| 1. | +447951234567 | +36201244578 | Test Message | 03/08/2020 12:16:55 | 100% | Submitted |

Description

This user makes it possible to send and receive SMS messages from the graphical user interface (GUI) of Ozeki SMS Gateway. It provides interactive login with username and password.

Figure 13 - SMS in the second user inbox

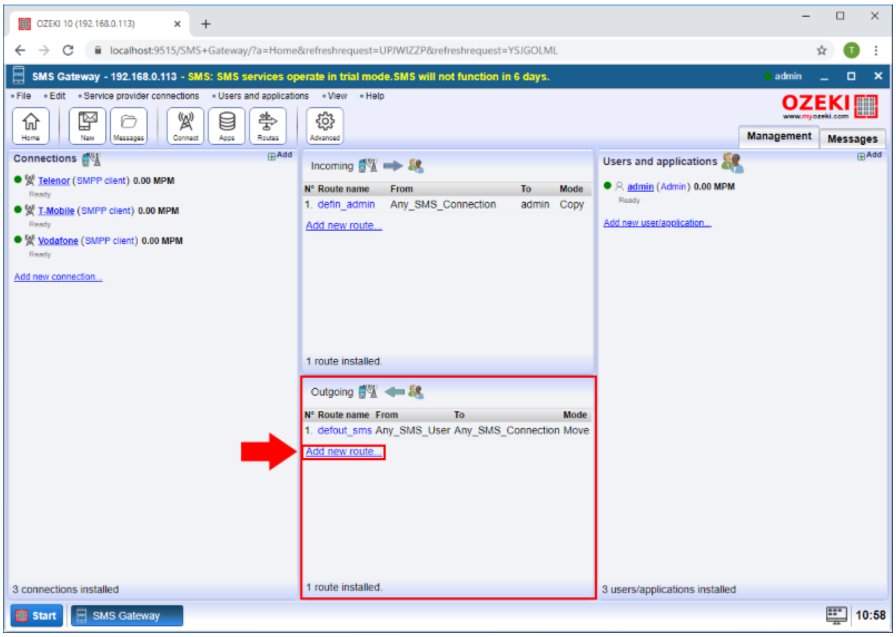
# Least cost SMS routing

This is a configuration example on how to setup least cost routing in Ozeki SMS Gateway. Least cost routing means, that SMS messages are routed to the appropriate SMS service provider connection based on the telephone number prefix.

In our example, there are three SMS Service provider connections: Telenor, T-Mobile, Vodafone. The routing is setup the following way:

- If an SMS is sent to a phone number that starts with +3620 it should be routed to Telenor
- If an SMS is sent to a phone number that starts with +3630 it should be routed to T-Mobile
- If an SMS is sent to a phone number that starts with +3670 it should be routed to Vodafone
- All other SMS messages should be routed to one of these providers randomly.

To setup this routing the outbound routing table should be modified. The outbound routing table, that determines which service provider connection will be used to send an SMS message, can be found in the bottom part of the center panel of the management form of Ozeki SMS Gateway (Figure 1).



SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.SMS will not function in 6 days.

Connections

- Telenor (SMPP client) 0.00 MPM — Ready
- T-Mobile (SMPP client) 0.00 MPM — Ready
- Vodafone (SMPP client) 0.00 MPM — Ready

Add new connection...

3 connections installed

Incoming

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defin_admin | Any_SMS_Connection | admin | Copy |

Add new route...

1 route installed.

Outgoing

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move |

Add new route...

1 route installed.

Users and applications

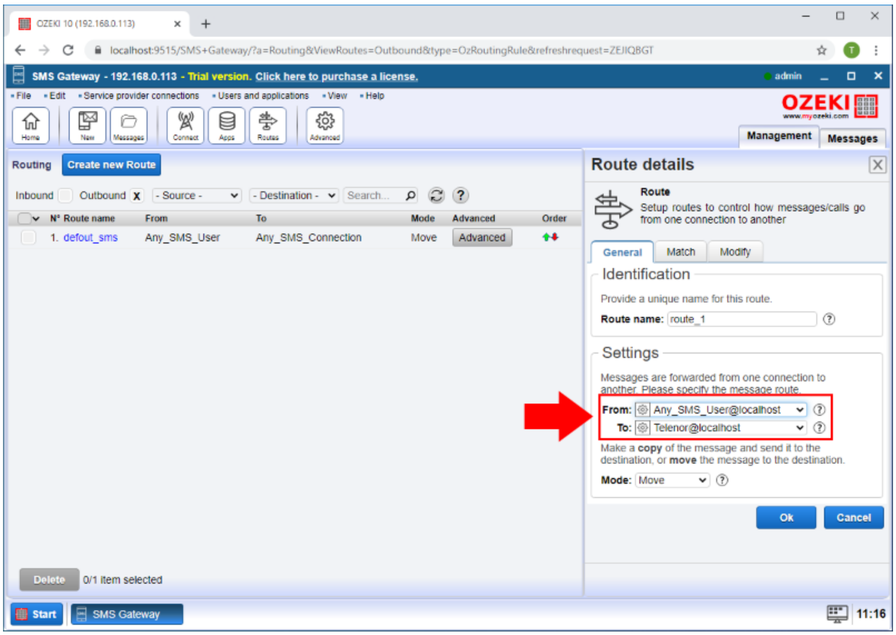
- admin (Admin) 0.00 MPM — Ready

Add new user/application...

3 users/applications installed

Figure 1 - Add new routing rule

In the Route details menu select the From and To connections which are Any SMS User and Telenor in this case (Figure 2).



SMS Gateway - 192.168.0.113 - Trial version. Click here to purchase a license.

| | N° | Route name | From | To | Mode | Advanced | Order |
|---|---|---|---|---|---|---|---|
| □ | 1. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move | Advanced | |

0/1 item selected

Route details

Route

Setup routes to control how messages/calls go from one connection to another

Identification

Provide a unique name for this route.

Route name: route_1

Settings

Messages are forwarded from one connection to another. Please specify the message route.

From: Any_SMS_User@localhost

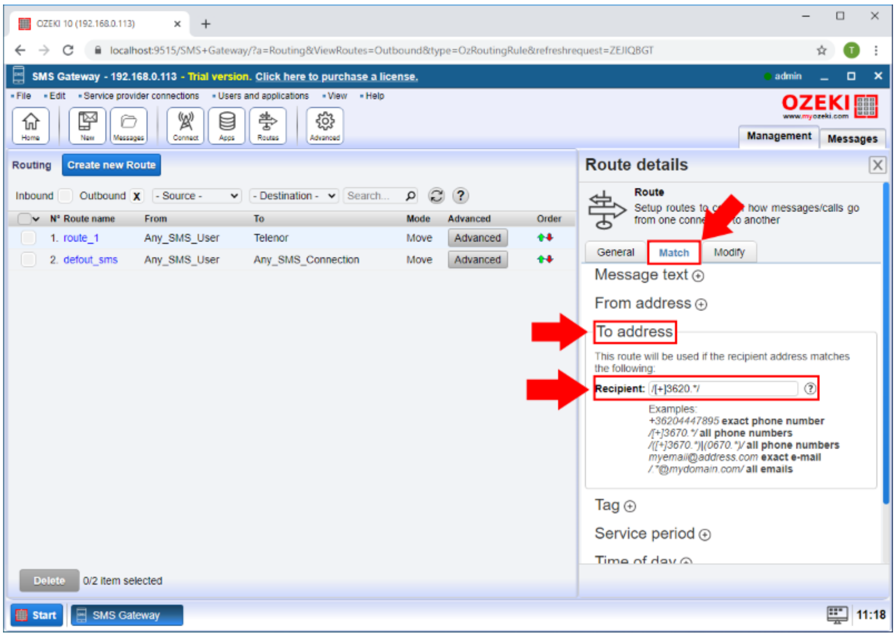
To: Telenor@localhost

Make a **copy** of the message and send it to the destination, or **move** the message to the destination.

Mode: Move

Figure 2 - Specify from and to connection

On the Match tab, in the To address field, you can enter the phone number prefix you want to use in the given routing rule (Figure 3).



SMS Gateway - 192.168.0.113 - Trial version. Click here to purchase a license.

Routing | Create new Route

| Nº | Route name | From | To | Mode | Advanced |
|---|---|---|---|---|---|
| 1. | route_1 | Any_SMS_User | Telenor | Move | Advanced |
| 2. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move | Advanced |

0/2 item selected

Route details

Route
Setup routes to control how messages/calls go from one connection to another

General | Match | Modify

Message text ⊕

From address ⊕

To address

This route will be used if the recipient address matches the following:

Recipient: /[+]3620.*/

Examples:
*+36204447895* **exact phone number**
*/[+]3670.*/* **all phone numbers**
*/([+]3670.*)|(0670.*)/* **all phone numbers**
*myemail@address.com* **exact e-mail**
*/.*@mydomain.com/* **all emails**

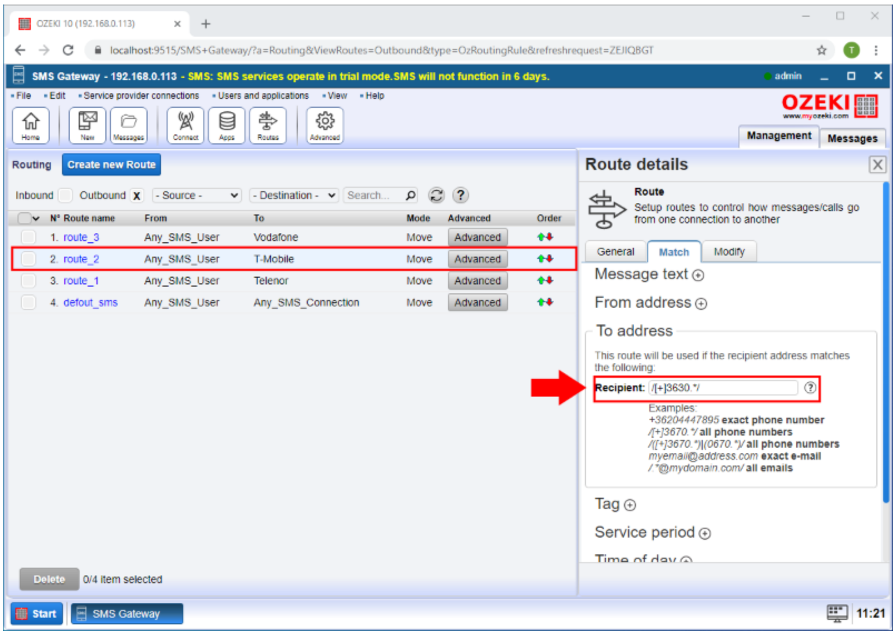
Tag ⊕

Service period ⊕

Time of day ⊕

Figure 3 - Specify to address prefix

Repeat above steps with the another two service provider connections as you can see on the Figure 4 and Figure 5.



SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.SMS will not function in 6 days.

Routing | Create new Route

| Nº | Route name | From | To | Mode | Advanced |
|---|---|---|---|---|---|
| 1. | route_3 | Any_SMS_User | Vodafone | Move | Advanced |
| 2. | route_2 | Any_SMS_User | T-Mobile | Move | Advanced |
| 3. | route_1 | Any_SMS_User | Telenor | Move | Advanced |
| 4. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move | Advanced |

0/4 item selected

Route details

Route
Setup routes to control how messages/calls go from one connection to another

General | Match | Modify

Message text ⊕

From address ⊕

To address

This route will be used if the recipient address matches the following:

Recipient: /[+]3630.*/

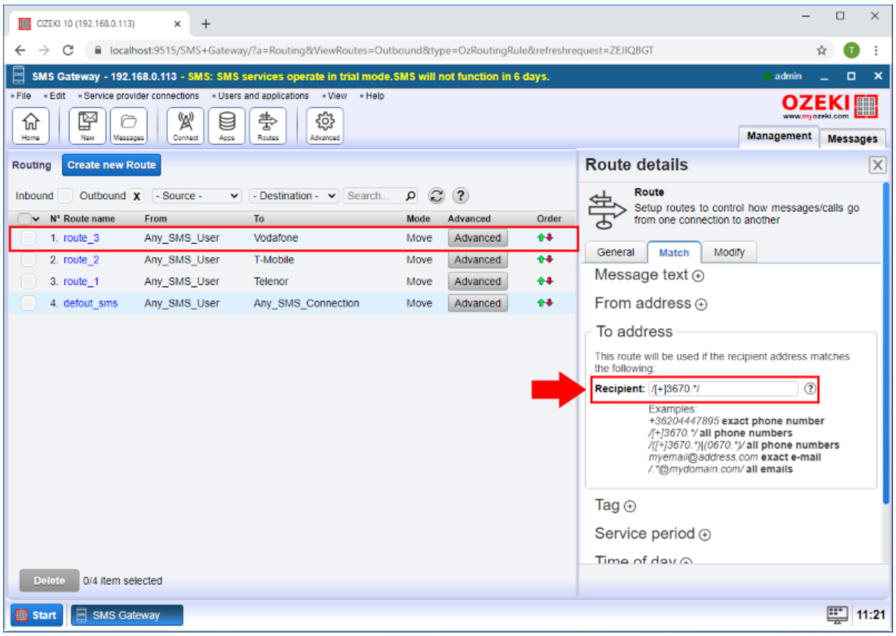
Examples:
*+36204447895* **exact phone number**
*/[+]3670.*/* **all phone numbers**
*/([+]3670.*)|(0670.*)/* **all phone numbers**
*myemail@address.com* **exact e-mail**
*/.*@mydomain.com/* **all emails**

Tag ⊕

Service period ⊕

Time of day ⊕

Figure 4 - T-Mobile routing rule



localhost:9515/SMS+Gateway/?a=Routing&ViewRoutes=Outbound&type=OzRoutingRule&refreshrequest=ZEJIQBGT

SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.SMS will not function in 6 days.

| N° | Route name | From | To | Mode | Advanced |
|---|---|---|---|---|---|
| 1. | route_3 | Any_SMS_User | Vodafone | Move | Advanced |
| 2. | route_2 | Any_SMS_User | T-Mobile | Move | Advanced |
| 3. | route_1 | Any_SMS_User | Telenor | Move | Advanced |
| 4. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move | Advanced |

0/4 item selected

Route details

Route

Setup routes to control how messages/calls go from one connection to another

To address

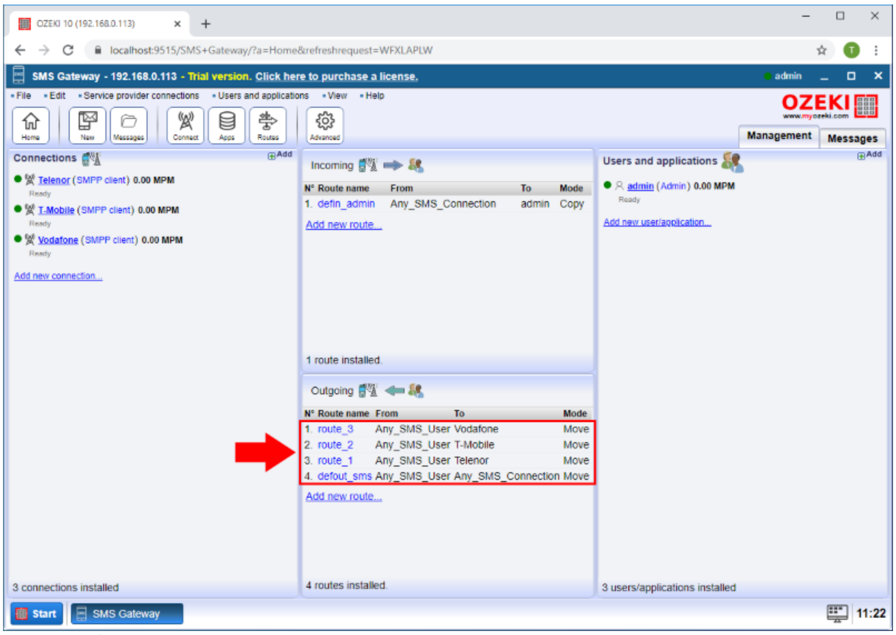
This route will be used if the recipient address matches the following:

Recipient: /[+]3670.*/

Examples:
+36204447895 exact phone number
/[+]3670.*/ all phone numbers
/([+]3670.*)|(0670.*)/ all phone numbers
myemail@address.com exact e-mail
/.*@mydomain.com/ all emails

Figure 5 - Vodafone routing rule

Finally you can see the routing rules in the Outbound section the bottom part of the center panel of the management form (Figure 6).



localhost:9515/SMS+Gateway/?a=Home&refreshrequest=WFXLAPLW

SMS Gateway - 192.168.0.113 - Trial version. Click here to purchase a license.

Connections

- Telenor (SMPP client) 0.00 MPM Ready
- T-Mobile (SMPP client) 0.00 MPM Ready
- Vodafone (SMPP client) 0.00 MPM Ready

Add new connection...

3 connections installed

Incoming

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defin_admin | Any_SMS_Connection | admin | Copy |

Add new route...

1 route installed.

Outgoing

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | route_3 | Any_SMS_User | Vodafone | Move |
| 2. | route_2 | Any_SMS_User | T-Mobile | Move |
| 3. | route_1 | Any_SMS_User | Telenor | Move |
| 4. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move |

Add new route...

4 routes installed.

Users and applications

- admin (Admin) 0.00 MPM Ready

Add new user/application...

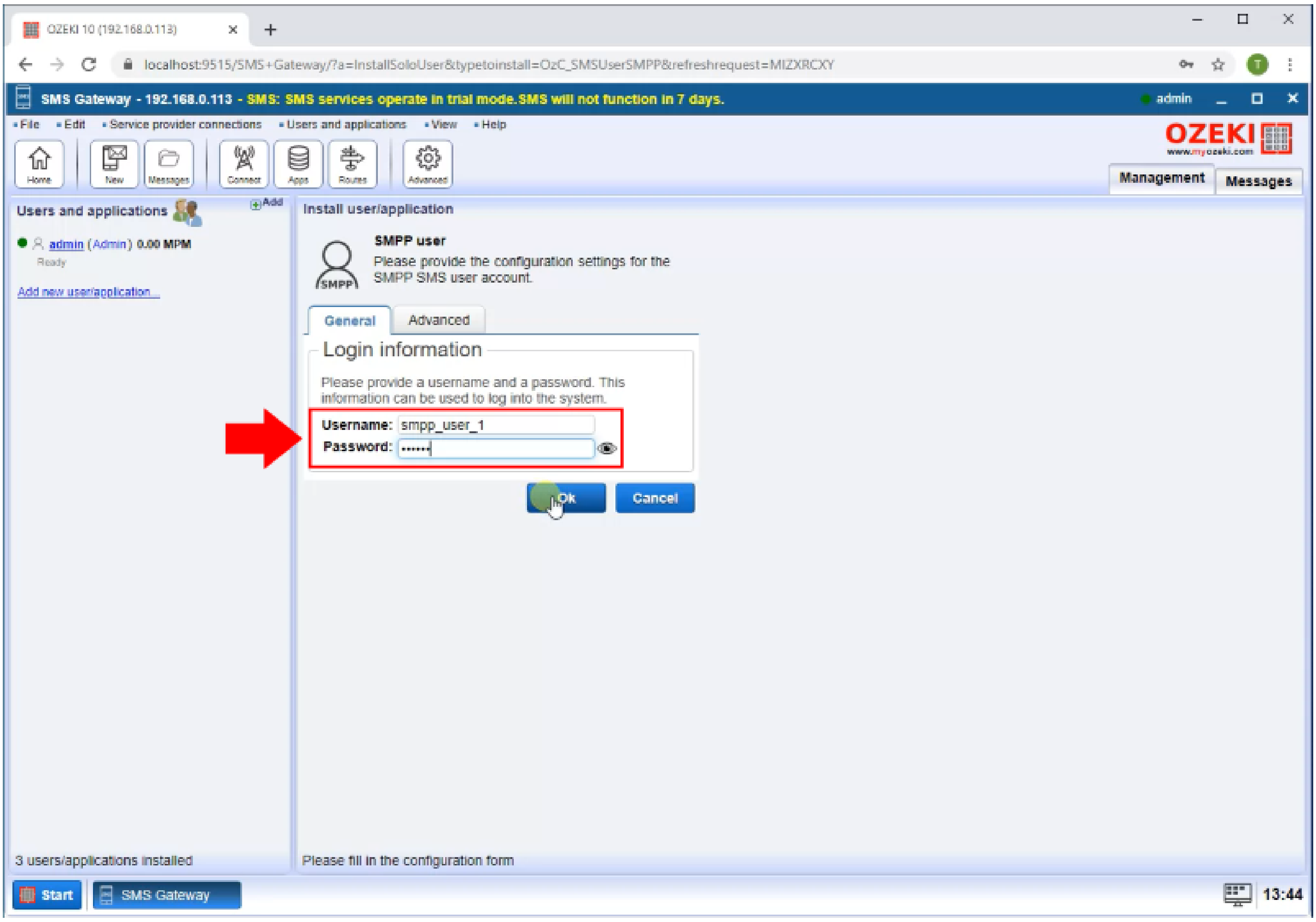
3 users/applications installed

Figure 6 - Add new routing rule

# How to block a sender number

***This section is about to show you how easy you can set sender phone number in a routing rule to drop the message. This modification process makes sure that every message received from a specific phone number will drop by the SMS Gateway. The document shows you all the configuration work that you have to perform in the routing rule. It also demonstrates how you can test your system by sending a test message and see if the routing rule works fine.***

Before you configured the outgoing routing rule, you just need to set up a SMPP user. For that purpose, click the Add new user/application... link on the right of the Management Console. An interface will open, search the SMPP User and click the blue 'install' button next to it. Clicking the Install link will bring up the SMPP user installation panel. Here, you need to enter a unique username in the Username field and a password in the Password filed (Figure 1).



localhost:9515/SMS+Gateway/?a=InstallSoloUser&typetoinstall=OzC_SMSUserSMPP&refreshrequest=MIZXRCXY

SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.

Install user/application

SMPP user

Please provide the configuration settings for the SMPP SMS user account.

Login information

Please provide a username and a password. This information can be used to log into the system.

Username: smpp_user_1

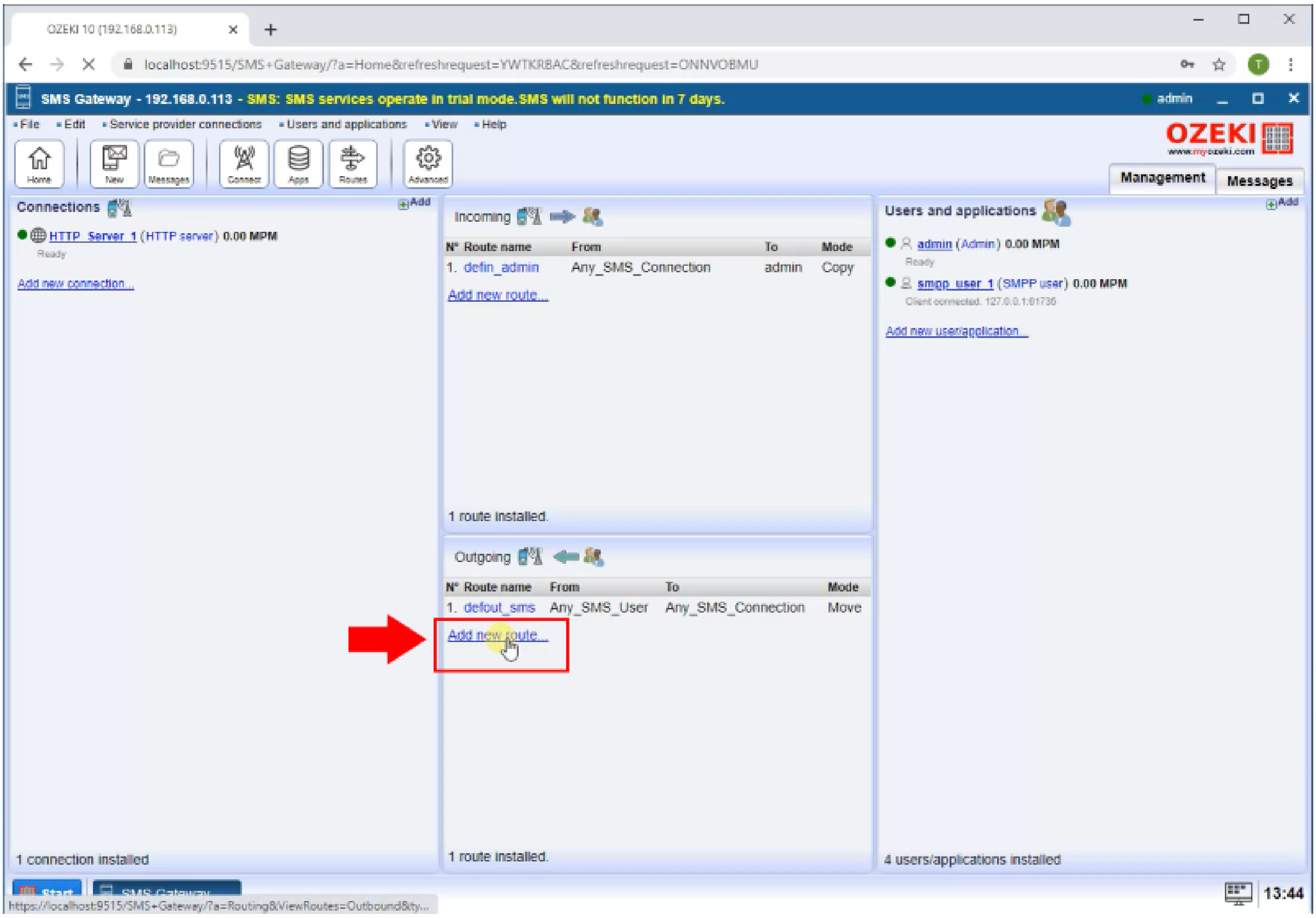
Password: ••••••

Please fill in the configuration form

3 users/applications installed

Figure 1 - Add SMPP user

You can add an outbound routing rule by clicking add new route in the Outbound section. Then in the Settings, for the From connection, select the User connection from which you are sending the SMS, and for the To connection, select the Service provider connection through which you want to send the SMS to the service provider. In the Mode section select the drop option to make sure that the message will rejected if it is sent from the phone number you will define.


OZEKI 10 (192.168.0.113)
localhost:9515/SMS+Gateway/?a=Home&refreshrequest=YWTKRBAC&refreshrequest=ONNVOBMU
SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
OZEKI
Home
New
Messages
Connect
Apps
Routes
Advanced
Management
Messages
Connections
Add
HTTP_Server_1 (HTTP server) 0.00 MPM
Ready
Add new connection...
1 connection installed
Incoming
Add new route...
1 route installed.
Outgoing
Add new route...
1 route installed.
Users and applications
Add
admin (Admin) 0.00 MPM
Ready
smpp_user_1 (SMPP user) 0.00 MPM
Client connected. 127.0.0.1:61736
Add new user/application...
4 users/applications installed
Start
SMS Gateway
13:44
https://localhost:9515/SMS+Gateway/?a=Routing&ViewRoutes=Outbound&ty...

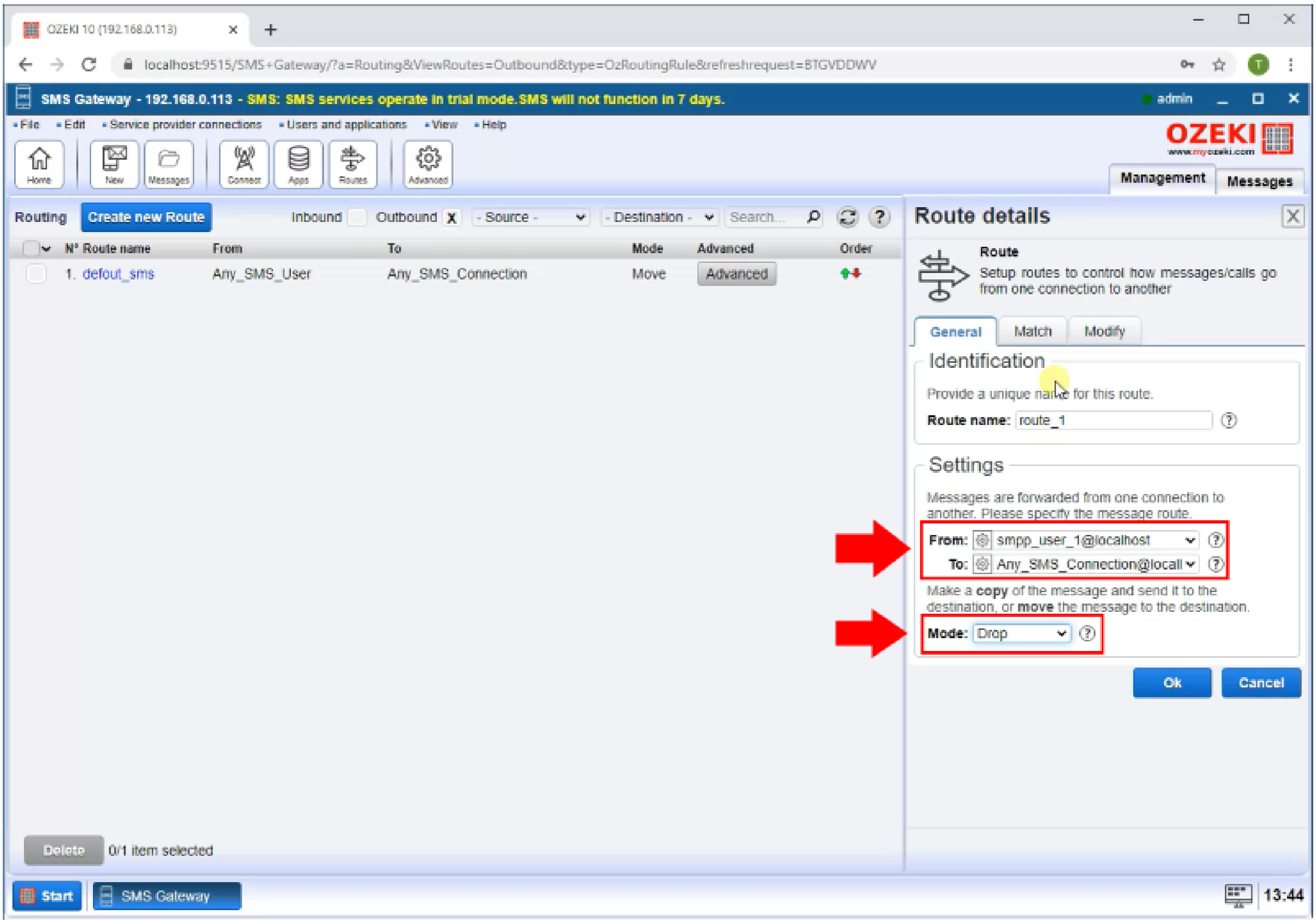
Incoming

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defin_admin | Any_SMS_Connection | admin | Copy |

Outgoing

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move |

Figure 2 - Add new route



| N° | Route name | From | To | Mode | Advanced | Order |
|---|---|---|---|---|---|---|
| 1. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move | Advanced | |

Route details

Route

Setup routes to control how messages/calls go from one connection to another

General | Match | Modify

Identification

Provide a unique name for this route.

Route name: route_1

Settings

Messages are forwarded from one connection to another. Please specify the message route.

From: smpp_user_1@localhost

To: Any_SMS_Connection@locall

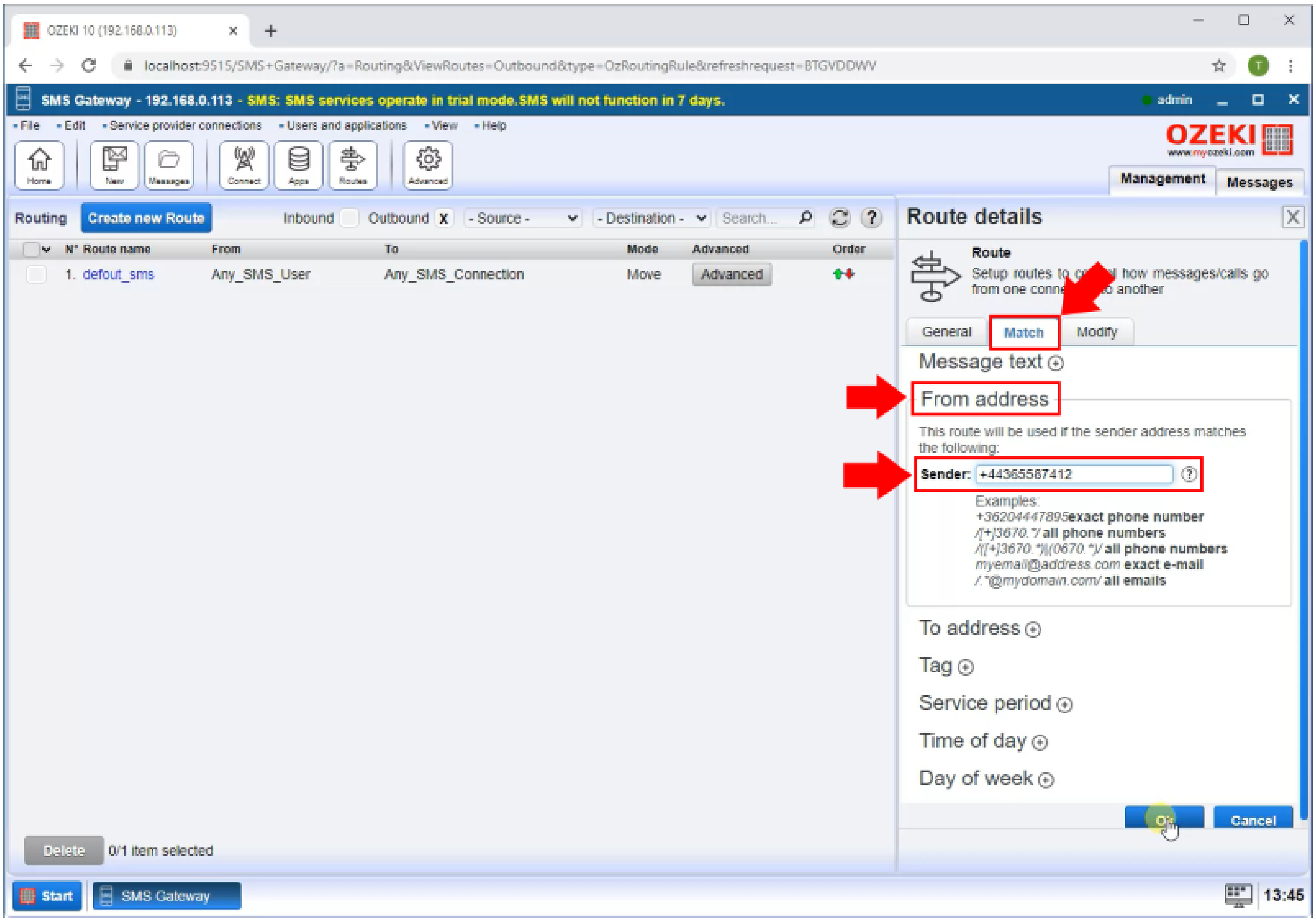
Make a **copy** of the message and send it to the destination, or **move** the message to the destination.

Mode: Drop

Ok Cancel

Figure 3 - Set message from and to connection and route mode

In the Match menu, you can perform all matching operation for the outgoing messages. At this point, you need to select the 'From address' submenu like in Figure 4, and here, you can just type the phone number that you want to reject every outgoing message. Finally, just click on OK to save the modifications.



localhost:9515/SMS+Gateway/?a=Routing&ViewRoutes=Outbound&type=OzRoutingRule&refreshrequest=BTGVDDWV

SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.

| N° | Route name | From | To | Mode | Advanced | Order |
|---|---|---|---|---|---|---|
| 1. | defaut_sms | Any_SMS_User | Any_SMS_Connection | Move | Advanced | |

Route details

Route

Setup routes to control how messages/calls go from one connection to another

From address

This route will be used if the sender address matches the following:

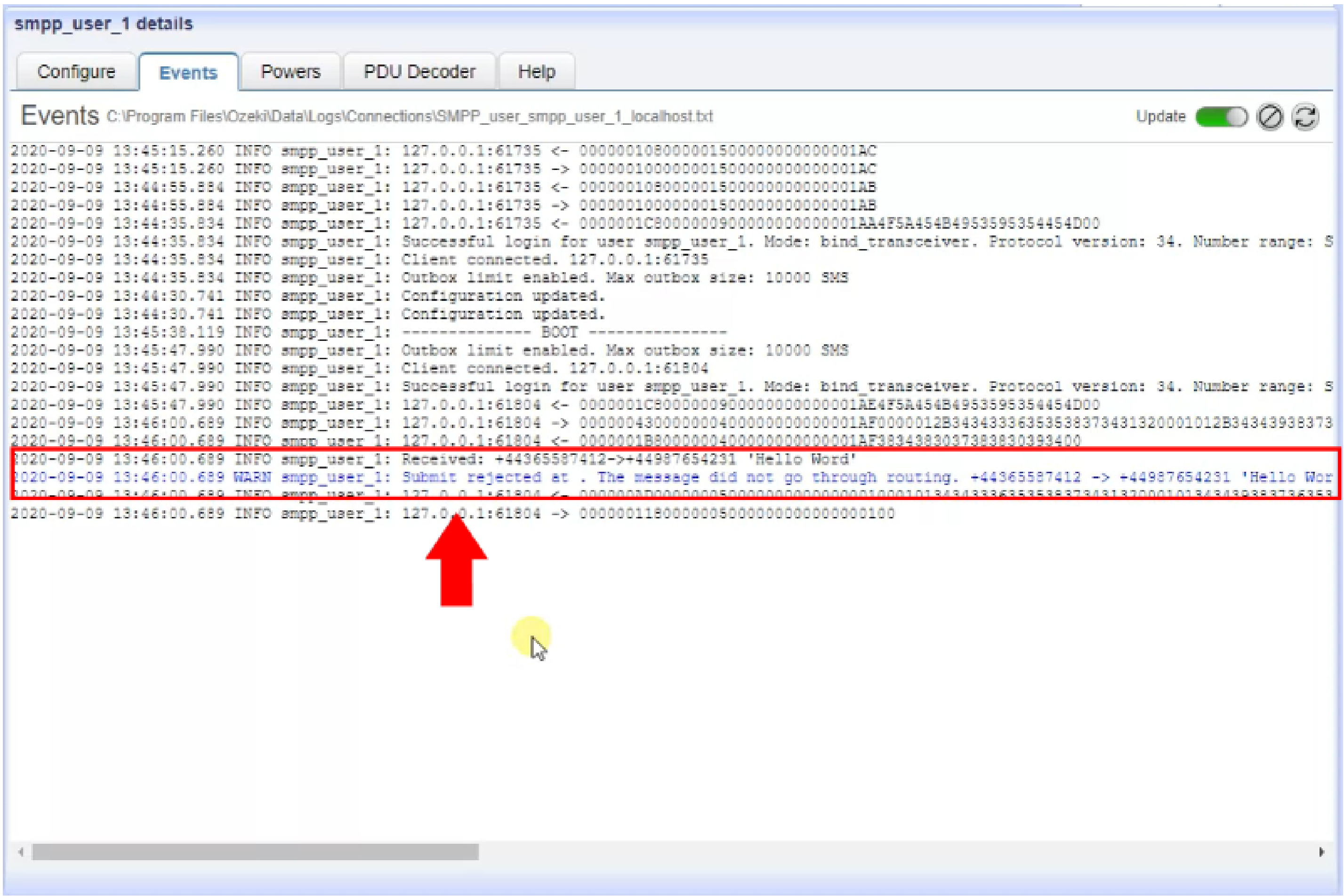
Sender: +44365587412

Examples:
+36204447895exact phone number
/[+]3670.*/ all phone numbers
/([+]3670.*)|(0670.*)/ all phone numbers
myemail@address.com exact e-mail
/.*@mydomain.com/ all emails

0/1 item selected

Figure 4 - Match sender address

To check if the modifications worked, you need to open the SMPP client and select the Events tab. Here, you can see every event of the SMPP client connection. As Figure 5 shows that, the routing rule reject the from address, so the SMPP client drop the message.

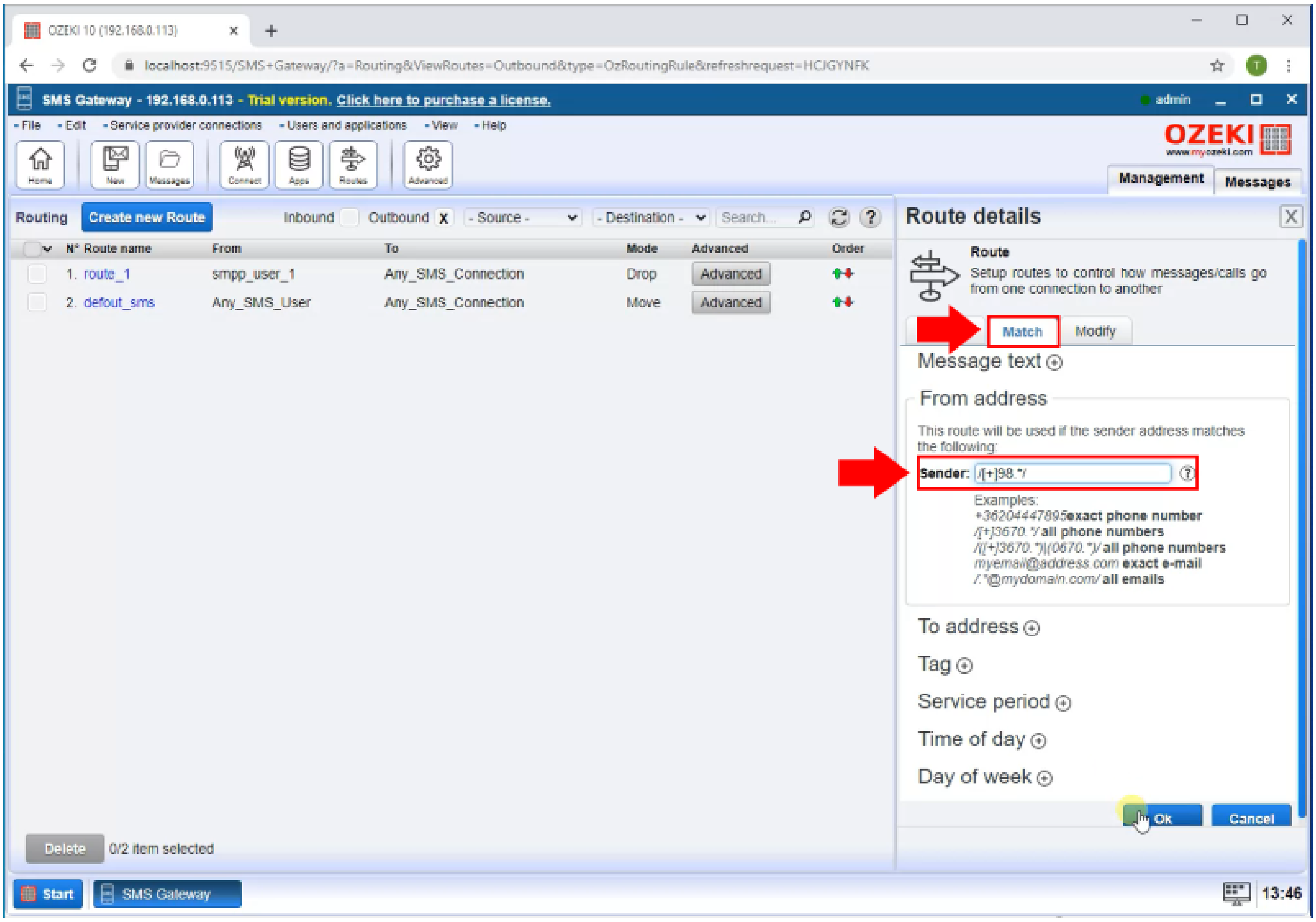


Events C:\Program Files\Ozeki\Data\Logs\Connections\SMPP_user_smpp_user_1_localhost.txt

```
2020-09-09 13:45:15.260 INFO smpp_user_1: 127.0.0.1:61735 <- 000000108000001500000000000001AC
2020-09-09 13:45:15.260 INFO smpp_user_1: 127.0.0.1:61735 -> 000000100000001500000000000001AC
2020-09-09 13:44:55.884 INFO smpp_user_1: 127.0.0.1:61735 <- 000000108000001500000000000001AB
2020-09-09 13:44:55.884 INFO smpp_user_1: 127.0.0.1:61735 -> 000000100000001500000000000001AB
2020-09-09 13:44:35.834 INFO smpp_user_1: 127.0.0.1:61735 <- 0000001C800000090000000000001AA4F5A454B4953595354454D00
2020-09-09 13:44:35.834 INFO smpp_user_1: Successful login for user smpp_user_1. Mode: bind_transceiver. Protocol version: 34. Number range: S
2020-09-09 13:44:35.834 INFO smpp_user_1: Client connected. 127.0.0.1:61735
2020-09-09 13:44:35.834 INFO smpp_user_1: Outbox limit enabled. Max outbox size: 10000 SMS
2020-09-09 13:44:30.741 INFO smpp_user_1: Configuration updated.
2020-09-09 13:44:30.741 INFO smpp_user_1: Configuration updated.
2020-09-09 13:45:38.119 INFO smpp_user_1: -------------- BOOT ---------------
2020-09-09 13:45:47.990 INFO smpp_user_1: Outbox limit enabled. Max outbox size: 10000 SMS
2020-09-09 13:45:47.990 INFO smpp_user_1: Client connected. 127.0.0.1:61804
2020-09-09 13:45:47.990 INFO smpp_user_1: Successful login for user smpp_user_1. Mode: bind_transceiver. Protocol version: 34. Number range: S
2020-09-09 13:45:47.990 INFO smpp_user_1: 127.0.0.1:61804 <- 0000001C800000090000000000001AE4F5A454B4953595354454D00
2020-09-09 13:46:00.689 INFO smpp_user_1: 127.0.0.1:61804 -> 00000043000000040000000000001AF0000012B34343336353538373431320001012B34343938373
2020-09-09 13:46:00.689 INFO smpp_user_1: 127.0.0.1:61804 <- 0000001B800000040000000000001AF3834383037383830393400
2020-09-09 13:46:00.689 INFO smpp_user_1: Received: +44365587412->+44987654231 'Hello Word'
2020-09-09 13:46:00.689 WARN smpp_user_1: Submit rejected at . The message did not go through routing. +44365587412 -> +44987654231 'Hello Wor
2020-09-09 13:46:00.689 INFO smpp_user_1: 127.0.0.1:61804 <- 0000003D0000000500000000000000010001013434333635353837343132000101343439383737363
2020-09-09 13:46:00.689 INFO smpp_user_1: 127.0.0.1:61804 -> 0000001180000005000000000000000100
```

Figure 5 - Received message rejected

In the SMS Gateway, during routing, you have the option to Match phone numbert using regular expressions. In this case you can reject for example a phone number prefix too. On the Match tab, in the To address field, you can enter the phone number prefix you want to use in the given routing rule.



| N° | Route name | From | To | Mode | Advanced |
|---|---|---|---|---|---|
| 1. | route_1 | smpp_user_1 | Any_SMS_Connection | Drop | Advanced |
| 2. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move | Advanced |

0/2 item selected

From address

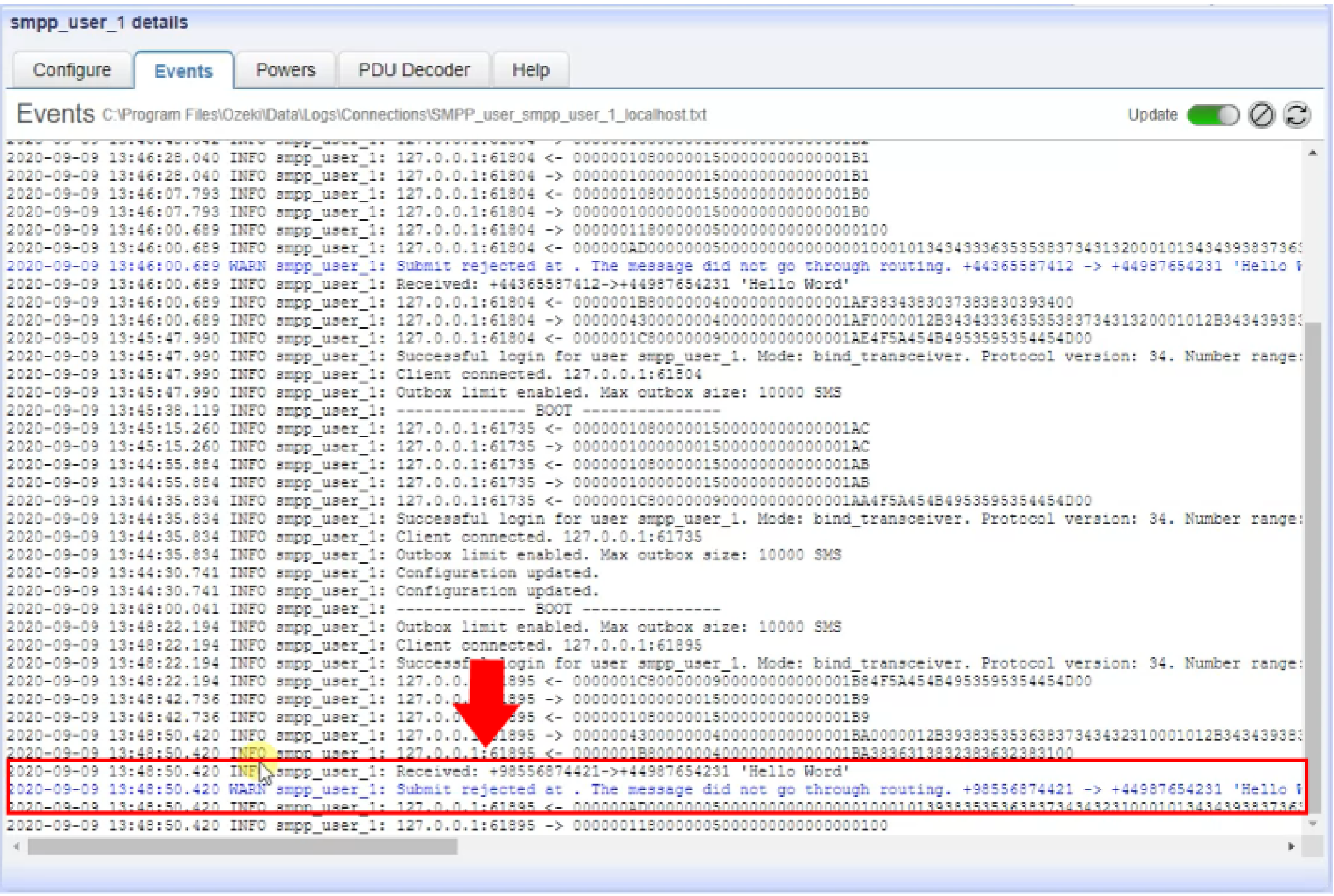
This route will be used if the sender address matches the following:

Sender: /[+]98.*/

Examples:
+36204447895exact phone number
/[+]3670.*/ all phone numbers
/([+]3670.*)|(0670.*)/ all phone numbers
myemail@address.com exact e-mail
/.*@mydomain.com/ all emails

Figure 6 - Set sender address regexp

To check if the modifications worked, you need to open the SMPP client and select the Events tab. Here, you can see every event of the SMPP client connection. As Figure 7 shows that, the routing rule reject the from address, so the SMPP client drop the message.



C:\Program Files\Ozeki\Data\Logs\Connections\SMPP_user_smpp_user_1_localhost.txt

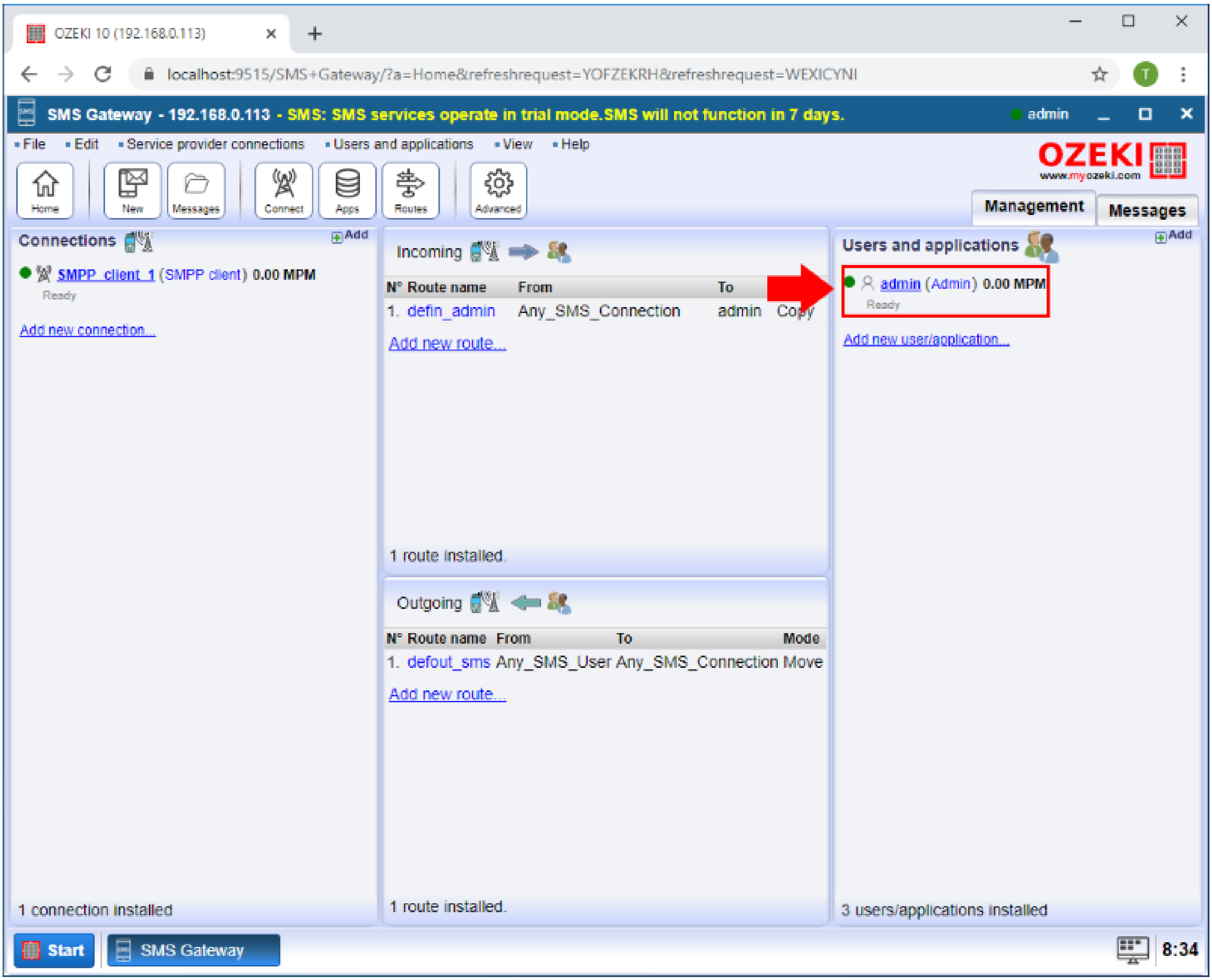
```
2020-09-09 13:46:28.040 INFO smpp_user_1: 127.0.0.1:61804 <- 000000108000001500000000000001B1
2020-09-09 13:46:28.040 INFO smpp_user_1: 127.0.0.1:61804 -> 000000100000001500000000000001B1
2020-09-09 13:46:07.793 INFO smpp_user_1: 127.0.0.1:61804 <- 000000108000001500000000000001B0
2020-09-09 13:46:07.793 INFO smpp_user_1: 127.0.0.1:61804 -> 000000100000001500000000000001B0
2020-09-09 13:46:00.689 INFO smpp_user_1: 127.0.0.1:61804 -> 00000011800000050000000000000000100
2020-09-09 13:46:00.689 INFO smpp_user_1: 127.0.0.1:61804 <- 000000AD00000005000000000000000010001013434333635353837343132000101343439383736
2020-09-09 13:46:00.689 WARN smpp_user_1: Submit rejected at . The message did not go through routing. +44365587412 -> +44987654231 'Hello W
2020-09-09 13:46:00.689 INFO smpp_user_1: Received: +44365587412->+44987654231 'Hello Word'
2020-09-09 13:46:00.689 INFO smpp_user_1: 127.0.0.1:61804 <- 0000001B8000000400000000000001AF3834383037383830393400
2020-09-09 13:46:00.689 INFO smpp_user_1: 127.0.0.1:61804 -> 000000430000000400000000000001AF0000012B343433363535383734313200001012B34343938
2020-09-09 13:45:47.990 INFO smpp_user_1: 127.0.0.1:61804 <- 0000001C8000000900000000000001AE4F5A454B4953595354454D00
2020-09-09 13:45:47.990 INFO smpp_user_1: Successful login for user smpp_user_1. Mode: bind_transceiver. Protocol version: 34. Number range:
2020-09-09 13:45:47.990 INFO smpp_user_1: Client connected. 127.0.0.1:61804
2020-09-09 13:45:47.990 INFO smpp_user_1: Outbox limit enabled. Max outbox size: 10000 SMS
2020-09-09 13:45:38.119 INFO smpp_user_1: -------------- BOOT ---------------
2020-09-09 13:45:15.260 INFO smpp_user_1: 127.0.0.1:61735 <- 000000108000001500000000000001AC
2020-09-09 13:45:15.260 INFO smpp_user_1: 127.0.0.1:61735 -> 000000100000001500000000000001AC
2020-09-09 13:44:55.884 INFO smpp_user_1: 127.0.0.1:61735 <- 000000108000001500000000000001AB
2020-09-09 13:44:55.884 INFO smpp_user_1: 127.0.0.1:61735 -> 000000100000001500000000000001AB
2020-09-09 13:44:35.834 INFO smpp_user_1: 127.0.0.1:61735 <- 0000001C8000000900000000000001AA4F5A454B4953595354454D00
2020-09-09 13:44:35.834 INFO smpp_user_1: Successful login for user smpp_user_1. Mode: bind_transceiver. Protocol version: 34. Number range:
2020-09-09 13:44:35.834 INFO smpp_user_1: Client connected. 127.0.0.1:61735
2020-09-09 13:44:35.834 INFO smpp_user_1: Outbox limit enabled. Max outbox size: 10000 SMS
2020-09-09 13:44:30.741 INFO smpp_user_1: Configuration updated.
2020-09-09 13:44:30.741 INFO smpp_user_1: Configuration updated.
2020-09-09 13:48:00.041 INFO smpp_user_1: -------------- BOOT ---------------
2020-09-09 13:48:22.194 INFO smpp_user_1: Outbox limit enabled. Max outbox size: 10000 SMS
2020-09-09 13:48:22.194 INFO smpp_user_1: Client connected. 127.0.0.1:61895
2020-09-09 13:48:22.194 INFO smpp_user_1: Successful login for user smpp_user_1. Mode: bind_transceiver. Protocol version: 34. Number range:
2020-09-09 13:48:22.194 INFO smpp_user_1: 127.0.0.1:61895 <- 0000001C8000000900000000000001B84F5A454B4953595354454D00
2020-09-09 13:48:42.736 INFO smpp_user_1: 127.0.0.1:61895 -> 000000100000001500000000000001B9
2020-09-09 13:48:42.736 INFO smpp_user_1: 127.0.0.1:61895 <- 000000108000001500000000000001B9
2020-09-09 13:48:50.420 INFO smpp_user_1: 127.0.0.1:61895 -> 000000430000000400000000000001BA0000012B393835353638373434323100010 12B34343938
2020-09-09 13:48:50.420 INFO smpp_user_1: 127.0.0.1:61895 <- 0000001B8000000400000000000001BA3836313832383632383100
2020-09-09 13:48:50.420 INFO smpp_user_1: Received: +98556874421->+44987654231 'Hello Word'
2020-09-09 13:48:50.420 WARN smpp_user_1: Submit rejected at . The message did not go through routing. +98556874421 -> +44987654231 'Hello W
2020-09-09 13:48:50.420 INFO smpp_user_1: 127.0.0.1:61895 <- 000000AD000000050000000000000000100010139383535363837343432310001013434393837363
2020-09-09 13:48:50.420 INFO smpp_user_1: 127.0.0.1:61895 -> 00000011800000050000000000000000100
```

Figure 7 - Received message rejected

# How to find the delivery log of an SMS message

*This guide explains where you can find the delivery history of a sent message. You will learn about the sent messages folder and how you can check the details of the delivery. This report contains information about the route of the message, what component took part during the process and the answer of the recipient if it subimtted the message or not. For that procedure, the guide is going to use the default admin user. Finally you will find information about where you can find the communicatoin logs of an SMS connection. So, let's get started.*

## Step 1 - Select the admin user



SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.

SMPP_client_1 (SMPP client) 0.00 MPM
Ready

Add new connection...

| N° | Route name | From | To | |
|---|---|---|---|---|
| 1. | defin_admin | Any_SMS_Connection | admin | Copy |

Add new route...

1 route installed.

| N° | Route name | From | To | Mode |
|---|---|---|---|---|
| 1. | defout_sms | Any_SMS_User | Any_SMS_Connection | Move |

Add new route...

1 route installed.

admin (Admin) 0.00 MPM
Ready

Add new user/application...

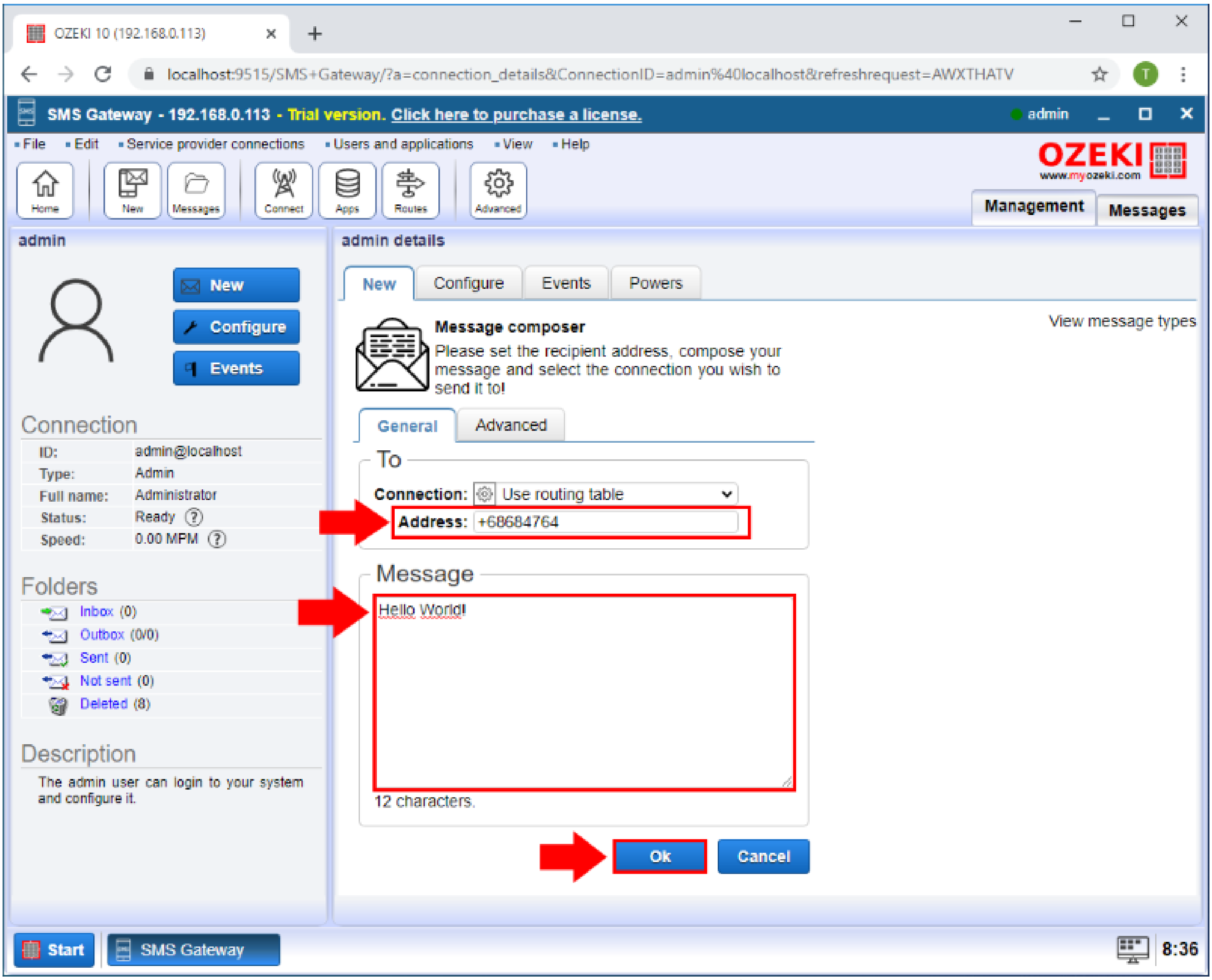
1 connection installed

3 users/applications installed

Figure 1 - Select the admin user in SMS Gateway

## Step 2 - Send the test message

From the Admin user you can send SMS message. Provide the recipient address, the message and click on the OK button (Figure 2).



Message composer

Please set the recipient address, compose your message and select the connection you wish to send it to!

Connection

| | |
|---|---|
| ID: | admin@localhost |
| Type: | Admin |
| Full name: | Administrator |
| Status: | Ready |
| Speed: | 0.00 MPM |

Folders

- Inbox (0)
- Outbox (0/0)
- Sent (0)
- Not sent (0)
- Deleted (8)

Description

The admin user can login to your system and configure it.

To

Connection: Use routing table

Address: +68684764

Message

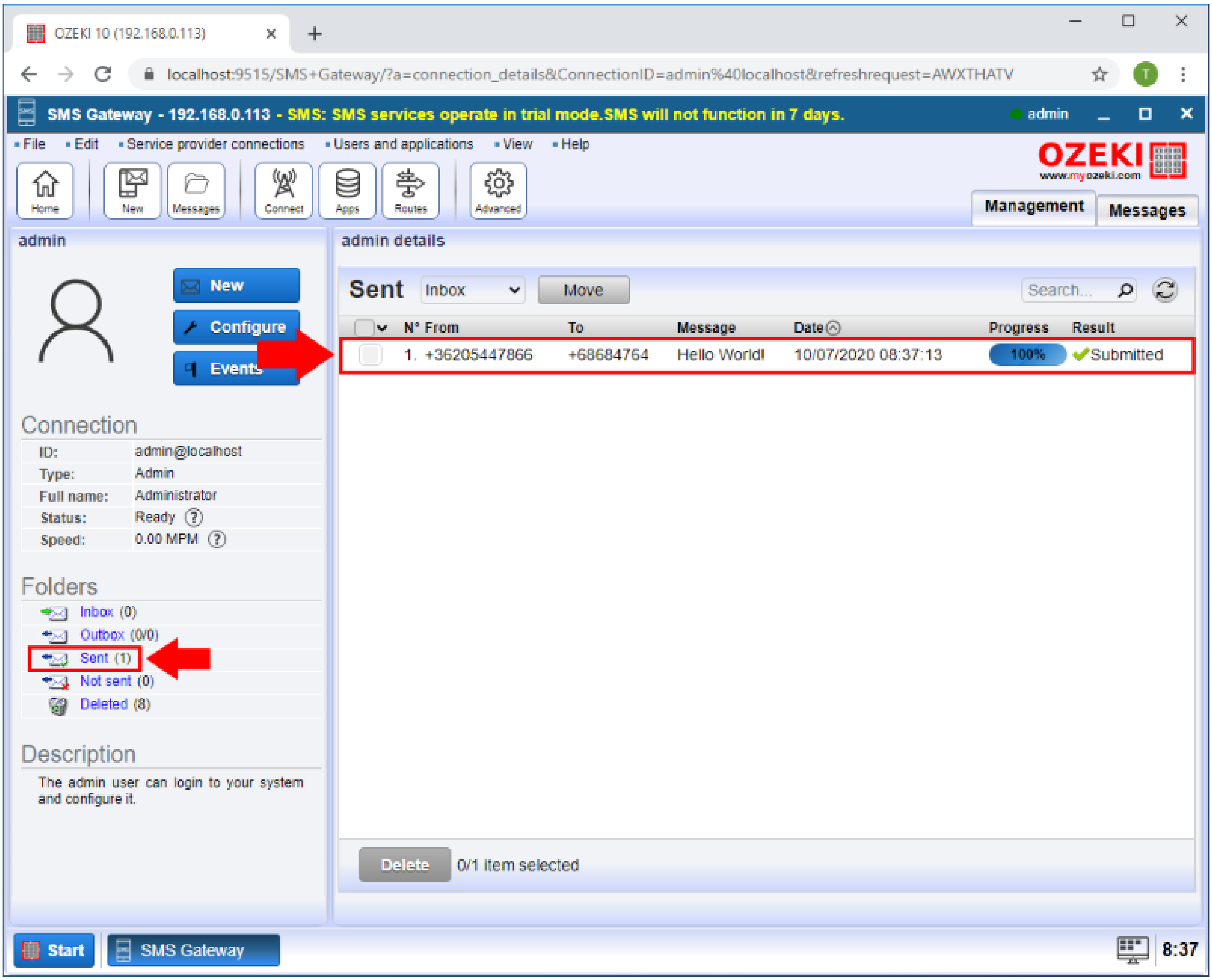
Hello World!

12 characters.

Figure 2 - Message composer page of admin user

# Step 3 - Message in sent folder

After it in the sent folder you can see the message. Thus the phone number of the sender and the recipient, the text of the message, and the time of sending (Figure 3).


OZEKI 10 (192.168.0.113)
localhost:9515/SMS+Gateway/?a=connection_details&ConnectionID=admin%40localhost&refreshrequest=AWXTHATV
SMS Gateway - 192.168.0.113 - SMS: SMS services operate in trial mode.SMS will not function in 7 days.
admin
File
Edit
Service provider connections
Users and applications
View
Help
Home
New
Messages
Connect
Apps
Routes
Advanced
OZEKI
www.myozeki.com
Management
Messages
admin
New
Configure
Events
admin details
Sent
Inbox
Move
Search...
Delete
Start
SMS Gateway
8:37

| N° | From | To | Message | Date | Progress | Result |
|---|---|---|---|---|---|---|
| 1. | +36205447866 | +68684764 | Hello World! | 10/07/2020 08:37:13 | 100% | ✓Submitted |

0/1 item selected

Connection

| | |
|---|---|
| ID: | admin@localhost |
| Type: | Admin |
| Full name: | Administrator |
| Status: | Ready |
| Speed: | 0.00 MPM |

Folders

- Inbox (0)
- Outbox (0/0)
- Sent (1)
- Not sent (0)
- Deleted (8)

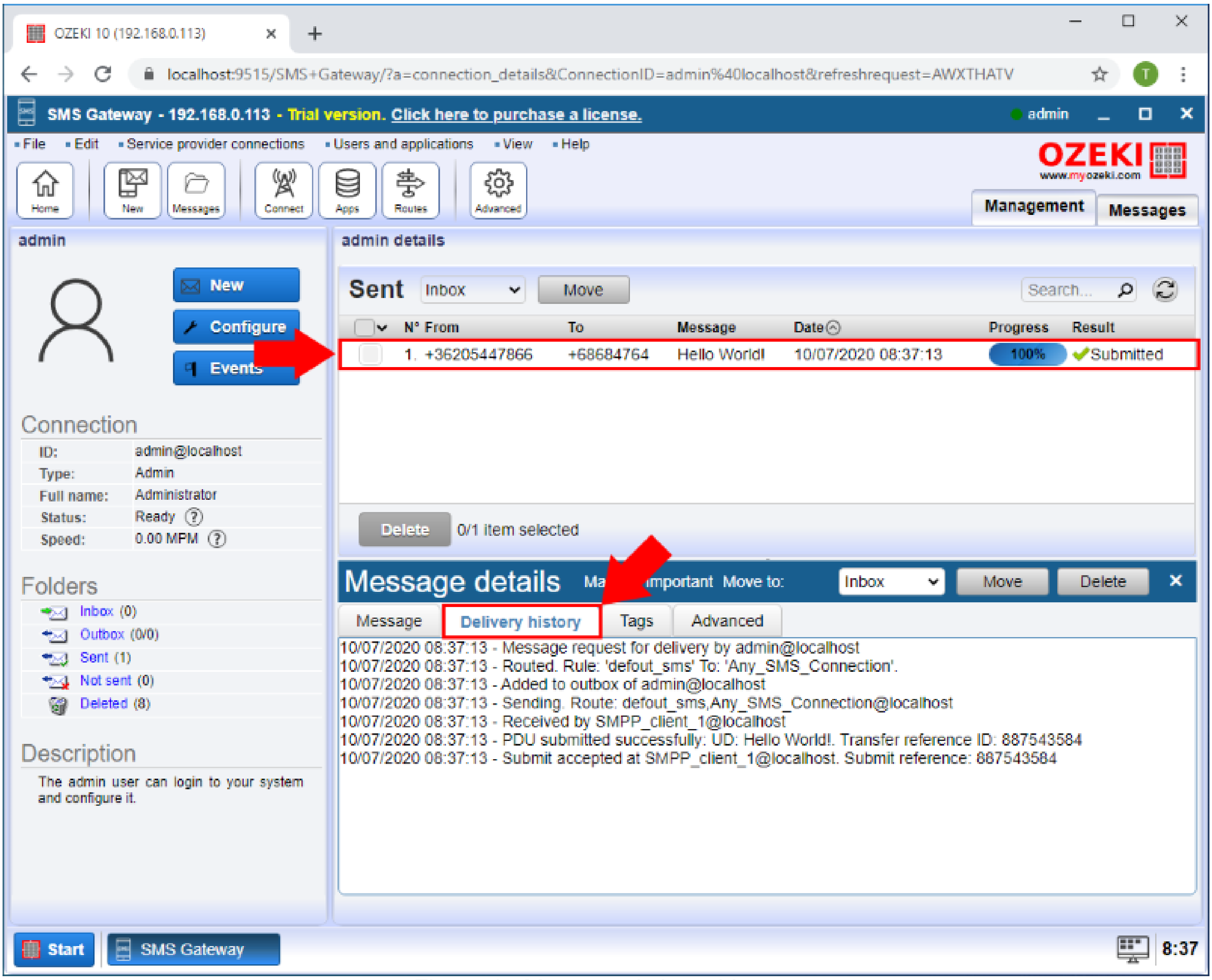
Description

The admin user can login to your system and configure it.

Figure 3 - Message in sent folder

## Step 4 - Message in sent folder

By clicking on the message you can see the details of it. You can still have further information if you click on "Delivery history" tab.



| N° | From | To | Message | Date | Progress | Result |
|---|---|---|---|---|---|---|
| 1. | +36205447866 | +68684764 | Hello World! | 10/07/2020 08:37:13 | 100% | ✓Submitted |

0/1 item selected

Connection

| | |
|---|---|
| ID: | admin@localhost |
| Type: | Admin |
| Full name: | Administrator |
| Status: | Ready |
| Speed: | 0.00 MPM |

Folders

- Inbox (0)
- Outbox (0/0)
- Sent (1)
- Not sent (0)
- Deleted (8)

Description

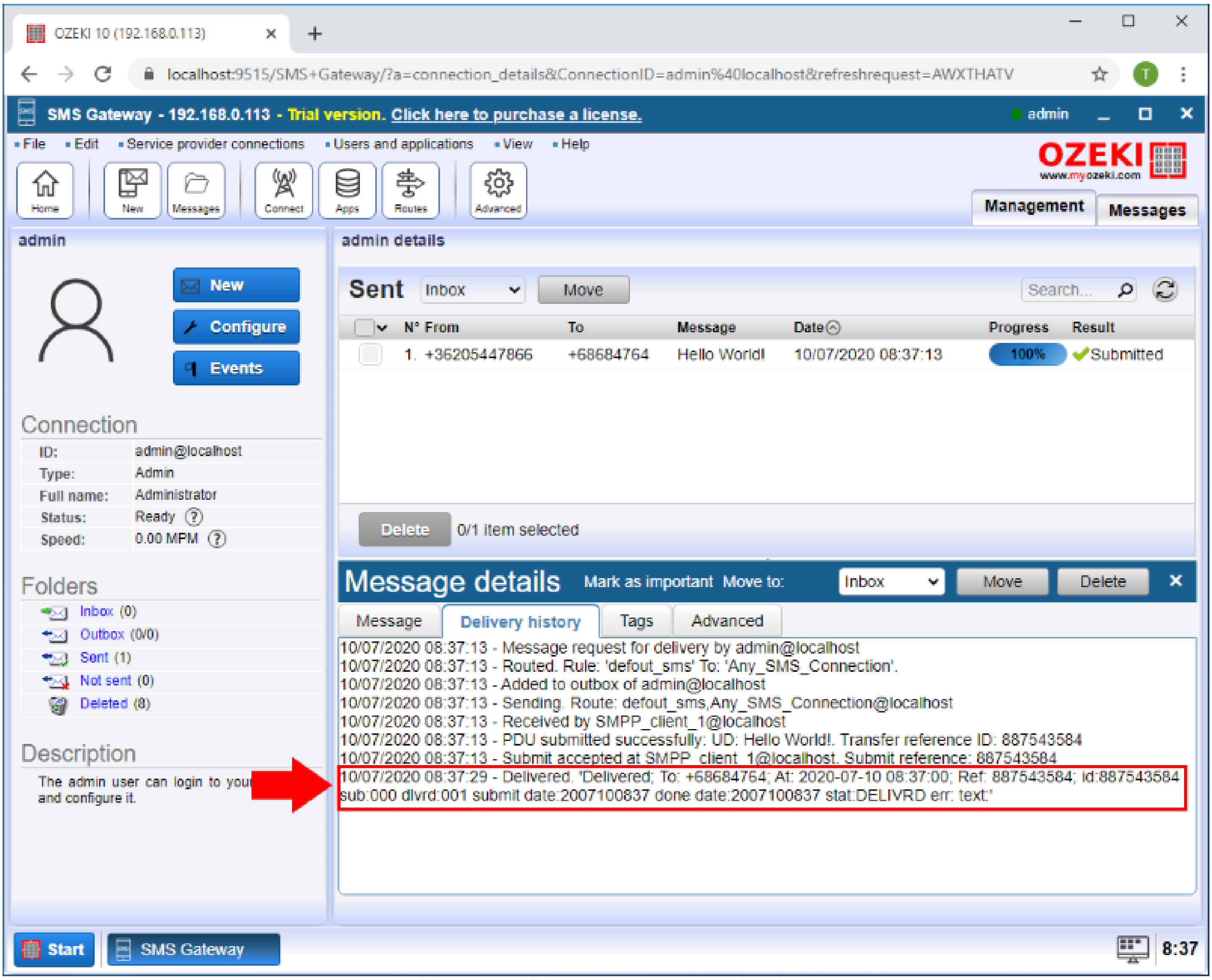
The admin user can login to your system and configure it.

```
10/07/2020 08:37:13 - Message request for delivery by admin@localhost
10/07/2020 08:37:13 - Routed. Rule: 'defout_sms' To: 'Any_SMS_Connection'.
10/07/2020 08:37:13 - Added to outbox of admin@localhost
10/07/2020 08:37:13 - Sending. Route: defout_sms,Any_SMS_Connection@localhost
10/07/2020 08:37:13 - Received by SMPP_client_1@localhost
10/07/2020 08:37:13 - PDU submitted successfully: UD: Hello World!. Transfer reference ID: 887543584
10/07/2020 08:37:13 - Submit accepted at SMPP_client_1@localhost. Submit reference: 887543584
```

Figure 4 - Message delivery history

## Step 5 - Message in sent folder

If the service provider send back the Delivery report for the message you can see this information here.



Connection

| | |
|---|---|
| ID: | admin@localhost |
| Type: | Admin |
| Full name: | Administrator |
| Status: | Ready |
| Speed: | 0.00 MPM |

Description

The admin user can login to your and configure it.

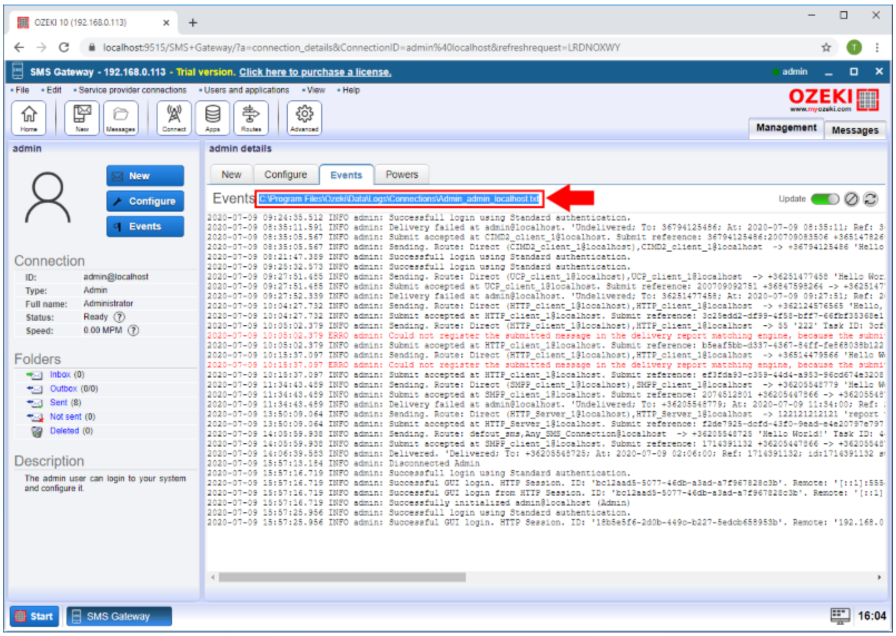
| N° | From | To | Message | Date | Progress | Result |
|---|---|---|---|---|---|---|
| 1. | +36205447866 | +68684764 | Hello World! | 10/07/2020 08:37:13 | 100% | ✓Submitted |

```
10/07/2020 08:37:13 - Message request for delivery by admin@localhost
10/07/2020 08:37:13 - Routed. Rule: 'defout_sms' To: 'Any_SMS_Connection'.
10/07/2020 08:37:13 - Added to outbox of admin@localhost
10/07/2020 08:37:13 - Sending. Route: defout_sms,Any_SMS_Connection@localhost
10/07/2020 08:37:13 - Received by SMPP_client_1@localhost
10/07/2020 08:37:13 - PDU submitted successfully: UD: Hello World!. Transfer reference ID: 887543584
10/07/2020 08:37:13 - Submit accepted at SMPP_client_1@localhost. Submit reference: 887543584
10/07/2020 08:37:29 - Delivered. 'Delivered; To: +68684764; At: 2020-07-10 08:37:00; Ref: 887543584; id:887543584
sub:000 dlvrd:001 submit date:2007100837 done date:2007100837 stat:DELIVRD err: text:'
```

Figure 5 - Delivery report received

# Step 6 - Message in the connection's log

To get more detailed information about how a message was sent through a connection, you can read the detailed log of the connection that was used to submit the messages. To this, you must open the details page of the connection and select the Events tab. This tab page contains the last 100 log messages, and the file name of the log file. To read the full log, copy the log file's location and use notepad to open the file (Figure 1).



Events C:\Program Files\Ozeki\Data\Logs\Connections\Admin_admin_localhost.txt

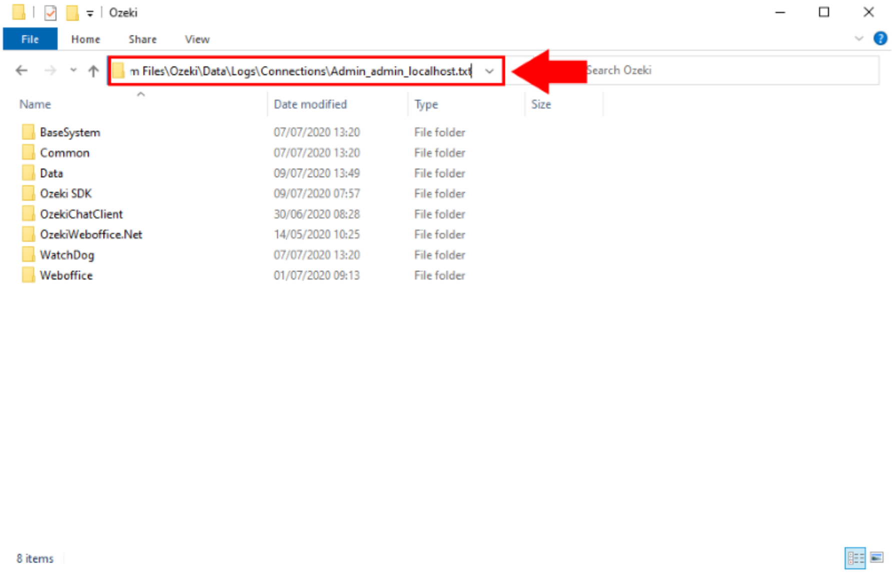
Connection
ID: admin@localhost
Type: Admin
Full name: Administrator
Status: Ready
Speed: 0.00 MPM

Description
The admin user can login to your system and configure it.

```
2020-07-09 09:24:35.512 INFO admin: Successfull login using Standard authentication.
2020-07-09 08:35:11.591 INFO admin: Delivery failed at admin@localhost. 'Undelivered; To: 36794125486; At: 2020-07-09 08:35:11; Ref: 3
2020-07-09 08:35:05.567 INFO admin: Submit accepted at CIMD2_client_1@localhost. Submit reference: 36794125486:200709083506 +36514782€
2020-07-09 08:35:05.567 INFO admin: Sending. Route: Direct (CIMD2_client_1@localhost),CIMD2_client_1@localhost  -> +36794125486 'Hello
2020-07-09 08:21:47.389 INFO admin: Successfull login using Standard authentication.
2020-07-09 09:25:32.573 INFO admin: Successfull login using Standard authentication.
2020-07-09 09:27:51.485 INFO admin: Sending. Route: Direct (UCP_client_1@localhost),UCP_client_1@localhost  -> +36251477458 'Hello Wor
2020-07-09 09:27:51.485 INFO admin: Submit accepted at UCP_client_1@localhost. Submit reference: 200709092751 +36847598264 -> +3625147
2020-07-09 09:27:52.339 INFO admin: Delivery failed at admin@localhost. 'Undelivered; To: 36251477458; At: 2020-07-09 09:27:51; Ref: 2
2020-07-09 10:04:27.732 INFO admin: Sending. Route: Direct (HTTP_client_1@localhost),HTTP_client_1@localhost  -> +36212476565 'Hello,
2020-07-09 10:04:27.732 INFO admin: Submit accepted at HTTP_client_1@localhost. Submit reference: 3c25edd2-df99-4f58-bff7-66fbf35368e1
2020-07-09 10:05:02.379 INFO admin: Sending. Route: Direct (HTTP_client_1@localhost),HTTP_client_1@localhost  -> 55 '222' Task ID: 3cf
2020-07-09 10:05:02.379 ERRO admin: Could not register the submitted message in the delivery report matching engine, because the submi
2020-07-09 10:05:02.379 INFO admin: Submit accepted at HTTP_client_1@localhost. Submit reference: b5eaf5bb-d337-4367-84ff-fe868038b122
2020-07-09 10:15:37.097 INFO admin: Sending. Route: Direct (HTTP_client_1@localhost),HTTP_client_1@localhost  -> +36514479566 'Hello W
2020-07-09 10:15:37.097 ERRO admin: Could not register the submitted message in the delivery report matching engine, because the submi
2020-07-09 10:15:37.097 INFO admin: Submit accepted at HTTP_client_1@localhost. Submit reference: ef3fda93-c359-44d4-a953-96cd674e3208
2020-07-09 11:34:43.489 INFO admin: Sending. Route: Direct (SMPP_client_1@localhost),SMPP_client_1@localhost  -> +36205548779 'Hello W
2020-07-09 11:34:43.489 INFO admin: Submit accepted at SMPP_client_1@localhost. Submit reference: 2074512801 +36205447866 -> +36205548
2020-07-09 11:34:43.489 INFO admin: Delivery failed at admin@localhost. 'Undelivered; To: +36205548779; At: 2020-07-09 11:34:00; Ref:
2020-07-09 13:50:09.064 INFO admin: Sending. Route: Direct (HTTP_Server_1@localhost),HTTP_Server_1@localhost  -> 12212121212 'report
2020-07-09 13:50:09.064 INFO admin: Submit accepted at HTTP_Server_1@localhost. Submit reference: f2de7925-dcfd-43f0-9ead-e4e20797e797
2020-07-09 14:05:59.938 INFO admin: Sending. Route: defout_sms,Any_SMS_Connection@localhost  -> +36205548725 'Hello World!' Task ID: 4
2020-07-09 14:05:59.938 INFO admin: Submit accepted at SMPP_client_1@localhost. Submit reference: 1714391132 +36205447866 -> +36205548
2020-07-09 14:06:39.583 INFO admin: Delivered. 'Delivered; To: +36205548725; At: 2020-07-09 02:06:00; Ref: 1714391132; id:1714391132 s
2020-07-09 15:57:13.184 INFO admin: Disconnected Admin
2020-07-09 15:57:16.719 INFO admin: Successfull login using Standard authentication.
2020-07-09 15:57:16.719 INFO admin: Successful GUI login. HTTP Session. ID: 'bc12aad5-5077-46db-a3ad-a7f967828c3b'. Remote: '[::1]:555
2020-07-09 15:57:16.719 INFO admin: Successful GUI login from HTTP Session. ID: 'bc12aad5-5077-46db-a3ad-a7f967828c3b'. Remote: '[::1]
2020-07-09 15:57:16.719 INFO admin: Successfully initialized admin@localhost (Admin)
2020-07-09 15:57:25.956 INFO admin: Successfull login using Standard authentication.
2020-07-09 15:57:25.956 INFO admin: Successful GUI login. HTTP Session. ID: '18b5e5f6-2d0b-449c-b227-5edcb658953b'. Remote: '192.168.0
```

Figure 6 - Copy Log file path

Paste the path into windows explorer and press enter (Figure 2).



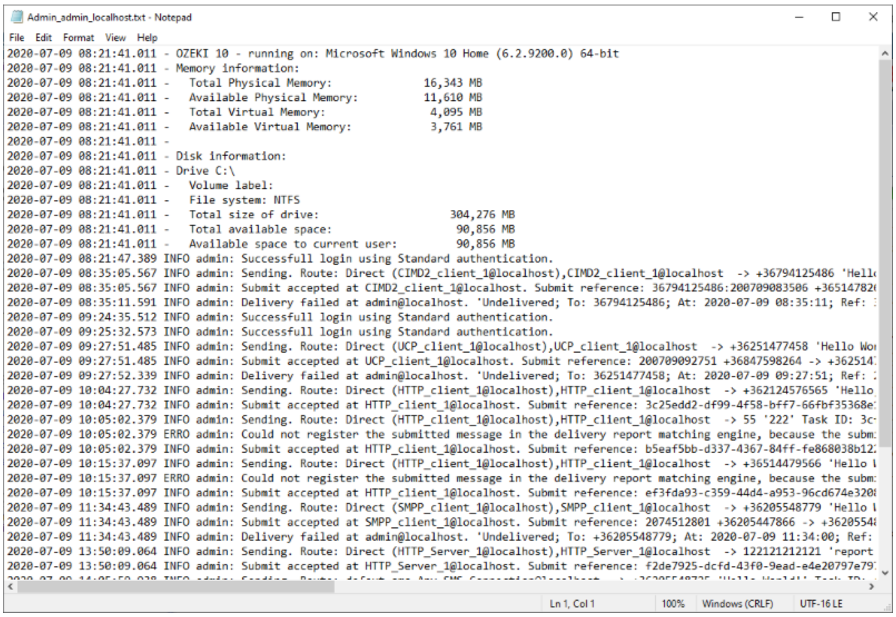
m Files\Ozeki\Data\Logs\Connections\Admin_admin_localhost.txt

| Name | Date modified | Type | Size |
|---|---|---|---|
| BaseSystem | 07/07/2020 13:20 | File folder | |
| Common | 07/07/2020 13:20 | File folder | |
| Data | 09/07/2020 13:49 | File folder | |
| Ozeki SDK | 09/07/2020 07:57 | File folder | |
| OzekiChatClient | 30/06/2020 08:28 | File folder | |
| OzekiWeboffice.Net | 14/05/2020 10:25 | File folder | |
| WatchDog | 07/07/2020 13:20 | File folder | |
| Weboffice | 01/07/2020 09:13 | File folder | |

8 items

Figure 7 - Paste the path in the windows explorer

Windows explorer will open notepad and display the contents of the logfile.



Admin_admin_localhost.txt - Notepad
File Edit Format View Help
Ln 1, Col 1
100%
Windows (CRLF)
UTF-16 LE

```
2020-07-09 08:21:41.011 - OZEKI 10 - running on: Microsoft Windows 10 Home (6.2.9200.0) 64-bit
2020-07-09 08:21:41.011 - Memory information:
2020-07-09 08:21:41.011 -   Total Physical Memory:            16,343 MB
2020-07-09 08:21:41.011 -   Available Physical Memory:        11,610 MB
2020-07-09 08:21:41.011 -   Total Virtual Memory:              4,095 MB
2020-07-09 08:21:41.011 -   Available Virtual Memory:          3,761 MB
2020-07-09 08:21:41.011 -
2020-07-09 08:21:41.011 - Disk information:
2020-07-09 08:21:41.011 - Drive C:\
2020-07-09 08:21:41.011 -   Volume label:
2020-07-09 08:21:41.011 -   File system: NTFS
2020-07-09 08:21:41.011 -   Total size of drive:                  304,276 MB
2020-07-09 08:21:41.011 -   Total available space:                 90,856 MB
2020-07-09 08:21:41.011 -   Available space to current user:       90,856 MB
2020-07-09 08:21:47.389 INFO admin: Successfull login using Standard authentication.
2020-07-09 08:35:05.567 INFO admin: Sending. Route: Direct (CIMD2_client_1@localhost),CIMD2_client_1@localhost  -> +36794125486 'Hell
2020-07-09 08:35:05.567 INFO admin: Submit accepted at CIMD2_client_1@localhost. Submit reference: 36794125486:200709083506 +36514782
2020-07-09 08:35:11.591 INFO admin: Delivery failed at admin@localhost. 'Undelivered; To: 36794125486; At: 2020-07-09 08:35:11; Ref:
2020-07-09 09:24:35.512 INFO admin: Successfull login using Standard authentication.
2020-07-09 09:25:32.573 INFO admin: Successfull login using Standard authentication.
2020-07-09 09:27:51.485 INFO admin: Sending. Route: Direct (UCP_client_1@localhost),UCP_client_1@localhost  -> +36251477458 'Hello Wo
2020-07-09 09:27:51.485 INFO admin: Submit accepted at UCP_client_1@localhost. Submit reference: 200709092751 +36847598264 -> +362514
2020-07-09 09:27:52.339 INFO admin: Delivery failed at admin@localhost. 'Undelivered; To: 36251477458; At: 2020-07-09 09:27:51; Ref:
2020-07-09 10:04:27.732 INFO admin: Sending. Route: Direct (HTTP_client_1@localhost),HTTP_client_1@localhost  -> +362124576565 'Hello
2020-07-09 10:04:27.732 INFO admin: Submit accepted at HTTP_client_1@localhost. Submit reference: 3c25edd2-df99-4f58-bff7-66fbf35368e
2020-07-09 10:05:02.379 INFO admin: Sending. Route: Direct (HTTP_client_1@localhost),HTTP_client_1@localhost  -> 55 '222' Task ID: 3c
2020-07-09 10:05:02.379 ERRO admin: Could not register the submitted message in the delivery report matching engine, because the subm
2020-07-09 10:05:02.379 INFO admin: Submit accepted at HTTP_client_1@localhost. Submit reference: b5eaf5bb-d337-4367-84ff-fe868038b12
2020-07-09 10:15:37.097 INFO admin: Sending. Route: Direct (HTTP_client_1@localhost),HTTP_client_1@localhost  -> +36514479566 'Hello
2020-07-09 10:15:37.097 ERRO admin: Could not register the submitted message in the delivery report matching engine, because the subm
2020-07-09 10:15:37.097 INFO admin: Submit accepted at HTTP_client_1@localhost. Submit reference: ef3fda93-c359-44d4-a953-96cd674e320
2020-07-09 11:34:43.489 INFO admin: Sending. Route: Direct (SMPP_client_1@localhost),SMPP_client_1@localhost  -> +36205548779 'Hello
2020-07-09 11:34:43.489 INFO admin: Submit accepted at SMPP_client_1@localhost. Submit reference: 2074512801 +36205447866 -> +3620554
2020-07-09 11:34:43.489 INFO admin: Delivery failed at admin@localhost. 'Undelivered; To: +36205548779; At: 2020-07-09 11:34:00; Ref:
2020-07-09 13:50:09.064 INFO admin: Sending. Route: Direct (HTTP_Server_1@localhost),HTTP_Server_1@localhost  -> 122121212121 'report
2020-07-09 13:50:09.064 INFO admin: Submit accepted at HTTP_Server_1@localhost. Submit reference: f2de7925-dcfd-43f0-9ead-e4e20797e79
```

Figure 8 - The file is open in the notepad

# Appendix

# Appendix - The 7 bit default alphabet of GSM phones

**This table gives you information about the GSM 7 bit alphabet used in text SMS messages. To see how you can send special symbols and international characters, please read the following document: SMS character set handling and multipart messages**

Your PC and your GSM phone use two different character sets: the ISO-8859-1 and the GSM 7 bit alphabet. Ozeki NG - SMS Gateway converts automatically between the corresponding character codes in these two tables. In some cases the automatic conversion might not fit your needs. If you want to be sure that a specific GSM character is used in your SMS message, you can enter its Hex value into the message. Before reading the explanation of how this is possible, please take a look at the characters that there are there in GSM.

The 7 bit GSM alphabet can be downloaded in PDF format from the following URL:
GSM-Alphabet-Character-Translations.pdf

This is the 7 bit default alphabet as specified by GSM 03.38. The corresponding ISO-8859-1 decimal codes are shown in the rightmost column. Note that the euro sign (€) is included in the ISO-8859-15 character set.

This is the 7 bit default alphabet as specified by GSM 03.38. The corresponding ISO-8859-1 decimal codes are shown in the rightmost column. Note that the euro sign (€) is a member of the ISO-8859-15 character set.

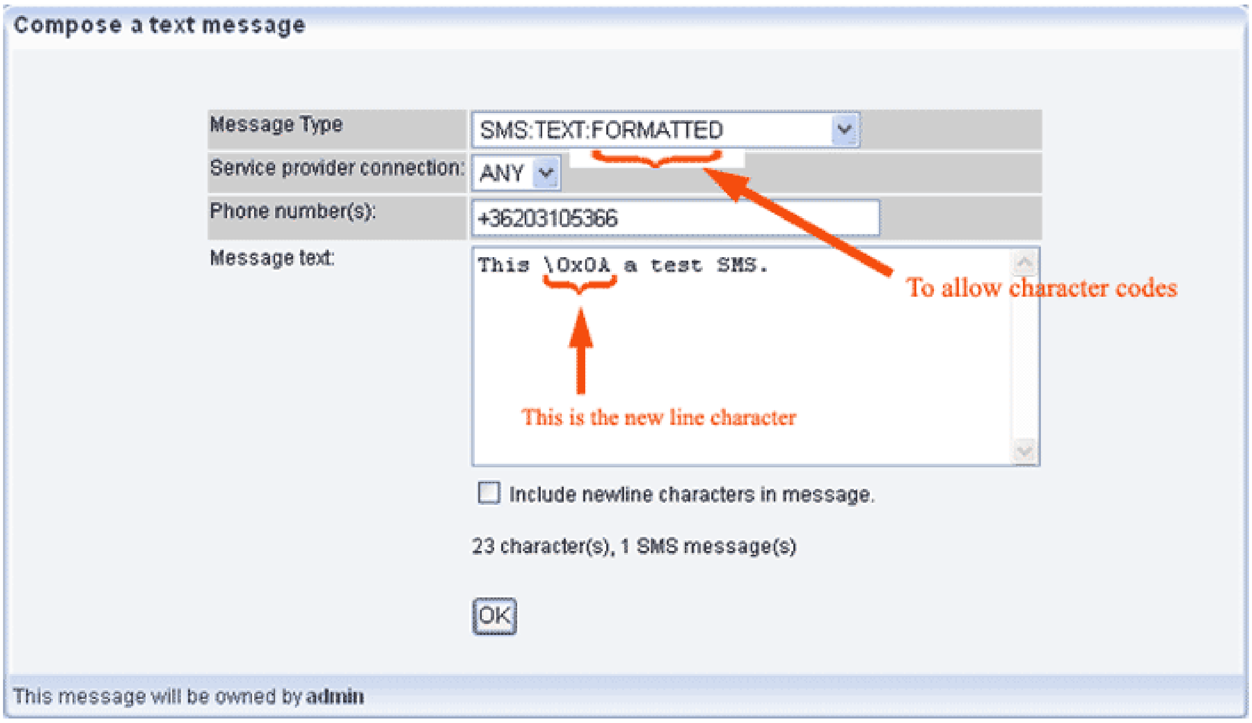
| Hex | Dec | Character name | Character | ISO-8859-1 DEC |
|---|---|---|---|---|
| 0x00 | 0 | COMMERCIAL AT | @ | 64 |
| 0x01 | 1 | POUND SIGN | £ | 163 |
| 0x02 | 2 | DOLLAR SIGN | $ | 36 |
| 0x03 | 3 | YEN SIGN | ¥ | 165 |
| 0x04 | 4 | LATIN SMALL LETTER E WITH GRAVE | è | 232 |
| 0x05 | 5 | LATIN SMALL LETTER E WITH ACUTE | é | 233 |
| 0x06 | 6 | LATIN SMALL LETTER U WITH GRAVE | ù | 249 |
| 0x07 | 7 | LATIN SMALL LETTER I WITH GRAVE | ì | 236 |
| 0x08 | 8 | LATIN SMALL LETTER O WITH GRAVE | ò | 242 |
| 0x09 | 9 | LATIN CAPITAL LETTER C WITH CEDILLA | Ç | 199 |
| 0x0A | 10 | LINE FEED |  | 10 |
| 0x0B | 11 | LATIN CAPITAL LETTER O WITH STROKE | Ø | 216 |
| 0x0C | 12 | LATIN SMALL LETTER O WITH STROKE | ø | 248 |
| 0x0D | 13 | CARRIAGE RETURN |  | 13 |
| 0x0E | 14 | LATIN CAPITAL LETTER A WITH RING ABOVE | Å | 197 |
| 0x0F | 15 | LATIN SMALL LETTER A WITH RING ABOVE | å | 229 |
| 0x10 | 16 | GREEK CAPITAL LETTER DELTA | Δ |  |
| 0x11 | 17 | LOW LINE | _ | 95 |
| 0x12 | 18 | GREEK CAPITAL LETTER PHI | Φ |  |
| 0x13 | 19 | GREEK CAPITAL LETTER GAMMA | Γ |  |
| 0x14 | 20 | GREEK CAPITAL LETTER LAMBDA | Λ |  |
| 0x15 | 21 | GREEK CAPITAL LETTER OMEGA | Ω |  |
| 0x16 | 22 | GREEK CAPITAL LETTER PI | Π |  |
| 0x17 | 23 | GREEK CAPITAL LETTER PSI | Ψ |  |
| 0x18 | 24 | GREEK CAPITAL LETTER SIGMA | Σ |  |
| 0x19 | 25 | GREEK CAPITAL LETTER THETA | Θ |  |
| 0x1A | 26 | GREEK CAPITAL LETTER XI | Ξ |  |
| 0x1B | 27 | ESCAPE TO EXTENSION TABLE |  |  |
| 0x1B0A | 27 10 | FORM FEED |  | 12 |
| 0x1B14 | 27 20 | CIRCUMFLEX ACCENT | ^ | 94 |

| | | | | |
|---|---|---|---|---|
| 0x1B28 | 27 40 | LEFT CURLY BRACKET | { | 123 |
| 0x1B29 | 27 41 | RIGHT CURLY BRACKET | } | 125 |
| 0x1B2F | 27 47 | REVERSE SOLIDUS (BACKSLASH) | \ | 92 |
| 0x1B3C | 27 60 | LEFT SQUARE BRACKET | [ | 91 |
| 0x1B3D | 27 61 | TILDE | ~ | 126 |
| 0x1B3E | 27 62 | RIGHT SQUARE BRACKET | ] | 93 |
| 0x1B40 | 27 64 | VERTICAL BAR | \| | 124 |
| 0x1B65 | 27 101 | EURO SIGN | € | 164 *(ISO-8859-15)* |
| 0x1C | 28 | LATIN CAPITAL LETTER AE | Æ | 198 |
| 0x1D | 29 | LATIN SMALL LETTER AE | æ | 230 |
| 0x1E | 30 | LATIN SMALL LETTER SHARP S (German) | ß | 223 |
| 0x1F | 31 | LATIN CAPITAL LETTER E WITH ACUTE | É | 201 |
| 0x20 | 32 | SPACE | | 32 |
| 0x21 | 33 | EXCLAMATION MARK | ! | 33 |
| 0x22 | 34 | QUOTATION MARK | " | 34 |
| 0x23 | 35 | NUMBER SIGN | # | 35 |
| 0x24 | 36 | CURRENCY SIGN | ¤ | 164 (ISO-8859-1) |
| 0x25 | 37 | PERCENT SIGN | % | 37 |
| 0x26 | 38 | AMPERSAND | & | 38 |
| 0x27 | 39 | APOSTROPHE | ' | 39 |
| 0x28 | 40 | LEFT PARENTHESIS | ( | 40 |
| 0x29 | 41 | RIGHT PARENTHESIS | ) | 41 |
| 0x2A | 42 | ASTERISK | * | 42 |
| 0x2B | 43 | PLUS SIGN | + | 43 |
| 0x2C | 44 | COMMA | , | 44 |
| 0x2D | 45 | HYPHEN-MINUS | - | 45 |
| 0x2E | 46 | FULL STOP | . | 46 |
| 0x2F | 47 | SOLIDUS (SLASH) | / | 47 |
| 0x30 | 48 | DIGIT ZERO | 0 | 48 |
| 0x31 | 49 | DIGIT ONE | 1 | 49 |
| 0x32 | 50 | DIGIT TWO | 2 | 50 |
| 0x33 | 51 | DIGIT THREE | 3 | 51 |
| 0x34 | 52 | DIGIT FOUR | 4 | 52 |
| 0x35 | 53 | DIGIT FIVE | 5 | 53 |
| 0x36 | 54 | DIGIT SIX | 6 | 54 |
| 0x37 | 55 | DIGIT SEVEN | 7 | 55 |
| 0x38 | 56 | DIGIT EIGHT | 8 | 56 |
| 0x39 | 57 | DIGIT NINE | 9 | 57 |
| 0x3A | 58 | COLON | : | 58 |
| 0x3B | 59 | SEMICOLON | ; | 59 |
| 0x3C | 60 | LESS-THAN SIGN | < | 60 |
| 0x3D | 61 | EQUALS SIGN | = | 61 |
| 0x3E | 62 | GREATER-THAN SIGN | > | 62 |
| 0x3F | 63 | QUESTION MARK | ? | 63 |
| 0x40 | 64 | INVERTED EXCLAMATION MARK | ¡ | 161 |
| 0x41 | 65 | LATIN CAPITAL LETTER A | A | 65 |
| 0x42 | 66 | LATIN CAPITAL LETTER B | B | 66 |
| 0x43 | 67 | LATIN CAPITAL LETTER C | C | 67 |
| 0x44 | 68 | LATIN CAPITAL LETTER D | D | 68 |
| 0x45 | 69 | LATIN CAPITAL LETTER E | E | 69 |
| 0x46 | 70 | LATIN CAPITAL LETTER F | F | 70 |
| 0x47 | 71 | LATIN CAPITAL LETTER G | G | 71 |
| 0x48 | 72 | LATIN CAPITAL LETTER H | H | 72 |

| | | | | |
|---|---|---|---|---|
| 0x49 | 73 | LATIN CAPITAL LETTER I | I | 73 |
| 0x4A | 74 | LATIN CAPITAL LETTER J | J | 74 |
| 0x4B | 75 | LATIN CAPITAL LETTER K | K | 75 |
| 0x4C | 76 | LATIN CAPITAL LETTER L | L | 76 |
| 0x4D | 77 | LATIN CAPITAL LETTER M | M | 77 |
| 0x4E | 78 | LATIN CAPITAL LETTER N | N | 78 |
| 0x4F | 79 | LATIN CAPITAL LETTER O | O | 79 |
| 0x50 | 80 | LATIN CAPITAL LETTER P | P | 80 |
| 0x51 | 81 | LATIN CAPITAL LETTER Q | Q | 81 |
| 0x52 | 82 | LATIN CAPITAL LETTER R | R | 82 |
| 0x53 | 83 | LATIN CAPITAL LETTER S | S | 83 |
| 0x54 | 84 | LATIN CAPITAL LETTER T | T | 84 |
| 0x55 | 85 | LATIN CAPITAL LETTER U | U | 85 |
| 0x56 | 86 | LATIN CAPITAL LETTER V | V | 86 |
| 0x57 | 87 | LATIN CAPITAL LETTER W | W | 87 |
| 0x58 | 88 | LATIN CAPITAL LETTER X | X | 88 |
| 0x59 | 89 | LATIN CAPITAL LETTER Y | Y | 89 |
| 0x5A | 90 | LATIN CAPITAL LETTER Z | Z | 90 |
| 0x5B | 91 | LATIN CAPITAL LETTER A WITH DIAERESIS | Ä | 196 |
| 0x5C | 92 | LATIN CAPITAL LETTER O WITH DIAERESIS | Ö | 214 |
| 0x5D | 93 | LATIN CAPITAL LETTER N WITH TILDE | Ñ | 209 |
| 0x5E | 94 | LATIN CAPITAL LETTER U WITH DIAERESIS | Ü | 220 |
| 0x5F | 95 | SECTION SIGN | § | 167 |
| 0x60 | 96 | INVERTED QUESTION MARK | ¿ | 191 |
| 0x61 | 97 | LATIN SMALL LETTER A | a | 97 |
| 0x62 | 98 | LATIN SMALL LETTER B | b | 98 |
| 0x63 | 99 | LATIN SMALL LETTER C | c | 99 |
| 0x64 | 100 | LATIN SMALL LETTER D | d | 100 |
| 0x65 | 101 | LATIN SMALL LETTER E | e | 101 |
| 0x66 | 102 | LATIN SMALL LETTER F | f | 102 |
| 0x67 | 103 | LATIN SMALL LETTER G | g | 103 |
| 0x68 | 104 | LATIN SMALL LETTER H | h | 104 |
| 0x69 | 105 | LATIN SMALL LETTER I | i | 105 |
| 0x6A | 106 | LATIN SMALL LETTER J | j | 106 |
| 0x6B | 107 | LATIN SMALL LETTER K | k | 107 |
| 0x6C | 108 | LATIN SMALL LETTER L | l | 108 |
| 0x6D | 109 | LATIN SMALL LETTER M | m | 109 |
| 0x6E | 110 | LATIN SMALL LETTER N | n | 110 |
| 0x6F | 111 | LATIN SMALL LETTER O | o | 111 |
| 0x70 | 112 | LATIN SMALL LETTER P | p | 112 |
| 0x71 | 113 | LATIN SMALL LETTER Q | q | 113 |
| 0x72 | 114 | LATIN SMALL LETTER R | r | 114 |
| 0x73 | 115 | LATIN SMALL LETTER S | s | 115 |
| 0x74 | 116 | LATIN SMALL LETTER T | t | 116 |
| 0x75 | 117 | LATIN SMALL LETTER U | u | 117 |
| 0x76 | 118 | LATIN SMALL LETTER V | v | 118 |
| 0x77 | 119 | LATIN SMALL LETTER W | w | 119 |
| 0x78 | 120 | LATIN SMALL LETTER X | x | 120 |
| 0x79 | 121 | LATIN SMALL LETTER Y | y | 121 |
| 0x7A | 122 | LATIN SMALL LETTER Z | z | 122 |
| 0x7B | 123 | LATIN SMALL LETTER A WITH DIAERESIS | ä | 228 |
| 0x7C | 124 | LATIN SMALL LETTER O WITH DIAERESIS | ö | 246 |
| 0x7D | 125 | LATIN SMALL LETTER N WITH TILDE | ñ | 241 |
| | | | | |

|  |  |  |  |  |
|---|---|---|---|---|
| 0x7E | 126 | LATIN SMALL LETTER U WITH DIAERESIS | ü | 252 |
| 0x7F | 127 | LATIN SMALL LETTER A WITH GRAVE | à | 224 |

If you wish to use any of these characters, you can do it by entering a special string into your SMS messages. For example, if you wish to add a new line character you should enter \0x0A instead of pressing the enter key (Figure 1). Make sure you write out all the characters of this code. It is a five-character-string!



Compose a text message

Message Type: SMS:TEXT:FORMATTED

Service provider connection: ANY

Phone number(s): +36203105366

Message text: This \0x0A a test SMS.

☐ Include newline characters in message.

23 character(s), 1 SMS message(s)

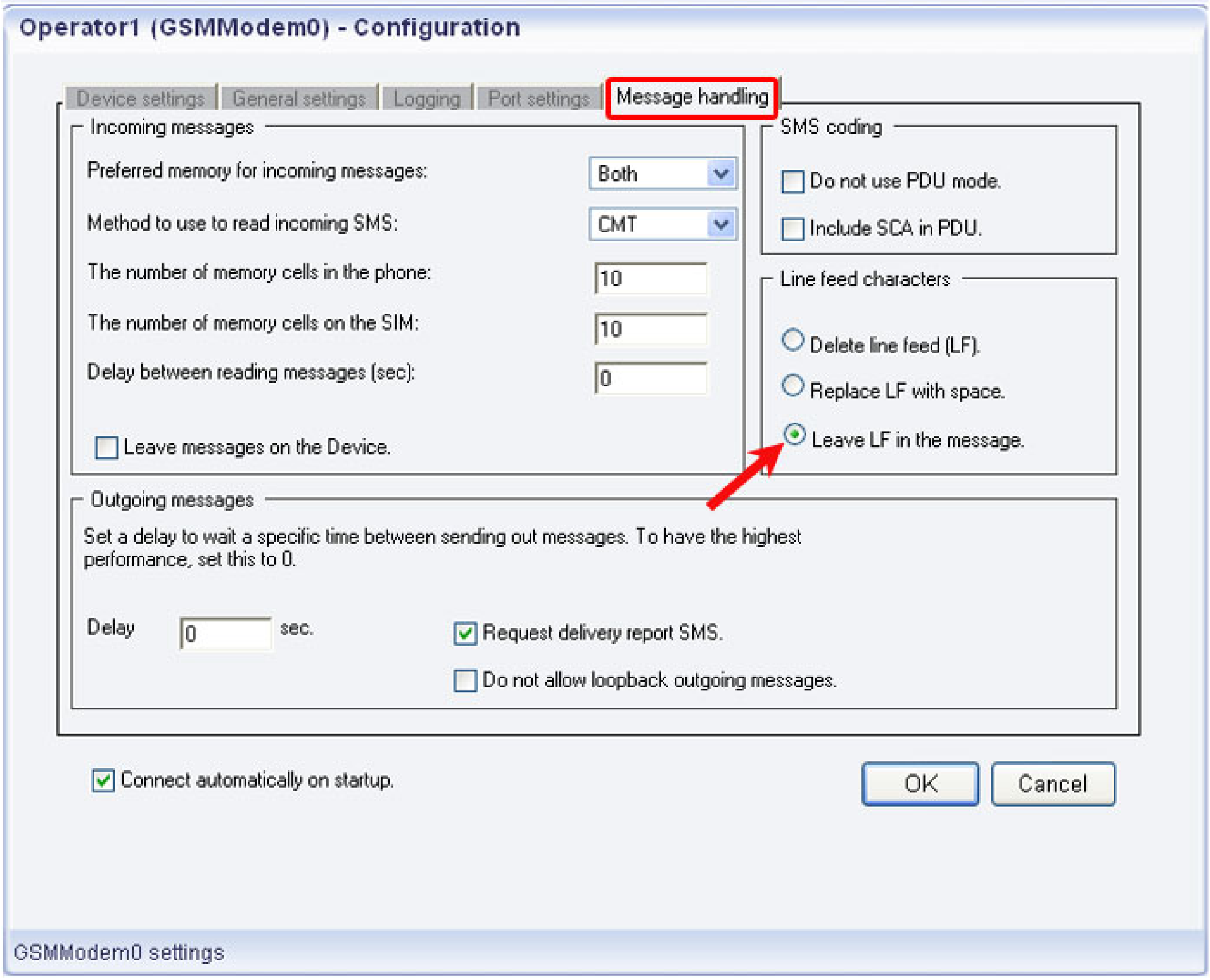
This message will be owned by **admin**

To allow character codes

This is the new line character

Figure 1 - Using the new line character

If you wish to receive SMS messages with a new line character, you need to set up the function in the GSM modem configuration window (Figure 2).



Operator1 (GSMModem0) - Configuration

Incoming messages

Preferred memory for incoming messages: Both

Method to use to read incoming SMS: CMT

The number of memory cells in the phone: 10

The number of memory cells on the SIM: 10

Delay between reading messages (sec): 0

□ Leave messages on the Device.

SMS coding

□ Do not use PDU mode.

□ Include SCA in PDU.

Line feed characters

○ Delete line feed (LF).

○ Replace LF with space.

● Leave LF in the message.

Outgoing messages

Set a delay to wait a specific time between sending out messages. To have the highest performance, set this to 0.

Delay 0 sec.

☑ Request delivery report SMS.

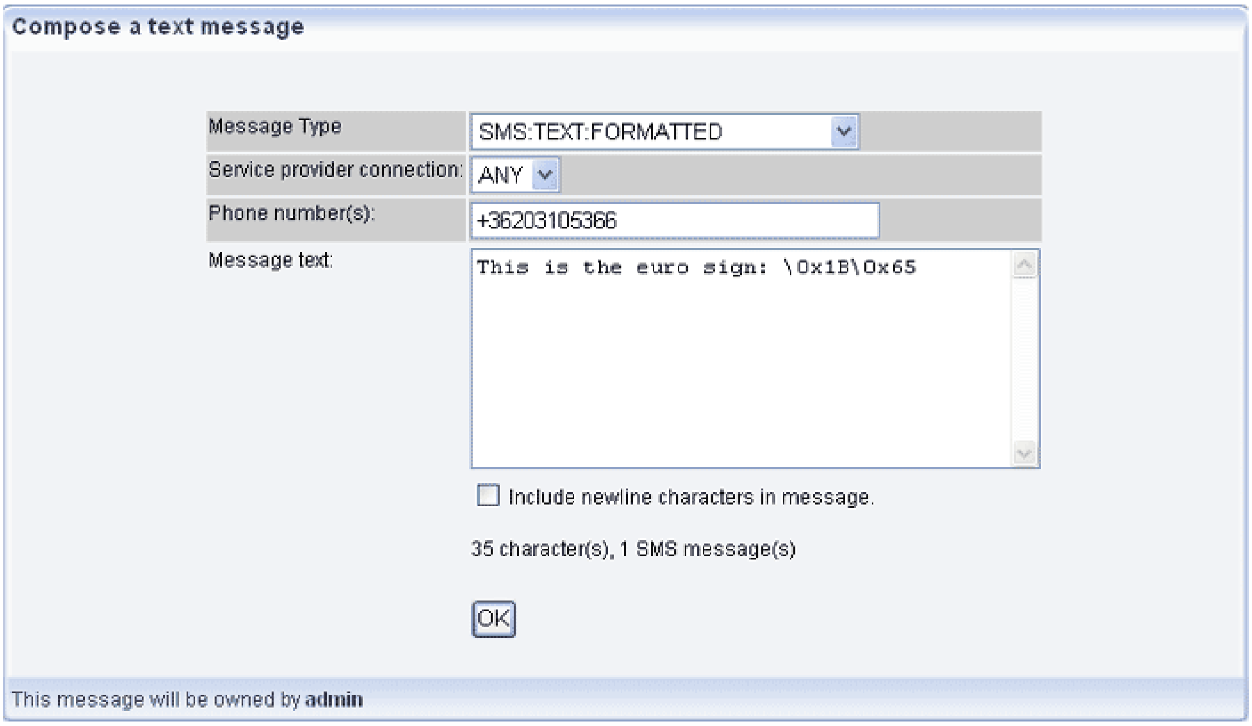
□ Do not allow loopback outgoing messages.

☑ Connect automatically on startup.

GSMModem0 settings

Figure 2 - Setting up the new line character for incoming messages

To send characters that require more than one byte, use the format in Figure 3 below.



Compose a text message

Message Type: SMS:TEXT:FORMATTED

Service provider connection: ANY

Phone number(s): +36203105366

Message text: This is the euro sign: \0x1B\0x65

□ Include newline characters in message.

35 character(s), 1 SMS message(s)

This message will be owned by **admin**

Figure 3 - Including a euro sign (€) in the message

Note: Please use version v6.3.2 or a more recent one for proper character encoding.

# Appendix - GSM Error Codes

***If a message send attempt fails, Ozeki NG - SMS Gateway creates a log entry, containing one of the following error codes:***

| | |
|---|---|
| 1 | Unassigned (unallocated) number |
| 8 | Operator determined barring |
| 10 | Call barred |
| 21 | Short message transfer rejected |
| 27 | Destination out of service |
| 28 | Unidentified subscriber |
| 29 | Facility rejected |
| 30 | Unknown subscriber |
| 38 | Network out of order |
| 41 | Temporary failure |
| 42 | Congestion |
| 47 | Resources unavailable, unspecified |
| 50 | Requested facility not subscribed |
| 69 | Requested facility not implemented |
| 81 | Invalid short message transfer reference value |
| 95 | Invalid message, unspecified |
| 96 | Invalid mandatory information |
| 97 | Message type non-existent or not implemented |
| 98 | Message not compatible with short message protocol state |
| 99 | Information element non-existent or not implemented |
| 111 | Protocol error, unspecified |
| 127 | Interworking, unspecified |
| 128 | Telematic interworking not supported |
| 129 | Short message Type 0 not supported |
| 130 | Cannot replace short message |
| 143 | Unspecified TP-PID error |
| 144 | Data coding scheme (alphabet) not supported |
| 145 | Message class not supported |
| 159 | Unspecified TP-DCS error |
| 160 | Command cannot be actioned |
| 161 | Command unsupported |
| 175 | Unspecified TP-Command error |
| 176 | TPDU not supported |
| 192 | SC busy |
| 193 | No SC subscription |
| 194 | SC system failure |
| 195 | Invalid SME address |
| 196 | Destination SME barred |
| 197 | SM Rejected-Duplicate SM |
| 198 | TP-VPF not supported |
| 199 | TP-VP not supported |
| 208 | D0 SIM SMS storage full |
| 209 | No SMS storage capability in SIM |
| 210 | Error in MS |
| 211 | Memory Capacity Exceeded |
| 212 | SIM Application Toolkit Busy |
| 213 | SIM data download error |
| 255 | Unspecified error cause |

| | |
|---|---|
| 300 | ME failure |
| 301 | SMS service of ME reserved |
| 302 | Operation not allowed |
| 303 | Operation not supported |
| 304 | Invalid PDU mode parameter |
| 305 | Invalid text mode parameter |
| 310 | SIM not inserted |
| 311 | SIM PIN required |
| 312 | PH-SIM PIN required |
| 313 | SIM failure |
| 314 | SIM busy |
| 315 | SIM wrong |
| 316 | SIM PUK required |
| 317 | SIM PIN2 required |
| 318 | SIM PUK2 required |
| 320 | Memory failure |
| 321 | Invalid memory index |
| 322 | Memory full |
| 330 | SMSC address unknown |
| 331 | no network service |
| 332 | Network timeout |
| 340 | NO +CNMA ACK EXPECTED |
| 500 | Unknown error |
| 512 | User abort |

# Appendix - SMSC settings - SMS Service Center Addresses

**This page provides you with a list of Short Message Service Center (SMSC) addresses in different countries. You need this information for the configuration of a GSM modem connection or set up IP SMS connection.**

**AFGHANISTAN**
Afghan Wireless +9370290009
Roshan (Old) +9379900100
Roshan +93799900100

**ALBANIA**
AMC +3553820
AMC +355681000000
Vodafone +355692000200

**ALGERIA**
Nedjma +21350001701
AMN +21361000750
Djezzy GSM +21370048571

**ANDORRA**
STA-MOBILAND +376301004

**ANGOLA**
Unitel +244920010026

**ARGENTINA**
Movistar +54079000801
Personal +54079000803
MOVISTAR & Personal +541151740011
Personal +541151740050
Personal +541151740055
Personal +541151740056
CTI +54320000000
CTI+543200000000
CTI Mobile+543200000001
CTI+543200000059

**ARMENIA**
Beeline (Old) +3749000301
Beeline +37491000301
Vivacell +37493297333
Karabakh Telecom +37497200622

**AUSTRALIA**
OPTUS +61411990000
OPTUS +6141190001
OPTUS +61412025989
OPTUS +61411990003
TELSTRA AUSTRALIA +61418706700
TELSTRA AUSTRALIA +61418706800
TELSTRA AUSTRALIA +61418706900
VODAFONE +61415011501
ONE.TEL AUSTRALIA +61411990001
Three +61430004010

**AUSTRIA**
A1 +436640501
MAX.MOBIL +43676021
Connect ONE +436990001999
TELE.RING +4365009000000
Three +4366000660
A1 Mobilkom +436640561
A1 Mobilkom +4366477
T-Mobile +43676023
Tele2 (domestic) +436888590001
One GMBH +436990004999

**MONACO**
Valamobile +37744019000

**MONGOLIA**
MobiCom +97699000030

**MONTENEGRO**
Pro monte +38169200000
Monet +38167100100
T-Mobile +38267100100
Promonte +38269200000

**NAMIBIA**
MTC +264811900200

**NEPAL**
Mobilink +9779851028801
Mobilink +9779841208365
Spice & Mero (Old) +977980000900
Spice & Mero +9779800009000
NTC (Old) +977981028801

**NETHERLANDS ANTILLES**
Uniqa +59995199300

**NEW CALEDONIA**
Mobilis +687770009

**NEW ZELAND**
Vodafone New Zealand +64-21600600

**NICARAGUA**
Telgua pcs +5056090200
C&W (Cable & Wireless) +5058105137
Enitel +5058500130

**NIGER**
Telecel +227940000
Zain +22796960200

**NIGERIA**
Econet Wireless +234802000000
MTN Nigeria +234803000000
Nitel +234801000000
Mtn (Business) +2348030000420
GloWorld +2348050001501

**NORWAY**
NetCom Norway +47-92001000
NetCom Norway +47-9208977
TeleNor Norway +47-90002100
TeleNor Norway +47-90007777
Sense Mobil +47 90002100
NWN +4747919000
NWN +4747919009
Telenor +47900002100
Telenor +4790002198

**OMAN**
Oman Mobile (Old) +968300610
Oman Mobile (Old) +9689300610
Nawras & Gull +96895001016

Yesss +436990008999

### AZERBAIJAN
AZERCELL +994509103300
BAKCELLI +994557070707
NarMobile +994702000700

### BAHRAIN
+973-9600179
Batelco +97392
Batelco +97394
Batelco +97396
Batelco+973973
Batelco +97398
Batelco +97339600179
Zain +97336135135

### BANGLADESH
GRAMEENPHONE +88017099999
GRAMEENPHONE +88017500569
GRAMEENPHONE +880170000600
Warid Tel +8801600006001
GrameenPhone +8801700000600
Aktel +8801801000004
Aktel +88018010004
Banglalink +88019900557
Iridium +881662900005
Thuraya +882161900000

### BELARUS (BELORUSSIA)
Velcom +375296009000
MTC (MTS) +375297770000

### BELGIUM
Mobistar Belgium +32-95955205
Mobistar Belgium +32495002530
Proximus Belgium +32 475 161616
Orange Belgium +32-486000005
Proximus +32475161612
Proximus +32475161622
SMobistar +32495955205
Proximus (old) +3275161612
Proximus (Old) +3275161616
(Old) +3275161617

### BENIN
MTN +22997976903

### BHUTAN
Bhutan Mobile (BMobile) +97517009997

### BOLIVIA
Telef & Tigo +59177830007

### BOSNIA AND HERZEGOVINA
PTT GSM BIH +38766125522
BH Mobile +38761125522
BH Telecom (BA) +38761768085
BH Telecom (BA) +38761768087
Eronet +387633051
Mtel (Old) +3876500009
MTEL +38765500009

### BOTSWANA
MASCOM WIRELESS +26771010024
VISTA CELLULAR +26772000003
Orange +26772820133

### BRAZIL
Plus GSM +550112102010
BRT GSM +550160000060
Oi +550310000010
Nextel +551178313930

Oman Mobile +96899300610

### PAKISTAN
Mobilink +92300000042
UFONE +923330005150
Mobilink (Old) +9230000042
Paktel +923040000011
warid +9231600006001
Warid +9232100006001
Warid +923210006001
Aur +923330005250
Ufone (Old) +92333005150
SCOM +923358000000
Telenor +923455000010

### PANAMA
Zona Movil +50766105136
C&W (Cable & Wireless) +5076999904

### PAPUA NEW GUINEA
Pacific Mobile (BMobile) +6756818801

### PARAGUAY
Vox De +595961233
Tigo +595981499003
Porthable +595991799504

### PERU
Tim Peru +51-1-7990000
Movistar & Telefonica +51-1-95599000
Claro +51-1-97990000

### PHILIPPINES
Globe Philippines +63-91702
Globe Philippines +63-91703
Globe Philippines +63-91704
Globe Philippines +63-91709
Islacom Philippines +63-9150200003
Smart Philippines +63-9180000101
Smart+63-910000101
Globe telecom +63-9170000017
Globe +63-9170000020
Globe +63-9170000118
Globe +63-9170000130
Globe +63-91708
Smart (Gold) +63-9180000115
Smart +63-9198961000
Sun +63-9220001501

### POLAND
Era GSM Poland +48-602951111
Era GSM Poland +48-602951112
Polkomtel Poland +48-601000310
Polkomtel Poland +48-601000311
IDEA Centertel Poland +48-501200777
IDEA Centertel & Orange +48-501200777
Plus GSM & Polkomtel +48-601000351
P4 (Play)+48-790998250

### PORTUGAL
Telecel is +351-911616161
TMN Portugal +351-936210000
OPTIMUS +35193121314
+351-911350460 Vodafone
Vodafone +351-911350610
Vodafone +351-911616162
Vodafone +351-911616163
Telcel +351-931770077
UZO (TMN)+351-962100000
TMN +351-962100005
TMN +351-962100012

### QATAR

TIM +551181136200
TIM & Vodafone +551181138200
TIM +551181138310
TIM +5511831382000
Claro +551188015300
Claro +551991015300
Claro +552191105300
Claro +552194995300
Claro +552194995350
TIM +553191938200
TIM +554199138200
Brasil Telecom +55510160000060
TIM +555181136200
TIM +556181138200
TIM +558199238200
TIM +559181119200

**BRUNEI DARUSSALAM**
DST (Old) +6738795000
Jabatan Telekom & BMobile +6738101010
DST +67387950000

**BULGARIA**
MOBILTEL +35988000301
BTC & Vivatel +359878000333
Globul +35989100000
Cosmo +35989100001

**BURKINA FASO**
Zain +22676600150

**CAMBODIA**
StarCell +855098009900
Camshin & Qbmore +85511001555
Cellcard +85512000024
IQ2 Mobile +85516800000

**CAMEROON**
MTN +23779000002

**CHAD**
Zain +2356000999

**CHILE**
ENTEL PCS +5698890005
Movistar +5691600130
TELEFONICA MOVIL DE CHILE S.A.
+5691600131
NN +5691600132
Movistar +5691600160
Smartcompcs S.A. & Entel PCS
+5692099000

**CHINA**
CHINA TELECOM +861390591500
CHINA TELECOM +8613800100500
Unicom (Beijing) +8613010112500
Unicom (tianjin) +8613010130500
China Unicom GSM +8613010150500
Unicom (Shandong) +8613010171500
Unicom (Hebei) +8613010180500
Unicom (Guangdong) +8613010200500
Unicom (Liaoning) +8613010240500
Unicom (Anhui) +8613010305500
Unicom (Shanghai) +8613010314500
Unicom (Jiangsu) +8613010341500
China Unicom GSM +8613010350500
Unicom (Zhejiang) +8613010360500
Unicom (Fujian & guizhou) +8613010380500
Unicom (Hainan) +8613010501500
Unicom (Guangxi) +8613010591500
Unicom (Shanxi) +8613010701500
Unicom (Hubei) +8613010710500

QATARNET +974 5589955

**REUNION**
Reunion +33609001390

**ROMANIA**
CONNEX ROMANIA: +40722004000
ORANGE ROMANIA: +40744946000
Vodafone +40722004400
Vodafone +40722006000
Cosmote (Cosmorom) +40766000510
Connex GSM & Mobifon SA +4092004000
Connex GSM +4092006008
Connex GSM & Mobifon SA +4092006057
Connex GSM +4092006064
CONNEX GSM +4092006083
Dialogue +4094574574
Dialog +4094946000

**RUSSIA & KAZAKHSTAN**
MTS RUSSIA +70957699100
NORTH WEST RUSSIA +78129600096
BEELINE 1800 +790173100
Extel +70119349900
MTS +70957699101
MTS +70957699102
MTS (Moscow) +70957699800
K Cell (Kazakhstan) +73009300300
K Cell (Kazakhstan) +73009300301
Beeline & K-Mobile Kazakhstan
+73339077000
ITT (Ekaterinburg) & Smarts & Uraltel
+73434800248
UTEL & USI +73519030100
KCELL +77010002525
Beeline +77059077000
HCC & NCC (Nizhny Novgorod)
+78319090000
NCC & HCC +78319090909
North Caucasian GSM +78652949494
Di Ex +79022509900
Tele2 +79022859999
SMARTS UFA & ZAO SMARTS
+79023100101
Smarts +79023708080
Smarts +79023708393
Taif Telekom +7902390000
Kuban GSM (MTS-Kuban) +79024330000
Baykalvestkom +79025110010
VNTC +79025570055
VNTC +79025570077
MTS (Smolensk) +79027899999
YCC +79028710010
SCS900 +79029869990
ITT (Omsk) +79029889991
ETK +79029910000
Beeline +79033619502
Beeline +79037011110
Beeline & VimpleCOMR +79037011111
VimpelComR +79037011120
Tele2 +79042700003
Tele2 +79043090000
Tele2 +79043290000
Tele2 +79043490000
Tele2 +79045290000
Tele2 +79045390000
MTC +79104999104
ITT (Kostroma) +79106609999
ITT (Rekom) +79107459999
MTC +79107899999
ITT (St. Petersburg) +79112009993
MTS +79114009993
Peter +79115509993

Unicom (Jiangxi) +8613010720500
Unicom (Hunan) +8613010731500
Unicom (Henan) +8613010761500
Unicom (Qinghai) +8613010776500
China Unicom GSM +8613010788500
Unicom (Ningxia) +8613010796500
Unicom (Sichuan) +8613010811500
Unicom (chongqing) +8613010831500
Unicom (yunnan) +8613010868500
Unicom (gansu) +8613010879500
Unicom (shenzhen) +8613010888500
Unicom (jilin) +8613010911500
Unicom +8613010940500
Unicom (Inner Mongolia) +8613010350500
Unicom (Xinjiang) +8613010969500
Unicom (Heilongjiang) +8613010980500
Unicom +861301101901
China Mobile +8613800200500
China Mobile +8613800210500
China Mobile +8613800220500
China Mobile +8613800250500
China Mobile +8613800280500
Hangzhou +8613800571500
CM Mobile +8613800752500
CM Mobile (Shenzhen) +8613800755500
CM Mobile +8613800756500
CM Mobile +8613800769500

### COLOMBIA

Tigo +573003690025
Comcel +573103150030
Movistar +573160001021

### COMOROS

COMORES TELECOM +2693300030

### CONGO

Zain (Brazzaville) +2425899900
Zain (Point-Noire) +2425999900
Mtn +2426660016

### COSTA RICA

I.C.E. +5063005007
I.C.E. +5063005100

### CROATIA

VIPNET +385910401
CRONET +385980501
Tele2 +385951000100
T-Mobile HR +385980504

### CYPRUS

CYPRUS +3579700000
Areeba +35796961001
Cyta Mobile +35799700000

### CZECH

Eurotel Czech rep +420-602909909
Radiomobil Czech rep +420-603051
Daniela Slusova +420 603 052 000
Oskar +420296363777
Oskar/Vodafone +420608005681

### DEMOCRATIC REPUBLIC OF THE CONGO

Vodacom +243811030
Celtel & Zain +2439961234

### DENMARK

Sonofon Denmark +45-40590000
TeleDanmark Denmark +45-40390999
Telia DK Denmark +45-28187000
Mobilix Denmark +45-26265151

ITT (Novgorod) +79116009993
ITT (Ekaterinburg) +79126313456
MTS (Perm) +79128800003
MTS (Chelyyabinsk & Magnitogorsk) +79128900003
MTS (Tyumen & Sergut & Nizhnevartovsk) +79129200003
ITT (Atlanta) +79135330003
MTC (Atlanta) +79139869990
ITT (Irkutsk) +79147991000
MTS +79168960220
MTS (Moscow) & ITT +79168999100
MTS +79168999101
MTS +79168999102
MTS (Moscow) +79168999800
ITT (Samara) +79171002003
BMTelekom (MTS-Bashkortostan) +79173400670
Kazan +79173911800
Kuban +79184330000
Smolensk +79206909090
Megafon & Mtn (North West)+79219909090
Megafon+79222909090
Megafon (Moscow)+79262909080
Megaphone (Megafon Moscow) +79262909090
Megaphone (Caucasus) & Mobikom +79282000002
Mobikom +79289900028
Tele2 +79508090000
Tele2 +79508404040
Global Star +79549903444

### RWANDA

+25008110333

### SAUDI ARABIA

Sauditel +966 5 0503 1999
Sauditel +966 5 0503 2999
AL JAWAL +96655053999
Al Jawal & Saudi Telecom Company +96655075999
Al Jawal +96655096999
SaudiTel +96655842855
Zain +966590100880

### SENEGAL

Tigo +221766600990
Orange +221776380010

### SERBIA

MobTel +38163100100
MobTel +38163100200
MobTel +38163100300
MobTel +38163100400
MobTel +38163100300
PTT Telekom Serbia +381-650000900
PTT Telekom Serbia +381-640000900
VIP Mobile +381610401
T-Mobile Cg & Monet +38167100100
Promonte (Old) +38169200000

### SEYCHELLES

Airtel: +248700000
C&W (Cable & Wireless) +248510000
Airtel +248700001

### SIERRA LEONE

Tigo +23230100996
Zain +23276010101
Africell +23277180000

### SINGAPORE

Free! +45 40590006
Tele2 +4525700000
Three +4531300000
TDC +4543625250
Bite (GSM) +4560990999

**ECUADOR**
Movistar +59395897705
Porta Conecel +59397995040

**EGYPT**
CLICK GSM +20105996500
MOBINIL +20122000020

**EL SALVADOR**
Digicel +50377601515

**ESTONIA**
EMT Estonia +372-5099000
Ritabell Estonia +372-5509911
Radiolinja Estonia +372-568771010
EMT +3725099001

**FAROE ISLANDS**
Faroese Telecom +298249020

**FIJI**
Vodafone +679901400

**FINLAND**
Radiolinja Finland +358-508771010
Sonera Finland +358-405202000
Sonera +358405202002
Sonera +358405202006
Tele +358405202330
Telia +358410400
DNA (Aina) +358447983500
Saunalahti +358451100100
Alands Mobiltelefon +3584570001040

**FRANCE**
SFR France +33-609001390
Bouygues France +33-660003000
Itineris France +33-689004000
Itineris France +33-689004431
Itineris France +33-689004581
Tele2 +33-607281155
(Tambahan) +33-609003347

**GABON**
Telecel Gabon +241-308009
Zain +24107910101

**GAMBIA**
Africell +220780001

**GEORGIA**
Geocell +995779104400
Beeline +99597290589
Magticom +99599599995

**GERMANY**
D1 Germany +49-1710760000
D1 Germany +49-1715990000
D1 Talkline Germany +49-1710760900
D2 Germany +49-172227033
D2 Germany +49-1722270000 [to D2 only]
D2 Germany +49-1722270333
D2 Talkline Germany +49-1722270258
Debitel Germany +49-1722270222
IC3S Germany +49-1722270201
Dr Materna Germany +49-1722270111
E2 VIAG Germany +49-1760000443

Mobile One Singapore +65-96845999
Mobile One Singapore +65-96845997
SingTel Singapore +65-96400001
SingTel Singapore +65-96500001
SingTel Singapore +65-98189999
SingTel Singapore +65-96197777
Starhub +6598540020
Mobile One +652431389

**SLOVAKIA**
GLOBTEL SLOVAKIA GSM +421905303303
EUROTEL SLOVAKIA +421903333000

**SLOVENIA**
Mobitel Slovenia +386-41001333
Si.Mobil Slovenia +386-40441000
Vodafone +386-40441003
Vodafone +386-40441013
TusMobil +386-70007007

**SOMALIA**
Telecom Somaliland +25265370000

**SOUTH AFRICA**
MTN +27-831000002
MTN (Prepaid) +27-831000113
Vodacom +27-829119
Vodacom +27-829129
Virgin Mobile +27-741000050
CellC +27-841000000

**SPAIN**
Telefonica Spain +34-609090909
Airtel Spain +34-607133000
Airtel Spain +34-607003110
Amena Spain +34-656000311
TELEFONICA +3409090909
Movistar +34-609090401
TIM (Telecom Italia Mobile) +34-609090403
Movistar & Telefonica +34-609090806
Movistar +34-609090870
TELEFONICA +34-60909090
Movistar +34-609090999
Simo +34-644109030
Movistar +34-690525100

**SRI LANKA**
Celltell +94-72000004
Mobitel +94-7100003
Dialog GSM +94-77000003
Bell+94-115000001
Dialog +94-117000003
Suntel +94-334088884
Hutch +94-785000005

**SUDAN**
Zain +249912020000
Areeba +249921999999

**SURINAME**
Digicel +5978199004
Telesur +5978690009

**SWAZILAND**
Vodacom +2686011033

**SWEDEN**
Comviq +46-707990001
Comviq +46-707990002
Comviq +46-707990003
Comviq +46-707773078
Europolitan +46-708000708
Telia +46-705008999

E-Plus Germany +49-1770600000
E-Plus Germany +49-1770610000
E-Plus Germany +49-1770620000
Solomon +4915700100100
Cellway D1 & Mobilcom D1 +491710760315
Victorvox D1 +49-1710760322
Debitel D1 +49-1710760333
TalkLine D1 +49-1710760345
The Phone House D1 +49-1710760374
D1 +49-1710762222
D1 T-Mobile +49-1712092522
D1 T-Mobile Deutschland GMBH +49-1712521001
DR Materna +49-1722241265
Victorvox D2 +49-1722270104
D2 Vodafone GMBH +49-1722270210
D2 Vodafone +49-1722270229
D2 Vodafone +49-1722270230
DR Materna +49-1722270265
D2 +49-1722270300
D2 Vodafone GMBH +49-1722270581
D2 Vodafone +49-1722270685
D2 Vodafone GMBH +49-1722270811
Vodafone +49-1722270827
Cellway D2 +49-1722270880
DR Materna +49-1722271222
D2 Vodafone +49-1722277000
D2 +49-1722278000
D2 +49-1722278010
D2 +49-1722278025
O2 Loop & Telefonica +49-1760000433
o2 +49-1760000442
Debitel +49-1760000461
E-Plus +49-1770600010
E-plus +49-177062000
E-Plus Mobilfunk +49-1770620007
E-plus +49-1770620014
O2 & GMBH & Co. OHG +49-1793959507

**GHANA**
Ghana Telecom +233200000007
Ghana Telecom +233200000010
Scancom & MTN +233244500000
Irancell (MT SMSC) +233244550190000
Tigo+233277500800

**GIBRALTAR**
Gibtelecom +35058000003

**GREECE**
Panafon Greece +30-94219000
Telestet Greece +30-93599000
Telestet Greece +30-93597000
Cosmote Greece +30-97100000
WIND (Prepaid) +30-693597000
Telestet +30-693599000
Vodafone +30-6942190000
Q Telecom +30-6990000010
Panafon +30-942940000
Cosmote +30-97100003
Cosmote +30-97100005

**GREENLAND**
SMSC +299500005
Tele +299551495

**GUADELOUPE**
Orange +590350012

**GUINEA**
Intercel +22463000996

**HOLLAND**

Tele2 +46-70799007
Tele2 +46-707990076
Tele2 +46-707990081
Tele2 (While Roaming) +46-707990086
Three +46-735480000

**SWITZERLAND**
Swisscom Switzerland +41-794999000
diAX Switzerland +41-765980000
Orange +41-787777070
Cablecom +41-435400000
Swisscom +41-740900103
Swisscom +41-740900108
SUNRISE & TDC +41-765980017
Tele2 +41-772099999
Swisscom Mobile LTD +41-794990900
Swisscom Mobile LTD +41-794991300
Viag Swisscom Mobile LTD +41-794998123
Swisscom +41-794998990
Swisscom +41-794999001
Swisscom Mobile LTD +41-799799888
Swisscom +41-900900941

**SYRIA**
Syriatel +96393111193
Syriatel (Sritil Old) +9639311193
MTN (Old) +9639444444
MTN +96394944444

**TAIWAN**
Chung Wa Tele Taiwan +886-932400821
(Old) +88691744001
Far EasTone +886-931000096
Far Eastone +886-931000099
Mobitai +886-931413131
Pan Asia Telecom (PAT) +886-931744010
Chungwa Telecom +886-932400841
Chungwa Telecom +886-932400851
Chungwa Telecom +886-932400881
Chungwa Telecom +886-932400882
TWN GSM +886-935074443
Taiwan Mobile +886-935874258
Catch +886-935874443
Far Eastone +886-936000097
Far Eastone +886-936000099
KGT +886-938348404
AndALetter +886-938749104

**TAJIKISTAN**
Babilon +992918602602
Beeline +992919038989
Somoncom & Lenta +992929999999

**TANZANIA**
Tritel +255812904000
Celtel +255748000004
Vodacom +255744114
Vodacom (Old) +25575114
Zantel +255774999100
Zain +255780000004

**THAILAND**
AIS +6618110888
WP1800 +6616110400
AIS +6613101800
Mobile Life +6618310808
AIS +66813101800
Smart Mobile +66816110400
AIS +66818110888
AIS +66818310808
TrueWorld +66891009120
Trueworld & Orange (Old) +6691009120
(Old) +6691009122

Libertel Netherlands +31-6540881000
PTT Netherlands +31-653131313
Telfort Netherlands +31-626000230
BEN Netherlands: +31-624000000
Dutchtone, Netherlands: +31628500561
Tele2 +31640191919
KPN & Hi & Ortel mobile +31653131314
KPN Mobile +31653131315
PTT +31653141414

### HONDURAS

Clear +5043990100
Tigo +5049526911

### HONG KONG

Hutchinson Hong Kong +852-94985795
HK Telecom Hong Kong +852-90288000
Smartone Hong Kong +852-90100000
Mandarin Communication (aka Sunday) +852-92347999
Hong Kong +852-92040032
CM Mobile +852-62444402
Three +852-63353535
HKCSl +852-90200006
Hong Kong Telecom +852-90250109
CM Mobile +852-90257224
Peoples +852-92040031
New World +852-92141238
New World +852-92141239
PCCW 3G +852-92347949
Three & Orange +852-94989995

### HUNGARY

Telenor +36-209300099
Telekom +36-309888000
Vodafone +36-709996500

### ICELAND

Landssimi Islands hf +354-8900100
Tal +354-6999099

### INDIA

AT&T Network of India
BPL Mobile India +91-9821000005
Hutch +91-98840-05444
MAXtouch India +91-9820005446
Essar Cellphone +91-9811009998
Tata +919848001104
Command +919830099990
AirTel Delhi +91-98100-51914
Skycell Cellular +919840011003
Spice Karnataka +919844198441
Spicell +919831029222
Usha Martin Telekom ltd +919830099990
Reliance +914031893330
BSNL & CellOne +919417099997
BSNL & CellOne +919422099997
Calcutta Telephones & BSNL & CellOne +919434099997
BSNL & Excell (Guwahati) +919434599997
BSNL (Old) & Cell ONE +919440099997
BSNL (New)+919442099997
Hutch +919732099990
Hutch +919801005444
Hutch +919805005444
Escotel (Haryana) +919812099995
Hutch +919813005444
Spice +919814047105
Hutch +919815005444
Airtel & Spice +919815051914
Airtel +919816051914
Orange & Hutch (Mumbai) +919820005444
Maxtouch +919820205446

Thai Mobile +6693200101

### TOGO

Togocel +228040009

### TONGA

TonFon +6768900200

### TUNISIA

Tunisiana +21622000022
Tunisie Telecom +21698390003

### TURKEY

Turkcell Turkey +90-5329010000
Turkcell Turkey +90-5329020000
Turkcell Turkey +905329030000
Turkcell Turkey +90-5329040000
Telsim Turkey +90-5429800033
Turkcell +905329010004
Turkcell +90-5329010008
Turkcell Iletisim Hizmetleri A.S. +90-5329020006
Turkcell Iletisim Hizmetleri A.S. +90-5329110006
Turkcell Iletisim Hizmetleri A.S. +90-5329110018
Turkcell Iletisim Hizmetleri A.S. +90-5329200004
Turkcell Iletisim Hizmetleri A.S. +90-5329210007
Turkcell Iletisim Hizmetleri A.S. +90-5329300004
Turkcell Iletisim Hizmetleri A.S. +90-5329500018
Turkcell Iletisim Hizmetleri A.S. +90-5329510020
Turkcell +90-5329530001
Turkcell Iletisim Hizmetleri A.S. +90-5329550013
Turkcell Iletisim Hizmetleri A.S. +90-5329580001
Turkcell Iletisim Hizmetleri A.S. +90-5329580014
turkcell +90-5329580016
Turkcell Iletisim Hizmetleri A.S. +90-5329590002
Turkcell +90-5329590011
Turkcell Iletisim Hizmetleri A.S. +90-5329590012
Turkcell Iletisim Hizmetleri A.S. +90-5329590020
KKT Cell +90-5338783000
(Hazir Kart) +90-5359010000
Avea+90-5598008000

### TURKMENISTAN

MTS +99366322222

### UGANDA

Waridtel +256700000088
Uganda Telecom & TeleChoice +25671000050
Zain +25675010004
MTN +25677110005

### UK

Isle of Man Pronto GSM +447624499955
Vodafone UK +44-7785016005
CellNet UK +44-7802000332
Virgin Mobile UK +447958879890
Orange UK +44-7973100973
Orange UK internal +44-7973100974
One2One UK +44-7958879879

Orange & Hutchison Max Telecom +919820405444
BPL +919821100005
Idea (Maharashtra) +919822078000
Idea (Maharashtra) +919822178000
BPL +919823000040
Idea +919824078000
Celforce & Hutch (Gujarat) +919825001002
Idea (chattisgarh) +919826012311
Oasis (Rajasthan) +919829003333
Hutch +919830005444
(kolkata) +919831029007
Airtel +919831051914
Escotel (Up West) +919837099995
RPG +919841044446
Aircel +919842201155
BPL (Tamil Nadu) +919843000040
Airtel (Karnataka) +919845087001
Hutch & BPL (Kerala) & Vodafone +919846000040
Idea & Escotel (Kerala)+919847099996
Airtel +919849051914
Airtel (Andhra Pradesh) +919849087001
Aircel (Guwahati) +919854099060
Hutch +919860005444
Reliance (Guwahati) +919864002222
Dolphin +919868099994
MTNL & Dolphin +919869099994
Hutch AP +919885005444
Hutch +919886005444
Vodafone +919888009998
Idea +919889199996
Airtel +919890051914
Idea +919891030039
Idea (Delhi) +919891030099
Airtel +919892051914
Airtel +919893051914
Airtel +919894051914
Airtel (Kerala) +919895051914
Airtel +919896051914
Airtel +919897051914
Airtel +919898051914
(Patna) +919934029007
Airtel (Guwahati) +919954029007

**INDONESIA**
Satelindo Indonesia +62-81615
Satelindo : +62 816 124
Satelindo : +62 816 125
Satelindo : +62 816 126
Satelindo : +62 816 127
Satelindo : +62 816 128
Telkomsel Indonesia is +62-81100000
Exelcomindo Indonesia +62-818445009
Telkomsel +62811000000
Lippo Telecom +628315000031
Axis +628315000032
Indosat +62855000000
Three +6289644000001

**IRAN**
Taliya +989320900004
TKC +989347691001
Irancell +989350001400
Irancell +989350001420
Irancell +989350001450
Irancell +989350002400

**IRAQ**
Korektel +964750001140
Mobitel +9647605912010

**IRELAND**

Guernsey Telecoms [+44-4481/7781]
Jersey +44-7781
Vodafone (Old) +44-385016005
Ekit +44-7624499904
Ekit travel SIM +44-7624499970
Global sim +44-7624499977
Hutchison 3G +44-7723564240
Hutchison 3G +44-7723564241
Hutchison 3G +44-7723564342
Hutchison 3G +44-7723564343
Hutchison 3G +44-7723564344
Cable & Wireless (Guernsey) +44-7781001000
Cable & Wireless (Guernsey) +44-7781001001
Cable & Wireless (Guernsey) +44-7781001003
Cable & Wireless (Guernsey) +44-7781001004
Cable & Wireless (Guernsey) +44-7781001008
Cable & Wireless (Guernsey) +44-7781001010
Cable & Wireless (Guernsey) +44-7781001013
Beemme (Cable and Wireless Guernsey) +44-7781484668
Hutchison 3G +44-7782000800
Hutchison 3G +44-7782000801
Hutchison 3G +44-7782000802
Vodafone +44-7785011300
Vodafone+44-7785011301
Vodafone +44-7785011302
Vodafone +44-7785011303
Vodafone +44-7785011304
Vodafone +44-7785011928
Vodafone +44-7785012520
Vodafone +44-7785012891
Vodafone +44-7785012990
Vodafone +44-7785013977
Vodafone +44-7785013978
Vodafone +44-7785013979
Vodafone +44-7785013987
Vodafone +44-7785013988
Vodafone +44-7785013993
Vodafone +44-7785013998
Vodafone +44-7785014208
Vodafone +44-7785014306
Vodafone +44-77850143090
Vodafone +44-7785014310
Vodafone +44-77850143100
Asda Mobile +44-7785014315
Lebara Mobile +44-7785014317
Talk Mobile Travel SIM +44-7785014318
Vodafone +44-77850143100
Vodafone +44-7785499993
Vodafone +44-7785499999
Vodafone +44-7785600100
Jersey Telecom +44-7797700000
Jersey Telecom +44-7797701000
Jersey Telecom +44-7797704000
Jersey Telecom +44-7797704001
Jersey Telecom +44-7797704002
Jersey Telecom +44-7797704003
United Mobile +44-7797704004
Jersey Telecom +44-7797704005
Jersey Telecom +44-7797704006
Jersey Telecom +44-7797704041
Jersey Telecom +44-7797704043
Jersey Telecom +44-7797704444
Jersey Telecom +44-7797706000
Jersey Telecom +44-7797706003
Jersey Telecom +44-7797706004

Eircell Ireland +35387699989
Three +353830000800
Meteor +353857000000
Vodafone +35387699959
Eircell+35387699985
Esat Ireland +353868002000

### ISLE OF MAN

Pronto GSM +447624499955

### ISRAEL

ORANGE ISRAEL +97254120032
Jawwal +972059141999
CELL COM+972521100059
Orange +972544120032
Jawwal (Old) +97259141999

### ITALY

Omni Italy +39-3492000200
Omni Italy +39-3492000300
Omni Italy +39-3492000400
Omni Italy +39-3492000500
TIM Italy +39-3359609600
TIM Italy +39-3359608000
TIM Italy +39-338960960
TIM Italy +39-338980000
Wind Italy +39-3205858500
Three +3916263333
Wind +39-3205952500
Wind +39-3505956500
Wind +39-3205959100
Wind +39-3205959300
Wind +39-3205959500
Tim +39-3358831200
Coop Voce +39-3359602000
MTV Mobile +39-3359604000
Vodafone +39-3492000256
Vodafone Omnitel N.V. +39-3492000586
Vodafone +39-3492001155
Vodafone +39-3492001156
Vodafone +39-3492001157
Vodafone +39-3492001158
Vodafone +39-3492001159
Vodafone +39-3492001311
Tiscali +39-3709999130
Uno Mobile +39-3770001004
BT Mobile +39-3770001005
Poste Mobile +39-3770001006
Conad (Insim) +39-3770001009
DTM +39-3770001010
Blu Spa +39-3801000100
Three +39-3916263008
Three +39-3916263019
Three +39-3916263333
Fastweb Mobile +39-3916263900
Fastweb Mobile +39-3916263901
Digitel +39-3916263921

### IVORY COAST

MTN +22505889999
Orange +22507070002

### JORDAN

Zain +962745490001
Orange +96277700016
Umniah +96278899088
Zain & Fastlink +96279000023

### KENYA

Safaricom +25472500010
Kencell +254733000810
Safaricom +254722500029
Safaricom (Old) +25472500029

Jersey Telecom +44-7797706007
Jersey Telecom +44-7797706008
Jersey Telecom +44-7797706009
Jersey Telecom +44-7797706010
Jersey Telecom +44-7797706011
Jersey Telecom +44-7797706017
Jersey Telecom +44-7797706018
Jersey Telecom +44-7797706020
Jersey Telecom +44-7797706022
Jersey Telecom +44-7797706024
Jersey Telecom +44-7797706025
Jersey Telecom +44-7797706027
Jersey Telecom +44-7797706029
Jersey Telecom +44-7797706031
Jersey Telecom +44-7797706032
Jersey Telecom +44-7797706033
NOMI MOBILE +44-7797706037
Jersey Telecom +44-7797706041
Jersey Telecom +44-7797706043
Jersey Telecom +44-7797706071
Jersey Telecom +44-7797706072
Jersey Telecom +44-7797706074
Jersey Telecom +44-7797707002
Jersey Telecom +44-7797707070
Jersey Telecom +44-7797707071
Jersey Telecom +44-7797891005
Jersey Telecom +44-7797895210
Jersey Telecom +44-7797895511
O2 +44-7802000334
O2 +44-7802000341
O2 +44-7802000531
O2 +44-7802000532
O2 +44-7802000720
O2 +44-7802000724
O2 +44-7802005701
o2 (PAYG) +44-7802092035
Airtel+Vodafone (Jersey) +44-7829791004
Blyk Mobile +44-7870002108
Lyca Mobile +44-7870002208
Vodafone +44-7909582529
Wave Telecom +44-7911704000
T-Mobile +44-7958879883
Orange +447973074999
Orange +447973100979
O2 (Old) (prepaid) +44802000332
o2 (Old) (contract) +44802000334
o2 (Old) +44802000335
o2 (Old) +44802000341
Orange (Old) +44973100973
Orange (Old) +44973100974

### UKRAINE

UMC +38050000501
Kyivstar +380672021111
Golden Telecom +380444990000
(number@sms.gt.kiev.ua)
Wellcome +38044 251 7777
GOLDEN TELECOM +380390001008
Golden Telecom (UNI) +380390001009
Golden Telecom +3804444990007
Golden Telecom +380444939907
GOLDEN TELECOM +380444939908
UMC +38050000502
Life & Kyivstar +380639010000
KYIVSTAR +380672020000
Kyivstar GSM JSC +38067202010
Beeline +380682953333
Beeline +380683201111
WellCOM & Ukrainian Radio Systems
+380683211111
Beeline +380689008708
Utel +380910440601

Orange +254770000040

**KUWAIT**
MTC +96596000303
Al-Wataniya +9656373717
Wataniya +96566373717

**KYRGYZ REPUBLIC (KIRGHIZIA)**
Bitel +996502588800
Megacom +996555500005

**LAOS**
Tigo +85620700024

**LATVIA**
LMT +3719202020
Baltcom GSM +371 9599994
Tele2 +37125850115
Amigo +37129202020
Tele2 +37129599994

**LEBANON**
FTML Cellis +961-3488888
MTC Touch +9613996060

**LESOTHO**
Vodacom Lesotho +26655820088

**LIBYA**
Al Madar +218919190120
Libyana +218929290120

**LIECHTENSTEIN**
United Mobile +42379010599

**LITHUANIA**
Bite GSM Lithuania +370-9950115
Omnitel Lithuania +370-9899992
Tele2 +37068499199
Bite GSM +3706992323
Omnitel +37069899993

**LUXEMBOURG**
PTT Luxembourg +352-021100003
Tango Luxembourg +352-091000030
VOX +352061000060

**MACAU**
Macau CTM +85366
Macau CTM +85368
HT +8536344503
CTM +8536200855
TELEMOVEL & CTM +8536800000
TELEMOVEL +8536800855

**MACEDONIA**
MobiMak [389 - 70]
T-Mobile +389 70 000501
Mobimak +389706622
Cosmofon +38975000200
Vip +38977000301

**MADAGASCAR**
Orange +261323232707
Zain +261331110006

**MALAWI**
TNM +2658820767
TNM +2658820777
Zain +2659900140

**MALAYSIA**
ADAM Malaysia +60-173600010

**UNITED ARAB EMIRATES**
PTT UAE +97150 6014994
Etisalat +971506060000
DU +971555515515

**URUGUAY**
Movistar +59894000080
Claro +59896998001
Tigo & Ancel+59899998932

**UZBEKISTAN**
Mts +998711300897
Beeline +998901850488
INFOCOM +998901850499
Coscom +998930190000

**VENEZUELA**
Digitel +58-412-8000000
Digicel +58-417-1002000
Infonet +58-418-0000013
Movistar +584240000954

**VIETNAM**
Mobifone Vietnam +84 90400012
Vinaphone Vietnam +84 91020005
Vinaphone Vietnam +84 91020010
Viettel (North and Central region) +84900000012
Viettel (South) +84900000018
Viettel +84900000040
Viettel +8490200030
Viettel +84980200030

**YEMEN**
Sabafon +96771580000
MTN +96773100004

**YUGOSLAVIA**
MobTel +381-63-100400
MobTel +381-63100300
MobTel +381-63100200
MobTel +381-63100100
MobTel +3816310030034
MobTel +3816310040034
PTT Telekom Serbia +381-650000900

**ZAMBIA**
Mtn +260966060015
Zain +260971911200

**ZIMBABWE**
Econet Wireless +26391010030
Zimbabwe net*one +26311191201
Telecel +26323100007
Telecel (pvt) LTD +26323100009
Econet +26391010045

**USA & CANADA & JAMAICA & ST. KITTS AND NEVIS & TRINIDAD AND TOBAGO & DOMINICAN REPUBLIC& PUERTO RICO & ST. VINCENT AND THE GRENADINES & DOMINICA & ST. LUCIA & AMERICAN SAMOA & GUAM & NORTHERN MARIANAS & MONTSERRAT & TURKS AND CAICOS ISLANDS & GRENADA (INCL CARRIACOU) & BERMUDA & CAYMAN ISLANDS & VIRGIN ISLANDS (USA) & +1284 VIRGIN ISLANDS (UK) & ANTIQUA AND BARBUDA & ANGUILLA & BARBADOS & BAHAMAS**
Aerial Comms USA +1-8132630025
Aerial Comms USA +1-2812350025
Voicestream USA +1-2063130004
Pacific Bell USA +1-2099042010

Celcom Malaysia +60-193900000
Mutiara Malaysia +60-162999000
Maxis Malaysia +60-120000015
TMTOUCH +60-132400000
DIGI +60-162999902
Digi Malaysia+60-166782176
CELCOM GSM +60-193900020

**MALDIVES**
Dhiraagu DHI-Mobile +9607780000
(Old)+960780000

**MALI**
Orange +2236000000
Malitel +2236700050

**MALTA**
Vodafone +356-941816
goMobile +35679700003

**MAROCCO**
Maroc Telecom +212 61 00 00 21
Maroc Telecom +212 61 00 00 22
Maroc Telecom +212 61 00 00 23
Meditel +2123992000
Maroc Telecom (Old) +2121000021
Meditel +21263992000

**MAURITANIA**
Mauritel +2226400850

**MAURITIUS**
EMTEL +230 7290999
cellplus +230 2500005

**MAYOTTE & REUNION**
Orange +262692000606

**MAZAMBIQUE**
Mcel +25882200030

**MEXICO**
Telcel +524700001410
Nextel +5253309904
Optel +525512235100
MOVISTAR +525512235110
Tigo +525512235135
Telcel & Orange +5294100001410

**MOLDOVA (MOLDAVIA)**
EventIS +37365000090
Orange +37369101030
Moldcell +37379499011
Voxtel +3739101020
Voxtel +3739101030

Pacific Bell USA +1-2099042020
Pacific Bell +1-2099042030
Powertel USA +1-3343338200
DigiPH PCS USA +1-3342090307
Omnipoint USA +1-9179070004
Sprint USA +1-7044100000
Washington PCS USA +1-410258953
FIDO Canada +1-5149931123
T-mobile +1-2063130003
Unicel +1-2149320902
Centennial Wireless +1-2605300005
Cinglar (great lakes region) +1-3123149600
Cingular & o2wireless & AT&T +1-3123149810
Cinglar (great lakes region) +1-3123149860
Cinglar (great lakes region) +1-3123149870
Cinglar (great lakes region) +1-3123149880
Cellular One & Dobson +1-3305310005
Southern Linc +1-3348000011
Cinglar (northeast region) +1-4047259000
Cinglar (northeast region)+1-4047259015
Cinglar (northeast region) +1-4047259016
Cinglar (northeast region) +1-4047259060
Cinglar (northeast region) +1-4047259245
Cinglar (northeast region) +1-4047259246
Cinglar (northeast region) +1-4047259247
Washington PCS & Sprint PCS +1-4102589530
Einstein +1-4147139800
Mobility +1-4413360930
Telecom Bermuda +1-4415907672
Cincinnati Bell +1-5132400140
Nextel & Telus +1-5148210053
Suncom +1-5405559150
Rogers +1-6044183471
Epic PCS +1-6205755996
Cinglar (central region) +1-6363848801
Cinglar (central region) +1-6363848815
Cinglar (central region) +1-6363848845
Cinglar (central region) +1-6363848870
Iconnect +1-6718886809
Cingular (SouthEast Coast) +1-7045020600
Cingular (southeast region) +1-7045020700
Cingular (southeast region) +1-7045020800
Rogers Wireless (Canada) & Speakout +1-7057969300
Xit Communications +1-8065270011
Orange +1-8098599990
Orange +1-8098599999
Aerial CDT +1-8132630026
Digicel (Incl C&W Bahamas) +1-8763800154
Cable & Wireless Caymen Islands & St Lucia & St Kitts and Nevis +1-8768141109
Alaska Wireless +1-9073590153
Cingular & AT&T +1-9078319301
(old AT&T) +1-9703769301
Nortel +1-9703769316
Nortel +1-9703769317
Nortel +1-9703769318
Nortel +1-9703769319

# GSM Operator and Country Codes for Operator Logos

***The following country and operator codes can be used when an operator logo is sent to a GSM device.***

| | |
|---|---|
| 202-01 | COSMOTE - Greece |
| 202-05 | Greece Panafon S A Panafon |
| 202-10 | Greece STET Hellas STET Hellas |
| 204-04 | Vodafone NL (Libertel) (GSM 900) |
| 204-08 | KPN Telecom (GSM 900) |
| 204-12 | Telfort (GSM 1800) |
| 204-16 | Ben (GSM 1800) |
| 204-20 | Dutchtone (GSM 1800) |
| 206-01 | Belgium Belgacom BEL PROXI |
| 206-10 | Belgium Mobistar BEL mobistar |
| 206-20 | KPN Orange Belgium SA |
| 208-01 | France France Telecom F Itineris |
| 208-01 | Monaco France Telecom F Itineris |
| 208-10 | France SFR F SFR |
| 208-10 | Monaco SFR F SFR |
| 208- | France Bouygues Telecom Bouygues |
| 213-03 | Andorra STA Andorra AND M-AND |
| 214-01 | Spain Airtel Airtel |
| 214-03 | Spain AMENA |
| 214-07 | Spain Telefonica Moviles MSTAR |
| 216-01 | Hungary Telenor |
| 216-30 | Hungary Magyar Telekom |
| 216-70 | Hungary Vodafone (GSM 900/1800) |
| 218-01 | Bosnia Cronet Cronet |
| 218-90 | Bosnia PTT Bosnia BIH PTT-GSM |
| 219-01 | Croatia HPT Croatian PTT HR-CRONET |
| 219-10 | Croatia VIPNET |
| 220-01 | Serbia Mobile Telecomm. MOBTEL |
| 220-03 | YU PTT |
| 220-02 | Montenegro PROMONTE GSM |
| 220-04 | Montenegro MoNet GSM |
| 220-07 | Serbia Promonte PROMONTE GSM |
| 222-01 | Italy Telecom Italia Mobile TIM |
| 222-01 | San Marino Telecom Italia Mobile TIM |
| 222-01 | The Vatican Telecom Italia Mobile TIM |
| 222-10 | Italy Omnitel Pronto Italia Omni |
| 222-10 | San Marino Omnitel Pronto Italia Omni |
| 222-10 | The Vatican Omnitel Pronto Italia Omni |
| 222-88 | Wind Telecomunicazioni S.P.A, Vatican and San Marino, Italy |
| 222-98 | Blu s.p.a. Telefonia mobile Italy |
| 226-01 | Romania MobiFon |
| 226-03 | Romania Cosmorom |
| 226-10 | Romania MobilRom |
| 228-01 | Switzerland Swiss PTT CH NAT D |
| 228-01 | Liechtenstein Swiss PTT CH NAT D |
| 228-03 | Orange Switzerland |
| 228-08 | Tele2 Switzerland |

| | |
|---|---|
| 230-01 | Czech republic Radio Mobil CZ Paegas |
| 230-02 | Czech republic Eurotel Praha EUROTEL-CZ |
| 230-03 | Czech republic OSKAR |
| 231-01 | Slovakia Globtel |
| 231-02 | Slovakia Eurotel Bratislava Eurotel Bratislava |
| 232-01 | Austria Mobilkom A1 |
| 232- | Austria O Call TS |
| 232-03 | Austria Max Mobil Telekoms A max. |
| 232-05 | Connect Austria Gesellschaft fur Telekommunikation GmbH - ONE |
| 232-07 | tele.ring Telekom Service GmbH - Telering |
| 234-10 | UK Cellnet Cellnet |
| 234-15 | UK Vodafone Vodafone |
| 234-30 | UK One2One/Virgine |
| 234-33 | UK Orange |
| 234-90 | Jersey Jersey Telecoms Jer 1 |
| 234-55 | Guersney Guersney Telecom GSY-TEL |
| 234-58 | Isle of Man Manx Telecom Manx |
| 235- | UK |
| 238-01 | Denmark Tele Danmark Mobil DK TDK-MOBIL |
| 238-02 | Denmark SONOFON |
| 238- | Denmark Telia Denmark Telia DK |
| 238- | Denmark Mobilix Mobilix |
| 240-01 | Sweden Telia Mobitel S TELIA MOBITEL |
| 240-07 | Sweden Comviq S COMVIQ |
| 240-08 | Sweden Europolitan Europolitan |
| 240- | Sweden Telenordia |
| 242-01 | Norway Telenor Mobil N Tele-mobil |
| 242-02 | Norway NetCom GSM A/S N NetCom GSM |
| 244-05 | Finland Telecom Finland Telecom Finland |
| 244-91 | Finland OY Radiolinja Radiolinja |
| 244- | Finland Telivo |
| 244- | Aland Alands Mobil |
| 246-01 | Lithuania Omnitel LT OMNITEL |
| 246-02 | Lithuania Mobilios Telekomunikacijos LT BITE |
| 246-03 | Lithuania LT TELE2 |
| 246-04 | ir Lithuanian X-GSM Tele2 |
| 247-01 | Latvia Latvias Mobilais Telefons LV LMT GSM |
| 247-02 | Latvia Baltcom LV BALTCOM |
| 248-01 | Estonia Eesti Mobiiltelefon EE EMT GSM |
| 248-02 | Estonia Radiolinja Eesti AS RLE-GSM |
| 248-03 | Estonia Ritabell Q-GSM |
| 290-01 | Russia Mobil Telesystems MTS-RUS |
| 290-02 | Russia NW GSM St Petersburg RUS NWGSM |
| 290- | Russia DonTelecom |
| 290- | Russia United Tele Moscow UTM |
| 290-38 | Russia Wireless Technology |
| 290- | Russia Extel Mobile Comms Systems |
| 290-99 | Russia KB Impuls |
| 255-01 | UKR FLASH |
| 257-01 | VELCOM Belarus |
| 260-01 | Poland Polkomtel SA PL-PLUS |
| 260-02 | Poland Polska TelefoniaCyfrowa PL-ERA GSM |
| | |

| | |
|---|---|
| 260-03 | Poland Polska TelefoniaKomorkowa Idea Centertel |
| 262-01 | Germany DeTeMobil D1 D1-Telekom |
| 262-02 | Germany Mannesmann Mobilfunk D2 D2-Privat |
| 262- | Germany E-plus E-plus |
| 262- | Germany VIAG E2 |
| 265-01 | Ukraine UA UMC UA UMC |
| 265- | Ukraine Ukrainian Radio System |
| 265- | Ukraine Bancomsvyaz |
| 266-01 | Gibraltar Gibtel Gibtel GSM |
| 268-01 | Portugal Telecel Telecel |
| 268-06 | Portugal Telemovel P TMN |
| 268-03 | Optimus |
| 270-01 | Luxembourg P&T Luxembourg LUXGSM |
| 270-77 | Luxemburg (900/1800) Millicom.SA L TANGO |
| 272-01 | EIR Eircell IRL EIR-GSM |
| 272-02 | EIR Esat Digifone IRL DIGIFONE |
| 274-01 | Iceland Postur og simi IS SIMINN |
| 276-01 | Albania AMC AMC |
| 278- | Malta Advanced |
| 278-01 | Malta Telecell Telecell |
| 280-01 | Cyprus Cyprus Telecom Auth CY CYTAGSM |
| 283-01 | RA-ARMGSM |
| 284-01 | Bulgaria MobiTel CITRON BG |
| 286-01 | Turkey TurkCell TURKCELL |
| 286-02 | Turkey Turk Telekom TR TELSIM |
| 293-41 | Slovenia Mobitel DD SI-GSM |
| 293- | Slovenia Digitel Digitel |
| 294-01 | Macedonia PTT Makedonija MKD-MOBIMAK |
| 302-37 | Canada (PCS) Microcell FIDO |
| 310-02 | USA (PCS) Sprint Spectrum Sprint |
| 310-15 | USA (PCS) BellSouth Mobility PCS |
| 310- | USA (PCS) Pacific Bell Mobile Srvs |
| 310- | USA (PCS) Western Wireless Corp |
| 310-20 | USA (PCS) American Portable Telecoms Sprint |
| 310-16 | USA (PCS) Omnipoint Corporation |
| 310-17 | USA (PCS) |
| 310-27 | USA (PCS) Powertel PCS Partners |
| 310-66 | USA (PCS) DigiPH DigiPH |
| 400-01 | Azerbadjan Azercell ACELL |
| 401-01 | Kazakhstan K-Mobile |
| 401-02 | Kazakhstan K-Cell |
| 404-07 | India TATA INA-TATA |
| 404-10 | India Airtel AIRTEL |
| 404-11 | India Essar ESSAR |
| 404-12 | India Escotel INA-ESCOTEL |
| 404-19 | India Escotel INA-ESCOTEL |
| 404-20 | India Max Touch MAXTOUCH |
| 404-21 | India BPL Mobile BPL MOBILE |
| 404-27 | India BPL Mobile BPL MOBILE |
| 404-30 | India Command COMMAND |
| 404-31 | India Mobilenet MOBILENET |
| 404-40 | India Skycell SKYCELL |
| 404-41 | India RPG MAA RPG MAA |

| | |
|---|---|
| 404- | India Usha Martin |
| 404- | India Modi Telstra |
| 404- | India Sterling Cellular SCL |
| 404- | India Mobile Telecom |
| 404- | India Airtouch |
| 404-43 | India BPL Mobile BPL MOBILE |
| 404-46 | India BPL USWest BPL MOBILE |
| 404- | India Koshiki |
| 404- | India Bharti Telenet |
| 404- | India Birla Comm |
| 404- | India Cellular Comms |
| 404-56 | India Escotel INA-ESCOTEL |
| 404- | India JT Mobiles |
| 404- | India Evergrowth |
| 404- | India Modicom |
| 404- | India Fascel |
| 410-01 | Pakistan Mobilink MOBILINK |
| 413-02 | Sri Lanka MTN Networks Pvt Ltd SRI DALOG |
| 415-01 | Lebanon Cellis (FTML) RL Cellis |
| 415-03 | Lebanon Libancell RL LibCL |
| 416-01 | Jordan FastLink JOR FSTLNK |
| 417-09 | Syria Mobile Syria SYR MOB |
| 418- | Iraq Iraq Telecoms + Posts |
| 419-02 | Kuwait Mobile Telecom KT MTCNet |
| 420-01 | Saudi Arabia MoPTT, DMTS-1 KSA ALJAWWAL |
| 420-07 | Saudi Arabia EAE KSA EAE |
| 422-02 | Oman General Telecoms GTO |
| 424-01 | Un Arab Emirat Etisalat UAE ETSLT |
| 424-02 | Un Arab Emirat Etisalat UAE EG2 |
| 425- | Israel Cellcom Israel Ltd Cellcom |
| 426-01 | Bahrein Batelco BHR MPLUS |
| 427-01 | Qatar QTel QAT Q-NET |
| 432-11 | Iran T.C.I. |
| 432- | Iran Celcom |
| 432- | Iran Kish Free Zone |
| 438-01 | Turkmenistan BCTI |
| 452-01 | Vietnam MTCS VMS-GSM |
| 452-02 | Vietnam Vinaphone (GSM 900) |
| 454-00 | Hong Kong Hong Kong Telecom HK TCSL GSM |
| 454-04 | Hong Kong Hutchinson HutchinsonGSM |
| 454-06 | Hong Kong Smartone Mobile Comm SmarTone |
| 454- | Hong Kong Peoples Telephone Co Ltd |
| 455-01 | Macao C.T.M. CTM GSM |
| 457-01 | Laos Lao Shinawatra Telecom |
| 460-00 | China Guangdong MCC CHNTELGSM |
| 460-01 | China China United Telecom CHN-CUGSM |
| 460- | China Guangxi PTB |
| 460- | China Liaoning PPTA |
| 460- | China Beijing Wireless |
| 460- | China Zhuhai Comms |
| 460- | China Jiaxing PTT |
| 460- | China Tjianjin Toll Telecom |
| 460- | China DGT MPT |
| | |

| | |
|---|---|
| 466-92 | Taiwan Shungwa Telecom LDM ROCLDGSM |
| 902-01 | Malaysia My Digi MY DIGI |
| 902-02 | Malaysia My BSB MY BSB |
| 902-12 | Malaysia Binariang Maxis Mobile MY maxis |
| 902-13 | Malaysia Telecom Malaysia Touch MY MRTEL |
| 902-16 | Malaysia Mutiara Telekom DIGI 1800 |
| 902-17 | Malaysia Sapura Digital Adam PHS MY ADAM |
| 902-19 | Malaysia Celcom GSM Celcom |
| 905-01 | Australia Telstra Corporation Limited Telstra |
| 905-02 | Australia Cable & Wireless Optus Limited YES OPTUS |
| 905-03 | Australia Vodafone Network Pty Limited VODAFONE |
| 905-08 | Australia One-Tel (GSM 1800) |
| 510- | Indonesia PT Kartika Ekamas |
| 510-01 | Indonesia PT Satelindo IND SAT-C |
| 510-10 | Indonesia Telekomsel TELKOMSELGSM |
| 510-11 | Indonesia Excelcom IND-EXCELCOM |
| 510-15 | Indonesia Telekomindo Telekomindo |
| 515-01 | The Philipines IslaCom ISLA |
| 515-02 | The Philipines Globe Telecom GLOBE |
| 515-03 | SMART telecommunications, Phlippines |
| 520-01 | Thailand Advanced Info Serv Pcl TH AIS GSM |
| 520-18 | Thailand Tacs WP1800 |
| 525-01 | Singapore Singapore Telecom ST-GSM-SGP |
| 525-03 | Singapore Mobile One M1-GSM-SGP |
| 528-01 | Brunei Jabatan Telekom |
| 528-11 | Brunei DSTCom |
| 530-01 | New Zealand Bell South BELLSOUTH021 |
| 542-01 | Fiji Vodafone Fiji VODAFONE |
| 546-01 | New Caledonia Mobilis Mobilis |
| 547-20 | Fr Polynesia Tikiphone F VINI Fr Pacific Isls Guinea International Wireless |
| 604-01 | Morocco ONPT Marocko MOR ONPT |
| 605-02 | Tunesia |
| 608-01 | Senegal Sonatel |
| 612-01 | Ivory Coast Comstar |
| 612-03 | Ivory Coast Ivoiris |
| 612-05 | Telecel |
| 617-01 | Mauritius Cellplus Mobile Comms MRU-CELLPLUS |
| 624-01 | Cameroon PTT Cameroon Cellnet CAM CELLNET |
| 633-01 | The Seychelles SEZ SEYCEL SEZ SEYCEL |
| 634-01 | Sudan MobiTel SDN MobiTel |
| 636-01 | Ethiopia ETA ETH-MTN |
| 640-01 | Tanzania TriTel TZ-Tritel |
| 641-01 | Uganda Celtel Cellular CELTEL |
| 641-10 | Uganda Mobile Telephone Networks MTN Uganda. |
| 646-01 | Madacom (Madagascar) |
| 646-02 | Antaris SMM (Madagascar) |
| 646-03 | SACEL (Madagascar) |
| 647-10 | Reunion SRR F SFR RU |
| 648-01 | Zimbabwe PTC Zimbabwe NET ONE |
| 649-01 | Namibia MTC Network NAM MTC |
| 690-01 | Malawi Telekom Network MW CP 900 |
| 651-01 | Lesotho Vodacom VCL COMMS |
| 652-01 | Botswana, Mascom Wireless |

| | |
|---|---|
| 652-02 | Botswana, Vista Cellular |
| 655-01 | South Africa Vodacom VodaCom-SA |
| 655-10 | South Africa Mobile Telephone Networks MTN-SA |

# Appendix - SMS Gateway - Error Codes

| Errorcode | Errorname | Errormessage |
|---|---|---|
| **Error messages** | | |
| **1020** | **ERROR_MAIN_BASE** | |
| 1021 | ERROR_MAIN_GLOBALEXCEPTION | Global exception catched: [MSG] |
| 1022 | ERROR_MAIN_LOCALEXCEPTION | Local exception catched: [MSG] |
| 1023 | ERROR_MAIN_UNEXPECTED_STOP | Stopping engine, because of unknown error. [CODE] |
| 1024 | ERROR_MAIN_CANNOTLOAD_EXTENSION_PHASE1 | Cannot load extenstions. [MSG] |
| 1025 | ERROR_MAIN_CANNOTLOAD_EXTENSION_PHASE2 | Cannot load extension [NAME]. [REASON] |
| **1030** | **ERROR_LICENSING_BASE** | |
| 1031 | ERROR_LICENSING_CANNOTCONTACT_ACTIVATIONSERVER | HTTP Download Error: [MSG] |
| 1032 | ERROR_LICENSING_CANNOTFIND_LICENSEFILE | Could not find license file [FILENAME]. |
| 1033 | ERROR_LICENSING_CANNOTLOAD_LICENSEFILE | Cannot load file [FILENAME]. Reason: [MSG] |
| 1034 | ERROR_LICENSING_CANNOTSAVE_LICENSEFILE | Cannot save file [FILENAME]. Reason: [MSG] |
| **1040** | **ERROR_CONFIG_BASE** | |
| 1041 | ERROR_CONFIG_CANNOTCREATE_DIR | Unable to create directory '[DIR]'. [MSG] |
| 1042 | ERROR_CONFIG_CANNOTWRITE_FILE | Cannot write file. Reason: '[REASON]' |
| 1043 | ERROR_CONFIG_CANNOTREAD_FILE | Cannot read configuration files. Reason: '[REASON]' |
| 1044 | ERROR_CONFIG_CANNOTDELETE_FILE | Cannot delete configuration file. [FILENAME] [MSG] |
| 1045 | ERROR_CONFIG_INVALIDSECTION | Invalid configuration directive in '[SECTIONNAME]' configuration. Directive: '[DIRECTIVE]' |
| 1046 | ERROR_CONFIG_INVALIDSECTION_SECTIONUNCLOSED | Unclosed section in configuration file. Check for closing ,, tags. |
| 1047 | ERROR_CONFIG_INVALIDPARAM_ALREADYEXISTS | Duplicate values for parameter '[PARAM]' in [SECTIONNAME1] and [SECTIONNAME2]. Only value [VALUE] is used. |
| 1048 | ERROR_CONFIG_INVALIDVALUE | Invalid configuration value '[VALUE]' for parameter '[PARAM]' in section '[SECTIONNAME]'. |
| 1049 | ERROR_CONFIG_INVALIDVALUE_LISTENERPORT | Cannot listen on port '[PORT]'. Invalid port number, using default port: 9500 |
| 1050 | ERROR_CONFIG_INVALIDVALUE_UNKNOWNSMSCPROTOCOL | Unknown protocol in configuration file: [PROTOCOL]. |
| 1051 | ERROR_CONFIG_INVALIDVALUE_NOPAIRFOUND | Parameter '[PARAM1]' in section '[SECTIONNAME]' must have a pair named '[PARAM2]'. |
| **1100** | **ERROR_IFTCPCLIENT_BASE** | |
| 1101 | ERROR_IFTCPCLIENT_CANNOTLISTEN | Cannot open TCP listener socket to accept clients on port '[PORT]'. ([MSG]) |
| 1102 | ERROR_IFTCPCLIENT_CONNECTTIMEOUT | Connect timeout. The connected TCP client [CLIENTNAME] did not send any data for more then 10 seconds. |
| 1103 | ERROR_IFTCPCLIENT_SOCKETERROR | Error during communication with UI: '[MSG]' |
| 1104 | ERROR_IFTCPCLIENT_CONNECTIONCLOSED | Client has closed connection ([ERRORMSG]) |
| 1105 | ERROR_IFTCPCLIENT_UNKNOWNPROTOCOL | Unknown protocol. SMPP, UCP and OZTEXT are supported. |
| 1106 | ERROR_IFTCPCLIENT_TEXT_SYNTAXERROR | User '[USER]' has entered unknown command '[COMMAND]' on console. |
| 1107 | ERROR_IFTCPCLIENT_TEXT_UNKNOWNPARAM | User '[USER]' has entered unknown parameter '[PARAMETER]' for command '[COMMAND]' on console. |
| 1108 | ERROR_IFTCPCLIENT_TCP_CLIENTTIMEOUT | Timeout. No response received for PDU '[PDUTYPE]' |
| 1109 | ERROR_IFTCPCLIENT_SMPP_INVALIDPDU | Invalid PDU received: '[PDU]' |
| 1110 | ERROR_IFTCPCLIENT_SMPP_PROTOCOLUNSUPPORTED | Protocol '[PROT]' not supported for this user. Check 'Type' settings in configuration file. |
| 1111 | ERROR_IFTCPCLIENT_SMPP_CLIENTTIMEOUT | Timeout. No response received for PDU '[PDUTYPE]' |
| 1112 | ERROR_IFTCPCLIENT_SMPP_FORCEOUT | Another SMPP client ([CLIENTNAME]) is logged in with this username and password. That client is going to be disconnected because only one client instance is allowed to connect with a single username and password. Use a different username to connect with a second instance! Configure your client to connect as SMPP transceiver! |
| 1113 | ERROR_IFTCPCLIENT_SMTP_MAILPARSE | Cannot parse incoming mail message from [FROM]. Mail format error. [MSG] |
| **1150** | **ERROR_IFHTTP_BASE** | |
| 1151 | ERROR_IFHTTP_LISTENER_GENERALERROR | ozHTTPListener error code:[CODE] says:[MSG] |
| 1152 | ERROR_IFHTTP_LISTENER_GENERALERROR2 | HTTPListener error code:[CODE] says:[MSG] |
| 1153 | ERROR_IFHTTP_INVALID_HTTPPORT | Cannot serve HTTP requests on this port. HTTP API can be accessed on following port: [PORT]. |
| 1154 | ERROR_IFHTTP_CANNOTSERVE_STATICCONTENT | Cannot serve static content [PATH]. [MSG] |
| 1155 | ERROR_IFHTTP_CANNOTSERVE_DYNAMICCONTENT | Cannot serve dynamic content [PATH]. [MSG] |
| 1156 | ERROR_IFHTTP_CANNOTREAD_TEMPLATE | Cannot read template [TEMPLATENAME], [FILENAME]. [MSG] |
| 1157 | ERROR_IFHTTP_UNKNOWNUSER | Invalid username or password. |
| 1158 | ERROR_IFHTTP_CANNOTFIND_ACTIONPARAM | No action parameter specified in HTTP query. |
| 1159 | ERROR_IFHTTP_INVALID_ACTIONPARAMVALUE | Action parameter specified in HTTP query unknown: '[PARAM]'. |
| 1160 | ERROR_IFHTTP_INVALID_PARAMVALUE | Parameter '[PARAMNAME]' has invalid value: '[VALUE]' |
| 1161 | ERROR_IFHTTP_MISSING_MANDATORYPARAM | Following mandatory parameter missing from request: '[PARAMNAME]'. |
| 1162 | ERROR_IFHTTP_INVALID_REPSONSEFORMAT | Invalid response format: [FORMAT]. |
| 1163 | ERROR_HTTPAPI_USERNOTFOUND | Cannot find user [USERNAME]. |
| 1164 | ERROR_HTTPAPI_PARAMETERMISSING | Mandatory http parameter is missing: [PARAMNAME]. |
| 1165 | ERROR_HTTPAPI_ACTIONUNKONWN | Invalid action parameter value: [NAME]. |
| 1166 | ERROR_HTTPAPI_CANNOTCREATEENV | Cannot create envelope. [MSG] |
| 1167 | ERROR_HTTPAPI_PARAMETERVALUEINVALID | Invalid parameter value for parameter [PARAMNAME]: '[PARAMVAL]' |
| **1180** | **ERROR_IFREMOTING_BASE** | |
| 1181 | ERROR_IFREMOTING_TRANSPORT_ERROR | Problem with API transport. [MSG] |
| 1182 | ERROR_IFREMOTING_GENERALERROR | Invalid API action. [MSG] |
| 1183 | ERROR_IFREMOTING_MESSAGEEVENT | Invalid event handling in API on event [EVENT]. [MSG] |
| 1184 | ERROR_IFREMOTING_STARTUP | Cannot start API. [MSG] |

| | | |
|---|---|---|
| **1200** | **ERROR_ENGINE_BASE** | |
| 1201 | ERROR_ENGINE_CANNOTGET_ENVELOPE2SEND | Cannot get envelope from user [USERNAME]. [MSG] |
| **1210** | **ERROR_ROUTER_BASE** | |
| 1211 | ERROR_ROUTER_UNKNOWUSER | Cannot add route. Unknown user '[USERNAME]'. [LINENO] |
| 1212 | ERROR_ROUTER_ROUTENAMENOTUNIQUE | Cannot add route. Route name not unique '[ROUTENAME]'. [LINENO] |
| 1213 | ERROR_ROUTER_OPERATORNOTAVAILABLE | Cannot add route. Service provider name does not exist: '[OPERATORNAME]'. [LINENO] |
| 1214 | ERROR_ROUTER_INVALIDACTION | Cannot add route. Invalid routing mode: '[ACTION]'. [LINENO] |
| 1215 | ERROR_ROUTER_UNKNOWNDIRECTION | Cannot add route. Unknown direction: '[DIRECTION]' |
| 1216 | ERROR_ROUTER_COULDNOTFINDROUTEOUT | Message dropped. No outgoing route found. Message '[MSG]'. |
| 1217 | ERROR_ROUTER_COULDNOTFINDROUTEIN | Message dropped. No incoming route found for message coming from operator [OPERATOR] '[MSG]'. |
| 1218 | ERROR_ROUTER_COULDNOTFINDCALLBACKID | Delivery reported dropped. Corresponding message was sent more than 1 day ago. [MSG] |
| 1219 | ERROR_ROUTER_INVALIDREGEXP | Invalid regular expression [PATTERN] in routing condition. [ERROR]. [LINENO] |
| 1220 | ERROR_ROUTER_COULDNOTFINDROUTE | Could not find route [NAME] |
| 1221 | ERROR_ROUTER_LIMITREACHED | Sorry, you have reached the maximum number of routing rules (which is [NUMROUTES]) that you can add. |
| 1222 | ERROR_ROUTER_COULDNOTPLAYSOUND | Could not play sound: [NAME]. [MSG] |
| 1223 | ERROR_ROUTER_INVALIDCOST | Invalid cost value for route [NAME]: '[COST]' |
| 1224 | ERROR_ROUTER_COULDNOTFINDROUTEOUT_ROUTENAME | Message dropped. No outgoing route found. Route '[ROUTENAME]' does not exist. Message '[MSG]'. |
| 1225 | ERROR_ROUTER_COULDNOTFINDROUTEOUT_SIMPLE | Message dropped. No outgoing route found. |
| 1226 | ERROR_ROUTER_COULDNOTFINDROUTEOUT_ROUTENAME_SIMPLE | Message dropped. No outgoing route found. Route '[ROUTENAME]' does not exist. |
| 1227 | ERROR_ROUTER_INVALIDBACKUPROUTE | Invalid routing configuration. Routename [BACKEDROUTENAME] specifed as route to backup in route [MYROUTENAME], does not exist. |
| 1228 | ERROR_ROUTER_COULDNOTFIND_BACKUPROUTE | Could not find backup route to send message. [MSG] |
| 1229 | ERROR_ROUTER_COULDNOTFIND_BACKUPROUTE_SIMPLE | Could not find backup route to send message. |
| **1240** | **ERROR_SQLLOGING_BASE** | |
| 1241 | ERROR_SQLLOGING_CANNOTCREATE_LOGDIR | Cannot create directory to store SQL log entries. [PATH]. [MSG] |
| 1242 | ERROR_SQLLOGING_CANNOTOPEN_LOGFILE | Cannot open log file for writing [PATH]. [MSG] |
| 1243 | ERROR_SQLLOGING_CANNOTWRITE_LOGENTRY | Cannot write SQL log entry [SQL]. [MSG] |
| 1244 | ERROR_SQLLOGING_CANNOTDELETE_QUEUEFILE | Cannot delete SQL queue file [FILENAME]. [MSG] |
| 1245 | ERROR_SQLLOGING_CANNOTWRITE_QUEUEFILE | Cannot write SQL queue file [FILENAME]. [MSG] |
| 1246 | ERROR_SQLLOGING_CANNOTREAD_QUEUEFILE | Cannot read SQL queue file [FILENAME]. [MSG] |
| **1250** | **ERROR_ENCODING_BASE** | |
| 1251 | ERROR_ENCODING_SCANOTFOUNDINPDU | Cannot find service center address in PDU. |
| 1252 | ERROR_ENCODING_UNKNOWNMESSAGETYPE | Unknown message type identifier in PDU. |
| 1253 | ERROR_ENCODING_MTNOTFOUNDINPDU | Cannot find message type field in PDU. |
| 1254 | ERROR_ENCODING_PHONENOLENNOTFOUNDINPDU | Cannot find phone number length byte in PDU. |
| 1255 | ERROR_ENCODING_INVALIDPHONENUMINPDU | Invalid phone number format in PDU. |
| 1256 | ERROR_ENCODING_PIDNOTFOUNDINPDU | Cannot find protocol identifier in PDU. |
| 1257 | ERROR_ENCODING_DCSNOTFOUNDINPDU | Cannot find data coding scheme in PDU. |
| 1258 | ERROR_ENCODING_SCTSNOTFOUNDINPDU | Cannot find service center timestamp in PDU. |
| 1259 | ERROR_ENCODING_UDLNOTFOUNDINPDU | Cannot find user data length in PDU. |
| 1260 | ERROR_ENCODING_TPMRNOTFOUNDINPDU | Cannot find TPMR in PDU. |
| 1261 | ERROR_ENCODING_TPDTNOTFOUNDINPDU | Cannot find TPDT in PDU. |
| 1262 | ERROR_ENCODING_STNOTFOUNDINPDU | Cannot find ST in PDU. |
| 1263 | ERROR_ENCODING_DTNOTFOUNDINPDU | Cannot find ST in PDU. |
| 1264 | ERROR_ENCODING_CANNOTENCODE | Cannot encode PDU. Reason: [MSG] |
| 1265 | ERROR_ENCODING_CANNOTREADORDELSEGMENT | Cannot read or delete multipart segment. [MSG] |
| 1266 | ERROR_ENCODING_TRNNOTFOUNDINSENTTRN | Cannot find TRN value '[TRN]' in sent transactions. |
| 1267 | ERROR_ENCODING_TRNNOTFOUNDINSENTTRN2 | Cannot find TRN value '[TRN]' in sent transactions (2). |
| 1268 | ERROR_ENCODING_CHECKSUMERROR | Checksum error in pdu. [PDU] |
| 1269 | ERROR_ENCODING_CANNOTDECODE | Cannot decode PDU. Reason: [MSG] |
| 1270 | ERROR_ENCODING_CANNOTENCODEMESSAGE | Cannot encode message [MSG] |
| 1271 | ERROR_ENCODING_CANNOTCREATE_UNKNOWNTYPE | Unknown message type value '[MSG]'. Using default message type (SMS:TEXT) |
| 1273 | ERROR_ENCODING_CANNOTDECODE_UD | Cannot decode User Data of envelope. [MSG] |
| 1274 | ERROR_ENCODING_CANNOTDECODE_TEXTCOMPRESSIONNOTSUPPORTED | |
| 1275 | ERROR_ENCODING_CANNOTDECODE_UNKNOWNDCS | |
| 1276 | ERROR_ENCODING_CANNOTDECODE_UNKNOWNALPHABET | |
| 1277 | ERROR_ENCODING_CANNOTDECODE_UNKNOWNBINARYDCS | |
| 1278 | ERROR_ENCODING_CANNOTDECODE_UNKNOWNINDICATION | |
| 1279 | ERROR_ENCODING_CANNOTDECODE_INVALIDUDHLENGTH | |
| 1280 | ERROR_ENCODING_CANNOTDECODE_INVALIDTLVINUDH | |
| 1281 | ERROR_ENCODING_CANNOTDECODE_INVALIDUDSIZE | |
| 1282 | ERROR_ENCOCING_CANNOTDECODE_SMPPDELIVERYREPORT | Cannot decode delivery report [PDU] |
| 1283 | ERROR_ENCODING_CANNOTDECODE_SMPPDATE | Cannot decode date value [DATE] in PDU [PDU] |
| **1300** | **ERROR_DRIVER_BASE** | |
| 1301 | ERROR_DRIVER_CANNOTCREATEDRIVER_ALREADY_EXISTS | Only one SMSC connection can be configured per configuration file. [FILENAME] |
| 1302 | ERROR_DRIVER_CANNOTCREATEDRIVER_LIMIT_REACHED | Sorry, you have reached the maximum number of drivers (which is [NUMDRIVERS]) that you can install. |
| 1303 | ERROR_DRIVER_CANNOTCREATEDRIVER_PROTOCOL_NOTALLOWED | Sorry, you are not authorized to install a service provider connection for [TYPE] protocol. |
| 1304 | ERROR_DRIVER_CANNOTCREATEDRIVER_PROTOCOL_UNKNOWN | Cannot set up connection with SMSC. Unknown protocol: '[PROTOCOL]'. '[MSG]' |
| 1305 | ERROR_DRIVER_CANNOTPROCESS_WINDOWSMSG | Message processing error: [MSG]; [PARAM] |
| | | |

| | | |
|---|---|---|
| 1306 | ERROR_DRIVER_CANNOTSAVE_CONFIG | Cannot save configuration file. [FILENAME] |
| 1307 | ERROR_DRIVER_CANNOTCREATE_MULTIPART_DIR | Cannot create directory [DIR]. [MSG] |
| 1308 | ERROR_DRIVER_KEEPALIVE_TIMEOUT | Keepalive packet timeout. |
| 1309 | ERROR_DRIVER_KEEPALIVE_NACK | Keepalive packet rejected. |
| 1310 | ERROR_DRIVER_CANNOTCONNECT | |
| 1311 | ERROR_DRIVER_CONNECTIONERROR | Cannot connect to SMSC. [MSG] |
| 1312 | ERROR_DRIVER_CONNECTION_ERROR | Unsuccessful connection. [MSG] |
| 1313 | ERROR_DRIVER_SHUTDOWN | |
| 1314 | ERROR_DRIVER_INVALIDUSERNAME | Invalid username or password. |
| 1315 | ERROR_DRIVER_NORESPONSE | No response received from server |
| 1316 | ERROR_DRIVER_UNKNOWNNORESPONSE | Unknown response received from server |
| 1317 | ERROR_DRIVER_TCPCONNECTIONLOST | No TCP connection to SMSC. |
| 1318 | ERROR_DRIVER_INVALIDUSERNAME2 | Could not log in to server. [MSG] |
| 1319 | ERROR_DRIVER_TCP_CONNECTIONERROR | TCP connection error. (Check http://www.ozekisms.com/index.php?owpn=423) for more information on this error. Error messages: [MSG] |
| 1320 | ERROR_DRIVER_UCP_TRNANOMAILY | Could not find oldest TRN, and could not give out TRN for transaction. This should not have happened. Report this error! |
| **1350** | **ERROR_DRIVER_ENVSUBMIT_BASE** | |
| 1351 | ERROR_DRIVER_INVALIDRECIPIENT | Invalid recipient address in message '[MSG]'. |
| 1352 | ERROR_DRIVER_SUBMITTIMEOUTERROR1 | Submit timeout ([SEC]). No response received from SMSC while sending message [MSG]. Message considered sent. |
| 1353 | ERROR_DRIVER_SUBMITTIMEOUTERROR2 | Submit timeout ([SEC]). No response received from SMSC while sending message [MSG]. Message needs resending. |
| 1354 | ERROR_DRIVER_SUBMITTIMEOUTERROR3 | Submit timeout ([SEC]). No response received from SMSC while sending message [MSG]. Message considered undeliverable. |
| 1355 | ERROR_DRIVER_ENVSUBMIT_ERROR_NOCONNECTION | No connection to the service provider. |
| 1356 | ERROR_DRIVER_ENVSUBMIT_ERROR_NORESPONSE | No response from SMSC. Connection timeout on connection: [DRIVERINSTANCE] |
| 1357 | ERROR_DRIVER_ENVSUBMIT_ERROR_NORESPONSE2 | Undelivered message. Submit timeout. No repsonse received from SMSC to submit request. |
| 1358 | ERROR_DRIVER_ENVSUBMIT_ERROR_NOTSMS | Unknown envelope content. This envelope does not seem to contain a valid SMS message. Possible routing problem? |
| 1359 | ERROR_DRIVER_ENVSUBMIT_ERROR_NOTACCEPTED | Envelope rejected by SMSC. |
| 1360 | ERROR_DRIVER_ENVSUBMIT_ERROR | Envelope could not be sent. [ERRORMSG] |
| 1361 | ERROR_DRIVER_ENVSUBMIT_ERROR_TIMEOUT | Timeout ([SUBMITTIMEOUT] sec.) Envelope could not be sent. [ERRORMSG] |
| 1362 | ERROR_DRIVER_ENVSUBMIT_ERROR_NOREFERENCE | Could not find message reference in submit response. Delivery reports will not be handled correctly. |
| 1363 | ERROR_DRIVER_ENVSUBMIT_ERROR_NOTACCEPTED_REASON | Envelope rejected by SMSC. [MSG] |
| 1364 | ERROR_DRIVER_ENVSUBMIT_ERROR_NOTDELIVERED | Delivery report received that sais could not deliver message to recipient. [ERRORMSG] |
| 1365 | ERROR_DRIVER_ENVSUBMIT_NOTSENT | Message could not be sent. Reason: [MSG] |
| 1366 | ERROR_DRIVER_ENVSUBMIT_ERROR_INVRESP | Envelope could not be sent by modem. [ERRORMSG] |
| 1367 | ERROR_DRIVER_ENVSUBMIT_NOTACCEPTED | The SMSC returned a not accepted response. [MSG] |
| **1400** | **ERROR_DRIVER_GSMMODEM_BASE** | |
| 1401 | ERROR_DRIVER_GSMMODEM_SIM_SMSC_DIFFERENCE | The address of SIM SMSC ([SIMSMSC]) is different from the address of SMSC specified in the configuration form ([SETSMSC]) |
| 1402 | ERROR_DRIVER_GSMMODEM_SIM_SMSC_AND_SET_SMSC_EMPTY | Both, the address of SIM SMSC and the address of SMSC specified in the configuration form are empty! You have to specify an SMSC to make SMS working. Check: http://www.ozekisms.com/index.php?owpn=199 |
| 1403 | ERROR_DRIVER_GSMMODEM_SIM_SMSC_SET_SMSC_EMPTY | Cannot read SMSC from SIM and the SMSC specified on the configuration form is empty! |
| 1404 | ERROR_DRIVER_GSMMODEM_PINEMPTY | A PIN code needs to be entered to start the modem, but no PIN code is defined in the 'Pin code' field of the 'Port settings' tab of the configuration form. Please specify the PIN code in the configuration form! |
| 1405 | ERROR_DRIVER_GSMMODEM_CANNOTDECODE_DELIVERYREPORT | Cannot decode delivery report [PDU] |
| 1406 | ERROR_DRIVER_GSMMODEM_MMS_ CANNOTDOWNLOAD_NODOWNLOADURL | Cannot download MMS message, because no download URL was found in the MMS Indication message. |
| 1407 | ERROR_DRIVER_GSMMODEM_MMS_ CANNOTDOWNLOAD_DOWNLOADFAILED | Cannot download MMS message from [URL]. [MSG] |
| 1408 | ERROR_DRIVER_GSMMODEM_MMS_ CANNOTDOWNLOAD_CANNOTCREATEFILE | Cannot create support file for downloading MMS message. Filename: [FILENAME]. [MSG] |
| 1409 | ERROR_DRIVER_GSMMODEM_MMS_ CANNOTDOWNLOAD_CANNOTDELETEFILE | Cannot delete support file after MMS download attempt. Filename: [FILENAME]. [MSG] |
| 1410 | ERROR_DRIVER_GSMMODEM_MMS_ CANNOTDOWNLOAD_CANNOTCREATEDIR | Cannot create support directory for MMS downloads. [DIRNAME] [MSG] |
| 1411 | ERROR_DRIVER_GSMMODEM_MMS_ CANNOTDOWNLOAD_CANNOTCONTINUEFAILEDDOWNLOAD | Cannot process entries in MMS download support directory, to continue previously failed MMS downloads. [DIRECTORY] [MSG] |
| 1412 | ERROR_DRIVER_GSMMODEM_MMS_ CANNOTDOWNLOAD_NOMORETRIES | MMS download failed. We have tried to download this MMS message too many times from the MMSC without success. [URL] |
| 1413 | ERROR_DRIVER_GSMMODEM_CANNOTOPENPORT | Cannot connect to port [PORT]. [MSG]. The port you want to access might be used by another process. Please make sure that no other SMS or modem software is running, that uses this port. It might also be possible that an MMS sending/receiving procedure was unexpecdly interrupted. If you issue the following command in a windows command shell: rasdial /disconnect, the problem might be resolved. As a final option if you reboot your computer, there is a good chance this error will not occur. |
| 1414 | ERROR_DRIVER_GSMMODEM_NORESPONSE | Port has been successfully opened, but no response has been received from modem on [PORT].Check data cable, or try different baud rate! Is your GSM phone or modem powered on? |
| 1415 | ERROR_DRIVER_CANNOTCREATEDRIVER_COULDNOTINIT | Could not initialize service provider connection [INSTANCE]. [MSG] |
| 1416 | ERROR_DRIVER_SUBMITPDU_ERROR | Could not submit PDU. Reason: [MSG] |
| 1417 | ERROR_DRIVER_COULDNOTPARSE_RESPONSE | Could not parse response from [SOURCE]. Response was: [RESPONSE] |
| 1418 | ERROR_DRIVER_GSMMODEM_CANNOTREADMODEMLIST | Cannot read list of modems. [MSG] |
| 1419 | ERROR_DRIVER_GSMMODEM_USSD_NOTSUPPORTED | Cannot initiate USSD request. USSD is not supported by this modem. |
| 1420 | ERROR_DRIVER_GSMMODEM_VOICE_NOTSUPPORTED | Cannot set modem into voice mode. This is not a voice modem. |
| 1421 | ERROR_DRIVER_GSMMODEM_VOICE_CANNOTSEND | Cannot send voice through opened modem channel. [MSG] |
| 1422 | ERROR_DRIVER_GSMMODEM_VOICE_COULDNOTDIAL | Cannot dial number [NUM]. |
| 1423 | ERROR_DRIVER_GSMMODEM_VOICE_VTXERROR | Cannot start voice stream transmission. AT+VTX command did not return with CONNECT. |
| 1424 | ERROR_DRIVER_GSMMODEM_VOICE_SENDERROR | Could not send voice stream |

| | | |
|---|---|---|
| 1425 | ERROR_DRIVER_GSMMODEM_PORT_UNEXPECTEDLY_CLOSE | Port was unexpecedly closed while reading input data. [MSG] |
| 1426 | ERROR_DRIVER_GSMMODEM_CANNOTUPDATEROUTINGTABLE | Cannot update routing table. [MSG] |
| 1427 | ERROR_DRIVER_GSMMODEM_GPRSCANNOTCLOSE | Cannot close GPRS connection. Try to execute RASDIAL /DISCONNECT manually at the command prompt. Error information: [MSG] |
| 1428 | ERROR_DRIVER_GSMMODEM_GPRSCANNOTOPEN | Cannot open GPRS connection. [MSG] |
| 1429 | ERROR_DRIVER_GSMMODEM_INVALIDSETTING | Invalid configuration for [SETTINGNAME]: [VALUE] |
| 1430 | ERROR_DRIVER_GSMMODEM_CANNOTSENDMMS | Cannot send MMS message. [MSG] |
| **1450** | **ERROR_DRIVER_HTTPSERVER_BASE** | |
| 1451 | ERROR_DRIVER_HTTPSERVER_SIMULATEDERROR | |
| 1452 | ERROR_DRIVER_HTTPSERVER_DUPLICATEBASEURL | Duplicate base URL found: [MSG] |
| 1453 | ERROR_DRIVER_HTTPSERVER_NOORIGINATOR | Incoming message does not contain originator address. |
| 1454 | ERROR_DRIVER_HTTPSERVER_CANNOTOPENOUTBOX | Cannot open outbox file [FILENAME]. ([ERRORMSG]) |
| 1455 | ERROR_DRIVER_HTTPSERVER_OUTBOXNOTAVAILABLE | Outbox file [FILENAME] not available. |
| 1456 | ERROR_DRIVER_HTTPSERVER_INVALIDOUTBOX | Invalid outbox file format in [FILENAME]. ([ERRORMSG]) |
| 1457 | ERROR_DRIVER_HTTPSERVER_HTTPLISTENERNOTSUPPORTED | Windows XP SP2 or Server 2003 required to use HttpListener class. |
| **1470** | **ERROR_DRIVER_PARLAYX_BASE** | |
| 1471 | ERROR_DRIVER_PARLAYX_CANNOTSUBMIT | Cannot submit SMS message. Reason: [MSG] |
| 1472 | ERROR_DRIVER_PARLAYX_CANNOTRECEIVE | Cannot complete SMS polling, to download incoming SMS messages from ParlayX server. [MSG] |
| 1473 | ERROR_DRIVER_PARLAYX_CANNOTREGISTERCALLBACK | Cannot register callback URL '[URL]' on ParlayX server. [MSG] |
| 1474 | ERROR_DRIVER_PARLAYX_CANNOTFINDDRIVER | Cannot find an installed ParlayX connection, that can receive the message with correlator [CORRELATOR]. [ENV] |
| 1475 | ERROR_DRIVER_PARLAYX_CANNOTREGISTERCALLBACKSERVER | Cannot register server on localhost to accept callback request on URL '[URL]' from ParlayX server. [MSG] |
| 1476 | ERROR_DRIVER_PARLAYX_CANNOTUNREGISTERCALLBACK | Cannot unregister callback URL '[URL]' on ParlayX server. [MSG] |
| 1477 | ERROR_DRIVER_PARLAYX_CANNOTPROCDELIVERYREPORT | Cannot process delivery report. [EXP] |
| **1500** | **ERROR_USER_BASE** | |
| 1501 | ERROR_USER_CANNOTCREATE_USER_NOUSERNAME | Cannot create user. Username field cannot be left empty. |
| 1502 | ERROR_USER_CANNOTCREATE_USER_ALREADYEXISTS | Cannot create user. Another user with username '[USERNAME]' already exists. [LINENO] |
| 1503 | ERROR_USER_CANNOTCREATE_USER_UNKNOWNTYPE | Cannot create user. Invalid user type: '[TYPE]'. [LINENO] |
| 1504 | ERROR_USER_CANNOTCREATE_USER_TYPENOTALLOWED | Sorry, you are not authorized to add a user of [TYPE] type. |
| 1505 | ERROR_USER_CANNOTCREATE_USER_LIMITREACHED | Sorry, you have reached the maximum number of users (which is [NUMUSERS]) that you can add. |
| 1506 | ERROR_USER_CANNOTCREATE_ADDRESSBOOK | Cannot load addressbok. Try to remove the addressbook directory '[DIR]'! [MSG] |
| 1507 | ERROR_USER_CANNOTENABLE | Error happend while enabling user. [MSG] |
| 1508 | ERROR_USER_CANNOTENABLE_NOTCONFIGURED | Cannot connect to database. The database connection is not yet configured. |
| 1509 | ERROR_USER_CANNOTDISABLE | Error happend while disabling user. [MSG] |
| 1510 | ERROR_USER_MESSAGE_NOTACCEPTED | The message posted by [USER] was notaccepted for delivery. [REASON] [ENVSTR] |
| 1511 | ERROR_USER_MESSAGE_NOTACCEPTED_OUTBOXFULL | The outbox queue is full |
| 1512 | ERROR_USER_MESSAGE_NOTACCEPTED_TYPENOTSUPPORTED | Message type [MSGTYPE] not supported. |
| 1513 | ERROR_USER_MESSAGE_NOTACCEPTED_INVALIDRECIPIENT | Invalid recipient address or no recipient address specified |
| 1514 | ERROR_USER_MESSAGE_NOTACCEPTED_CANNOTENCODE | Cannot encode message. Reason [MSG] |
| 1515 | ERROR_USER_MESSAGE_NOTACCEPTED_CANNOTPARSE | Parsing stopped, becase there are too many parsing errors. More then 5 lines had errors. |
| 1520 | ERROR_USER_MESSAGESTORE_INVALID_QUEUE_SIZE | Invalid queue size for users: [SIZE] |
| 1521 | ERROR_USER_MESSAGESTORE_CANNOTCREATE_DIR | Cannot create directory [DIR]. [MSG] |
| 1522 | ERROR_USER_MESSAGESTORE_CANNOTSAVE_ENVELOPE | Cannot save envelope [FILE]. [MSG] |
| 1523 | ERROR_USER_MESSAGESTORE_CANNOTSAVE_ERRORENVELOPE | Cannot create error file [FILE]. [MSG] |
| 1524 | ERROR_USER_MESSAGESTORE_CANNOTLOAD_ENVELOPE | Cannot load envelope [FILE]. [MSG] |
| 1525 | ERROR_USER_MESSAGESTORE_CANNOTDELETE_ENVELOPE | Cannot delete envelope [FILE]. [MSG] |
| 1526 | ERROR_USER_MESSAGESTORE_CANNOT_DESERIALIZE | Cannot load envelope from file. Ozeki NG version incompatibility. [FILE]. Reason: [MSG] |
| 1527 | ERROR_USER_MESSAGESTORE_CANNOT_SERIALIZE | Cannot save envelope to file: [FILE]. Reason: [MSG] |
| 1540 | ERROR_USER_ACCOUNTING_REQUEST_INSUFFICIENT_CREDITS | Cannot reserve credits: Insufficient credits '[BALANCE]' but needed '[NEEDED]'. [ENVSTR] |
| 1541 | ERROR_USER_ACCOUNTING_UPDATE_INSUFFICIENT_CREDITS | Cannot start operation: Insufficient credits ([BALANCE]). [ENVSTR] |
| 1542 | ERROR_USER_ACCOUNTING_ACCOUNT_NOT_FOUND | Account for user '[USER]' not found. [ENVSTR] |
| 1543 | ERROR_USER_ACCOUNTING_TRANSACTION_ADD | Cannot save transaction to file: [MSG] |
| 1544 | ERROR_USER_ACCOUNTING_CREATE | Cannot create account to file [FILENAME], [MSG] |
| 1545 | ERROR_USER_ACCOUNTING_LOAD | Cannot load account from file [FILENAME], [MSG] |
| 1546 | ERROR_USER_ACCOUNTING_ACCOUNT_EXPIRED | Account has been expired at [EXPRDATE]. [ENVSTR] |
| 1547 | ERROR_USER_ACCOUNTING_INSUFFICIENT_CREDITS | Insufficient credits. |
| **1600** | **ERROR_USERIMPL_BASE** | |
| 1601 | ERROR_USERAUTOREPLY_CANNOTFIND_SCRIPTFILE | Cannot find configuration file: [FILENAME] |
| 1602 | ERROR_USERAUTOREPLY_RESPONSELIMITREACHED | Cannot send response message. The response limit forbids sending more then [NUM] responses to the same number. For more information, please visit http://www3.ozekisms.com/index.php?owpn=203. Message: [MSG] |
| 1610 | ERROR_USERFILE_CLEANUP_EXCEPTION | Error wile deleteing lock files in '[FUNCTION]'. Exception: [MSG] |
| 1611 | ERROR_USERFILE_NOTFOUND | Could not find file [FILENAME]. It was probably deleted while beeing processed. [MSG] |
| 1612 | ERROR_USERFILE_CANNOTPROCESS | Cannot process file [FILENAME]. The format of the file content might be invalid. [MSG] |
| 1613 | ERROR_USERFILE_CANNOTREAD | Cannot read file [FILENAME]. There was an IO error. [MSG] |
| 1614 | ERROR_USERFILE_CANNOTCREATEDIR_ALREADYEXISTS | Directories cannot be same for parameter '[PARAM1]' and '[PARAM2]' value of '[VALUE]' in [SECTIONNAME]. Resetting all directory values to default. |
| 1615 | ERROR_USERFILE_DIRERROR | Directory for polling not found. [MSG] |
| 1616 | ERROR_USERFILE_OUTDIRERROR | Output directory not found. [MSG] |
| 1617 | ERROR_USERFILE_INVLALID_CSVLINE | Invalid CSV line. The line should have 3 sections: phone number, message text, date |
| 1630 | ERROR_USERTCP_CANNOTPROCINCOMING | Client did not accept [NOTIFICATION]. [MSG] |
| 1640 | ERROR_USERHTTPCLIENT_INVALIDRESPONSEFORMAT | Cannot send SMS message(s). Content of response has invalid format (See http://www.ozekisms.com/index.php?owpn=355). [ERRORS] [RESPONSE] |
| | | |

| | | |
|---|---|---|
| 1641 | ERROR_USERHTTPCLIENT_REQUESTTIMEOUT | Request Timeout: [MSG] |
| 1642 | ERROR_USERHTTPCLIENT_INVALIDURL | Format of target URL could not be determined: [MSG] |
| 1643 | ERROR_USERHTTPCLIENT_CANNOTRESOLVEHOSTNAME | Remote name could not be resolved: [MSG] |
| 1644 | ERROR_USERHTTPCLIENT_CANNOTOPENURL | HTTP Error. [MSG] Cannot open URL [URL]. |
| 1650 | ERROR_USER_APPSTARTER_CANNOTEXEC | Cannot execute application. Reason: [MSG] |
| 1651 | ERROR_USER_APPSTARTER_INVALIDRESPONSE | Cannot process response message. Invalid format. [MSG] |
| 1660 | ERROR_USER_API_CANNOTLOADFORM | Cannot load form settign. [MSG] |
| 1661 | ERROR_USER_API_CANNOTSAVEFORM | Cannot save form settign. [MSG] |
| 1662 | ERROR_USER_API_CANNOTREADDESCR | Cannot read API description. [MSG] |
| 1670 | ERROR_USER_CANNOTPOLLMESSAGES | Exception catched during polling. [MSG] |
| **Warnings** | | |
| **2100** | **WARN_CONFIG_BASE** | |
| 2101 | WARN_CONFIG_DIRCREATE | Unable to create directory '[DIR]'. [MSG] |
| 2102 | WARN_CONFIG_DUPLICATEDIR | Directories cannot be same for parameter '[PARAM1]' and '[PARAM2]' value of '[VALUE]' in [SECTIONNAME]. Resetting all directory values to default. |
| 2103 | WARN_CONFIG_PARAMETERVALUENOTUSED | Value '[VALUE1]' for parameter '[PARAM1]' is not used because '[PARAM2]'='[VALUE2]' in section '[SECTIONNAME]' |
| 2104 | WARN_CONFIG_PARAMETERFILENOTUSED | File '[VALUE1]' for parameter '[PARAM1]' is not used because [PARAM2]=[VALUE2] in section '[SECTIONNAME]' |
| **2150** | **WARN_IFTCPCPLIENT_BASE** | |
| 2151 | WARN_IFTCPCPLIENT_CLIENTIDLE | Client has been idle for more then [IDLETIME] seconds. Disconneting client. |
| **2200** | **WARN_ROUTER_BASE** | |
| 2201 | WARN_ROUTER_RESENDINGMESSAGE_ONBACKUPROUTE | Message send failed. Trying to find backup route for message [MSG]. |
| 2202 | WARN_ROUTER_DELIVERYREPORT_REFERENCEDMESSAGENOTFOUND | Cannot find corresponding envelope for incoming delivery report in sent items. Missing envelope ID: [ENVID] |
| 2203 | WARN_ROUTER_DELIVERYREPORT_MESSAGEREFERENCE_ DOESNOTMATCHCOMPLETELEY | Envelope callback ID '[CID]' information does not match received callback ID information. Envelope ID: [ENVID] |
| 2204 | WARN_ROUTER_DELIVERYREPORT_MESSAGEREFERENCE_ALREADYUSED | This callback ID is already registered in the delivery report router [CID]. |
| 2205 | WARN_ROUTER_DELIVERYREPORT_COULDNOTFIND_PARLAYXMSG | Could not find message with ID '[ID]' |
| **2250** | **WARN_ENCODING_BASE** | |
| 2251 | WARN_ENCODING_UNEXPECTEDFORMAT_PDUNONHEX | Message PDU contains unexpected (non-hex) characters. Message might not be parsed correctly. [PDU] |
| 2252 | WARN_ENCODING_UNSUPPORTED_CHARSET | The following charset is not supported on this system: [CHARSET]. Using default charset. |
| 2253 | WARN_ENCODING_UNSUPPORTED_SMPPDCS | Unsupported data coding scheme [DCS] |
| 2254 | WARN_ENCODING_INVALID_GSMDATETIME | Invalid message. Cannot decode date time value [DATETIME] |
| 2255 | WARN_ENCODING_NOVALIDRECIPIENT_IN_MESSAGE | Could not find a valid recipient in the message. |
| **2300** | **WARN_DRIVER_BASE** | |
| 2301 | WARN_DRIVER_CANNOTCONNECT_UNCONFIGURED | Cannot connect with default parameters. Please configure this connection first! |
| 2302 | WARN_DRIVER_CANNOTINITDRIVER_FROMGUI | Cannot initialize driver for GUI access [DRV]. [MSG] |
| 2303 | WARN_DRIVER_AUTOCONNECTDISABLED | Autoconnect feature disabled. You have to connect to the service manually by clicking on the connect link. |
| 2304 | WARN_DRIVER_SUBMITTING_EMPTYMESSAGE | The messagedata of the envelope is empty. |
| 2310 | WARN_DRIVER_GSMMODEM_NOTCONFIGURED | Some mandatory parameters are empty on the configuration form. This indicates, that the connection has not yet been configured. Please configure your connection! |
| 2311 | WARN_DRIVER_GSMMODEM_SIM_SMSC_EMPTY | SMSC address in SIM Card is empty! |
| 2312 | WARN_DRIVER_GSMMODEM_SIM_SMSC_MODIFIED | SIM card SMSC number has been modified to [SMSC]. |
| 2313 | WARN_DRIVER_GSMMODEM_SIM_SMSC_READ_UNSUCCESSFUL | Getting SMSC address from SIM is unsuccessful! |
| 2314 | WARN_DRIVER_GSMMODEM_SIM_SMSC_WAIT | Waiting [WAITTIME] millisec. |
| 2320 | WARN_DRIVER_GSMMODEM_CMTNOTSUPPORTED | This modem does not support GSM Phase 2, so CMT memory reading option is not available. Using CMGL instead. |
| 2321 | WARN_DRIVER_GSMMODEM_NOTREGISTERED_TO_NETWORK | The modem is NOT registered to the GSM network. Check your antenna! Disable PIN Code! |
| 2322 | WARN_DRIVER_GSMMODEM_TEXTMODE_LONGER160 | Some modems cannot send messages longer then 160 characters in text mode. |
| 2323 | WARN_DRIVER_GSMMODEM_LOCALNUMBERFORMAT | The number format for the recipient phone number is a local number format. It does not start with a + sign. If you cannot submit this messages, please try to use international number format (plus sign + country code + number). It might be better to use '+[NUM]' instead of '[NUM]'. |
| 2324 | WARN_DRIVER_GSMMODEM_RESPONSE_TIMEOUT | Modem Timeout. The modem did not respond to the command. Is it plugged in? |
| 2325 | WARN_DRIVER_GSMMODEM_MMS_CANNOTCREATE_ATTACHMENTDIR | Cannot create directory to store MMS attachments [PATH] |
| 2340 | WARN_DRIVER_CIMD2_INVALIDADDRESS | '[ADDRESS]' is not a valid CIMD2 address nor an alphanumeric address. |
| 2341 | WARN_DRIVER_CIMD2_NOTSTARTWSENDERPREFIX | '[ADDRESS]' doesnt start with '[PREFIX]'. |
| 2342 | WARN_DRIVER_CIMD2_NOTANADDRESS | '[ADDRESS]' contains some invalid characters and it's not an address. |
| 2343 | WARN_DRIVER_CIMD2_ALPHANUMADDRESSTOOLONG | Alphanumeric address '[ADDRESS]' is too long. Maximum [NUMCHARS] characters are allowed. Truncated address will be: [TRUNCADDR] |
| 2350 | WARN_DRIVER_UCP_NORESP_ON_TRANSACTION | No response received from SMSC to transaction [TRN], timeout value is [TIMEOUT]. |
| 2360 | WARN_DRIVER_PARLAYX_CALLBACKNOTSUPPORTED | Callbacks are not available in ParlayX version 2.0. Disabling callback requests. |
| 2361 | WARN_DRIVER_PARLAYX_DELIVERYCALLBACK_NOT_SUPPORTED | Delivery report callbacks in ParlayX Version 2.0 are not supported |
| **2500** | **WARN_USER_BASE** | |
| 2501 | WARN_USER_ALREADY_ENABLED | User has already been enabled. |
| 2502 | WARN_USER_ALREADY_DISABLED | User has already been disabled. |
| 2510 | WARN_USER_MESSAGESTORE_CANNOTDOWNLOAD_MESSAGELIST | Cannot read message from queue [QUEUE]. [REASON] |
| 2511 | WARN_USER_MESSAGESTORE_CANNOTREAD_MESSAGE | Error while reading message in outbox queue. [MSG] |
| 2512 | WARN_USER_MESSAGESTORE_CANNOTMOVE_MESSAGE | Cannot move file from [SOURCEFILE] to [DESTFILE]. [MSG] |
| 2513 | WARN_USER_MESSAGESTORE_CANNOTSAVE_ENVELOPE | Cannot save envelope [FILE]. Saving it to file [NEWFILE]. [MSG] |
| 2514 | WARN_USER_MESSAGESTORE_CANNOTDELETE_ENVELOPE | Cannot delete envelope [FILE]. Creating error file [ERRFILE]. [MSG] |
| 2515 | WARN_USER_MESSAGESTORE_MESSAGE_NOTFOUND | Message [MSGID] not found in queue. Removing reference |
| 2530 | WARN_USER_SQL_INVALIDCHAR_IN_SQLTEMPLATE | Invalid character(s) in SQL template. There is a (hidden) unicode character in the template. It is possible that you have used copy paste to insert it. Type your templates manually. If you want to use |

| | | |
|---|---|---|
| | | unicode characters, you must enable unicode template support on the Advanced tab of the confiugration form! Unicode characters are now removed from the template. |
| 2531 | WARN_USER_SQL_ODBCNOTSUPPORTED_FORMSSQLSERVER | This seems like an ODBC connection string for Microsoft SQL Server. We recommend to use OLEDb provider SQLNCLI for Microsoft SQL Server connections. ODBC can cause problems. More information is available at: http://www.ozekisms.com/index.php?owpn=167 |
| 2540 | WARN_USER_EMAIL_MESSAGE_FILTERED | Can't send the following SMS, because e-mails are filtered: [SMS] |
| 2541 | WARN_USER_EMAIL_MESSAGELOOP_DANGER | This e-mail will be not sent as SMS to prevent SMS loops remove the SMS/REPORT tag from the subject line. [SMS] |
| 2542 | WARN_USER_EMAIL_CANNOTCONNECTTOPOP3 | Can't connect to POP3 server: [MSG]. |
| 2543 | WARN_USER_EMAIL_CANNOTCLOSEPOP3 | Can't close POP3 server: [MSG]. |
| 2544 | WARN_USER_EMAIL_CANNOTLOGINTOPOP3 | Can't log in to POP3 server with [MSG]. |
| 2545 | WARN_USER_EMAIL_SMSTOEMAIL_FORWARDINGFAILED | SMS -> E-MAIL: [MSG] |
| 2546 | WARN_USER_EMAIL_EMAILTOSMS_FORWARDINGFAILED | E-MAIL -> SMS: [MSG] |
| 2560 | WARN_USER_AUTOREPLYEASY_INVALIDMSGTYPE | Invalid message type defined [MSGTYPE] |
| **Information** | | |
| **3010** | **INFO_MAIN_BASE** | |
| 3011 | INFO_MAIN_SERIALNUMBER | Serial numer: [SERIAL] |
| 3012 | INFO_MAIN_ACTIVATIONCODE | Activation code: [CODE] |
| 3013 | INFO_MAIN_SERVICE_SHUTDOWN_REASON | Stop reason: [REASON] |
| 3014 | INFO_MAIN_SERVICE_INITIALIZED | Service initialized successfully. |
| 3015 | INFO_MAIN_SERVICE_STOPPING_HEARTBEAT | Stopping heartbeat |
| 3016 | INFO_MAIN_SERVICE_STOPPING_SMSC | Stopping SMSC connections |
| 3017 | INFO_MAIN_SERVICE_STOPPING_GUI | Stopping user interfaces |
| 3018 | INFO_MAIN_SERVICE_STOPPING_MESSAGESTORE | Stopping user queue manager |
| 3019 | INFO_MAIN_SERVICE_STOPPING_ROUTER_PHASE1 | Stopping routing engine, phase I |
| 3020 | INFO_MAIN_SERVICE_STOPPING_ROUTER_PHASE2 | Stopping routing engine, phase II |
| 3021 | INFO_MAIN_SERVICE_STOPPING_ENGINE | Stopping message engine |
| 3022 | INFO_MAIN_SERVICE_STOPPING_SQLLOGGER | Stopping SQLLogger |
| 3023 | INFO_MAIN_SERVICE_STOPPING_SERVICE | Stopping service |
| 3024 | INFO_MAIN_SERVICE_STOPPED | Service stopped |
| 3025 | INFO_MAIN_LOADING_EXTENSTION | Loading extension: [NAME] |
| **3050** | **INFO_CONFIG_BASE** | |
| 3051 | INFO_CONFIG_AUTOCONNECT_NOTCONFIGURED | Autoconnect is disabled. To start the connection, click on connect! |
| **3100** | **INFO_IFTCPCLIENT_BASE** | |
| 3101 | INFO_IFTCPCLIENT_CONNECTED | TCP client connected from '[IPADDRESS]'. Connection id: [CONNID] |
| 3102 | INFO_IFTCPCLIENT_DISCONNECTED | TCP client disconnect '[IPADDRESS]' disconnected. Connection id: [CONNID] |
| 3103 | INFO_IFTCPCLIENT_SERVER_STATUS | TCP Server status information: [STATUS] |
| 3104 | INFO_IFTCPCLIENT_LOGGEDIN | Client connected with username '[USERNAME]' from '[CLIENTNAME]'. |
| 3105 | INFO_TCPCLIENT_LOGGEDOUT | Client disconnected. (username '[USERNAME]', connected from '[CLIENTNAME]') |
| **3150** | **INFO_ROUTER_BASE** | |
| 3151 | INFO_ROUTER_ROUTING_INCOMING_MESSAGE | Incoming message routed to '[USER]' using route '[ROUTENAME]' [POLICY]. Message: [ENVSTR] |
| 3152 | INFO_ROUTER_ROUTING_OUTGOING_MESSAGE | Outgoing message routed to '[OPERATOR]' using route '[ROUTENAME]'. Message: [ENVSTR] |
| 3153 | INFO_ROUTER_ROUTING_DELIVERYREPORT | Matching delivery report received to reference '[CALLBACKID]'. |
| 3154 | INFO_ROUTER_DELIVERYREPORT_COULDNOTFIND_MATCHINGENVELOPE | Delivery report dropped, because no matching envelope found in system for callback ID '[CID]'. This error can happen in two cases: The message was sent more then 1 week ago or the message reference (callback ID) returned by the SMSC during sending does not match the callback ID in the delivery report. [ENV] |
| **3200** | **INFO_ENCODING_BASE** | |
| 3201 | INFO_ENCODING_PDUCOUNT | Message will be sent using [COUNT] PDU(s). |
| 3202 | INFO_ENCODING_CHARSETINFO | Encoding information: Message length is [MSGLEN]. It is encoded as [CHARSET]. It will be sent in [PARTCOUNT] message(s). |
| 3203 | INFO_ENCODING_SEGMENTRECEIVED | Multipart message segment received. Segment saved as [MSG] |
| 3204 | INFO_ENCODING_LASTSEGMENTRECEIVED | Last multipart message segment received. |
| 3205 | INFO_ENCODING_SEGMENT_ACCEPTED | |
| 3206 | INFO_ENCODING_SEGMENT_NOTFOUND | |
| 3207 | INFO_ENCODING_DECODING_TO_MESSAGETYPE | Decoding SMS to type [TYPE] |
| 3208 | INFO_ENCODING_DECODING_FAILED_PROCESSING_AS_BINARY | Decoding failed. Trying to decoding as standard binary message. |
| **3300** | **INFO_DRIVER_BASE** | |
| 3301 | INFO_DRIVER_CONNECTIONONLINE | Connection online. |
| 3302 | INFO_DRIVER_CONNECTIONOFFLINE | Connection offline. |
| 3303 | INFO_DRIVER_TCPCONNECTING | Connecting to [IP] |
| 3304 | INFO_DRIVER_TCPCONNECTED | Connected from [FROMIP] to [TOIP] |
| 3305 | INFO_DRIVER_TCPFORCEDISCONNECT | Disconnected from [HOST]:[PORT] |
| 3306 | INFO_DRIVER_LOGIN_SUCCESS | Login successful. |
| 3307 | INFO_DRIVER_LOGIN_FAILED | Login failed. [MSG] |
| 3308 | INFO_DRIVER_LOGIN_TIMEOUT | Login failed due to timeout. |
| 3309 | INFO_DRIVER_LOGIN_DISCONNECTED | Login canceled due to disconnect. |
| 3310 | INFO_DRIVER_LOGIN_STOPPED | Login canceled due to service provider connection being stopped. |
| 3321 | INFO_DRIVER_RECEIVED_MESSAGE | Message received on connection '[DRIVER]'. [MSG] |
| 3322 | INFO_DRIVER_SENDING_MESSAGE | Sending message (try [TRY]) [ENVSTR] using service provider connection '[DRIVER]' |
| 3323 | INFO_DRIVER_SENDING_MESSAGE_SUBMITTING_PDU | PDU ([PART]/[COUNT]) accepted by SMSC with message reference: [REF] |
| 3324 | INFO_DRIVER_SENDING_MESSAGE_CONSIDERINGSENT | Considering sent. Deleting message from outbox [MSG] |
| 3325 | INFO_DRIVER_SENDING_MESSAGE_BACKTOOUTBOX | Returning message to outbox [MSG] |
| 3326 | INFO_DRIVER_SENDING_MESSAGE_NOTSENT | Considering not sent. Deleting message from outbox [MSG] |

| | | |
|---|---|---|
| 3327 | INFO_DRIVER_SENDING_MESSAGE_SENT | Message successfully sent. Reference: [REF] |
| 3328 | INFO_DRIVER_SENDING_MESSAGE_PDU_SEQUENCE | Sending message with sequence number [SEQ] |
| 3329 | INFO_DRIVER_SENDING_MESSAGEPART_ACCEPTED | Message part sent as sequence [SEQNO] [ENVELOPE]was accepted by SMSC with callback reference id [CALLBACKID] |
| 3330 | INFO_DRIVER_SENDING_MESSAGEPART_NOTFOUND | Corresponding message for message part sent as sequence [SEQNO] was not found. It may have been deleted, because previous part was not accepted. Message part dropped. |
| 3331 | INFO_DRIVER_SENDING_USINGBACKUPROUTE | Sending message [ENVSTR] through backup route using service provider connection '[DRIVER]' |
| 3341 | INFO_DRIVER_GSMMODEM_SIMSMSC_MATCHES_CONFIGUREDSMSC | The address of SIM SMSC ([SIMSMSC]) and the address of SMSC specified in the configuration form ([SETSMSC]) are the same. No configuration is neccessary. |
| 3342 | INFO_DRIVER_GSMMODEM_SIMSMSC_WILLBEUSED | Address of SIM SMSC ([SIMSMSC]) will be used! |
| 3343 | INFO_DRIVER_GSMMODEM_USSD | Sending USSD request to network [USSD] |
| 3344 | INFO_DRIVER_GSMMODEM_REGISTRATIONINFO_UNAVAILABLE | GSM Network registration information is not available. |
| 3345 | INFO_DRIVER_GSMMODEM_PDUMODE | PDU mode is used. |
| 3346 | INFO_DRIVER_GSMMODEM_REGISTERED_TO_NETWORK | Good news! The GSM modem is registered to the GSM network. |
| 3347 | INFO_DRIVER_GSMMODEM_REGISTERING | The modem is currently searching for a GSM network. It is not connected. There is no signal. Check your antenna! |
| 3348 | INFO_DRIVER_GSMMODEM_MESSAGESTORES | The GSM modem and the SIM card have the following message stores: [STORES] |
| 3349 | INFO_DRIVER_GSMMODEM_NOPDUMODE | Text mode is used. To get full functionality, we recommend you to use PDU mode! |
| 3350 | INFO_DRIVER_GSMMODEM_COPS | GSM network information: [CODE] |
| 3351 | INFO_DRIVER_GSMMODEM_PINRESULT | Pin code status: [RESULT] |
| 3352 | INFO_DRIVER_GSMMODEM_DISABLEPIN | Please put the SIM card into a mobile handset and disable the requested code! |
| 3353 | INFO_DRIVER_GSMMODEM_QUICKRECONNECT | There were more then 5 consecutive errors. Ozeki NG will now reset the modem and try to send the message again. This might solve the problem. |
| 3354 | INFO_DRIVER_GSMMODEM_QUICKRECONNECT_FAKE | Ozeki NG will now try to send the message again. This might solve the problem. |
| 3355 | INFO_DRIVER_GSMMODEM_AUTODETECT_START | GSM Modem auto detection is starting at port [PORT]. |
| 3356 | INFO_DRIVER_GSMMODEM_AUTODETECT_END | GSM Modem auto detection has finished at port [PORT]. |
| 3357 | INFO_DRIVER_GSMMODEM_AUTODETECT_BAUDRATE | Port speed check ([PORTSPEED]) at port [PORT] is [SUCCESS]. |
| 3358 | INFO_DRIVER_GSMMODEM_AUTODETECT_MODEMNAME | Result of GSM Modem name check at port [PORT] is "[NAME]". |
| 3359 | INFO_DRIVER_GSMMODEM_AUTODETECT_SMSCENTRE | The SIM SMSC is "[SMSC]". |
| 3360 | INFO_DRIVER_GSMMODEM_AUTODETECT_PDU_SUPPORT | GSM Modem [SUPPORTING] PDU mode at port [PORT]. |
| 3361 | INFO_DRIVER_GSMMODEM_AUTODETECT_SENDAT | Command to GSM Modem: "[COMMAND]" at port [PORT]. |
| 3362 | INFO_DRIVER_GSMMODEM_AUTODETECT_ATRESPONSE | Response of GSM Modem: "[RESPONSE]" at port [PORT]. |
| 3363 | INFO_DRIVER_GSMMODEM_AUTODETECT_OPERATION_UNSUCCESSFUL | Autodetection of GSM Modem [OPERATION] was unsuccessful at port [PORT]. |
| 3364 | INFO_DRIVER_GSMMODEM_MMS_DOWNLOAD | Downloading MMS messages (try [TRY]). Download URL: [URL]. |
| 3365 | INFO_DRIVER_GSMMODEM_MMS_GPRSOPENING | Opening GPRS connection. |
| 3366 | INFO_DRIVER_GSMMODEM_MMS_GPRSCLOSING | Closing GPRS connection. |
| 3380 | INFO_DRIVER_SMPP_DELIVERYREPORT | Delivery report: [MSG] |
| 3381 | INFO_DRIVER_SMPP_DELIVERYID | Message reference id: [ID] |
| 3390 | INFO_DRIVER_UCP_OPTIONALFIELD_USED | User defined optional field is used for parameter '[FIELD]', with value '[VALUE]' |
| 3400 | INFO_DRIVER_PARLAYX_REGISTERRECEIVESERVICE | Registering callback service for incoming messages. Listening on URL: [URL] |
| 3401 | INFO_DRIVER_PARLAYX_REGISTERDELIVERYSERVICE | Registering callback service for delivery reports. Listening on URL: [URL] |
| 3410 | INFO_DRIVER_HTTPSERVER_MESSAGESAVEDTOOUTBOX | Message [MSGID] saved to outbox file [FILENAME]. |
| **3500** | **INFO_USER_BASE** | |
| 3501 | INFO_USER_CONNECTED | User enabled. |
| 3502 | INFO_USER_DISCONNECTED | User disabled. |
| 3510 | INFO_USER_MESSAGE_INCOMING | Incoming message for user '[USER]'. [ENVSTR] |
| 3511 | INFO_USER_MESSAGE_SENDING | Message accepted for delivery from user '[USER]'. [ENVSTR] |
| 3512 | INFO_USER_MESSAGE_SENT | Message successfully sent[OPERATOR]. [ENVSTR] |
| 3513 | INFO_USER_MESSAGE_NOTSENT | Message not sent. [REASON] [ENVSTR] |
| 3514 | INFO_USER_MESSAGE_DELIVERED | Message delivery is acknowledged by returned delivery reports. [ENVSTR] |
| 3515 | INFO_USER_MESSAGE_UNDELIVERED | Message delivery rejected. [REASON] [ENVSTR] |
| 3516 | INFO_USER_MESSAGE_DELIVERYINPROGRESS | Message delivery in progress ([CODE]) [REASON]. [ENVSTR] |
| 3517 | INFO_USER_MESSAGE_DELETED_SUCCESSFULPROC | Messsage removed from [QUEUE], because it was successfully forwarded to a client or processed by an application. [ENV] |
| 3518 | INFO_USER_MESSAGE_RESENDING | Could not send message to network. Trying to resend message (try [TRYCOUNT]). The reason of failure was [ERR]. The message is: [MSG]. |
| 3530 | INFO_USER_SQL_INSERT | Message received. Inserting it into database. Message: '[MSG]' |
| 3531 | INFO_USER_SQL_SELECT_MESSAGESFOUND | Outgoing message found in database (database id: [DBID]). [MSG] |
| 3532 | INFO_USER_SQL_UPDATE | Message [EVENT]. Updating correspondig record (database id: [DBID]) in database. Message: '[MSG]' |
| 3540 | INFO_USER_EMAIL_GENERAL | [PRE][MSG] |
| 3541 | INFO_USER_EMAIL_READINGMESSAGES_INTOMEMORY | Read uid of processed e-mails into memory from: '[MSG]' |
| 3542 | INFO_USER_EMAIL_READMESSAGES_INTOFILE | Write uid of processed e-mails from memory into : '[MSG]' |
| 3550 | INFO_USER_HTTPCLIENT_POST_REQUEST | Initiating HTTP [METHOD] request to [URL]. Data: [DATA]. |
| 3551 | INFO_USER_HTTPCLIENT_GET_REQUEST | Initiating HTTP [METHOD] request to [URL]. |
| 3552 | INFO_USER_HTTPCLIENT_RESPONSE | HTTP response received from [URL]. |
| 3553 | INFO_USER_HTTPCLIENT_REQUEST_ACCEPTED | Received HTTP 200 OK, request accepted. [URL]. |
| 3554 | INFO_USER_HTTPCLIENT_NORESPONSE | No response SMS was returned by the HTTP server. That is OK. |
| 3599 | INFO_USER_GENERAL_INFORMATION | API Information. [MSG] |