A MITEL
PRODUCT
GUIDE

OpenScape Deployment Service V10

PKI Basic Configuration Guide

Service Documentation

09/2025

A31003-S23A0-S101-02-76A9

Mitel

Notices

The information contained in this document is believed to be accurate in all respects but is not warranted by Mitel Europe Limited. The information is subject to change without notice and should not be construed in any way as a commitment by Mitel or any of its affiliates or subsidiaries. Mitel and its affiliates and subsidiaries assume no responsibility for any errors or omissions in this document. Revisions of this document or new editions of it may be issued to incorporate such changes. No part of this document can be reproduced or transmitted in any form or by any means - electronic or mechanical - for any purpose without written permission from Mitel Networks Corporation.

Trademarks

The trademarks, service marks, logos, and graphics (collectively “Trademarks”) appearing on Mitel’s Internet sites or in its publications are registered and unregistered trademarks of Mitel Networks Corporation (MNC) or its subsidiaries (collectively “Mitel), Unify Software and Solutions GmbH & Co. KG or its affiliates (collectively “Unify”) or others. Use of the Trademarks is prohibited without the express consent from Mitel and/or Unify. Please contact our legal department at iplegal@mitel.com for additional information. For a list of the worldwide Mitel and Unify registered trademarks, please refer to the website: http://www.mitel.com/trademarks.

Contents

PKI DLS Basic Configuration Guide

1 Configuration overview

1.1 PKI Basics

Public key infrastructure is the term used to describe the laws, policies, procedures, standards, and software that regulate or control the operation of certificates and public and private keys. More specifically, a PKI is a system of digital certificates, certification authorities, and other registration authorities that verify and authenticate the validity of each party involved in an electronic transaction.

A PKI consists of the following basic components:

Digital certificates . Electronic credentials, consisting of public keys, which are used to sign and encrypt data. Digital certificates provide the foundation of a PKI.
One or more certification authorities (CAs) . Trusted entities or services that issue digital certificates. When multiple CAs are used, they are typically arranged in a carefully prescribed order and perform specialized tasks, such as issuing certificates to subordinate CAs or issuing certificates to users.
Certificate policy and practice statements . The two documents that outline how the CA and its certificates are to be used, the degree of trust that can be placed in these certificates, legal liabilities if the trust is broken, and so on.
Certificate repositories . A directory service or other location where certificates are stored and published. In a Windows Server 2003 domain environment, the Active Directory® directory service is the most likely publication point for certificates issued by Windows Server 2003–based CAs.
Certificate revocation lists (CRL) . Lists of certificates that have been revoked before reaching the scheduled expiration date.
Certificate trust lists . These are signed lists, which are located on the client, of trusted CA certificates. Certificate trust means that a certificate is part of a certificate trust list (CTL) or that the CTL contains a trusted certificate from another CA that is part of the certificate's certificate chain. Windows Server 2003 domain administrators can use Group Policy objects (GPOs) to publish and maintain CTLs.
Key archival and recovery . A feature that makes it possible to archive and recover the private key portion of a public-private key pair, in the event that a user loses his or her private keys, or an administrator needs to assume the role of a user for data access or data recovery. Private key recovery does not recover any data or messages; it merely enables the recovery process.
Public key standards . Standards developed to describe the syntax for digital signing and encrypting of messages and to ensure that a user has an appropriate private key. To maximize interoperability with third-party applications that use public key technology, the Windows Server 2003 PKI is based on the standards recommended by the Public-Key Infrastructure (X.509) (PKIX) working group of the Internet Engineering Task Force (IETF). Other standards that the IETF has recommended also have a significant impact on public key infrastructure interoperability, including standards for Transport Layer Security (TLS), Secure/ Multipurpose Internet Mail Extensions (S/MIME), and Internet Protocol security (IPSec).

A PKI system makes it possible for an organization to do the following:

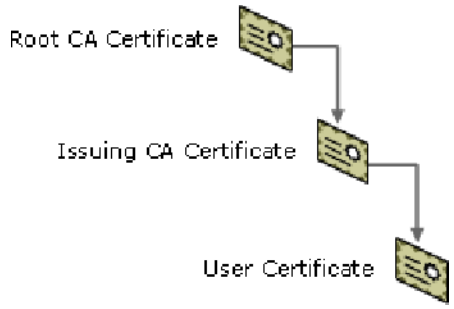
1.1.1 How a Public Key Infrastructure Works

- **Publih ctificates.** The PKI administrator makes certificate templates available to client**s** (us**er**s, services, applications, and computers) and enables additional CAs to issue certificates.

- **Enroll clients.** To participate in a PKI, users, services, or computers must request and receive certificates from an issuing CA or a Registration Authority (RA). Typically, enrollment is initiated when a requester provides unique information and a newly generated public key. The CA administrator or enrollment agent uses the information provided to authenticate the identity of the requester before issuing a certificate.
- **Use certificates.** Clients use their certificates, which are validated or invalidated in a timely manner if CAs and certificate revocation lists are available to verify or deny their authenticity. If they are validated, a PKI provides an easy way for users to use keys in conjunction with applications that perform public key cryptographic operations, making it possible to provide security for e-mail, e-commerce, and networks.
- **Renew or revoke certificates.** A well-designed PKI makes it easy for you to renew or revoke existing certificates, and to manage the trust level associated with certificates used by different clients or for different applications.

The status of a public key certificate is determined by means of the chain building process. Chain building is the process of building a trust chain, or certification path, from the end certificate to a root CA that is trusted by the security principal. Figure 16.2 shows a certification path in a two-level CA hierarchy.

Certification Path in a Two-Level CA Hierarchy



In this example, the issuing CA issued the User certificate, and the root CA issued the certificate of the issuing CA. This is considered a trusted chain, because it terminates with a root CA certificate that has been designed and implemented to meet the highest degree of trust.
The chain building process validates the certification path by checking each certificate in the certification path from the end certificate to the certificate of the root CA. If the CryptoAPI discovers a problem with one of the certificates in the path, or if it cannot find a certificate, the certification path is either considered invalid or is given less weight than a fully validated certificate.

This guide is based on an example PKI system comprised of the following Windows 2008 R2 servers.

- DNS server
- MS Root Certification Authority – Enterprise CA
- MS Subordinate Certificate Authority - Enterprise CA
- DLS Server

DLS with a configured PKI connector (chapter 6) is integrated in this PKI system. The previous chapters are included in the guide for reference purposes. In fact if the customers PKI environment is already in operation only the integration of DLS to that environment is required. In that case the reader can consult directly chapter 6. He can look at previous chapters in case he needs clarifications or cross references to a standard PKI environment.

1.2 DNS server

All CAs (Certification Authorities) and DLS server must be in the same Active directory.

1.3 Microsoft Root Certification Authority

The DLS PKI Connector provides a ready to use plug-in for standard PKI environments based on Windows Server 2003, Windows Server 2008 / 2008 R2 and Windows Server 2012. Integration with Windows PKI will show following restrictions:
- The CAs used from DLS PKI Connector to request and enroll certificate may only be Enterprise CAs (**Standalone CA is not supported,** although a Standalone CA may be the trust anchor on customers infrastructures, Standalone CAs can't be used as issuing CAs for DLS PKI Connector)
- All CAs are accessed by the same windows credentials, this means DLS must be configured using a domain account with proper rights on all CAs (issuing and revoking certificates). I.e. CAs in different windows domain forests are not supported.

1.4 Microsoft Subordinate Certification Authority

If a two-tier CA hierarchy is needed, then a subordinate CA can be installed. Root CA will sign this subordinate CA.

1.5 DLS server

The main components of PKI DLS is the PKI connector and the PKI plugin.
The PKI Connector is a new integrated component of the DLS. The connector acts as an abstraction layer to be able to communicate with individual standard and non-standard PKI environments used by different customers.

The figure below shows the main components of the overall architecture:

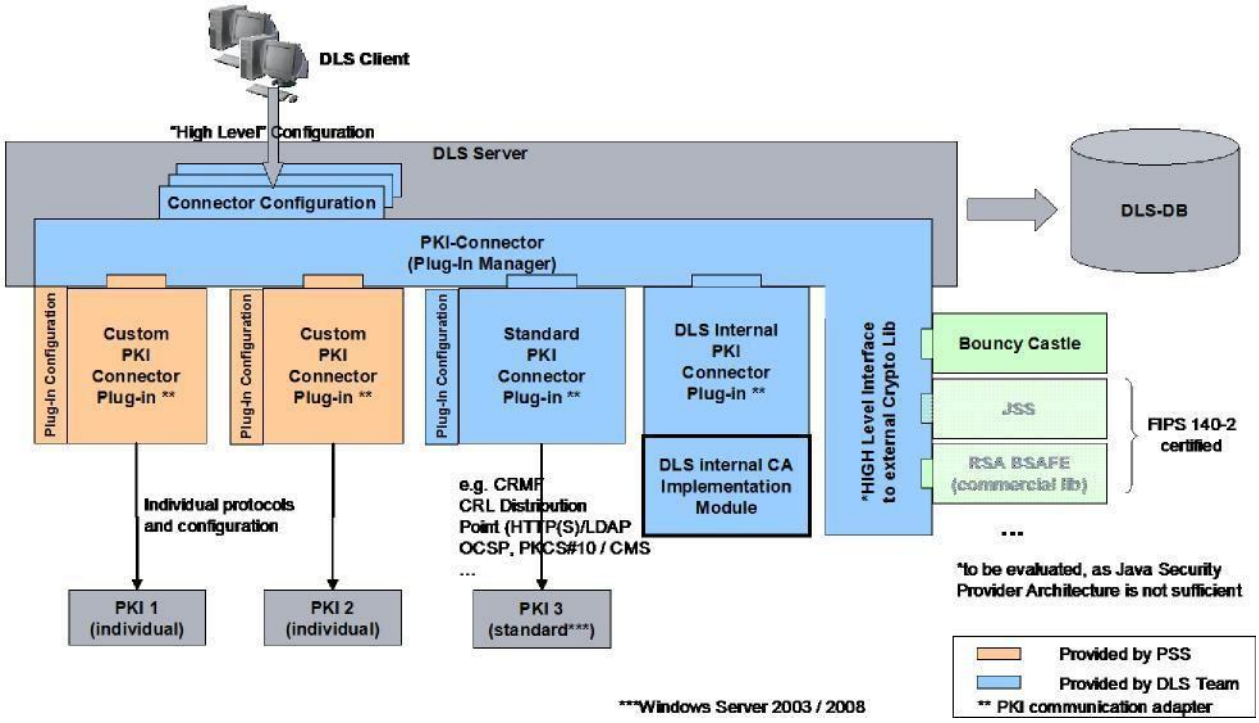


Figure 1 Overview: DLS PKI Connector Architecture

To be able to support standard and non-standard PKI infrastructures, pluggable connectors, hiding the details of communication and the concrete adaption and configuration to interact with an individual PKI, can be used and plugged into the DLS PKI Connector.

1.5.1 *Supported Configuration Scenarios*

The DLS can be used in a single tenant as well as in a multitenant environment This scenario has also an impact on the DLS PKI Connector. The following scenarios are supported:

- Customer uses
  - own PKI
  - DLS internal CA
  - both, DLS internal CA and own PKI, depending on the certificate type and use
- Customer provides DLS services to other customers (multi-tenancy) using
  - own centralized PKI
  - DLS internal CA
  - both, DLS internal CA and centralized own PKI, depending on the certificate type and use
  - either DLS internal CA, centralized own PKI or an individual customer PKI provided and maintained by its customer, depending on the certificate type and use

These mixed scenarios can be configured and assigned to a customer or tenant via the DLS Client GUI.

2 DNS Configuration

The active directory is implemented in A Windows 2008 R2 server which is installed and configured as follows.

2.1 DNS Installation

Set the IP of your DNS as shown in Figure 1. If you have a primary DNS above PKI DNS then enter its address in 'Alternate DNS server' field.

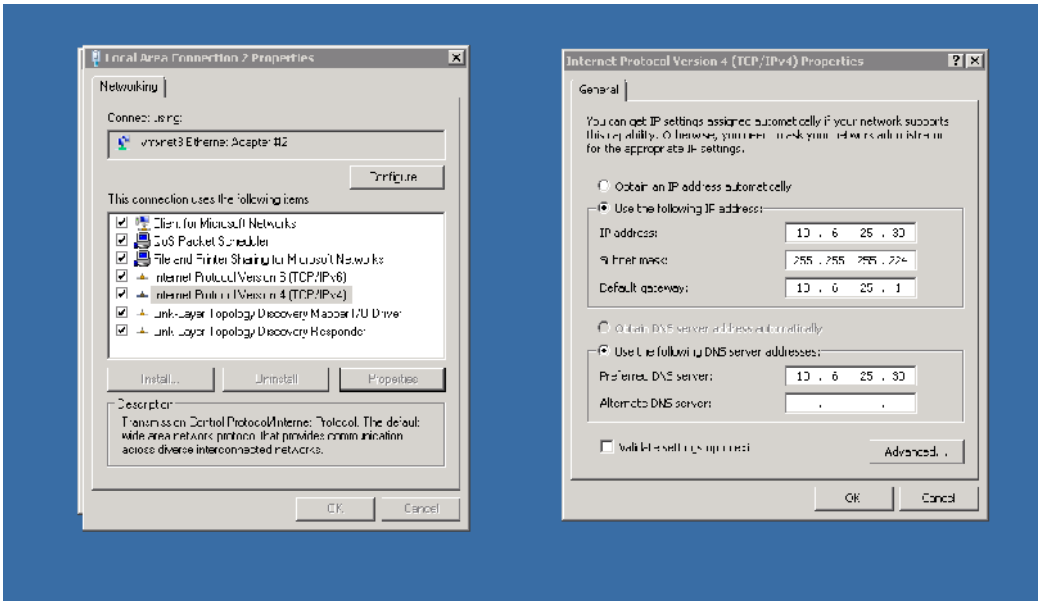


Figure 2 DNS network settings

In command line execute the command 'dcpromo' to install active directory of DNS as shown in Figure 2.

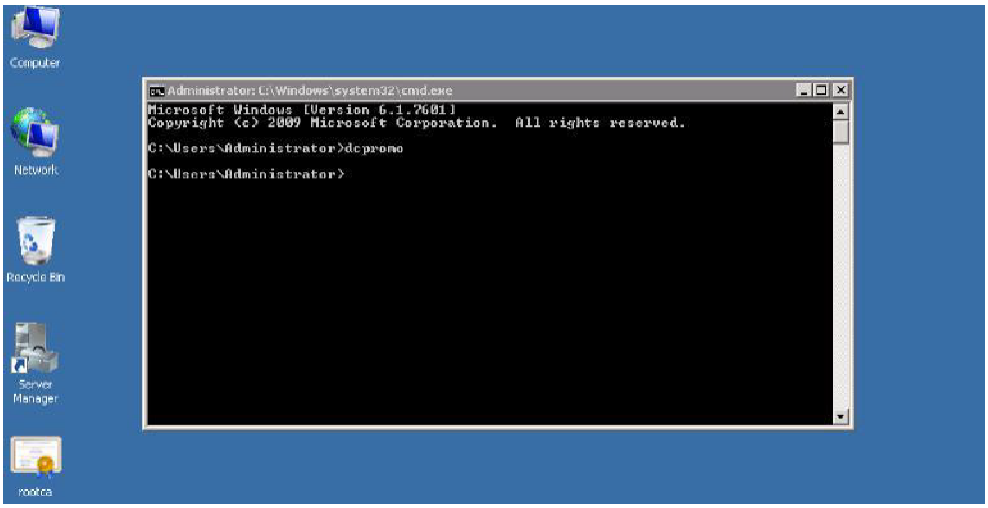


Figure 3 `dcpromo` to install active directory

Select ”Create a new domain in a new forest”, as shown in Figure 4.

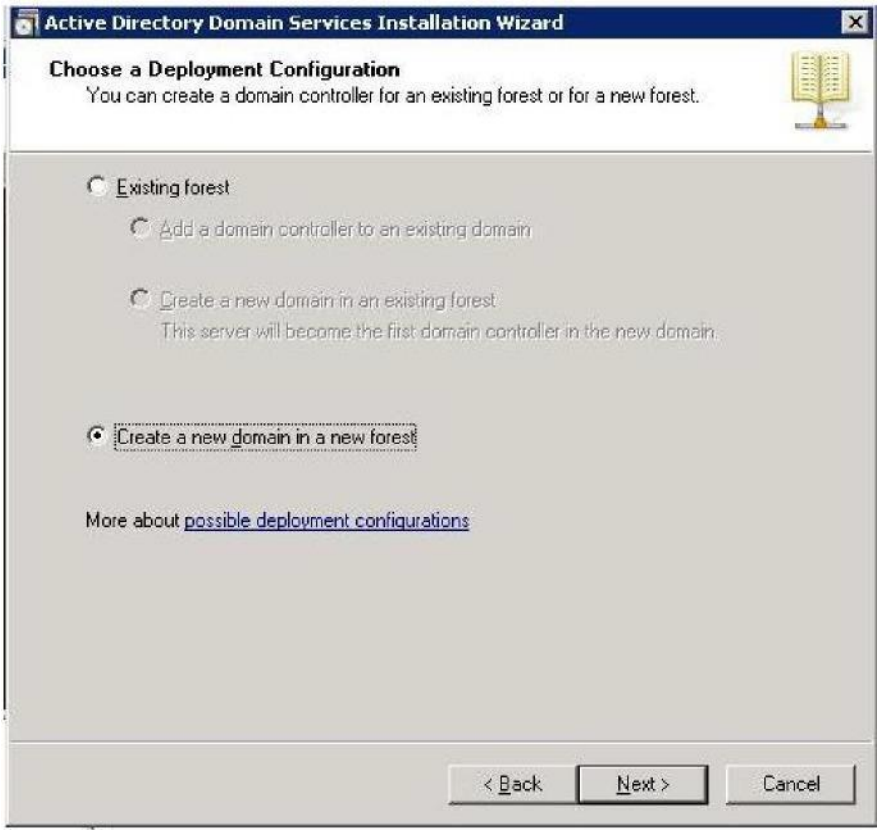


Figure 4 domain controller type

Specify the domain names as shown in Figures 5 and 6.

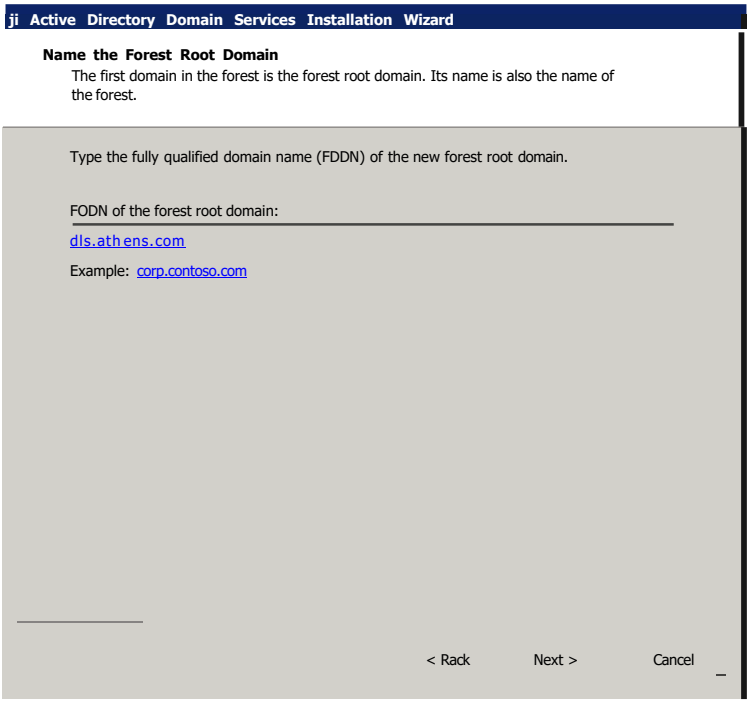


Figure 5 Specify Domain Name

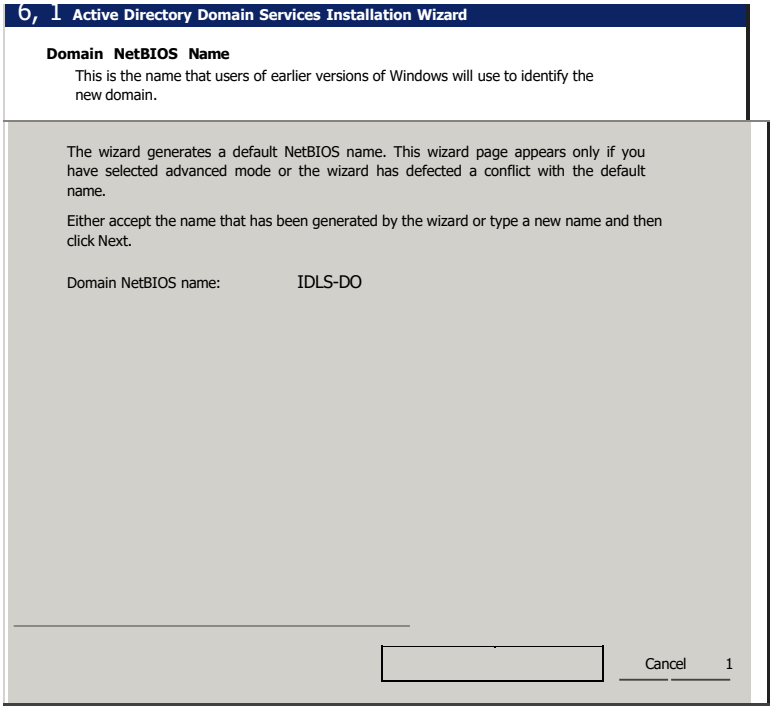


Figure 6 Domain NetBios name

Follow default settings as shown in Figures 7, unless local configuration requests for different settings.

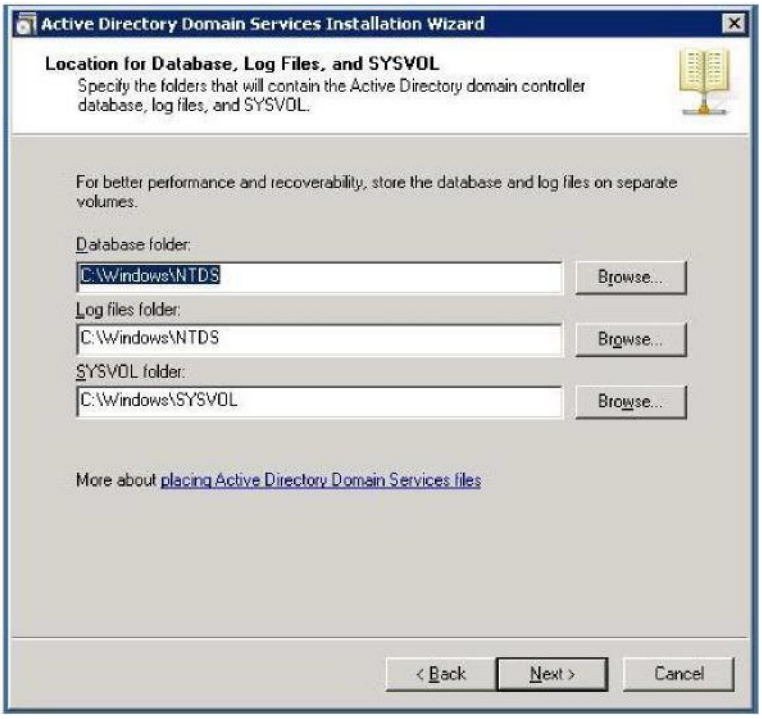


Figure**7** Db and log settings

Set a restore password for the Administrator account as shown in Figure 8.

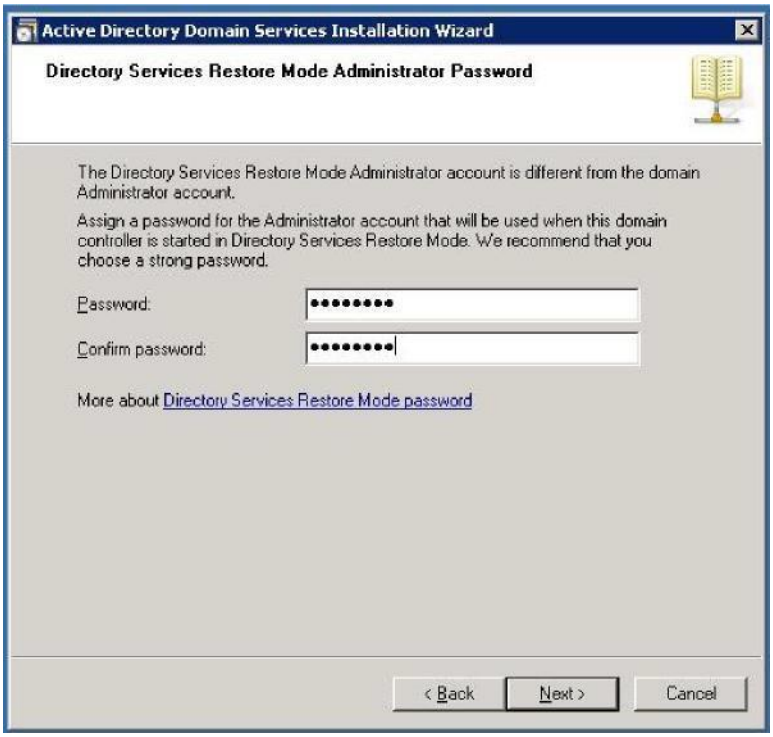


Figure 8 Set Admin password

DNS installation continues with respective review options. Select "Next" in order to proceed with the installation process. Active directory installation is finalized (figure 9).

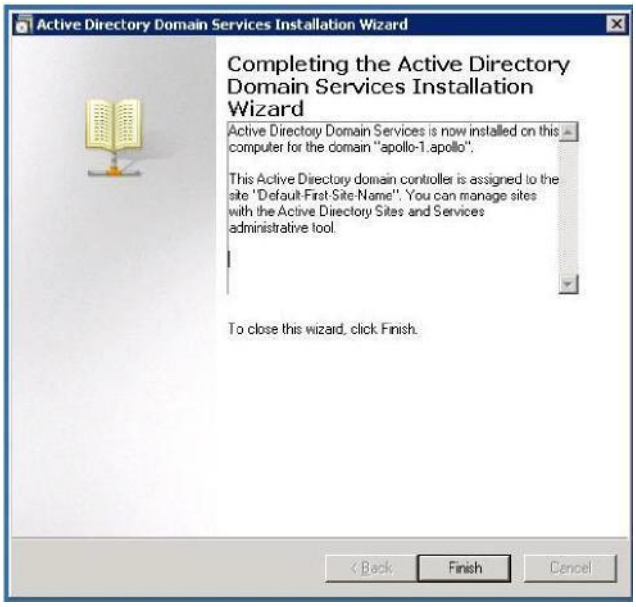


Figure 9 Finalize Active Directory installation

2.2 Create 'dls' user in DNS

The system needs a 'dls' user to be created to facilitate the communication between its entities.

Respectively to each customer's infrastructure 'dls' user may be individually granted with all permissions mentioned in this guide (on user level), or he may be a part of a user group which will be granted with respective permissions.

This guide is based on user level permissions.

Through Server Manager, navigate to Active Directory Domain Services -> Active Directory Users and Computers ->User and select to create new user
Right click in 'Users' folder to create a new user as shown in figure 10

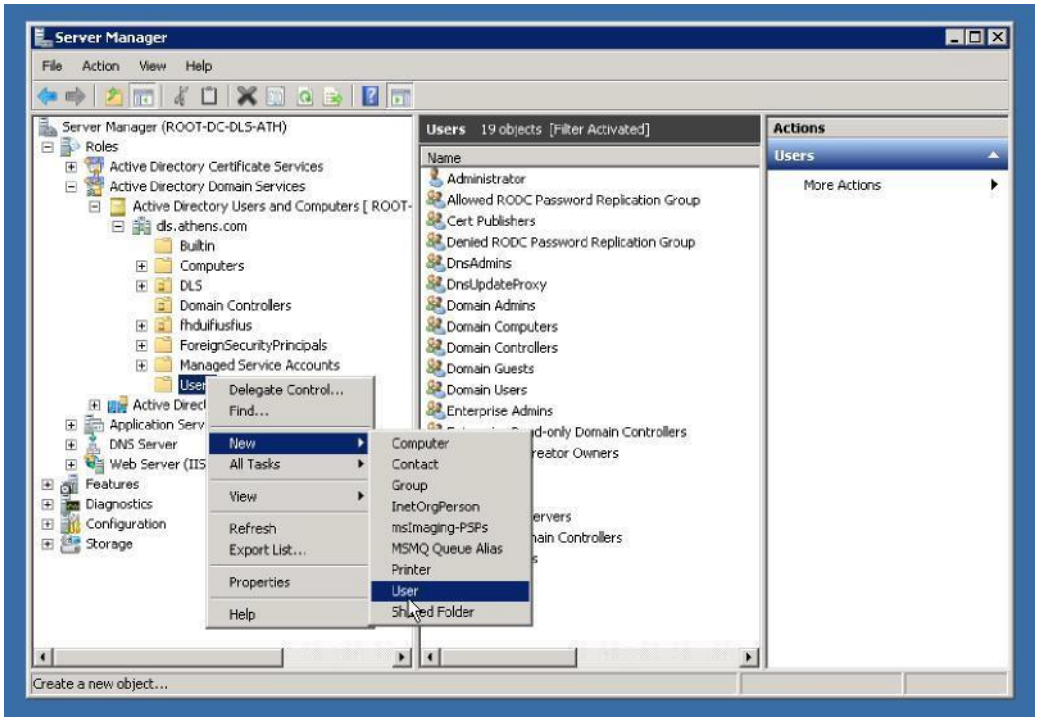


Figure 10 Create new user

Create domain user "dls" as shown in figure 11.

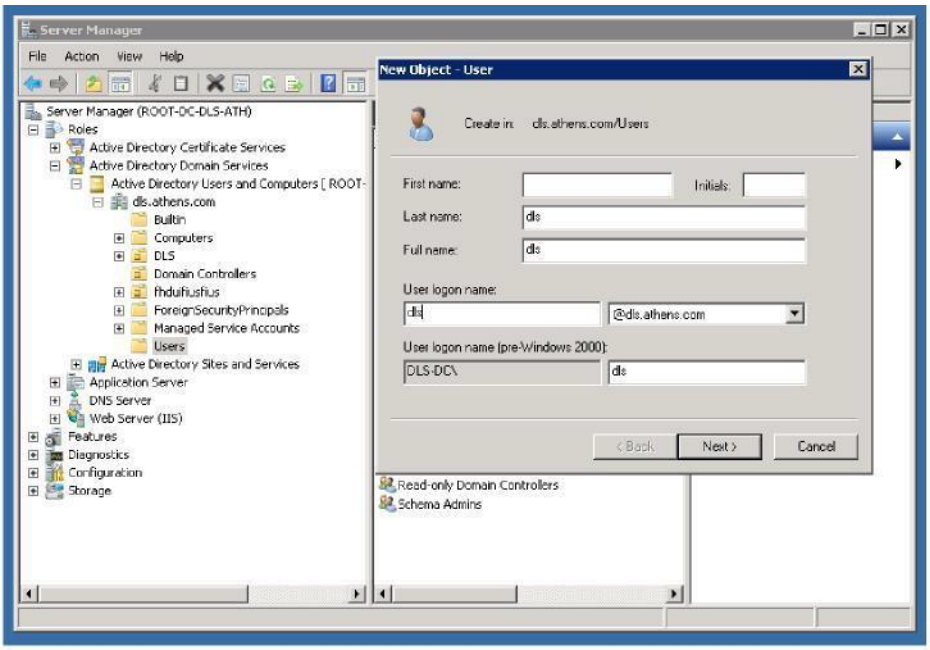


Figure 11 Create 'dls' user in the active directory

Assign a password to this user and select 'Password never expires' as shown in figure 12.

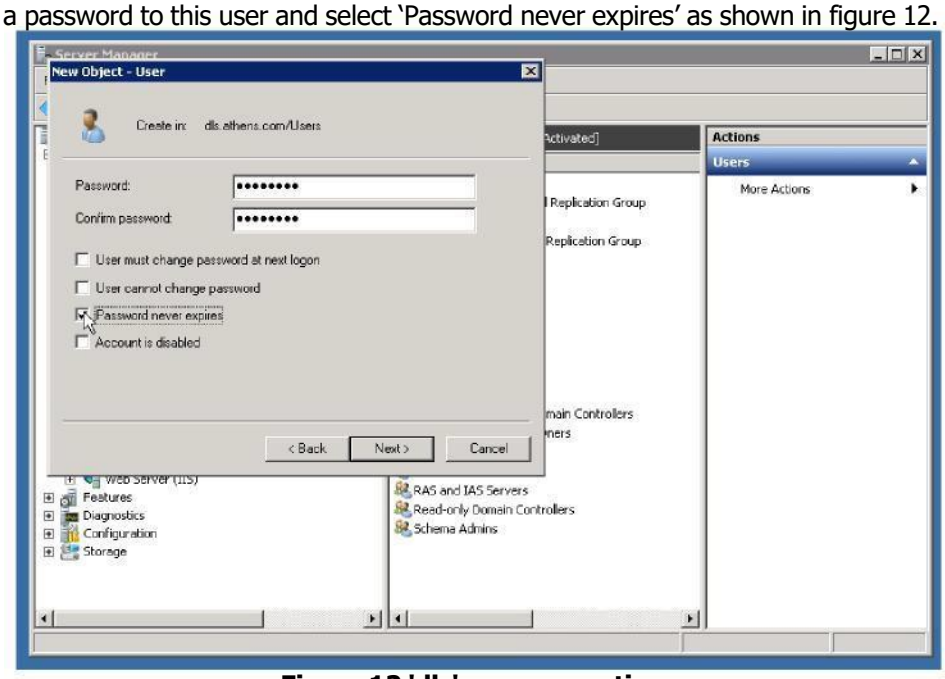


Figure 12 'dls' user properties

Add 'dls' user to Remote Desktop users as shown in figure 13.

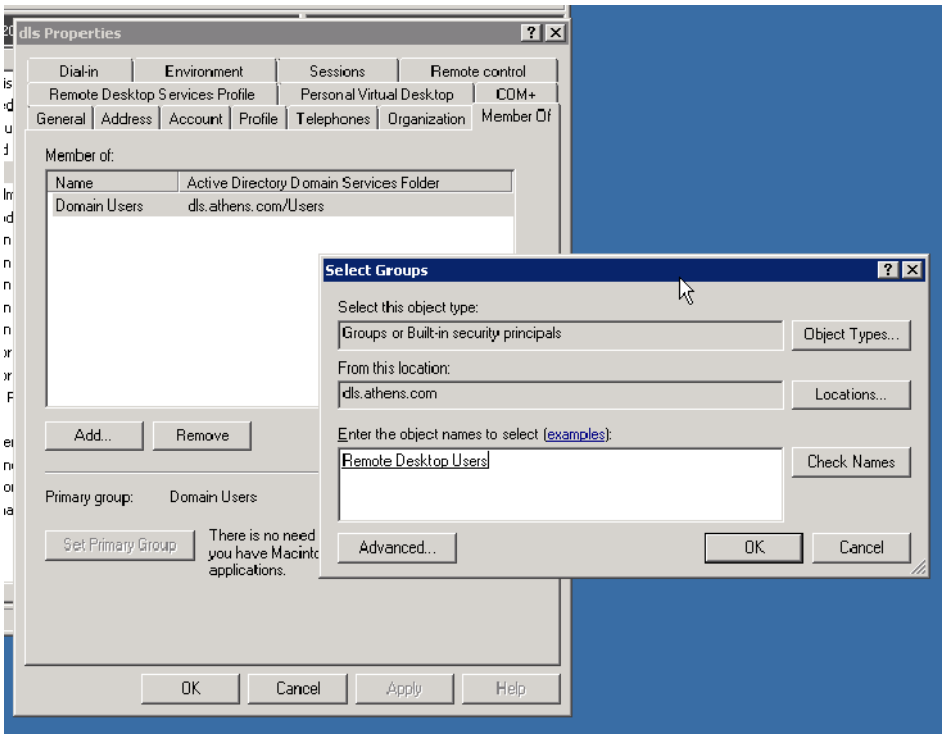


Figure 13 Add 'dls' to Remote Desktop Users

Enable remote control to 'dls' user (figure 14). This is required by the DLS PKI operation.

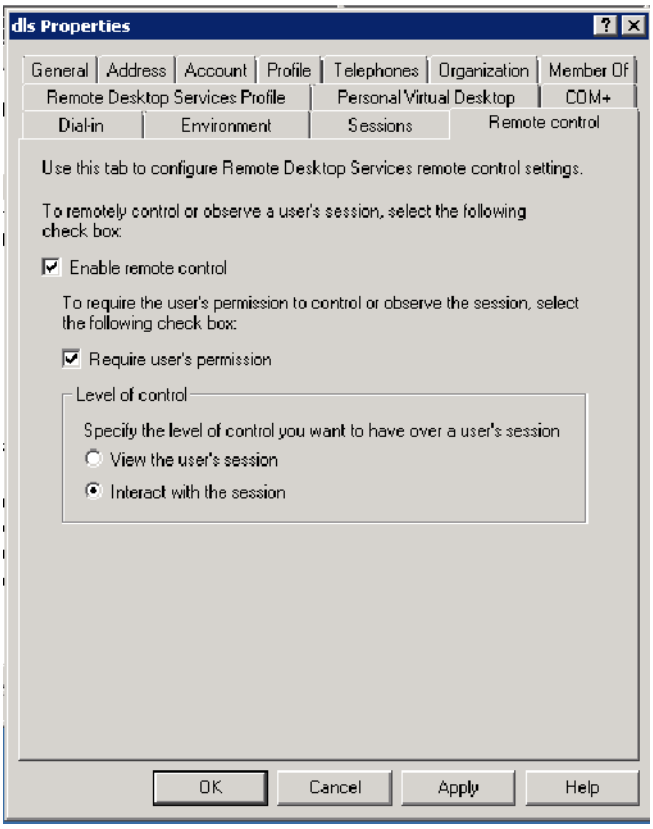


Figure 14 'dls' user remote control properties

From now on login to the DNS domain (not to the local machine) as in figure 15.

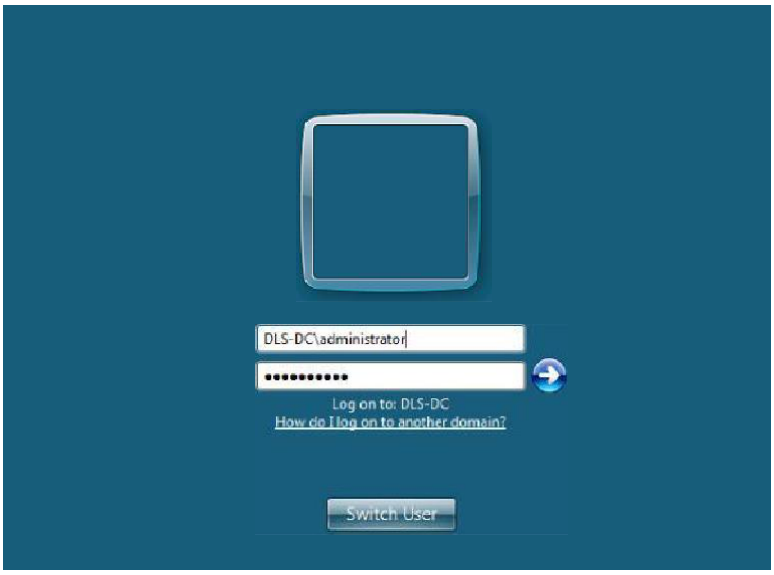


Figure **15** Login Administrator to the domain

3 Microsoft Root Certification Authority

Only **Enterprise CAs** are supported by DLS PKI. Enterprise CAs are integrated with Active Directory. They publish certificates and CRLs to Active Directory. Enterprise CAs use information stored in Active Directory, including user accounts and security groups, to approve or deny certificate requests. Enterprise CAs use certificate templates. When a certificate is issued, the enterprise CA uses information in the certificate template to generate a certificate with the appropriate attributes for that certificate type.

A **Root CA** is the CA that is at the top of a certification hierarchy and must be trusted unconditionally by clients in your organization. All certificate chains terminate at a root CA. Whether you use enterprise or stand-alone CAs, you need to designate a root CA. Because there is no higher certifying authority in the certification hierarchy, the subject of the certificate issued by a root CA is also the issuer of the certificate. Likewise, because the certificate chain terminates when it reaches a self-signed CA, all self-signed CAs are root CAs. Windows Server 2003 only allows you to designate a self-signed CA as a root CA. The decision to designate a CA as a trusted root CA can be made at either the enterprise level or locally, by the individual IT administrator.

A root CA serves as the foundation upon which you base your certification authority trust model. It guarantees that the subject public key belongs to the subject identity information that is contained in the certificates it issues. Different CAs might also verify this relationship by using different standards; therefore it is important to understand the policies and procedures of the root certification authority before choosing to trust that authority to verify public keys.

The root CA is the most important CA in your hierarchy. If your root CA is compromised, every other CA and certificate in your hierarchy might have been compromised. You can maximize the security of the root CA by keeping it disconnected from the network and using subordinate CAs to issue certificates to other subordinate CAs or to end users.

3.1 CA Server settings

The network settings of the root CA will have as preferred DNS the previously installed DNS server and as default gateway any other higher in the hierarchy DNS (figure 16). Give a computer name to the CA and add it to the active directory as shown in figure17 & figure 18.

You will see a welcome message as soon as the CA enters the domain (figure 19). Now you can login root CA in the domain as shown in figure 20.

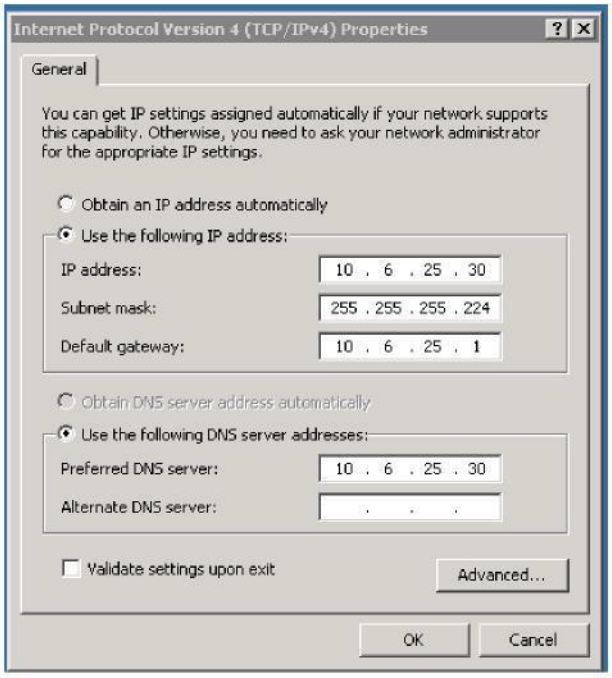


Figure 5 Root CA network settings

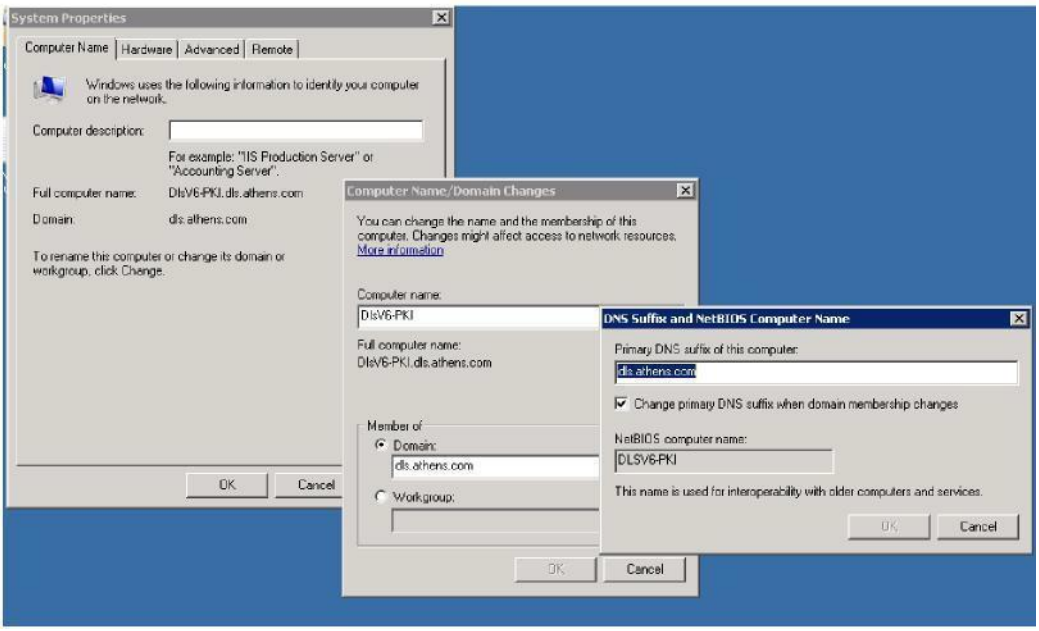


Figure 17 Root6 CA joins the domain

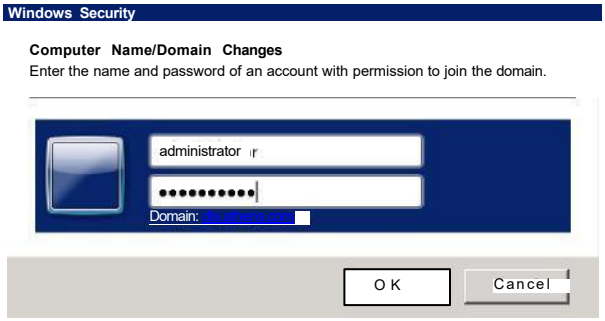


Figure 7 Account with permission to join the domain

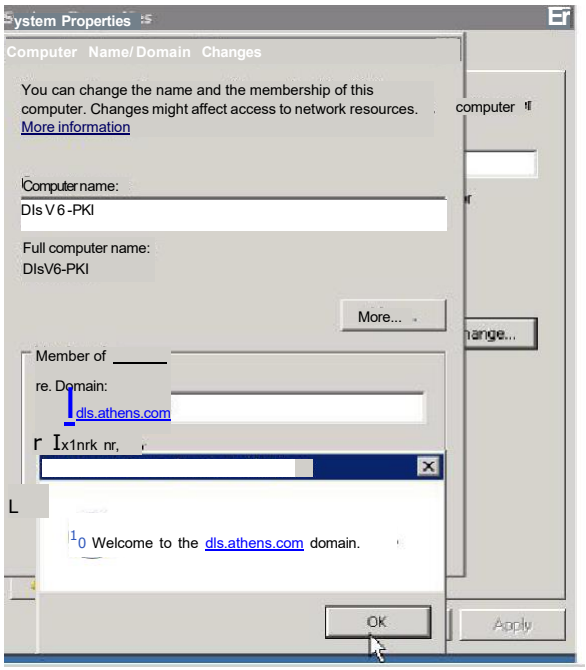


Figure 8 Root CA joined the domain

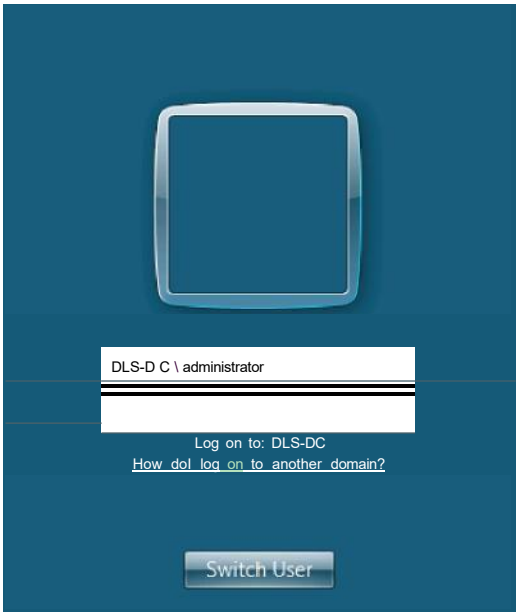


Figure 20 Login CA to the domain

3.2 Install CA and other Application server services

A root CA MS server will need some extra windows components apart from the default. Install extra Application server components as shown in figure 21.

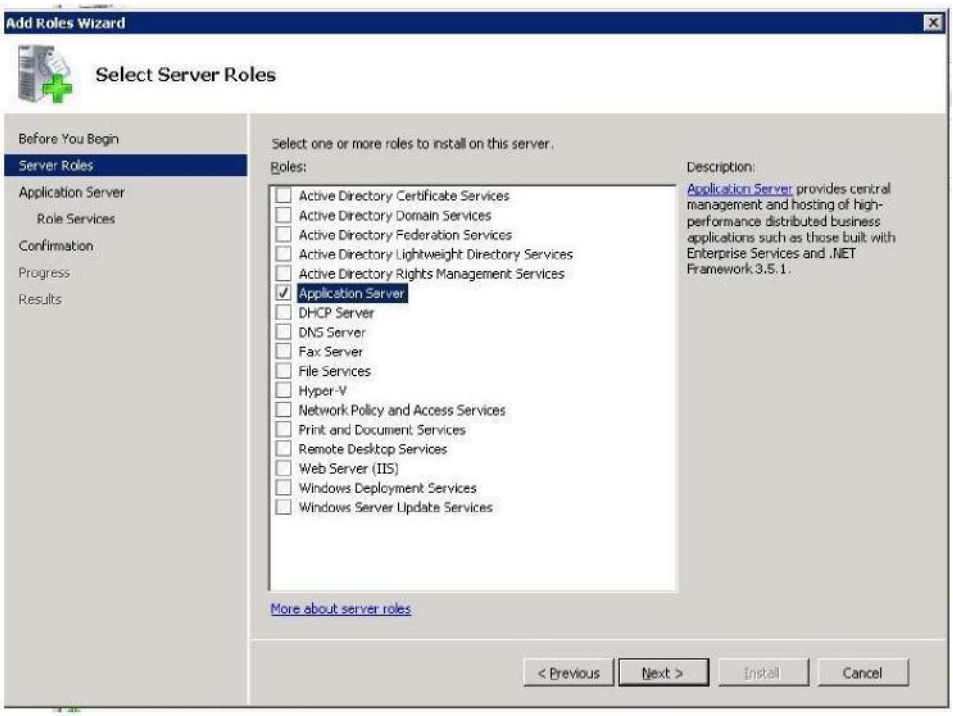
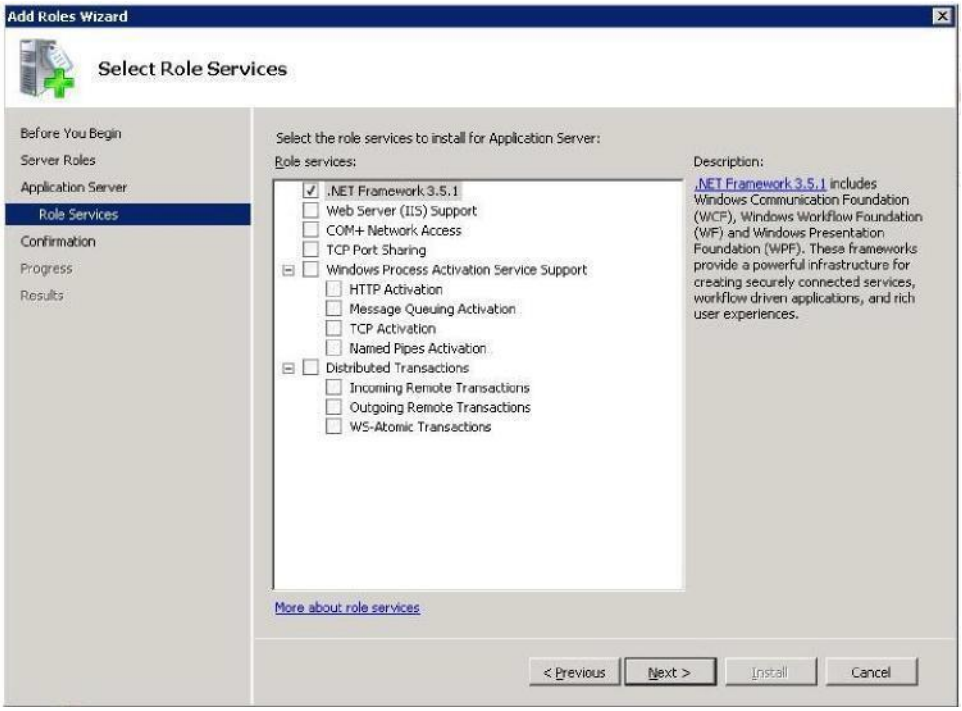


Figure 9 Install needed windows components if missing

Select to install NET Framework 3.5.1 as in figure 22.



It is also necessary to install Certificate services as in figure 23 & 24. Select to install a certification authority. Select CA type 'Enterprise Root CA' as in figure 25.

Figure 22 Install missing windows components

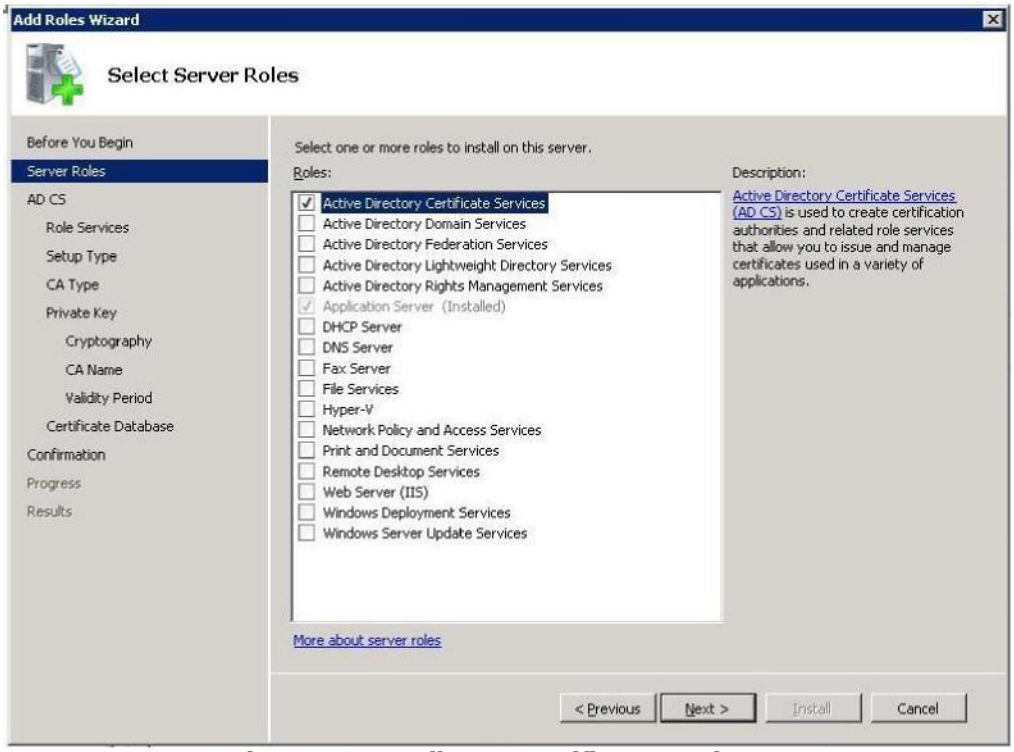


Figure 10 Install WIN Certificate services

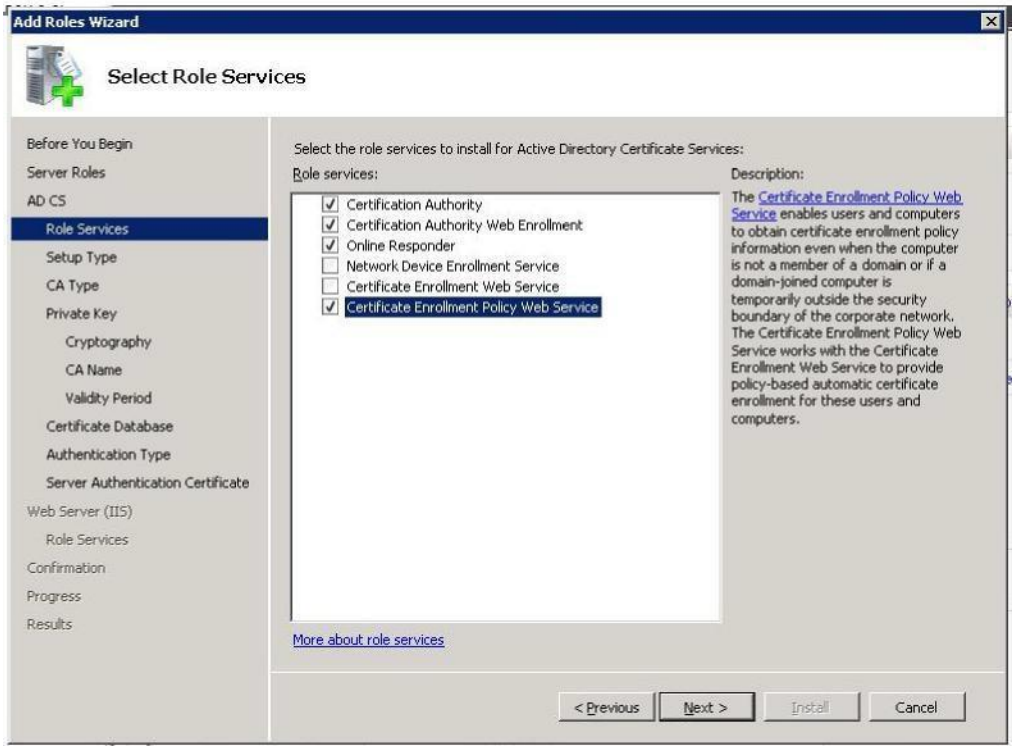


Figure 24 Install WIN Certificate services

Note: Network Device Enrollment Service and Certificate Enrollment Web Service can not be installed at the same time Certification Authority is being installed. You should finished the installation as described here and if necessary you should return and add respective roles after the installation of Certificate authority is finished.

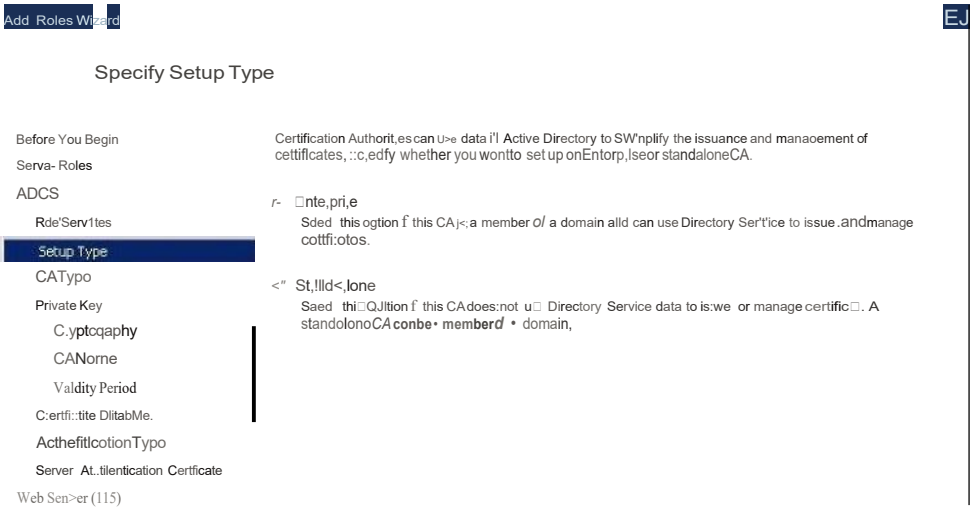

Add Roles Wizard
EJ
Specify Setup Type
Before You Begin
Serva- Roles
ADCS
Rde'Serv1tes
Setup Type
CATypo
Private Key
C.yptcqaphy
CANorne
Valdity Period
C:ertfi::tite DlitabMe.
ActhefitlcotionTypo
Server At..tilentication Certficate
Certification Authorit,es can U>e data i'l Active Directory to SW'nplify the issuance and manaoement of cettifIcates, ::c,edfy whether you wontto set up onEntorp,lseor standaloneCA.
r- □nte,pri,e
Sded this ogtion f this CA J<;a member o/ a domain alld can use Directory Ser't'ice to issue .andmanage cottfi:otos.
<" St,!lld<,lone
Saed thi□QJltion f this CA does:not u□ Directory Service data to is:we or manage certific□. A standolonoCA conbe• member• • domain,

Figure 25 Select CA Type "Enterprise"

Select Root CA as in figure 26


Add Role-□Wizard
EI
Specify CA Type
Before'iou fl:el)r'l
Server Roles
AI) CS
R.ole ServlcM
SetLOType
CA Type
Prlvotel(ey
uyptogr,pl,y
CA□e
Valdty Pertocl
Ciertific.M:ieD.&tM),!11□
Authenbc<lban Type
Server Authenticetion Certficate
Wobs,...,,(115)
A COl)'tIY!atlonc/ root□ndsl.M1dh□e CJu con be co!Vlgi.red to create□l-.er8'tlic:□Ipubic keyWrMtrucbse (PKO.
A root CAls ,aCA th!IlkSLll!IS1$ ownseMl□utd certlfc,ata. A subordnate CA reteives i:.$certllcate fromollOther
CA,Spet:ify whether vou -wont to setU?e rootorSYbotdnoto CA.
C-- !torl'.CA
sele<t tilts opuon f,..,•• rostaltnoti-.. frst or ony certttcatta, outhor1tyn • p<Alltckey1nfrostru,Me,
/"" Sybordnote CA
Soloct tilts option f ,...,CA will obtants CA mt:ifbto fromanolnor CA iigho, In• pubk key
rifrastnJctu-e.

Figure 26 Select "Root CA"

Select whether you want to use a pre-existing private key or create new (figure 27) and specify the cryptographic service provider, hash algorithm and key length as shown in figure 28.

Set Up Private Key

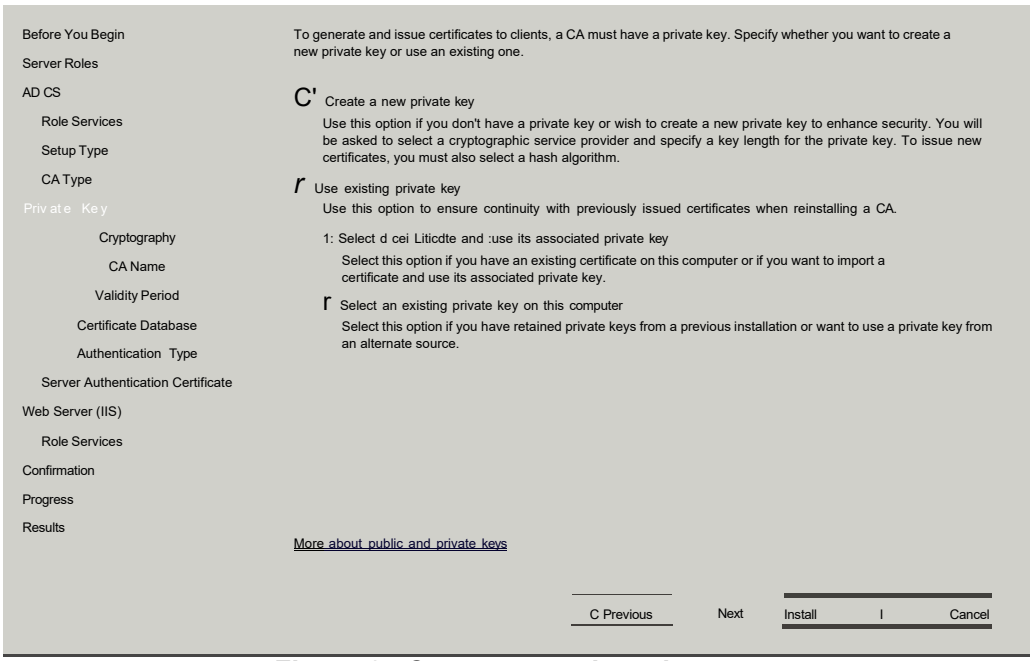


Figure 27 Create new private key

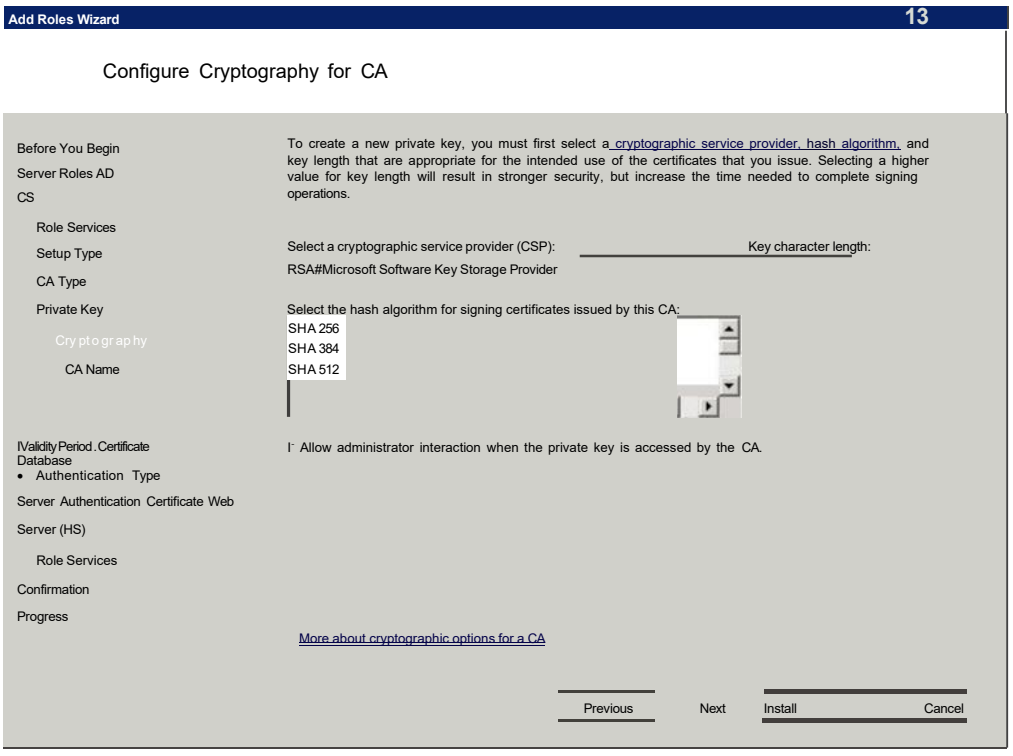


Figure 28 Cryptographic service provider, hash algorithm and key length

Give a CA common name (CN) and DN suffix (figure 29).

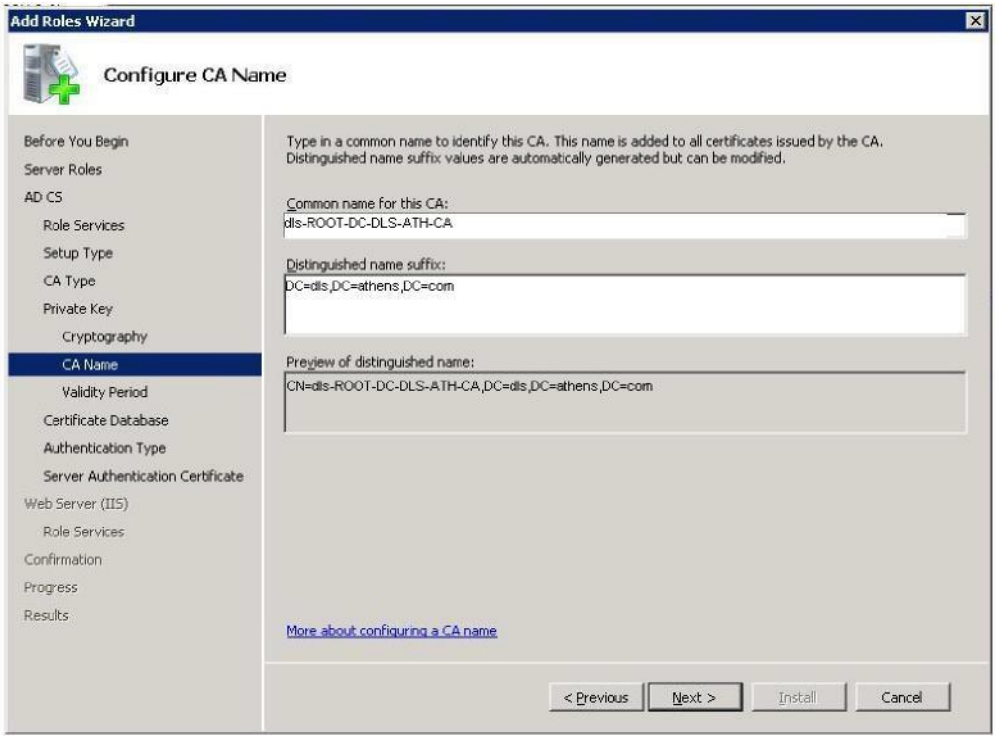


Figure 29 Configure CA common name and DN suffix

Specify validity period for the certificate that will be generated for your Root CA as shown in figure 30.

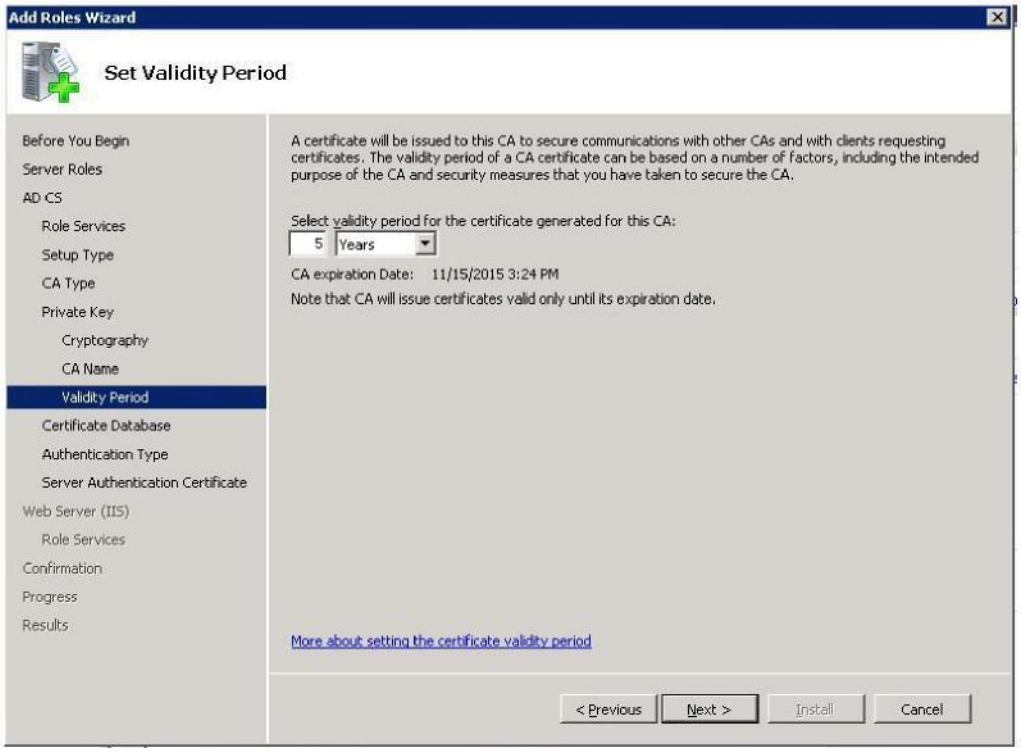


Figure 30 CA certificate validity period

Finally, specify Certificate database and Certificate database log location (figure 31), the type of authentication clients will use (figure 32) and a server authentication certificate suitable for ssl encryption (figure 33).

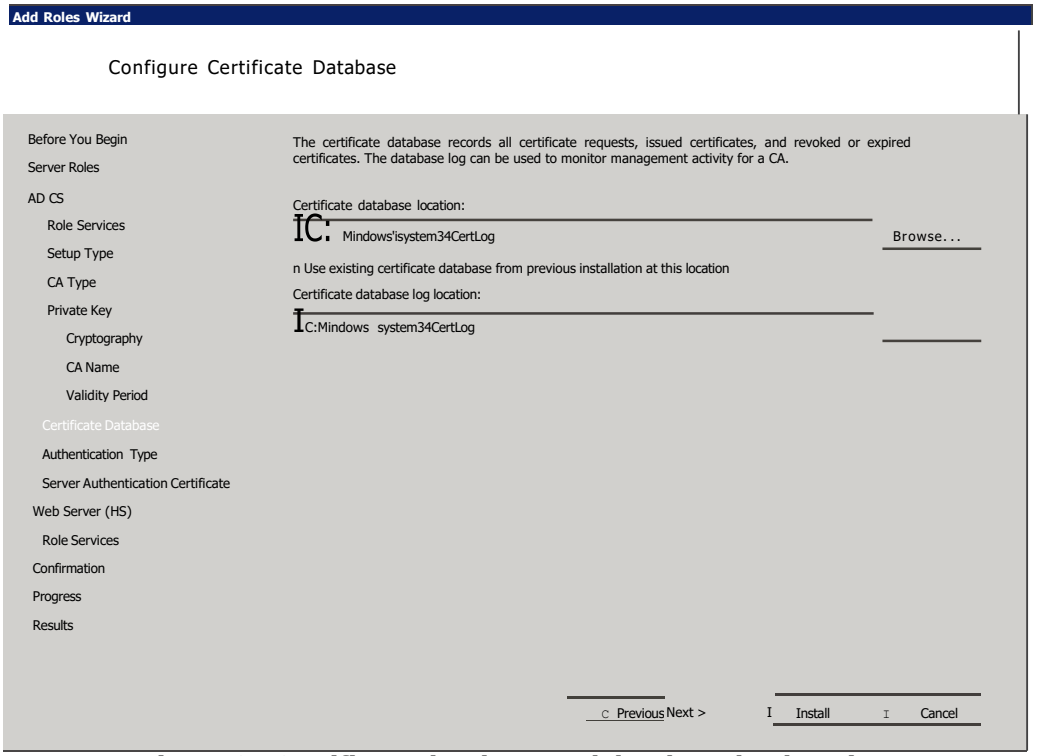


Figure 31 Certificate database and database log location

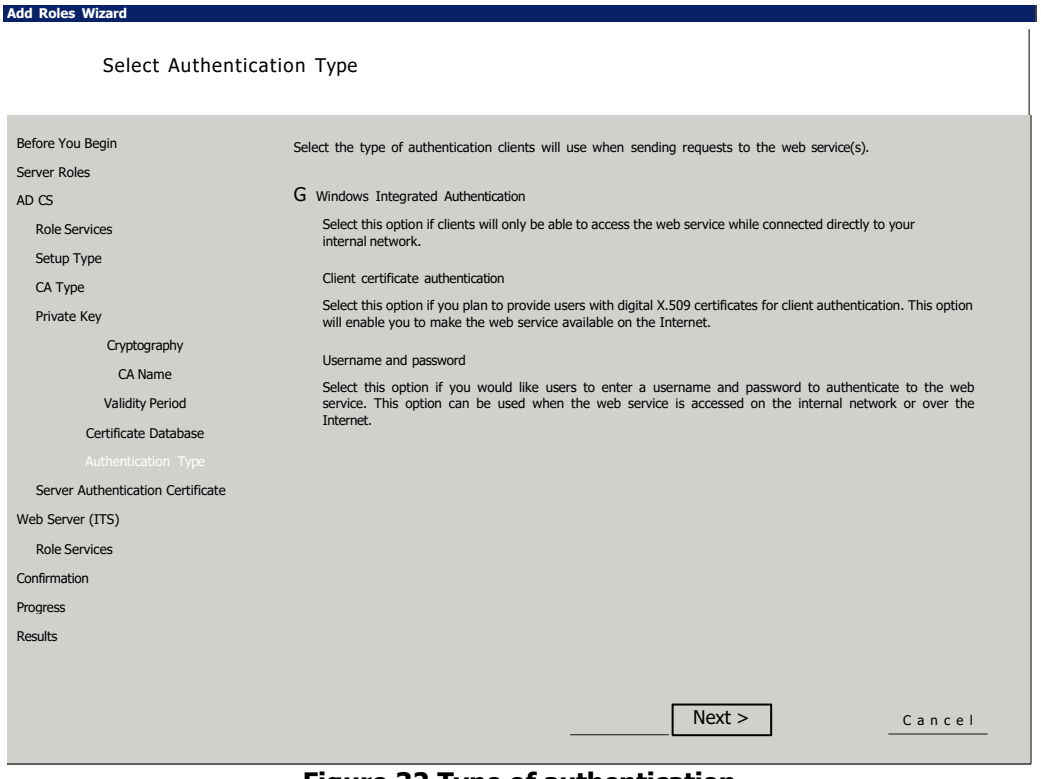


Figure 32 Type of authentication

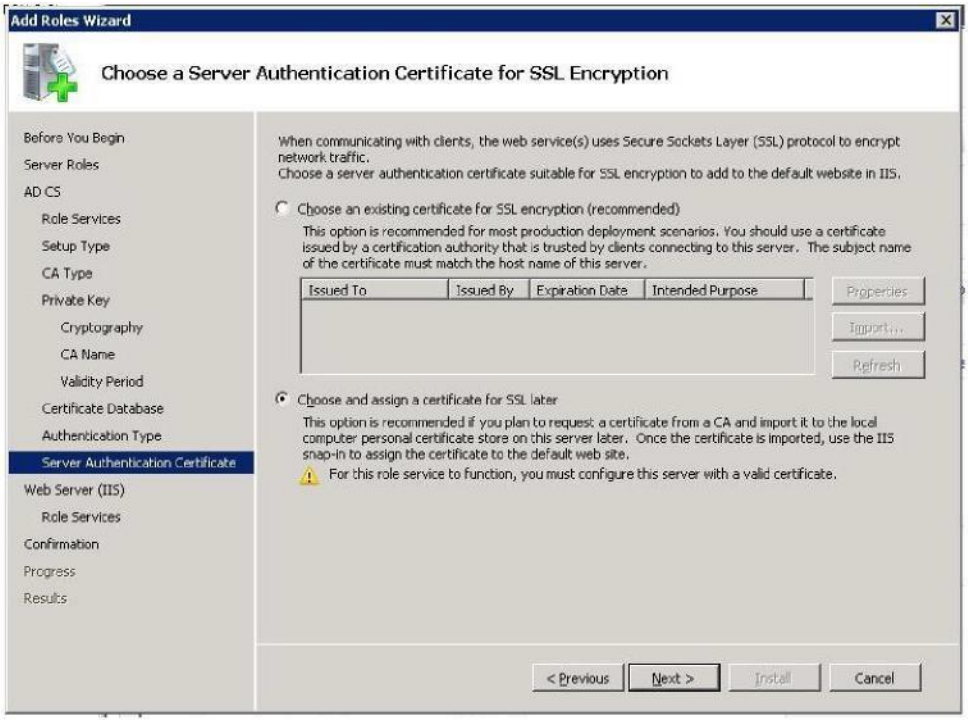


Figure 33 Certificate for SSL Encryption

Finishing the installation, system will prompt you with installation wizard for Web Server (IIS) Role services (needed for Certication Authority Web Enrollment). Proceed with default values as shown in figure 34.

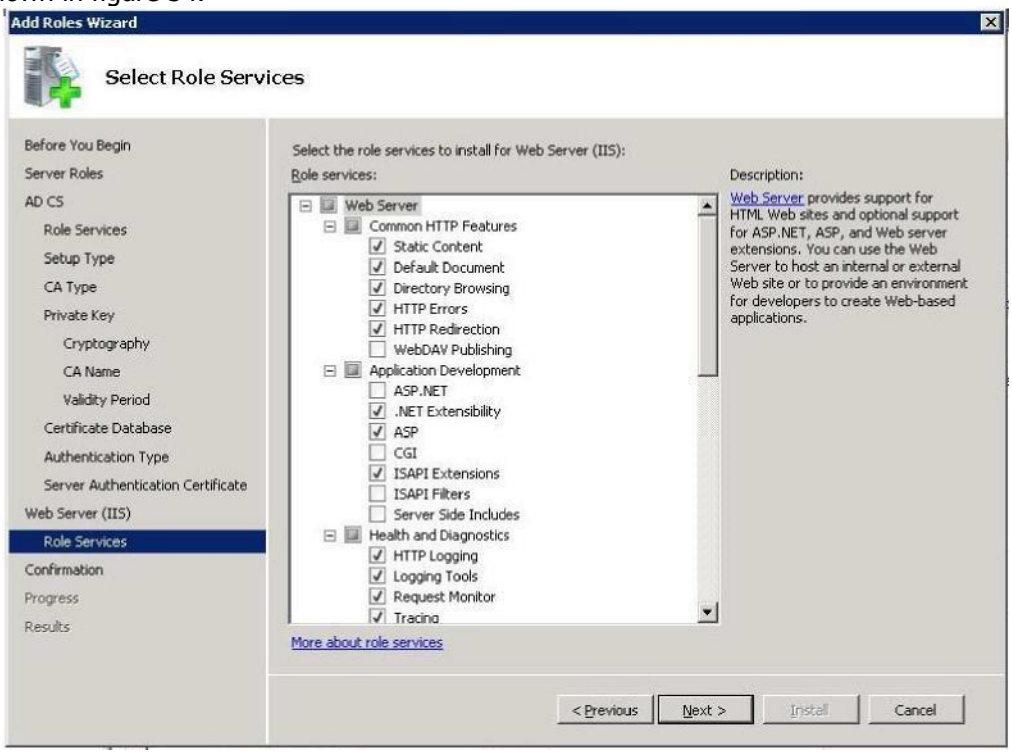


Figure 34 Installation wizard for Web Server (IIS) Role services

Proceed with "Install" and wait for the installation to finish. A success message will be displayed with the end of the installation (figure 35)

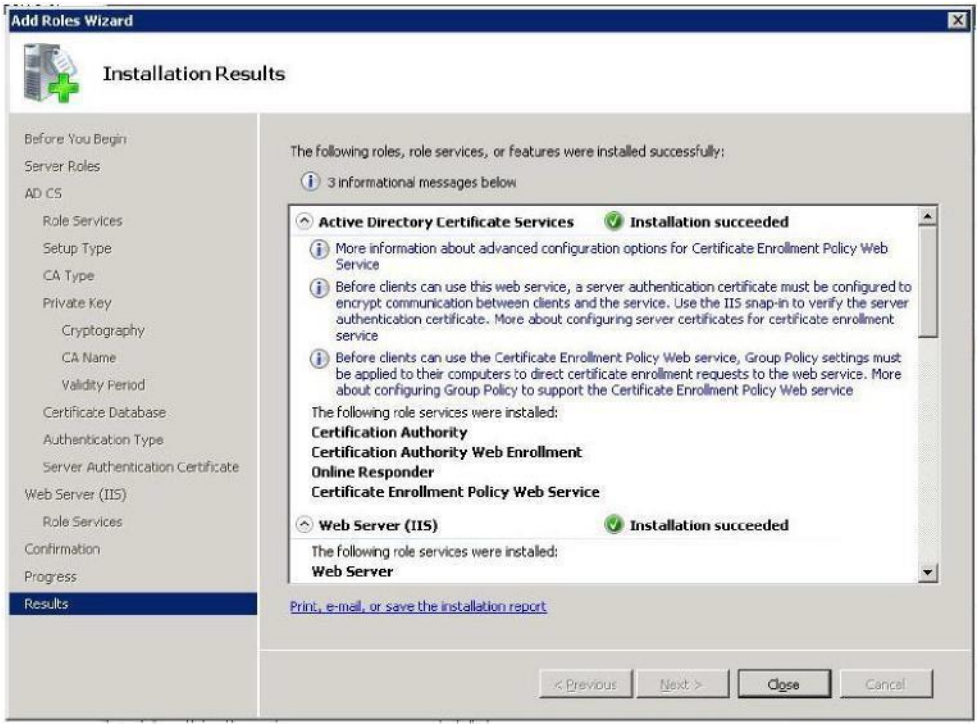


Figure 35 End of installation

3.3 Certificate templates configuration

PKI DLS will be using templates to request on demand certificates from the CA. Hence a number of templates can be configured in the CA for different purposes. For the WBM interface the Web Server template would be used. For the phone's client authentication (e.g. for secure communication with DLS) the User Template is needed and is actually a mandatory configuration parameter for the plugin.

In order to configure the certificate templates open the CA management window (figure 36).

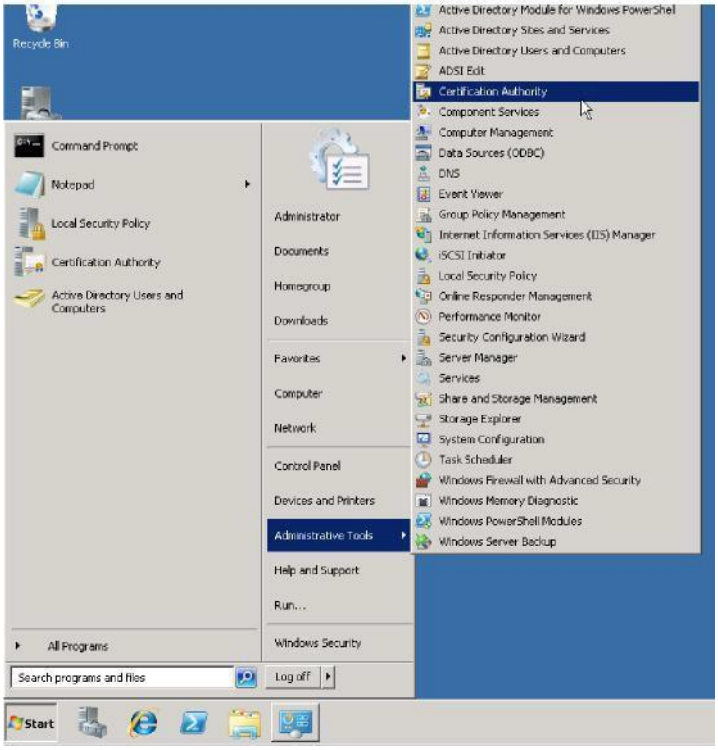


Figure 36 Open Certification Authority management window

There select your previously installed CA and right click to open its properties. Set a 'template type' request handling as shown in figure 37.

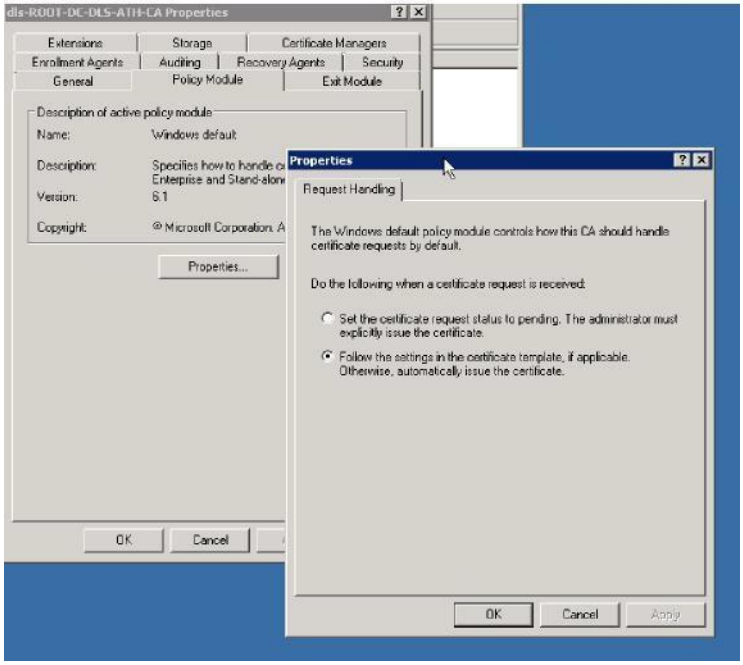


Figure 37 CA properties

Open templates manager window by right clicking 'Certificate Templates' and selecting 'Manage' as in figure 38. A new window opens named **certtmpl** – **[Certificate Templates].**

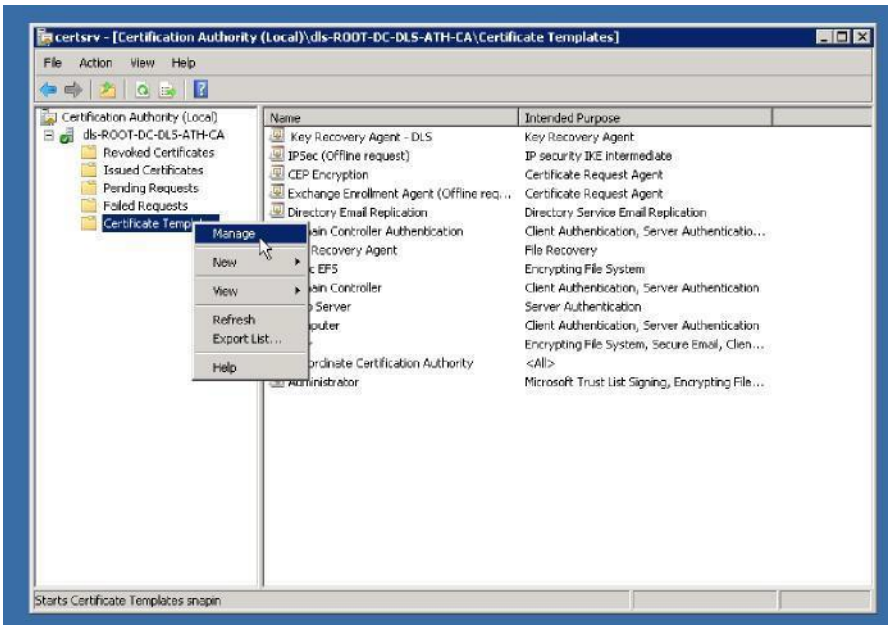


Figure 38 Open Certificate Templates manager

In Certificate Templates window we want to modify two templates, 'User' template and 'Web Server' template. It is better instead of modifying the existing templates to copy them and modify the new ones. Select 'User' template and right click to duplicate it as in figure 39. Select Windows Server version for the new template as in figure 40

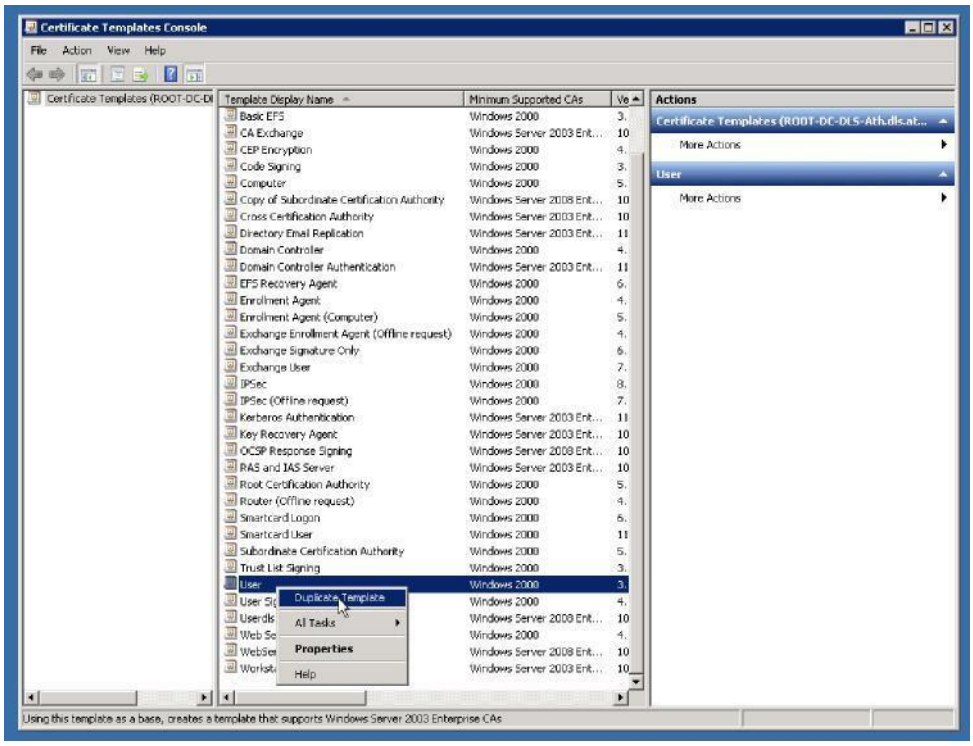


Figure 39 Duplicate User certificate template

Figure 40 Select Windows Server version

In the "General" tab of the new template, insert desirable template name and validity period (figure 41).

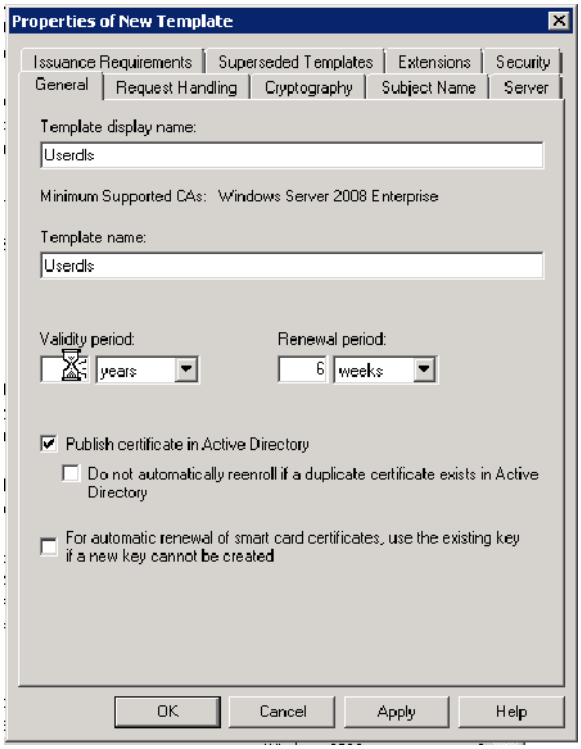


Figure 41 Copy the template with a new name

On "Subject Name" tab make sure "Supply in the request" is selected as in figure 42.

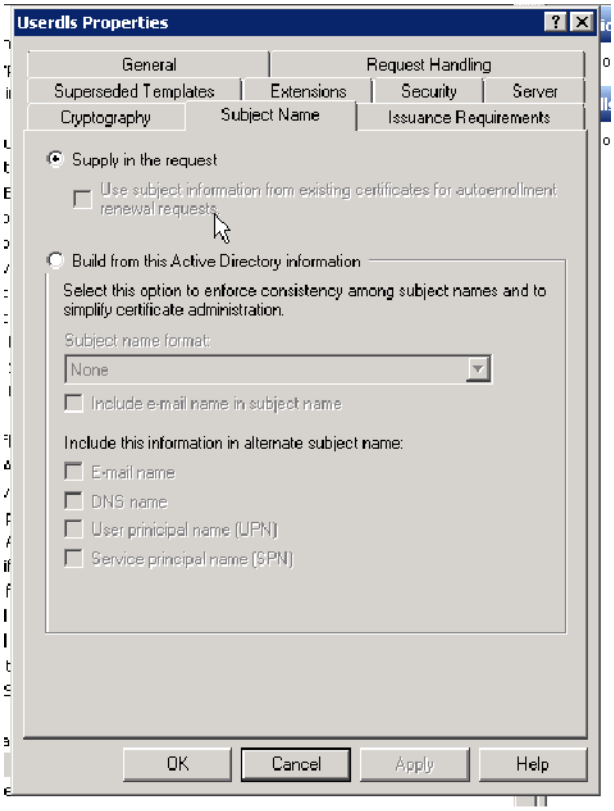


Figure 42 Template Subject Name properties

Now 'dls' user should be added in the template users. Select tab 'Security' and press button 'Add' (figure 43). Enter 'dls' object name in the 'Select users...' window and press button 'Check Name'. The user is discovered in the domain. Select 'OK' and 'dls' user is added to the

'Group or user names' box. In order for "dls" user to be able to request and enroll certificates you need to set user permissions to allow "Read" and "Enroll" as shown in figure 44.

Note: 'dls' user may be added as an individual user or as a group of users created for certificates requests. In any way user designated to request certificates through DLS PKI connector should be (individually or as part of a user group) granted with all permissions mentioned in this guide.

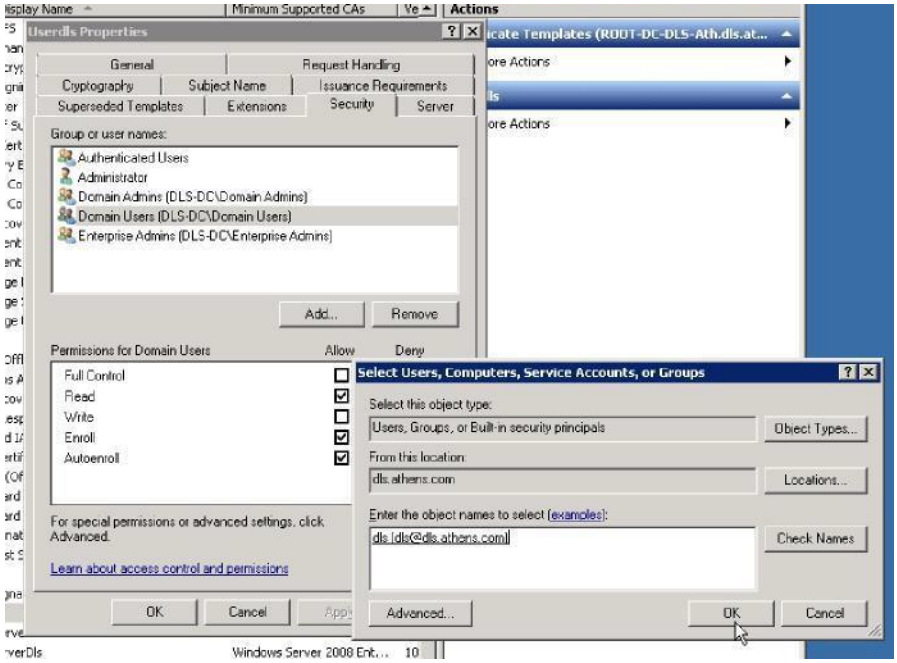


Figure 43 Add 'dls' user in template users

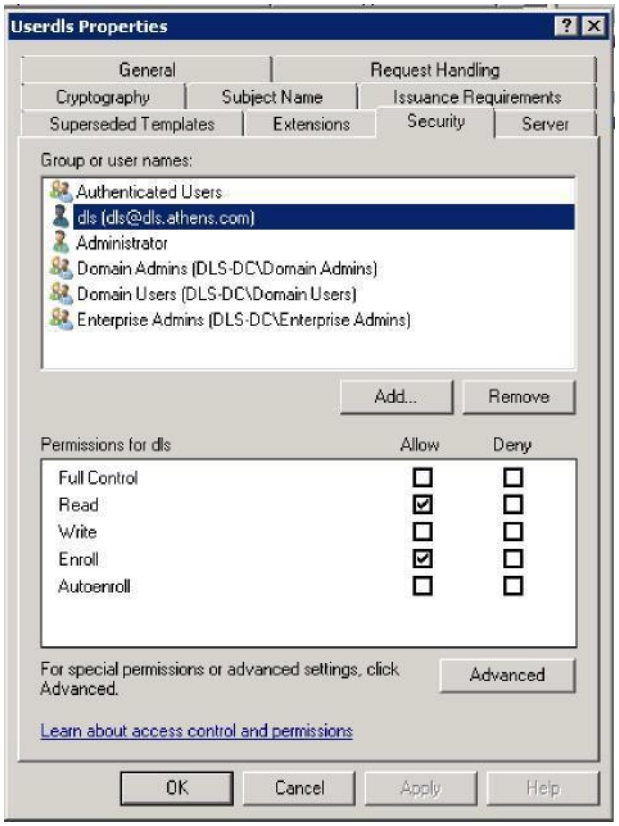


Figure 44 'dls' user added and granted permissions

Make the same modification for a Web Server template. Duplicate the template as shown in and set the 'General' and 'Subject Name' as shown in figures 45 and 46 respectively. Then add 'dls' user in the template users (figure 47). Give permissions to 'dls' user as in figure 48.

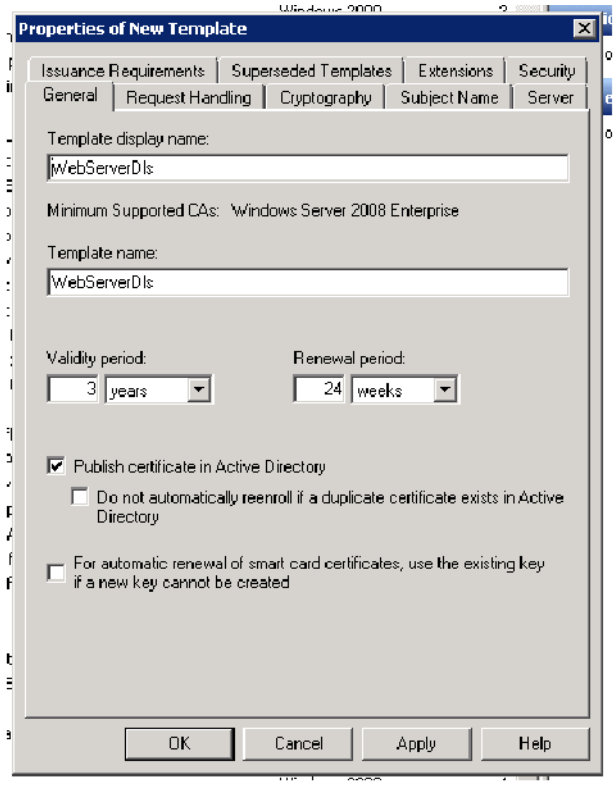


Figure 45 Template general properties

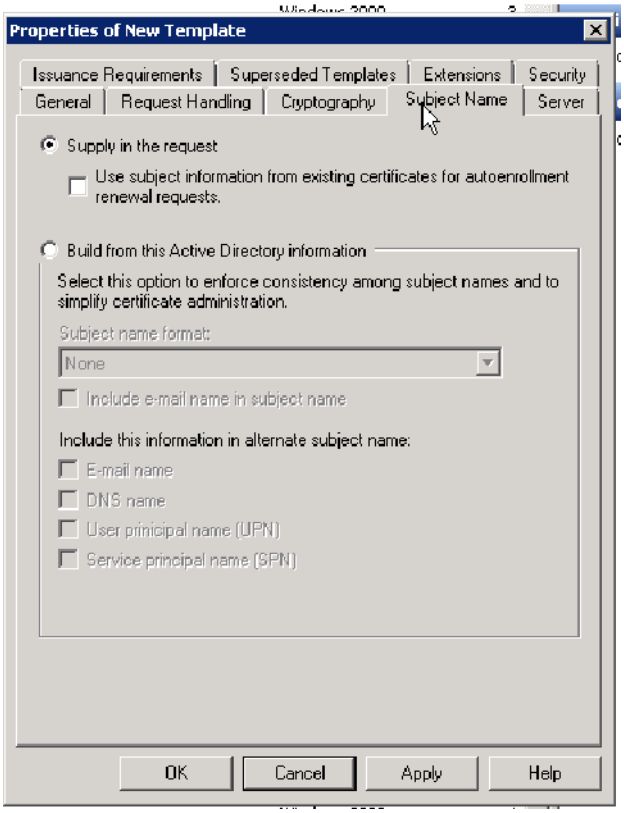


Figure 46 Template Subject Name properties

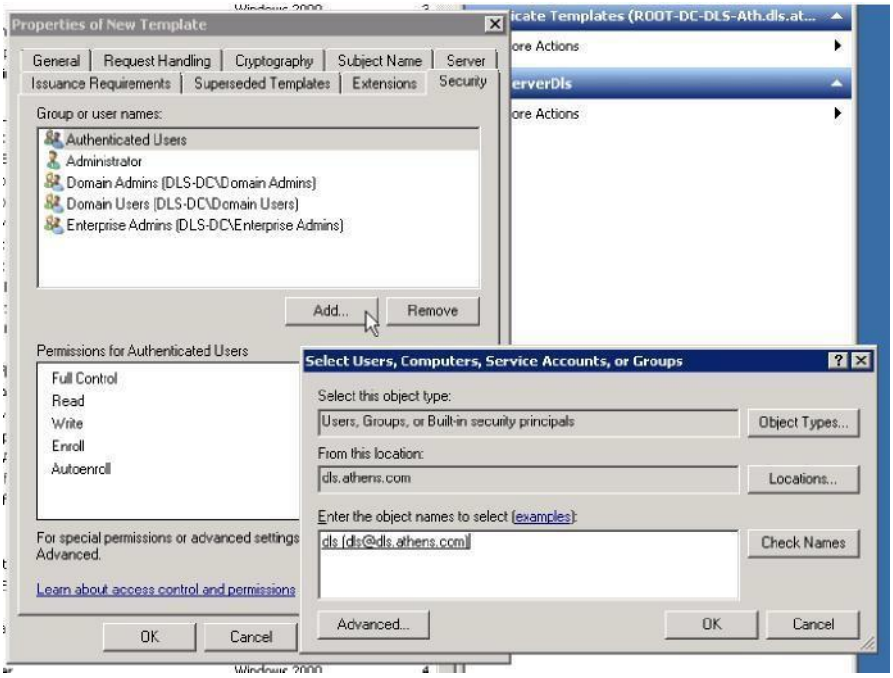


Figure 47 Add 'dls' user in template users

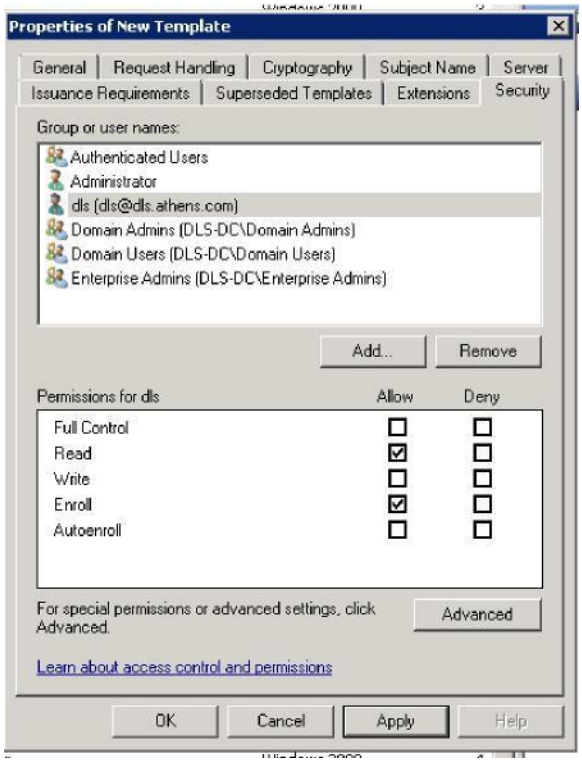


Figure 48 'dls' user added **and granted permissions**

3.4 Issuing Certificate Templates

After completing steps described in section 3.3 you will have two configured certificate templates in CA template pool. These templates though are neither issued nor enabled to your CA. That means the templates exist in the pool but cannot be used. You have to issue and enable the templates to respective issuing CA.
In 'Certification Authority' window right click to 'Certificate Templates'. Select 'New' and 'Certificate Template to Issue' as shown in figure 49.

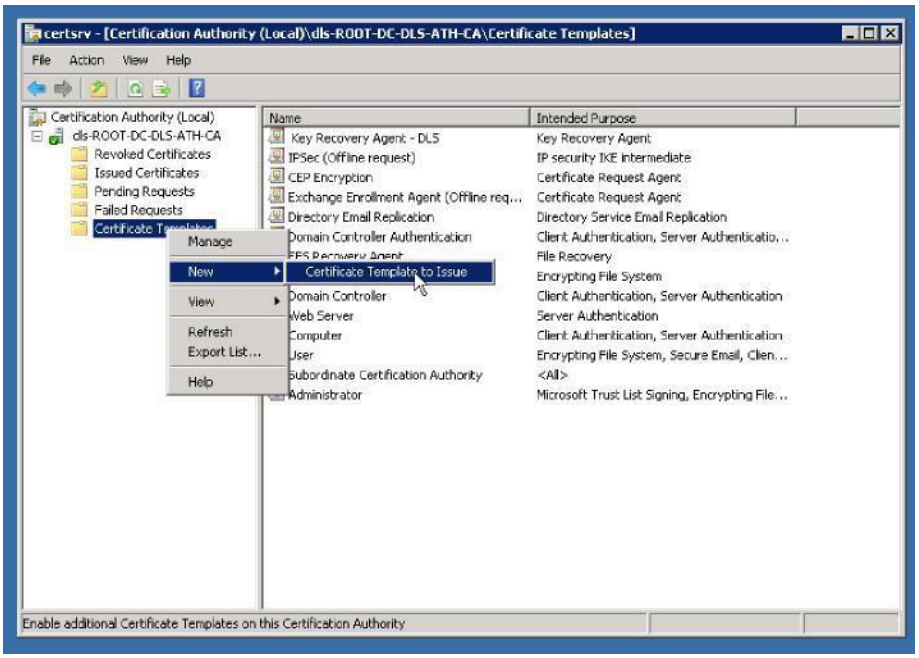


Figure 49 Issue the two templates to your specific CA

In the 'Enable Certificate Templates' window select the previously configured templates and press 'OK'. This is highlighted in figure 50.

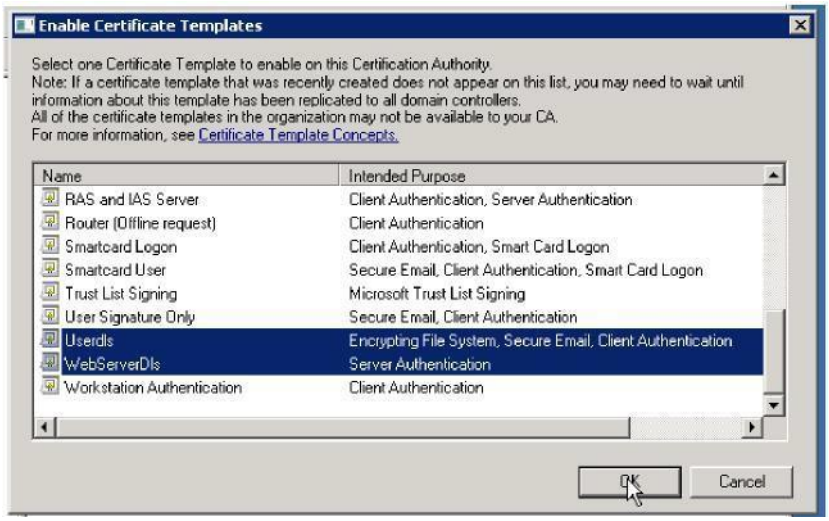


Figure 50 Enable the templates on your CA

Now the two new templates are listed under 'Certification Authority' -> 'Certificate Templates' (figure 51).

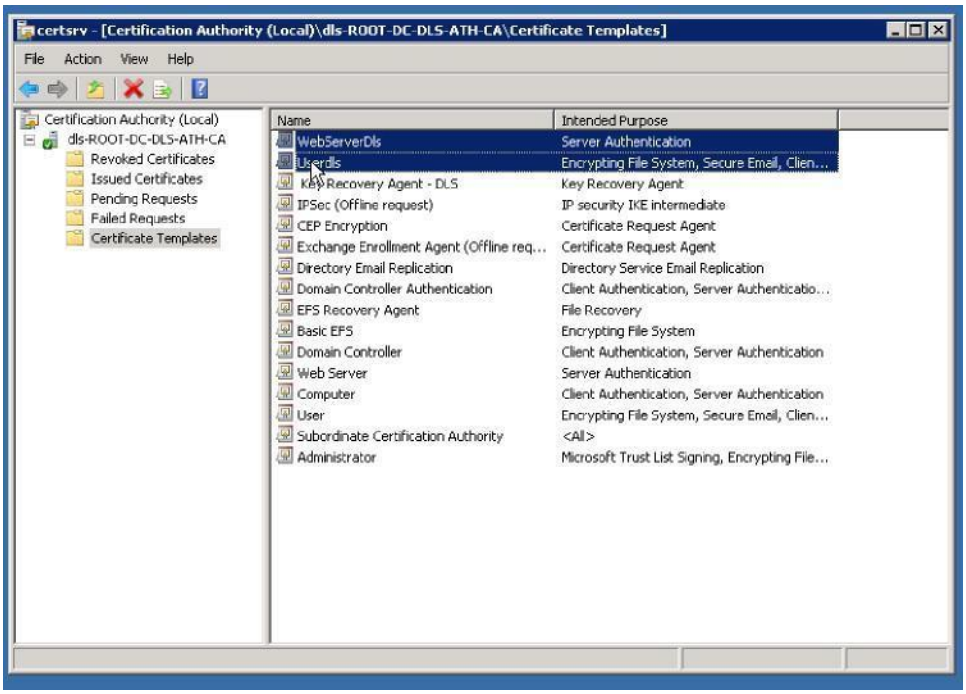


Figure 51 The two templates are now assigned to the CA

Then add 'dls' user to CA through Properties – Security tab and set user permissions to allow "Issue and Manage Certificates" and "Request Certificates" as shown in figure 52.

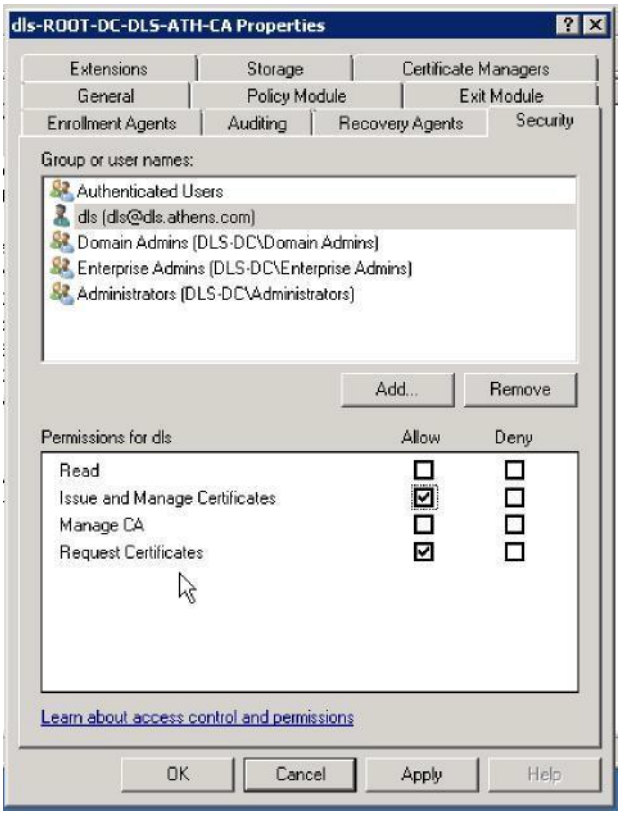


Figure 52 Root CA properties - dls user permissions

3.5 How to modify Certificate Properties

Assume there is a need to modify the validity period of the issued certificates. An example procedure is described below.
Go to the server where Certification Authority (CA) is configured.
Navigate to the Certificate Authority -> Certificate Templates and with right click select 'Manage' as in figure 53.

In the Certificate templates select the type of template according to the certificate you want to modify. E. g. select 'Web Server' template for web server certificates. If you would like to keep your current template settings too then make a copy of the template and change only the copy, not the original one.
Right-click your original template and select 'Duplicate Template' as shown in figure 54.

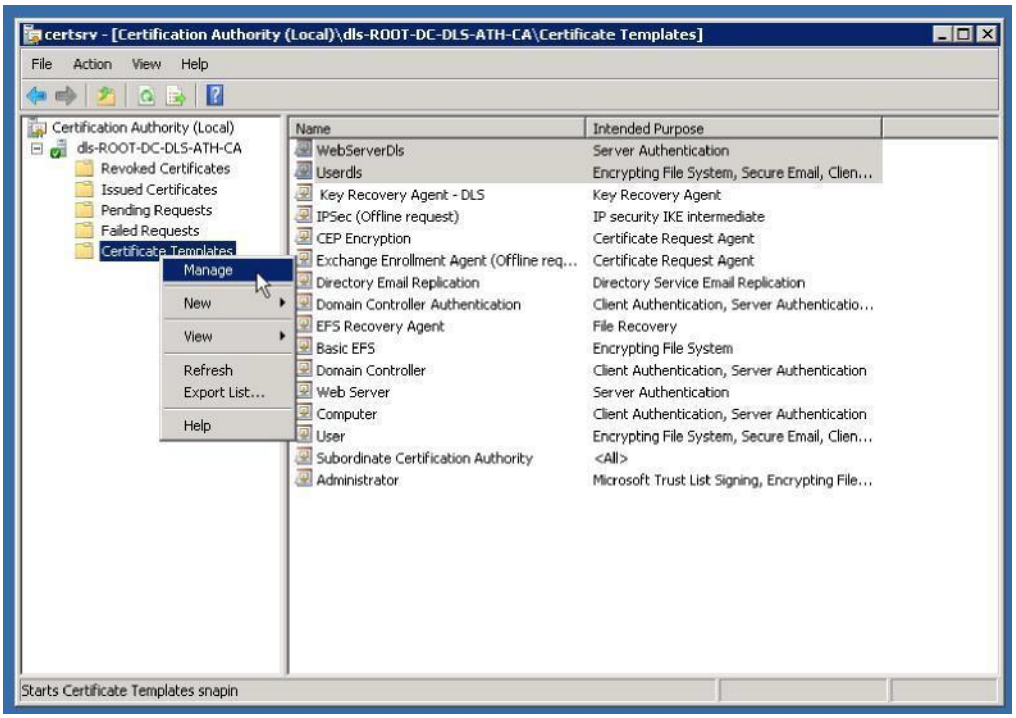


Figure 53 Manage certificate templates

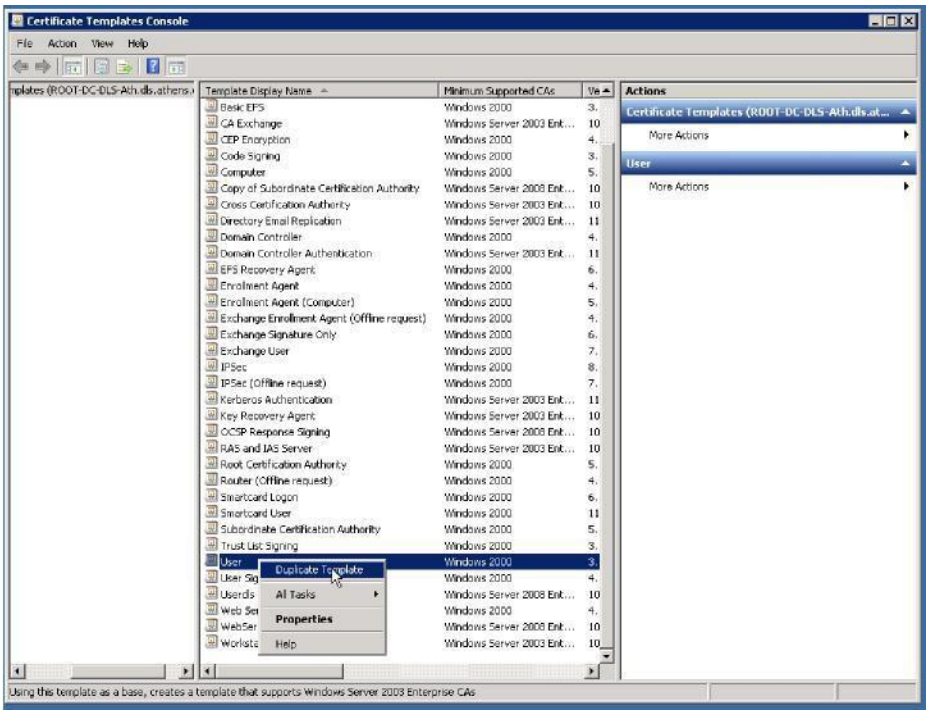


Figure 54 Duplicate template

Go to the duplicate template and with right-click select 'Properties' as shown in figure 55. Make any changes needed at the CA configuration. E.g. change the validity period of the Web server certificate as shown in figure 56.

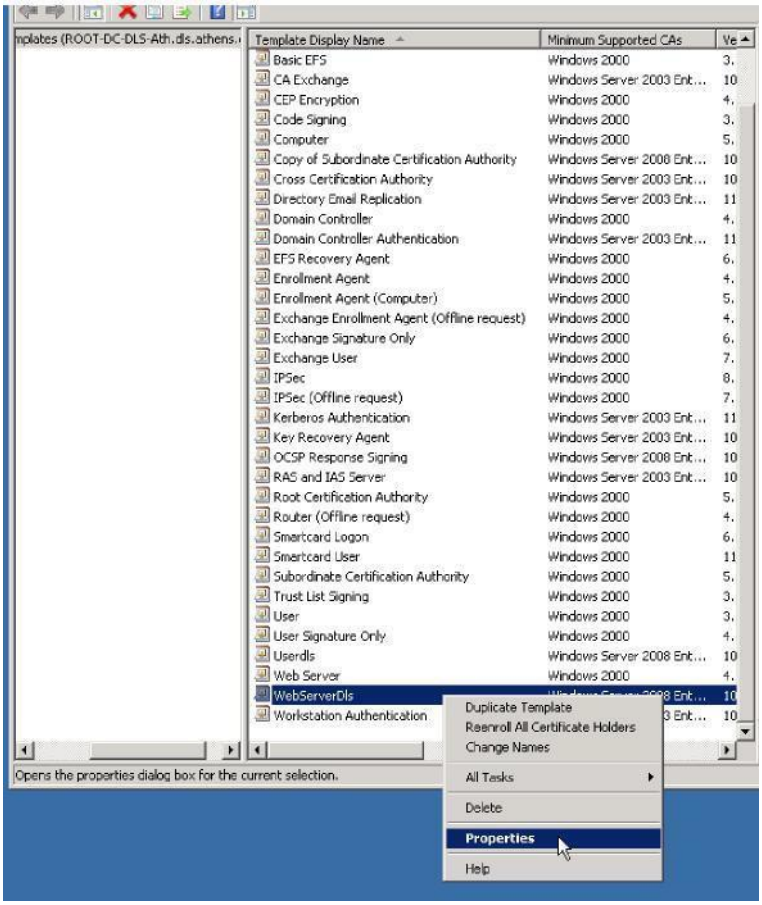


Figure 55 Change the properties of the new template

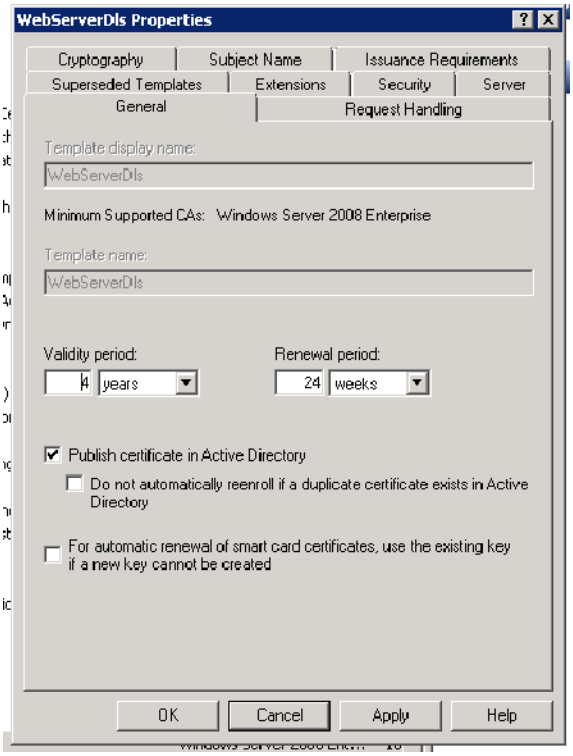


Figure 56 Change validity period (for example) 4 Microsoft Subordinate Certification Authority

Subordinate CA is a certificate authority (CA) at a level beneath the root CA. A certificate that one CA issues to another CA is referred to as a *cross-certificate.* A superior CA may issue a cross-certificate to a subordinate CA as commonly found in hierarchical trust models.

CAs that are not root CAs are considered subordinate. The first subordinate CA in a hierarchy obtains its CA certificate from the root CA. This first subordinate CA can, in turn, use this key to issue certificates that verify the integrity of another subordinate CA. These higher subordinate CAs are referred to as intermediate CAs. An intermediate CA is subordinate to a root CA, but also serves as a higher certifying authority to one or more subordinate CAs. An intermediate CA is often referred to as a policy CA because it is typically used to separate classes of certificates that can be distinguished by policy. For example, policy separation includes the level of assurance that a CA provides or the geographical location of the CA to distinguish different end-entity populations. A policy CA can be online or offline.

Note

- Most organizations use one root CA and two policy CAs — one to support internal users, the second to support external users.

The next level in the CA hierarchy usually contains the issuing CA. The issuing CA issues certificates to users and computers and is almost always online. In many CA hierarchies, the lowest level of subordinate CAs is replaced by RAs, which can act as an intermediary for a CA by authenticating the identity of a user who is applying for a certificate, initiating revocation requests, and assisting in key recovery. Unlike a CA, however, an RA does not issue certificates or CRLs; it merely processes transactions on behalf of the CA.

4.1 Subordinate CA Server Configuration

Put the server in the domain in the same way as described in chapter 3, figures 16 to 19. Log the server in the domain. Ensure that the server has joined the domain by looking at the system properties (figure 57).

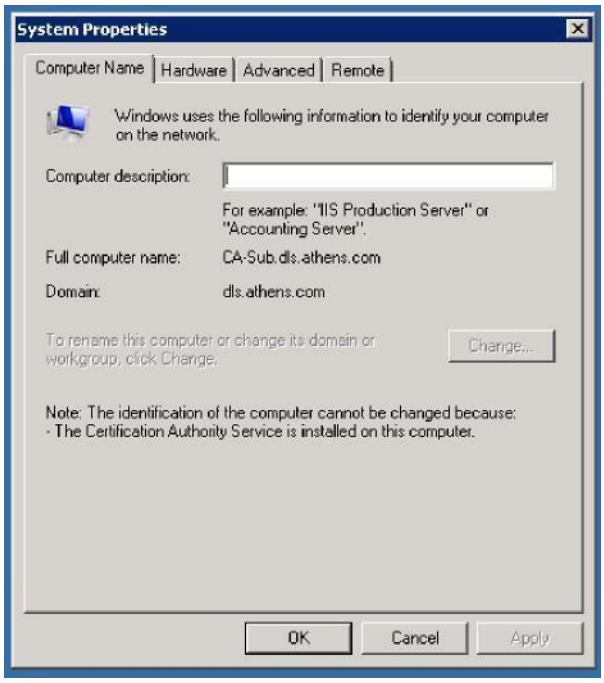


Figure 57 Ensure the server have joined the domain

4.2 Install CA and other Application services

Install Application Server Windows components that may be missing (figure 58).

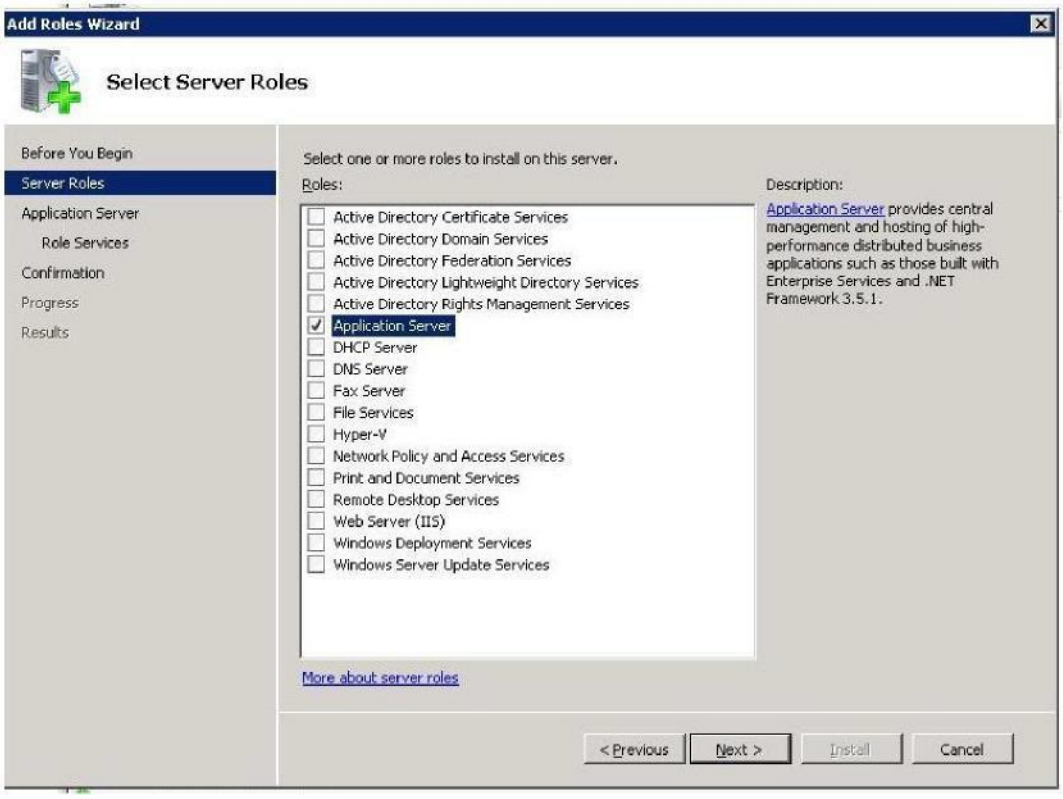


Figure 58 Install Application Server if missing

Select to install NET Framework 3.5.1 as in figure 59. It is also necessary to install Certificate Services package (figure 60). During that installation choose to install an 'Enterprise' and 'Subordinate CA' as illustrated in figures 61 & 62.

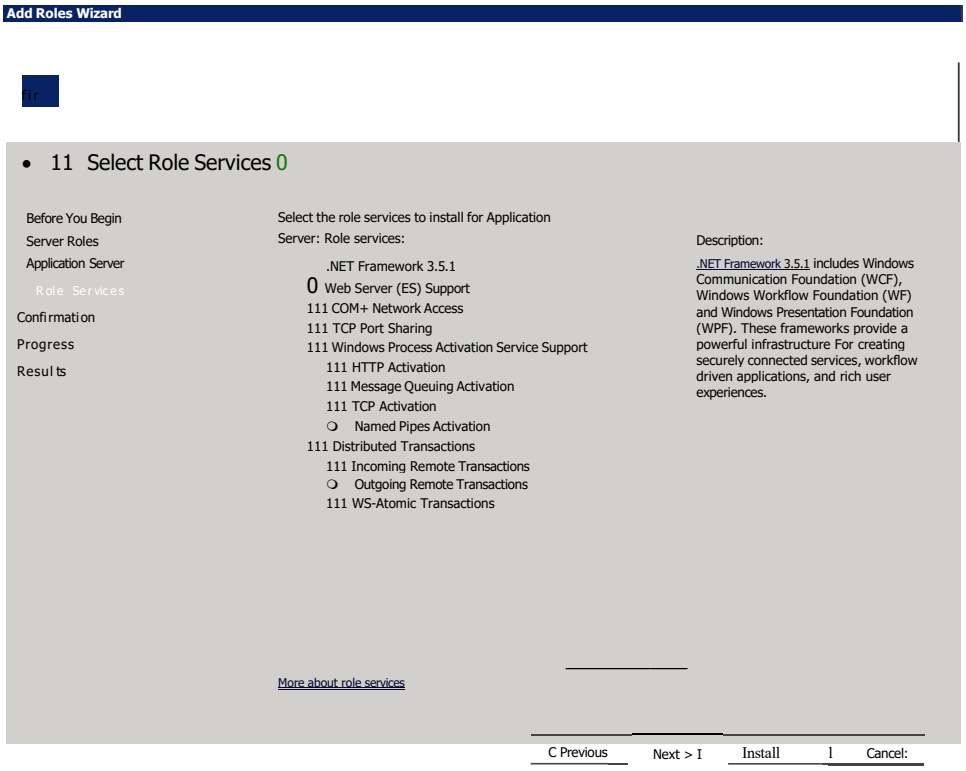
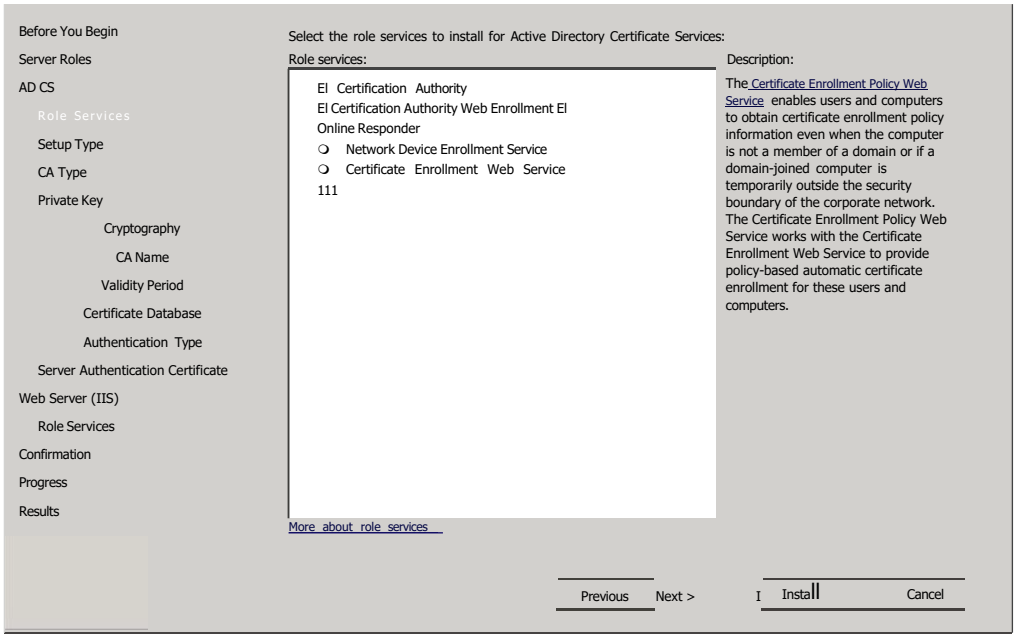


Figure 59 Install necessary WIN applications if missing

Figure 60 Install Certificate services



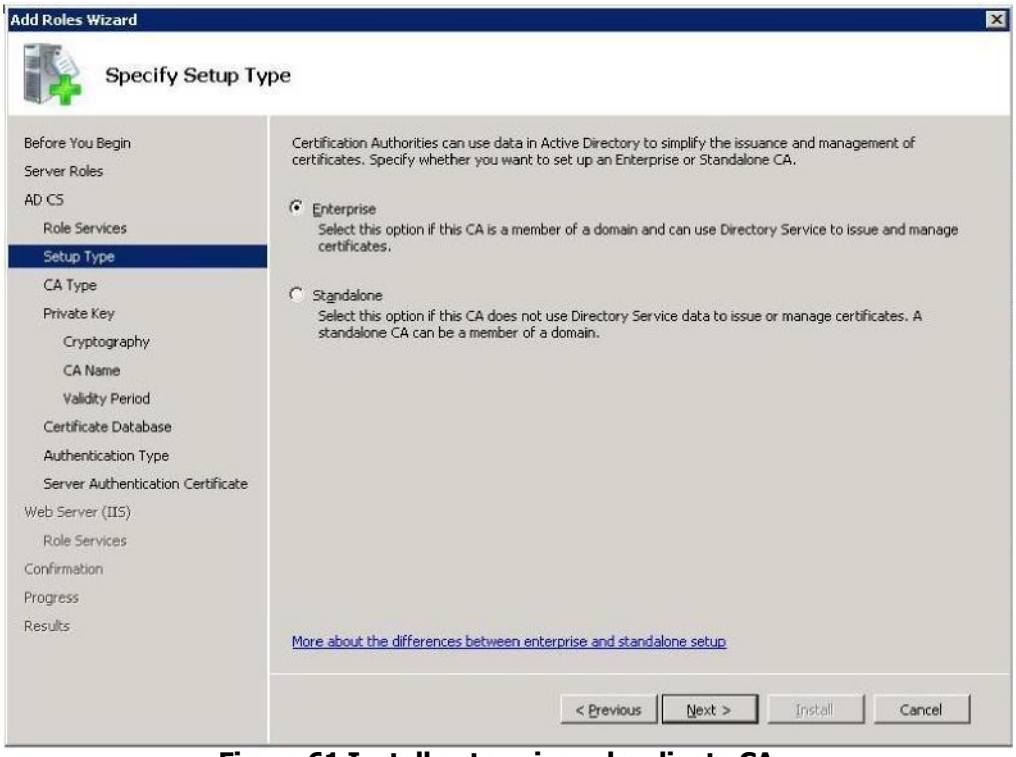


Figure 61 Install enterprise subordinate CA

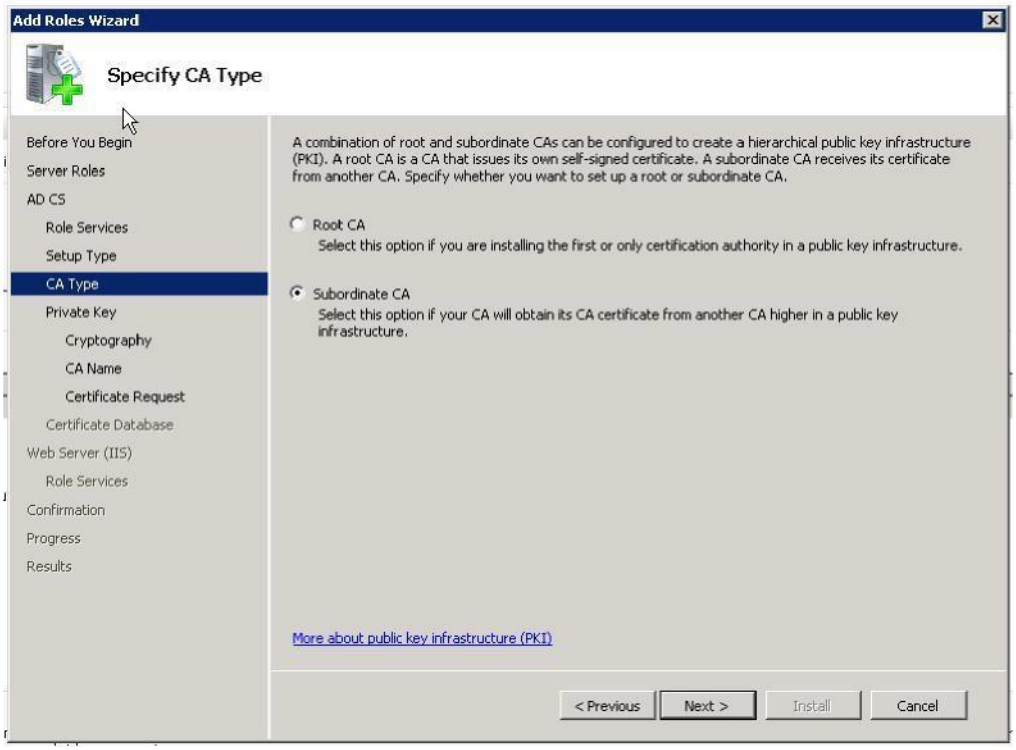


Figure 62 Install enterprise subordinate CA

Select whether you want to use a pre-existing private key or create new (figure 63) and specify the cryptographic service provider, hash algorithm and key length as shown in figure 64.

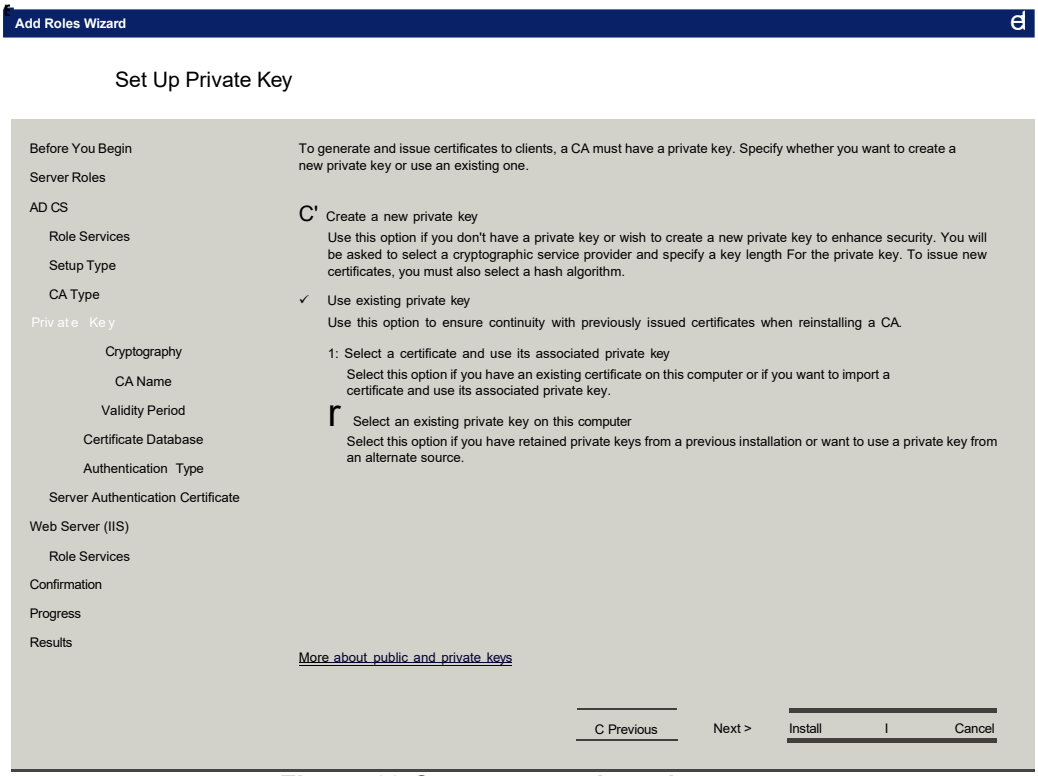


Figure 63 Create new private key

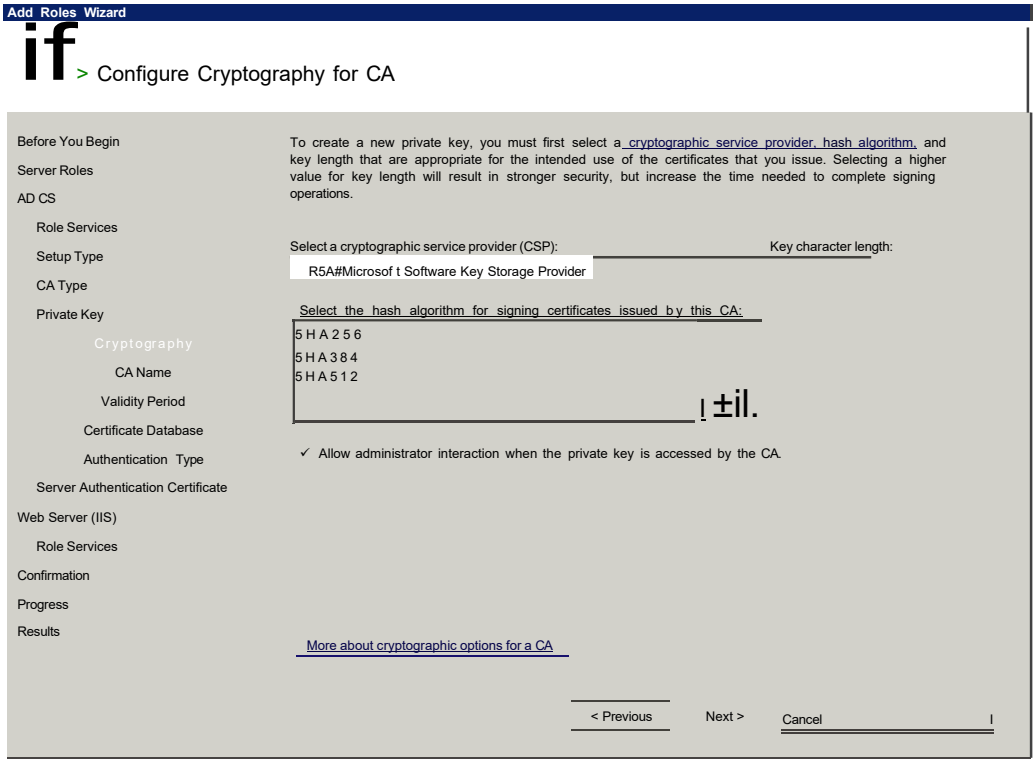


Figure 64 Cryptographic service provider, hash algorithm and key length

Give a CA common name (CN) and DN suffix (figure 65).

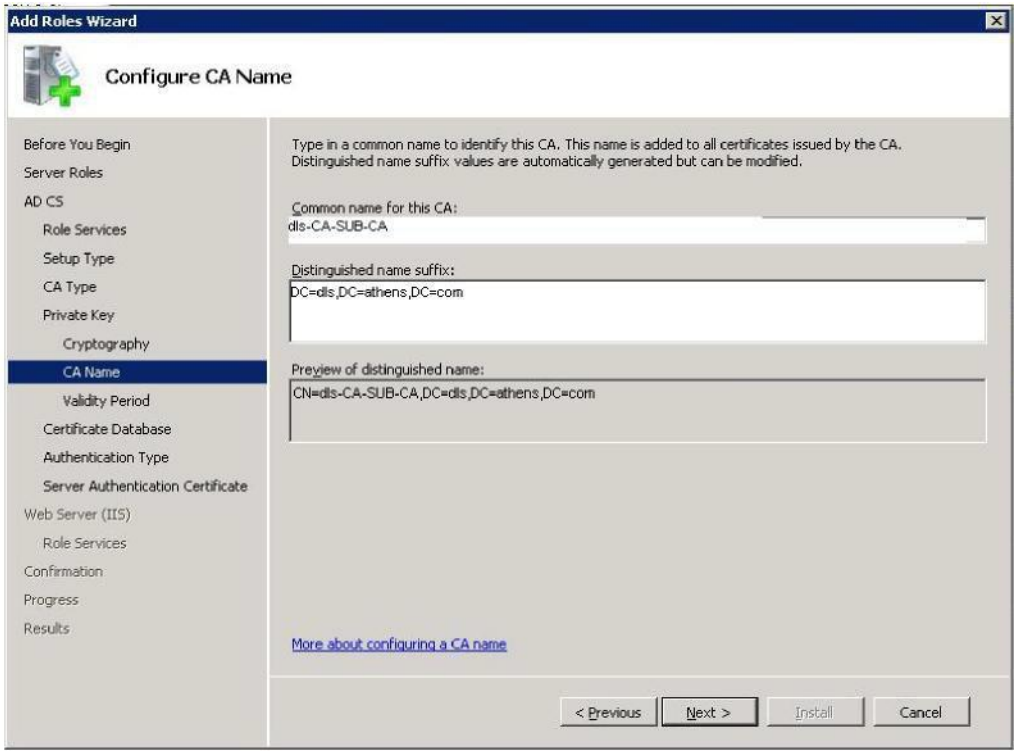


Figure 65 Configure CA common name and DN suffix

Define Root CA that will be used to request Certificate for this subordinate CA. (figure 66)

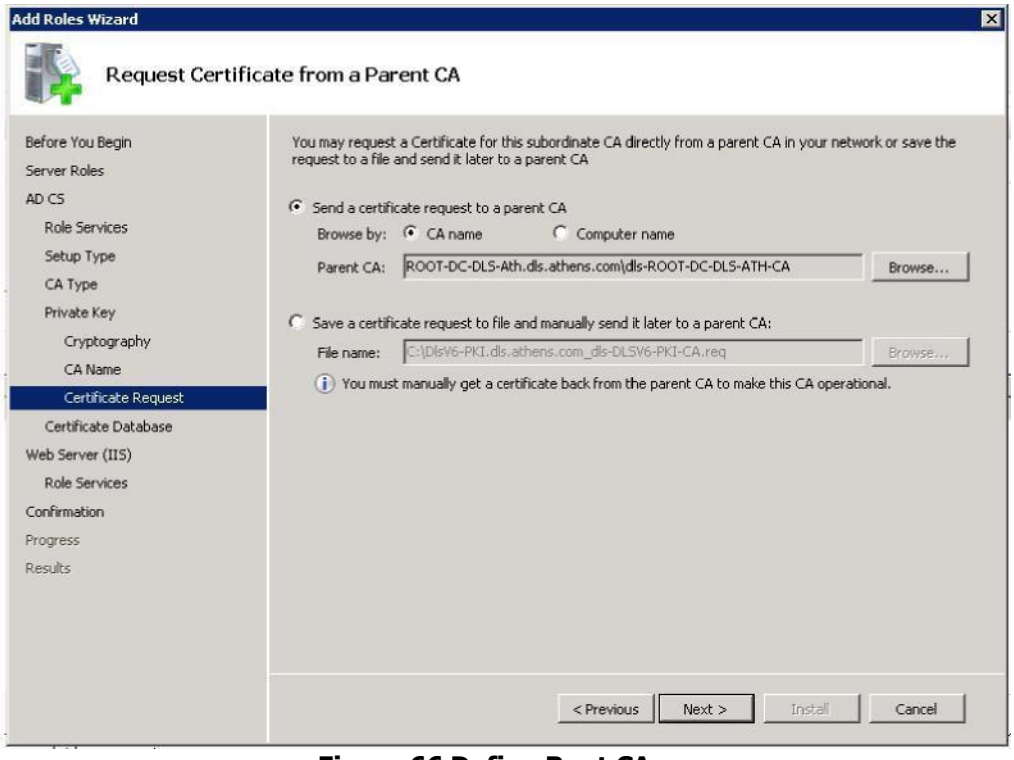


Figure 66 Define Root CA

Specify validity period for the certificate that will be generated for your Root CA as shown in figure 67.

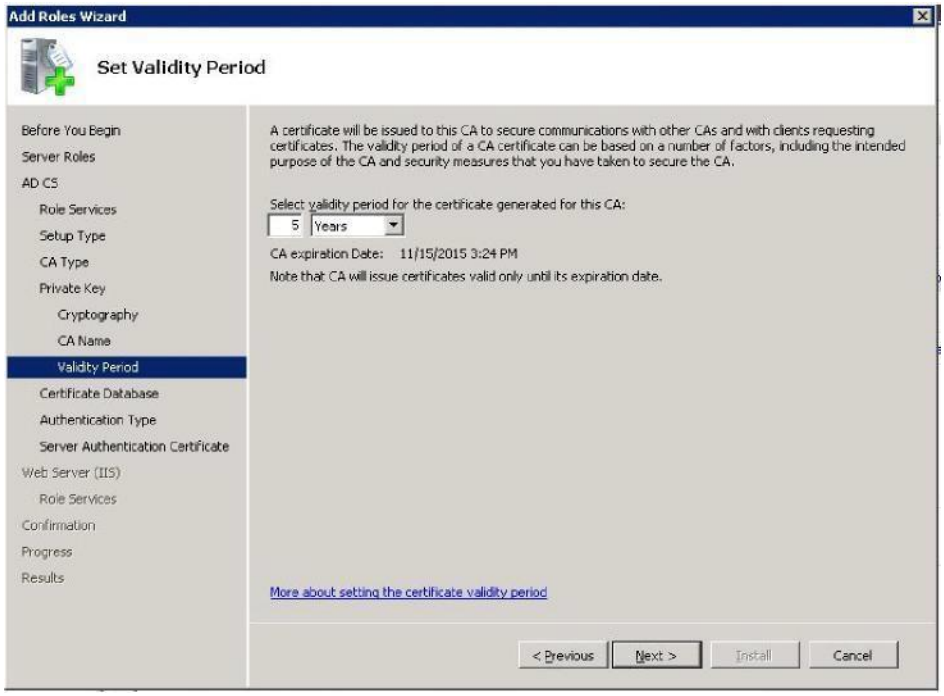


Figure 67 CA certificate validity period

Finally, specify Certificate database and Certificate database log location (figure 68), the type of authentication clients will use (figure 69) and a server authentication certificate suitable for ssl encryption (figure 70).

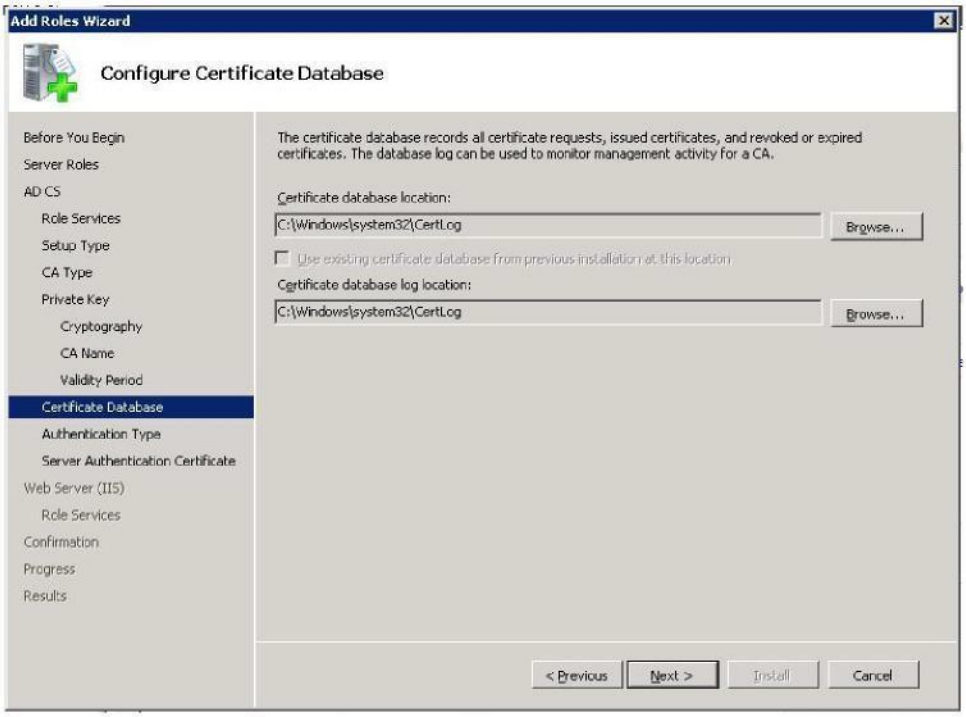


Figure 68 Certificate database and database log location

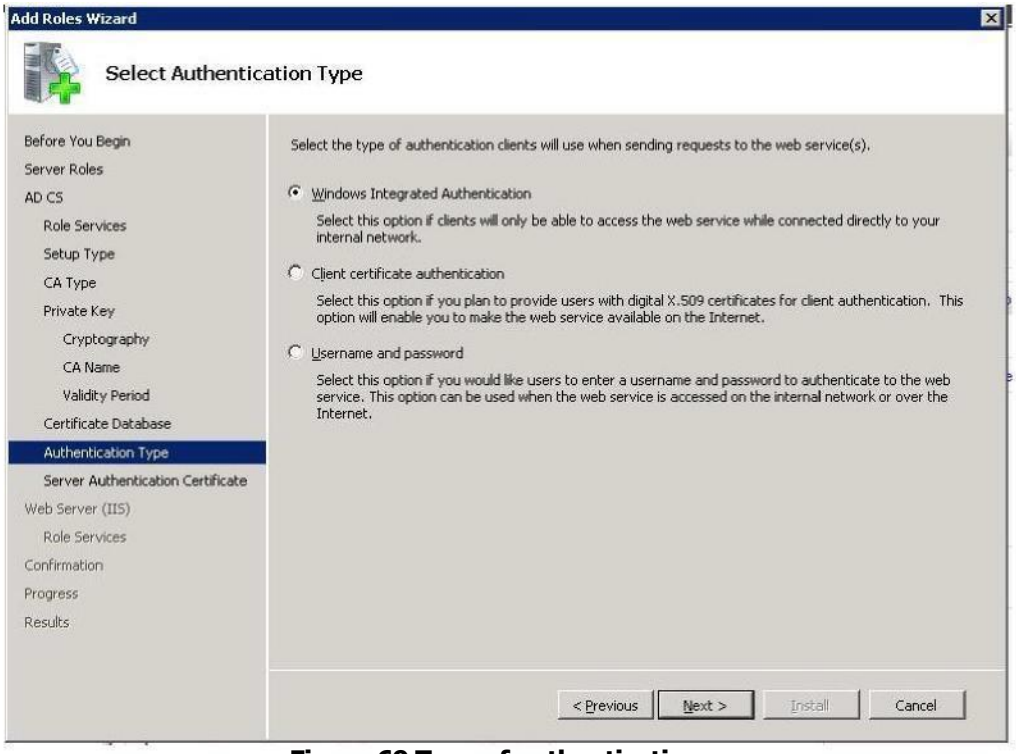


Figure 69 Type of authentication

Figure 70 Certificate for SSL Encryption

Finishing the installation, system will prompt you with installation wizard for Web Server (IIS) Role services (needed for Certication Authority Web Enrollment). Proceed with default values as shown in figure 71.

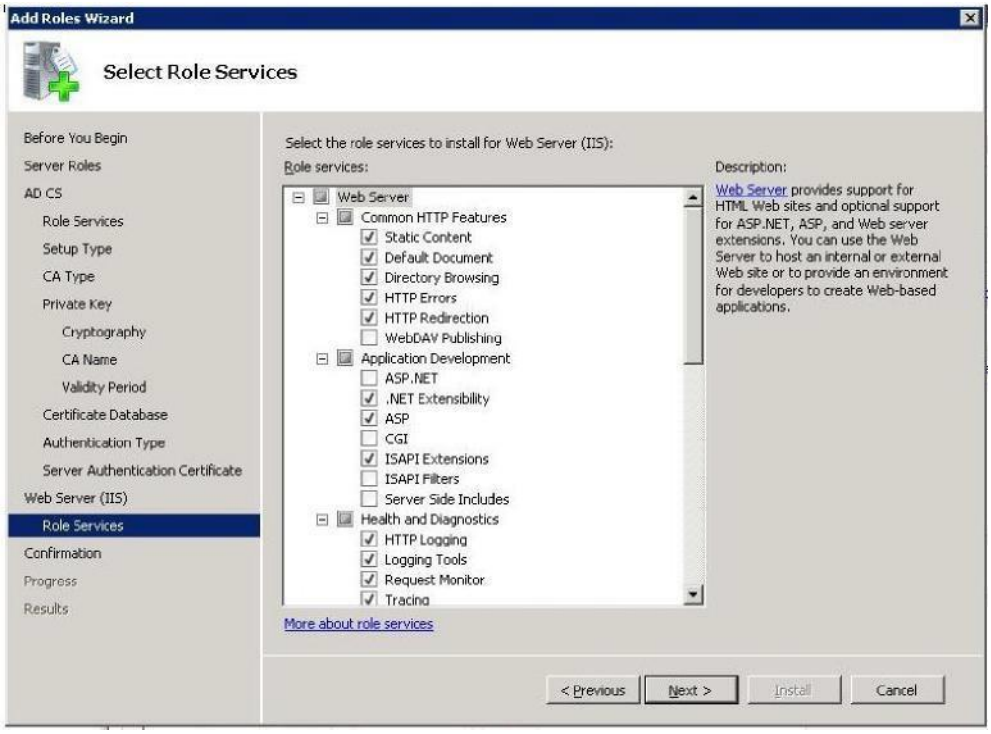


Figure 71 Installation wizard for Web Server (IIS) Role services

Proceed with "Install" and wait for the installation to finish. A success message will be displayed with the end of the installation (figure 72)

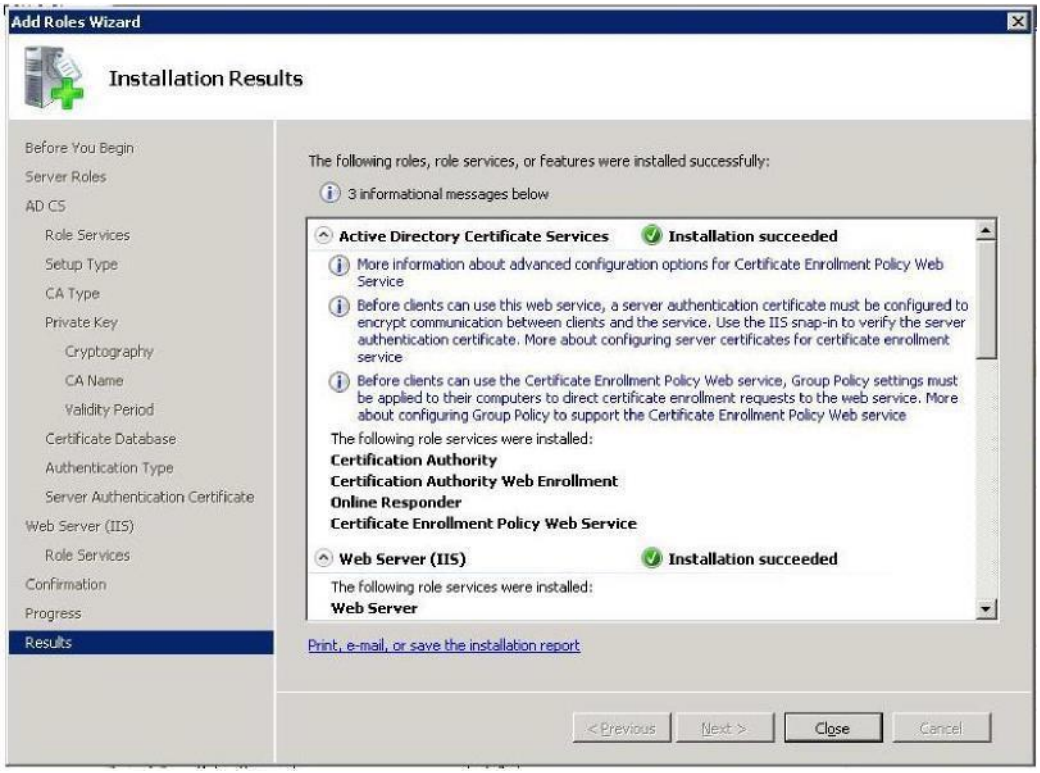


Figure 72 End of installation

4.3 Sub CA certificate templates configuration

Open subordinate CA management window as in figure 73. Right click your sub CA and edit its properties. In 'Policy Module' tab set a template-wise request handling as shown in figure 74.

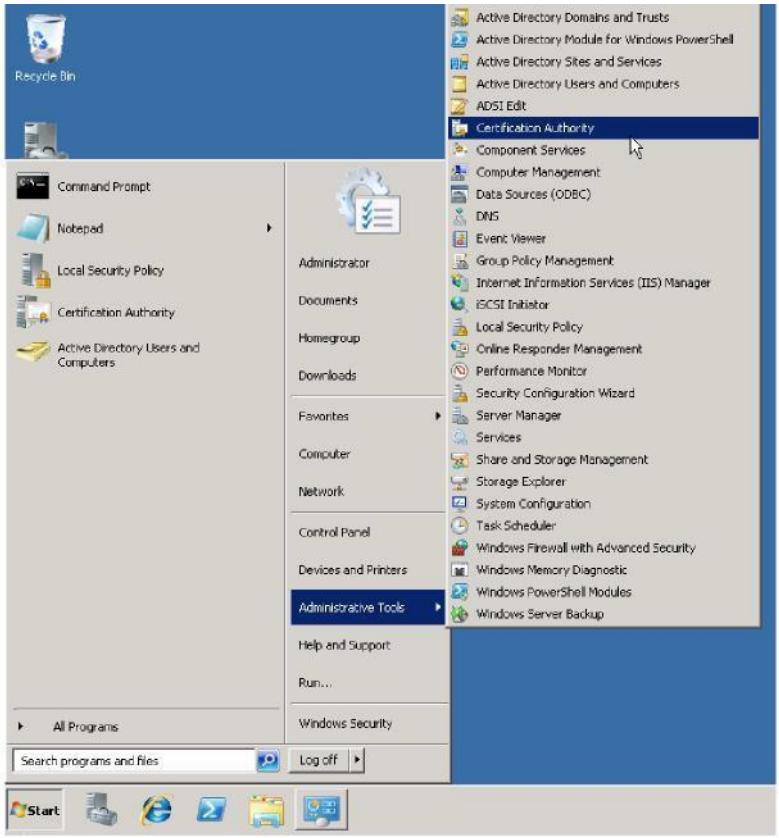


Figure 73 Open CA management window

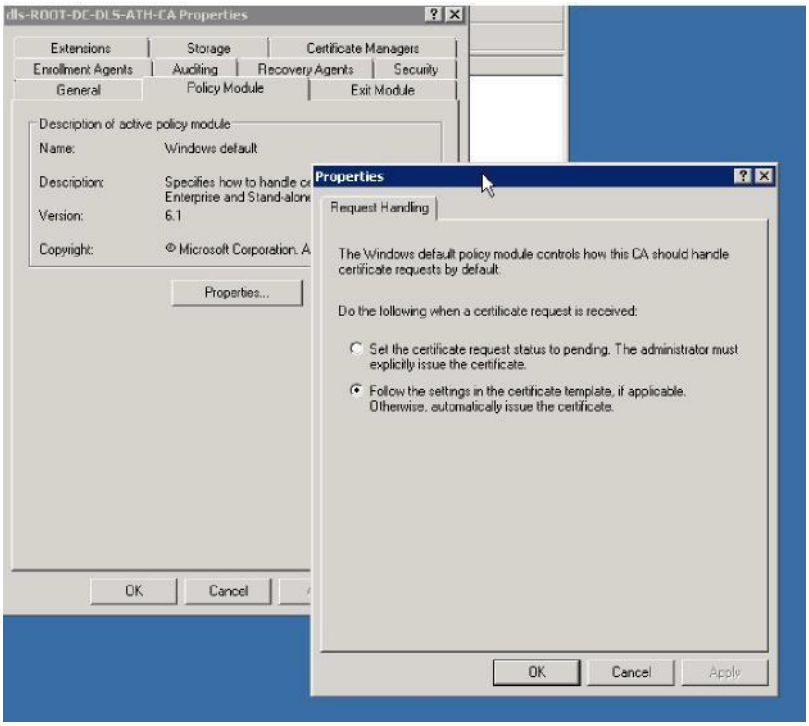


Figure 74 CA properties

Two templates must be configured for client and server certificate request handling. PKI DLS will send to CA his requests based on these templates. A user and a Web Server template are needed. Firstly duplicate the existing 'User' template as in figure 75. Set the general properties as in figure 76.

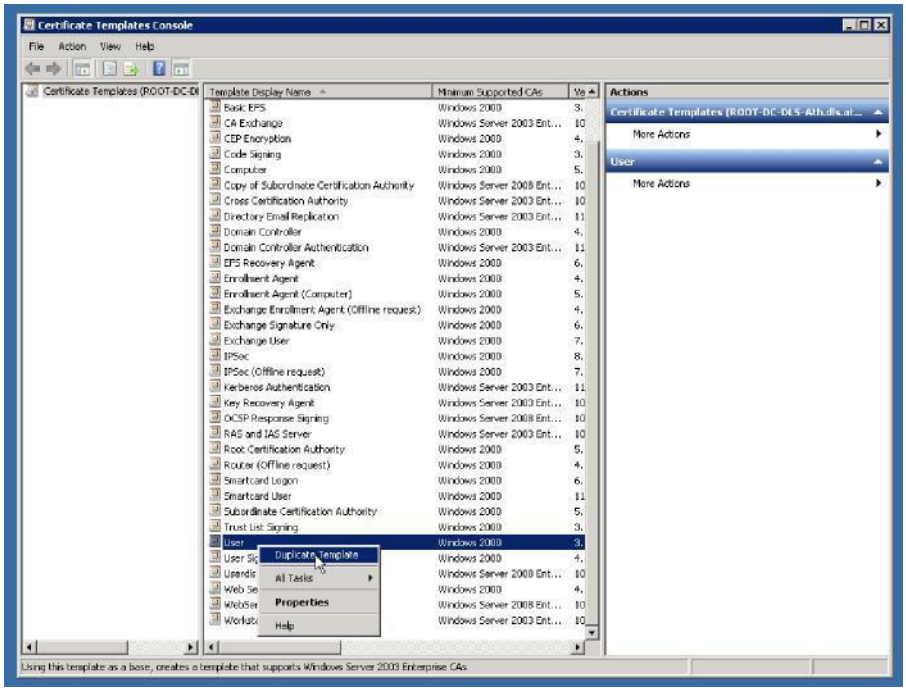


Figure 75 Duplicate user template

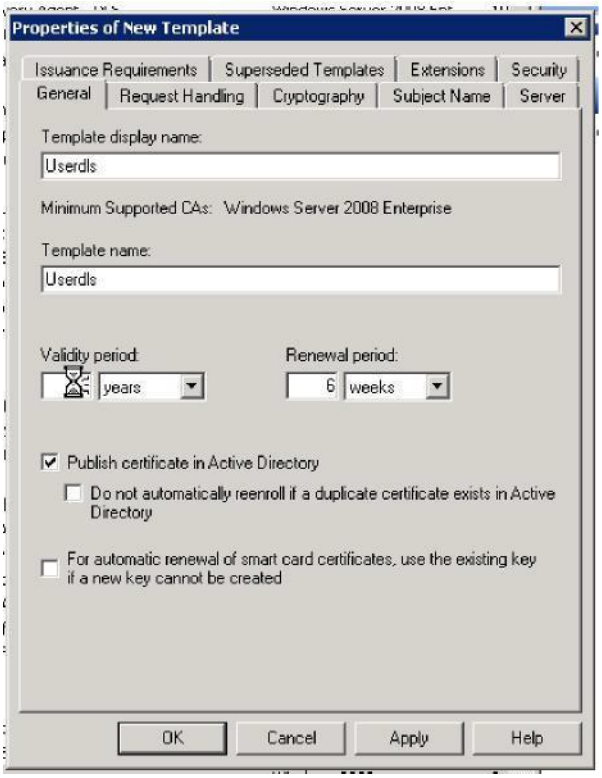


Figure 76 User template general properties (default)

Set the Subject Name properties of the new template as in figure 77. In the security tab add 'dls' user (figure 78).

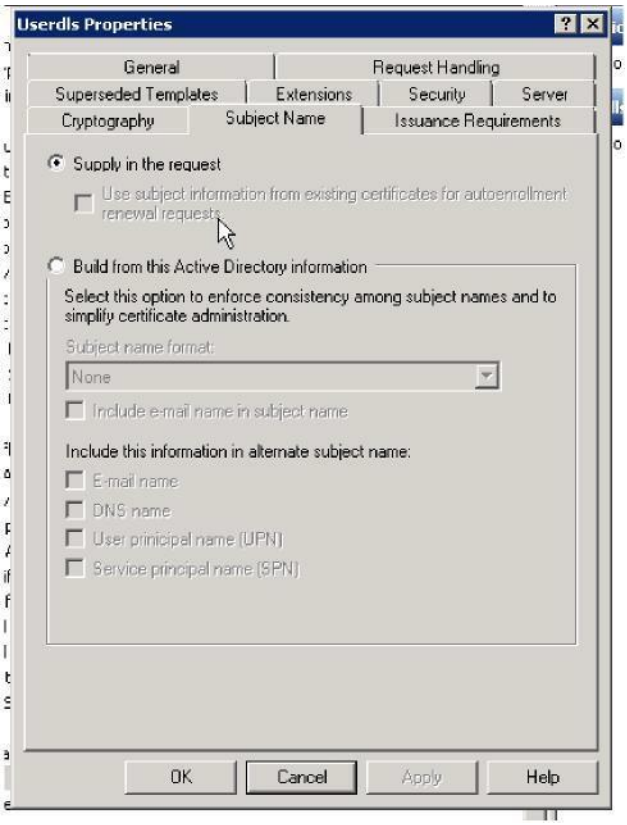


Figure 77 User template Subject Name properties (default)

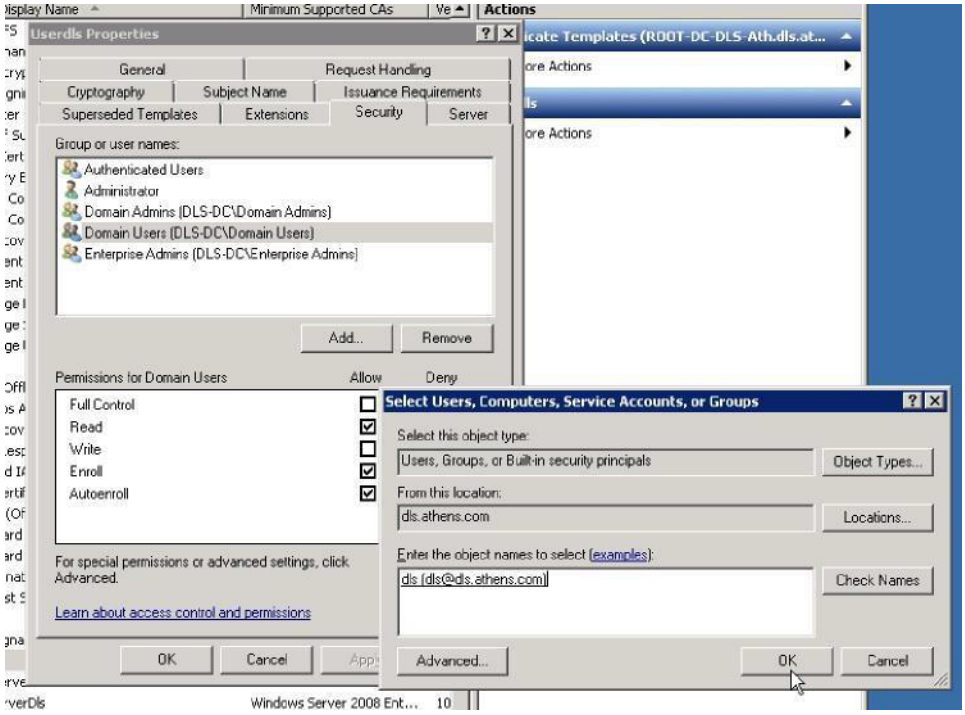


Figure 78 Enter 'dls' in template users

Give permissions to 'dls' user (figure 79). Then duplicate 'Web Server' template and set its properties as in figures 80 and 81.

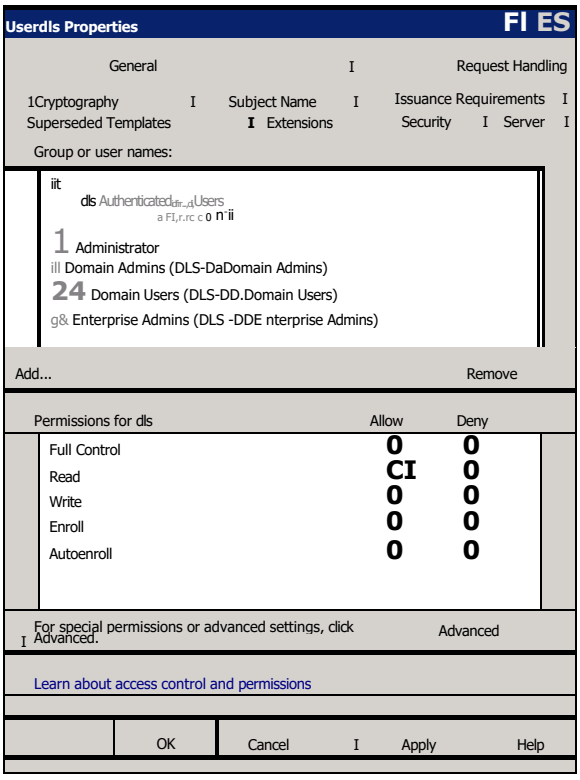


Figure 79'dls' user permissions

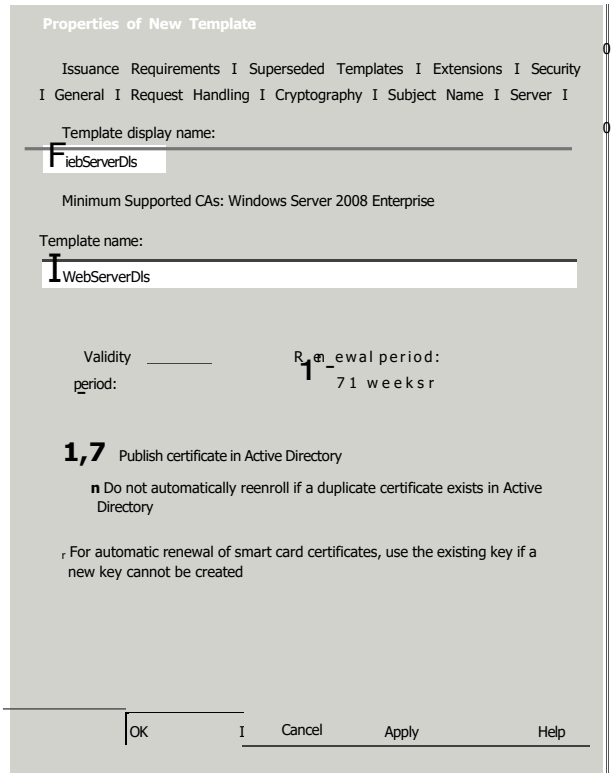


Figure 80 Web user template general properties

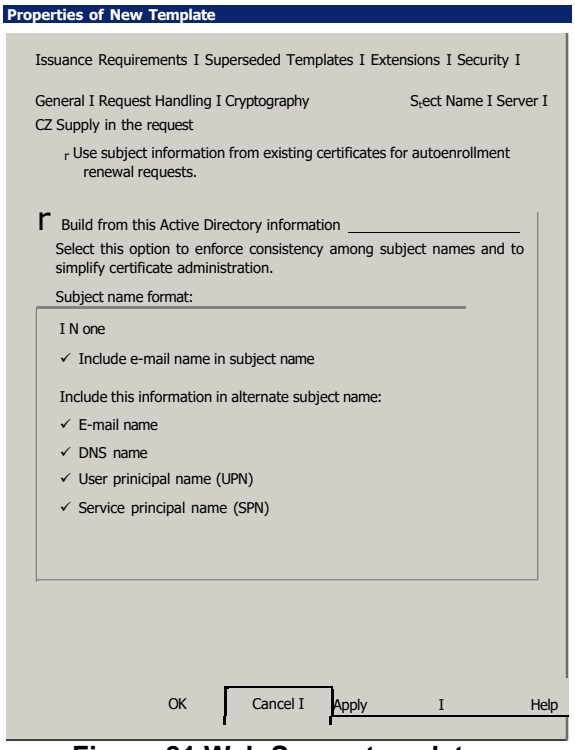


Figure 81 Web Server template

Next to that add 'dls' user to the template users (figure 82). Then set 'Dls' user permissions as in figure 83.

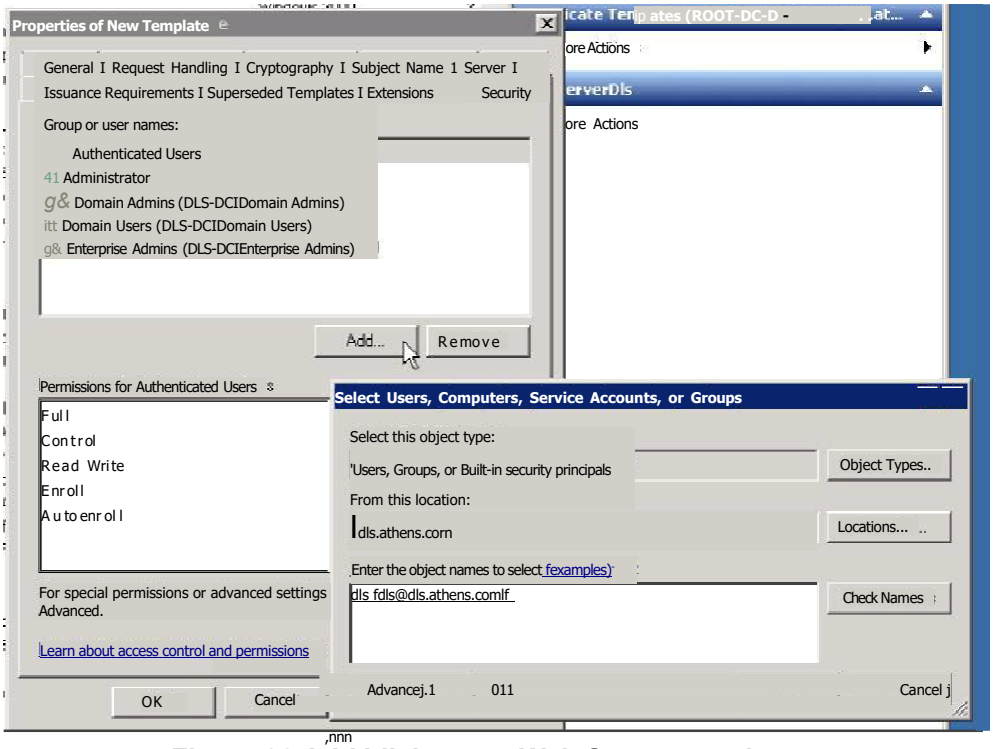


Figure 82 Add Idlsiuser to Web Server template

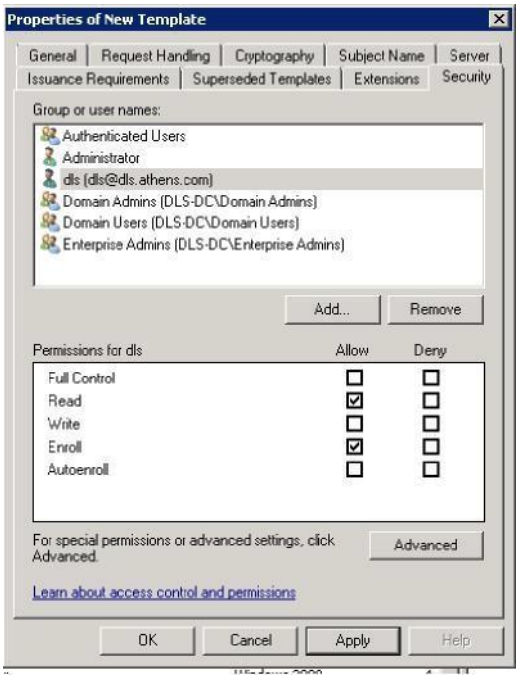


Figure 83 'dls' user permissions

Note: For infrastructures that a Root CA (Enterprise CA only) is configured and certificate templates designated to use with DLS are already configured as described in section 3.3 on Root CA, then configuration of certificate templates on Sub CA is not required. (Templates are already created and configured with respective values and user permissions on Root CA)

4.4 Issuing Certificate Templates

Issue the two templates in your specific sub CA in the same way as that was done for root CA (section 3.4). The templates are issued and enabled for the specific subordinate CA as highlighted in figures 84 and 85.

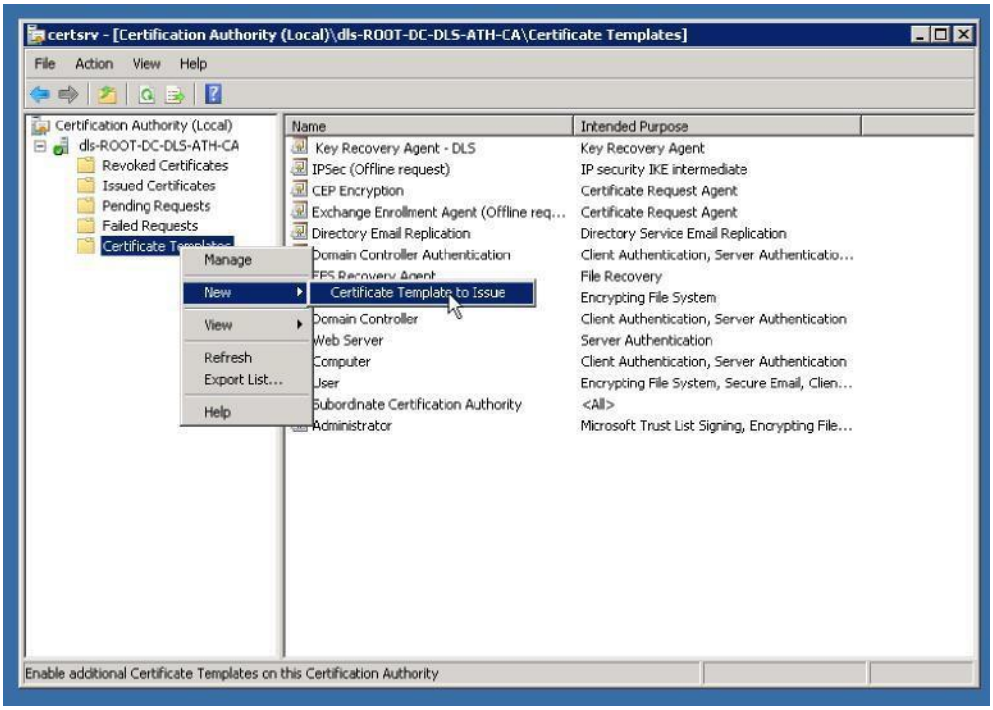


Figure 84 Issue the two templates to your specific subCA

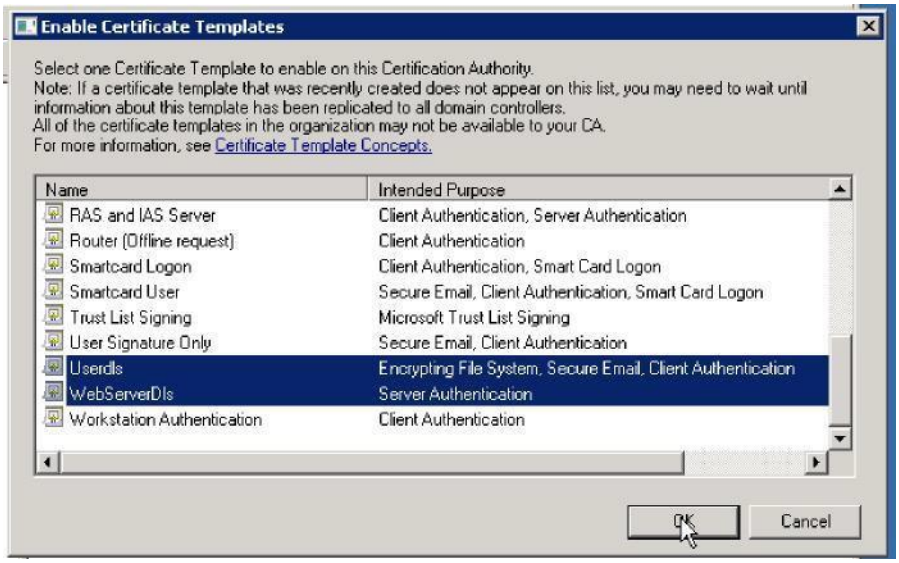


Figure 8**5** Enable the two templates on your subCA

Then add dls users to Sub CA authenticated users and set user permissions to allow "Issue and Manage Certificates" and "Request Certificates" as shown in figure below.

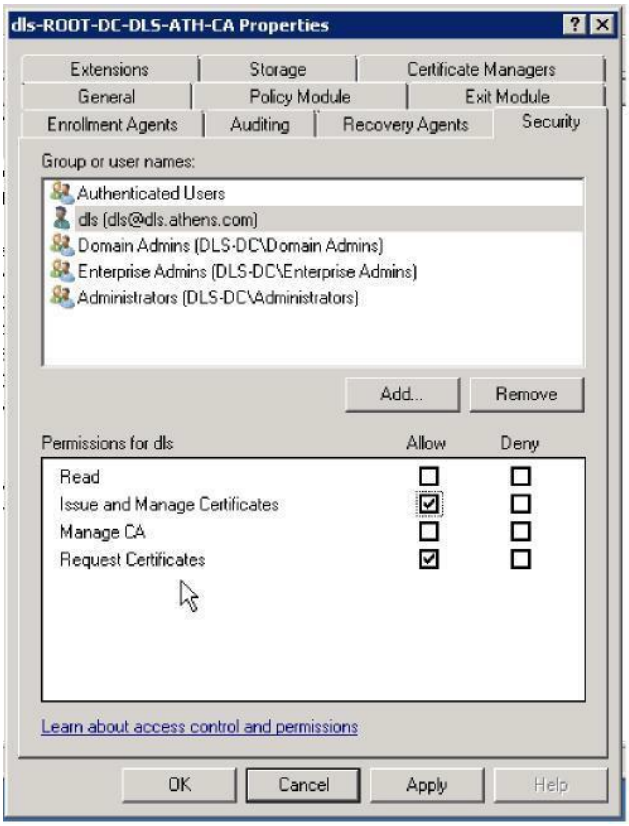


Figure 86 Root CA and dls user properties

5 CA backup and restore

In case a CA server experiences outages or failures due to hardware failures it is recommended to backup the CA certificate in order to avoid overheads and delays. As soon as the repaired or new CA server is operative its CA certificate can be restored. In that way the certificates already issued to the devices will be still valid. In the Certificate Authority window right click and open All Tasks menu as shown in figure 82. Select 'backup CA'.

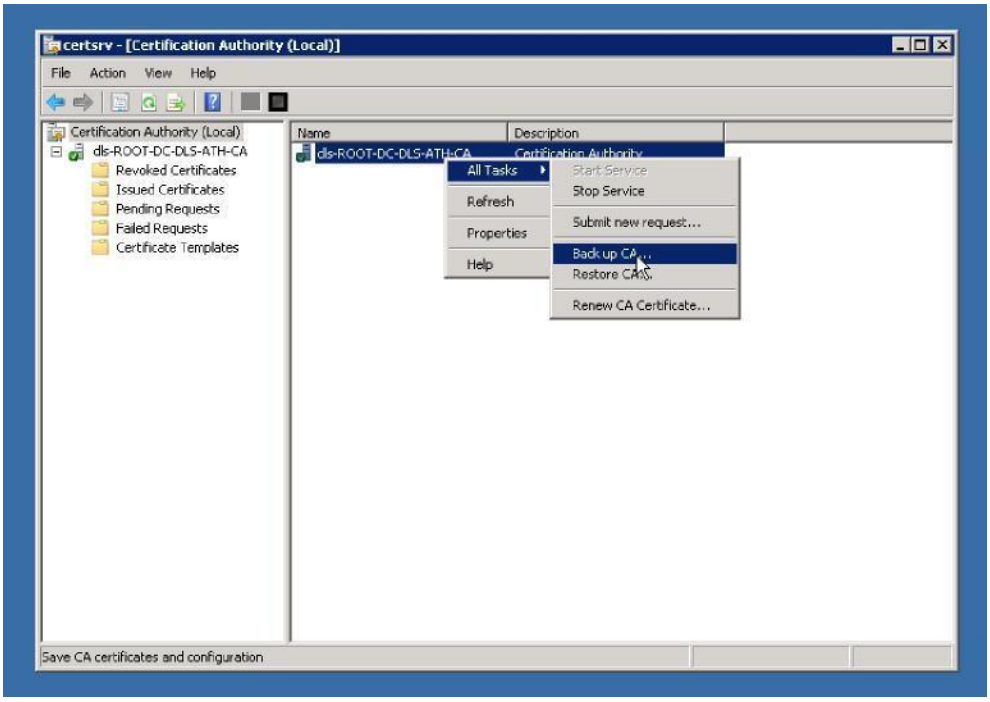


Figure 87 Right click to backup CA

A backup wizard starts (figure 88). Browse for an empty backup folder (figure 89).

Figure 88 Backup wizard starts

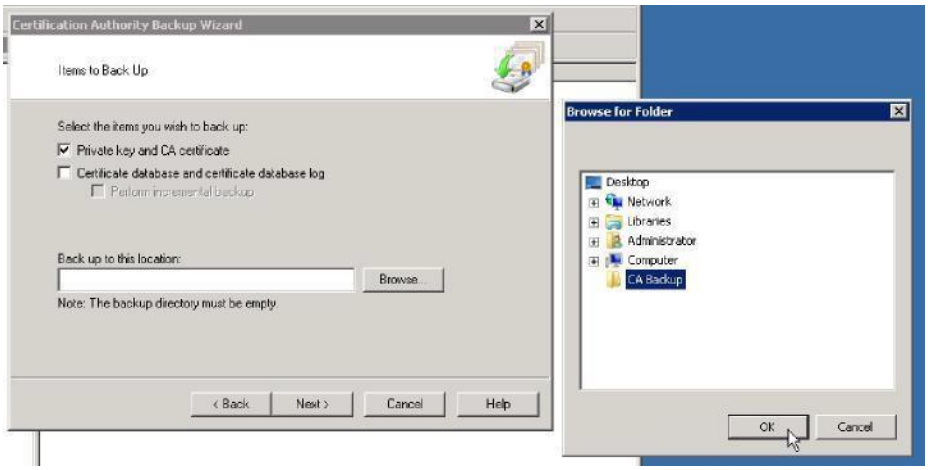


Figure 89 Select backup directory

Set password needed for access to the private key and the CA certificate file (figure 90)

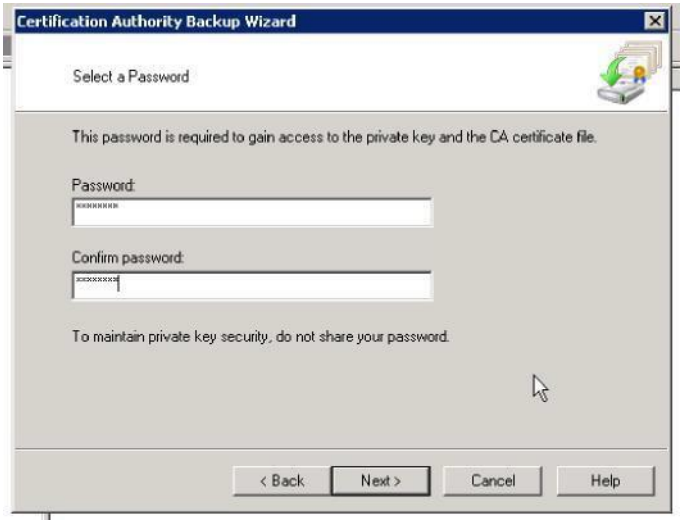


Figure 90 Password for access to the private key and the CA certificate file

Backup wizard is completed (figure 91).To restore the CA select the 'Restore' option in the menu shown in figure 92. The procedure is similar to the backup procedure.

Figure 91 Backup completed

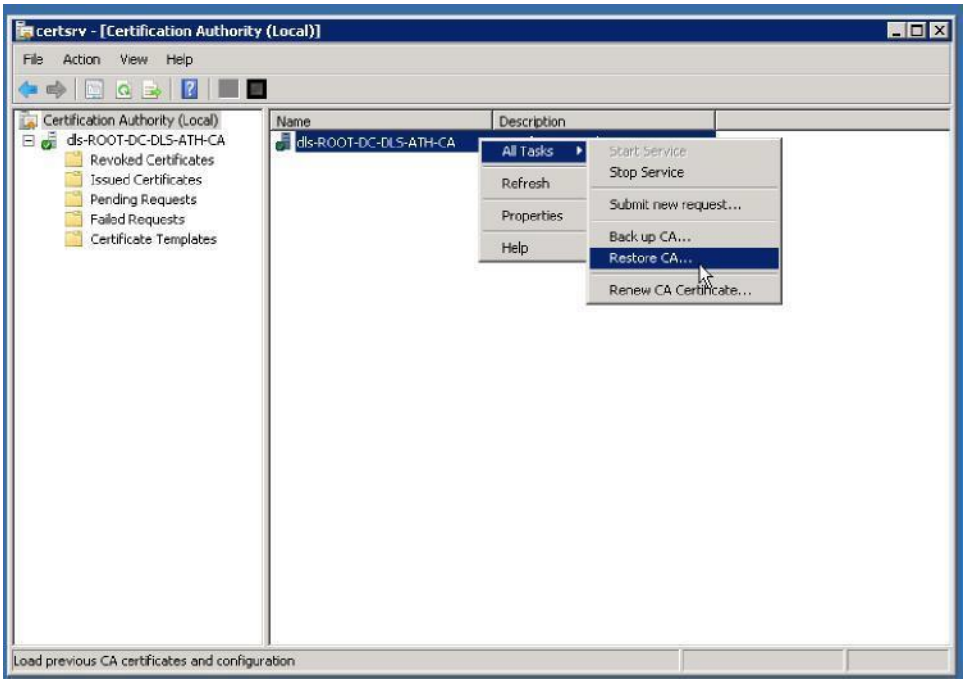


Figure 92 Restore CA

6 DLS server with PKI plugin and connector configured

6.1 Server settings

Set as preferred DNS the PKI DNS server. Set as alternate DNS any other DNS that is higher in the hierarchy as shown in figure 93.

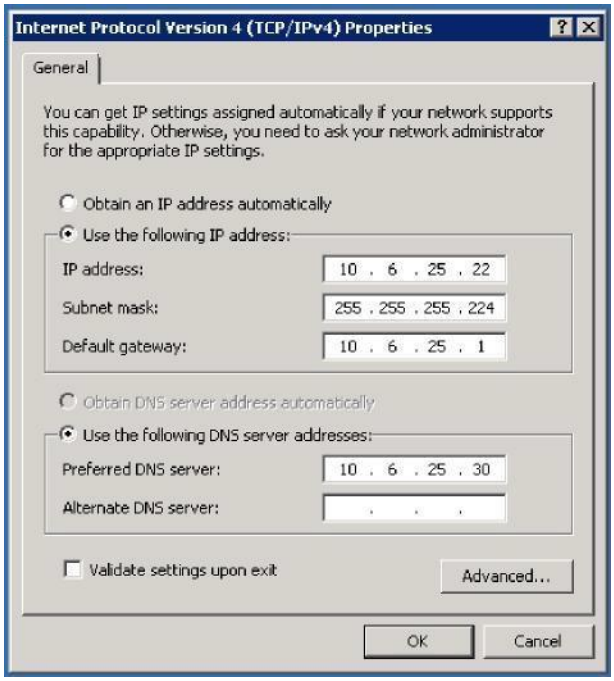


Figure 93 DLS server network settings

Give a computer name to the server and put it in the domain as done for the previously installed servers (for example see section 3.1). Add 'dls' user in the local Administrators group as shown in figure 94. Log off from the server and log on using 'dls' user, in the domain as shown in figure 95.

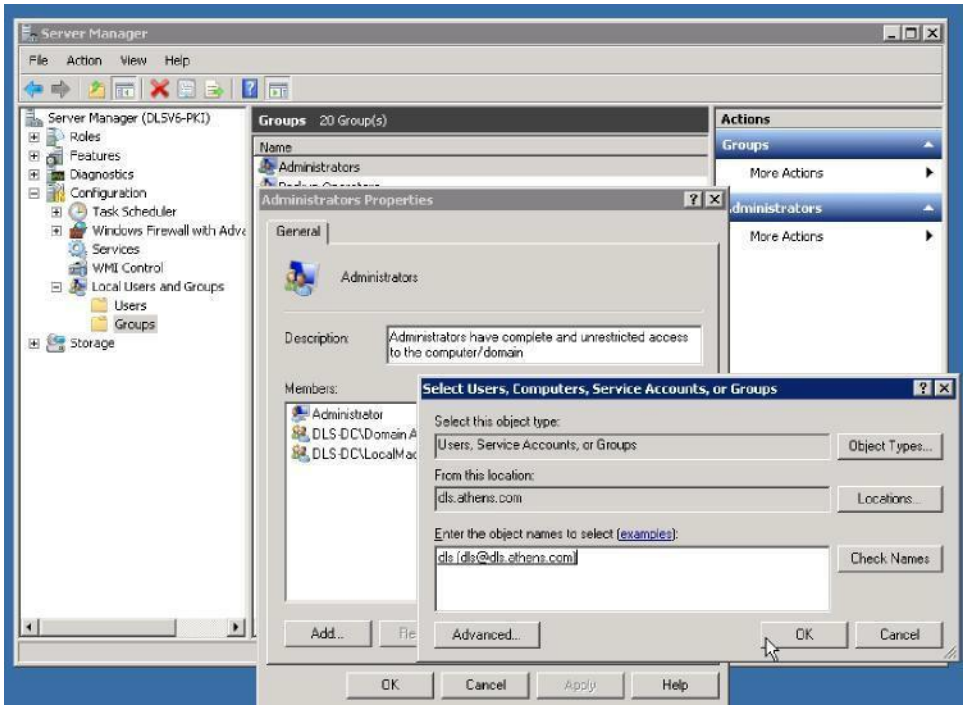


Figure **94** Add 'dls' to local **Administrators group**

Figure 95 Log server to the domain

Configure 'dls' user account to log on to the DLS service:

Through start up menu Navigate to Administrative Tools -> Services

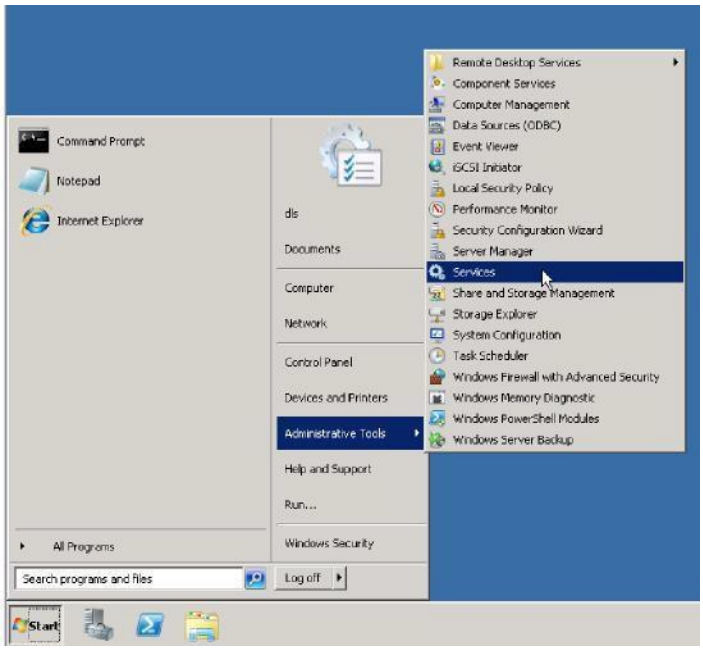


Figure 96 Open Services

Right click on "DeploymentService"and select "Stop" to stop the service. After the service is successfully stopped, right click again on "DeploymentService" and select Properties. Navigate to "Log On" tab and browse for "dls" domain user, e.g.:dls@dls.athens.com. Select dls user and set respective password, as shown in figure 97

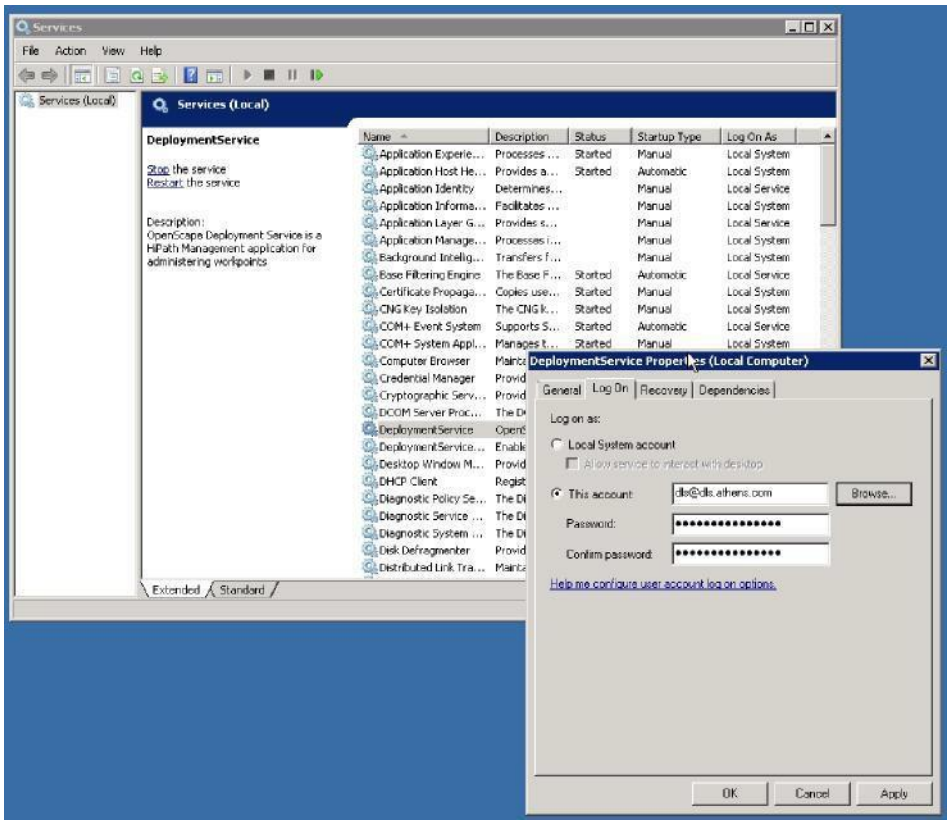


Figure 97 Store Logon account credentials for DLS service

Right click on "DeploymentService" again and start the service. Now dls service will be running under 'dls' user in order to facilitate the communication between DLS server and PKI entities

***Note**: There is a restriction regarding user selected to log on Deployment Service. This user should be the user that performed the installation, or a user with local administration rights and respective rights in SQL server used by DLS.*

6.2 PKI Plug-in Configuration

PKI plug-In and Connector must be configured as the minimum PKI configuration in the DLS server.

6.2.1 PKI Internal Plug-In configuration

This connector plug-in to an internal CA can be used in configurations, where an external PKI is not provided, not available or not accessible from the DLS. In such environm ents it is still possible to manually import certificates and to generate certificates or even CA certificates internally by the DLS. For a better and modularized architecture, this Internal CA Implementation Module is accessed in the same way as external PKI infrastructures using an internal connector plug-in providing the same required interfaces according to the specification.

Figure 98 illustrates the mask of a plug-In to an internal CA. Table 1 provides an explanation of the general tab fields.

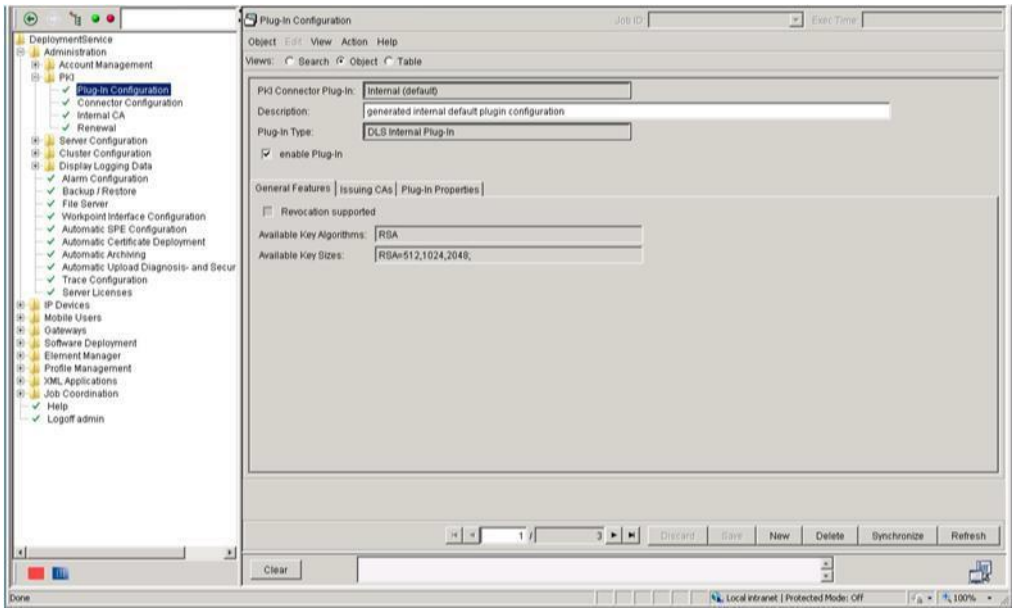


Figure 98 Internal plug-in general features

Field	Info
PKI Connector Plug-In	Unique name of this plug-in configuration
Description	Some additional description
Plug-In Type	The plug-in module used for this configuration. The list contains all installed (and accessible) modules
Enable Plug-In	Enable this plug-in configuration. A PKI connector license may be required to be able to use external PKI
Revocation Supported	Flag to state if this plug-in and configuration supports revocation requests. There are several standards available to support

	revocation (e.g. CMC, CMS, SCEP ...) and even PKI individual solutions. The technical details are plug-in dependent.
Available Key Algorithms	The key algorithms supported by this plug-in. As the PKI connector itself offers access to Crypto Libraries, the plug-in may choose to use its own implementation or even external hardware to create public private key pairs.
Available Key Sizes	The available key sizes

Table 1 Fields in General Features tab sheet

Each Plug-In configuration must be able to provide a list of supported issuing CAs. There may be plug-ins capable of requesting this information from the external CA. It is also assumed that this information can be configured using plug-in dependent properties.
This information is required due to the fact that an external registration authority (RA) may serve more than one CA. In this case the requestor (DLS) may choose the signer (issuer) for the requested certificate.
This information is necessary to be able to view details of the individual issuing CAs, especially the validity period of the signing CA. This enables the DLS to issue alarms when the validity period is running out.

Figure 99 shows the issuing CA for an internal plug-in.

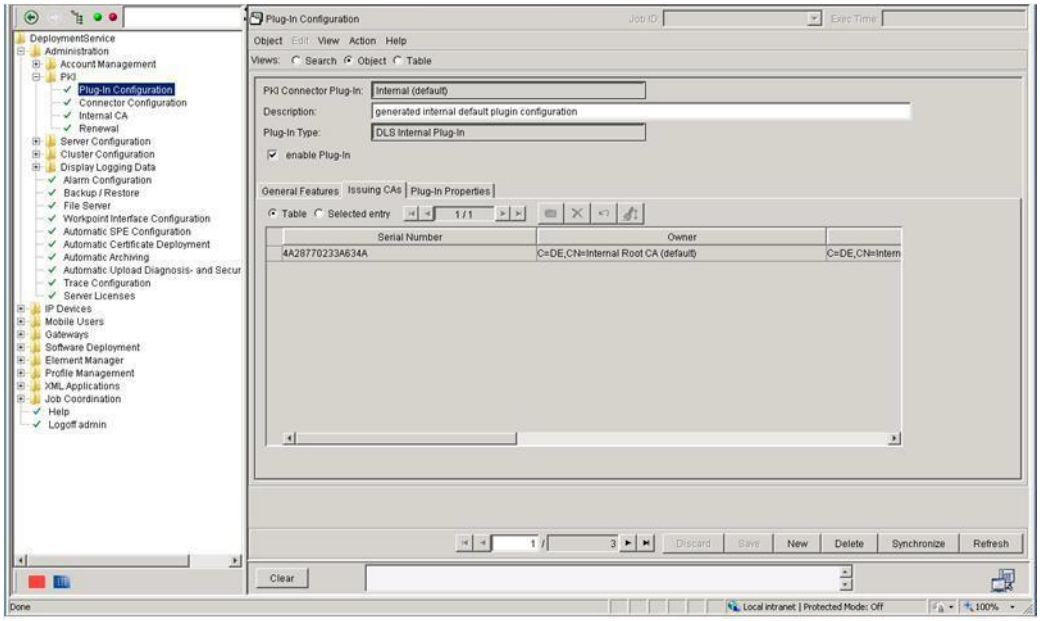


Figure 99 Issuing CAs of internal plug-in

Each plug-in must provide a set of key value pairs used for its configuration. These parameters are vendor specific but can be configured using a generic key value table. Values must be configured with care, as the user interface cannot provide validity checks. A message may be displayed after setting wrong values or keys. These messages are vendor dependent and more information must be obtained from the vendors' plug-in hand book.
There will be a special handling for security relevant properties like passwords. The plug-in must provide information on which properties (key name) should be handled as password. The value of those properties will not be displayed as plain text and must be entered in a generic way using a password dialog. Figure 100 shows the properties of an internal plug-in.
Regarding property 'internal.x509name.template' it specifies the basic items of the template that will be used to request x509 certificates. Its value contains two items:

- the common name (CN), this is the part that must match the host being authenticated. The CN part is not configurable i.e. DLS expects a string having CN=? - C stands for CA country

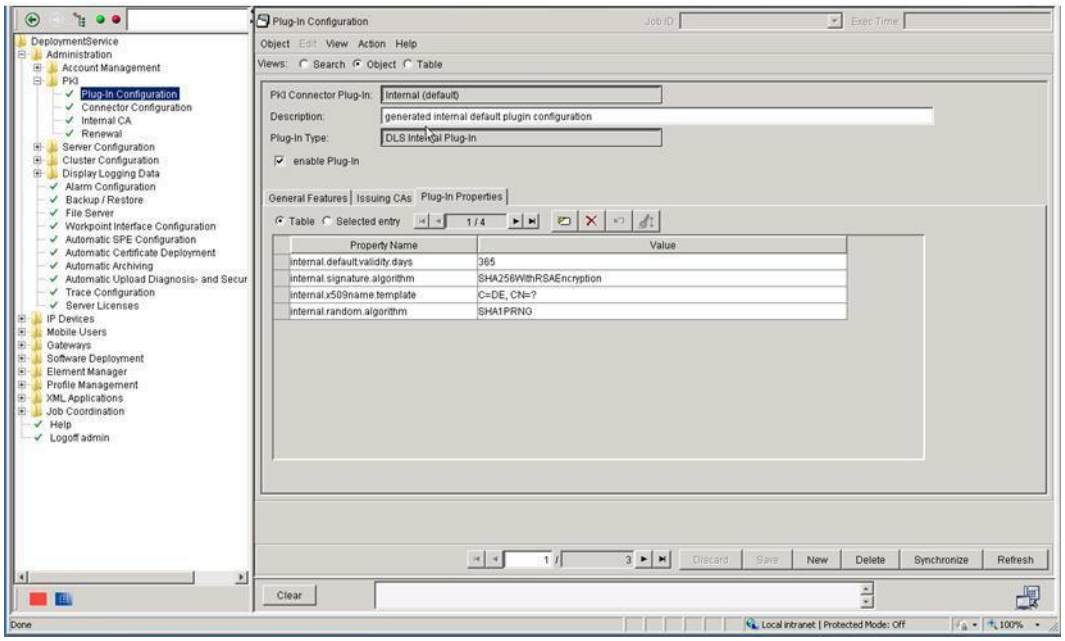


Figure 100 Internal plug-in properties

To create a new plug-in press 'New', enter plug-in name, description and type (e.g.internal). Do not check 'enable plug-in' yet.
Before enabling the Plug-In Configuration there must be executed a synchronization with the PKI. This will be done by pressing the ***Synchronize*** Button. During synchronization the data from tab sheets "General Features" and "Issuing CAs" will be read out from the PKI and will be written into DLS DB. The result of the synchronization will be displayed in the status bar.

The possible actions for the plug-in window are listed and described in table 2.

Actions	Info
Discard	Changes on the table will be discarded and the previous values will be loaded.
New	Create a new Plug-In Configuration.
Save	Table changes will be saved. The connector will be informed about the new changes and may respond with an error message which will be displayed in the status message field. On success: the values will also be saved in the DLS database.
Refresh	Reload actual values from the DLS database
Synchronize	This Button **must be used** to synchronize the configuration and the PKI Plug-In. The General Features data and the Issuing CAs belonging to this Plug-In Configuration will be read out and will be written into DLS DB. In case of errors a status message will be displayed and some controls within this window will show success / failure status.

Table 2 Actions for Plug-In window

After the synchronization was successful enable internal plug-in by selecting the respective checkbox as in figure 101.

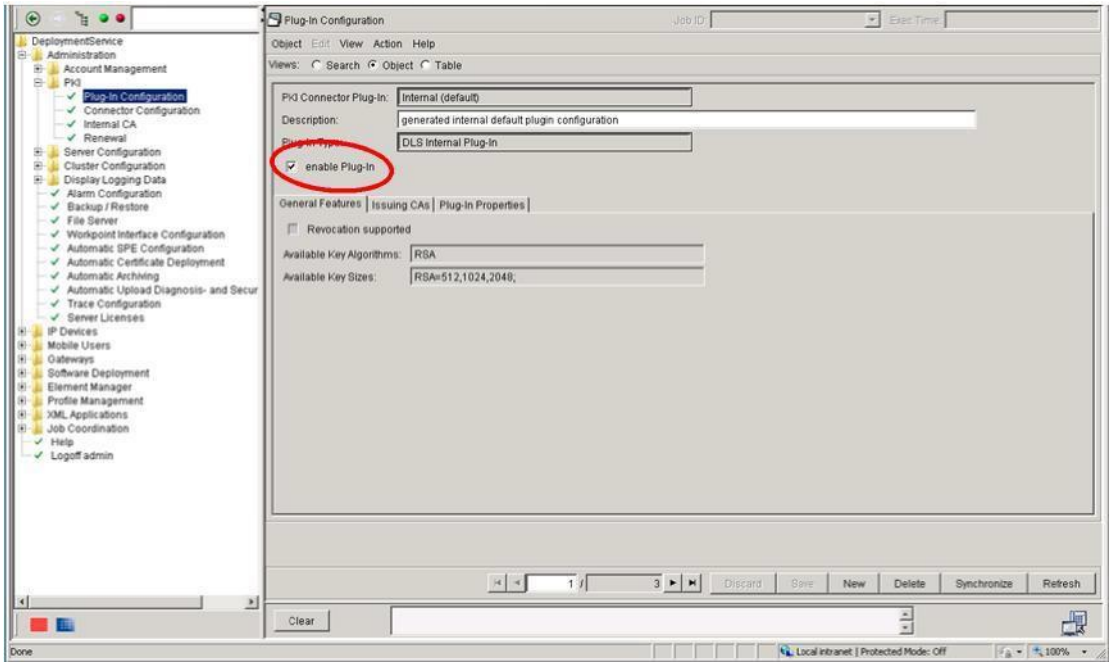


Figure **101** Enable plug-in

6.2.2 PKI Root CA Plug-In configuration

Each connector plug-in may support different configuration options as individually defined by its provider. Therefore the plug-in must only provide a minimal set of common information (feature set, supported key algorithms ...) which will be displayed to the user but also used in other windows for configuration.

For common configuration scenarios, each plug-in must provide and maintain a list of key value pairs which can be configured using a generic DLS user interface.

As the supported list of plug-ins is a deployment and installation issue, the list of available plug-ins is fixed during runtime and cannot be configured by the user interface. There can be more than one configuration per plug-in. This is required to be able to use the same Plug-In module in different configurations and to access more than one external CA instance:

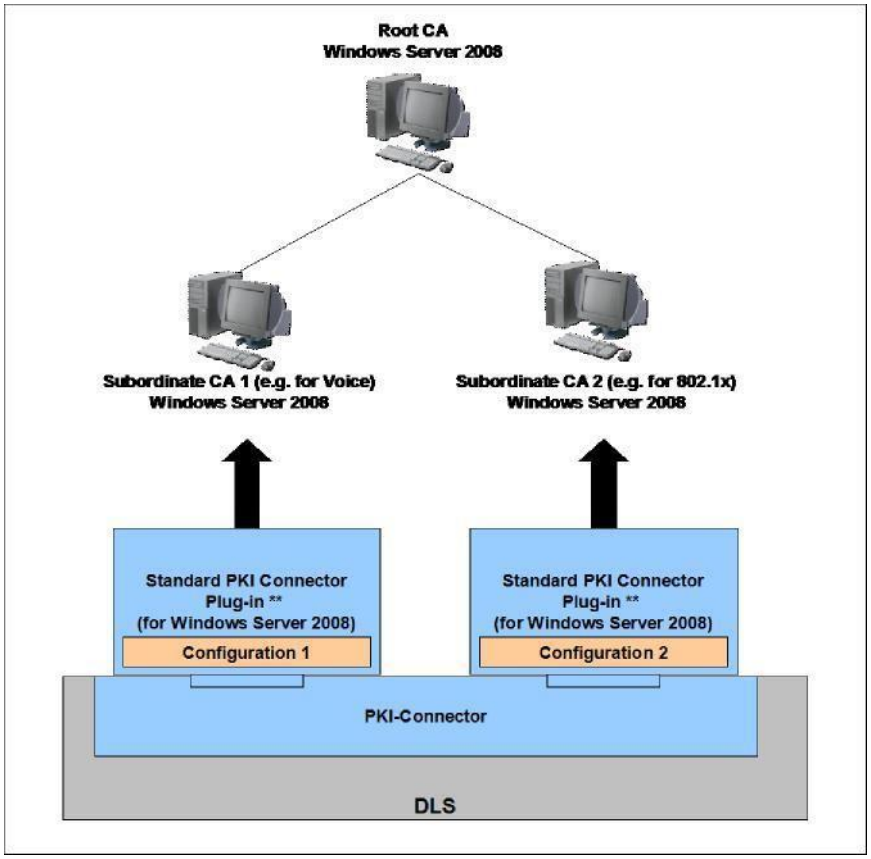


Figure 102 Individual Plug-In configurations

The DLS PKI Connector provides a ready to use plug-in for standard PKI environments based on Windows Server 2003, Windows Server 2008 and Windows Server 2008 R2. The aim is to support common customer infrastructures with a ready to use and tested plug-in to minimize time to market for common customer project scenarios. Integration with Windows PKI will show following restrictions:

- Enterprise CA only (Standalone CA is not supported)
- All CAs are accessed by the same windows credentials, this means DLS must be configured using a domain account with proper rights on all CAs (issuing and revoking certificates). I.e. CAs in different windows domain forests are not supported.

The general features tab presents some common information of services and features each plug-in provides. This information cannot be edited. The general properties of an external plug-in (to an MS root CA) are shown in figure 103. Table 1 describes the fields of this mask.

For infrastructures that uses Subordinate CAs to request and enroll certificates, the only mandatory Plug In Configuration is the one for Subordinate CA (Root CA plug in is not required)

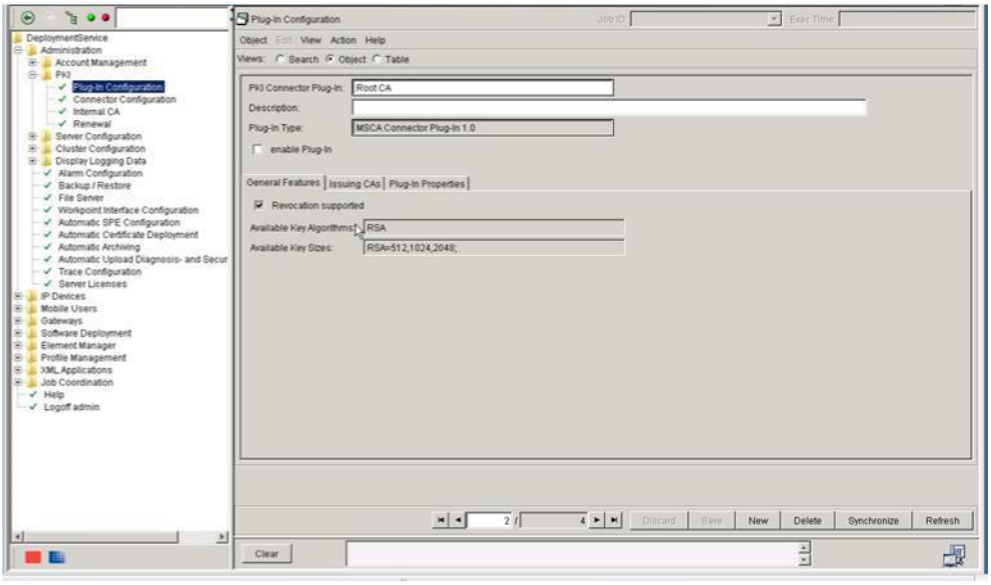


Figure 102 PKI connector External plug-in general properties

The role of issuing CAs was described in section 6.2.1. Figure 103 shows that a root and a subordinate CA are the issuing CAs for that plug-in.

Note: Issuing CAs are retrieved from PKI during synchronization. During that process server request and retrieves all issuing CAs as those are configured in Active Directory. In case a CA server has not joined the domain, i.e. not configured in Active Directory, then this CA w ill not be retrieved and displayed as a issuing CA in below mask. This could be the case of a Root CA that hasn't joined the domain (Standalone Root CA)

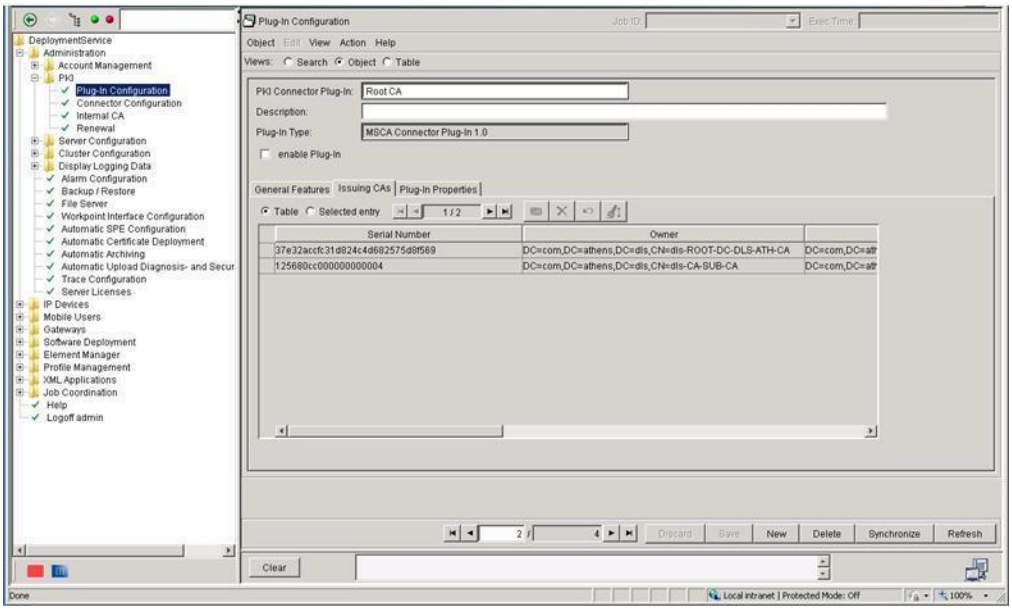


Figure 103 Issuing CAs tab

Figure 104 shows the properties of the root CA. A short explanation follows.

- **msca.ldap.trustedCertificate** Path to the file where the trusted (as it regards to the connection with LDAP) certificate is stored. LDAP server certificate shall be verified according to msca.ldap.trustedCertificate. This file should be located in the common data of the environment.
- **msca.user** There the 'dls' user can be entered
- **msca.server** The domain name of the MS subordinate CA server

- **msca.clientcert.template.name** This is the name of the client certificate template configured in the subordinate CA ('Userdls' in this example)
- **msca.ldap.certificateVerification** can be set to:
  - 'None',
  - 'Trusted',
  - 'Full'
- **msca.ldap.server** The domain name of **DNS server** (which acts as LDAP)
- **msca.signature.algorithm** Encryption algorithm of CA
- **msca.ldap.port** default is 389
- **msca/x509name.template** This is information that will be incorporated in certificate request to the CA. Section 6.2.2.1 provides a description of of x509 certificate request fields.
- **msca.servercert.template.name** This is the name of the server certificate template configured in the subordinate CA ('WebServerDls' in this example)
- **msca.ldap.security** Boolean type where:
  - True: use TLS protocol to connect to LDAP. Secure LDAP port should be set as msca.ldap.port.
  - False: not use TLS

  When is set to true, DLS should try to connect to LDAP using TLS.
- **msca.password** The password for user 'Administrator' in subordinate CA
- **msca.revocation.enabled** To state if this plug-in and configuration supports revocation requests
- **mscaldap.clientCertificate** Path to the .p12 file where the DLS client certificate is stored. This file should be located in the common data of the environment.
- **mscap.ldap.clientCertificatePassword** the password for the .p12 file.

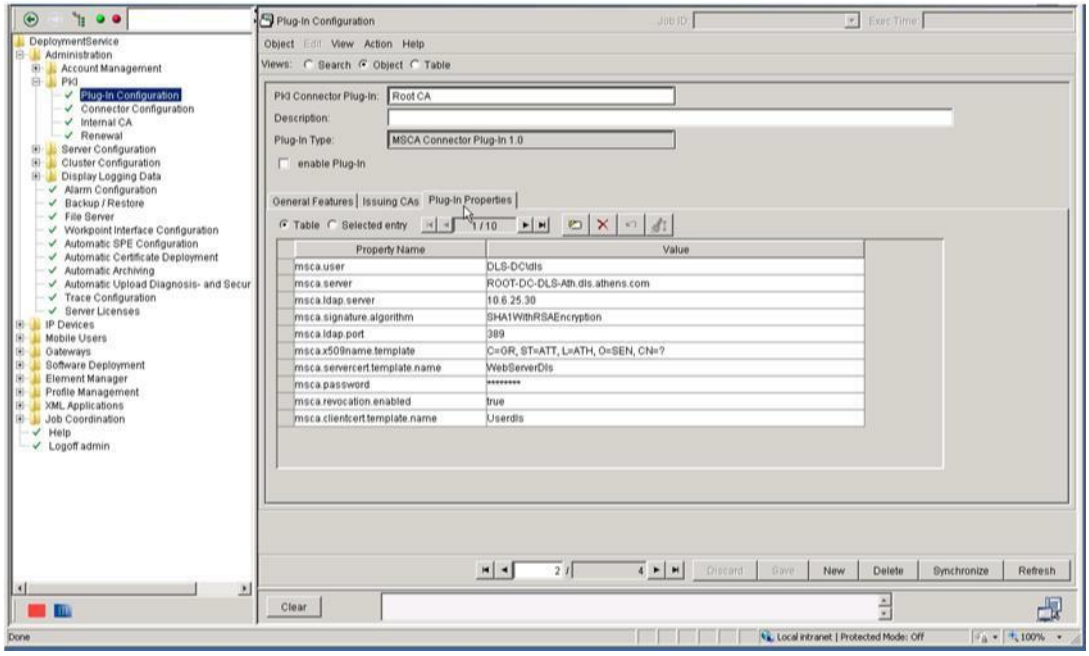


Figure 104 External plug-in properties

To create a new External plug-in press 'New" in the 'Plug-in Configuration' mask, give name and description, select as Plug-in Type, 'MSCA Connector Plug-in 1.0' and save. Then go to Plug-in Properties tab and enter the properties according to your deployment.

6.2.2.1 Fields of x509 certificate request

This is an example of x509 certificate request fields to facilitate DLS PKI plugin configuration
Country Name (2 letter code) [AU]: *TW*
State or Province Name (full name) [Some-State]: *Taiwan*

Locality Name (eg, city) []: *Taipei City*
Organization Name (eg, company) [Internet Widgits Pty Ltd]: *Tavern*
IMACAT's Organizational Unit Name (eg, section) []: *Owner*
Common Name (eg, YOUR name) []: *Tavern IMACAT's*

1. You have to fill everything in English (ASCII characters). Only ASCII English characters are allowed in X.509 certificates.
2. Country Name is the two-letter upper-cased country code. The country code of Taiwan is TW. Refer to the ISO-3166 two-letter country code list if you are not in Taiwan.
3. State Name is the full name of your country. You cannot fill in the country code above here. Fill in Taiwan here if you are in Taiwan.
4. Locality Name is your place name. Fill in your city or your county here.
5. Organization Name is the name of your organization. Fill in the name of your company name, your school or your institution here.
6. Organizational Unit Name is the name of your department. Fill in the name of your department or your unit here.
7. Organizational Unit Name is the name of this certificate. If this is a root CA, fill in the previous organization name here. You may append a RSA/2048 after the name to identify this CA in the future. If this is a server certificate, fill in the full qualified domain name of the server (www.abc.com) here. If this is an e-mail certificate, fill in your e-mail here.

6.2.3 PKI Plug-In to Subordinate CA

If a two-tier hierarchy system is installed a subordinate CA is needed. Figure 105 shows the properties tab of a subordinate CA.

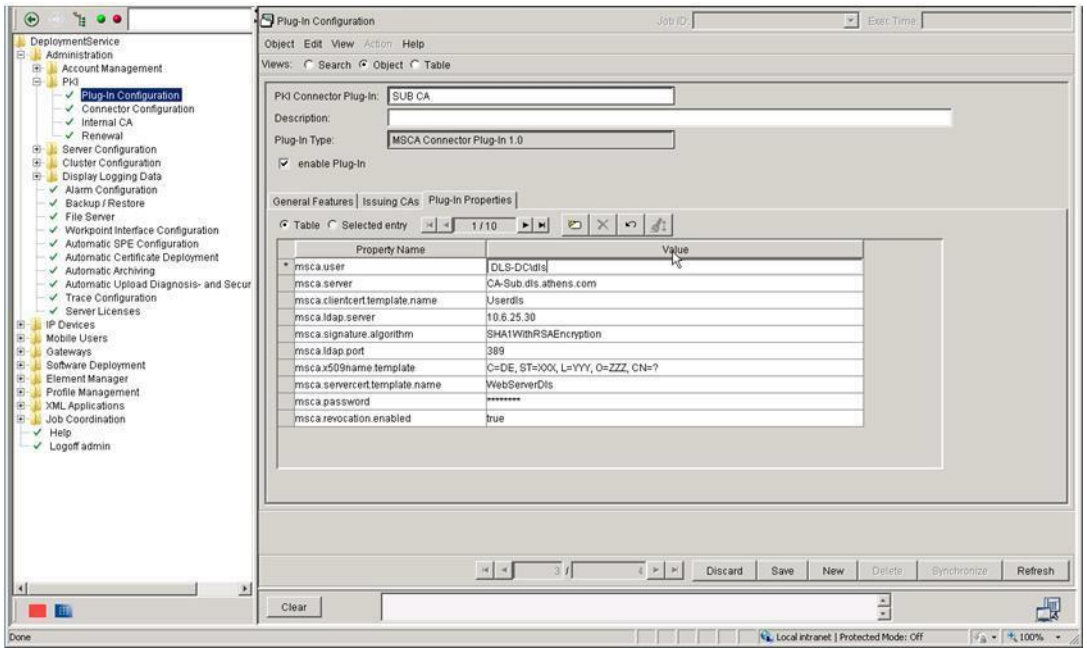


Figure 105 Subordinate CA Plug-in properties

The following properties are contained in the table. After saving a new plug-in, some of them need manual configuration by the user.

- **msca.ldap.trustedCertificate** Path to the file where the trusted (as it regards to the connection with LDAP) certificate is stored. LDAP server certificate shall be verified according to msca.ldap.trustedCertificate. This file should be located in the common data of the environment.
- **msca.user** There the 'dls' user can be entered
- **msca.server** The domain name of the MS subordinate CA server

- **msca.clientcert.template.name** This is the name of the client certificate template configured in the subordinate CA ('Userdls' in this example)
- **msca.ldap.certificateVerification** can be set to:
  - 'None',
  - 'Trusted',
  - 'Full'
- **msca.ldap.server** The domain name of **DNS server** (which acts as LDAP)
- **msca.signature.algorithm** Encryption algorithm of CA
- **msca.ldap.port** default is 389
- **msca/x509name.template** This is information that will be incorporated in certificate request to the CA. Section 6.2.2.1 provides a description of of x509 certificate request fields.
- **msca.servercert.template.name** This is the name of the server certificate template configured in the subordinate CA ('WebServerDls' in this example)
- **msca.ldap.security** Boolean type where:
  - True: use TLS protocol to connect to LDAP. Secure LDAP port should be set as msca.ldap.port.
  - False: not use TLS

  When is set to true, DLS should try to connect to LDAP using TLS.
- **msca.password** The password for user 'Administrator' in subordinate CA
- **msca.revocation.enabled** To state if this plug-in and configuration supports revocation requests
- **mscaldap.clientCertificate** Path to the .p12 file where the DLS client certificate is stored. This file should be located in the common data of the environment.
- **mscap.ldap.clientCertificatePassword** the password for the .p12 file.

Figure 106 shows the default general features of the plug-in. There is the option to support certificate revocation or not depending on the customers needs.

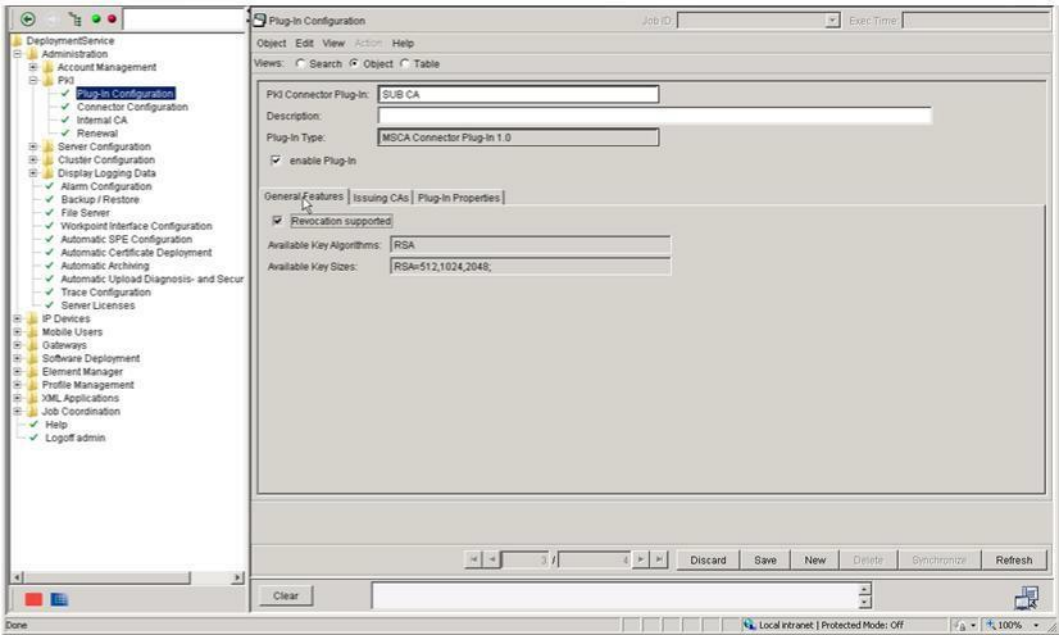


Figure 106 Subordinate plug-in General features (default)

To create a new plug-in to subordinate CA press 'New" in the 'Plug-in Configuration' mask, give name and description, select as Plug-in Type, 'MSCA Connector Plug-in 1.0' and save. Then go to Plug-in Properties tab and enter the properties according to your deployment. The main difference from the properties of a root CA plug-in is that for the subordinate CA in **msca.server** the subordinate domain name is entered.

For infrastructures that uses Subordinate CAs to request and enroll certificates, the only mandatory Plug In Configuration is the one for Subordinate CA (Root CA plug in is not required)

6.3 PKI Connector Configuration

Once installed and plugged into the DLS, different connectors can be used and configured using the DLS Client. As the variety of configuration parameters is immense (e.g. used for certificate protocols, authentication, additional and optional parameters when requesting certificates ...), only common high level attributes are available for configuration via the DLS Client GUI. Very specific or individual parameters are to be configured using common or individual configuration files.

A connector configuration is used to

- define the plug-in configuration used to access the external PKI
- choose an issuing CA used to request certificates from and to be able to verify relationship with the configured trust anchor
- define some high level parameters used in the certificate request sent to the CA
  - define and import the trust anchor to be deployed to devices when using this configuration
- A connector configuration can be dedicated to a certificate type (e.g. SPE, 802.1x, WBM) or can be used as a global configuration for all / some types of certificates to be deployed.

6.3.1 PKI Internal Connector Configuration

The following figure illustrates the request parameters of an internal connector. Table 4 gives a description of the parameters. Once the parameters are saved a trust anchor can be imported as figure 107 illustrates. Once trust anchor certificate is imported (figure 108) the corresponding mask shows the certificate parameters (figure 109). Then the connector can be tested by pressing the 'Test' button (figure 110).
Following connector configuration, the connector can be enabled by checking the respective checkbox.

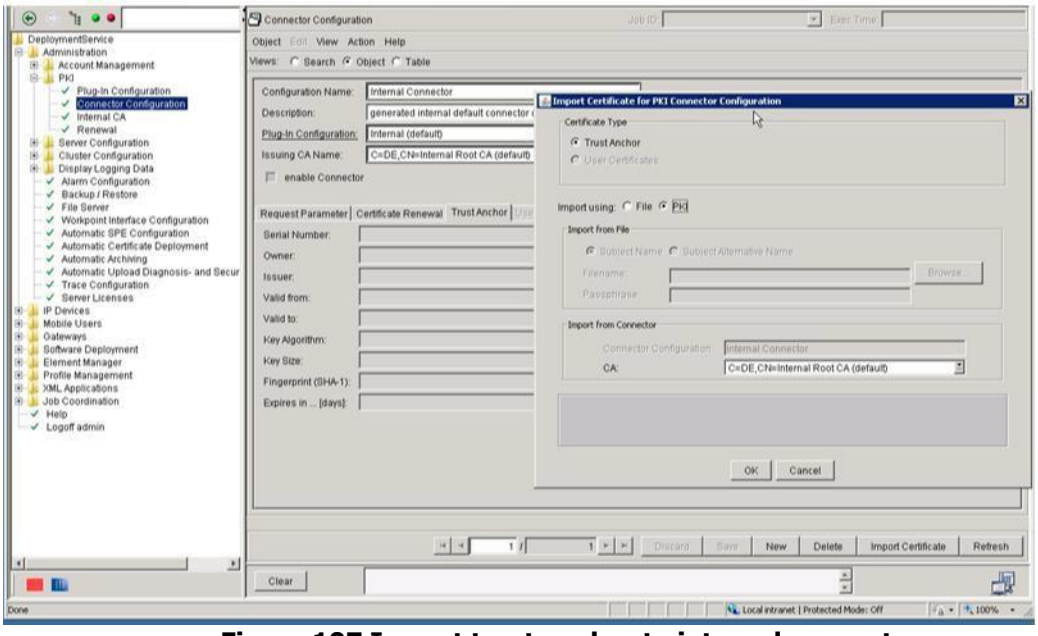


Figure 107 Import trust anchor to internal connector

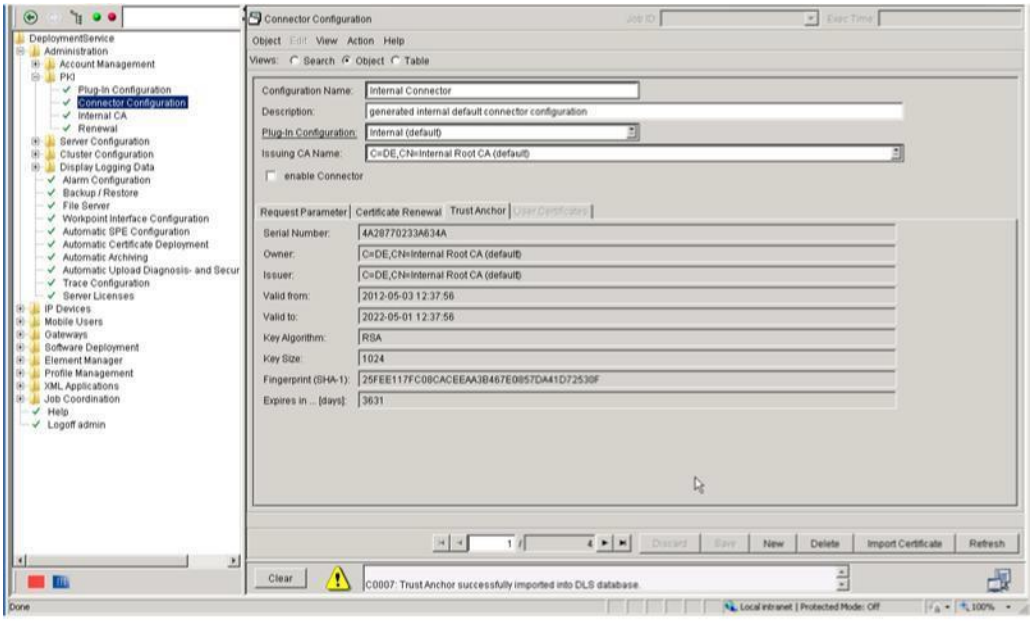


Figure 108 Trust anchor of the internal connector

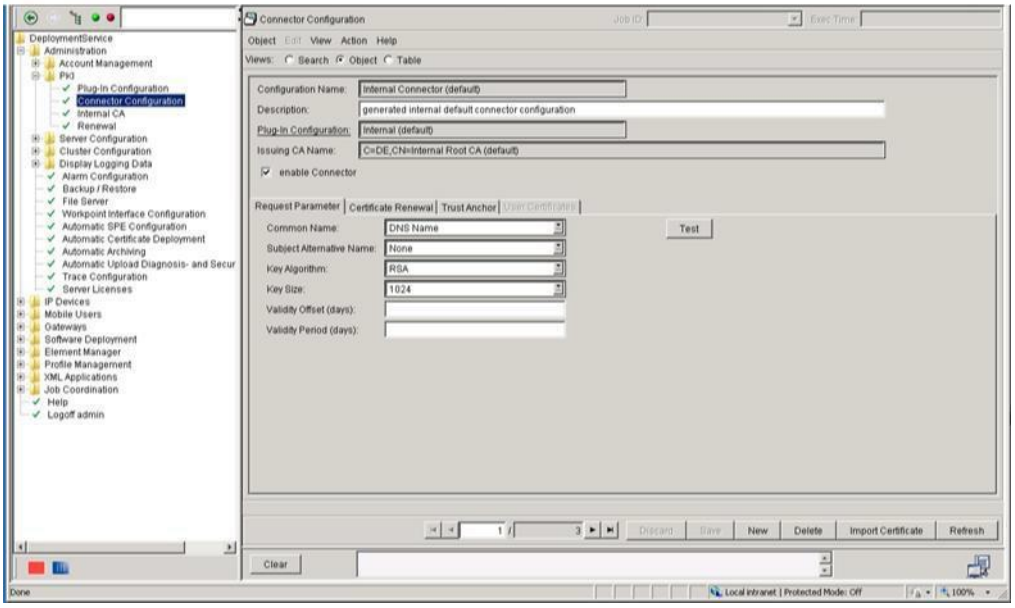


Figure 10**9** Request parameters of the default internal connector

Field	**Info**	**Value**
Configuration Name	The unique name for this configuration (global)	Required
Description	Some additional description (global)	Optional
Plug-In Configuration	The plug-in configuration to be used (global)	Required
Issuing CA Name	The issuing CA name to be used for signing when requesting new certificates. This is required if an RA is serving more than one CA. This field is also required for integrity checks: the issuing CA and the trust anchor must be related (part of the certificate chain). (global)	Required
Enable Connector	Enable this Connector Configuration (global)	Required
Common Name	The Subject Name selection provides either □ MAC Address	Required

	• DNS Name • IP Address To comply with standards DNS Name should be used! **In case of certificate requests for gateways this selection will not be used (it will always be DNS Name, if available, or IP Address)**	
Subject Alternative Name	The Subject Alternative Name extension allows either • None • DNS Name • IP Address • MAC Address The DLS provides no information of DNS names; therefore a DNS reverse lookup is required. If the DNS reverse lookup fails, the IP Address will be used instead. **Please Note:The CA may ignore this field (depending on its policy).** **In case of gateways this selection is not used (will always be DNS Name or IP Address)**	Optional
Key Algorithm	The algorithm to be used for generating public private key pairs. The available list of algorithms depends on the plug-in	Required
Key Size	The list of available key sizes; plug-in dependent	Required
Validity Offset	The offset in days to be used for the validity period. Negative values are accepted, too. (see also validity period)	Optional
Validity Period	Defines how long (in days) shall new certificates be valid (counted from the time of the request on). **Please Note:** **In most cases the validity values are ignored / overruled by the CA. Thus: these settings are PKI (policy) dependent.**	Optional

Table 3 Fields in Request Parameter tab sheet

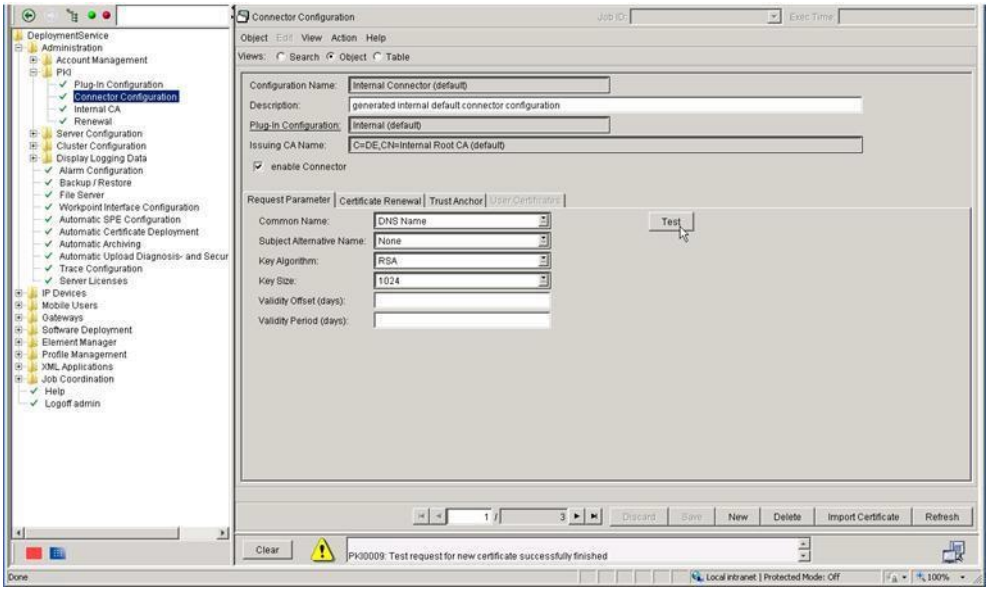


Figure 110 Test connector to internal plugin

6.3.2 PKI Connector to Root CA

A connector to the root CA is also configured. To configure the new connector press 'New' in the 'Connector Configuration' mask. Then select the needed parameters in the required fields. The fields are described in table 3. The request parameter tab of the example connector is shown in figure 111.

For infrastructures that uses Subordinate CAs to request and enroll certificates, the only mandatory Connector Configuration is this for Subordinate CA (Root CA connector is not required)

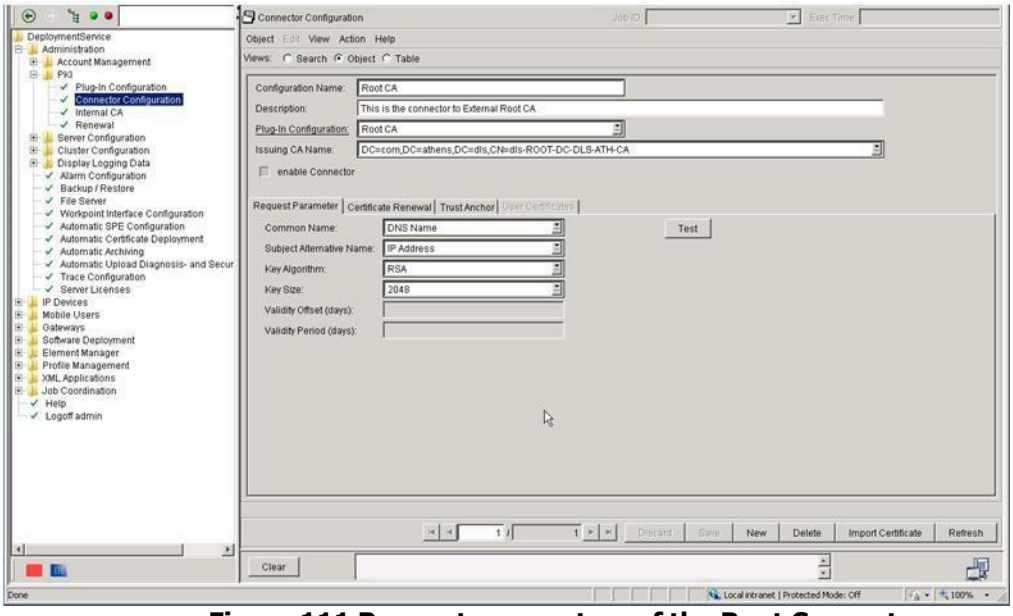


Figure 111 Request parameters of the Root Connector

Import trust anchor respective to selected plug in configuration. The CA selected on "Import Certificate for PKI Connector Configuration" page will be the trust anchor for this Connector Configuration.In most scenarios this will be the Root-CA itself, but can be a subordinate CA, as well. If a trust anchor is not available, the configuration cannot be saved!

The DLS Connector will verify the relationship to the issuing CA. It is absolutely necessary that the issuing CA is part of the certificate chain with the trust anchor as the overall trusted parent node.

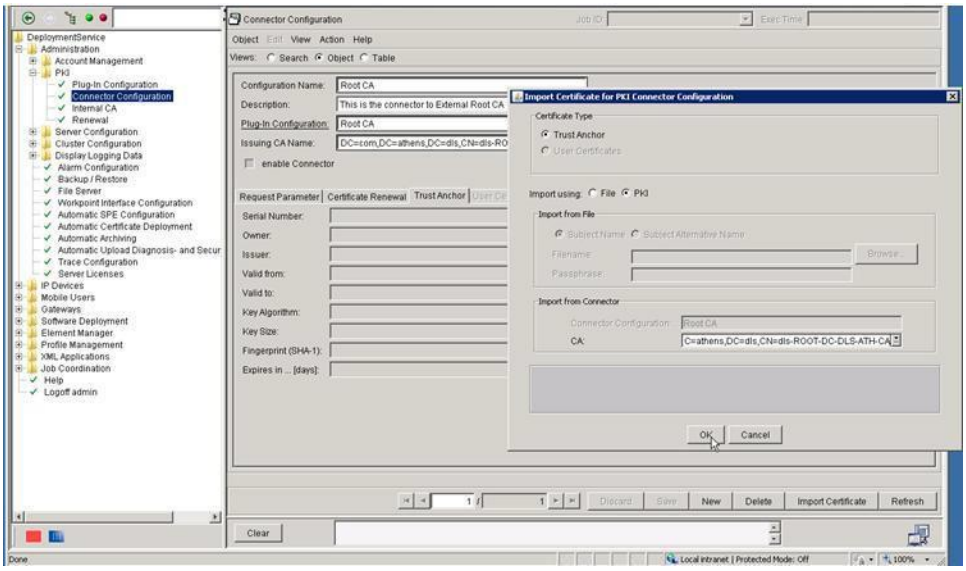


Figure 112 Import trust anchor to the connector

Figure 113 shows the certificate parameters of the imported trust anchor.

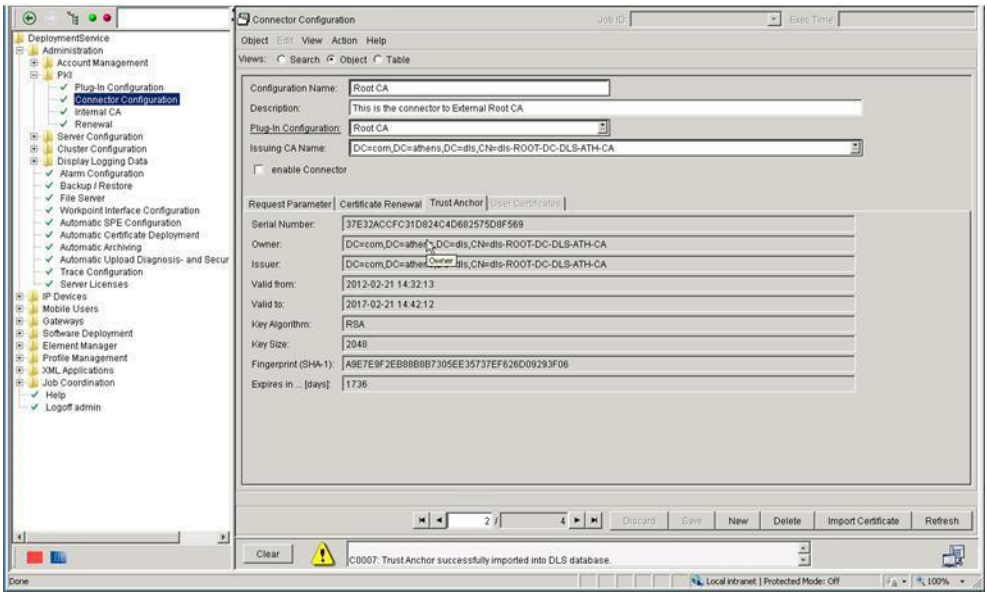


Figure 113 Trust anchor of the Root Connector

Following the Connector Configuration you may send a test certificate request by pressing 'Test'. If the request was successful a message is displayed as in figure 114.

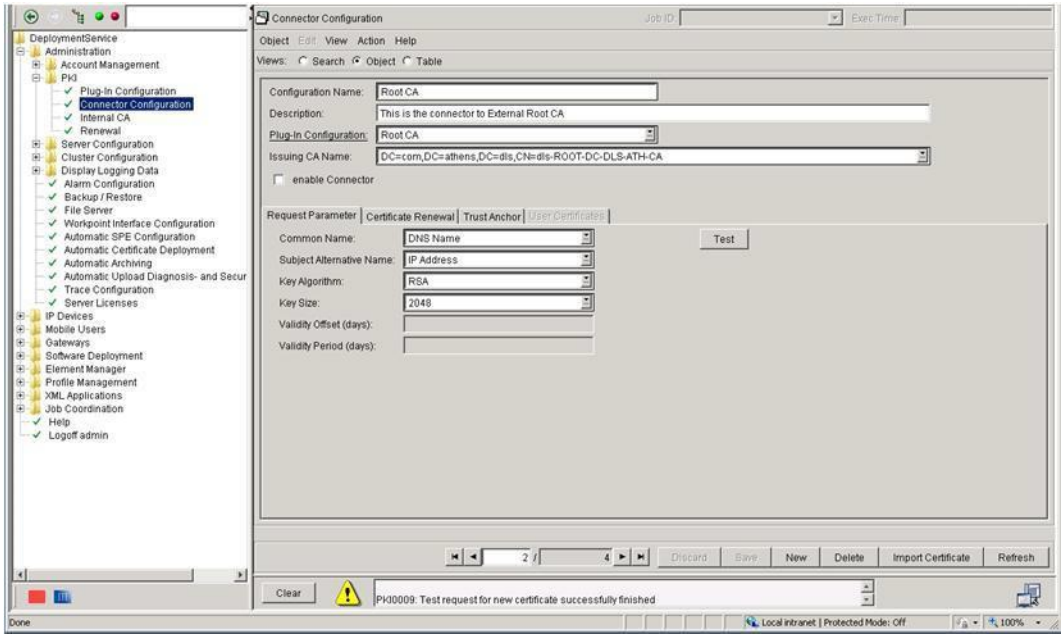


Figure 114 Test connector to External Root CA

Finally enable the connector by selecting the respective checkbox.

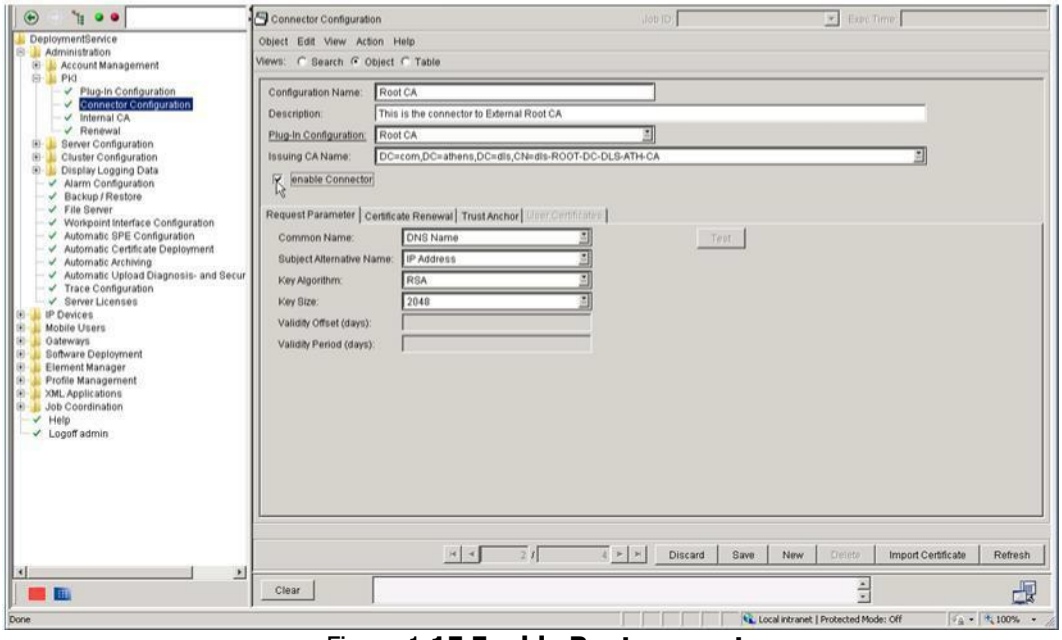


Figure 1 **15 Enable Root connector**

6.3.3 PKI Connector to Subordinate CA

For a subordinate CA a different connector can be configured. The procedure is the same as the one described in the previous two sections except that the used plug-in and CA are that of the subordinate.

For infrastructures that uses Subordinate CAs to request and enroll certificates and Root CA is an offline server, the only mandatory Connector Configuration is this for Subordinate CA (Root CA connector is not required)

Figure 116 shows the request parameters.

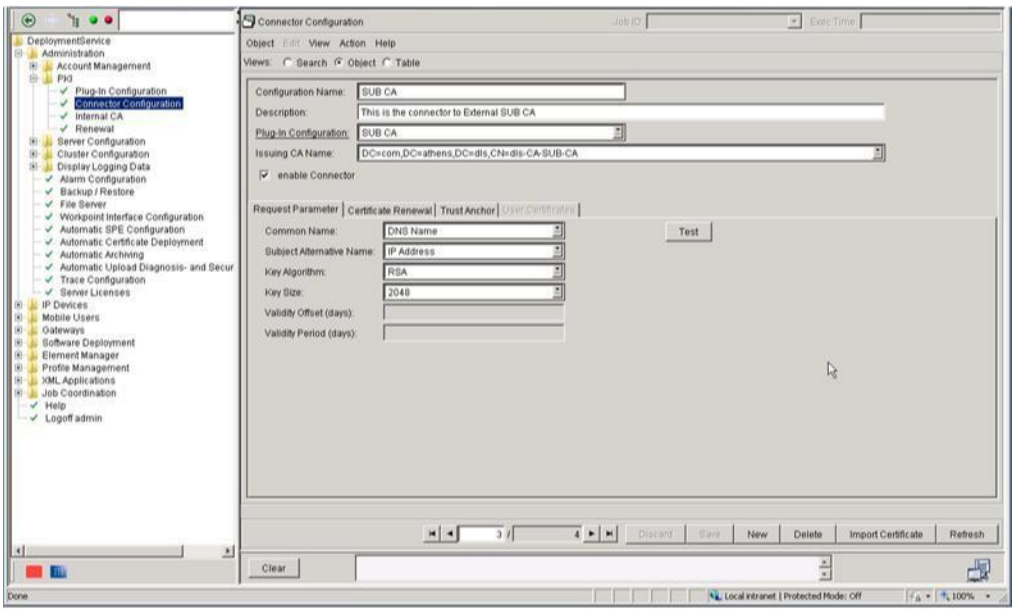


Figure 116 Subordinate Connector request parameters

In The fields 'Plug-in Configuration' and 'Issuing CA Name' use the respective items of the subordinate CA.

Next step is to import trust anchor respectively to selected plug in configuration. The CA selected on "Import Certificate for PKI Connector Configuration" page will be the trust anchor for this Connector Configuration.In most scenarios this will be the Root-CA itself, but can be a subordinate CA, as well. If a trust anchor is not available, the configuration cannot be saved!

The DLS Connector will verify the relationship to the issuing CA. It is absolutely necessary that the issuing CA is part of the certificate chain with the trust anchor as the overall trusted parent node.

Note: In case the trust anchor isn't displayed in the list of CAs (e.g. trust anchor is an offline Root CA not configured in Active Directory) then user is able to select a manual import of trust anchor certificate through file.

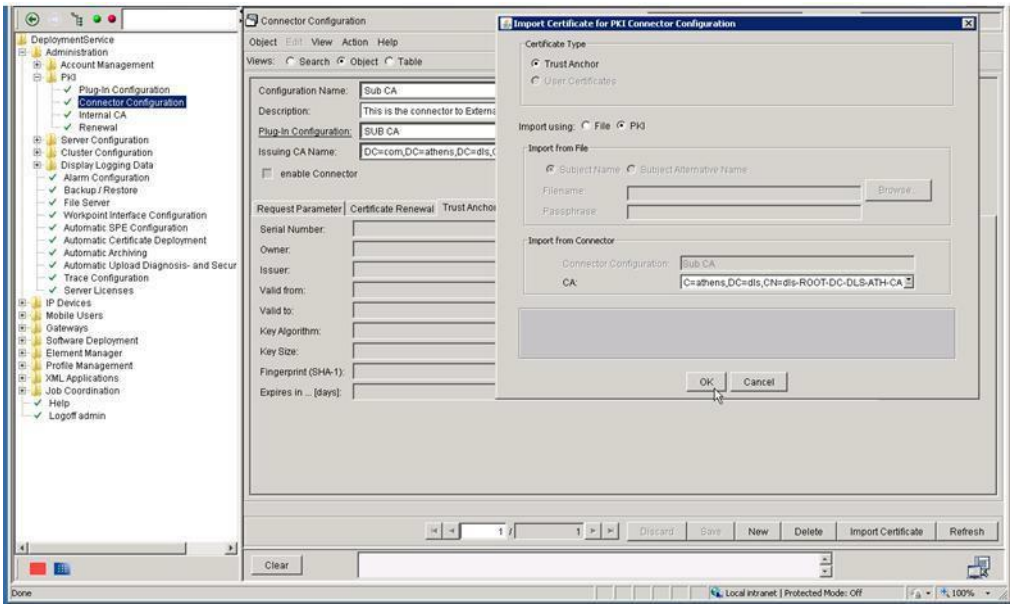


Figure 117 Trust anchor Sub CA

Figure 118 shows the parameters of the imported trust anchor.

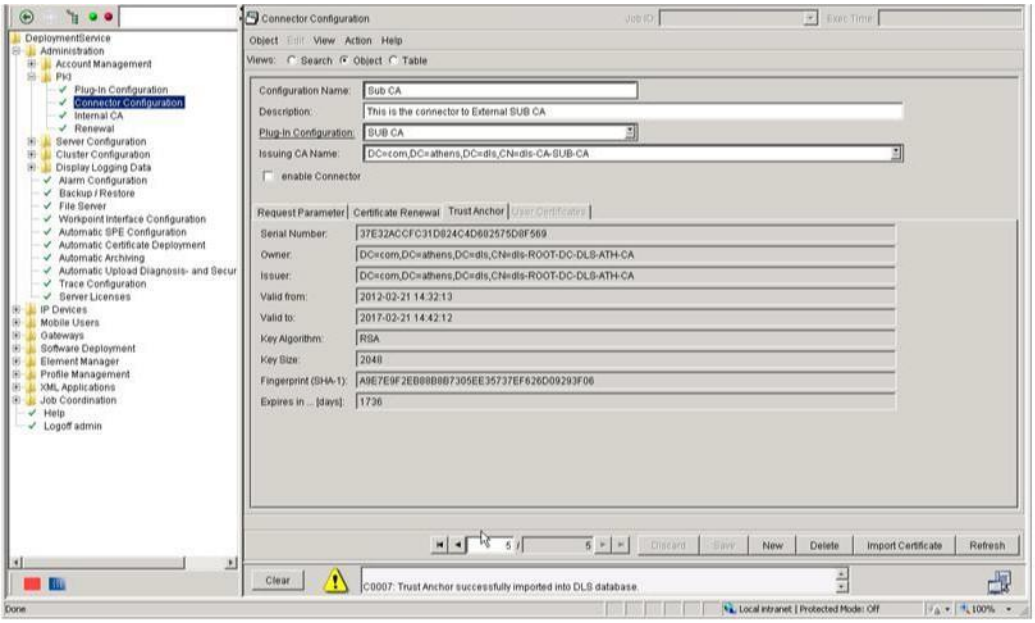


Figure 118 Trust Anchor parameters

Then test that the connector can request a certificate from the subordinate CA (figure 119).

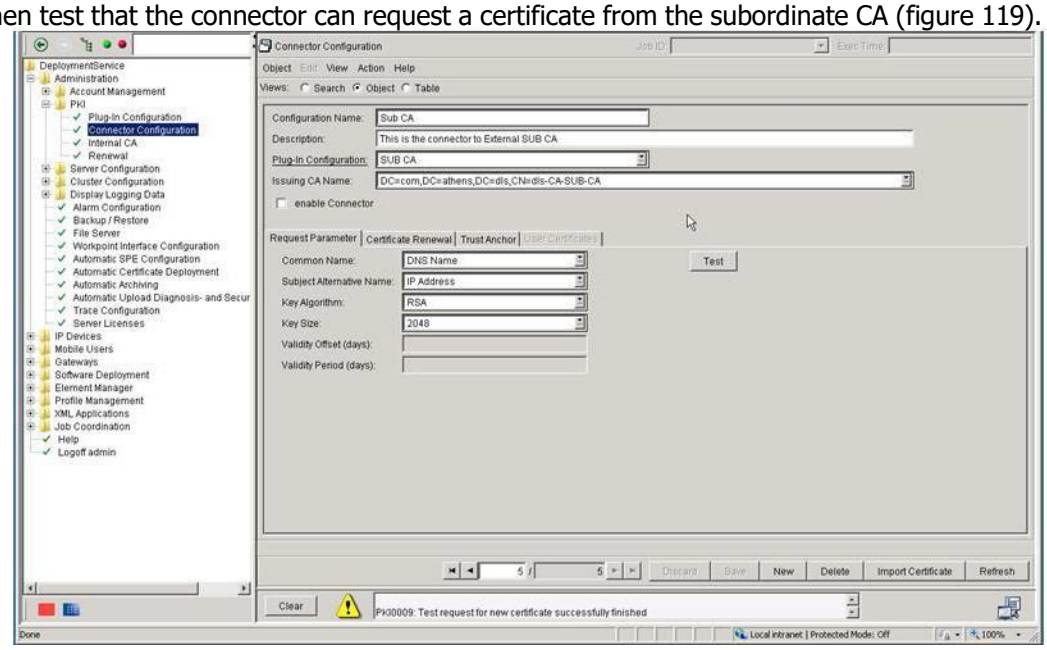


Figure 11**9** Test request for the subordinate connector

6.3.4 Internal CA Configuration

In case a customer does not provide and maintain an own PKI, an internal CA can be configured and used instead. This internal CA does not offer services like a real PKI (e.g. revocation, revocation lists, CRL, OCSP ...) and is therefore not recommended if security is a critical issue.

To support certificate creation and signing either a self-signed CA certificate (root CA) is created using internal crypto libraries, or an external CA certificate with its private key can be imported. An internal CA is shown in figure 120. The following tables explain the related fields and items.

Fields	Description	Value
CA Name	A unique name identifying this internal CA certificate	Required
Description	A more detailed description of this certificate	Optional

Table Fields in Internal CA window (object view)

Actions	Description
Create CA	Open the create dialog to create a self-signed root CA used for issuing certificates
Import CA	Open the import dialog to import a .p12, .pfx file containing a CA certificate and its associated private key. The imported CA certificate must contain all necessary extensions required to present itself as a valid issuing CA
Save	Save the new imported file and its name and description
Delete	Delete this internal CA certificate **if no connector configuration is assigned**.

Table 4 Internal CA window actions

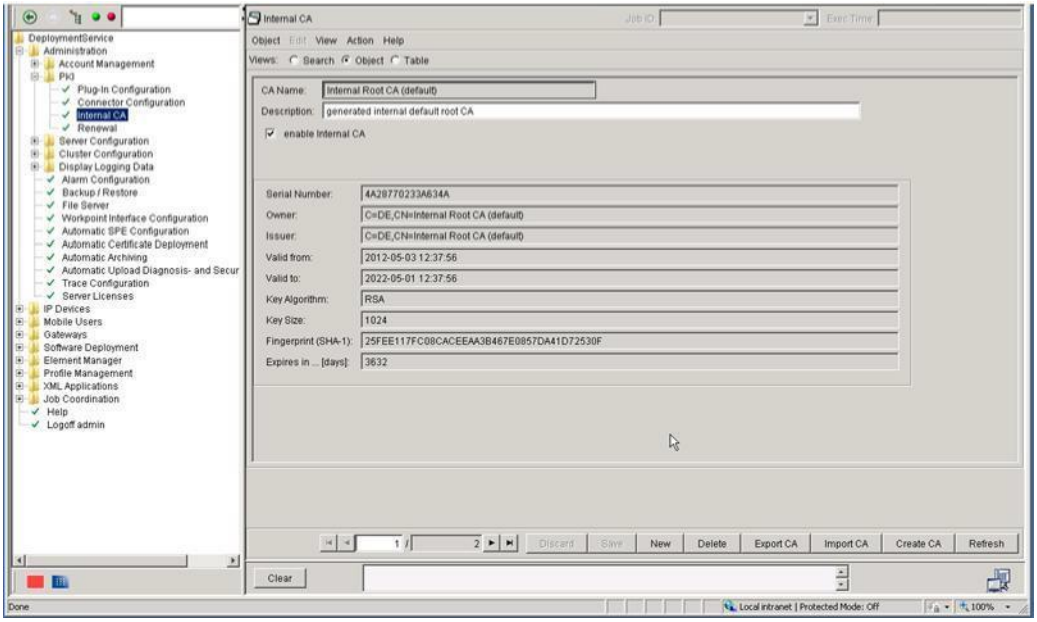


Figure 1 **20** Internal CA

6.3.5 PKI Licenses

PKI is implemented in DLS V6 where a base license for the DLS application is needed. Internal plugin does not need licenses to be used. Licenses are only required for MS Plugin. DLS license includes a number of PKI User licenses. In case a CLA is connected to multiple DLS the PKI User licenses can be distributed to the different DLS servers as shown in figure 121.

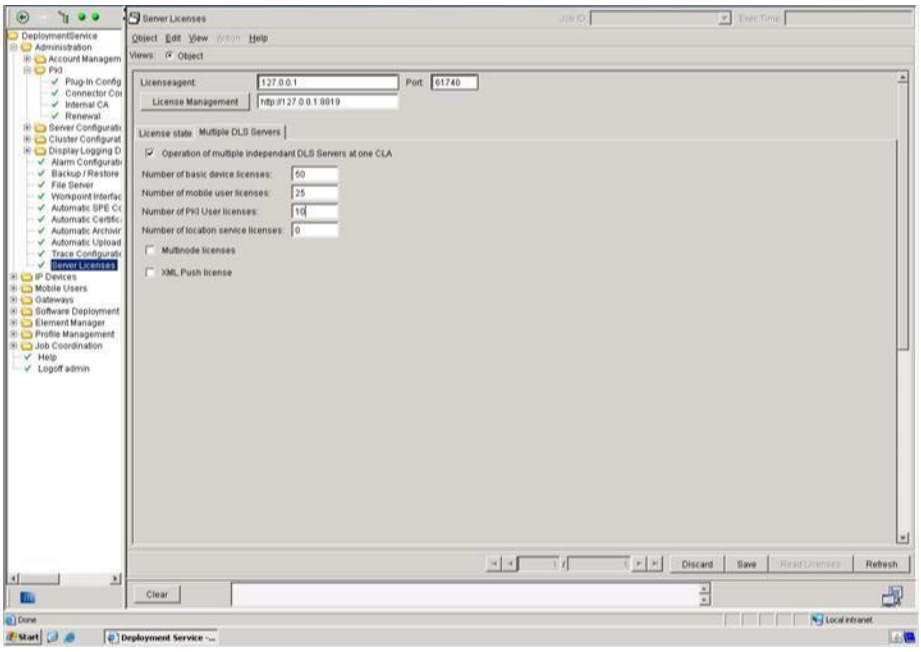


Figure 121 PKI user licenses distribution

In case of a Multi Tenant environment licenses should be distributed to selected locations accordingly. In order for PKI to properly function and dls to be able to request certificates for devices of a specific location then "PKI Connector enabled" should be checked on location level (figure 122)

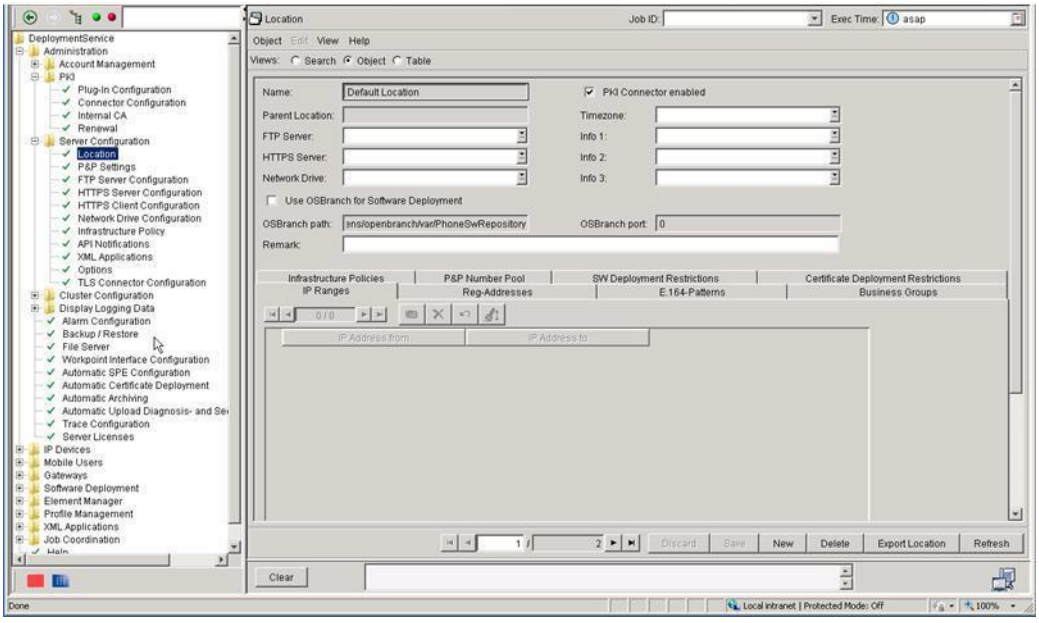


Figure 122 PKI Connector enabled on

6.3.6 Import a WBM certificate to a Phone

This is a basic task often used as the first test to verify that the freshly configured system works. Navigate to IP Devices -> IP Phone Configuration -> Security Settings
Select the tab 'WBM Server certificate' as shown in figure 123.

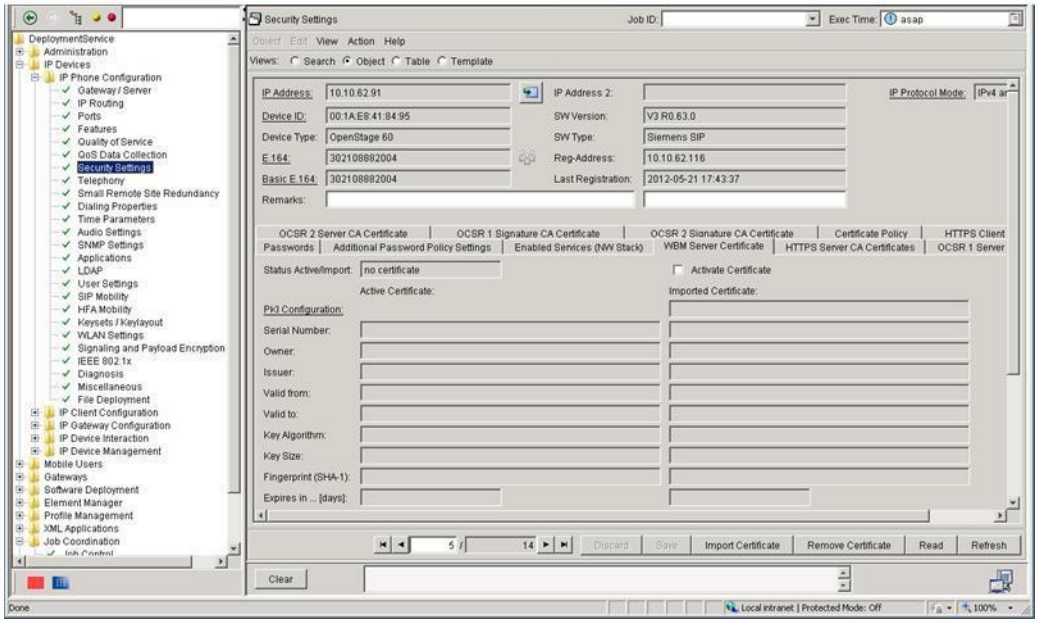


Figure 123 WBM Server certificate tab

Press 'Import Certificate' button and configure the import mask (figure 124). You can choose to request the certificate from the root or the subordinate CA configuration or you could have a specific connector configured to serve only WBM certificate requests.

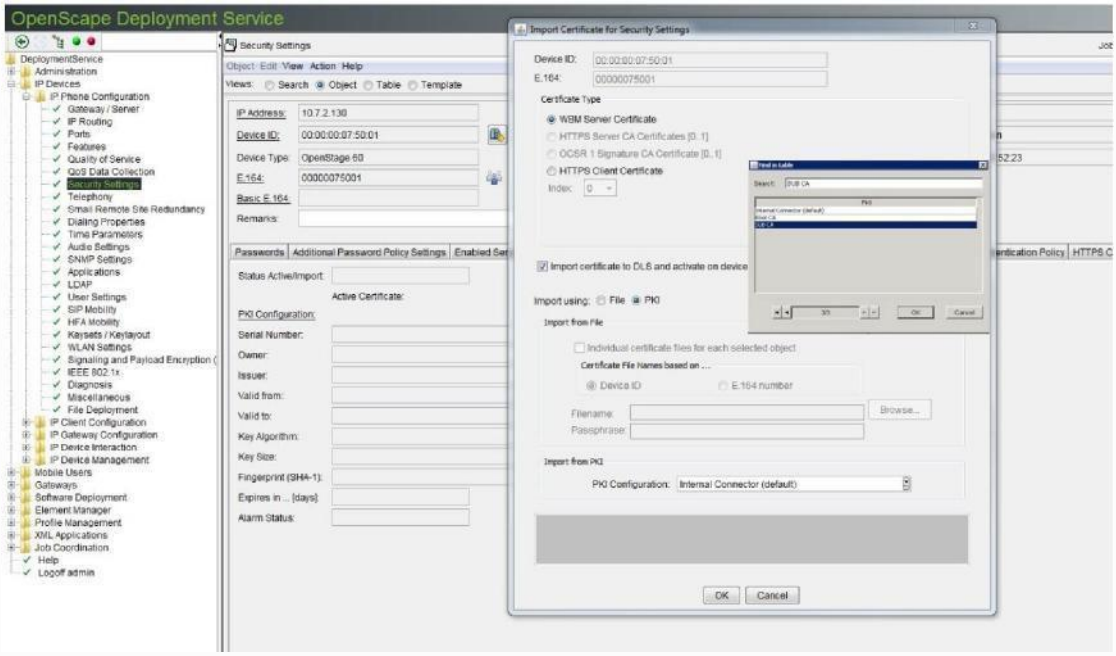


Figure 124 Import WBM certificate – select a connector

The certificate is imported to the phone via PKI and its parameters are shown in the 'Imported Certificate' column (on the right) in the figure 125.

As indicated in respective message that is displayed to message box "If the certificate had not been activated on the device automatically (1-step), check "Activate certificate" and then "Save" record"

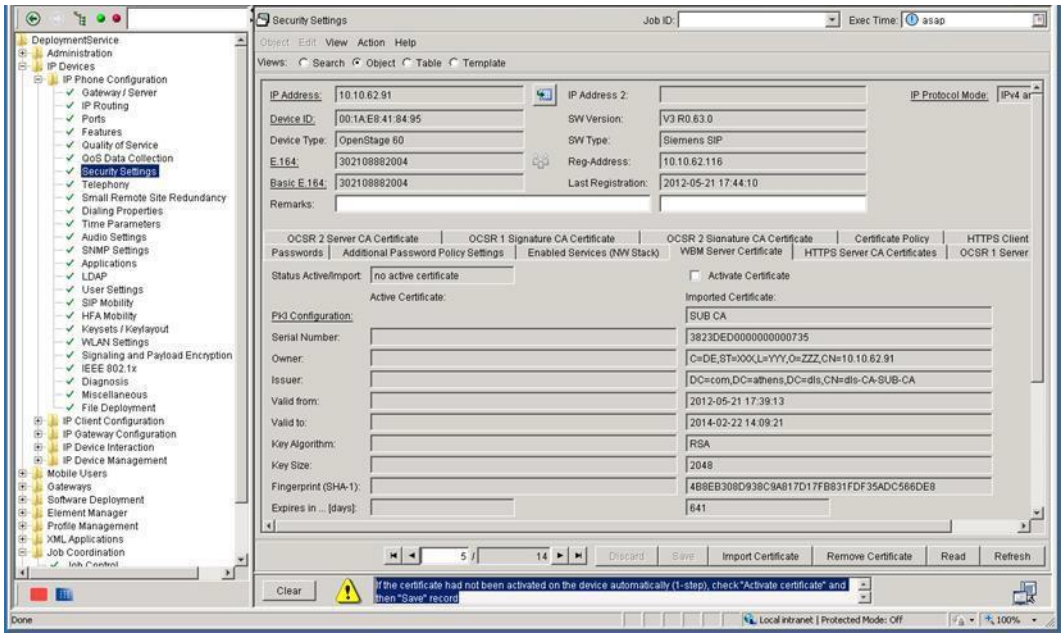


Figure 125 Imported WBM certificate

In case the "Import certificate to DLS and activate to device (1-step)" was selected on first step (during certificate import), then proceed with "Refresh" or "Read" and the certificate will be displayed in the 'Active Certificate' column (on the left) as well. Now the 'Status Active/ Import' box indicates that the imported and the activated certificates are 'equal' (figure 126).

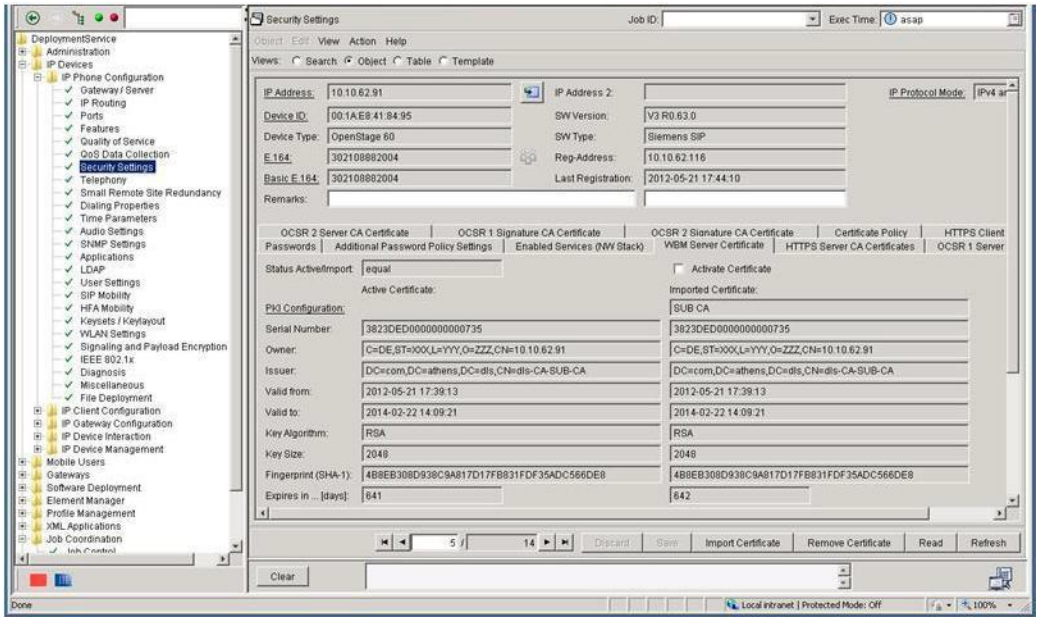


Figure 126 Activated WBM certificate

You can further verify that the WBM certificate is imported and activated by trying to access the WBM of the phone. As shown in figure 127, by clicking on the 'Certificate Error' box of the internet Explorer you can inspect the WBM certificate information (issued to the phone, issued by PKI CA, validity period as set in PKI CA).

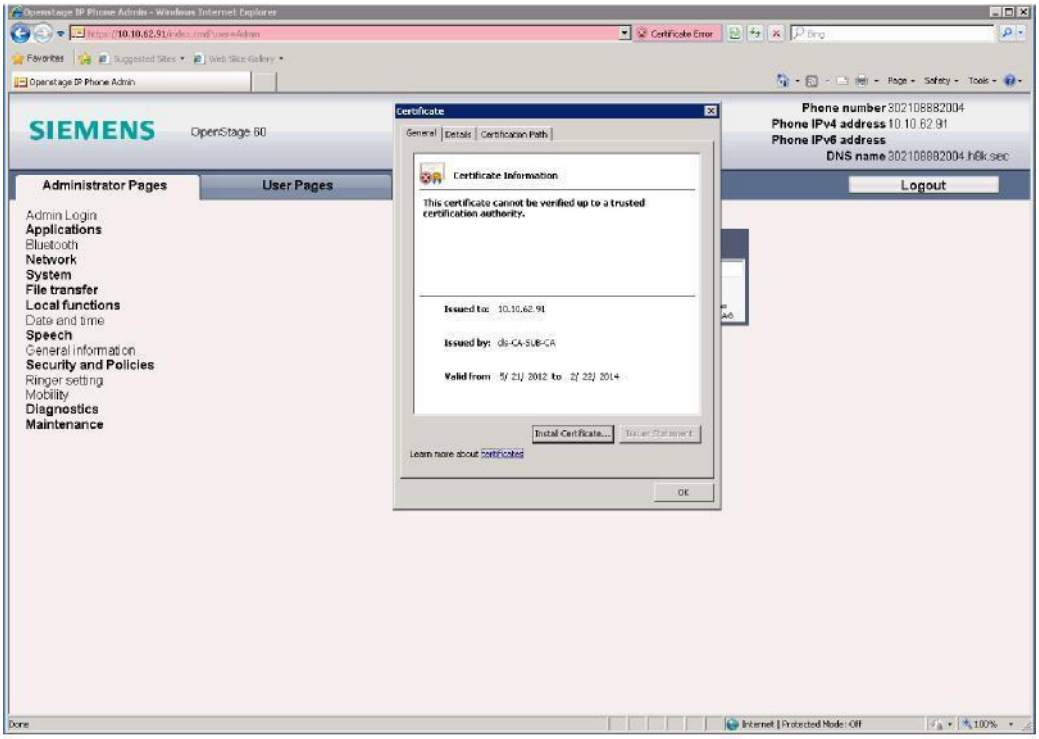


Figure 127 Phones WBM certificate information

6.3.7 Required Permissions on the Microsoft CA Server

When using the MSCA connector, the service account used by DLS requires specific permissions on the **Microsoft CA server** to successfully request and enroll certificates. These permissions fall into the following categories:

1. **Certificate Templates**

    - The DLS service account must have at least **Read** and **Enroll** permissions on the certificate templates used (e.g., *Web Server*, *Client Authentication*).

    Without these permissions, DLS cannot submit enrollment requests based on the configured templates.

2. **DCOM Permissions**

    - The DLS service account must have **Local Launch** and **Local Activation** rights on the Microsoft CA server.
    - These can be configured using dcomcnfg or by adding the account to the Windows group **Certificate Service DCOM Access**.

    Without these rights, DLS cannot request certificates successfully, even if certificate templates are accessible and a manual certreq operation succeeds.

3. **CA Security Tab (certsrv.msc → Properties → Security)**

    - The DLS service account must have the **Request Certificates** permission.
    - In most deployments, **Enroll** permission is also required.
    - **Autoenroll** is not mandatory for DLS but may be enabled if the customer's CA policies or Group Policy settings require it.

    These settings are configured in the CA Management Console under *Properties → Security*.

Note: All of the above configurations must be performed on the **Microsoft CA server**, not on the DLS server.

mitel.com