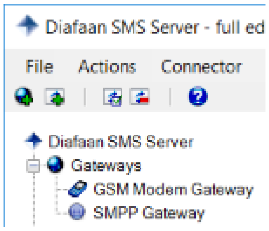
Diafaan SMS Server 4.8 Manual

Table of Contents

Welcome to Diafaan SMS Server

I

1 Welcome to Diafaan SMS Server



Diafaan SMS Server makes it easy to send and receive SMS text messages from and to a web server, email, SQL database or C#/Visual Basic .NET script.

Simply attach a GSM modem to your PC/server or set up a connection to an online SMS- or paging operator, use a wizard to add the SMS gateway and connections to the database, web server or email and your SMS gateway is ready for use.

Getting started

- Study Getting started to familiarize yourself with the basics of the program.
- Check out the **How to** pages for scripting examples and instructions for specific database software like Microsoft SQL Server (Express), MySQL, Oracle, PostgreSQL or Microsoft Access.

Advanced features

- Get an overview of all available Gateway and Connector features in the Reference section.
- See how to integrate with web applications using the HTTP API specifications.
- Send messages from the Windows command line with the Commandline client.

Getting started

II

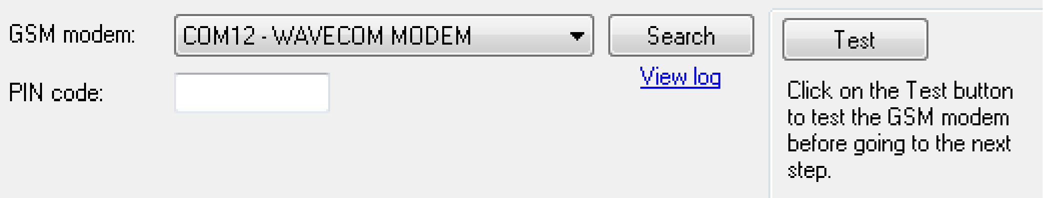
2 Getting started

After installing Diafaan SMS Server and starting up the program for the first time the installation wizard enables you to add an SMS gateway and the default HTTP web interface. You can create additional SMS gateways and Connectors later.

Enter the administrator password

The installation wizard asks for a password of the administrator (**admin)** account. This password is required to log in to the web application of Diafaan SMS Server.

Add a GSM Modem Gateway



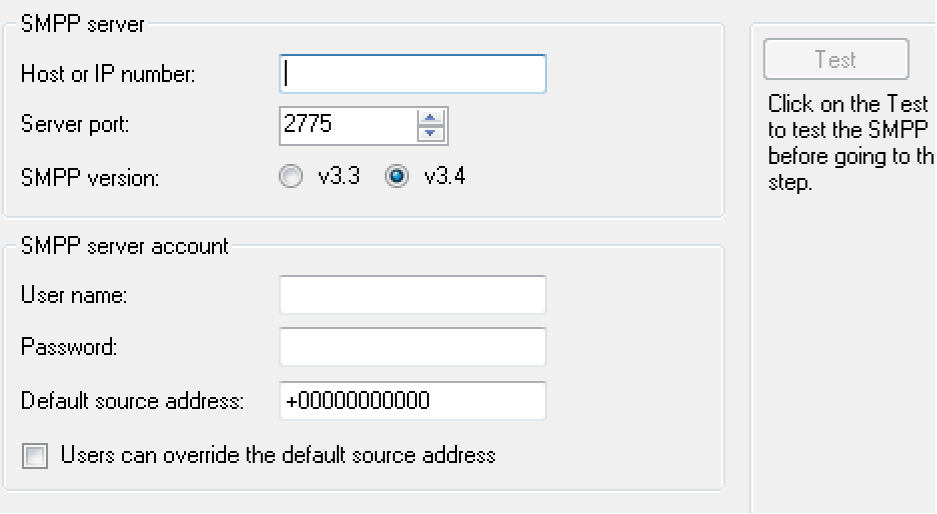
GSM modem: COM12 - WAVECOM MODEM

PIN code:

Click on the Test button to test the GSM modem before going to the next step.

- Click **Search** to get a list of all installed GSM modems.
- Select the GSM modem from the list.
- If the SIM card needs a PIN code, enter it in the PIN code field.
- Click **Test** to test the GSM modem and retrieve all information necessary to send and receive SMS messages with the modem.
- If required enter the GSM number and SMSC number of the SIM card.

or Add an SMPP Gateway



SMPP server

Host or IP number:

Server port: 2775

SMPP version: ○ v3.3 ● v3.4

SMPP server account

User name:

Password:

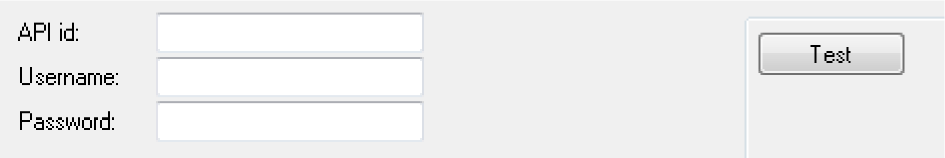
Default source address: +00000000000

□ Users can override the default source address

Click on the Test to test the SMPP before going to th step.

- Enter the SMPP host name or IP number, SMPP server port, user name and password.
- Click **Test** to test the connection to the SMPP server

or Add a Clickatell Gateway



API id:

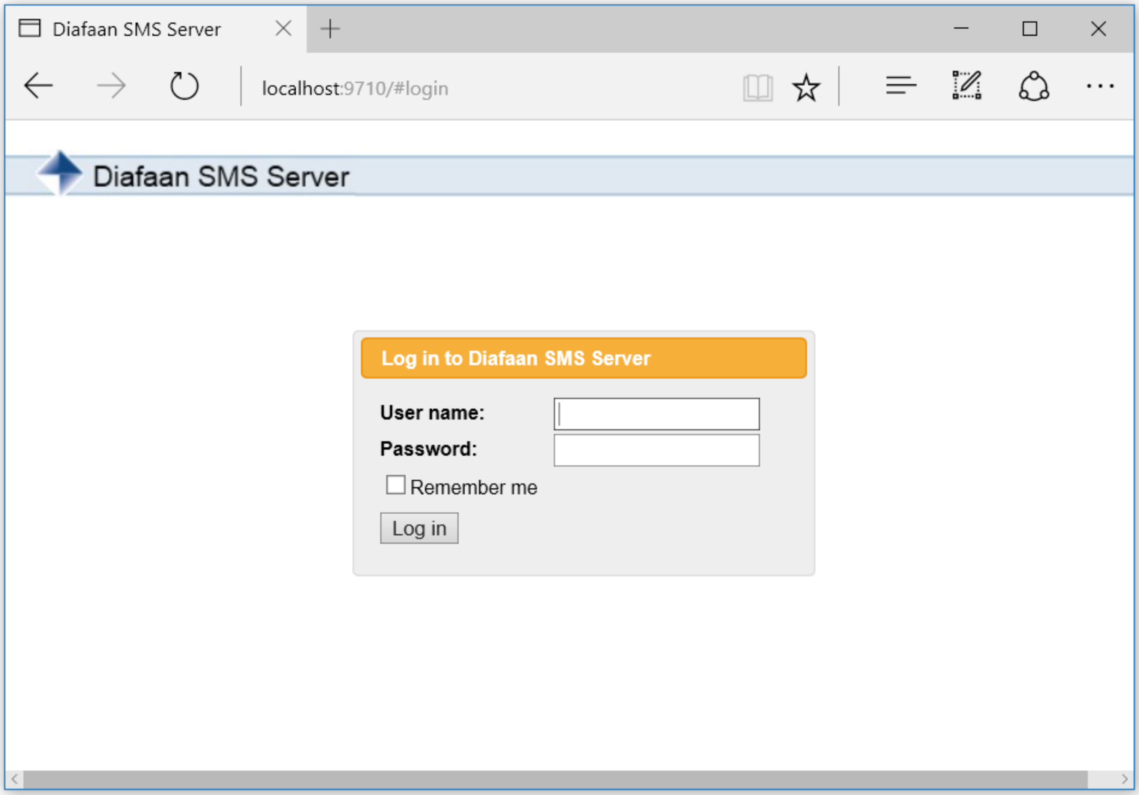
Username:

Password:

- Create a Clickatell HTTP account on the Clickatell website.
- Enter the Clickatell HTTP API id, user name and password.
- Click **Test** to test the connection to Clickatell

Send your first message

- Start a web browser and use the web application on http://localhost:9710/ to log in to Diafaan SMS Server.
- Log in to Diafaan SMS Server with the user name **admin** and the password you entered in the setup wizard.

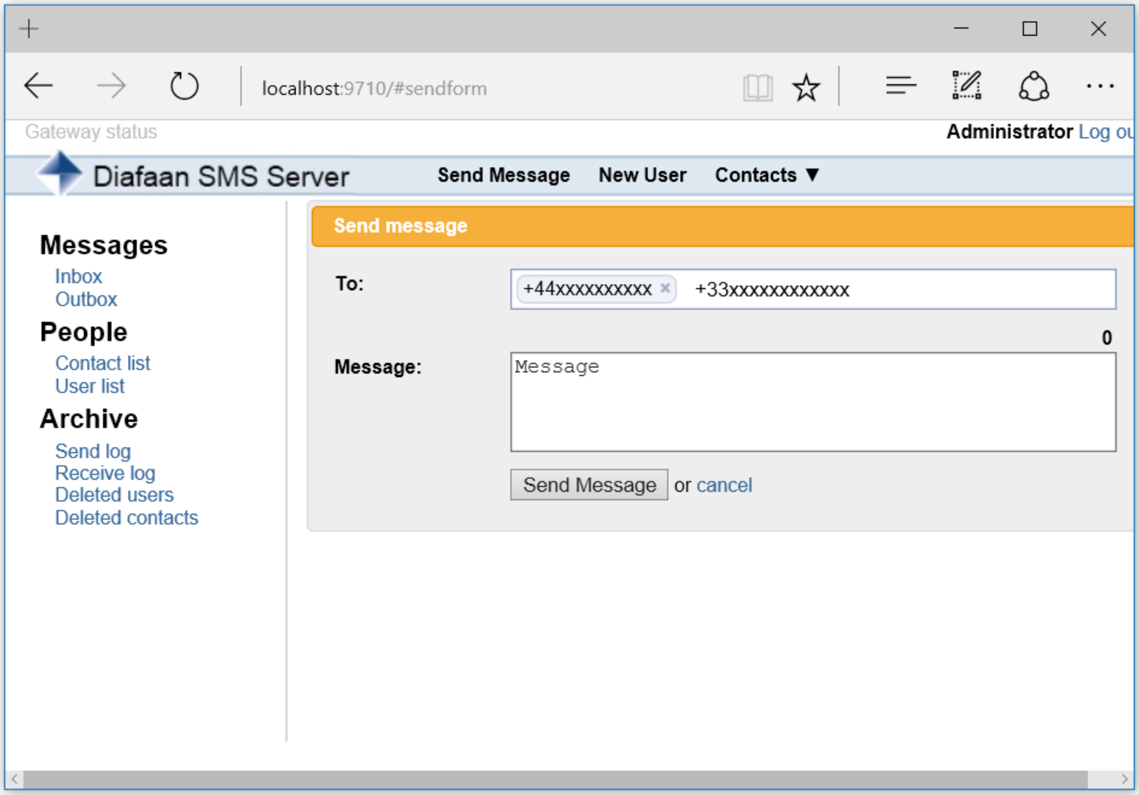


User name:

Password:

□ Remember me

- In the To field you can enter multiple mobile numbers, separated by comma's.
- **Important!** It is good practice to enter all mobile numbers in Diafaan SMS Server in international number format, starting with a '**+**'. Not all mobile operators accept numbers in other formats. Short numbers or other special numbers can be prefixed with a '**#**' (national address type) or an '*' (unknown address type).



To: +44xxxxxxxxxx × +33xxxxxxxxxxxx

0

Message: Message

2.1 Gateways and connectors

Gateways

Diafaan SMS Server uses gateways to interface with SMS and other messaging services. The following gateways are supported:

All editions:

- **GSM Modem gateway**
  Diafaan SMS Server can use GSM modems and SIM boxes to send and receive SMS messages. This includes a broad range of devices like the Sierra Wireless Fastrack, Cinterion, Wavecom, Siemens, Nokia, Huawei or Option GSM modems and a large number of normal GSM phones.

- **Clickatell gateway**
  Clickatell is a world leading communication provider specializing in bulk messaging services and SMS gateway connectivity. Diafaan SMS Server makes use of the Clickatell HTTP interface to send SMS messages.

- **Emulator gateway**
  The Emulator Gateway emulates sending and receiving SMS messages. This gateway can be used to test the configuration of Diafaan SMS Server without actually sending or receiving any SMS messages.

Basic- and full editions only:

- **SMPP gateway**
  SMPP is a high performance two-way SMS protocol that is supported by a large number of online SMS providers. Use the SMPP Gateway to send and receive SMS messages with an SMS operator over an Internet connection. The SMPP Gateway supports SMPP version 3.3 or 3.4.

- **Paging gateway**
  Send messages to pagers and GSM phones with an analog or ISDN modem or an Internet connection. The Paging Gateway supports the IXO/TAP and UCP 01 protocols over a modem connection and the SNPP, SMTP (email), WCTP and UCP 01 protocols over an Internet connection.

- **Scripting gateway**
  Use the scripting gateway to send and receive messages to messaging services like email, SMS or internet services. The scripting gateway is also used to change the recipient, message text etc. and to dispatch the message to a different gateway. The scripting gateway supports scripts in C# and Visual Basic .NET.

- **Hypermedia SMS PRO gateway**
  Send and receive SMS messages with the Hypermedia HG-7000. The Hypermedia HG-7000 SMS PRO gateway is a professional multi-port wireless SMS gateway device with support for 4 - 32 GSM modem modules.

- **HTTP gateway**
  Send and receive SMS messages with an online SMS service.

Connectors

Connectors are used to connect Diafaan SMS Server with other software. This can be a database, email system, web server and more. The following connectors are supported:

All editions:

- **Web connector**
  The Web connector uses a built-in web server to send SMS messages and query the status of previously sent messages. The web connector can also be used to post received messages to webpages. It has a built-in web application to make sending and receiving SMS messages very easy.

- **Email connector**
  Email is converted into SMS messages with a built-in SMTP server or with a connection to an external POP server. The email connector can also forward SMS messages to an email address.

- **SQL connector**
  The SQL connector provides the connection to the database you want to use. It supports three database tables, one for messages to send (MessageOut), one to store received messages (MessageIn) and a table to log sent or failed calls (MessageLog).

- **Scripting connector**
  Use a script to send SMS messages and process received messages. A simple script might send an SMS response to a received SMS message. The scripting connector supports scripts in C# and Visual Basic .NET.

- **Excel connector**
  The Excel Connector can import messages from spreadsheet files and export received messages and logs to .CSV (comma separated values) files.

Basic- and full editions only:

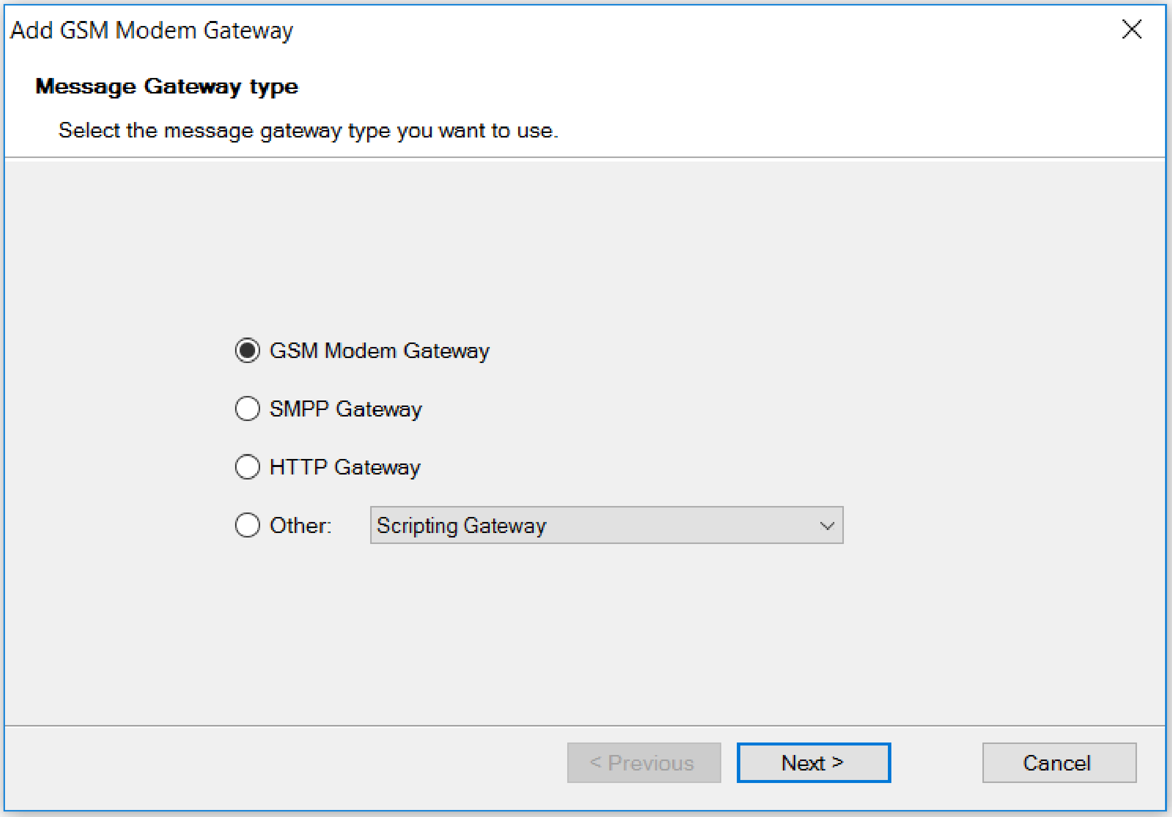
- **SMPP connector**
  The SMPP connector accepts incoming connections from one or more SMPP clients. The SMPP connector supports SMPP protocol versions 3.3, 3.4 and 5.0.

- **Protocol connector**
  Use legacy paging clients that support the TAP/IXO, SNPP, UCP or ESPA 4.4.4 protocols to send messages via Diafaan SMS Server.

2.2 Add a gateway

Add gateway wizard

Start the **Add gateway** wizard and select the gateway type. If the **Add gateway** option is not available in the **Actions** list, you have to remove one or more gateways first.



Add GSM Modem Gateway

Message Gateway type

Select the message gateway type you want to use.

● GSM Modem Gateway

○ SMPP Gateway

○ HTTP Gateway

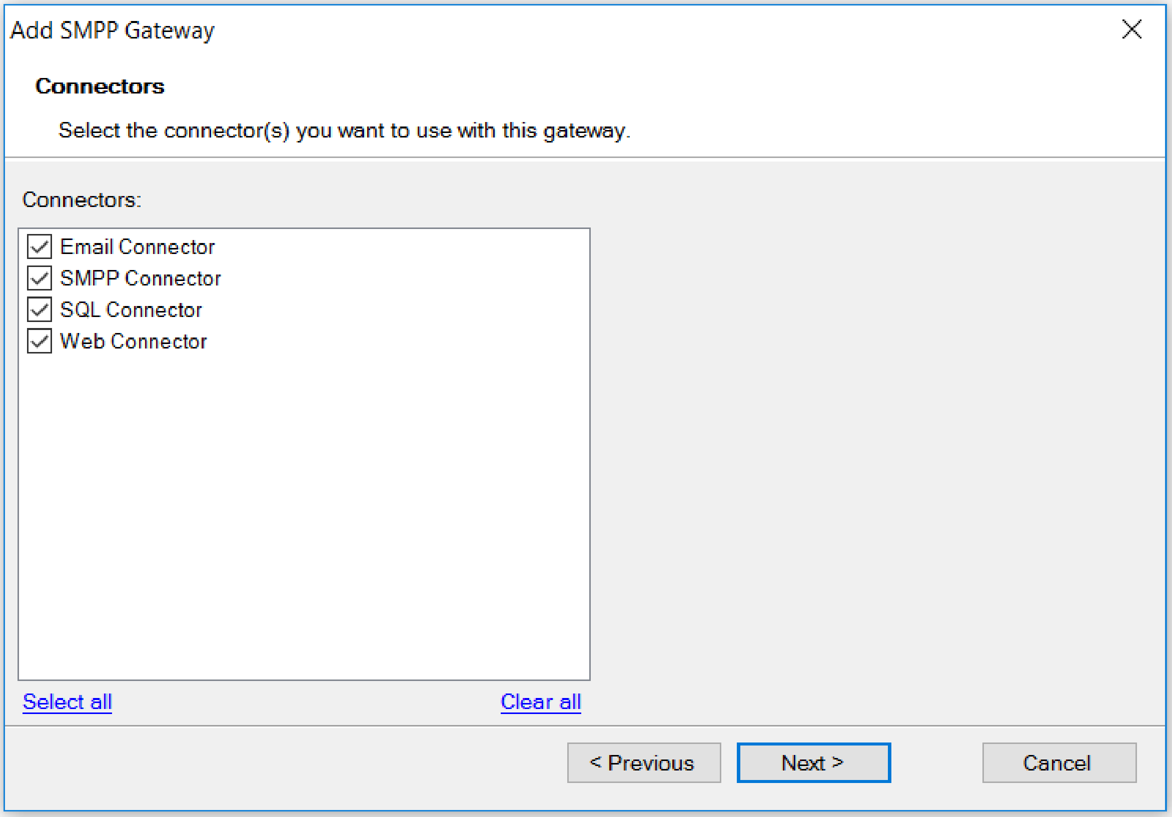
○ Other: Scripting Gateway

Select one of the available gateway

- GSM Modem gateway
- HTTP gateway (basic and full editions only)
- SMPP gateway (basic and full editions only)
- Clickatell gateway
- Emulator Gateway
- Paging gateway (basic and full editions only)
- Scripting gateway (basic and full editions only)
- Hypermedia SMS PRO gateway (basic and full editions only)

Assign the connectors

If you have installed more than one connector you can select the connectors you want to use with the new gateway.



Add SMPP Gateway

Connectors

Select the connector(s) you want to use with this gateway.

Connectors:

☑ Email Connector
☑ SMPP Connector
☑ SQL Connector
☑ Web Connector

2.2.1 Add a GSM Modem gateway

Prepare the SIM card

It is strongly advised to remove the PIN code from the SIM card before you use it with the GSM modem. You can enter the PIN code in Diafaan SMS Server and it will use this before logging on to the GSM network. However, if you provide the wrong PIN code to the software or change the SIM card afterwards the SMS software might repeatedly try to enter the wrong PIN code, resulting in a blocked SIM card. When this happens you need to reset the SIM card on a GSM phone using the PUC (Personal Unblocking Code, also known as PUK) you got with the SIM card.

Search modem

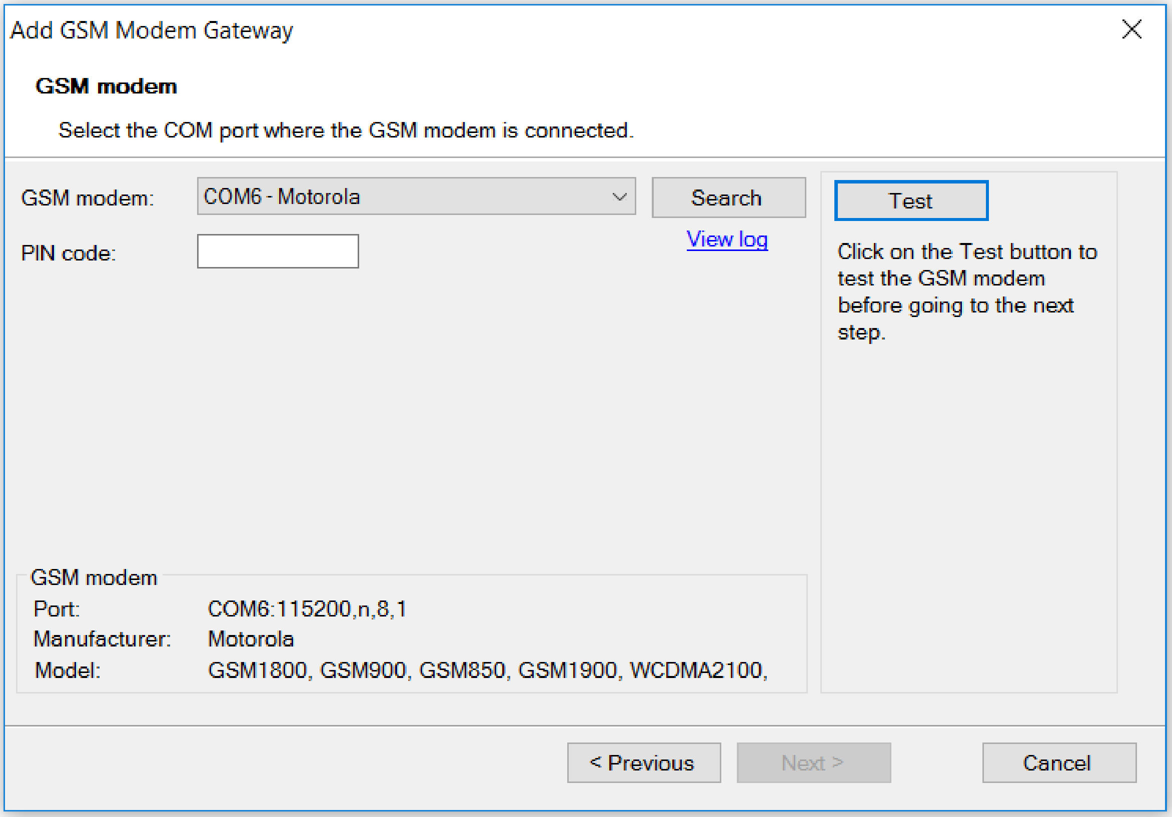
Click on the **Search** button to find all the GSM modems attached to the PC. If you already know the serial port where your GSM modem is connected you can skip this step, just select the serial port and click on the **Test** button to test the GSM modem.

TCP/IP modems

The GSM Modem Gateway can directly use GSM modems connected over an Internet TCP/IP connection, without the need for additional drivers. The GSM Modem Gateway cannot find TCP/IP modems automatically, to add the modem manually select **TCP/IP** from the GSM modem port list and enter the host name or IP number and the attached port number. The GSM Modem Gateway can use a raw TCP/IP connection or a Telnet connection to the modem.

Test the GSM modem

Before going to the next wizard page you have to click the **Test** button to test the GSM modem. The wizard will query the GSM Modem to find the SMSC number, the GSM number and check if a PIN code is required. If one of this numbers is not found you can enter them manually.



GSM modem

Select the COM port where the GSM modem is connected.

GSM modem: COM6 - Motorola

PIN code:

Click on the Test button to test the GSM modem before going to the next step.

GSM modem

Port: COM6:115200,n,8,1
Manufacturer: Motorola
Model: GSM1800, GSM900, GSM850, GSM1900, WCDMA2100,

Advanced options

Sometimes it is necessary to change an advanced parameter in order to get the GSM Modem to work. You can do this later in the **GSM Modem Gateway Properties** dialog box. If the GSM modem is not found you can still add the GSM Modem gateway by finishing the wizard.

2.2.2 Add an HTTP gateway

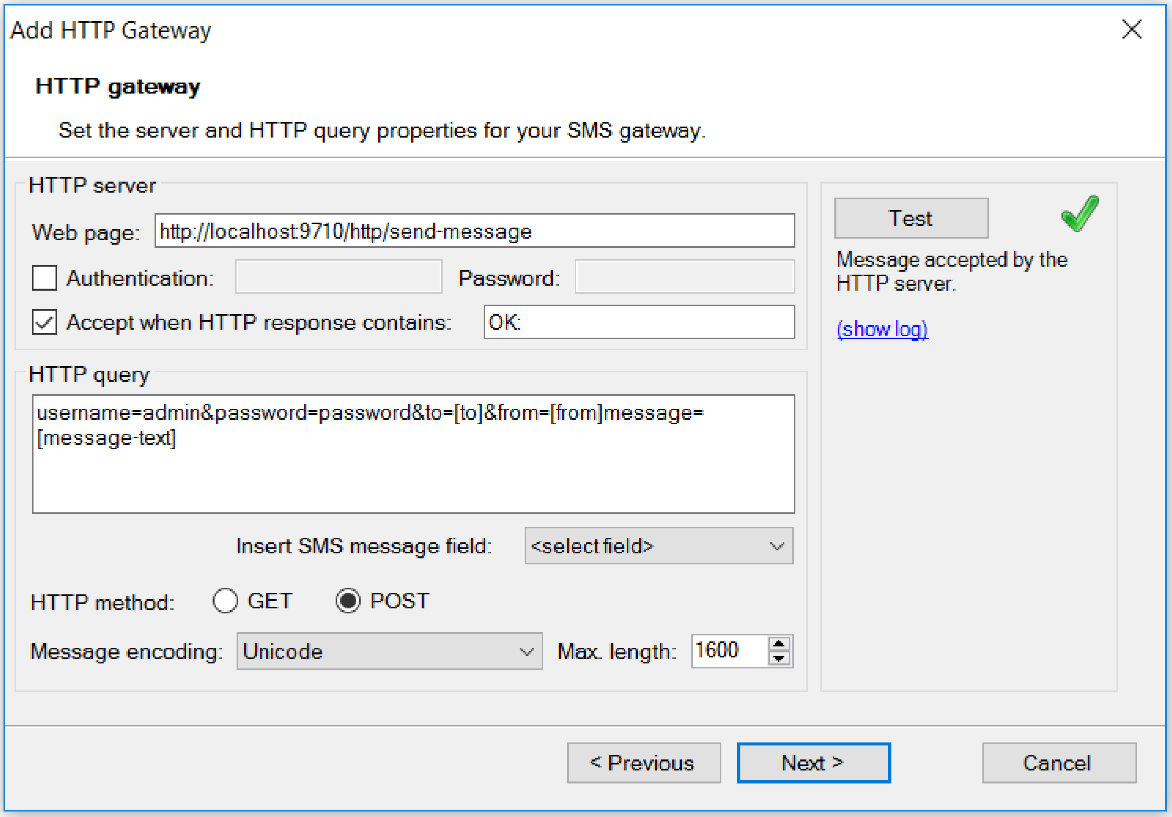
Only available in Diafaan SMS Server - basic and full editions.

The HTTP gateway sends messages to a web server with an HTTP GET or POST call. It supports sending messages to a generic HTTP server but it also has specific support to send and receive messages with the following SMS services:

- BulkSMS
- ClickSend
- Clickatell
- Esendex
- MessageBird
- Vonage/Nexmo
- Twilio
- Wassenger (WhatsApp)

Add a generic HTTP gateway

The generic HTTP gateway can send messages to the HTTP server of most SMS service providers. But a generic HTTP gateway is send-only, it does not support receiving messages or status updates (receipts).



Set the server and HTTP query properties for your SMS gateway.

HTTP server

Web page: http://localhost:9710/http/send-message

☐ Authentication: Password:

☑ Accept when HTTP response contains: OK

HTTP query

username=admin&password=password&to=[to]&from=[from]message=[message-text]

Insert SMS message field: <select field>

HTTP method: ○ GET ● POST

Message encoding: Unicode Max. length: 1600

✓ Message accepted by the HTTP server.

Web page
The full HTTP address of the HTTP server plus the path of the SMS service of the HTTP API.

Authentication
If the HTTP service requires Basic Access Authentication, you can enter the user name and password here.

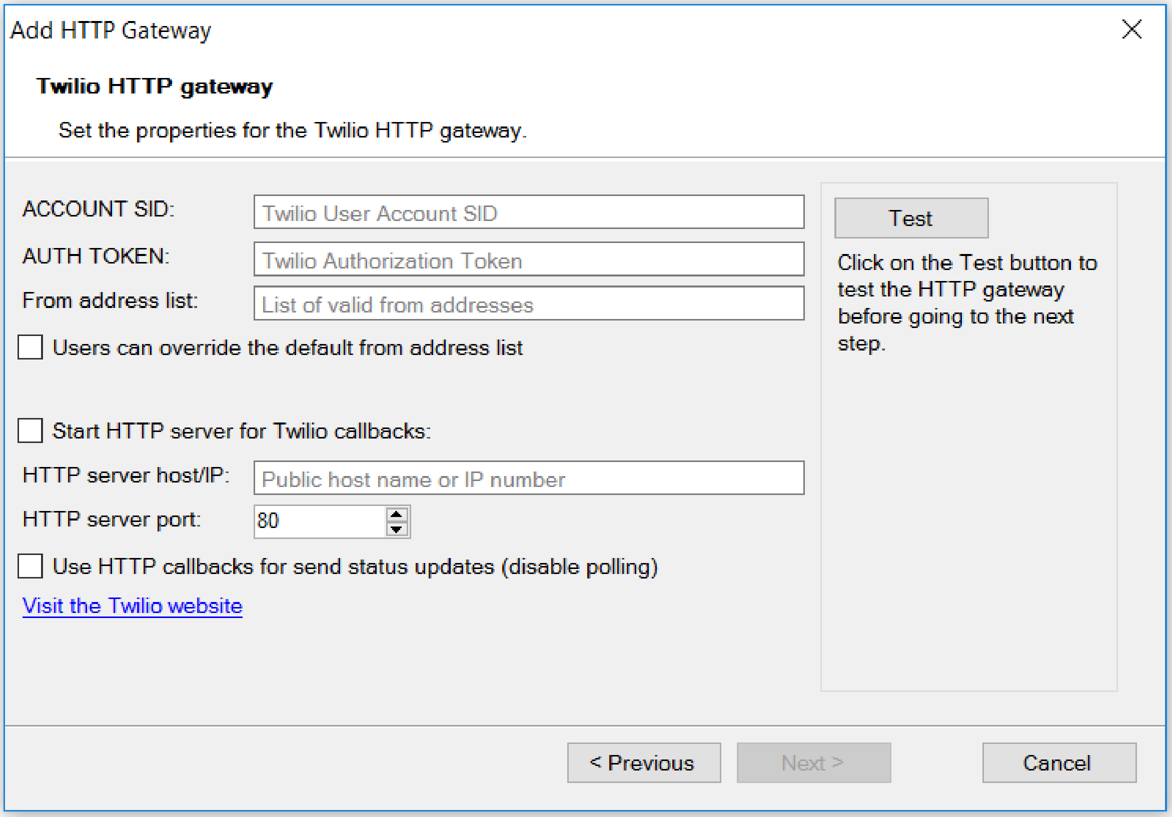
Accept when HTTP response contains
Regard the message only as sent when the text of the HTTP response matches the given text. This field can also contain a regular expression when the content starts with the text "**regex:**".

HTTP query
The query part of the URI when the GET HTTP method is used or the body content of the HTTP request when the POST method is used. The content type for the POST method is **application/x-www-form-urlencoded**.

Message encoding
Select **Unicode** to encode the message-text field as the default URL encoded text for the full Unicode character set, **GSM default alphabet** replaces the character that do not belong in the GSM default alphabet with question marks and **ASCII** only allows basic ACSII characters.

Add a specific HTTP gateway

When you want to use the HTTP API of Twilio, Nexmo, Clickatell or MessageBird to send and/or receive messages, you can select the desired gateway from the list and you will be presented with the options that are appropriate for the selected SMS service.



Add HTTP Gateway

Twilio HTTP gateway

Set the properties for the Twilio HTTP gateway.

ACCOUNT SID: Twilio User Account SID
AUTH TOKEN: Twilio Authorization Token
From address list: List of valid from addresses
□ Users can override the default from address list

□ Start HTTP server for Twilio callbacks:
HTTP server host/IP: Public host name or IP number
HTTP server port: 80
□ Use HTTP callbacks for send status updates (disable polling)

Visit the Twilio website

Click on the Test button to test the HTTP gateway before going to the next step.

The properties for each SMS service are different but most services require a user name and a password. There are some settings that are common for different SMS services:

From address list
To prevent number spoofing and spam, most SMS services enforce limits to the 'From' number or alphanumeric sender ID that can be used when a message is sent. In the From address list you can enter a list of valid from addresses, separated by commas, semicolons or spaces. The first number or alphanumeric sender ID in the list is used as the default number. All the other numbers/alphanumeric sender IDs in the list are only used when the message has the identical From number.

Alphanumeric sender IDs can be double quoted when they contain a comma, semicolon or space character, e.g. **+44xxxxxxxxxx,"my number",+441xxxxxxxxxx**.

Users can override the default from address list
When the SMS service does not enforce limits to the from address, or when you are certain that the from address in the messages is always correct, this option will use the from address in the message without checking it against the from address list.

Start HTTP server
This option starts an HTTP server to accept HTTP callbacks from the SMS service. Most SMS service providers send HTTP callbacks for incoming SMS messages and for delivery report receipts when this is enabled in the settings on the website of the SMS service provider.

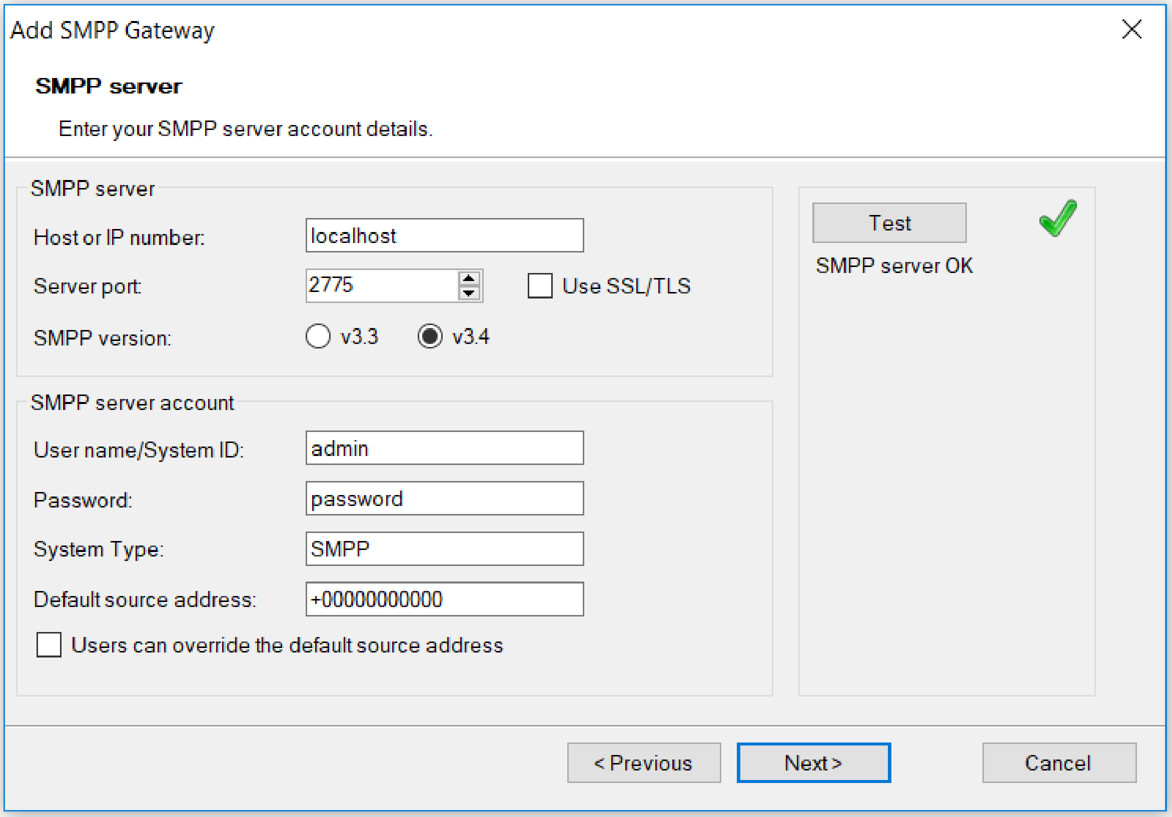
When your computer runs in a LAN network behind a NAT router, you have to create a port forwarding in the NAT router to allow HTTP callbacks from the SMS service provider to reach the HTTP server on the local network.

Accept HTTP callbacks for delivery reports (disable polling)
Normally, the HTTP gateway uses polling to retrieve the status of previously sent messages. With polling, there is no need to start an HTTP server and this is easier to set up than using HTTP callbacks for delivery reports. But if you use the HTTP server to receive SMS messages, it is better to use HTTP callbacks for status updates (delivery reports) and disable status polling.

2.2.3 Add an SMPP gateway

Only available in Diafaan SMS Server - basic and full editions.

SMPP is a high performance SMS protocol that is supported by a large amount of online SMS providers, SMPP stands for Short Message Peer to Peer protocol. In the SMPP server setup wizard you can enter the account information supplied by your SMS provider.



Add SMPP Gateway

SMPP server

Enter your SMPP server account details.

SMPP server

Host or IP number: localhost

Server port: 2775 □ Use SSL/TLS

SMPP version: ○ v3.3 ● v3.4

SMPP server account

User name/System ID: admin

Password: password

System Type: SMPP

Default source address: +00000000000

□ Users can override the default source address

SMPP server OK ✓

Source address

Some SMPP operators require that you send a source address or sender id with each message. The default source address will be used for each message where the user does not provide a source address. If the SMPP operator only allows a limited number of valid source addresses it is possible to disable user specific source addresses or to add a list of valid source addresses in the advanced settings.

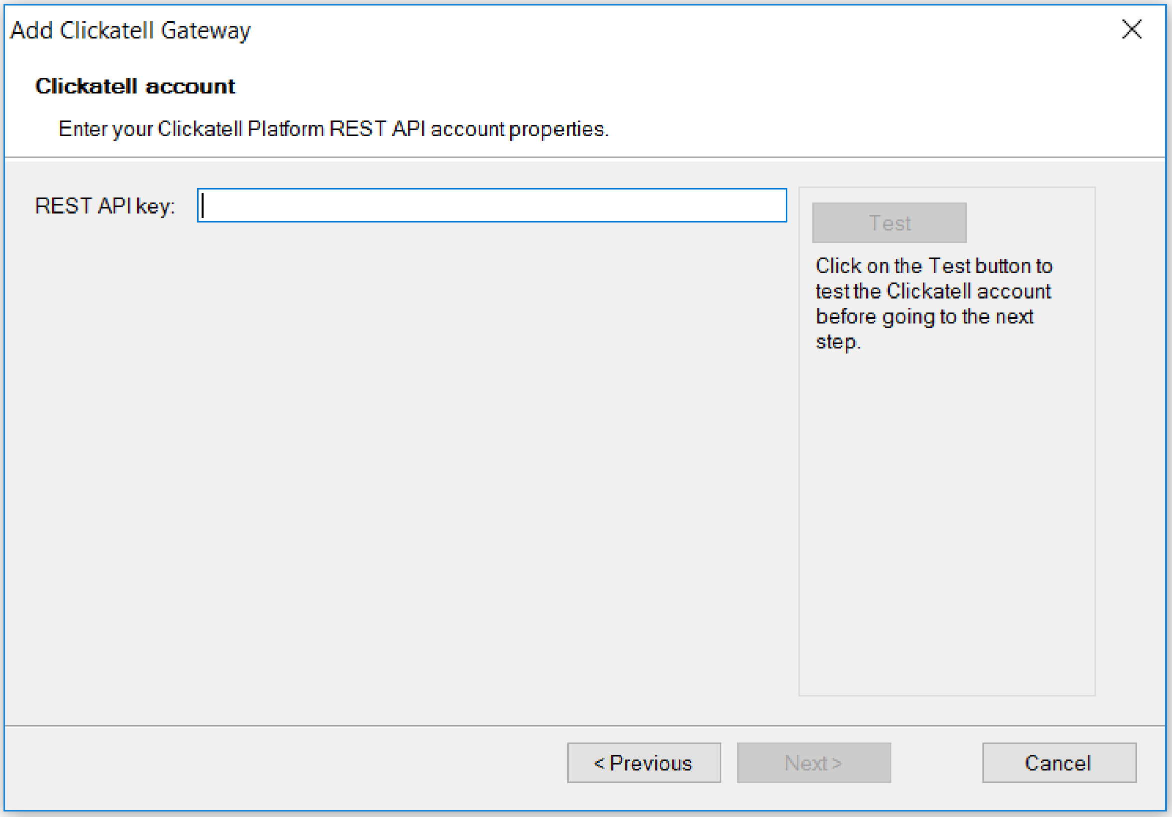
Performance

After initial installation, the SMPP gateway uses a conservative setting that is compatible with most SMPP servers. If your SMS provider supports advanced features like asynchronous (windowed) operation enabling the AsyncModeWindowSize option in the advanced settings can increase the sending speed considerably.

2.2.4 Add a Clickatell gateway

Get a Clickatell Platform REST API id

After creating a Clickatell account on the Clickatell website you also have to create a new REST API integration and switch the integration on. You then get a REST API key that you use in the Clickatell gateway to access your Clickatell account.



Add Clickatell Gateway

Clickatell account

Enter your Clickatell Platform REST API account properties.

REST API key:

Click on the Test button to test the Clickatell account before going to the next step.

Clickatell Communicator / Central API

If you use an older Communicator / Central Clickatell account you can still use that account with Diafaan SMS Server. The Communicator / Central API uses an (HTTP) API id, a username and a password to identify your account.

2.2.5 Add a Paging Gateway

Only available in Diafaan SMS Server - basic and full editions.

With Diafaan SMS Server you can send messages to pagers and GSM phones with an analog or ISDN modem, a direct serial port connection or an Internet connection. The Paging Gateway supports the IXO/TAP and UCP 01 protocols over a modem or serial port connection and the SNPP, SMTP (email), WCTP and UCP 01 protocols over an Internet connection.

Multiple services

Each Paging Gateway can contain multiple paging or SMS services, you can add new services to the Paging Gateway in the Paging Gateway Properties. Services that use the same modem port must be grouped in one Paging Gateway, other services can also be combined in the same Paging Gateway but this is not necessary.

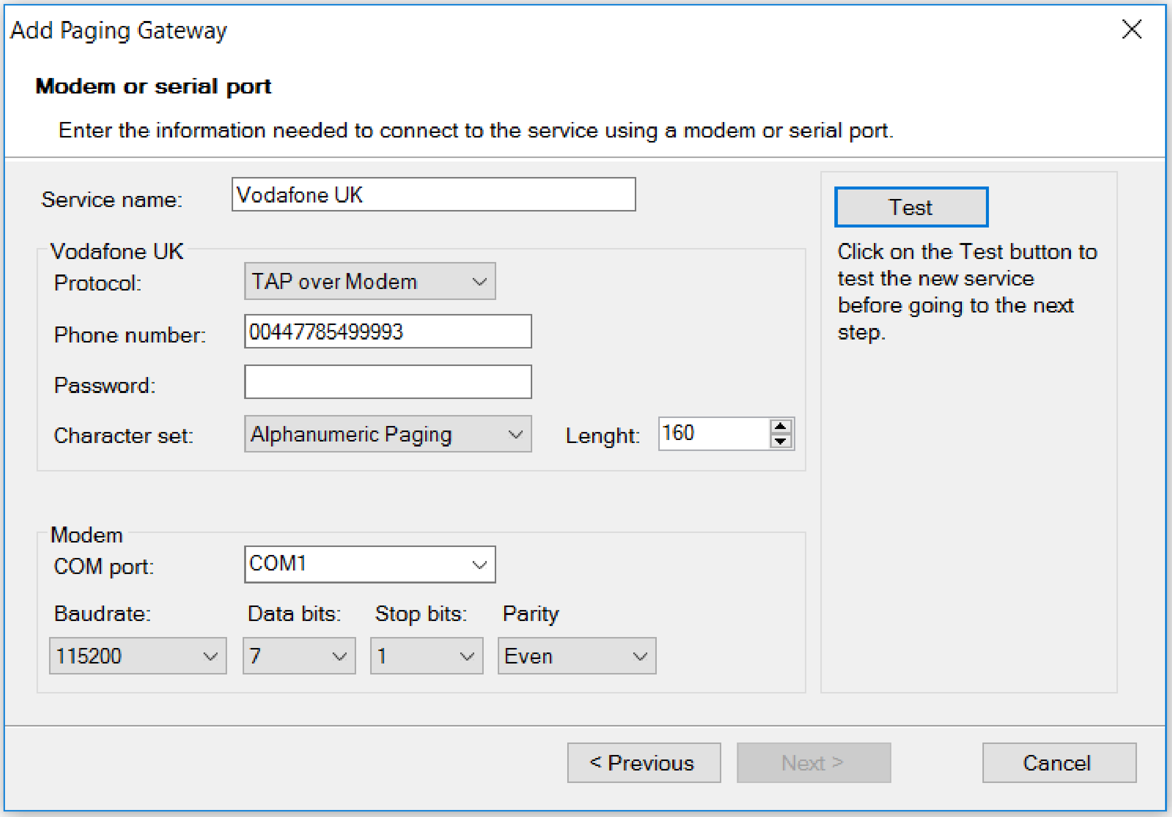
Find the service settings

Before you set up the connection to the paging service you have to find the proper settings for the service you want to add. You can get the proper settings from your paging company or mobile operator.

Supported protocols

Protocol	*Connection*	*Description*
IXO/TAP	Modem or serial port	IXO/TAP is an industry standard and is the predominant communications protocol for paging over a modem connection in the US, Canada, South America, Australia, and the majority of Europe.
UCP 01	Internet, modem or serial port	Universal Computer Protocol (UCP), is a protocol primarily used to connect to short message service centres (SMSCs) for mobile telephones.
SNPP	Internet	Simple Network Paging Protocol (SNPP) is a protocol that is used to send messages to pagers over the Internet
SMTP (email)	Internet	Email is a widely used method to send messages to pagers and mobile phone. Since the email system is not time sensitive it is not ideal for use with urgent paging or SMS messages. The SMTP service can also be used to send messages to normal email users.
WCTP	Internet	Wireless Communication Transfer Protocol (WCTP) is a protocol that supports sending messages to pagers and mobile phones over a plain (http://)or secured (https://) HTTP connection.

Add a modem service



Add Paging Gateway

Modem or serial port

Enter the information needed to connect to the service using a modem or serial port.

Service name: Vodafone UK

Vodafone UK

Protocol: TAP over Modem

Phone number: 00447785499993

Password:

Character set: Alphanumeric Paging

Lenght: 160

Modem

COM port: COM1

Baudrate:	Data bits:	Stop bits:	Parity
115200	7	1	Even

Click on the Test button to test the new service before going to the next step.

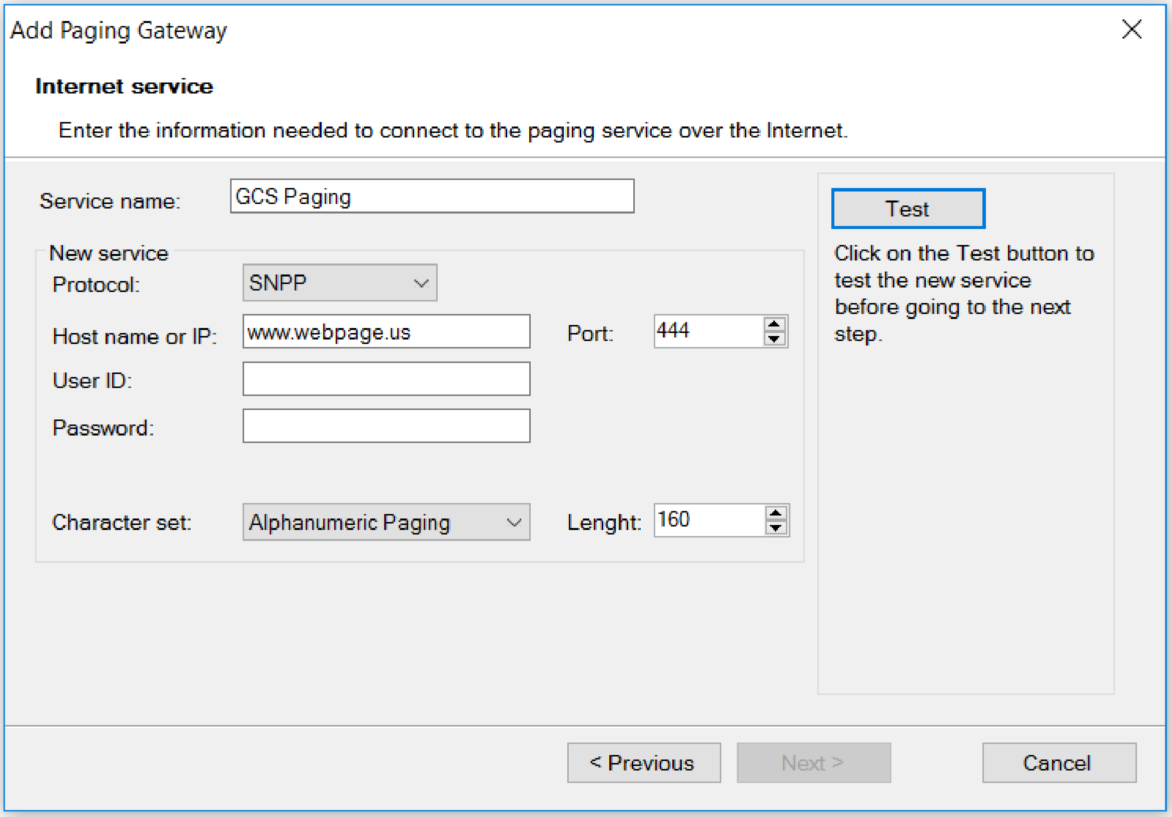
To add a new Paging Gateway with a modem dial-in service just provide the correct protocol, phone number of the service, an optional password and the modem properties. Then click on the Test button to send a test message to the service and see if the settings you entered are correct.

Modem settings
Most TAP services use a connection with 7 data bits, 1 stop bit and even parity.
Most UCP services use a connection with 8 data bits, 1 stop bit and no parity.

If the connection to the pager service does not work and additional modem settings are required to set up the connection, you can change them later in the Paging Gateway Properties.

Character set
Most paging services will accept basic ASCII characters only, for those services you can choose Alphanumeric Paging as the character set. Numeric Paging can be used if you want the service to send messages to numeric pagers only. Some SMS dial-in services accept the full SMS Default Alphabet, if you are unsure if your SMS service uses the SMS Default Alphabet, you can test this by sending an email address to the pager or mobile phone. If the "@" character is shown correctly on the pager or mobile phone you chose the right character set. In the Paging Gateway Properties you can specify additional character sets for some protocols.

Add an Internet service



Add Paging Gateway

Internet service

Enter the information needed to connect to the paging service over the Internet.

Service name: GCS Paging

New service

Protocol: SNPP

Host name or IP: www.webpage.us

Port: 444

User ID:

Password:

Character set: Alphanumeric Paging

Lenght: 160

Click on the Test button to test the new service before going to the next step.

To add a new Paging Gateway with an Internet service just provide the correct protocol, host name or URL of the service and an optional user id and password. Then click on the Test button to send a test message to the service and see if the settings you entered are correct.

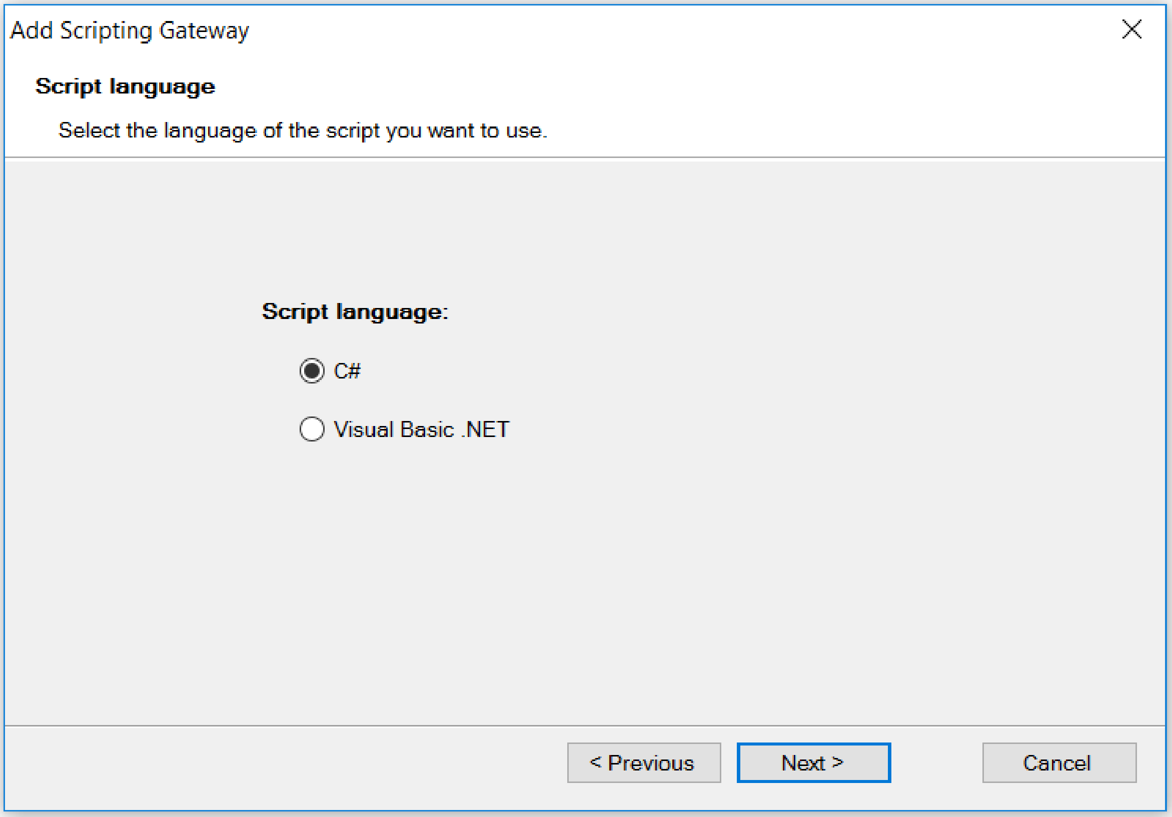
If the connection to the pager service does not work and additional settings are required to set up the connection, you can change them later in the Paging Gateway Properties.

2.2.6 Add a Scripting gateway

Only available in Diafaan SMS Server - basic and full editions.

Select scripting language

Diafaan SMS Server supports two scripting languages, Visual Basic .NET or C#. After adding the scripting gateway a default skeleton script is installed that provides the basic messaging functions. This skeleton script can be adapted in the **Scripting Gateway Properties** dialog.



Add Scripting Gateway

Script language

Select the language of the script you want to use.

Script language:

● C#

○ Visual Basic .NET

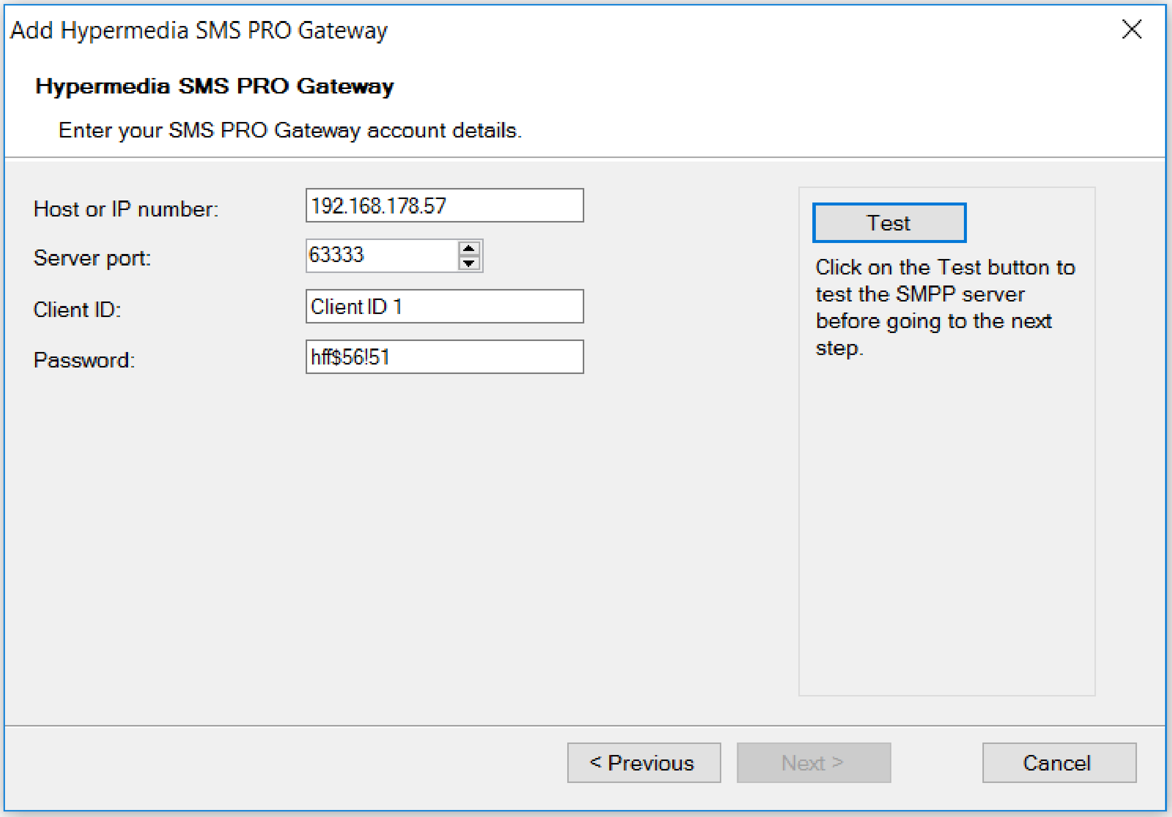
2.2.7 Add a Hypermedia SMS PRO gateway

Only available in Diafaan SMS Server - basic and full editions.

Hypermedia systems provides a range of wireless voice and SMS hardware gateways aimed at the banking sector, health care, manufacturing, governments, mobile service operators and other demanding markets. The Hypermedia SMS PRO Gateway is a professional multi-port wireless SMS gateway device with support for 4 - 32 ports.

Client ID

The Client ID can be freely chosen. If another client logs in with the same ID as an existing client then the new client will cause the disconnection of the old client and a message will be sent to each indicating this, therefore it is recommended that each new client that connects to the gateway should use a unique ID. Note that if no ID is given, a unique ID will be generated for the client by the gateway.



Add Hypermedia SMS PRO Gateway

Hypermedia SMS PRO Gateway

Enter your SMS PRO Gateway account details.

Host or IP number: 192.168.178.57

Server port: 63333

Client ID: Client ID 1

Password: hff$56!51

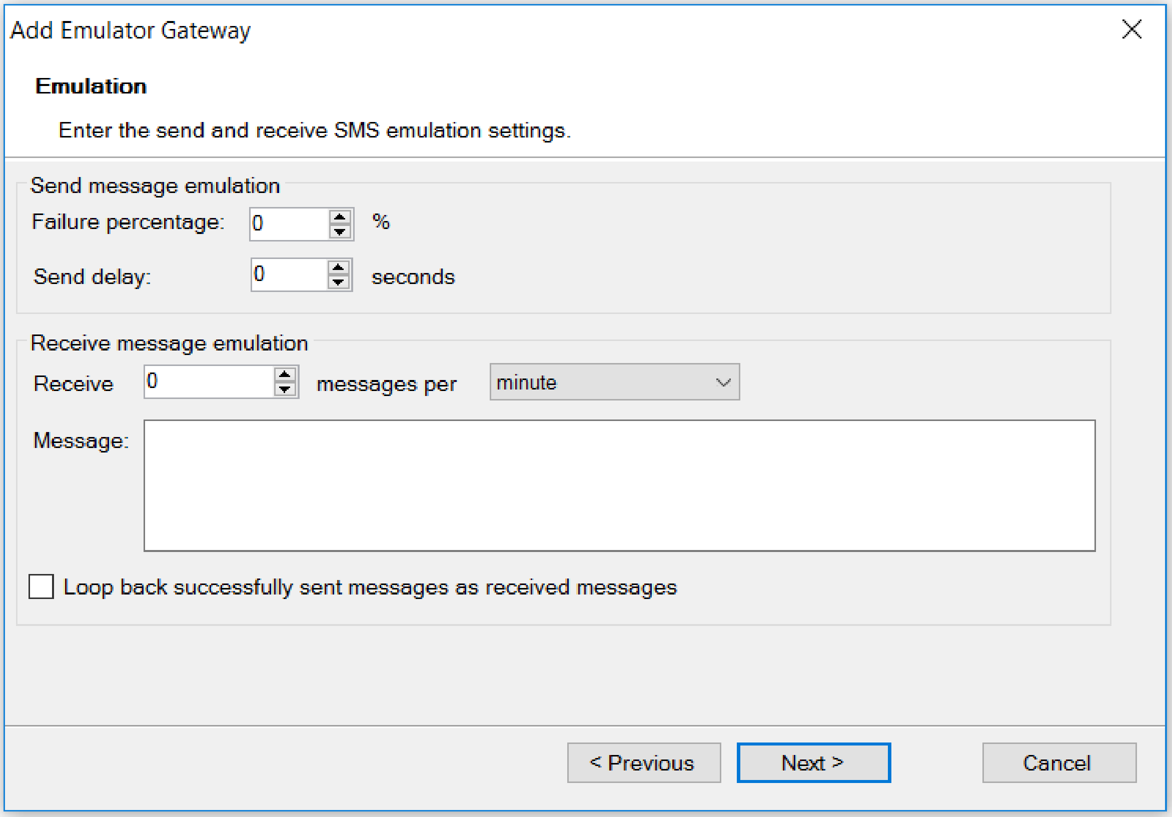
Click on the Test button to test the SMPP server before going to the next step.

SIM card routing

The SIM card routing options can be used to assign a specific SIM card in the Hypermedia SMS PRO Gateway to this Gateway. It is also possible to assign a GSM number to the SIM card that is used as the **To** field for messages that are received on this SIM card. In the Hypermedia SMS PRO gateway properties you can assign more than one SIM card to the Gateway.

2.2.8 Add an Emulator Gateway

The Emulator Gateway emulates sending and receiving SMS messages. This gateway can be used to test the configuration of Diafaan SMS Server without actually sending or receiving any SMS messages.



Add Emulator Gateway

Emulation

Enter the send and receive SMS emulation settings.

Send message emulation

Failure percentage: 0 %

Send delay: 0 seconds

Receive message emulation

Receive 0 messages per minute

Message:

□ Loop back successfully sent messages as received messages

Failure percentage

The percentage of messages that are flagged as failed by the Emulator Gateway, the messages fail with error code 1640 and error text "Randomly generated error".

Send delay

The minimum number of seconds between each emulated message.

Receive ... messages per ...

If this value is larger than '0', the Emulator Gateway generates one or more received SMS messages every second, minute, hour or day.

Message

The text of the emulated received messages.

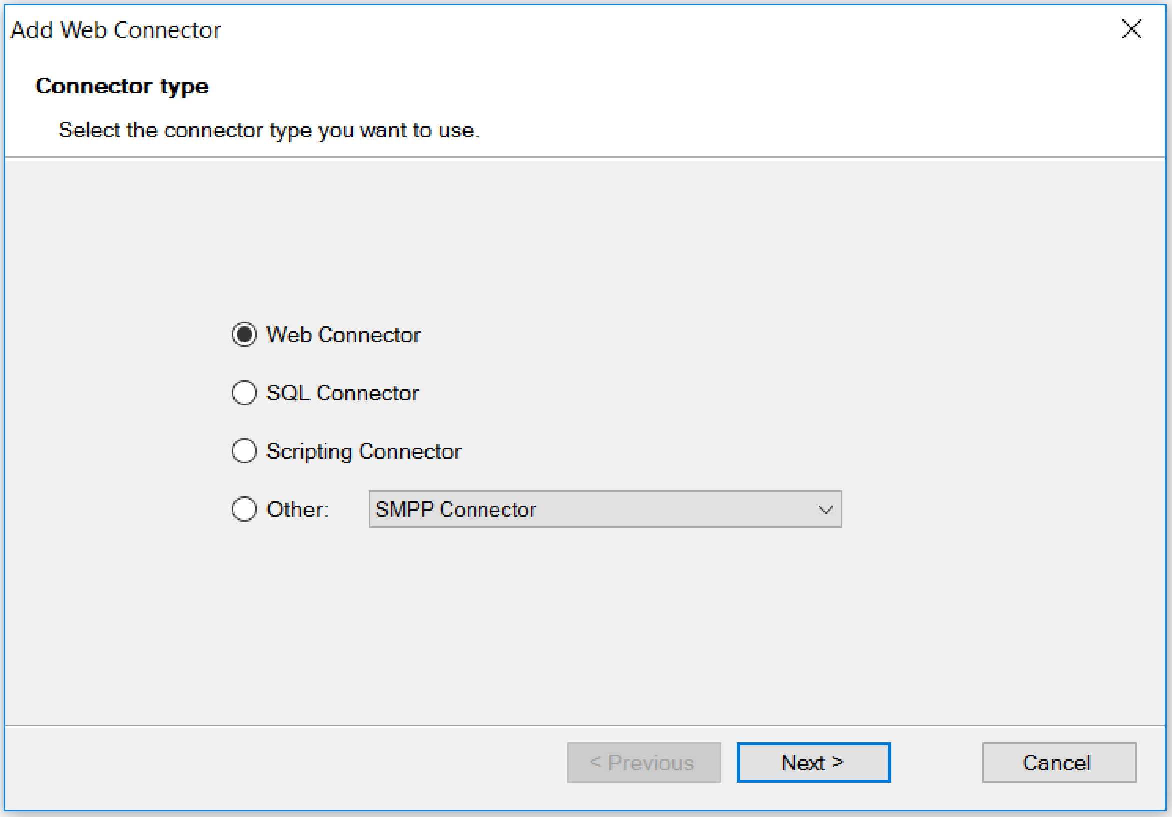
Loop back successfully sent messages as received messages

Convert all messages that are flagged as 'sent' by the Emulator Gateway to a received SMS message with the same message properties.

2.3 Add a connector

Add connector wizard

Start the **Add connector** wizard and select the gateway type. If the **Add connector** option is not available in the **Actions** list, you will have to remove one or more connectors first.



Add Web Connector

Connector type

Select the connector type you want to use.

● Web Connector

○ SQL Connector

○ Scripting Connector

○ Other: SMPP Connector

Select one of the available connectors

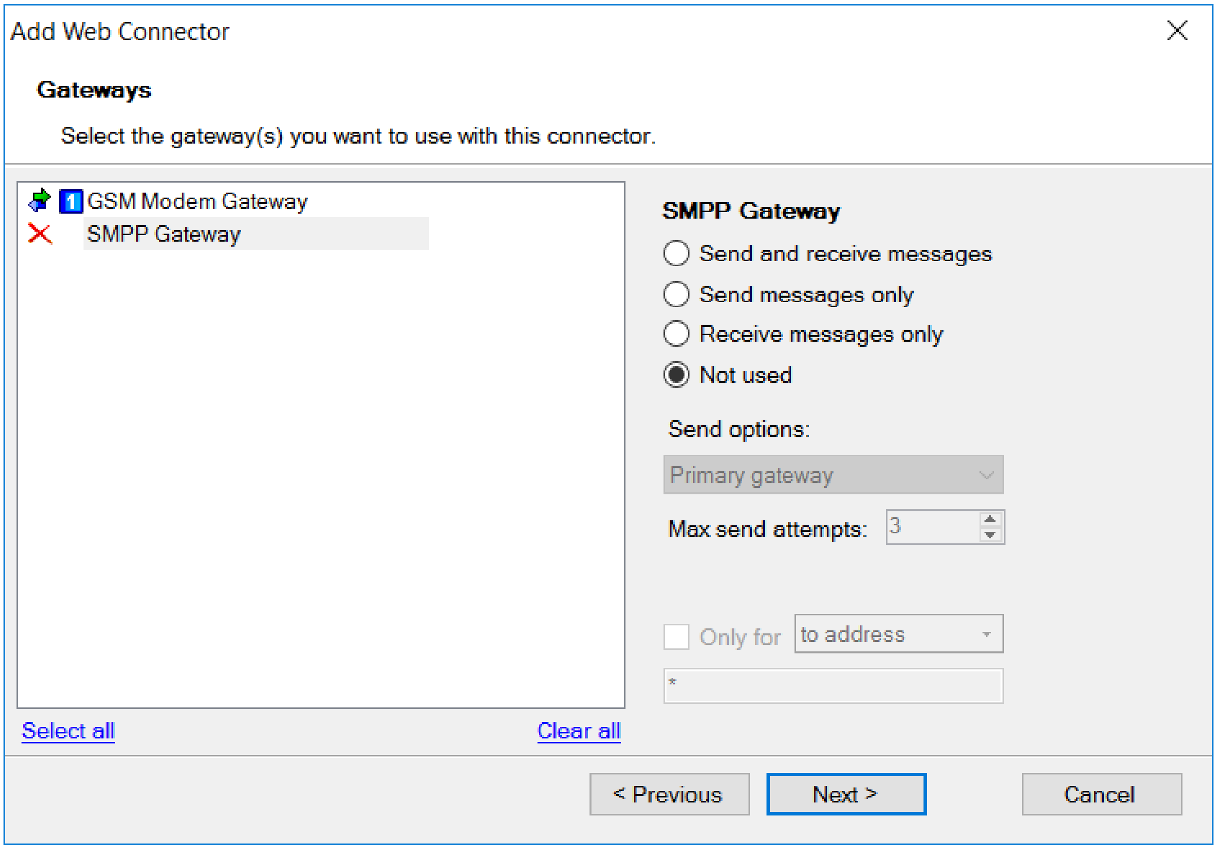
Web connector
Email connector
SQL connector
Scripting connector
Excel connector
SMPP connector (basic and full editions only)
Protocol connector (basic and full editions only)

Assign the gateways

If you have installed more than one gateway you can select the gateways you want to use with the new connector. For each available gateway you can choose if you want to use the gateway to send and receive messages, send or receive only or not at all.

You can assign the gateway as a primary gateway or a backup gateway. Diafaan SMS Server will first try to send the message with one of the assigned primary gateways. If sending the message failed it will be repeated with a maximum of **Max send attempts** for each primary gateway.

If sending the message is still not successful on all primary gateways, the same procedure is used with the assigned backup gateway(s).



Add Web Connector

Gateways

Select the gateway(s) you want to use with this connector.

- GSM Modem Gateway
- SMPP Gateway

SMPP Gateway

○ Send and receive messages
○ Send messages only
○ Receive messages only
● Not used

Send options: Primary gateway

Max send attempts: 3

□ Only for to address

*

Receive messages

If a gateway receives an SMS message, it is forwarded to all connectors that have the receive messages option enabled for that gateway. This means that one received message may be processed by more than one connector.

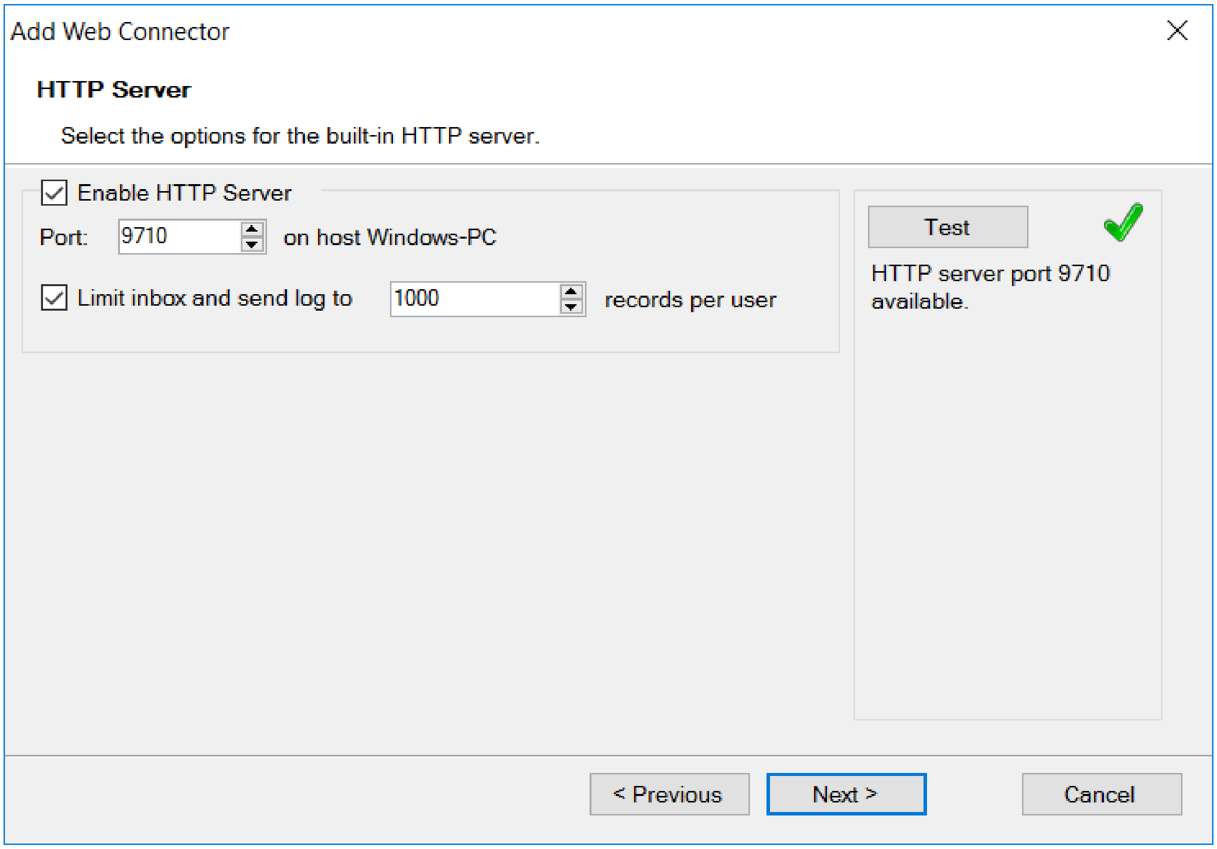
Override the assigned gateways

With most connectors you can use the **Gateway** field to specify a gateway list for each message. The gateway list can be a single gateway name or multiple gateway names separated by semicolons, e.g.: **GSM Modem Gateway** or **GSM Modem Gateway;Clickatell Gateway**. Only gateways that are assigned to the connector can be used, other gateway names are ignored. If a gateway is assigned as backup gateway it will still be used as a backup gateway. Please note that the gateway name must be an exact match, otherwise it will be ignored.

2.3.1 Add a Web connector

Enable the HTTP Web server

The Web connector starts an HTTP web server on the provided port. The HTTP Server enables access to the the SMS Web Application Client and the HTTP API. See the **HTTP API specifications** for an overview of the HTTP commands.



Add Web Connector

HTTP Server

Select the options for the built-in HTTP server.

☑ Enable HTTP Server

Port: 9710 on host Windows-PC

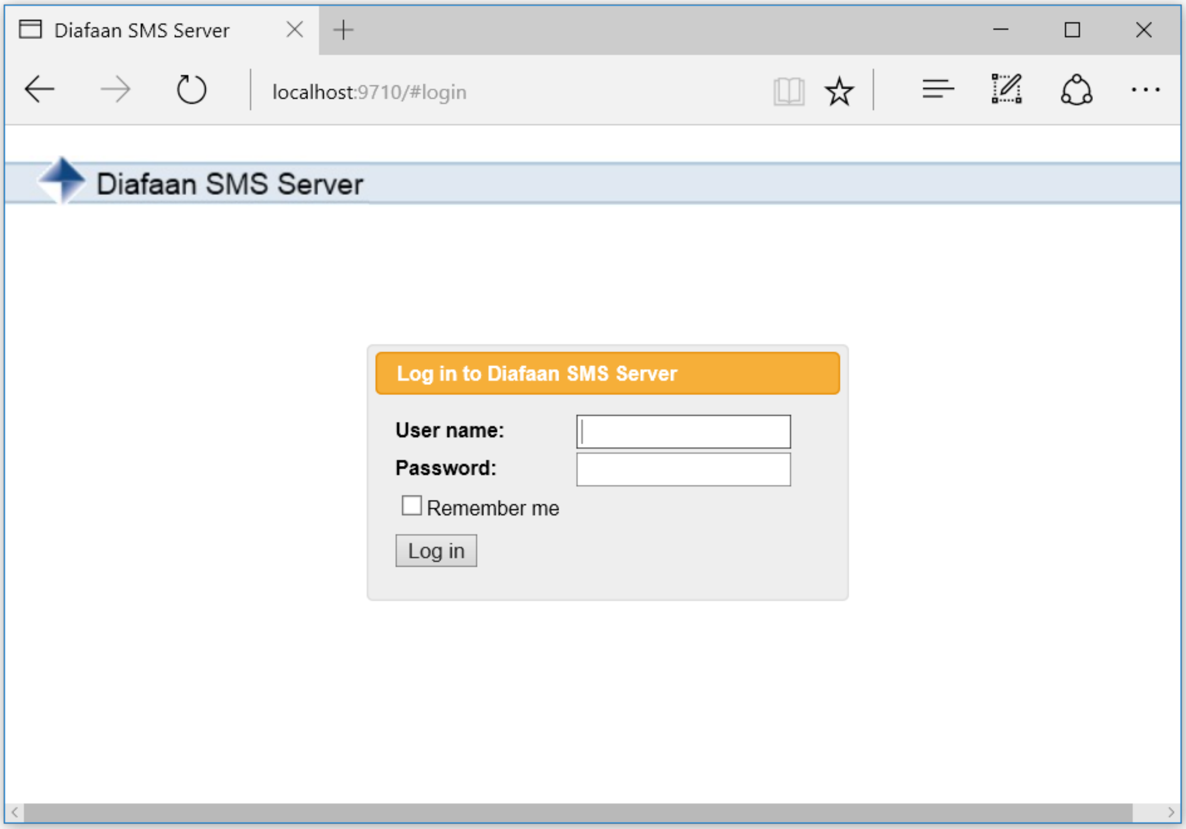
☑ Limit inbox and send log to 1000 records per user

HTTP server port 9710 available.

If the **Limit inbox and send log to** option is selected, the number of log items per user in the Send Log and the Inbox/Receive Log is limited to a fixed number of records. If the log database is getting too large, viewing the logs will get gradually slower and a fixed log limit helps to prevent this.

Access the SMS Web Application Client

On the computer where Diafaan SMS Server is installed, you can access the web application client on url http://localhost:9710 with the user name **admin** and the password you entered in the setup wizard. If the web application can be accessed from the local computer but not from the rest of the network there might be a problem with the firewall. Diafaan SMS Server sets an exception to the Windows firewall on installation but it is possible that other firewalls block access to the HTTP server.



Log in to Diafaan SMS Server

User name:

Password:

□ Remember me

2.3.2 Add an Email connector

Forward email to SMS

Diafaan SMS Server can receive email in one of two ways, with a built-in SMTP server or with a connection to an external POP box. In the Add Email connector wizard you can choose which method you want to use or if you do not want to forward email to SMS.

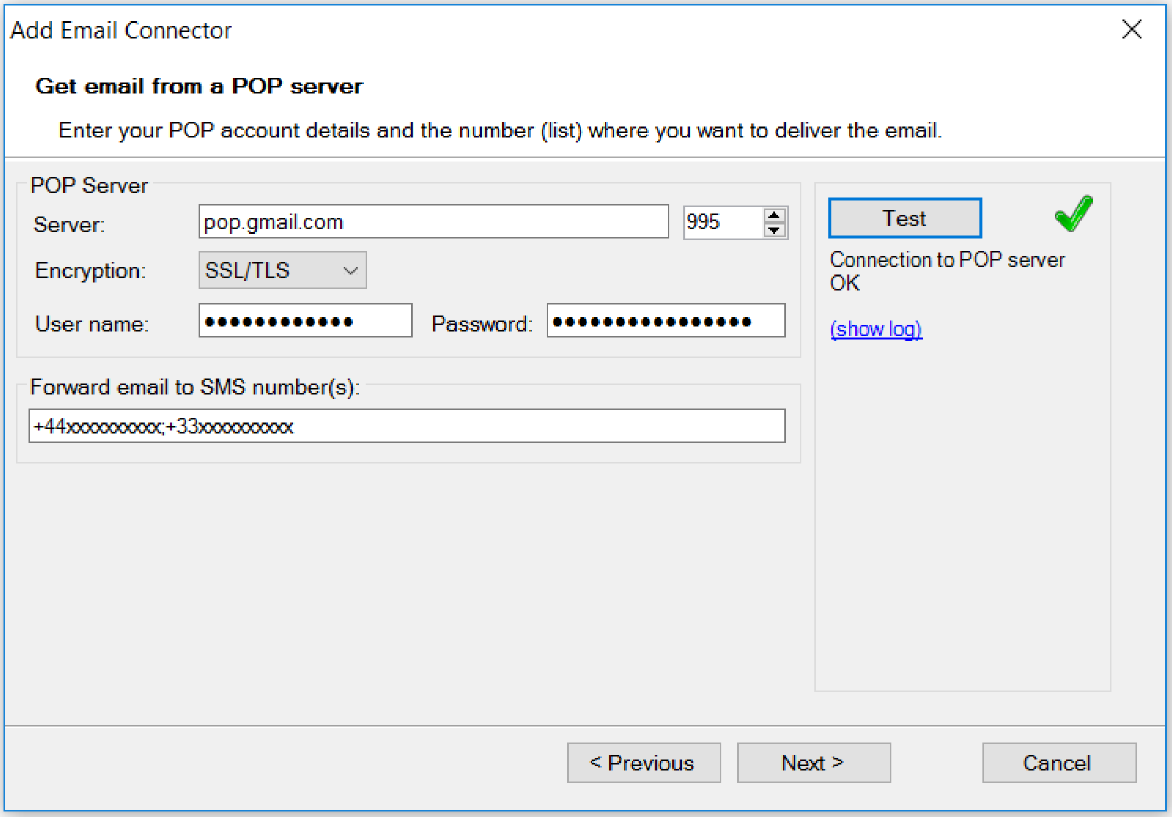
SMTP Server

By default, the new SMTP Server will start on port 25. You can change the SMTP port number in the **Email connector properties**. It is also possible to change the message template, the parts of the email messages that are converted to SMS.
The to address of the mail message is converted into the number and an optional message type, e.g.:

+44xxxxxxxxx@sms.com	is sent to number +44xxxxxxxxx.
p44xxxxxxxxx@sms.com	is also sent to number +44xxxxxxxxx.
+44xxxxxxxxx.sms.text@sms.com	is sent to number +44xxxxxxxxx with message type sms.text.

POP3 Client

The Email connector can retrieve email from a POP box and convert the mail to an SMS message. After installing the connector, the email is forwarded to the GSM number (list) you entered. You can change the message template and add the option to extract the GSM number (list) from the mail subject or body in the **Email connector properties**.



Add Email Connector

Get email from a POP server

Enter your POP account details and the number (list) where you want to deliver the email.

POP Server

Server: pop.gmail.com 995

Encryption: SSL/TLS

User name: ●●●●●●●●●●●● Password: ●●●●●●●●●●●●●●●●

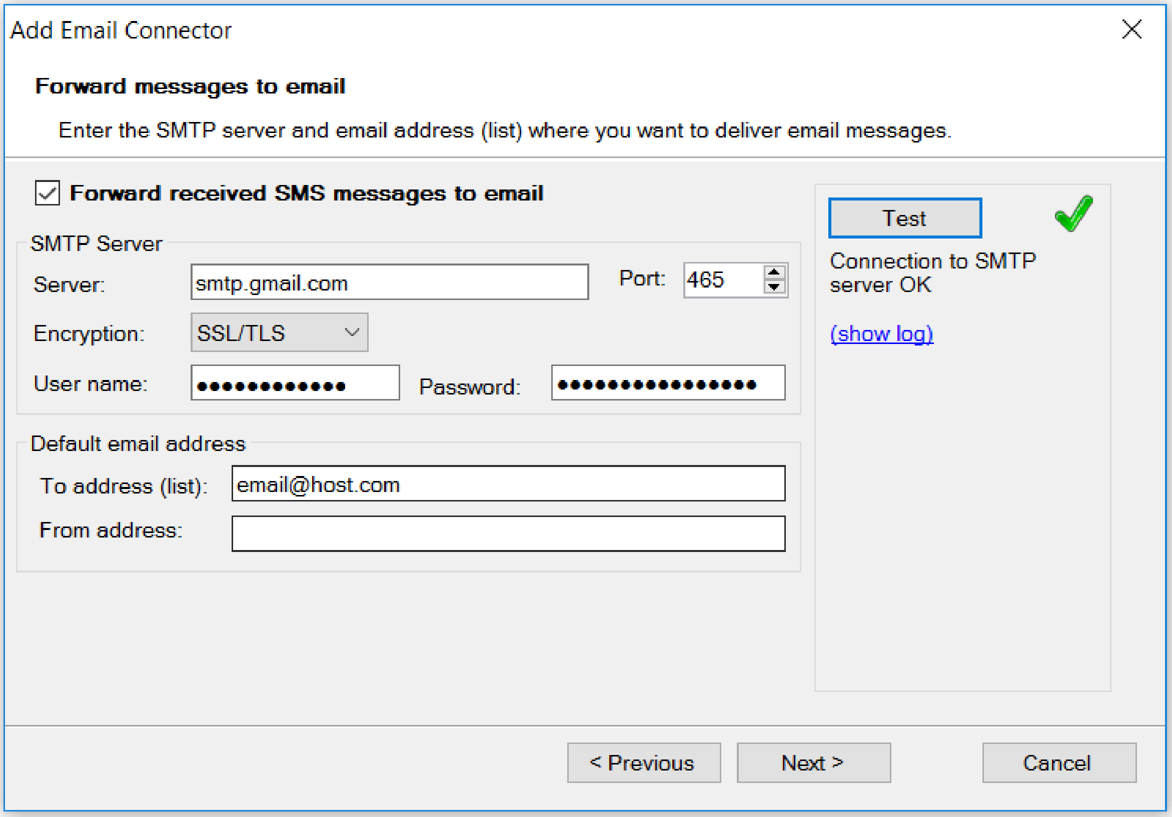
Forward email to SMS number(s):

+44xxxxxxxxxx;+33xxxxxxxxxx

Connection to POP server OK

Forward received SMS message to email

Enable the **Forward received SMS messages** to email to convert received SMS messages to one or more email messages.



Add Email Connector

Forward messages to email

Enter the SMTP server and email address (list) where you want to deliver email messages.

☑ **Forward received SMS messages to email**

SMTP Server

Server: smtp.gmail.com

Port: 465

Encryption: SSL/TLS

User name: ●●●●●●●●●●●●

Password: ●●●●●●●●●●●●●●●●

Default email address

To address (list): email@host.com

From address:

Connection to SMTP server OK

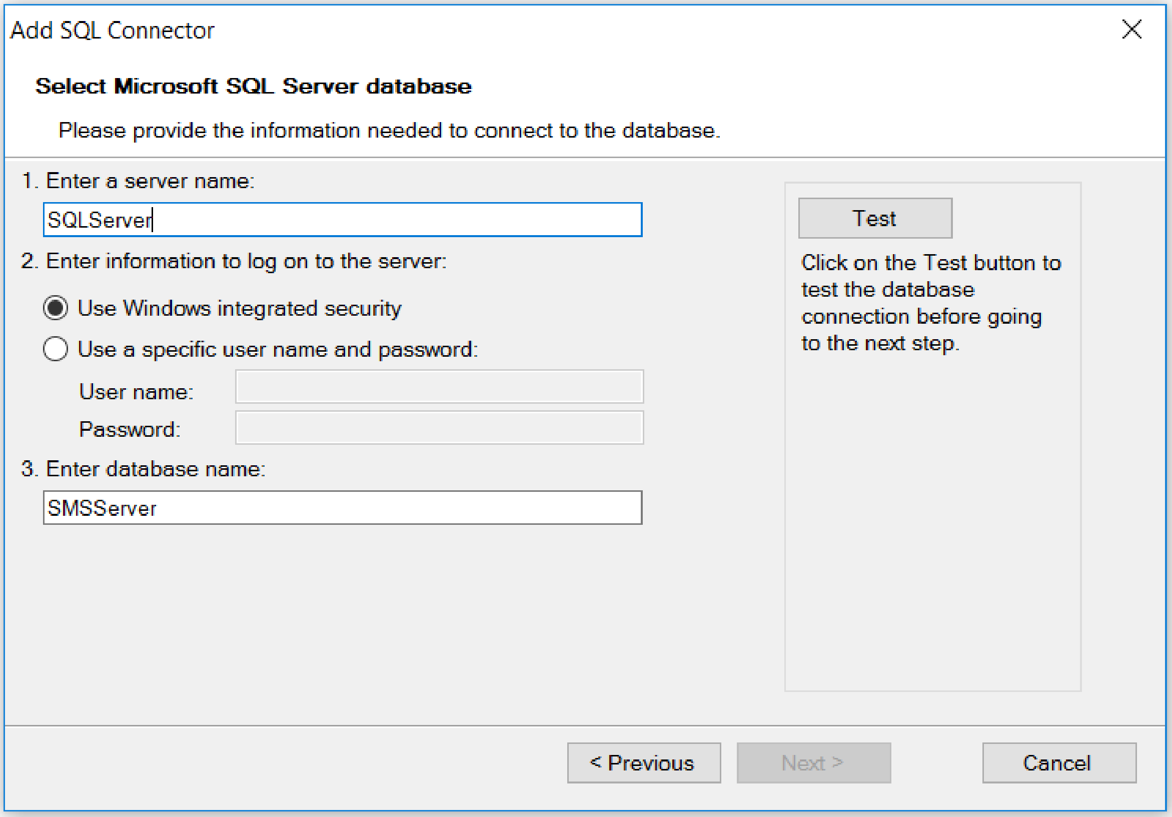
After installing the connector, the message is forwarded to the email address (list) you entered. You can change the email template and add the option to extract the email address from the SMS message or to reply to a previously sent email message in the **Email connector properties**.

2.3.3 Add an SQL connector

The SQL connector provides the connection to the database you want to use. It supports three database tables, one for messages to send (MessageOut), one to store received messages (MessageIn) and a table to log sent or failed calls (MessageLog).

Connect to the database

If you use a Microsoft SQL Server (Express) database you can provide the information necessary to connect to the database in the dialog box.



Add SQL Connector

Select Microsoft SQL Server database

Please provide the information needed to connect to the database.

1. Enter a server name:

SQLServer

2. Enter information to log on to the server:

● Use Windows integrated security

○ Use a specific user name and password:

User name:

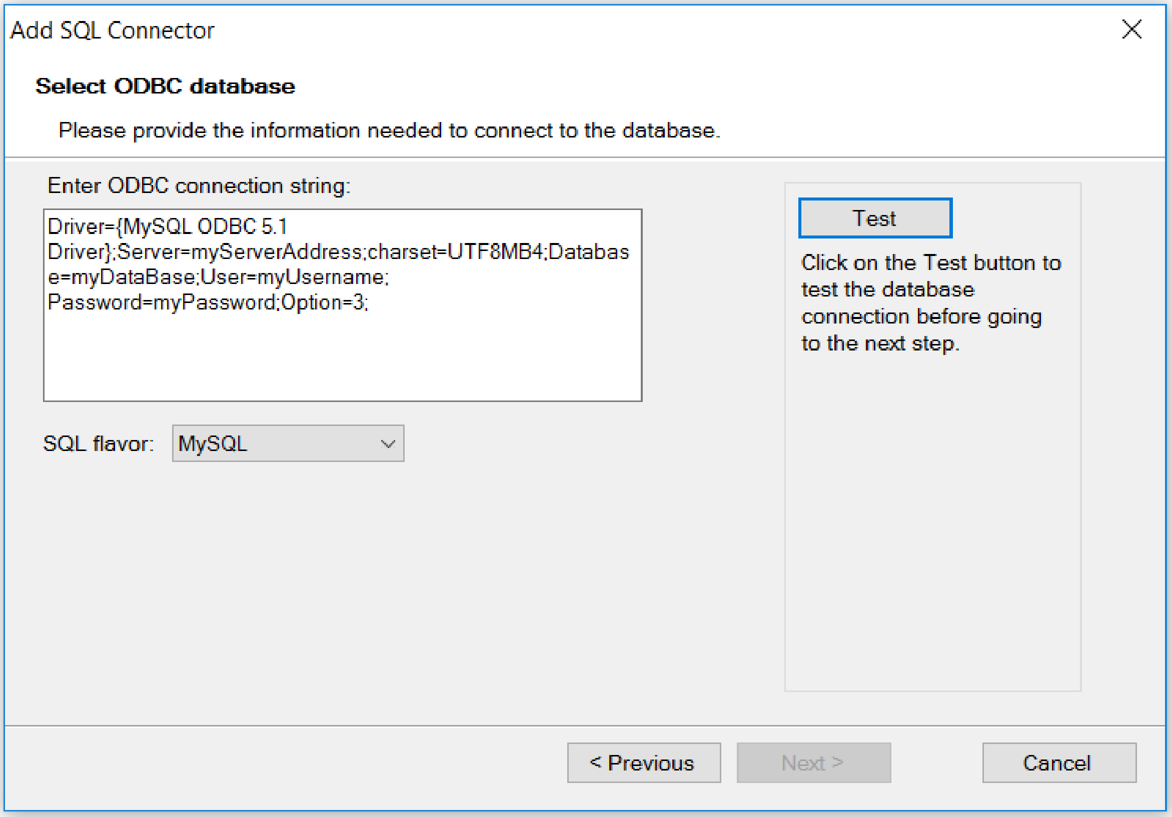
Password:

3. Enter database name:

SMSServer

Click on the Test button to test the database connection before going to the next step.

For other database programs you have to enter the ODBC or OLE DB connection string for your database and the SQL flavor (use ANSI SQL-92 if your database manufacturer is not in the list).



Add SQL Connector

Select ODBC database

Please provide the information needed to connect to the database.

Enter ODBC connection string:

```
Driver={MySQL ODBC 5.1 Driver};Server=myServerAddress;charset=UTF8MB4;Database=myDataBase;User=myUsername;
Password=myPassword;Option=3;
```

SQL flavor: MySQL

Click on the Test button to test the database connection before going to the next step.

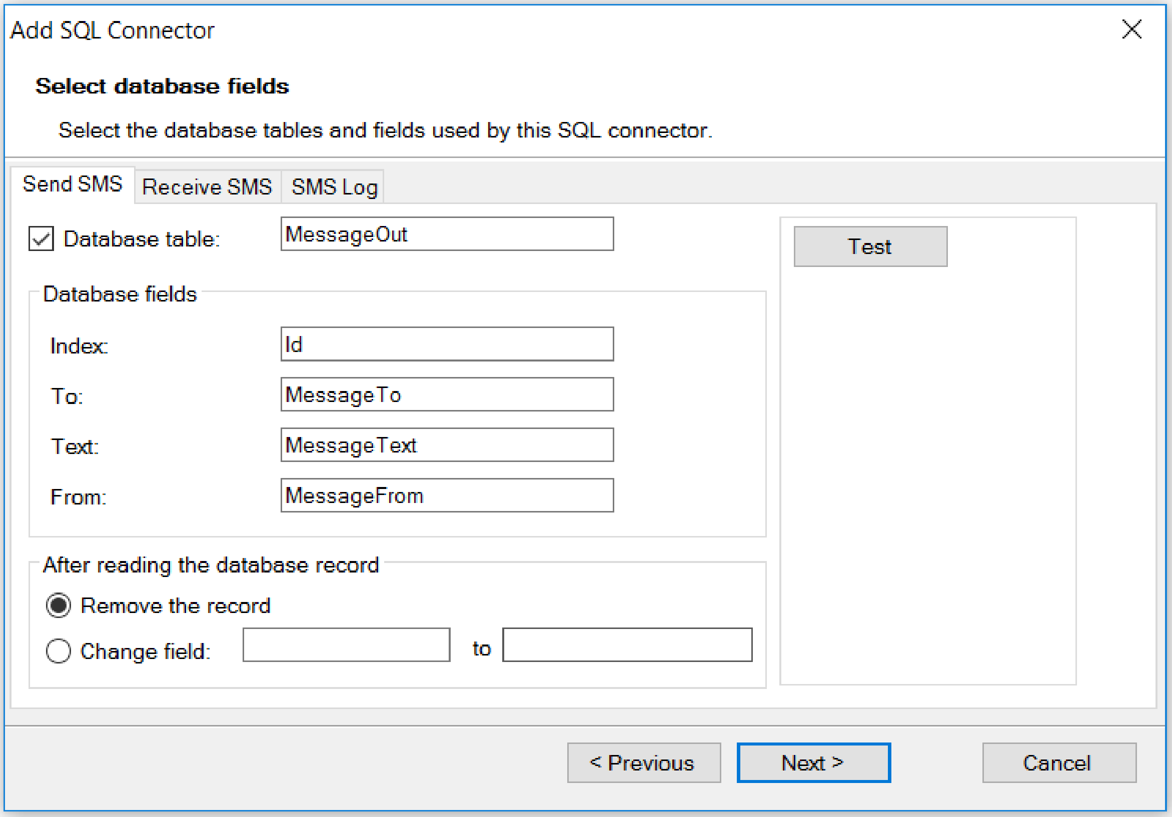
Create the database

The wizard lets you create a Microsoft SQL Server (Express) automatically, if you want to use a different database you have to create the database tables manually. On the following pages you can find the database table definitions for some popular database programs.

Create a MySQL database
Create an Oracle database
Create a Microsoft Access database
Create a PostgreSQL database

Use an existing database

If you used the wizard to create the Microsoft SQL Server (Express) database you are done. However if you use another database program or want the SQL connector to use an existing database with a different table layout you have to provide the table and field names manually.



Select database fields

Select the database tables and fields used by this SQL connector.

☑ Database table: MessageOut

Database fields

Index: Id

To: MessageTo

Text: MessageText

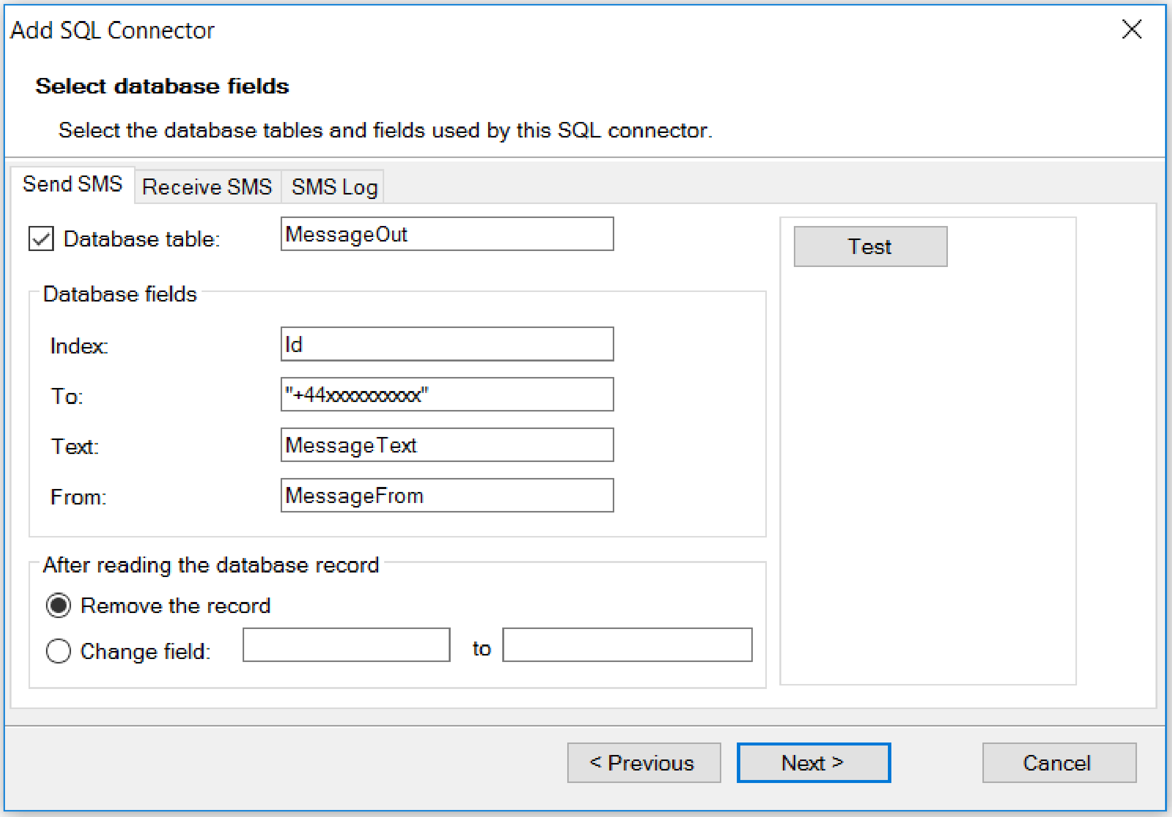
From: MessageFrom

After reading the database record

● Remove the record

○ Change field: to

For each SMS message field you can enter the corresponding database field name. In the Send SMS tab page you can also use double quotes to indicate fixed values. In the example below the **To** part of the SMS message is not retrieved from the database but all messages are sent to number +44xxxxxxxxxx.



Select database fields

Select the database tables and fields used by this SQL connector.

☑ Database table: MessageOut

Database fields

Index: Id

To: "+44xxxxxxxxxx"

Text: MessageText

From: MessageFrom

After reading the database record

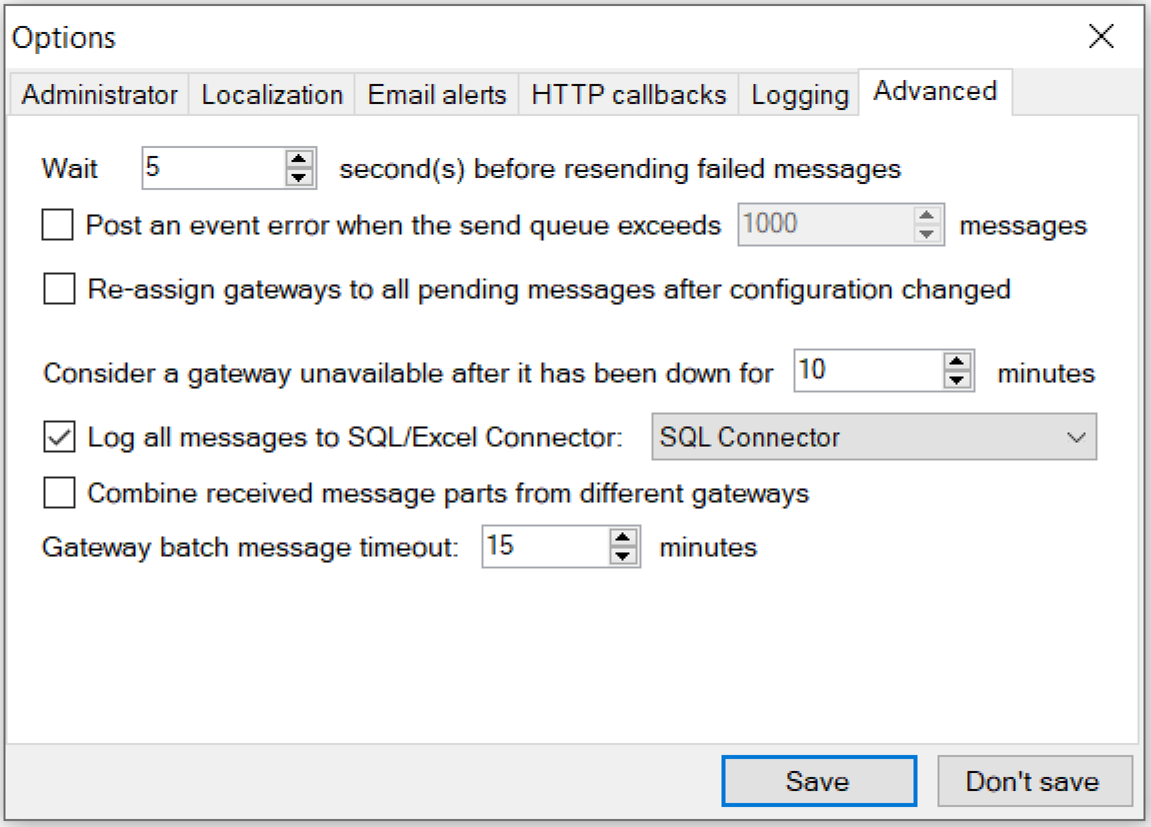
● Remove the record

○ Change field: to

The SQL connector can use more fields, like the SMS message type or the gateway name you want to use to send the messages. These fields can be entered in the advanced properties of the **SQL Connector Properties**.

Log all messages to the SQL Connector

By default, the SQL connector only logs the messages that are sent from the same connector but it is also possible to designate one SQL connector to log all messages in Diafaan SMS Server. This option can be enabled in advanced settings of the general options (menu options 'Actions-Options-Advanced').



Wait 5 second(s) before resending failed messages

□ Post an event error when the send queue exceeds 1000 messages

□ Re-assign gateways to all pending messages after configuration changed

Consider a gateway unavailable after it has been down for 10 minutes

☑ Log all messages to SQL/Excel Connector: SQL Connector

□ Combine received message parts from different gateways

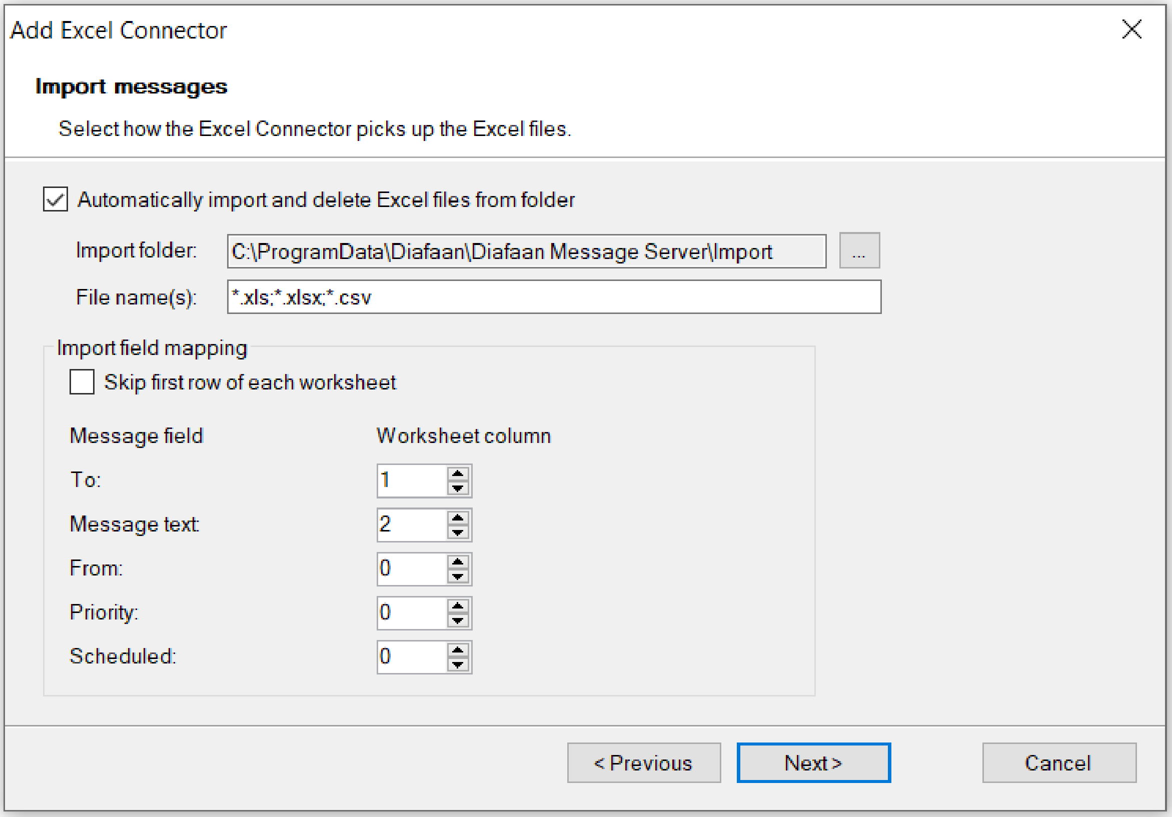
Gateway batch message timeout: 15 minutes

2.3.4 Add an Excel connector

The Excel Connector can import messages from spreadsheet files and export received messages and logs to .CSV (comma separated values) files. Messages can be imported from the Excel files formats .XLS, .XLSX and from generic .CSV (comma separated values) files.

Import messages from spreadsheet files

Select the folder and set the file name(s) of the spreadsheet file(s) that the Excel Connector should import. The Excel Connector reads each file, extracts the messages from the file(s) and then removes the file(s). Each row in the spreadsheet corresponds to one separate message. The settings for the import field mapping maps the columns of the spreadsheet worksheets to the corresponding message field. The mapping of other message fields can be set later in the advanced settings of the Excel Connector.



Add Excel Connector

Import messages

Select how the Excel Connector picks up the Excel files.

☑ Automatically import and delete Excel files from folder

Import folder: C:\ProgramData\Diafaan\Diafaan Message Server\Import

File name(s): *.xls;*.xlsx;*.csv

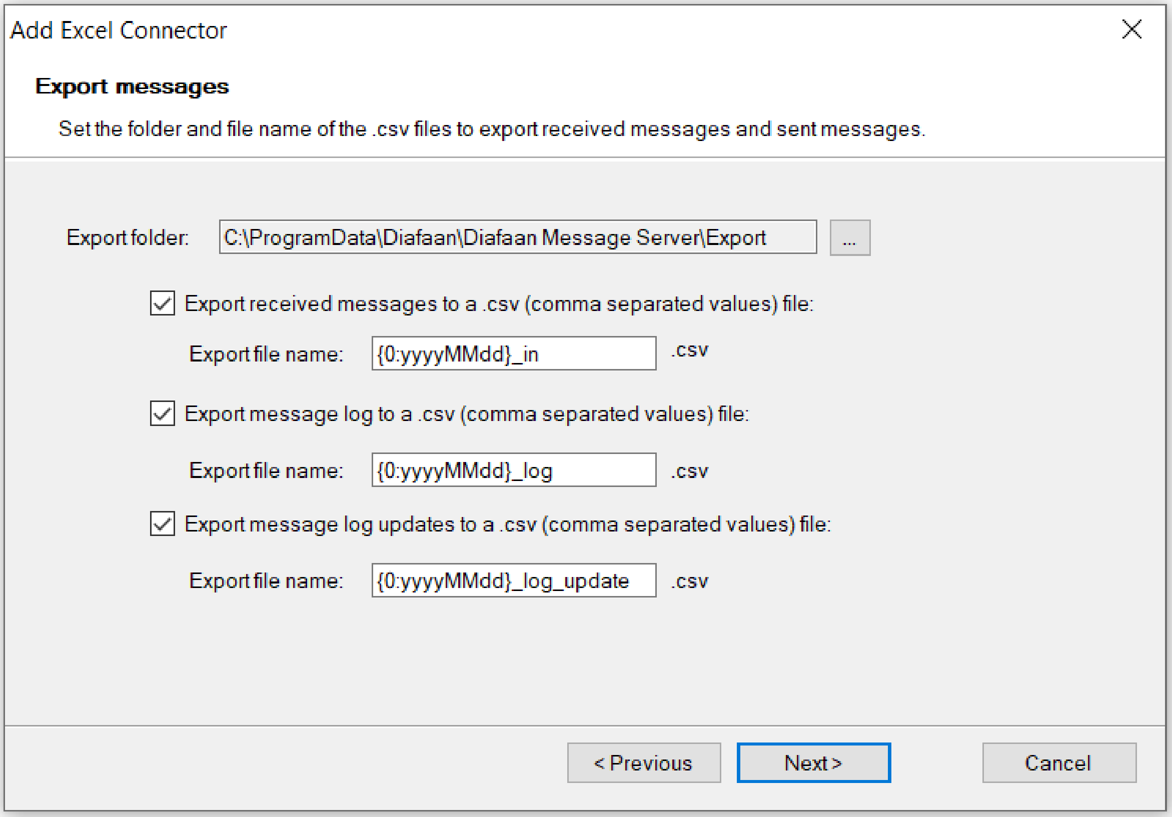
Import field mapping

□ Skip first row of each worksheet

Message field	Worksheet column
To:	1
Message text:	2
From:	0
Priority:	0
Scheduled:	0

Export received messages and logs to .CSV files

Received messages and the log records of the messages that are sent from this Excel Connector can be exported to .CSV files. By default only a limited number of message fields are stored in the .CSV files, message fields can be removed and additional messages fields can be added later in the advanced settings of the Excel Connector.



Add Excel Connector

Export messages

Set the folder and file name of the .csv files to export received messages and sent messages.

Export folder: C:\ProgramData\Diafaan\Diafaan Message Server\Export

☑ Export received messages to a .csv (comma separated values) file:

Export file name: {0:yyyyMMdd}_in .csv

☑ Export message log to a .csv (comma separated values) file:

Export file name: {0:yyyyMMdd}_log .csv

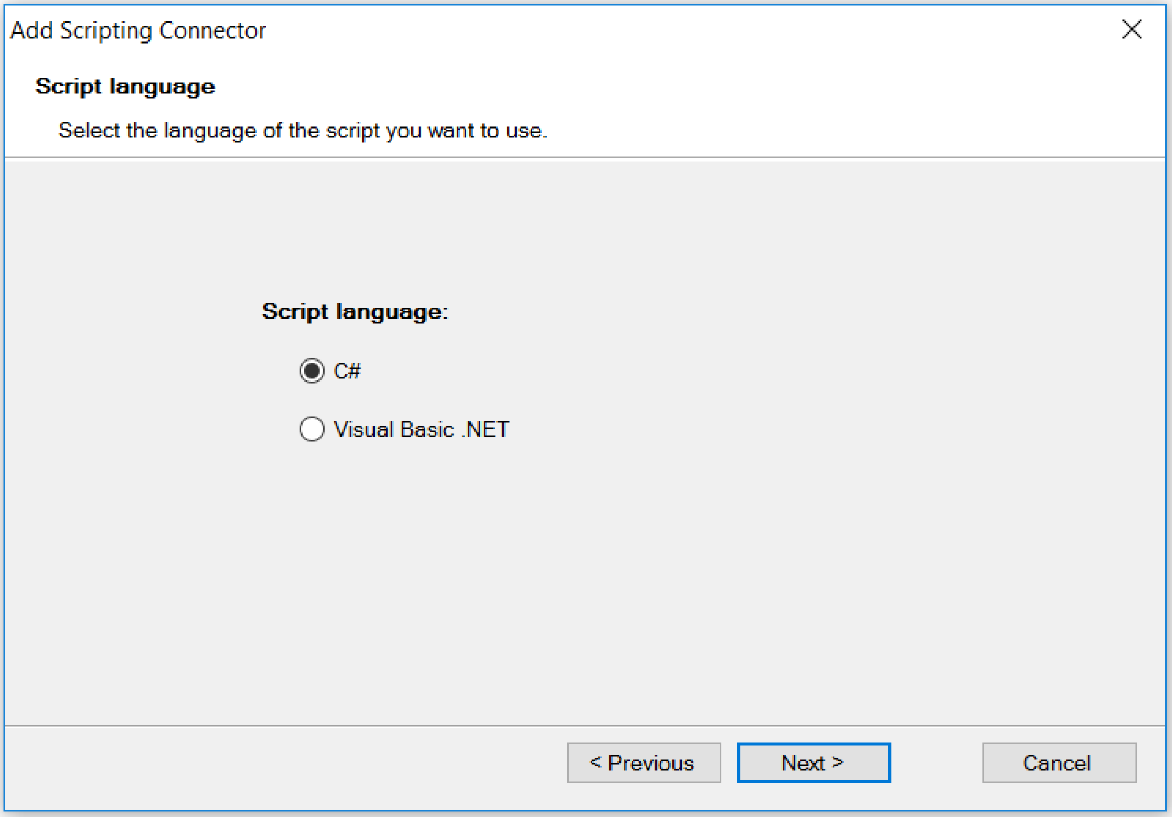
☑ Export message log updates to a .csv (comma separated values) file:

Export file name: {0:yyyyMMdd}_log_update .csv

2.3.5 Add a Scripting connector

Select scripting language

Diafaan SMS Server supports two scripting languages, Visual Basic .NET or C#. After adding the scripting connector a default skeleton script is installed that provides the basic messaging functions. This skeleton script can be adapted in the **Scripting Connector Properties** dialog.



Add Scripting Connector

Script language

Select the language of the script you want to use.

Script language:

● C#

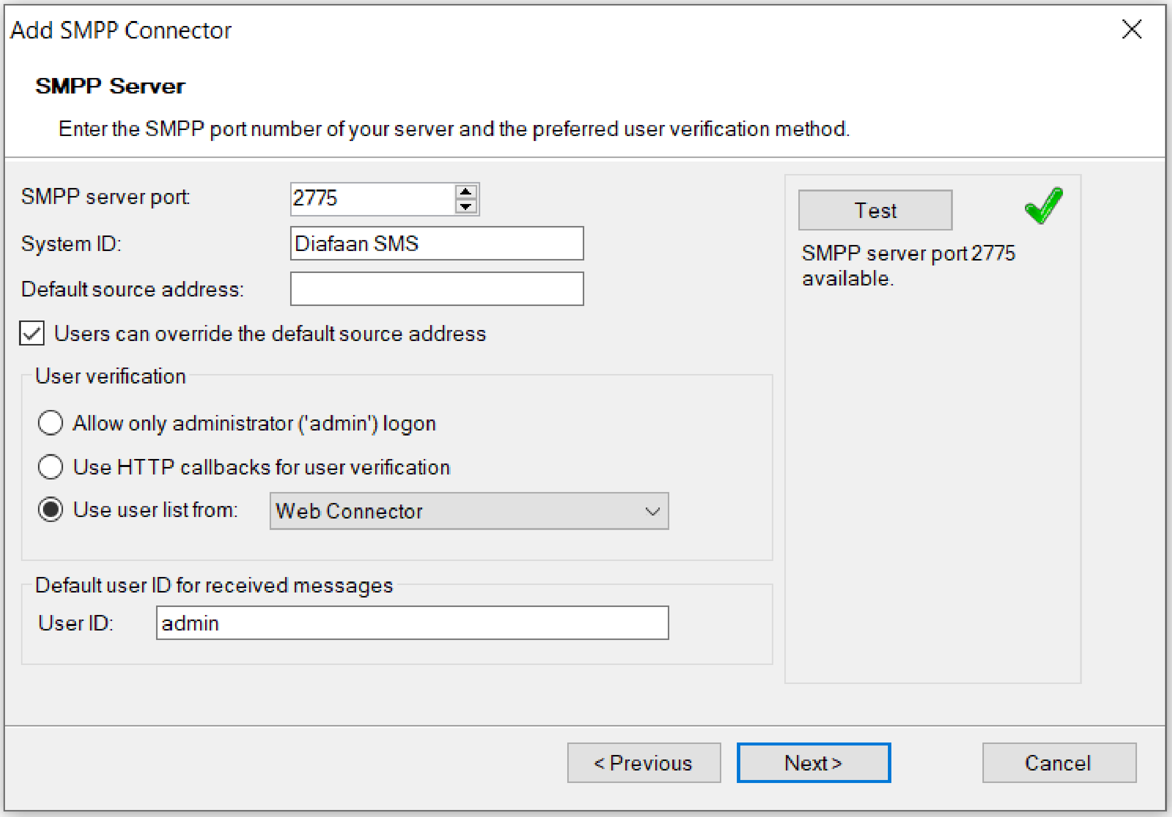
○ Visual Basic .NET

2.3.6 Add an SMPP connector

The SMPP Connector starts an SMPP server that accepts network connections from one or more SMPP clients. The SMPP server supports SMPP versions 3.3, 3.4 and 5.0.

User verification

The SMPP Connector supports three user verification options to check the user name and password of the SMPP clients. A basic 'admin' user verification where only the administrator can log in, verification using HTTP callback requests and verification with the user list of one of the Web Connectors in Diafaan SMS Server. The HTTP callback user verification option is grayed out when HTTP callbacks are not yet set up in the general options. The Web Connector user verification option is grayed out when there is no Web Connector set up in Diafaan SMS Server.



Add SMPP Connector

SMPP Server

Enter the SMPP port number of your server and the preferred user verification method.

SMPP server port: 2775

System ID: Diafaan SMS

Default source address:

☑ Users can override the default source address

User verification

○ Allow only administrator ('admin') logon

○ Use HTTP callbacks for user verification

● Use user list from: Web Connector

Default user ID for received messages

User ID: admin

SMPP server port 2775 available.

Default user ID

Received messages can be assigned a default user ID, this is the user ID of the SMPP client where the received messages are delivered. Diafaan SMS Server can assign a specific user ID for each individual received message using the HTTP callback functionality.

Default source address

Set a fixed source address for all outgoing messages. This source address is used when the source address of the message is empty or if the option 'Users can override the default source address' is disabled.

Advanced options

After the SMPP Connector is added to Diafaan SMS Server, you can change some of the default SMPP options, like the character encoding or the maximum number of messages a client can send per minute in the advanced settings of the **SMPP Connector properties**. Some of these options can be changed for each individual client connection using the HTTP callback user verification.

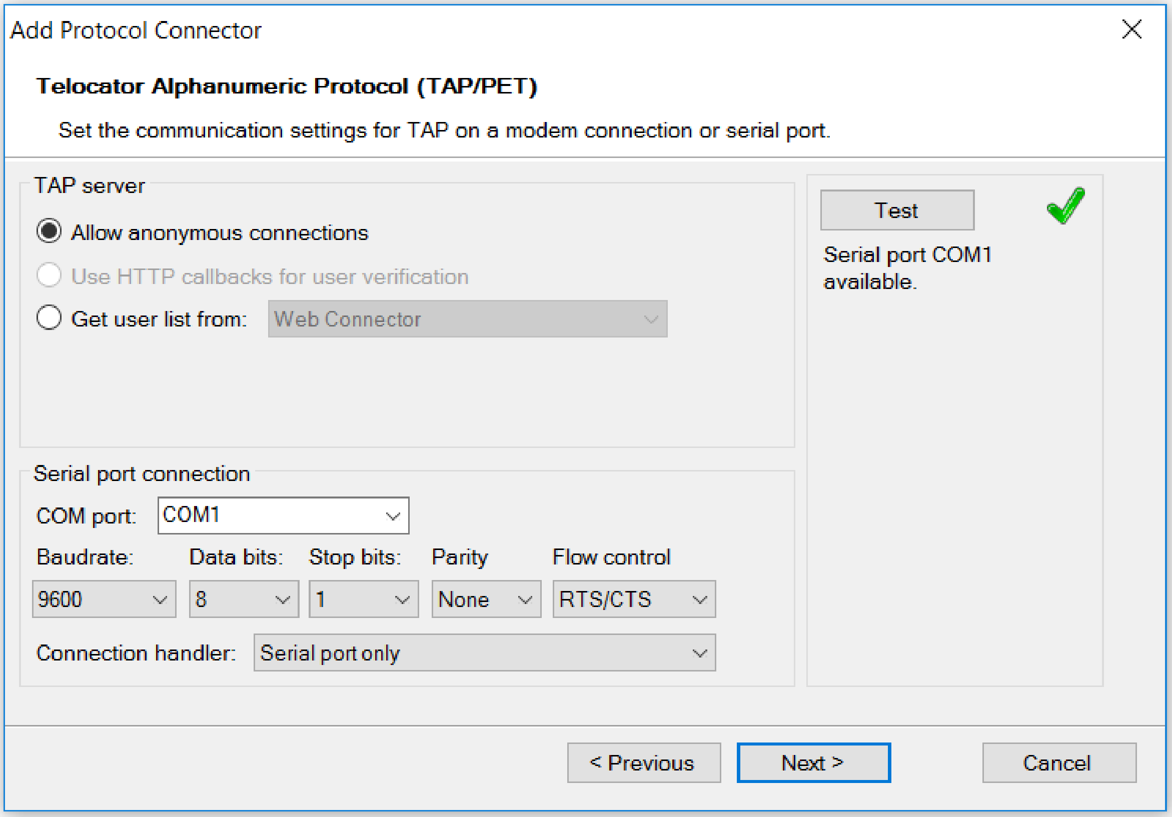
2.3.7 Add a Protocol connector

The Protocol Connector adds support for a number of legacy paging protocols. It allows legacy paging client hardware and software to send messages with Diafaan SMS Server to GSM phones and pagers. The Protocol Connector supports the following protocols:

- TAP/PET/IXO (Telocator Alphanumeric Protocol)
- SNPP (Simple Network Paging Protocol)
- UCP 01/30/51/60 (Universal Computer Protocol)
- ESPA 4.4.4

The paging protocols in the Protocol Connector are quite old and over the years there have been many different interpretations of the standards. The Protocol Connector might therefore not work with all the paging hardware and software that is still in operation. Make sure that you test your paging client thoroughly before using the Protocol Connector in a production environment.

Add a TAP server (Telocator Alphanumeric Protocol)



Add Protocol Connector

Telocator Alphanumeric Protocol (TAP/PET)

Set the communication settings for TAP on a modem connection or serial port.

TAP server

● Allow anonymous connections

○ Use HTTP callbacks for user verification

○ Get user list from: Web Connector

Serial port connection

COM port: COM1

Baudrate:	Data bits:	Stop bits:	Parity	Flow control
9600	8	1	None	RTS/CTS

Connection handler: Serial port only

Serial port COM1 available.

User verification

The TAP server supports three user verification options to check the user name and password of the TAP clients. Anonymous connections (no log on parameters required), verification using HTTP callback requests and verification with the user list of one of the Web Connectors in Diafaan SMS Server. The HTTP callback user verification option is grayed out when HTTP callbacks are not yet set up in the general options. The Web Connector user verification option is grayed out when there is no Web Connector set up in Diafaan SMS Server.

The TAP protocol only supports a single password without user ID. If the contact list from a Web Connector is used to verify the user, the password of the user must be used in the TAP client, not the user name. If the connection handler is set to **serial port only** the password is not enforced.

TAP on a modem connection or serial port

The TAP server listens on the serial port for incoming TAP client connections. The connection handler must be set to **Serial port only** when there is a direct serial port connection between the TAP client and Diafaan SMS Server and the TAP client does not send modem commands to initiate the connection. **Serial port with attached modem** is used when the COM port is attached to a modem and the TAP clients dial in with modems to set up the connection. **Serial port with modem emulation** can be used when the TAP client is connected directly to the serial port but the client expects that a modem is connected, in this mode the Protocol Connector emulates the modem response that the TAP client expects.

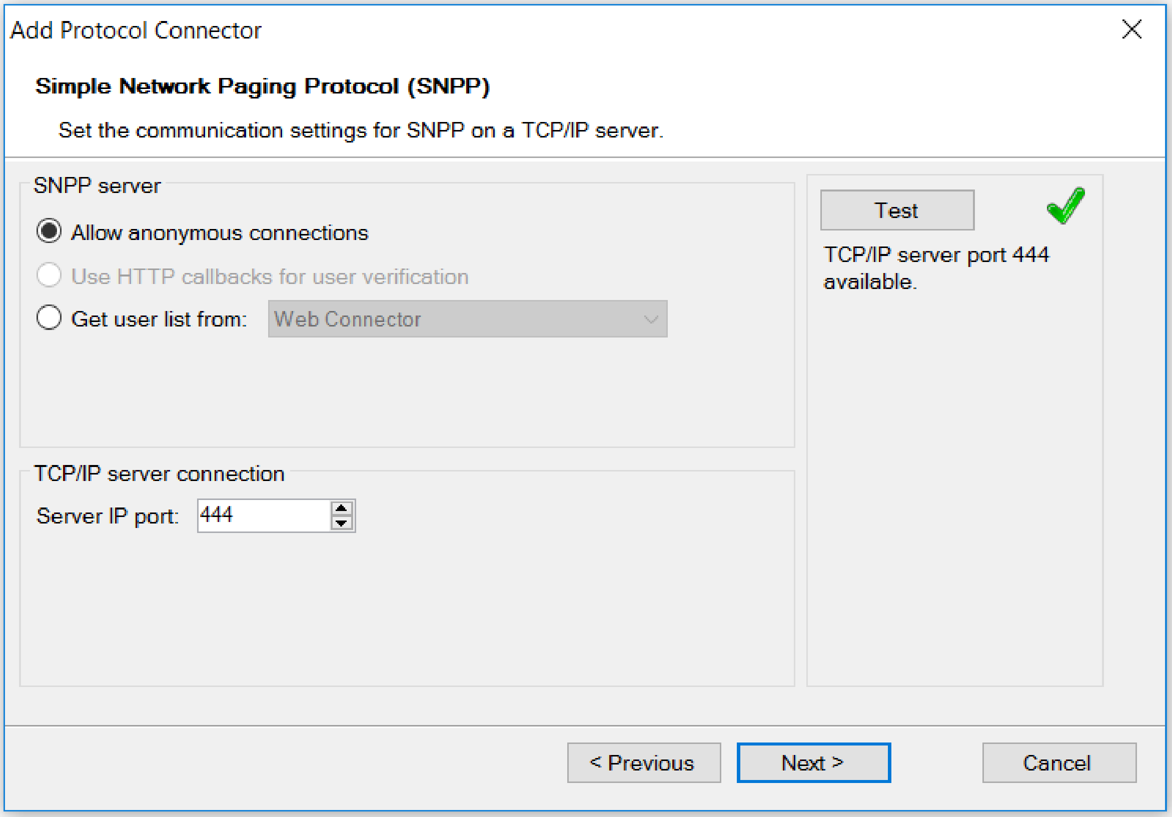
TAP on a TCP/IP server

The TAP server listens on the IP port for incoming TAP client internet connections. The connection handler must be set to **Plain TCP/IP connection** when there is a direct connection between the TAP client and Diafaan SMS Server and the TAP client does not send modem commands to initiate the connection. **TCP/IP with modem emulation** can be used when the TAP client is connected over the internet but the client expects that a modem is connected, in this mode the Protocol Connector emulates the modem response that the TAP client expects.

TAP on a TCP/IP client connection

The TAP server opens an Telnet connection to a TAP client. This mode can be useful when a legacy serial port TAP client is connected to a serial to TCP/IP converter so that it can be accessed over a network connection. The connection handler must be set to **Plain TCP/IP connection** when there is a direct connection between the TAP client and Diafaan SMS Server and the TAP client does not send modem commands to initiate the connection. **TCP/IP with modem emulation** can be used when the TAP client expects that a modem is connected, in this mode the Protocol Connector emulates the modem response that the TAP client expects.

Add an SNPP server (Simple Network Paging Protocol)



Add Protocol Connector

Simple Network Paging Protocol (SNPP)

Set the communication settings for SNPP on a TCP/IP server.

SNPP server

● Allow anonymous connections

○ Use HTTP callbacks for user verification

○ Get user list from: Web Connector

TCP/IP server connection

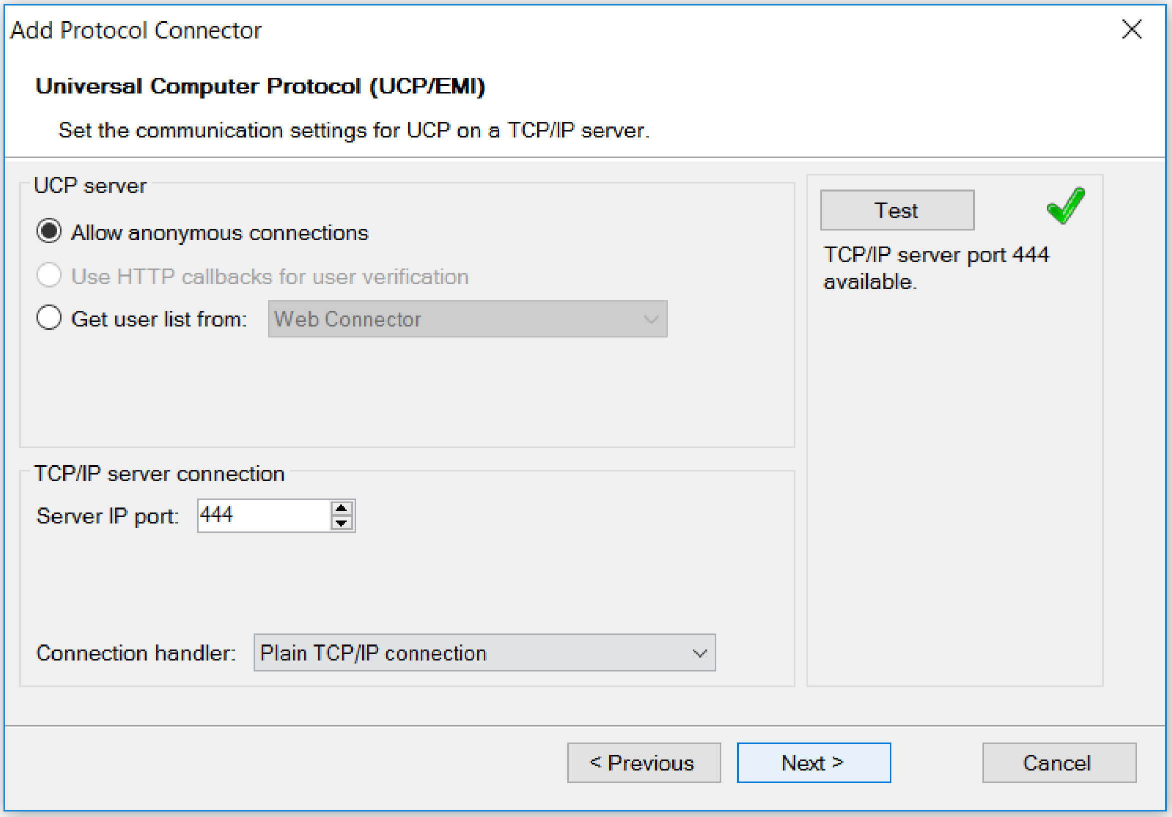
Server IP port: 444

TCP/IP server port 444 available.

User verification

The SNPP server supports three user verification options to check the user name and password of the SNPP clients. Anonymous connections (no log on parameters required), verification using HTTP callback requests and verification with the user list of one of the Web Connectors in Diafaan SMS Server. The HTTP callback user verification option is grayed out when HTTP callbacks are not yet set up in the general options. The Web Connector user verification option is grayed out when there is no Web Connector set up in Diafaan SMS Server.

Add a UCP server (Universal Computer Protocol)



Add Protocol Connector

Universal Computer Protocol (UCP/EMI)

Set the communication settings for UCP on a TCP/IP server.

UCP server

● Allow anonymous connections

○ Use HTTP callbacks for user verification

○ Get user list from: Web Connector

TCP/IP server connection

Server IP port: 444

Connection handler: Plain TCP/IP connection

TCP/IP server port 444 available.

User verification

The UCP server supports three user verification options to check the user name and password of the UCP clients. Anonymous connections (no log on parameters required), verification using HTTP callback requests and verification with the user list of one of the Web Connectors in Diafaan SMS Server. The HTTP callback user verification option is grayed out when HTTP callbacks are not yet set up in the general options. The Web Connector user verification option is grayed out when there is no Web Connector set up in Diafaan SMS Server.

If the connection handler is set to **serial port only** the password is not enforced.

TAP on a modem connection or serial port

The UCP server listens on the serial port for incoming TAP client connections. The connection handler must be set to **Serial port only** when there is a direct serial port connection between the UCP client and Diafaan SMS Server and the UCP client does not send modem commands to initiate the connection. **Serial port with attached modem** is used when the COM port is attached to a modem and the UCP clients dial in with modems to set up the connection. **Serial port with modem emulation** can be used when the UCP client is connected directly to the serial port but the client expects that a modem is connected, in this mode the Protocol Connector emulates the modem response that the UCP client expects.

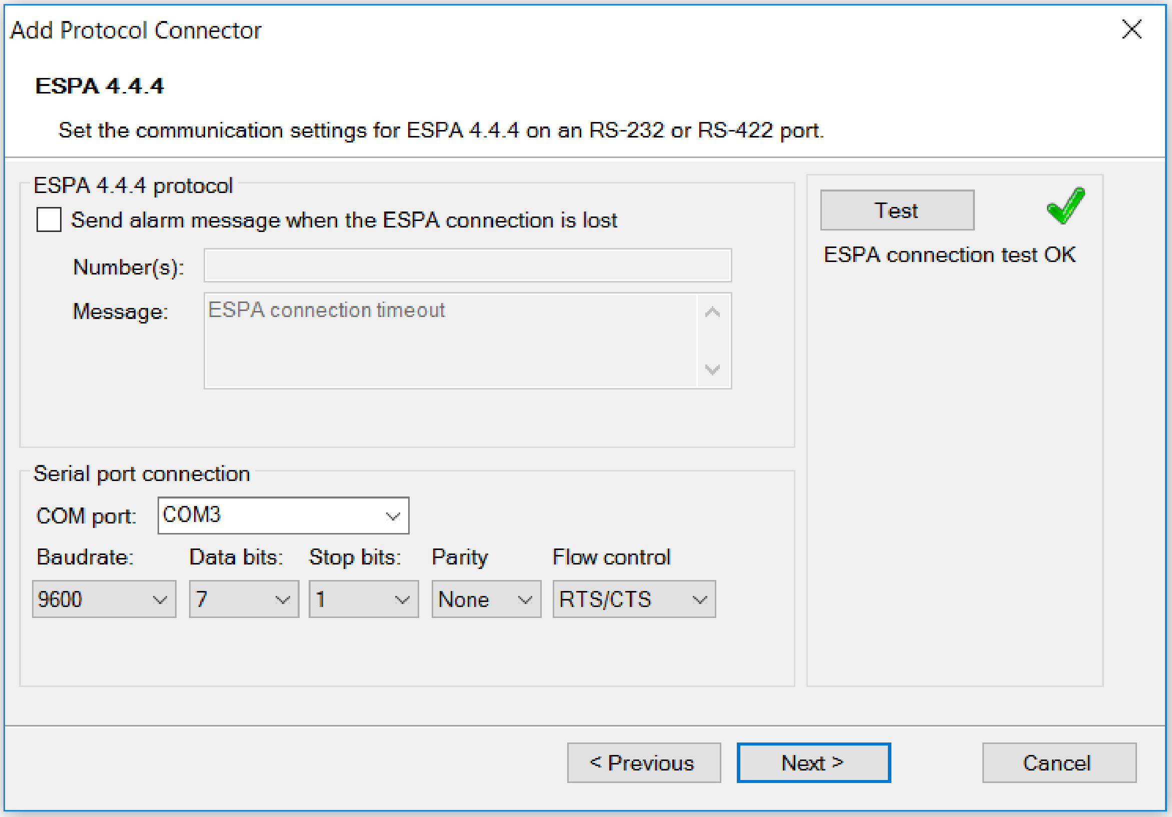
UCP on a TCP/IP server

The UCP server listens on the IP port for incoming UCP client internet connections. The connection handler must be set to **Plain TCP/IP connection** when there is a direct connection between the UCP client and Diafaan SMS Server and the UCP client does not send modem commands to initiate the connection. **TCP/IP with modem emulation** can be used when the UCP client is connected over the internet but the client expects that a modem is connected, in this mode the Protocol Connector emulates the modem response that the UCP client expects.

UCP on a TCP/IP client connection

The UCP server opens an Telnet connection to a UCP client. This mode can be useful when a legacy serial port UCP client is connected to a serial to TCP/IP converter so that it can be accessed over a network connection. The connection handler must be set to **Plain TCP/IP connection** when there is a direct connection between the UCP client and Diafaan SMS Server and the UCP client does not send modem commands to initiate the connection. **TCP/IP with modem emulation** can be used when the UCP client expects that a modem is connected, in this mode the Protocol Connector emulates the modem response that the UCP client expects.

Add an ESPA 4.4.4 station



Add Protocol Connector

ESPA 4.4.4

Set the communication settings for ESPA 4.4.4 on an RS-232 or RS-422 port.

ESPA 4.4.4 protocol

□ Send alarm message when the ESPA connection is lost

Number(s):

Message: ESPA connection timeout

Serial port connection

COM port: COM3

Baudrate:	Data bits:	Stop bits:	Parity	Flow control
9600	7	1	None	RTS/CTS

✓ ESPA connection test OK

Send alarm message when the ESPA connection is lost

The Protocol Connector can send an SMS or pager message when the connection to the ESPA device is lost.

How to...

III

3 How to...

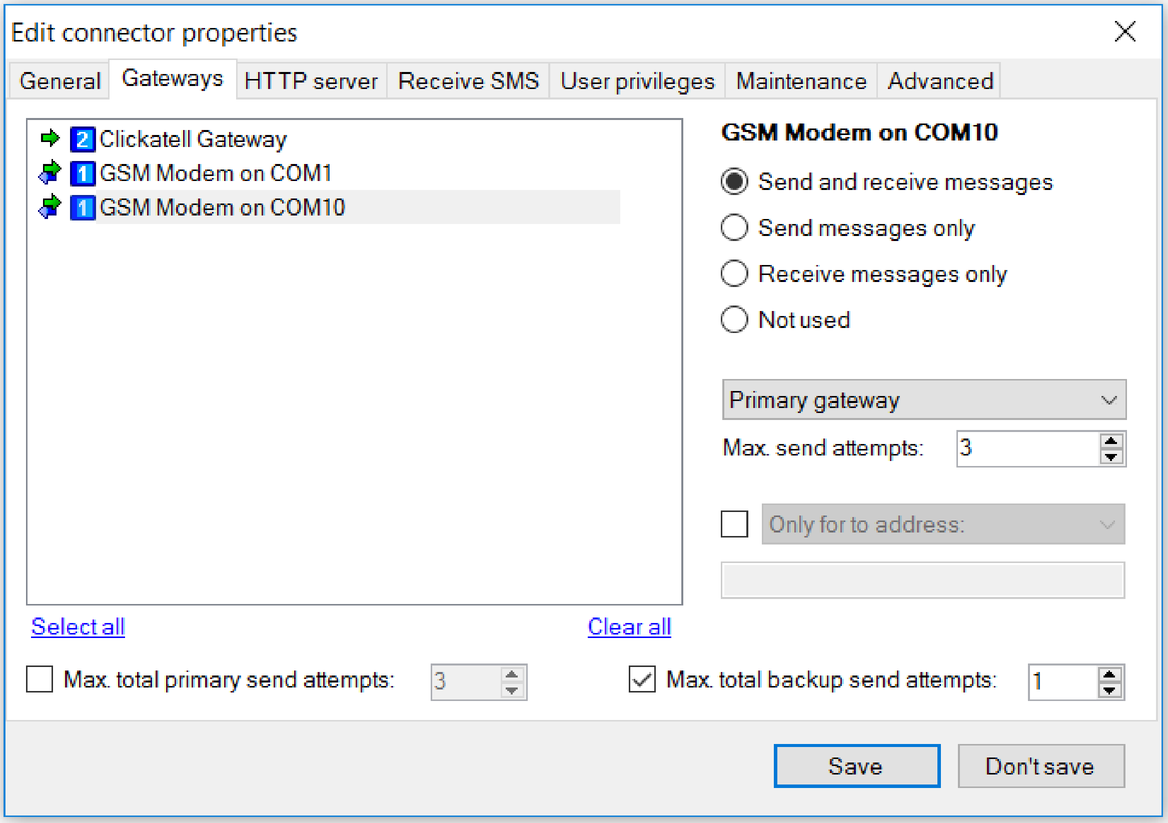
3.1 Route messages

Connector specific routing

The main routing procedure in Diafaan SMS Server is connector specific routing. For each connector you can choose if you want to use the available gateways to send and receive messages, send or receive only or not at all.

Each gateway can be designated as a primary gateway or as a backup gateway. Diafaan SMS Server will first try to send the message with one of the assigned primary gateways. If sending the message failed it will be repeated with a maximum of **Max send Attempts** for each primary gateway. If sending the message is still not successful on all primary gateways, the same procedure is used with the assigned backup gateway(s).

The **Max. total primary send attempts** can be used to limit the total amount of send attempts that are made with the primary gateways. For instance, if this value is set to **1** Diafaan SMS Server will only try to send the messages one time with one of the primary gateways. If the message fails it will switch to the backup gateways or mark the message as failed. The **Max. total backup send attempts** does the same for all the the backup gateways.



Edit connector properties

General | Gateways | HTTP server | Receive SMS | User privileges | Maintenance | Advanced

- 2 Clickatell Gateway
- 1 GSM Modem on COM1
- 1 GSM Modem on COM10

Select all   Clear all

GSM Modem on COM10

● Send and receive messages
○ Send messages only
○ Receive messages only
○ Not used

Primary gateway

Max. send attempts: 3

□ Only for to address:

□ Max. total primary send attempts: 3

☑ Max. total backup send attempts: 1

Save   Don't save

Meaning of the routing icons

1 Use this gateway as a primary gateway
2 Use this gateway as a backup gateway
✕ The gateway is not used for this connector
➡ Use the gateway to send message only
⬅ Use the gateway to receive message only
⇄ Use the gateway to send and receive message

Message specific routing using the To Address, From address or Message Type

For each Gateway in the Connector Properties you can specify whether you want to use the Gateway for all messages or only for specific messages. Messages can be routed depending on the **to address**, **from address** or the **message type** of the message.

The routing filter can consist of one or more wildcards, separated by semicolons or comma's, or a regular expression match when the filter starts with the text "**regex:**". If the routing filter starts with a minus character (-), the message is routed through the gateway when the **to address** or **message type** does not match the filter.

If the routing filter contains a valid file name of an existing file enclosed in brackets (for example: **[c:\temp\whitelist.txt]**) the routing is handled according to the white list / black list rules in the corresponding file.

Routing examples for the "to address" filter

Use this gateway for messages with numbers starting with 07 or +447:
_07*;+447*_

Do not use this gateway for messages with numbers starting with 07 or +447:
_-07*;+447*_

Use this gateway for messages with British mobile numbers, using a regular expression:
regex:^(07|\+447)[0-9]{9}$

Do not use this gateway for messages with British mobile numbers, using a regular expression:
-regex:^(07|\+447)[0-9]{9}$

Use this gateway only for messages with destination numbers that follow the whitelist and blacklist rules in the file c:\temp\whitelist.txt:
[c:\temp\whitelist.txt]

Message specific routing using the Gateway field

With most connectors you can use the **Gateway** field to specify a gateway (list) for each message. The gateway list can be a single gateway name or multiple gateway names separated by semicolons. Only gateways that are enabled for sending in the connector specific routing options can be used, other gateway names are ignored.

By default, if a gateway is assigned as backup gateway in the connector specific routing options, it will still be used only as a backup gateway. But it is possible to change the primary and backup designation of the gateways with the '|' separator. The gateways before the '|' character become the primary gateways and the gateways after the '|' characters become the backup gateways. An optional value can be added after the gateway name to specify the maximum number of send attempts that is used for that gateway using a colon plus the maximum number of send attempts, e.g. **Clickatell Gateway:2**.

In the first example below you can use the GSM Modem on COM1 as a primary gateway and the Clickatell Gateway as the backup gateway by setting the Gateway field to **GSM Modem on COM1;Clickatell Gateway:2**. The gateway **GSM Modem on COM10** is not used for this message.

In the second example the gateway field is set to **Clickatell Gateway|GSM Modem on COM1;GSM Modem on COM10**. The gateway **Clickatell** is used as a primary gateway and the gateways **GSM Modem on COM10** and **GSM Modem on COM10** will be used as backup gateways.

Using message specific routing in the Web Connector:
http://localhost:9710/http/send-message?to=%2B44xxx&message=Message&gateway=GSM+Modem+on+COM1%3BClickatell+Gateway*%3A2*

Change the primary/backup order of the message specific routing:
http://localhost:9710/http/send-message?to=%2B44xxx&message=Message&gateway=Clickatell+Gateway%7CGSM+Modem+on+COM1%3BGSM Modem on COM10

Message specific routing with the HTTP callback request

The message out HTTP callback request can return the **gateway** query field to specify a gateway (list) for each individual message. The gateway list can be a single gateway name or multiple gateway names separated by semicolons with an optional '|' character to designate the primary and backup gateways (see Message specific routing using the Gateway field).

Example HTTP callback response with a 'gateway' field:
result=1&gateway=GSM%20Modem%20Gateway%3A2

Advanced routing

Only available in Diafaan SMS Server - basic and full editions.

The most flexible way to route messages is with a gateway script. Depending on the destination address, message type or message text a different gateway (list) can be used to send the messages. It is also possible to change the destination address, message type and message text before it is sent.

In the gateway skeleton script the message is routed with the PostDispatchMessage method.

```
private void PostDispatchMessage(
  string recordId,
  string toAddress,
  string fromAddress,
  string message,
  string messageType,
  string gateway,
  string backupGateway)
```

The gateway name list is set in the gateway and backupGateway parameters. An optional value can be added after the gateway name to specify the maximum number of send attempts that is used for that gateway, **GSM Modem on COM10:4** for a maximum of 4 send attempts. E.g.:

```
private void OnMessageToSend(string recordId, string toAddress, string fromAddress,
                             string message, string messageType)
{
  PostDispatchMessage(
    recordId,
    toAddress,
    fromAddress,
    "Dispatched: " + message,
    messageType,
    "GSM Modem on COM1;Clickatell Gateway",
    "GSM Modem on COM10:4");
}
```

See for a routing example the **Dispatch SMS to gateway** script.

3.2 Use message types

Message types are used to specify what the message means and how it should be sent. Message types are gateway specific, not all messages types will be understood by all gateways. In those cases the gateway will resort to its default message handling to send the message. In the **Paging Gateway** the message type is used to select which service is used to send the message. The following message types are the standard message types used in Diafaan SMS Server.

Text messages

Text messages can be split and later recombined into multiple SMS messages to increase the message length. Splitting the messages takes a couple of character spaces from each message. The maximum number of characters sent in each SMS message also depends on the character set used for the message.

sms.automatic

If possible the text is sent in the default GSM SMS alphabet. If the message contains characters that are not in the default alphabet the text is sent as unicode. Since an SMS unicode message has a maximum length of 70 characters and a text in the default GSM SMS alphabet has a maximum of 160 characters the number of characters that are sent to the phone will depend on the characters that are used in the message.

sms.text

The default GSM SMS alphabet is always used to send the message. Characters that are not available in the default GSM SMS alphabet are converted to question marks. The message length of a message sent with the default alphabet is usually 160 characters but some characters, like the € symbol take the place of two normal characters and will decrease the maximum message length.

sms.unicode

The message will always be sent in unicode text. This means that the maximum amount of characters in one message is always 70 characters, even if the message contains only characters from the default GSM SMS alphabet.

Flash message

If implemented on the GSM phone, flash messages are displayed directly on the phone's screen instead of being stored in the message inbox. You can send a flash message with one of the message types:
sms.automatic.flash
sms.text.flash
sms.unicode.flash

Binary messages

Binary message can contain up to 140 bytes per SMS message. In Diafaan SMS Server all bytes of the binary messages are encoded in two uppercase hexadecimal digits, for example the 'A' byte is encoded to **41**. In addition to the data itself, a binary message can also contain a User Data Header. This UDH usually defines the purpose of the data.

sms.binary

Binary message can be sent in five ways: data only, with user data header, with data coding scheme, with user data length or with protocol identifier.

Data only:
<Data>
Data message:
3C3C6E0000F01CF00C

User Data Header plus Data:
<UDH>,<Data>
Data message:
04010203,3C3C6E0000F01CF00C

Data Coding Scheme plus User Data Header plus Data:
Usable for data or unicode messages
<DCS>,<UDH>,<Data>
Unicode message:
08,,0055006E00690063006F006400650020006D006500730073006100670065
Data message:
04,04010203,3C3C6E0000F01CF00C

Data Coding Scheme plus User Data Length plus User Data Header plus Data:
Usable for data, text or unicode messages
<DCS>,<UDL>,<UDH>,<Data>
Text message:
00,0C,,D4329E0E6A97E7F3F0B90C
Unicode message:
08,1E,,0055006E00690063006F006400650020006D006500730073006100670065
Data message:
04,0D,04010203,3C3C6E0000F01CF00C

Data Coding Scheme plus Protocol Identifier plus User Data Length plus User Data Header plus Data:
Usable for data, text or unicode messages with a special protocol identifier
<DCS>,<PID>,<UDL>,<UDH>,<Data>
Text message:
00,00,0C,,D4329E0E6A97E7F3F0B90C
Unicode message
08,00,1E,,0055006E00690063006F006400650020006D006500730073006100670065
Data message:
04,00,0D,04010203,3C3C6E0000F01CF00C
Empty text message with PID value 0x40 (Type 0, silent message):
00,40,00,,

See **Send a WAP push message** for an example of a binary messages.

sms.pdu

Only available for GSM Modem Gateways
GSM modems send SMS messages as binary data strings called Packet Data Units. If you use the **sms.pdu** message type you can send a fully customized SMS PDU.

smpp.pdu

Only available for SMPP Gateways
The SMPP protocol uses a binary Packet Data Unit that is different from the SMS PDU format. If you use the **smpp.pdu** message type you can send a fully customized SMPP PDU.

USSD commands

USSD is a standard for transmitting information over GSM signaling channels. It is mostly used as a method to query the available balance and other similar information in pre-paid GSM services. The function that is triggered when sending USSD is network-dependent and depends on the specific services the operator is offering.

gsm.ussd

Only available for GSM Modem Gateways
The USSD code should be sent in the **to** field of the message. The response is sent back as a received SMS message.

Example USSD codes:
 * *101#
 * *109*72348937857623#

3.3 Use number templates

Number templates are used to translate the To address from one format to another. You can use a number template if the gateway accepts only specific number formats (e.g. only numbers in international number format) and the source of the messages might generate the numbers in other formats.

Number templates can also be used if you use different gateways to send the messages and if each gateway accepts numbers in a slightly different format. For instance when you use a GSM modem as a primary gateway to send the messages and a Paging Gateway with an email service as a backup gateway. Since the email service only accepts an email address as a valid recipient address you can use a number template to translate the GSM phone number to a valid email address.

Number templates in gateways or connectors

All the gateways and connectors in Diafaan SMS Server have a number template property. The number template of the connectors change the number format immediately after they are received in the connector, the result is that the transformed number is used throughout Diafaan SMS Server, including in the logs. In the gateways, the number is changed just before sending the message. This means that the original number is used throughout Diafaan SMS Server, including in the logs

Number template format

The number template uses a simple format to check if the number confirms to the template and to translate the number from one format to another.

# The hash tag character is used for each individual digit or character that is available in the source address and should be sent to the gateway.

[...] Everything in brackets is not used to check if the number matches the number format but is inserted in the resulting recipient address.

Other characters All other characters are used to check if the number matches the number format but are removed from the resulting recipient address.

The number template field can contain multiple number templates, each one separated by a semicolon, e.g.:
07[+447]#########;447[+447]#########

The number template also supports regular expression replacement if the field conforms to the following format:
regex:<regular expression>replace:<replacement>
or
regex:<regular expression>replace:<replacement>template:<number template>

If the number does not conform to the template format it is not translated to a different number format but sent 'as is'.

Examples

Translate local UK GSM numbers to international format:
07[+447]######### translates "07848000000" to "+447848000000".

Translate UK GSM number to an international format starting with 00:

07[00447]#########;+447[00447]######### translates "07848000000" and "+447848000000" to "00447848000000".

Translate a pager phone number to a 5 digit PIN:
+124812#####;024812##### translates "+12481234567" and "02481234567" to "34567"

Translate a national GSM number to an email address:
0[0]##########[@t-mobile.de] translates "01511111111" to "01511111111@t-mobile.de"

Translate national German GSM numbers to international format:
regex:^01(?=[0-9]{9,13}$)replace:+491

Strip all non digits from the number:
regex:[^\d]replace:

Strip all non digits (0-9) from the number and then translate local UK GSM numbers to international format:
regex:[^\d]replace:template:07[+447]#########

3.4 Use white lists and black lists

In most of the advanced gateway and connector settings in Diafaan SMS Server white list and black lists can be used to allow or block messages from or to specific numbers or number ranges.

Note: The uniform white list/black list was introduced in Diafaan SMS Server 4.2.0.0. White lists in gateways and connectors that were created in older versions can still behave in the original way. White lists in gateways and connectors that are created in Diafaan SMS Server version 4.2.0.0 or higher and white lists that start with a header bracket (**[**) all conform to the new format.

Sections

Each section of the white list consists of a header and a list of numbers or keywords, one number/keyword on each line. The header specifies how the numbers/keywords are interpreted, as a white list or black list, case sensitive or case insensitive etc.

Headers

The header appears on a line by itself, in square brackets ('[' and ']'). Each header contains one or more options, separated by comma's, that define how the number list below it is handled. The header can contain the following keywords:

allow	Allow the numbers/keywords in the list (default)
deny	Block the numbers/keywords in the list
break	Stop processing the rest of the list when a match is found (default)
nobreak	Continue processing the rest of the list when a match is found
ignorecase	Ignore upper and lower case differences in the match (default)
enforcecase	Only allow an exact match

If the header contains a valid file name and if it is on the first line of the list then the file is used as the white list/black list and the rest of the list is ignored. The file must have the white list/black list format as is described in this text.

Number list

Under each header is a list of numbers or keywords, one number or keyword on each line. The numbers may contain the wildcard characters '*' (zero or more characters), '?' (a single character) and '#' (a single digit) to define a number range instead of a single number.

If the line starts with 'regex:' the rest of the line is handled as a regular expression.

Example 1

Allow only German and British numbers except number +49123456789:

```
[allow,nobreak]
+49*
+44*

[deny]
+49123456789
```

Example 2

Use the file 'C:\temp\whitelist.txt' as the white list and/or black list:

```
[C:\temp\whitelist.txt]
```

Example 3

Allow only valid email addresses (using a regular expression) but block gmail.com email addresses:

```
[allow,nobreak]
regex:\b[A-Z0-9._%+-]+@[A-Z0-9.-]+\.[A-Z]{2,6}\b

[deny]
*@gmail.com
```

Example 4

Allow only valid Dutch mobile numbers:

```
[allow]
+316########
```

Example 5

Allow only numbers +49123456789 and +49987654321 (without a header, see the note at the top of this page)

```
+49123456789
+49987654321
```

3.5 Use HTTP callbacks

Only available in Diafaan SMS Server - basic and full editions.

Diafaan SMS Server can use HTTP callbacks to send information about the message flow in Diafaan SMS Server to a web server. The web server can use this information for message and user accounting. For messages to send (action message_out) or for received messages (action message_in) it is also possible to block the messages completely or to change the message fields with the web server script.

The response of all the HTTP callback should be a URL (percentage) encoded query string. The minimum response for all callbacks is 'result=1', this means that the message will be handled normally by Diafaan SMS Server. It is important that the response from the HTTP server is as fast as possible since the handling of other messages in the Gateway or Connector is blocked until the HTTP callback is finished. When the HTTP callback takes more than 5 seconds, the message is rejected.

There is an example PHP HTTP callback script available on the download page of the *Diafaan SMS Server* website that can be used a a base for a customized script. The script includes a minimal implementation of the HTTP callback functionality and a basic user accounting system, intended for use with the SMPP Connector.

Message out HTTP callback

The message out callbacks are sent from each individual connector, just before the message is added to the message queue of Diafaan SMS Server. The message can be discarded with the response 'result=0' and most of the message fields can be changed in the HTTP response query string.

HTTP query parameters	
source	The name of the connector that is sending the HTTP callback.
action	"message_out".
user_id	User ID of the user sending the message.
user_name	User name of the user sending the message, for some connectors it is the same as user_id.
ip_number	The IP number of the sender, if applicable.
from	The source number or name to be sent with the message.
to	The destination address of the message.
message	The message text of the message.
message_type	The message type for the message.
message_guid	Unique message GUID for the message.
message_info	User generated message info, this value is not used by Diafaan SMS Server and can be set freely.
user_info	The optional user info field.
priority	The priority of the message, default 0.
scheduled	Optional date and time when the message should be sent, in ISO 8601 format.
gateway	The name of the gateway(s) that are allowed to send the message.
tlv_list	Optional TLV list when the message is sent with an SMPP server.
session_count	The index of the message in the current session. This field is only used in connection based connectors like the SMPP Connector.
example: http://host.com/?source=SQL+Connector&action=message_out&user_id=&ip_number=&from=%2B44xxxxxxxx&to=%2B44xxxxxxxx&message=Message+text	

The HTTP response may contain the following parameters, the optional parameters should only be used if you want to change one or more of the message fields		
result	M	"0" (reject message) or "1" (accept message).
result_error_text	O	Optional error text to facilitate debugging if result=0. This parameter is ignored by Diafaan SMS Server.
user_id	O	User ID of the user sending the message.
from	O	The source number or name to be sent with the message.
to	O	The destination address of the message.
message	O	The message text of the message.
message_type	O	The message type for the message.
message_guid	O	Unique message GUID for the message.
message_info	O	User generated message info, this value is not used by Diafaan SMS Server and can be set freely.
user_info	O	The optional user info field.
error_code	O	A connector dependent error code that is sent back to the sender when the message is rejected. For messages from the SMPP Connector, the error code must be one of the ESME_R.... error numbers.
error_text	O	The error text that is sent back to the sender when the message is rejected. Not all connector protocols support an error text.
priority	O	The priority of the message, default 0.
scheduled	O	The date and time when the message should be sent, in ISO 8601 format. Set to an empty string to reset/remove the schedule. E.g.: '2020-12-31T19:35:00', or including time zone: '2020-12-31T19:35:00.0000000-07:00'
gateway	O	The name of the gateway(s) that are allowed to send the message.
tlv_list	O	Optional TLV list when the message is sent with an SMPP Gateway.
close_connection	O	Set to "1" to request that the connector closes the connection to the client after the message is handled (sent or rejected).
example: result=1&user_id=64535&from=%2B4400000000&to=%2B4411111111&message=Changed+message+text		

Message in HTTP callback

The message in callbacks are sent from each individual gateway, just before the message is added to the message queue of Diafaan SMS Server. The message can be discarded with the response 'result=0' and most of the messages fields can be changed in the HTTP response query string.

HTTP query parameters	
source	The name of the gateway that is sending the HTTP callback.
action	"message_in".
from	The source number or name to be sent with the message.
to	The destination address of the message.
sms_central	The SMSC number of the SMSC that delivered the message.
message	The message text of the message.
message_type	The message type for the message.
message_id	The message ID of the message, used to be able to combine multi-part messages.
message_part	The message part number when an incomplete multi-part message is received.
message_parts_received	The received number of message parts when an incomplete multi-part message is received.

message_parts_total	The total amount of messages parts of this message.
send_time	The send date and time of the message.
receive_time	The date and time the message is received by Diafaan SMS Server.
pdu	The PDU of the message.
tlv_list	The TLV list of the SMPP message (only for messages from an SMPP Gateway).
example: http://host.com/?source=GSM+Modem+Gateway&action=message_in&from=%2B44xxxxxxxx&to=%2B44xxxxxxxx&message=Message+text	

The HTTP response may contain the following parameters, the optional parameters should only be used if you want to change one or more of the message fields		
result	M	"0" (reject message) or "1" (accept message).
result_error_text	O	Optional error text to facilitate debugging if result=0. This parameter is ignored by Diafaan SMS Server.
user_id	O	User ID of the user sending the message.
from	O	The source number or name to be sent with the message.
to	O	The destination address of the message.
message	O	The message text of the message.
message_type	O	The message type for the message.
connector_list	O	The list of connectors that may receive the message.
tlv_list	O	Optional TLV list when the message forwarded to an SMPP Connector
example: result=1&user_id=64535&from=%2B4400000000&to=%2B4411111111&message=Changed+message+text		

Message log HTTP callback

The message log callbacks are sent from each individual connector when the initial send result of the message is received.

HTTP query parameters	
source	The name of the connector that is sending the HTTP callback.
action	"message_log".
user_id	User ID of the user sending the message.
from	The source number or name to be sent with the message.
to	The destination address of the message.
message	The message text of the message.
message_type	The message type for the message.
message_guid	Unique message GUID for the message.
message_info	User generated message info.
message_id	The message ID that was returned by the gateway.
message_parts	The number of sent message parts for the message.
gateway	The name of the gateway that has sent the message.
status_code	"200" (message sent) or "300" (message failed).
status_text	The status text of the message.
error_code	The error code that was returned by the gateway.
error_text	The error text that was returned by the gateway.
send_time	The date and time the message was sent, in ISO 8601 UTC format.
user_info	The optional user info field.
pdu	The PDU or comma separated PDU list of the sent message.
result_trace	An error trace of the message, this field can give an indication of the reliability of the gateways.

The HTTP response is ignored by Diafaan SMS Server, it is best to return 'result=1' for future compatibility.

Message log update HTTP callback

The message log update callbacks are sent from each individual connector when an update is received for a previous message log callback. This callback is usually sent after a delivery confirmation is received for one of the messages.

HTTP query parameters	
source	The name of the connector that is sending the HTTP callback.
action	"message_log_update".
user_id	User ID of the user sending the message.
message_id	The message ID that was returned by the gateway.
message_guid	Unique message GUID for the message.
message_info	User generated message info.
status_code	"201" (message received on phone) or "301" (message could not be delivered to the phone).
status_text	The status text of the message.
error_code	The error code that was returned by the gateway.
error_text	The error text that was returned by the gateway.
receive_time	The date and time the message was received on the phone (only when status_code is "201"), in ISO 8601 UTC format.
user_info	The optional user info field.
receipt	The receipt text of the delivery report (only for receipts from an SMPP Gateway).
tlv_list	The TLV list of the SMPP receipt (only for receipts from an SMPP Gateway).

The HTTP response is ignored by Diafaan SMS Server, it is best to return 'result=1' for future compatibility.

Event log HTTP callback

The event log callbacks are sent for all error, warning or information events in Diafaan SMS Server. Repeating errors are usually throttled to a maximum of one per hour, similar to the event log in Diafaan SMS Server.

HTTP query parameters	
source	The name of the module in Diafaan SMS Server that triggered the event.
action	"event_log"
event_time	The date and time when the event was triggered, in ISO 8601 UTC format.
event_type	"information", "warning", "error" or "unknown".
event_message	The event description.
event_trace	The event description, including optional additional information about the source of the problem.

The HTTP response is ignored by Diafaan SMS Server, it is best to return 'result=1' for future compatibility.

Validate user HTTP callback

Validate the user name or password, depending on the type of connector this callback is sent once for every connection or every time a message has to be send. Connectors that do not support user/password logon, like the SQL Connector, don't send this callback at all.

HTTP query parameters	
source	The name of the connector that is sending the HTTP callback.
action	"validate_user".
ip_number	The IP number of the sender (if applicable).
user_id	The login user ID (user name).
password	The login password.
example: http://host.com/?source=Web+Connector&action=validate_user&user_id=&ip_number=127%2E0%2E0%2E1&user_name=User&password=Password	

The HTTP response may contain the following parameters, the optional parameters should only be used if you want to change the user ID		
result	M	"0" (reject username and password) or "1" (accept user name and password).
result_error_text	O	Optional error text to facilitate debugging if result=0. This parameter is ignored by Diafaan SMS Server.
user_id	O	User ID override.
example: result=1&user_id=64535		

User validation requests from an SMPP Connector can also return these connection specific parameters		
smpp_maxmessagesperminute	O	A numeric value of 0 (unlimited) or higher.
smpp_messageidformat	O	"hexadecimal", "decimal" or "guid".
smpp_characterencoding	O	"gsmdefaultalphabet", "iso_8859_1", "iso_8859_15", "gsmdefaultalphabet7bitspacked" or "gsmdefaultalphabet7bitspackedifudh".
smpp_packettimeout	O	A numeric value between 10 and 120 (seconds).
smpp_connectiontimeout	O	A numeric value between 10 and 3600 (seconds).
smpp_maxuseridconnections	O	A numeric value of 0 (unlimited) or higher.
smpp_use16bitconcatenationid	O	"0" (no) or "1" (yes).
smpp_messageouthandling	O	"default" or "smpp_packets".
smpp_messageinhandling	O	"deliver_sm", "deliver_sm_usepayloadtlv" or "data_sm".
smpp_clientasyncwindowsize	O	A numeric value of 1 (no asynchronous window) or higher.
smpp_includetlvparameters	O	"0" (deliver_sm packets are sent without TLV parameters) or "1" (default, deliver_sm packets can be sent with TLV parameters).
example: result=1&smpp_maxmessagesperminute=600&smpp_use16bitconcatenationid=1		

Accept connection HTTP callback

This callback is sent when a client wants to connect to an IP server based Connector, like the SMPP Connector. The connection can be refused if the response is 'result=0'.

HTTP query parameters

source	The name of the connector that is sending the HTTP callback.
action	"accept_connection".
ip_number	The IP number of the sender (if applicable).
example: http://host.com/?source=SMPP+Connector&action=accept_connection&user_id=&ip_number=127%2E0%2E0%2E1	

The HTTP response may contain the following parameters, the optional parameters should only be used if you want to change the user ID.		
result	M	"0" (reject connection) or "1" (accept connection).
result_error_text	O	Optional error text to facilitate debugging if result=0. This parameter is ignored by Diafaan SMS Server.
user_id	O	User ID override.
example: result=1&user_id=64535		

3.6 Send a WAP push message

A WAP push service indication message is a message to indicate that (new) WAP content is available, it can also be used to send a general URL to a phone user. WAP push messages can be binary encoded and sent as a binary SMS message.

Compose a WAP push service indication message

Compose the User Data Header (UDH)

The User Data Header specifies how the mobile phone should handle the message. For a WAP push message, it contains the "Application Port Addressing 16 bit address" information of the GSM 03.40 specification:

06	UDH length
05	Application Port Addressing, 16 bit address
04	UDH Application Port Addressing content length
0B84	Destination port number, IANA WAP-push Multimedia Messaging Service port (2948)
0000	Source port number

If the combination of the UDH and the content of the message has a length of more than 140 bytes, the message must be split in multiple message parts of no more than 140 bytes each and the UDH is expanded with the concatenation information:

0B	UDH length
05	Application Port Addressing, 16 bit address
04	Content length
0B84	Destination port number, IANA WAP-push Multimedia Messaging Service port (2948)
0000	Source port number
00	Concatenated short message
03	Content length
00	Message reference number (8-bits)
01	Total message parts (1 for a single message part)
01	Sequence number of the message part

And the same with a 16-bit concatenation reference number:

0C	UDH length
05	Application Port Addressing, 16 bit address
04	Content length
0B84	Destination port number, IANA WAP-push multimedia messaging service port (2948)
0000	Source port number
08	Concatenated short message, 16-bit reference number
04	Content length
0000	Message reference number (16-bits)
01	Total message parts (1 for a single message part)
01	Sequence number of the message part

Compose the Wireless Session Protocol header

The Wireless Session Protocol (WSP) header specifies the type of message according to the "Wireless Application Protocol Wireless Session Protocol" specification of the Wireless Application Protocol Forum. In this example the message is a binary encoded WAP push application message with content type "application/vnd.wap.sic" and character set UTF-8:

00	WAP push session reference number (8-bits)

06	WSP type (WAP push, 0x06)
04	Content length
03	Length of content type and accept charset
AE	Content type "application/vnd.wap.sic" (0x2E), bit 7 is always set to "1"
81	Accept charset indication (0x01), bit 7 is always set to "1"
EA	Accept charset type "UTF-8" (0x6A), bit 7 is always set to "1"

Compose the WAP Service Indication message

XML formated WAP push service indication message according to the "Wireless Application Protocol Service Indication" specification of the Wireless Application Protocol Forum:

```
<?xml version="1.0"?>
<!DOCTYPE si PUBLIC "-//WAPFORUM//DTD SI 1.0//EN" "http://www.wapforum.org/DTD/si.dtd">
<si>
  <indication href="http://www.diafaan.com" action="signal-medium">
    WAP Push message text
  </indication>
</si>
```

To be able to send the message in a binary SMS, the message is encoded with the WBXML binary encoding according to the "Wireless Application Protocol
Service Indication" specification:

01	WBXML Version 1.1
05	SI 1.0 Public Identifier
6A	Charset UTF-8
00	String table length 0
45	<si>
C6	<indication
0D	href="http://www.
03	Start of string value
6469616661616E	"diafaan"
00	End of string value
85	.com
07	Attribute: action="signal-medium"
01	>
03	Start of string value
5741502050757368 206D657373616765 2074657874	"WAP Push message text"
00	End of string value
01	</indication>
01	</si>

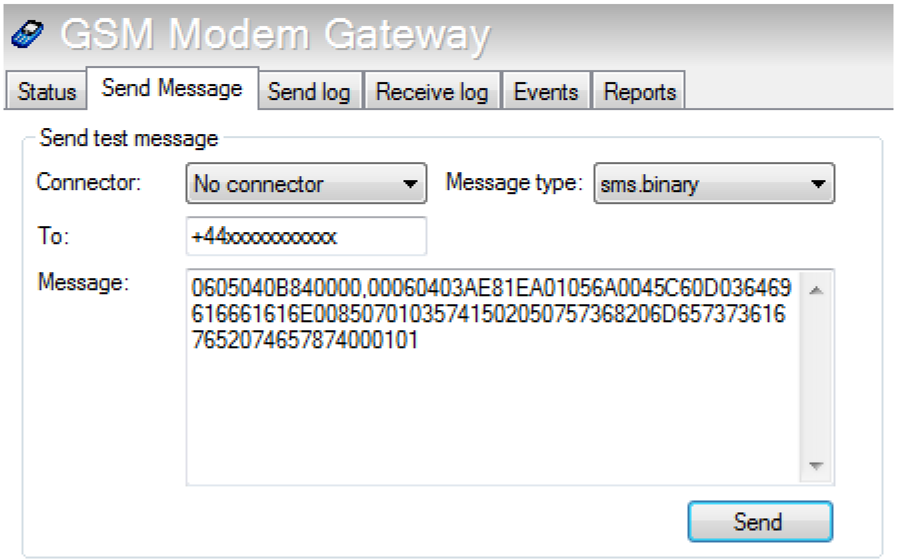
Service indication message:
01056A0045C60D036469616661616E00850701035741502050757368206D657373616765207465787 4
000101

WSP plus service indication message:
00060403AE81EA01056A0045C60D036469616661616E00850701035741502050757368206D6573736
167652074657874000101

UDH plus WSP plus service indication message:
0605040B840000,00060403AE81EA01056A0045C60D036469616661616E00850701035741502050757
368206D657373616765207465787400 0101

Send the WAP push message

You can send the binary encoded WAP push service indication message in Diafaan SMS Server using the **sms.binary** message type. To send the service indication message, use the encoded message that includes the UDH, WSP plus service indication message as specified above. The UDH and the rest of the binary message must be separated by a comma. If you send a binary message, make sure that the message data does not contain any new line or space characters. If it does, the message will fail.



Connector: No connector

Message type: sms.binary

To: +44xxxxxxxxxxx

Message:

```
0605040B840000,00060403AE81EA01056A0045C60D036469
616661616E00850701035741502050757368206D657373616
7652074657874000101
```

3.7 Connect to Microsoft Exchange Server

Your entire organization can send and SMS messages from Outlook and receive SMS messages in their Outlook Inbox if you connect Diafaan SMS Server with your Microsoft Exchange Server 2007, 2010 or 2013. Diafaan SMS Server uses standard internet protocols to connect with Exchange Server.

Step 1: Install Diafaan SMS Server

Diafaan SMS Server can be installed on the server where Microsoft Exchange Server 2007 or 2010 is installed or on a separate server or PC in your network. When you choose to install Diafaan SMS Server on the machine where Microsoft Exchange Server is installed, you have to perform an extra step to ensure that the SMTP servers in Diafaan SMS Server and Exchange Server do not interfere with each other.

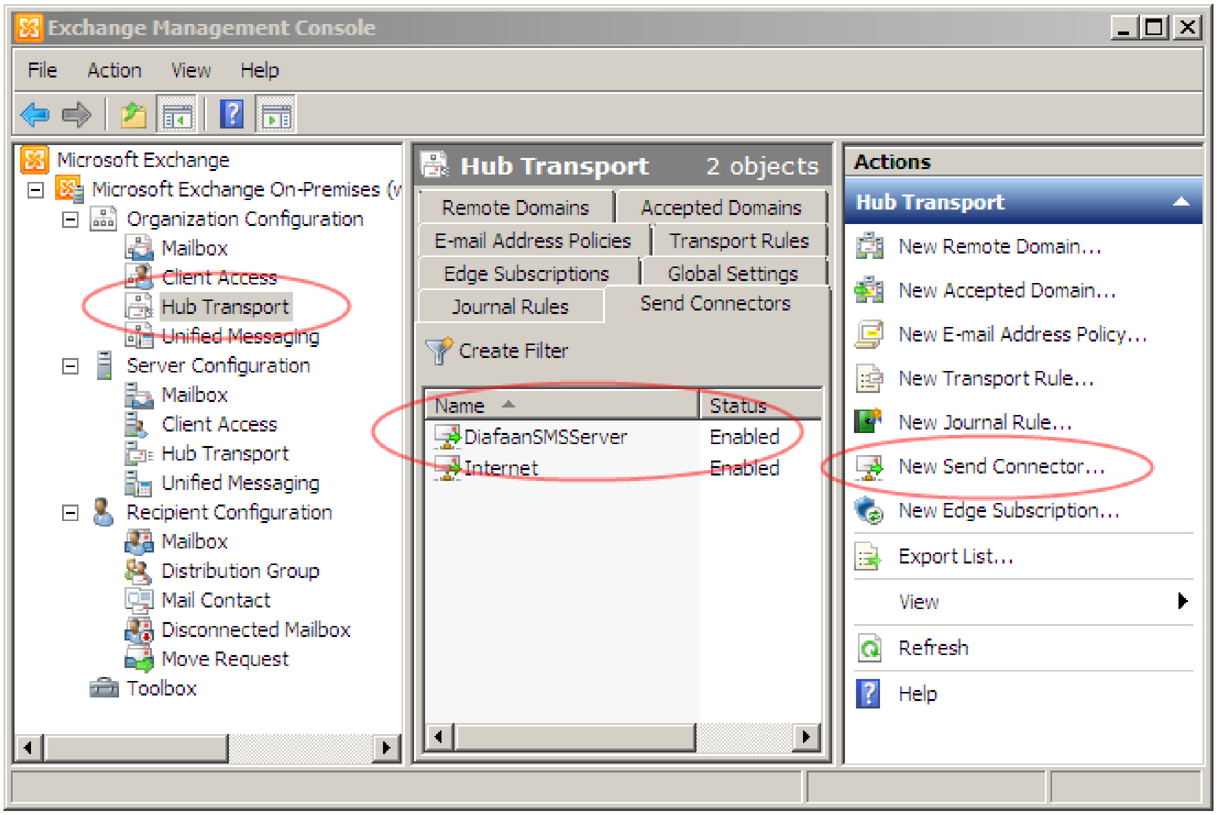
Older versions of Microsoft Exchange Server can also be connected with Diafaan SMS Server, but only when the two programs are installed on different servers in your network.

Step 2: Add an Email Connector to Diafaan SMS Server

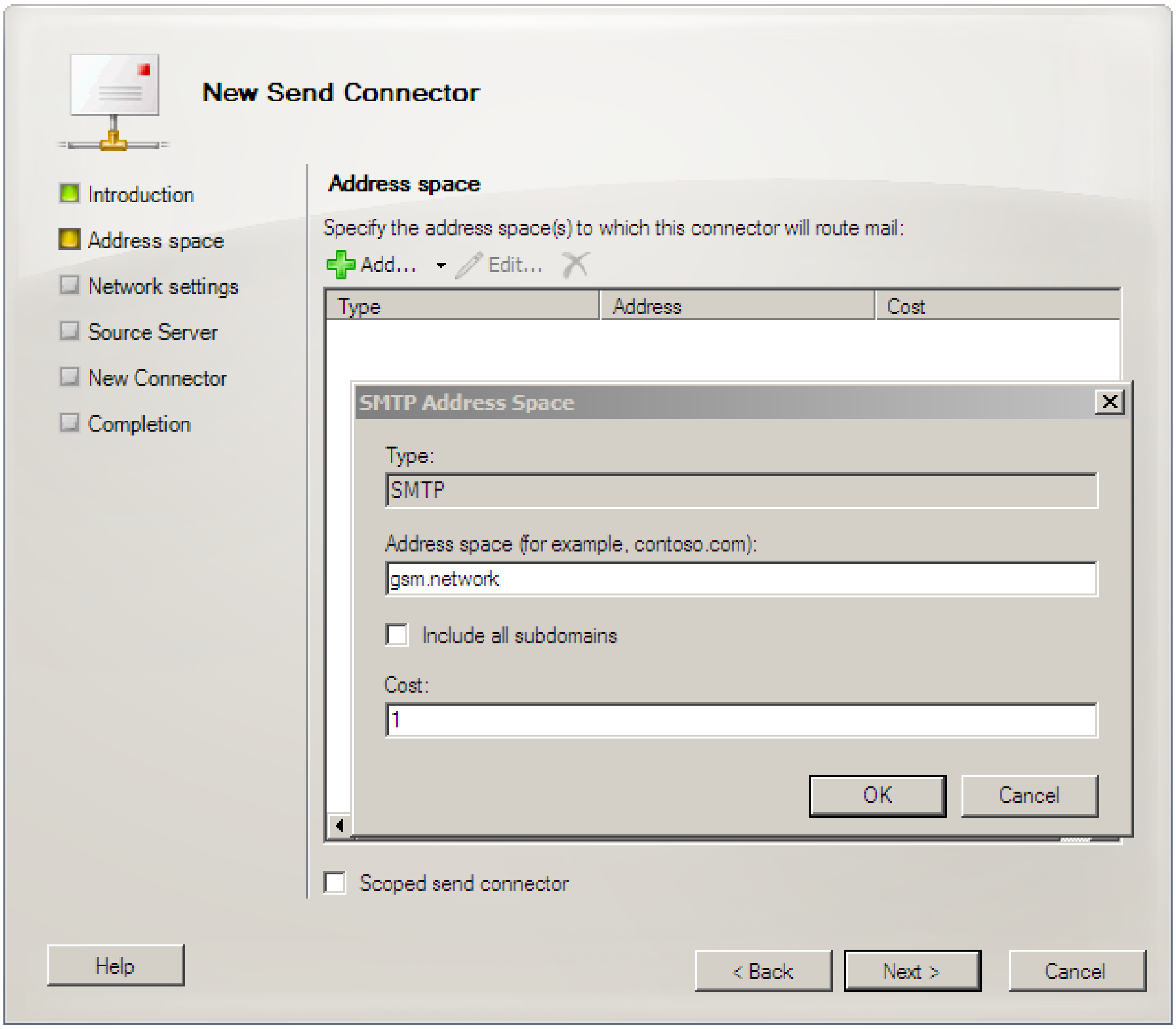
The second step in the integration with Microsoft Exchange Server is to add an Email Connector in Diafaan SMS Server. You can select the default options in the setup wizard, specific options necessary to integrate with your Exchange Server installation can be changed later in the Email Connector Properties.

Step 3: Add a Send Connector to Exchange

Exchange uses Send Connectors to forward email to an external email server. You can add a new Send Connector in the Exchange Management Console.



Send Connector Address Space
The address space defines which email messages will be forwarded to Diafaan SMS Server. In this example the address space is set to **gsm.network**. This means that all email messages to the domain gsm.network (e.g. +44xxxxxxxx@gsm.network) will be sent through Diafaan SMS Server.



Specify the address space(s) to which this connector will route mail:

Type: SMTP

Address space (for example, contoso.com): gsm.network

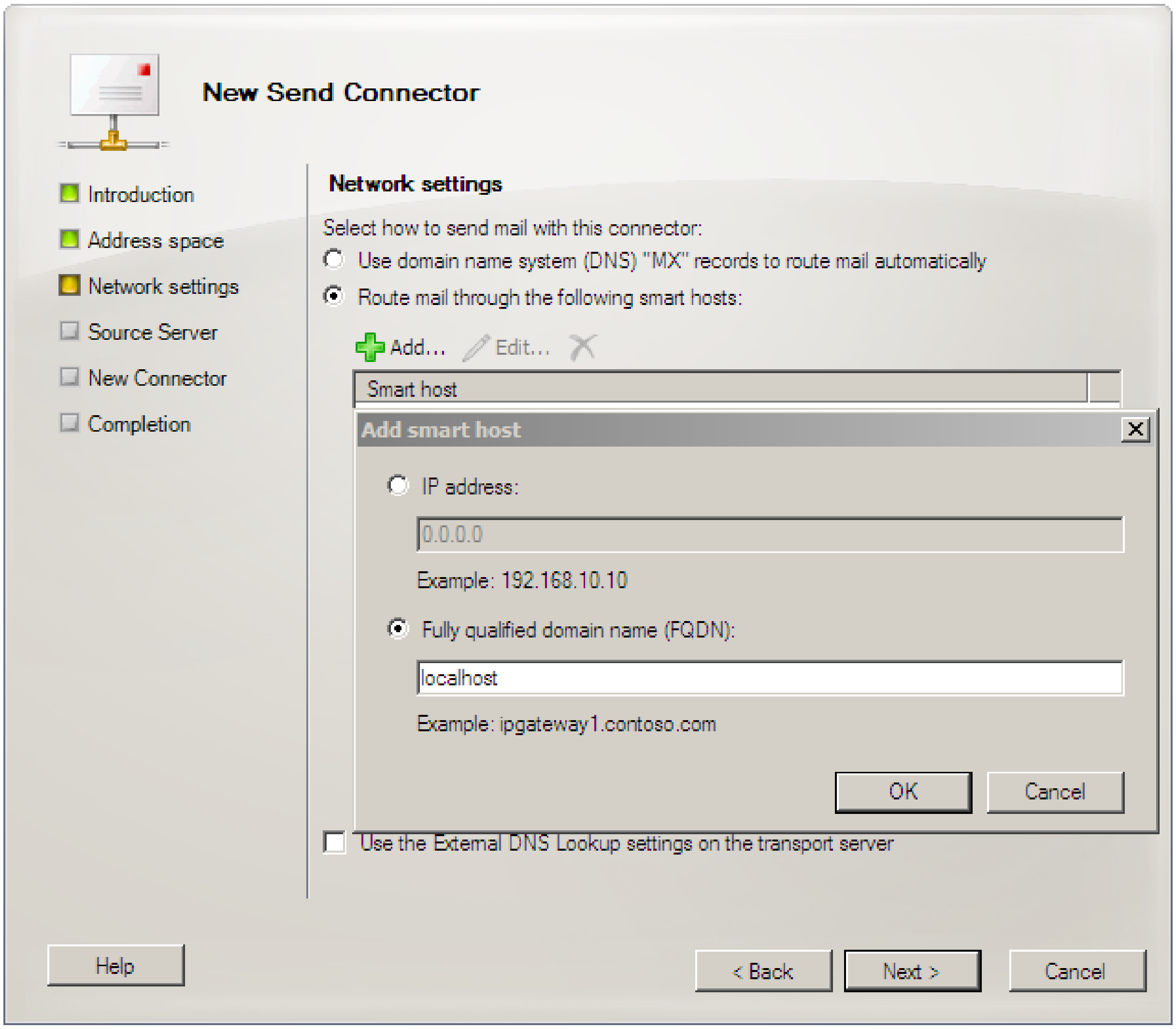
□ Include all subdomains

Cost: 1

□ Scoped send connector

Send Connector Smart Host

In the Network Settings you can specify the location of the Diafaan SMS Server Email Connector. In the picture below the location is set to **localhost**, the same computer where Exchange is installed. If Diafaan SMS Server is installed on another computer you can enter the IP address or fully qualified domain name of the computer.



New Send Connector

Network settings

Select how to send mail with this connector:

○ Use domain name system (DNS) "MX" records to route mail automatically

● Route mail through the following smart hosts:

Add smart host

○ IP address:

0.0.0.0

Example: 192.168.10.10

● Fully qualified domain name (FQDN):

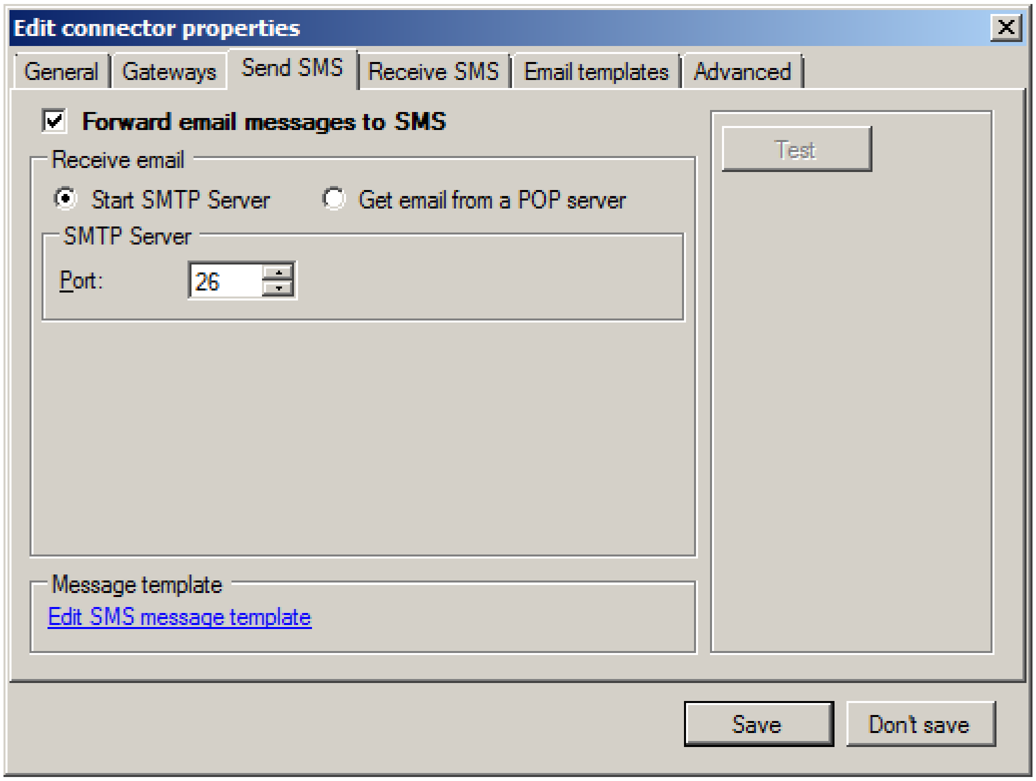
localhost

Example: ipgateway1.contoso.com

□ Use the External DNS Lookup settings on the transport server

Step 3a: Change Diafaan SMTP Port

In the installation example in step 3 there is an obvious conflict between Diafaan SMS Server and Exchange. Since both programs are installed on the same computer and both use an SMTP Server on the same SMTP port one of the SMTP Servers will not work (probably the SMTP Server used by Diafaan SMS Server). To solve this conflict the SMTP port in the Email Connector of Diafaan SMS Server should be set to an unused port, for example 26. You can also change the message template to define which parts of the email messages are sent as the SMS message.



☑ Forward email messages to SMS

Receive email

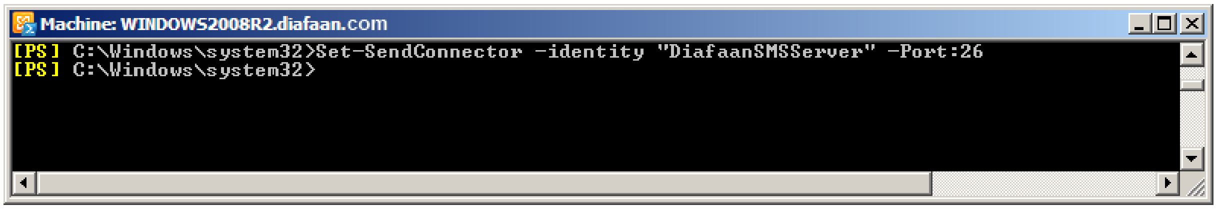
● Start SMTP Server ○ Get email from a POP server

SMTP Server

Port: 26

Message template

The same port number must now be set in the Send Connector in Exchange. Since this is not possible in the Exchange Management Console, you have to do this with a shell command in the Exchange Management Shell:
Set-SendConnector -identity "DiafaanSMSServer" -Port:26

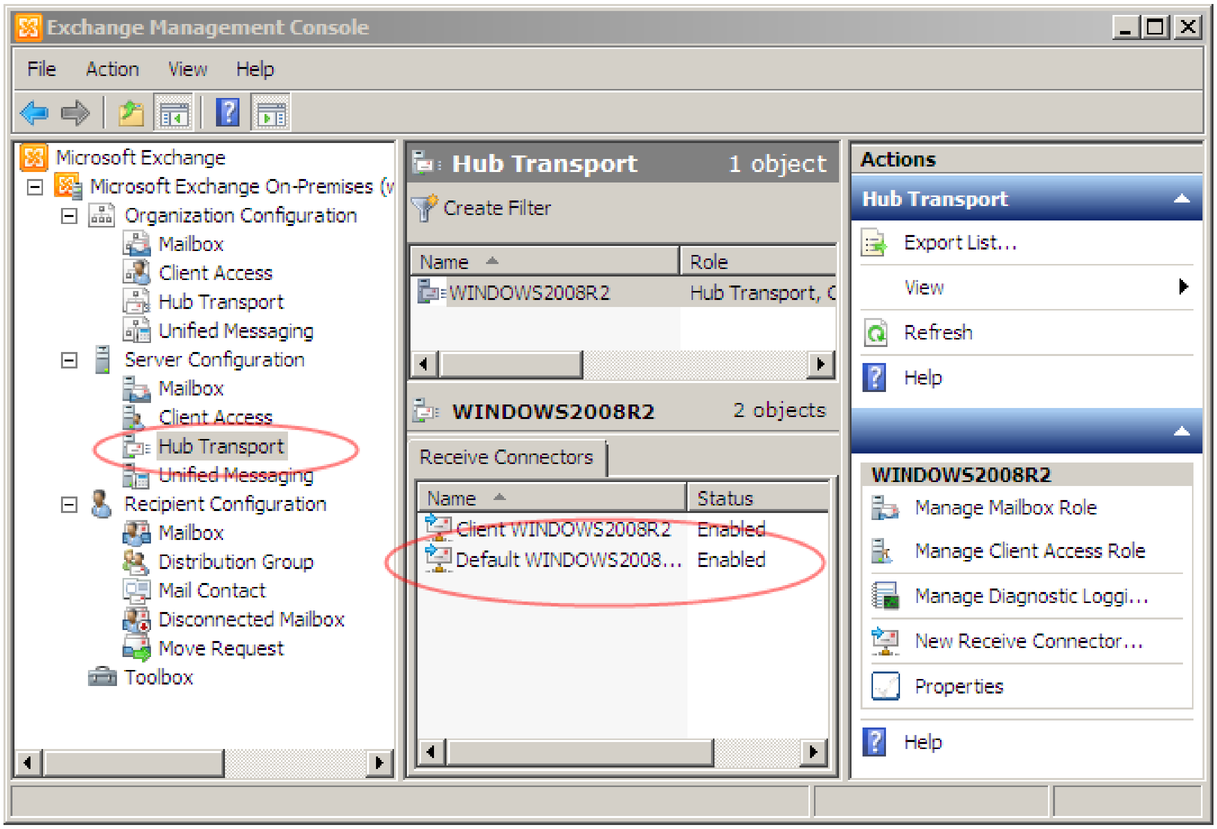


```
[PS] C:\Windows\system32>Set-SendConnector -identity "DiafaanSMSServer" -Port:26
[PS] C:\Windows\system32>
```

Sending SMS messages from one of the Outlook clients should now work, you can try it out by sending an email message to email address **<gsm-number>@gsm.network**, (depending on the address space for the Send Connector you supplied).

Step 4: Use the Receive Connector in Exchange

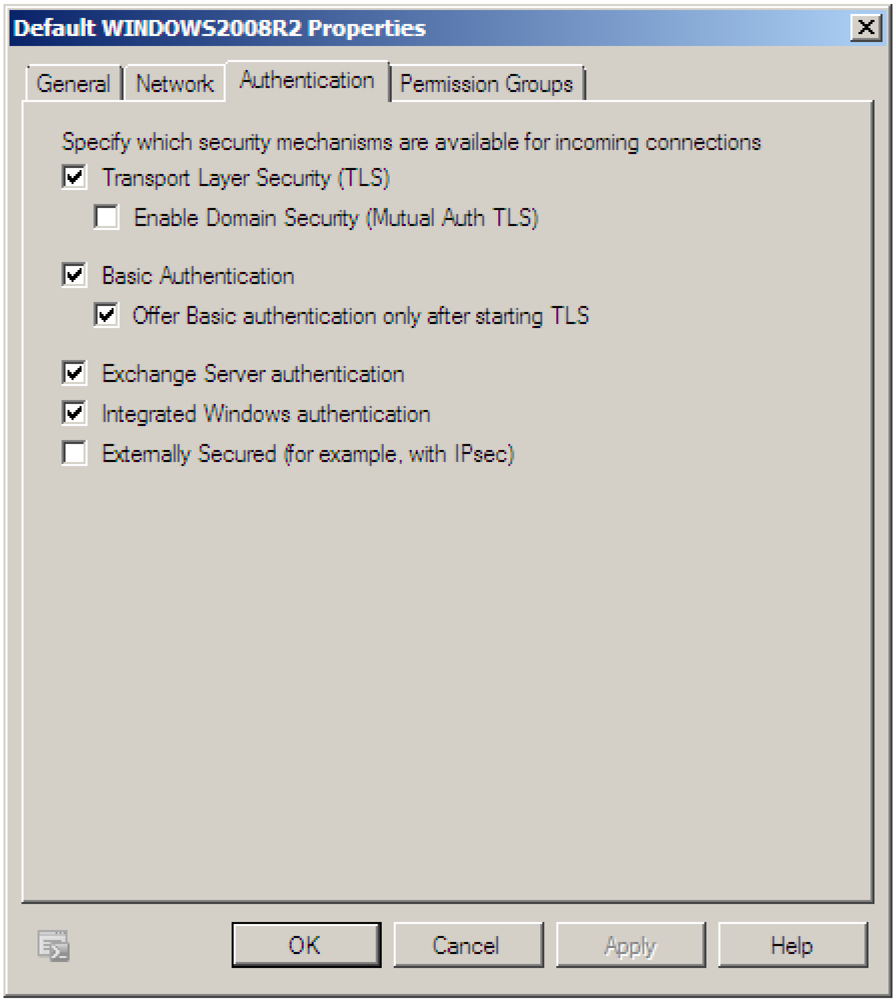
If you want to use Diafaan SMS Server to forward received SMS messages to email users you can use (one of) the SMTP Server in Exchange. The default SMTP Server in Exchange accepts email from an SMTP Client with basic authentication after TLS encryption is enabled. Diafaan SMS Server can use basic authentication in plain text or over a TLS encrypted connection.



Name	Role
WINDOWS2008R2	Hub Transport, C

Name	Status
Client WINDOWS2008R2	Enabled
Default WINDOWS2008...	Enabled

Check to see if Basic Authentication is enabled.



Default WINDOWS2008R2 Properties

Specify which security mechanisms are available for incoming connections

☑ Transport Layer Security (TLS)

☐ Enable Domain Security (Mutual Auth TLS)

☑ Basic Authentication

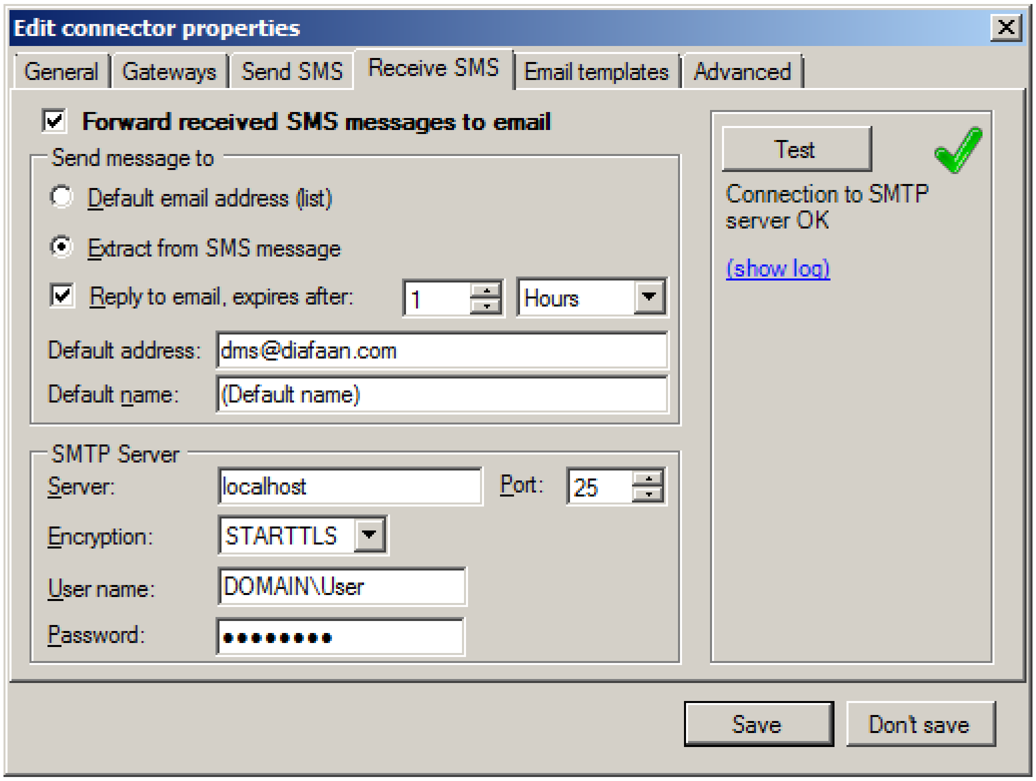
☑ Offer Basic authentication only after starting TLS

☑ Exchange Server authentication

☑ Integrated Windows authentication

☐ Externally Secured (for example, with IPsec)

Set the email properties
In the Receive SMS tab page you can specify what Diafaan SMS Server should do with the received SMS messages. In the example below the SMS message is replied to a previously sent email message (if available)



☑ Forward received SMS messages to email

Send message to

○ Default email address (list)

● Extract from SMS message

☑ Reply to email, expires after: 1 Hours

Default address: dms@diafaan.com

Default name: (Default name)

SMTP Server

Server: localhost Port: 25

Encryption: STARTTLS

User name: DOMAIN\User

Password: •••••••••

Connection to SMTP server OK

(show log)

Set the Exchange SMTP Server
If Exchange Server is installed on the same computer as Diafaan SMS Server you can set the server to **localhost**. If it is installed on another server, you have to enter the IP address or Fully qualified Domain Name of the computer where Exchange is installed. If your Exchange Server accepts email over a TLS encrypted connection you should also set encryption to **STARTTLS**.

To change the format of the email message, select **Edit email template for received SMS messages** on the **Email templates** page. The From Address Template can be set to **[from]@gsm.network** (depending on the address space for the Send Connector) to allow receivers of the SMS message to reply to the message.

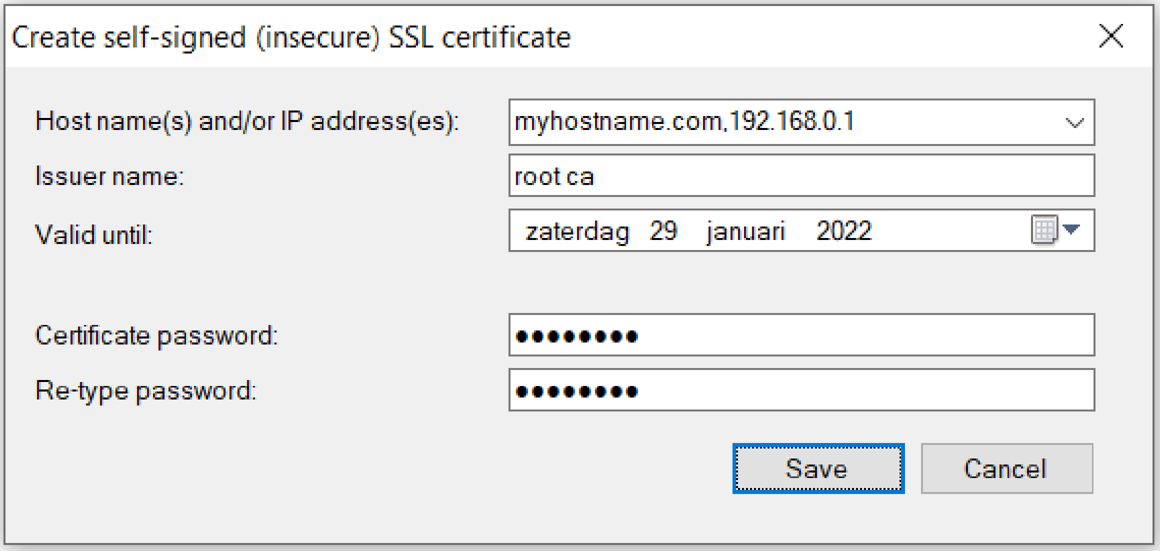
That is all, forwarding received SMS messages to email should now work. You can test it by sending an SMS message to the GSM modem.

3.8 Add encryption option to connectors

Some of the IP servers in Diafaan SMS Server, like the HTTP server in the Web Connector and the SMPP server in the SMPP Connector, offer support for encrypted SSL connections. The IP servers in Diafaan SMS Server can use an X509 (OpenSSL) certificate that is stored in a password protected .pfx file. For optimal security and compatibility the certificate must be signed by a trusted root certificate authority and be set to the name of the host name or IP address of the server where Diafaan SMS Server is installed. But it is also possible to use a self-signed certificate if the SSL clients will accept them, however some SSL clients only accept connections to properly signed certificates with a perfect host name match.

Diafaan SMS Server can be used to generate a self-signed certificate for test purposes or when encryption without secure server identification is sufficient. To create the certificate file select menu options **Help-Create SSL certificate**, fill in the Host name or IP address and the password of the certificate file and select **Save** to create the certificate file.

You can fill in multiple DNS names and/or IP addresses in the first field (separated by comma's) if the server can be accessed by more than one DNS name or IP address.


Create self-signed (insecure) SSL certificate

Host name(s) and/or IP address(es): myhostname.com,192.168.0.1

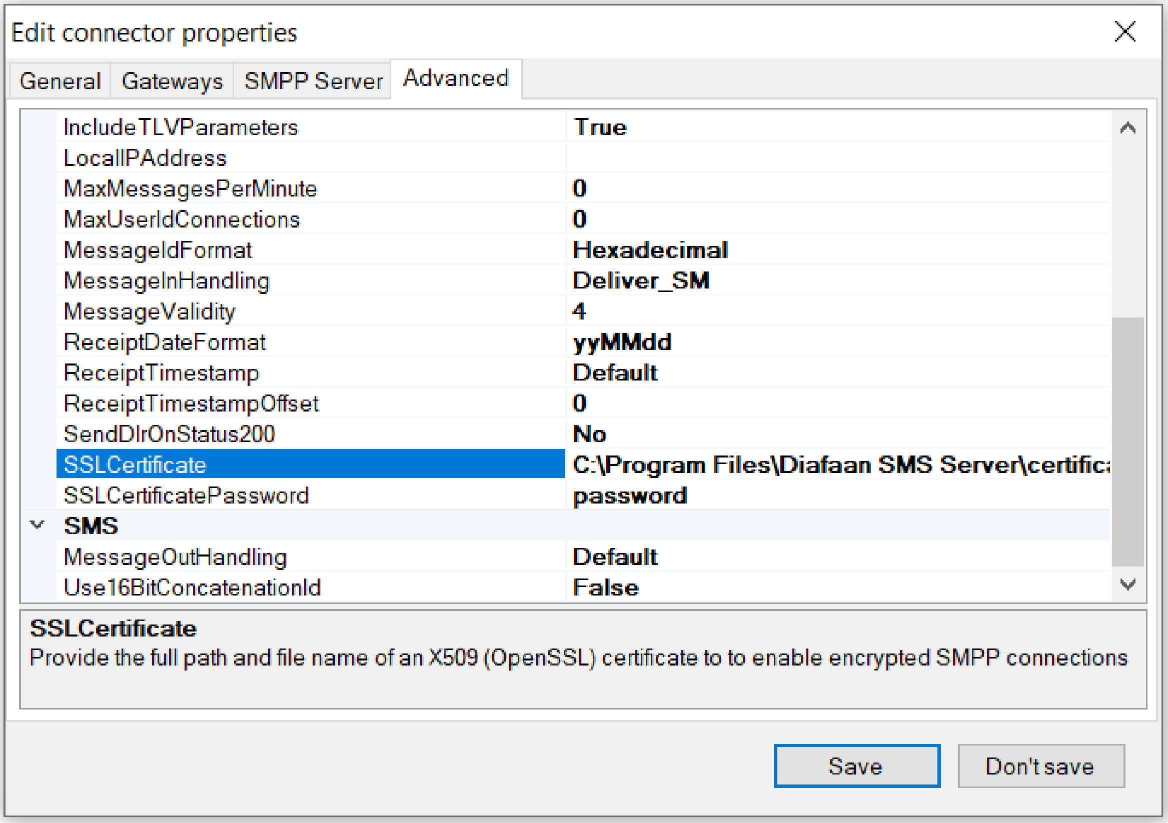
Issuer name: root ca

Valid until: zaterdag 29 januari 2022

Certificate password: ●●●●●●●●

Re-type password: ●●●●●●●●

To enable encryption, the certificate file has to be added to the connector. In the example below there are two properties that must be set in the advanced settings of the SMPP Connector: **SSLCertificate** and **SSLCertificatePassword**:



IncludeTLVParameters	True
LocalIPAddress	
MaxMessagesPerMinute	0
MaxUserIdConnections	0
MessageIdFormat	Hexadecimal
MessageInHandling	Deliver_SM
MessageValidity	4
ReceiptDateFormat	yyMMdd
ReceiptTimestamp	Default
ReceiptTimestampOffset	0
SendDlrOnStatus200	No
SSLCertificate	C:\Program Files\Diafaan SMS Server\certific
SSLCertificatePassword	password
SMS	
MessageOutHandling	Default
Use16BitConcatenationId	False

SSLCertificate
Provide the full path and file name of an X509 (OpenSSL) certificate to to enable encrypted SMPP connections

The **SSLCertificate** property is set to full path and file name of the certificate file. Since this file is password protected, the **SSLCertificatePassword** must also be set.

When the certificate is installed correctly the connector now accepts only secure connections from clients.

3.9 Access your database

Learn how to send and receive SMS messages using an existing SQL database with Diafaan SMS Server. Get instructions about the installation of the ODBC or OLE DB driver, how to connect to the database with a connection string and use the database tables and fields of your database.

Connect to an existing database

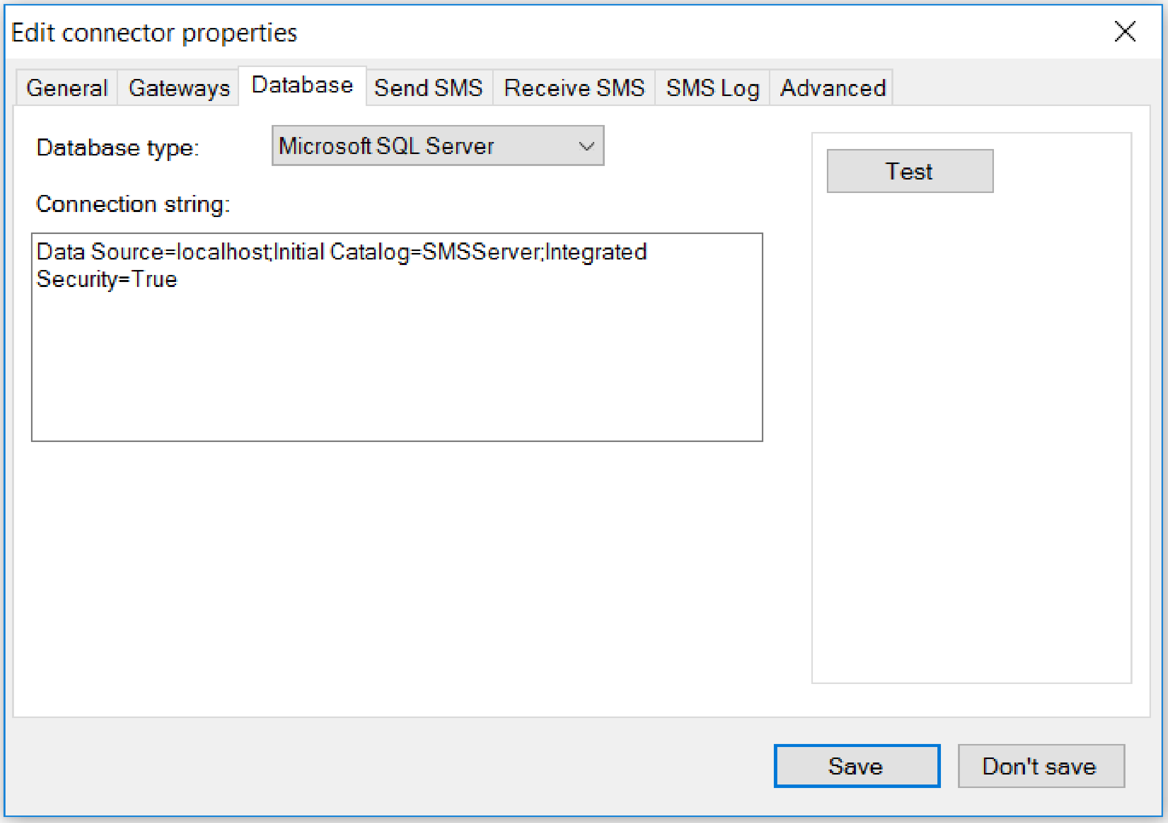
Install the driver

Diafaan SMS Server supports three driver models to connect to your database, Microsoft SQL Server, ODBC and OLE DB. Database drivers for some databases like Microsoft SQL Server or dBase are already installed by default in Windows but for most databases you will have to install the ODBC or OLE DB driver for your specific database. Usually you can find the right drivers on the website of the database manufacturer.

Create the connection string

All information necessary to connect to the database is stored in the connection string. The format of the connection string is different for each ODBC/OLE DB driver and database. Examples of connection strings to connect to a large number of databases can be found on the website http://www.connectionstrings.com.

In the example below you can see that the connection string contains information about the server on which the database runs, the name of the database and the name and password needed to connect to the database. You can use the **Test** button to test if the connection string is set up right.



Database type: Microsoft SQL Server

Connection string:

```
Data Source=localhost;Initial Catalog=SMSServer;Integrated
Security=True
```

Use the database tables and fields

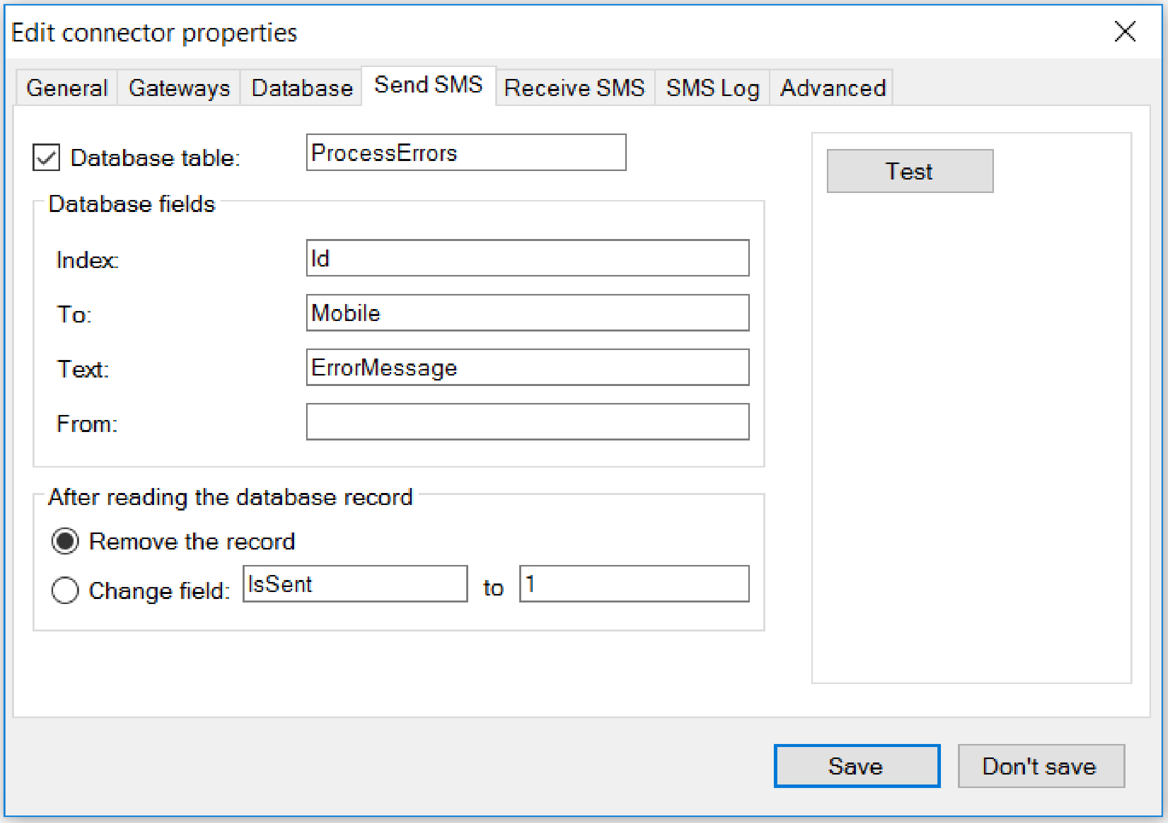
Database tables

The SQL connector supports three database tables, one for messages to send (MessageOut), one to store received messages (MessageIn) and a table to log sent or failed calls (MessageLog). For each database table you can provide the table name of your database and each database field name you want to use.

Send SMS table

Diafaan SMS Server polls the Send SMS table for new records and uses the information to send SMS messages. After the records are read in Diafaan SMS Server either deletes the record or updates one of the fields to a fixed value to prevent that messages are sent more than once. The only required field for the Send SMS table is **Index** which should have a unique value, without this field Diafaan SMS Server does not know which records should be deleted or updated.

example 1
If you have a database table **ProcessErrors** with three fields, a unique **Id** field, a **Mobile** field that contains a GSM number and an **ErrorMessage** field that contains the text that should be sent to the mobile number you can enter these table and field names as in the screenshot below:



Edit connector properties

☑ Database table: ProcessErrors

Database fields

Index: Id

To: Mobile

Text: ErrorMessage

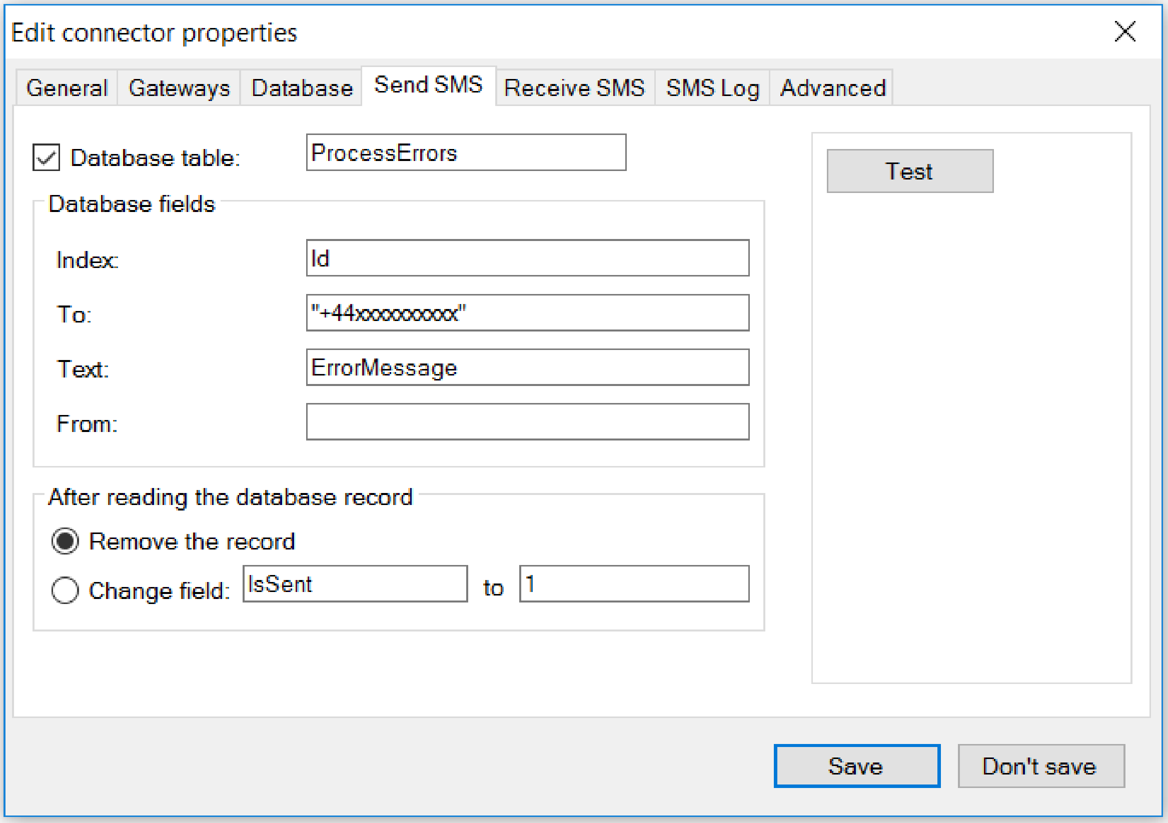
From:

After reading the database record

● Remove the record

○ Change field: IsSent to 1

example 2
If you have a database with the same database table but without the **Mobile** field it is also possible to use a fixed value instead of a database field name by enclosing the text in double quote characters. In this case all messages will be sent to GSM number +44xxxxxxxx. Quoted strings can be used in all values for the Send SMS table except the **Index** and **Change field to** values.



☑ Database table: ProcessErrors

Database fields

Index: Id

To: "+44xxxxxxxxxx"

Text: ErrorMessage

From:

After reading the database record

● Remove the record

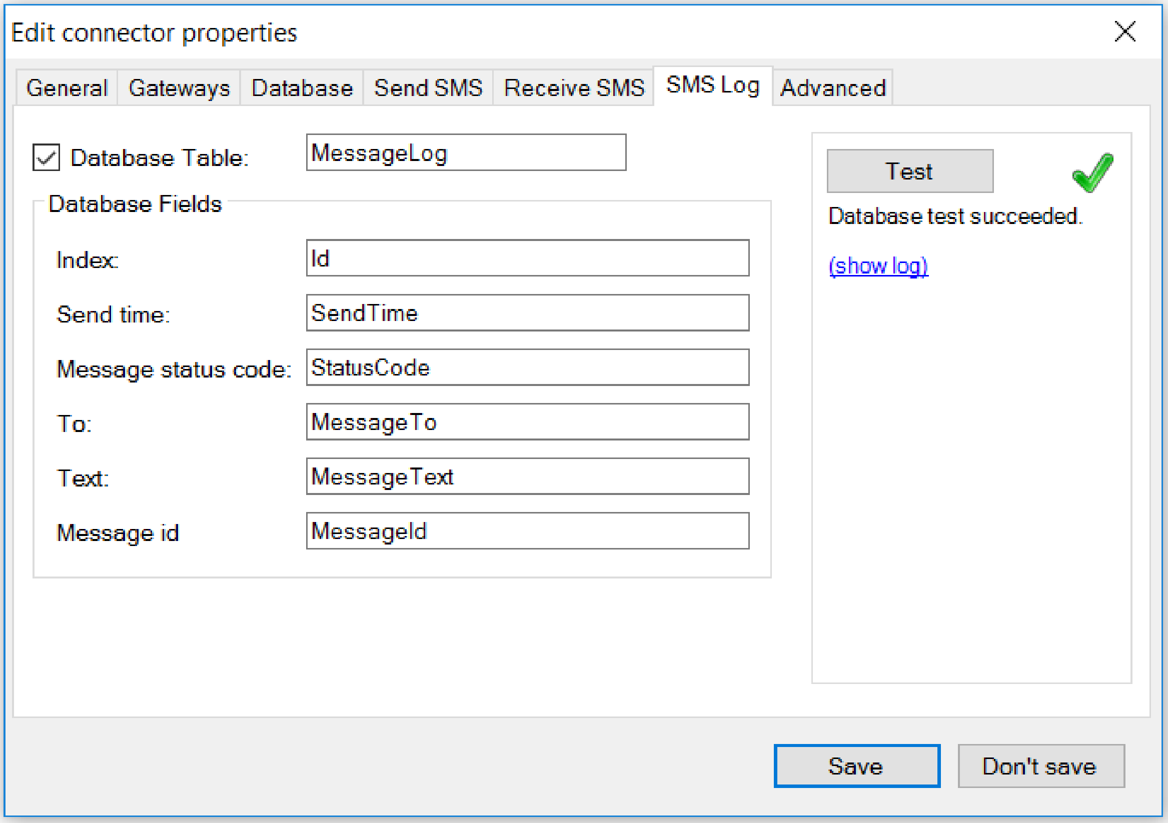
○ Change field: IsSent to 1

Receive SMS table

In the Receive SMS tab page the database table name and field names where received SMS messages are stored can be added. All fields are optional, you can choose to add only the information about the received SMS message you are interested in to the database.

SMS Log table

Messages that are sent or failed to send can be stored in the SMS Log database table. All fields are optional but when the Index and Message Id fields are not used, the status of the message cannot be updated. Status updates occur when a the status of a previously sent message changes. This happens for example when the message is received on the GSM phone or if the message could not be delivered on the GSM phone because it was switched off.



☑ Database Table: MessageLog

Database Fields

Index: Id

Send time: SendTime

Message status code: StatusCode

To: MessageTo

Text: MessageText

Message id MessageId

Database test succeeded.

The **Message Status Code** field stores the message result. Status codes between 200 and 299 indicate that the message is sent successfully. Status codes between 300 and 399 means that the message failed.

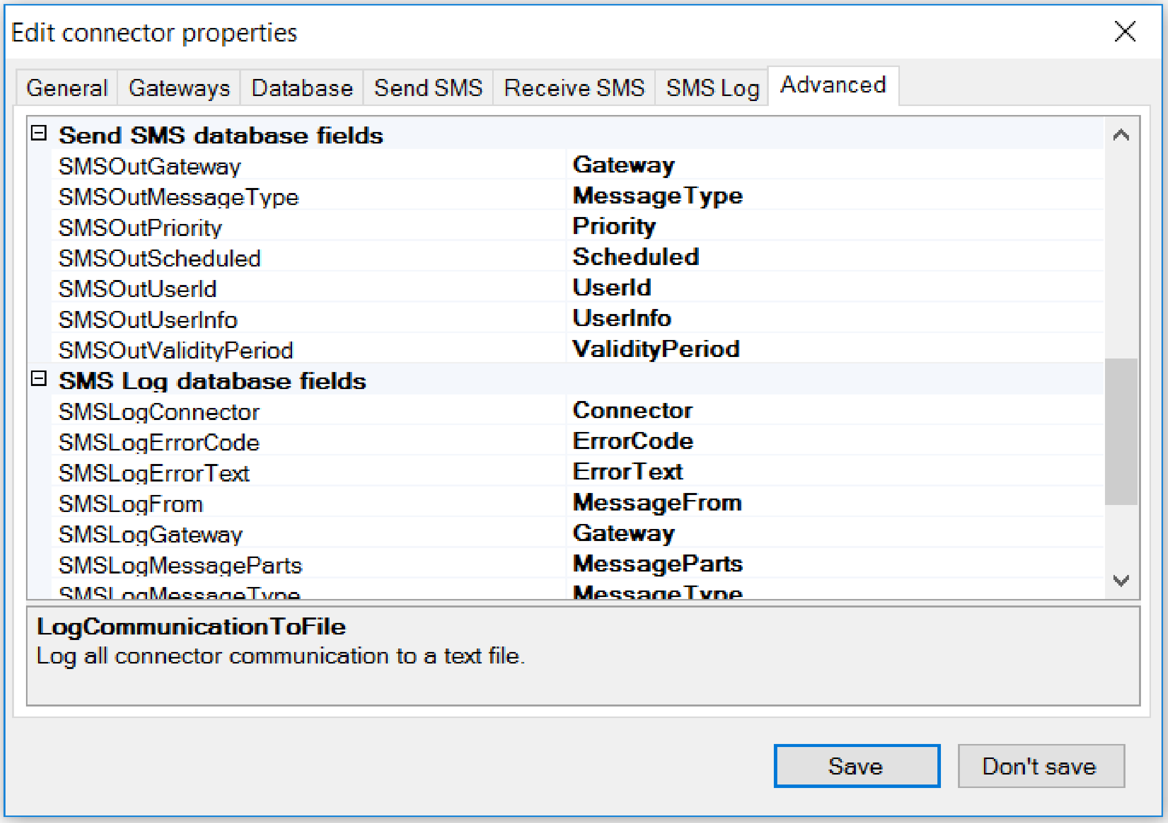
status codes

200	Message sent and accepted by the gateway.
201	Message received on the GSM phone.
300	Message failed or rejected by the gateway.
301	Status error received after the message was accepted by the gateway

Advanced database fields

In the advanced tab page you can specify additional database fields that you want to use. The purpose of most fields should be clear, but some might need explaining.

Send SMS database fields



⊟ Send SMS database fields	
SMSOutGateway	**Gateway**
SMSOutMessageType	**MessageType**
SMSOutPriority	**Priority**
SMSOutScheduled	**Scheduled**
SMSOutUserId	**UserId**
SMSOutUserInfo	**UserInfo**
SMSOutValidityPeriod	**ValidityPeriod**
⊟ **SMS Log database fields**	
SMSLogConnector	**Connector**
SMSLogErrorCode	**ErrorCode**
SMSLogErrorText	**ErrorText**
SMSLogFrom	**MessageFrom**
SMSLogGateway	**Gateway**
SMSLogMessageParts	**MessageParts**
SMSLogMessageType	**MessageType**

LogCommunicationToFile
Log all connector communication to a text file.

SMSOutGateway
Use this field if you want to send the message through one or more specific gateways. See Route messages for more information about routing options.

SMSOutUserId and SMSOutUserInfo
These two fields are not used by Diafaan SMS Server, but are transparent and written without change in the **SMSLogUserId** and **SMSLogUserInfo** fields of the **MessageLog** database table. The fields can be used to track the messages you send or to add your own information to the log database table.

3.9.1 Microsoft SQL Server (Express)

Learn how to send and receive SMS messages using a Microsoft SQL Server (Express) database with Diafaan SMS Server. Let Diafaan SMS Server create the database tables or create the database tables manually with an SQL script.

Microsoft SQL Server

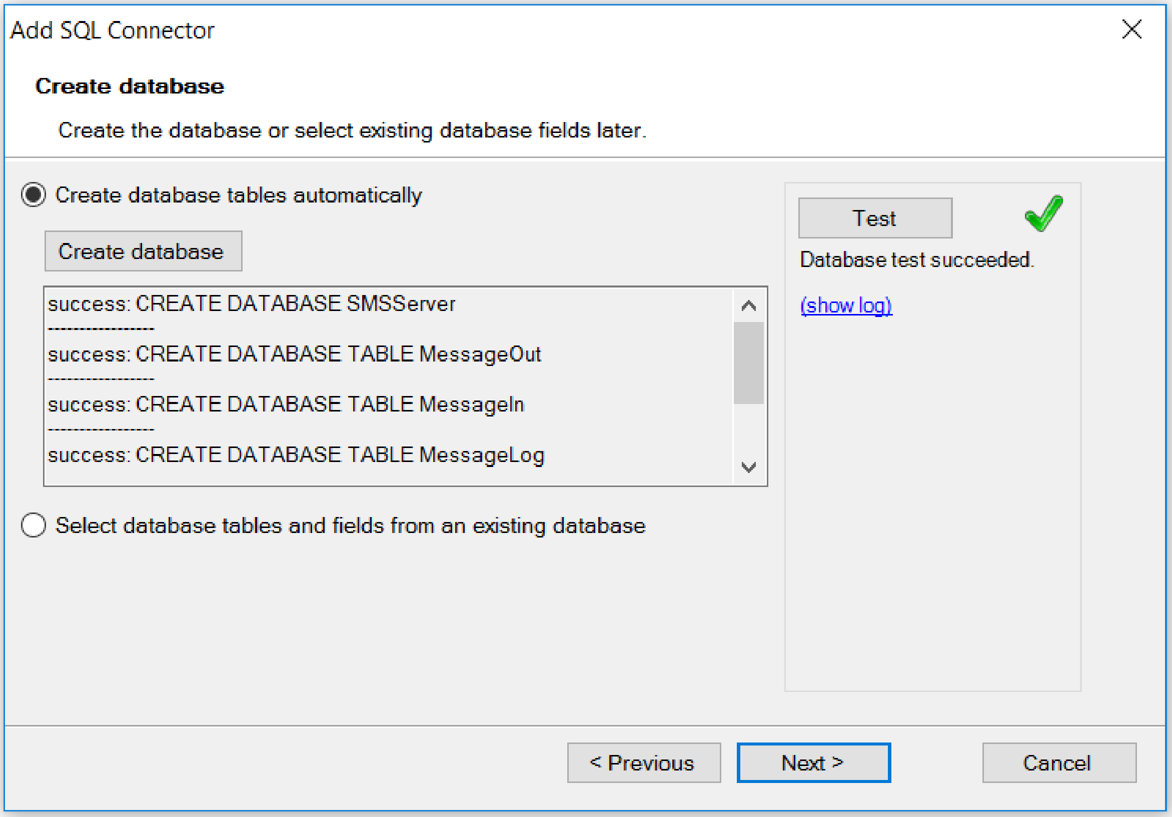
Microsoft SQL Server is Microsoft's high performance SQL Server relational database management system. Drivers to access the database are built-in in Microsoft Windows so it is not necessary to install any additional drivers to use the database.

Microsoft SQL Server Express

Microsoft SQL Server Express is a freely-downloadable and distributable version of Microsoft's SQL Server relational database management system. For all intent and purposes it is compatible with the regular version of Microsoft SQL Server and uses the same drivers. The only practical difference is that **\SQLEXPRESS** should be added to the data source (server) name in the connection string.

Create the SMSServer database

You can use the setup wizard to create the database tables and fields for the MessageOut, MessageIn and MessageLog tables.



Add SQL Connector

Create database

Create the database or select existing database fields later.

● Create database tables automatically

```
success: CREATE DATABASE SMSServer
-----------------
success: CREATE DATABASE TABLE MessageOut
-----------------
success: CREATE DATABASE TABLE MessageIn
-----------------
success: CREATE DATABASE TABLE MessageLog
```

○ Select database tables and fields from an existing database

Database test succeeded.

The setup wizard creates a database with the following table and field layout:

```
CREATE DATABASE SMSServer;
CREATE TABLE MessageOut (
    Id int IDENTITY (1,1) PRIMARY KEY,
    MessageTo nvarchar(80),
    MessageFrom nvarchar(80),
    MessageText nvarchar(max),
    MessageType nvarchar(80),
    MessageGuid nvarchar(max),
    MessageInfo nvarchar(max),
    Gateway nvarchar(80),
    UserId nvarchar(80),
    UserInfo nvarchar(max),
    Priority int,
    Scheduled datetime,
    ValidityPeriod int,
    TLVList nvarchar(max),
    IsSent bit NOT NULL DEFAULT 0,
    IsRead bit NOT NULL DEFAULT 0);

CREATE INDEX IDX_IsRead ON MessageOut (IsRead);

CREATE TABLE MessageIn (
    Id int IDENTITY (1,1) PRIMARY KEY,
    SendTime datetime NOT NULL DEFAULT getdate(),
    ReceiveTime datetime,
    MessageFrom nvarchar(80),
    MessageTo nvarchar(80),
    SMSC nvarchar(80),
    MessageText nvarchar(max),
    MessageType nvarchar(80),
    MessageParts int,
    MessagePDU nvarchar(max),
    Gateway nvarchar(80),
    UserId nvarchar(80));

CREATE TABLE MessageLog (
    Id int IDENTITY (1,1) PRIMARY KEY,
    SendTime datetime NOT NULL DEFAULT getdate(),
    ReceiveTime datetime,
    StatusCode int,
    StatusText nvarchar(80),
    MessageTo nvarchar(80),
    MessageFrom nvarchar(80),
    MessageText nvarchar(max),
    MessageType nvarchar(80),
    MessageId nvarchar(80),
    MessageGuid nvarchar(max),
    MessageInfo nvarchar(max),
    ErrorCode nvarchar(80),
    ErrorText nvarchar(max),
    Gateway nvarchar(80),
    MessageParts int,
    MessagePDU nvarchar(max),
    Connector nvarchar(80),
    UserId nvarchar(80),
    UserInfo nvarchar(max));
```

```
CREATE INDEX IDX_MessageId ON MessageLog (MessageId, SendTime);
```

3.9.2 MySQL database

Learn how to send and receive SMS messages using a MySQL database with Diafaan SMS Server. Get instructions about the installation of the ODBC driver, create the database with an SQL script and how to connect to the database with the connection string for MySQL.

MySQL

MySQL is one of the most popular database programs available. Diafaan SMS Server can use an ODBC driver to connect to MySQL. For this article MySQL version 8.0.20 was used with the MyODBC 8.0 driver.

Install the MyODBC driver

Before Diafaan SMS Server can send and receive SMS messages from the database you have to make sure the MyODBC 8.0 driver is installed on the computer running Diafaan SMS Server. You can find the MyODBC driver on the MySQL website.

Create the SMSServer database

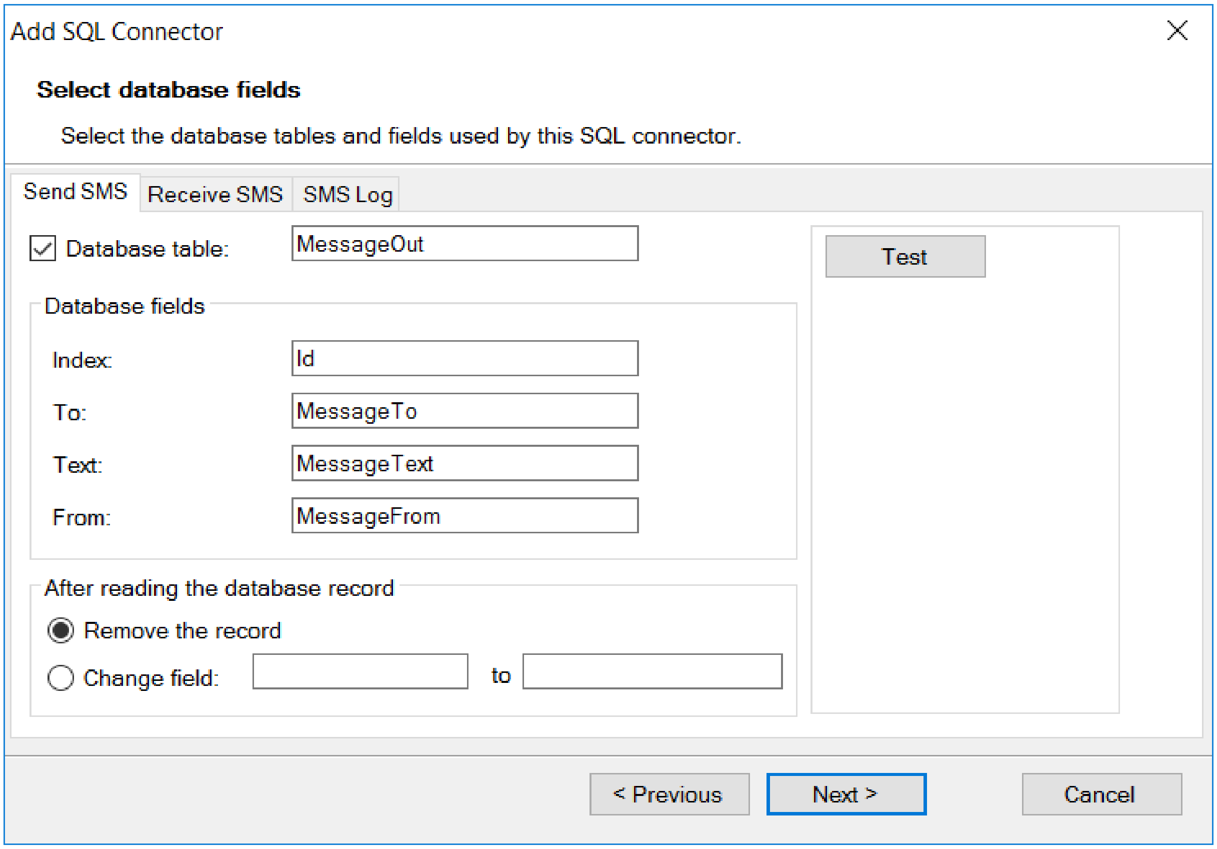
You can use the MySQL command line client to create the database tables. Of course it is also possible to use an existing database, but if you want to be sure you can use all options available in Diafaan SMS Server you may use the SQL commands below necessary to create the database tables to send and receive SMS messages and to keep the send log.

```
CREATE DATABASE SMSServer DEFAULT CHARACTER SET utf8mb4;

CREATE TABLE MessageOut (
    Id INT NOT NULL AUTO_INCREMENT PRIMARY KEY,
    MessageTo VARCHAR(80),
    MessageFrom VARCHAR(80),
    MessageText TEXT,
    MessageType VARCHAR(80),
    MessageGuid TEXT,
    MessageInfo TEXT,
    Gateway VARCHAR(80),
    UserId VARCHAR(80),
    UserInfo TEXT,
    Priority INT,
    Scheduled DATETIME,
    ValidityPeriod INT,
    TLVList TEXT,
    IsSent TINYINT NOT NULL DEFAULT 0,
    IsRead TINYINT NOT NULL DEFAULT 0) CHARACTER SET utf8mb4;

CREATE INDEX IDX_IsRead ON MessageOut (IsRead);

CREATE TABLE MessageIn (
    Id INT NOT NULL AUTO_INCREMENT PRIMARY KEY,
    SendTime DATETIME,
    ReceiveTime DATETIME,
    MessageFrom VARCHAR(80),
    MessageTo VARCHAR(80),
    SMSC VARCHAR(80),
    MessageText TEXT,
    MessageType VARCHAR(80),
    MessageParts INT,
    MessagePDU TEXT,
    Gateway VARCHAR(80),
    UserId VARCHAR(80)) CHARACTER SET utf8mb4;

CREATE TABLE MessageLog (
    Id INT NOT NULL AUTO_INCREMENT PRIMARY KEY,
    SendTime DATETIME,
    ReceiveTime DATETIME,
    StatusCode INT,
    StatusText VARCHAR(80),
    MessageTo VARCHAR(80),
    MessageFrom VARCHAR(80),
    MessageText TEXT,
    MessageType VARCHAR(80),
    MessageGuid TEXT,
    MessageInfo TEXT,
    MessageId VARCHAR(80),
    ErrorCode VARCHAR(80),
    ErrorText TEXT,
    Gateway VARCHAR(80),
    MessageParts INT,
    MessagePDU TEXT,
    Connector VARCHAR(80),
    UserId VARCHAR(80),
    UserInfo TEXT) CHARACTER SET utf8mb4;

CREATE INDEX IDX_MessageId ON MessageLog (MessageId, SendTime);
```

Add the SQL Connector

Now you can add the new database to Diafaan SMS Server. First create a new SQL Connector with the connector wizard and select ODBC as the database type. On the next page you can provide the ODBC connection string, the following connection string should be adapted for your database installation:

```
Driver={MySQL ODBC 8.0
Driver};Server=myServerAddress;charset=UTF8MB4;Database=myDataBase;User=myUsername;
Password=myPassword;Option=3;
```



Add SQL Connector

Select database fields

Select the database tables and fields used by this SQL connector.

☑ Database table: MessageOut

Database fields

Index: Id

To: MessageTo

Text: MessageText

From: MessageFrom

After reading the database record

● Remove the record

○ Change field: to

Select the database fields

On the next wizard pages you can select the database tables and fields you want Diafaan SMS Server to use. You can only select the most used fields here. Additional database fields can be selected later on the advanced tab page in the connector properties dialog.

For more information see **Access your database**.

It is a good idea to change the database flavor option on the same advanced tab page to MySQL. This will instruct Diafaan SMS Server to use a specific SQL syntax for MySQL enhancing performance somewhat.

3.9.3 Oracle database

Learn how to send and receive SMS messages using an Oracle database with Diafaan SMS Server. Get instructions about the installation of the OLE DB driver, create the database with an SQL script and how to connect to the database with the connection string for Oracle.

Oracle database

Diafaan SMS Server can use OLE DB to connect to the Oracle database. For this article Oracle 10g Express Edition was used with the Oracle 10g Provider for OLE DB.

Install the Oracle OLE DB provider

Before Diafaan SMS Server can send and receive SMS messages from the database you have to make sure the OLE DB provider is installed on the computer running Diafaan SMS Server. You can get the Oracle OLE DB provider software on the Oracle website.

Create the SMSServer database

On the database home page of your Oracle server you can use the SQL commands to create the database tables. Of course it is also possible to use an existing database, but if you want to be sure you can use all options available in Diafaan SMS Server you may use the SQL commands below. Since Oracle does not support auto increment fields in the CREATE TABLE statements, a number of sequences and triggers must be created to provide the same functionality.

```
CREATE DATABASE SMSServer;

CREATE TABLE MessageOut (
    Id NUMBER(19) NOT NULL PRIMARY KEY,
    MessageTo NVARCHAR2(80),
    MessageFrom NVARCHAR2(80),
    MessageText NCLOB,
    MessageType NVARCHAR2(80),
    MessageGuid NCLOB,
    MessageInfo NCLOB,
    Gateway NVARCHAR2(80),
    UserId NVARCHAR2(80),
    UserInfo NCLOB,
    Priority NUMBER,
    Scheduled DATE,
    ValidityPeriod NUMBER,
    TLVList NCLOB,
    IsSent NUMBER(1) DEFAULT 0,
    IsRead NUMBER(1) DEFAULT 0);

CREATE INDEX IDX_IsRead ON MessageOut (IsRead);

CREATE SEQUENCE MessageOutSequence START WITH 1 INCREMENT BY 1 NOMAXVALUE;

CREATE TRIGGER MessageOutTrigger
    BEFORE INSERT ON MessageOut
    FOR EACH ROW
    BEGIN
      SELECT MessageOutSequence.NEXTVAL INTO :NEW.Id FROM DUAL;
    END;

CREATE TABLE MessageIn (
    Id NUMBER(19) NOT NULL PRIMARY KEY,
    SendTime DATE,
    ReceiveTime DATE,
    MessageFrom NVARCHAR2(80),
    MessageTo NVARCHAR2(80),
    SMSC NVARCHAR2(80),
    MessageText NCLOB,
    MessageType NVARCHAR2(80),
    MessageParts NUMBER,
    MessagePDU NCLOB,
    Gateway NVARCHAR2(80),
    UserId NVARCHAR2(80));

CREATE SEQUENCE MessageInSequence START WITH 1 INCREMENT BY 1 NOMAXVALUE;

CREATE TRIGGER MessageInTrigger
    BEFORE INSERT ON MessageIn
    FOR EACH ROW
    BEGIN
      SELECT MessageInSequence.NEXTVAL INTO :NEW.Id FROM DUAL;
    END;

CREATE TABLE MessageLog (
    Id NUMBER(19) NOT NULL PRIMARY KEY,
    SendTime DATE,
    ReceiveTime DATE,
    StatusCode NUMBER,
    StatusText NVARCHAR2(80),
    MessageTo NVARCHAR2(80),
    MessageFrom NVARCHAR2(80),
    MessageText NCLOB,
    MessageType NVARCHAR2(80),
    MessageId NVARCHAR2(80),
    MessageGuid NCLOB,
    MessageInfo NCLOB,
    ErrorCode NVARCHAR2(80),
    ErrorText NCLOB,
    Gateway NVARCHAR2(80),
    MessageParts NUMBER,
    MessagePDU NCLOB,
    Connector NVARCHAR2(80),
    UserId NVARCHAR2(80),
    UserInfo NCLOB);

CREATE INDEX IDX_MessageId ON MessageLog (MessageId, SendTime);

CREATE SEQUENCE MessageLogSequence START WITH 1 INCREMENT BY 1 NOMAXVALUE;

CREATE TRIGGER MessageLogTrigger
    BEFORE INSERT ON MessageLog
    FOR EACH ROW
    BEGIN
      SELECT MessageLogSequence.NEXTVAL INTO :NEW.Id FROM DUAL;
    END;
```

Add the SQL Connector

Now you can add the new database to Diafaan SMS Server. First create a new SQL Connector with the connector wizard and select OLE DB database as the database type. On the next page you can provide the connection string, the connection string below is an example of a connection string for the OLE DB provider and can be adapted to suit your database installation:

Connection string with TNS:

```
Provider=OraOLEDB.Oracle;Data Source=myOracleDB;User
Id=myUsername;Password=myPassword;
```

Connection string without TNS:

```
Provider=OraOLEDB.Oracle;Data Source=(DESCRIPTION=(CID=GTU_APP)
(ADDRESS_LIST=(ADDRESS=(PROTOCOL=TCP)(HOST=myServerAddress)(PORT=1521)))
(CONNECT_DATA=(SID=myOracleSID)(SERVER=DEDICATED)));User
Id=myUsername;Password=myPassword;
```

Select the database fields

On the next wizard pages you can select the database tables and fields you want Diafaan SMS Server to use. You can only select the most used fields here. Additional database fields can be selected later on the advanced tab page in the connector properties dialog.

For more information see **Access your database**.

It is a good idea to change the database flavor option on the same advanced tab page to Oracle. This will instruct Diafaan SMS Server to use a specific SQL syntax for Oracle, enhancing performance somewhat.

3.9.4 Microsoft Access

Learn how to send and receive SMS messages using a Microsoft Access database with Diafaan SMS Server. Get instructions about the OLE DB driver, create the database with an SQL script and how to connect to the database with the connection string for Microsoft Access.

Microsoft Access

Microsoft Access is not optimized for multi-user configurations and therefore it is not the best database for high-volume messaging. For low to medium message volume however it is an excellent database, especially if you already have your contact data stored in an Access database. Using the Access database connector makes it easy to add SMS functionality to your Microsoft Access or Visual Basic applications.

Use the Microsoft JET OLE DB provider

If you have Microsoft Access installed on the computer where Diafaan SMS Server is running, the OLE DB provider is already installed with it. If not, you might have to download the provider from the Microsoft website.

Create the database

You can use Microsoft Access to create the database file. The tables and fields can be entered manually in the Access user interface or with the SQL commands below. Of course it is also possible to use an existing database, but if you want to be sure you can use all options available in Diafaan SMS Server you may use the SQL commands below.

You can find an empty Access database (**SMSServer.mdb**) with this layout in the Examples folder of Diafaan SMS Server.

```
CREATE TABLE MessageOut (
    Id AUTOINCREMENT,
    MessageTo TEXT,
    MessageFrom TEXT,
    MessageText MEMO,
    MessageType TEXT,
    MessageGuid TEXT,
    MessageInfo TEXT,
    Gateway TEXT,
    UserId TEXT,
    UserInfo MEMO,
    Priority NUMBER,
    Scheduled DATE,
    ValidityPeriod NUMBER,
    TLVList TEXT,
    IsSent YESNO,
    IsRead YESNO);

CREATE UNIQUE INDEX MessageOutNDX ON MessageOut (Id);

CREATE INDEX IDX_IsRead ON MessageOut (IsRead);

CREATE TABLE MessageIn (
    Id AUTOINCREMENT,
    SendTime DATE,
    ReceiveTime DATE,
    MessageFrom TEXT,
    MessageTo TEXT,
    SMSC TEXT,
    MessageText MEMO,
    MessageType TEXT,
    MessageParts NUMBER,
    MessagePDU MEMO,
    Gateway TEXT,
    UserId TEXT);

CREATE UNIQUE INDEX MessageInNDX ON MessageIn (Id);

CREATE TABLE MessageLog (
    Id AUTOINCREMENT,
    SendTime DATETIME,
    ReceiveTime DATETIME,
    StatusCode NUMBER,
    StatusText TEXT,
    MessageTo TEXT,
    MessageFrom TEXT,
    MessageText MEMO,
    MessageType TEXT,
    MessageId TEXT,
    MessageGuid TEXT,
    MessageInfo TEXT,
    ErrorCode TEXT,
    ErrorText TEXT,
    Gateway TEXT,
    MessageParts NUMBER,
    MessagePDU MEMO,
    Connector TEXT,
    UserId TEXT,
    UserInfo MEMO);

CREATE UNIQUE INDEX MessageLogNDX ON MessageLog (Id);

CREATE INDEX IDX_MessageId ON MessageLog (MessageId, SendTime);
```

Add the SQL Connector

Now you can add the new database to Diafaan SMS Server. First create a new SQL Connector with the connector wizard and select OLE DB database as the database type. On the next page you can provide the connection string, the connection string below is an example of a connection string for the OLE DB provider and can be adapted to suit your database configuration:

Connection string for Access 2010 and older:
Provider=Microsoft.ACE.OLEDB.12.0;Data Source=C:\smsserver.accdb;User Id=myUserId;Password=myPassword;

Alternative connection string for Access 2002 and Access 2003:
Provider=Microsoft.Jet.OLEDB.4.0;Data Source=C:\smsserver.mdb;User Id=myUserId;Password=myPassword;

Select the database fields

On the next wizard pages you can select the database tables and fields you want Diafaan SMS Server to use. You can only select the most used fields here. Additional database fields can be selected later on the advanced tab page in the connector properties dialog.

For more information see **Access your database**.

Important note

If you want to keep the messages in the MessageOut table you can select the “Change field: .. to ..” option. In Microsoft Access the YES value of a YESNO field is -1. Other values have unpredictable results and may cause repeated sending of the messages. For the database layout in this article this would mean Change field: IsSent to -1.

It is a good idea to change the database flavor option on the same advanced tab page to Microsoft. This will instruct Diafaan SMS Server to use a specific SQL syntax for Microsoft, enhancing performance somewhat.

3.9.5 PostgreSQL database

Learn how to send and receive SMS messages using a PostgreSQL database with Diafaan SMS Server. Get instructions about the installation of the ODBC driver, create the database with an SQL script and how to connect to the database with the connection string for PostgreSQL.

PostgreSQL

PostgreSQL is an open source database server. Diafaan SMS Server can use an ODBC driver to connect to PostgreSQL. For this article PostgreSQL version 8.4.2 was used with the psqlODBC 8.4.2 driver.

Install the psqlODBC driver

Before Diafaan SMS Server can send and receive SMS messages from the database you have to make sure the psqlODBC driver is installed on the computer running Diafaan SMS Server. You can find the psqlODBC driver on the PostgreSQL website.

Create the SMSServer database

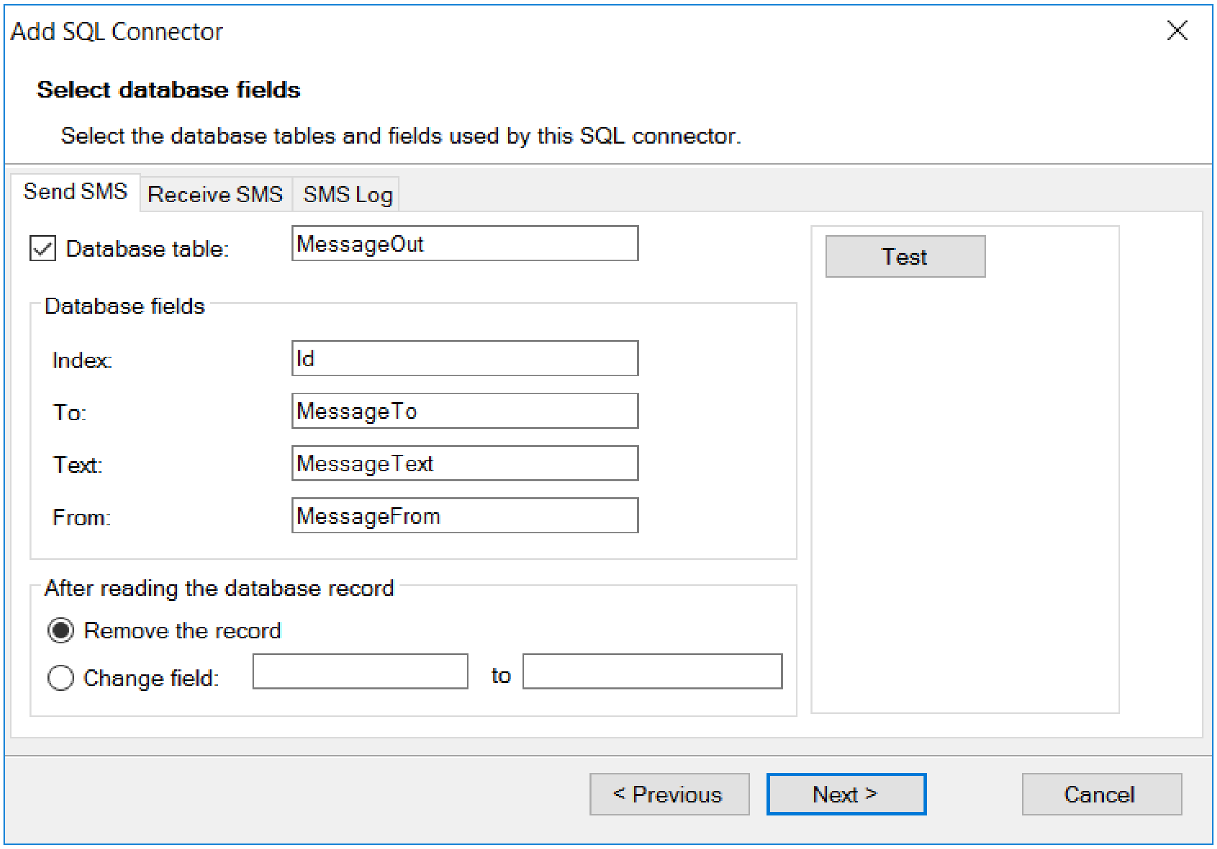
You can use the PostgreSQL SQL shell to create the database tables. Of course it is also possible to use an existing database, but if you want to be sure you can use all options available in Diafaan SMS Server you should use the SQL commands below necessary to create the database tables to send and receive SMS messages and to keep the send log.

```
CREATE DATABASE SMSServer WITH ENCODING 'UTF8';

CREATE SEQUENCE messageout_seq;

CREATE TABLE MessageOut (
    Id INTEGER NOT NULL PRIMARY KEY DEFAULT nextval('messageout_seq'),
    MessageTo VARCHAR(80),
    MessageFrom VARCHAR(80),
    MessageText TEXT,
    MessageType VARCHAR(80),
    MessageGuid TEXT,
    MessageInfo TEXT,
    Gateway VARCHAR(80),
    UserId VARCHAR(80),
    UserInfo TEXT,
    Priority INTEGER,
    Scheduled TIMESTAMP,
    ValidityPeriod INTEGER,
    TLVList TEXT,
    IsSent SMALLINT NOT NULL DEFAULT 0,
    IsRead SMALLINT NOT NULL DEFAULT 0);

CREATE INDEX IDX_IsRead ON MessageOut (IsRead);

CREATE SEQUENCE messagein_seq;

CREATE TABLE MessageIn (
    Id INTEGER NOT NULL PRIMARY KEY DEFAULT nextval('messagein_seq'),
    SendTime TIMESTAMP,
    ReceiveTime TIMESTAMP,
    MessageFrom VARCHAR(80),
    MessageTo VARCHAR(80),
    SMSC VARCHAR(80),
    MessageText TEXT,
    MessageType VARCHAR(80),
    MessageParts INTEGER,
    MessagePDU TEXT,
    Gateway VARCHAR(80),
    UserId VARCHAR(80));

CREATE SEQUENCE messagelog_seq;

CREATE TABLE MessageLog (
    Id INTEGER NOT NULL PRIMARY KEY DEFAULT nextval('messagelog_seq'),
    SendTime TIMESTAMP,
    ReceiveTime TIMESTAMP,
    StatusCode INTEGER,
    StatusText VARCHAR(80),
    MessageTo VARCHAR(80),
    MessageFrom VARCHAR(80),
    MessageText TEXT,
    MessageType VARCHAR(80),
    MessageId VARCHAR(80),
    MessageGuid TEXT,
    MessageInfo TEXT,
    ErrorCode VARCHAR(80),
    ErrorText TEXT,
    Gateway VARCHAR(80),
    MessageParts INTEGER,
    MessagePDU TEXT,
    Connector VARCHAR(80),
    UserId VARCHAR(80),
    UserInfo TEXT);

CREATE INDEX IDX_MessageId ON MessageLog (MessageId, SendTime);
```

Add the SQL Connector

Now you can add the new database to Diafaan SMS Server. First create a new SQL Connector with the connector wizard and select ODBC as the database type. On the next page you can provide the ODBC connection string, the following connection string should be adapted for your database installation:

```
Driver={PostgreSQL
UNICODE};Server=192.168.0.0;Port=5432;Database=smsserver;Uid=postgres;Pwd=password;
```



Add SQL Connector

Select database fields

Select the database tables and fields used by this SQL connector.

☑ Database table: MessageOut

Database fields

Index: Id

To: MessageTo

Text: MessageText

From: MessageFrom

After reading the database record

● Remove the record

○ Change field: to

Select the database fields

On the next wizard pages you can select the database tables and fields you want Diafaan SMS Server to use. You can only select the most used fields here. Additional database fields can be selected later on the advanced tab page in the connector properties dialog.

For more information see **Access your database**.

It is a good idea to change the database flavor option on the same advanced tab page to MySQL (PostgreSQL is not available but MySQL has the same syntax). This will instruct Diafaan SMS Server to use a specific SQL syntax for PostgreSQL enhancing performance somewhat.

3.10 Scripting examples

3.10.1 Gateway scripting examples

3.10.1.1 BulkSMS.com script

The BulkSMS.com gateway script provides a connection to the bulksms.com SMS gateway. The script offers support for sending text, unicode and flash SMS messages. Status update requests are also supported, making it possible to see when a message is received on the GSM phone.

You can find the BulkSMS.com.cs script in the Examples folder of Diafaan SMS Server.

References

The script makes use of the System.Web assembly so System.Web.dll has to be added to the ReferenceList property in the Advanced tab page.

Script

Before you can use the script you have to change the userName and password variables in the top of the script.

Sending the message

In the **OnMessageToSend** event method the query string is built with different parameters depending on the message type. The binary message type is not supported but it will not be very difficult to implement this as well. The query string is sent with a POST method to the url using an HttpWebRequest.

If the response indicates that the message is successfully sent, the script calls **PostSendResult** with a positive status code, the last parameter of this call is set **true**. This instructs Diafaan SMS Server to fire the **OnStatusRequest** event one or more times depending on the message status.
If the response does not indicate that the message is successfully sent, the script calls **PostSendResult** with a negative status code.

If you make any change to the script make sure that the **PostSendResult** is always called, even if an exception is caused. Otherwise the script will only continue with the next message after a timeout of several minutes.

Handling status updates

Status updates are handled in the **OnStatusRequest** event. The script sends an HTTP query to BulkSMS.com and calls **PostSendResultUpdate** if the response indicates that the message was received on the GSM phone. In this case **OnStatusRequest** is not called again.
If the response does not indicate that the message was received on the GSM phone the **OnStatusRequest** event will be called a couple of times more, each time with a larger time interval.

Permanent errors are not handled in the example script, but it is possible to use **PostSendResultUpdate** for permanent errors as well.

3.10.1.2 Dispatch SMS to gateways

Learn how to use a C# or Visual Basic .NET script with Diafaan SMS Server to send an SMS message through a specific gateway, depending on the GSM number.

Scenario

In this scenario three gateways are defined, **GSM Modem 1**, **GSM Modem 2** and **Clickatell**. SMS messages to British GSM numbers should be sent with GSM Modem 1 and messages to French GSM numbers to GSM Modem 2. All messages to other destinations should be rejected. For both destinations the Clickatell gateway is used as a backup gateway. The maximum send attempts on both GSM modems is 2, this is indicated in the script with "GSM Modem 1:2" and "GSM Modem 2:2".

Script

The only adaptation to the skeleton script is in the **OnMessageToSend** event:

C# script:

```
private void OnMessageToSend(string recordId, string toAddress, string fromAddress,
                             string message, string messageType)
{
  if (toAddress.StartsWith("+44")) {
    PostDispatchMessage(recordId, toAddress, fromAddress, message, messageType,
                        "GSM Modem 1:2", "Clickatell");
  }
  else if (toAddress.StartsWith("+33")) {
    PostDispatchMessage(recordId, toAddress, fromAddress, message, messageType,
                        "GSM Modem 2:2", "Clickatell");
  }
  else {
    PostSendResult(recordId, "", StatusCode.SendError,
                   "Error: Invalid country destination", "", "", false);
  }
}
```

Visual Basic .NET script:

```
Private Sub OnMessageToSend(recordId As String, toAddress As String, _
                            fromAddress As String, message As String, _
                            messageType As String)
  If toAddress.StartsWith("+44") Then
    PostDispatchMessage(recordId, toAddress, fromAddress, message, messageType, _
                        "GSM Modem 1:2", "Clickatell")
  ElseIf toAddress.StartsWith("+33") Then
    PostDispatchMessage(recordId, toAddress, fromAddress, message, messageType, _
                        "GSM Modem 2:2", "Clickatell")
  Else
    PostSendResult(recordId, "", StatusCode.SendError, _
                   "Error: Invalid country destination", "", "", False)
  End If
End Sub
```

If you make any change to the script make sure that the **PostSendResult** or **PostDispatchMessage** is always called, even if an exception is caused. Otherwise the script will only continue with the next message after a timeout of several minutes.

MaximumBatchSize

To enhance performance when a large amount of messages is sent, you can set the **MaximumBatchSize** in the Advanced properties to a higher value.

3.10.1.3 Send messages to e-mail

Learn how to use a C# or Visual Basic .NET script with Diafaan SMS Server to send an email message. This simple script uses the System.Net.Mail namespace to send messages to the email address in the toAddress field of the message.

Script

The only adaptation to the skeleton script is in the headers and the **OnMessageToSend** event:

C# script:

```
using System.Net.Mail;

private void OnMessageToSend(string recordId, string toAddress, string fromAddress,
                             string message, string messageType)
{
   string smtpHostName = "smtp.host.com";
   string defaultFromAddress = "name@host.com";
   string defaultFromName = "Sender Name";
   string[] messageLines;
   SmtpClient smtpClient;
   MailAddress from;
   MailAddress to;
   MailMessage emailMessage;

   try {
      smtpClient = new SmtpClient(smtpHostName);
      if (fromAddress == "") {
        from = new MailAddress(defaultFromAddress, defaultFromName,
                System.Text.Encoding.UTF8);
      }
      else {
        from = new MailAddress(fromAddress, fromAddress,
                System.Text.Encoding.UTF8);
      }
      to = new MailAddress(toAddress);
      emailMessage = new MailMessage(from, to);
      messageLines = message.Split(new char[] {'\n','\r'},
                StringSplitOptions.RemoveEmptyEntries);
      if (messageLines.Length > 0)   emailMessage.Subject = messageLines[0];
      emailMessage.SubjectEncoding = System.Text.Encoding.UTF8;
      emailMessage.Body = message;
      emailMessage.BodyEncoding =  System.Text.Encoding.UTF8;
      smtpClient.Send(emailMessage);
      PostSendResult(recordId, "", StatusCode.Sent,
                     "Success: Message accepted by the SMTP server",
                     "", "", false);
   }
   catch (Exception e) {
     PostSendResult(recordId, "", StatusCode.SendError, "Error: " + e.Message,
                "", "", false);
   }
}
```

Visual Basic .NET script:

```
Imports Microsoft.VisualBasic
Imports System.Net.Mail

Private Sub OnMessageToSend(recordId As String, toAddress As String, _
                            fromAddress As String, _
                            message As String, messageType As String)
   Dim smtpHostName As String = "smtp.host.com"
   Dim defaultFromAddress As String = "name@host.com"
   Dim defaultFromName As String = "Sender Name"
   Dim messageLines As String()
   Dim smtpClient As SmtpClient
   Dim from As MailAddress
   Dim [to] As MailAddress
   Dim emailMessage As MailMessage

   Try
      smtpClient = New SmtpClient(smtpHostName)
      If fromAddress = "" Then
         from = New MailAddress(defaultFromAddress, defaultFromName, _
                System.Text.Encoding.UTF8)
      Else
         from = New MailAddress(fromAddress, fromAddress, System.Text.Encoding.UTF8)
      End If
      [to] = New MailAddress(toAddress)
      emailMessage = New MailMessage(from, [to])
      messageLines = message.Split(New Char() {ControlChars.Lf, ControlChars.Cr}, _
         StringSplitOptions.RemoveEmptyEntries)
      If messageLines.Length > 0 Then
         emailMessage.Subject = messageLines(0)
      End If
      emailMessage.SubjectEncoding = System.Text.Encoding.UTF8
      emailMessage.Body = message
      emailMessage.BodyEncoding = System.Text.Encoding.UTF8
      smtpClient.Send(emailMessage)
      PostSendResult(recordId, "", StatusCode.Sent, _
                "Success: Message accepted by the SMTP server", _
                "", "", False)
   Catch e As Exception
      PostSendResult(recordId, "", StatusCode.SendError, "Error: " & e.Message, _
                "", "", False)
   End Try
End Sub
```

If you make any change to the script make sure that the **PostSendResult** is always called, even if an exception is caused. Otherwise the script will only continue with the next message after a timeout of several minutes.

3.10.2 Connector scripting examples

3.10.2.1 Auto reply SMS messages

Learn how to use a C# or Visual Basic .NET script with Diafaan SMS Server to send an automatic reply to a received SMS message. This simple script sends a reply message every time an SMS message is received.

Script

The only adaptation to the skeleton script is in the **OnMessageReceived** event:

C# script:

```
private void OnMessageReceived(string fromAddress, string toAddress,
                               string message, string messageType,
                               string messageId, string smsc, string pdu,
                               string gateway, DateTime sendTime,
                               DateTime receiveTime)
{
  PostSendMessage(fromAddress, "", "We have received your message.",
                  "sms.text", "", "", "");
}
```

Visual Basic .NET script:

```
Private Sub OnMessageReceived(fromAddress As String, toAddress As String, _
                              message As String, messageType As String, _
                              messageId As String, smsc As String, _
                              pdu As String, gateway As String, _
                              sendTime As DateTime, receiveTime As DateTime)
  PostSendMessage(fromAddress, "", "We have received your message.", _
                  "sms.text", "", "", "")
End Sub
```

If you want to be sure the message is sent with the same (GSM Modem) gateway that received the message, add the gateway name to the **PostSendMessage** method:

C# script:

```
private void OnMessageReceived(string fromAddress, string toAddress,
                               string message, string messageType,
                               string messageId, string smsc, string pdu,
                               string gateway, DateTime sendTime,
                               DateTime receiveTime)
{
  PostSendMessage(fromAddress, "", "We have received your message.",
                  "sms.text", gateway, "", "");
}
```

Visual Basic .NET script:

```
Private Sub OnMessageReceived(fromAddress As String, toAddress As String, _
                              message As String, messageType As String, _
                              messageId As String, smsc As String, _
                              pdu As String, gateway As String, _
                              sendTime As DateTime, receiveTime As DateTime)
  PostSendMessage(fromAddress, "", "We have received your message.", _
                  "sms.text", gateway, "", "")
End Sub
```

Reference

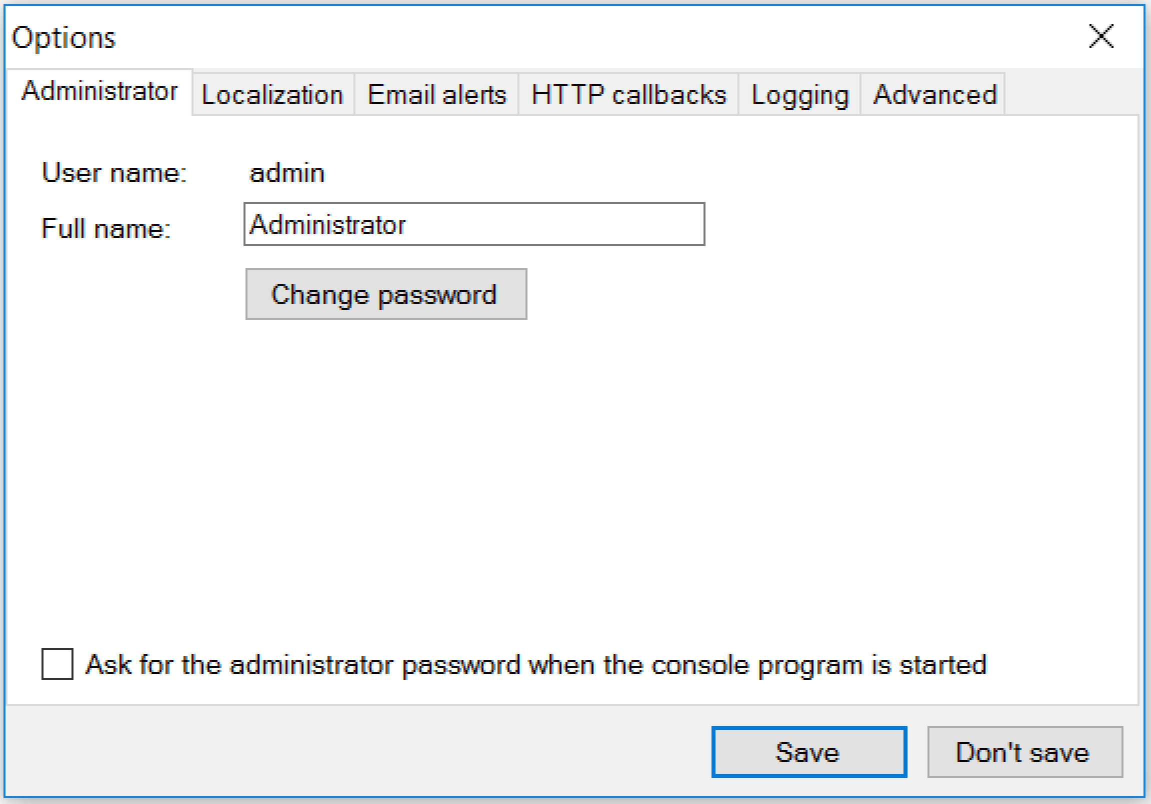
IV

4 Reference

4.1 General options

Administrator

Set the full name and/or password for the administrator (**admin**) account. The **admin** account is used in the web application and the HTTP API of the Web Connector.



User name: admin

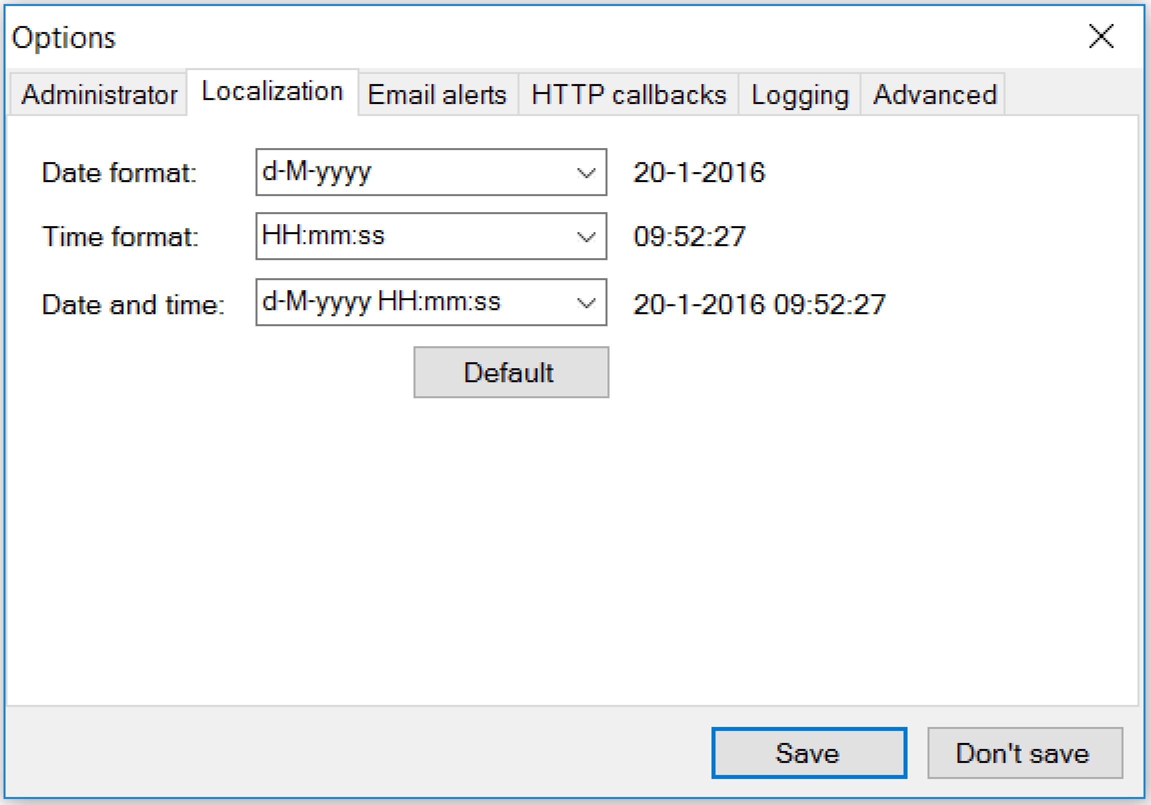
Full name: Administrator

☐ Ask for the administrator password when the console program is started

If you want to prevent unauthorized access to the Diafaan SMS Server console program you can check the option to ask for the administrator password when a the console program is started.

Localization

In the localization settings the date and time format that is used in Diafaan SMS Server can be changed. The **Date and time** format is used in all situations where the date and time is used in combination. The **Date format** and **Time format** values are used for date only and time only fields.

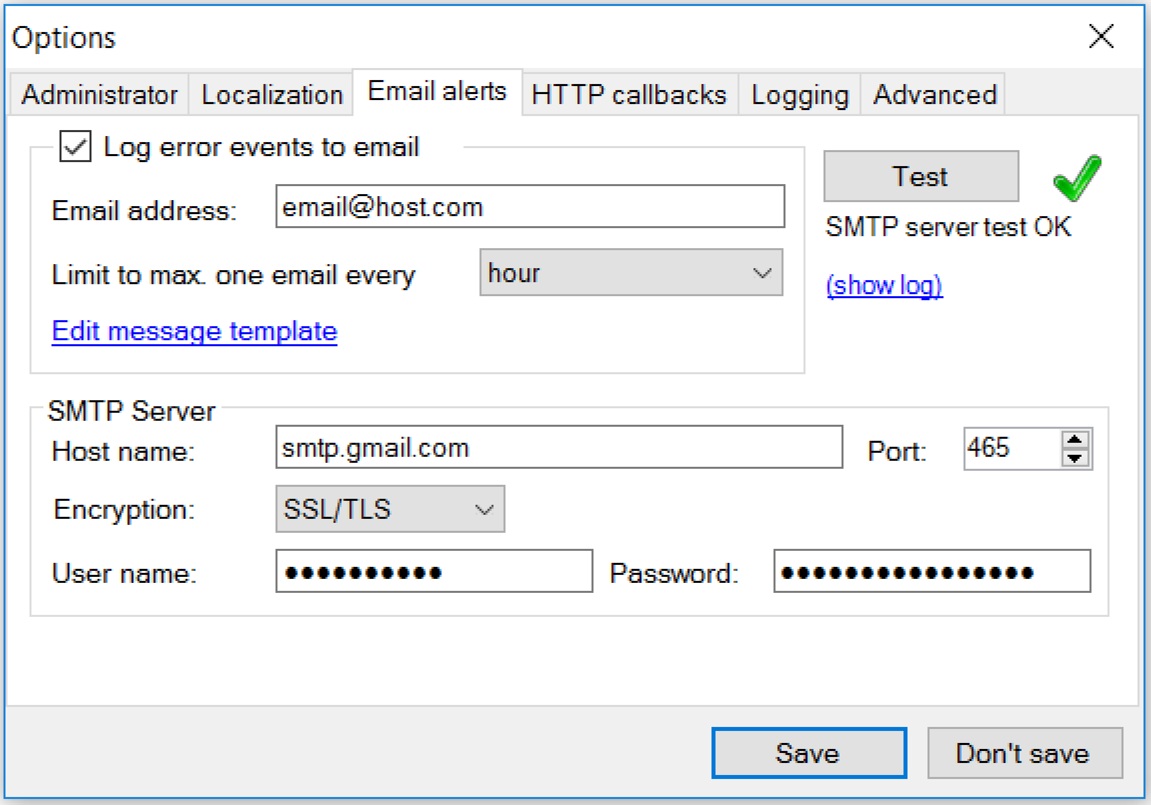


Date format: d-M-yyyy 20-1-2016

Time format: HH:mm:ss 09:52:27

Date and time: d-M-yyyy HH:mm:ss 20-1-2016 09:52:27

Email alerts



☑ Log error events to email

Email address: email@host.com

Limit to max. one email every hour

✓ SMTP server test OK

SMTP Server

Host name: smtp.gmail.com

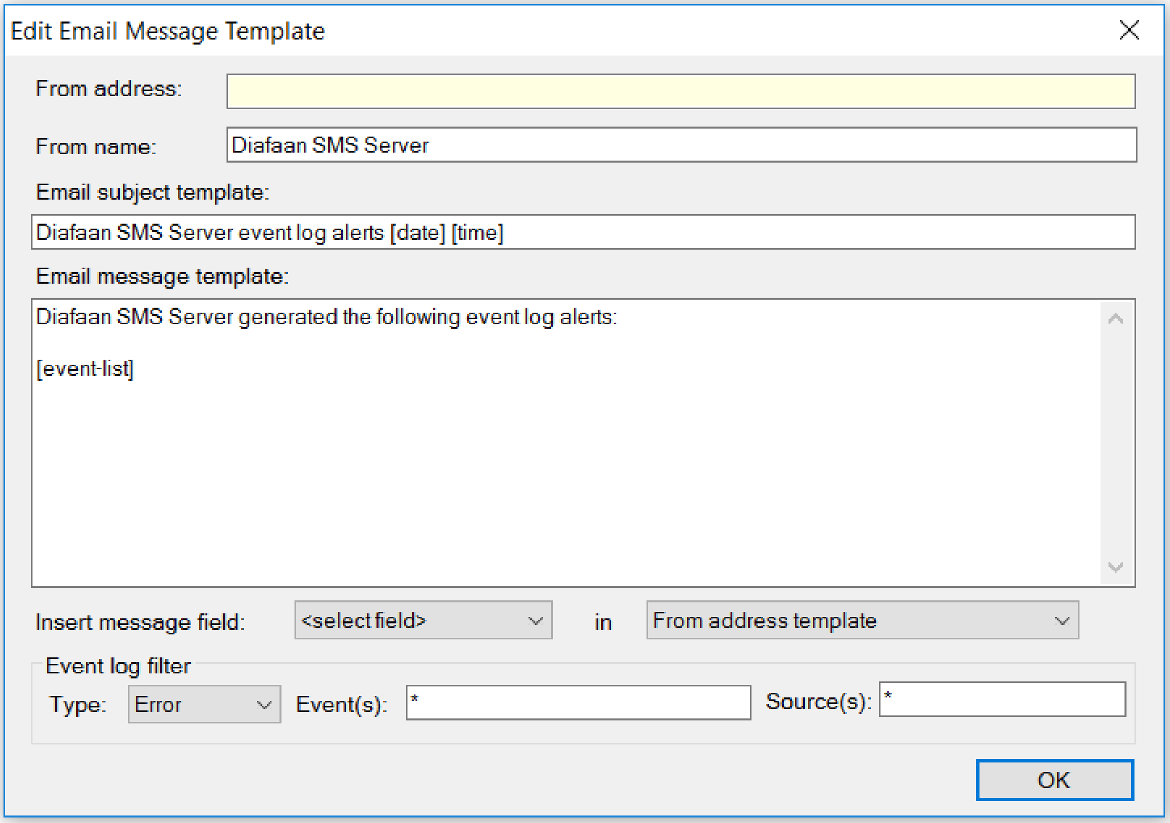
Port: 465

Encryption: SSL/TLS

User name: ●●●●●●●●●●● Password: ●●●●●●●●●●●●●●●●●

Log error events to email

Check this option if you want to receive an email alert if an error is added to the event log. To prevent an avalanche of email messages after repeated errors are generated, the number of emails can be limited to one email in a predefined time span.



From address:

From name: Diafaan SMS Server

Email subject template:

Diafaan SMS Server event log alerts [date] [time]

Email message template:

Diafaan SMS Server generated the following event log alerts:

[event-list]

Insert message field: <select field> in From address template

Event log filter

Type: Error Event(s): * Source(s): *

Event log filter

In the Email Message Template the email fields for the alert email can be changed and a filter can be set to limit the events for which an email alert is sent. The Event(s) field filters on the event text and the Source(s) field filters on the name of the gateway or connector. Both fields can contain multiple filters with wildcards (separated by semicolons) and it is possible to use a regular expression when the field starts with the keyword **regex:** (when a match is found) or **-regex:** (when no match is found).

Examples for the Event(s) field

Allow events that contain the text "Timeout" (wildcard filters are case insensitive):
timeout

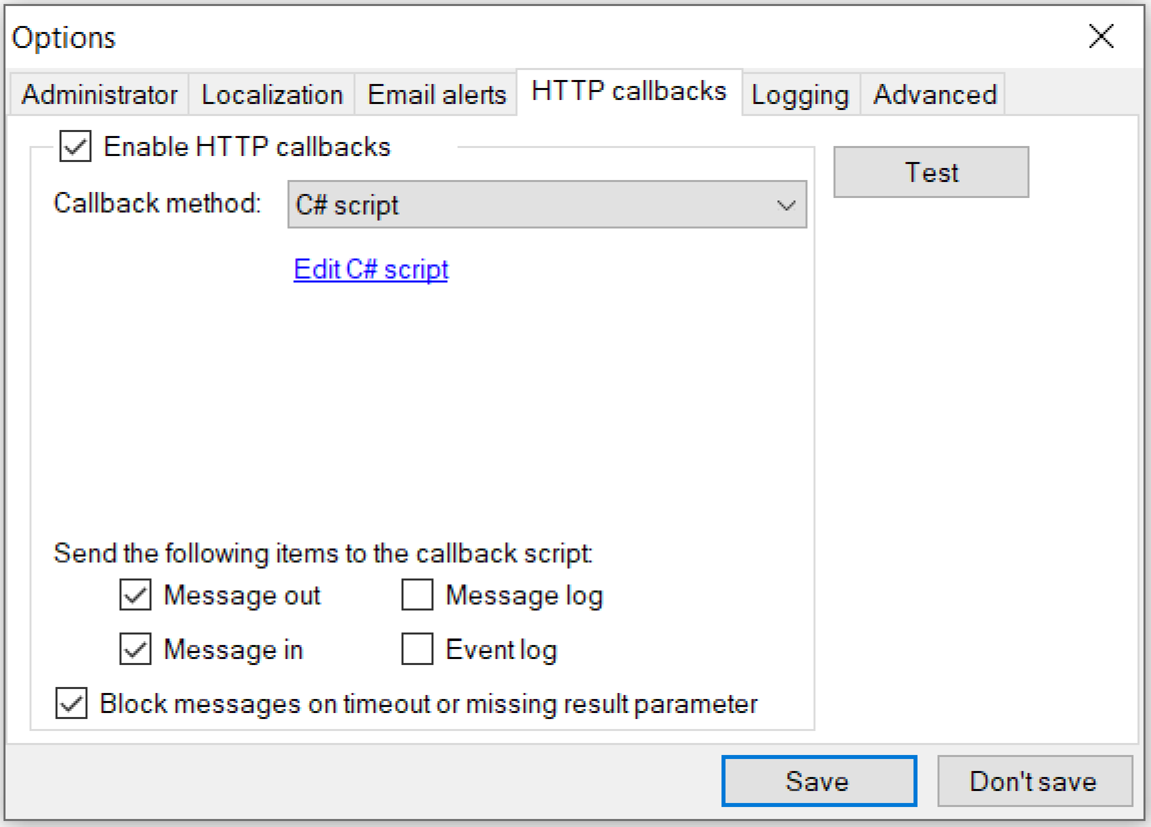
Allow events that contain the text "Timeout" or "Reset":
timeout*;*reset

Allow events that do not contain the text "Timeout" or "Reset":
-**timeout*;*reset**

Allow events that start with the text "Reset" using a regular expression (regex filters are case sensitive):
*regex:^Reset.**

HTTP callbacks

Only available in Diafaan SMS Server - basic and full editions.
See Use HTTP callbacks for the HTTP specifications



☑ Enable HTTP callbacks

Callback method: C# script

Edit C# script

Send the following items to the callback script:

☑ Message out ☐ Message log

☑ Message in ☐ Event log

☑ Block messages on timeout or missing result parameter

Enable HTTP callbacks

Check this option to enable HTTP callbacks in Diafaan SMS Server. This setting enables all HTTP callbacks for individual messages and events and also enables user authentication callbacks when the connector supports it.

Callback method (C# script)

The HTTP callbacks are handled by a C# script. Because the script runs internally in Diafaan SMS Server there is no need to start an external process and as a result it is the fastest HTTP callback option. However, if the script makes a call to an external process (like a database) it can still slow down the message throughput.

Edit C# script

Click to edit the C# script that handles the callback calls. The default script returns a 'result=1' response for every HTTP callback.

Callback method (using HTTP server)

Use the full HTTP client only when the HTTP callbacks don't work with the minimal HTTP client. The minimal HTTP client is much faster but lacks advanced features like automatic proxy server detection and may not be compatible with all web servers.

Web page

Set the full URI of the web page that handles the HTTP callback request.

Callback method (using local CGI application)

Instead of sending an HTTP call to an HTTP server, Diafaan SMS Server can also use a local installation of a CGI application, for instance the PHP program **php-cgi.exe**. Instead of sending the callback request to a web server, this option lets you process the callback on the local server. This option was tested with the CGI applications **PHP** (php-cgi.exe), **Perl** (perl.exe), **Ruby** (ruby.exe) and **Python** (python.exe). **php-cgi.exe** is part of the PHP installation package for Windows.

CGI application
Select the .exe CGI application on your local server.

Script file
Provide the full path and file name of the local script that handles the HTTP callback request.

Callback method (using local php-cgi.exe FastCGI application)

PHP and some other CGI applications can also be used in a FastCGI mode where TCP/IP is used for the CGI communication instead of the relatively slow process communication of normal CGI. For reliable PHP scripts, FastCGI is always better than normal CGI. But if the PHP script has memory leaks or might otherwise be occasionally unresponsive, it is better to use normal CGI.

CGI application
Select the .exe CGI application on your local server.

Script file
Provide the full path and file name of the local script that handles the HTTP callback request.

FastCGI port
The TCP/IP port number that the FastCGI program uses for the communication with Diafaan SMS Server.

HTTP method

Select the POST or GET HTTP method depending on the expectations in the script.

Timeout

Timeout (in seconds) for the HTTP calls, increase this value if the HTTP server does not always handle the HTTP request in time.

Message out

Use an HTTP callback for each outbound message before it is added to the message queue.

Message in

Use an HTTP callback for each inbound message before it is added to the message queue.

Message log

Use an HTTP callback for each message result and message result update or delivery report.

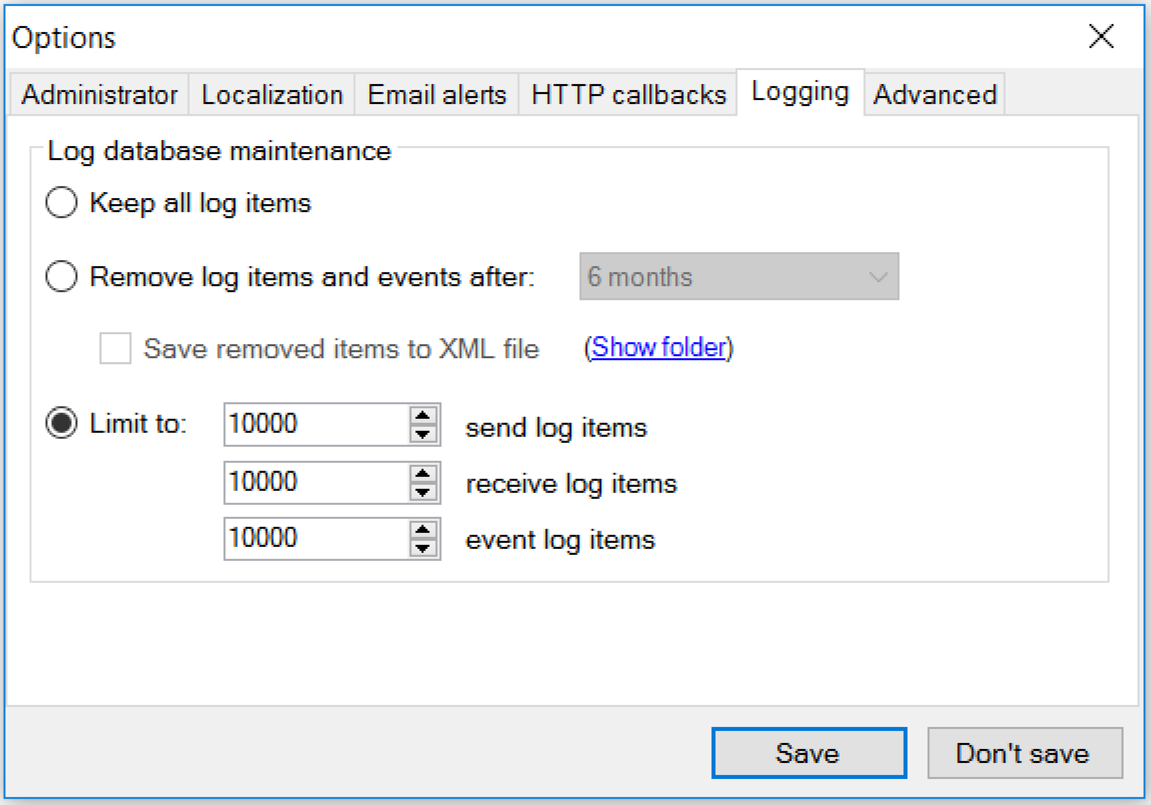
Event log items

Use an HTTP callback for all event log items.

Block messages on timeout or missing result parameter

Reject (block) outbound and inbound message when the HTTP callback call fails or when the **result** parameter is missing in the HTTP response.

Logging



Log database maintenance

○ Keep all log items

○ Remove log items and events after: 6 months

□ Save removed items to XML file (Show folder)

● Limit to: 10000 send log items

10000 receive log items

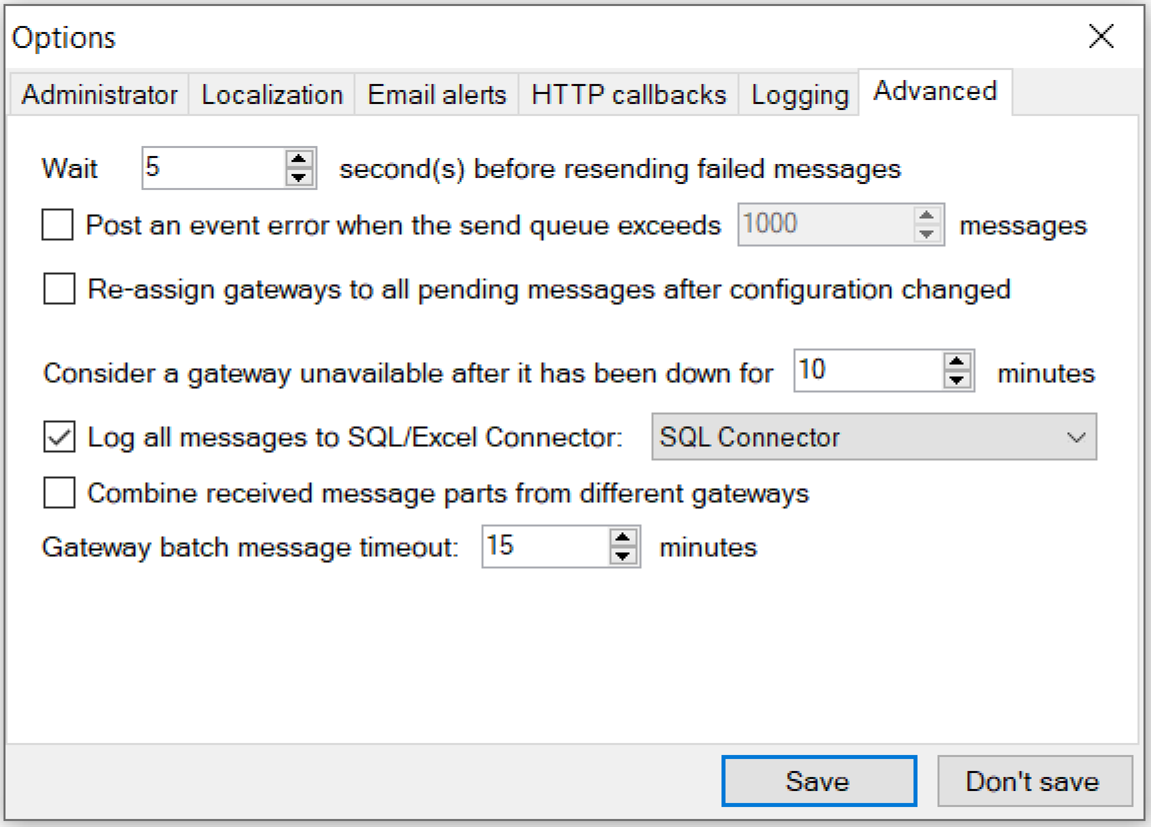
10000 event log items

Log database maintenance

If the **Limit to** option is selected the number of log items in the Send Log, Receive Log and Event Log is limited to a fixed number of records. If the internal log database, and specifically the send log table, is getting too large, updating and viewing the logs will get gradually slower and a fixed log limit helps to prevent this.

An alternative to a fixed log length is to periodically remove older items from the database. If the **remove log items** option is checked, the oldest items are cleared from the database once a day. If you send tens of thousands of messages a day and are not very interested in the log, the time period can be set on 7 days or less. If you do not send large amounts of messages a day, the time period can be set to 6 months or one year.
You can make a backup copy of the removed records by checking the **save removed items** option.

Advanced



Wait 5 second(s) before resending failed messages

□ Post an event error when the send queue exceeds 1000 messages

□ Re-assign gateways to all pending messages after configuration changed

Consider a gateway unavailable after it has been down for 10 minutes

☑ Log all messages to SQL/Excel Connector: SQL Connector

□ Combine received message parts from different gateways

Gateway batch message timeout: 15 minutes

Resend wait time
If the message failed, wait x seconds before it is send again.

Post an error
Trigger an error in the event log when the message queue exceeds a specific number of messages.

Re-assign gateways
Check this option if you want to re-assign the (possible) gateways to all messages in the send queue after each configuration change. If the option is not enabled then the new gateway configuration will only be applied to new messages.

If there are a large number of messages in the send queue it may take a long time to assign the new gateway list to each message after each time the gateway- or connector properties are changed. It is advised to use this option only for situations where the number of messages in the send queue is limited.

Gateway unavailability timeout
Set the timeout, in minutes, when a gateway that is (temporarily) down is considered to be permanently unavailable. When the gateway is unavailable, messages that are routed through this gateway are flagged as failed. When the gateway is down but not yet unavailable, the pending messages are kept in the send queue until the gateway is working again.

Log to SQL or Excel connector
Send the log records and updates (delivery confirmation) of all messages in Diafaan SMS Server to a specific SQL or Excel connector.

Combine received message parts from different gateways

When there are multiple SMPP Gateways connected to the same SMPP server and the same account, there is a chance that multiple received messages parts for the same message are received with different SMPP Gateways. By default the message parts are only combined when they are received in the same SMPP Gateway. If this option is enabled, message parts from different SMPP Gateways are also combined to the original message length.

To improve the handling speed of received multiple message parts from different SMPP Gateways you can set the **MessageInHandling** property in the intended SMPP Gateways to **SingleMessageParts**.

Gateway batch message timeout

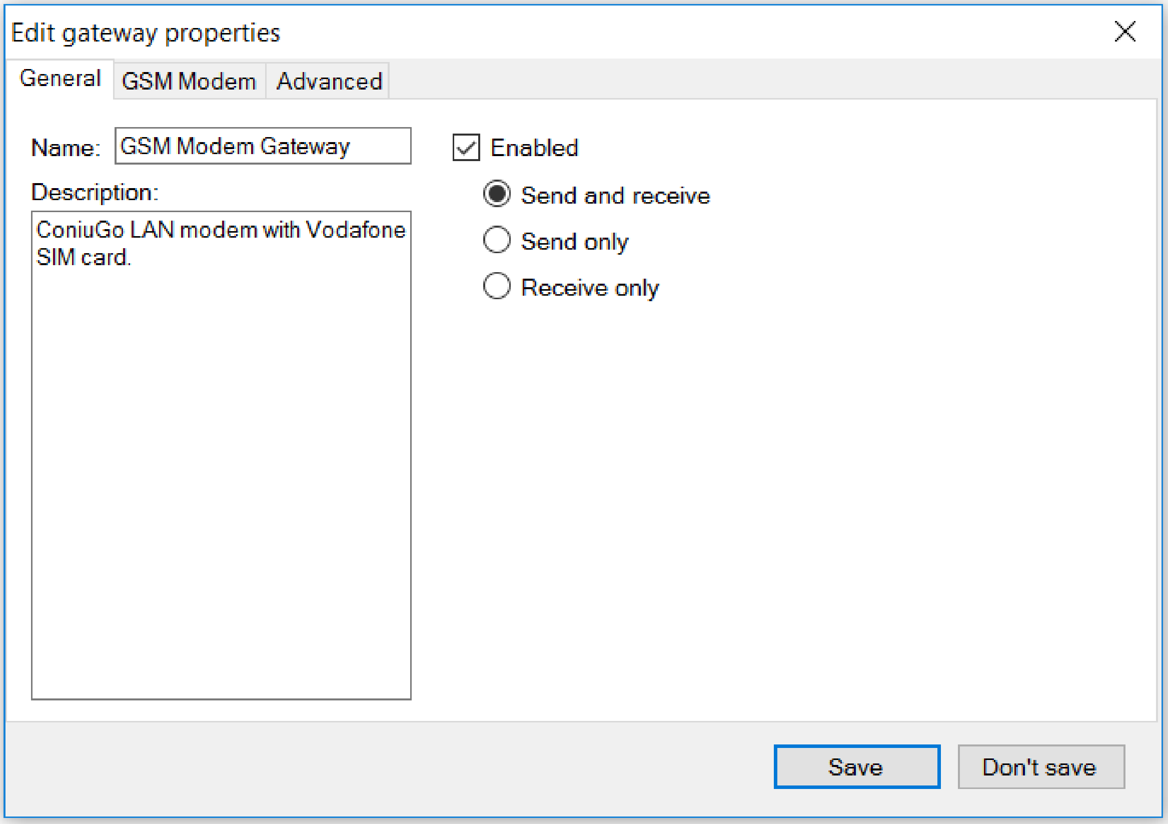
Set the timeout, in minutes, to re-send a message when it is pre-loaded in one of the gateways. This is a security measure for when a message is 'stuck' for too long in a gateway. The default value is **15 minutes** When this message timeout is set too low it might result in duplicated messages, if it is set too high it might lead to a long send delay when one of the gateways malfunctions.

4.2 Gateway properties

General properties

The gateway name is used to identify the gateway and to route messages through a specific gateway.

If you want to temporarily disable the gateway or set the gateway to send or receive only you can set the appropriate options here.



Name: GSM Modem Gateway

Description: ConiuGo LAN modem with Vodafone SIM card.

☑ Enabled

● Send and receive

○ Send only

○ Receive only

Gateway specific properties

- GSM Modem gateway
- Clickatell gateway
- Emulator gateway
- SMPP gateway (basic and full editions only)
- HTTP gateway (basic and full editions only)
- Paging gateway (basic and full editions only)
- Scripting gateway (basic and full editions only)

4.2.1 GSM Modem gateway

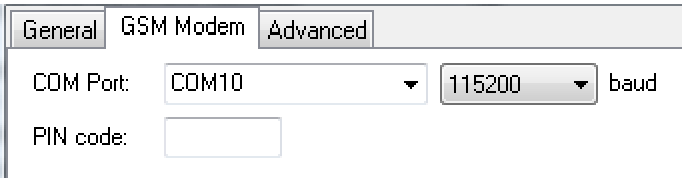
GSM Modem properties

COM Port
Select the name of the serial port where the GSM modem is connected. If the GSM modem is connected to a USB port you must select the name of the virtual COM port of the USB port driver.

Baudrate
Select the baudrate (speed) for the serial port. Often the baudrate is detected automatically by the GSM modem but some GSM modems require a fixed baudrate.

TCP/IP connected modems
The GSM Modem Gateway can directly use GSM modems connected over an Internet TCP/IP connection, without the need for additional drivers. To use a TCP/IP connected GSM modem select **TCP/IP** from the COM port list and enter the host name or IP number and the attached port number. The GSM Modem Gateway can use a raw TCP/IP connection or a Telnet connection to the modem.

PIN code
Type the PIN code of the SIM card. Using a PIN code on your SIM card is not recommended. If your SIM card does not have a PIN code you can leave this field empty.



COM Port: COM10 115200 baud

PIN code:

Advanced properties

LogCommunicationToFile
Log all modem communication to file. Use this option only if you encounter problems with the GSM modem. If enabled, the status windows of the GSM modem provides a link to the location on you hard disk where the file is stored.

MaximumBatchSize
The maximum number message this gateway will process in one batch. If this value is higher than one and a number of messages are sent at the same time, the gateway loads multiple messages concurrently. This can increase the speed in which the messages are processed but it also might prevent normal load balancing for multiple GSM modems. The default value of MaximumBatchSize is **Automatic**, this sets the batch size to to an optimal value of 1 to 10, depending on the other settings in the GSM Modem Gateway.

NumberTemplate
The number template used to translate the 'To' number format for outbound message.

For more information see Use number templates.

NumberTemplateReceive
The number template used to translate the 'From' number format for inbound message.

For more information see Use number templates.

NumberWhiteList

List of allowed mobile numbers for received SMS messages, messages from other numbers are blocked. When the list does not have any items, received SMS messages from all numbers are allowed.

For more information about the white list and black list format see: Use white lists and black lists.

QuietHours
A list of 24 hour numbers (separated with semicolons) when sending messages with this gateway is paused. The hour number can be appended with **.1**, **.2**, **.3** or **.4** to specify the first, second, third or fourth quarter of the hour. If the messages cannot be sent with another gateway, they stay in the send queue until the quiet hours are over and this gateway resumes message sending.
00;01;23 pauses message sending from 23:00:00 to 01:59:59.
12.1;12.2;18.1;18.2 pauses message sending from 12:00:00 to 12:29:59 and from 18:00:00 to 18:29:59.

DisableOnFailedMessages
The number of consecutive failed messages after which the gateway will be disabled, set to 0 to disable this option.

DisableOnModemErrors
Set an error number and error number count after which the gateway will be disabled. This option can be used when a specific modem error is an indication of a permanent error. Use format e:c where e is the error number and c the number of errors that causes the gateway to be disabled, leave empty to disable this option.

EnableHTTPCallbacks
Enable or disable HTTP callbacks for this gateway. This option is only available when the HTTP callback option is enabled in the general options.

CompatibilityMode
Besides **Normal Mode**, two other compatibility mode options are available when you encounter problems with the operation of your GSM modem, Basic PDU Mode and Basic Text Mode.
Basic PDU Mode is a minimal implementation of the GSM 07.05 PDU standard. Only the necessary modem commands are used and options like delivery confirmation or signal quality are not supported. In this mode you will have to set the value for the SMSC number manually since the software will not query the GSM modem for the right SMSC number.
Basic Text Mode is a minimal implementation of the GSM 07.05 text mode standard. Support for unicode characters or binary messages is not available. Use this mode only if your GSM modem does not support PDU mode, some CDMA modems for instance can send and receive SMS messages in text mode only.

ModemInitializationBeginning
Additional modem initialization. This modem command is sent to the GSM modem before the rest of the initialization. A possible use for this property is the modem command to select the GSM operator you want to use. E.g. if the network operator code is 26202 (Vodafone Germany) the modem command is:
AT+COPS=1,2,26202

ModemInitializationCommand
The modem initialization command command, the default value is **ATE0**.

ModemInitializationEnd
Additional modem initialization. This modem command is sent to the GSM modem after the rest of the initialization.

ModemLogon
Logon parameters for TCP/IP modems that require a clear text logon procedure before it can be used to send and receive SMS messages. The logon parameter consist of multiple pairs of a trigger and

response text separated by semicolons, the trigger and response parameter is separated by a "@" character. The GSM modem gateway waits for the trigger and sends the response when the trigger is received. If all triggers are handled, the modem is expected to be available to send and receive SMS messages. An example ModemLogon for the PORTech MV-370 modem is:

username:@voip;password:@1234;module1@module1;got!!

This means:
Wait for "username:" and send "voip" to the modem.
Wait for "password:" and send "1234" to the modem.
Wait for "*module1*" and send "module1" to the modem.
Wait for "got!!", the modem is now available to send and receive SMS messages.

ModemRebootCommand
The modem reboot command, default value is the **AT+CFUN=1,1** command for a full reboot. Not all modems support a full reboot of the modem.

ModemRebootTime
After a modem reboot command the GSM modem needs some time for the reboot to complete and to reconnect to the mobile network. The ModemRebootTime specifies the number of seconds before the GSM Modem Gateway sends the next command.

ModemResetCommand
The modem reset command, the default value is **AT&F**.

PermanentErrorList
A list of GSM modem error codes (separated by semicolons) that are regarded as permanent errors. No more send attempts will be made for the message.

PermitModemCommandMessages
Permit messages with message type "modem.command", the text of the message is executed as an AT command on the modem.

RebootAfterNetworkFailure
If the GSM modem gateway is active, every 10 minutes the gateway queries the signal strength and network status of the GSM modem. If RebootAfterNetworkFailure is enabled the gateway sends a hard reset to the GSM modem when no GSM network is detected (rssi 99) or when it is not logged on to the GSM network.

RebootAfterNetworkRoaming
Reboot GSM modem when the modem reports it is not logged on to the home network but is roaming on another network.

RebootAfterReceiveError
Send the reboot command to the GSM modem if an error is reported after a receive SMS or delete received SMS command. This is often an indication of a network or GSM modem problem and can sometimes be resolved by a modem reboot.

RebootAfterSendError
Send the reboot command to the GSM modem if an error is reported after a send SMS command. This is often an indication of a network or GSM modem problem and can sometimes be resolved by a modem reboot.

RebootAfterTimeout

Send the reboot command to the GSM modem if no valid response to a modem command was received after a certain timeout. If supported by the GSM modem it will restart the connection to the GSM network.

RebootBeforeInitialization
Send the reboot command to the GSM modem every time the modem is initialized. This setting is useful if the modem sometimes does not recover from an internal error state.

RebootDaily
Send the reboot command to the GSM modem once every day. Use this only if the GSM modem starts getting problems when it runs for long periods.

ReceivePollAfterMaximumSendBatch
The maximum number of SMS messages that are sent immediately after each other before the GSM modem is checked for new received messages.

ReceivePollIntervalTime
The time (in seconds) between each check for new received messages when the GSM modem is not sending a message.

ReportCallerId
Get the caller identification of normal (voice) phone calls to the modem and forward them as received SMS messages with message type **gsm.cli**.

ReportCallerIdHangup
Send a hangup command (ATH0) to the modem when an incoming phone call is detected.

ReportCallerIdHangupCommand
The modem hangup command, default is **ATH0**.

SendDelayTime
Time (in seconds) between sending two consecutive SMS messages. This option can be used to delay sending of the messages if the mobile network is not able to process the messages fast enough.

SendLimit
The maximum number of message parts the GSM Modem Gateway sends per whole hour, day, week or month. 100H means 100 message parts per hour, 300D 300 message parts per day, 1000W 1000 message parts per week and 1000M 1000 message parts per month. When the maximum number of messages is reached, the GSM Modem Gateway will pause message sending until the start of the next time period. The gateway is not considered down during this period.
By default all send attempts are subtracted from the send limit, also when the GSM modem returns an error. You can add a list of error codes to the **SendLimitErrorCodeExceptionList** property for errors that should not be counted in the send limit.

SendLimitErrorCodeExceptionList
A list of GSM modem error codes that will not affect the send limit (Include 'timeout' in the list for timeout errors. Include 'all' for all errors, including timeout errors). Multiple error codes can be separated with semicolons.

SendTimeout
Response timeout value (in seconds) after sending a message (PDU) to the GSM modem.

SMSOverGPRS
Some GSM operators support sending SMS messages over a GPRS connection. If enabled this will speed up the sending messages considerably. Most GSM operators do not support SMS over GPRS or

have only limited support. Sending messages will fail if this option is enabled and the GSM operator does not support it.

DataBits
Serial port data bits. default is 8.

FlowControl
Serial port flow control, default is **None**.

Parity
Serial port parity. default is no parity.

StopBits
Serial port stop bits. default is 1.

GSMNumber
The mobile phone number of the SIM card (optional). This number is only used to identify which GSM modem received an SMS message. If this value is set, it overrides the GSM number found on the SIM card.

SMSCNumber
The SMS Central number is the address of the SMS Central of your GSM operator. This number must be in international format, e.g. +491722270333 for the German Vodafone SMSC and is necessary if you want to send SMS messages. If this value is set, it overrides the SMSC number found on the SIM card.

CombineMultiPartMessage
If enabled, partial messages received on the GSM modem are combined to the original message length. Since delivery of some message parts might be delayed this could mean that the delivery of the whole message will be delayed. If, after a certain timeout, not all message parts are received, the partial messages are delivered individually. If disabled the partial messages are delivered individually. CombineMultipartMessages can also be set to **Binary**. In this mode the received messages are not decoded and combined but forwarded as message type **sms.binary** with the message field set to **<DCS>,<PID>,<UDL>,<UDH>,<Data>**. This may be useful if you intend to use Diafaan SMS Server to retransmit the messages unaltered to one or more alternative receivers.

DefaultAddressType
The address type used for mobile numbers that do not start with a **+** (international), **#** (national) or ***** (unknown). Most mobile operators prefer numbers in the International number format address type.

DeliveryConfirmation
Require status updates for the SMS messages after it is sent to the SMSC. If you do not need to know if the SMS messages are delivered on the GSM phone or if the status updates causes errors sending the messages you can disable status updates.

MaxMessageParts
Messages to send that are longer than 160 characters or 70 characters for unicode messages can be split in multiple SMS messages that are recombined to one message on the handset of the receiver. The MaxMessagePart property limits the amount of partial messages that the gateway sends with a long message. If the message is longer than can be sent with the maximum message parts, the rest of the message is discarded.

MaxShortCodeLength
The maximum number length where the number is regarded as a short code, short codes are encoded as Network Specific numbers.

MessageType
The message type is the default value for the SMS message encoding, the default value may be replaced with the message type specified by the sending Connector.
The gateway uses four possible value for the default message type:

Automatic	Standard Text encoding is the default, Unicode encoding will be used only if the message contains characters that are not in the default GSM SMS character set.
StandardText	The default 7-bit GSM SMS character set is used, all other characters are converted to question marks.
Unicode	All messages are sent in Unicode 16-bit encoding.
Data	All messages are sent in binary 8-bit encoding.

The GSM Modem gateway supports the following custom message types:

sms.automatic	Standard Text encoding is the default, Unicode encoding will be used only if the message contains characters that are not in the default GSM SMS character set.
sms.text	The default 7-bit GSM SMS character set is used, all other characters are converted to question marks.
sms.unicode	The message is sent in Unicode 16-bit encoding.
sms.binary	The message is sent in binary 8-bit encoding.
sms.automatic.flash	Flash SMS, standard Text encoding is the default, Unicode encoding will be used only if the message contains characters that are not in the default GSM SMS character set.
sms.text.flash	Flash SMS, the default 7-bit GSM SMS character set is used, all other characters are converted to question marks.
sms.unicode.flash	Flash SMS, all messages are sent in Unicode 16-bit encoding.
gsm.ussd	Send a USSD command to the GSM operator (e.g. *101#). The response is sent as a received SMS message.

Use16BitConcatenationId
Use a 16 bit concatenation ID to send long concatenated SMS messages, the default concatenation ID for SMS is an 8-bit number.

ValidityPeriod
The validity period of sent messages, specified in number of days (1-30) or as hexadecimal value (0x00-0xFF).

The hexadecimal values translate to the following validity period:

0x00 - 0x8F	(**Value** + 1) x 5 minutes (i.e. 5 minutes intervals up to 12 hours)
0x90 - 0xA7	12 hours + ((**Value** -143) x 30 minutes)
0xA8 - 0xC4	(**Value** - 166) x 1 day
0xC5 - 0xFF	(**Value** - 192) x 1 week

**Value is the decimal value of the hexadecimal number*

4.2.2 HTTP Gateway

Only available in Diafaan SMS Server - basic and full editions.

HTTP gateway properties

ShowAdvancedOptions
Show advanced options in the settings for the specific HTTP gateway. This also adds a link to edit the gateway script.

Advanced properties

LogCommunicationToFile
Log all internet communication to file. Use this option only if you encounter problems with the connection to your SMPP account. If enabled, the status windows of the SMPP gateway provides a link to the location on you hard disk where the file is stored.

MaximumBatchSize
The maximum number of messages this gateway will process in one batch. If this value is higher than one and a number of messages are sent at the same time, the gateway loads multiple messages concurrently. This increases the speed in which the messages are sent.

NumberTemplate
The number template can be used to translate the number from one format to another.

For more information see Use number templates.

SourceAddressTemplate
The number template can be used to translate the **From** field from one format to another.

For more information see Use number templates.

QuietHours
A list of 24 hour numbers (separated with semicolons) when sending messages with this gateway is paused. The hour number can be appended with **.1**, **.2**, **.3** or **.4** to specify the first, second, third or fourth quarter of the hour. If the messages cannot be sent with another gateway, they stay in the send queue until the quiet hours are over and this gateway resumes message sending.
00;01;23 pauses message sending from 23:00:00 to 01:59:59.
12.1;12.2;18.1;18.2 pauses message sending from 12:00:00 to 12:29:59 and from 18:00:00 to 18:29:59.

EnableHTTPCallbacks
Enable or disable HTTP callbacks for this gateway. This option is only available when the HTTP callback option is enabled in the general options.

4.2.3 SMPP gateway

Only available in Diafaan SMS Server - basic and full editions.

SMPP server properties

Host or IP number
The host name or IP number of the SMPP server.

Server port
The port number of the SMPP server. Port number 2775 is the default port for the SMPP protocol but other port numbers are also common.

Use SSL/TLS
Check this option if the SMPP server requires an encrypted SSL or TLS connection.

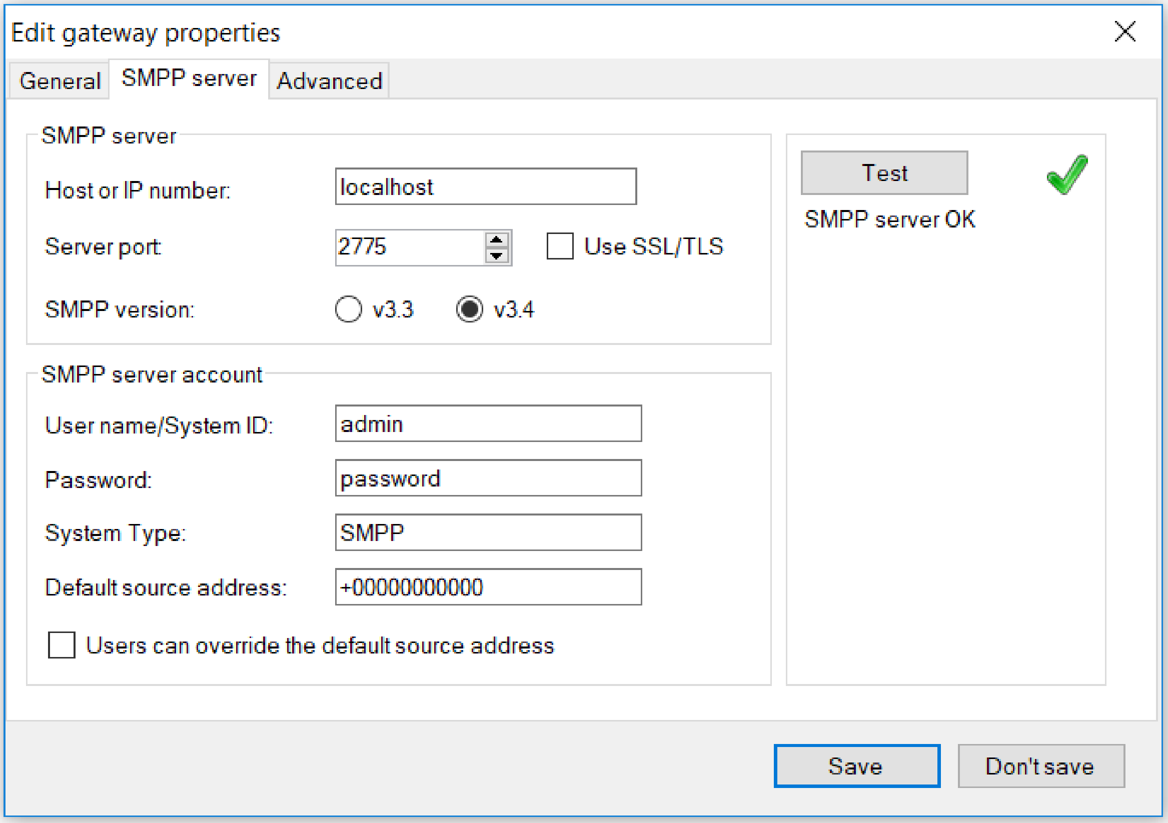
SMPP version
Most SMPP servers support both version 3.3 and 3.4 of the SMPP protocol. Some advanced features of the protocol are only supported in version 3.4 but basic messaging is the same for both versions.
.

User name
The user name (or System ID) of your SMPP account.

Password
The password of your SMPP account.

Default source address
Some SMPP operators require that you send a source address or sender id with each message. The default source address will be used for each message where the user does not provide a source address.

Users can override the default source address
If you do not want users to send their own source addresses with the messages you can disable this option here. Some SMPP operators block messages with an unapproved source address, in the **ApprovedSourceAddress** property in the advanced settings you can provide a list with valid source addresses.



Edit gateway properties

SMPP server

Host or IP number: localhost

Server port: 2775 □ Use SSL/TLS

SMPP version: ○ v3.3 ● v3.4

SMPP server account

User name/System ID: admin

Password: password

System Type: SMPP

Default source address: +00000000000

□ Users can override the default source address

SMPP server OK ✓

Advanced properties

LogCommunicationToFile
Log all internet communication to file. Use this option only if you encounter problems with the connection to your SMPP account. If enabled, the status windows of the SMPP gateway provides a link to the location on you hard disk where the file is stored.

MaximumBatchSize
The maximum number of messages this gateway will process in one batch. If this value is higher than one and a number of messages are sent at the same time, the gateway loads multiple messages concurrently. This increases the speed in which the messages are sent.

NumberTemplate
The number template can be used to translate the number from one format to another.

For more information see Use number templates.

SourceAddressTemplate
The number template can be used to translate the **From** field from one format to another.

For more information see Use number templates.

NumberWhiteList
List of allowed mobile numbers for received SMS messages, messages from other numbers are blocked. When the list does not have any items received SMS messages from all numbers are allowed.

For more information about the white list and black list format see: Use white lists and black lists.

QuietHours
A list of 24 hour numbers (separated with semicolons) when sending messages with this gateway is paused. The hour number can be appended with **.1**, **.2**, **.3** or **.4** to specify the first, second, third or fourth quarter of the hour. If the messages cannot be sent with another gateway, they stay in the send queue until the quiet hours are over and this gateway resumes message sending.
00;01;23 pauses message sending from 23:00:00 to 01:59:59.
12.1;12.2;18.1;18.2 pauses message sending from 12:00:00 to 12:29:59 and from 18:00:00 to 18:29:59.

DisableOnFailedMessages
The number of consecutive failed messages after which the gateway will be disabled, set to 0 to disable this option.

EnableHTTPCallbacks
Enable or disable HTTP callbacks for this gateway. This option is only available when the HTTP callback option is enabled in the general options.

AddressRange
A parameter used when receiving messages. Set this field only if instructed to do so by your SMPP service provider.

AsyncModeWindowSize
If this property is set to a value higher than 1, the SMPP gateway does not wait for a response from the SMPP server before the next message is sent. This can increase message throughput considerably but if the AsyncModeWindowSize is set too high or if the SMPP server does not support asynchronous mode properly it can lead to overloading of the SMPP server.

BindNPI
The Number Plan Indicator parameter for the SMPP bind command, default is 0 (Unknown).

BindTON
The Type Of Number parameter for the SMPP bind command, default is 1 (International number).

CharacterSetReceive
The character set that the SMPP server uses for incoming text messages (not used for Unicode or data messages). Use ISO-8859-1 or ISO-8859-15 when the '@' character is not displayed properly.

CharacterSetReceiveMapping
Character set mapping for outgoing text messages. This field may contain a list of character number pairs in hexadecimal number format. The character pairs are separated by semicolons and the two character numbers are separated by a normal colon. E.g. '2A:2B;32:3F' maps number '0x2A' to character 2B ('+') and number '0x32' to character 3F ('?').

CharacterSetSend
The character set for outgoing text messages. Use IA5, ISO-8859-1 or ISO-8859-15 when the '@' character is not displayed properly.

CharacterSetSendDCS
The Data Coding Scheme (DCS) for outgoing text messages. Default value is 0 for the default character set, 3 for ISO-8859-1/ISO-8859-15.

CharacterSetSendMapping

Character set mapping for outgoing text messages. This field may contain a list of character number pairs in hexadecimal number format. The character pairs are separated by semicolons and the two character numbers are separated by a normal colon. E.g. '2A:2B;32:3F' maps character '2A' ('*') to 0x2B and character '32' ('2') to 0x3F.

IgnoreACCEPTDReceiptStatus
Ignore receipts with status **ACCEPTED**.

KeepAliveInterval
The SMPP Gateway sends an enquire_link message to the SMPP server after a period of inactivity to keep the connection to the SMPP server intact. If the SMPP server keeps breaking the connection after a certain time the timeout interval can be set here.

MaxMessagesPerMinute
This setting limits the amount of messages that the SMPP Gateway will send each minute. Some SMPP servers enforce a strict message throttling and sending more messages then are allowed might actually decrease message throughput.

MaxMessagesPerSecond
This setting limits the amount of messages that the SMPP Gateway will send each second, leave at **0** to disable this limit. This setting should only be used for SMPP servers that have a very strict throttling policy.

MessageIdFormat
Specify the message ID format that the SMPP server uses. **Hexadecimal** is the default option for SMPP 3.3, it translate the message ID from a hexadecimal value to a numeric value and assumes that the message ID in a receipt is a numeric value.

MessageOutHandling
This property specifies how the messages are sent to the SMPP server (only for SMPP versions 3.4 and 5.0). **Submit_SM** sends long messages in multiple message parts, this is the standard option for most SMPP servers. The **Submit_SM_UsePayloadTlv** and **Data_SM** settings send long messages in one SMPP packet using the SMPP 'Payload' TLV field.

PermanentErrorList
A list of GSM modem error codes (separated by semicolons) that are regarded as permanent errors. No more send attempts will be made for the message.

ReceiptTimestampOffset
Offset in (fractional) hours for the receipt time stamp. Use a positive number to increase, and a negative number to decrease the time.

SendLimit
The maximum number of message parts the SMPP Gateway sends per whole hour, day, week or month. 100H means 100 message parts per hour, 300D 300 message parts per day, 1000W 1000 message parts per week and 1000M 1000 message parts per month. When the maximum number of messages is reached, the SMPP Gateway will pause message sending until the start of the next time period. The gateway is not considered down during this period.
By default all send attempts are subtracted from the send limit, also when the SMPP server returns an error. You can add a list of error codes to the **SendLimitErrorCodeExceptionList** property for errors that should not be counted in the send limit.

SendLimitErrorCodeExceptionList
A list of SMPP server error codes that will not affect the send limit (Include 'timeout' in the list for timeout errors. Include 'all' for all errors, including timeout errors). Multiple error codes can be separated with semicolons.

SystemType
The system type is a text that the SMPP Gateway sends to the SMPP server to signal what sort of service it is. Some SMPP servers require a specific system type value.

TransceiverMode
By default, the SMPP Gateway starts up two connections to the SMPP server., one for sending messages and one for receiving messages. SMPP version 3.4 supports a transceiver mode where one connection is used for both sending and receiving messages.

UseMessageIdPrefix
Set to **True** if you want to attach the gateway ID to the message ID, leave this option to **False** if you use multiple SMPP gateway with the same user ID.

ValidityPeriod
The validity period of sent messages, specified as a relative date and time in the format YYMMDDhhmmss000R. E.g. 000003000000000R specifies a validity period of 3 days.
Leave empty for the default validity period of the SMPP server.

ApprovedSourceAddressList
SMPP supports both numeric and text source addresses. To prevent spoofing, some SMPP operators require that source addresses have to be approved before you can use them. the ApprovedSourceAddressList can contain a list of valid source addresses, if the source address is not in the list the default source address is used instead.
If the ApprovedSourceAddressList is empty all source addresses are valid.

DefaultAddressType
The address type (TON) that is used when no specific address type is detected. The detection of the address type is as follows:

- If the address starts with a + it is regarded as an international phone number (TON 1 and NPI 1).
- If characters are detected it is an alphanumeric source address (TON 5 and NPI 0)
- If it has less than **MaxShortCodeLength** numbers it is a shortcode (TON 3 and NPI 1)
- The default address type value is used with NPI 1.

DeliveryConfirmation
Require status updates for the SMS messages after it is sent to the SMSC. If you do not need to know if the SMS messages are delivered on the mobile phone you can disable status updates here.

MaxMessageParts
Messages to send that are longer than 160 characters or 70 characters for unicode messages can be split in multiple SMS messages that are recombined to one message on the handset of the receiver. The MaxMessagePart property limits the amount of partial messages that the gateway sends with a long message. If the message is longer than can be sent with the maximum message parts, the rest of the message is discarded.

MaxShortCodeLength
The maximum number length where the number is regarded as a short code, short codes are encoded as Network Specific numbers.

MessageInHandling
By default, partial received messages are combined to the original message length. Since delivery of some message parts might be delayed this could mean that the delivery of the whole message will be delayed. If, after a certain timeout, not all message parts are received, the partial messages are delivered individually. The messages can be forwarded in on of the the following ways:

CombineMessageParts	This is the default option, multipart message parts are combined to the original long message.
SingleMessageParts	The message parts are not combined but forwarded immediately as individual message parts.
Binary	The received messages are not decoded and combined but forwarded as message type **sms.binary.smpp** with the message field set to **<DCS>,<PID>,<ESMCLASS>,<UDH>,<Data>**.
SMPP_PDU	The received messages are forwarded as raw SMPP PDU packets.

MessageType
The message type is the default value for the SMS message encoding, the default value may be replaced with the message type specified by the sending Connector.
The gateway uses four possible value for the default message type:

Automatic	Standard Text encoding is the default, Unicode encoding will be used only if the message contains characters that are not in the default GSM SMS character set.
StandardText	The default 7-bit GSM SMS character set is used, all other characters are converted to question marks.
Unicode	All messages are sent in Unicode 16-bit encoding.
Data	All messages are sent in binary 8-bit encoding.

The SMPP gateway supports the following custom message types:

sms.automatic	Standard Text encoding is the default, Unicode encoding will be used only if the message contains characters that are not in the default GSM SMS character set.
sms.text	The default 7-bit GSM SMS character set is used, all other characters are converted to question marks.
sms.unicode	The message is sent in Unicode 16-bit encoding.
sms.binary	The message is sent in binary 8-bit encoding.
sms.automatic.flash	Flash SMS, standard Text encoding is the default, Unicode encoding will be used only if the message contains characters that are not in the default GSM SMS character set.
sms.text.flash	Flash SMS, the default 7-bit GSM SMS character set is used, all other characters are converted to question marks.
smpp.pdu	Raw SMPP PDU packet.

ReceiptHandling
This option changes the way SMPP receipts for previously sent messages are handled.

Default	Only set the Status Code, Status Text and, in case of an error, Error Code and readable Error Text.
WriteToErrorText	The same as **Default**, in case of error, write the full receipt to the Error Text .
HandleAsReceivedMessage	Handle the receipt as a received SMS message with message type **smpp.receipt**.
HandleAsReceivedMessageWithTLV	Handle the receipt as a received SMS message with message type **smpp.xml.receipt** and formatted as an XML message with all TLV parameters of the SMPP packet.

Use16BitConcatenationId
Use a 16 bit concatenation ID to send long concatenated SMS messages, the default concatenation ID for SMS is an 8-bit number.

4.2.4 Clickatell gateway

Clickatell properties (Platform)

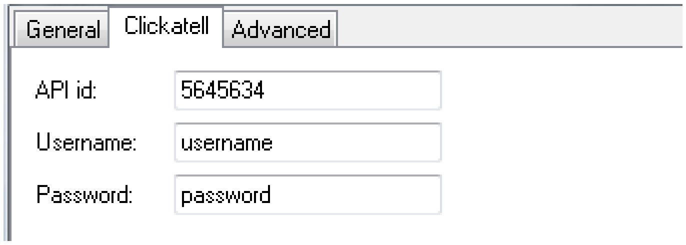
REST API key
The REST API key of your Clickatell Platform REST integration.

Clickatell properties (Communicator / Central)

API id
The HTTP API ID of your Clickatell http account.

Username
The user name of your Clickatell account.

Password
The password of your Clickatell account.



API id: 5645634
Username: username
Password: password

Advanced properties

ApprovedFromList
Clickatell supports both numeric and text sender id's. To prevent spoofing, Clickatell requires that sender id's have to be approved by them before you can use them. To use an originator you have to list them here, one originator per line. All originator addresses that are not in this list are filtered out by the Clickatell gateway.

DefaultFrom
If no originator is supplied or the originator is not in the ApprovedFromList this default originator will be used.

HttpHost (Communicator / Central only)
The Clickatell host name. Both secure and unsecured connections are supported.

https://api.clickatell.com	secure Clickatell host.
http://api.clickatell.com	unsecured Clickatell host.

HttpParameters (Communicator / Central only)
Optional extra parameters that are used when sending an SMS message to Clickatell. An example is the parameter **MO=1**, an indication that the message is Mobile Originated and that the receiver can send a response back.

LogCommunicationToFile
Log all internet communication to file. Use this option only if you encounter problems with the connection to your Clickatell account. If enabled, the status windows of the Clickatell gateway provides a link to the location on you hard disk where the file is stored.

MaximumBatchSize
The maximum number of messages this gateway will process in one batch. If this value is higher than one and a number of messages are sent at the same time, the gateway loads multiple messages concurrently. This increases the speed in which the messages are sent.

NumberTemplate
The number template can be used to translate the number from one format to another.

For more information see Use number templates.

QuietHours
A list of 24 hour numbers (separated with semicolons) when sending messages with this gateway is paused. The hour number can be appended with **.1**, **.2**, **.3** or **.4** to specify the first, second, third or fourth quarter of the hour. If the messages cannot be sent with another gateway, they stay in the send queue until the quiet hours are over and this gateway resumes message sending.
00;01;23 pauses message sending from 23:00:00 to 01:59:59.
12.1;12.2;18.1;18.2 pauses message sending from 12:00:00 to 12:29:59 and from 18:00:00 to 18:29:59.

MaxMessageParts
Messages to send that are longer than 160 characters or 70 characters for Unicode messages can be split in multiple SMS messages that are recombined to one message on the handset of the receiver. The MaxMessagePart property limits the amount of partial messages that the gateway sends with a long message. If the message is longer than can be sent with the maximum message parts, the rest of the message is discarded.

MessageType
The message type is the default value for the SMS message encoding, the default value may be replaced with the message type specified by the sending Connector.
The gateway uses four possible value for the default message type:

Automatic	Standard Text encoding is the default, Unicode encoding will be used only if the message contains characters that are not in the default GSM SMS character set.
StandardText	The default 7-bit GSM SMS character set is used, all other characters are converted to question marks.
Unicode	All messages are sent in Unicode 16-bit encoding.
Data	All messages are sent in binary 8-bit encoding.

The Clickatell gateway supports the following custom message types:

sms.automatic	Standard Text encoding is the default, Unicode encoding will be used only if the message contains characters that are not in the default GSM SMS character set.
sms.text	The default 7-bit GSM SMS character set is used, all other characters are converted to question marks.
sms.unicode	The message is sent in Unicode 16-bit encoding.
sms.binary	The message is sent in binary 8-bit encoding.
sms.automatic.flash	Flash SMS, standard Text encoding is the default, Unicode encoding will be used only if the message contains characters that are not in the default GSM SMS character set.
sms.text.flash	Flash SMS, the default 7-bit GSM SMS character set is used, all other characters are converted to question marks.
sms.unicode.flash	Flash SMS, all messages are sent in Unicode 16-bit encoding.

4.2.5 Paging gateway

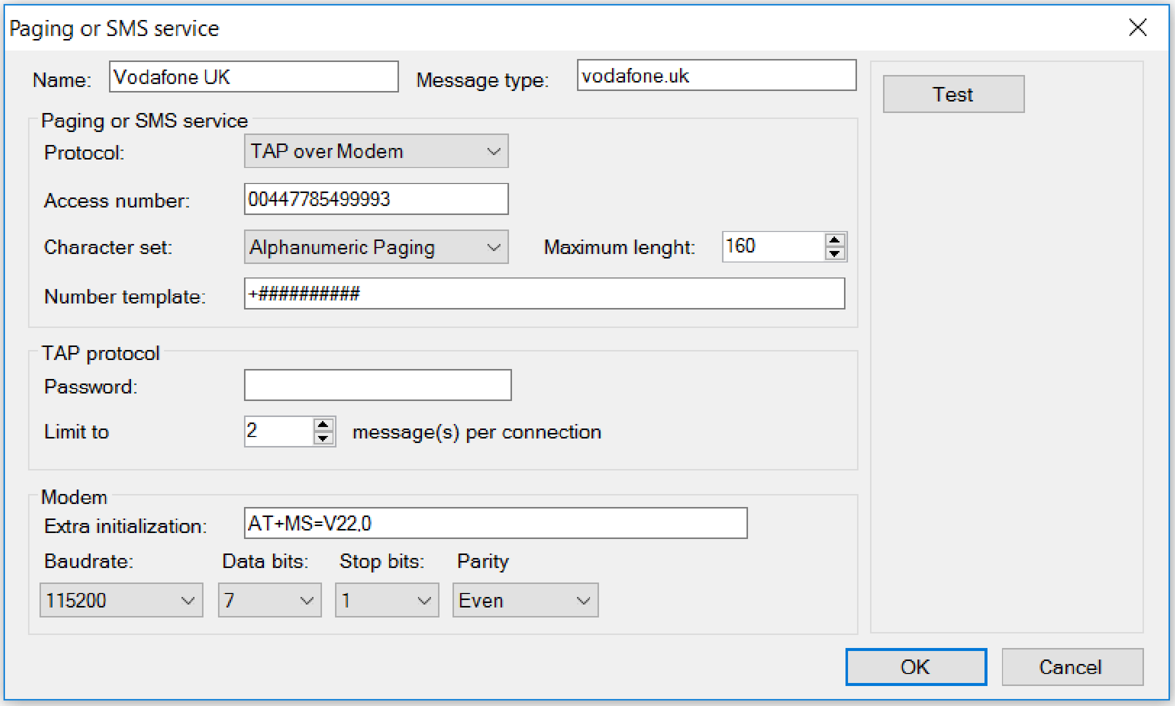
Only available in Diafaan SMS Server - basic and full editions.

Service list

In the Service list tab page you can add, edit or remove the paging services. If one of the services uses a modem you must also provide the COM port where the modem is connected to.

Default
All services must have a different Message Type. The Message Type is used to determine which service is used to send the message. The **Default** button can be used to change the default service that is used to send the message when no corresponding Message Type is found within the message.

Service



Paging or SMS service

Name: Vodafone UK
Message type: vodafone.uk

Paging or SMS service
Protocol: TAP over Modem
Access number: 00447785499993
Character set: Alphanumeric Paging
Maximum lenght: 160
Number template: +##########

TAP protocol
Password:
Limit to 2 message(s) per connection

Modem
Extra initialization: AT+MS=V22,0
Baudrate: 115200
Data bits: 7
Stop bits: 1
Parity: Even

Each service must have a Name and a Message Type. The Message Type is only relevant if the Paging Gateway has more than one service.

Number template
The number template can be used to translate the number from one format to another. For more information see Use number templates.

TAP over modem or serial port

Access Number
The Access Number is the phone number used to connect to the TAP service.

Password
Some TAP Services require a password to accept messages.

Limit to
This value limits the number of messages sent within 1 phone call. If you know that the TAP service accepts more messages per connection you can change this number.

Modem settings
Most TAP services need a connection with 7 data bits, 1 stop bit and even parity. Sometimes it is necessary to set the baudrate to a lower value of 300, 1200 or 2400.

UCP over modem or serial port

Access Number
The Access Number is the phone number used to connect to the UCP service.

Password
Some UCP Services require a password to accept messages.

Limit to
This value limits the number of messages sent within 1 phone call. If you know that the UCP service accepts more messages per connection you can change this number.

Modem settings
Most UCP services need a connection with 8 data bits, 1 stop bit and no parity.

UCP over the Internet

Host name or IP
The host name or IP number used to connect to the UCP service.

Port
Enter the port number of the UCP service here, there is no standard port number for UCP services.

Password
Some UCP Services require a password to accept messages.

Limit to
This value limits the number of messages sent within 1 phone call. If you know that the TAP service accepts more messages per connection you can change this number.

SMTP over the Internet

Host name or IP
The host name or IP number used to connect to the email service.

Port
Enter the port number of the email service here, the standard SMTP port is 25.

User name and Password
If the SMTP server requires a user name and password you can enter them here.

Start TLS and SSL
If the SMTP server requires a secure connection you can enter them here.

Password
Some UCP Services require a password to accept messages.

SNPP over the Internet

Host name or IP
The host name or IP number used to connect to the SNPP service.

Port
Enter the port number of the SNPP service here, the standard SNPP port is 444.

Login ID and Password
If the SNPP server requires a user name and password you can enter them here.

Limit to
This value limits the number of messages sent within 1 phone call. If you know that the SNPP service accepts more messages per connection you can increase this number.

WCTP over the Internet

Host URL
Provide the full URL of the WCTP host. Diafaan SMS Server supports both unsecure (http://) and secure (https://) connections.

Sender ID and Password
If the WCTP server requires a user name and password you can enter them here. Some WCTP servers require a password in the miscInfo field and some in the securityCode field.

DTD URL
If the WCTP server requires a specific DTD URL you can enter it here.

Modem command

The Modem command is not a real protocol but can be used to send a specific AT command to the modem. It can be used to query the modem state or to send a specific instruction to the modem (for instance a watchdog command). When you send a message to the Modem command service the To field must contain the AT command. The modem response is stored in the Status Text of the result.

Advanced properties

LogCommunicationToFile
Log all internet communication to file. Use this option only if you encounter problems with one of the services. If enabled, the status windows of the Paging gateway provides a link to the location on you hard disk where the file is stored.

SendFromAddress
Allow sending the from address with the messages if the protocol supports a sender ID.

QuietHours
A list of 24 hour numbers (separated with semicolons) when sending messages with this gateway is paused. The hour number can be appended with **.1**, **.2**, **.3** or **.4** to specify the first, second, third or fourth quarter of the hour. If the messages cannot be sent with another gateway, they stay in the send queue until the quiet hours are over and this gateway resumes message sending.
00;01;23 pauses message sending from 23:00:00 to 01:59:59.
12.1;12.2;18.1;18.2 pauses message sending from 12:00:00 to 12:29:59 and from 18:00:00 to 18:29:59.

EnableHTTPCallbacks
Enable or disable HTTP callbacks for this gateway. This option is only available when the HTTP callback option is enabled in the general options.

ModemInitialization
The AT modem command used to initialize the modem. If a special initialization command for only one of the services you can add an extra initialization in the **Paging or SMS Service** properties.

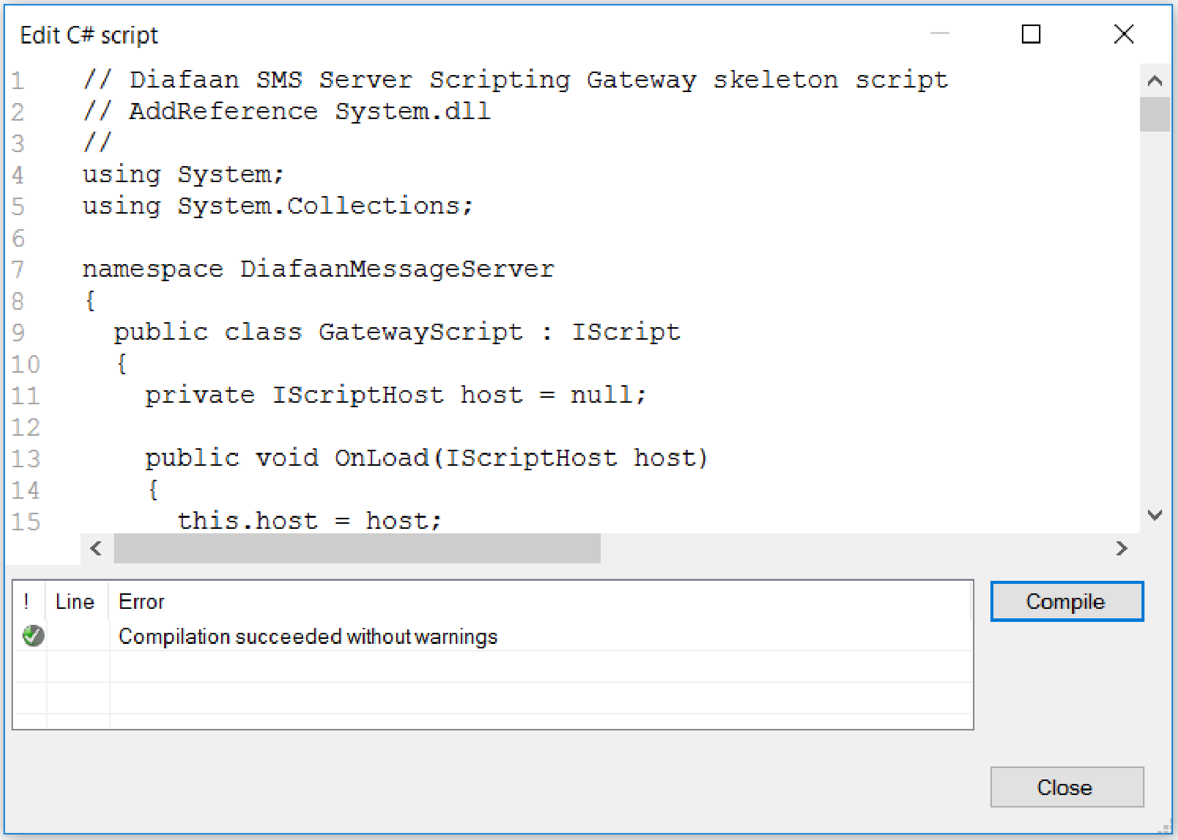
ModemReset
The AT modem command used to reset the modem.

4.2.6 Scripting gateway

Only available in Diafaan SMS Server - basic and full editions.

Edit gateway script

Click on the **Edit C# Script** or **Edit Visual Basic .Net Script** link to edit the script. Use the **Compile** button to check if the script has syntax errors.



```
1   // Diafaan SMS Server Scripting Gateway skeleton script
2   // AddReference System.dll
3   //
4   using System;
5   using System.Collections;
6
7   namespace DiafaanMessageServer
8   {
9     public class GatewayScript : IScript
10    {
11      private IScriptHost host = null;
12
13      public void OnLoad(IScriptHost host)
14      {
15        this.host = host;
```

!	Line	Error
✓		Compilation succeeded without warnings

If the script uses functions that are not in the System.dll assembly, a reference to the required assembly DLL must be added in the comment section at the top of the script. For instance to add a reference to the System.Data.dll assembly in a C# script you can insert:

// AddReference System.Data.dll

At line 3 in the script.

The reference can be the full path name of the DLL file if it is not a standard .NET assembly.

Advanced properties

LogCommunicationToFile
Log all trace log communication to file. You can use this option in combination with the PostTraceLog script function to debug your script. If enabled, the status windows of the Scripting gateway provides a link to the location on the hard disk where the file is stored.

MaximumBatchSize
The maximum number of messages this gateway will process in one batch. If this value is higher than one and a number of messages are sent at the same time, the gateway loads multiple messages concurrently. This increases the speed in which the messages are sent. Leave the MaximumBatchSize on 1 if the script is processor heavy or takes a long time to send a message (more than a few seconds). For small scripts you can increase this value to speed up handling multiple messages.

QuietHours
A list of 24 hour numbers (separated with semicolons) when sending messages with this gateway is paused. The hour number can be appended with **.1**, **.2**, **.3** or **.4** to specify the first, second, third or fourth quarter of the hour. If the messages cannot be sent with another gateway, they stay in the send queue until the quiet hours are over and this gateway resumes message sending.
00;01;23 pauses message sending from 23:00:00 to 01:59:59.
12.1;12.2;18.1;18.2 pauses message sending from 12:00:00 to 12:29:59 and from 18:00:00 to 18:29:59.

EnableHTTPCallbacks
Enable or disable HTTP callbacks for this gateway. This option is only available when the HTTP callback option is enabled in the general options.

Gateway skeleton script

The gateway skeleton script that is installed after adding the gateway makes it easier to handle events and provides a number of methods you can use to send messages, enter received messages and add log entries.

Events

OnMessageToSend is called when there is a message the script should send out.

OnStatusRequest is called when the script should query the status of a previously sent message.

Methods

PostSendResult sets the send result of a previous handled OnMessageToSend message.

PostSendResultUpdate updates the send result of a previously handled OnMessageToSend message.

PostDispatchMessage dispatches the OnMessageToSend message to a different gateway.

PostReceiveMessage adds a new received message.

PostEventLog adds a line to the event log.

PostTraceLog Adds a line to the communication file.

4.2.7 Hypermedia SMS PRO gateway

Only available in Diafaan SMS Server - basic and full editions.

SMS PRO Gateway properties

Host or IP number
Enter the host name or IP number of the Hypermedia SMS PRO gateway.

Server port
The port number of the Hypermedia SMS PRO gateway. Port number 63333 is the default port for the device.

Client ID
A unique Client ID (user name), the Client ID can be freely chosen by the user. If another client logs in with the same ID as an existing client then the new client will cause the disconnection of the old client and a message will be sent to each indicating this, therefore it is recommended that each new client that connects to the gateway should use a unique ID. Note that if no ID is given, a unique ID will be generated for the client by the gateway.

Password
The password of the Hypermedia SMS PRO gateway.

SIM card routing properties

Use all SIM cards in the SMS PRO Gateway
Use all SIM cards in the SMS PRO Gateway to send and receive SMS messages.

Use SIM cards:
Enter a list with all SIM cards in the Hypermedia SMS PRO Gateway you want to use to send and receive SMS messages. Click the **Update** button to refresh the available SIM card list. The SIM card list may contain multiple SIM cards separated by semicolons, e.g. **<21#1>;<21#2>**

You can assign a GSM number to each SIM card that is inserted in the **To** field of received messages, e.g. **+11111111111 <21#1>;+22222222222 <21#2>**

Advanced properties

LogCommunicationToFile
Log all internet communication to file. Use this option only if you encounter problems with the connection to your SMPP account. If enabled, the status windows of the SMPP gateway provides a link to the location on you hard disk where the file is stored.

MaximumBatchSize
The maximum number of messages this gateway will process in one batch. If this value is higher than one and a number of messages are sent at the same time, the gateway loads multiple messages concurrently. This increases the speed in which the messages are sent.

NumberTemplate
The number template can be used to translate the number from one format to another.

For more information see Use number templates.

NumberWhiteList
List of allowed mobile numbers for received SMS messages, messages from other numbers are blocked. When the list does not have any items received SMS messages from all numbers are allowed.

For more information about the white list and black list format see: Use white lists and black lists.

QuietHours
A list of 24 hour numbers (separated with semicolons) when sending messages with this gateway is paused. The hour number can be appended with **.1**, **.2**, **.3** or **.4** to specify the first, second, third or fourth quarter of the hour. If the messages cannot be sent with another gateway, they stay in the send queue until the quiet hours are over and this gateway resumes message sending.
00;01;23 pauses message sending from 23:00:00 to 01:59:59.
12.1;12.2;18.1;18.2 pauses message sending from 12:00:00 to 12:29:59 and from 18:00:00 to 18:29:59.

EnableHTTPCallbacks
Enable or disable HTTP callbacks for this gateway. This option is only available when the HTTP callback option is enabled in the general options.

DeliveryConfirmation
Require status updates for the SMS messages after it is sent to the SMSC. If you do not need to know if the SMS messages are delivered on the GSM phone or if the status updates causes errors sending the messages you can disable status updates.

MaxMessageParts
Messages to send that are longer than 160 characters or 70 characters for unicode messages can be split in multiple SMS messages that are recombined to one message on the handset of the receiver. The MaxMessagePart property limits the amount of partial messages that the gateway sends with a long message. If the message is longer than can be sent with the maximum message parts, the rest of the message is discarded.

MessageType
The message type is the default value for the SMS message encoding, the default value may be replaced with the message type specified by the sending Connector.
The gateway uses four possible value for the default message type:

Automatic	Standard Text encoding is the default, Unicode encoding will be used only if the message contains characters that are not in the default GSM SMS character set.
StandardText	The default 7-bit GSM SMS character set is used, all other characters are converted to question marks.
Unicode	All messages are sent in Unicode 16-bit encoding.
Data	All messages are sent in binary 8-bit encoding.

The GSM Modem gateway supports the following custom message types:

sms.automatic	Standard Text encoding is the default, Unicode encoding will be used only if the message contains characters that are not in the default GSM SMS character set.
sms.text	The default 7-bit GSM SMS character set is used, all other characters are converted to question marks.
sms.unicode	The message is sent in Unicode 16-bit encoding.
sms.binary	The message is sent in binary 8-bit encoding.
sms.automatic.flash	Flash SMS, standard Text encoding is the default, Unicode encoding will be used only if the message contains characters that are not in the default GSM

	SMS character set.
sms.text.flash	Flash SMS, the default 7-bit GSM SMS character set is used, all other characters are converted to question marks.
sms.unicode.flash	Flash SMS, all messages are sent in Unicode 16-bit encoding.

ValidityPeriod
The validity period of sent messages, specified in number of days (1-30) or as hexadecimal value (0x00-0xFF).

The hexadecimal values translate to the following validity period (Value is the decimal value):

0x00 - 0x8F	(**Value** + 1) x 5 minutes (i.e. 5 minutes intervals up to 12 hours)
0x90 - 0xA7	12 hours + ((**Value** -143) x 30 minutes)
0xA8 - 0xC4	(**Value** - 166) x 1 day
0xC5 - 0xFF	(**Value** - 192) x 1 week

AdditionalMessageParameters
List of additional Hypermedia SMS PRO API message parameters, separated by semicolons. The parameter consists of a parameter name and a value, separated by a colon (':'). The value is a hexadecimal encoded representation of the information, for example '1' must be encoded as '31' and 'abcd' must be encoded as '61626364'.

Examples:

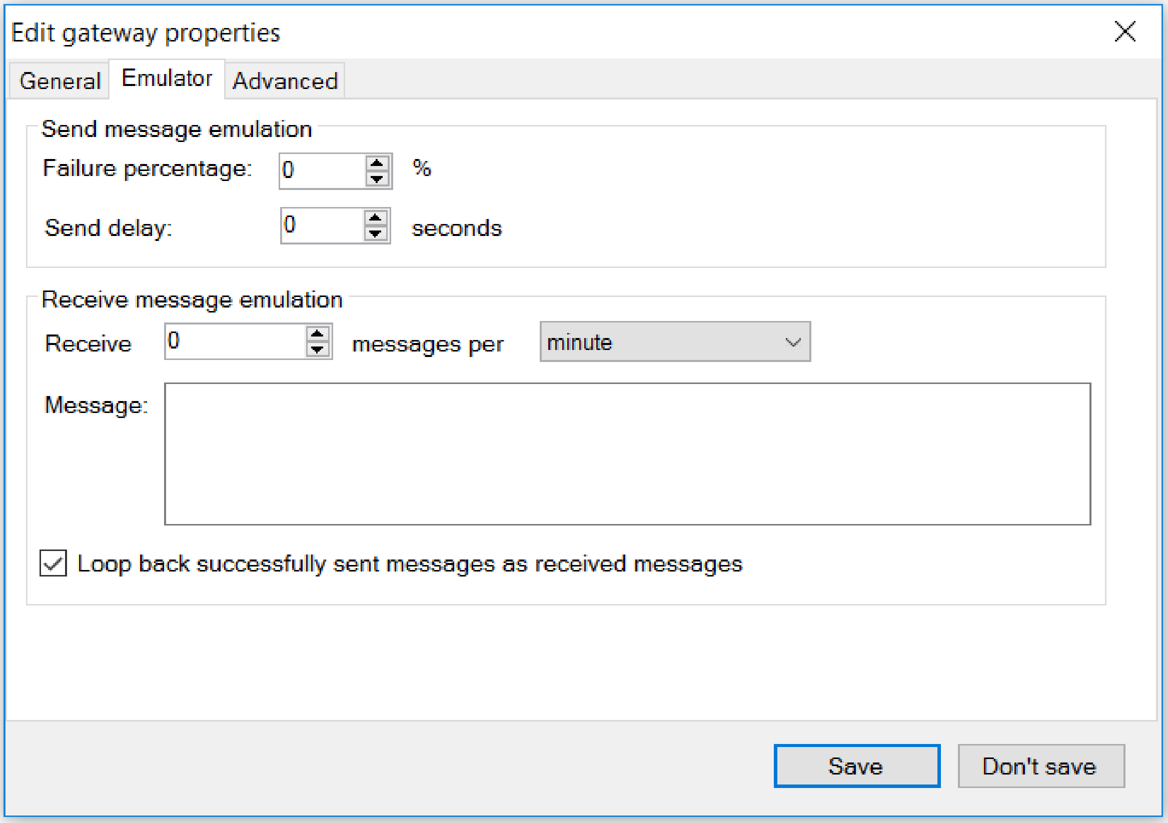
Set priority to 1:
priority:31

Set queue type to 'master':
queue_type:6d6173746572

Set both options:
priority:31;queue_type:6d6173746572

4.2.8 Emulator gateway

Emulator Gateway properties



Edit gateway properties

Send message emulation

Failure percentage: 0 %

Send delay: 0 seconds

Receive message emulation

Receive 0 messages per minute

Message:

☑ Loop back successfully sent messages as received messages

Failure percentage

The percentage of messages that are flagged as failed by the Emulator Gateway, the messages fail with error code 1640 and error text "Randomly generated error".

Send delay

The minimum number of seconds between each emulated message.

Receive ... messages per ...

If this value is larger than '0', the Emulator Gateway generates one or more received SMS messages every second, minute, hour or day.

Message

The text of the emulated received messages.

Loop back successfully sent messages as received messages

Convert all messages that are flagged as 'sent' by the Emulator Gateway to a received SMS message with the same message text.

Advanced properties

LogCommunicationToFile

Log all HTTP Callback message_in requests from this gateway to a file.

MaximumBatchSize

The maximum number of messages this gateway will process in one batch. If this value is higher than one and a number of messages are sent at the same time, the gateway loads multiple messages concurrently. This increases the speed in which the messages are sent.

QuietHours

A list of 24 hour numbers (separated with semicolons) when sending messages with this gateway is paused. The hour number can be appended with **.1**, **.2**, **.3** or **.4** to specify the first, second, third or fourth quarter of the hour. If the messages cannot be sent with another gateway, they stay in the send queue until the quiet hours are over and this gateway resumes message sending.
00;01;23 pauses message sending from 23:00:00 to 01:59:59.
12.1;12.2;18.1;18.2 pauses message sending from 12:00:00 to 12:29:59 and from 18:00:00 to 18:29:59.

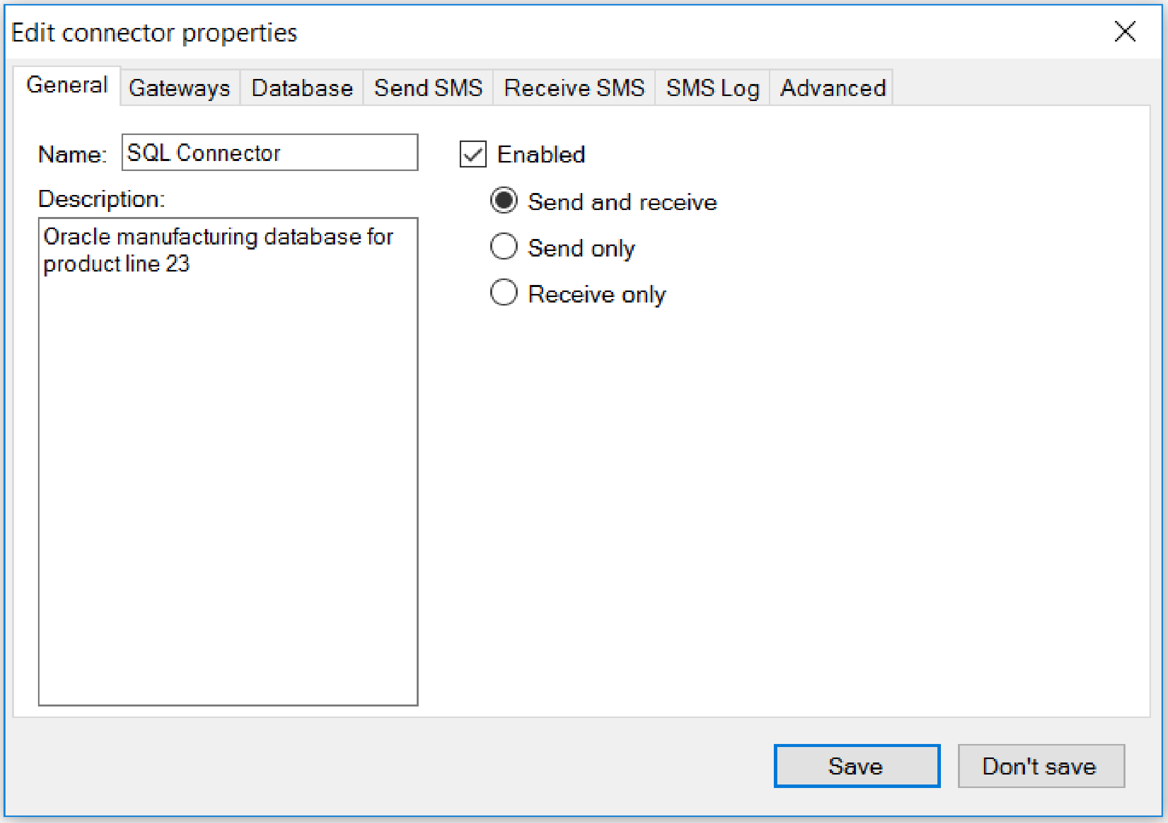
EnableHTTPCallbacks

Enable or disable HTTP callbacks for this gateway. This option is only available when the HTTP callback option is enabled in the general options.

4.3 Connector properties

General properties

If you want to temporarily disable the connector or set the connector to send or receive only you can set the appropriate options here.



Edit connector properties

Name: SQL Connector

Description: Oracle manufacturing database for product line 23

☑ Enabled

● Send and receive

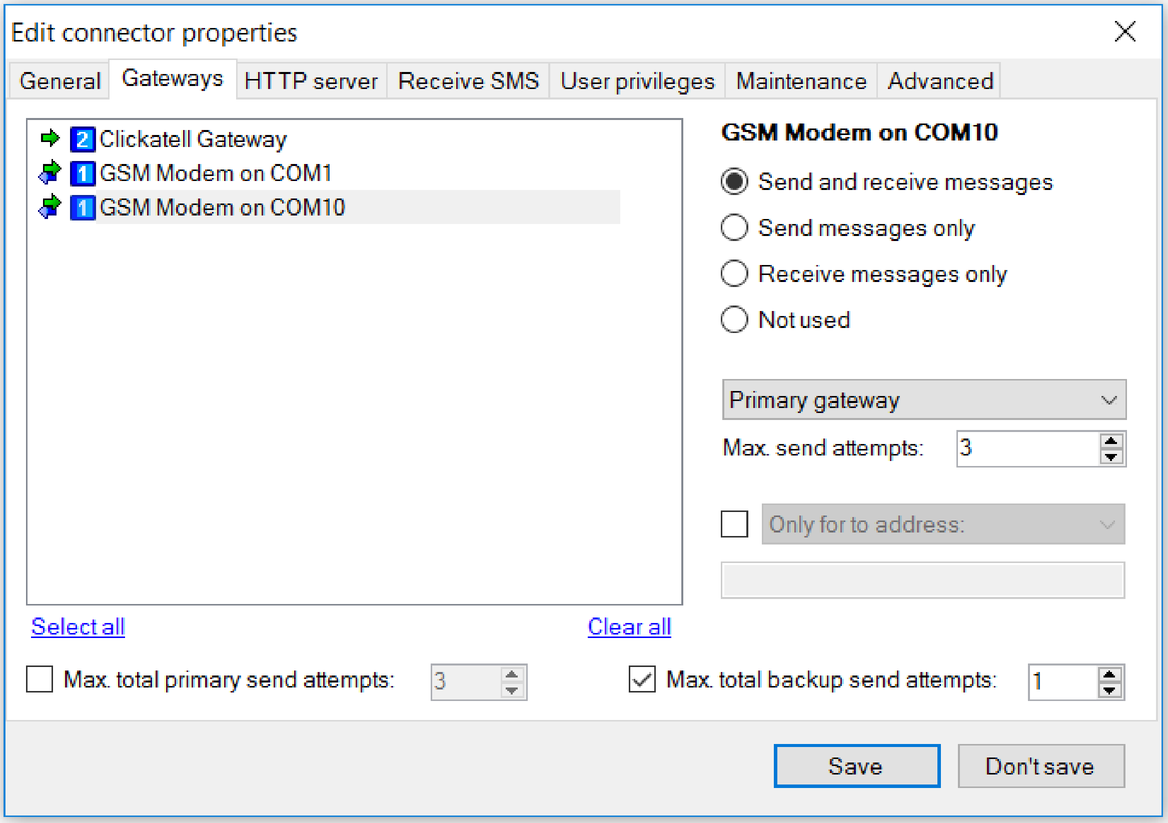
○ Send only

○ Receive only

Gateway routing properties

For each available gateway you can choose if you want to use the gateway to send and receive messages, send or receive only or not at all.

You can assign the gateway as a primary gateway or as a backup gateway. Diafaan SMS Server will first try to send the message with one of the assigned primary gateways. If sending the message failed it will be repeated with a maximum of **Max send Attempts** for each primary gateway. If sending the message is still not successful on all primary gateways, the same procedure is used with the assigned backup gateway(s).



2 Clickatell Gateway
1 GSM Modem on COM1
1 GSM Modem on COM10

GSM Modem on COM10

● Send and receive messages
○ Send messages only
○ Receive messages only
○ Not used

Primary gateway

Max. send attempts: 3

□ Only for to address:

□ Max. total primary send attempts: 3

☑ Max. total backup send attempts: 1

Connector specific properties

- Web connector
- Email connector
- SQL connector
- Scripting connector
- SMPP connector (basic and full editions only)

4.3.1 Web connector

HTTP server properties

Check **Enable HTTP Server** to start the web server on the requested port.

Enable Web Application

The Web Application API gives access to the SMS client web application. With the web application one (light edition) or more (basic- and full editions) users can send SMS messages, view received SMS messages and maintain a contact list using any modern Internet browser.

It is possible use the HTML files in the data folder of the Web Connector instead of the built-in HTML files. This makes it possible to translate the web application or to apply the styling of your company. Select Show folder to see the HTML folder of the Web Connector and Refresh files to (over)write the internal files to the HTML folder.

Translation to other languages

All language dependent information for the web application is located in the index.html file. This is the only file that needs to be changed when you want to translate the web application in another language. If you want to support multiple languages you can copy the index.html file and give each language its own html file with another name (e.g. index-de.html, index-sp.html etc.).

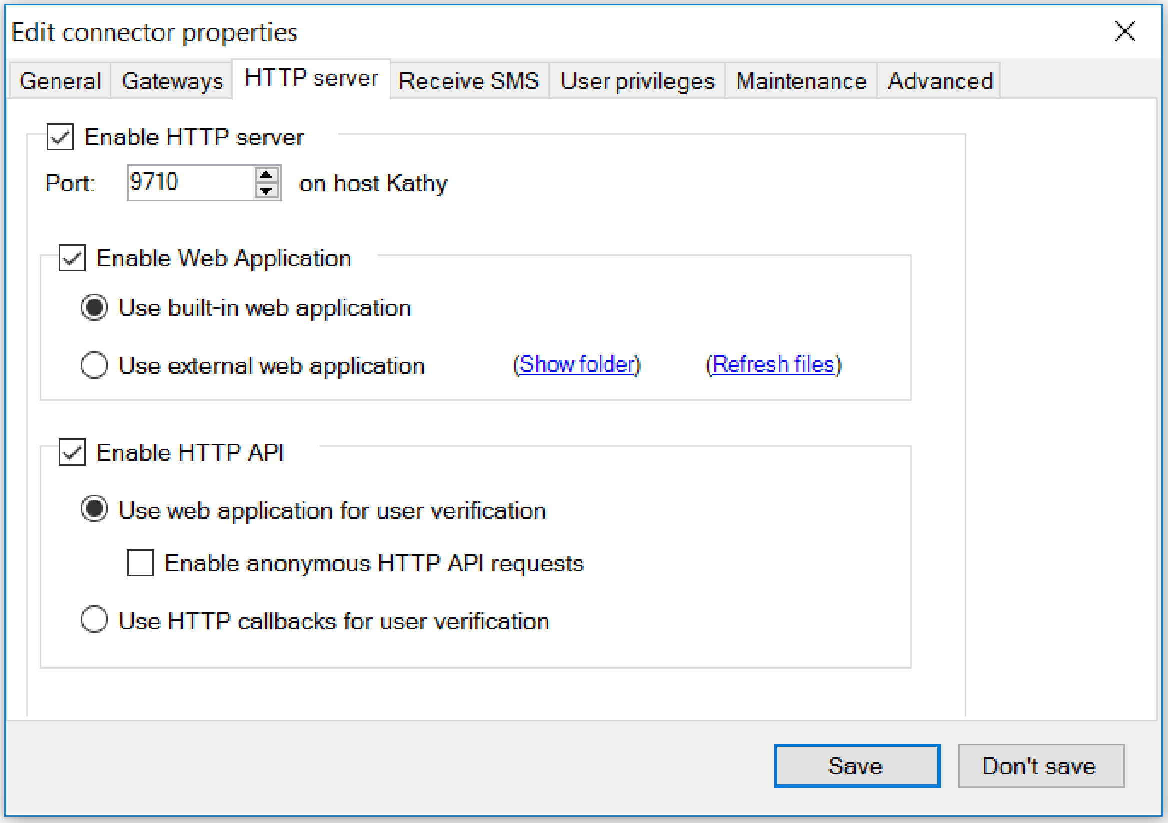
Be careful with changes to the HTML tags of the index.html file since this may interfere with the proper functioning of the web application.

Sound alert for received messages

If you use the external web application, you can replace the (empty) **alert-sms-received.mp3** file with another mp3 file to let the browser play a sound alert for every received message. The browser will only play the file when the Web Application is loaded and the user is logged in. Some browsers might not have the ability to play the alert file, or will only play it when the Web Application is the active web page.

Enable HTTP API

The HTTP API provides a simple way to send messages and to view received messages from a browser or external application. See the HTTP API specifications for more information about the HTTP API. If **Anonymous HTTP API requests** is enabled messages can be sent with the HTTP API without user name and password. In the basic and full edition of Diafaan SMS Server it is also possible to use HTTP callbacks instead of the user list from the Web Application for user verification.



☑ Enable HTTP server

Port: 9710 on host Kathy

☑ Enable Web Application

● Use built-in web application

○ Use external web application (Show folder) (Refresh files)

☑ Enable HTTP API

● Use web application for user verification

□ Enable anonymous HTTP API requests

○ Use HTTP callbacks for user verification

Receive SMS properties

Forward received SMS messages to user inbox

Received SMS messages can be forwarded to the inbox of the administrator and also to other users of the web application. When you create a new user in the web application you can also specify that the user will see all received messages in his or her inbox.

When **Reply to user** is selected the web connector looks if there was a previous message sent to the mobile phone number. If there is a mobile number match, the message is sent as a reply to the inbox of the last sender of the message to that particular mobile number.

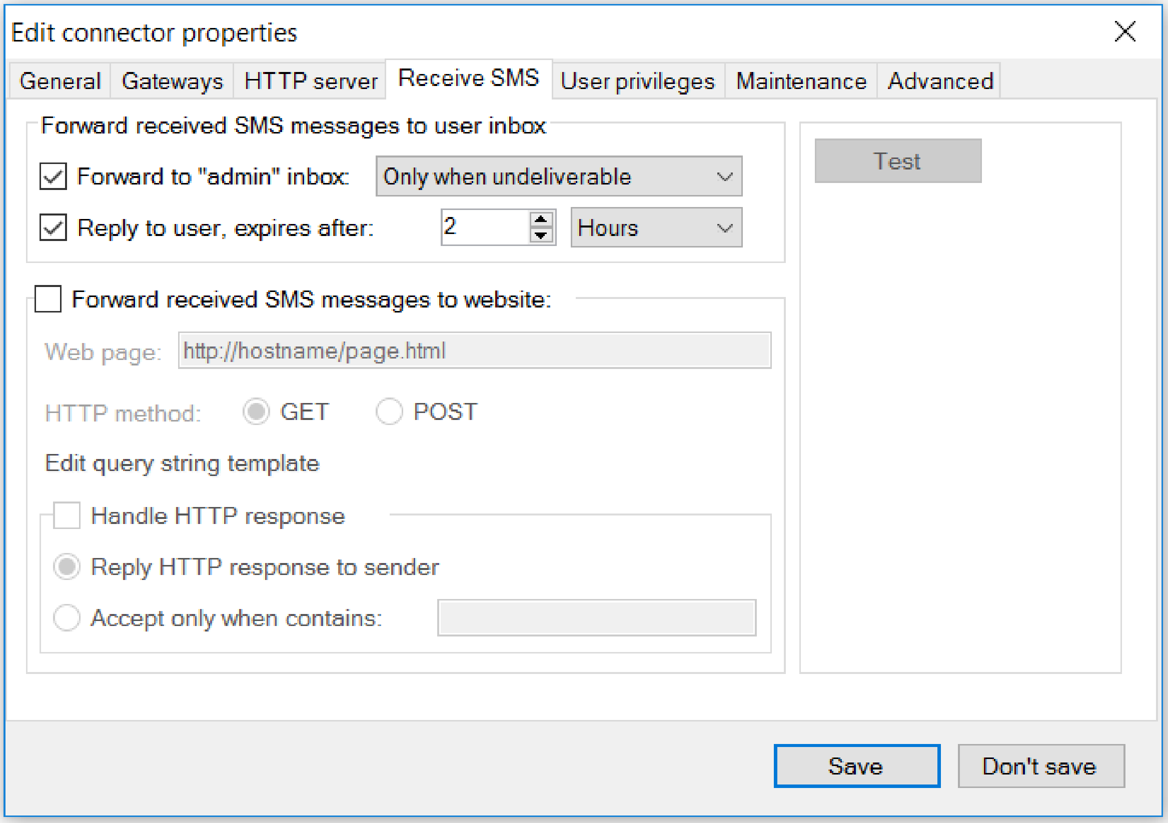
Forward received SMS messages to website

Enter the web page and HTTP method where you want to deliver the received SMS messages. You should also edit the query string template to indicate what parts of the received SMS messages are forwarded to the web server.

Handle HTTP response

The HTTP response from the web page can be used to send a reply back to the sender of the message. This option makes it possible to use PHP, ASP.NET or another web server script language to perform powerful database driver SMS queries. If the HTTP response is an empty text, no response will be sent back.

The HTTP response can also be used to determine if the web page accepts the message. The Web connector will repeat the HTTP command a number of times if the supplied text is not in the HTTP response.



Forward received SMS messages to user inbox

☑ Forward to "admin" inbox: Only when undeliverable

☑ Reply to user, expires after: 2 Hours

□ Forward received SMS messages to website:

Web page: http://hostname/page.html

HTTP method: ● GET ○ POST

Edit query string template

□ Handle HTTP response

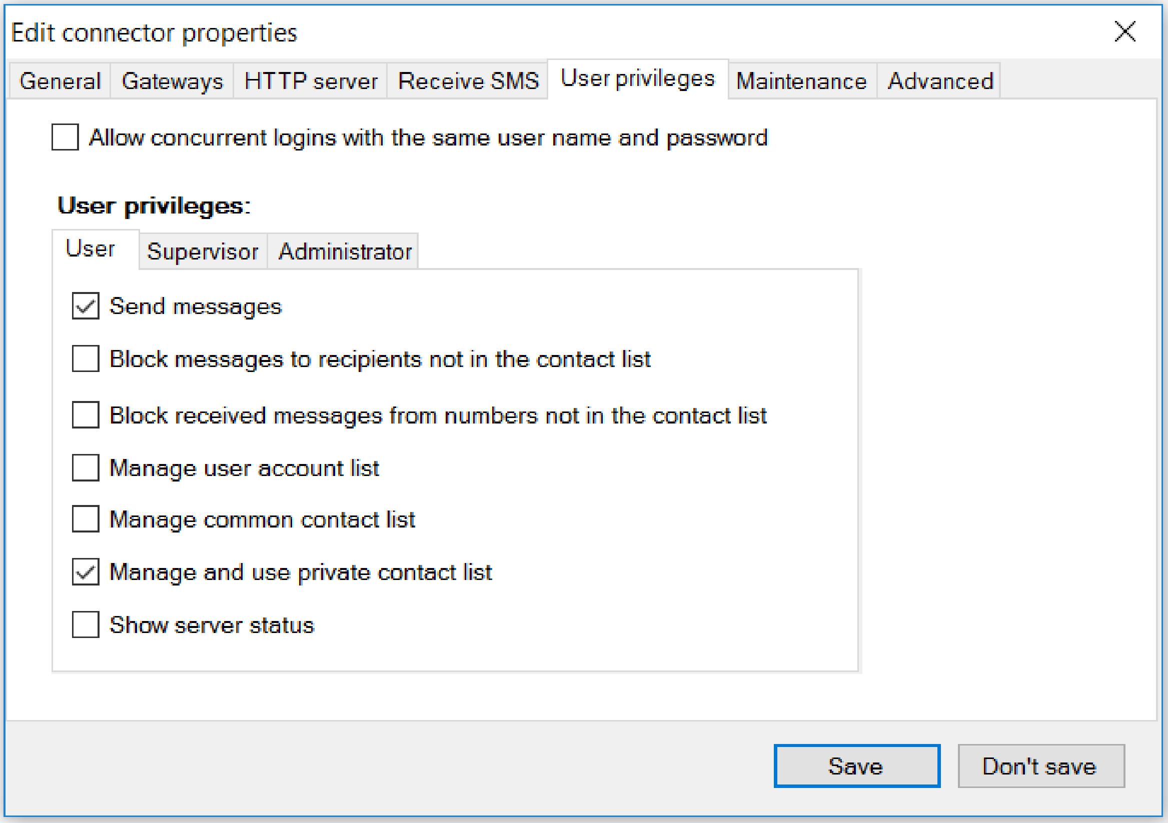
● Reply HTTP response to sender

○ Accept only when contains:

User privileges

In situation where multiple people need to monitor the same sent- and received messages, like small call centers or customer support departments it can be useful to log in to the same user account at the same time. If you check **Allow concurrent logins with the same user name and password** this is possible. All information, like the sent- and received messages and the contact list are synchronized on all web browser that are logged in to the same account.

For all three user types, **users**, **supervisors** and **administrators**, you can set the privileges that the user will have. There is no inherent difference between the three user types. you can enable or disable the same options for each user type.



□ Allow concurrent logins with the same user name and password

User privileges:

☑ Send messages

□ Block messages to recipients not in the contact list

□ Block received messages from numbers not in the contact list

□ Manage user account list

□ Manage common contact list

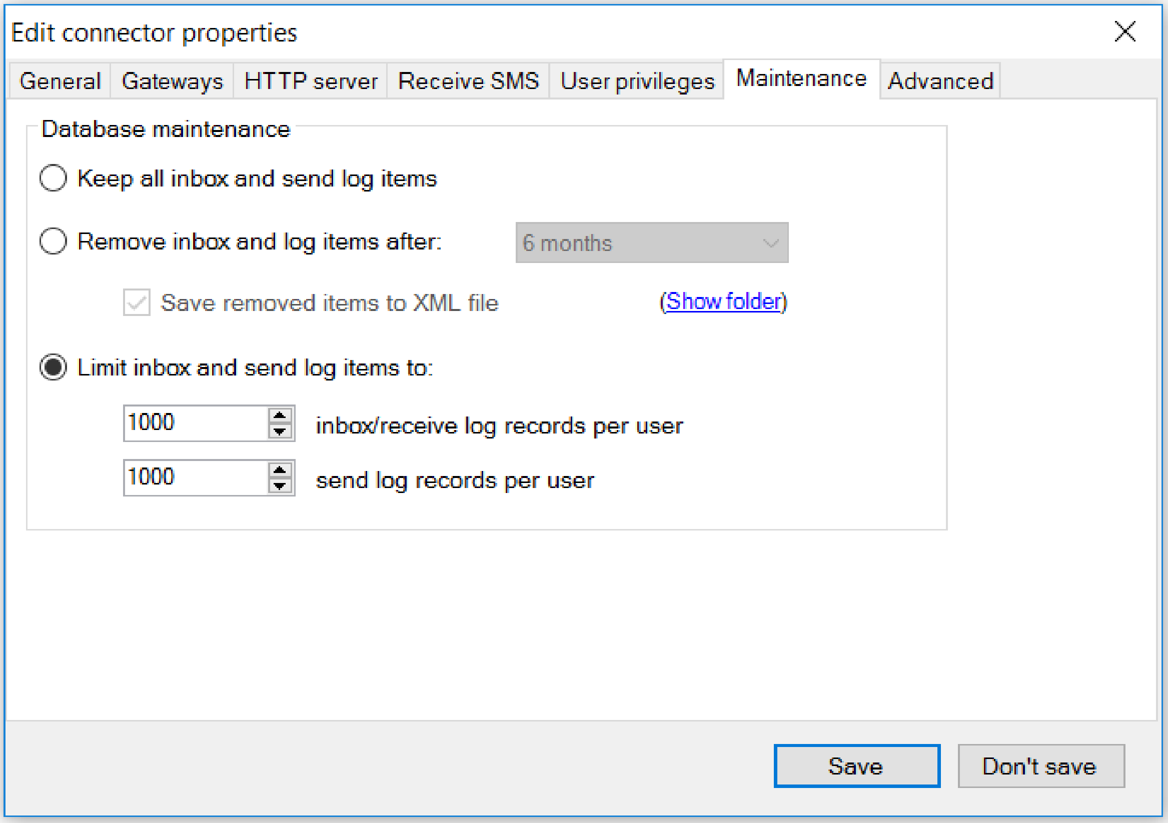
☑ Manage and use private contact list

□ Show server status

Maintenance

Received messages that are stored in one of the inboxes and messages sent with the Web Connector are stored in a database. To limit the number of records in the database you can check the **remove inbox and log items** option. If checked, the oldest database records are removed once a day. You can make a backup copy of the removed records by checking the **save removed items** option.

If the **Limit inbox and send log to** option is selected, the number of log items per user in the Send Log and the Inbox/Receive Log is limited to a fixed number of records. If the log database is getting too large, viewing the logs will get gradually slower and a fixed log limit helps to prevent this.



Database maintenance

○ Keep all inbox and send log items

○ Remove inbox and log items after: 6 months

☑ Save removed items to XML file (Show folder)

● Limit inbox and send log items to:

1000 inbox/receive log records per user

1000 send log records per user

Advanced properties

LogCommunicationToFile
Log all internet communication to file. Use this option only if you encounter problems with the connection to the Web connector or to the external web server. If enabled, the status windows of the Web connector provides a link to the location on you hard disk where the file is stored.

MaximumBatchSize
The maximum number of messages this connector will process in one batch. If this value is higher than one and a number of messages are sent at the same time, the connector loads multiple messages concurrently. This increases the speed in which the messages are processed.

SendPriorityAPI
Default message send priority for the HTTP API, set to 0 to allow a custom priority for individual messages. A higher number ensures that messages are placed higher in the message send queue and are sent before messages with a lower priority.

SendPriorityWebApp
Message send priority for the Web Application. A higher number ensures that messages are placed higher in the message send queue and are sent before messages with a lower priority.

NumberTemplate
The number template can be used to translate the number from one format to another.

For more information see Use number templates.

NumberWhiteList
List of allowed destination numbers (to address) for outgoing SMS messages, messages to other numbers are blocked. When the list does not have any items, messages to all numbers are allowed.

For more information about the white list and black list format see: Use white lists and black lists.

EnableHTTPCallbacks
Enable or disable HTTP callbacks for this gateway. This option is only available when the HTTP callback option is enabled in the general options.

ErrorResponse
ERROR response string for the HTTP API calls, leave empty for the default error response.

OkResponse
OK response string for the HTTP API calls, leave empty for the default OK response.

RemoveDuplicateNumbers
Remove duplicate numbers from the HTTP API call before sending the message.

SendServerCommandEnabled
Enables the 'send-server-command' HTTP API call. The 'send-server-command' HTTP API command can be used to change the settings of Diafaan SMS Server from a web browser.

HTTPClientSendAttempts
Maximum number of times the HTTP Client tries to deliver a received message.
2nd attempt after at least 30 seconds
3rd attempt after at least 5 minutes
4th attempt after at least 30 minutes
5th attempt after at least 1 hour
6th and further attempts after at least 12 hour

HTTPServerBacklog
The number of incoming HTTP connection the server accepts before denying subsequent connections.

HTTPServerIPNumber
The IP number of the local interface where the server should bind to, leave empty to bind all available IP numbers to the HTTP service.

HTTPServerSSLCertificate
The full path and filename of the X509 (OpenSSL) certificate required to enable secure connections to the HTTP server.

HTTPServerSSLCertificatePassword
The password of the X509 certificate used for SSL (https://) connections.

4.3.1.1 HTTP API specifications

HTTP commands, all editions

HTTP commands can be sent with both GET or POST methods, the POST method requires a url-encoded BODY with content-type 'application/x-www-form-urlencoded '. In this manual the GET method is used for all examples.

send-message

Send one or more messages.

parameters

username	optional	The name of the user, if omitted the guest account is used
password	optional	The password of the user
to	required	One or more phone/pager numbers, separated by semicolons
contact-name	required	One or more contact names and/or groups from the user's contact list, separated by semicolons
from	optional	Sender id
message	optional	Message text
message-type	optional	Message type to indicate if the message is text, Unicode, flash etc.
message-guid	optional	Unique message id, this parameter overrides the standard message ID of Diafaan SMS Server
message-info	optional	Freely usable message info, this parameter is not used internally by Diafaan SMS Server
gateway	optional	One or more gateway names, separated by semicolons, that may be used to send the message
priority	optional	A higher number ensures that the message is placed higher in the message send queue and is sent before messages with a lower priority.
validity-period	optional	The validity period of the SMS message, in seconds.
scheduled	optional	The date and time when the message should be sent, in ISO 8601 format. E.g.: '2020-12-31T19:35:00', or including time zone: '2020-12-31T19:35:00.0000000-07:00'

returns

OK: message-id
or
ERROR:error-code error-message

If multiple messages are sent this command returns one response line per message in the order at which the numbers are located in the to and contact-name fields. Since numbers may be blocked and the contact-name can be a group, the number of return lines may be different from the number of to or contact-name entries.

example

http://localhost:9710/http/send-message?username=admin&password=&to=%2B44xxxxxxxx&message-type=sms.automatic&message=Message+Text

request-status-update

Get the status of a previously sent message.

parameters

username	optional	The name of the user, if omitted the guest account is used
password	optional	The password of the user
message-id	mandatory	Message id of the previously sent message
info	optional	The type of status information that is returned: "status" status code and status message (default) "error-code" error code and error text (if available) "gateway" the name of the gateway that sent the message (if available)

returns

STATUS:status-code status-message

or
ERROR-CODE:error-code error-message
or
GATEWAY:gateway-name
or (if the API call fails)
ERROR:error-code error-message

example
http://localhost:9710/http/request-status-update?username=admin&password=&message-id=c3b0c20a-b63e-4bbf-919f-3d4251a8b9c0

remove-queued-message

Remove a previously sent message from the send queue.

parameters

username	optional	The name of the user, if omitted the guest account is used
password	optional	The password of the user
message-id	mandatory	Message id of the previously sent message

returns
OK: status-message
or
ERROR: error-message

example
http://localhost:9710/http/remove-queued-message?username=admin&password=&message-id=c3b0c20a-b63e-4bbf-919f-3d4251a8b9c0

request-server-status

Get the status of the Diafaan SMS Server and all the Gateways and Connectors statistics.

parameters

username	mandatory	The name of the user
password	mandatory	The password of the user

returns
An XML file with the status and statistics of Diafaan SMS Server.
or
ERROR:error-code error-message

example
http://localhost:9710/http/request-server-status?username=admin&password=

request-received-messages

Get a list of received messages in the inbox.

parameters

username	optional	The name of the user, if omitted the guest account is used
password	optional	The password of the user
limit	optional	The maximum number of messages returned
remove	optional	Set to "1" to remove the returned messages from the database
order	optional	Set to "newest" to get newest messages first, otherwise oldest messages are displayed first.

returns
An XML file with a list of the requested messages.
or
ERROR:error-code error-message

example
http://localhost:9710/http/request-received-messages?username=admin&password=&limit=10&order=newest

```
<?xml version='1.0' encoding='UTF-8'?>
<MessageList Count="2">
  <Message>
    <From>+44xxxxxx</From>
    <To />
    <MessageText>Message text 1</MessageText>
    <MessageType>sms.text</MessageType>
    <MessageId>ae97a8b7-5336-445a-ae67-e658aa8f6aed</MessageId>
    <MessagePart>-1</MessagePart>
    <ReceivedMessageParts>1</ReceivedMessageParts>
    <TotalMessageParts>1</TotalMessageParts>
    <SendTime>20100219200855</SendTime>
    <ReceiveTime>20100219200848</ReceiveTime>
    <Gateway>GSM Modem Gateway</Gateway>
    <SMSC>+44xxxxxxxxx</SMSC>
    <PDU>079458560488019539B53F50000012091028055400DD47719649987CD25D87506</PDU>
    <IsRemoved>0</IsRemoved>
  </Message>
  <Message>
    <From>+33xxxxxxxx</From>
    <To />
    <MessageText>Message text 2</MessageText>
    <MessageType>sms.text</MessageType>
    <MessageId>ae97a8b7-5336-445a-ae67-e658aa8f6aed</MessageId>
    <MessagePart>-1</MessagePart>
    <ReceivedMessageParts>1</ReceivedMessageParts>
    <TotalMessageParts>1</TotalMessageParts>
    <SendTime>20100219200855</SendTime>
    <ReceiveTime>20100219200848</ReceiveTime>
    <Gateway>GSM Modem Gateway</Gateway>
    <SMSC>+33xxxxxxxxx</SMSC>
    <PDU>07474523460488019539B53F50000012091028055400DD47719649987CD25D87506</PDU>
    <IsRemoved>0</IsRemoved>
  </Message>
</MessageList>
```

HTTP commands, basic- and full edition only

Warning: The 'send-server-command' HTTP API command can leave Diafaan SMS Server in an unstable state if one of the server setting, gateway settings, connector settings or routing settings is set to an unsupported value.

- *The 'send-server-command' HTTP API command can only be used by the administrator (admin) account or by a user with administrator privileges.*

- *In Diafaan SMS Server 4.0 or higher this command is disabled by default, it can be enabled with the* ***SendServerCommandEnabled*** *property in the advanced settings of the Web Connector.*

send-server-command

Send a command to the server to start and stop a gateway or connector or change one of the settings of the server

Start one or more gateways

username	mandatory	The name of the user, the user must have administrator privileges
password	mandatory	The password of the user
command	mandatory	"start-gateway"
gateway	optional	The name of the gateway or a list of gateway names, separated by semicolons. Leave empty to affect all gateways

example
http://localhost:9710/http/send-server-command?username=admin&password=&command=start-gateway&gateway=GSM+Modem+Gateway

Stop one or more gateways

username	mandatory	The name of the user, the user must have administrator privileges
password	mandatory	The password of the user
command	mandatory	"stop-gateway"
gateway	optional	The name of the gateway or a list of gateway names, separated by semicolons. Leave empty to affect all gateways

example
http://localhost:9710/http/send-server-command?username=admin&password=&command=stop-gateway&gateway=GSM+Modem+Gateway+1%3BGSM+Modem+Gateway+2

Restart (stop and start) one or more gateways

username	mandatory	The name of the user, the user must have administrator privileges
password	mandatory	The password of the user
command	mandatory	"restart-gateway"
gateway	optional	The name of the gateway or a list of gateway names, separated by semicolons. Leave empty to affect all gateways

example
http://localhost:9710/http/send-server-command?username=admin&password=&command=restart-gateway

Start one or more connectors

username	mandatory	The name of the user, the user must have administrator privileges
password	mandatory	The password of the user
command	mandatory	"start-connector"
connector	optional	The name of the connector or a list of connector names, separated by semicolons. Leave empty to affect all connectors

example
http://localhost:9710/http/send-server-command?username=admin&password=&command=start-connector&connector=Email+Connector

Stop one or more connectors

username	mandatory	The name of the user, the user must have administrator privileges
password	mandatory	The password of the user
command	mandatory	"stop-connector"
connector	optional	The name of the connector or a list of connector names, separated by semicolons. Leave empty to affect all connectors

example
http://localhost:9710/http/send-server-command?username=admin&password=&command=stop-connector&connector=Email+Connector%3BSQL+Connector

Restart (stop and start) one or more connectors

username	mandatory	The name of the user, the user must have administrator privileges
password	mandatory	The password of the user
command	mandatory	"restart-connector"
connector	optional	The name of the connector or a list of connector names, separated by semicolons. Leave empty to affect all connectors

example
http://localhost:9710/http/send-server-command?username=admin&password=&command=restart-connector&connector=Email+Connector%3BSQL+Connector

Reset send limit, only for GSM Modem Gateways

username	mandatory	The name of the user, the user must have administrator privileges
password	mandatory	The password of the user
command	mandatory	"reset-send-limit"
gateway	optional	The name of the GSM Modem Gateway or a list of gateway names, separated by semicolons. Leave empty to affect all GSM Modem Gateways

example
http://localhost:9710/http/send-server-command?username=admin&password=&command=reset-send-limit&gateway=GSM+Modem+Gateway

Pause sending messages with the selected gateway(s)

username	mandatory	The name of the user, the user must have administrator privileges
password	mandatory	The password of the user
command	mandatory	"pause-sending"
gateway	optional	The name of the gateway or a list of gateway names, separated by semicolons. Leave empty to affect all gateways

example
http://localhost:9710/http/send-server-command?username=admin&password=&command=pause-sending&gateway=GSM+Modem+Gateway

Resume sending messages with the selected gateway(s)

username	mandatory	The name of the user, the user must have administrator privileges
password	mandatory	The password of the user
command	mandatory	"resume-sending"
gateway	optional	The name of the gateway or a list of gateway names, separated by semicolons. Leave empty to affect all gateways

example
http://localhost:9710/http/send-server-command?username=admin&password=&command=resume-sending&gateway=GSM+Modem+Gateway

Get the server configuration XML data (all gateway and connector settings)

username	mandatory	The name of the user, the user must have administrator privileges
password	mandatory	The password of the user
command	mandatory	"get-server-configuration"

example
http://localhost:9710/http/send-server-command?username=admin&password=&command=get-server-configuration

Get the server settings XML data (all general settings of Diafaan SMS Server)

username	mandatory	The name of the user, the user must have administrator privileges
password	mandatory	The password of the user
command	mandatory	"get-server-settings"

example
http://localhost:9710/http/send-server-command?username=admin&password=&command=get-server-settings

Change one of the gateway properties (see the server configuration XML data for valid properties and values)

username	mandatory	The name of the user, the user must have administrator privileges
password	mandatory	The password of the user
command	mandatory	"set-gateway-property"
gateway	optional	The name of the gateway or a list of gateway names, separated by semicolons. Leave empty to affect all gateways
property	mandatory	The name of the gateway property
value	mandatory	The new value of the gateway property
reload	optional	Set to "0" to prevent restarting the gateway if you want to change more properties, by default the gateway is restarted immediately

example
http://localhost:9710/http/send-server-command?username=admin&password=&command=set-gateway-property&gateway=GSM+Modem+Gateway&property=Enabled&value=0

Change one of the connector properties (see the server configuration XML data for valid properties and values)

username	mandatory	The name of the user, the user must have administrator privileges
password	mandatory	The password of the user
command	mandatory	"set-connector-property"
connector	optional	The name of the connector or a list of connector names, separated by semicolons. Leave empty to affect all connectors
property	mandatory	The name of the connector property
value	mandatory	The new value of the connector property
reload	optional	Set to "0" to prevent restarting the connector if you want to change more properties, by default the connector is restarted immediately

example
http://localhost:9710/http/send-server-command?username=admin&password=&command=set-connector-property&connector=SQL+Connector&property=SMSOut.Enabled&value=0

Change one of the connector routing properties (see the server configuration XML data for valid properties and values)

username	mandatory	The name of the user, the user must have administrator privileges
password	mandatory	The password of the user
command	mandatory	"set-routing-property"
connector	optional	The name of the connector or a list of connector names, separated by semicolons. Leave empty to affect all connectors
gateway	optional	The name of the gateway or a list of gateway names, separated by semicolons. Leave empty to affect all gateways
property	mandatory	The name of the routing property
value	mandatory	The new value of the routing property
reload	optional	Set to "0" to prevent restarting the connector if you want to change more routing properties, by default the connector is restarted immediately

example
http://localhost:9710/http/send-server-command?username=admin&password=&command=set-routing-property&connector=SQL+Connector&gateway=GSM+Modem+Gateway&property=Receive&value=0

Change one of the settings properties (see the server settings XML data for valid properties and values)

username	mandatory	The name of the user, the user must have administrator privileges
password	mandatory	The password of the user
command	mandatory	"set-settings-property"
property	mandatory	The name of the settings property
value	mandatory	The new value of the settings property
reload	optional	Set to "0" to prevent reloading the new setting if you want to change more settings, by default the new setting is loaded immediately

example
http://localhost:9710/http/send-server-command?username=admin&password=&command=set-settings-property&property=LogEventsToEmail&value=1

HTTP API response error codes

300 Unknown error
301 No recipients found
302 Multiple to fields detected
303 Guest login not available
304 Authentication error
306 Unknown command or insufficient parameters
307 Operation not allowed
309 Message id not found
310 HTTP callback authentication not supported for this command
311 Message rejected
312 Web application user not found

4.3.1.2 Commandline client

Commandline client

Diafaan SMS Server includes a commandline client that makes use of the Web connector. It has the same functionality as the HTTP API but a different syntax.

Usage:
dmscmnd.exe "key1:value1" "key2:value2" "key3:value3" ...

keys:

host	mandatory	The host name and port number of the Web Connector. Use multiple host keys/value combination for redundancy support. e.g. "host:host1:9710" "host:host2:9710"
command	mandatory	HTTP page name, e.g. "command:send-message"

Commands

send-message

Send one or more messages.

keys

username	optional	The name of the user, if omitted the guest account is used

password	optional	The password of the user
to	required	One or more phone/pager numbers, separated by semicolons
contact-name	required	One or more contact names and/or groups from the user's contact list, separated by semicolons
from	optional	Sender id
message	optional	Message text
message-type	optional	Message type to indicate if the message is text, Unicode, flash etc.
gateway	optional	One or more gateway names, separated by semicolons, that may be used to send the message.
priority	optional	A higher number ensures that the message is placed higher in the message send queue and is sent before messages with a lower priority.

returns
OK: message-id
or
ERROR:error-code error-message

If multiple messages are sent this command returns one response line per message in the order at which the numbers are located in the to field.

example

```
dmscmnd.exe host:localhost:9710 command:send-message username:admin password: to:+44xxxxxxxx
message-type:sms.automatic "message:Message Text"
```

example with redundant host:

```
dmscmnd.exe host:host1:9710 host:host2:9710 command:send-message username:admin password: to:
+44xxxxxxxx message-type:sms.automatic "message:Message Text"
```

request-status-update

Get the status of a previously sent message.

keys

username	optional	The name of the user, if omitted the guest account is used
password	optional	The password of the user
message-id	mandatory	Message id of the previously sent message

returns
STATUS:status-code status-message
or
ERROR:error-code error-message

example

```
dmscmnd.exe host:localhost:9710 command:request-status-update username:admin password:
message-id:c3b0c20a-b63e-4bbf-919f-3d4251a8b9c0
```

request-server-status

Get the status of the Diafaan SMS Server and all the Gateways and Connectors statistics.

parameters

username	optional	The name of the user, if omitted the guest account is used
password	optional	The password of the user

returns
An XML file with the status and statistics of Diafaan SMS Server.

or
ERROR:error-code error-message

example
dmscmnd.exe host:localhost:9710 command:request-server-status username:admin password:

request-received-messages

Get a list of received messages in the inbox.

keys

username	optional	The name of the user, if omitted the guest account is used
password	optional	The password of the user
limit	optional	The maximum number of messages returned
remove	optional	Set to "1" to remove the returned messages from the database
order	optional	Set to "newest" to get newest messages first, otherwise oldest messages are displayed first.

returns
An XML file with a list of the requested messages.
or
ERROR:error-code error-message

example
dmscmnd.exe host:localhost:9710 command:request-received-messages username:admin password: limit:10 order:newest

4.3.2 Email connector

Send SMS properties

Check **Start SMTP Server** to start the SMTP server on the requested port. Check **Get email from a POP server** when you want to get the email messages from an external POP server instead.

SMTP Server

By default, the new SMTP Server will start on port 25. You can change the SMTP port number if it interferes with an existing SMTP server. It is also possible to change the message template, the parts of the email messages that are converted to SMS.
The to address of the mail message is converted into the number and an optional message type, e.g.:

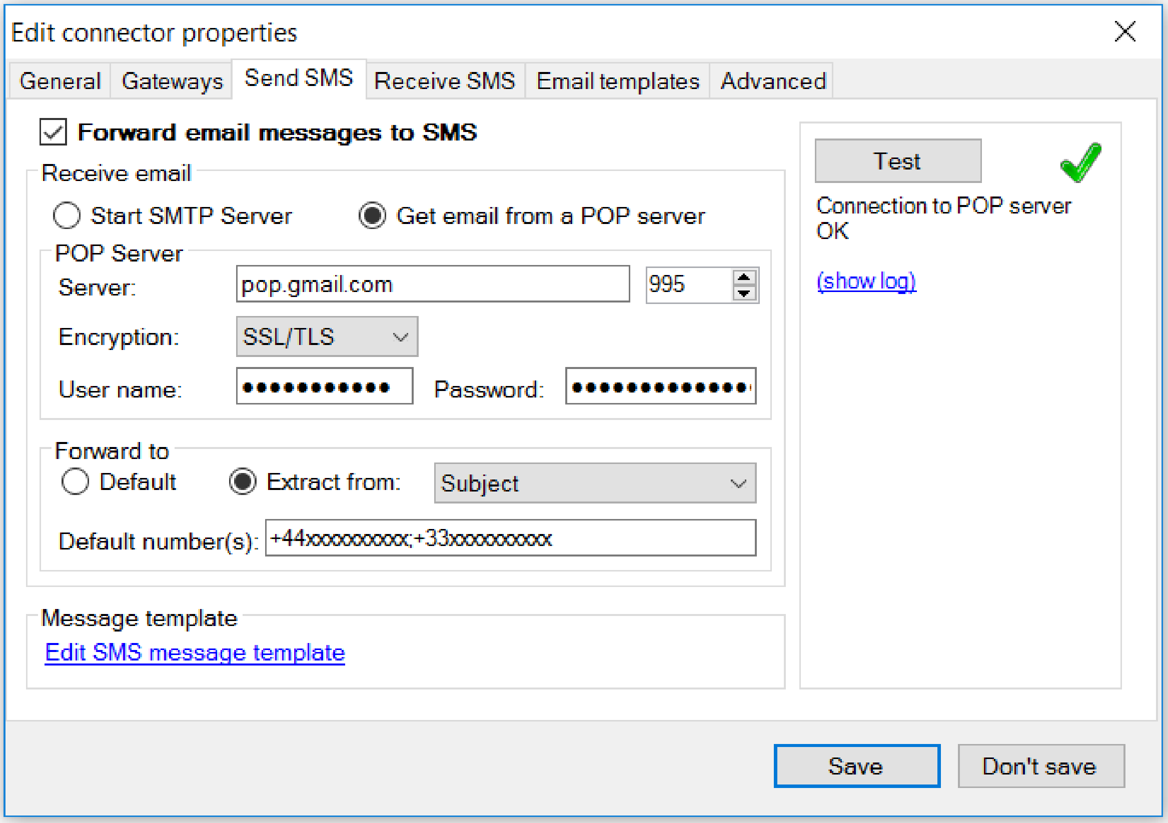
+44xxxxxxxxx@sms.com	is sent to number +44xxxxxxxxx.
p44xxxxxxxxx@sms.com	is also sent to number +44xxxxxxxxx.
+44xxxxxxxxx.sms.text@sms.com	is sent to number +44xxxxxxxxx with message type sms.text.

Some email environments might have trouble with a plus sign in the email address, to overcome this problem you can use a p character instead.

POP3 Client

The Email connector can retrieve email from a POP box and convert the mail to an SMS message. After installing the connector, the email is forwarded to the GSM number (list) you entered. You can change the message template and add the option to extract the GSM number (list) from the mail subject or body. If **Extract from Subject**, **Body, To address** or **From address** is enabled, the Email connector will try to find the GSM numbers in the email subject, body or the email address. If multiple numbers are detected in the subject or mail body the message will be sent to all those numbers. If no number is found the default number (list) is used. A number can only be detected in the subject or message body if it contains 4-20 digits with an optional leading '+' (e.g. +4400000000).

All retrieved messages from the POP server are deleted after retrieval.



☑ Forward email messages to SMS

Receive email

○ Start SMTP Server ● Get email from a POP server

POP Server

Server: pop.gmail.com 995

Encryption: SSL/TLS

User name: ●●●●●●●●●●● Password: ●●●●●●●●●●●●●

Forward to

○ Default ● Extract from: Subject

Default number(s): +44xxxxxxxxxx;+33xxxxxxxxxx

Message template

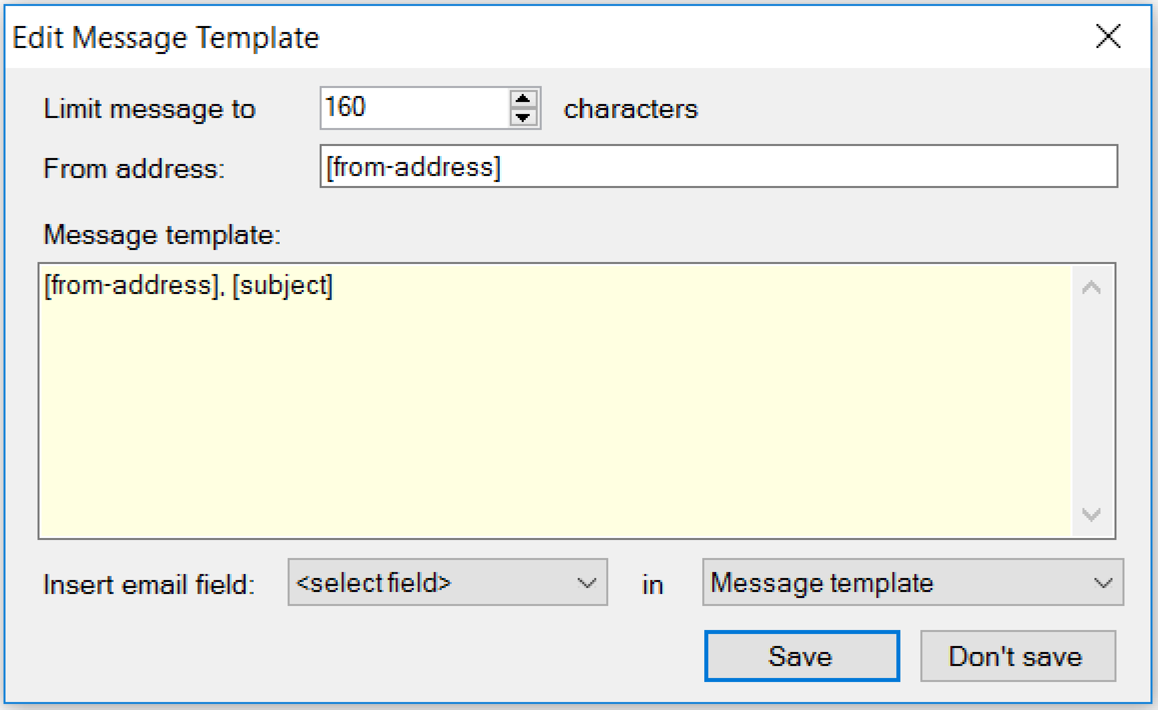
Edit SMS message template

Connection to POP server OK

(show log)

Edit message template

The message template defines what part(s) of the email message is forwarded to the SMS message. You can insert fields from the mail message in the template and limit the amount of characters of the message.



Edit Message Template

Limit message to 160 characters

From address: [from-address]

Message template:

[from-address]. [subject]

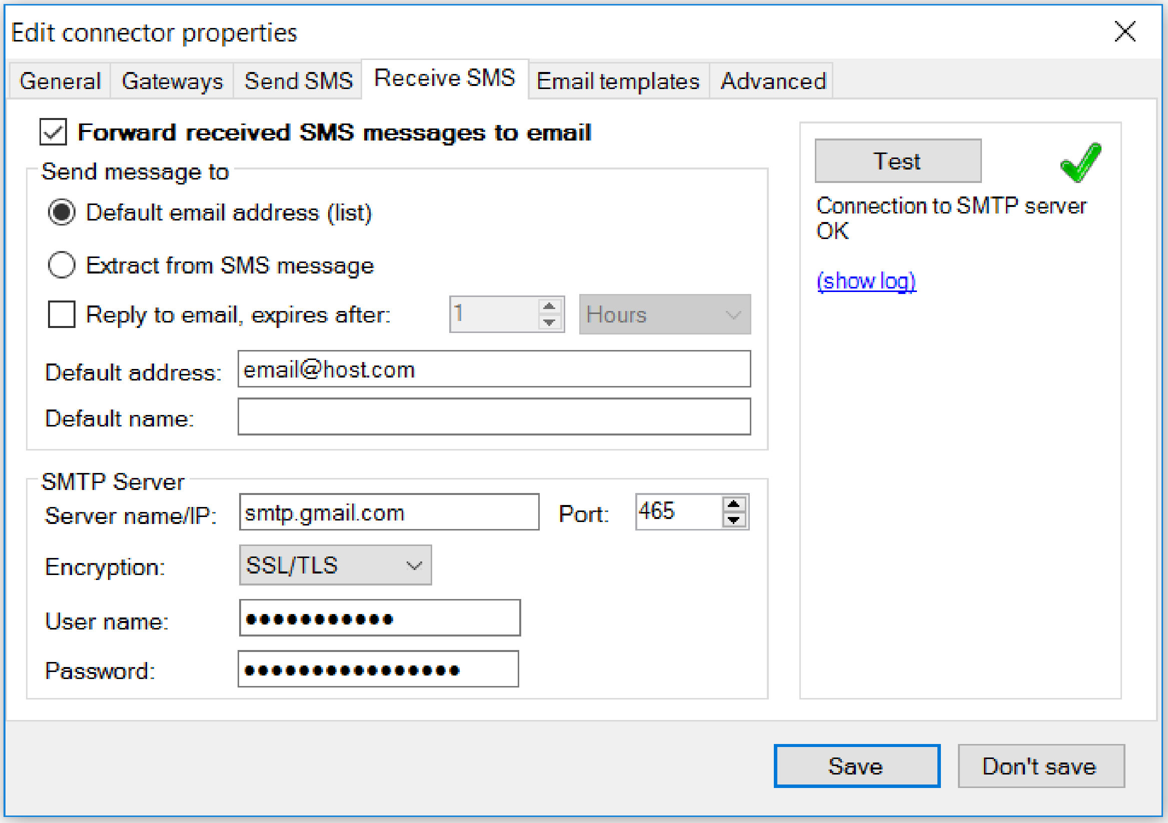
Insert email field: <select field> in Message template

Receive SMS properties

Enable the **Forward received SMS messages to email** option to convert received SMS messages to one or more email messages. If **Extract from SMS message** is enabled the Email connector tries to find a valid email address in the SMS message text. If no email address is found the default address (list) is used.

When **Reply to email** is selected the email connector looks if there was a previous email message forwarded to the mobile phone number. If there is a GSM number match, the message is sent as an email reply to the last sender of the original email to that specific phone number. If there is no match, the message is sent to the default address (list).
If **Reply to email** is selected in combination with the **Extract from SMS message** option, the procedure to forward the SMS message to email is as follows:

- If an email address is found in the SMS message, the email is sent to this email address.
- If no email address is found in the SMS message and there is a GSM number match from a previous email message, the email is sent as a reply to the last sender of the original email to that specific phone number.
- If no email address is found in the SMS message and there is no GSM number match, the email is sent to the default address list.



☑ Forward received SMS messages to email

Send message to

● Default email address (list)

○ Extract from SMS message

□ Reply to email, expires after: 1 Hours

Default address: email@host.com

Default name:

SMTP Server

Server name/IP: smtp.gmail.com Port: 465

Encryption: SSL/TLS

User name: ●●●●●●●●●●●●

Password: ●●●●●●●●●●●●●●●●

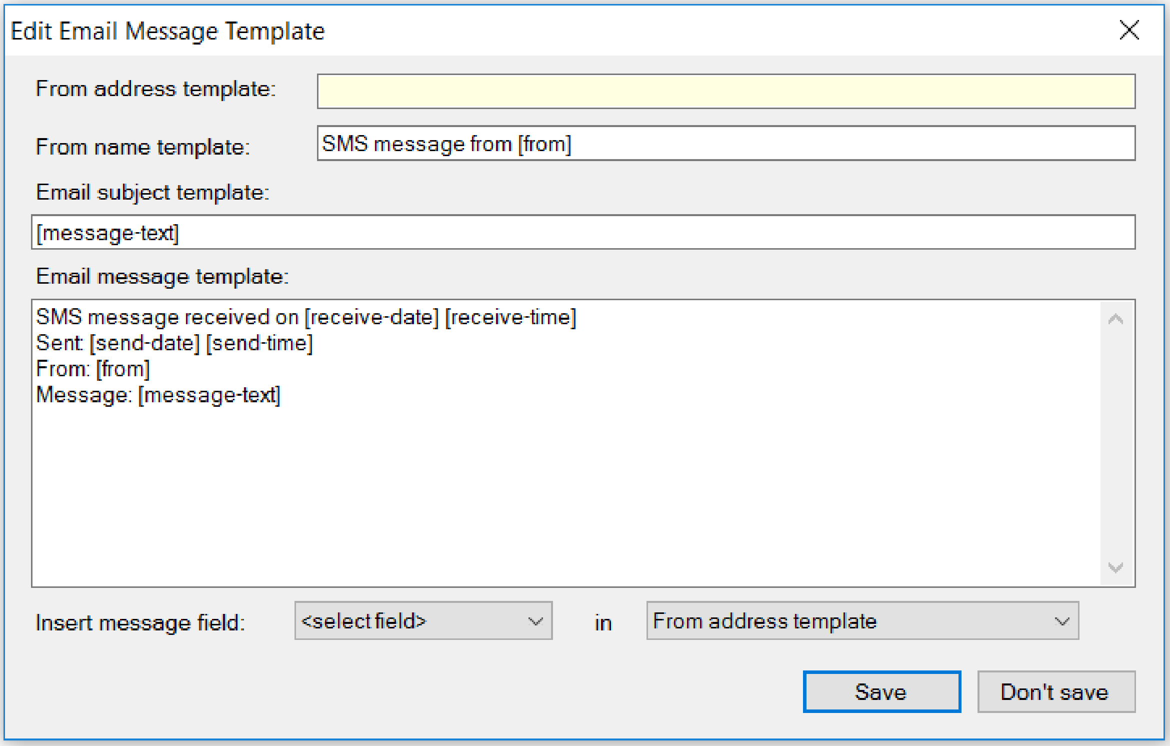
Connection to SMTP server OK

(show log)

Email templates

Edit email template for received SMS messages

The email template defines the layout of the email message. For each email field you can use one or more parts of the received SMS message.



Edit Email Message Template

From address template:

From name template: SMS message from [from]

Email subject template: [message-text]

Email message template:

SMS message received on [receive-date] [receive-time]
Sent: [send-date] [send-time]
From: [from]
Message: [message-text]

Insert message field: <select field> in From address template

Edit email template for SMS reply to email

This template is only used if the email is a reply to a previously sent email message. It supports a number of fields from the original email message, like the **To name**, **To address** and the email **Subject**.

Check the **Reply when send SMS failed** if you want to send a reply when the SMS message could not be sent. If you also want to send a reply when the SMS messages was sent you can check **Reply when SMS is sent**.

Using OAUTH2 authentication for Gmail, Exchange or Outlook (experimental)

The Email Connector offers experimental support for XOAUTH2 authentication in the POP and SMTP client for Google Gmail and Microsoft Outlook and Exchange and possibly other POP and SMTP services. To use XOAUTH2 authentication, you first have to sign up to Google Cloud Platform for Gmail or Azure Active Directory for Outlook to obtain an app dependent Oauth2 Client ID and (for GMail) a Client Secret. The Client ID and Client Secret must be filled in in the OAuth2ClientID and OAuth2ClientSecret advanced settings of the Email Connector. Other services might also work, but then the other Oauth2 options in the advanced settings must also be filled in, for GMail and Outlook these additional settings can be left empty.

With a valid Oauth2 Client ID and, for GMail, a Client Secret filled in, you can select Oauth2 as the **Authentication** option in the Send SMS (for POP) and Receive SMS (for SMTP) options. There are differences between the Oauth2 process of Gmail and Outlook so the right service must be selected in the **Authentication** type.

If all settings are correct, you can use the **Authenticate** button to start the OAuth2 authentication process in a web browser. If the authentication process succeeds, the red cross icon changes to a green check mark and you can use the 'Test' button to test the POP server or SMTP server with the new settings and authentication.

Advanced properties

LogCommunicationToFile
Log all internet communication to file. Use this option only if you encounter problems with the SMTP or POP3 connection to or from the Email connector. If enabled, the status windows of the Email connector provides a link to the location on you hard disk where the log files are stored.

MaximumBatchSize
The maximum number of messages this connector will process in one batch. If this value is higher than one and a number of messages are sent at the same time, the connector loads multiple messages concurrently. This increases the speed in which the messages are processed.

SendPriority
Message send priority, a higher number ensures that messages are placed higher in the message send queue and are sent before messages with a lower priority.

NumberTemplate
The number template can be used to translate the number from one format to another.

For more information see Use number templates.

EnableHTTPCallbacks
Enable or disable HTTP callbacks for this gateway. This option is only available when the HTTP callback option is enabled in the general options.

NumberWhiteList
List of allowed destination numbers (to address) for outgoing SMS messages, messages to other numbers are blocked. When the list does not have any items, messages to all numbers are allowed.

For more information about the white list and black list format see: Use white lists and black lists.

OAuth2ClientId
OAuth2 application client ID.

OAuth2ClientSecret
OAuth2 application client secret (when applicable).

OAuth2EndpointAuthorization
OAuth2 authorization endpoint (https://...), leave empty for the default endpoint.

OAuth2EndpointTokenRequest
OAuth2 token request endpoint (https://...), leave empty for the default endpoint.

OAuth2Scope
OAuth2 authentication scope, leave empty for the default POP and SMTP scope.

EmailWhiteList
List of allowed source email addresses or domain names, messages from other email addresses or domains are blocked. Leave empty to allow all sender addresses.

For more information about the white list and black list format see: Use white lists and black lists.

POP3CheckMailInterval
The interval time, in minutes, to check the POP server for new mail.

SMTPServerIPBinding
local IP address where the SMTP Server should bind to. If this value is empty the SMTP server binds to all available IP addresses.

SMTPServerIPWhiteList
List of allowed IP numbers, messages from other IP numbers are blocked. Leave empty to allow all IP numbers.

For more information about the white list and black list format see: Use white lists and black lists.

SMTPServerMaximumConnections
The maximum number of concurrent connections to the SMTP server.

SMTPServerMaximumTransactions
The maximum number of email transactions (messages) per connection to the SMTP server.

SMTPServerSSLCertificate
The full path and filename of the X509 (OpenSSL) certificate required to enable secure connections to the SMTP server.

SMTPServerCertificatePassword
The password of the X509 certificate used for SSL (https://) connections.

SMTPServerStopOnNoActiveGateways
Stop the SMTP server when there are no active gateways available to send the messages. This option can be used to let the SMTP client know that this route is (temporarily) disabled and that it must use another way to send the messages.

SMTPServerUserName
The user name required to log in to the SMTP server. Leave empty to allow anonymous user logon.

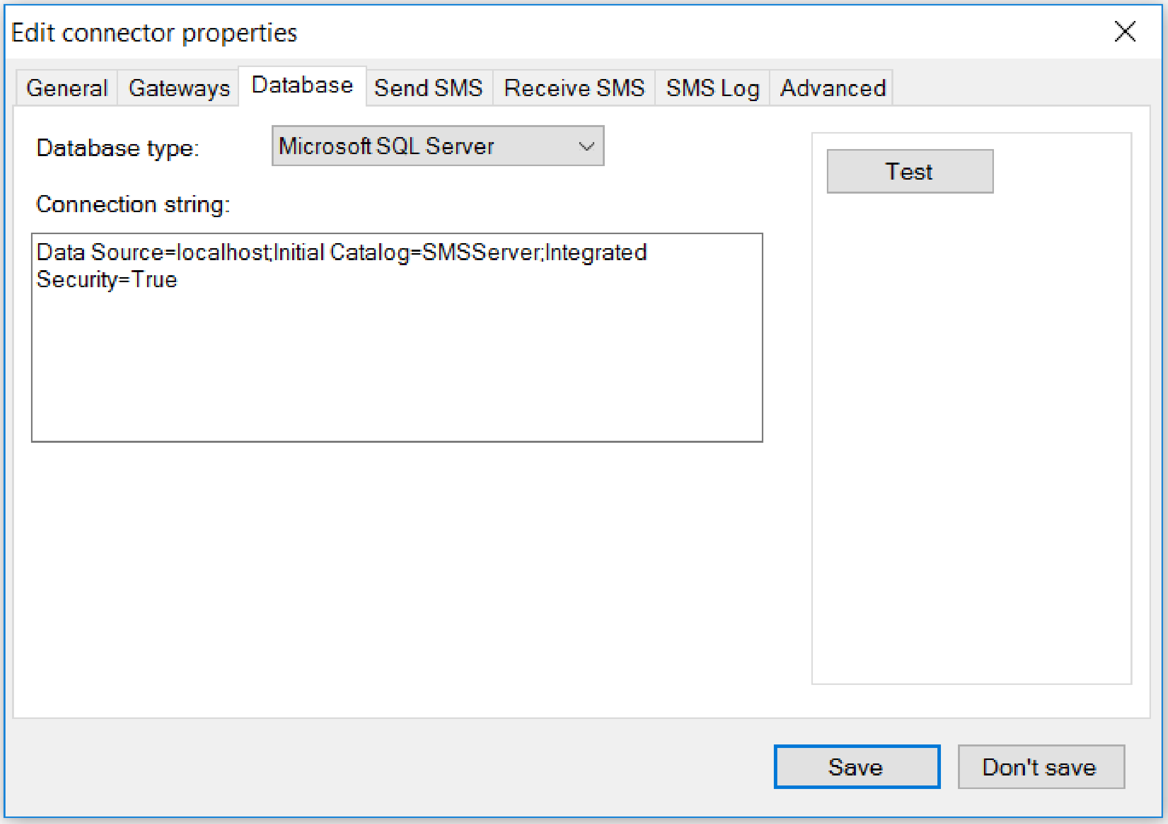
SMTPServerPassword
Password to log on to the SMTP server, only used when the SMTPServerUserName field is not empty.

SMTPClientSendAttempts
Maximum number of times the SMTP Client tries to deliver a received message.
2nd attempt after at least 30 seconds
3rd attempt after at least 5 minutes
4th attempt after at least 30 minutes
5th attempt after at least 1 hour
6th and further attempts after at least 12 hour

4.3.3 SQL connector

Database properties

Select the database driver type and connection string to the database you want to use. Diafaan SMS Server supports three database driver types, **Microsoft SQL Server (Express)**, **ODBC** and **OLE DB**.



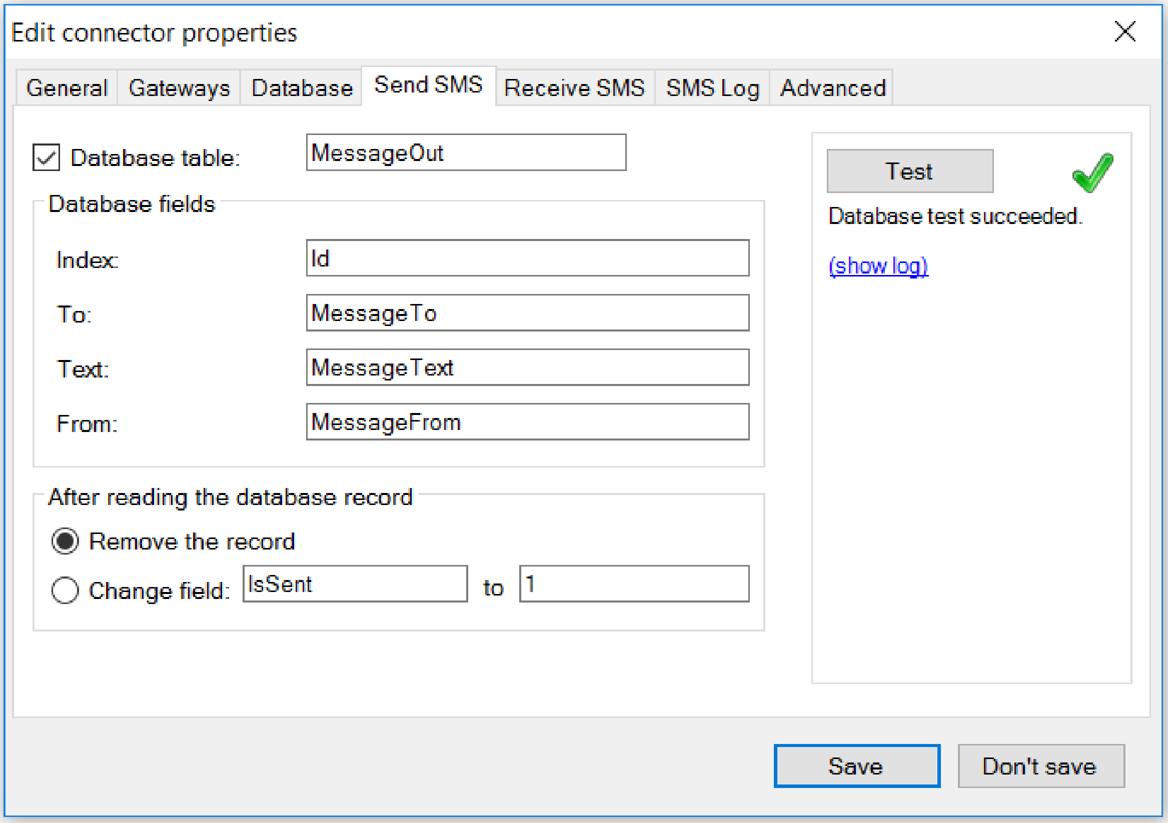
Database type: Microsoft SQL Server

Connection string:

```
Data Source=localhost;Initial Catalog=SMSServer;Integrated Security=True
```

Send SMS properties

In the Send SMS tab page you can enter the table name and the basic database field names of the database table that is polled for new messages to send. After the database records are read in, the records are either removed from the database or one of the database fields is set to a fixed value. Records where this value is set are not read anymore in successive database pollings.
If large numbers of messages will be sent it is better to remove the records from the database, leaving a large number of records in the database might have an impact on the database performance.


Edit connector properties

☑ Database table: MessageOut

Database fields

Index: Id

To: MessageTo

Text: MessageText

From: MessageFrom

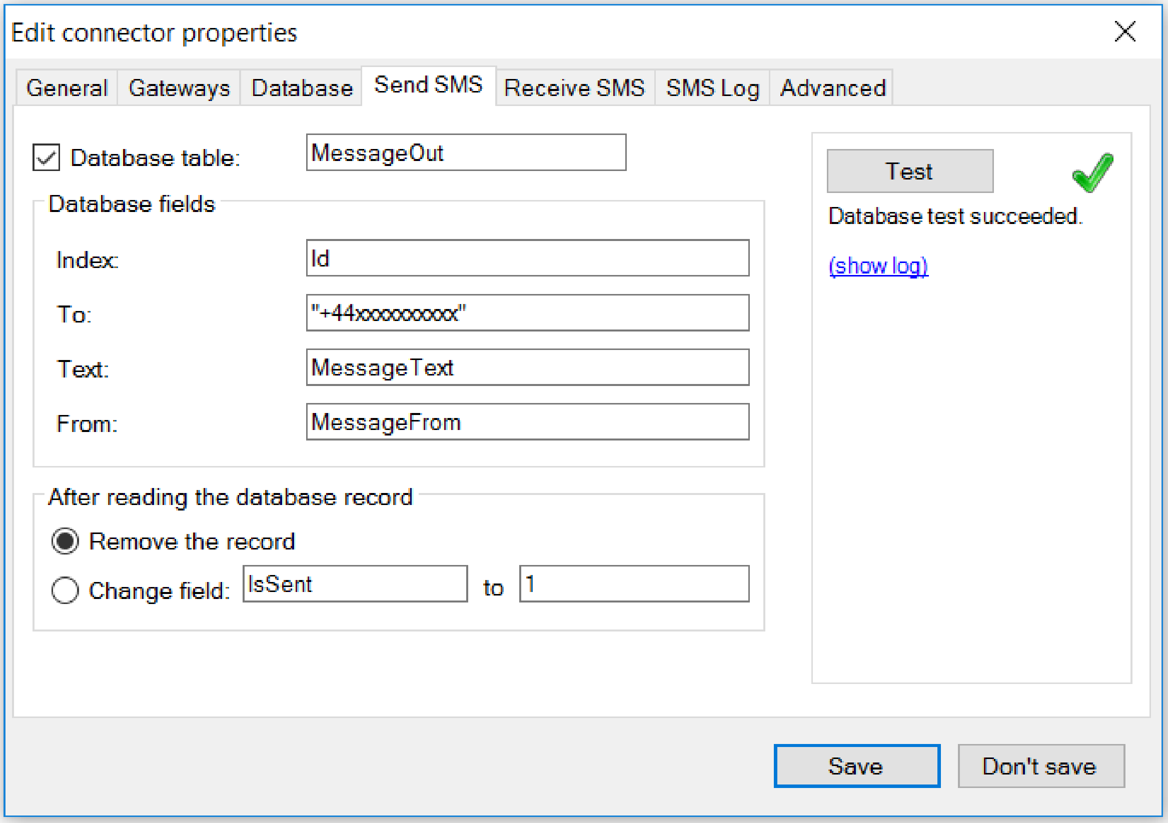
After reading the database record

● Remove the record

○ Change field: IsSent to 1

Database test succeeded.

In the Send SMS tab page you can also use double quotes to indicate fixed values. In the example below the **To** part of the SMS message is not retrieved from the database but all messages are sent to number +44xxxxxxxxxx. This enables you to use database tables where no GSM number is available. The only required database field is the Index, all other fields are optional.



Edit connector properties

☑ Database table: MessageOut

Database fields

Index: Id

To: "+44xxxxxxxxxx"

Text: MessageText

From: MessageFrom

After reading the database record

● Remove the record

○ Change field: IsSent to 1

Database test succeeded.

Scheduled messages

The Advanced settings offers support for an **SMSOutScheduled** date/time database field. If this database field is used individual messages are not read from the database before the specified date and time.

Receive SMS properties

Use the values in the Receive SMS tab page to specify the database table name and basic database field names where received messages are stored. More fields from the received message can be specified in the Advanced properties.

SMS Log properties

The SMS Log database table is used to store the result of message send attempts. Specify the database table name and basic database field names where the message logs are stored. More fields can be specified in the Advanced properties.

The **Message Status Code** field stores the message result. Status codes between 200 and 299 indicate that the message is sent successfully. Status codes between 300 and 399 means that the message failed.

status codes

200	Message sent and accepted by the gateway.

201	Message received on the GSM phone.
300	Message failed or rejected by the gateway.
301	Status error received after the message was accepted by the gateway

The message status can be updated, this means that a message with status code 200 (accepted by the gateway) can change to status code 301 if the GSM network cannot deliver the message or to status code 201 when the message is delivered on the phone.

Advanced properties

LogCommunicationToFile
Log all SQL transactions from this gateway to a file.

MaximumBatchSize
The maximum number of database records this connector will process in one batch. If this value is higher than one and a number of messages are sent at the same time, the connector loads multiple messages concurrently. This increases the speed in which the messages are processed considerably.

SendPriority
Default message send priority, set to 0 to allow a custom priority for individual messages. A higher number ensures that messages are placed higher in the message send queue and are sent before messages with a lower priority.

NumberTemplate
The number template can be used to translate the number from one format to another.

For more information see Use number templates.

NumberWhiteList
List of allowed destination numbers (to address) for outgoing SMS messages, messages to other numbers are blocked. When the list does not have any items, messages to all numbers are allowed.

For more information about the white list and black list format see: Use white lists and black lists.

EnableHTTPCallbacks
Enable or disable HTTP callbacks for this gateway. This option is only available when the HTTP callback option is enabled in the general options.

DatabaseCharacterset
Specifies the most extensive character set the database supports, the best character set for most modern databases is 'Unicode'. The UCS2 character set can solve a compatibility problem with some Unicode characters (like smileys) on some versions of the MySQL database. ASCII can be used for older databases that do not support Unicode at all. Characters in received messages that are not in the selected character set are converted to question marks (?).

DatabaseCheckInterval
The time interval in seconds between two queries of the Send SMS database table.

DatabaseUpdateInterval
Minimum interval time for updates of the MessageIn and MessageLog database tables. Increase this value to increase the number of messages that are written or updated to the database per transaction, this can reduce strain on the database in high-traffic systems.

HandleSchedulingInDatabase

Set to 'True' (default) to keep scheduled messages in the database until the schedule time. Set to 'False' to handle scheduled messages in the internal send queue of Diafaan SMS Server.

LockMessageOutTable
Lock the MessageOut table for reading when the polling transaction is in progress. This option can be set when multiple SQL Connectors or multiple installations of Diafaan SMS Server use the same MessageOut table to import messages. With some database configuration this setting prevents that messages are loaded multiple times by different SQL Connectors.

MessageInRetryOnError
If set to true, messages that cannot be written to the MessageIn table because of a database error are kept in memory until a follow up database transaction succeeds.

MessageLogRetryOnError
If set to true, messages that cannot be written to, or updated in, the MessageLog table because of a database error are kept in memory until a follow up database transaction succeeds.

SendQueuePauseThreshold
Pause loading new messages from the MessageOut table when the send queue exceeds this threshold. Set to 0 to disable the threshold.

SQLFlavor
The type of database that is used, changing this value causes small changes in the SQL query strings to optimize performance.

database types

AnsiSQL92	Standard ANSI SQL
Microsoft	Implements the TOP keyword
Oracle	Implements the ROWNUM keyword
MySQL	Implements the LIMIT keyword

Database field names
Additional database field names can be entered to query or store more parts of the sent or received message fields.

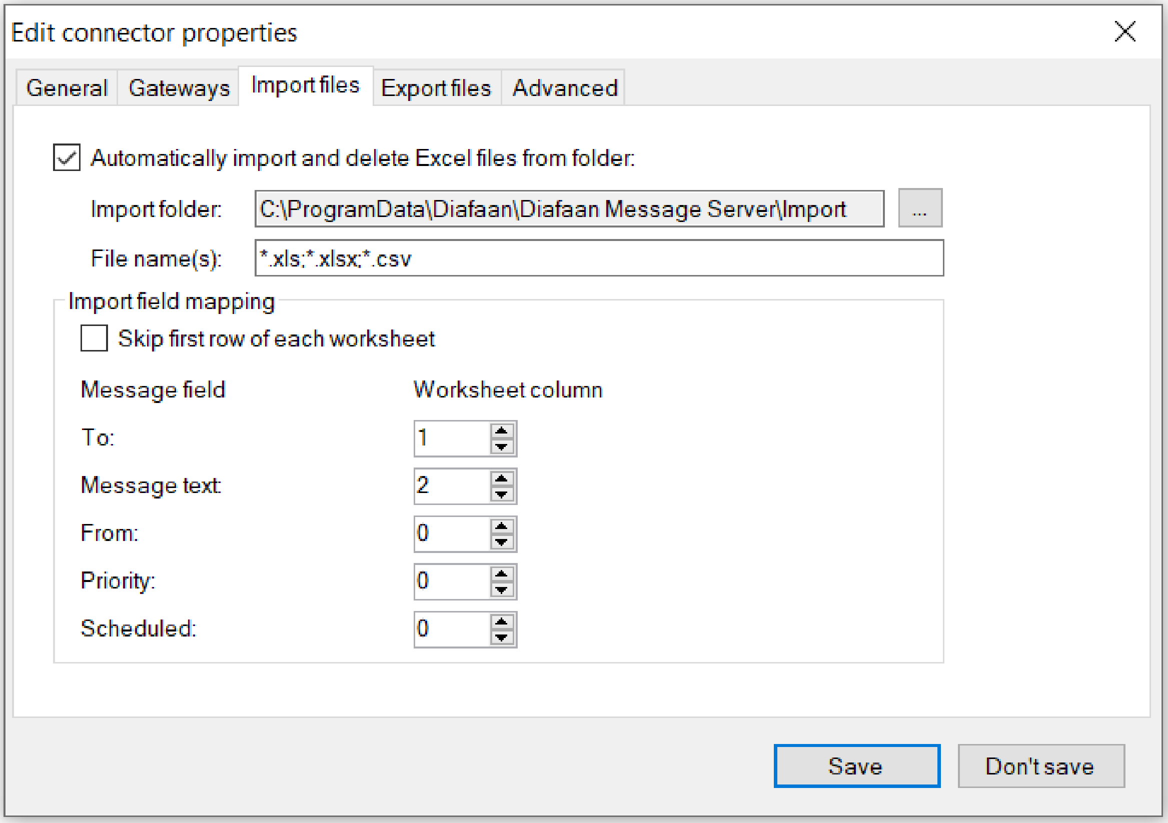
4.3.4 Excel connector

Import messages from spreadsheet files

Messages can be automatically imported from one or more spreadsheet files in a specific folder. After importing the messages from the spreadsheet file, the file is automatically removed.

The file name(s) setting can contain a list of file names that are imported, separated by semicolons. The file names may contain wildcards (* and ?). The Excel Connector can import three file types, the Excel file formats .XLS and .XLSX and the generic .CSV format.

Each row in each worksheet of the spreadsheet file is imported as one message (or multiple messages if the To field contains multiple numbers separated by comma's) and the message fields are filled in according to the **import field mapping** settings. Additional messages fields can be mapped to worksheet columns in the advanced settings of the Excel Connector.



☑ Automatically import and delete Excel files from folder:

Import folder: C:\ProgramData\Diafaan\Diafaan Message Server\Import

File name(s): *.xls;*.xlsx;*.csv

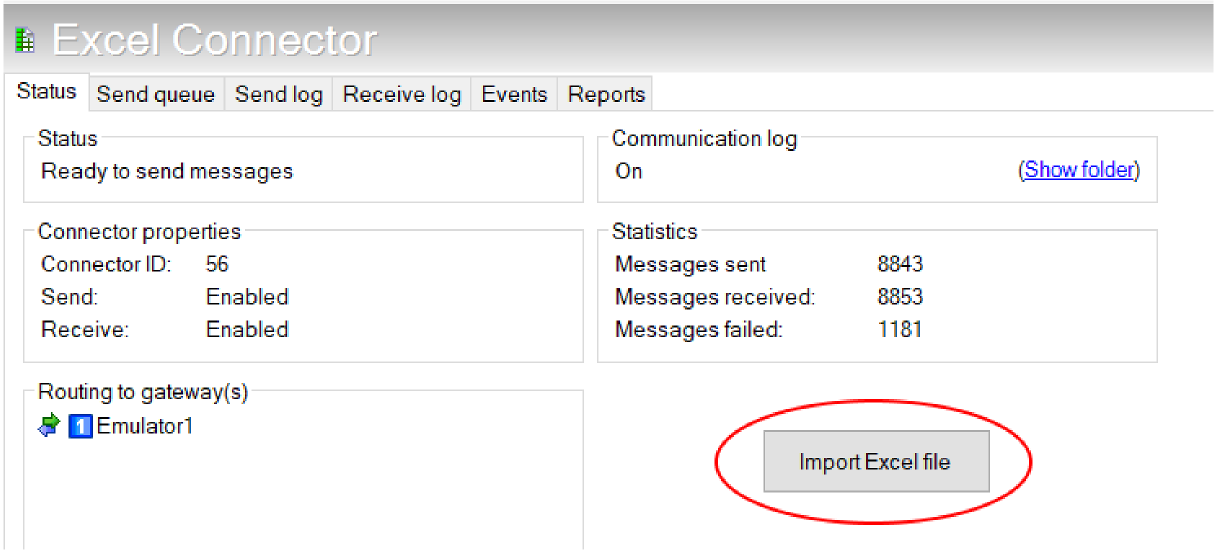
Import field mapping

□ Skip first row of each worksheet

Message field	Worksheet column
To:	1
Message text:	2
From:	0
Priority:	0
Scheduled:	0

Import messages manually from a spreadsheet file

The 'Import Excel file' button in the status window of the Excel Connector can be used to manually import messages from a single spreadsheet file. Unlike automatic imports the imported file is not removed after it is read.



Excel Connector

Status
Ready to send messages

Communication log
On (Show folder)

Connector properties

Connector ID:	56
Send:	Enabled
Receive:	Enabled

Statistics

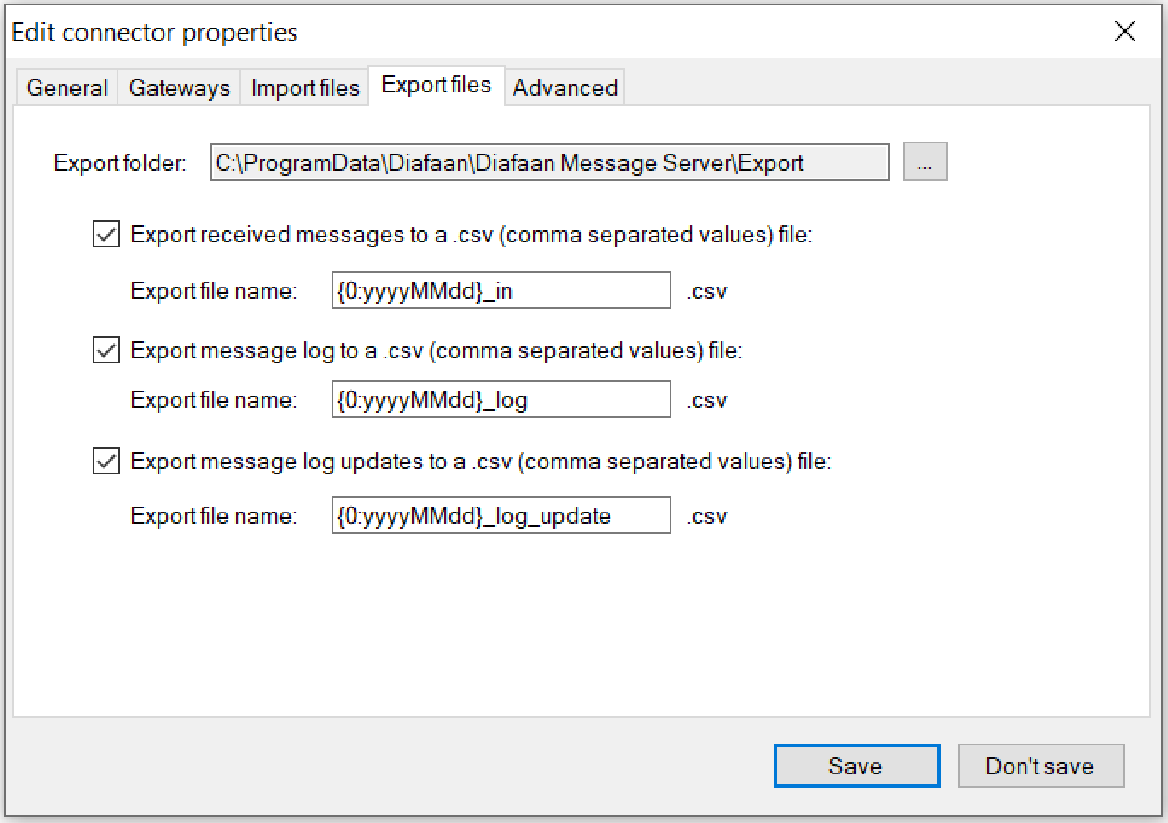
Messages sent	8843
Messages received:	8853
Messages failed:	1181

Routing to gateway(s)
1 Emulator1

Export received messages and message logs to .CSV files

The Excel Connector can export received SMS messages, send log messages and send log updates (delivery reports and delivery failures) to .CSV files. You can set the export folder where files are saved and the names of the three export files. The file names can contain date and time information in the form of **{0:yyyyMMdd}** so that the messages are written to different files dependent of the date. With the default names a new file is made each day. If you want to create a new file each month you can use **{0:yyyyMM}** instead. The message fields that are exported can be changed in the advanced settings.

By default, the Excel Connector only exports the send log messages that are sent from the Excel Connector itself. There is an option in the general setting to log all messages sent with Diafaan SMS Server to a specific Excel Connector or SQL Connector.



Export folder: C:\ProgramData\Diafaan\Diafaan Message Server\Export

☑ Export received messages to a .csv (comma separated values) file:

Export file name: {0:yyyyMMdd}_in .csv

☑ Export message log to a .csv (comma separated values) file:

Export file name: {0:yyyyMMdd}_log .csv

☑ Export message log updates to a .csv (comma separated values) file:

Export file name: {0:yyyyMMdd}_log_update .csv

Advanced properties

LogCommunicationToFile
Log all import and export actions from this connector to a file.

MaximumBatchSize
The maximum number of database records this connector will process in one batch. If this value is higher than one and a number of messages are sent at the same time, the connector loads multiple messages concurrently.

SendPriority
Default message send priority. A higher number ensures that messages are placed higher in the message send queue and are sent before messages with a lower priority.

NumberTemplate
The number template can be used to translate the number from one format to another.

For more information see Use number templates.

NumberWhiteList
List of allowed destination numbers (to address) for outgoing SMS messages, messages to other numbers are blocked. When the list does not have any items, messages to all numbers are allowed.

For more information about the white list and black list format see: Use white lists and black lists.

EnableHTTPCallbacks

Enable or disable HTTP callbacks for this connector. This option is only available when the HTTP callback option is enabled in the general options.

ExportMessageInFieldList
The list of message fields of received messages that are written to the .CSV file.

ExportMessageLogFieldList
The list of message fields of the send log that are written to the .CSV file.

ExportMessageLogUpdateFieldList
The list of message fields of the send log updates (delivery reports and delivery failures) that are written to the .CSV file.

FieldMapGateway
Worksheet column number in the import files that contains the (list of) gateway(s) that can send the messages, set to 0 to disable.

FieldMapMessageGUID
Worksheet column number in the import files that contains the message GUID, set to 0 to disable.

FieldMapMessageInfo
Worksheet column number in the import files that contains the message info, set to 0 to disable.

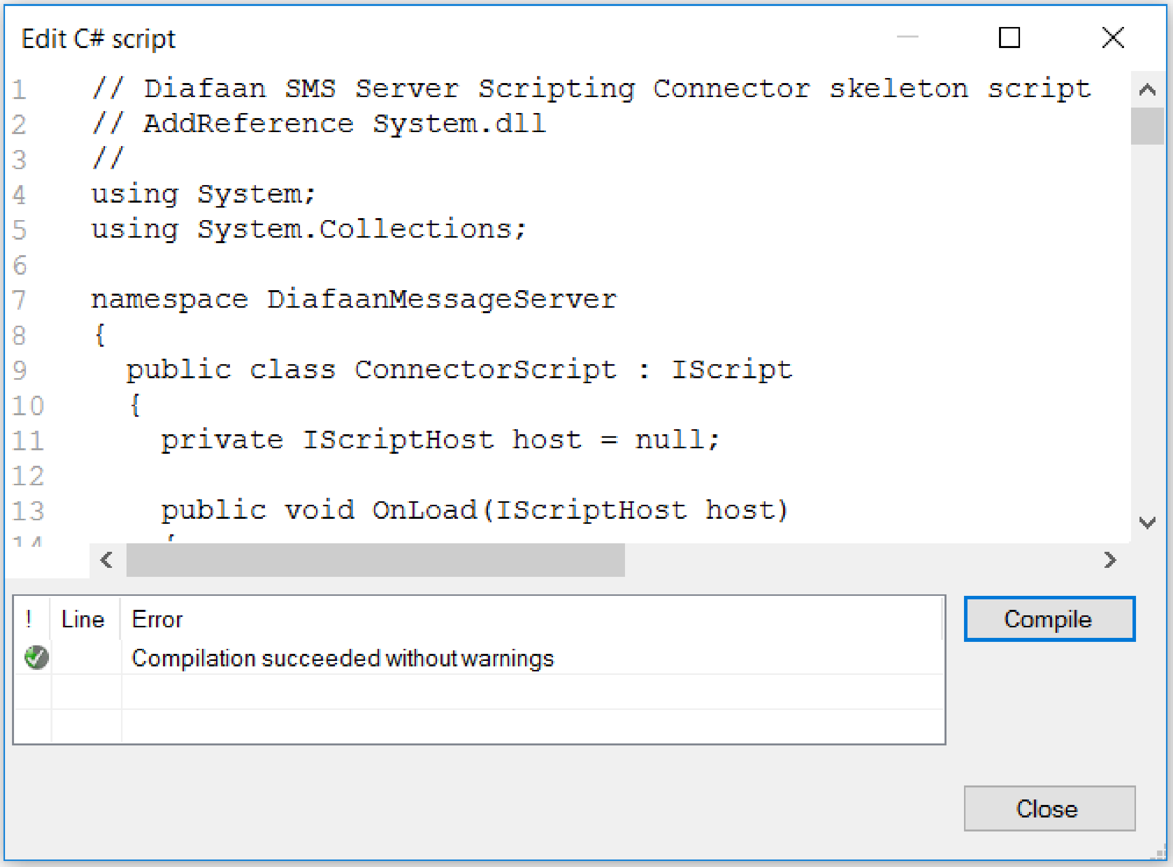
FieldMapMessageType
Worksheet column number in the import files that contains the message type, set to 0 to disable.

ImportFilesCheckInterval
Interval time to check the import folder for new Excel files (in seconds).

4.3.5 Scripting connector

Edit connector script

Click on the **Edit C# Script** or **Edit Visual Basic .Net Script** link to edit the script. Use the **Compile** button to check if the script has syntax errors.



```
1   // Diafaan SMS Server Scripting Connector skeleton script
2   // AddReference System.dll
3   //
4   using System;
5   using System.Collections;
6
7   namespace DiafaanMessageServer
8   {
9     public class ConnectorScript : IScript
10    {
11      private IScriptHost host = null;
12
13      public void OnLoad(IScriptHost host)
```

!	Line	Error
✓		Compilation succeeded without warnings

If the script uses functions that are not in the System.dll assembly, a reference to the required assembly DLL must be added in the comment section at the top of the script. For instance to add a reference to the System.Data.dll assembly in a C# script you can insert:

// AddReference System.Data.dll

At line 3 in the script.

The reference can be the full path name of the DLL file if it is not a standard .NET assembly.

Advanced properties

LogCommunicationToFile

Log all trace log communication to file. You can use this option in combination with the PostTraceLog script function to debug your script. If enabled, the status windows of the Scripting connector provides a link to the location on the hard disk where the file is stored.

EnableHTTPCallbacks
Enable or disable HTTP callbacks for this gateway. This option is only available when the HTTP callback option is enabled in the general options.

NumberWhiteList
List of allowed destination numbers (to address) for outgoing SMS messages, messages to other numbers are blocked. When the list does not have any items, messages to all numbers are allowed.

For more information about the white list and black list format see: Use white lists and black lists.

Connector skeleton script

The connector skeleton script that is installed after adding the gateway makes it easier to handle events and provides a number of methods you can use to send messages and add log entries.

Events

OnMessageReceived is called when an SMS message is received.

OnSendMessageResult is called when a message result is received from a previously sent message.

OnSendMessageResultUpdate is called when a message result update was received.

Methods

PostSendMessage initiate sending an SMS message.

PostEventLog adds a line to the event log.

PostTraceLog Adds a line to the communication file.

4.3.6 SMPP connector

SMPP server properties

SMPP server port

By default, a new SMPP server will start on port 2775. You can change the SMPP server port number if it interferes with an existing IP server on the computer.

System ID

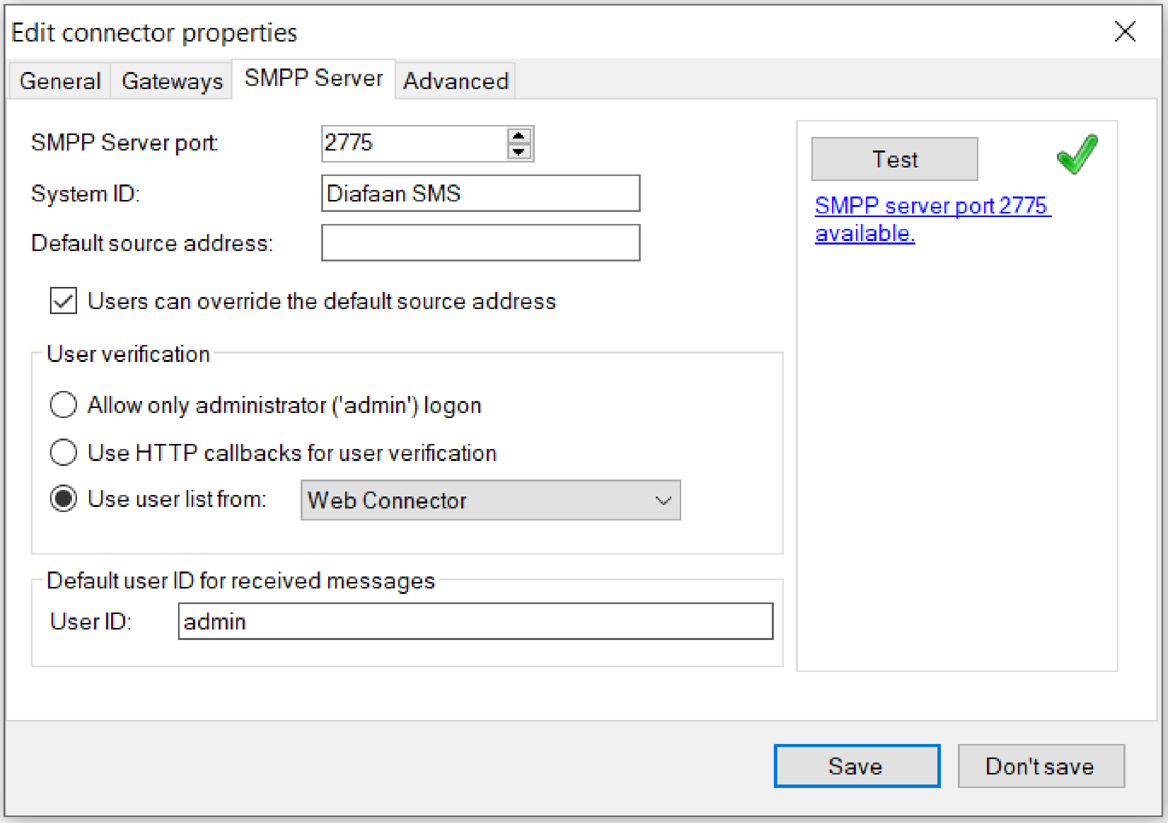
The system ID is the name of the SMPP server that is sent to the SMPP clients after a successful bind request from the client.

User verification

The SMPP Connector supports three user verification options to check the user name and password of the SMPP clients. A basic 'admin' user verification where only the administrator can log in, verification using HTTP callback requests and verification with the user list of one of the Web Connectors in Diafaan SMS Server. The HTTP callback user verification option is grayed out when HTTP callbacks are not yet set up in the general options. The Web Connector user verification option is grayed out when there is no Web Connector set up in Diafaan SMS Server.

Default user ID

Received messages can be assigned a default user ID, this is the user ID of the SMPP client where the received messages are delivered. Diafaan SMS Server can assign a specific user ID for each individual received message using the HTTP callback functionality.



SMPP Server port: 2775

System ID: Diafaan SMS

Default source address:

☑ Users can override the default source address

User verification

○ Allow only administrator ('admin') logon

○ Use HTTP callbacks for user verification

● Use user list from: Web Connector

Default user ID for received messages

User ID: admin

✓ SMPP server port 2775 available.

User specific options

Some of the advanced options in the SMPP Connector can be changed for each individual client connection with the HTTP callback user verification option. The following advanced properties can be overridden:
- CharacterEncoding
- ConnectionTimeout
- MaxMessagesPerMinute
- MaxUserIdConnections
- MessageIdFormat
- MessageInHandling
- MessageOutHandling
- Use16BitConcatenationId

Advanced properties

LogCommunicationToFile
Log all SMPP communication to this connector to a file.

MaximumBatchSize
The maximum number of records this connector will process in one batch. If this value is higher than one and a number of messages are sent at the same time, the connector loads multiple messages concurrently. This increases the speed in which the messages are processed considerably.

SendPriority
Default message send priority. A higher number ensures that messages are placed higher in the message send queue and are sent before messages with a lower priority.

NumberTemplate
The number template can be used to translate the number from one format to another.

For more information see Use number templates.

NumberWhiteList
List of allowed destination numbers (to address) for outgoing SMS messages, messages to other numbers are blocked. When the list does not have any items, messages to all numbers are allowed.

For more information about the white list and black list format see: Use white lists and black lists.

EnableHTTPCallbacks
Enable or disable HTTP callbacks for this connector. This option is only available when the HTTP callback option is enabled in the general options.

CharacterEncoding
The character encoding that is used for outgoing and incoming messages with Data Coding Scheme (DSC) 0, supported encodings are the GSM default alphabet (default), ISO 8859-1, ISO 8859-15 and 7-bits packed GSM default alphabet.

ClientAsyncWindowSize
Maximum number of **sm_deliver** packets that are sent to the SMPP client without waiting for the response, default is **1**. Setting this property to a higher number increases the speed at which received messages and delivery reports are sent to the SMPP clients.

ConnectionTimeout
Connection inactivity timeout (in seconds), the connection to the SMPP client is closed when the inactivity timeout is triggered. The SMPP client can send the **enquire link** packet to keep the connection alive.

LocalIPAddress
The local IP address where the SMPP server listens on, leave empty for all available IP addresses.

IncludeTLVParameters
Allow optional TLV parameters in deliver_sm packets to the SMPP client. Set to **False** if the SMPP client does not accept TLV parameters.

MaxMessagesPerMinute
Maximum message throughput per client connection per minute, set to '0' for unlimited message throughput.

MaxUserIdConnections
Maximum number of client connections per user ID, set to '0' for unlimited connections per user ID. Some SMPP clients require two connections for full functionality, a transmitter connection and a receiver connection.

MessageIdFormat
The SMPP message ID format for outgoing messages, the traditional message ID format for SMPP version 3.3 is **Hexadecimal**, with this format the message Id is formatted as a hexadecimal number in the packet response and as the correspondent decimal number in the receipt. A **GUID** formatted message ID can be selected if you require a unique message ID and if the SMPP clients accept it.

MessageInHandling
This property specifies how the received messages and receipts are sent to the SMPP client (only for SMPP versions 3.4 and 5.0). **Deliver_SM** sends long messages in multiple message parts, this is the standard option for most SMPP clients. The **Deliver_SM_UsePayloadTlv** and **Data_SM** settings send long messages in one SMPP packet.

MessageValidity
Validity period (in days) for received messages and receipts. The messages and receipts are removed from the internal database when the SMPP client did not retrieve them within the validity period.

ReceiptDateFormat
Date/time format used in receipts. Leave emtpy for the default format (yyMMddHHmm). Set to 'yyMMddHHmmss' to include seconds.

ReceiptTimestampOffset
Offset in (fractional) hours for the receipt time stamp. Use a positive number to increase, and a negative number to decrease the time.

SendDlrOnStatus200
Send a delivery report when the message is sent upstream (status code 200). The status in the delivery report can be set to delivered (DELIVERD), accepted (ACCEPTD) or unknown (UNKNOWN).

MessageOutHandling
Keep at **Default** to send as normal (combined) text messages. Set to **SMPP_PDU** to forward each individual SMPP packet as raw SMPP packet with message type **smpp.pdu**.

Use16BitConcatenationId

Use 16-bit concatenation ID header for received (sm_deliver) multi-part messages (the default concatenation ID for SMS is 8-bits).

SSLCertificate
The full path and filename of the X509 (OpenSSL) certificate required to enable secure connections to the SMPP server.

SSLCertificatePassword
The password of the X509 certificate used for SSL connections.

Index