

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:

```
```\n\n-----BEGIN CERTIFICATE-----\n\n\n\n\n-----END CERTIFICATE-----\n\n-----BEGIN ENCRYPTED PRIVATE KEY-----\n\n\n\n\n-----END ENCRYPTED PRIVATE KEY-----\n\n```\n
```

or this structure:

```
-----BEGIN CERTIFICATE-----\n\n\n\n\n-----END CERTIFICATE-----\n\n-----BEGIN PRIVATE KEY-----\n\n\n\n\n-----END PRIVATE KEY-----\n
```

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-----\n\n\n\n\n-----END PRIVATE KEY-----\n
```

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**.

---

Revision #4

Created 27 August 2022 17:14:16 by Val Redman

Updated 13 October 2022 15:41:31 by Val Redman