Guide to create and export custom certificates using XCA.

1. Download XCA : https://sourceforge.net/projects/xca/

2. After installing it you'll have to create a database within which the private keys and certificates will be stored.

File→New DataBase → set a file name on your disk, called for example test.xdb and a password to encrypt data within it.

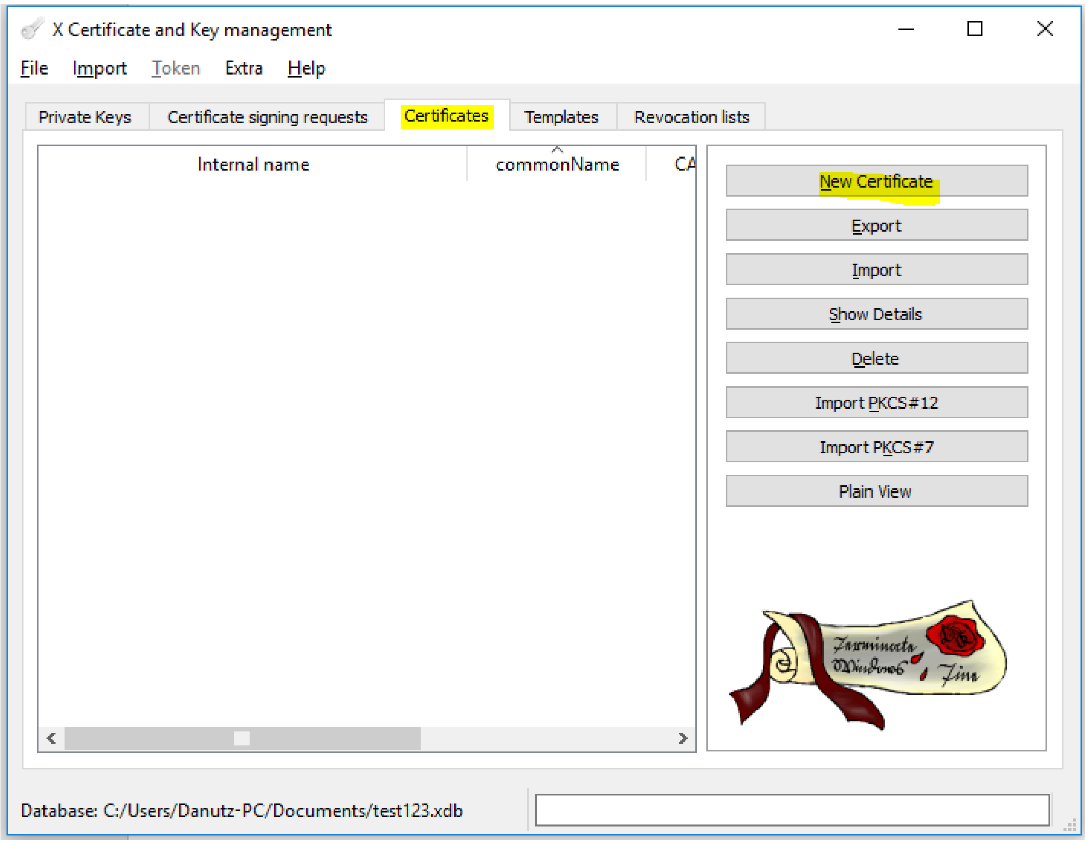
3. Open the new created xdb file.

4. We'll create a chain composed of 1 ROOT→ 1 CA→ and 2 user certificates:

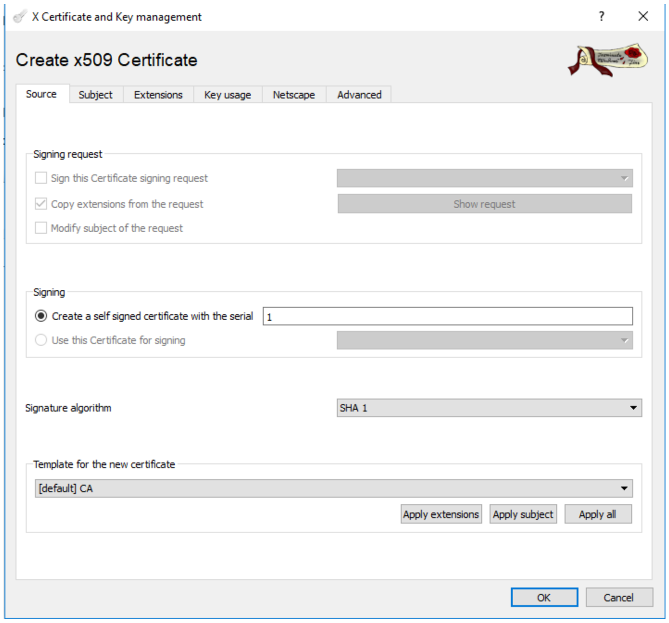
ROOT MyOrg_ROOT.crt
  CA MyOrg_CA.crt
    USER_SSL MyOrg_USER_SSL.p12
    USER_SSO MyOrg_USER_SSO.p12

a) ROOT

In the Certificates tab click on New Certificate



Database: C:/Users/Danutz-PC/Documents/test123.xdb



□ Sign this Certificate signing request

☑ Copy extensions from the request

□ Modify subject of the request

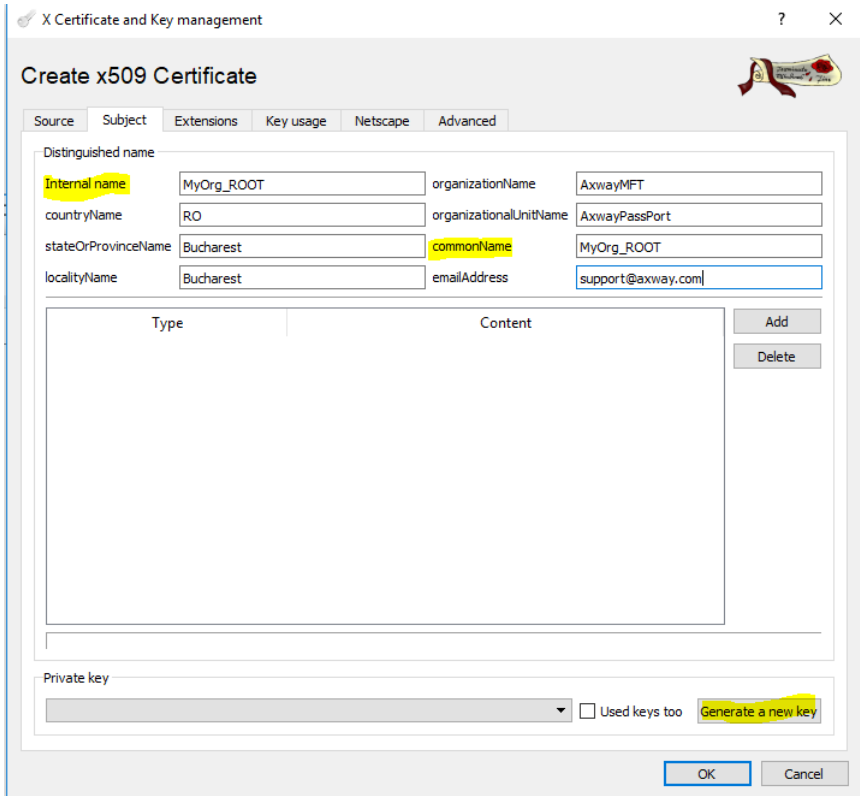
● Create a self signed certificate with the serial 1

○ Use this Certificate for signing

Signature algorithm SHA 1

Template for the new certificate [default] CA

Click on the second tab, **Subject** add the details of your ROOT certificate.
After adding the details we'll generate the Private key by clicking on "Generate a new key".



Distinguished name

Internal name: MyOrg_ROOT

countryName: RO

stateOrProvinceName: Bucharest

localityName: Bucharest

organizationName: AxwayMFT

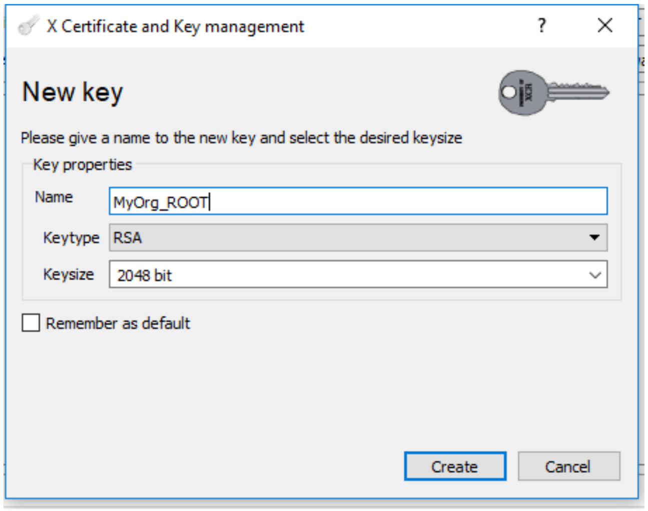
organizationalUnitName: AxwayPassPort

commonName: MyOrg_ROOT

emailAddress: support@axway.com

□ Used keys too

Generate a new key



New key

Please give a name to the new key and select the desired keysize

Key properties

Name: MyOrg_ROOT

Keytype: RSA

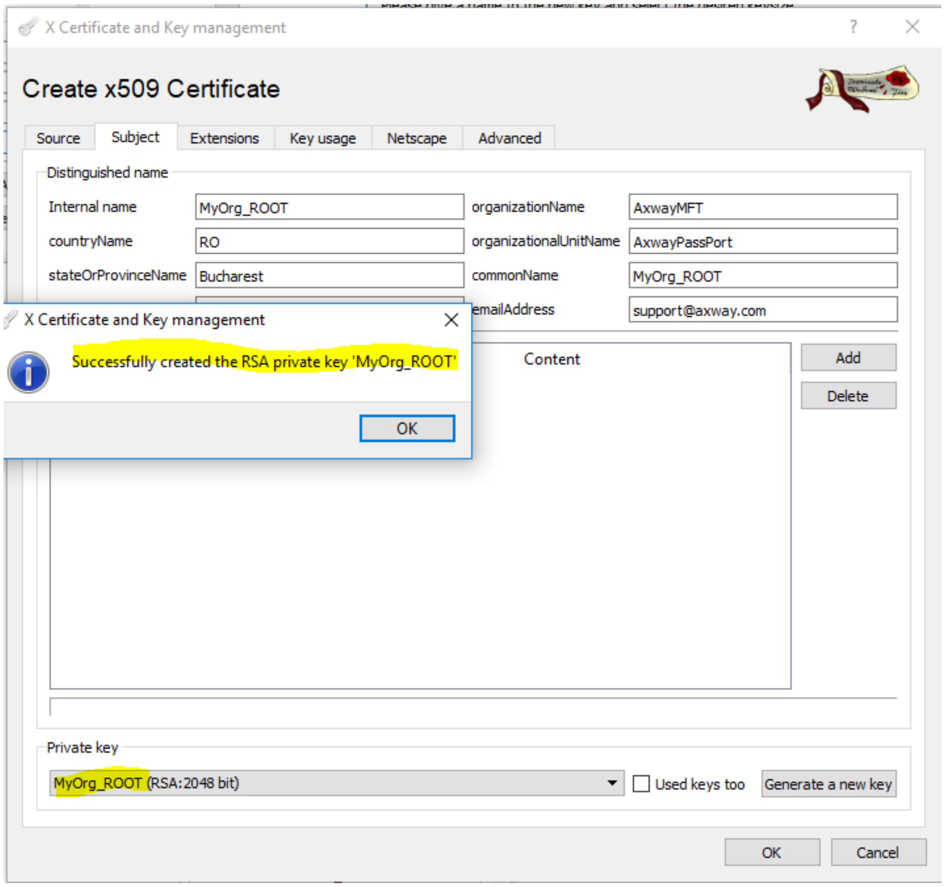
Keysize: 2048 bit

□ Remember as default

You can adjust the Name, Keytype and Keysize as per your need.In my case I’ll keep the default values.

Click on **Create**.

The private key for the MyOrg_ROOT has been created.



X Certificate and Key management

Create x509 Certificate

Distinguished name

Internal name	MyOrg_ROOT	organizationName	AxwayMFT
countryName	RO	organizationalUnitName	AxwayPassPort
stateOrProvinceName	Bucharest	commonName	MyOrg_ROOT
		emailAddress	support@axway.com

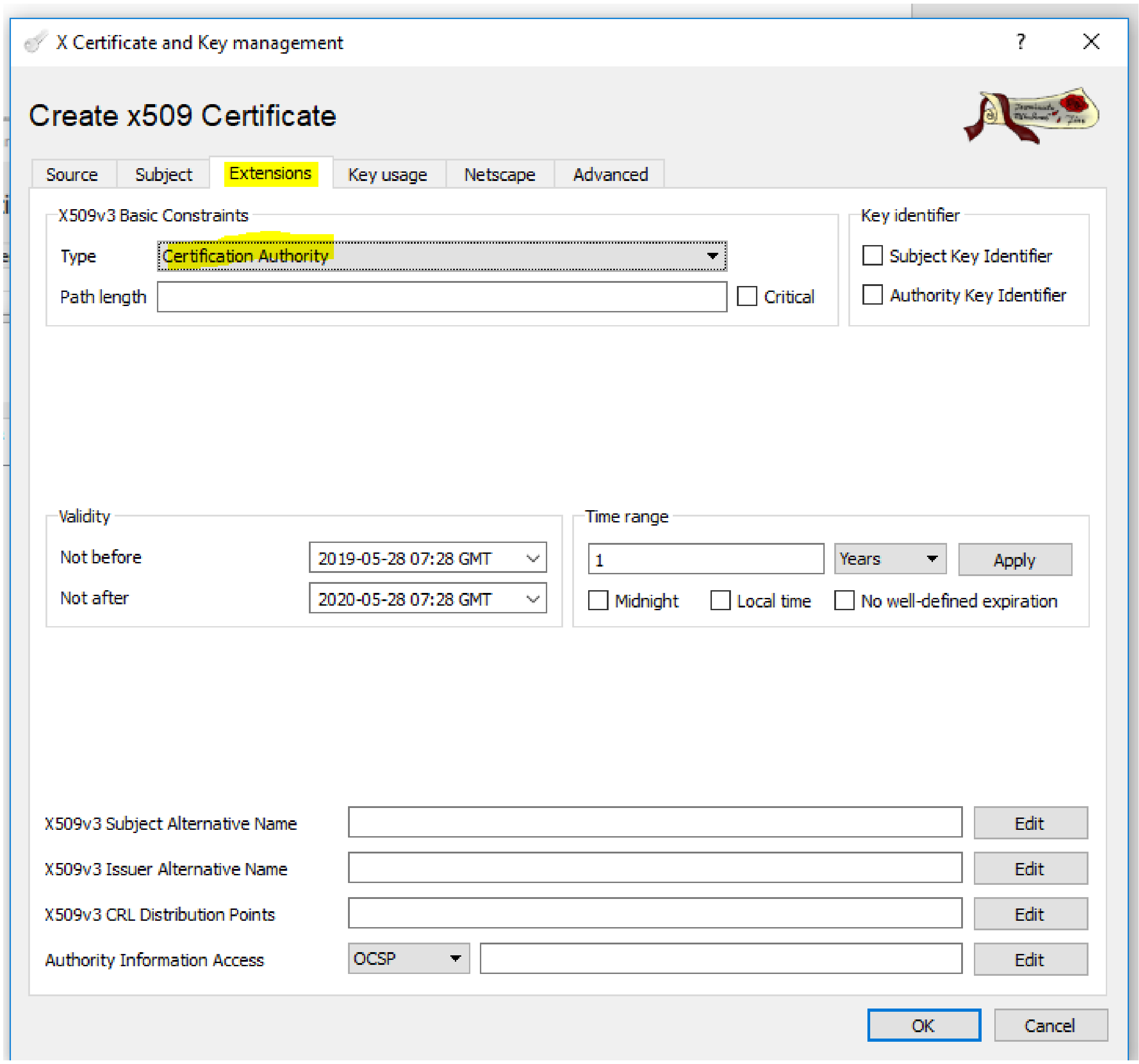
X Certificate and Key management

Successfully created the RSA private key 'MyOrg_ROOT'

Private key

MyOrg_ROOT (RSA:2048 bit)

Afterwards switch to the 3rd tab, **Extensions** and set the Type to **Certification Authority.** **Do not click on OK at this step.**



Create x509 Certificate

X509v3 Basic Constraints

Type: Certification Authority

Path length: □ Critical

Key identifier

□ Subject Key Identifier

□ Authority Key Identifier

Validity

Not before: 2019-05-28 07:28 GMT

Not after: 2020-05-28 07:28 GMT

Time range: 1 Years

□ Midnight □ Local time □ No well-defined expiration

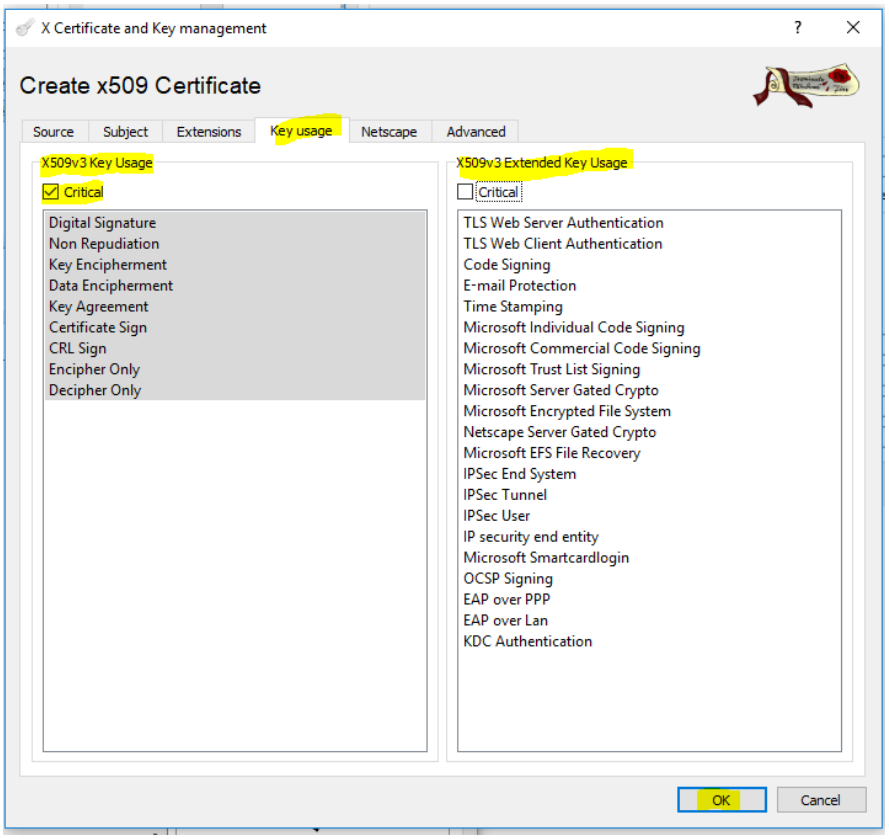
X509v3 Subject Alternative Name

X509v3 Issuer Alternative Name

X509v3 CRL Distribution Points

Authority Information Access: OCSP

Switch to the 4th tab,Key usage and set the needed, desired key usages and then **click OK.**



Create x509 Certificate

X509v3 Key Usage

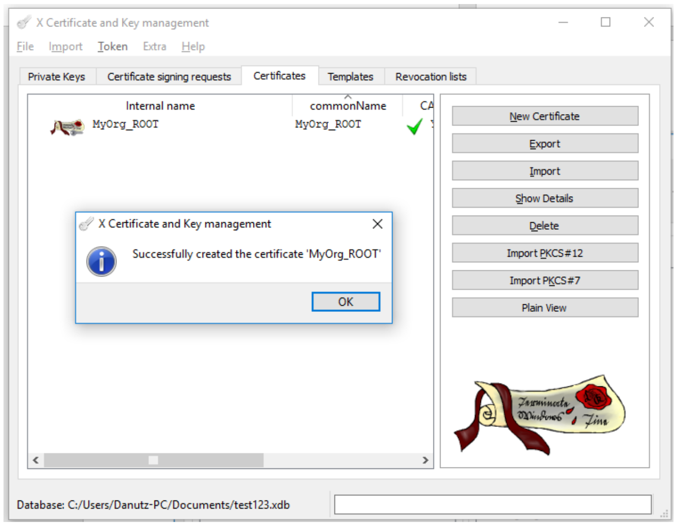
☑ Critical

Digital Signature
Non Repudiation
Key Encipherment
Data Encipherment
Key Agreement
Certificate Sign
CRL Sign
Encipher Only
Decipher Only

X509v3 Extended Key Usage

□ Critical

TLS Web Server Authentication
TLS Web Client Authentication
Code Signing
E-mail Protection
Time Stamping
Microsoft Individual Code Signing
Microsoft Commercial Code Signing
Microsoft Trust List Signing
Microsoft Server Gated Crypto
Microsoft Encrypted File System
Netscape Server Gated Crypto
Microsoft EFS File Recovery
IPSec End System
IPSec Tunnel
IPSec User
IP security end entity
Microsoft Smartcardlogin
OCSP Signing
EAP over PPP
EAP over Lan
KDC Authentication



Internal name	commonName	CA
MyOrg_ROOT	MyOrg_ROOT	✓

X Certificate and Key management

Successfully created the certificate 'MyOrg_ROOT'

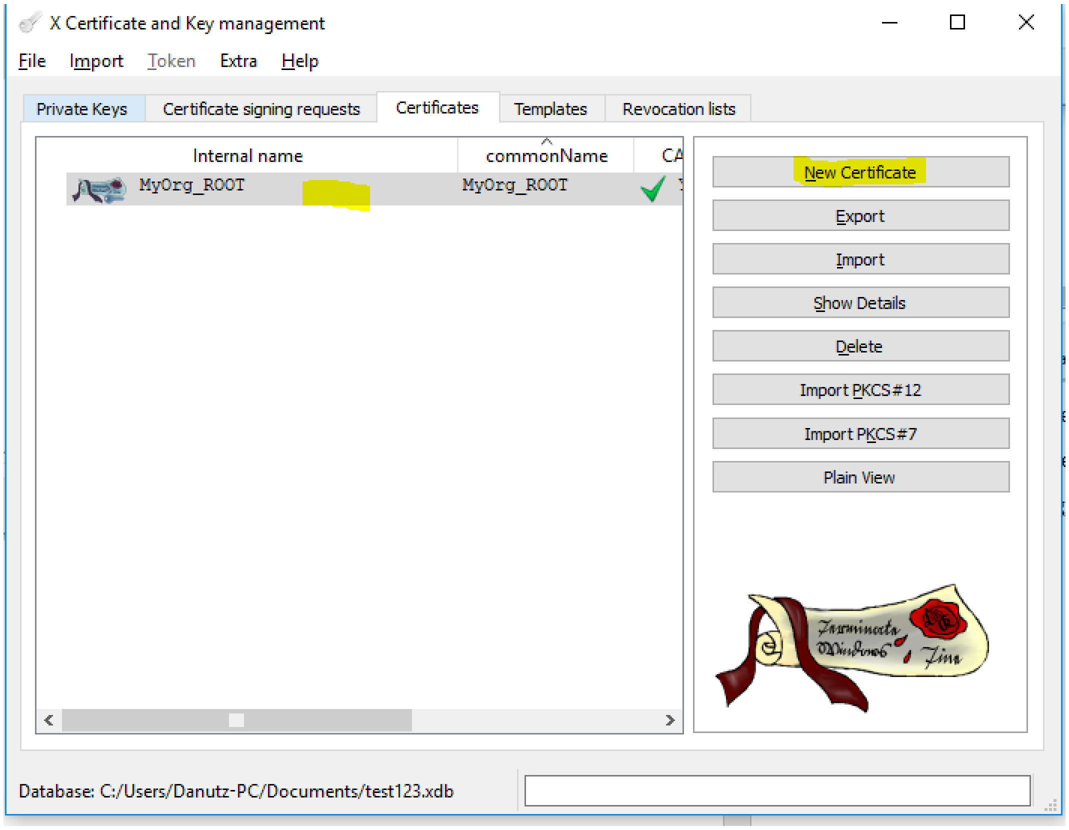
Database: C:/Users/Danutz-PC/Documents/test123.xdb

Congrats you've created the ROOT certificate.

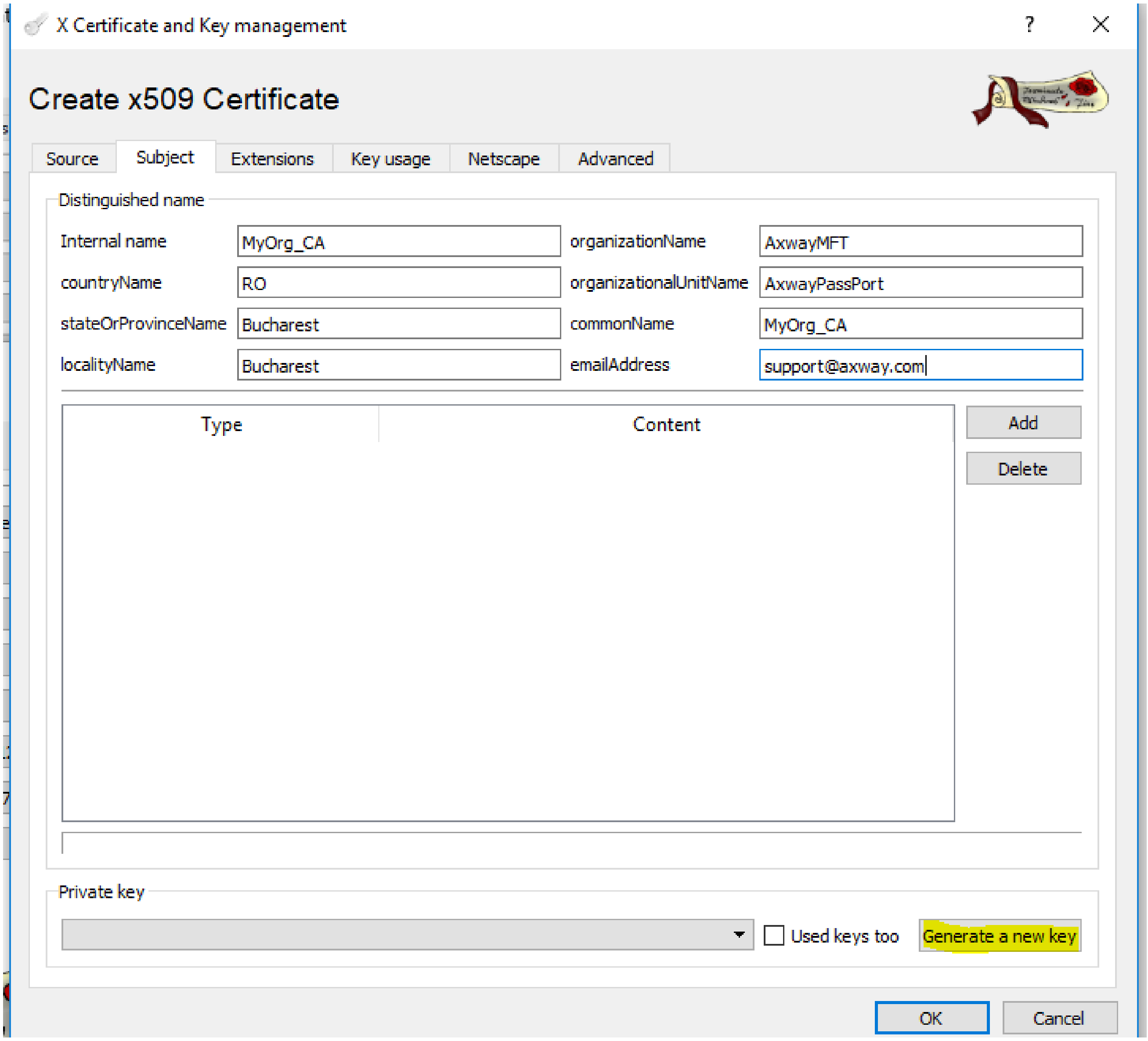
b) INTERMEDIATE CA

We'll proceed same way with the Intermediate CA certificate.

Click on the MyOrg_ROOT and then click the tab New Certificate to create the CA under it.



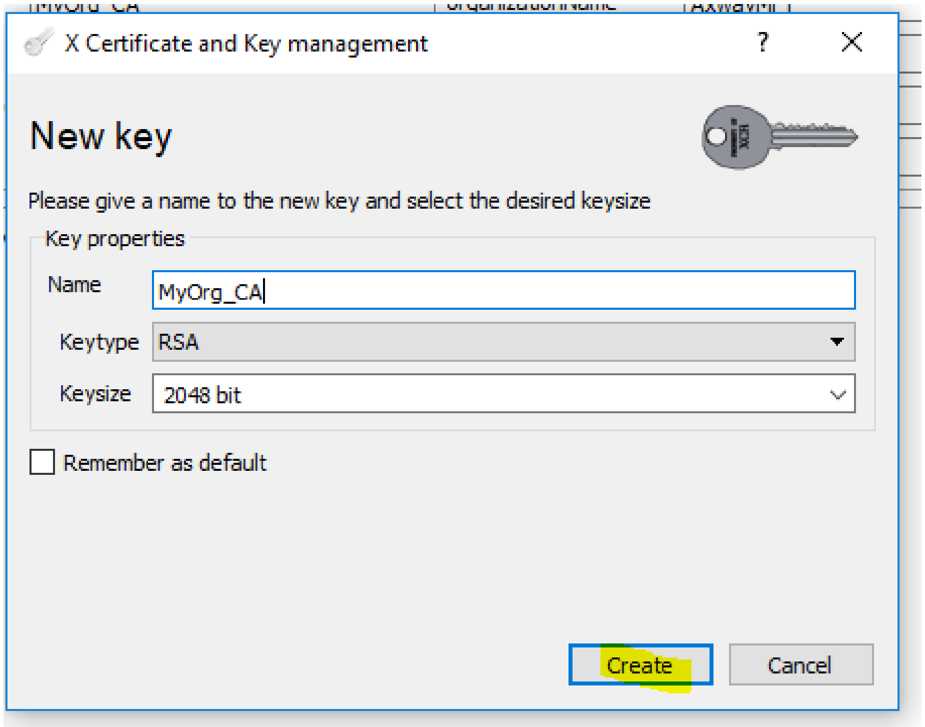
Click on Subject tab,add details of your Intermediate CA certificate, followed by Generate a new key:



Internal name	MyOrg_CA	organizationName	AxwayMFT
countryName	RO	organizationalUnitName	AxwayPassPort
stateOrProvinceName	Bucharest	commonName	MyOrg_CA
localityName	Bucharest	emailAddress	support@axway.com

☐ Used keys too

Click on Create and the private key for the Intermediate certificate will be created:



X Certificate and Key management

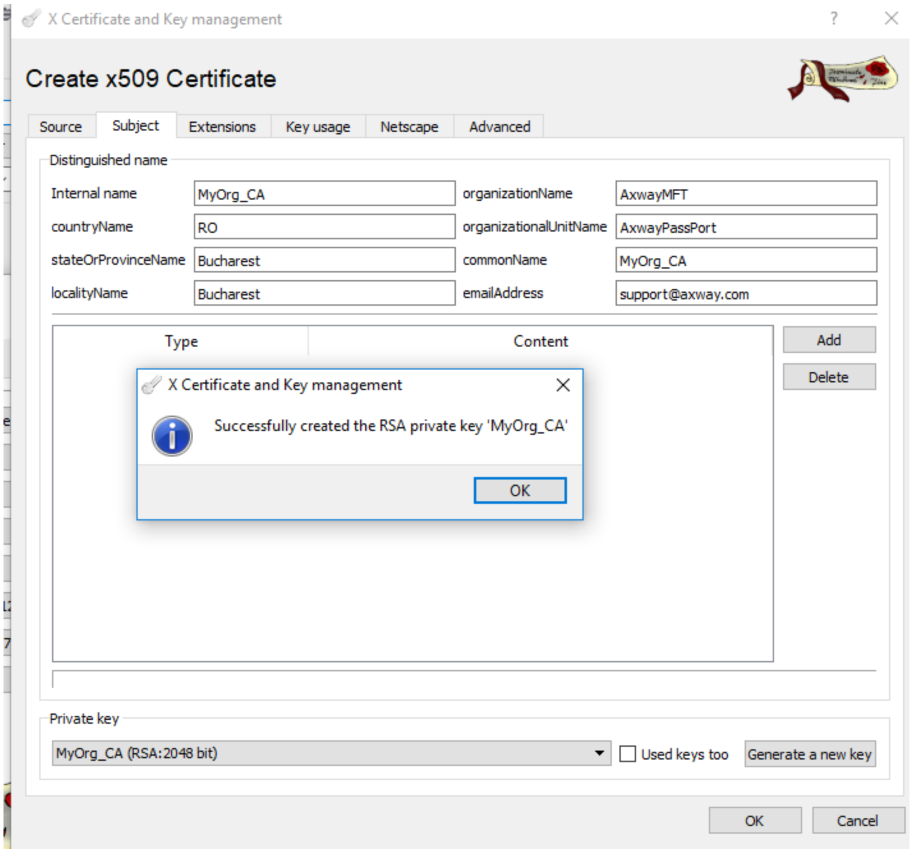
New key

Please give a name to the new key and select the desired keysize

Key properties

Name	MyOrg_CA
Keytype	RSA
Keysize	2048 bit

□ Remember as default



X Certificate and Key management

Create x509 Certificate

Distinguished name

Internal name	MyOrg_CA	organizationName	AxwayMFT
countryName	RO	organizationalUnitName	AxwayPassPort
stateOrProvinceName	Bucharest	commonName	MyOrg_CA
localityName	Bucharest	emailAddress	support@axway.com

X Certificate and Key management

Successfully created the RSA private key 'MyOrg_CA'

Private key

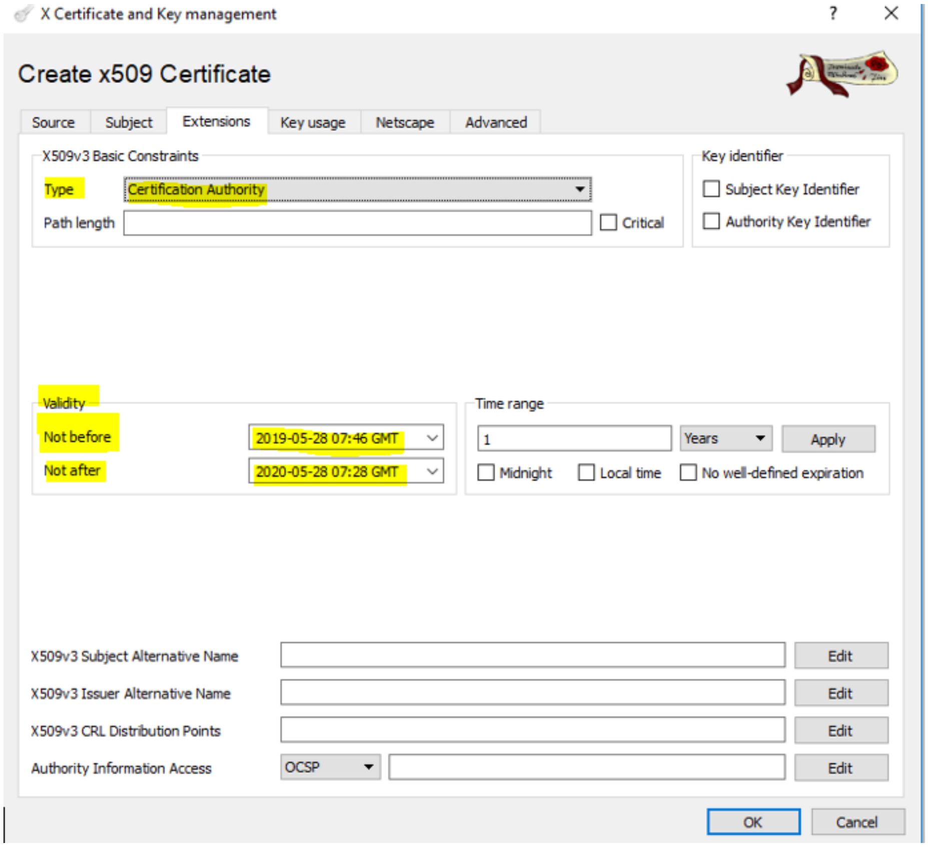
MyOrg_CA (RSA:2048 bit)

□ Used keys too

Then switch to the 3rd tab, Extensions and set Type as Certification Authority.

Note:As long as this certificate will not be the last one in the chain(entity) it will be set as CA(Certification Authority).Only the last certificate in the chain, the user one, will be set as the End Entity.

Validity of Certificate can be adjusted as well.



X Certificate and Key management

Create x509 Certificate

X509v3 Basic Constraints

Type: Certification Authority

Path length:

□ Critical

Key identifier

□ Subject Key Identifier

□ Authority Key Identifier

Validity

Not before: 2019-05-28 07:46 GMT

Not after: 2020-05-28 07:28 GMT

Time range: 1 Years

□ Midnight □ Local time □ No well-defined expiration

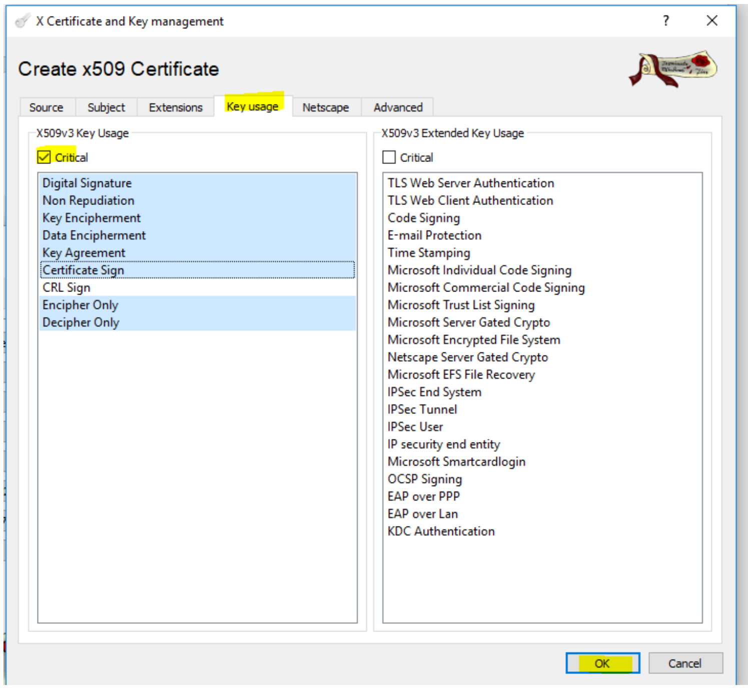
X509v3 Subject Alternative Name:

X509v3 Issuer Alternative Name:

X509v3 CRL Distribution Points:

Authority Information Access: OCSP

Set the Key usage and afterwards click OK to create the Intermediate CA certificate:



Create x509 Certificate

X509v3 Key Usage

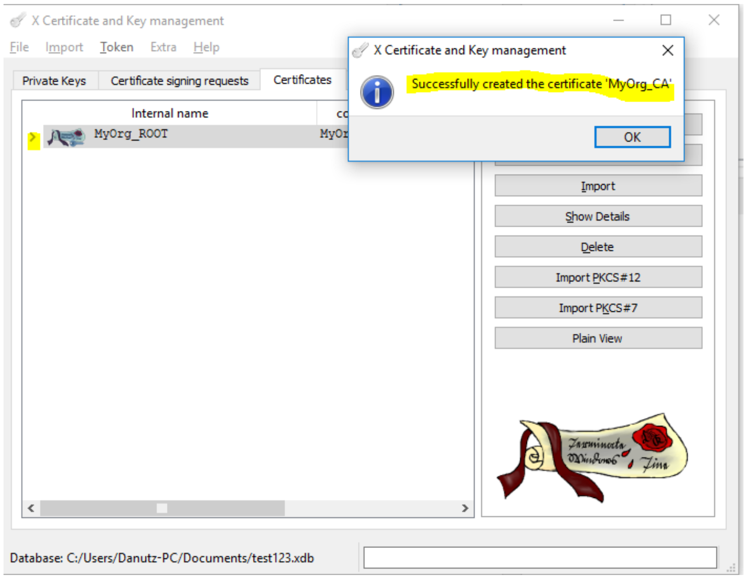
☑ Critical

Digital Signature
Non Repudiation
Key Encipherment
Data Encipherment
Key Agreement
Certificate Sign
CRL Sign
Encipher Only
Decipher Only

X509v3 Extended Key Usage

□ Critical

TLS Web Server Authentication
TLS Web Client Authentication
Code Signing
E-mail Protection
Time Stamping
Microsoft Individual Code Signing
Microsoft Commercial Code Signing
Microsoft Trust List Signing
Microsoft Server Gated Crypto
Microsoft Encrypted File System
Netscape Server Gated Crypto
Microsoft EFS File Recovery
IPSec End System
IPSec Tunnel
IPSec User
IP security end entity
Microsoft Smartcardlogin
OCSP Signing
EAP over PPP
EAP over Lan
KDC Authentication

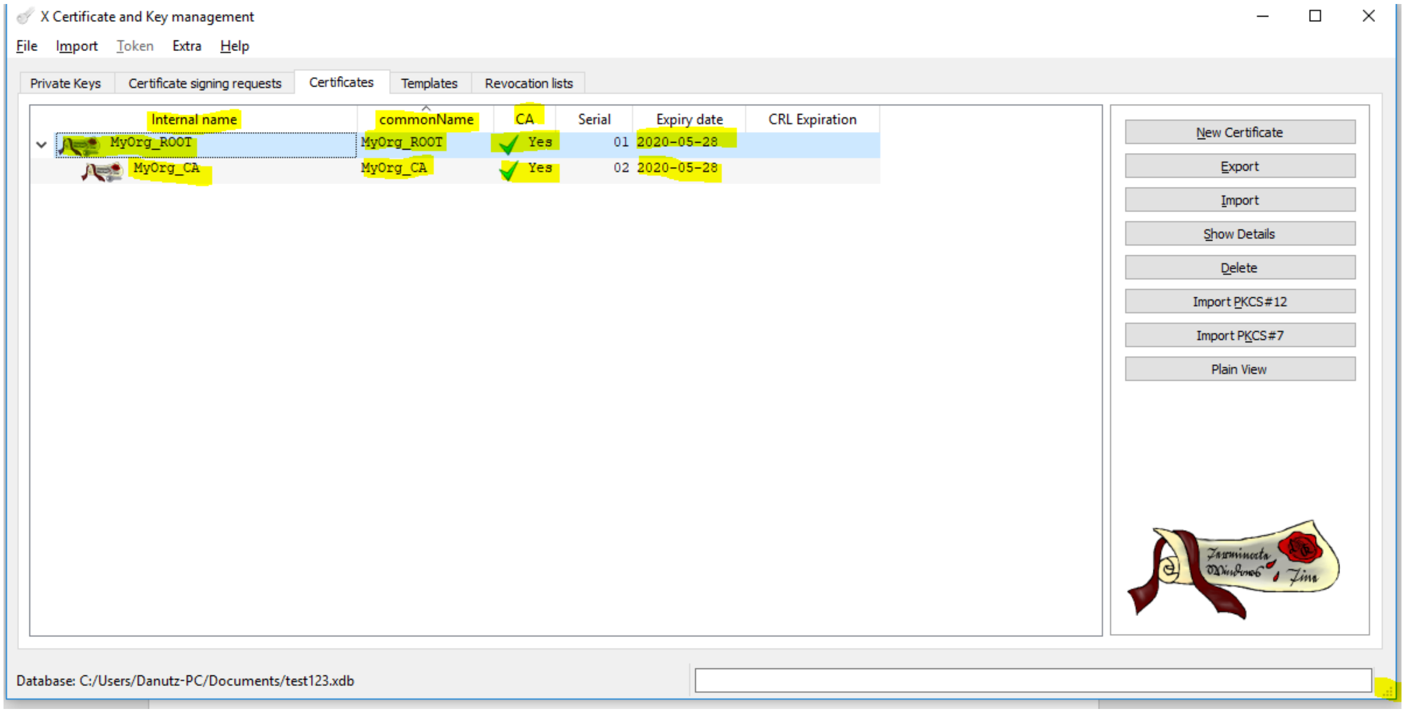


MyOrg_ROOT MyOr

Successfully created the certificate 'MyOrg_CA'

Database: C:/Users/Danutz-PC/Documents/test123.xdb

Click OK and afterwards expand the arrow and it should look like this:

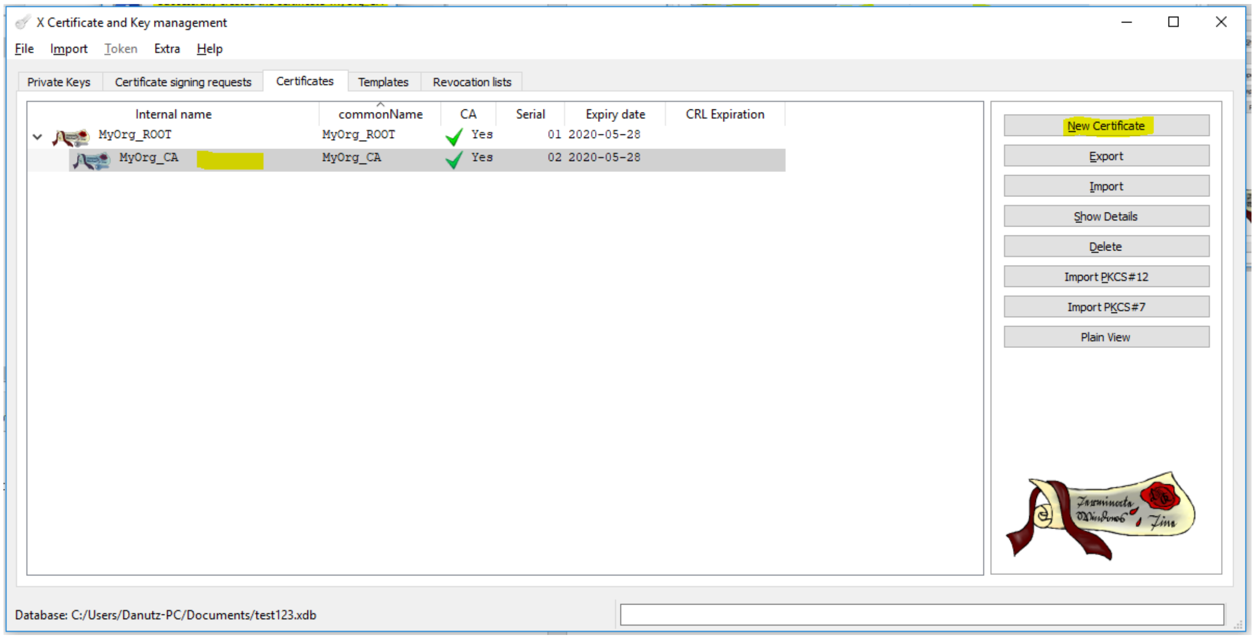


Internal name	commonName	CA	Serial	Expiry date	CRL Expiration
MyOrg_ROOT	MyOrg_ROOT	✓ Yes	01	2020-05-28	
MyOrg_CA	MyOrg_CA	✓ Yes	02	2020-05-28	

Database: C:/Users/Danutz-PC/Documents/test123.xdb

c. USER : Now let's create the USER certificate for the Passport SSL.

Click on the Intermediate Certificate and afterwards click on the New Certificate buton.


X Certificate and Key management

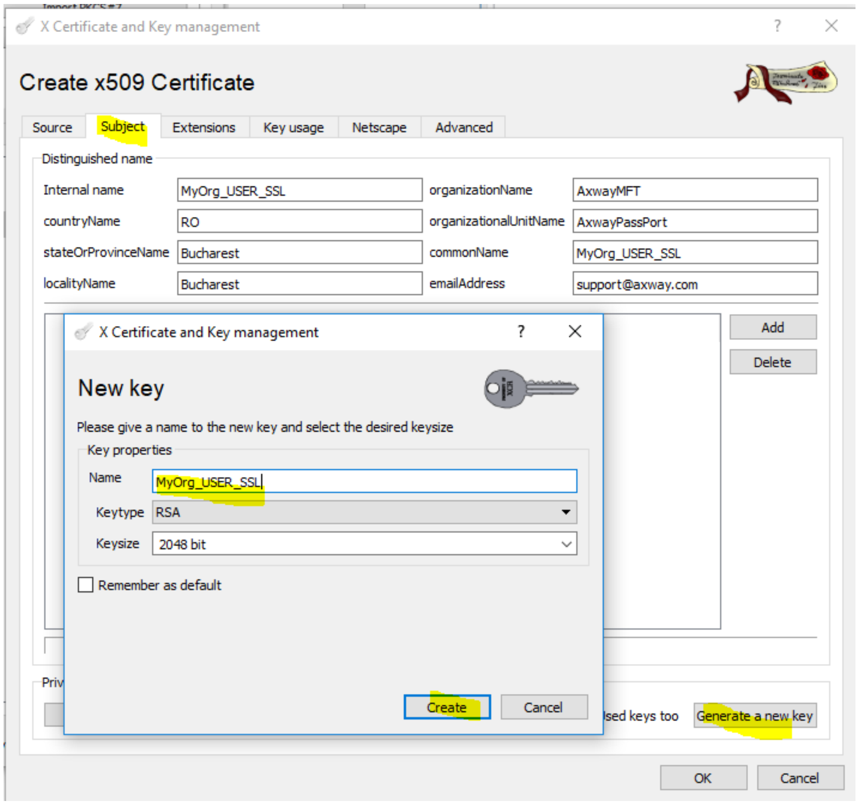
File Import Token Extra Help

Private Keys | Certificate signing requests | Certificates | Templates | Revocation lists

Internal name	commonName	CA	Serial	Expiry date	CRL Expiration
MyOrg_ROOT	MyOrg_ROOT	✓ Yes	01	2020-05-28	
MyOrg_CA	MyOrg_CA	✓ Yes	02	2020-05-28	

Database: C:/Users/Danutz-PC/Documents/test123.xdb

Add the details in the Subject tab and generate the Private key correspondent to this certificate.


X Certificate and Key management

Create x509 Certificate

Source | Subject | Extensions | Key usage | Netscape | Advanced

Distinguished name

Internal name	MyOrg_USER_SSL	organizationName	AxwayMFT
countryName	RO	organizationalUnitName	AxwayPassPort
stateOrProvinceName	Bucharest	commonName	MyOrg_USER_SSL
localityName	Bucharest	emailAddress	support@axway.com

X Certificate and Key management

New key

Please give a name to the new key and select the desired keysize

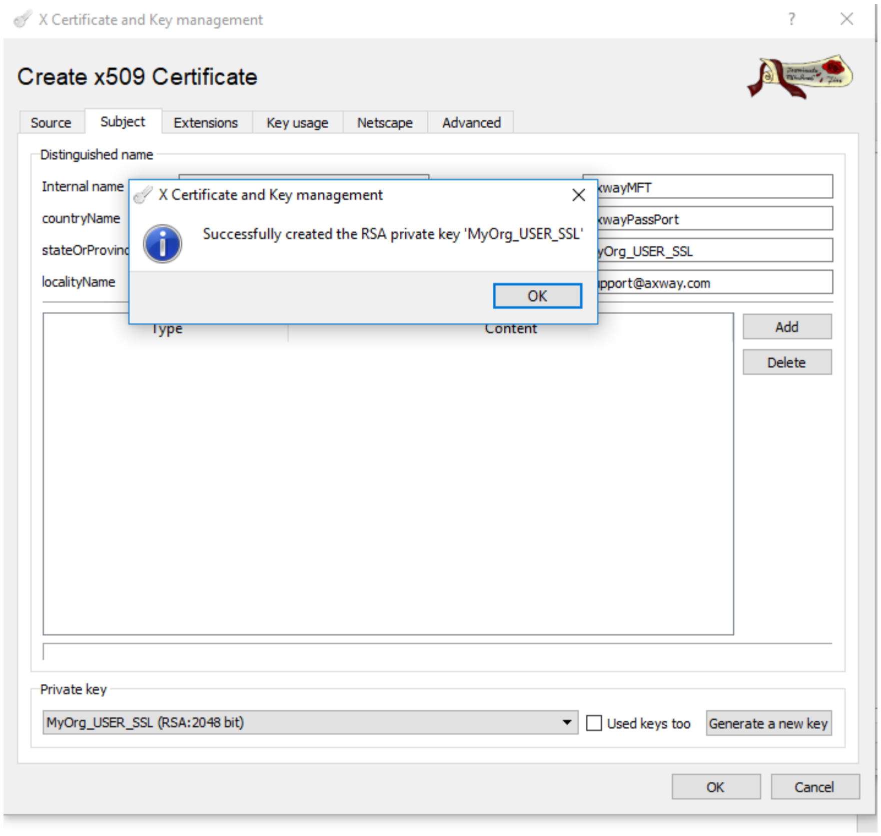
Key properties

Name: MyOrg_USER_SSL

Keytype: RSA

Keysize: 2048 bit

☐ Remember as default



X Certificate and Key management

Create x509 Certificate

Distinguished name

Internal name

countryName

stateOrProvinc

localityName

xwayMFT

xwayPassPort

yOrg_USER_SSL

pport@axway.com

X Certificate and Key management

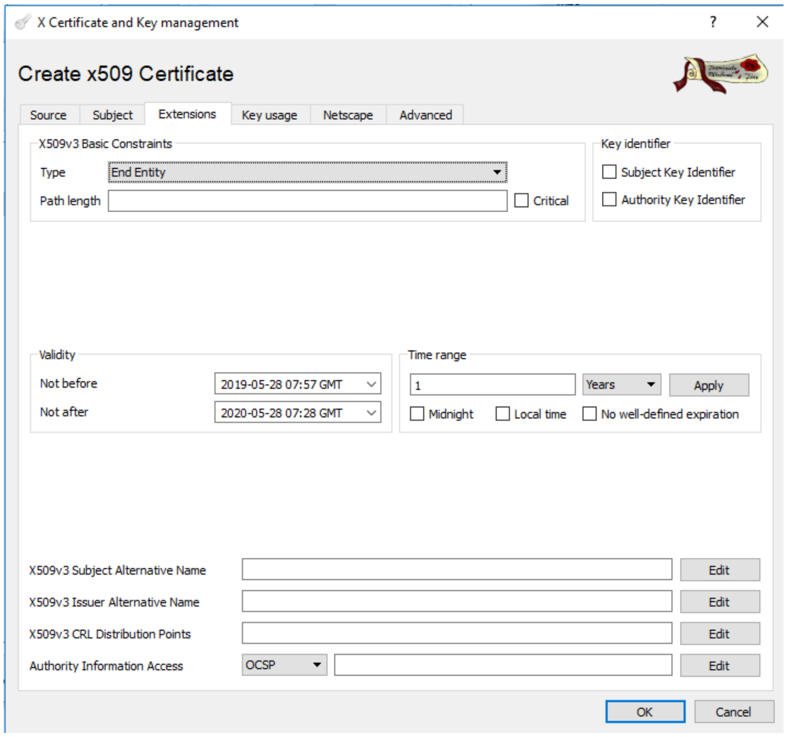
Successfully created the RSA private key 'MyOrg_USER_SSL'

Private key

MyOrg_USER_SSL (RSA:2048 bit)

□ Used keys too

In the 3rd tab, Extensions set the **Type** to **End Entity** being the last certificate in the chain :



X Certificate and Key management

Create x509 Certificate

X509v3 Basic Constraints

Type: End Entity

Path length

□ Critical

Key identifier

□ Subject Key Identifier

□ Authority Key Identifier

Validity

Not before: 2019-05-28 07:57 GMT

Not after: 2020-05-28 07:28 GMT

Time range

1 Years

□ Midnight □ Local time □ No well-defined expiration

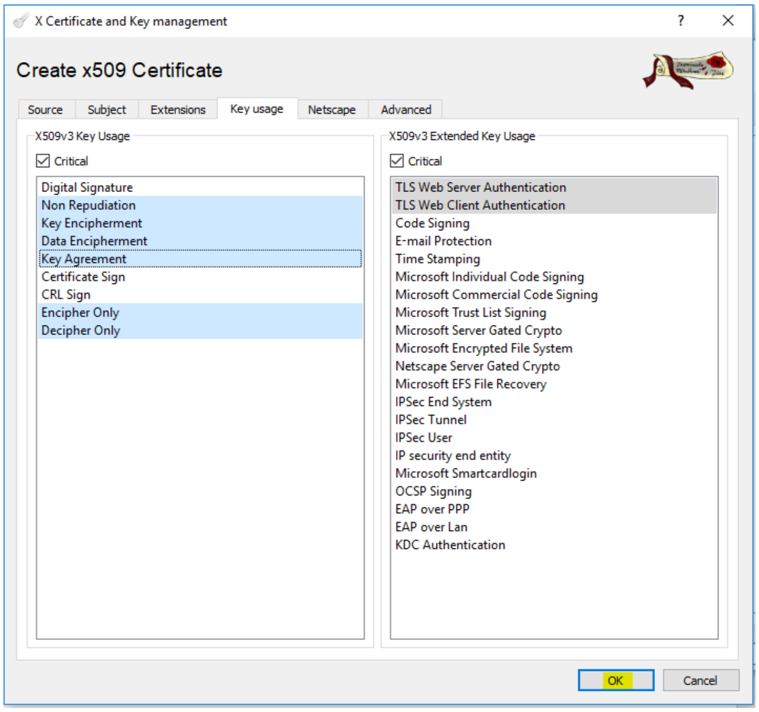
X509v3 Subject Alternative Name

X509v3 Issuer Alternative Name

X509v3 CRL Distribution Points

Authority Information Access: OCSP

Set the Key usages and the **click OK** to create it



Create x509 Certificate

X509v3 Key Usage

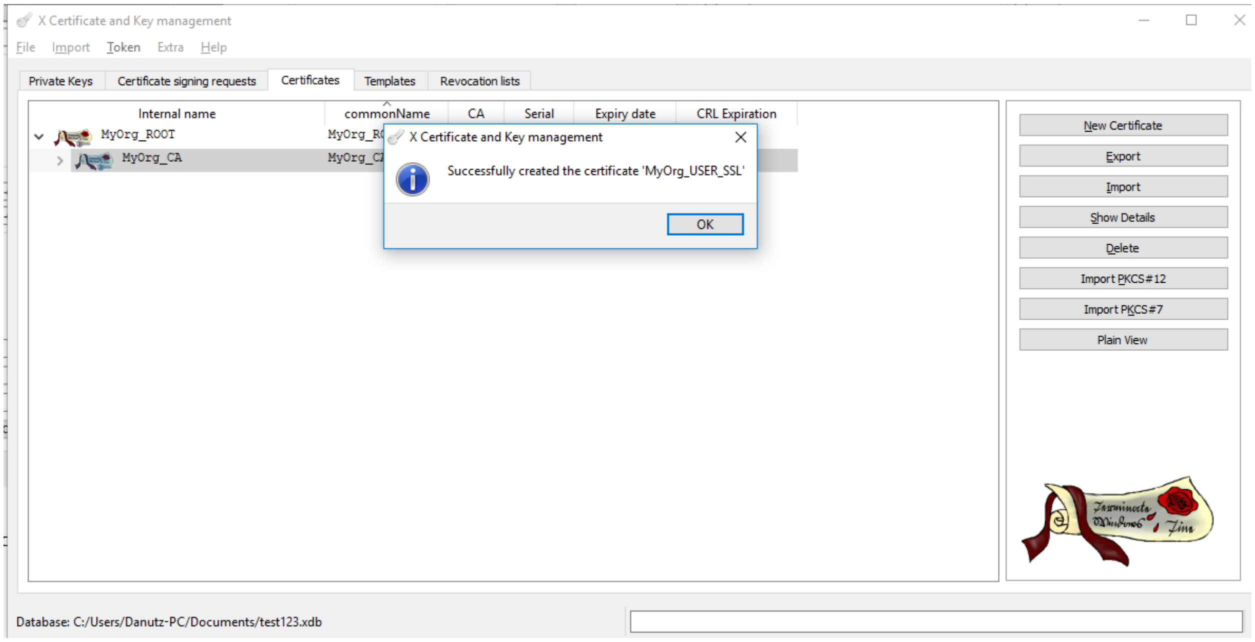
☑ Critical

- Digital Signature
- Non Repudiation
- Key Encipherment
- Data Encipherment
- Key Agreement
- Certificate Sign
- CRL Sign
- Encipher Only
- Decipher Only

X509v3 Extended Key Usage

☑ Critical

- TLS Web Server Authentication
- TLS Web Client Authentication
- Code Signing
- E-mail Protection
- Time Stamping
- Microsoft Individual Code Signing
- Microsoft Commercial Code Signing
- Microsoft Trust List Signing
- Microsoft Server Gated Crypto
- Microsoft Encrypted File System
- Netscape Server Gated Crypto
- Microsoft EFS File Recovery
- IPSec End System
- IPSec Tunnel
- IPSec User
- IP security end entity
- Microsoft Smartcardlogin
- OCSP Signing
- EAP over PPP
- EAP over Lan
- KDC Authentication



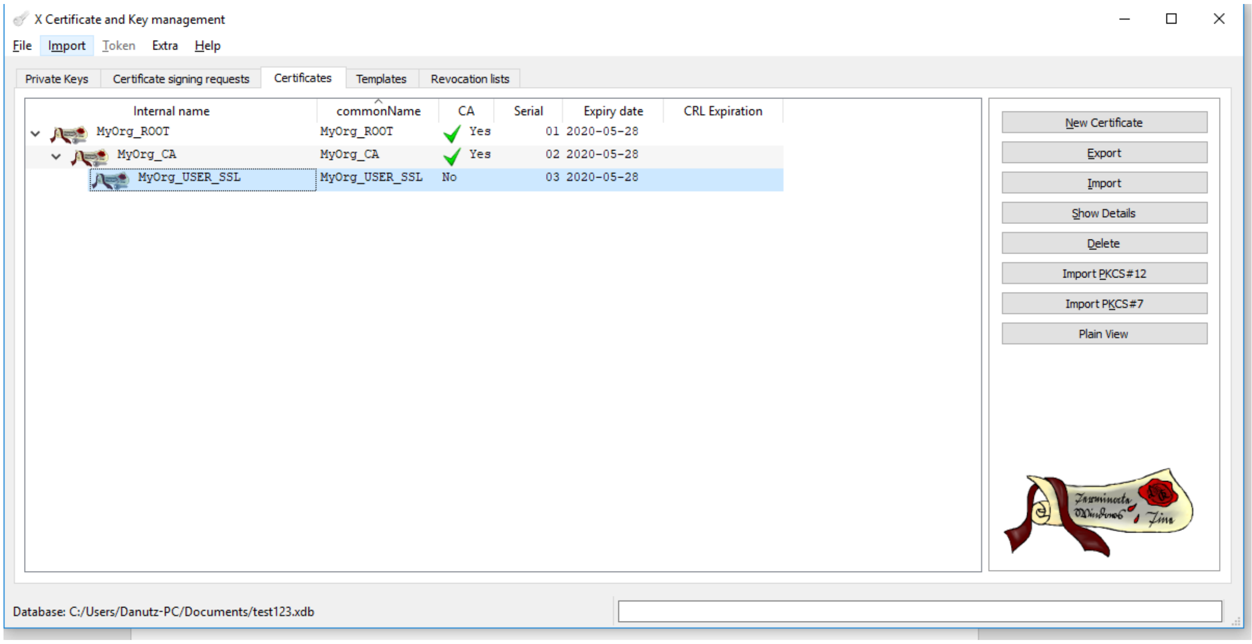
Internal name	commonName	CA	Serial	Expiry date	CRL Expiration
MyOrg_ROOT	MyOrg_R				
MyOrg_CA	MyOrg_CA				

X Certificate and Key management

Successfully created the certificate 'MyOrg_USER_SSL'

Database: C:/Users/Danutz-PC/Documents/test123.xdb

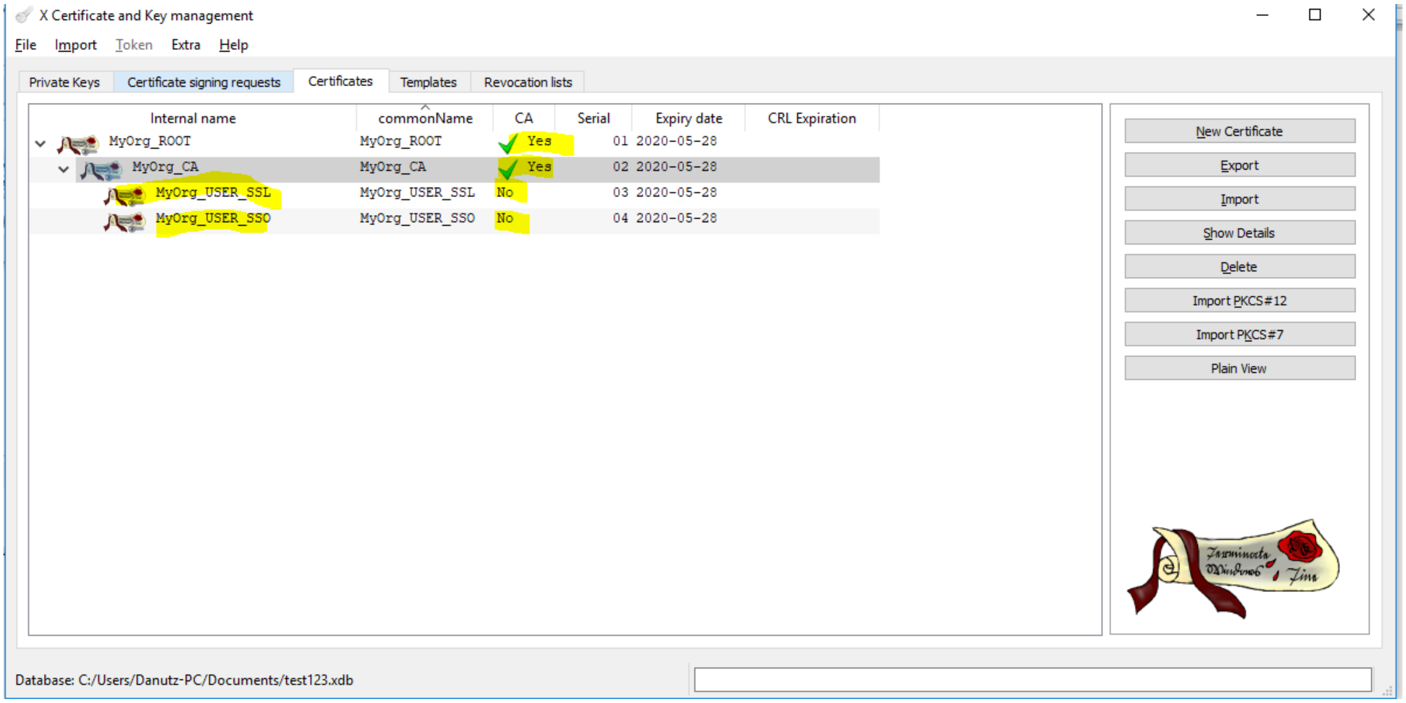
Expand the second arrow and you’ll see the new MyOrg_USER_SSL under the MyOrg_CA.



Internal name	commonName	CA	Serial	Expiry date	CRL Expiration
MyOrg_ROOT	MyOrg_ROOT	✓ Yes	01	2020-05-28	
MyOrg_CA	MyOrg_CA	✓ Yes	02	2020-05-28	
MyOrg_USER_SSL	MyOrg_USER_SSL	No	03	2020-05-28	

Database: C:/Users/Danutz-PC/Documents/test123.xdb

Redo the same step for the MyOrg_USER_SSO , step c).



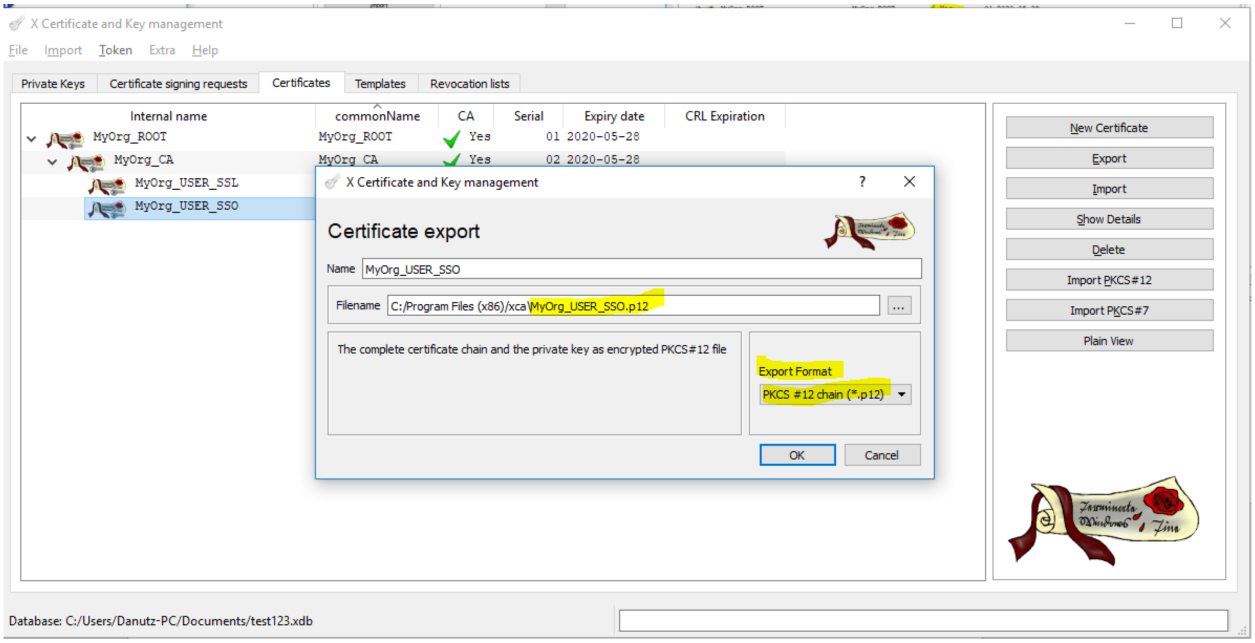
Internal name	commonName	CA	Serial	Expiry date	CRL Expiration
MyOrg_ROOT	MyOrg_ROOT	✓ Yes	01	2020-05-28	
MyOrg_CA	MyOrg_CA	✓ Yes	02	2020-05-28	
MyOrg_USER_SSL	MyOrg_USER_SSL	No	03	2020-05-28	
MyOrg_USER_SSO	MyOrg_USER_SSO	No	04	2020-05-28	

Database: C:/Users/Danutz-PC/Documents/test123.xdb

Export the certificates

We’ll export the USER SSL and SSO certificates in private format, pkcs12.

Right click on the MyOrg_USER_SSO →Export→File→Set the Export format for PKCS #12 chain (*.p12) , click OK



Internal name	commonName	CA	Serial	Expiry date	CRL Expiration
MyOrg_ROOT	MyOrg_ROOT	✓ Yes	01	2020-05-28	
MyOrg_CA	MyOrg_CA	✓ Yes	02	2020-05-28	
MyOrg_USER_SSL					
MyOrg_USER_SSO					

X Certificate and Key management

Certificate export

Name: MyOrg_USER_SSO

Filename: C:/Program Files (x86)/xca\MyOrg_USER_SSO.p12

The complete certificate chain and the private key as encrypted PKCS#12 file

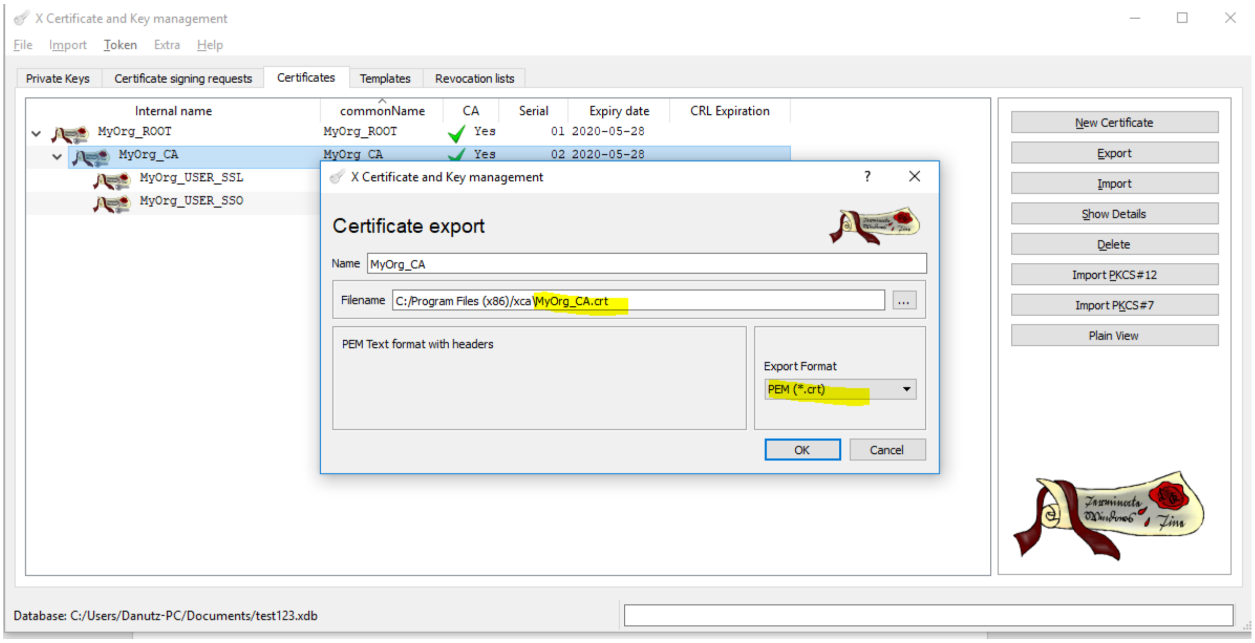
Export Format: PKCS #12 chain (*.p12)

Database: C:/Users/Danutz-PC/Documents/test123.xdb

Enter a password to encrypt the PCKS12 file.This password will be used/needed when importing the certificate in the sso.jks file.

Same procedure for the MyOrg_USER_SSL certificate, used for ssl.jks

We’ll export the ROOT certificate in public format , .pem or .crt



Internal name	commonName	CA	Serial	Expiry date	CRL Expiration
MyOrg_ROOT	MyOrg_ROOT	✓ Yes	01	2020-05-28	
MyOrg_CA	MyOrg_CA	✓ Yes	02	2020-05-28	
MyOrg_USER_SSL					
MyOrg_USER_SSO					

Certificate export

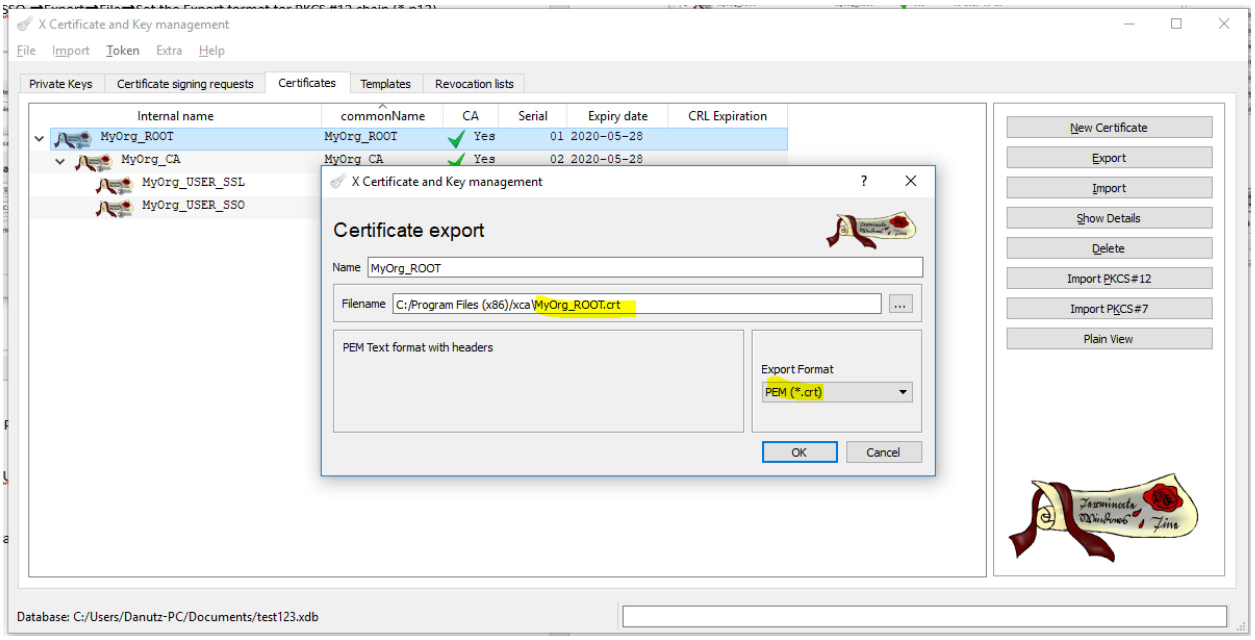
Name: MyOrg_CA

Filename: C:/Program Files (x86)/xca\MyOrg_CA.crt

PEM Text format with headers

Export Format: PEM (*.crt)

Database: C:/Users/Danutz-PC/Documents/test123.xdb



Internal name	commonName	CA	Serial	Expiry date	CRL Expiration
MyOrg_ROOT	MyOrg_ROOT	✓ Yes	01	2020-05-28	
MyOrg_CA	MyOrg_CA	✓ Yes	02	2020-05-28	
MyOrg_USER_SSL					
MyOrg_USER_SSO					

Certificate export

Name: MyOrg_ROOT

Filename: C:/Program Files (x86)/xca\MyOrg_ROOT.crt

PEM Text format with headers

Export Format: PEM (*.crt)

Database: C:/Users/Danutz-PC/Documents/test123.xdb

At the end of the exports we should have:

MyOrg_ROOT.crt (public format), correspondent to the ROOT certificate
    MyOrg_CA.crt (public format), correspondent to the INTERMEDIATE certificate
        MyOrg_USER_SSL.p12 (private format), correspondent to the SSL USER certificate
        MyOrg_USER_SSO.p12 (private format), correspondent to the SSO USER certificate