

DJIGZO EMAIL ENCRYPTION

Djigzo Gateway Separate Front-end and Back-end Configuration Guide



October 19, 2011, Rev: 6514

Contents

1 Introduction	3
2 Back-end	3
2.1 SSL certificate	3
2.2 Configure SOAP	4
3 Front-end	5
3.1 Specify back-end server	5
3.2 Trust the SSL certificate	5

1 Introduction

This guide will explain how to configure the Djigzo gateway to work with a separate front-end and back-end. This guide does not explain how to setup a Djigzo gateway (see the installation guide for installation instructions). This guide assumes that Djigzo is installed on Ubuntu. For installations on other systems, some of the instructions might be different.

Note: currently, the back-end does not have any notion of roles. Once an external user has successfully authenticated, the external user is allowed to access all SOAP services. Future versions of Djigzo will add support for roles on the back-end.

This guide assumes that the following IP addresses are used for the front-end and back-end server:

Front-end: **192.168.178.71**

Back-end: **192.168.178.72**

Note: commands that should be executed by the user, are shown on lines starting with a \$ sign (the \$ sign is not part of the command to execute). You can directly copy and paste the commands to the command line.

2 Back-end

SOAP is being used to communicate between the front-end and back-end. To protect the SOAP connection against sniffing and spoofing, the connection should be protected with TLS. A SSL certificate should therefore be available which is valid for the IP address (or domain name) of the back-end server. The following part will explain how to make a self signed SSL certificate with openssl. This part can be skipped if a valid SSL certificate is already available.

2.1 SSL certificate

A SSL certificate valid for the back-end IP address (or domain name if the back-end server has a domain name) will be created using openssl in the directory of the Djigzo server. In the last step, openssl asks for some user input like country, city etc. Only the *common name* is required. The *common name* should be equal to the IP address of the back-end server (in this case 192.168.178.72). Alternatively instead of using an IP address, the domain name can be used.

Create SSL certificate:

```
$ cd /usr/share/djigzo
$ sudo mkdir ssl
$ cd ssl
$ sudo sh -c "openssl genrsa 2048 > ssl-soap.key"
```

```
$ sudo sh -c "openssl req -new -x509 -nodes -sha1 -days 365 \  
-key ssl-soap.key > ssl-soap.cert"
```

Note: make sure that the *common name* is the correct IP address or domain name. If not, the SSL connection cannot be established.

Create pfx file

The generated key and certificate should be stored in a password protected pfx file. Use "djigzo" (without the quotes) for the pfx password¹.

```
$ sudo openssl pkcs12 -export -inkey ssl-soap.key \  
-in ssl-soap.cert -out ssl-soap.pfx
```

The pfx file should be owned by user *djigzo* and only readable by owner.

```
$ sudo chown djigzo:djigzo ssl-soap.pfx  
$ sudo chmod go-r ssl-soap.pfx
```

The generated key should be removed for safety reasons as it's no longer needed.

```
$ sudo rm ssl-soap.key
```

2.2 Configure SOAP

Set the IP address, port and protocol

The IP address, port and protocol of the SOAP server are specified in the Djigzo main properties file. The soap.server properties in the properties file */etc/djigzo/djigzo.properties* should be changed to make the Djigzo back-end bind to the external IP address.

```
protected.system.soap.server.protocol=https  
protected.system.soap.server.bind=192.168.178.72  
protected.system.soap.server.port=9001
```

Note: the bind address should be set to the IP address of the back-end server².

Change default SOAP password

Because the SOAP interface will be made available to external users, it's advised to change the default soap password. The soap password is specified in the main Djigzo properties file */etc/djigzo/djigzo.properties*. The soap password can be changed by modifying the property "protected.system.soap.password".

¹if a different password is used, make sure that the password specified in the soap configuration file soap.xml matches the pfx password.

²The reason the IP address should be specified, is that the WSDL of the SOAP services will specify the actual IP address of SOAP service. The address should therefore be equal to the IP address of the server itself because the server will handle SOAP.

Note: the new soap password should also be set on the front-end (this will be explained in the front-end section).

Allow access to port 9001

If a local firewall is running, for example `ufw`, the firewall should be opened to allow incoming connections to port 9001. For `ufw` this can be done using the following command:

```
$ sudo ufw allow 9001/tcp
```

Note: for added safety, access to port 9001 should only be allowed from the front-end server IP.

Restart Djigzo

Djigzo should be restarted to make sure the new settings are being used.

```
$ sudo /etc/init.d/djigzo restart
```

3 Front-end

3.1 Specify back-end server

The front-end web application should be setup to connect to the remote Djigzo back-end. The following settings should be added to the Tomcat default configuration file `/etc/default/tomcat6`.

```
JAVA_OPTS="$JAVA_OPTS -Ddjigzo.ws.server.host=192.168.178.72"  
JAVA_OPTS="$JAVA_OPTS -Ddjigzo.ws.server.port=9001"  
JAVA_OPTS="$JAVA_OPTS -Ddjigzo.ws.server.protocol=https"  
JAVA_OPTS="$JAVA_OPTS -Dsoap.password=*****"
```

Note: The host IP address should be replaced with the IP address of the back-end server and the soap password (****) should be replaced with the soap password used by the back-end.

3.2 Trust the SSL certificate

If a self signed SSL certificate is used, or an SSL certificate is used which is not by default trusted, the self signed SSL certificate or the root that issued the certificate should be trusted by Tomcat.

The following steps assume that the SSL certificate used is a self signed certificate which was generated using the steps specified in the previous section. If an existing SSL certificate issued by a non-trusted root is used, some of the following steps should be modified to match the root certificate used.

Copy the self signed SSL certificate

The self signed certificate should be copied from the back-end server to the front-end server.

```
$ scp djigzo-admin@192.168.178.72:/usr/share/djigzo/ssl/ssl-soap.cert .
```

Trust the self signed SSL certificate

By copying the self signed SSL certificate to the local ca certificate store, the certificate will be trusted.

```
$ sudo mv ssl-soap.cert /usr/local/share/ca-certificates/djigzo-ssl-soap.crt
$ sudo /usr/sbin/update-ca-certificates
```

Note: if djigzo-ssl-soap.crt was already stored in the ca certificates store, use the -f flag (*update-ca-certificates -f*) to force an update.

Restart Tomcat

Tomcat should be restarted to make sure the new settings are being used.

```
sudo /etc/init.d/tomcat6 restart
```