**PLEASE PROVIDE CERT ON FILE IN**
**.CRT .CER or .PEM FORMAT**

## How to generate private key and certificate on macOs Mojave and LINUX

Open terminal

Locate user folder:  cd ~/ Run command:

openssl req -new -newkey rsa:4096 -days 720 -nodes -x509 -sha256 -keyout superSecretPrivateKey.pem -out sendMeToCitadeleCert.pem

You are about to be asked to enter information that will be incorporated into your certificate request.

What you are about to enter is what is called a Distinguished Name or a DN.

Country Name (2 letter code) []:
[example] LV
State or Province Name (full name) []:
[example] Latvia
Locality Name (eg, city) []:
[example] Riga
Organization Name (eg, company) []:
[example] Sia ABC
Organizational Unit Name (eg, section) []:
[example] IT
Common Name (eg, fully qualified host name) []:
[example] example.com Email
Address []:
[example] janis.berzins@abc.lv

This example will create private key (superSecretPrivateKey.pem) and certificate (sendMeToCitadeleCert.pem)

**How to generate public key (certificate)  using Java kyetool (Windows)**

Instruction and short description about certificate generation uisng Java keytool you may found here:
http://docs.oracle.com/javase/7/docs/technotes/tools/solaris/keytool.html

Citadele bank requirements:

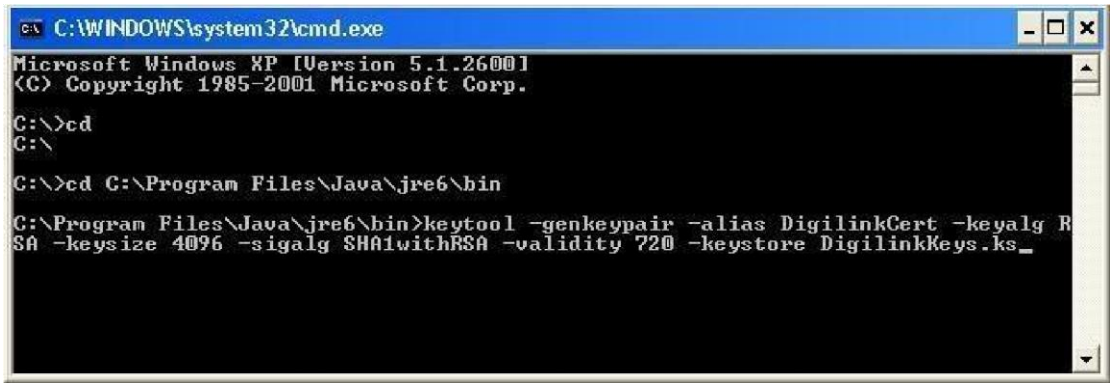
| | |
|---|---|
| Algorithm: | SHA256withRSA |
| Public key code algorithm: | RSA |
| Public key lengh: | 4096 |
| Max validation: | 4 years |

## 1. First step to generate sertificate

Call the command line and sign the following request:

keytool -genkeypair -alias [the name of the certificate in the certificate store] -keyalg RSA - keysize 4096 -sigalg SHA256withRSA -validity [sertificate validation days -keystore [sertificate store name ks]

Example: keytool -genkeypair -alias DigilinkCert -keyalg RSA -keysize 4096 -sigalg -SHA256withRSA -validity 720 keystore DigilinkKeys.ks




The program will prompt you to enter the information required to obtain a certificate –

Enter keystore password: ****** Password of sertificate store (min 6 simbols)
Re-enter new password: ******  Repeat the password
What is your first and last name? (name and surname)
 [Unknown]:  Janis Berzins
What is the name of your organizational unit? (department name)
 [Unknown]:  IT
What is the name of your organization? (company name)
 [Unknown]:  SIA ABC
What is the name of your City or Locality? (city)
 [Unknown]:  Riga
What is the name of your State or Province? (country)
 [Unknown]:  Latvia
What is the two-letter country code for this unit? (country code)
 [Unknown]:  LV
Is CN=Janis Berzins, OU=IT, O=SIA ABC, L=Riga, ST=Latvia, C=LV correct?
(confirm your choise)
 [no]:  yes (if all data are correct, accept YES)

Enter key password for <DigilinkCert> ****** (set the password of certificate)
(RETURN if same as keystore password):
Re-enter new password: ****** (repeat the password)

The result of this example will create a certificate repository for DigilinkKeys.ks  and the certificate is stored there DigilinkCert.

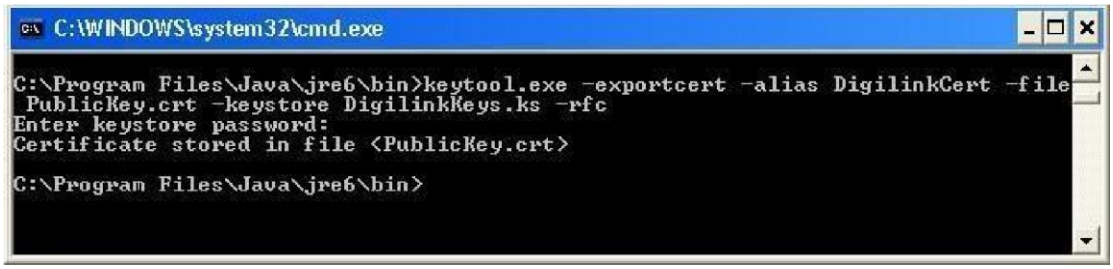
**2. The next step is to export from the certificate the public key to be sent to the Bank. The Public Key is required by the Bank to verify the authenticity of messages (payment requests) received from the Company.**

To do this, type the following command:

keytool.exe -exportcert -alias [the previously created certificate name in the

certificate store]
- file [The name
of the exported
key file.crt] -
keystore [The
name of the
previously
created
certificate
repository.ks] –
rfc

Example: keytool.exe -exportcert -alias DigilinkCert -file PublicKey.crt -keystore DigilinkKeys.ks
-rfc




The program will ask you to enter the certificate store password that you created in step 1:

Enter keystore password: ******
Certificate stored in file <PublicKey.crt> - the certificate is currently exported to the PublicKey.crt file in the directory you are working in and can be sent to the Bank as an email attachment. If the certificate system does not allow sending, it must be archived.

Fails izskatās sekojoši:

PublicKey.crt
Open fail the informtaion looks like this:

Fail opened in Notepad, looks like this:

-----BEGIN CERTIFICATE-----

MIIFRDCCAyygAwIBAgIEU6wWRTANBgkqhkiG9w0BAQUFADBkMQswCQYDVQQGEwJMVjEPMA0GA1UE

CBMGTGF0dmlhMQ0wCwYDVQQHEwRSaWdhMRAwDgYDVQQKEwdTSUEgQUJDMQswCQYDVQQLEwJJVDEW

MBQGA1UEAxMNSmFuaXMgQmVyemluczAeFw0xNDA2MjYxMjQ3MDFaFw0xNjA2MTUxMjQ3MDFaMGQx

CzAJBgNVBAYTAkxWMQ8wDQYDVQQIEwZMYXR2aWExDTALBgNVBAcTBFJpZ2

ExEDAOBgNVBAoTB1NJ

QSBBQkMxCzAJBgNVBAsTAklUMRYwFAYDVQQDEw1KYW5pcyBCZXJ6aW5zMII

CIjANBgkqhkiG9w0B

.....

Y6z4MD8pKTJNCr/ZWT7ii/57fJ9sVh+Iht/kqX/y6kIow30yPxLJ37

LPL+lPe3iyf/WpVoMNOmy5

V/3x7KL/Rv1ASaBuyjdIRuSb7e3a9MPBgT01xr/sOUl4jg2+4UZt mKuPqJx0OcVaInP

bWI8XnzOQ

0IkaDLQfnj/UsIdRcLLadW8KyVpbw+2qAhRw+p25gBx95vvDF

O6c/YkTUqUxaWmZ oj44/lUe6pf6

t0QA2e4Ay5zUYZHFdL42lT01qm2HGn4/M/E8Plw8QLIgcWoVr FNvE37Xm+N6JuM

w/w33NmBZ8ndM jRM976kfCBr+wxHlSuL0z9PZCZhTZDvzxVG70N/PEdTOLz9U

GRqmKl0= -----END CERTIFICATE-----