Difference between pem, crt, key files

Asked 4 years, 8 months ago Modified 2 months ago Viewed 295k times

163

I'm having problems understanding the difference between files produced by openssl and how to detect them.

For example I'm trying to generate Self-signed cert with private key and generate JKS file from p12 format. I'm googling like a madman but I still don't know how to generate it correctly to be able to use following commands.

```
openssl pkcs12 -export -in user.pem -inkey user.key -certfile user.pem -out
testkeystore.p12
keytool -importkeystore -srckeystore testkeystore.p12 -srcstoretype pkcs12 -
destkeystore wso2carbon.jks -deststoretype JKS
```

Source: https://www.ibm.com/support/pages/how-generate-jks-keystore-existing-private-key

I found a couple of different commands to generate Self-signed cert and private key but I don't know how to map resulting files to the commands above and whats worse I don't understand what those commands do. I mean I see what files they generate and understand that certificate and private key used to sign it (or maybe the other way around :|) but what is the difference between those commands and is cert.pem === certificate.crt - Those file extensions are driving me crazy.

```
openssl req -x509 -sha256 -nodes -days 365 -newkey rsa:2048 -keyout
privateKey.key -out certificate.crt
openssl req -newkey rsa:2048 -new -nodes -x509 -days 3650 -keyout key.pem -out
cert.pem
```

This is yet another situation where I'm having similar issues with the openssl command. At this point I'm even ready to read some RFC (I hope it won't come to this :))

ssl openssl certificate pkcs#12 jks

Share Improve this question Follow

edited Apr 20, 2024 at 9:10
BenMorel
36.7k ● 52 ● 205 ● 337

asked Jul 31, 2020 at 16:02
sebastian_t
2,889 ● 6 ● 27 ● 41

2 Informational note: originally Java by default required JKS-format keystore, and thus there are many websites manuals and other documentation giving advice like the IBM link above, as well as many older As on this and other Stacks. Java versions since 8u60 in 2015 can use PKCS12 files as well as JKS, and the conversion step is no longer necessary. – dave_thompson_085 Dec 24, 2022 at 20:51

4 Answers

Sorted by: Highest score (default)

227

Those file names represent different parts of the key generation and verification process. Please note that the names are just convention, you could just as easily call the files `pepperoni.pizza` and the content will be the same, so do be conscious of how you use the filenames.

A brief primer on PKI - Keys come in two halves, a public key and a private key. The public key can be distributed publicly and widely, and you can use it to verify, but not replicate, information generated using the private key. The private key must be kept secret.

`.key` files are generally the private key, used by the server to encrypt and package data for verification by clients.

`.pem` files are generally the public key, used by the client to verify and decrypt data sent by servers. PEM files could also be encoded private keys, so check the content if you're not sure.

`.p12` files have both halves of the key embedded, so that administrators can easily manage halves of keys.

`.cert` or `.crt` files are the signed certificates -- basically the "magic" that allows certain sites to be marked as trustworthy by a third party.

`.csr` is a certificate signing request, a challenge used by a trusted third party to verify the ownership of a keypair without having direct access to the private key (this is what allows end users, who have no direct knowledge of your website, confident that the certificate is valid). In the self-signed scenario you will use the certificate signing request with your own private key to verify your private key (thus self-signed). Depending on your specific application, this might not be needed. (needed for web servers or RPC servers, but not much else).

A JKS keystore is a native file format for Java to store and manage some or all of the components above, and keep a database of related capabilities that are allowed or rejected for each key.

The commands you list look fine to me, and I don't see a question beyond asking what the different files are for. If you need more information, please enrich your question.

Share Improve this answer Follow

edited Jan 18, 2023 at 10:35
Bruno Bieri
10.3k ● 11 ● 66 ● 96

answered Jul 31, 2020 at 16:15
PaulProgrammer
17.7k ● 4 ● 45 ● 60

1 Thanks for the thorough reply. After running openssl req -x509 ... command I got 2 files that contents begin with -----BEGIN one has PRIVATE KEY and other CERTIFICATE next. On stackoverflow.com/questions/991758/… I found that -----BEGIN means that I'm dealing with PEM format. Does this apply here to both of those files? – sebastian_t Jul 31, 2020 at 17:03

2 Yes, PEM format, but by convention, the one that says "PRIVATE KEY" is usually named `.key`. – PaulProgrammer Aug 1, 2020 at 3:03

33 According to this answer, `.crt` keeps a signed certificate, whereas `.csr` is the certificate signing request. Also, `.pem` just indicates that the content (can be a key, certificate, ...) is Base64 encoded. – Wolfson Apr 29, 2021 at 9:50

1 `.key files are generally the private key, used by the server to encrypt` Isn't encryption normallly done by a public key and decryption done by a private key? You're saying the exact opposite here. – Mugen Feb 24, 2022 at 4:23

@Mugen PKI is weird in this way. Server encrypts to the public key using the private key for verification. Client uses the public key to encrypt data to the server for privacy. – PaulProgrammer Mar 6, 2022 at 17:36

118

`.key` is the **private key.** This is accessible the key owner and no one else.

`.csr` is the **certificate request.** This is a request for a certificate authority to sign the key. (The key itself is not included.)

`.crt` is the **certificate** produced by the certificate authority that verifies the authenticity of the key. (The key itself is not included.) This is given to other parties, e.g. HTTPS client.

`.pem` is a text-based container using base-64 encoding. It could be any of the above files.

```
-----BEGIN EXAMPLE-----
...
-----END EXAMPLE-----
```

`.p12` is a PKCS12 file, which is a container format usually used to combine the private key and certificate.

There isn't only one extension. For example you may see certificates with either the `.crt` or a `.pem` extension.

Share Improve this answer Follow

answered Feb 7, 2022 at 20:17

Paul Draper
83.6k ● 53 ● 214 ● 301

Just to add more info: `.der`, another (binary) encoding (either public or private key, or csr)

7

Share Improve this answer Follow

answered Jan 8, 2023 at 0:27
Joseph Riopelle
179 ● 2 ● 6

0

After I've created this question a few years has passed and I had to generate an SSL certificate for docker and it turned out once again I had huge problems wrapping my head around it.

I've decided to create a cli generator as a practice (in PHP but wrapped in docker) and it definitely helped me better understand what I was dealing with. If after going through all the answers you are still having problems I would recommend checking out an advanced example in my project https://github.com/tarach/self-signed-ssl-generator?tab=readme-ov-file#advanced-examples because file types are only half of the fun if it comes to SSL certs. 😅

Share Improve this answer Follow

answered Jan 19 at 15:38
sebastian_t
2,889 ● 6 ● 27 ● 41

Start asking to get answers

Find the answer to your question by asking.

Ask question

Explore related questions

ssl openssl certificate pkcs#12 jks

See similar questions with these tags.