# Digital Certificate formats and filename extensions

In this article, we will learn about the available formats of Digital certificate, their purpose and some of the technical commands used for requesting and issuing the certificates.

We shall use ***OpenSSL***, which is an open source toolkit for performing cryptographic operations

Common formats for X.509 certificates are:

1. **pem - (Privacy Enhanced Mail)** - PEM formats file have Base64 encoded DER certificate, enclosed between the tags "BEGIN CERTIFICATE" and "END CERTIFICATE". This format can have multiple certificates. PEM standards are meant to provide message confidentiality and integrity to emails.
2. **cer, .crt, .der** - usually in binary format
3. **p7b, .p7c - PKCS#7** - PKCS #7 is a container which may contain plain data, signed data, encrypted data, or combination of these. It may also contain set of certificates needed to validate the certification chain.
4. **p12 - PKCS#12** - This format usually contains X509 certificates, public and private key. It is protected by password.
5. **pfx - PFX (Personal Information Exchange)** - Files have both the private and public keys. This format is preferred for creating certificates to authenticate applications or websites. Since this format has private keys, this file is password protected.

**Certificate Signing Request (CSR)**

Certificate Signing Requests are digital requests i.e. that are files which contain the information submitted by the person or organization requesting for the certificate. All information like Distinguished Name, Common name, Organization Name as well as the RSA Public key is part of CSR files. This CSR file is submitted to a CA.

**OpenSSL command for generating the CSR –** “Openssl req –newkey rsa:1024 -nodes –keyout C:\OpenSSL\RSAKeys.key –out C:\OpenSSL\CSR.csr”

This command will output two files

- RSAKeys.keys – contains the RSA private key.
- CSR.csr – Contains the Certificate Request, information about the requesting entity (this information will be contained in the certificate after the CA verifies it and the public key)

**Generating a X509 certificate from the CSR**

The command below takes a CSR file i.e. request as an input and outputs the Certificate.

OpenSSL req –x509 –in C:\OpenSSL\CSR.csr –out C:\OpenSSL\Certificate.cer –key C:\OpenSSL\RSAKeys.key

## Interconversion of certificate formats

- **Convert CER format certificate to PEM format certificate**
  The following command will convert an x509 certificate from cer to PEM format.
  ```
  x509 –in C:\OpenSSL\Certificate.cer –out C:\OpenSSL\Certificate.pem
  ```

- **Convert PEM Format Certificate to PFX Format Certificate**
  The following command will convert an x509 certificate from PEM to PFX format.
  ```
  pkcs12 -export -out C:\OpenSSL\Certificate.pfx -inkey C:\OpenSSL\RSAKeys.key -in certificate.pem
  ```

- **Convert PEM Format Certificate to PKCS12 Format Certificate**
  The following command will convert an x509 certificate from PEM to PKCS12 format.
  ```
  pkcs12 -export -out C:\OpenSSL\Certificate.p12 -inkey C:\OpenSSL\RSAKeys.key -in certificate.pem
  ```

- **Convert PKCS12 Format Certificate to PEM Format Certificate**
  The following command will convert an x509 certificate from PKCS12 to PEM format.
  ```
  pkcs12 -export -out C:\OpenSSL\Certificate.pem -inkey C:\OpenSSL\RSAKeys.key -in certificate.p12
  ```

**Source: http://www.go4expert.com/articles/digital-certificate-formats-filename-t24831/**