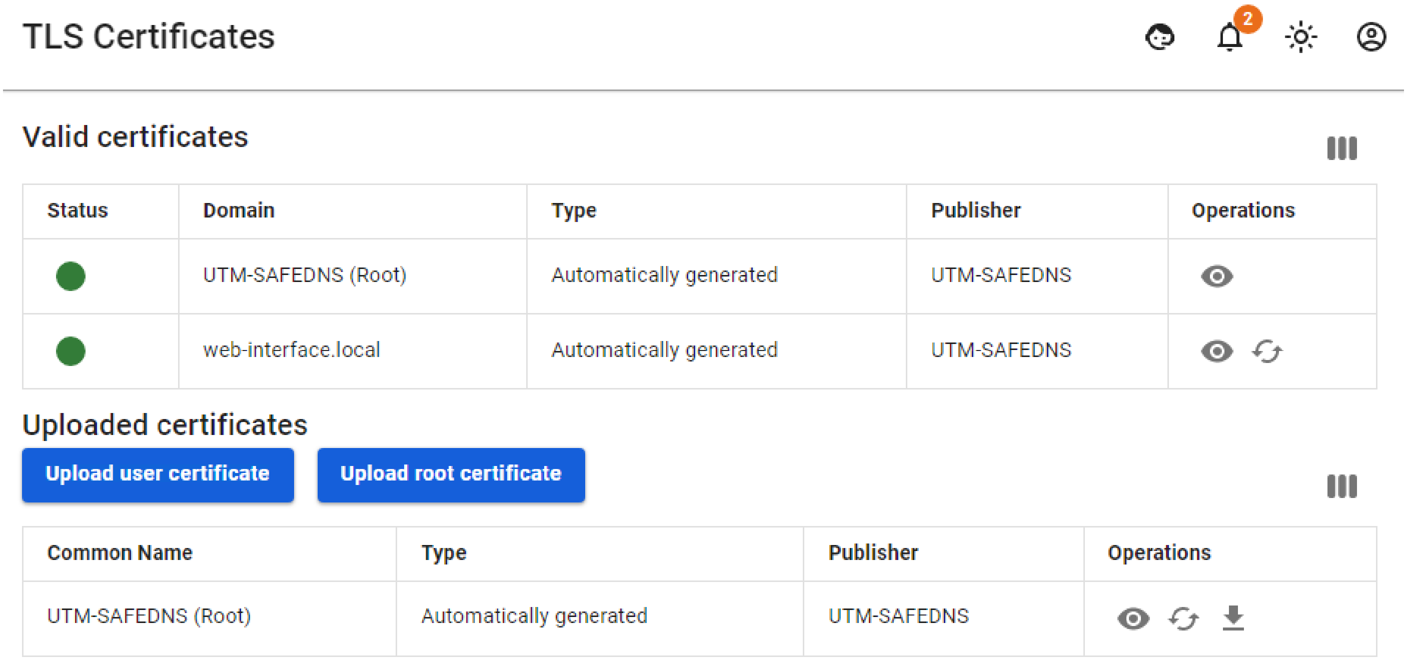
Certificates

- TLS Certificates
- Uploading your SSL certificate to server

TLS Certificates

Section with information about SSL certificates.

This section displays SSL certificates/certificate chains, the list of which is formed by the following modules: reverse proxying module, IKEv2, SSTP VPN servers, web interface, web authorization, mail, etc.



Status	Domain	Type	Publisher	Operations
●	UTM-SAFEDNS (Root)	Automatically generated	UTM-SAFEDNS	👁
●	web-interface.local	Automatically generated	UTM-SAFEDNS	👁 ⟳

Common Name	Type	Publisher	Operations
UTM-SAFEDNS (Root)	Automatically generated	UTM-SAFEDNS	👁 ⟳ ⬇

Valid certificates

The table *Valid Certificates* shows the ones generated automatically, as well as the downloaded certificate chains used by SafeUTM.

> If the same certificate chain is listed in several rows of the *Valid Certificates* table, then this chain is used by several modules.

Downloaded certificates

The *Downloaded Certificates* table shows all downloaded certificate chains, as well as the SafeUTM root certificate. For more information, see **Uploading your SSL certificate to server**.

> To view basic information about the certificate (serial number, expiration date, etc.), click the eye icon.

How is the certificate issued?

1. A local certificate chain is created, and signed by a root (self-signed) certificate.
2. Simultaneously with the creation of a local certificate chain, a request is sent to issue the chain to Let's Encrypt.
3. If the Let's Encrypt certificate chain is successfully issued, it will replace the local chain.
4. If the Let's Encrypt certificate chain issue fails, then the local certificate chain will be used.

How is the certificate reissued?

When reissuing a non-root certificate chain, UTM will try to update the chain as follows:

- It checks the downloaded certificates. If the certificate is found, it will replace the previous chain with the found downloaded one.
- If there are no downloaded certificates, then SafeUTM will turn to Let's Encrypt to issue a new certificate chain.
- If the chain from Let's Encrypt is received, it will be displayed in the table.
- If it was not possible to get a chain of certificates from Let's Encrypt, then a local chain of certificates is created and signed by the root certificate.

When the root certificate is reissued, UTM will replace the previous certificate with an automatically generated root certificate.

Features

If you want to try again to get a Let's Encrypt certificate instead of a self-signed one, you need to click **Reissue** in column **Management**.

When replacing/reissuing the root certificate chain, **IPsec connections Head office <-> Branch** will stop working and they will need to be recreated.
If you want to replace an automatically issued certificate chain with your own, then when uploading your own certificate chain, the **CN (Common name)** of the last certificate in the chain must match the domain for which the certificate is being uploaded.
Let's Encrypt certificate is **issued for 3 months** and will be **automatically reissued** upon expiration.
From this section, you can download the root (self-signed) certificate by clicking on the corresponding link.

To upload an SSL certificate to the server, see the article **Uploading your SSL certificate to server**.

Uploading your SSL certificate to server

After purchasing a trusted SSL certificate from Certificate Authority (CA), you need to create a text file of the type:

```
-----BEGIN PRIVATE KEY-----
.....
.....
-----END PRIVATE KEY-----
-----BEGIN CERTIFICATE-----
.....
.....
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
.....
.....
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
.....
.....
-----END CERTIFICATE-----
```

This file consists of two logical blocks:

- The block with a private key and the block of certificates consisting of a root certificate, a domain certificate, and vendor certificates.
- The block with certificates that the CA will send you follows the block with the private key.

Be careful: in addition to the root and domain certificate, the CA will most likely send additional vendor certificates consisting of several additional certificates in one file (bundle). This bundle of certificates must be added after the main certificate is issued for your domain. The order of the blocks in the file can be represented as follows:

```
Private key
Certificate for domain
Certificate from the vendor-certificates bundle
```

```
4Certificate from the vendor-certificates bundle

...

The main (root) certificate
```

After that, you can upload the received file with the private key and certificate to UTM via the web interface. To do this, go to **Services -> TLS Certificates**.

The generally accepted standard for creating a certificate chain file can also be found here: https://www.digicert.com/ssl-support/pem-ssl-creation.htm.

Encrypted private key

Only the standard private key format is supported: decrypted PEM. Such a key starts with the line:

`-----BEGIN RSA PRIVATE KEY-----`

Sometimes the CA issues an encrypted private key using a passphrase. In this case, you need to decrypt (convert) the encrypted key into a regular one using the `openssl` utility or, if the CA provides other tools for this, use them. The list of parameters for calling `openssl` to convert the key into an unencrypted form depends on CA's key encryption technology and should be described in the instructions for installing the certificate from the CA. **You cannot upload and use an encrypted private key on the SafeUTM server.**

Instructions for Creating Certificate on Windows OS.

To create a certificate, follow these steps:

1. Download the OpenSSL program. Link to the program:

http://slproweb.com/products/Win32OpenSSL.html.
2. Install OpenSSL.
3. If the certificate file is in **pkcs12** format**:** (if it is in .pem format, then you can immediately proceed to Subparagraph **d**):

- **a.** Place this file in the directory C:\OpenSSL-Win64\bin (in the folder with the OpenSSL program installed).
- **b.** Open the command prompt.
- **c.** In the command prompt, go to the directory with the OpenSSL program installed.
- **d.** Enter command `openssl pkcs12 -in certificate.pkcs12 -out certificate.pem` (with this command, you will convert the certificate to the desired format). **certificate.pkcs12** is the source certificate that you received from the certification authority (hereinafter CA); **certificate.pem** is the result of the conversion.
- **e.** Open the resulting file in a text editor (for example, in notepad).
- **f.** The file has the following structure:

~~~~
```
 -----BEGIN CERTIFICATE-----
 .............
 .............
 -----END CERTIFICATE-----
 -----BEGIN ENCRYPTED PRIVATE KEY-----
 .............
 .............
 -----END ENCRYPTED PRIVATE KEY-----
```
~~~~

or this structure:

```
-----BEGIN CERTIFICATE-----
.............
.............
-----END CERTIFICATE-----
-----BEGIN PRIVATE KEY-----
.............
.............
-----END PRIVATE KEY-----
```

If it is written in the certificate `--BEGIN ENCRYPTED PRIVATE KEY--`, then you need to decrypt it using the OpenSSL utility. **Command to decrypt:** `openssl rsa -in certificate.pem -out certificate_decoded.pem`. **certificate.pem** is the file that you received after conversion in Step d; **certificate_decode.pem** is the result of decryption. If in the certificate it says **--BEGIN PRIVATE KEY--**, then the certificate file has already been decrypted. You can proceed to the next step.

4. Create an empty file with extension .pem (**my_certificate.pem**).
5. Open it with a text editor.
6. Open the file that you got in Step 3 (**certificate_decode.pem).** From this file you need to copy the text of the type (private key):

```
-----BEGIN PRIVATE KEY-----
.............
.............
-----END PRIVATE KEY-----
```

7. Paste the copied text into the file created in Step 4 (**my_certificate.pem).**
8. Go to the file created in Step 3 (**certificate_decode.pem).** From this file you need to copy the text of the type (your domain certificate):

````
-----BEGIN CERTIFICATE-----
.............
.............
-----END CERTIFICATE-----
````

9. Paste the copied text into the file created in Step 4 (**my_certificate.pem).**
10. The CA, in addition to your certificate, should have sent you a certificate bundle (there may be several of them) and a root certificate. If you don't have these certificates, you can download them online or request them from your CA.
11. From the certificate bundle and the root certificate, copy the text of the type:

~~~
```text
-----BEGIN CERTIFICATE-----
.............
.............
-----END CERTIFICATE-----
```
~~~

12. Paste the copied text into the file created in Step 4 (**my_certificate.pem).** In the beginning, you will need to insert the text from the certificate bundle, and at the very end the text of the root certificate.
13. As a result, you will get a file of blocks:

~~~
```
Private key
domain certificate
Certificate from the vendor-certificates bundle
Certificate from the vendor-certificates bundle
.........
Root certificate
```
~~~

14. Upload the resulting file to UTM. To do this, go to **Services -> TLS Certificates**.