
KNOWLEDGE BASE ARTICLE

How to convert a certificate into the appropriate format

This article provides detailed instructions on using OpenSSL to convert certificates between various formats such as PEM, DER, P7B, and PFX, including command examples for each conversion.

PUBLISHED

4 Jul 2022

DOCUMENT

InfoHub — Knowledge, news, and updates from EssingtonITS

How to convert a certificate into the appropriate format

4 Jul 2022

Using OpenSSL to Convert Certificates

OpenSSL can be used to convert certificates into the appropriate format. Begin by installing OpenSSL:

```
apt-get install openssl
```

Note: The PEM format is the most common format used for certificates. Extensions used for PEM certificates are `.cer`, `.crt`, and `.pem`. They are Base64 encoded ASCII files. The DER format is the binary form of the certificate and does not contain the 'BEGIN CERTIFICATE/END CERTIFICATE' statements. DER formatted certificates most often use the `.der` extension.

Conversion Commands

Convert x509 to PEM

```
openssl x509 -in certificatename.cer -outform PEM -out certificatename.pem
```

Convert PEM to DER

```
openssl x509 -outform der -in certificatename.pem -out certificatename.der
```

Convert DER to PEM

```
openssl x509 -inform der -in certificatename.der -out certificatename.pem
```

Convert PEM to P7B

Note: The PKCS#7 or P7B format is stored in Base64 ASCII format and has a file extension of `.p7b` or `.p7c`. A P7B file only contains certificates and chain certificates (Intermediate CAs), not the private key. The most common platforms that support P7B files are Microsoft Windows and Java Tomcat.

```
openssl crl2pkcs7 -nocrl -certfile certificatename.pem -out certificatename.p7b -certfile CACert.cer
```

Convert PKCS7 to PEM

```
openssl pkcs7 -print_certs -in certificatename.p7b -out certificatename.pem
```

Convert PFX to PEM

Note: The PKCS#12 or PFX format is a binary format for storing the server certificate, intermediate certificates, and the private key in one encryptable file. PFX files usually have extensions such as `.pfx` and `.p12`. PFX files are typically used on Windows machines to import and export certificates and private keys.

```
openssl pkcs12 -in certificatename.pfx -out certificatename.pem
```

Convert PFX to PKCS#8

Step 1: Convert PFX to PEM

```
openssl pkcs12 -in certificatename.pfx -nocerts -nodes -out certificatename.pem
```

Step 2: Convert PEM to PKCS8

```
openssl pkcs8 -in certificatename.pem -topk8 -nocrypt -out certificatename.pk8
```

Convert P7B to PFX

Step 1: Convert P7B to CER

```
openssl pkcs7 -print_certs -in certificatename.p7b -out certificatename.cer
```

Step 2: Convert CER and Private Key to PFX

```
openssl pkcs12 -export -in certificatename.cer -inkey privateKey.key -out  
certificatename.pfx -certfile cacert.cer
```

Convert PFX to CRT, PEM, and KEY with Key Password Removal

Convert a PFX file to a PEM file that contains both the certificate and private key:

```
openssl pkcs12 -in filename.pfx -out cert.pem -nodes
```

Convert a PFX file to separate public and private key PEM files, extract the private key from a PFX to a PEM file:

```
openssl pkcs12 -in filename.pfx -nocerts -out key.pem
```

Export the certificate (includes the public key only):

```
openssl pkcs12 -in filename.pfx -clcerts -nokeys -out cert.pem
```

Remove the password (paraphrase) from the extracted private key (optional):

```
openssl rsa -in key.pem -out server.key
```

Convert PEM to KEY and CRT

```
openssl rsa -outform der -in privkey.pem -out privkey.key  
openssl x509 -outform der -in fullchain.pem -out fullchain.crt  
openssl x509 -outform der -in cert.pem -out cert.crt
```