




How to use SSL certificate converter (v1.0).

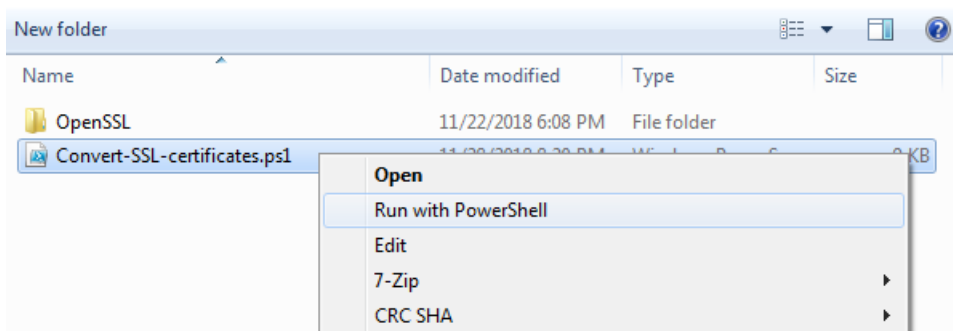
This is a useful tool that can convert a SSL certificate and its private key from PEM format into PKCS12 (.PFX) format. Additionally if needed it can also include the intermediate(s) and root CA certificates.

The described converter is available as a PowerShell script that provides the user a GUI.

Before running the script make sure that you have the **“OpenSSL”** folder next to the script file. This folder contains a pre-compiled version of OpenSSL and two cryptographic libraries that are used by the script.

Name	Size	Type
 libcrypto-1_1.dll	2,141 KB	Application extens...
 libssl-1_1.dll	392 KB	Application extens...
 openssl.exe	423 KB	Application

To start using it simply right click on the *“Convert-SSL-certificate.ps1”* file and select *“Run with PowerShell”*.



A window will open presenting 4 fields:

- **1) Cert file:** This is the actual SSL certificate file that will be converted. The script expects an already signed by a Certification Authority (CA) file in PEM format. PEM is an ascii representation of the actual SSL certificate content, you can identify this format if you open the file in Notepad and see the “-----BEGIN CERTIFICATE-----” and “-----END CERTIFICATE-----” tags.
- **2) Key file:** This is the private key generated when you created a CSR (Certificate Signing Request) for the actual SSL certificate to be signed by the CA. The script expects a private key file in PEM format (plain text ascii representation). You can identify this format by the opening “-----BEGIN PRIVATE KEY-----” and ending “-----END PRIVATE KEY-----” tags if you open the private key file in Notepad.
- **3) Intermediate CA:** This is one or more intermediate and/or root CA certificates bundled into one single file in PEM format. The “-----BEGIN CERTIFICATE-----” and “-----END CERTIFICATE-----” tags can be identified at the beginning and end of every certificate included in this file when opened in Notepad.
- **4) Password:** this is the password that will be used to protect the private key in the encrypted PFX file.

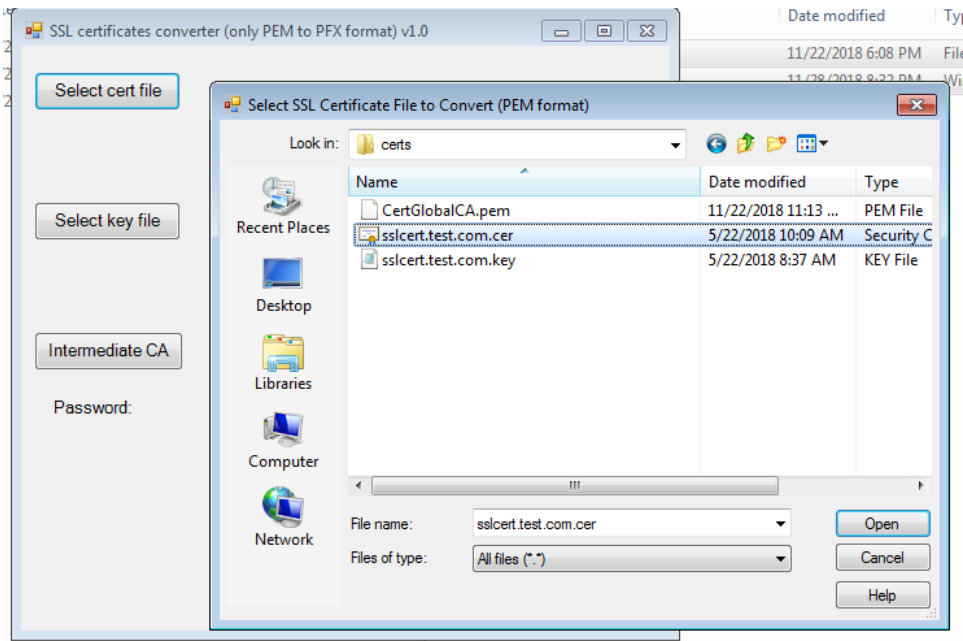


```
1 -----BEGIN CERTIFICATE-----
2 MIIEIzCCA3OgAwIBAgIQDI7gyQ1qiRWIBAYe4kH5rzANBgkqhkiG9w0BAQsFADBh
3 MQswCQYDVQQGEwJVUzEVMBMGA1UEChMMRGlnaUNlcnQgSW5jMRkwFwYDVQQLExB3
4 d3cuZGlnaWNaUNlcnQuY29tMSAwHgYDVQQDExdEaWdpQ2VydCBHbG9iYWwgUm9vdCBH
5 MjAeFw0xMzA4MDExMjAwMDBaFw0yODA4MDExMjAwMDBaMEQxCzAJBgNVBAYTA1VT
6 MRUwEwYDVQQKEwxEaWdpQ2VydCBHbG9iYWwgUm9vdCBHbG9iYWwgUm9vdCBH
7 bCBBDQSBHMjCCASIwDQYJKoZIhvcNAQEBBQADggEPADCCAQoCggEBANNIfL7zBYzd
8 W9UvhU5L4IatFaxhz1uvPmoKR/uadpFgC4przc/cv35gmAvkVN1W7SHMArZagV+X
9 au4CLyMnuG3UsOcGAngLH1ypmTb+u6wbBfpXzYEQQGfWMItyNdSWYb7QjHqXnrx5
10 IuYUL6nG6AEfq/gmD6yOTSwyOR2Bm40cZbIc22Gois89g5+vCShjEbyrpeJiJ7RfR
11
12
13
14
15
16
17
18
19
20 IwQYMBaAFE4iVCAYlebjbuYP+vq5Eu0GF485MA0GCSqGSIb3DQEBCwUAA4IBAQA
21 OYSR+ZfrqoGvhOlaOJL84mxZvzbIRacxAXHhBsCsMsdaVSnaT0AC9aHesO3ewFj2
22 dz12uYf+QYB6z13jAMZbAuabeGLJ3LhmnftiQjXS8X9Q9ViIyFEBFltcT8jW+rZ
23 8uckJ2/0lYDbliZkVivP6hnZf1WZUXoOLRg9eFhSvGNvVvvdRLNXSmDmyHBW4co
24 atc7TlJFGa8kBPJIERqLrqwYElasA8u49L3KJg6nwd3jM+/AVTAN1VlOnAM2BvjA
25 jxS2nE0qnsHhfTuvcdFuhOWKU4Z0BqYBvQ3lBetoxi6PrABDJXWKTUgNX31EGDk
26 92hiHuwZ4STyhxGs6QiA
27 -----END CERTIFICATE-----
28
```

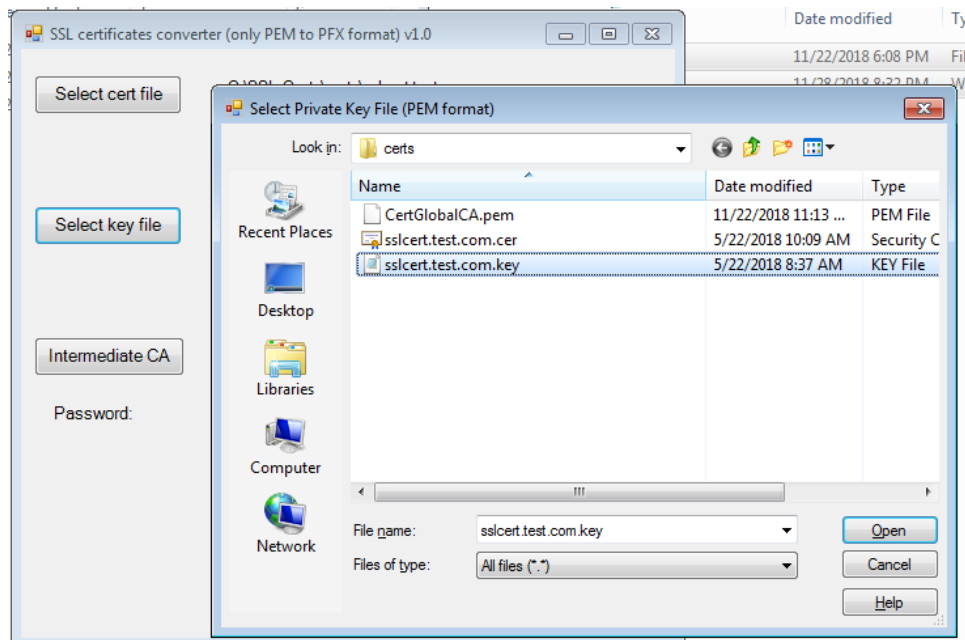
Figure 1 Example of a certificate file in PEM format.

STEPS TO CONVERT FROM PEM TO PFX.

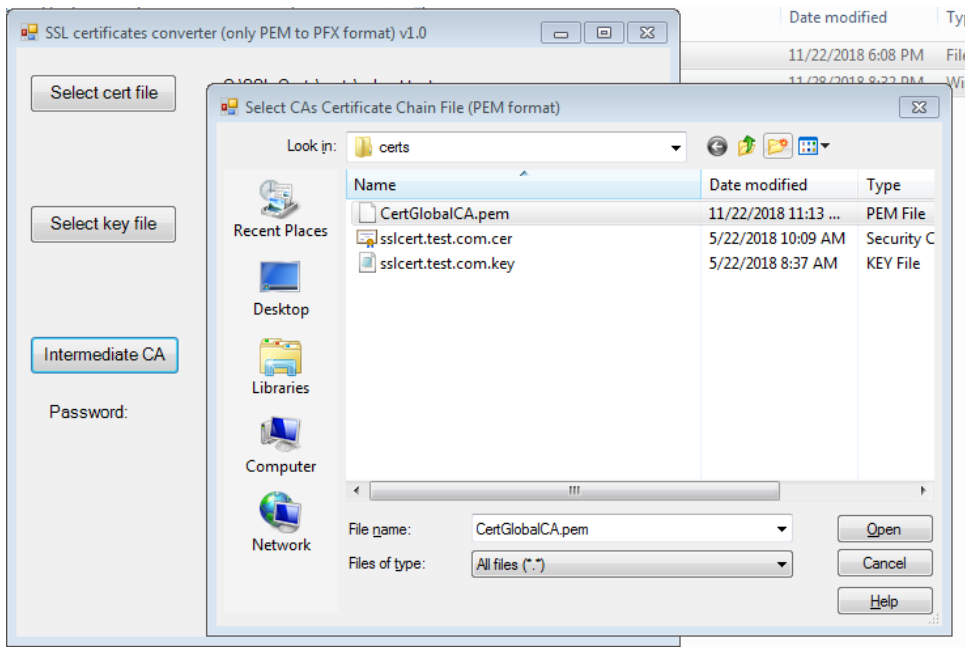
Step 1: Select a certificate file to convert to PFX format:



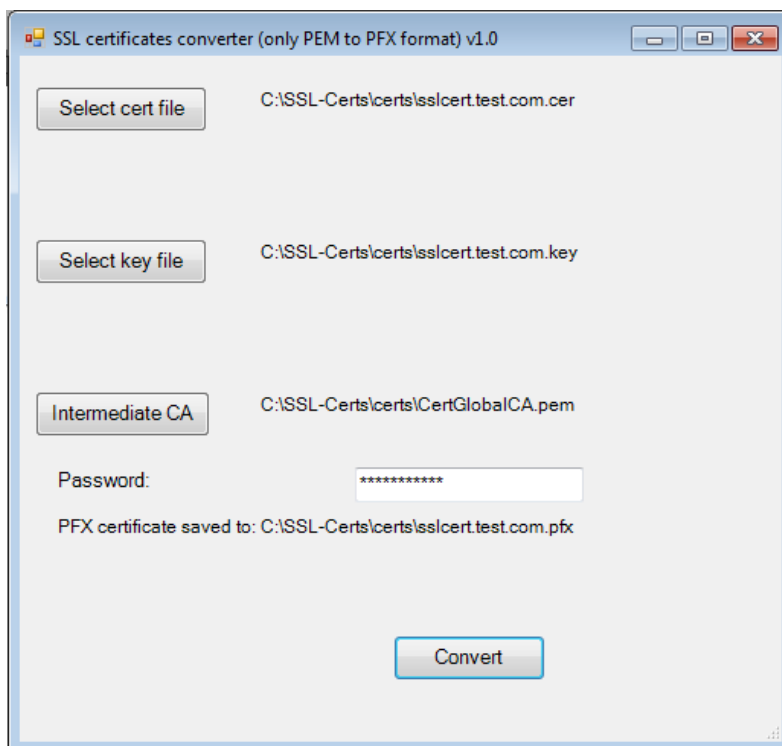
Step 2: Select the corresponding private key file:







Step 3 (Optional): Select the intermediate/root CA certificate. Only needed if you want to install the intermediate(s) and/or root CA certificates in the destination server. Notice that all intermediate and root CA certificates must be contained in a single file. This step is not needed if you only want to install the actual SSL certificate for your Website and the private key in the destination server.



Step 4: Provide a password to protect the private key and click on “Convert” button.



After a successful conversion the GUI will show you the path where the new PFX file was created. The output PFX file will be saved on the same folder from the input file and it will be named like the input file but with a *.pfx* extension as shown below.

Name	Type	Size
 CertGlobalCA.pem	PEM File	2 KB
 sslcert.test.com.cer	Security Certificate	3 KB
 sslcert.test.com.key	KEY File	4 KB
 sslcert.test.com.pfx	Personal Information Exchange	6 KB