

ITProTM
SERIES

WindowsITPro  **eBooks**

Keeping Your Business
SAFE from Attack:

Encryption and Certificate Services

By Jan De Clercq

Microsoft[®]



Contents

Chapter 4: The Certificate Lifecycle Part 1: Enrollment and Key Archival and Recovery	62
Overview of the Certificate Life Cycle	62
Certificate Enrollment	63
Certificate Autoenrollment	63
Setting Up Machine Autoenrollment for Windows 2000 PKI Clients	64
Setting Up Machine and User Autoenrollment for Windows XP PKI Clients	65
Forcing Automatic Enrollment and Renewal	69
Advanced Autoenrollment Options	71
Certificate-Manager Approval	71
The Self RA Feature	72
Superseding Certificate Templates	72
“Do Not Automatically Reenroll” Property	73
How Autoenrollment Works	73
Enrollment Interfaces	75
Web-Based Enrollment Interface	77
Scripted Enrollment Options for Custom Enrollment Interfaces	78
Key Generation	79
Certificate-Request Creation	79
Requestor Identification	80
Certificate Generation	81
Certificate Publication and Distribution	81
Key Achival and Recovery	82
Manual Key Archival and Recovery	83
Automatic Key Archival and Recovery Architecture	84
Configuring Automatic Key Archival and Recovery	87
Key Recovery from the CA Database	88
Conclusion	90

Chapter 4:

The Certificate Life Cycle, Part 1: Enrollment and Key Archival and Recovery

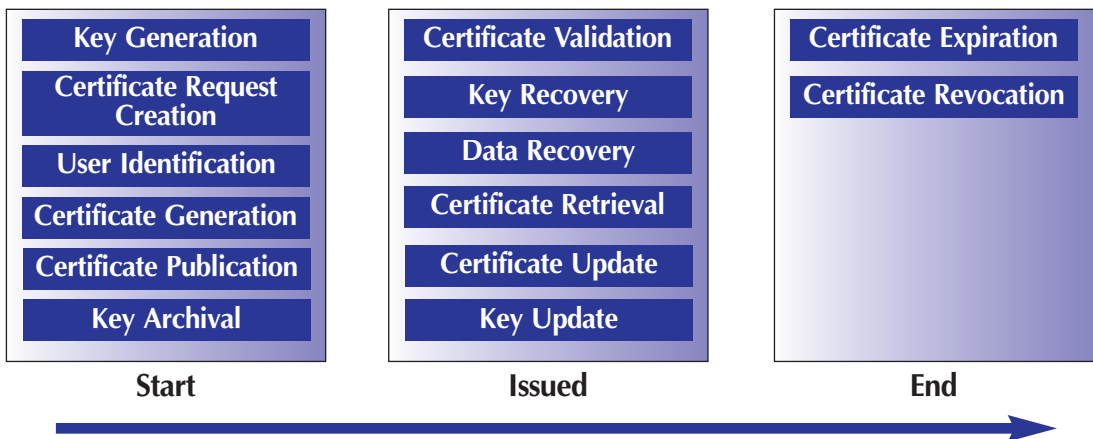
This chapter and the next focus on the Windows Server 2003 public key infrastructure (PKI) certificate life cycle and its different subprocesses. In this chapter, we concentrate on certificate enrollment, key archival, and key recovery.

Overview of the Certificate Life Cycle

The life of a certificate can be subdivided into three main phases, within which different events can occur. The phases are the start phase, the issued phase, and the end phase. Figure 4-1 illustrates the complete certificate life cycle, its different phases, and their events.

Figure 4-1

The certificate life cycle



A significant aspect of the certificate life cycle is the degree of automation for the different processes. The degree of automation is important from an end user's ease-of-use perspective and from an administrator's ease-of-management point of view. Increased automation is the main advantage of what is called a *managed PKI solution*: In a managed PKI, most processes are automated. For example, Windows 2000 PKI comes with much more automation than its predecessor, Windows NT4 PKI. The degree of automation is even higher in Windows Server 2003 PKI, and that is why we can call it a true managed PKI solution.

Certificate Enrollment

Certificate enrollment enables a user, machine, or service to participate in and use PKI-enabled applications or processes. Certificate enrollment also consists of a cycle of events: key generation, certificate request, identification, certificate generation and publishing, and encryption-key archival.

A user or an administrator can manually start certificate enrollment. In some PKI-enabled applications, the process requires an initiative from both the user and an administrator. A good example of the dual requirement is the scenario in which a Windows enterprise Certification Authority (CA) is issuing certificates. In this case, before the user can initiate certificate enrollment, the administrator first must authorize which users the machines can enroll for a particular certificate type. This authorization can be established by setting access-control permissions on the certificate templates.

In most cases, a user (whether a user, machine, or service account) initiates certificate enrollment. In Windows 2000 and Windows Server 2003, there is one exception to this rule: smart card enrollment. In the smart card enrollment model, an administrator who has a special smart card enrollment-agent certificate is allowed to enroll for a certificate on a user's behalf. He or she can also load the user's certificate on the (administrator's) smart card.

Enroll permission on the Active Directory (AD) CA object is required for a user to enroll for a certificate from a Windows 2000 or Windows Server 2003 standalone CA that's a member of a Windows domain. Before a user can enroll for a certificate from a Windows 2000 or Windows Server 2003 enterprise CA, the following conditions must be met:

- The user must have read and enroll permission on the enterprise CA level.
- This permission must be set on the AD CA object.
- The appropriate permissions (enroll and read) must be set on the certificate template.
- The correct certificate template must be configured on the CA.

Enrollment can also be initiated automatically for both user accounts and machine accounts that are part of a Windows domain environment. This feature is known as *certificate autoenrollment*. We explain autoenrollment in more detail in the next section.

In the following sections, we look more specifically at the following key elements and aspects of the Windows certificate-enrollment process:

- Certificate autoenrollment
- Enrollment interfaces
- Key and certificate request generation
- Requestor identification
- Certificate generation, publication, and distribution

Certificate Autoenrollment

Certificate autoenrollment is the Windows 2000, Windows XP, and Windows Server 2003 OSs' capability to automatically enroll users and machines for certificates. Windows 2000 PKI supports certificate autoenrollment only for machines and Encrypting File System (EFS) user certificates. Windows Server 2003 PKI extends certificate autoenrollment to users and all certificate types, greatly enhancing the PKI's ease of use. Compared to the feature set of other PKI products on the market,

certificate autoenrollment is also a unique feature that gives Windows Server 2003 PKI an important advantage over those products.

Certificate autoenrollment not only handles certificate enrollment; it also automates certificate renewal and certain certificate housekeeping tasks, such as removing revoked certificates from a user's or machine's certificate store, or downloading the trusted-root CA certificates and cross-certificates from AD. Following are some typical examples of how certificate autoenrollment is or can be used:

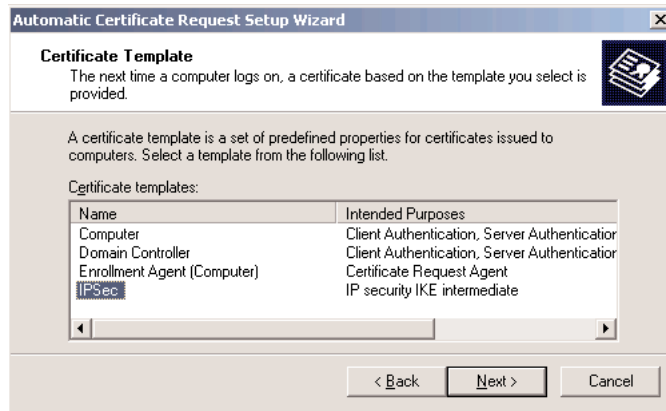
- Every Windows 2000 and Windows Server 2003 domain controller automatically gets a domain controller certificate when the machine joins a domain in which an enterprise certification authority is defined.
- An administrator can set a Group Policy Object (GPO) setting that automatically enrolls machines for an IP Security (IPSec) or Secure Sockets Layer (SSL) certificate.
- An administrator can set a GPO setting that automatically enrolls a number of users for a user or secure-mail certificate.
- When the CA administrator wants to change a property (e.g., the lifetime) of a particular certificate type, he or she can duplicate the old certificate template to create a new certificate template and let the new template supersede the old one. Autoenrollment will then automatically distribute a new certificate, based on the new template, to the concerned PKI users.

Certificate autoenrollment for users requires extra client-side code that at the time of this writing was bundled only with Windows XP and Windows Server 2003 clients. Autoenrollment requires both the machine and the user to be part of a Windows AD domain. Also, autoenrollment properties can be set only on version 2 certificate templates. And remember that only a domain with a Windows Server 2003 schema supports version 2 templates.

Setting Up Machine Autoenrollment for Windows 2000 PKI Clients

To enable machine-certificate autoenrollment from a Windows Server 2003 CA for Windows 2000 PKI clients, you must enable machine-certificate autoenrollment in the GPO Public Key Policies' Automatic Certificate Request Settings container. If you right-click this container, and then select New\Automatic Certificate Request..., the Automatic Certificate Request Setup Wizard (as Figure 4-2 shows) will start and will guide you through the machine-certificate autoenrollment process.

Figure 4-2
The Automatic Certificate Request Wizard



To enable machine-certificate autoenrollment from a Windows Server 2003 CA for Windows XP PKI clients, use the procedure outlined in the next section.

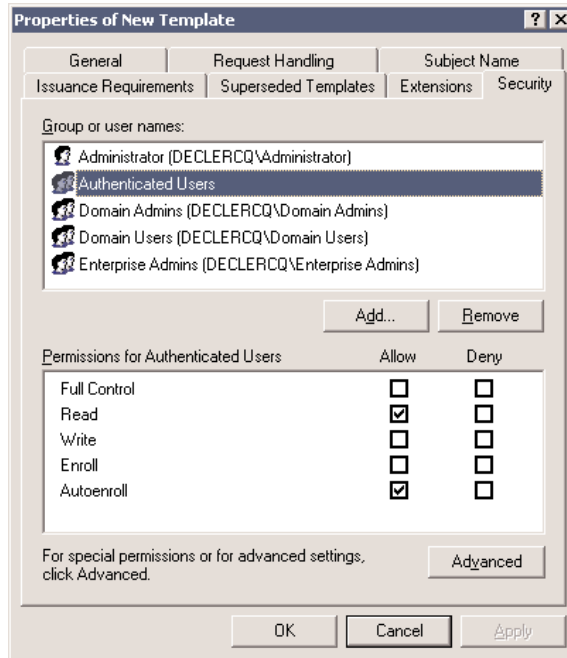
Setting Up Machine and User Autoenrollment for Windows XP PKI Clients

To set up machine- or user-certificate autoenrollment from a Windows Server 2003 CA for Windows XP PKI clients, you must make configuration changes in both the MMC Certificate Templates snap-in and the MMC Group Policy snap-in.

To enable autoenrollment at the template level, open the Certificate Templates snap-in, open the template, go to the Security tab, and set the appropriate ACL settings to give users, machines, or groups the Autoenroll permission (as Figure 4-3 shows).

Figures 4-3

Setting autoenrollment permissions on the Certificate Template Level—Security tab

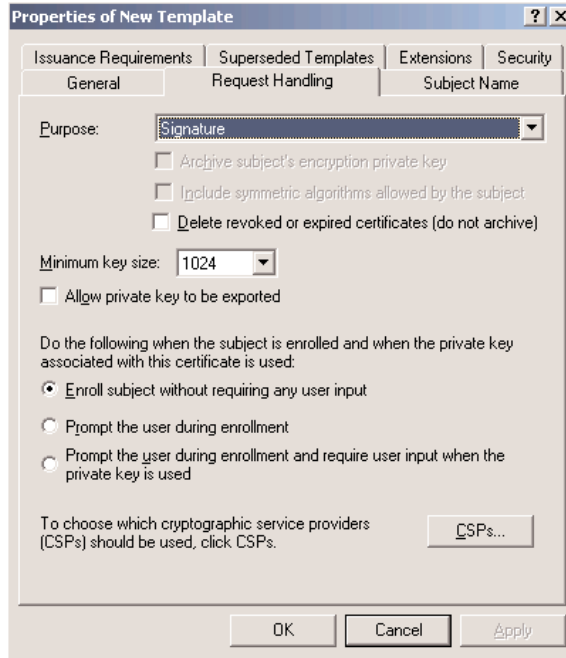


These autoenrollment properties can be set only on version 2 certificate templates. Only a domain with a Windows Server 2003 schema supports version 2 templates, and only a Windows Server 2003 Enterprise Edition or Datacenter Edition AD-integrated CA can issue certificates that are based on version 2 certificate templates.

If you want autoenrollment to occur without any PKI user intervention, leave the default settings on the Request Handling tab unchanged (as Figure 4-4 shows). If you want to prompt the user to start the autoenrollment process, click the “Prompt the user during enrollment” button.

Figures 4-4

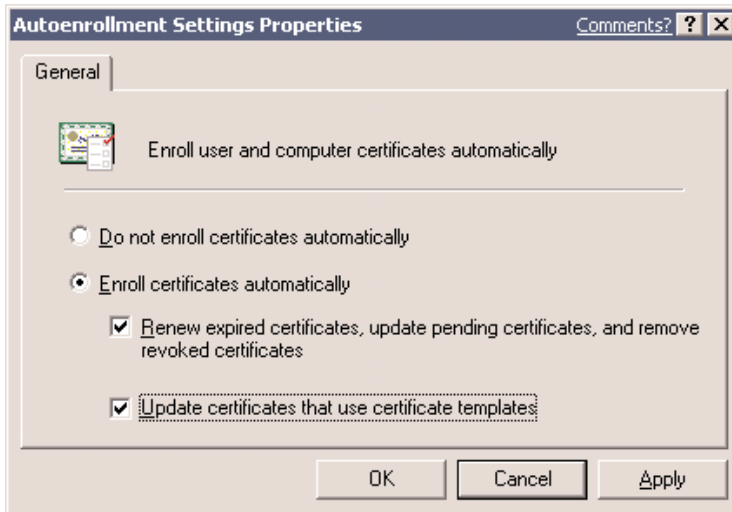
Setting autoenrollment permissions on the Certificate Template Level—Request Handling tab



Autoenrollment typically cannot be done when you are using smart cards. User input is required when you are enrolling smart card certificates. Enrolling for certificates that must be stored on a smart card requires the user to enter a smart card in the smart reader. In most cases, this enrollment also requires the user to enter a PIN code. Requiring user input on machine certificate templates will make machine autoenrollment fail.

To enable autoenrollment at the GPO level, open the Group Policy snap-in, go to Computer Configuration\Windows Settings\Security Settings\Public Key Policies, and then open the Autoenrollment Settings Properties dialog box (shown in Figure 4-5). In this dialog box, click the “Enroll certificates automatically” button, and select the “Update certificates that use certificate templates” check box.

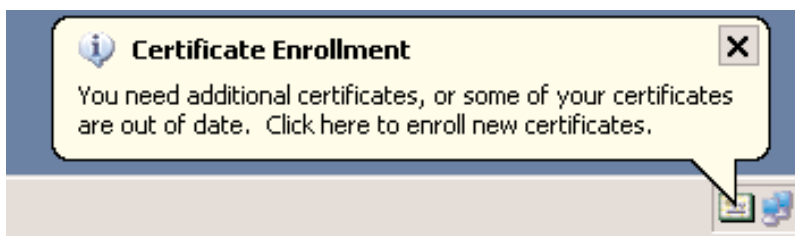
Figure 4-5
Setting autoenrollment properties at the GPO level



If you also want the autoenrollment process to take care of certificate renewal and other certificate housekeeping tasks, make sure that, in addition, you select the “Renew expired certificates, update pending certificates, and remove revoked certificates” check box.

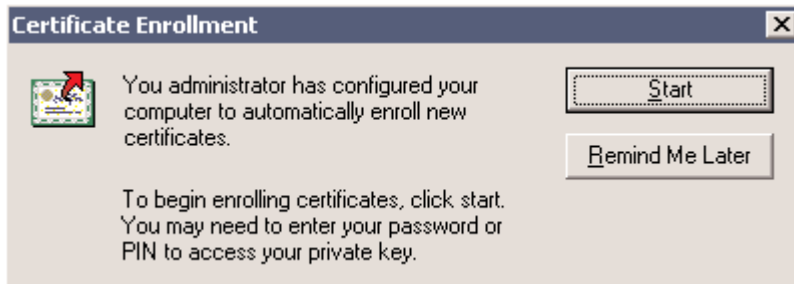
When autoenrollment occurs, and it has been set up to occur without user input, everything will happen automatically without user intervention. If it has been set up to occur with user input, a warning balloon will appear in the user’s taskbar tray (as Figure 4-6 shows).

Figure 4-6
Autoenrollment text balloon



After approximately 15 seconds, the warning balloon is replaced by a certificate icon. When the user clicks the balloon or the certificate icon, a dialog box appears and prompts the user to choose whether to start the autoenrollment process (as Figure 4-7 illustrates). If the user clicks the Remind Me Later button in this dialog box, the warning balloon will reappear at the next group-policy refresh interval, or at the next interactive logon.

Figure 4-7
User Autoenrollment Confirmation dialog box



The user-input warning balloon appears with a delay of 60 seconds after the interactive logon sequence. If you want the balloon to appear immediately after the interactive logon sequence, make the following registry hack on the user's machine: Set the HKEY_CURRENT_USER \SOFTWARE\Microsoft\Cryptography\AutoEnrollment\AEExpress registry key to value 1 (REG_DWORD).

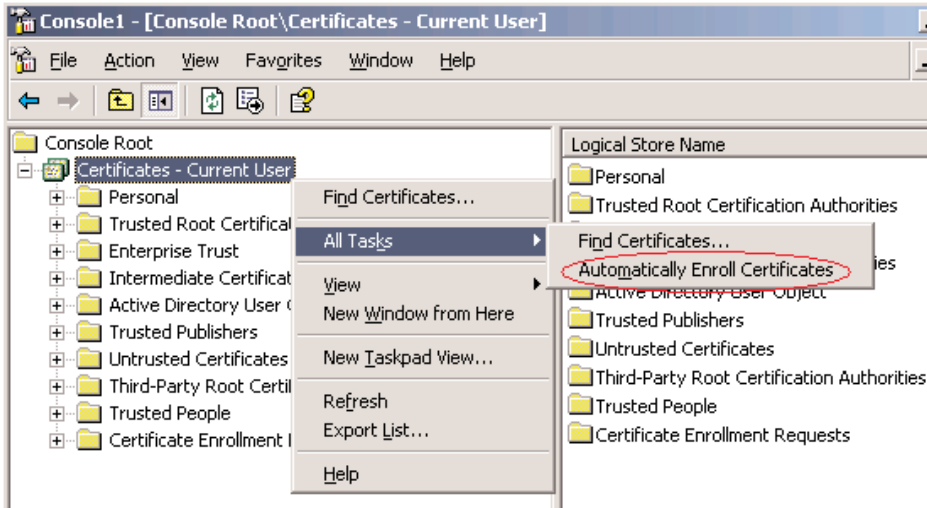
Forcing Automatic Enrollment and Renewal

You can force certificate enrollment to occur without waiting for the next logon or automatic GPO refresh. To force automatic certificate enrollment for both user certificates and machine certificates, you can manually force a group-policy update using the gpupdate.exe command-line utility. Triggering a GPO update will, in turn, trigger an autoenrollment event.

To force automatic certificate enrollment for user certificates only, open the MMC Certificates snap-in, and then open your Personal Certificates container. Then right-click the Certificates—Current User container, and select All Tasks\Automatically Enroll Certificates... from the context menu (as Figure 4-8 shows). A confirmation dialog box will appear, from which you can start the autoenrollment process, or ask to be reminded later.

Figure 4-8

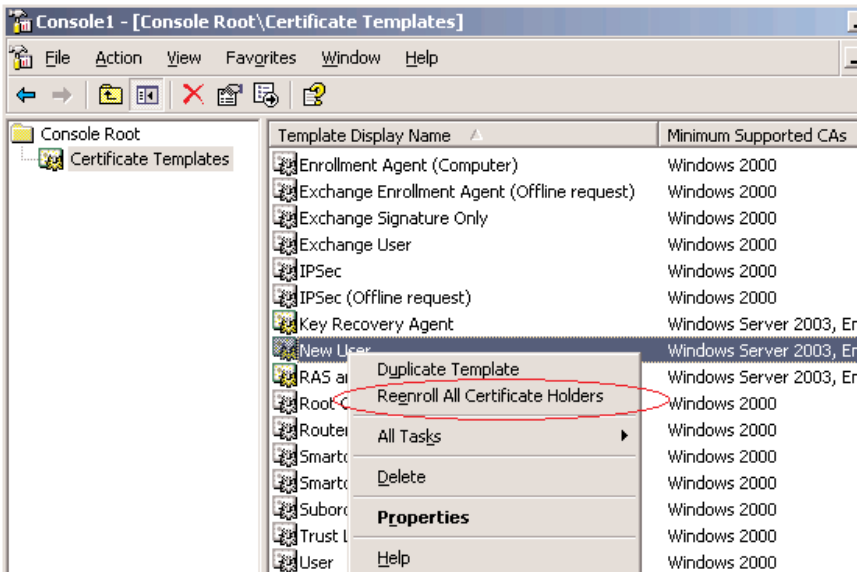
Forcing user certificate autoenrollment—user certificates only



You can also force renewal of specific user- or machine-certificate types. To do so, in the Certificate Templates snap-in, right-click the template, and select “Reenroll All Certificate Holders” (as Figure 4-9 shows).

Figure 4-9

Forcing user certificate autoenrollment—user or machine certificates



When you select this option, the version number of the certificate template is increased, and this version number update will actually trigger the autoenrollment event. To manually force a download of the root CA and cross-certificates stored in AD that are downloaded as part of the autoenrollment process, you must delete the following registry key and all subordinate keys on the client machine:

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Cryptography\AutoEnrollment\AEDirectoryCache.

Advanced Autoenrollment Options

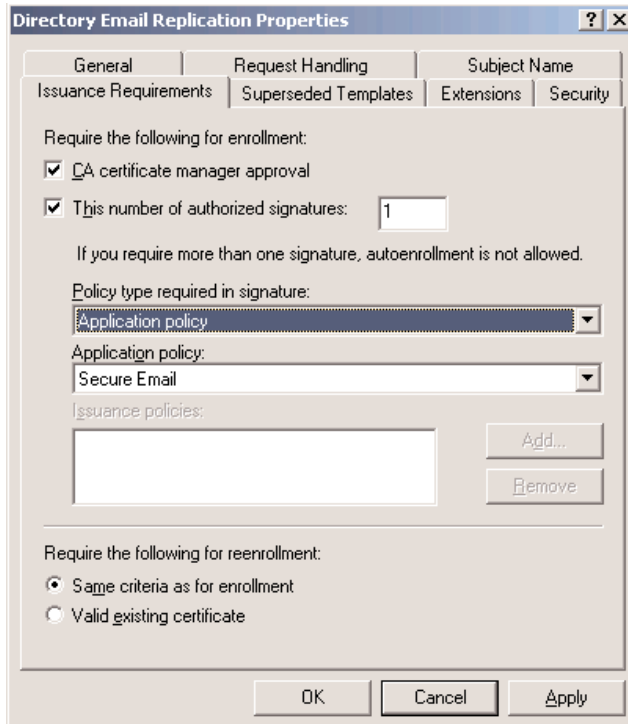
Next, we discuss some of the advanced autoenrollment options: the requirement for certificate-manager approval, the self registration authority (self RA) feature, the concept of superseding certificate templates, and the meaning of the “Do not automatically reenroll if a duplicate certificate exists in Active Directory” certificate-template property. All of these options are available only on version 2 certificate templates.

Certificate-Manager Approval

Version 2 certificate templates have a property called “CA certificate manager approval” (on the Issuance Requirements tab, as Figure 4-10 shows). If this property is set, CA manager approval is required before the CA will actually issue the certificate.

Figure 4-10

Issuance requirements in certificate template properties



Until the CA manager approves the request, it is added to the CA's Pending Requests container. This feature works in conjunction with certificate autoenrollment. The autoenrollment process will periodically check the CA for approved requests and, when it finds them, automatically install the certificates on the client machine.

The Self RA Feature

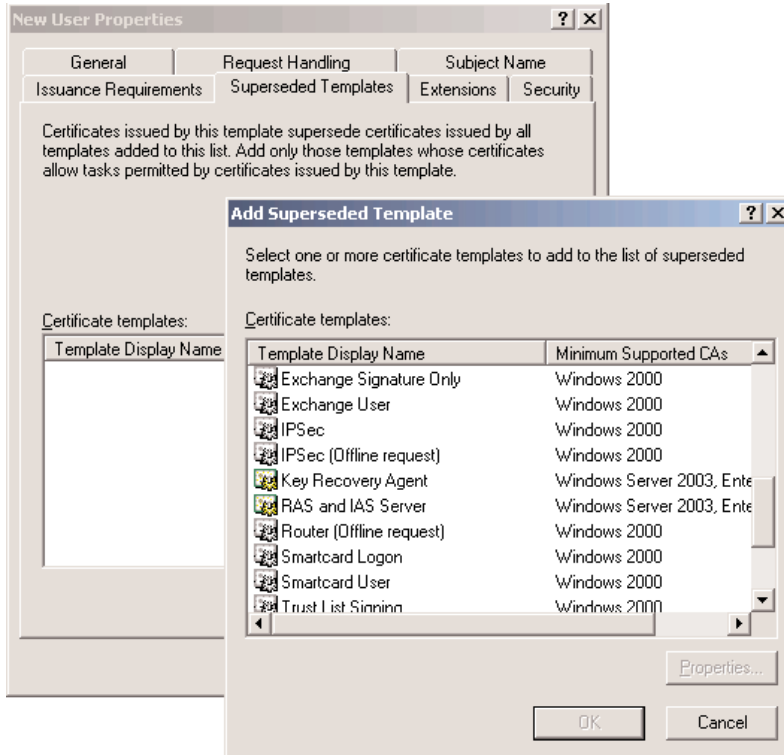
Self RA is a new Windows Server 2003 PKI feature that lets you set special enrollment requirements on version 2 certificate templates. To sign a new certificate request, self RA requires the presence of an existing—previously issued—certificate and its associated private key. Self RA is also configured from the Issuance Requirements tab, in the certificate-template Properties dialog box. Self RA can work in conjunction with autoenrollment. However, autoenrollment cannot deal with requests that require more than a single signature to authorize the enrollment request. You can set the following self RA-related properties (as Figure 4-9 illustrates):

- The number of signatures required to authorize the certificate request. Remember, when you are using autoenrollment, the number of signatures is limited to one.
- The content of the application and/or issuance policy fields in the X.509 certificate extensions of the authorization certificate.
- The requirements for automatic reenrollment. Use the same criteria as those used for the original enrollment (these are the criteria listed in the upper part of the Issuance Requirements tab), or just check to determine whether a valid certificate of the type mentioned in the certificate template is present in the PKI user's certificate store.

Superseding Certificate Templates

Superseding certificate templates lets CA administrators automatically reenroll users for certain certificate types. This option allows you, for example, to change a property of a particular certificate type (e.g., the lifetime, or the content of an X.509 extension) by issuing a new template. To set up superseding templates, use the Superseded Templates tab in the New User Properties of a version 2 certificate template (as Figure 4-11 shows).

Figure 4-11
Setting up superseding certificate templates



“Do Not Automatically Reenroll” Property

Another very useful certificate-template property related to autoenrollment is “Do not automatically reenroll if a duplicate certificate exists in Active Directory” (available from the General tab of a version 2 certificate template). When this property is enabled, autoenrollment will not enroll a user for a certificate when a similar certificate exists in the AD object of the user—even if a certificate does not exist in the My container of the user’s certificate store. In this case, the autoenrollment process will query AD to determine whether the user should be enrolled. This is an interesting option for users who do not have roaming profiles and who log on to multiple machines. Without this setting, these users would be automatically enrolled for a certificate on every machine to which they are logging on.

How Autoenrollment Works

Certificate autoenrollment (or the *autoenrollment event*, as we call it in the following explanation) is triggered by the winlogon process. The winlogon process is initiated every time an interactive logon is performed, and every time machine- or user-based group policies are applied. By default, group policies are applied every eight hours. The autoenrollment event is also triggered eight hours after the

last autoenrollment event occurred. As we mentioned above, GPO updates can also be triggered manually. Unlocking a workstation, however, does not trigger a certificate autoenrollment event.

Here is what happens following an autoenrollment event:

1. The client OS queries AD to download the content of a set of predefined certificate stores to the local store on the client machine. These stores include the NTAUTH, the trusted-root CA, certificate templates, and Authority Information Access (AIA) (for cross-certificates and subordinate CA certificates) AD containers.
2. The autoenrollment process processes the certificate templates, analyzes their properties, and creates a list—referred to as the requirements list—of tasks to be done during the autoenrollment event. The requirements list includes
 - Certificate enrollment tasks. All templates that have autoenroll and read permission set for the current machine or user will be added to the requirements list.
 - Certificate renewal tasks. The autoenrollment process processes the user's or machine's Personal certificate store container to look for expired certificates or certificates that are about to expire; it will add these certificates to the requirements list. Automatic certificate renewal starts when 80 percent of the certificate lifetime has passed, or when the renewal interval period specified in the certificate template has been reached. The renewal interval period is specified on the General tab of a version 2 certificate template.
 - Certificate enrollment tasks based on template-supersede rules. The autoenrollment process evaluates certificate template-supersede rules and makes the appropriate additions and deletions to the requirements list.
3. The autoenrollment process then searches AD for an enterprise CA that can issue the certificates.
4. If a CA is found, the autoenrollment process will pass the requirements list to the CA using certificate enrollment and renewal requests.
5. The CA will process the certificate enrollment and renewal requests.
6. If a certificate is issued from the CA, the autoenrollment process will install the certificate in the My container of the user's or machine's certificate store. If the certificate's state is set to "pending" (for certificate requests that require administrator approval), the autoenrollment process will save the certificate request information in the Pending Request container of the user's or machine's certificate store.
7. At the end of the autoenrollment process, the outcome (success or failure) of the process will be logged in the local system's application event log. If the autoenrollment failed, a summary dialog box will appear.
8. For requests that were set to "pending," the autoenrollment process will regularly query the CA to check whether the request has been approved. This query will happen every time the group policies are refreshed. Autoenrollment will even retrieve pending certificates that were enrolled manually through the CA Web interface.

As pointed out earlier, the autoenrollment event triggers more than just certificate autoenrollment:

- Trusted-root CA certificates are downloaded from the AD-based Certification Authorities and NTAAuth stores to the local machine's Trusted Root Certification Authorities certificate store container. The autoenrollment process doesn't download the complete NTAAuth store; only the differences between the content of the user certificate and the NTAAuth store are downloaded from the NTAAuth store.
- Cross-certificates and subordinate CA certificates are downloaded from AD to the local machine certificate store. As for trusted-root CA certificates, only the differences are downloaded.
- The autoenrollment process enumerates the pending certificate requests in the Pending Request container of the user's certificate store. It downloads the certificate, once it has been issued, from the issuing CA and installs it in the user's certificate store. If the request has been pending for more than 60 days, the autoenrollment event removes it from the Pending Request container in the user's certificate store.
- The autoenrollment process also deletes expired and revoked certificates in the userCertificate attribute of the user's AD object and in the user's local certificate store. Expired and revoked certificates are deleted from the user's local certificate store only if the "Delete revoked or expired certificates" property has been set on the Request Handling tab of the certificate template Properties dialog box.

Enrollment Interfaces

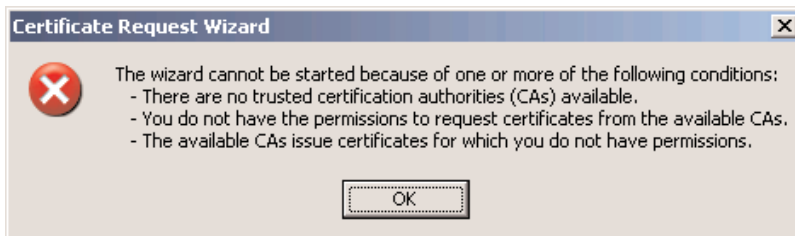
Both Windows 2000 PKI and Windows Server 2003 PKI can support four different certificate-enrollment interfaces—a Web interface, a GUI interface, a command-prompt interface, and a scripting interface:

- Web interface. To use a Web browser to enroll for certificates from an enterprise CA or a standalone CA, the user can go to the following URL: <http://<CAservername>/certsrv>. This is the default URL—you can change it using the MMC Internet Information Services (IIS) Manager snap-in.
- GUI interface. A user can enroll for certificates from an enterprise CA or a standalone CA using the Certificate Request Wizard (as Figure 4-12 shows) that's available from the Certificates snap-in. GUI-based enrollment will work only if the client-side PKI logic can find a CA object in AD, and if the user's machine is joined to an AD domain. A message like the one illustrated in Figure 4-13 will be displayed.

Figure 4-12
Certificate Request Wizard



Figure 4-13
Certificate Request Wizard error message



- Command-line interface. A user can enroll for certificates from an enterprise CA using the command-prompt utility `certreq`. He or she can also use `certreq` to retrieve approved pending certificate requests from a CA. To enroll for certificates using the command-line interface, use the following `certreq` syntax:

```
CertReq [-Submit] [Options] [RequestFileIn [CertFileOut [CertChainFileOut
[FullResponseFileOut]]]]
```

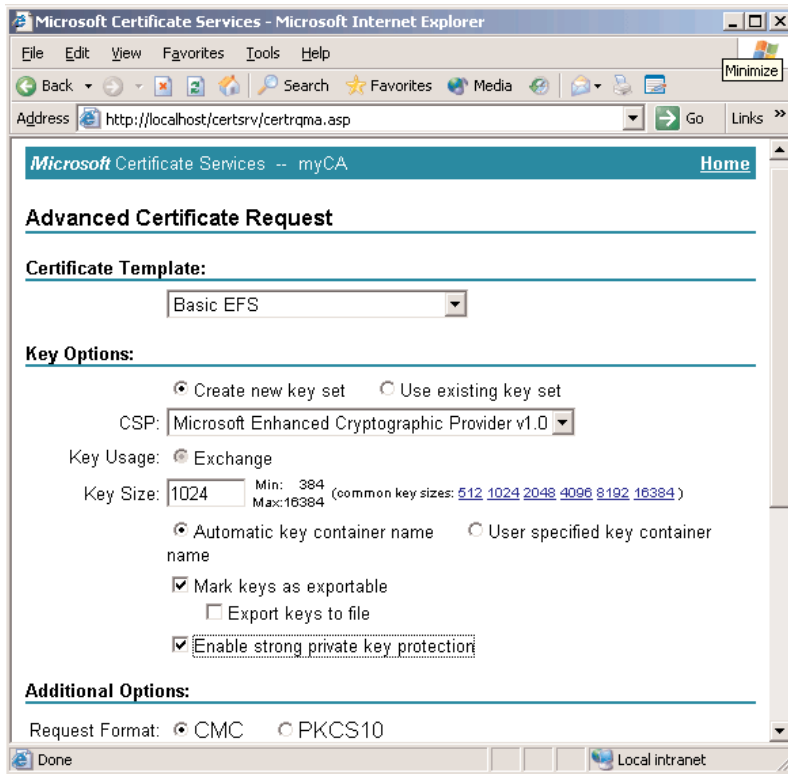
```
CertReq [-Retrieve] [Options] RequestId [CertFileOut [CertChainFileOut
[FullResponseFileOut]]]
```

- Scripting interface. You can use CAPICOM in conjunction with the `xenroll.dll` DLL to create custom certificate-enrollment interfaces, which we explain in more detail in the following section.

Web-Based Enrollment Interface

The Web-based enrollment interface (as Figure 4-14 shows) is made available through the Certificate Services Web Enrollment Support, an installation option that is selected by default when you are installing a CA on a server that has IIS installed.

Figure 4-14
Web-based enrollment interface



The Web server hosts the `Certsrv` virtual directory and application, and the `Certcontrol` and `CertEnroll` virtual directories. All of these directories are automatically created during the Web-interface installation process. You can also use the `certutil.exe` command line tool with the `-vroot` switch to manually create the CA virtual directories.

The CA with which the Web interface is communicating should not necessarily be on the same machine as the Web interface itself. When you are installing the Web interface on a server that does

not have a CA installed, the installation program will prompt you for the name of a CA server to which the interface must point. This means you can deploy multiple CA Web interfaces on the same machine or on a different machine that is pointing to the same certificate server. These options give you an elegant way to deal with, for example, the different language requirements of your PKI clients: Just deploy one Web interface per language that you need to support.

The CA Web interface is built on Active Server Pages (ASP) that detects the client's browser type. If ASP detects Internet Explorer, it will use the Certificate Enrollment Control and its ActiveX controls (explained later). If ASP detects a Netscape browser, it will generate a key and request through the Netscape-specific KeyGen method.

When you use the Web-enrollment interface in Windows Server 2003, you will notice that the inclusion of the Microsoft Internet Explorer (IE) Enhanced Security Configuration in Windows Server 2003 brings along some interesting error and warning messages (like the one that Figure 4-15 illustrates).

Figure 4-15

Web-enrollment warning message (following the IE-enhanced security configuration)



Scripted Enrollment Options for Custom Enrollment Interfaces

In case your organization has special enrollment requirements, Windows 2000 and Windows Server 2003 make it relatively easy for a developer to modify existing certificate-enrollment interfaces, or to develop custom certificate-enrollment interfaces.

You can use the CAPICOM automation object for these tasks. You can use CAPICOM for signing enrollment requests or adding multiple signatures to a CMC-formatted (CMC stands for “Certificate Management Messages over CMS”) enrollment request. CAPICOM is not installed by default on a Windows Server 2003 platform. You can download CAPICOM version 2.1.0.1 (cc2rinst.exe) from the Windows Platform SDK Redistributables Web site at <http://www.microsoft.com/downloads/details.aspx?displaylang=en&familyid=860ee43a-a843-462f-abb5-ff88ea5896f6>.

Windows 2000 and Windows Server 2003 also come with two reusable software modules that can be called from any customized certificate-enrollment application: the Certificate Enrollment Control and the Smartcard Enrollment Control. The Certificate Request Wizard, discussed earlier, uses the Certificate Enrollment Control.

Both modules are located in the %windir%\system32\certsrv\certcontrol\x86 directory on the server side and in the %windir%\system32 directory on the client side. A client-side copy of the controls is needed to enable the module to write to the local certificate stores. The Certificate Enrollment Control and the Smartcard Enrollment Control are extensively documented in the Windows 2000 security-platform software development kit (SDK).

The Software Enrollment Control module (screnl.dll) enables an administrator who has an enrollment-agent certificate to request certificates on behalf of other users and to store those certificates on a smart card.

Key Generation

Certificate enrollment starts with key generation. During the key-generation process, an asymmetric key pair, or a private key and a public key, are generated. Most PKI-enabled applications use a single-key pair; some use a dual-key pair, or even a triple-key pair. Secure mail applications that are providing key recovery, for example, use a dual-key pair to enable both the support for encryption key recovery and nonrepudiation services based on the user's signing key. Both services have different requirements.

Nonrepudiation and digital signatures require that the access to and the use of the signing private key is strictly limited to a single user. Key recovery, alternatively, requires the private decryption key to be archived in some centralized database. In a triple-key-pair model, one key pair is used to identify a user, one is used for digital signatures, and another one is used for data encryption.

Once the private key has been generated, it must be stored in a secure place, such as a secure file-system location (e.g., a DPAPI-secured part of the user profile) or, in the best case, a smart card.

In Windows 2000 and Windows Server 2003, the tasks of key generation, secure private-key storage, and public-key embedment in a certificate request are all performed by the Certificate Enrollment Control (xenroll.dll) and Smartcard Enrollment Control (screnl.dll) (both modules were explained in the previous section). Next, we discuss the generation of certificate requests.

Certificate-Request Creation

When the public key is prepared for certification, it is embedded in a certificate request. In addition to the public key, a certificate request contains the identity of the certificate requestor and some other request attributes that depend on the type of CA to which the request is sent. Commonly used certificate-request formats that are supported in both Windows 2000 PKI and Windows Server 2003 PKI are PKCS 10 and PKCS 7. More information about PKCS 10 is available from <http://www.rsasecurity.com/rsalabs/node.asp?id=2132>. More information about PKCS 7 is available from <http://www.rsasecurity.com/rsalabs/node.asp?id=2129>. Windows Server 2003 PKI also supports the Certificate Management messages over CMS (CMC) syntax. CMC is defined in RFC 2797 (available from <http://www.ietf.org/rfc/rfc2797.txt>). CMS stands for Cryptographic Message Syntax and is defined in PKCS7 (you can find more information on PKCS7 at <http://www.rsasecurity.com/rsalabs/node.asp?id=2129>).

There are three ways to look at the content of a Windows 2000, Windows XP, or Windows Server 2003 certificate request as it is submitted to a CA:

- On the PKI user side—Open the Certificate Enrollment Requests certificate container in a user's certificate store.
- On the CA side: —Open the Pending Requests container, right-click a pending request in the right pane of the MMC Certification Authority snap-in, and select All Tasks\View Attributes/Extensions....
- For a request saved as a *.req file—You can look at the certificate-request content using the certutil.exe command-line tool.

Windows 2000 and Windows Server 2003 also support the Simple Certificate Enrollment Protocol (SCEP), a certificate-enrollment protocol developed by Verisign and Cisco. The support for SCEP enables, for example, a Cisco router to enroll for an IPsec certificate with a Windows 2000 or Windows Server 2003 CA. You can find more information on SCEP at http://www.cisco.com/warp/public/cc/pd/sqsw/tech/scep_wp.htm. You can download the SCEP support add-on for Windows Server 2003 from the following URL: <http://www.microsoft.com/downloads/details.aspx?FamilyID=9f306763-d036-41d8-8860-1636411b2d01&DisplayLang=en>.

To enable a Windows 2000 or Windows Server 2003 CA to support SCEP, you must install the `mscep.dll` (an Internet Server API [ISAPI] filter) that comes with the Windows 2000 or Windows Server 2003 resource kit. A detailed setup procedure is also available in the resource kit, and in the Q249125 Microsoft Knowledge Base article titled “Using Certificates for Windows 2000 and Cisco IOS VPN Interoperation,” available at <http://support.microsoft.com/?kbid=249125>.

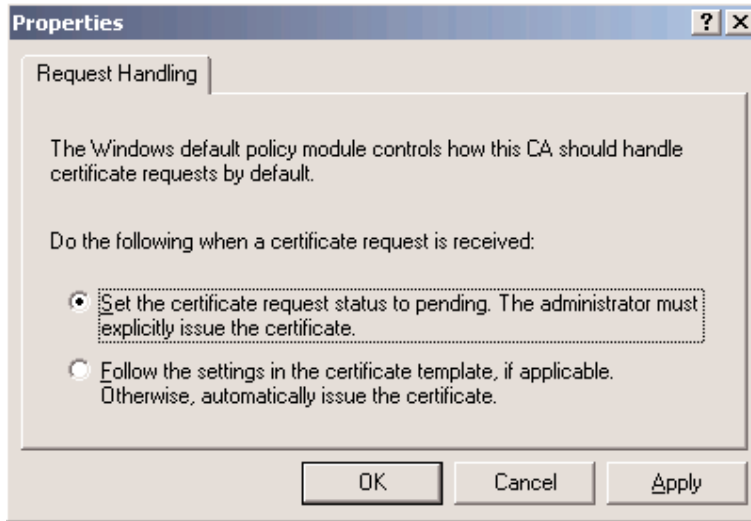
Requestor Identification

Before the certificate is actually generated by the CA, the CA will validate the certificate request (this includes validating the request format) and identify the requesting user. Identification is an often-forgotten, critical step of certificate enrollment. Identification means that the CA will check whether the entity requesting the certificate is really the entity mentioned in the request message, and whether the requesting entity possesses the private key corresponding to the public key in the certificate request.

In Windows 2000 and Windows Server 2003, the identification method depends upon the enrollment interface the PKI user uses to enroll for a certificate. When users enroll for certificates using the Certificate Enrollment Wizard, the CA authenticates users by means of the Kerberos protocol. The CA will use the NT LAN Manager (NTLM) protocol to authenticate a client when the client does not support Kerberos or when there is no Kerberos Key Distribution Center (KDC) available. The authentication protocol that is used when a user enrolls through a Web interface depends on how authentication has been configured on the CA Web directory (`certsrv`).

CA administrators who want to employ user-identification methods other than the ones that are based on the Windows and IIS authentication protocols can configure the Windows CA to require CA certificate-manager approval before the certificate is actually issued; this means that the request will go to a pending state when it arrives at the CA. As part of the CA certificate-manager-approval process, the manager can then use a custom method (e.g., face-to-face identification, or identification based on the user's driver's license) to identify the user. After the manager has used the custom method to uniquely identify the user, he or she can then manually issue the certificate. When you are using an enterprise CA, certificate-manager approval can be configured on the certificate-template level. A standalone CA by default puts all incoming requests in a pending state. This default can be changed from the policy Properties dialog box of the CA object (as Figure 4-16 shows).

Figure 4-16
Changing a standalone CA's policy properties



Certificate Generation

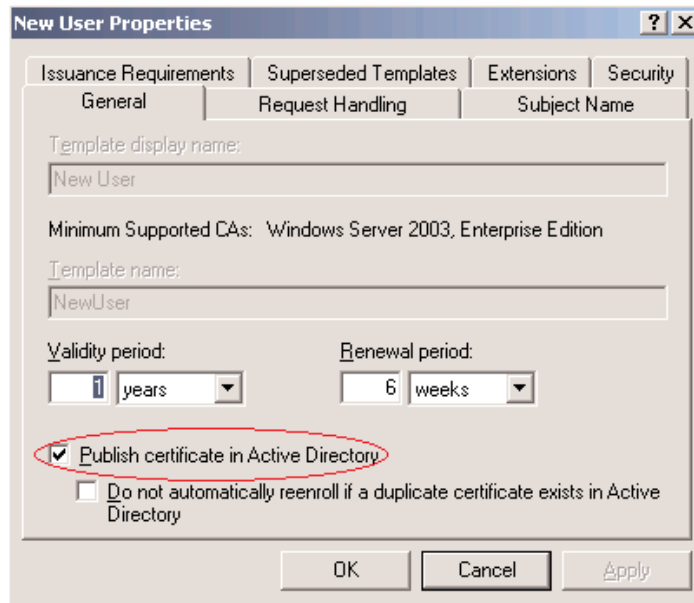
During certificate generation, the CA will sign the certificate content using the CA's private key. The content of a certificate depends on the content of the certificate request. In the case of an Enterprise CA, some of the requestor-specific data are retrieved from the AD. The layout of the certificate is defined in certificate templates that the CA retrieves from the AD or from the registry.

Certificate Publication and Distribution

Once a certificate has been generated, it can be published and distributed. Most PKI systems publish the certificate in a directory. In Windows 2000 and Windows Server 2003, certificate publication to AD depends on the certificate template setting:

- On version 1 certificate templates, the AD publication behavior cannot be changed.
- On version 2 certificate templates, AD publication depends on a changeable template property, Publish certificate in Active Directory. You can change this property from the General tab of the certificate template Properties dialog box (as Figure 4-17 shows).

Figure 4-17
Certificate template property for certificate AD publication



To publish certificates to locations other than AD (e.g., other Lightweight Directory Access Protocol [LDAP] directories or Web sites), you can develop custom exit modules that can be plugged into the Windows CA architecture. In case you want to do this, remember that no matter what exit modules you create and install, certificates will not be published unless you have specified the publication location in the certificate request.

To distribute the certificate to the user, a Windows enterprise CA sends a copy to the user using a CMC- or PKCS7-formatted message. Certificates returned from the CA are then automatically installed in the user's certificate store. A standalone CA stores the newly generated certificate on the CA's Web site, from which the user can then retrieve it.

An interesting detail related to certificate publication is that the machine account of the server hosting the CA must be a member of the Cert Publishers global group of every domain in the forest. Members of this group are the only ones who can write certificates to the usercertificate attribute of AD user objects.

Key Archival and Recovery

Key archival and key recovery are PKI services that are used to recover lost or simply unavailable private encryption keys. Encryption-key archival and recovery is a major requirement in PKI-enabled applications that are dealing with persistent data; good examples are secure mail applications.

Microsoft first introduced automatic and centralized private-encryption-key archival and recovery in the Key Management Service (KMS), which is part of the Secure MIME (S/MIME)-based secure mail application that ships with Microsoft Exchange Server.

Windows Server 2003 PKI archival and recovery builds on the central database concept as it is provided by the Exchange KMS: It includes an automated and centralized key archival and recovery service with every Windows Server 2003 enterprise CA. Key archival and recovery can also be done manually by the PKI user.

A very important detail related to the KMS archival service is that the latest Exchange release—Exchange 2003—does not come with a KMS service anymore. This means that, if you have an operational KMS in an Exchange 2000 environment, and you plan to migrate to Exchange 2003, you must migrate the KMS key-archival database to the Windows Server 2003 CA key-archival database.

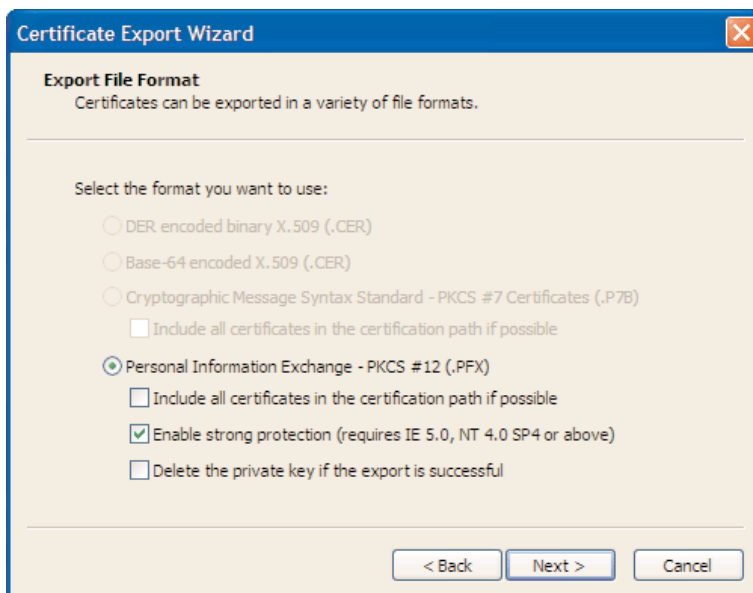
Manual Key Archival and Recovery

There are different ways for a PKI user to manually back up his or her private encryption keys. The preferred and most commonly used format in which to store archived private keys is PKCS 12. Access to and confidentiality of a PKCS 12 file's (*.pfx) content can be secured using a password. To manually back up your private keys to a PKCS 12 file, you can do one of the following:

- Open your personal certificate store using the Certificates snap-in. Right-click the certificate whose private key you want to manually back up, and select All Tasks\Export.... This action brings up the Certificate Export Wizard. Make sure that you check the “Yes, export the private key” option, and that you enable strong key protection (as Figure 4-18 shows). Selecting the “Enable strong protection...” option will make the Wizard prompt you for a password to protect the PKCS 12 file's content. Do not check the “Delete the private key if export is successful” option because this would delete your private key from your system.

Figure 4-18

Backing up the private key using the Certificate Export Wizard



- From IE: Select Internet Options from the Tools menu. In the Internet Options dialog box, go to the Content tab. Click the Certificates button to bring up the Certificates dialog box. Select the certificate whose private key you want to export, and click Export.... This action will also bring up the Certificate Export Wizard. Then use the same options as mentioned above.

You can also archive your private encryption keys from Microsoft Outlook. Outlook does not store the keys in a PKCS 12-formatted file, though; it uses a special Outlook export format (*.epf). Like PKCS12, this format can be secured using a password that is provided by the user. To export your private encryption keys from Outlook, select Options in the Tools menu, and go to the Security tab. At the bottom of the Security tab, click the Import\Export button. Doing this brings up the Import\Export Digital ID dialog box. In this box, click the “Export your Digital ID to a file” button. Select the Digital ID whose private key you want to export, and fill in a filename and password. Once more, make sure that you do not select the “Delete Digital ID from system” check box.

The .epf extension shows the historical roots of Outlook’s secure mail technology: “epf” stands for *Entrust profile*. One reason Outlook still uses this format is that it supports X.509 version 1 certificates, which were used in the early Exchange KMS implementations.

Automatic Key Archival and Recovery Architecture

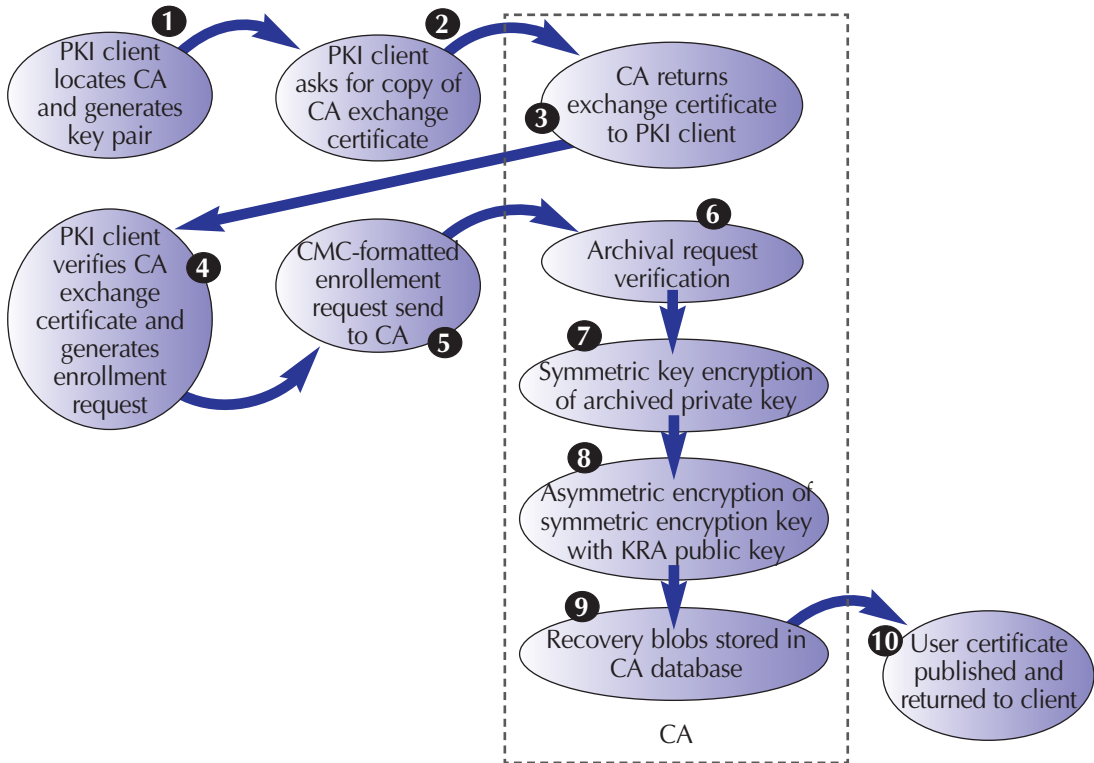
A Windows Server 2003 CA securely stores the archived private keys in the CA database. The CA uses a symmetric 3DES key to encrypt a private key, and then uses a key-recovery agent’s public key to encrypt the symmetric key. A new symmetric encryption key is randomly generated for every new key-archival request.

A key-recovery agent (KRA) is the owner of a key-recovery certificate and private key. For advanced security reasons, both the KRA certificate and private key can be stored on a smart card. This option is supported in Windows Server 2003 PKI.

If more than one KRA is defined, the symmetric encryption key will be encrypted with each KRA’s public key. Windows Server 2003 PKI comes with a predefined key-recovery-agent certificate template, so it is relatively easy to set up a key-recovery certificate for a particular account.

The key-archival process occurs in a completely transparent way for the PKI user, as part of the certificate-enrollment process. Whether or not a private key is archived depends on a certificate template setting that we will explain next.

During a certificate enrollment that incorporates an automatic key archival, the following steps occur (Figure 4-19 illustrates this process):

Figure 4-19*Windows Server 2003 key-archival process*

1. The PKI client queries AD for a CA. It specifically looks for CA entries in the Enrollment Services container in the configuration-naming context. AD returns the name and location of CA.
2. The PKI client requests the CA for a copy of its CA exchange certificate.
3. The CA returns the CA exchange certificate to the client.
4. The PKI client validates the CA exchange certificate. It verifies the signature, performs a revocation check, and validates the certificate format.
5. The PKI client encrypts the private key that must be archived with the CA exchange certificate's public key. This encrypted blob is then embedded in a CMC-formatted request object and forwarded to the CA.
6. The CA decrypts the encrypted private key of the client with the private key associated with the CA's exchange certificate. The CA then encrypts the private key with a random 3DES symmetric key.
7. The CA checks whether the private key in the CMC cryptographically pairs with the public key in the certificate request. The CA also validates the signature on the request, using the public key that comes with the request.

8. Finally, the CA encrypts the symmetric key with the public keys of one or more KRAs, based on the CA configuration.
9. The CA saves, in the CA database, the encrypted blob that contains the encrypted private key, and the symmetric key encrypted with one or more KRA public keys.
10. The CA processes the certificate request. The CA then forwards the certificate to the user and publishes it in the directory (if this option has been set in the certificate template).

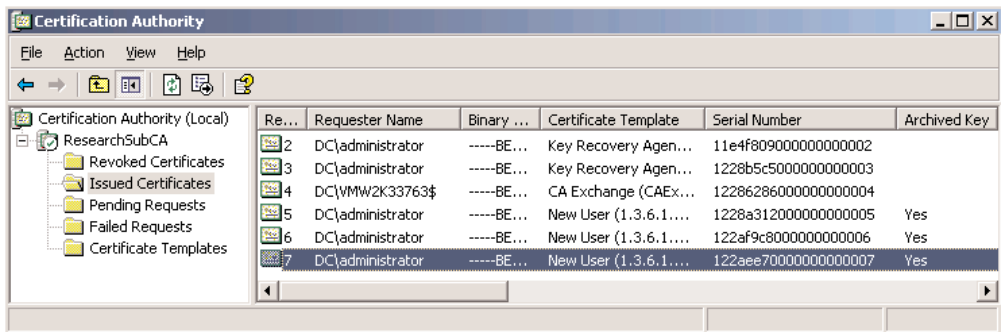
In this process, the CA exchange certificate is used to provide confidentiality and integrity protection when the PKI client's private key is forwarded to the CA for archival. The CA exchange-certificate's content is defined in a new certificate template that comes with Windows Server 2003. A CA's CA exchange certificate is physically stored in the attributes of the CN=<CAName>-Xchg,DC=<domainname> AD object. The CA's private key is stored in a secured part of the CA server's registry. For obvious security reasons, the CA exchange certificate and key pair have a very short lifetime (seven days). If this new template is not available, a set of predefined values stored in the registry is used to define the content of the CA exchange certificate.

The Windows Server 2003 CA stores the encrypted private key in the CA database's RawArchivedKey column and stores the encrypted symmetric key in the KeyRecoveryHashes column. You can view these columns and the rest of the CA database's schema from the command line by typing the following certutil command at the command line:

```
certutil schema
```

Whether or not the private key of a given certificate is archived in the CA database can also be seen from the Certification Authority snap-in. To view this information, you must change the columns that are displayed in the Issued Certificates CA container: You must add the Archived Key column (as Figure 4-20 shows). To add this column, right-click the Issued Certificates container, and then select View/Add/Remove columns.... Then add the archived key in the available column's list.

Figure 4-20
Archived key column in CA interface

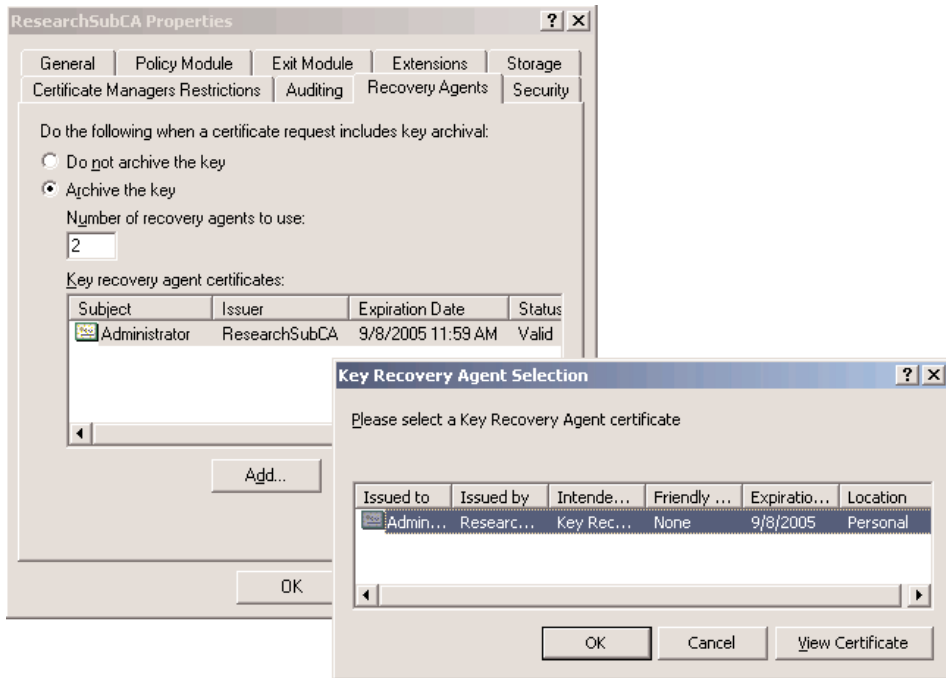


Configuring Automatic Key Archival and Recovery

Configuring automatic key archival and key recovery in Windows Server 2003 requires configuration changes in the CA object's and the certificate template properties. Here's how to make those changes:

To configure a CA object's key-archival settings, open the Certification Authority snap-in, open the CA object's Properties dialog box, and then go to the Recovery Agents tab, which Figure 4-21 shows. Click the "Archive the key" button to enable key recovery.

Figure 4-21
CA key recovery settings

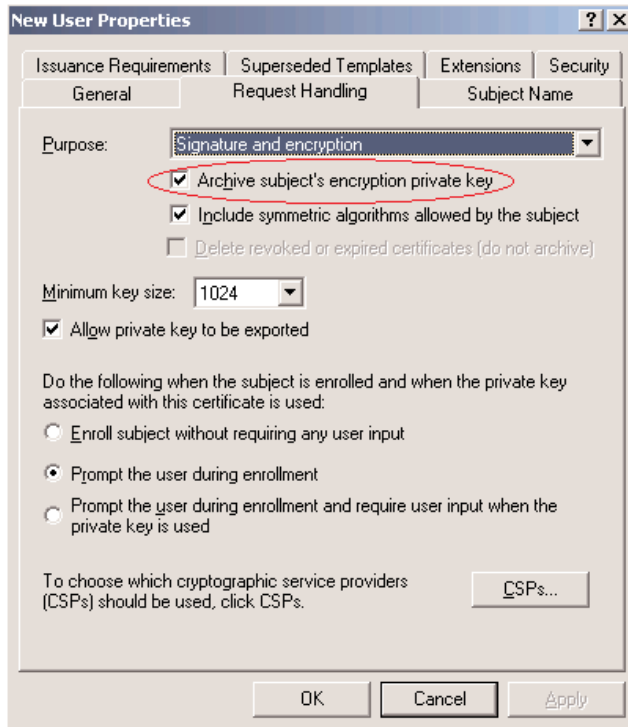


You can specify the number of KRAs you want to define in the "Number of recovery agents to use" text box. At the bottom of the Recovery Agents tab, you can select the KRA certificates you want to use for key archival.

When you click the Add button, the CA logic will query the KRA container in the AD configuration-naming context and retrieve a list of available KRA certificates. Each time you add a new KRA certificate, you must restart the CA service. As long as the CA service has not been restarted, the status column of the KRA certificate will say Not Loaded.

To enable key archival at the certificate-template level, you must use the Certificate Templates snap-in. To automatically archive the private key when a user requests a certificate based on a particular template, open the template, go to the Request Handling tab, and select the "Archive subject's encryption private key" check box (as Figure 4-22 shows). Note that you can set the key-archival option only on version 2 certificate templates.

Figure 4-22
Key archival settings in certificate template properties



Key Recovery from the CA Database

A key recovery is typically initiated by a PKI or PKI-enabled application user. Key recovery requires the intervention of at least one KRA, depending on how key archival and recovery has been configured in the CA properties.

Key recovery requires the intervention of both a certificate manager and a KRA: a certificate manager to retrieve the recovery data from the CA database, and a KRA to retrieve the archived private key from the recovery data. An archived private key can be recovered from the command line or from the Windows GUI.

A full Windows Server 2003 private-key recovery sequence from the command line consists of the following steps:

1. The KRA identifies the user who requests a key recovery.
2. The KRA writes down the user principal name (UPN), user common name (CN), account name (domain\username), and SHA-1 thumbprint (hash) or serial number of the user certificate whose private key the agent wants to recover. In this step, the most important task is to find a unique identifier for the key to be recovered. If more than one key is archived for a particular user, the safest thing is to first retrieve a list of all archived keys. You can do this by using the following command:

```
Certutil getkey <user common name, account name or UPN>
```

- Then, to export the recovery data from the CA database, the KRA opens a command prompt and types

```
certutil -getkey <Unique identifier> <output file>
```

- To transform the output file to a PKCS #12 file (which will contain the recovered private key and is secured using the password "test"), the KRA types

```
certutil p "test" -recoverkey <output file> <pkcs12 file>
```

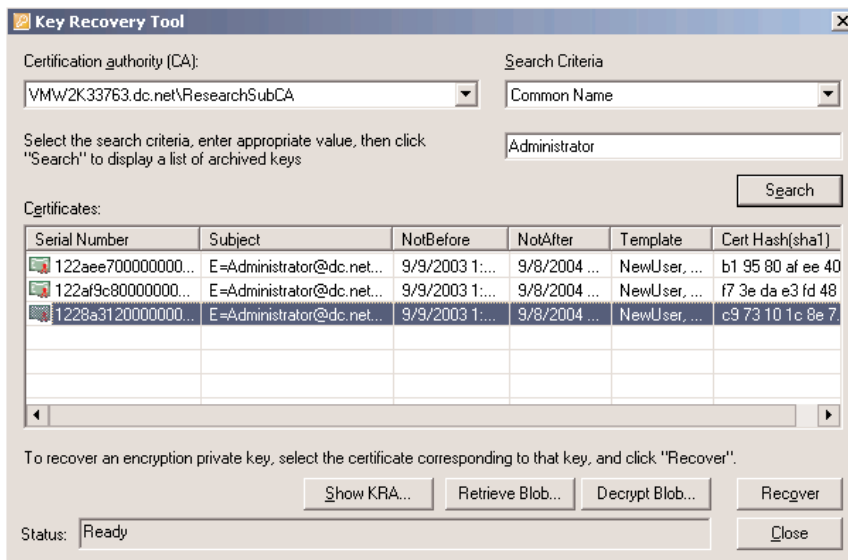
- The KRA provides the PKCS #12 file to the appropriate user, who can then import it to his or her certificate store.

If the KRA recovered multiple keys for the same user, the KRA can merge them all into a single PKCS 12 file. Doing this will facilitate the installation of the recovered keys for the user. To merge the keys into a single PKCS 12 file, the user must type the following certutil command (the password used to secure the PKCS12 file is again "test"):

```
CertUtil -p "test" -MergePFX -user "<PKCS12_ File1>,<PKCS12_File2>"
"<NameofCombined_PKCS12"
```

To recover keys from the GUI, you must use the Key Recovery Tool (krt.exe) that is part of the Windows Server 2003 Resource Kit (as Figure 4-23 shows). This tool offers a very convenient GUI-based way to recover keys that have been archived in the CA database.

Figure 4-23
Key recovery tool



To recover keys using the Key Recovery Tool, you must do the following:

1. Select a CA from whose database you want to recover keys (in the CA drop-down box).
2. Search for the archived private keys and certificates for a particular user. To do so, select a search criterion (common name, UPN, serial number, hash, or account name) in the Search Criteria drop-down box, and then fill in the user identifier and click the Search button.
3. You can then
 - Recover all archived keys at once (using the Recover button).
 - Recover a single key-certificate pair (using the Retrieve Blob... and Decrypt blob... buttons).

Conclusion

This chapter has focused on two important aspects of the certificate life cycle: certificate enrollment and key archival and recovery. In Windows Server 2003, these are also areas where Microsoft has added lots of new functionality. Every IT architect planning to build a Windows Server 2003-rooted PKI should understand the operation and benefits of the new certificate autoenrollment and centralized key archival and recovery services. In the next chapter, we look on the following aspects of the certificate lifecycle: certificate validation and certificate revocation.