# Correctly Setup Encryption

## Introduction

Libraesva ESG has multiple functionalities to ensure that your traffic is always transmitted securely: HTTPS (TLS), SMTPS (TLS), end-to-end encryption. All these options come with the best defaults possible on installation, but reviewing the configuration is something very important to increase the security.

×**NOTE** Steps 4 and 5 are only required if you plan on using end-to-end Mail Encryption, otherwise the default are good enough.

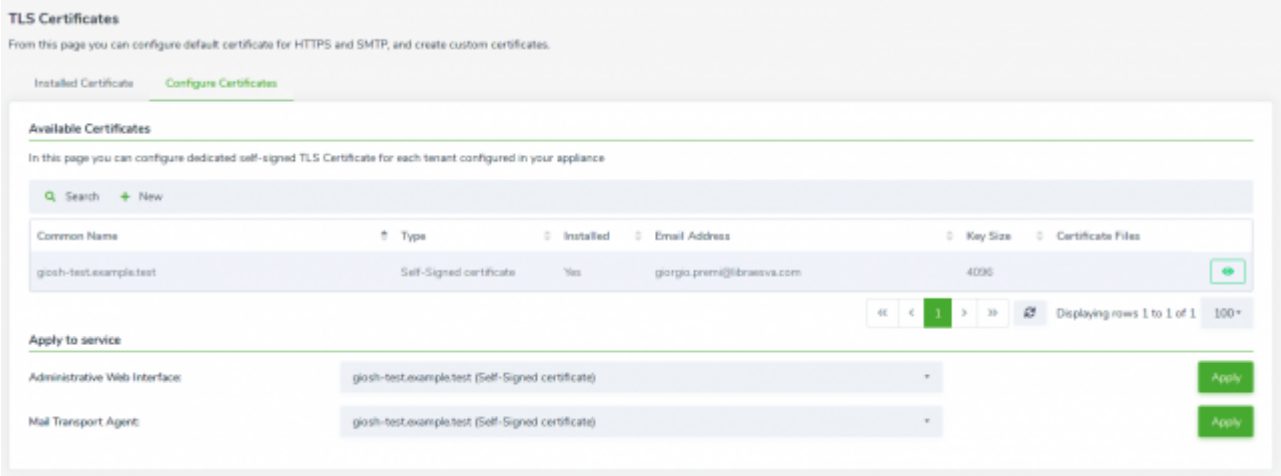
×**NOTE** This guide is for Libraesva ESG version 5.0 and above. Please do upgrade if you are still using previous versions.

## Creating TLS certificate

The first step in improving security is to configure a valid certificates using a well known Certificate Authority (CA) .Libraesva ESG comes with a self-signed certificate, which is enough to send and receive encrypted data but remote senders cannot verify your identity since there is no external Authority for your certificate.

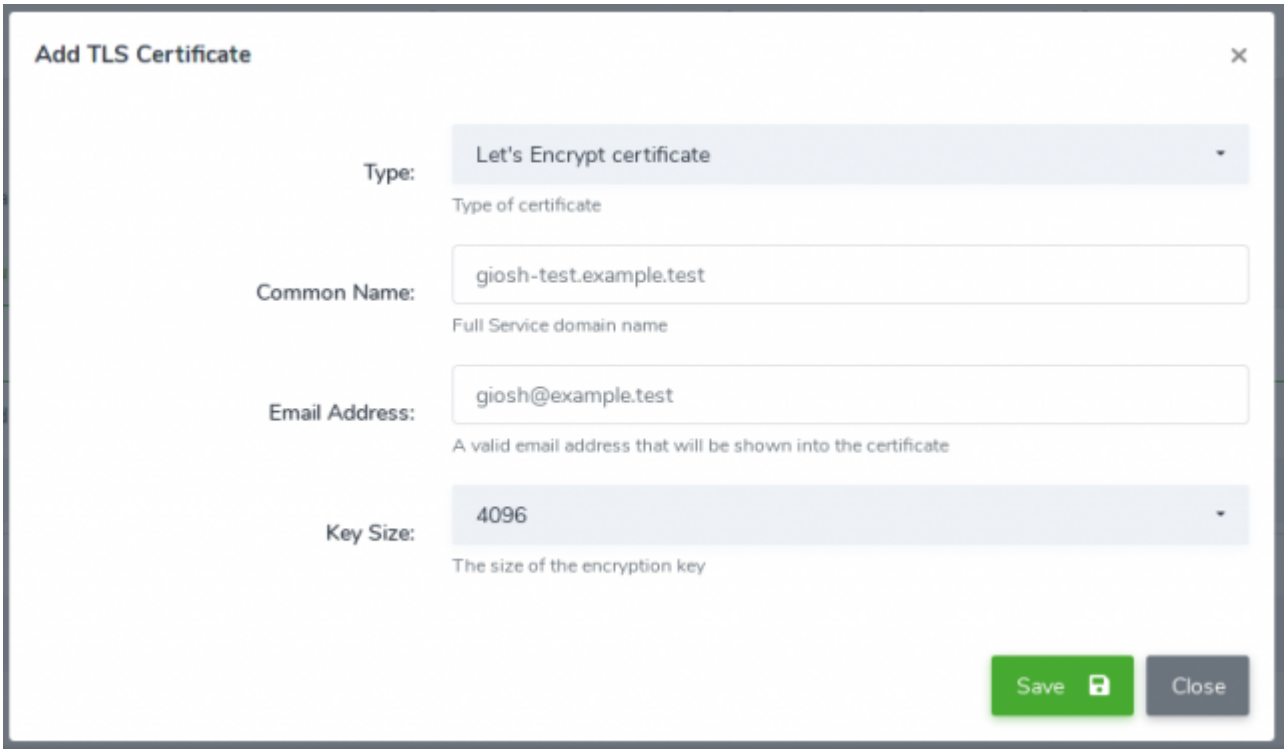
The main problem with self-signed certificates is that they are not trusted by the browser, so the user is presented a scary warning “Untrusted Certificate” and there is no (easy) way to make sure that someone isn’t trying to do spoofing.

To create a new certificate go to “Admin Area > Appliance > TLS Certificates”, then on tab “Configure Certificates” create a new configuration.




# a. Let’s Encrypt: auto-renew free certificate (Recommended)

If you plan on publishing the web interface of appliance also from outside the company, Let’s Encrypt is the way to go. Let’s encrypt provides an easy to use certificates free of charge, auto-renewed and with proper trusted public Certification Authority. It’s just a one click setup and you’re done!

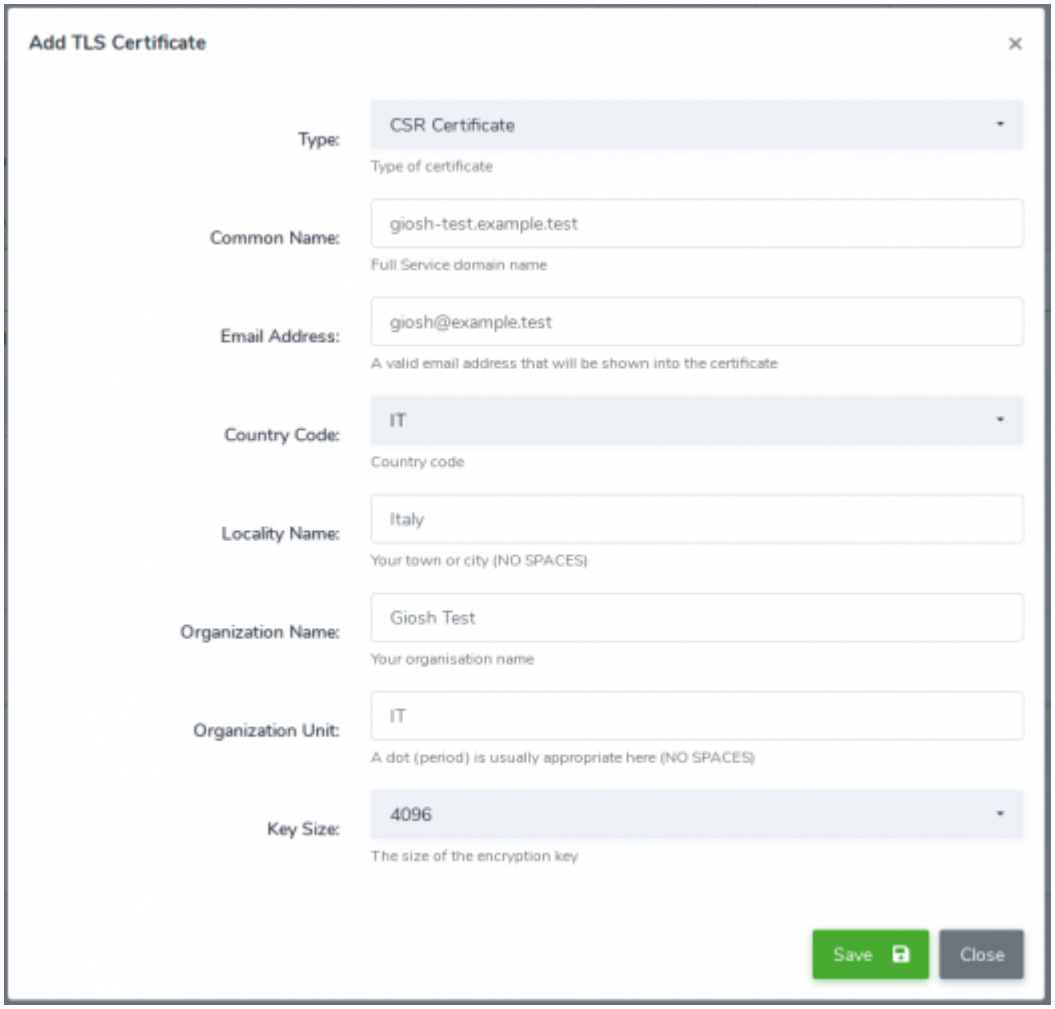



×**NOTE** Let’s Encrypt requires you to map both port 443 and port 80 for the chosen name to Libraesva ESG from all IP sources (e.g. from any address to your.tls-name.test:80 and your.tls-name.test:443). This is not a security issue as Libraesva ESG only use port 80 to for Let’s Encrypt validation and user are forced to use HTTPS.

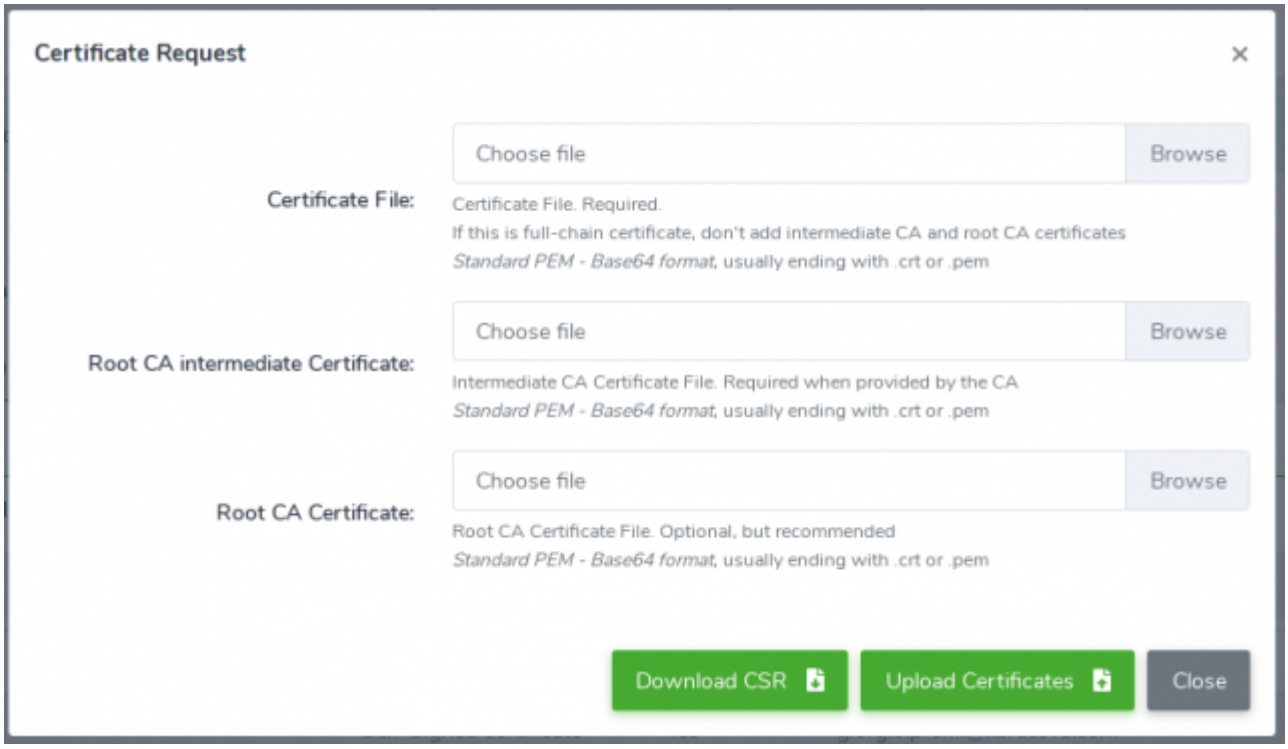
# b. Certificate Request: new hostname certificate

If you prefer to buy a certificate from a Certificate Authority for the hostname assigned to Libraesva ESG, then you have to create a Certificate Request from Libraesva ESG.

On Libraesva ESG create a new Certificate of type “CSR” and fill in all the information requested for the certificate.




From the newly created record you can **download the Certificate Request file** (file extension is *.csr), and use this on the Certification Authority panel to create properly signed certificates.




Once you receive the certificates back from the CA, make sure you **upload your certificate**, all **intermediate certificate files** and (optionally) the **CA root certificate** (these files extension is usually *.pem or *.crt).

If everything is done correctly and certificates are validated for the hostname, your record will now shows “Installed: yes”.

×**RENEWAL** before the certificates expires, you have to download the Request Certificate again and use it to create a new certificate on your CA. It’s better to update the current certificate, instead of creating a new one.

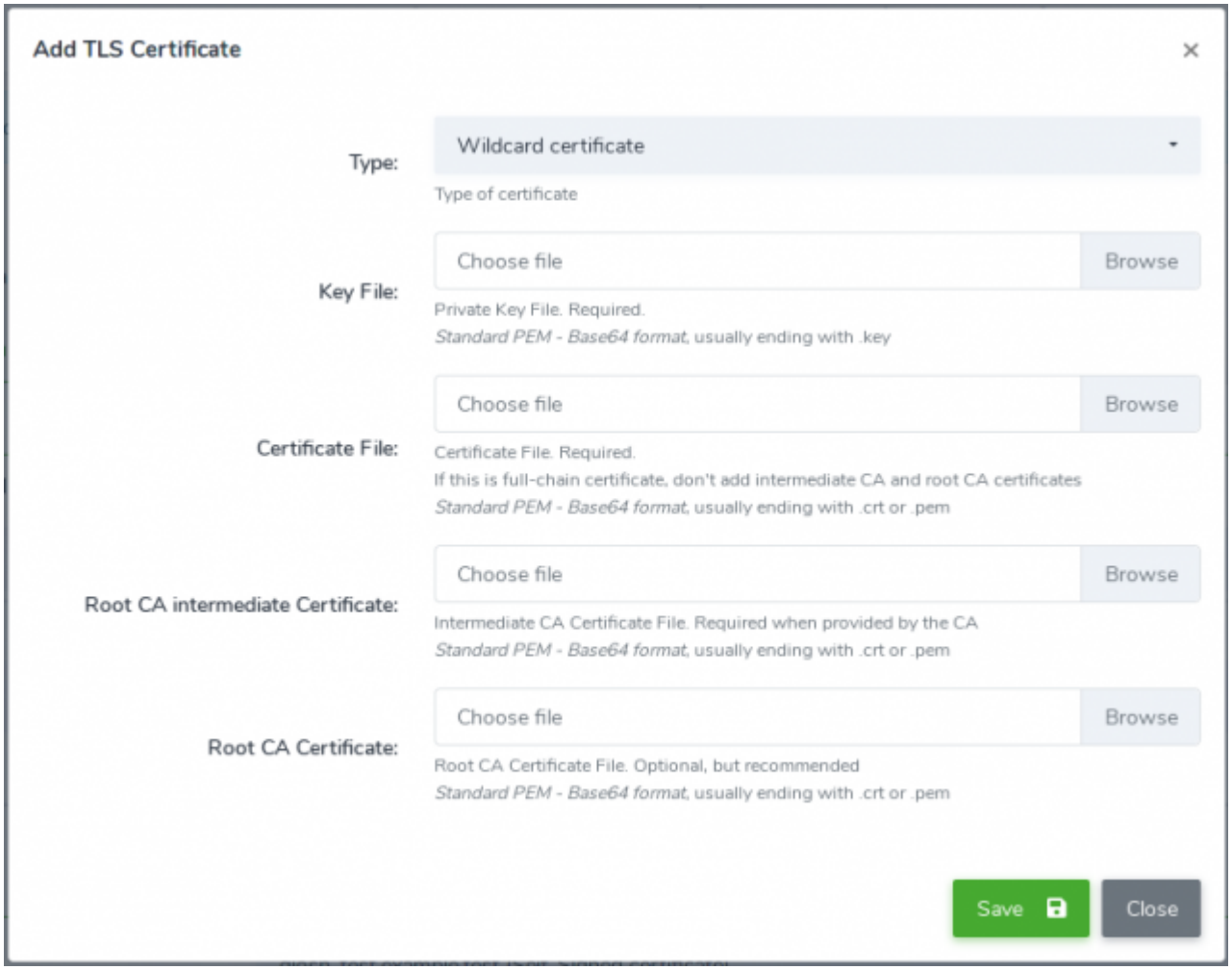
# c. Wildcard Certificate: if your company already owns it

Larger organizations often prefer to buy a single wildcard certificate to be used for all the hosts of your company. A wildcard certificate is similar to normal certificate, but as the name suggests the common name is not restricted to a single value (e.g. *.your-domain.test may be used for mail1.your-domain.test and mail2.your-domain.test, but not for two.sub-level.your-domain.test).

When installing wildcard certificate **you must also install the private key that the CA provides you when buying the wildcard certificates.**

On Libraesva ESG create a new certificate and select type “Wildcard”, than make sure you **upload your certificate private key**, the **wildcard certificate**, all **intermediate**

**certificate files** and (optionally) the **CA root certificate** (these files extension is usually *.pem or *.crt).




If everything is done correctly and certificates are validated, your record will be created and will show “Installed: yes”.

×**RENEWAL** before the certificates expires you have to upload the new certificates. It’s better to update the current certificate, instead of creating a new one.

# Use installed certificates for public services

Now that valid certificates are installed on Libraesva ESG, the can be assigned to Libraesva ESG services.

## Web access (HTTPS)

The web interface entry controls HTTPS service, that is the service used by **web browsers**. The certificate chosen should match the **hostname you want your users to use** while

browsing; with that in mind, may be a good idea to review the configuration for “Report Link URL” from “Admin area > Appliance > Quarantine settings”.

Plain text connection over HTTP are disabled on Libraesva ESG and there’s no override to it.

# Mail Transport (SMTPS)

The mail transport agent controls SMTPS service, that is the service used by **mail servers** to relay your email. The certificate chosen should match **hostname that is advertised by the SMTP banner**, which defaults to the appliance hostname; with that in mind, may be a good idea to review the configuration for “My Hostname” from “Admin area > Mail Transport > Advanced Configuration”.

Libraesva ESG is a mail gateway so it must accept both encrypted and plain text traffic as specified by RFC 2487. If you prevent the use of TLS for email you may bounce back messages to the sender, especially when you talk with small companies or outdated Mail Servers.

By default Libraesva ESG tries to use the highest protection available, but you may want to *configure stricter requirements for your domains*.

# Set TLS policies for email flow

## Incoming email

The responsibility of encrypting via TLS an inbound message *depends on the Mail Server sending to Libraesva ESG*. If you have uploaded a TLS Certificate signed by Certification Authority then you should enforce all your servers sending to Libraesva ESG to use an “*encrypt+verify*”

Review the list of all your trusted servers in “Admin Area > Mail Transport > Relay Configuration > Trusted Network” and the account list “System > Mail Transport > Relay Configuration > SMTP Auth”; for each of your trusted servers and make sure that strict TLS policies are configured.

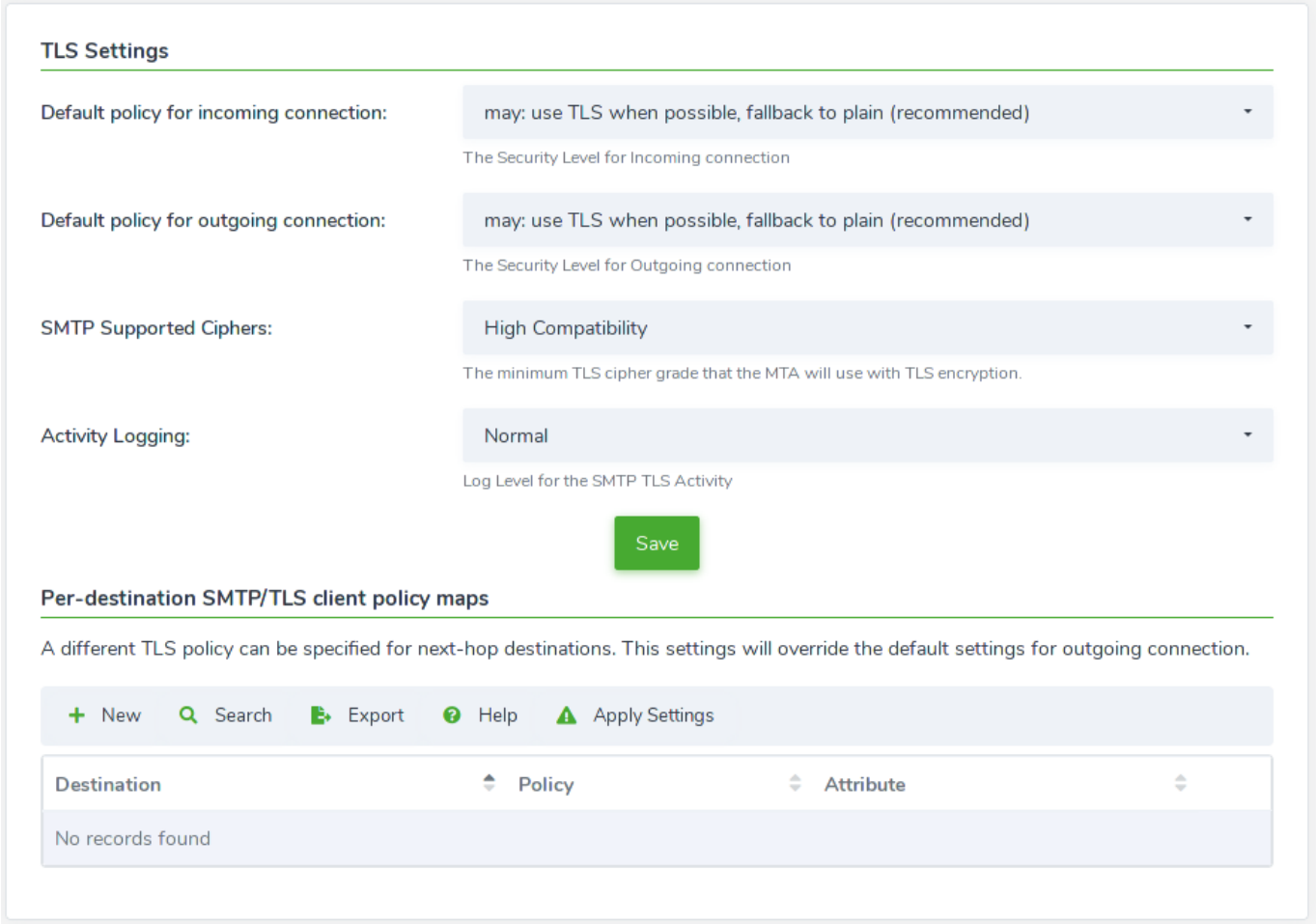
External resources:

- Setup secure mail flow in Exchange
- Zimbra Outgoing SMTP Authentication

# Outgoing email: TLS policies

Outgoing email traffic cannot be enforced to use TLS with a public Mail Server as specified by RFC 2487; if you do, you may see some message bounced back either because a receiver doesn't support one (or all) of the standard encryption algorithms used by Libraesva ESG.

It's a good practice to enforce TLS for Mail Servers managed by Libraesva ESG.




**Default policy for outgoing connection (required)**

Make sure that *"Default policy for outgoing connection"* is set to "*may*". This will grant proper reception of emails from all sources.

**Force Encryption for Trusted Senders (recommended)**
For each of the trusted servers, it is pretty straightforward to setup the policy "encrypt"; this ensures that all the traffic is never sent in plain text and you will grant email privacy for internal communication.

**Verify Domain and Encrypt (whenever possible)**
For each server for which you know that the certificate is valid and signed by a public

Certification Authority, you should add a “verify” policy. This means that before setting up the encrypted communication channel, the server is verified against public information (CA, DNS, SMTP banner).

# End-to-End Mail Encryption

Libraesva ESG end-to-end encryption begins transparently on the gateway and ensures your encrypted emails is only readable by the intended recipient.

You must review some of your setting to make sure that the communication between your servers and Libraesva ESG is safe. If you follow common security guidelines or you have followed this guide up to this point, you already have everything up and running.

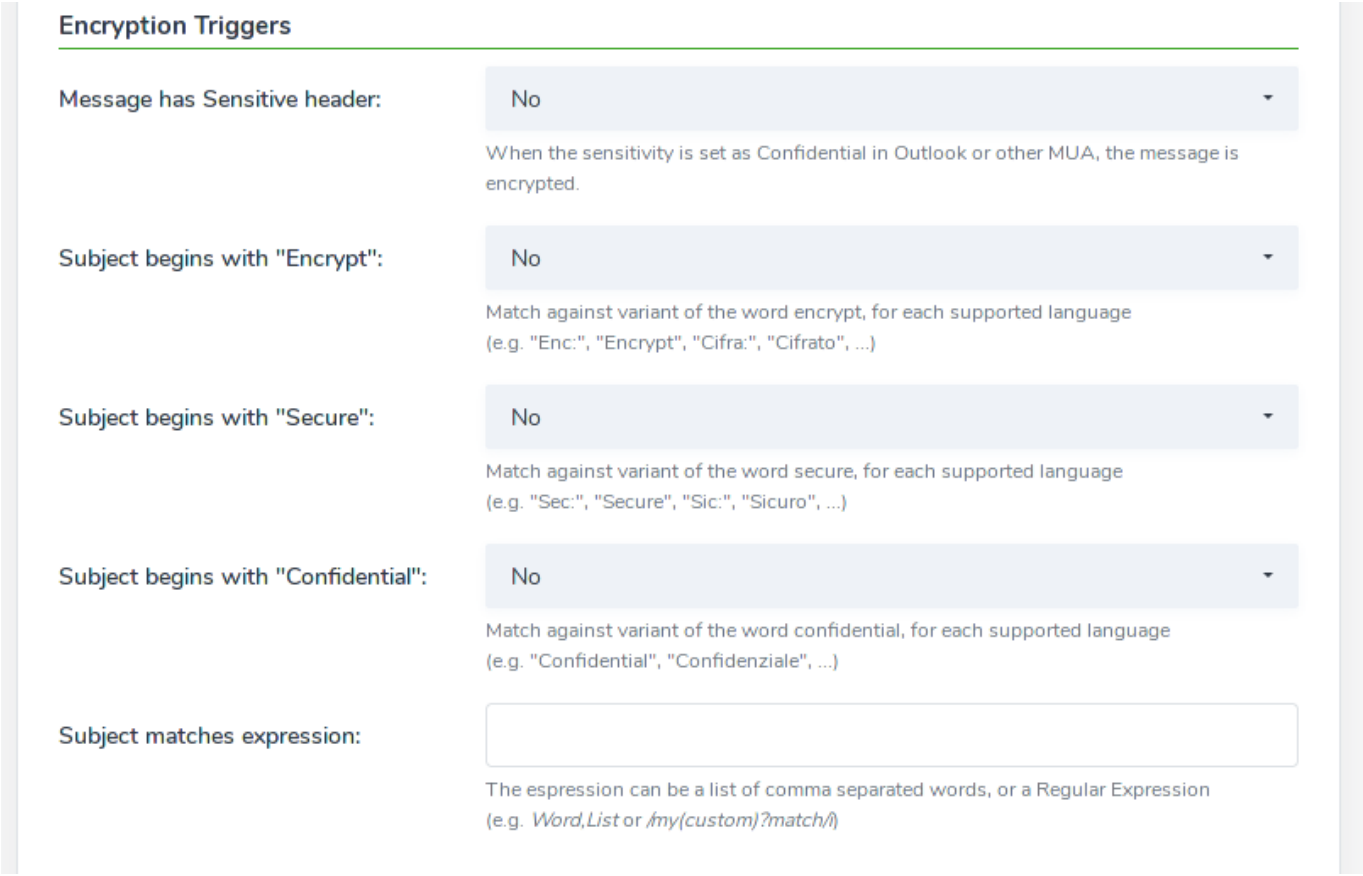
To recap the requirements, make sure that:

- a valid (non self-signed) certificate is installed, and assigned to Web Interface and Mail Transport Agent;
- Outgoing e-mail are secured by TLS in Libraesva ESG, possibly by forcing “encrypt/verify” policies for all your servers;
- Incoming e-mail are secured by TLS in all your Mail Server, possibly forcing the use of TLS.

# Give the users a way to explicitly encrypt

No matter which automation you have in your company, training the user to encrypt sensible data is one of the best way to prevent data loss.

Change the general settings of Mail Encryption, with something like the following:




It is suggested to keep “Sensitive” header detection enabled. Most of the users are comfortable with their Mail Clients (e.g. Outlook), by enabling this every message marked as confidential in a Mail Client is encrypted by Libraesva ESG.

For any other users, who prefers to tag the message in the subject, choose one of the available keywords: “Encrypt”, “Secure” and “Confidential”. You may enable more keywords, but be aware this could incur in too many messages being encrypted, so having a few is preferred.

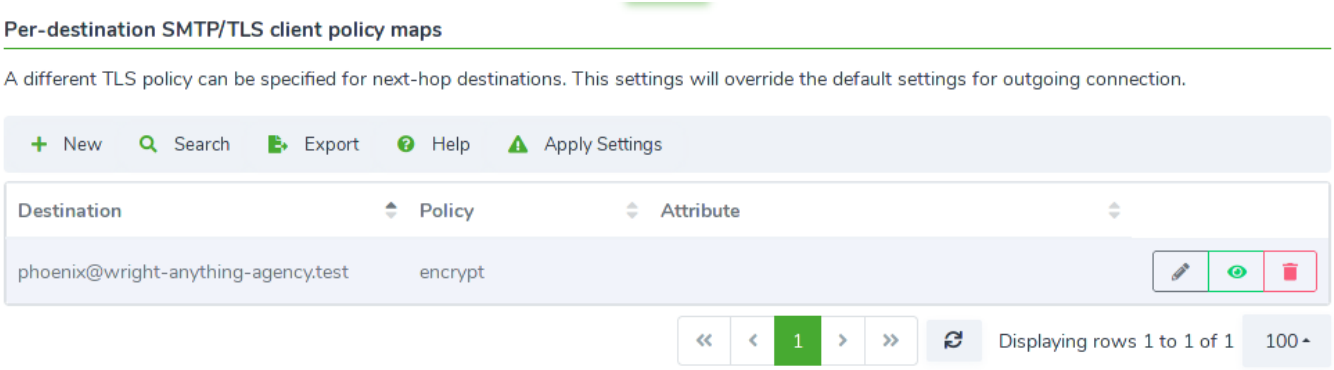
×**Localization** The encryption keyword in the subject are detected in many variants and for all supported languages (e.g. English, Italiand) and in many variants (e.g. Encrypt, Encrypted, Encryption, …).
If you prefer simple tagging to encrypt a message, you may want to add as a custom expression like *“[encrypt] [cypher]”*, and then you can prefix the subject with *“[encrypt]”*, or *“[cypher]”*.

# Configure End-to-End policy for sensible destination

You have the possibility to encrypt all the message for a single destination (or source),

independently of confidentiality level or content analysis.

As an example, you may encrypt all the outgoing message to you lawyer or the company business consultant.

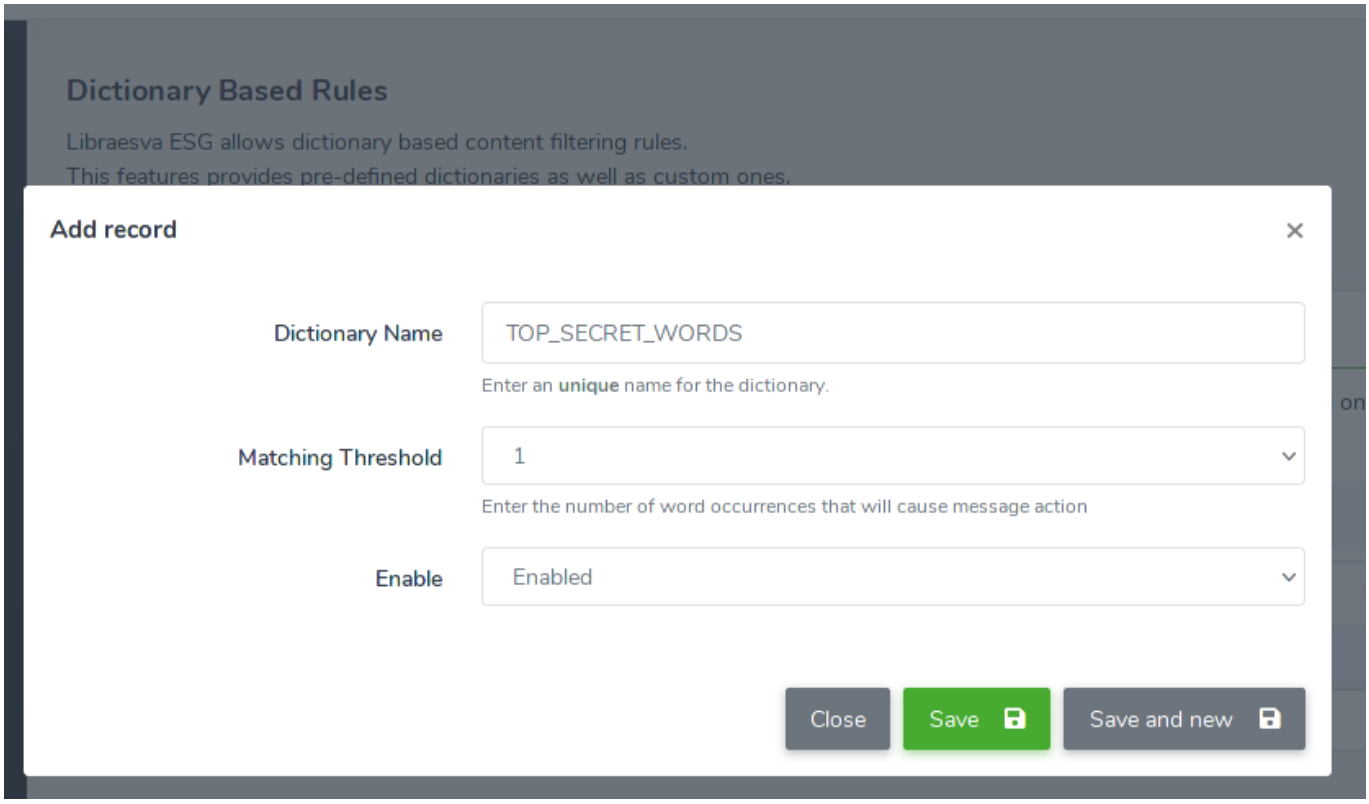



# Use dictionaries of reserved words to prevent data loss

Mail Encryption is integrated seamlessly in dictionaries content detection, both for inbound and outbound messages. To safeguard secrets you may want to create a dictionary of sensible words to trigger encryption.

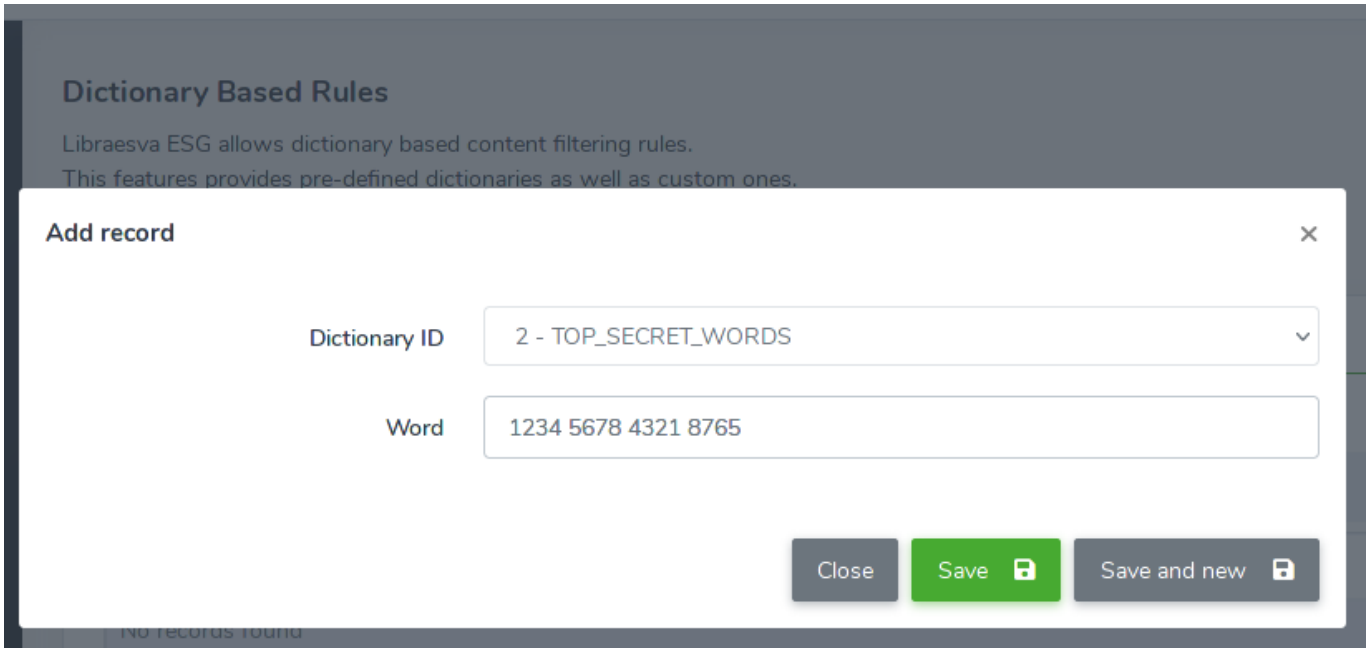
## Step #1: New dictionary

First create a new Dictionary for top secret words:



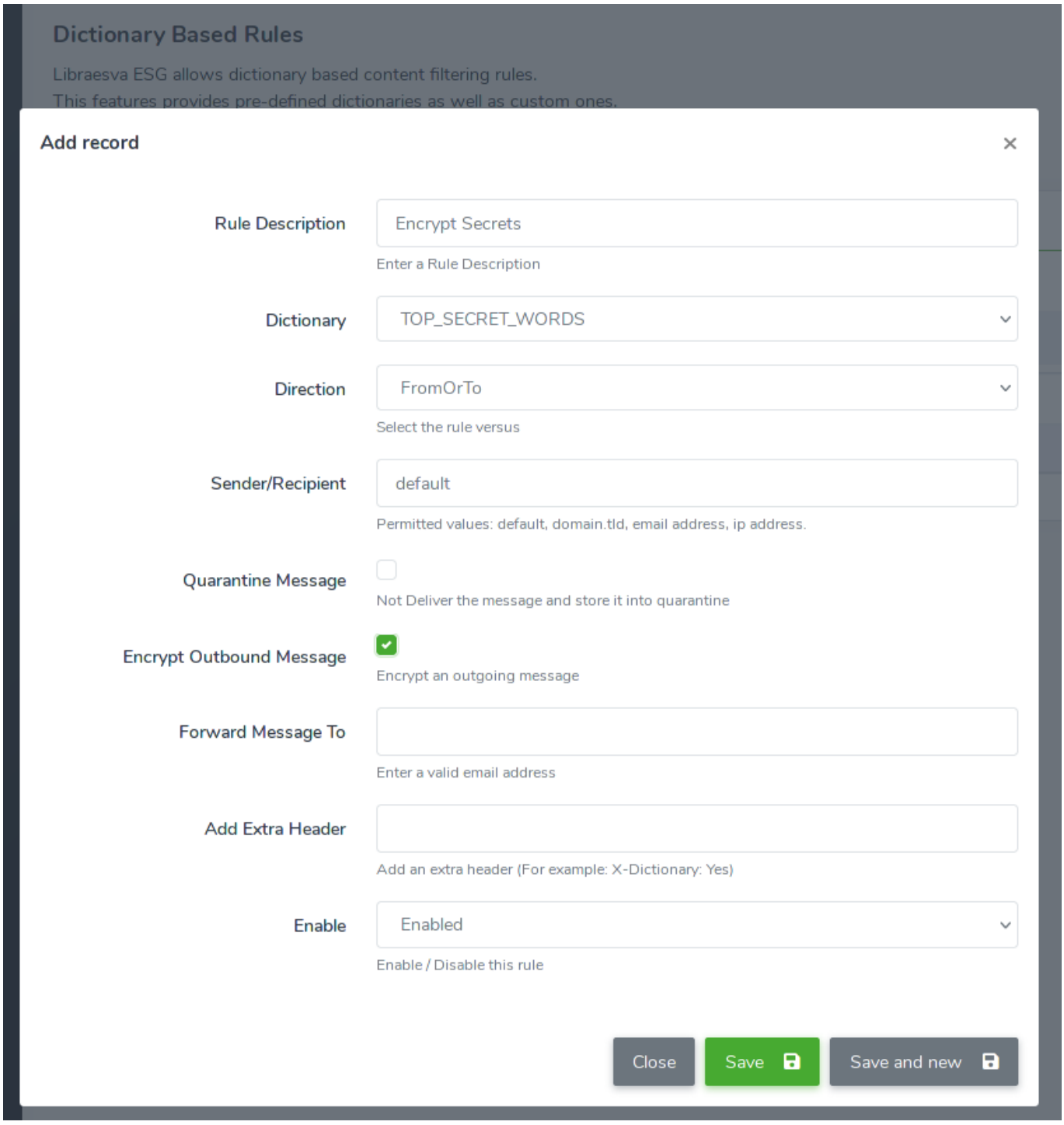

## Step #2: Add sensible information

Then add sensible words or numbers to dictionaries. Good candidates are Credit Card Numbers, VAT Number, National Insurance Numbers, … You may want to add one word at a time or use text import for bulk inserts.



# Step #3: Encryption trigger

Finally create an encryption rule for your dictionary and apply settings to make if effective




×**HINT** From now on whenever there is the need to track a new secret word or code, you can simple add an entry in your dictionary.