Import private key and certificate into Java Key Store (JKS)

Apache Tomcat and many other Java applications expect to retrieve SSL/TLS certificates from a Java Key Store (JKS). Jave Virtual Machines usually come with keytool[1] to help you create a new key store.

Keytool helps you to:

- create a new JKS with a new private key
- generate a Certificate Signung Request (CSR) for the private key in this JKS
- import a certificate that you received for this CSR into your JKS

Keytool **does not** let you import an existing private key for which you already have a certificate. So you need to do this yourself, here's how:

Let's assume you have a private key (**key.pem**) and a certificate (**cert.pem**), both in PEM format as the file names suggest.

PEM format is 'kind-of-human-readable' and looks like e.g.

```
-----BEGIN CERTIFICATE-----
Ulv6GtdFbjzLeqlkelqwewlq822OrEPdH+zxKUkKGX/eN
.
. (snip)
.
9801asds3BCfu52dm7JHzPAOqWKaEwIgymlk=
----END CERTIFICATE-----
```

Convert both, the key and the certificate into DER format using openssl[2]:

```
openssl pkcs8 -topk8 -nocrypt -in key.pem -inform PEM -out key.der -outform DER
openssl x509 -in cert.pem -inform PEM -out cert.der -outform DER
```

Now comes the tricky bit, you need something to import these files into the JKS. ImportKey will do this for you, get the ImportKey.java[3] source or the compiled (Java 1.5 !) ImportKey.class[4] and run it like

```
user@host:~$ java ImportKey key.der cert.der
Using keystore-file : /home/user/keystore.ImportKey
One certificate, no chain.
Key and certificate stored.
Alias:importkey  Password:importkey
```

Now we have a proper JKS containing our private key and certificate in a file called keystore.ImportKey, using 'importkey' as alias and also as password. For any further changes, like changing the password we can use keytool.

1. http://java.sun.com/j2se/1.5.0/docs/tooldocs/windows/keytool.html
2. http://www.openssl.org/
3. daisy:80 (ImportKey.java)
4. daisy:81 (ImportKey.class)