Skapa certifikat till ARX

ASSA ABLOY
Opening Solutions

Instruktion för att skapa och signera certifikat för säker webbanslutning till ARX

UG905342A

The global leader in
door opening solutions

Copyright © 2018-2019 ASSA ABLOY Opening Solutions Sweden AB

Teknisk dokumentation och support

Vi förbehåller oss rätten att korrigera eventuella tryckfel och uppdatera informationen efter utskrift. På hemsidan finns utförliga manualer tillgängliga för ARX passersystem.

Om du inte hittar svar på dina frågor i manualen hänvisar vi till ASSAs tekniska support, de nås på +46 (0)8 775 16 60 alternativt: technical.arx@assaabloy.com

ASSA ABLOY Opening Solutions Sweden AB
Förmansvägen 11
SE-117 43 Stockholm
Sweden

Phone: +46 (0)8-775 16 00
Fax: +46 (0)8-775 16 20

Innehållsförteckning

Generera och signera ett certifikat för ARX

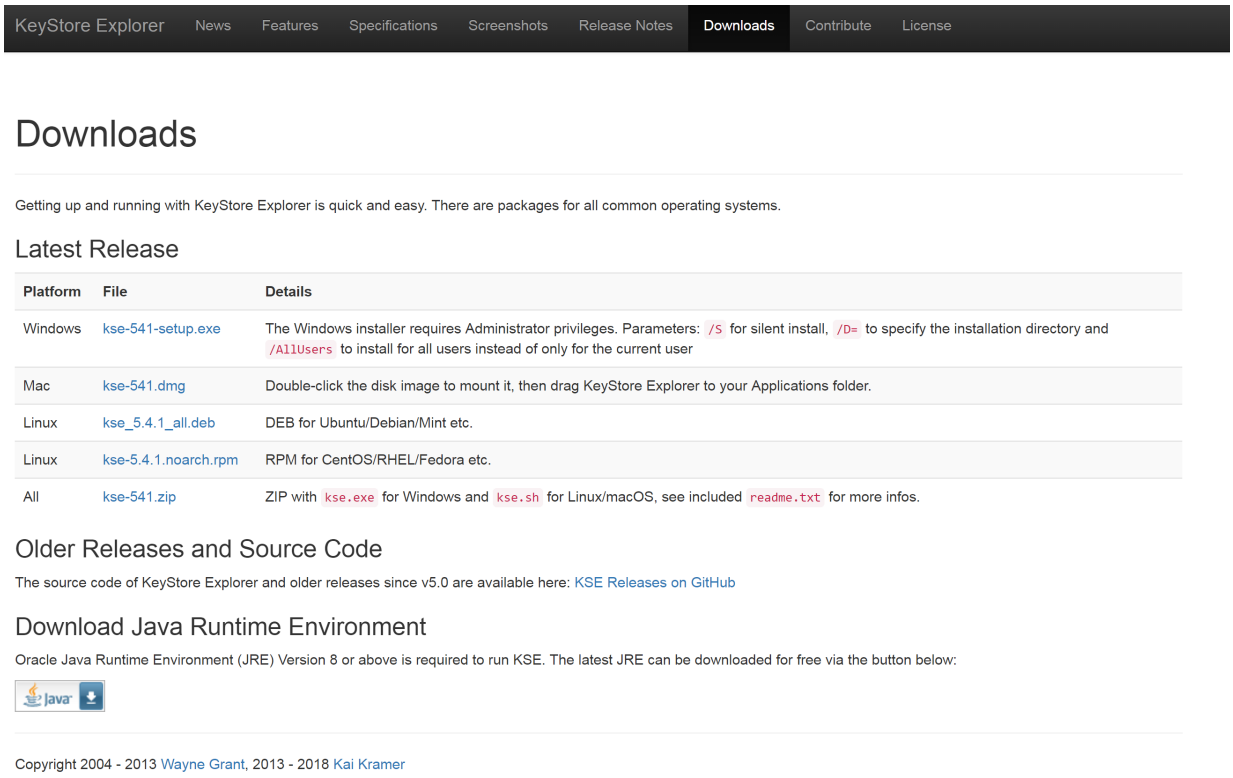
I det här kapitlet förklarar vi hur man skapar, signerar och lägger in ett betrott certifikat i ARX som gör att man kan komma åt olika funktioner i ARX via en webbläsare eller en integration på ett säkert sätt, utan att få certifikatvarningar (SSL / https).

VIKTIGT

Den här guiden kommer att innehålla grafiska stegvisa instruktioner om hur du kommer igång, men du som användare förväntas använda programmet KeyStoreExplorer.

Du kan också använda keytoolen som ingår i Java JDK, men vi rekommenderar endast mycket avancerade användare att göra det manuellt. Vi kan inte garantera att du bara kan följa denna guide blint när du skriver kommandon i din terminal.

För att ladda ner programmet, gå till: https://keystore-explorer.org/downloads.html och installera den rätta releasen för din OS-distribution



KeyStore Explorer | News | Features | Specifications | Screenshots | Release Notes | Downloads | Contribute | License

Downloads

Getting up and running with KeyStore Explorer is quick and easy. There are packages for all common operating systems.

Latest Release

Platform	File	Details
Windows	kse-541-setup.exe	The Windows installer requires Administrator privileges. Parameters: `/S` for silent install, `/D=` to specify the installation directory and `/AllUsers` to install for all users instead of only for the current user
Mac	kse-541.dmg	Double-click the disk image to mount it, then drag KeyStore Explorer to your Applications folder.
Linux	kse_5.4.1_all.deb	DEB for Ubuntu/Debian/Mint etc.
Linux	kse-5.4.1.noarch.rpm	RPM for CentOS/RHEL/Fedora etc.
All	kse-541.zip	ZIP with `kse.exe` for Windows and `kse.sh` for Linux/macOS, see included `readme.txt` for more infos.

Older Releases and Source Code

The source code of KeyStore Explorer and older releases since v5.0 are available here: KSE Releases on GitHub

Download Java Runtime Environment

Oracle Java Runtime Environment (JRE) Version 8 or above is required to run KSE. The latest JRE can be downloaded for free via the button below:

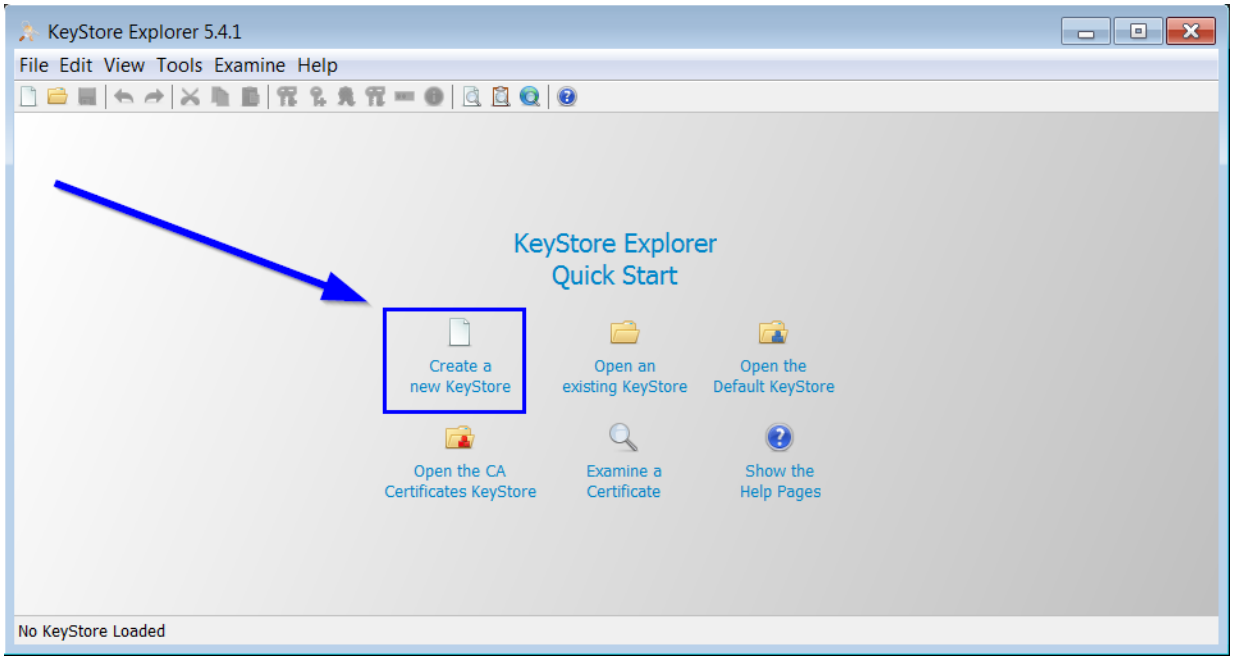
Java

Copyright 2004 - 2013 Wayne Grant, 2013 - 2018 Kai Kramer

Generering av Java Keystore och ett självsignerat certifikat

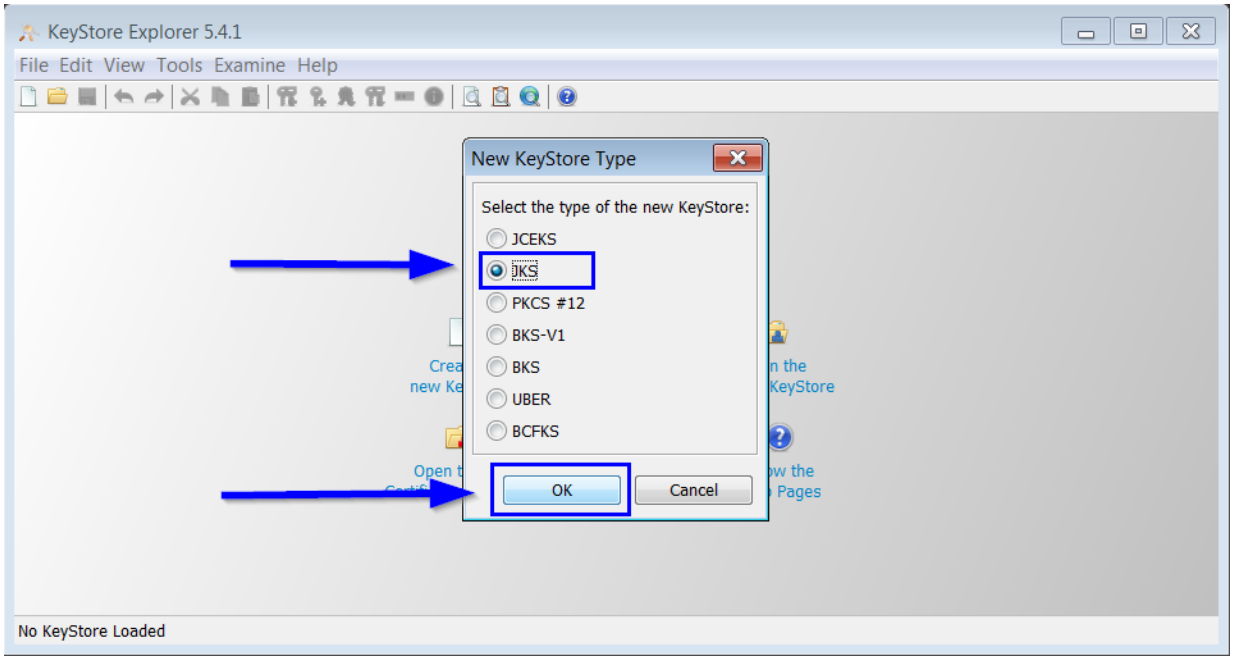
När Keystore Explorer installeras och startas behöver du skapa en keystore.

Steg 1



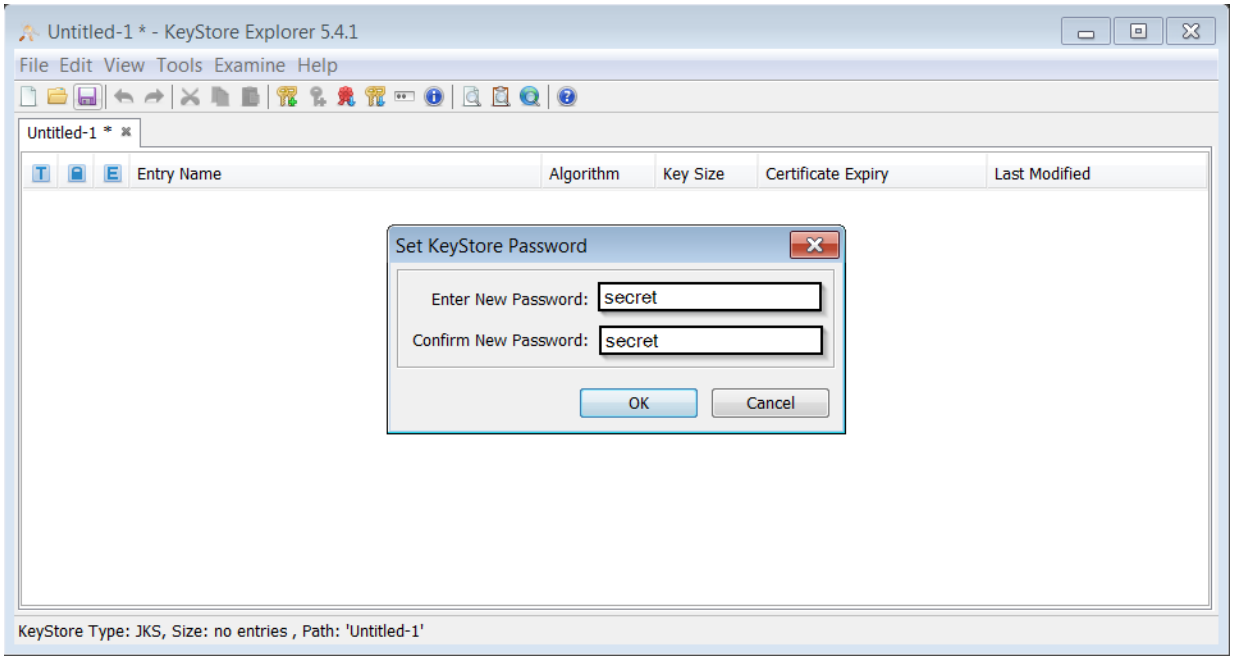
Klicka på den markerade knappen för att börja skapa en ny Keystore

Steg 2



*Välj JKS Keystore-typen och tryck på **OK***

Steg 3

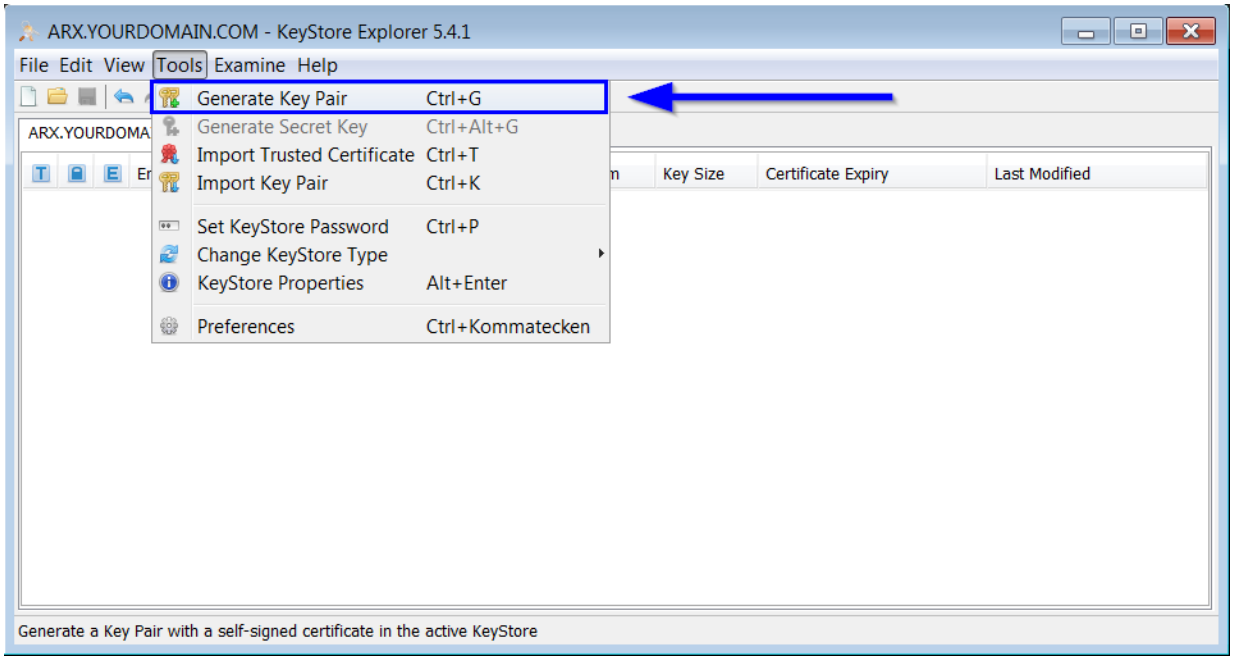


Spara JKS-filen du just skapat, lösenordet måste sättas till ***secret***

VIKTIGT
Vi rekommenderar att du namnger filen till det domännamn systemet tillhör och med tillägget .jks

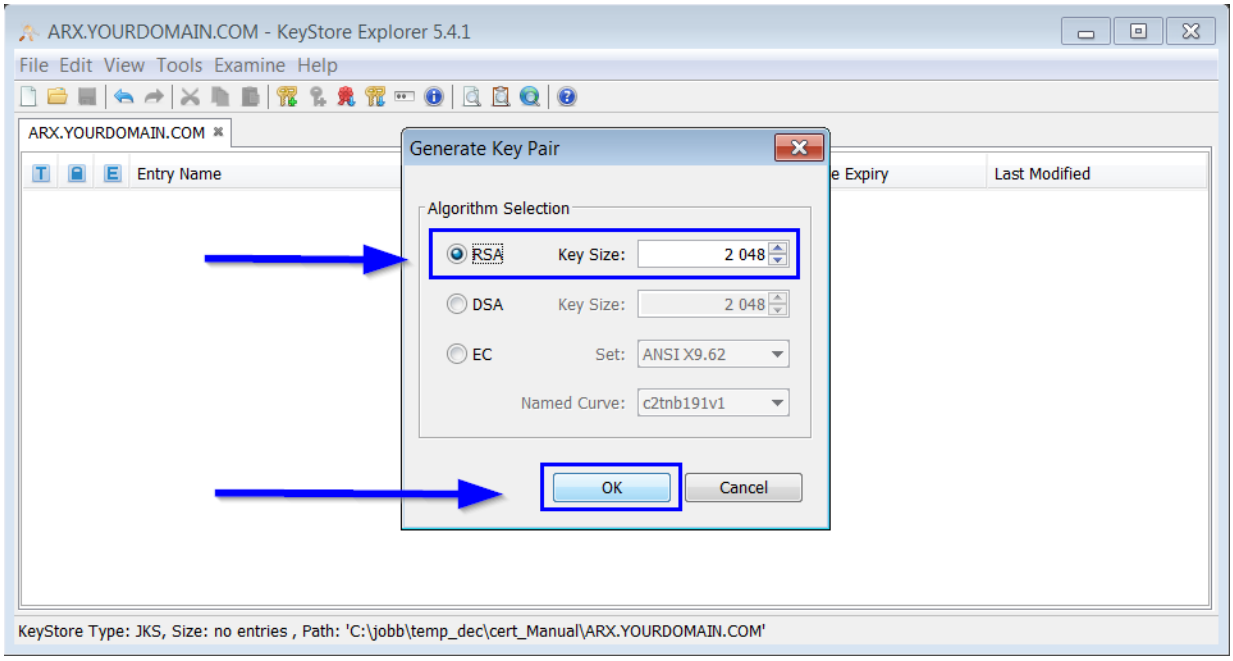
Till exempel, i vårt fall lagrar vi filen som **ARX.YOURDOMAIN.COM.jks**

Steg 4



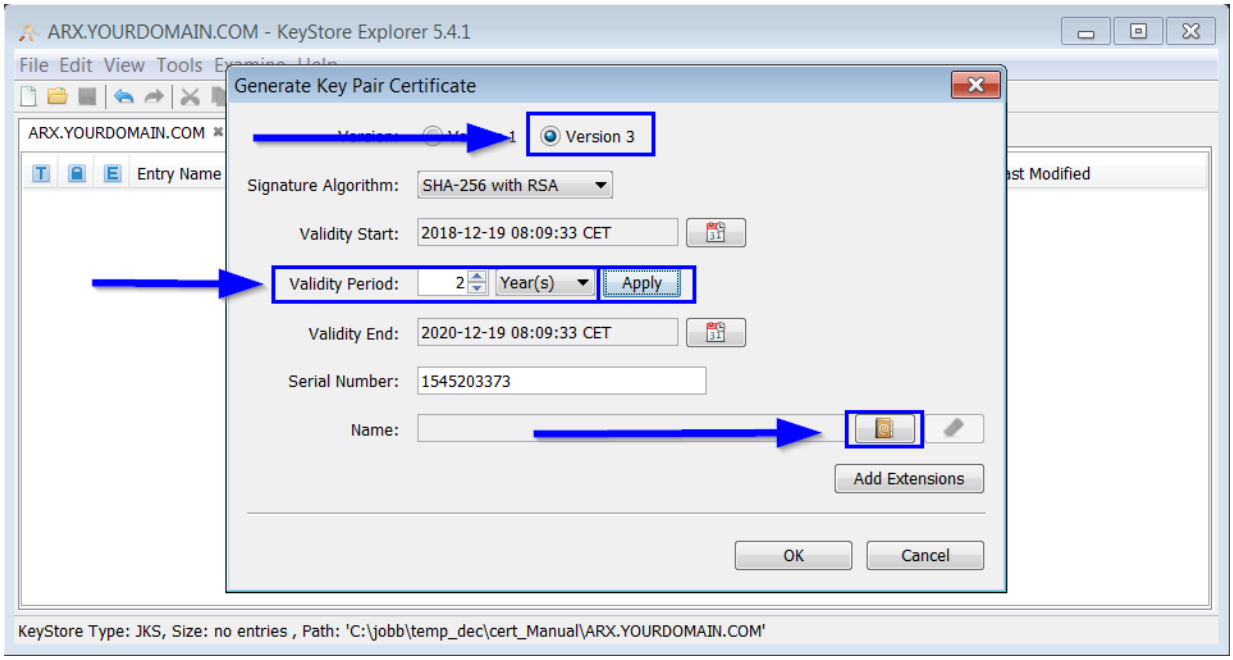
Nu när du har sparat JKS-filen, gå till Tools -> Generate Key Pair

Steg 5



*Alrogitmen måste vara **RSA** och nyckelstorlek ska vara **2048**, när detta är inställt trycker du på **OK***

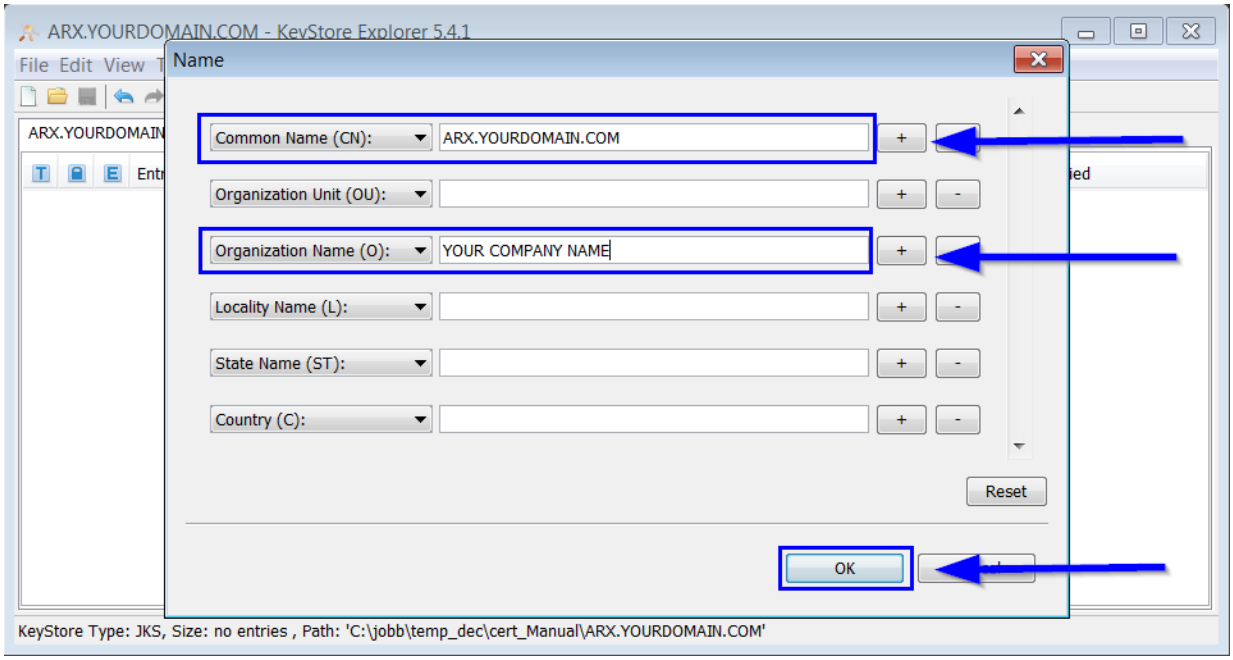
Steg 6



*Se till att du använder **Version 3** och ange giltighetsperioden för certifikatet till minst **ett år**. Kom ihåg att trycka på **Apply** för att spara din valideringsperiod*

När instruktionerna ovan är gjorda trycker du på den markerade knappen (ovanför knappen "Add extensions").

Steg 7



VIKTIGT

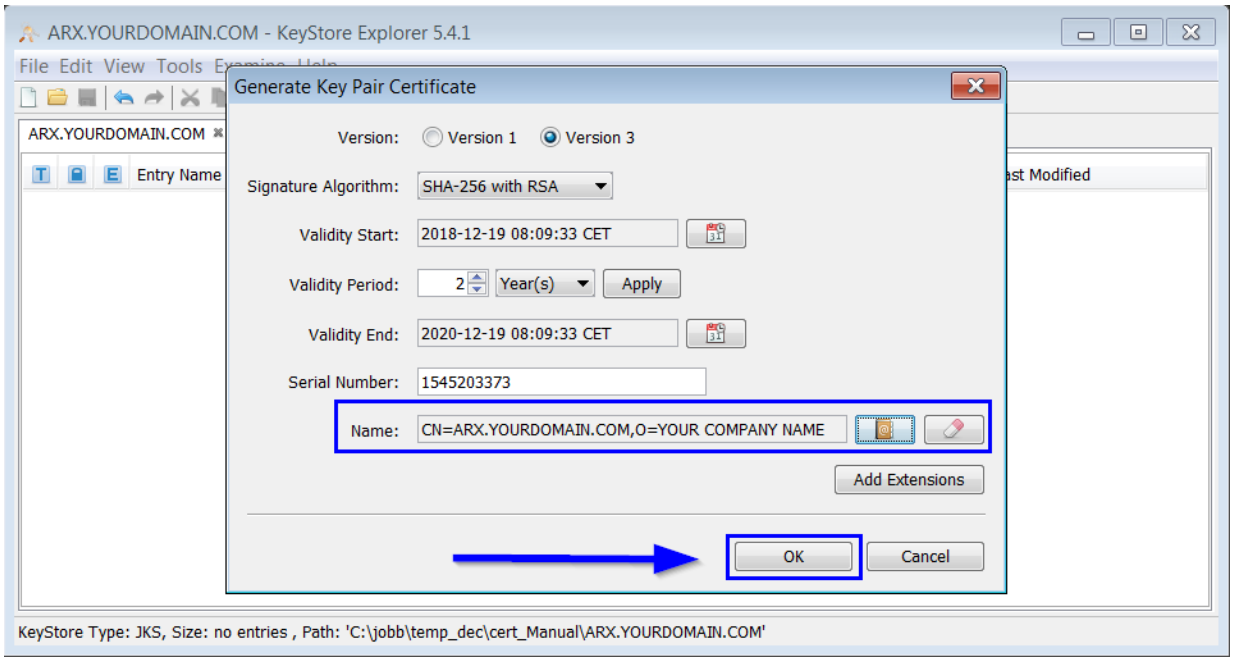
Om du ska använda ett wildcard-SSL-certifikat måste du använda wildcard-referensen i din URL.

*I denna bild hänvisar vi till adressen **ARX.YOURDOMAIN.COM**, men om du använder wildcard SSL skulle du behöva skriva ***.YOURDOMAIN.COM***

***CN**: krävs. Här anger du din adress (om ARX kommer att vara värd på din adress **arx.yourdomain.com** då är det detta du skriver).*

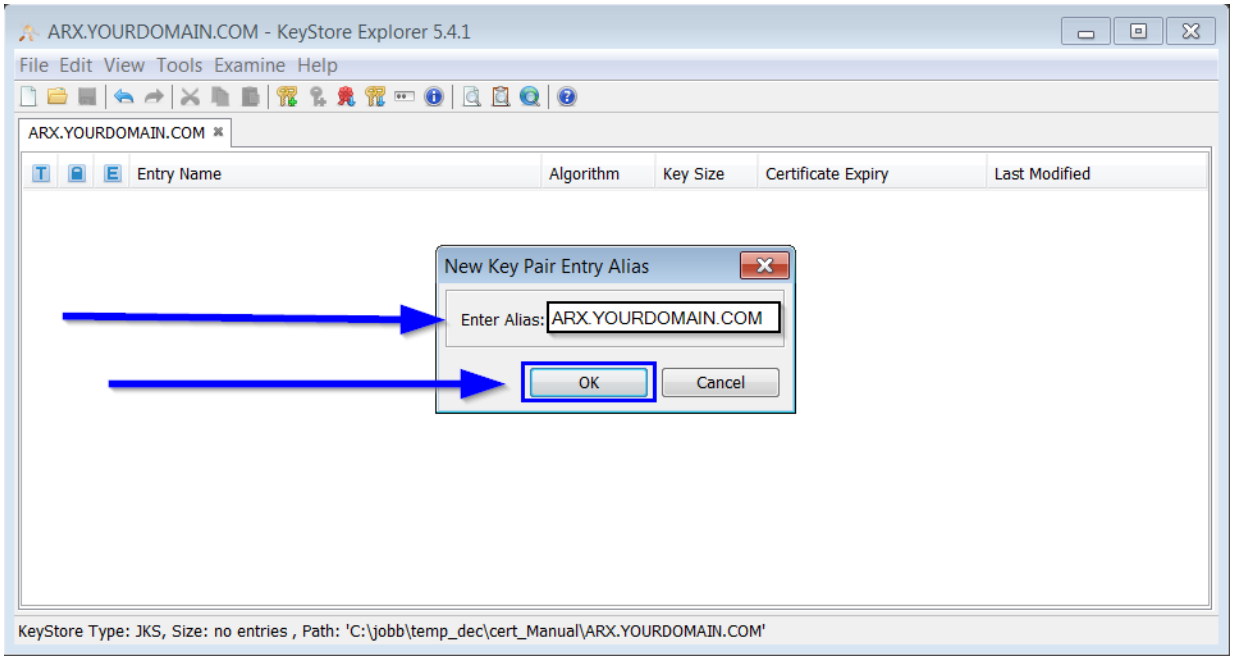
*Alla andra fält är valfria, men vi rekommenderar att du anger ett organisationsnamn (**Organization Name**).*

Steg 8



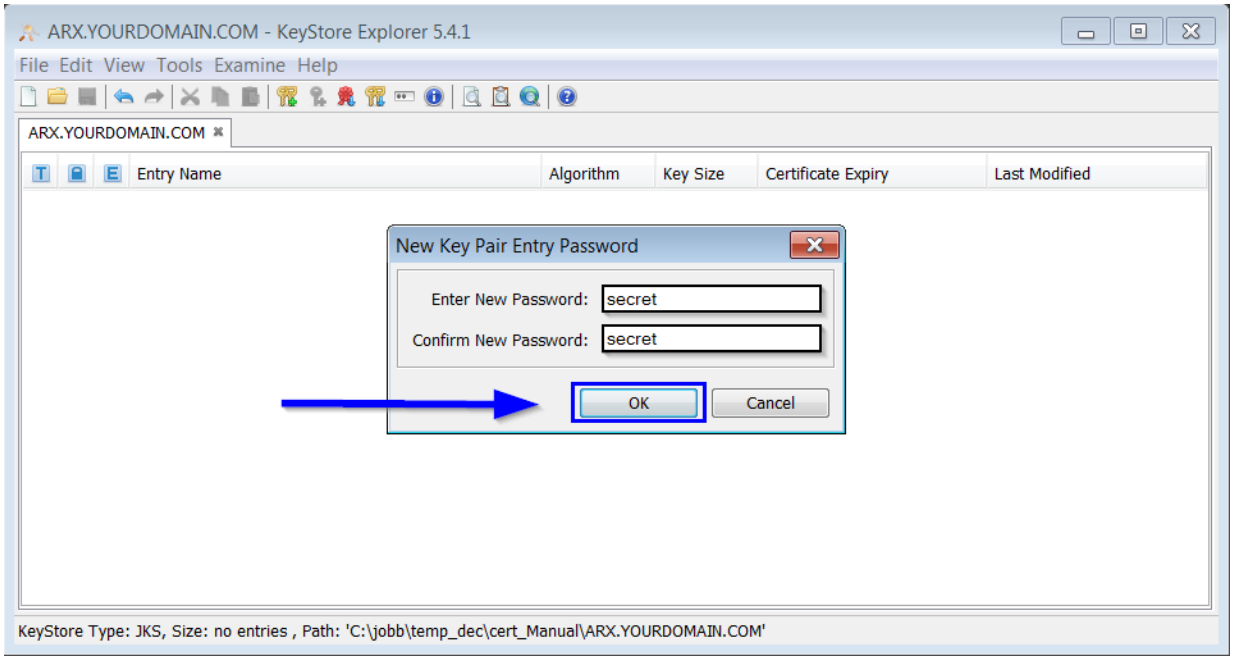
*Se till att du kan se **CN=** innan du trycker på **OK***

Steg 9



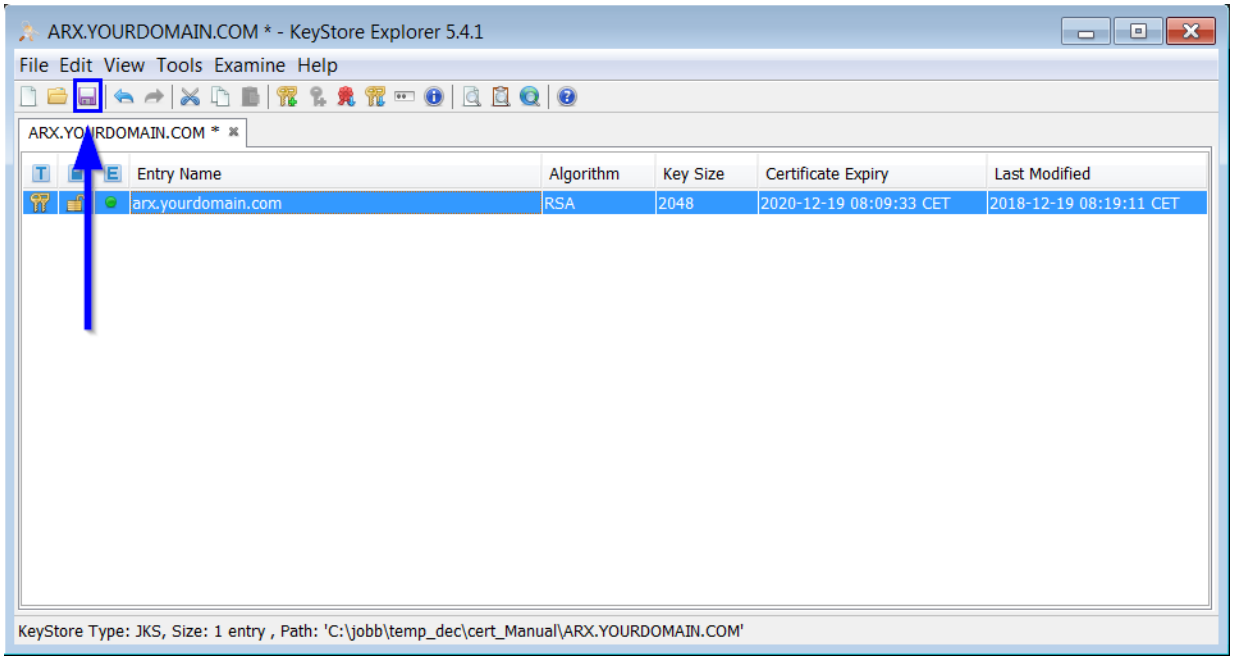
*Du kan spara ditt certifikat som vad som helst, men för att göra det enklare att navigera i rätt certifikat rekommenderar vi att du namnger det som den webbadress du angav tidigare och tryck **OK**.*

Steg 10



*I detta steg krävs det att du anger lösenordet: **secret**. Avsluta med **OK**.*

Steg 11 (klart)

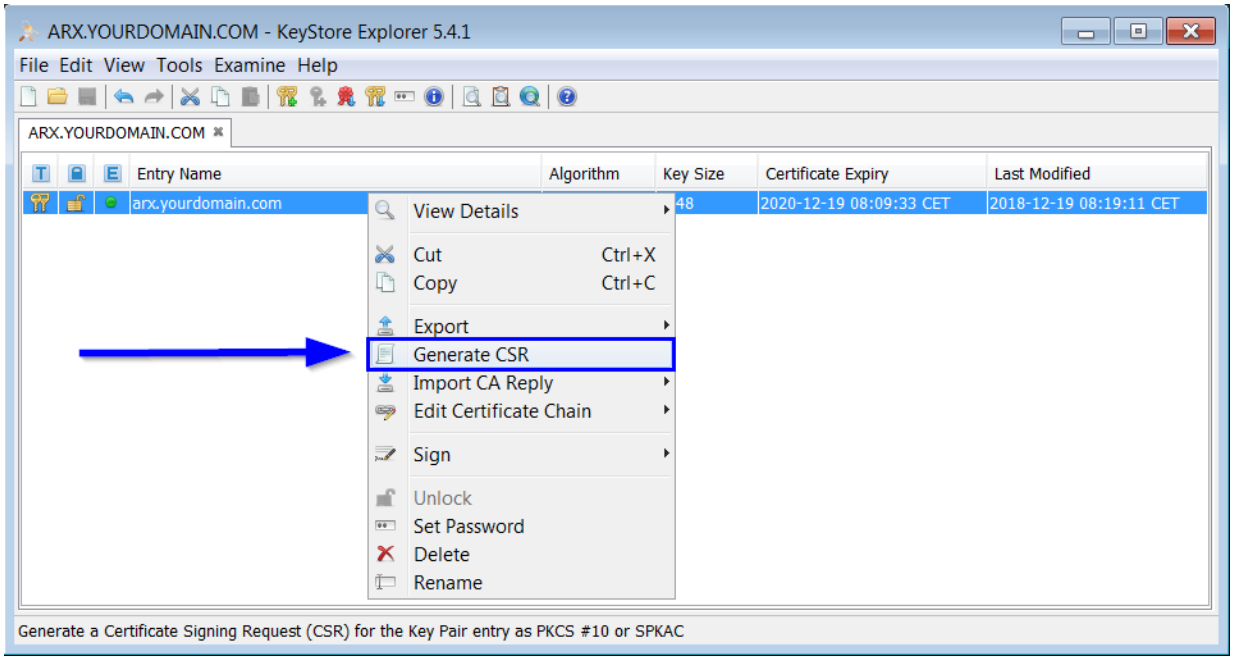


Spara dina ändringar och nu är ditt självsignerade certifikat skapat!

Du kan enkelt dubbelklicka på certifikatet för att få mer information om det. Nästa steg innan du kan använda certifikatet är att skapa en CSR (Certificate Signing Request)

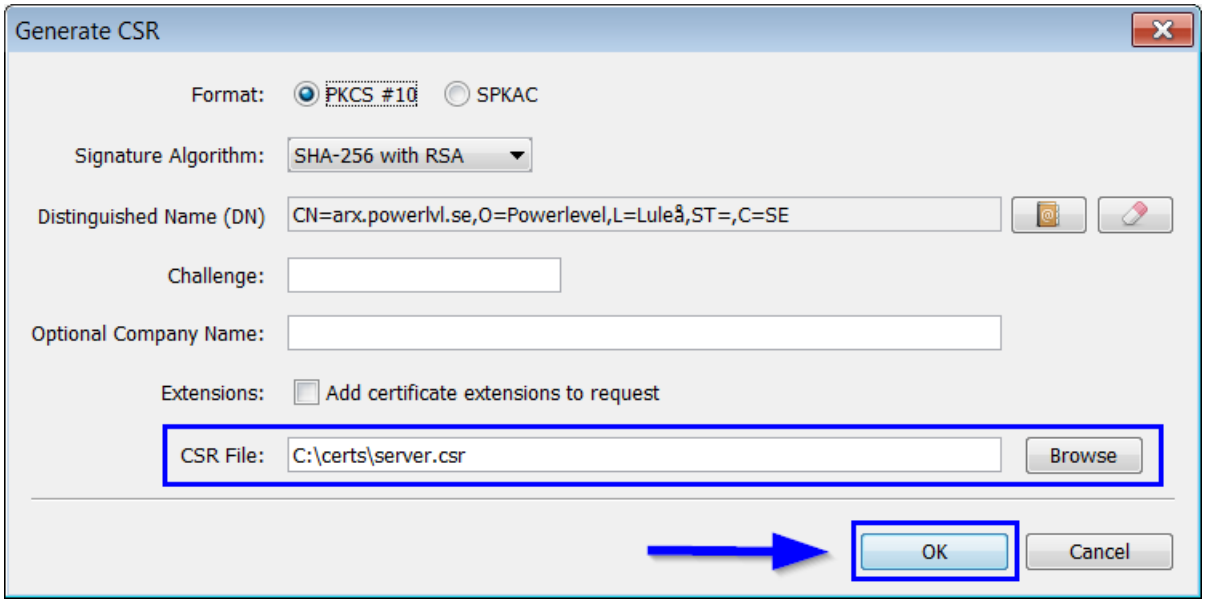
Generera en CSR (certifikat-signeringsförfrågan från certifikat)

Steg 1



*Högerklicka på ditt certifikat och väl **Generate CSR***

Steg 2



*Välj lämplig plats för att spara filen och tryck **OK***

VIKTIGT

Nu när du har skapat din CSR måste du ladda upp den till en SSL-återförsäljare för att göra den signerad och betrodd.

I denna guide har vi två olika exempel A och B. **A** är import från p7b-fil och **B** där du själv skapar kedjan.

Dessa exempel använder filerna som kommer ifrån **Digicert** eller **Comodo** respektive

Importera det signerade SSL-certifikatet

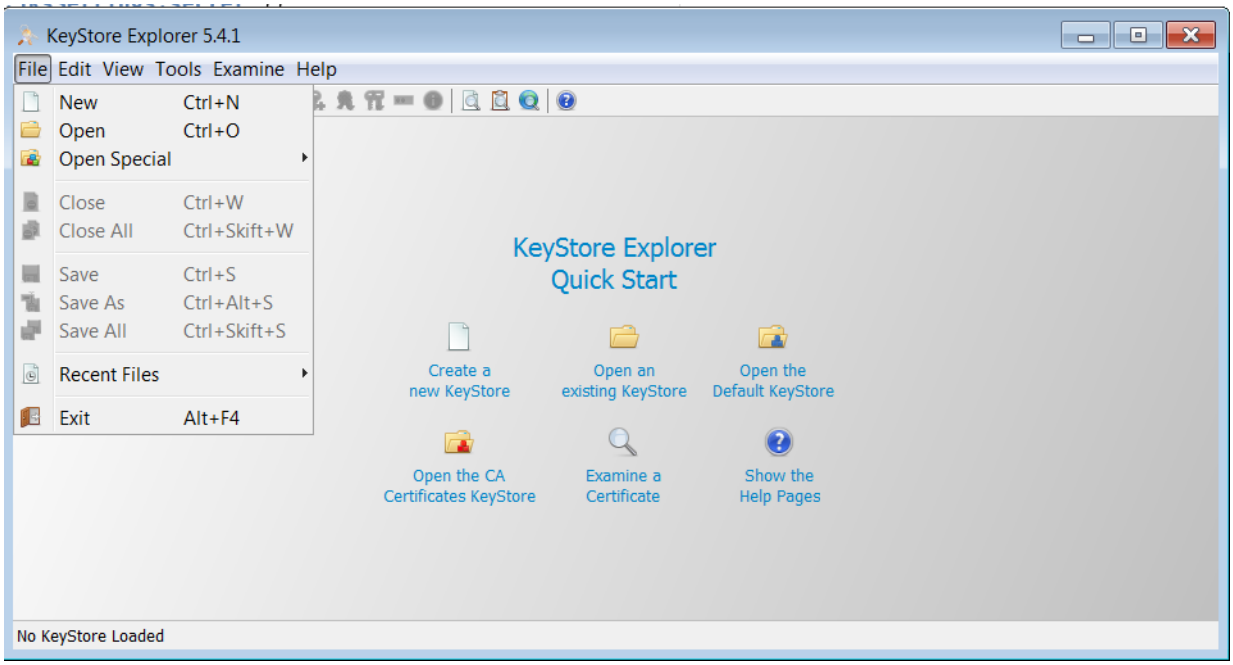
Vi kommer att förklara två olika sätt att importera SSL-certifikatet till din skapade Java Keystore (som genereras ovan)

> **VARNING**
> Kom ihåg att innan du kan fortsätta härifrån måste du ha köpt ett SSL-certifikat från en återförsäljare och fått det signerade certifikatet.

A. Importera SSL-certifikat (.p7b-fil)

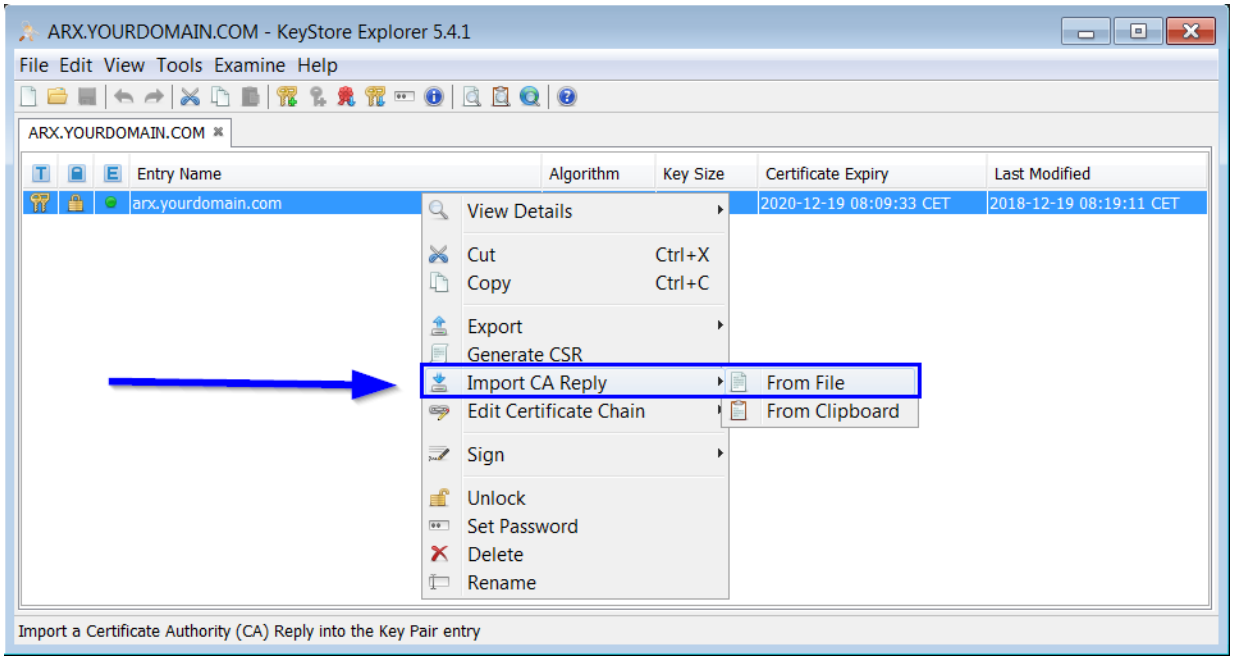
.p7b-filer innehåller både certifikat och certifikatkedja, vilket gör att de importeras till Keystore mycket lättare än en .crt-fil

Steg 1



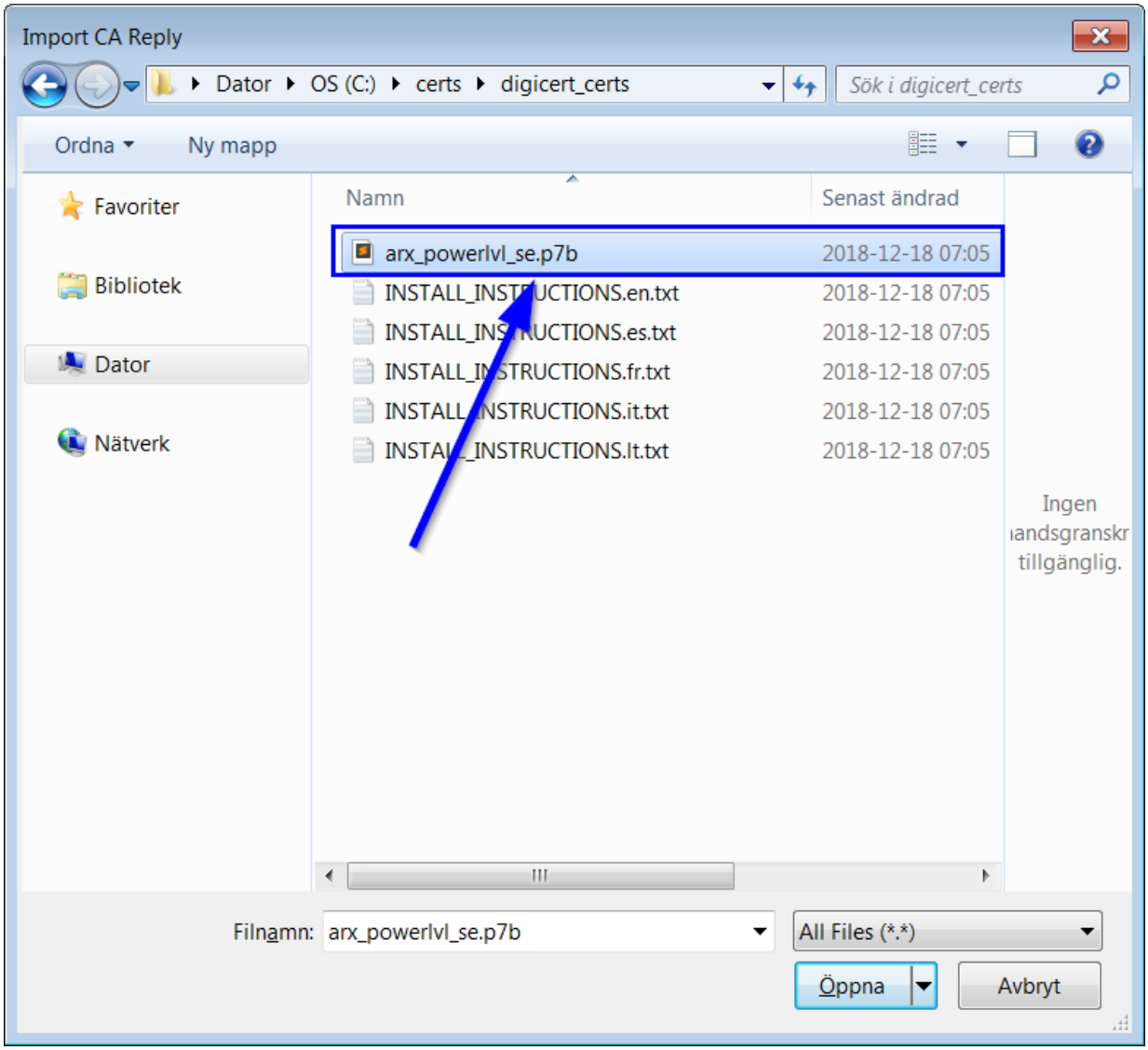
Starta programmet KeyStore Explorer och öppna din tidigare självsignerade certifikatfil (du skickade till SSL-återförsäljaren).

Steg 2



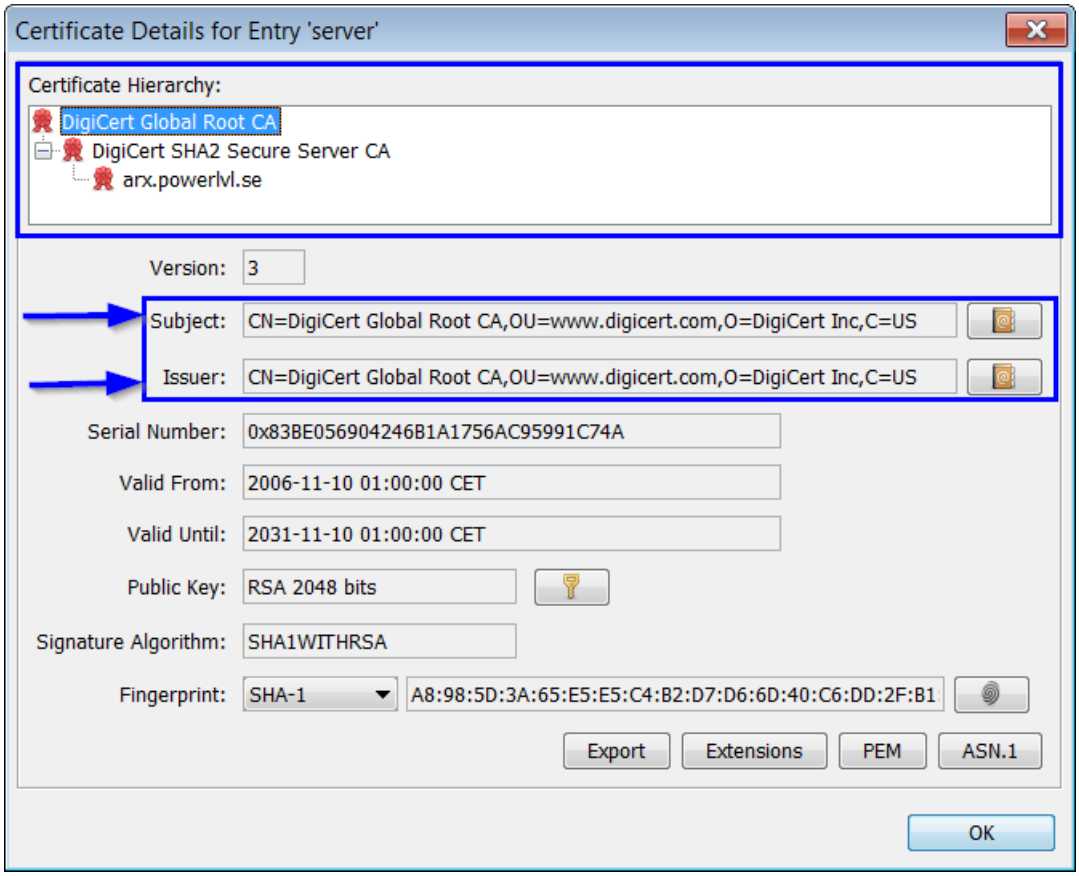
*Högerklicka på certifikatet och välj **Import CA Reply** och välj **From file***

Steg 3



*Välj **p7b-filen** du fick i retur från SSL-återförsäljaren.*

Steg 4 (klart)



Dubbelklicka på ditt certifikat och verifiera att certifikatkedjan ser ut som något liknande det här.

*Klicka på certifikatet som ligger längst upp i trädet (vilket har namnet **Root CA** eller liknande) och verifiera att den har samma **Subject** och **Issuer**, Det betyder att det inte finns några fler certifikat att lägga till i kedjan. (Detta är ett certifikat från en betrodd certifikatmyndighet CA).*

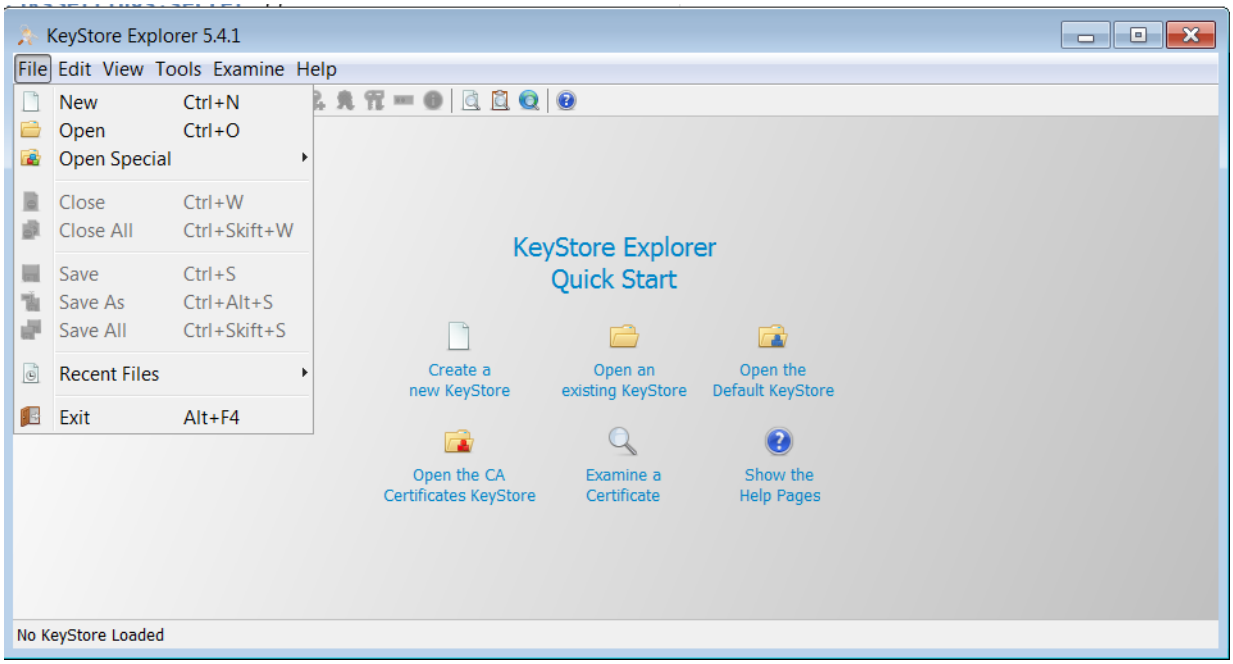
OBSERVERA
Kom ihåg att spara innan du fortsätter

VIKTIGT
Nu när du har gjort dessa steg kan du gå till det sista steget: Konfigurera ARX för att använda din nya JavaKeystore [25]

B. Importera SSL Certitfikat (.crt-filer)

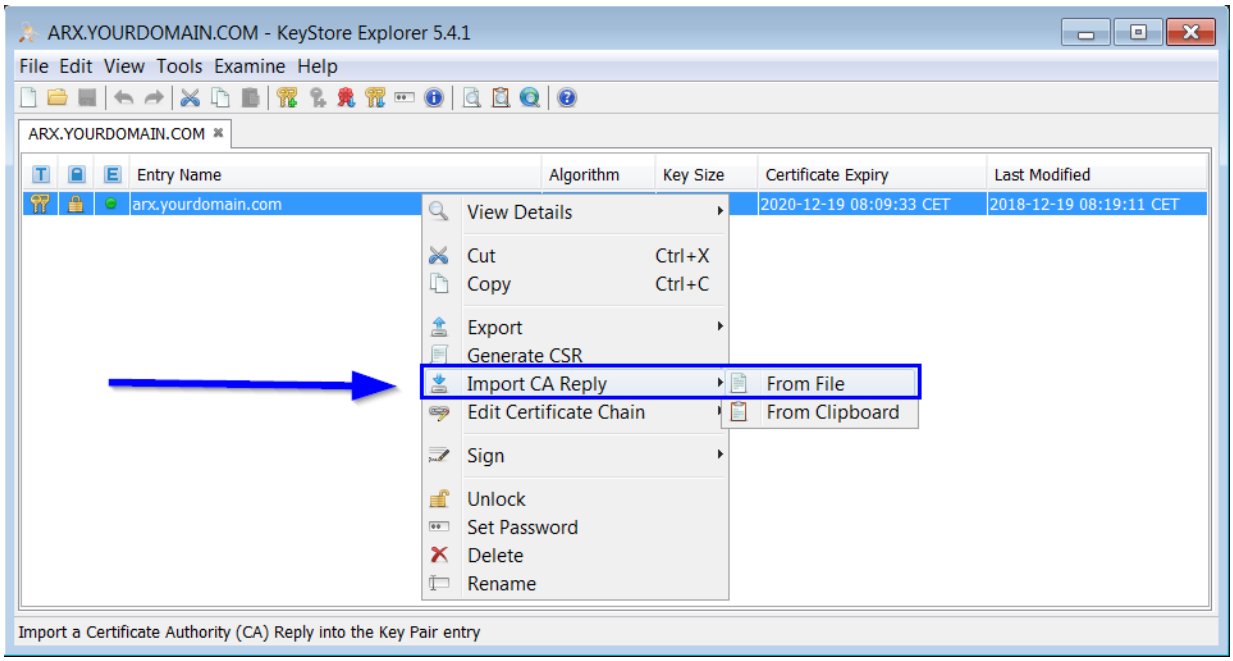
.crt-filer innehåller certifikatet från vilket du kan skapa certifikatkedjan

Steg 1



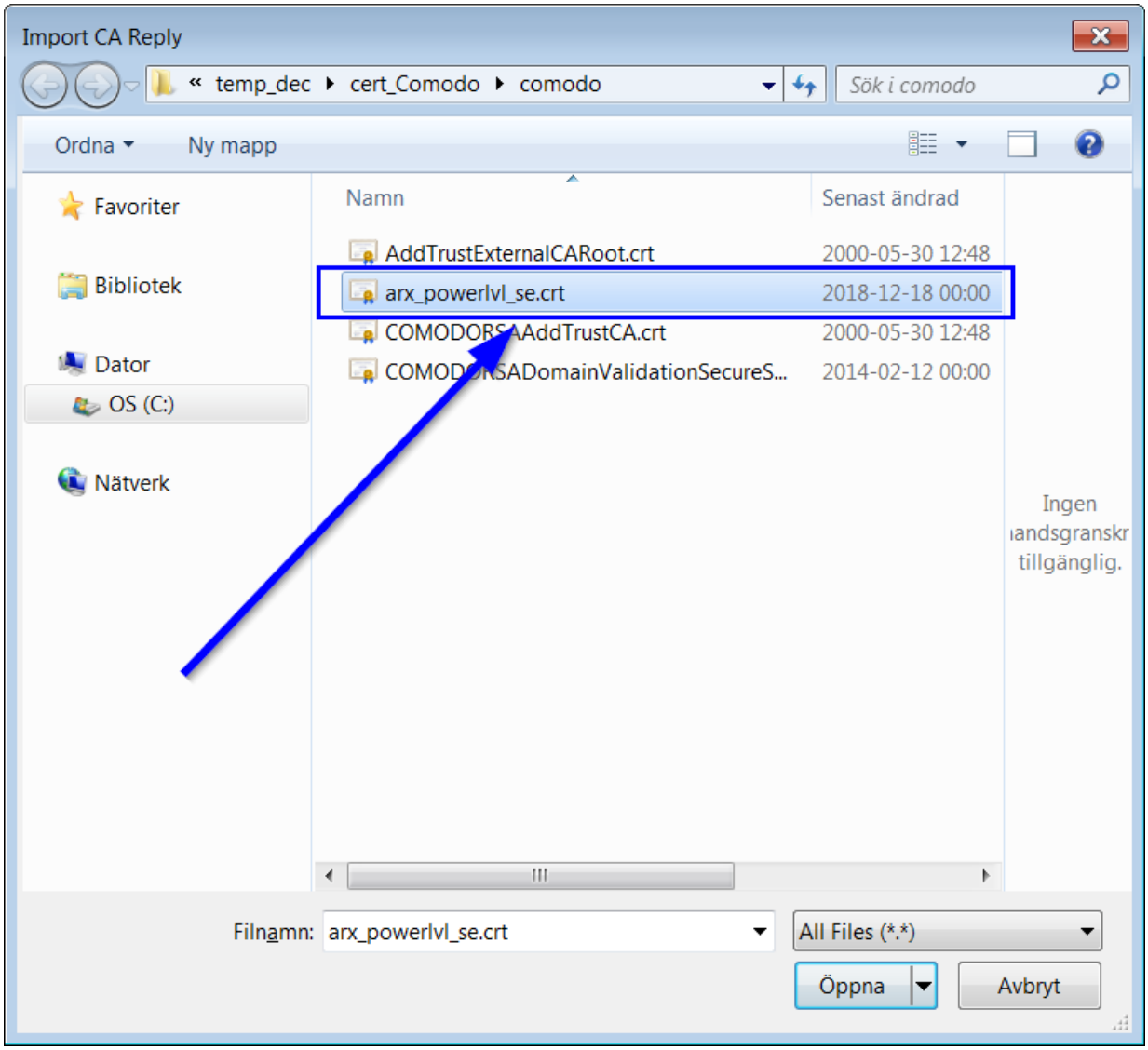
Starta programmet KeyStore Explorer och öppna din tidigare självsignerade certifikatfil (du skickade till SSL-återförsäljaren).

Steg 2



*Högerklicka på certifikatet och välj **Import CA Reply** och välj **From File***

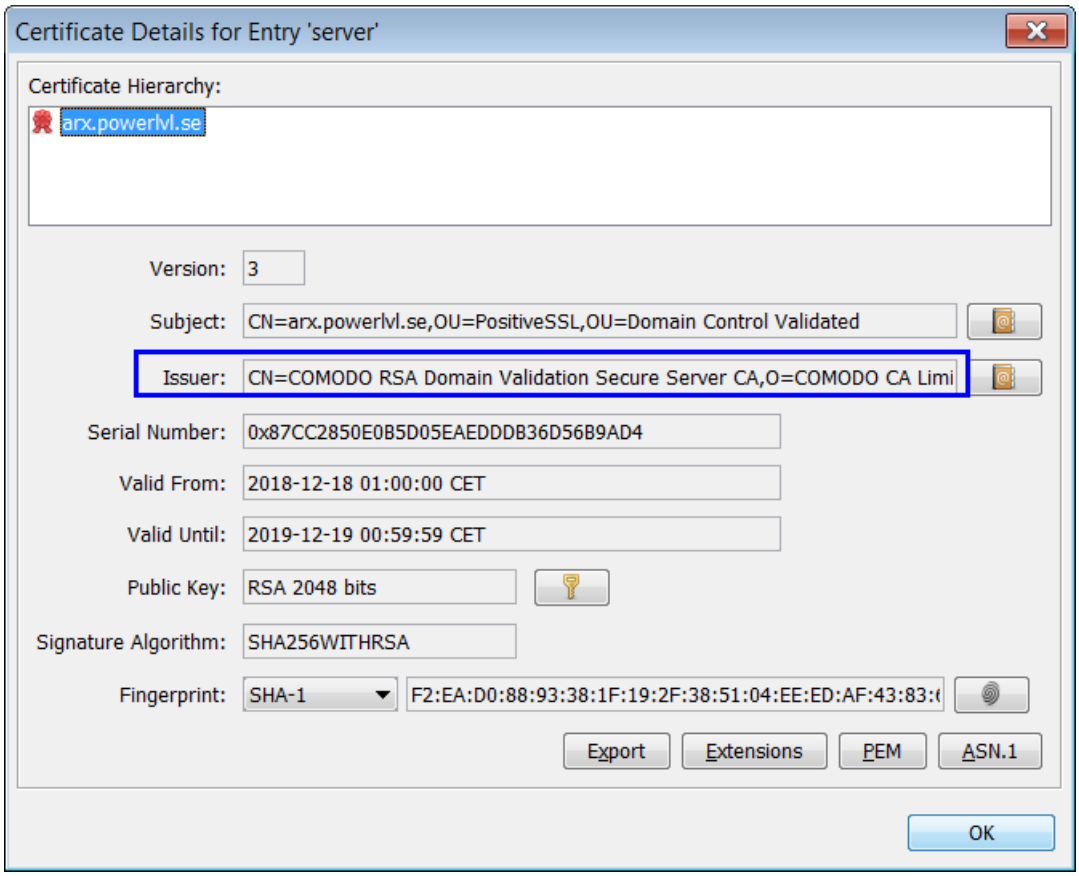
Steg 3



Välj ***crt*** *filen du fick i retur från SSL-återförsäljaren*

I det här exemplet har vår fil namnet ***arx_powerlvl_se.crt***

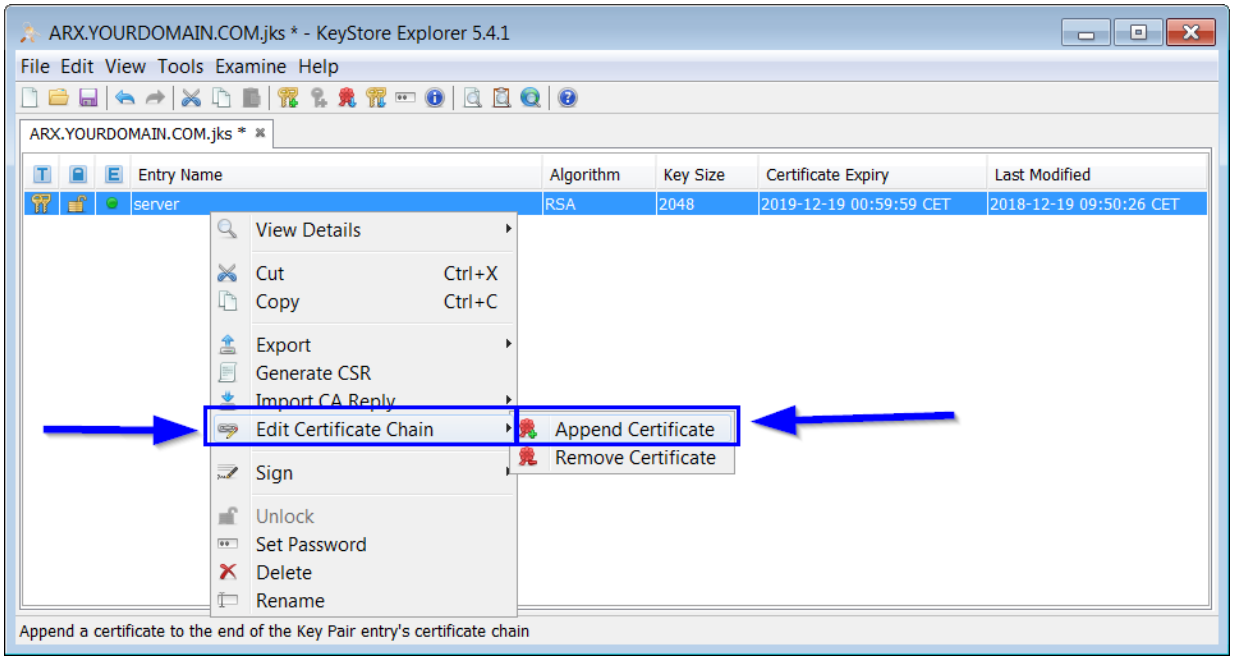
Steg 4



Dubbelklicka på ditt certifikat och kontrollera att utgivaren har uppdaterats.

> ***VIKTIGT***
> *Observera att utgivaren i detta fall är **COMODO RSA Domain Validation Secure Server.** Det här är relevant för oss när du väljer nästa certifikat (i nästa steg)*

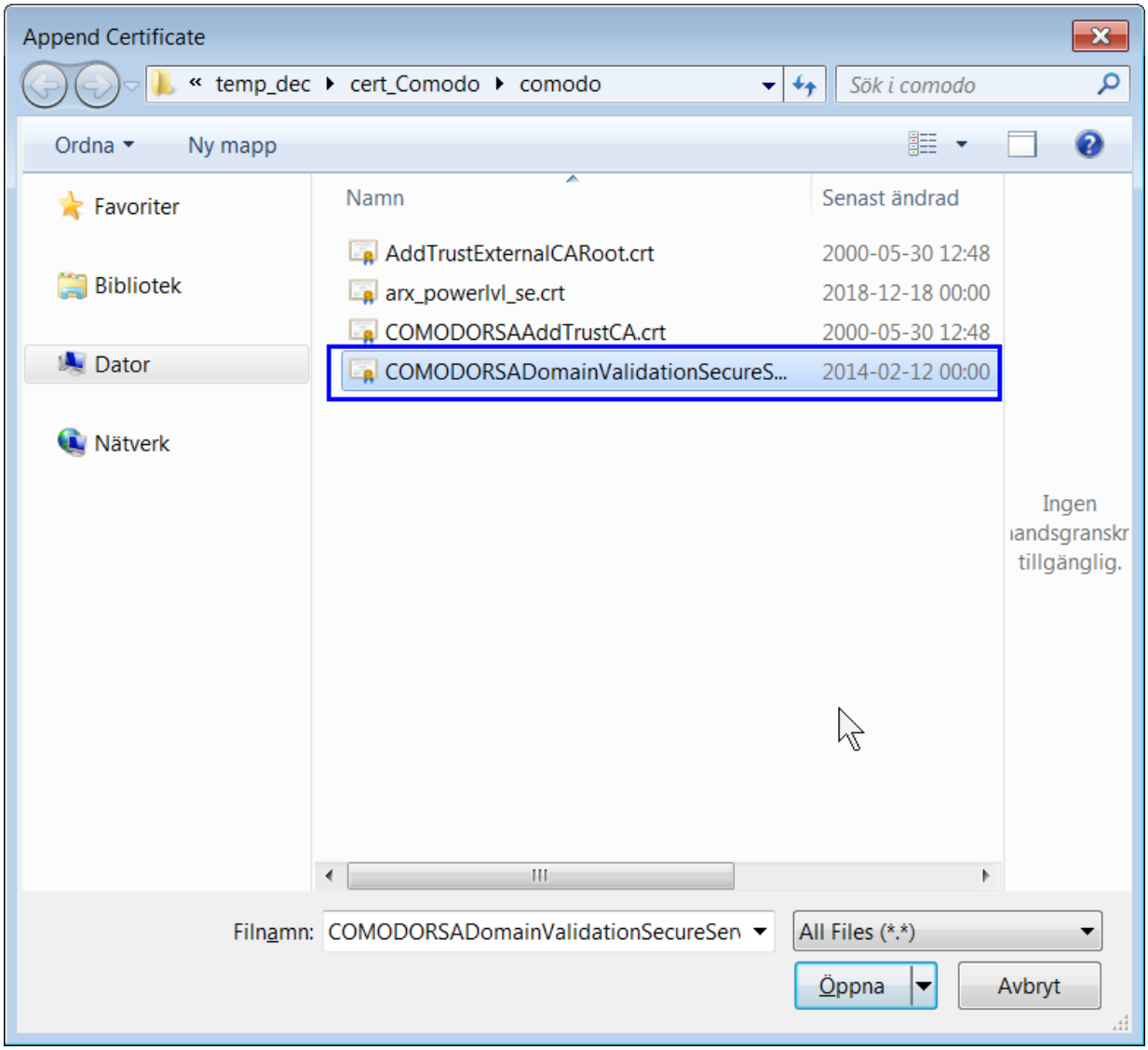
Steg 5



*Högerklicka på ditt certifikat igen, men välj i det här fallet **Edit Certificate Chain** och välj **Append Certificate***

Här kommer vi att börja bygga förtroendekedjan, du kommer att se vad det betyder i de senare stegen.

Steg 6

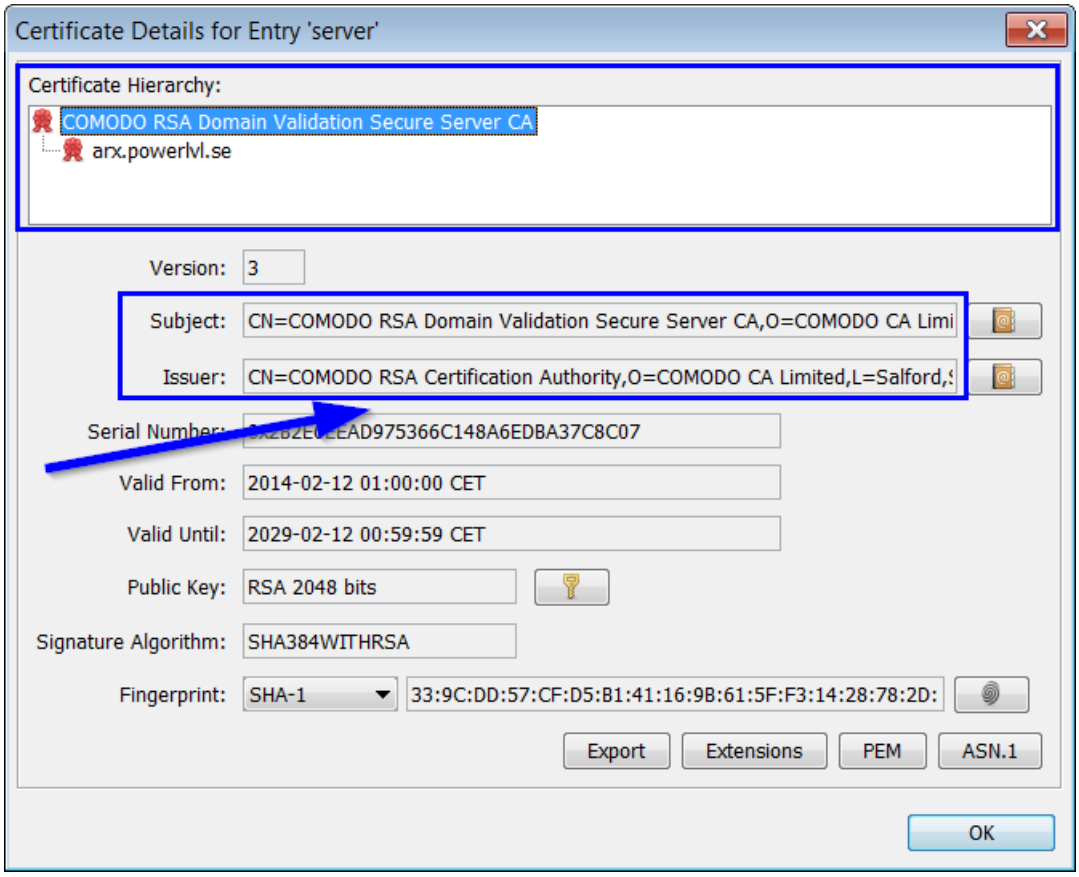


*Nu väljer vi certifikatet som har ett liknande namn som den som utfärdade vårt certifikat (se **steg 4**)*

VIKTIGT
*Till exempel: Vår utgivare som vi hittat i steg 4 har namnet **COMODO RSA Domain Validation Secure Server***

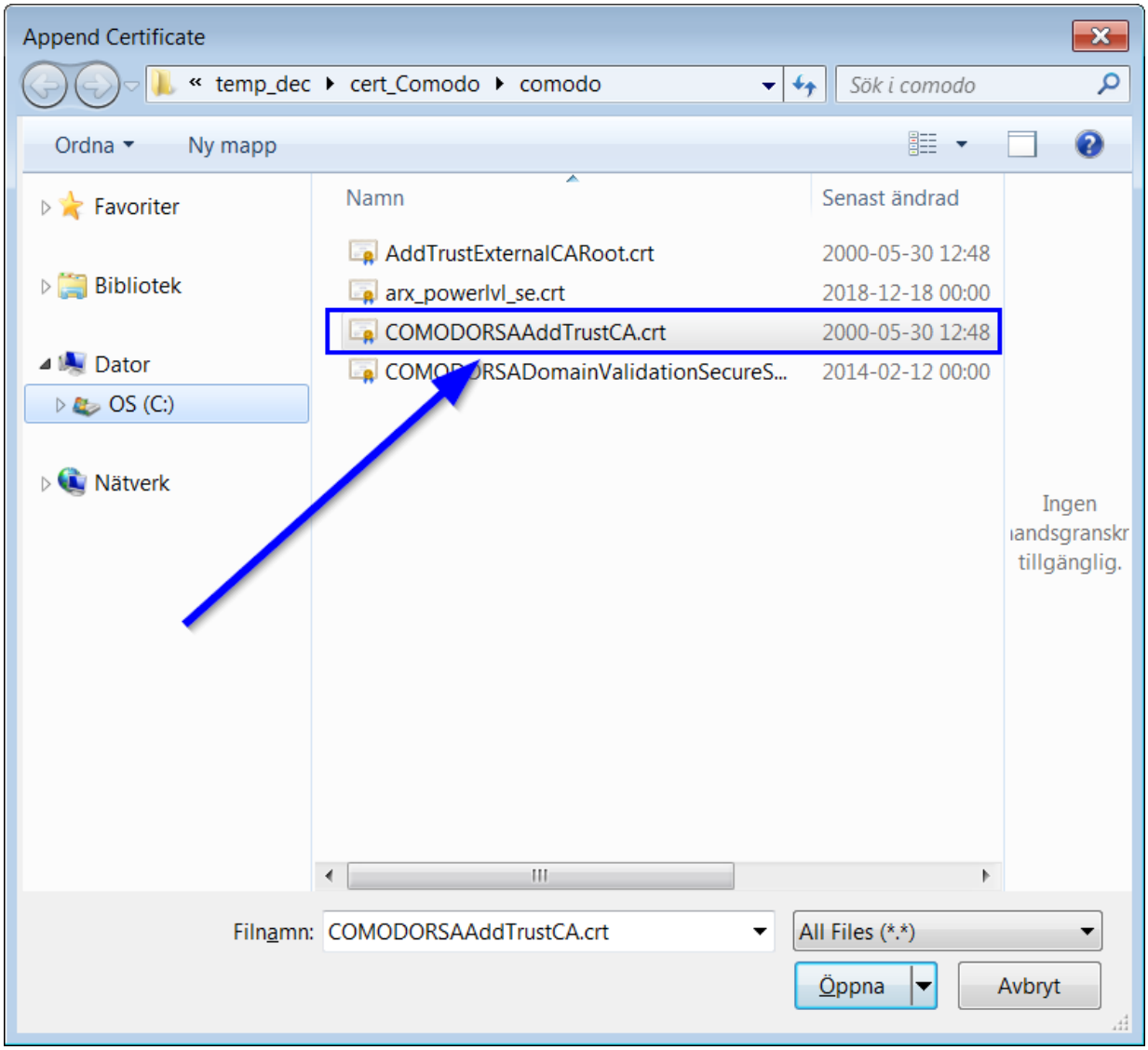
*Så filen vi måste välja i detta fall är **COMODORSADomainValidationSecureServerCA.crt***

Steg 7



*Dubbelklicka på ditt certifikat igen, nu ser du att det har fått en trädstruktur (i toppen). Klicka på det nya certifikatet och notera namnet på **utfärdaren (Issuer)** eftersom detta är relevant i nästa steg*

Steg 8

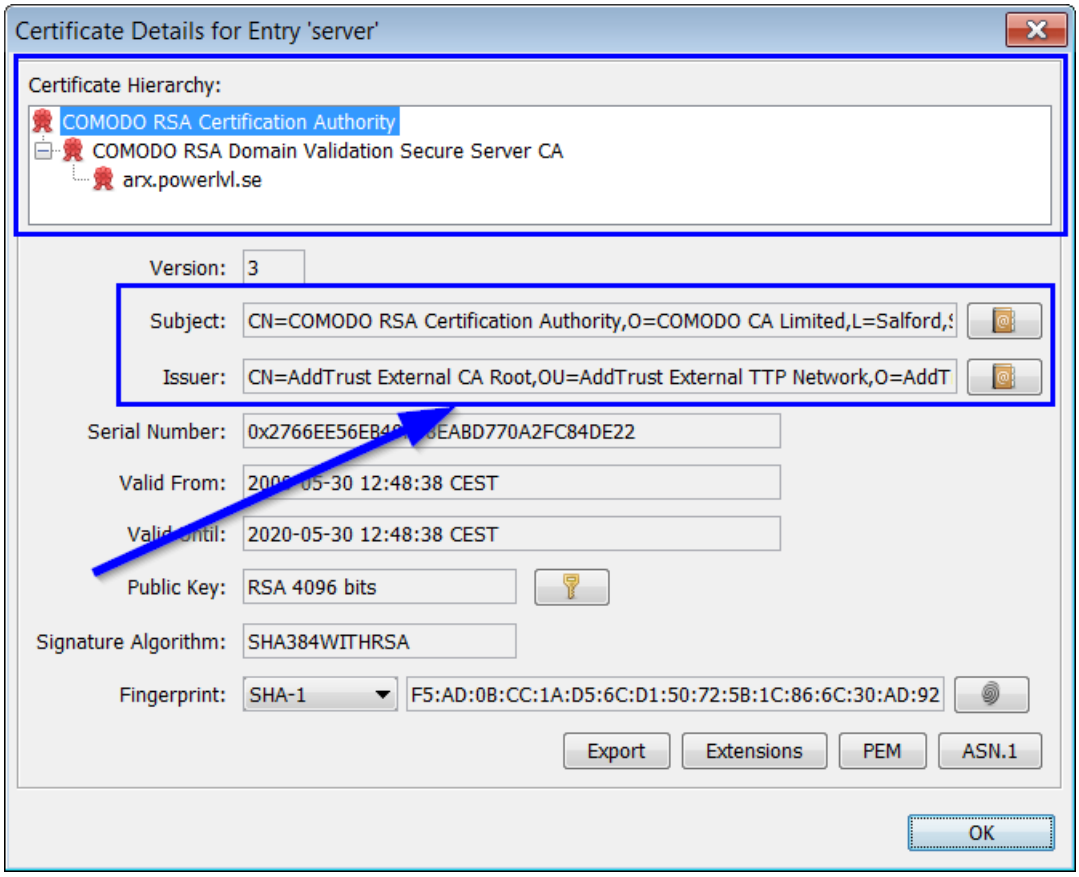


VIKTIGT
*Först upprepa **Steg 5** att lägga till ett nytt certifikat.*

*Nu väljer vi certifikatet som har ett liknande namn som **utfärdaren** vi hittade i **Steg 7***

*i det här fallet var utfärdarens namn **COMODO RSA Certificate Authority** och vi väljer filen **COMODORSAAddTrustCA.crt***

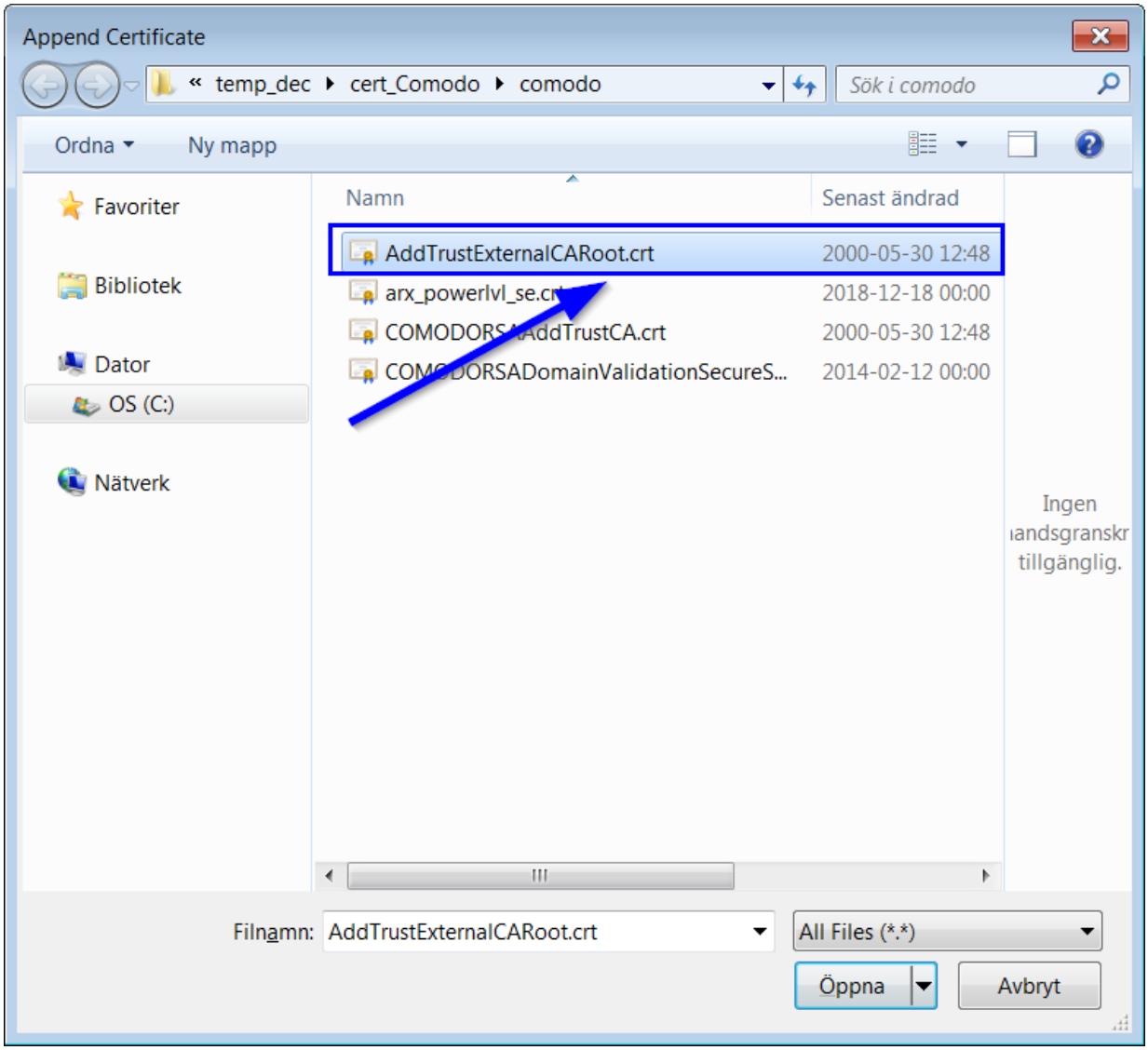
Steg 9



*Som i **Steg 7** dubbelklicka på ditt certifikat och nu har vi ännu ett certifikat i vår förtroendekedja.*

*Igen notera namnet på utfärdaren, i detta fall **AddTrust External CA Root***

Steg 10

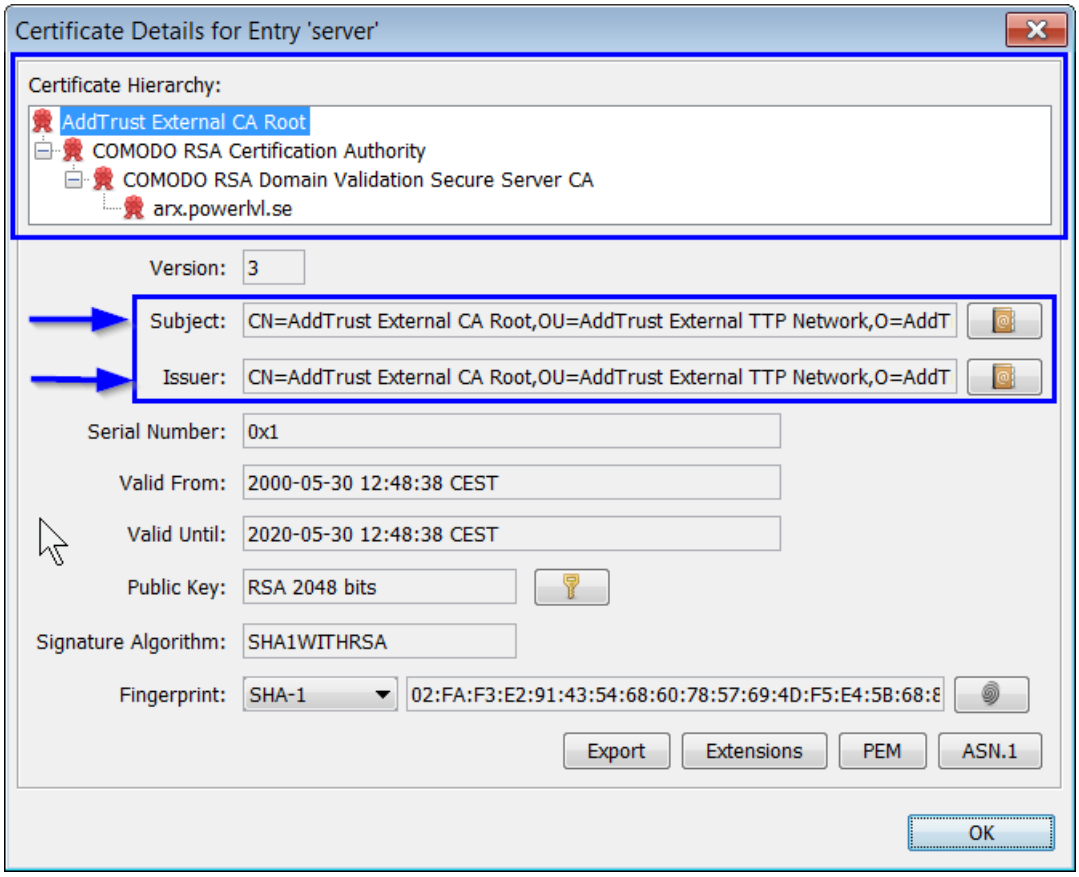


VIKTIGT
*Först upprepa **Steg 5** att lägga till ett nytt certifikat.*

*Nu väljer vi certifikatet som har ett liknande namn som **utfärdaren** vi hittade i**Steg 9***

*I det här fallet var utfärdarens namn **AddTrust External CA Root** och vi väljer filen **AddTrustExternalCARoot.crt***

Steg 11 (klart)



Dubbelklicka på ditt certifikat och verifiera att certifikatkedjan ser ut som något liknande det här.

*Klicka på certifikatet som ligger längst upp i trädet (vilket har namnet **Root CA** eller liknande)*

*Verifiera att den har samma **Subject** och **Issuer***

> ⚠ **OBSERVERA**
> Kom ihåg att spara innan du fortsätter

Konfigurera ARX för att använda din nya JavaKeystore

Om du har lyckats skapa ditt certifikat och din JavaKeystore enligt ovan.

Börja med att hitta din JavaKeystore-fil, i den här guiden har vi kallat filen **ARX.YOURDOMAIN.COM.jks**

Nu måste vi kopiera den här filen till mappen **secrets** där ARX är installerad

> **VIKTIGT**
> Som standard är ARX installerad i katalogen:
>
> **C:\Program Files\ASSA\ARX**

Nästa steg är nu att redigera filen **system.properties** filen finns i ARX installationskatalog. Leta upp följande textrader och se till att det står "true" på två av raderna och att rätt certifikatfil-namn hamnar på dom andra två raderna. Om någon av textraderna saknas i filen så lägg till dom i slutet av filen.

system.httpsIntegrationServerEnabled=true

system.httpsIntegrationServerJks=**ARX.YOURDOMAIN.COM.jks**

system.httpsWebServerEnabled=true

system.httpsWebServerJks=**ARX.YOURDOMAIN.COM.jks**

> **VIKTIGT**
> Du kan bara använda samma JavaKeystore-fil för integrationsservern och webbservern om de använder samma adress.

Generera manuellt via Keytool-kommandon

Det här är de kommandon som behövs om du inte vill installera Keystore Explorer-programmet. (Rekommenderas bara för avancerade användare.)

> **VIKTIGT**
> Se till att du kan använda kommandot keytool i din terminal.
>
> Enklast att verifiera detta är att öppna konsolen och skriva **keytool**

Generera en Javakeystore-fil

Här skapar vi en JavaKeystore-fil liknande den som skapades ovan med programmet *KeyStore Explorer*

```
keytool -genkeypair -alias server -keyalg RSA -keysize 2048 -storepass secret -
keypass secret
-keystore ARX.YOURDOMAIN.COM.jks -dname "CN=ARX.YOURDOMAIN.COM,
O=YOUR COMPANY NAME, ST=, C=SE"
```

Följande värden **måste ändras:**

Detta kommando kommer att generera en nyckelkatalogfil som heter **ARX.YOURDOMAIN.COM.jks**

Certifikatet kommer att ha det gemensamma namnet **ARX.YOURDOMAIN.COM**

Organisationen i certifikatet skall vara **DITT FÖRETAGS NAMN**

Det land som certifikatet kommer att innehålla kommer att vara **SE** vilket är sverige

Generera en "Certificate Signing Request"

När du har ditt självsignerade certifikat, då kan du skapa en CSR.

```
keytool -certreq -alias server -file ARX.YOURDOMAIN.COM.csr -storepass secret -
keypass secret -keystore ARX.YOURDOMAIN.COM.jks
```

Filnamnet kan vara vad som helst, i det här fallet var det **ARX.YOURDOMAIN.COM.csr**

Importera CSR Response-filerna

I den här guiden hanterar vi bara två exempel, en är där du har fått en **.p7b** fil och en annan där du fått en **.crt** fil.

.p7b fil import

För att lägga till det signerade certifikatet i din keystore:

```
keytool -import -alias server -file ARX.YOURDOMAIN.COM.crt -
keystore ARX.YOURDOMAIN.COM.jks -storepass secret
```

.crt fil import

Det här kan vara lite knepigt. Först måste du lägga till det signerade certifikatet i din keystore, sen måste du importera rätt filer i rätt ordning.

Vi måste nu först lägga till Root CA-certifikatet och sen ta det en efter en, här är dom kommandon som vi behöver gå igenom för att få vår SSL från återförsäljaren **Comodo** att fungera.

```
keytool -import -alias RootCert -file fromComodo\AddTrustExternalCARoot.crt -
keystore ARX.YOURDOMAIN.COM.jks -storepass secret
```

```
keytool -import -alias Middle1 -file fromComodo\COMODORSAAddTrustCA.crt -
keystore ARX.YOURDOMAIN.COM.jks -storepass secret
```

```
keytool -import -alias Middle2 -file fromComodo
\COMODORSADomainValidationSecureServerCA.crt -keystore
ARX.YOURDOMAIN.COM.jks -storepass secret
```

keytool -import -alias **server** -file **fromComodo\ARX.YOURDOMAIN.COM.crt** -keystore ARX.YOURDOMAIN.COM.jks -storepass secret

OBSERVERA

Kom ihåg att de olika aliaserna ska vara unika utom i det sista steget där du lägger till ditt signerade SSL-certifikat i din Keystore, det är därför du måste använda **server** alias i sista steget.