

Generate Request and Certificate PKI Application

User Guide

Table of Contents

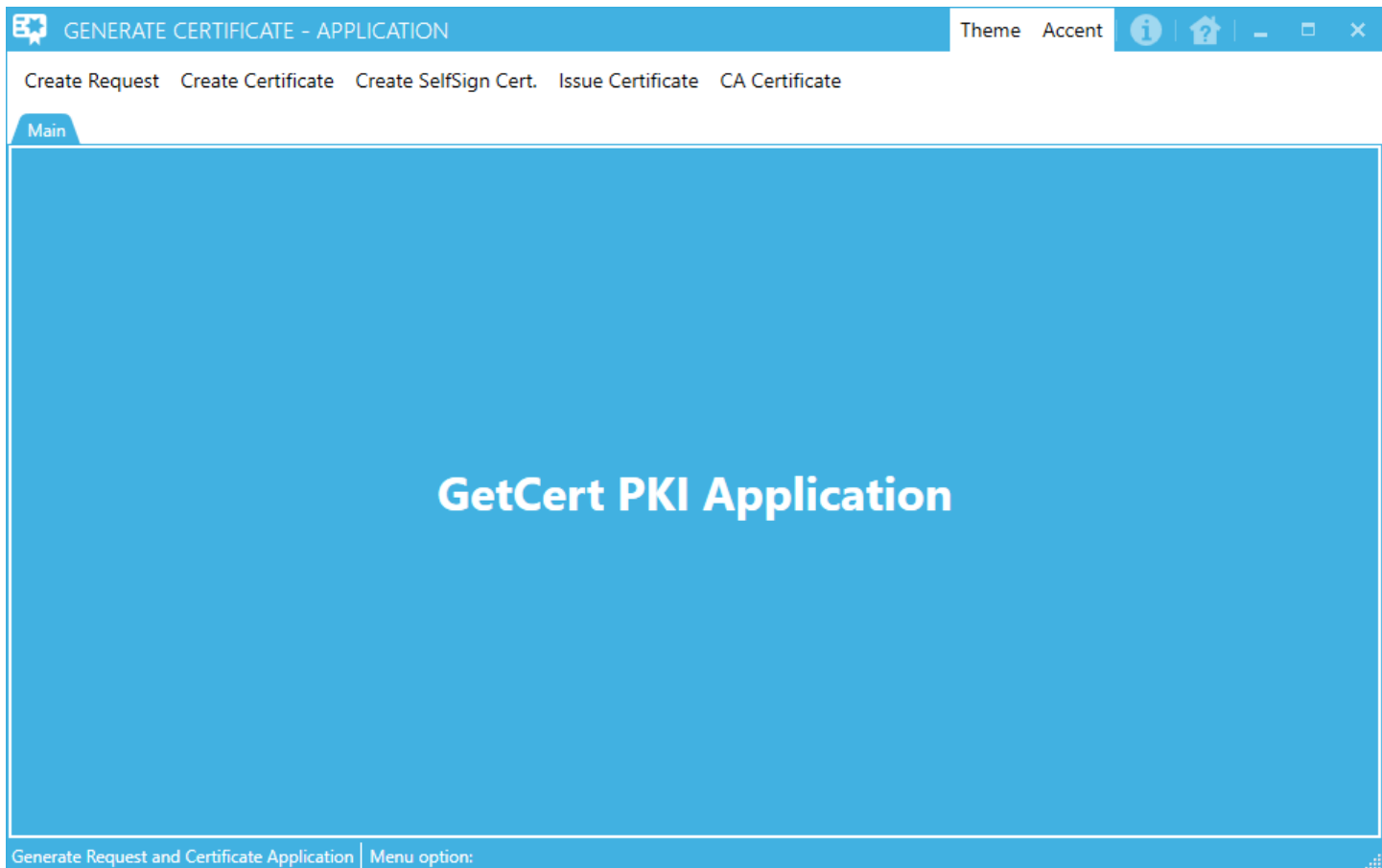
1. INTRODUCTION	1
1.1. User Interface.....	2
2. CREATE REQUEST	11
3. CREATE CERTIFICATE	17
4. CREATE SELFSIGN CERT	21
5. ISSUE CERTIFICATE	24
6. CA CERTIFICATE	27
7. IMPORT DATA FROM GENERATED SIGNED CERTIFICATE FILE TO SERVER CERTIFICATE STORE	34
8. IMPORT DATA FROM GENERATED CA CERTIFICATE FILE(S) TO CLIENT CERTIFICATE STORE	41
8.1. Import three level CA certificate (issuer CA)	43
8.2. Import two level CA certificate (intermediate CA)	46
8.3. Import one level CA certificate (master CA).....	49
9. APPLICATION REQUIREMENTS.....	53

1. INTRODUCTION

This document describes how to use Generate Request and Certificate PKI Application (GenCert) application.

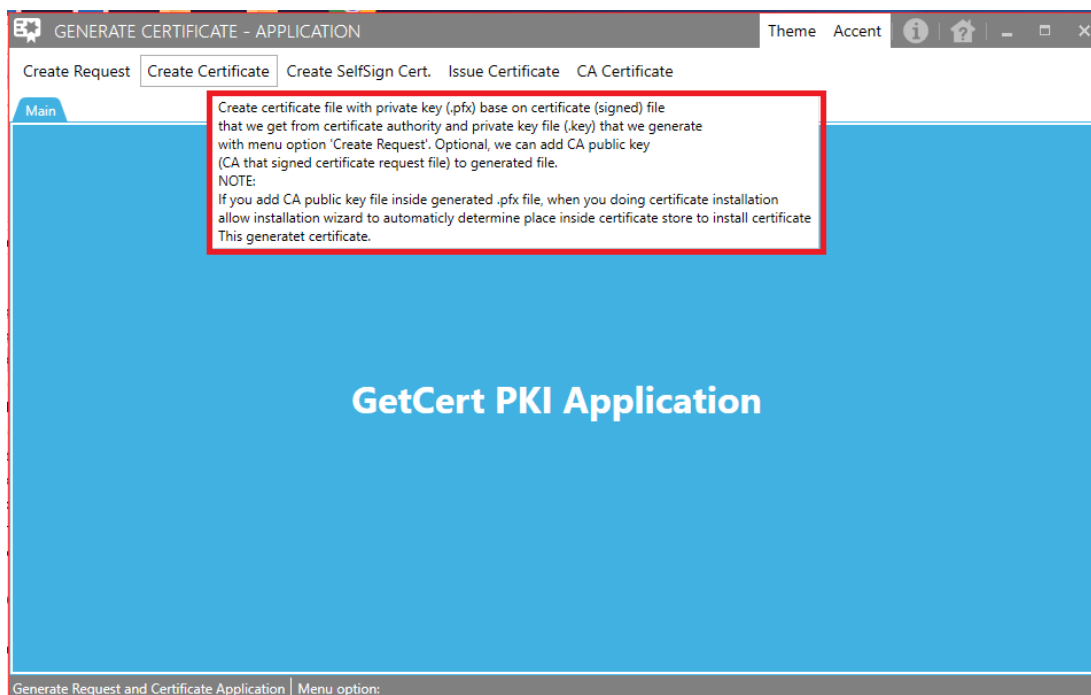
The application consists of 5 menu options:

- Create Request
- Create Certificate
- Create SelfSign Cert.
- Issue Certificate
- CA Certificate



1.1. User Interface

If you position mouse pointer over each menu option, you can get ToolTip help, what does that menu option do.



When you click on any menu option at the top of the window inside, appropriate form will be open inside new tab.

The screenshot shows the 'GENERATE CERTIFICATE - APPLICATION' window with the 'Create Request' form open. The form includes the following fields and options:

- Common Name:** For example server1.webserver.local
- Subject Alternative Names:** (Empty text area)
- Is this CA certificate:** No
- Key Length:** 1024
- Signature Algorithm:** SHA1WITHRSA
- Country Code:** For example "RS"
- State or Province Name:** For example "Serbia"
- Locality Name:** For example "Novi Sad"
- Organization:** For example "Company123"
- Choose Key Usage:** (Dropdown menu)
- Choose Extended Key Usage:** ServerAuthetification
- Path to store generate files:** G:\PKI\GenCert\out
- Private Key File Name:** Name for private key file without extension (.key)
- Request Key File Name:** Name for cert. request key file without extension (.csr)

Buttons: Generate, Continue, Gen.Alternative Names, Browse.

To close new opened tab, you can use 'x' circle inside each opened tab

GENERATE CERTIFICATE - APPLICATION

Theme Accent

Create Request Create Certificate Create SelfSign Cert. Issue Certificate CA Certificate

Main Create Request

Common Name: For example server1.webserver.local

Subject Alternative Names:

Is this CA certificate: No

Key Length: 1024

Signature Algorithm: SHA1WITHRSA

Country Code: For example "RS"

State or Province Name: For example "Serbia"

Locality Name: For example "Novi Sad"

Organization: For example "Company123"

Choose Key Usage:

Choose Extended Key Usage: ServerAuthentication

Path to store generate files: G:_PKI\GenCert\out

Private Key File Name: Name for private key file without extension

Request Key File Name: Name for cert. request key file without extension

Generate

Continue

Generate Request and Certificate Application | Menu option: Create Request

If you wish to reorder opened tab you can drag, move and drop each opened tab

GENERATE CERTIFICATE - APPLICATION

Theme Accent

Create Request Create Certificate Create SelfSign Cert. Issue Certificate CA Certificate

Main Create Request Create Certificate

Certificate Friendly name: Certificate Friendly name

Path for signed request file (.cer): Select .cer file path

Path for private key file (.key): Select .key file path

Path for generate certificate file (.pfx): Select folder to store generate cert.file .pfx

Certificate File Name: Name for certificate file without extension

Password for export private key: Password

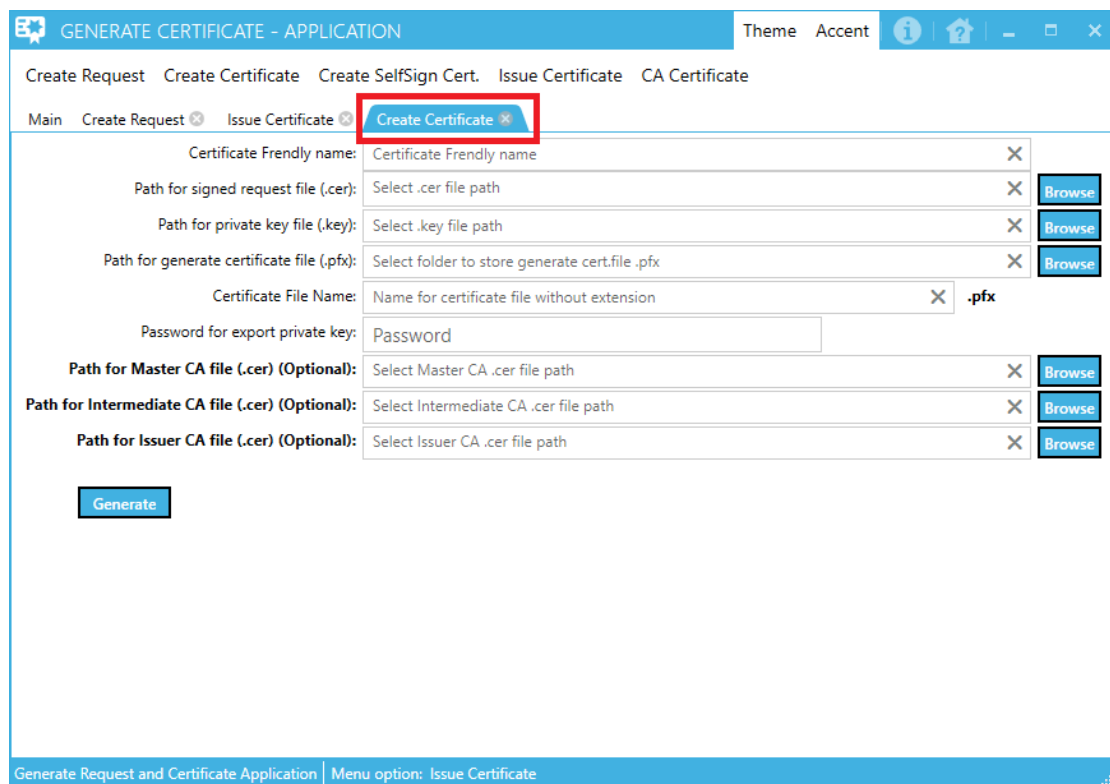
Path for Master CA file (.cer) (Optional): Select Master CA .cer file path

Path for Intermediate CA file (.cer) (Optional): Select Intermediate CA .cer file path

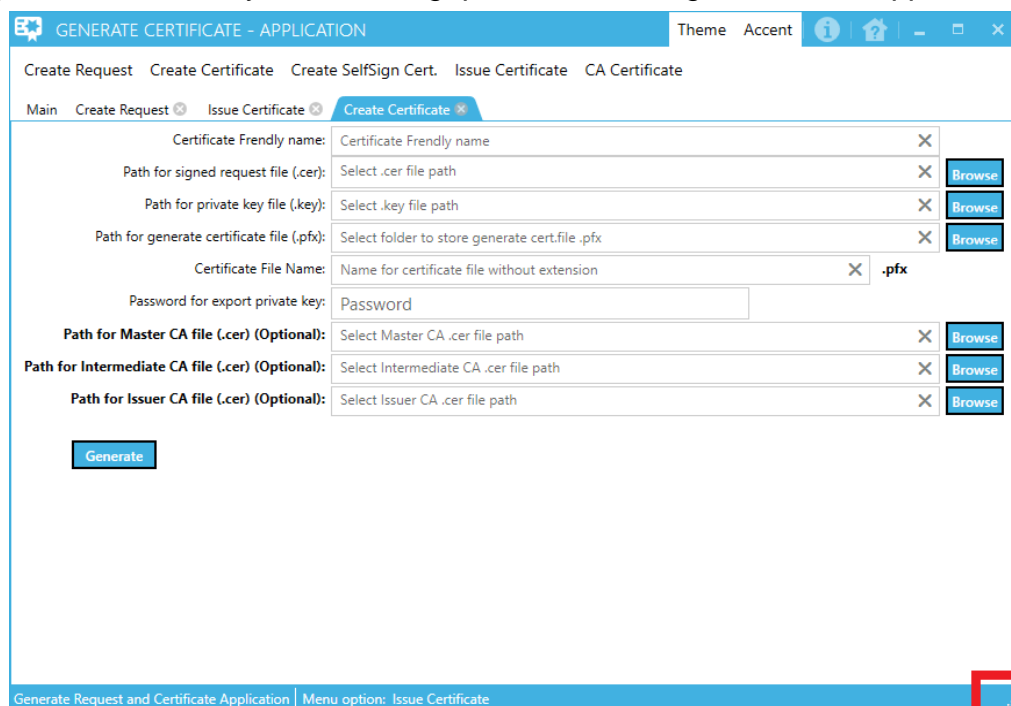
Path for Issuer CA file (.cer) (Optional): Select Issuer CA .cer file path

Generate

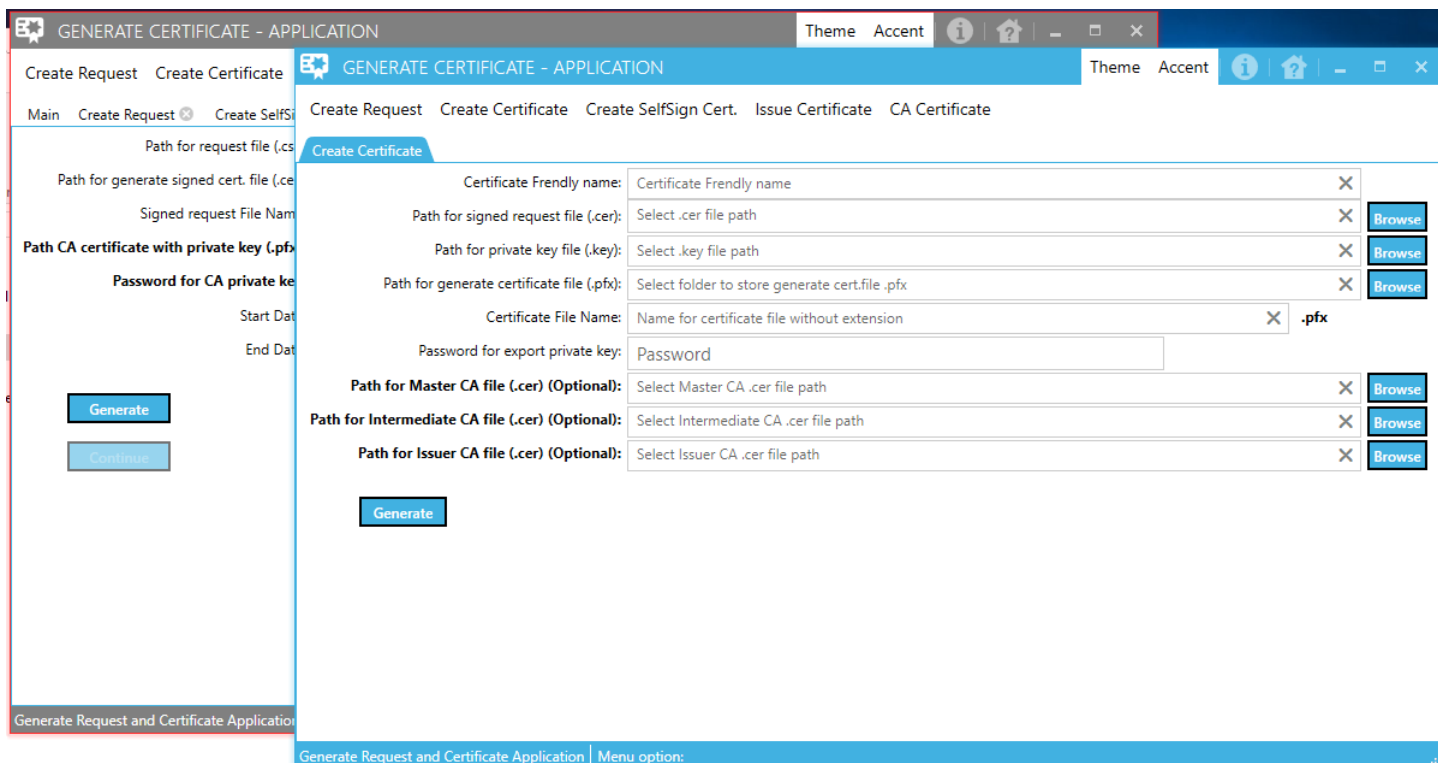
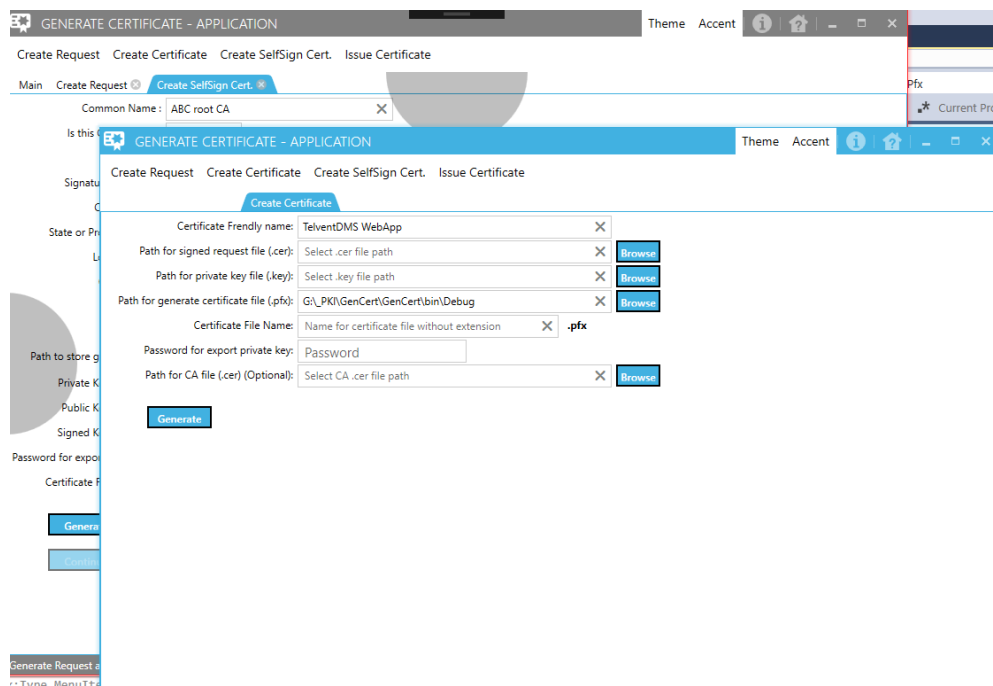
Generate Request and Certificate Application | Menu option: Issue Certificate



To resize application window, you can use grip inside down right corner of application window



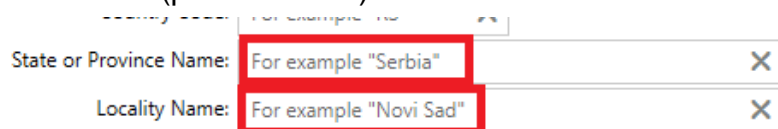
You can drag and drop each opened tab inside main application window to totally new window by clicking on any tab, press left mouse button, hold and move tab outside the main application window.



But, be careful, if you close any application window, all created application windows also will be closed.

On the open form for any menu option you will found different UI elements (textboxes, comboboxes, checkcomboboxes, passwords, ... etc.).

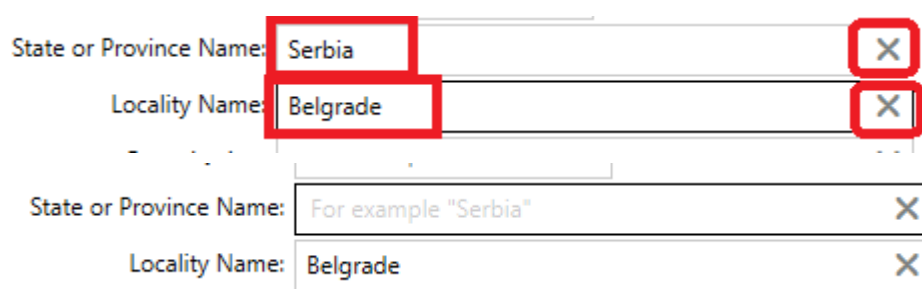
When textbox is empty, you will see watermark message as help, what you need to enter inside that textbox (picture below).



State or Province Name: For example "Serbia" X

Locality Name: For example "Novi Sad" X

At the end of each textbox you will find "X" mark. If you enter some value inside textbox and after that you click on "X" mark at the end of that textbox, entered value inside textbox will be deleted (picture below).



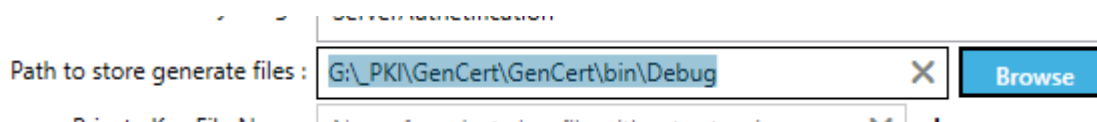
State or Province Name: Serbia X

Locality Name: Belgrade X

State or Province Name: For example "Serbia" X

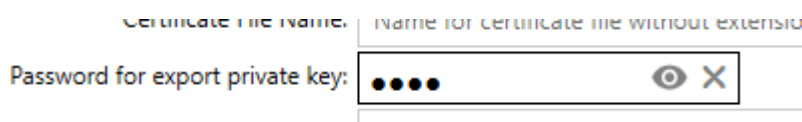
Locality Name: Belgrade X

When you click on textboxes that use to store value from Browse button, all content of that textboxes will be automatically selected (picture below)



Path to store generate files : G:_PKI\GenCert\GenCert\bin\Debug X Browse


When you are entering value inside password box, you will see dot character instead of character you entered (picture below)




Certificate file name: Name for certificate file without extension

Password for export private key: X

When password box has some value, at the end of password box you will see new mark "eye"

 . When you click on "eye" you will see real content inside password box (picture below)



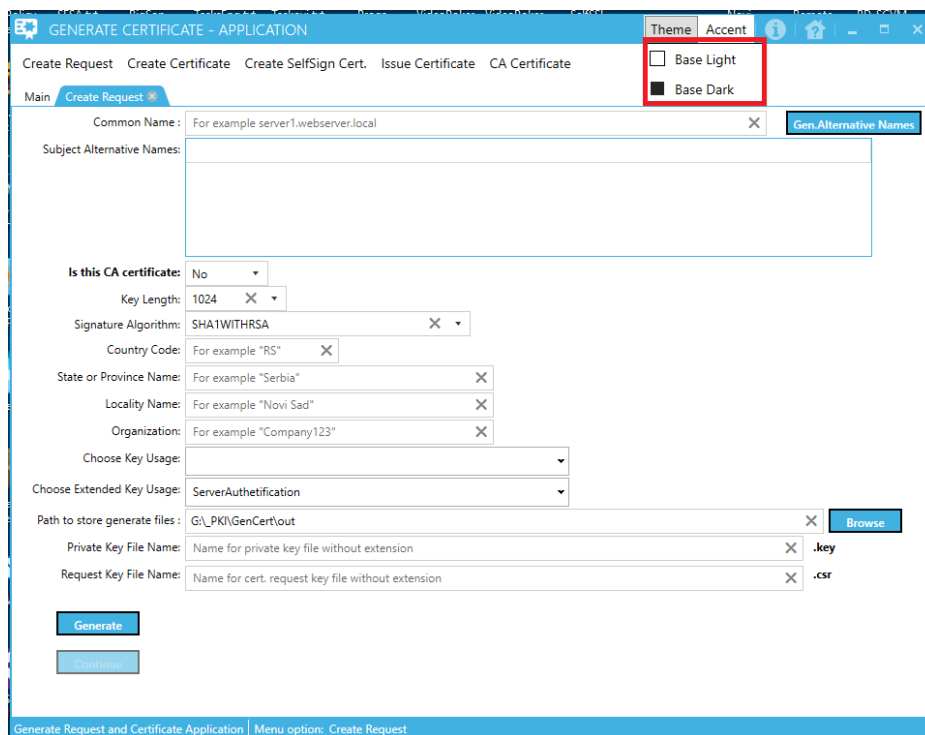
Certificate file name: Name for certificate file without extension

Password for export private key: 1234 X

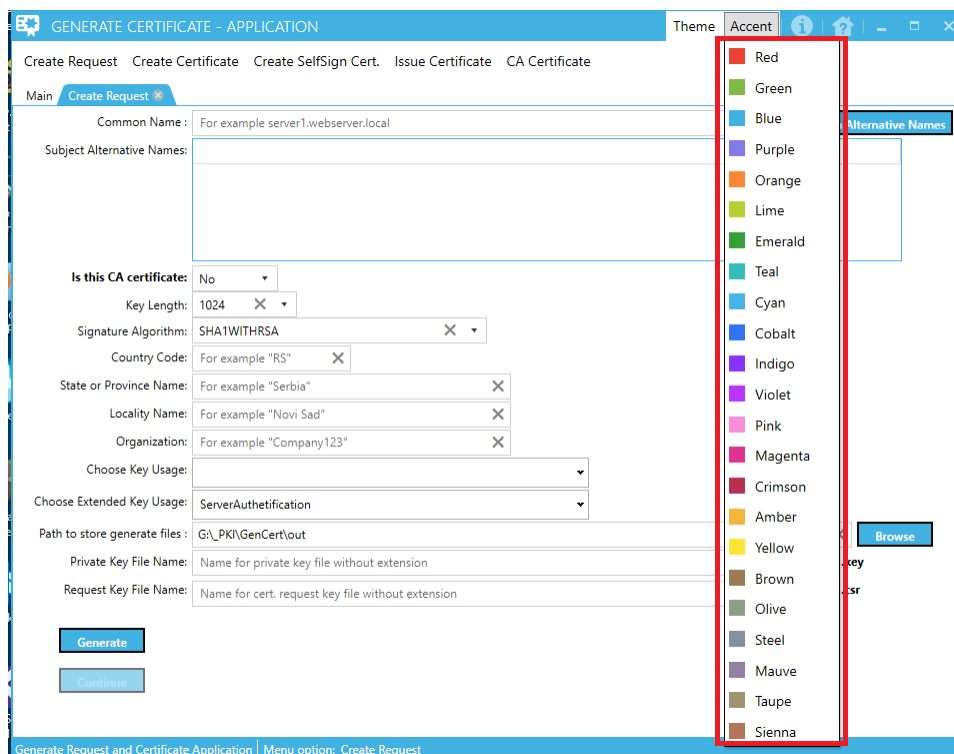
When you need to enter value for date field on the form, you can enter that value manually or click on calendar symbol at the end of date field and choose value (picture below)

If you position mouse pointer over each button inside opened form, you will get help for current button under the mouse pointer. For example, mouse pointer located over button “Generate”, picture below:

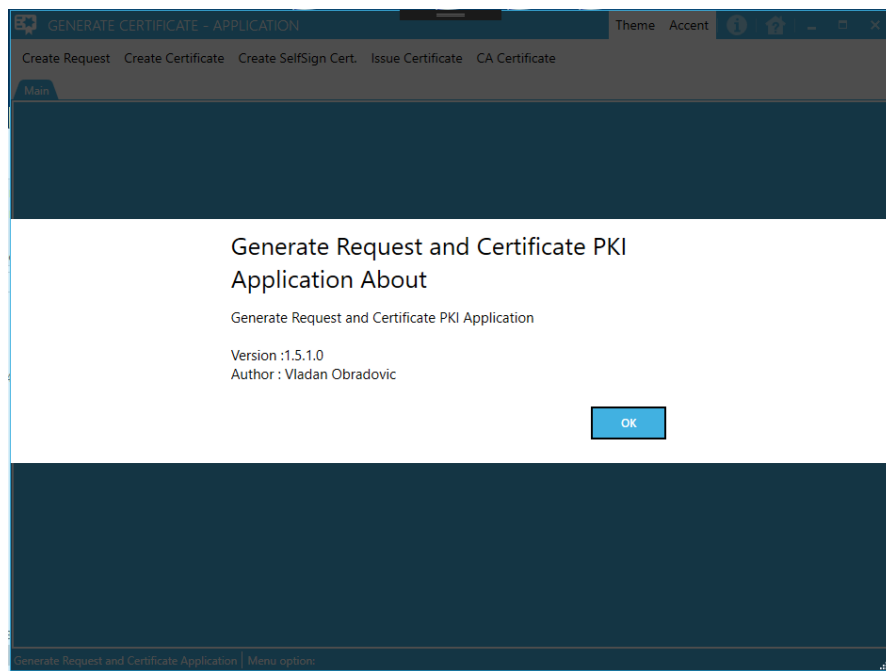
When you click on the Theme button on the application title bar, menu options for application themes will be open. You can choose between “Base Light” (see picture under) and “Base Dark” theme.



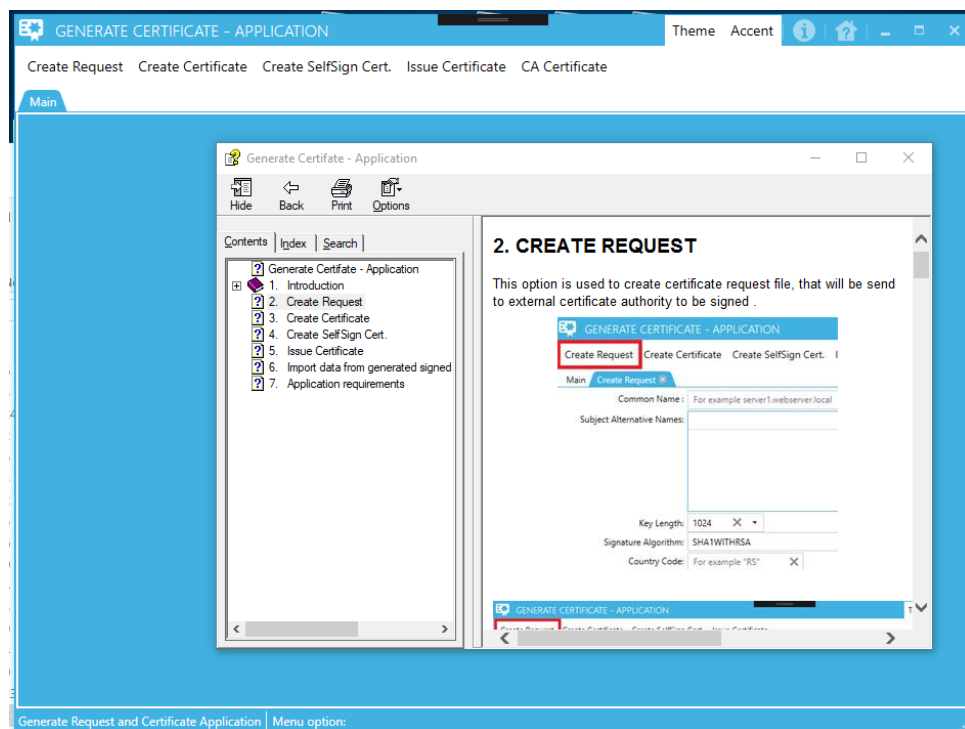
When you click on the Accent button on the application title bar, menu options for application color theme will be open. You can choose between 23 color themes.



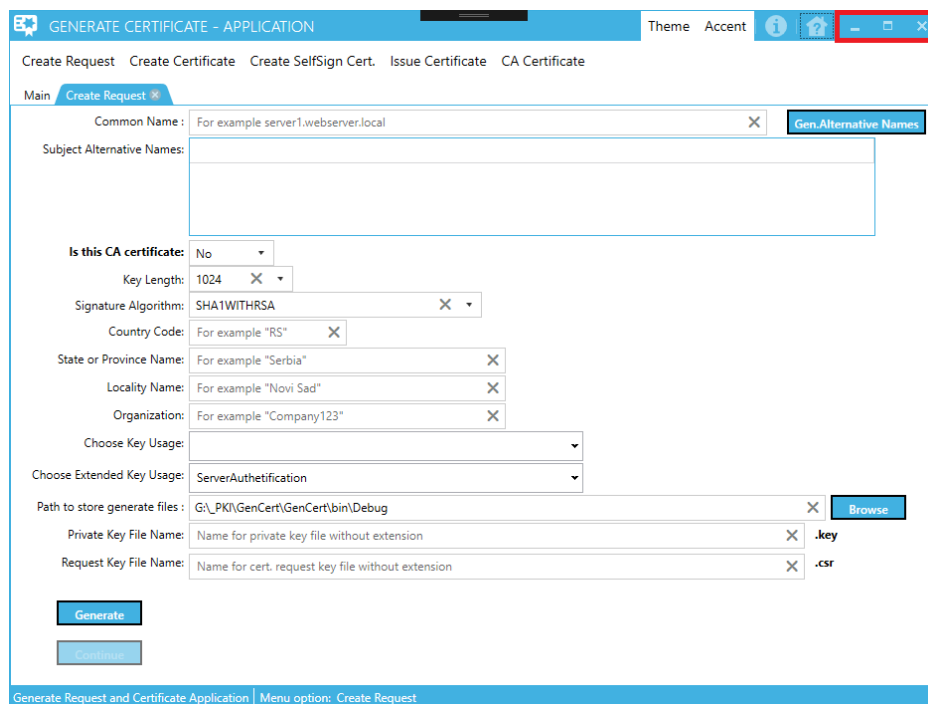
When you click on the Application Info button on the application title bar, application info will be open.



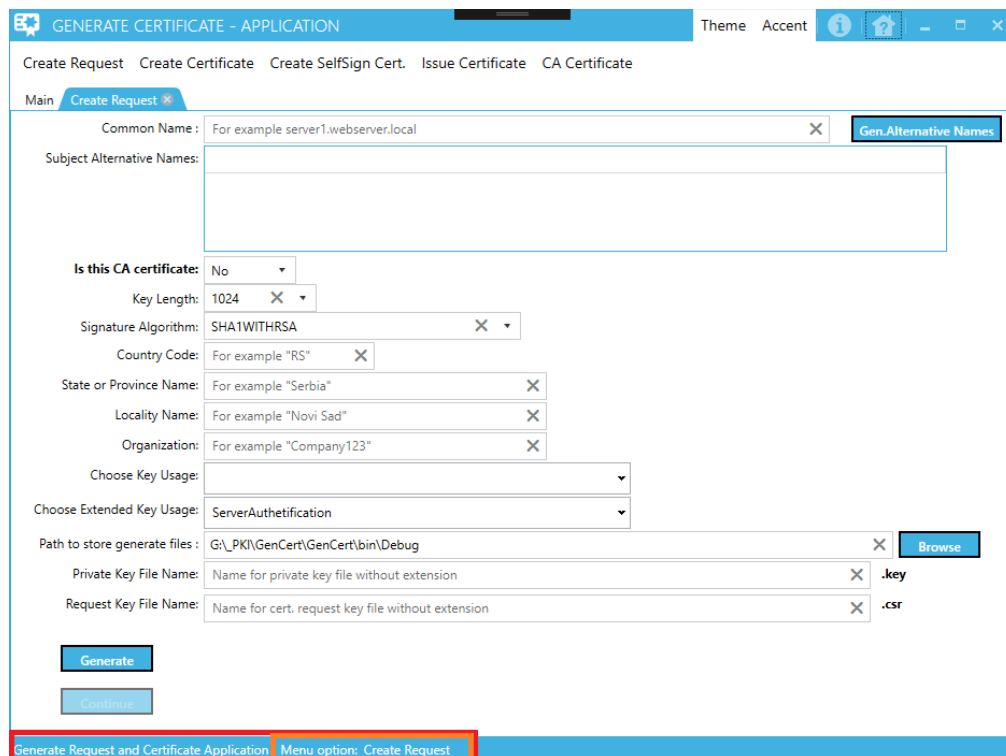
When you click on the Get Application Help button on the application title bar, application help in .chm format will be open.



When click on the main window command buttons (upper red rectangle in the picture below), you can minimize, maximize and close (exit) application.



Inside application status bar, you can see name of currently activated menu option (see picture below)



Inside windows task bar you can see application icon (see picture below)



2. CREATE REQUEST

This option is used to create certificate request file, that will be send to external certificate authority to be signed .

The screenshot shows the 'GENERATE CERTIFICATE - APPLICATION' window with the 'Create Request' tab selected. The form contains the following fields and options:

- Common Name:** For example server1.webserver.local
- Subject Alternative Names:** (Empty table)
- Key Length:** 1024
- Signature Algorithm:** SHA1WITHRSA
- Country Code:** For example "RS"
- Is this CA certificate:** No
- State or Province Name:** For example "Serbia"
- Locality Name:** For example "Novi Sad"
- Organization:** For example "Company123"
- Choose Key Usage:** (Dropdown menu)
- Choose Extended Key Usage:** ServerAuthentification
- Path to store generate files:** G:_PKI\GenCert\GenCert\bin\Debug
- Request Key File Name:** Name for cert. request key file without extension
- Private Key File Name:** Name for private key file without extension
- Password for private key file:** Password

Buttons: 'Generate', 'Continue', 'Gen.Alternative Names', 'Browse'.

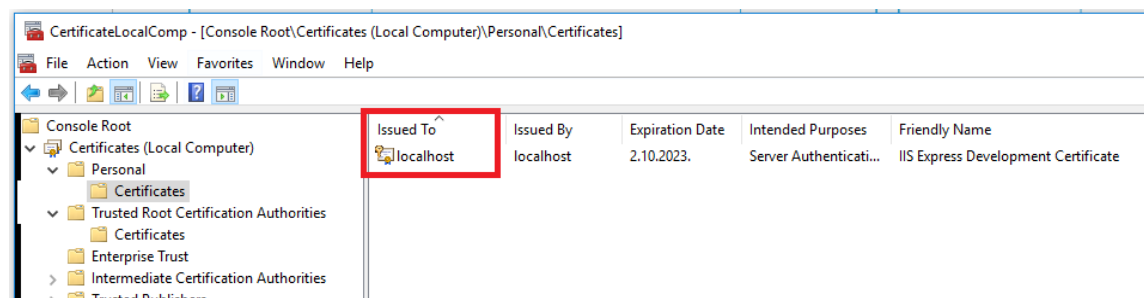
You need to fill all displayed fields on the form under before generate certificate request file. Some fields offer default value that can be changed. Another fields need to be fill with appropriate value(s).

Watermark inside fields that need to be fill with value(s) show example value for each field.

When enter value inside field "Common Name:" you can click on the button "Gen.Alternative Names" to generate alternative web server names to fill "Subject Alternative Names:" table or you can fill this table with alternative names manually. Or you can generate and then change names of generated alternative names.

Tip:

In field “Common Name:” you can enter value for field “Issued To”. This is what you can see when you open Certificate mmc console and look for certificate



When you fill all fields inside form click on the button “Generate” to create following files:

1. File with certificate private key (.key)
2. File with certificate request (.csr)

If everything is OK, form will be generated 2 files. Inside directory path entered inside field "Path to store generate files:", you will find two files. First file with .key extension (certificate private key file) and the second file with .csr extension (certificate request file).

Tip:

If you wish to add additional security to generated private key file, you can enter password inside field “Password for private key file” and that password will be used to encrypt data inside generated private key file. See the picture below (orange rectangle). This field is optional.

GENERATE CERTIFICATE - APPLICATION

Theme Accent

Create Request Create Certificate Create SelfSign Cert. Issue Certificate CA Certificate

Main Create Request *

Common Name: webserver.local Gen Alternative Names

Subject Alternative Names: webserver.local
s1.webserver.local
s2.webserver.local
s3.webserver.local

Is this CA certificate: No

Key Length: 1024

Signature Algorithm: SHA512WITHRSA

Country Code: RS

State or Province Name: Serbia

Locality Name: Novi Sad

Organization: Company 123

Choose Key Usage:
Choose Extended Key Usage: ServerAuthentication

Path to store generate files: G:\PKI\GenCert\GenCert\bin\Debug\webserver Browse

Request Key File Name: webserver_request .csr

Private Key File Name: webserver_private .key

Password for private key file: [masked] toggle icon

Generate Continue

Generate Request and Certificate Application | Menu option: Create Request

File with .csr extension need to be send to external certificate authority for signing and generate file with .cer extension. This file will be use lately to generate signed certificate file with private key (file with .pfx extension). Also, you can generate own CA root authority certificate and sign certificate request file with generated CA root certificate (menu option "Create SelfSign Cert."), or if you already have generated CA root certificate file you immediately can proceed with sign certificate file with that root CA certificate (menu option "Issue Certificate").

NOTE:

If data inside form not filled as need, you will get error screen, for example:

GENERATE CERTIFICATE - APPLICATION

Theme Accent

Create Request Create Certificate Create SelfSign Cert. Issue Certificate CA Certificate

Main Create Request *

Common Name: webserver.local Gen Alternative Names

Subject Alternative Names: s1.webserver.local
s2.webserver.local
s3.webserver.local

Error

You MUST enter Country Code.
You MUST enter name for State or Province.
You MUST enter Location name.
You MUST enter Organization name.
You MUST enter file name to save certificate private key (.key extension).
You MUST enter file name to save certificate request (.csr extension).

OK

Path to store generate files: G:\PKI\GenCert\GenCert\bin\Debug Browse

Private Key File Name: Name for private key file without extension .key

Request Key File Name: Name for cert. request key file without extension .csr

Generate Continue

Generate Request and Certificate Application | Menu option: Create Request

You need to fix all error messages and then try to generate certificate request file again. If everything fills correctly on the form, when you press Generate button, two files (.key and .csr) will be generate and button "Continue" on the form will be enabled.

GENERATE CERTIFICATE - APPLICATION Theme Accent

Create Request Create Certificate Create SelfSign Cert. Issue Certificate CA Certificate

Main Create Request

Common Name: webserver.local Gen Alternative Names

Subject Alternative Names: webserver.local, s1.webserver.local, s2.webserver.local, s3.webserver.local

Is this CA certificate: No

Key Length: 1024

Signature Algorithm: SHA512WITHRSA

Country Code: RS

State or Province Name: Serbia

Locality Name: Novi Sad

Organization: Company 123

Choose Key Usage:

Choose Extended Key Usage: ServerAuthentification

Path to store generate files: G:\PKI\GenCert\GenCert\bin\Debug\webserver Browse

Request Key File Name: webserver_request .csr

Private Key File Name: webserver_private .key

Password for private key file: Generate Continue

Generate Request and Certificate Application | Menu option: Create Request

Click Generate

GENERATE CERTIFICATE - APPLICATION Theme Accent

Create Request Create Certificate Create SelfSign Cert. Issue Certificate CA Certificate

Main Create Request

Common Name: webserver.local Gen Alternative Names

Subject Alternative Names: webserver.local, s1.webserver.local, s2.webserver.local, s3.webserver.local

Is this CA certificate: No

Key Length: 1024

Signature Algorithm: SHA512WITHRSA

Country Code: RS

State or Province Name: Serbia

Locality Name: Novi Sad

Organization: Company 123

Choose Key Usage:

Choose Extended Key Usage: ServerAuthentification

Path to store generate files: G:\PKI\GenCert\GenCert\bin\Debug\webserver Browse

Request Key File Name: webserver_request .csr

Private Key File Name: webserver_private .key

Password for private key file: Generate Continue

File with private key: G:\PKI\GenCert\GenCert\bin\Debug\webserver\webserver_private.key successfully generated and saved.

Generate Request and Certificate Application | Menu option: Create Request

GENERATE CERTIFICATE - APPLICATION

Create Request Create Certificate Create SelfSign Cert. Issue Certificate CA Certificate

Main Create Request

Common Name:

Subject Alternative Names:

Is this CA certificate:

Key Length:

Signature Algorithm:

Country Code:

State or Province Name:

Locality Name:

Organization:

Choose Key Usage:

Choose Extended Key Usage:

Path to store generate files:

Request Key File Name:

Private Key File Name:

Password for private key file:

File with private key: G:\PKI\GenCert\GenCert\bin\Debug\webserver\webserver_private.key successfully generated and saved.
 File with certificate request: G:\PKI\GenCert\GenCert\bin\Debug\webserver\webserver_request.csr successfully generated.
 File with certificate request: G:\PKI\GenCert\GenCert\bin\Debug\webserver\webserver_request.csr successfully saved.

Generate Request and Certificate Application | Menu option: Create Request

Messages shown upper, inside green rectangle, informing you that .key and .csr files has been generated.

Tip:

You can use Browse button to locate folder where generated files will be stored or you can enter path manually and if path not exist, that folders path will be created.

If you click on button Continue, new wizard dialog will be open.

GENERATE CERTIFICATE - APPLICATION

Create Request Create Certificate Create SelfSign Cert. Issue Certificate CA Certificate

Main Create Request

Common Name:

INFO

You need to send generated .csr file to internal or external certificate authority to generate certificate based on data inside certificate request file (.csr) => 'Send to CA authority'

To test generated .csr file you can generate self sign certificate file that can act as CA root and use it to sign certificate request and generate new certificate based on data inside certificate request (.csr) file => 'Sign locally - Don't have CA cert'

If you wish to use generate certificate file inside internal network, technically speaking you can use self-sign CA generate certificate file for that purpose. You need to import generate CA root certificate file (inside menu option 'Create SelfSign Cert.') to Trusted Root Certification Authorities inside certification store and everything will be work fine for certificate generated base on certificate request and signed with generated CA root certificate.

If you already have generated certificate for CA root authority you can sign request with that certificate => 'Sign locally - Have CA cert'

Generate Request and Certificate Application | Menu option: Create Request

1. Option "Send to CA authority" will be use if you wish to send generated .csr file to internal or external CA authority for issuing
2. Option "Cancel" will close current dialog and return you to previous screen
3. Option "Sign locally-Don't have CA cert" will activate menu option "Create SelfSign Cert." and set value inside opened form "Is this CA certificate: "=Yes. We can use this option if we wish to generate certificate for CA root authority that we will use to sign certificate request (.csr) file
4. Option "Sign locally-Have CA cert" will activate menu option "Issue Certificate" to sign certificate request file (.csr) with previously generated CA root authority certificate (option "Create SelfSign Cert.").

3. CREATE CERTIFICATE

This option is used to create certificate request file, that will be send to external certificate authority to be signed.

The screenshot shows the 'GENERATE CERTIFICATE - APPLICATION' window with the 'Create Certificate' tab selected. The 'Certificate Friendly name' is 'Web Server'. The 'Path for signed request file (.cer)' is 'G:_PKI\GenCert\Ger'. The 'Path for private key file (.key)' is 'C:\GenCert\GenCert\'. The 'Generate' button is visible at the bottom.

You need to fill all displayed fields on the form under before generate file signed certificate file with private key.

You can enter data manually or click on the “Browse” buttons to select appropriate files and output directory where generate file (.pfx extension) will be generate.

Inside field "Password for export private key:" you need to enter password that will be use when import data for generated certificate file to computer store. When enter data inside this field, you will see black dots. When click on the right icon on right inside this field, you will see what you typed.

Watermark inside fields show help message what kind of data need to be enter to each field.

Tip:

You can use Browse button to locate folder where generated files will be stored or you can enter path manually and if path not exist, that folders path will be created.

GENERATE CERTIFICATE - APPLICATION

Theme Accent

Create Request Create Certificate Create SelfSign Cert. Issue Certificate CA Certificate

Main Create Request CA Certificate Issue Certificate **Create Certificate**

Certificate Friendly name: Web Server

Path for signed request file (.cer): G:_PKI\GenCert\GenCert\bin\Debug\webserver\webserver_signed2.cer

Path for private key file (.key): G:_PKI\GenCert\GenCert\bin\Debug\webserver\webserver_private2.key

Password for read data from private key file:

Path for generate certificate file (.pfx): G:_PKI\GenCert\GenCert\bin\Debug\webserver

Certificate File Name: webserver .pfx

Password for export private key:

Path for Master CA file (.cer) (Optional): G:_PKI\GenCert\GenCert\bin\Debug\masterCA\masterCA_public.cer

Path for Intermediate CA file (.cer) (Optional): G:_PKI\GenCert\GenCert\bin\Debug\intermediateCA\intermediateCA_public.cer

Path for Issuer CA file (.cer) (Optional): G:_PKI\GenCert\GenCert\bin\Debug\issuerCA\issuerCA_public.cer

Generate

Generate Request and Certificate Application | Menu option: Create Certificate

When you fill all fields inside form click on the button “Generate” to create files.

If everything is OK, form will generate certificate file with private key inside directory path entered inside field "Path for generate certificate file (.pfx):".

This file and external CA public key file need to be installed inside computer store on each computer that will access web server.

NOTE:

If data inside form not filled as need, you will get error screen, for example:

GENERATE CERTIFICATE - APPLICATION

Theme Accent

Create Request Create Certificate Create SelfSign Cert. Issue Certificate CA Certificate

Main **Create Certificate** CA Certificate

Certificate Friendly name: Web Server

Path for signed request file (.cer): G:_PKI\GenCert\out\abcserver01\signed.cer

Path for private key file (.key): G:_PKI\GenCert\out\abcserver01\private.key

Path for generate certificate file (.pfx): G:_PKI\GenCert\out\abcserver01

Error

You MUST enter name for generate certificate file (.pfx).
You MUST enter password for export private key from certificate file.

OK

Generate Request and Certificate Application | Menu option: CA Certificate

You need to fix all error messages and then try to generate certificate request file again.

Tip:

If you previously generated private key file and protected it with password, you must enter that password inside field "Password for read data from private key file". See picture below (orange rectangle).

The screenshot shows the 'GENERATE CERTIFICATE - APPLICATION' window with the 'Create Certificate' tab selected. The 'Password for read data from private key file' field is highlighted with an orange rectangle. The window contains the following fields and options:

- Certificate Friendly name: Web Server
- Path for signed request file (.cer): G:_PKI\GenCert\GenCert\bin\Debug\webserver\webserver_signed2.cer
- Path for private key file (.key): G:_PKI\GenCert\GenCert\bin\Debug\webserver\webserver_private2.key
- Password for read data from private key file: (highlighted with orange rectangle)
- Path for generate certificate file (.pfx): G:_PKI\GenCert\GenCert\bin\Debug\webserver
- Certificate File Name: webserver
- Password for export private key: (masked with dots)
- Path for Master CA file (.cer) (Optional): G:_PKI\GenCert\GenCert\bin\Debug\masterCA\masterCA_public.cer
- Path for Intermediate CA file (.cer) (Optional): G:_PKI\GenCert\GenCert\bin\Debug\intermediateCA\intermediateCA_public.cer
- Path for Issuer CA file (.cer) (Optional): G:_PKI\GenCert\GenCert\bin\Debug\issuerCA\issuerCA_public.cer

A 'Generate' button is located at the bottom left of the form area.

You can use Browse button to locate folder where generated files will be stored or you can enter path manually and if path not exist, that folders path will be created.

This screenshot shows the same 'GENERATE CERTIFICATE - APPLICATION' window, but with blue dots placed on the 'Browse' buttons for the following fields:

- Path for generate certificate file (.pfx)
- Path for Master CA file (.cer) (Optional)
- Path for Intermediate CA file (.cer) (Optional)
- Path for Issuer CA file (.cer) (Optional)

The 'Generate' button is also visible at the bottom left.

GENERATE CERTIFICATE - APPLICATION

Theme Accent

Create Request Create Certificate Create SelfSign Cert. Issue Certificate CA Certificate

Main Create Request CA Certificate Issue Certificate **Create Certificate**

Certificate Friendly name: Web Server X

Path for signed request file (.cer): G:_PKI\GenCert\GenCert\bin\Debug\webserver\webserver_signed2.cer X Browse

Path for private key file (.key): G:_PKI\GenCert\GenCert\bin\Debug\webserver\webserver_private2.key X Browse

Password for read data from private key file: X

Path for generate certificate file (.pfx): G:_PKI\GenCert\GenCert\bin\Debug\webserver X Browse

Certificate File Name: webserver X .pfx

Password for export private key: X

Path for Master CA file (.cer) (Optional): G:_PKI\GenCert\GenCert\bin\Debug\masterCA\masterCA_public.cer X Browse

Path for Intermediate CA file (.cer) (Optional): G:_PKI\GenCert\GenCert\bin\Debug\intermediateCA\intermediateCA_public.cer X Browse

Path for Issuer CA file (.cer) (Optional): G:_PKI\GenCert\GenCert\bin\Debug\issuerCA\issuerCA_public.cer X Browse

Generate

Certificate file with private key: G:_PKI\GenCert\GenCert\bin\Debug\webserver\webserver.pfx and Master CA file: G:_PKI\GenCert\GenCert\bin\Debug\masterCA\masterCA_public.cer

Generate Request and Certificate Application | Menu option: Create Certificate

Generated .pfx file, now can be import to appropriate computer certificate store.

See chapter 7, how to import data from generated certificate file to computer certificate store.

4. CREATE SELFSIGN CERT.

This option is used to create self-sign certificate or create self-sign certificate for CA root.

The screenshot displays the 'GENERATE CERTIFICATE - APPLICATION' window. The 'Main' tab is active, and the 'Create SelfSign Cert.' option is selected and highlighted with a red box. The form contains the following fields:

- Common Name:** ABC root CA
- Is this CA certificate:** No
- Key Length:** 1024
- Signature Algorithm:** SHA256withRSA
- Country Code:** For example "RS"
- State or Province Name:** For example "Serbia"
- Locality Name:** For example "Novi Sad"
- Organization:** For example "Company123"
- Start Date:** 12.3.2019.
- End Date:** 12.3.2019.
- Path to store generate files:** Enter path to save generate files
- Private Key File Name:** Name for cert. private key file without extension
- Public Key File Name:** Name for cert. public key file without extension
- Signed Key File Name:** Name for cert. signed key file without extension
- Password for export private key:** Password
- Certificate Friendly name:** Certificate Friendly name

Buttons at the bottom include 'Generate' and 'Continue'. The status bar at the bottom indicates 'Generate Request and Certificate Application | Menu option: Create SelfSign Cert.'

You need to fill all displayed fields on the form over before generating self-sign certificate file.

If you use option "Is this CA certificate"=Yes, generated certificate file can be use as CA root certificate.

GENERATE CERTIFICATE - APPLICATION

Create Request Create Certificate Create SelfSign Cert. Issue Certificate CA Certificate

Main Create Request Issue Certificate Create Certificate Create SelfSign Cert.

Common Name: ABC root CA

Is this CA certificate: Yes

Key Length: 1024

Signature Algorithm: SHA1WITHRSA

Country Code: RS

State or Province Name: Serbia

Locality Name: Belgrad

Organization: Cert Authority CO

Start Date: 12.3.2019.

End Date: 12.3.2029.

Path to store generate files: G:_PKI\GenCert\out\rootCA

Private Key File Name: rootCA_private

Public Key File Name: rootCA_public

Signed Key File Name: rootCA_signed

Password for export private key:

Certificate Friendly name: ABC root CA Friendly name

Generate

Continue

Generate Request and Certificate Application | Menu option: Create SelfSign Cert.

When you click on Generate, application will generate 3 files:

1. File with certificate private key (.key)
2. File with certificate public key (.cer)
3. File with private and public key (.pfx)

If all files successfully generated and you choose Yes value for field "Is this CA certificate", button Continue will be enabled.

GENERATE CERTIFICATE - APPLICATION

Create Request Create Certificate Create SelfSign Cert. Issue Certificate CA Certificate

Main Create Request Issue Certificate Create Certificate Create SelfSign Cert.

Common Name: ABC root CA

Is this CA certificate: Yes

Key Length: 1024

Signature Algorithm: SHA1WITHRSA

Country Code: RS

State or Province Name: Serbia

Locality Name: Belgrad

Organization: Cert Authority CO

Start Date: 12.3.2019.

End Date: 12.3.2029.

Path to store generate files: G:_PKI\GenCert\out\rootCA

Private Key File Name: rootCA_private

Public Key File Name: rootCA_public

Signed Key File Name: rootCA_signed

Password for export private key:

Certificate Friendly name: ABC root CA Friendly name

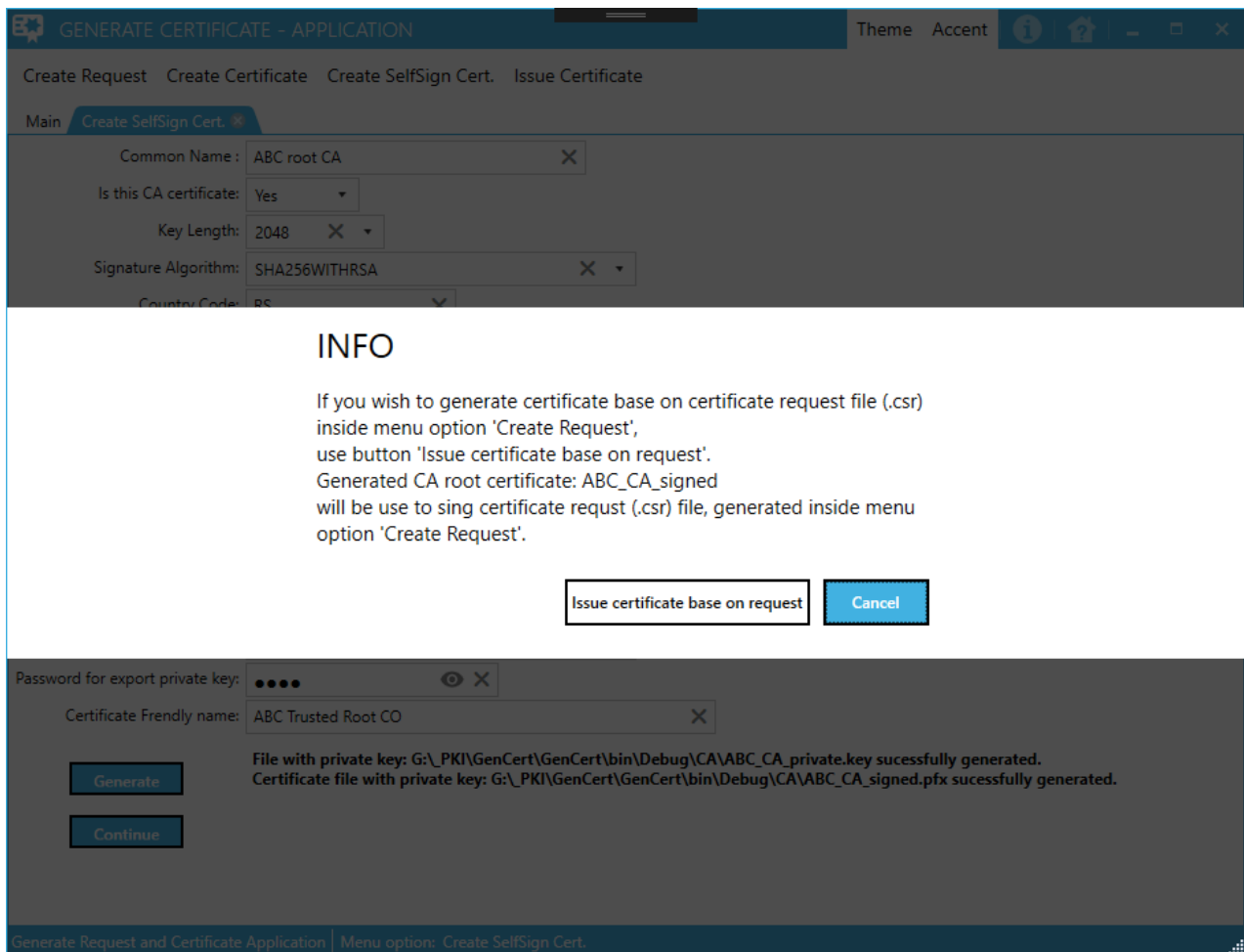
Generate

Continue

File with private key: G:_PKI\GenCert\out\rootCA\rootCA_private.key successfully generated.
Certificate file with private key: G:_PKI\GenCert\out\rootCA\rootCA_signed.pfx successfully generated.

Generate Request and Certificate Application | Menu option: Create SelfSign Cert.

If you click Continue button, you will get dialog wizard to explain what to do next.



If you use option "Create Certificate base on generated (.cer) file", this will activate menu option "Create Certificate" and automatically fill all fields with appropriate value:

"Path for signed request file (.cer):"

"Path for private key file (.key):"

"Path for generate certificate file (.pfx):"

"Path for CA file (.cer) (Optional):"

5. ISSUE CERTIFICATE

This option is used to create signed certificate file base on request certificate file which will be signed by generated CA root certificate file inside menu option "Create SelfSign Cert."

GENERATE CERTIFICATE - APPLICATION

Theme Accent

Create Request Create Certificate Create SelfSign Cert. **Issue Certificate** CA Certificate

Main Create Request > Issue Certificate >

Path for request file (.csr): G:_PKI\GenCert\GenCert\bin\Debug\webserver\webserver_request.csr

Path for generate signed cert. file (.cer): G:_PKI\GenCert\GenCert\bin\Debug\webserver

Signed request File Name: Name for signed request file without extension .cer

Path CA certificate with private key (.pfx): Select .pfx file path

Password for CA private key: Password

Start Date: 12.3.2019.

End Date: 12.3.2020.

Generate Request and Certificate Application | Menu option: Create Request

You need to fill all displayed fields on the form over before generating certificate signed file.

GENERATE CERTIFICATE - APPLICATION Theme Accent

Create Request Create Certificate Create SelfSign Cert. Issue Certificate CA Certificate

Main Create Request **Issue Certificate**

Path for request file (.csr): G:_PKI\GenCert\GenCert\bin\Debug\webserver\webserver_request.csr

Path for generate signed cert. file (.cer): G:_PKI\GenCert\GenCert\bin\Debug\webserver

Signed request File Name: webserver_signed .cer

Path CA certificate with private key (.pfx): G:_PKI\GenCert\out\issuerCA\issuerCA.pfx

Password for CA private key: ●●●●

Start Date: 12.3.2019.

End Date: 12.3.2020.

Generate Request and Certificate Application | Menu option: Create Request

When you click on Generate, application will generate file:

File with certificate public key (.cer)

GENERATE CERTIFICATE - APPLICATION Theme Accent

Create Request Create Certificate Create SelfSign Cert. Issue Certificate CA Certificate

Main Create Request **Issue Certificate**

Path for request file (.csr): G:_PKI\GenCert\GenCert\bin\Debug\webserver\webserver_request.csr

Path for generate signed cert. file (.cer): G:_PKI\GenCert\GenCert\bin\Debug\webserver

Signed request File Name: webserver_signed .cer

Path CA certificate with private key (.pfx): G:_PKI\GenCert\out\issuerCA\issuerCA.pfx

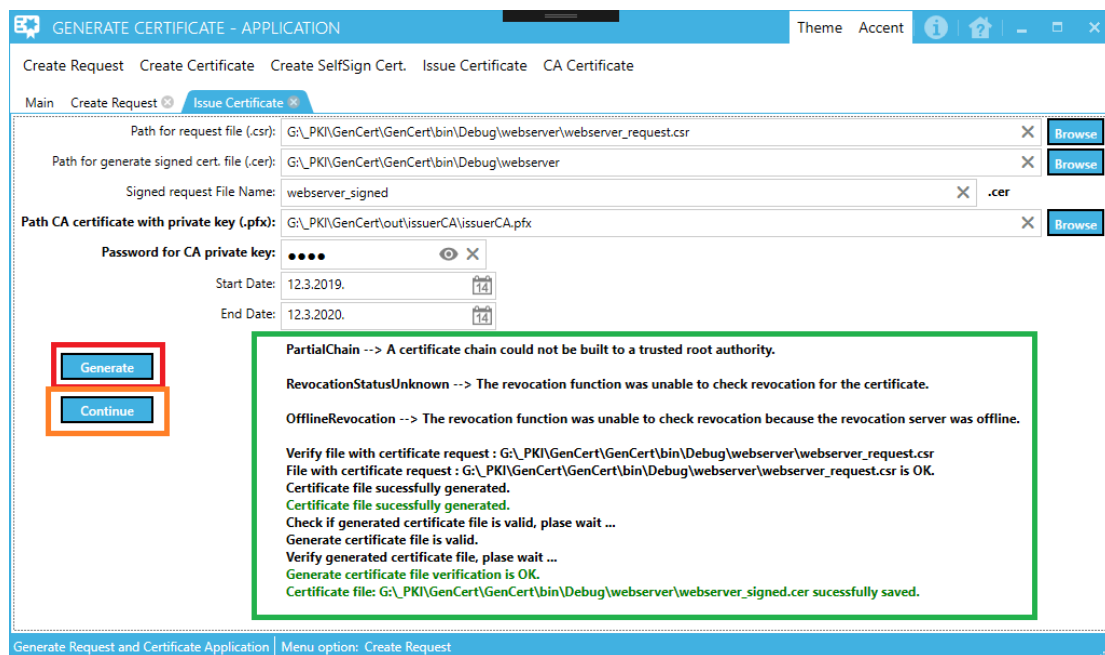
Password for CA private key: ●●●●

Start Date: 12.3.2019.

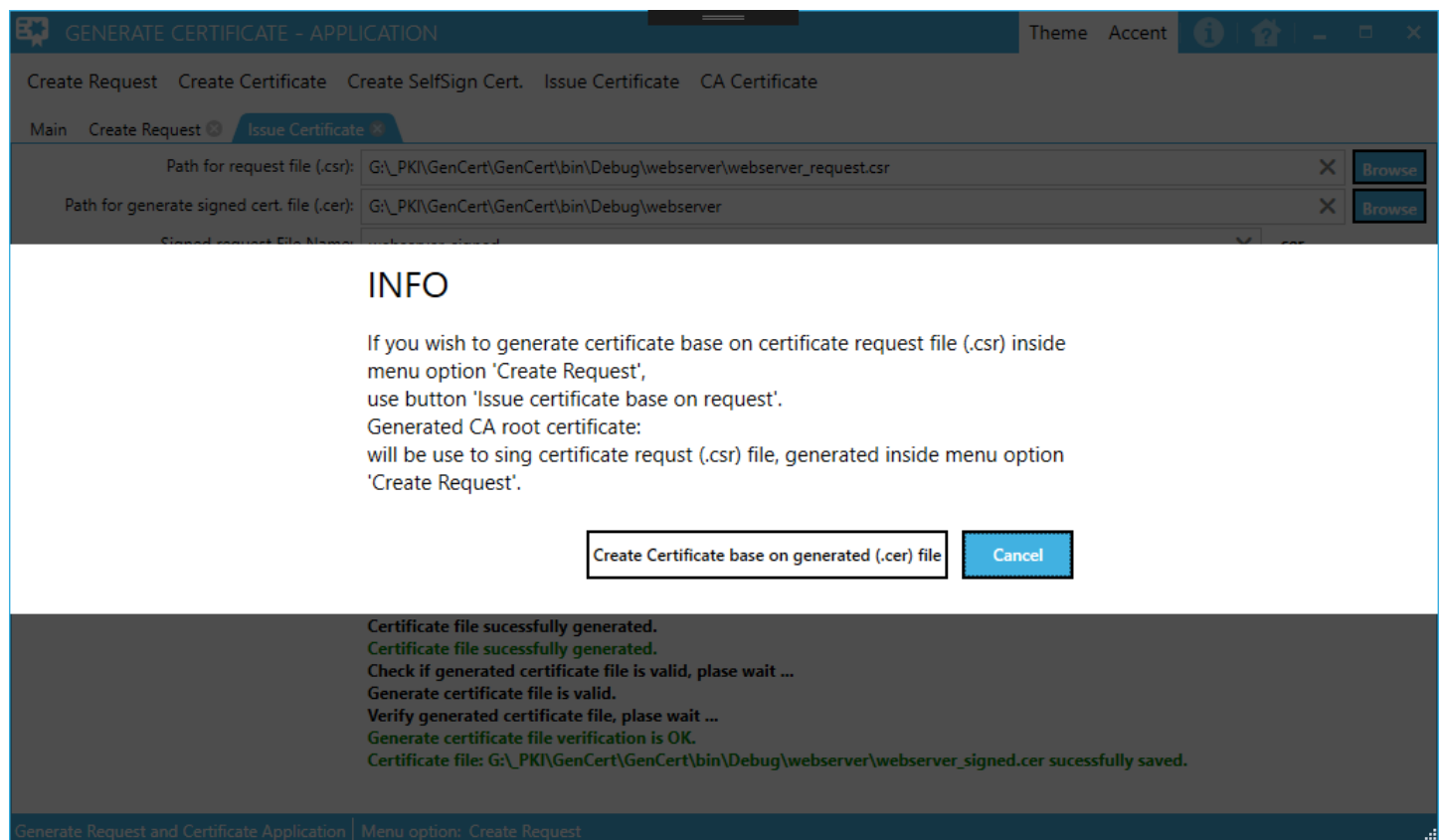
End Date: 12.3.2020.

Generate Request and Certificate Application | Menu option: Create Request

If all files successfully generated, button Continue will be enabled.



If you click Continue button, you will get dialog wizard to explain what to do next.



6. CA CERTIFICATE

This option is used to create signed certificate file base on request certificate file which will be signed by generated CA root certificate file inside menu option "Create SelfSign Cert.".

The screenshot displays the GenCert application interface. The top menu bar includes 'Generate Certificate - Application' and a sub-menu with 'Create Request', 'Create Certificate', 'Create SelfSign Cert.', 'Issue Certificate', and 'CA Certificate'. The 'CA Certificate' option is highlighted with a red box. Below the menu bar, the 'Main' tab is active, and the 'CA Certificate' sub-tab is selected. The main form is titled 'Master CA :'. It contains several input fields for configuring a Master CA, including 'Common Name', 'Key Length', 'Signature Algorithm', 'Country Code', 'State or Province Name', 'Locality Name', 'Organization', 'Start Date', 'End Date', 'Path for generate files', 'Public Key File Name', 'Signed Key File Name', 'Password for export', and 'Certificate Friendly name'. The 'Path for generate files' field has a 'Browse' button. At the bottom of the form, there are buttons for 'Test Data L1', 'Test Data L2', 'Test Data L3', 'Generate', 'Continue', and 'Clean data'. The bottom status bar indicates 'Generate Request and Certificate Application | Menu option: CA Certificate'.

Master CA :

Common Name : For example masterCA.test.local X

Key Length: 4096 X

Signature Algorithm: SHA512WITHRSA X

Country Code: For example "RS" X

State or Province Name: For example "Serbia" X

Locality Name: For example "Novi Sad" X

Organization: For example "Master CA CO" X

Start Date: 14.3.2019. 14

End Date: 14.3.2019. 14

Path for generate files : G:_PKI\GenCert\GenCert\bin\Debug X Browse

Public Key File Name: Name for Master cert. public key file without extension X .cer

Signed Key File Name: Name for Master cert. signed key file without extension X .pfx

Password for export: Password

Certificate Friendly name: Master CA name X

Test Data L1

Test Data L2

Test Data L3

Generate

Continue

Clean data

Generate Request and Certificate Application | Menu option: CA Certificate

This form has tree expander form parts.

The first expander form can be used to enter data for master CA (usually this is workgroup computer, keep as offline).

The second expander form can be used to enter data for intermediate CA (usually this computer is part of some Active Directory domain, always online – for II level CA infrastructure, use to issue certificate for computers and users, for III level CA infrastructure, use to issue certificate for issuer CA server).

The third expander form can be used to enter data for issuer CA (usually this computer is part of some Active Directory domain, always online – for III level CA infrastructure, use to issue certificate for computers and users).

The screenshot displays the 'GENERATE CERTIFICATE - APPLICATION' window. It features a top navigation bar with tabs: 'Create Request', 'Create Certificate', 'Create SelfSign Cert.', 'Issue Certificate', and 'CA Certificate'. The 'CA Certificate' tab is active. Below the navigation bar, there are three main sections, each with an 'Expand / Collapse' button on the left. The sections are:

- Master CA :** Fields include Common Name (For example masterCA.test.local), Key Length (4096), Signature Algorithm (SHA512WITHRSA), Country Code (For example "RS"), State or Province Name (For example "Serbia"), Locality Name (For example "Novi Sad"), Organization (For example "Master CA CO"), Start Date (14.3.2019), End Date (14.3.2019), Path for generate files (G:\PKI\GenCert\GenCert\bin\Debug), Public Key File Name (Name for Master cert. public key file without extension), Signed Key File Name (Name for Master cert. signed key file without extension), Password for export (Password), and Certificate Friendly name (Master CA name). Buttons: Test Data L1, Test Data L2, Test Data L3, Generate, Clean data.
- Intermediate CA :** Fields include Common Name (For example intermediateCA.test.local), Key Length (2048), Signature Algorithm (SHA512WITHRSA), Country Code (For example "RS"), State or Province Name (For example "Serbia"), Locality Name (For example "Novi Sad"), Organization (For example "Intermediate CA CO"), Start Date (14.3.2019), End Date (14.3.2019), Path for generate files (G:\PKI\GenCert\GenCert\bin\Debug), Public Key File Name (Name for Intermediate CA cert. public key file without extension), Signed Key File Name (Name for Intermediate CA cert. signed key file without extension), Password for export (Password), and Certificate Friendly name (Intermediate CA name). Buttons: Test Data L1, Test Data L2, Test Data L3, Generate, Clean data.
- Issuer CA :** Fields include Common Name (For example issuerCA.test.local), Key Length (1024), Signature Algorithm (SHA512WITHRSA), Country Code (For example "RS"), State or Province Name (For example "Serbia"), Locality Name (For example "Novi Sad"), Organization (For example "Issuer CA CO"), Start Date (14.3.2019), End Date (14.3.2019), Path for generate files (G:\PKI\GenCert\GenCert\bin\Debug), Public Key File Name (Name for Issuer CA cert. public key file without extension), Signed Key File Name (Name for Issuer CA cert. signed key file without extension), Password for export (Password), and Certificate Friendly name (Issuer CA name). Buttons: Test Data L1, Test Data L2, Test Data L3, Generate, Clean data.

Buttons:

"Test Data L1" - This button will fill master CA form with entered data. Use this button if you generate only mater CA certificate to sign certificate request

"Test Data L2" - This button will fill master CA + intermediate CA forms with entered data. Use this button if you generate mater CA + intermediate CA certificates to sign certificate request

"Test Data L3" - This button will fill master CA + intermediate CA + issuer CA forms with entered data. Use this button if you generate mater CA + intermediate CA + issuer CA certificates to sign certificate request

"Clean data" - This button will clean entered data on current form

When click one of "Test Data" buttons, application will open new form to enter data that will be used to automatically generate data these need to generate appropriate certificates files.

GENERATE CERTIFICATE - APPLICATION

Theme Accent

Create Request Create Certificate Create SelfSign Cert. Issue Certificate CA Certificate

Main CA Certificate

Expand / Collapse master CA

Master CA :

Common Name : For example masterCA.test.local

Key Length: 4096

Signature Algorithm: SHA512WITHRSA

Country Code: For example "RS"

State or Province Name: For example "Serbia"

Locality Name: For example "Novi Sad"

Organization: For example "Master CA CO"

Start Date: 14.3.2019.

End Date: 14.3.2019.

Path for generate files : G:\PKI\GenCert\GenCert\bin\Debug

Public Key File Name: Name for Master cert. public key file with

Signed Key File Name: Name for Master cert. signed key file with

Password for export: Password

Certificate Friendly name: Master CA name

Test Data L1

Test Data L2

Test Data L3

Generate

Continue

Clean data

CA PARAMETERS

Organization : For example ABC

Domain name : For example: test.local

Country Code : For example: RS

State or Province name : For example: Serbia

Locality Name : For example: Novi Sad

Password : Password

Ok Cancel

Generate Request and Certificate Application | Menu option: CA Certificate

On opened form for "CA PARAMETERS", enter data and press "Ok" button to continue.

CA PARAMETERS

Organization : Proba

Domain name : dmz.local

Country Code : RS

State or Province name : Serbia

Locality Name : Belgrad

Password : ●●●●●●●●

Ok Cancel

If data entered correctly, application will automatically generate and fill form(s) with appropriate data.

GENERATE CERTIFICATE - APPLICATION

Theme Accent

Create Request Create Certificate Create SelfSign Cert. Issue Certificate CA Certificate

Main CA Certificate

Expand / Collapse master CA

Master CA :

Common Name: Proba masterCA.dmz.local

Key Length: 4096

Signature Algorithm: SHA512WITHRSA

Country Code: RS

State or Province Name: Serbia

Locality Name: Belgrad

Organization: Master CA CO

Start Date: 11.3.2019.

End Date: 14.3.2029.

Path for generate files: G:_PKI\GenCert\GenCert\bin\Debug\masterCA

Public Key File Name: masterCA_public

Signed Key File Name: masterCA

Password for export:

Certificate Friendly name: Proba Master CA

Test Data L1

Test Data L2

Test Data L3

Generate

Continue

Clean data

Expand / Collapse Intermediate CA

Intermediate CA :

Common Name: For example

Key Length: 2048

Signature Algorithm: SHA512WITHRSA

Country Code: For example

State or Province Name: For example

Locality Name: For example

Organization: For example

Start Date: 14.3.2019.

End Date: 14.3.2019.

Path for generate files: G:_PKI\GenCert\GenCert\bin\Debug\intermediateCA

Public Key File Name: Name for Intermediate CA

Signed Key File Name: Name for Intermediate CA

Password for export: Password

Certificate Friendly name: Intermediate CA

Continue

Clean data

Generate Request and Certificate Application | Menu option: CA Certificate

GENERATE CERTIFICATE - APPLICATION

Theme Accent

Create Request Create Certificate Create SelfSign Cert. Issue Certificate CA Certificate

Main CA Certificate

Expand / Collapse master CA

Master CA :

Common Name: Proba masterCA.dmz.local

Key Length: 4096

Signature Algorithm: SHA512WITHRSA

Country Code: RS

State or Province Name: Serbia

Locality Name: Belgrad

Organization: Master CA CO

Start Date: 11.3.2019.

End Date: 14.3.2029.

Path for generate files: G:_PKI\GenCert\GenCert\bin\Debug\masterCA

Public Key File Name: masterCA_public

Signed Key File Name: masterCA

Password for export:

Certificate Friendly name: Proba Master CA

Test Data L1

Test Data L2

Test Data L3

Generate

Continue

Clean data

Expand / Collapse Intermediate CA

Intermediate CA :

Common Name: Proba intermediateCA.dmz.local

Key Length: 2048

Signature Algorithm: SHA512WITHRSA

Country Code: RS

State or Province Name: Serbia

Locality Name: Belgrad

Organization: Intermediate CA CO

Start Date: 12.3.2019.

End Date: 14.3.2028.

Path for generate files: G:_PKI\GenCert\GenCert\bin\Debug\intermediateCA

Public Key File Name: intermediateCA_public

Signed Key File Name: intermediateCA

Password for export:

Certificate Friendly name: Proba Intermediate CA

Continue

Clean data

Expand / Collapse Issuer CA

Issuer CA :

Common Name: Proba issuerCA.dmz.local

Key Length: 1024

Signature Algorithm: SHA512WITHRSA

Country Code: RS

State or Province Name: Serbia

Locality Name: Belgrad

Organization: Issuer CA CO

Start Date: 13.3.2019.

End Date: 14.3.2027.

Path for generate files: G:_PKI\GenCert\GenCert\bin\Debug\issuerCA

Public Key File Name: issuerCA_public

Signed Key File Name: issuerCA

Password for export:

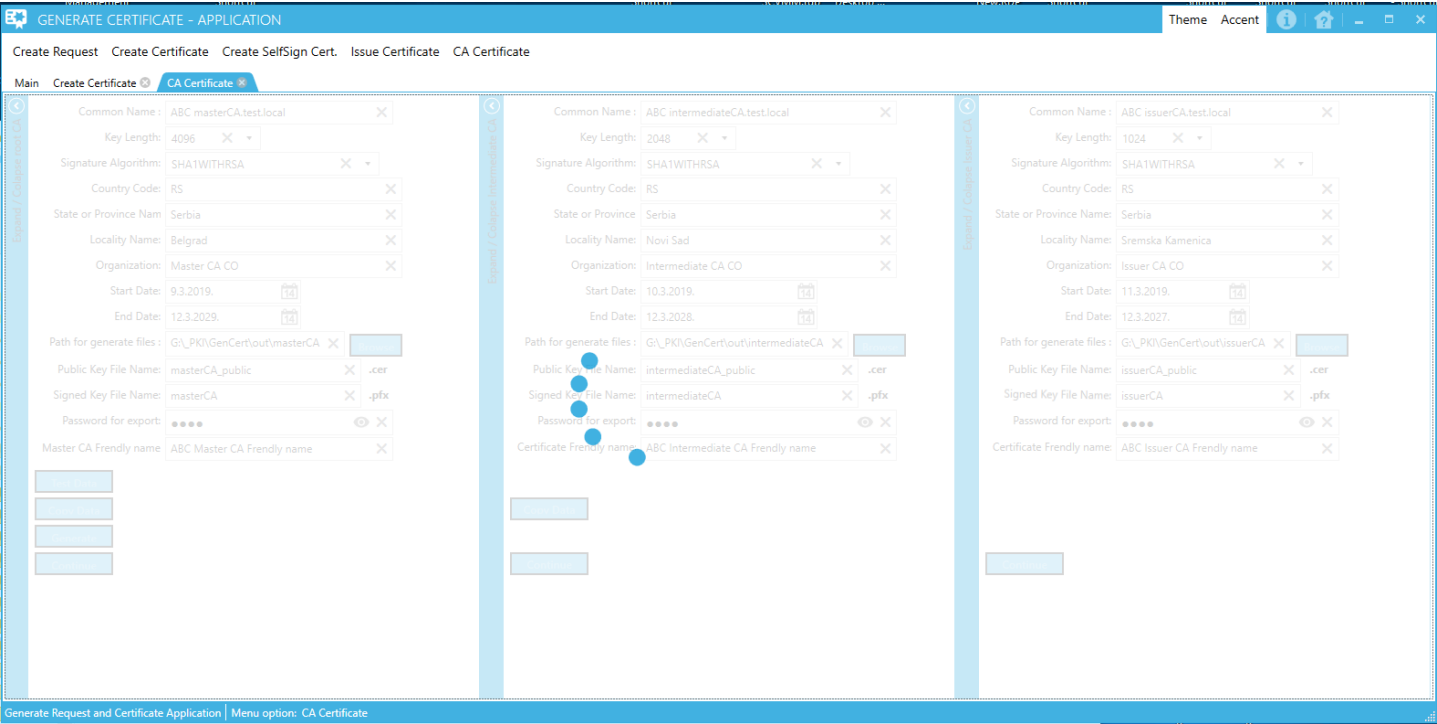
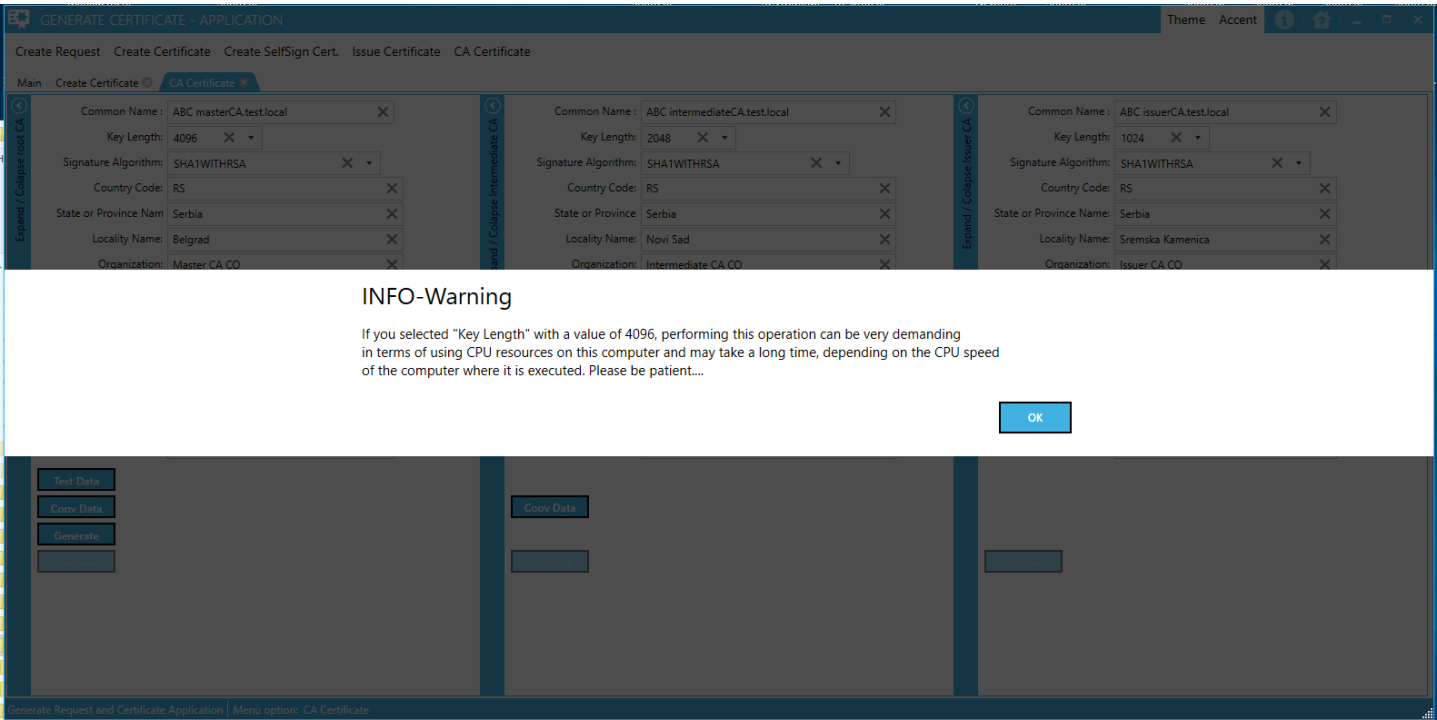
Certificate Friendly name: Proba Issuer CA

Continue

Clean data

Generate Request and Certificate Application | Menu option: CA Certificate

When you click “Generate” button you will get next message



Master CA :

Common Name: Proba masterCA.dms.local

Key Length: 4096

Signature Algorithm: SHA512WITHRSA

Country Code: RS

State or Province Name: Serbia

Locality Name: Belgrad

Organization: Master CA CO

Start Date: 11.3.2019.

End Date: 14.3.2029.

Path for generate files: G:\PKI\GenCert\GenCert\bin\Debug\masterCA

Public Key File Name: masterCA_public

Signed Key File Name: masterCA

Password for export:

Certificate Friendly name: Proba Master CA

Intermediate CA :

Common Name: Proba intermediateCA.dms.local

Key Length: 2048

Signature Algorithm: SHA512WITHRSA

Country Code: RS

State or Province Name: Serbia

Locality Name: Belgrad

Organization: Intermediate CA CO

Start Date: 12.3.2019.

End Date: 14.3.2028.

Path for generate files: G:\PKI\GenCert\GenCert\bin\Debug\intermediateCA

Public Key File Name: intermediateCA_public

Signed Key File Name: intermediateCA

Password for export:

Certificate Friendly name: Proba Intermediate CA

Issuer CA :

Common Name: Proba issuerCA.dms.local

Key Length: 1024

Signature Algorithm: SHA512WITHRSA

Country Code: RS

State or Province Name: Serbia

Locality Name: Belgrad

Organization: Issuer CA CO

Start Date: 13.3.2019.

End Date: 14.3.2027.

Path for generate files: G:\PKI\GenCert\GenCert\bin\Debug\issuerCA

Public Key File Name: issuerCA_public

Signed Key File Name: issuerCA

Password for export:

Certificate Friendly name: Proba Issuer CA

When you successfully generate appropriate CA certificate file(s), button “Continue” will be enabled.

If you click on “Continue” button, you will get next dialog:

Master CA :

Common Name: Proba masterCA.dms.local

Key Length: 4096

Signature Algorithm: SHA512WITHRSA

Country Code: RS

State or Province Name: Serbia

Locality Name: Belgrad

Organization: Master CA CO

Start Date: 11.3.2019.

End Date: 14.3.2029.

Path for generate files: G:\PKI\GenCert\GenCert\bin\Debug\masterCA

Public Key File Name: masterCA_public

Signed Key File Name: masterCA

Password for export:

Certificate Friendly name: Proba Master CA

Intermediate CA :

Common Name: Proba intermediateCA.dms.local

Key Length: 2048

Signature Algorithm: SHA512WITHRSA

Country Code: RS

State or Province Name: Serbia

Locality Name: Belgrad

Organization: Intermediate CA CO

Start Date: 12.3.2019.

End Date: 14.3.2028.

Path for generate files: G:\PKI\GenCert\GenCert\bin\Debug\intermediateCA

Public Key File Name: intermediateCA_public

Signed Key File Name: intermediateCA

Password for export:

Certificate Friendly name: Proba Intermediate CA

Issuer CA :

Common Name: Proba issuerCA.dms.local

Key Length: 1024

Signature Algorithm: SHA512WITHRSA

Country Code: RS

State or Province Name: Serbia

Locality Name: Belgrad

Organization: Issuer CA CO

Start Date: 13.3.2019.

End Date: 14.3.2027.

Path for generate files: G:\PKI\GenCert\GenCert\bin\Debug\issuerCA

Public Key File Name: issuerCA_public

Signed Key File Name: issuerCA

Password for export:

Certificate Friendly name: Proba Issuer CA

INFO

If you wish to generate certificate base on certificate request file (.csr) inside menu option 'Create Request', use button 'Issue certificate base on request'.
Generated CA root certificate: masterCA
will be use to sing certificate request (.csr) file, generated inside menu option 'Create Request'.

Issue certificate base on request Cancel

If click Cancel you will stay on current form or if you click button “Issue certificate base on request”, application will open new form for issuing (sign) certificate file, based on created request file.

The screenshot shows the 'GENERATE CERTIFICATE - APPLICATION' window with the 'Issue Certificate' tab selected. The 'Path CA certificate with private key (.pfx)' field is highlighted with a red box, containing the path 'G:_PKI\GenCert\GenCert\bin\Debug\issuerCA\issuerCA.pfx'. Other fields include 'Path for request file (.csr)' with path 'G:_PKI\GenCert\out\DemoCertFiles\webserver\webserver_request.csr', 'Path for generate signed cert. file (.cer)' with path 'G:_PKI\GenCert\out\DemoCertFiles\webserver', 'Signed request File Name' with value 'webserver_signed', 'Password for CA private key' with masked characters, 'Start Date' as '14.3.2019.', and 'End Date' as '14.3.2020.'. Buttons for 'Generate' and 'Continue' are at the bottom left. The status bar at the bottom indicates 'Generate Request and Certificate Application | Menu option: CA Certificate'.

Based on generated certificate file(s), (masterCA / intermediateCA / issuerCA) new form for issue certificate will be automatically fill field "Path CA certificate with private key (.pfx)" with appropriate generated certificate file.

This means, if you generate only certificate for masterCA and click Continue button, field "Path CA certificate with private key (.pfx)" will be filled with masterCA file path.

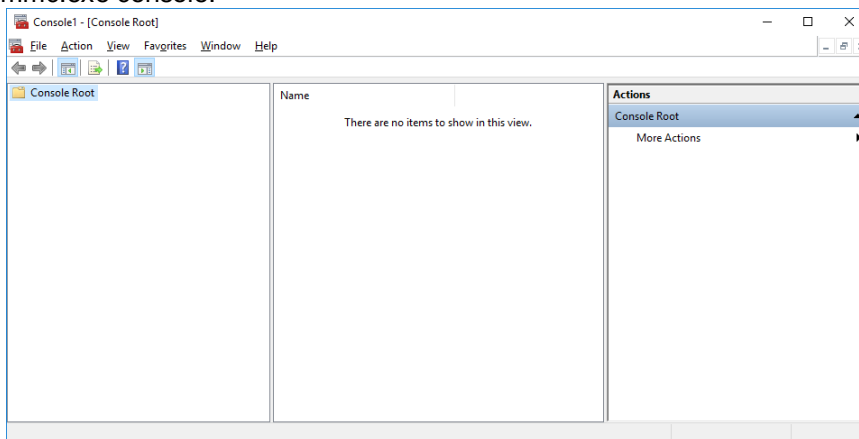
If you generate masterCA and intermediateCA and click Continue button, field "Path CA certificate with private key (.pfx)" will be filled with intermediateCA file path.

If you generate masterCA, intermediateCA and issuerCA and click Continue button, field "Path CA certificate with private key (.pfx)" will be filled with issuerCA file path.

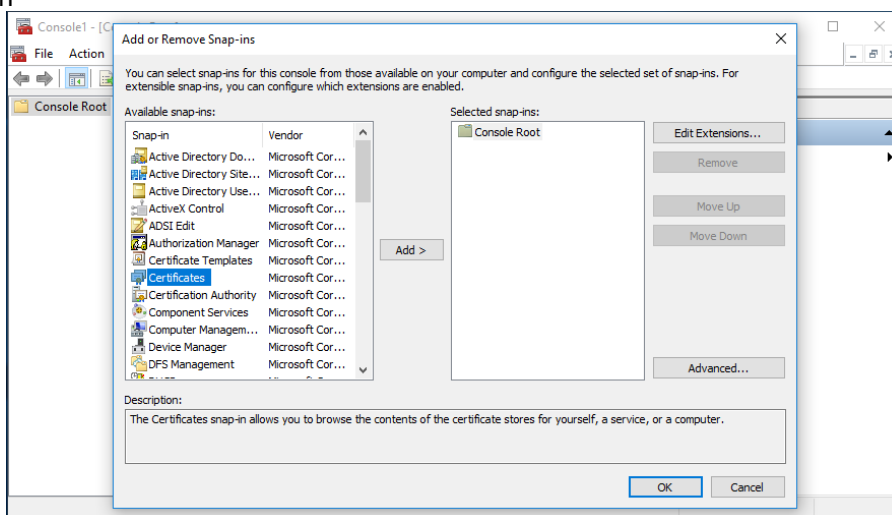
Field "Path CA certificate with private key (.pfx)" always will be filled with last generated certificate file path inside certificates CA chain.

7. IMPORT DATA FROM GENERATED SIGNED CERTIFICATE FILE TO SERVER CERTIFICATE STORE

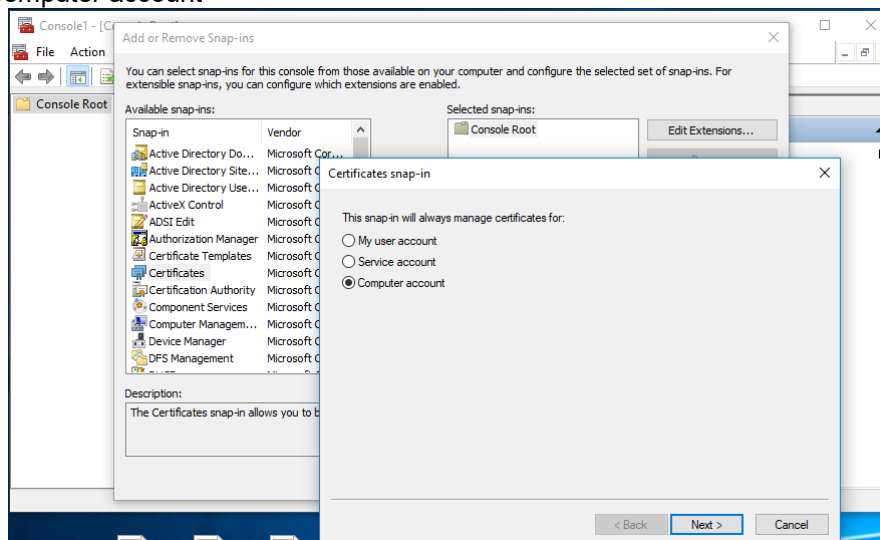
On the computer, open mmc.exe console.



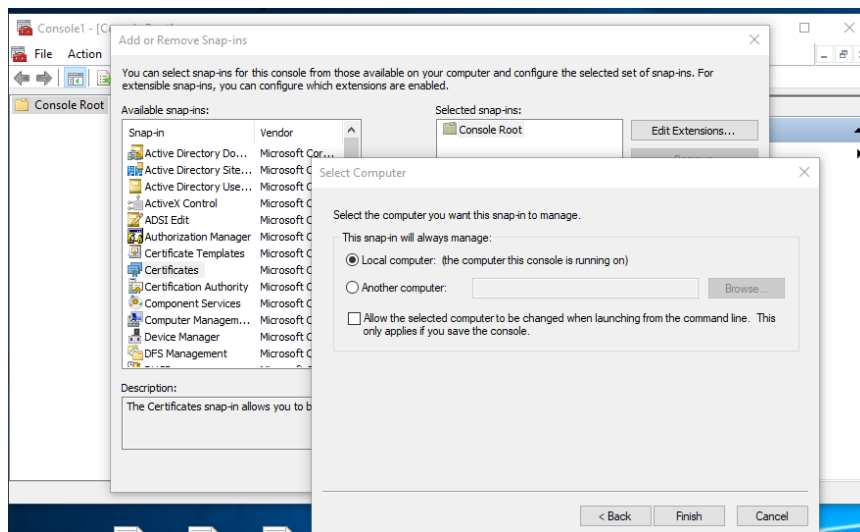
Add Certificate Snap-in



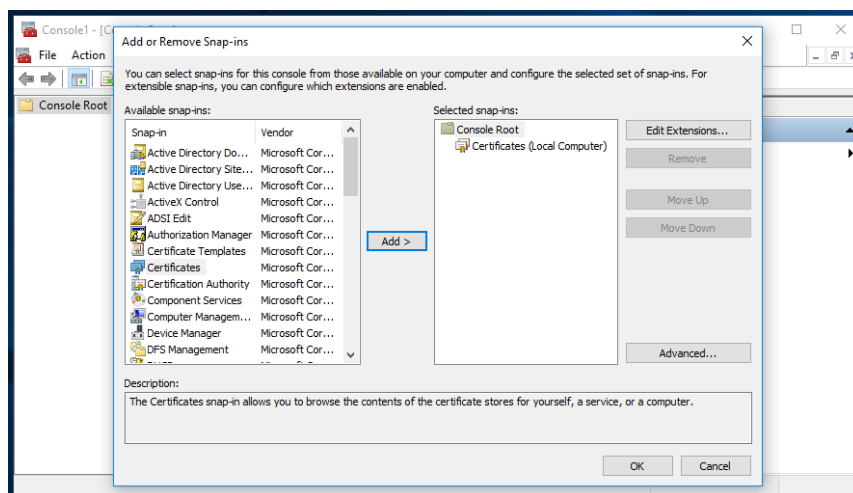
Add, Choose option "Computer account"



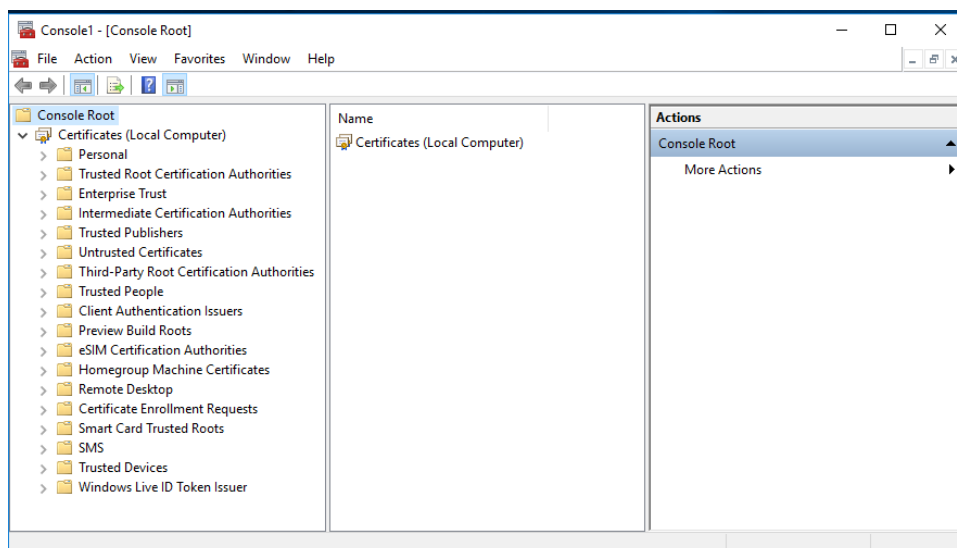
Next



Finish

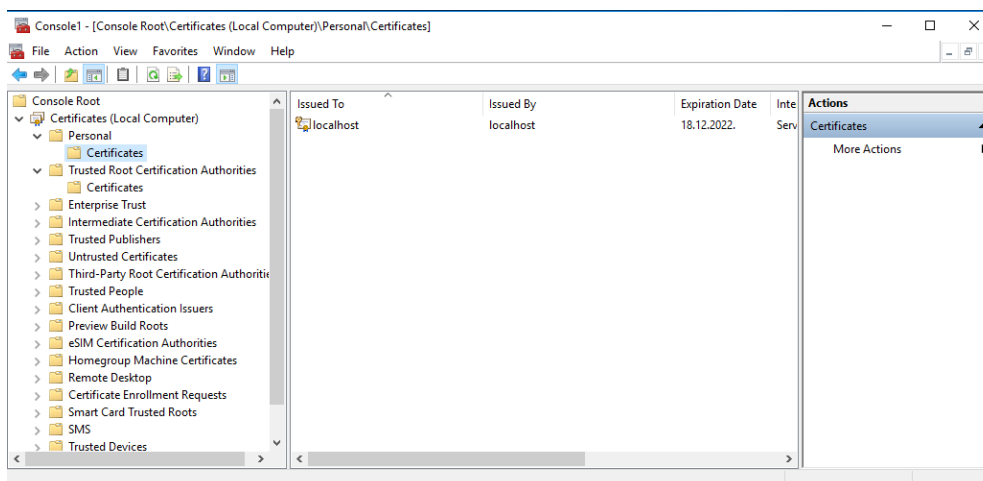


OK

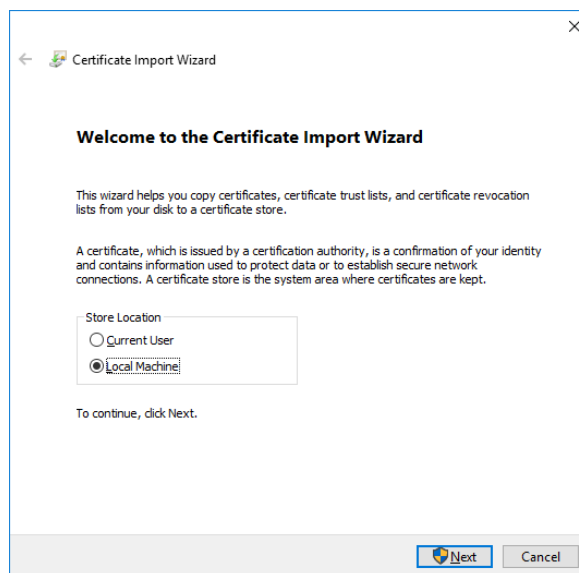


Parts of local computer Certificate store for interest are:

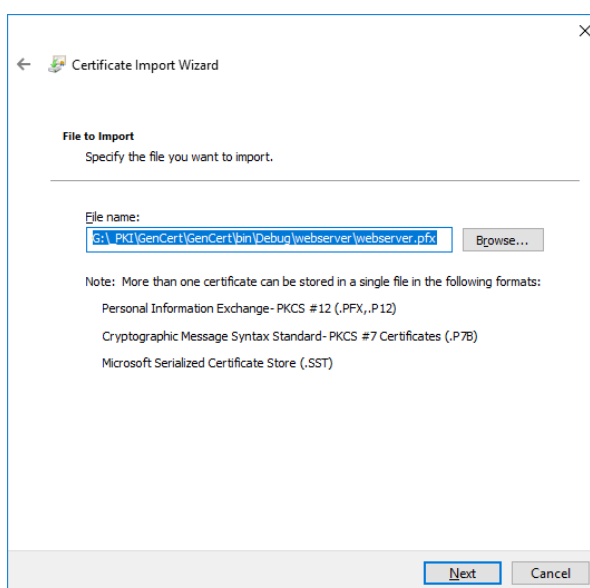
1. Personal -> Certificates
2. Trusted Root Certification Authorities -> Certificates
3. Intermediate Certification Authorities -> Certificates



Double click on generated signed certificate request file with .pfx extension will open Certificate Import Wizard. Select Store location -> Local Machine



Next



Next

Enter password you set when generated .pfx file. Optionally you can allow that this certificate can by exportable from certificate store by checking option “Mark this key as exportable.”

← Certificate Import Wizard

Private key protection

To maintain security, the private key was protected with a password.

Type the password for the private key.

Password:

•••••

☐ Display Password

Import options:

☐ Enable strong private key protection. You will be prompted every time the private key is used by an application if you enable this option.

☒ Mark this key as exportable. This will allow you to back up or transport your keys at a later time.

☐ Protect private key using virtualized-based security(Non-exportable)

☒ Include all extended properties.

Next

Cancel

Next

← Certificate Import Wizard

Certificate Store

Certificate stores are system areas where certificates are kept.

Windows can automatically select a certificate store, or you can specify a location for the certificate.

☒ Automatically select the certificate store based on the type of certificate

☐ Place all certificates in the following store

Certificate store:

Browse...

Next

Cancel

Next

← Certificate Import Wizard

Completing the Certificate Import Wizard

The certificate will be imported after you click Finish.

You have specified the following settings:

Certificate Store Selected

Automatically determined by the wizard

Content

PFX

File Name

G:_PKI\GenCert\GenCert\bin\Debug\webserver\webserver....

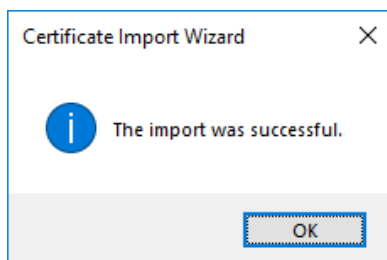
<

>

Finish

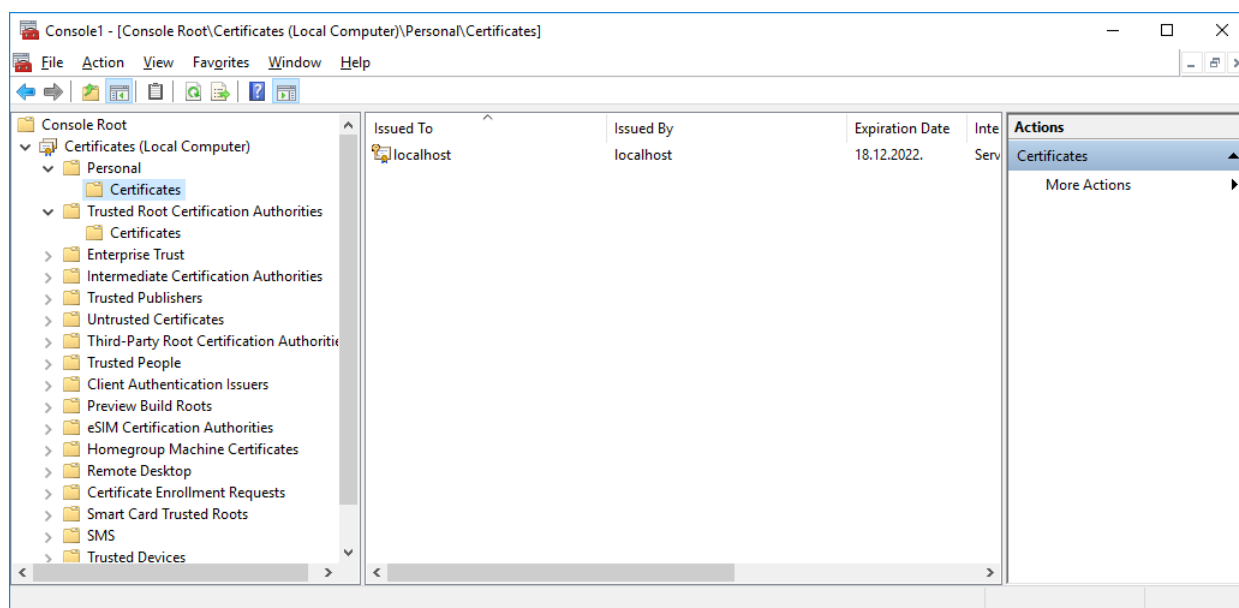
Cancel

Finish

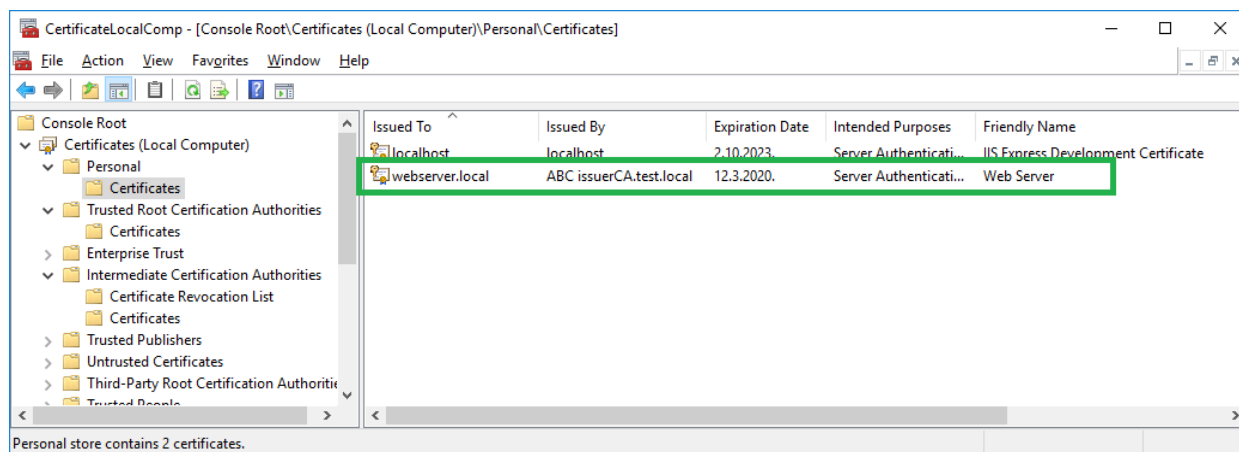


OK

Refresh Personal and Trusted Root Certification Authorities inside mmc console.



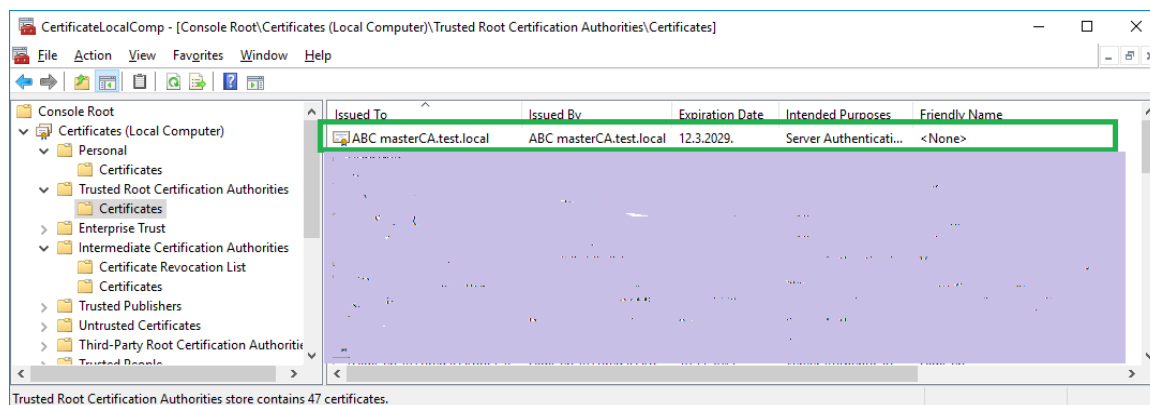
After refresh



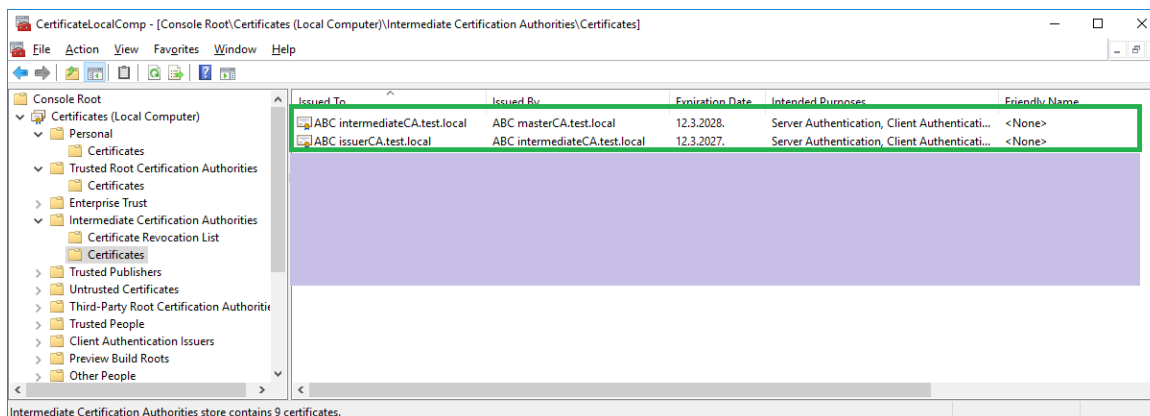
You can see data from imported certificate file inside Personal, Certificate store.

On the picture, you can see that certificate issued to webserver.local by CA root name "ABC issuerCA.test.local", expiration day for imported certificate and purpose (Server Authentication)

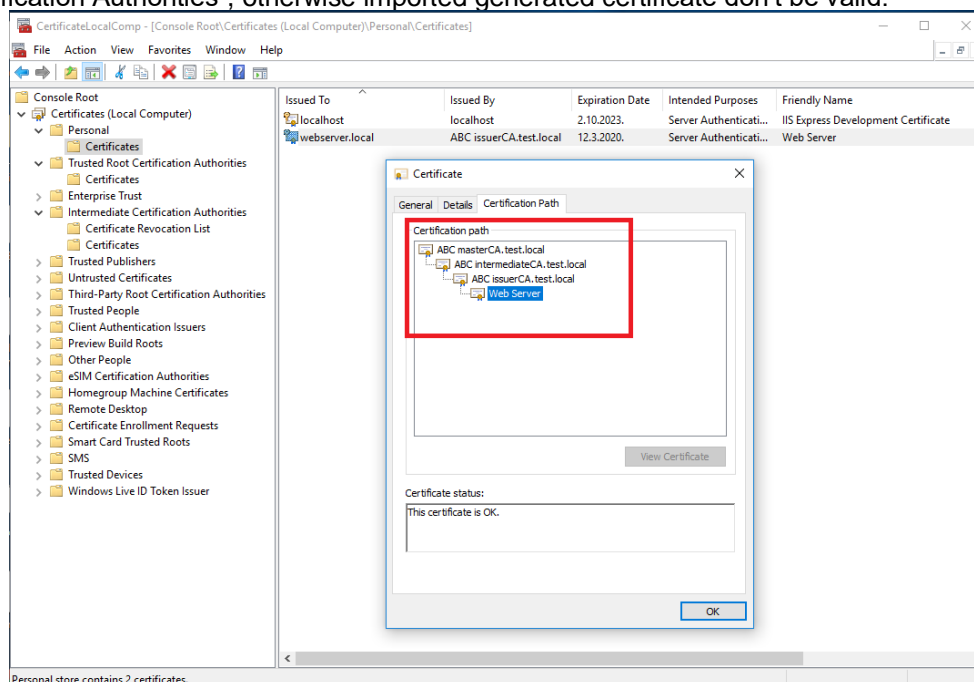
If Refresh "Trusted Root Certification Authorities" -> "Certificates" you can see certificate for root CA that signed our generated certificate file, that we imported when generate our certificate file. In our case, that is certificate named "ABC masterCA.test.local"



If Refresh “Intermediate Certification Authorities” -> “Certificates” you can see certificate for intermediate CA and issuer CA that signed our generated certificate file, that we imported when generate our certificate file. In our case, that is certificate named “ABC intermediateCA.test.local” and “ABC issuerCA.test.local”.

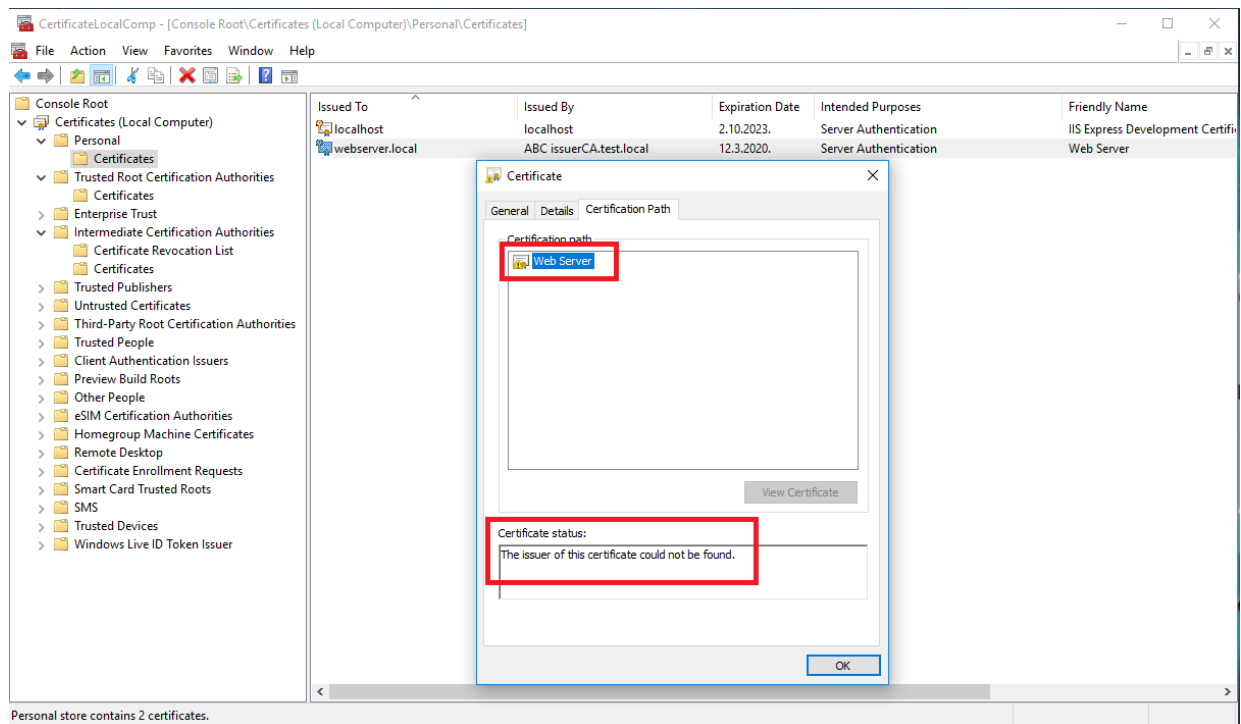


If you sign your certificate request file by some internal or external certificate authorities, you need to import data from that CA certificate authorities to “Trusted Root Certification Authorities” and if need also import appropriate certificate inside “Intermediate Certification Authorities”, otherwise imported generated certificate don't be valid.



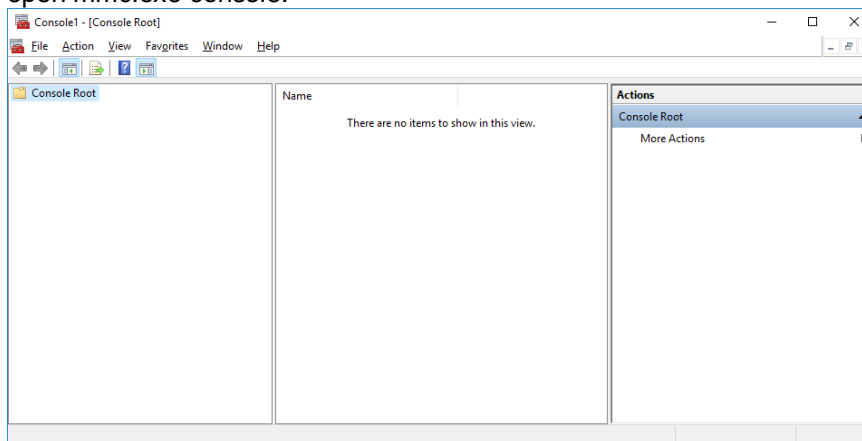
Valid imported certificate look.

If you didn't import data for CA root public key to certificate store, certificate look like this:

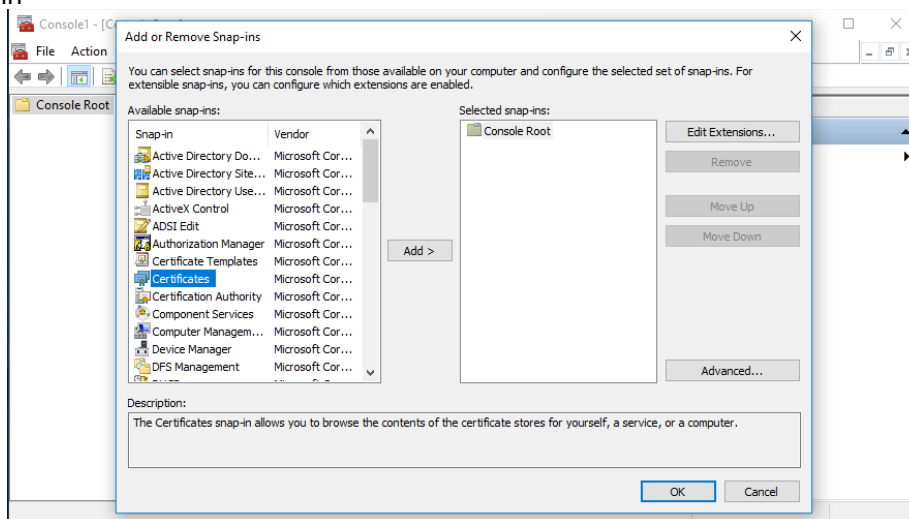


8. IMPORT DATA FROM GENERATED CA CERTIFICATE FILE(S) TO CLIENT CERTIFICATE STORE

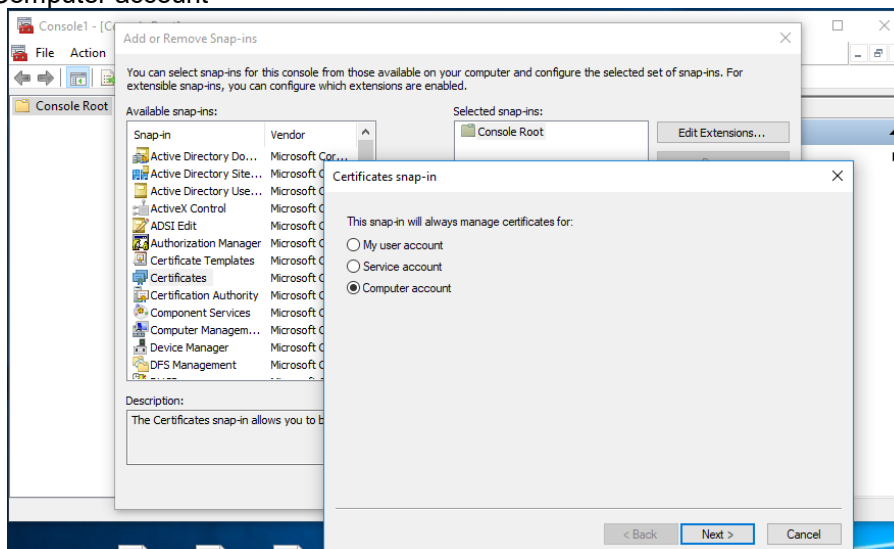
On the client computer, open mmc.exe console.



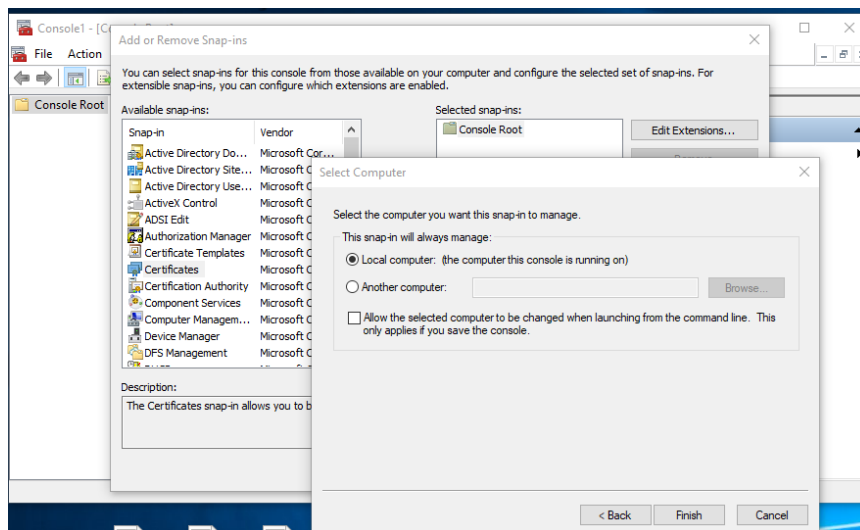
Add Certificate Snap-in



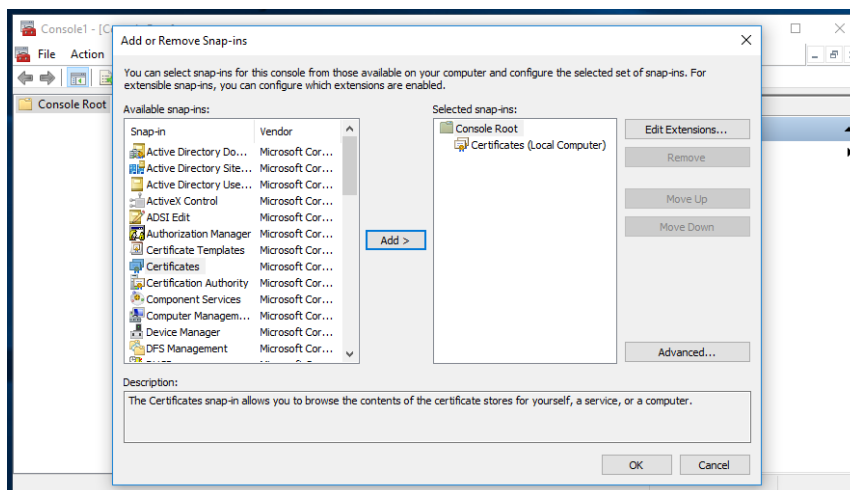
Add, Choose option "Computer account"



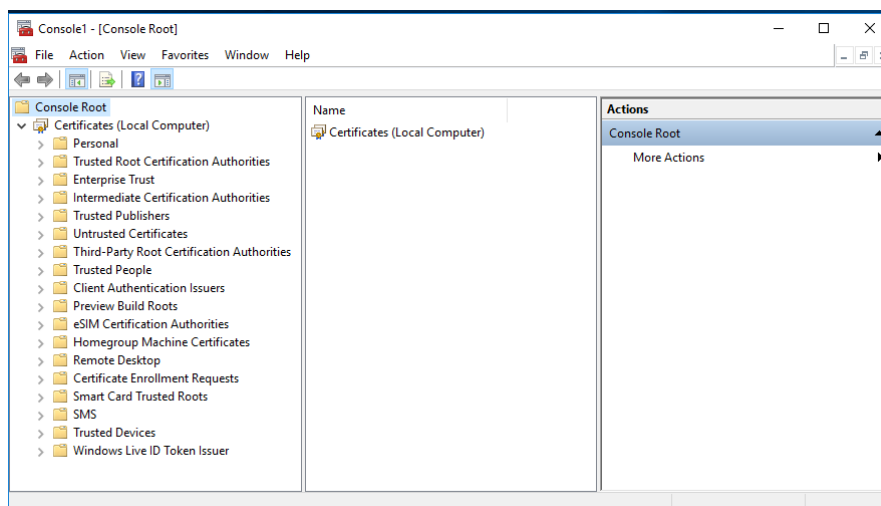
Next



Finish

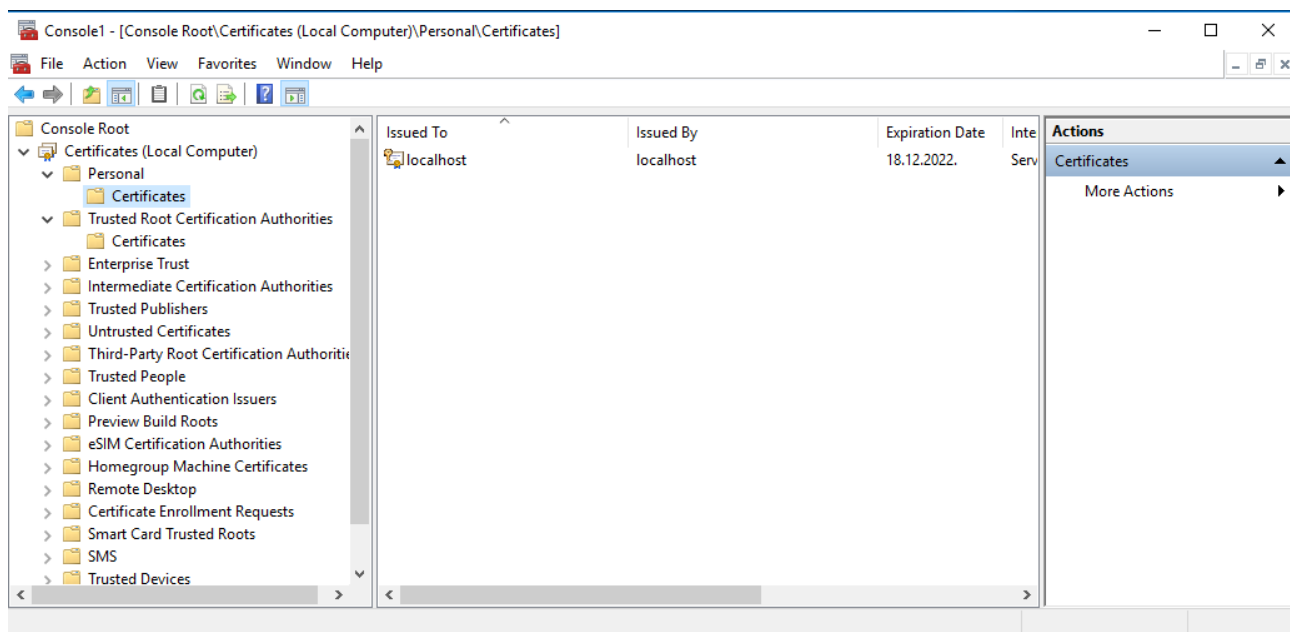


OK



Parts of local computer Certificate store for interest are:

1. Personal -> Certificates
2. Trusted Root Certification Authorities -> Certificates
3. Intermediate Certification Authorities -> Certificates

**NOTE:**

Always use CA file that was used when issuing certificate for server.

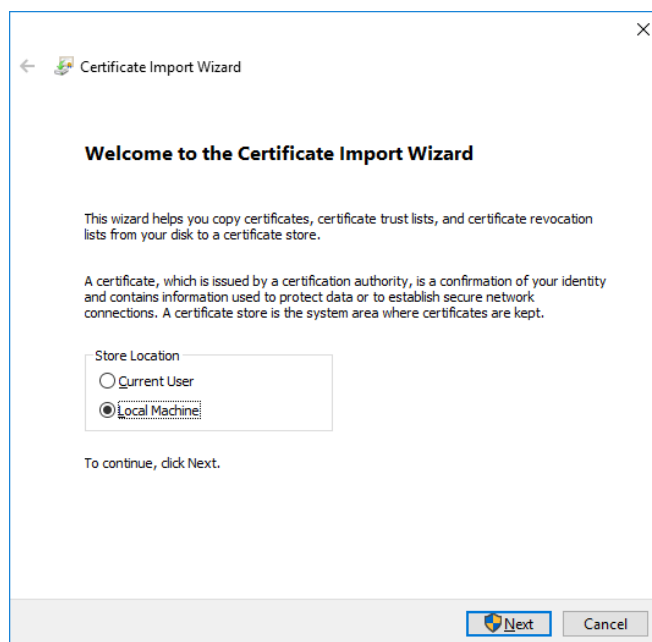
If you use one level CA (only master CA) to issue certificate for server use master CA .pfx file.

If you use two level CA (master CA + intermediate CA) to issue certificate for server use intermediate CA .pfx file.

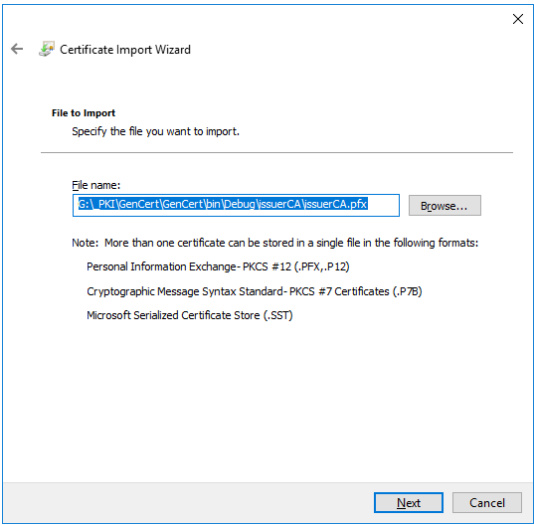
If you use three level CA (master CA + intermediate CA + issuer CA) to issue certificate for server use issuer CA .pfx file.

8.1. Import three level CA certificate (issuer CA)

Double click on generated CA file with .pfx extension will open Certificate Import Wizard. Select Store location -> Local Machine

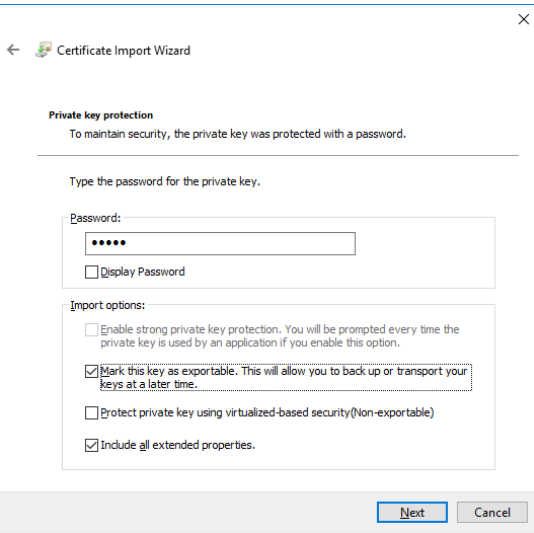


Next

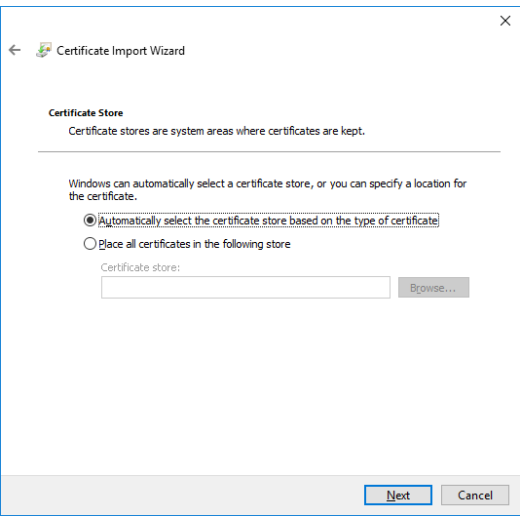


Next

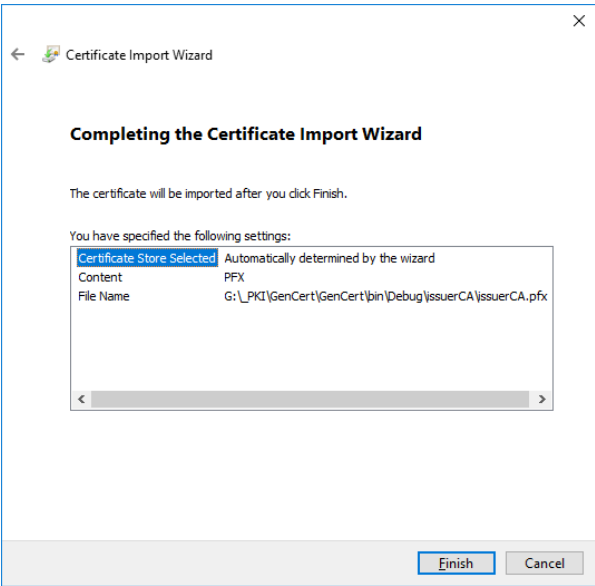
Enter password you set when generated .pfx file. Optionally you can allow that this certificate can by exportable from certificate store by checking option “Mark this key as exportable.”



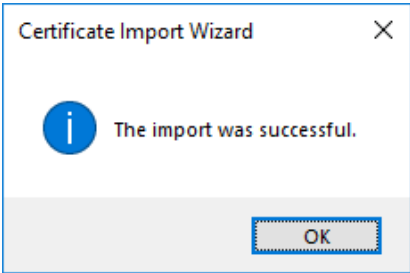
Next



Next



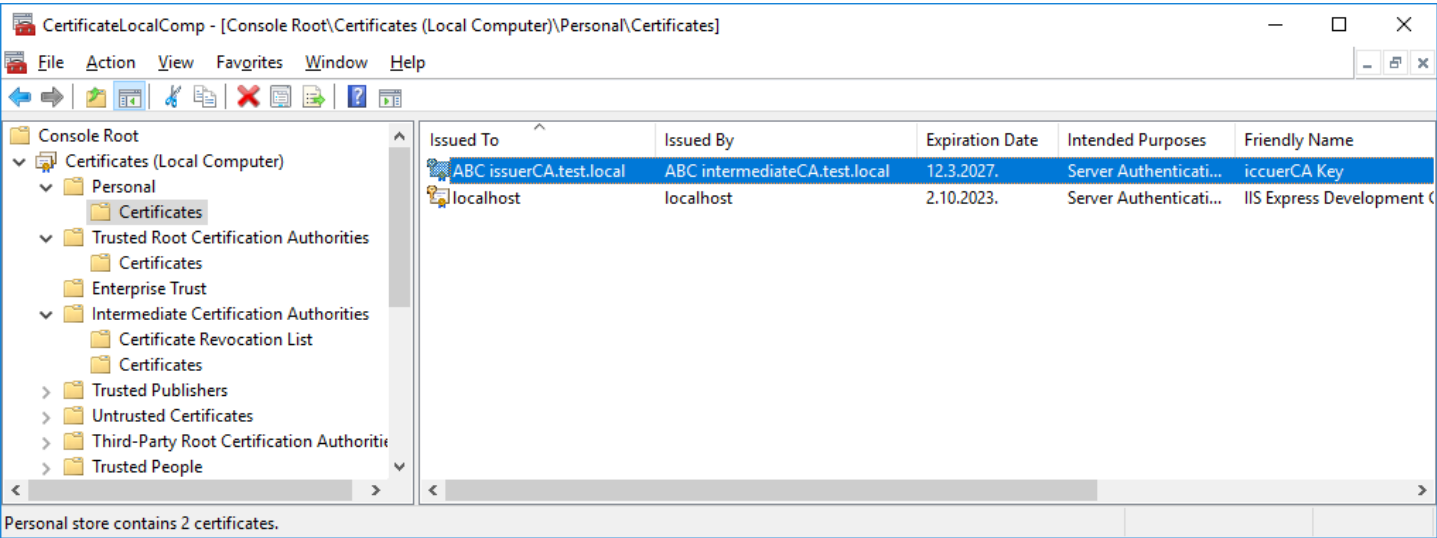
Finish

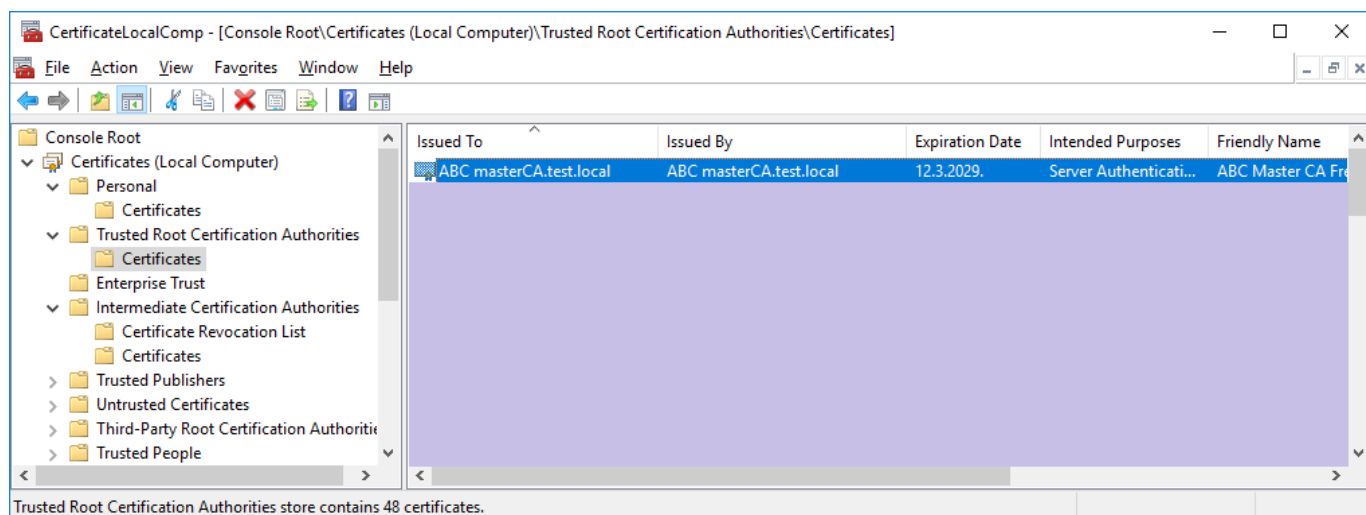
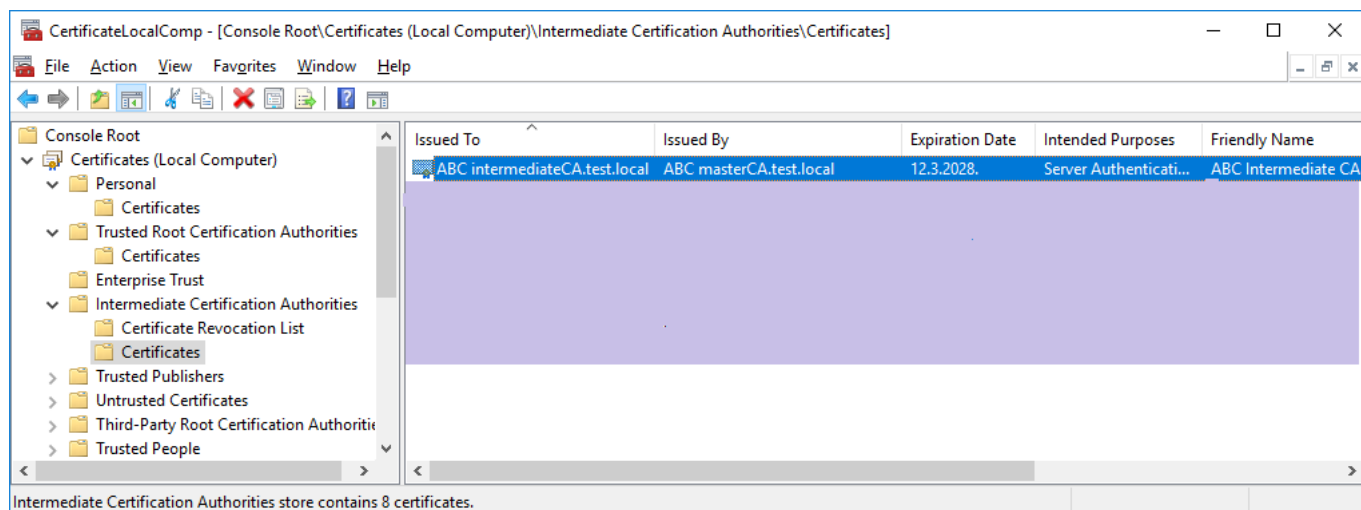


OK

If you import three level certificate (issuer CA) you need to refresh Personal, Trusted Root Certification Authorities and Intermediate Certification Authorities inside mmc console to see new imported certificates.

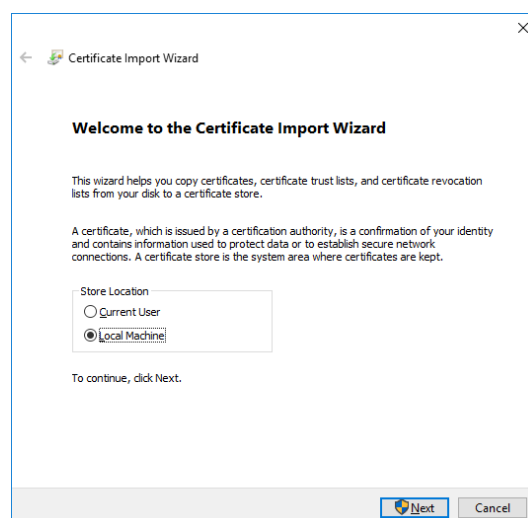
- issuer CA certificate will be found inside Personal->Certificates store.
- intermediate CA certificate will be found inside Intermediate Certification Authorities ->Certificates store.
- master CA certificate will be found inside Trusted Root Certification Authorities ->Certificates store.



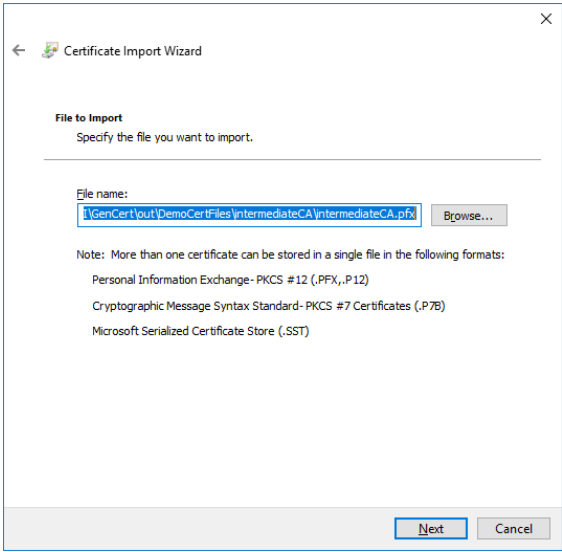


8.2. Import two level CA certificate (intermediate CA)

Double click on generated CA file with .pfx extension will open Certificate Import Wizard. Select Store location -> Local Machine

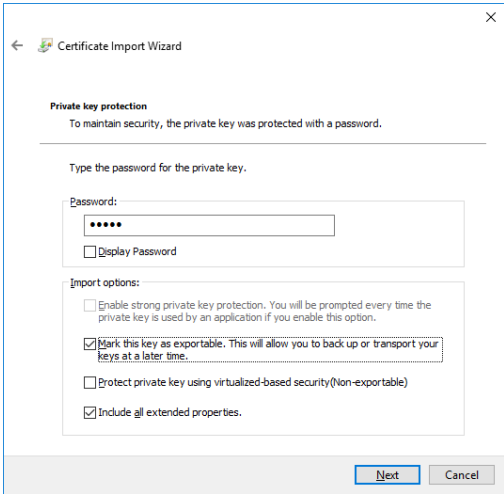


Next

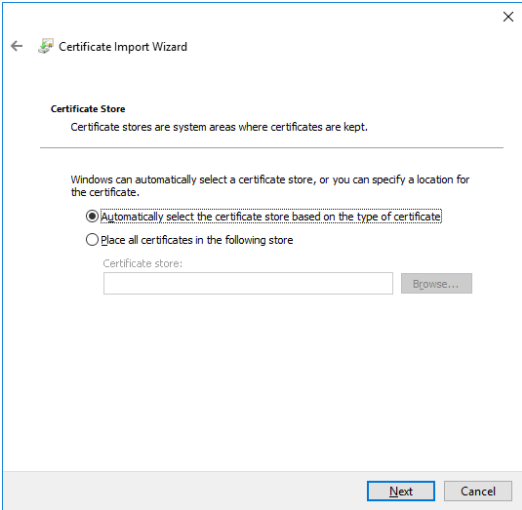


Next

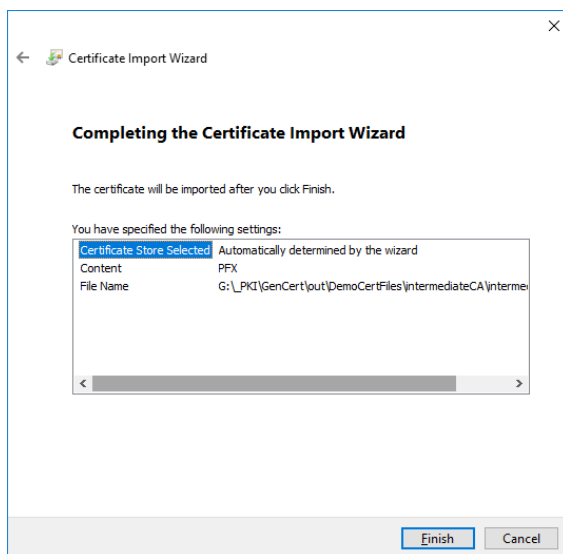
Enter password you set when generated .pfx file. Optionally you can allow that this certificate can by exportable from certificate store by checking option “Mark this key as exportable.”



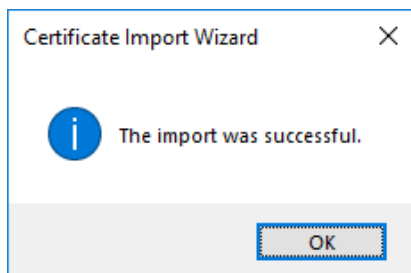
Next



Next

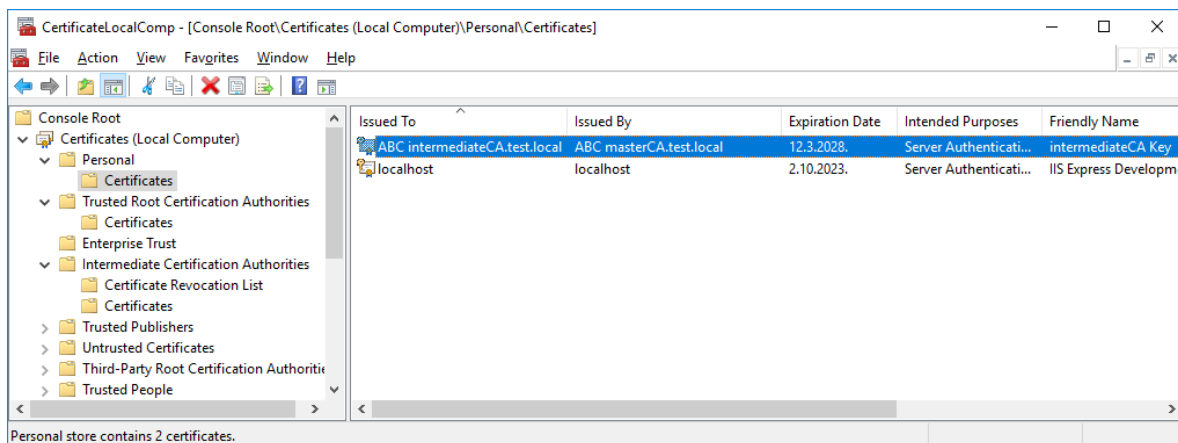


Finish

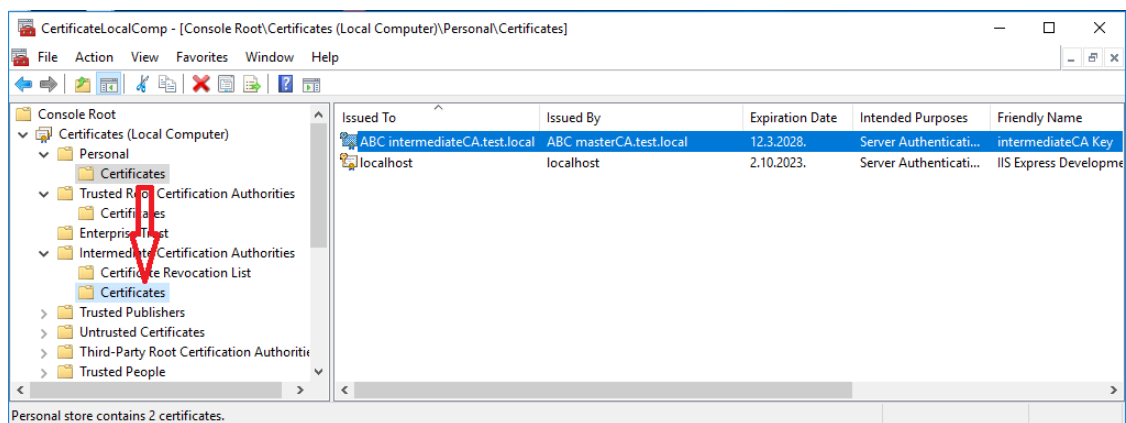


OK

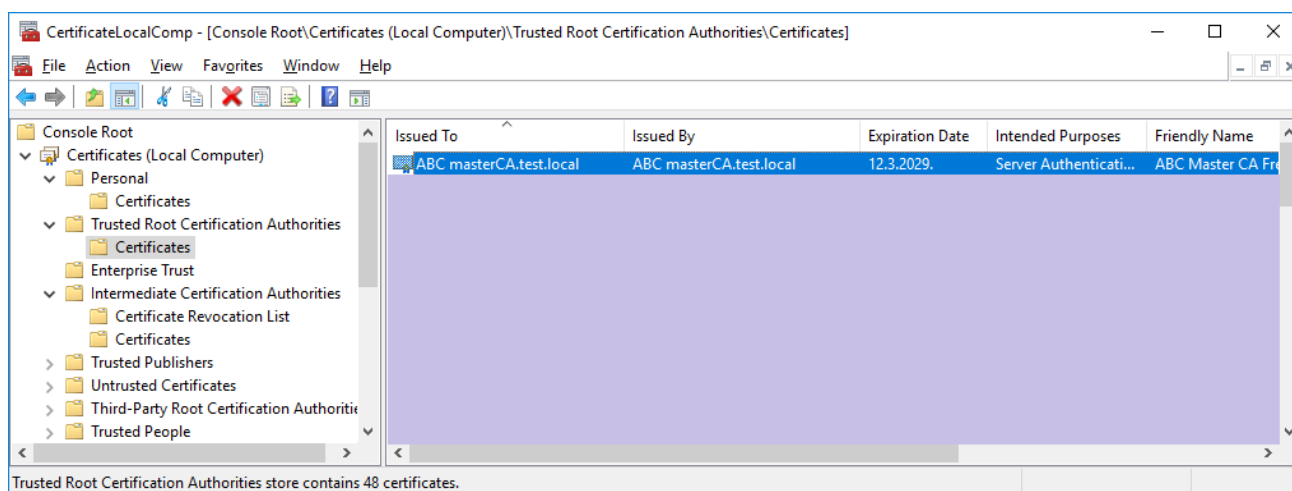
If you import two level certificate (intermediate CA) you need to refresh Personal, Trusted Root Certification Authorities. Intermediate CA certificate will be found inside Personal->Certificates store.



This certificate need to be moved to Intermediate Certification Authorities->Certificates store. Select imported intermediate certificate inside Personal->Certificates store and drag and drop that certificate to Intermediate Certification Authorities ->Certificates store.

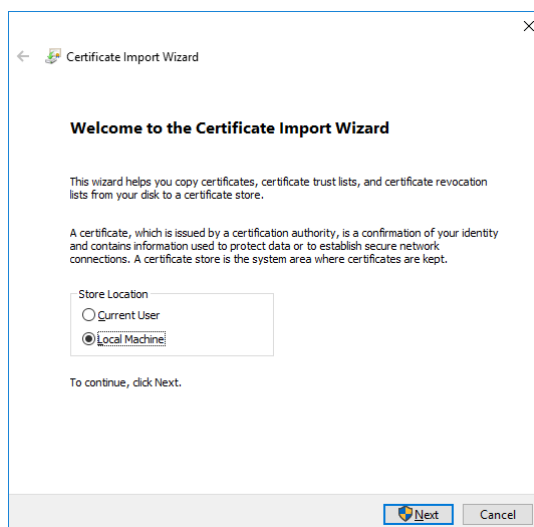


Master CA certificate will be found inside Trusted Root Certification Authorities->Certificates store.

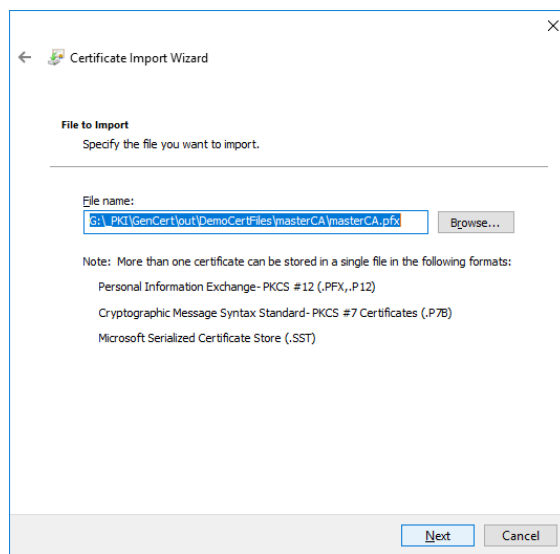


8.3. Import one level CA certificate (master CA)

Double click on generated CA file with .pfx extension will open Certificate Import Wizard. Select Store location -> Local Machine

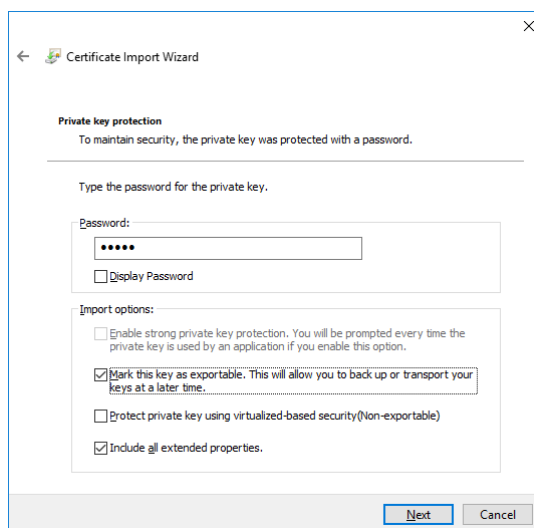


Next



Next

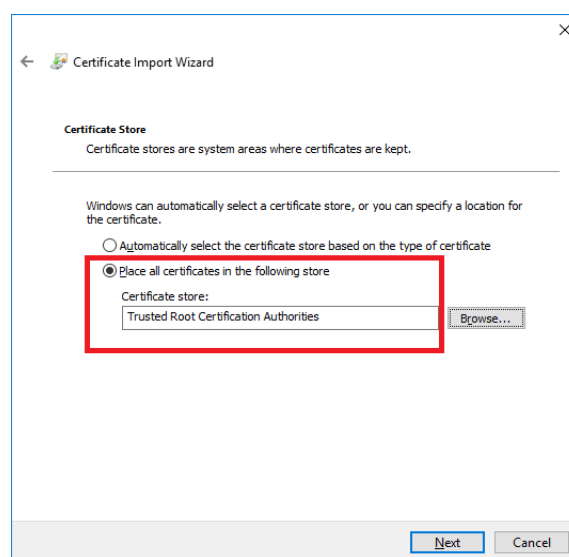
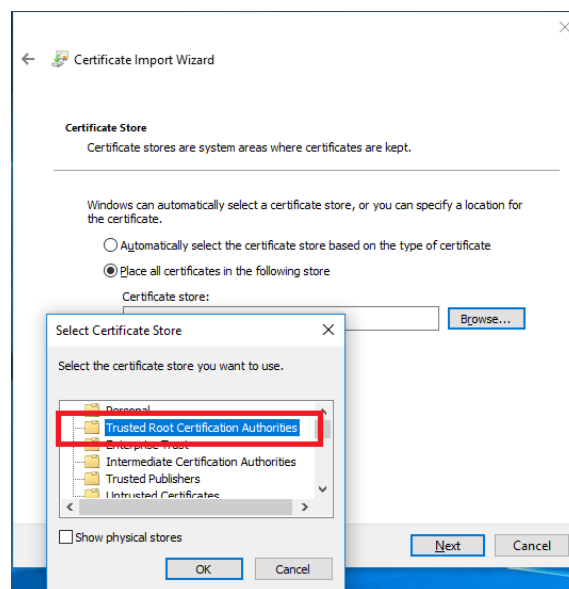
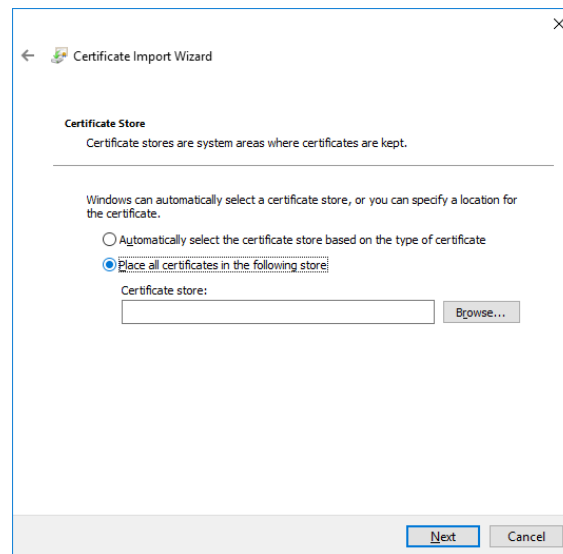
Enter password you set when generated .pfx file. Optionally you can allow that this certificate can be exportable from certificate store by checking option “Mark this key as exportable.”



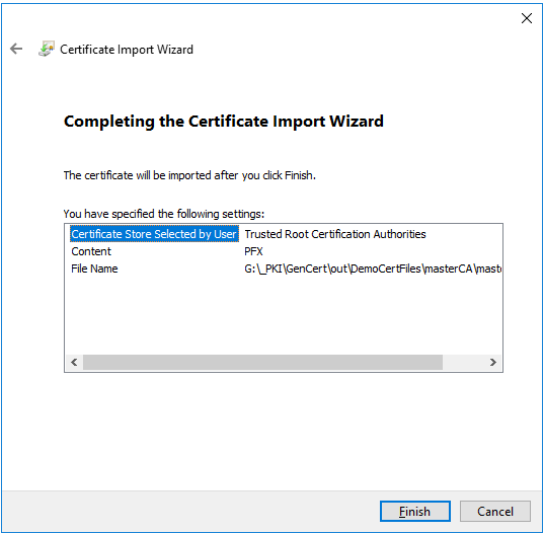
Next

NOTE:

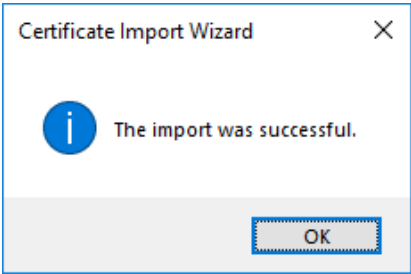
In next step be careful, use option “Place all certificates in the following store” and use certificate store “Trusted Root Certification Authorities”



Next

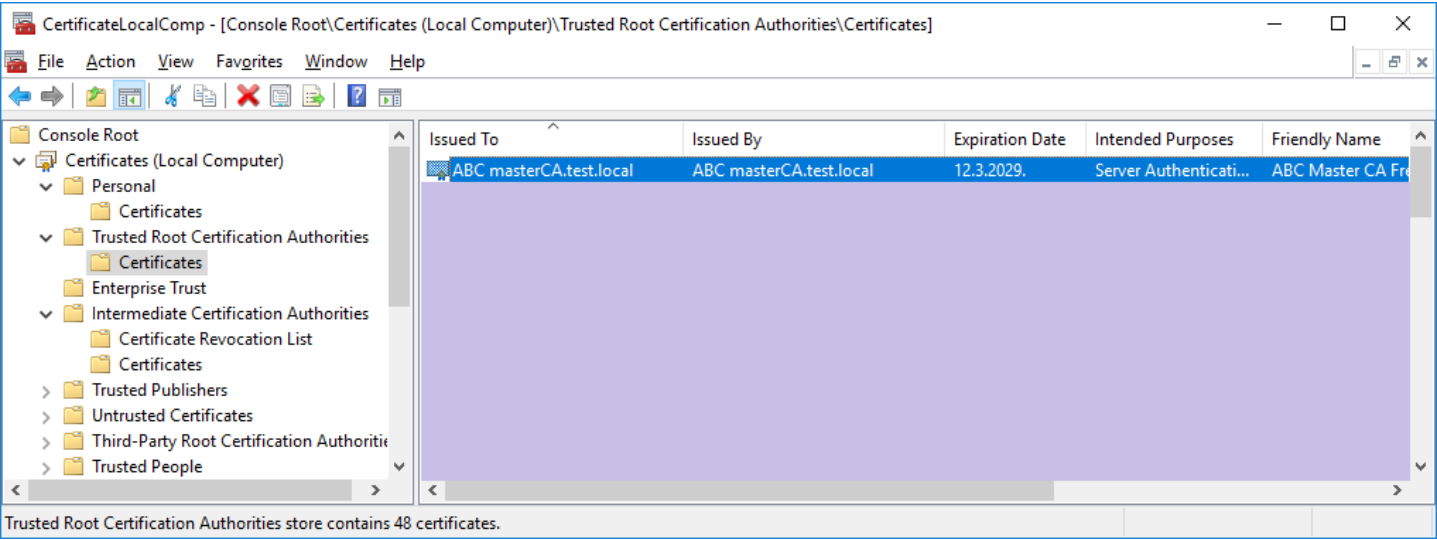


Finish



OK

If you import one level certificate (master CA) you need to refresh master CA certificate will be found inside Trusted Root Certification Authorities->Certificates store.



9. APPLICATION REQUIREMENTS

To successfully start application, you need .NET Framework 4.x installed on computer where application need to be started.

For minimal configuration, you ONLY need a file GenCert.exe to run application.


Inside Help folder you can find generated application user manual in different file formats: pdf, chm, xps, html.

For complete configuration, you need following files:

GenCert.exe

GenCert.chm

config.ini

File GenCert.chm extension is help file for application. If you put GenCert.chm file inside the same folder where you start application, this help will be open when you click on Home button 

When you start application for the first time 3 new files will be created:

Log4NetApplicationLog.log – log4net application log file in txt format

Log4NetApplicationLog.xml – log4net application log file in xml format

config.ini – application configuration file

Inside config.ini file, you can configure default value for option Certificate Friendly Name which is used inside application on different menu options.