Microsoft TechNet

Advanced Certificate Enrollment and Management

Published: May 17, 2004
By Carsten B. Kinder and David B. Cross

Complex infrastructure environments and branch-office deployment environments often dictate unique and advanced management techniques for managing a public key infrastructure (PKI) or certificate deployment to remote servers. This white paper explains several remote deployment scenarios along with the step-by-step procedures to perform X.509 certificate enrollment to implement a secure infrastructure.

On This Page

Introduction

Complex infrastructure environments and branch-office deployment environments often dictate unique and advanced management techniques to manage a PKI or certificate deployment to remote servers. Network and infrastructure services, such as domain controllers, Internet Authentication Servers (IAS), Internet Information Servers (IIS), and other stand-alone applications, often require X.509 certificate enrollment or provisioning to provide or enable secure protocols, messaging, or application services. In many of these deployment scenarios, automatic system provisioning or even traditional certificate enrollment may not be possible due to one or more reasons, such as:

- Stand-alone servers with no relationship to an Active Directory® domain
- Firewalls blocking required communication ports
- Complete lack of connectivity without a virtual private network (VPN) or Internet Protocol Security (IPSec) certificate credential to authenticate to the master network

For example, a branch office domain controller may be connected to the central site only through a firewall, and only port 25 is open for Simple Mail Transfer Protocol (SMTP) replication. The domain controller cannot automatically or manually enroll a domain controller certificate over Remote Procedure Call/Distributed Component Object Model (RPC/DCOM) and, therefore, SMTP replication will fail. Because the certification authority is located in the central site and the firewall is blocking RPC traffic, the branch office domain controller cannot contact the certification authority to enroll its certificate. In this situation, a domain controller certificate must be requested, processed, and installed in an asynchronous or offline process. This white paper explains several remote deployment scenarios along with the step-by-step procedures to perform X.509 certificate enrollment to implement a secure infrastructure.

Supported Scenarios

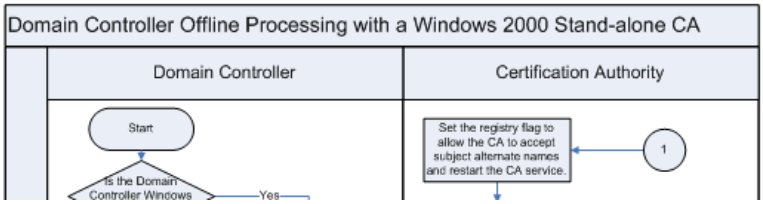
Although it is preferable to describe and support all environments, it is not always possible due to technology limitations or documentation requirements. This white paper documents and focuses on domain controller certificate enrollment for Windows 2000 and Windows Server™ 2003 domain controllers from a Windows 2000 or Windows Server 2003 stand-alone certificate authority (CA) as well as from a Windows Server 2003 enterprise CA. Because of technical constraints, manual certificate enrollment from a Windows 2000 enterprise CA is not covered in detail.

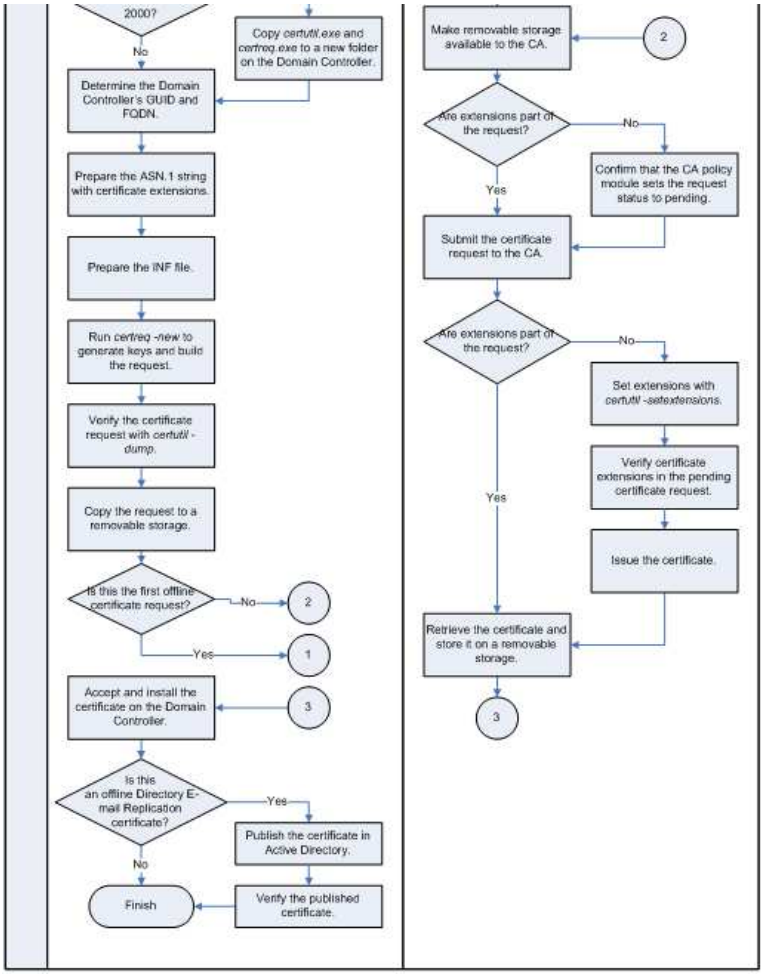
Assumptions

In this white paper, it is assumed that all domain controllers have been configured and function properly. It is also assumed that the CA has been implemented according to the recommended best practices by Microsoft® as documented in the white paper at
http://www.microsoft.com/technet/prodtechnol/windowsserver2003/technologies/security/ws3pkibp.mspx

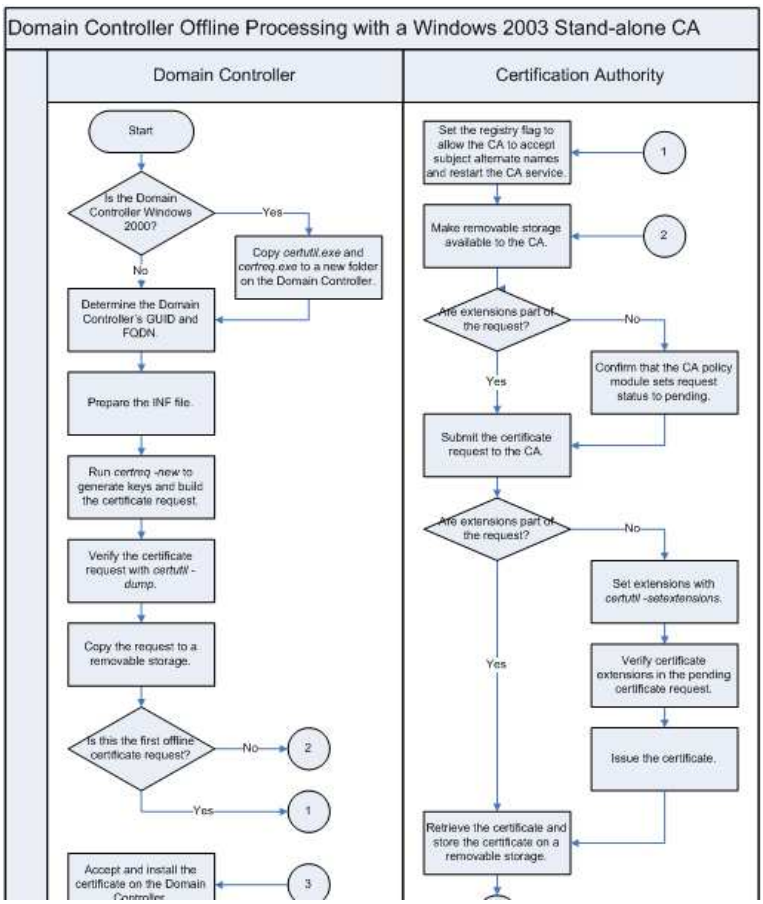
Process Overview

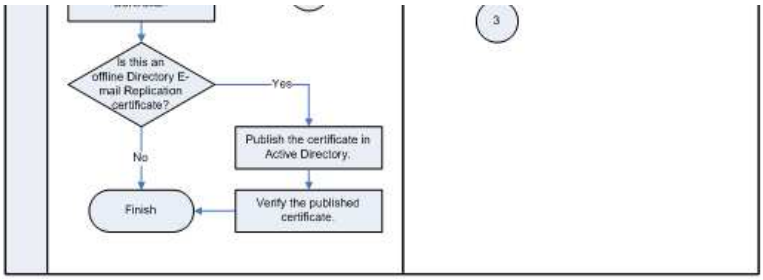
To provide a high-level overview of an advanced enrollment scenario, the following diagrams illustrate step-by-step references of the various procedures and processes that are described in detail in later sections.



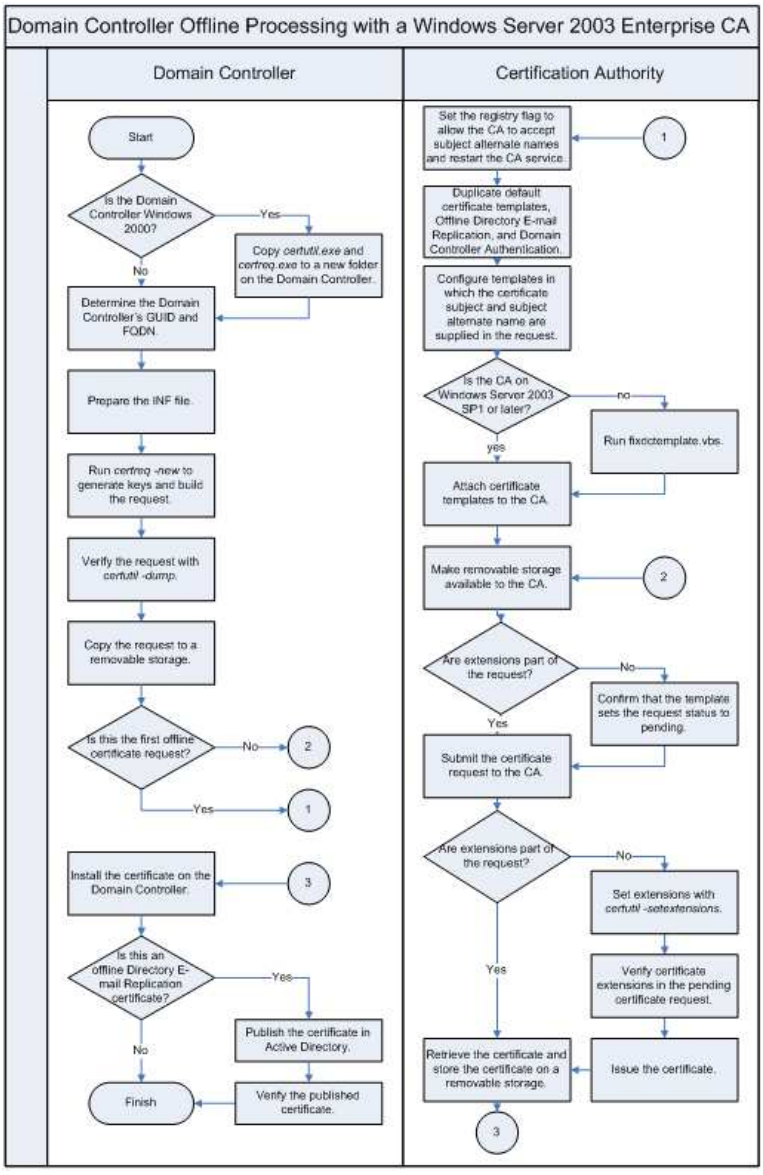


See full-sized image





See full-sized image



See full-sized image

Top of page

Requesting Offline Domain Controller Certificates

In branch-office scenarios with limited connectivity or those requiring a VPN or IPSec connection to the primary data center will often require an asynchronous enrollment process to enroll or provision the domain controller with an X.509 certificate. Asynchronous or (offline) certificate requests for domain controllers must be created with command-line tools because the operating system does not natively support an offline enrollment Wizard such as the one found in IIS. Since any type of certificate request can be created with the command-line tools, you can also build other certificate request types with these tools once you understand the general procedures. The following sections provide the steps required to enroll a domain controller for a certificate through an offline process.

Note: Some differences and capabilities exist between a Windows 2000 system and a Windows Server 2003 family operating system. The differences and required procedures specific to an operating system family are noted accordingly.

Preparing a Windows 2000 Domain Controller

For a Windows 2000 family operating system, the first step is to prepare the Windows 2000 domain controller. Since Windows 2000 has no natively installed tools to create certificate requests at a command line, a Windows Server 2003 tool must be used. The version of the certreq.exe and certutil.exe command-line tools available on Windows 2000 have only limited capabilities and do not satisfy the requirements for offline certificate request processing, which is required for this scenario. For

example, the Windows 2000 version of certreq.exe does not support the –new option which is required to create new certificate requests. Certutil.exe has many more options to verify and process certificates.

The Windows Server 2003 Administration Tools Pack includes the latest version of certreq.exe and certutil.exe and is required as part of this process. To install the tools on a Windows 2000 computer, you must first install the Windows Server 2003 Administration Tools Pack on a Windows XP or Windows Server 2003 system because it cannot be installed directly on a Windows 2000 computer.

The Windows Server 2003 Administration Tools Pack can be downloaded from the Web site at
http://www.microsoft.com/downloads/details.aspx?FamilyID=c16ae515-c8f4-47ef-a1e4-a8dcbacff8e3&DisplayLang=en

To install the tools on a Windows 2000 system, perform the following steps.

Important: Once you have copied the new files to the Windows 2000 domain controller, two versions of these files will reside on the Windows 2000 computer. Do not remove the natively installed files on the Windows 2000 system because other applications, like the Certificates MMC Snap-in, depend on these native versions. Also, do not register certcli.dll and certadm.dll on the Windows 2000 system through the regsvr32.exe command.

1. Log on to a computer that runs Windows XP Service Pack 1 or Windows Server 2003.
2. Install the Windows Server 2003 Administration Tools Pack.
3. Copy the following files to a removable storage medium such as a diskette.
   - certreq.exe
   - certutil.exe
   - certcli.dll
   - certadm.dll
4. Log off the computer.
5. Make the removable storage medium available to the domain controller that requires an offline domain controller certificate.
6. Log on to the domain controller as a domain administrator and create a new directory to store the files, for example, *%HOMEDRIVE%\W2K3AdmPak.*

   Note: Do not include this path in your system search path to avoid conflicts with the tool versions that already exist on your computer.
7. From the removable storage medium, copy the four files to the newly created directory.

Generating an Offline Certificate Request

This section applies to both Windows 2000 and Windows Server 2003 domain controllers. The next step is to generate an offline certificate request for the domain controller. As mentioned previously, you must create an offline certificate request with a command-line tool called certreq.exe. This tool supports a rich set of command-line options, but only a few options are required in this procedure. For more information on the certreq.exe tool and its syntax, see Appendix 3: Certreq.exe Syntax.

Certreq.exe requires a text (instruction) file to generate an appropriate X.509 certificate request for a domain controller. You can create the file with your preferred (ASCII) text editor and save the file with an *.inf extension to any directory on your hard drive.

In general, a domain controller certificate that is to be used for SMTP replication must meet the following requirements.

- The certificate must contain the "Certificate Template Name" (referenced by the object identifier 1.3.6.1.4.1.311.20.2) extension.
- The certificate must contain a "Subject Alternative Name" extension that includes the Globally Unique Identifier (GUID) of the domain controller account and the fully qualified domain name (FQDN) of the domain controller Domain Name System (DNS) host name. The subject alternative name is uniquely identified with object identifier 2.5.29.17.

Typically, both extensions are automatically inserted in the certificate by an enterprise CA when the certificate is enrolled automatically or manually while connected to a CA. However, when you request the domain controller certificate offline, you must provide these extensions explicitly in the offline request.

There are several different approaches to adding extensions in certificate requests and, ultimately, issued X.509 certificates. For a Windows 2000 CA, you can either include the extension(s) in the INF instruction file, or you can add the extension(s) to a pending certificate request. If you add the extension(s) to a pending certificate request, it is not necessary to also use the INF instruction file to perform the same task.

For a Windows Server 2003 CA, you can either use the same procedures as a Windows 2000 CA, or you can specify certificate extensions using certreq.exe from a command-line when the certificate request is submitted.

Important: When you submit extensions from the certreq.exe command-line tool, you cannot set the critical flag for the submitted extensions. Therefore, if an extension is required to be marked critical, you must use an INF instruction file as the submission method.

Identifying the Domain Controller GUID

Active Directory replication via SMTP requires the domain controller GUID as an attribute in the domain controller certificate. Thus, you must identify the domain controller GUID if you are going to issue certificates for SMTP replication from a Windows Server 2003 enterprise or stand-alone CA. Since the GUID of the domain controller is stored in Active Directory, there are several ways to read this attribute from the domain controller's computer object. You do not need to be logged on to a domain controller or the CA to read the GUID from Active Directory; any computer that has read permissions to objects in the domain is sufficient to complete this task. The following is a partial list of tools that can be used to determine the domain controller GUID from Active Directory.

- Adsiedit.exe (available as part of the Windows® support tools)
- Ldp.exe (available as part of the Windows support tools)
- Dsquery.exe (available as part of the Windows support tools)
- Netdiag.exe (available as part of the Windows support tools)
- Replmon.exe (available as part of the Windows support tools)
- Windows Management Instrumentation (WMI)

For more information, see the Microsoft Knowledge Base article "Determining the Server GUID of a Domain Controller" at

http://support.microsoft.com/default.aspx?scid=kb;en-us;224544

Since it may be complicated in some scenarios to easily identify a domain controller GUID, a script is provided in Appendix 2 to look up the GUID from Active Directory automatically when the certificate request is generated. This script simplifies the process for administrators. For more information, see Reqdccert.vbs – Generates Domain Controller Certificate Requests.

Creating an Offline Certificate Request
Every X.509 certificate that is issued by a CA requires a unique certificate request to initiate the issuance process. If required, you can create several certificate requests as a batch process before submitting them to a CA.

However, to avoid potential errors and confusion, ensure that you are using unique names for your request files. It is necessary to create the certificate request as a member of the local Administrators group because only this group has interactive logon permissions to a domain controller by default. The first step is to define the input information in an INF file.

To create the INF file that will supply input information for the request, perform the following steps.

1. Log on as a member of the local Administrators group on the domain controller that requires a certificate.
2. Make the reqdccert.vbs script, found in Appendix 2, available locally to the domain controller.
3. If you plan to submit the certificate request to a Windows 2000 or Windows Server 2003 stand-alone CA, run the script from a command-line prompt without any additional parameters. In this case, the *Domain Controller* certificate template is assumed and the domain controller's GUID and DNS name are included as a subject alternative name in the issued certificate.

   If you plan to submit the certificate request to a Windows Server 2003 enterprise CA, you must specify the specific name of the certificate template that is used for enrollment. The name depends on the certificate template name that was previously chosen in **Certificate Template Creation**. In addition, you must specify if the certificate is enrolled for either domain controller authentication or for e-mail replication. This is required because the authentication and e-mail replication templates differ in the subject alternative name construction.

   According to your requirement, run one of the following commands at the command-line prompt.

   To create a certificate request for a domain controller authentication certificate with a custom template:

   ```
   cscript reqdccert.vbs <Templatename> A
   ```

   To create a certificate request for a directory e-mail replication certificate with a custom template:

   ```
   cscript reqdccert.vbs <Templatename> E
   ```

   To create a certificate request with the default domain controller template:

   ```
   cscript reqdccert.vbs
   ```

   The script creates an INF file that is used as an input file to build the certificate request. It also creates some additional syntax that contains the appropriate command syntax to submit the request to the CA and verify the certificate.

   For a detailed description of the tasks that are performed by the script, see Reqdccert.vbs – Generates Domain Controller Certificate Requests in Appendix 2.
4. Open a command-prompt window. On a Windows 2000 domain controller, type

   ```
   %HOMEDRIVE%\W2K3AdmPak\CERTREQ -new <dcname>.inf <dcname>.req
   ```

   On a Windows Server 2003 domain controller, type

   ```
   CERTREQ -new <dcname>.inf <dcname>.req
   ```

   Replace the <dcname> variable with the hostname of the domain controller. If you are unsure of the name, look up the name of the INF file saved in your current directory. Note that it may take a few seconds or minutes before the command prompt returns. Depending on the size and available cycles of the system CPU as well as the key length that was set in the INF file, it may take several minutes for the key material to be generated.

   Important: The certreq.exe tool performs several tasks that are often not apparent. First, it generates the public and private key material, and then, it generates the actual certificate request. The key material generation process is the reason why certificate requests must not be generated for several domain controllers on a single machine. The key material must be unique per domain controller to ensure a secure solution.

Viewing the Certificate Request
At this stage, you have successfully generated the public and private key material and created the certificate request. The certificate request is stored in a file and is kept in the *Certificate Enrollment Requests* store of the local machine. To find the enrollment request that was generated, perform the following steps.

1. While still logged on as a member of the local Administrators group, start the Microsoft Management console.
2. Add the Certificates MMC Snap-In.
3. Select Computer Account when the Add window asks for the certificate store that should be managed.
4. In the Certificates MMC Snap-In, navigate to **Certificate Enrollment Requests** in the left pane.

   The certificate request appears in the right pane. It is expected that there is no more than one certificate request pending in this container. Otherwise, this would imply that other certificate request attempts have been submitted but have not yet been approved or accepted.

5. When you double-click the certificate request, the message "You have a private key that corresponds to this certificate" is displayed in the General tab at the bottom. As mentioned previously, certreq.exe has generated the certificate request and the key material. You can safely ignore the warning at the top of the window that says the certificate might be altered. This is an expected warning since the certificate has not yet been issued.

Verifying a Certificate Request

It is recommended that the offline domain controller certificate request be validated before it is sent to a CA for processing. Verification will ensure that all request fields are set properly and that the certificate, when processed and issued, will provide the expected functionality. To examine the previously generated certificate request, perform the following step.

- While still logged on as a member of the local Administrators group to the domain controller, type certutil –dump <dcname>.req at a command-line prompt, and then press Enter.

Replace the <dcname> option with the name used in the previous section to generate the offline request. If you have properly created a request that has used the DomainController certificate template, the command will display an output similar to the following:

```
PKCS10 Certificate Request:
Version: 1
Subject:
    CN=W2K3-BO-DC.contoso2.com
Public Key Algorithm:
    Algorithm ObjectId: 1.2.840.113549.1.1.1 RSA
    Algorithm Parameters:
    05 00
Public Key Length: 1024 bits
Public Key: UnusedBits = 0
    0000  30 81 89 02 81 81 00 95  b8 e9 18 84 df 95 ce 37
    0010  ce f6 af 32 40 43 14 d5  0f 7b 3b 76 36 8c dd 8b
    0020  7c 03 29 33 26 d3 84 c3  7e ae 25 34 ea 1e db 3a
    0030  b9 01 e3 a7 02 3c 6b 8f  66 99 c8 ac 51 70 03 bc
    0040  47 06 ef 2f 62 3e c3 8d  e1 51 bd 9d c8 7d 95 8c
    0050  08 0a bf 54 a6 f3 1d 2f  cd b8 7d 17 fc 4c 7d a5
    0060  a6 ce 90 d0 a3 21 c5 b0  c1 f0 de ae 00 43 16 cb
    0070  eb 73 01 e7 71 79 ed dd  72 d0 cc 4a 55 26 a2 99
    0080  03 21 dc d1 5b b4 b9 02  03 01 00 01
Request Attributes: 4
  4 attributes:
  Attribute[0]: 1.3.6.1.4.1.311.13.2.3 (OS Version)
    Value[0][0]:
        5.2.3790.2
  Attribute[1]: 1.3.6.1.4.1.311.21.20 (Client Information)
    Value[1][0]:
    Unknown Attribute type
    Client Id: = 1
    XECI_XENROLL -- 1
    User: CONTOSO2\administrator
    Machine: W2K3-BO-DC.contoso.com
    Process: certreq
  Attribute[2]: 1.2.840.113549.1.9.14 (Certificate Extensions)
    Value[2][0]:
    Unknown Attribute type
Certificate Extensions: 4
    2.5.29.14: Flags = 0, Length = 16
    Subject Key Identifier
        55 93 fd 45 5b 22 87 33 95 96 4a 77 e3 ff 08 08 f6 83 de fc
    2.5.29.17: Flags = 1(Critical), Length = 3c
    Subject Alternative Name
        Other Name:
             1.3.6.1.4.1.311.25.1= 0410 6661 6135 3636 3234 3831 6263 3866 6662
        DNS Name=W2K3-BO-DC.contoso.com
    2.5.29.37: Flags = 0, Length = 16
    Enhanced Key Usage
        Server Authentication (1.3.6.1.5.5.7.3.1)
        Client Authentication (1.3.6.1.5.5.7.3.2)
    2.5.29.15: Flags = 0, Length = 4
    Key Usage
        Digital Signature, Key Encipherment (a0)
  Attribute[3]: 1.3.6.1.4.1.311.13.2.2 (Enrollment CSP)
    Value[3][0]:
    Unknown Attribute type
    CSP Provider Info
    KeySpec = 1
    Provider = Microsoft RSA SChannel Cryptographic Provider
    Signature: UnusedBits=0
    0000  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
    Remaining 78 bytes are zero
Signature Algorithm:
    Algorithm ObjectId: 1.2.840.113549.1.1.5 sha1RSA
    Algorithm Parameters:
    05 00
Signature: UnusedBits=0
    0000  45 4c 4f ca 00 52 36 88  a0 2d 2e 45 f3 87 be ac
    0010  89 4f 4e d4 19 42 a7 e5  cb 7a 15 d5 eb e2 a0 96
    0020  6c 39 94 c5 71 d6 e7 03  10 9c 45 a0 ad c7 34 e6
    0030  f9 f2 31 da ce e2 2e 6f  b7 23 6f 12 53 6a 40 89
    0040  ba a9 2e 2f bc cf d4 00  72 18 82 05 33 ea 0e 20
    0050  0b e2 5c 4c 5a 57 b5 71  08 e0 3e d8 8b 0d 18 05
    0060  7b 11 4d b1 e9 db 16 e5  78 e8 c2 b2 ff bb c2 9d
    0070  e6 2b 17 83 dc 1d 43 fd  4e 0e 37 58 f0 ac a9 95
Signature matches Public Key
Key Id Hash(sha1): 55 93 fd 45 5b 22 87 33 95 96 4a 77 e3 ff 08 08 f6 83 de fc
CertUtil: -dump command completed successfully.
```

It should be expected from the output of the certificate request that the majority of attributes from the generated INF file appear in the actual encoded certificate

request.

The reqdccert.vbs script creates an INF instruction file that contains the extension information for DNS name and GUID of the domain controller in an encoded format. This enables the certificate requested to be constructed with certreq.exe –new, whereby the extension information will be automatically included in the certificate request from the INF file.

Deleting a Certificate Request
Sometimes it may be necessary to either delete an unneeded or unused certificate request or even delete erroneous certificate requests due to mistakes in the INF input file. If you are logged on as a normal user, you can remove certificate requests only from your own certificate store. However, if you are logged on as Administrator, you can also remove certificate requests from the computer's certificate store. To delete a certificate request, perform the following steps.

1. Open the Certificates MMC Snap-In for the computer account (as described previously).
2. Navigate to Certificate Enrollment Requests in the left pane.
3. Remove the pending certificate request in the right pane by pressing Delete on your keyboard.
4. Close the Certificates MMC Snap-In.
5. If the request has already been submitted to a CA, it may be necessary to also deny the pending request on the CA.

The key material that was generated for the certificate request will remain on the system in the local system (computer) profile unless explicitly deleted by an administrator. Orphaned key material may only be removed manually at the command-line prompt using certutil.exe. However, the administrator will need to know the key container.

Warning: If you delete keys manually from your computer, you will invalidate all data that was encrypted with an encryption key. If you have not implemented a key recovery mechanism, it is recommended that you leave unused keys on the system instead of deleting them.

Use the following command to display the list of available key containers for the machine context.

```
certutil –store my
```

Use the following command for the current user's context.

```
certutil –store –user my
```

The output will display the key container name for each certificate. Key container names typically are created with random GUID strings as the name.

If you have determined the *keycontainername* for a specific certificate, you can delete the key container with the following command.

```
certutil.exe -delkey <KeyContainerName>
```

The -delkey option is supported only with the Windows Server 2003 version of certutil. On Windows 2000, you must add a prefix to the commands. The prefix is the path you have copied the Windows Server 2003 version of certutil to. In this white paper, the *%HOMEDRIVE%\W2K3AdmPak* path is used.

Transferring the Request to a CA
Once the request file has been generated and validated on the domain controller, it can be transferred to the CA for processing and issuance. Transfer the following files to the CA.

- <dcname>-req This file contains the certificate request that was generated on the domain controller.
- <dcname>-req.bat This batch file was created by reqdccert.vbs and contains the correct command-line parameters to submit the request to the CA.

At this point, you can log off from your domain controller.

Top of page

Processing Domain Controller Certificates

Due to differences in the Certificate Services components in Windows 2000 and Windows Server 2003, the issuing process for domain controller certificates depends on the certificate template version and the specific certificate template. Since the processes and steps required are different, both types are outlined separately in the following sections.

Certificate Templates

The Windows 2000 and 2003 Server enterprise certification authority (CA) supports the concept of certificate templates. Certificate templates define how an enterprise CA should process a certificate request and generate a specific certificate type when issued. A difference exists between the *Domain Controller* certificate templates in a Windows 2000 and Windows Server 2003 Active Directory environment. It is necessary to understand the differences before you submit the certificate request to a CA. For a broader discussion on certificate templates, see the following references.

- Implementing and Administering Certificate Templates in Windows Server 2003 at http://www.microsoft.com/technet/prodtechnol/windowsserver2003/technologies/security/ws03crtm.mspx
- Selecting Certificate Templates at http://www.microsoft.com/technet/prodtechnol/windowsserver2003/library/depkit/C71D2CD3-82EF-4E3C-8746-1340D0EF4E9A.mspx

Windows 2000 Certificate Templates
A Windows 2000 CA provides only one certificate template for domain controller certificates: *Domain Controller*. It may be manually or automatically enrolled through Automatic Certificate Request Service (ACRS) settings in Group Policy. The certificate template name is hard-coded in the operating system and it must be used to enroll Windows 2000 domain controllers. In Windows 2000, you cannot modify certificate templates and thus, only the *Domain Controller* template may be used.

With a domain controller certificate that was issued by a Windows 2000 CA, a domain controller can use the certificate for any of the following purposes.

- Provide mutual authentication for Public Key Cryptography for Initial Authentication in Kerberos (PKINIT) or smart card logon.
- Encrypt Active Directory replication traffic if SMTP replication is enabled.

- Enable Lightweight Directory Access Protocol (LDAP) over Secure Sockets Layer (SSL) connections to Active Directory.
- Authenticate using SSL client authentication for an application that requires this functionality.

If Windows 2000 style auto-enrollment (ACRS in Group Policy) is enabled in a connected environment, a Windows 2000 domain controller will auto-enroll certificates based on the *Domain Controller* template.

Windows Server 2003 Certificate Templates
In contrast to a Windows 2000 enterprise CA, a Windows Server 2003 enterprise CA provides three templates for domain controller certificates. In Windows Server 2003, the *Domain Controller* certificate template is known as a V1 certificate template and exists to support auto-enrollment for Windows 2000 domain controllers. The two other certificate templates are *Directory Email Replication* and *Domain Controller Authentication*. Both certificate templates are V2 templates, which implies they can be modified, duplicated, and used for the new style certificate auto-enrollment.

Windows XP and Windows Server 2003 systems that are joined to an Active Directory domain support a more advanced auto-enrollment mechanism than previously available in Windows 2000. To support a Windows 2000 computer in a mixed environment with Windows XP and Windows Server 2003 systems, both methods are supported by a Windows Server 2003 CA. For more information on the new style auto-enrollment, see the white paper at
http://www.microsoft.com/technet/prodtechnol/windowsserver2003/plan/autoenro.mspx

The capabilities of the Windows 2000 V1 *Domain Controller* certificate template have been divided into two V2 certificate templates to provide more granular functionality. Instead of having one multi-purpose domain controller certificate that can be used for almost everything, the *Domain Controller Authentication* certificate template is tailored for smart card logon support; the *Directory Email Replication* certificate is made for supporting SMTP e-mail replication. Thus, if you do not require Active Directory replication via SMTP, you do not have to deploy *Directory Email Replication* certificates.

Note: Both certificate templates are configured by default to supersede the former *Domain Controller* certificate template, which will be performed when both a Windows 2000 enterprise CA and domain controller are upgraded to Windows Server 2003. For more information on superseding templates, see the Implementing and Administering Certificate Templates in Windows Server 2003 white paper at
http://www.microsoft.com/technet/prodtechnol/windowsserver2003/technologies/security/ws03crtm.mspx

To summarize the previous points, the following table provides an overview of the certificate templates that are used by default in Windows 2000 and Windows Server 2003.

Domain Controller OS	Windows 2000 Stand-alone CA	Windows 2000 Enterprise CA	Windows Server 2003 Stand-alone CA	Windows Server 2003 Enterprise CA
Windows 2000	Domain Controller	Domain Controller	Domain Controller	Domain Controller
Windows Server 2003	Domain Controller	Domain Controller	Domain Controller	• Domain Controller Authentication • Directory Email Replication

These same templates can also be used for domain controller offline certificate enrollment. In the case of a Windows Server 2003 enterprise CA, it will also be required to customize the *Domain Controller* certificate template. For more information on this process and requirements, see **Certificate Template Configuration**.

Using Version 2 Templates with Windows 2000 Computers
In a mixed environment where you may support or have multiple operating system versions installed, your choice of template version may be limited by the client operating system. Only Windows XP client systems are able to enroll for version 2 templates through the Certificates MMC Snap-In. Windows 2000 clients are not able to display or enroll for version 2 templates through the Certificates MMC Snap-In or automatic certificate request settings in Group Policy. Nevertheless, a Windows 2000 computer can be used to enroll for user certificates that are based on version 2 templates through the Windows Server 2003 Web enrollment pages. As an alternative, you could install certreq.exe on a Windows 2000 computer and request certificates based on version 2 templates manually using that method.

Domain Controller Certificate Details
Since many of the offline domain controller enrollment processes involve complex, manual procedures, the following section is provided to assist in understanding the most important characteristics of a domain controller certificate as a reference.

Certificate Purpose	Domain Controller	Domain Controller Authentication	Directory Email Replication
Domain controller authentication–A domain controller can prove its identity to another party, such as a client machine, during smart card logon.	Yes	Yes	No
SSL–If a Web server is installed on the domain controller, the Web server can leverage the domain controller certificate to establish SSL connections with clients. This also supports LDAP over SSL connections to the domain controller.	Yes	Yes	No
Client authentication (SSL)–This would be used if the machine acts as an SSL client to another application on a separate server.	Yes	Yes	No
E-mail encryption–SMTP e-mail can be encrypted and signed with this certificate.	Yes	No	Yes

A certificate that was issued by a CA based on the *Domain Controller* certificate template has the following characteristics.

- The subject contains the domain controllers FQDN prefix with the "CN=" relative distinguished name (RDN) element.
- The certificate purposes (also known as extended key usage) are set to "Client Authentication (1.3.6.1.5.5.7.3.2)" and "Server Authentication (1.3.6.1.5.5.7.3.1)". The numbers in parentheses are the corresponding standard object identifier (OID) for each certificate purpose.
- The common name of the template is set to "DomainController".
- The Subject Alternative Name contains the domain controller's GUID in OID 1.3.6.1.4.1.311.25.1 and the FQDN of the domain controller.

A certificate that was issued based on the *Domain Controller Authentication* certificate template has the following characteristics.

- The subject of the certificate is empty.
- The certificate purposes (also known as extended key-usage) are set to "Client Authentication (1.3.6.1.5.5.7.3.2)", "Server Authentication (1.3.6.1.5.5.7.3.1)", and "Smart Card Logon (1.3.6.1.4.1.311.20.2.2)". The numbers in parentheses are the corresponding OID for each certificate purpose.

- The common name of the template is set to "Domain Controller Authentication" or the name of the template that was specified in the certificate request for this certificate type.
- The Subject Alternative Name extension contains the domain controller's fully qualified DNS name of the domain controller.

A certificate that was issued based on the *Directory Email Replication* certificate template has the following characteristics.

- The subject of the certificate is empty.
- The certificate purpose (also known as extended key-usage) is set to "Directory Service Email Replication (1.3.6.1.4.1.311.21.19)". The numbers in parentheses are the corresponding OID for each certificate purpose.
- The common name of the template is set to "DirectoryEmailReplication" or the name of the template that was specified in the certificate request for this certificate type.
- The Subject Alternative Name extension contains the domain controller's GUID in OID 1.3.6.1.4.1.311.25.1 and the FQDN of the domain controller.

Windows Server 2003 Certificates and Publishing in Active Directory
Certificates, which are enrolled from an enterprise CA with the *Domain Controller* or the *Directory Email Replication* certificate template, are published by default into the requester's object in Active Directory. When a certificate is auto-enrolled, the requestor is naturally a domain controller. These certificate templates publish certificates in Active Directory primarily to facilitate encrypted replication of Active Directory content with SMTP; both replication partners must have access to the public key (certificate) of their replication partner.

Only enterprise certification authorities publish certificates in Active Directory as an automatic process. If you enroll a *Domain Controller* certificate manually from a stand-alone CA, you must publish the certificate manually in Active Directory. Remember that stand-alone CAs do not offer the *Domain Controller Authentication* and *Directory Email Replication* certificate template format(s), so regardless of the CA version, you can only issue *Domain Controller* certificates from a stand-alone CA.

By default, certificates that have been built with the *Domain Controller Authentication* certificate template are not published in Active Directory because smart card logon requires the domain controller's certificate in Active Directory. Therefore, it is not recommended that *Domain Controller Authentication* certificates be published in Active Directory by manipulating the default *Domain Controller Authentication* certificate template.

Windows 2000 Server CA Configuration

By default, a Windows 2000 Server CA does not permit subject alternative names that are specified in a certificate request to be accepted and inserted in the issued certificate. This applies for both stand-alone and enterprise CAs. This functionality is required to submit and process offline domain controller certificate requests. To permit this functionality, the CA configuration must be modified.

Important: Changing the CA configuration of the CA to permit subject alternative names in certificate requests will be a global setting and is not limited to a single template. Once this setting is enabled, the CA will accept attributes for subject alternative names for all certificate requests.

To change the CA configuration to accept subject alternative names in requests, perform the following steps.

1. Log on to the CA as Administrator.
2. At a command-line prompt, type

   ```
   CERTUTIL -setreg policy\EditFlags +EDITF_ATTRIBUTESUBJECTALTNAME2
   ```

   This enables the CA to accept the "Subject Alternative Name" to be submitted in the request. If you submit more offline domain controller requests to this CA, you do not have to set the parameter again. It persists until it is manually reset to the default setting by an administrator. To reset the CA to the default setting, type the following at a command-line prompt.

   ```
   CERTUTIL -setreg policy\EditFlags -EDITF_ATTRIBUTESUBJECTALTNAME2
   ```

   Note: This setting will affect all certificate requests that are submitted to your CA. Any certificate request that provides a SubjectAltName2 will be recognized and processed by the CA.
3. The following command will display the configuration parameters (EditFlags) that have been set before and after the change. You can verify the parameters explicitly with the following command.

   ```
   CERTUTIL -getreg policy\EditFlags
   ```
4. To enable the changes in all subsequently issued certificates, restart the certification authority service. To restart the CA, type the following at a command prompt, and then press Enter.

   ```
   NET STOP certsvc & NET START certsvc
   ```

Issuing Domain Controller Certificates with a Windows 2000 CA

A CA that is installed on a Windows 2000 computer can issue domain controller certificate requests for both Windows 2000 and Windows Server 2003 domain controllers.

With a Windows 2000 CA, no option is available to customize the *Domain Controller* certificate template. However, as a limited option, you may set the validity time of a domain controller certificate on a specific CA. For information on how to change the validity time, see the Microsoft Knowledge Base article *HOW TO: Change the Expiration Date of Certificates That Are Issued by a Windows Server 2003 or a Windows 2000 Server Certificate Authority* at http://support.microsoft.com/default.aspx?id=254632

Windows 2000 Stand-alone CA
To issue domain controller certificates, it does not matter if the stand-alone CA is a domain member or a member of a workgroup. With both configurations, domain controller certificates can be issued. A stand-alone CA does not support certificate templates, therefore, no choice is required in this regard. A stand-alone CA will process a certificate request according to the information supplied in the request.

To issue a certificate for domain controllers from a Windows 2000 CA, you can use one of two approaches.

- Prepare a certificate request with the Windows Server 2003 version of certreq.exe that includes the domain controllers GUID and its DNS name in the subject alternative name.
- Use a generic certificate request and add the certificate extensions to the certificate request while it is pending at the CA.

In the first case where you have included the subject alternative name in the certificate request, it is not required to set the certificate request status to pending before issuing the certificate.

Since this white paper illustrates how to include the subject alternative name in the certificate request in **Certificate Template Configuration**, the following section explains how to manipulate a pending request. You may use the following procedure with a different set of command-line parameters for other certificate types as well.

Recommendation: From a technical viewpoint, both methods result in the same certificate being issued. However, it is more convenient to include extensions in the certificate request because no additional manipulation of a pending request is required. This method, however, does require the Windows Server 2003 version of certreq.exe to be used.

Windows 2000 CA Configuration
Once you have set the *EditFlag* (see Windows 2000 Server CA Configuration), you should verify that the policy module will put all certificates into a pending mode before they are issued. To confirm the request handling of the CA, perform the following steps.

1. Log on to the computer where the Windows Server 2000 stand-alone CA is installed.
2. Open the Certification Authority MMC Snap-In.
3. In the left pane, select the CA object and select Properties on the Action menu.
4. In the Policy Module tab, click Properties.
5. Verify that the policy module of the stand-alone CA is set to the following value.

   Set the certificate request status to pending.

   Note: A stand-alone CA sets the certificate request status to pending by default.
6. Click **OK** twice to confirm the setting and close the Properties page.
7. Close the Certification Authority MMC Snap-In.

Issuing a Domain Controller Certificate
Compared to a Windows 2003 CA, it is more complicated to issue certificates that require a given set of subject alternative names in the certificate. Unfortunately, unlike a Windows Server 2003 CA, you cannot specify certificate extensions such as the subject alternative name when you submit the certificate. The Windows 2000 CA does not support such functionality. To work around this limitation, you must include the certificate extensions in the certificate request or perform the certificate enrollment process in three distinct steps, with an optional fourth step.

1. The certificate request is submitted.
2. Certificate extensions are set in the pending certificate request.
3. The certificate is approved and issued.
4. The certificate can be manually verified for accuracy.

The following steps show how to submit the request, process the request, verify the request, and issue the certificate. In this case, the certificate request does not include the certificate extensions.

Submit the Certificate Request
1. Log on to the computer where the Windows Server 2000 stand-alone CA is installed.
2. Copy the certificate request (<dcname>.req) and the batch script (<dcname>-req.bat) that were previously created on the domain controller (see Creating an Offline Certificate Request) into a working file folder on the CA.
3. At a command-line prompt, use the <dcname>-req.bat script to run the required certreq command. The batch file will simply submit the certificate request to the CA with the following command.

   ```
   certreq -attrib "CertificateTemplate:DomainController" <dcname.req>
   ```
4. A window will appear where you can select the CA that will issue the certificate. Select the Windows 2000 issuing CA and click OK.
5. Note the RequestID that is shown after the command has finished.

Process the Certificate Request
The next step is to set the certificate extensions in the pending certificate request using either the Windows 2000 or the Windows Server 2003 version of certutil.

Important: The following command will override any existing information in the certificate request that matches the given OID. If you have specified, for example, a subject alternative name as part of the certificate request, the subject alternative name will be overridden with this command.

1. Copy the <dcname>.asn file from the domain controller to the CA computer.
2. At a command-line prompt, run the following command.

   ```
   certutil -setextension <RequestID> 2.5.29.17 1 @<dcname>.asn
   ```

   The subject alternative name, which is identified by the OID 2.5.29.17, is set with the attributes that are defined in the <dcname>.asn file. The fourth parameter that is set to "1" marks the extension as critical. For a description of the file structure of the ASN file, see Appendix 5: ASN.1 File Structure.

Verify the Pending Certificate Request
You may verify the pending certificate request to validate that the extensions are properly inserted in the certificate request. If something is not correct or if the

certificate request does not meet your requirements, you can validate the information at this point to ensure accuracy. Unfortunately, the Windows 2000 CA does not support viewing the properties of a pending certificate request in the Certificate Authorities MMC Snap-In. To verify that the certificate will be issued with the correct certificate template, you can use either the Windows 2000 or Windows Server 2003 version of certutil.

1. From a command-line prompt, run the following command.

   ```
   certutil -view -restrict RequestID=<RequestID> -out RequestAttributes
   ```

   Replace the <RequestID> with the RequestID that was recorded previously.

   The command output should look similar to the following:

   ```
   Row 1:
     Request Attributes: "CertificateTemplate:DomainController"
   ```

2. To view the request attributes of all pending certificate requests, run the following command.

   ```
   certutil -view -restrict disposition=9 -out RequestID,RequestAttributes
   ```

3. Verify that the certificate extensions have been set correctly. Type the following command at a command-line prompt.

   ```
   certutil -view -restrict RequestID=<RequestID> -out ext:2.5.29.17
   ```

   Replace the <RequestID> with the RequestID that was recorded previously. The "2.5.29.17" value represents the OID of the certificate extension that you are referencing. If you do not know the exact OID of your extension, use "all" instead of the specific OID. For example:

   ```
   certutil -view -restrict RequestID=<RequestID> -out ext:all
   ```

   Ensure that the extension you have configured in the previous section appears in the pending certificate request.

Note: All of the certutil -view commands may be used to query the certification authority database on a Windows 2000 or Windows Server 2003 CA. Once you understand the schema, you can query the database providing the appropriate field names. You can also use certutil -view -? for more samples to query the database.

Issue and Retrieve the Certificate
Next, you must issue the pending certificate. For this section, you can use either the Windows 2000 or Windows Server 2003 version of certutil.

1. To issue the pending request, open the Certificate Authority MMC Snap-In. Run certsrv.msc at a command-line prompt, and then press Enter.
2. In the left pane, click the Pending Certificates container.
3. In the right pane, select the certificate that corresponds to the RequestID recorded previously.
4. On the Action menu, select All Tasks – Issue.

   Alternatively, you can issue certificates from the command line with the following command.

   ```
   certutil –resubmit <RequestID>
   ```

   Replace <RequestID> with the RequestID value that was recorded previously by the certreq command.

5. Once the certificate has been issued, you must store the certificate as a file to transfer it to the domain controller. The following command will create two certificate files. The CER file contains only the domain controller certificate; the P7B file contains the domain controller certificate and all of its parent certificates. At a command-line prompt, run the following command.

   ```
   CERTREQ -retrieve <RequestID> <dcname>.cer <dcname>.p7b
   ```

   Replace <RequestID> with the RequestID that was used in the previous commands.

   A window appears where you can select the CA that has issued the certificate.

6. Select the issuing CA and click OK.
7. Store the retrieved certificates on a diskette or other medium to transfer to the domain controller.
8. Log off the CA.

To install the certificate on the domain controller, see Domain Controller Certificate Installation.

Windows 2000 Enterprise CA
Unfortunately, domain controller certificates cannot be issued manually from a Windows 2000 enterprise CA for the following reasons.

- A Windows 2000 CA supports only hard-coded certificate templates so that you cannot duplicate and customize the default certificate template. If the original domain controller certificate was manipulated, it would affect all other domain controllers that are able to connect to the CA and use auto-enrollment to request certificates.
- A Windows Server 2000 CA issues certificates immediately without pending. Since the submission interface on the CA has limited functionality in Windows 2000 and, therefore, certificate manipulation is required in a pending state, support for offline domain controllers is particularly problematic.

Therefore, if support for offline certificate request processing is required, it is recommended that you install a Windows 2000 stand-alone CA or a Windows Server 2003

CA.

Issuing Domain Controller Certificates with a Windows Server 2003 CA

A Windows Server 2003 enterprise CA may support V2 templates and pending requests on a per template basis, as well as support for submitting request attributes in enrollment request from both enterprise and stand-alone CAs. Therefore, the following sections document the unique procedures for each CA type. V2 templates are only available if the CA was installed on Windows Server 2003, Enterprise Edition.

Windows Server 2003 Stand-alone CA

A stand-alone CA issues certificates based on fixed rules similar to those in templates. As described previously, a stand-alone CA issues certificates similar to the *Domain Controller* certificate template; thus, you cannot enroll *Directory Email Replication* or *Domain Controller Authentication* certificates with this CA type.

When issuing certificates manually, it does not matter if the stand-alone CA is a member of a domain or a workgroup. Both configurations are supported, and since the subject alternative name information was included in the certificate request when reqdccert.vbs was run, it is not required to place the certificate requests in a pending state to process them correctly.

To issue a domain controller certificate from a stand-alone Windows Server 2003 CA, perform the following steps.

1. Log on to the CA computer.
2. Copy the certificate request and the batch script (<dcname>-req.bat) that was created on the domain controller previously into a working folder on the CA.
3. From a command-line prompt, run the <dcname>-req.bat script to perform the certreq command. The script will request a certificate based on the *DomainController* certificate template because no other template exists for this purpose.

   The bat-file contains the following command.

   ```
   CERTREQ -attrib "CertificateTemplate:DomainController" <requestfile>
   ```

4. A window will appear where you can select the CA that will issue the certificate. Select the issuing CA and click OK.
5. Make a note of the RequestID that is shown after the command has finished. This value will be needed later.
6. If the enrollment handling is set to the default configuration on a stand-alone CA, you will have to approve and issue the pending certificate. If the CA is configured to issue certificates automatically, continue to step 15.
7. To verify and issue the pending request, open the **Certificate Authority MMC Snap-In**.
8. In the left pane, click the *Pending Certificates* container.
9. In the right pane, select the certificate that corresponds to the RequestID noted in step 5.
10. On the Action menu, select All Tasks - View Attributes/Extensions.
11. A window opens that displays the request properties. Click the Extensions tab.
12. Verify the value of the Subject Alternative Name tag. It should display the hexadecimal encoded GUID and the FQDN of the domain controller.
13. Click OK to close the window.
14. With the certificate request still selected in the right pane, choose All Tasks - Issue on the Action menu.

    Alternatively, you can issue certificates from the command-line prompt with the following command.

    ```
    certutil –resubmit <RequestID>
    ```

    Replace the <RequestID> variable with the RequestID that was noted in step 5.
15. Once the certificate has been issued, you will have to store the certificate as a file to transfer it to the domain controller. The following command will create two certificate files. The CER file contains only the domain controller certificate; the P7B file contains the domain controller certificate and all of its parent certificates. At a command-line prompt, run the following command.

    ```
    CERTREQ –retrieve <RequestID> <dcname>.cer <dcname>.p7b
    ```

    Replace the <RequestID> variable with the RequestID that was used in the previous commands.
16. A window will appear where you can select the CA that has issued the certificate. Select the issuing CA and click OK.
17. Store the retrieved certificates on a diskette or other medium to transfer to the domain controller.
18. Log off the CA.

Next, continue to Domain Controller Certificate Installation.

Windows Server 2003 Enterprise CA

An enterprise CA maintains and uses certificate templates in Active Directory for all certificate request processing and issuance. Therefore, domain controller certificates are formatted and issued based on the templates that are available and assigned to a CA. In the case of offline domain controller certificates, modification of the default certificate templates in Active Directory is necessary to support the enrollment processing.

Since the following section requires V2 templates, it is assumed that the enterprise CA was installed on a server with Windows Server 2003, Enterprise Edition.

Certificate Template Creation

In the following steps, duplicates of the existing *Domain Controller* certificate templates are created to support offline request processing. The following are the two primary reasons for modifications.

- By default, the offline *Directory Email Replication* certificate template publishes certificates in Active Directory. Normally, this would be satisfactory; however, the CA

publishes an issued certificate into the certificate requestors Active Directory object and not into the certificate subject's Active Directory object. When a certificate request is submitted manually, the requestor is always the account of the user who created the certificate request. Thus, the CA would publish a domain controller certificate in the administrator's Active Directory user object. To work around this issue, a new certificate template is created that does not require the CA to publish the domain controller certificate in Active Directory. For information on manual certificate publication, see Publishing Domain Controller Certificates.

- Both *Directory Email Replication* and *Domain Controller Authentication* certificate templates require a special template flag to enable offline request processing. To leave the existing templates intact for all normal domain controller certificates processing, duplicates are made of both templates with the special settings.

  Note: It is recommended that you do not manipulate the *Directory Email Replication* and *Domain Controller Authentication* default templates. They are used for auto-enrollment by Windows Server 2003 domain controllers that can connect to the enterprise CA. Modification of these templates may cause an interruption of service for auto-enrollment of connected domain controllers.

Perform the following steps to duplicate the default *Domain Controller* certificate templates.

1. Log on with Enterprise Admin permissions to a domain computer that has Windows Server 2003 Administration Tools (AdminPak.msi) installed and is a member of the forest where the enterprise CA is installed. Any computer joined to the Active Directory forest can be used to maintain certificate templates since they are stored in Active Directory. Connectivity to a CA is not required at this point. By default, only members of the Enterprise Administrators group have permission to change certificate templates.
2. Click the Start button, and then point to Run. Type *certtmpl.msc* and press Enter.
3. The certificate Templates MMC Snap-In opens. In the right pane, select the Directory Email Replication template.
4. On the Action menu, select Duplicate template.
5. The Properties window of the new template appears. On the General tab, type *Offline Directory Email Replication* as Template display name. Click to clear the Publish certificate in Active Directory check box.

   Technically, any template name may be used; however, a meaningful label is recommended.
6. Click the Subject Name tab. Click to select the Supply in the request option and click OK.
7. In the right pane, select the Domain Controller Authentication template.
8. On the Action menu, select Duplicate template.
9. The Properties window of the new template appears. In the General tab, type *Offline Domain Controller Authentication* as Template display name. The Publish certificate in Active Directory check box is already clear because, by default, authentication certificates are not published in Active Directory.
10. Click the Subject Name tab. Click to select the Supply in the request option and click OK.
11. Close the Certificate Templates MMC Snap-In.

Certificate Template Configuration
Note that the template change is not required for CAs running Windows Server 2003 Service Pack 1 or later. If you have installed Windows Server 2003 Service Pack 1 or a later version on your CA, continue to **Issuing a Domain Controller Certificate**.

Typically, an enterprise CA will read information that is stored in several certificate attributes from the requestor's object in Active Directory. For example, the domain controller's common name or its GUID are derived from the domain controller's computer object and inserted into a certificate when a new domain controller certificate is manually or automatically enrolled through normal processes. If a domain controller certificate is requested manually through an offline or asynchronous process, all information that is required in the certificate must be explicitly specified in the certificate request.

When the certificate request is submitted to the CA, the request contains a field with the certificate template's common name, which is subsequently used by the CA to determine the enrollment and issuance policies. One such policy is whether the request may define the subject and the subject alternative name. By default, these attributes must not be set in a certificate request, and the CA will ignore such information in a request. It is important to note that auto-enrollment requires that the request not contain the subject information to process requests properly.

To configure the CA to allow the subject alternative name to be specified in certificate requests, the certificate templates that have been created in a previous section must be modified. Fortunately, this is a one-time operation, and it is not necessary to undo the change after a domain controller certificate has been enrolled. The template change will replicate normally such as any other attribute value change in your Active Directory environment and will not generate any schema changes.

To configure and apply the template modification before issuing an offline domain controller certificate, perform the following steps.

1. Log on as the Enterprise Administrator to a computer that is a member of the forest where the enterprise CA was installed.
2. Make the script available to your local computer as shown in FixDCtemplate.vbs.
3. Run the following script with the specified parameters.

   ```
   fixdctemplate.vbs <Templatename>
   ```

   Replace <Templatename> with the template's common name *OfflineDirectoryEmailReplication*. (Do not use blank spaces.)

   For more information on all the steps performed by the script, see FixDCtemplate.vbs.
4. Run the following script again with the specified parameters.

   ```
   fixdctemplate.vbs <Templatename>
   ```

   In this case, replace <Templatename> with the template's common name *OfflineDomainControllerAuthentication*. (Do not use blank spaces.)
5. Log off the computer.

Windows Server 2003 CA Configuration
In the previous section, the certificate templates have been created and configured. Once the changes in Active Directory have been replicated throughout the forest, the CA will be able to access and use these templates. However, in order for the CA to use and issue certificates based on the templates, the templates must be

manually published on each CA before they will be available.

Perform the following steps to publish a template in an enterprise CA.

1. Log on to the computer where the CA was installed as a local machine administrator.
2. Open the Certificate Authority MMC Snap-In.
3. Expand the CA object in the right pane and select Certificate Templates.
4. On the Action menu, select New - Certificate Template to Issue.
5. The template selection window appears. Press and hold the <CTRL> key and select the templates Offline Directory Email Replication and Offline Domain Controller Authentication.
6. Click OK to add the templates to the CA.

   Both templates will appear in the right pane of the Certificate Authority MMC Snap-In.
7. Close the Certificate Authority MMC Snap-In.

Issuing a Domain Controller Certificate with a Windows Server 2003 CA
The steps to issue a domain controller certificate from a Windows Server 2003 enterprise CA are similar to the Windows Server 2003 stand-alone CA procedures. However, since the Windows Server 2003 enterprise CA supports the new *Directory Email Replication* and *Domain Controller Authentication* certificate templates, these templates are used instead of the *Domain Controller* template.

Depending on the specific certificate template that is used, different attributes are specified as the subject alternative name. The *Directory Email Replication* requires the domain controllers GUID and its fully qualified DNS name (FQDN) in this extension. The *Domain Controller Authentication* template requires the FQDN only to be included in the subject alternative name extension. Therefore, you will find that the INF file and the certificate request look different for both certificate types.

Important: For offline domain controller certificate requests, never use the *Domain Controller* certificate template because a Windows Server 2003 CA supersedes that template with the *Directory Email Replication* and *Domain Controller Authentication* certificate templates. You will receive error 0x80094803 if you use the *Domain Controller* certificate template.

To issue a domain controller certificate from a Windows Server 2003 enterprise CA, perform the following steps.

1. Log on to the CA computer.
2. Copy the certificate request and the batch script (<dcname>-req.bat) that was created on the domain controller in a previous step into a working folder on the CA.
3. From a command-line prompt, use the <dcname>-req.bat script to run the certreq command. The script will request a certificate based on the given certificate template.

   The bat file contains the following command if the *Directory Email Replication* certificate template is used.

   ```
   CERTREQ -attrib "CertificateTemplate:<TemplateName>" <requestfile>
   ```

   A window will appear where you can select the CA that will issue the certificate.
4. Select the issuing CA and click OK.
5. If you have changed the default enrollment handling in one of the *Domain Controller* certificate templates or in the CA policy module configuration, make a note of the RequestID that is shown after the previous command has finished to issue the pended certificate. Otherwise, a Save window appears where you can set the name of the certificate file. Type the name of the certificate and click OK. Continue to step 9.
6. If the certificate request was pended, run the following command at a command-line prompt. Otherwise, continue to step 7.

   ```
   certutil –resubmit <RequestID>
   ```

   Replace <RequestID> with the RequestID that was recorded previously.
7. Once the certificate is issued, you have to store it as a file to transfer it to the domain controller. The following command will create two certificate files. The CER-file contains only the domain controller certificate; the P7B file contains the domain controller certificate and all of its parent certificates. At a command-line prompt, run the following command.

   ```
   CERTREQ –retrieve <RequestID> <dcname>.cer <dcname>.p7b
   ```

   Replace <RequestID> with the RequestID that was used in the previous commands.
8. A window will appear where you can select the CA that has issued the certificate. Select the issuing CA and click OK.
9. Store the retrieved certificates on a diskette or other medium to transfer to the domain controller.
10. Log off the CA.

Top of page

Domain Controller Certificate Installation

Installing a certificate on a domain controller implies that the certificate and associated private key are available to the local system (computer) account. Since the key material has been previously generated and the certificate request is still pending at the domain controller, the certificate must be accepted (installed). Acceptance refers to the operation whereby the certificate and the key material are linked, and the certificate request is deleted from the *Certificate Enrollment Requests* container.

When an Administrator accepts a certificate, certreq will look into the *Certificate Enrollment Requests* container of the local machine store first, and if no corresponding

request is found, it looks into the current user's (Administrator) *Certificate Enrollment Requests* container. Thus, you can accept machine certificate requests as Administrator.

Installing Domain Controller Certificates

The following procedure will install the domain controller certificate locally in the local system profile but not in Active Directory. To install the certificate on the target domain controller, perform the following steps with the Windows Server 2003 version of certreq and certutil. On a Windows 2000 domain controller, you must add a prefix to the commands. The prefix is the path you have copied the commands to. In this white paper, the *%HOMEDRIVE%\W2K3AdmPak* path is used.

1. Log on to the target domain controller.
2. Make the CER- and P7B-file from the previous section available to the domain controller.
3. From a command-line prompt, run the following command.

   ```
   CERTREQ –ACCEPT <dcname>.p7b
   ```

   Replace <dcname> with the name of the target domain controller. The command will not report any confirmation of success.
4. Verify that the certificate has been installed in the local system personal certificate store by running the following command at a command-line prompt:

   ```
   certutil –viewstore My
   ```

   A window will appear that displays all certificates that are available in the local computer personal store.
5. Log off your domain controller.

Publishing Domain Controller Certificates

As mentioned previously, a stand-alone CA is not capable of publishing certificates in Active Directory. However, SMTP replication requires the use of domain controller certificates published in Active Directory. For certificates that are enrolled asynchronously through an offline process, you must manually publish these certificates in Active Directory.

The following steps instruct you to examine the certificates that reside in the domain controller's local machine certificate store before the new certificate is published in Active Directory. The new certificate is published and you will be able to verify that the certificate was published properly.

Publishing certificates into computer objects in Active Directory requires write permissions for the userCertificate attribute that is part of any computer object. Administrators and members of the built-in domain group "Cert Publishers" have this permission by default.

The reqdccert.bat script creates a file called <dcname>-vfy.bat that contains the correct certutil command to verify the domain controller's certificate in Active Directory. You can use this batch file instead of typing the certutil –viewstore commands manually as described in the following steps.

Perform the following steps to manually view and publish a domain controller certificate in Active Directory.

1. Log on as domain administrator or a member of the Cert Publishers global group for the target domain controller. Technically, the publication can be performed at any computer that is a domain member, but for convenience, the domain controller is used in this scenario.
2. Verify that there are no certificates already published on the domain controller's Active Directory object.

   Note: The following steps work only with the Windows Server 2003 version of certutil.exe. Thus, if you perform the following steps from a Windows 2000 domain controller, you must add a prefix to the certutil command. The prefix is the path you have copied the certutil command to. In this white paper, *%HOMEDRIVE%\W2K3AdmPak* is used.

   Run the following command from a command-line prompt.

   ```
   certutil –viewstore "ldap:///cn=<dcname>,ou=domain controllers,dc=<domainname>,dc=<com>?usercertificate"
   ```

   Replace the <dcname> variable with the name of the target domain controller and <domainname> and <com> variable names with the appropriate domain suffix.

   A window should appear with no certificates displayed. This is expected since no certificates have been published yet.
3. Click Cancel to close the window.
4. The certificate is published in Active Directory using the *userCertificate* attribute on the machine account object for the domain controller. Run the following command to write the certificate to the domain controller's Active Directory object.

   ```
   certutil –f –dspublish <dcname>.cer machine
   ```

   Replace the <dcname> variable with the name of the target domain controller.

   The command determines the proper Active Directory object by the subject information in the certificate. The publication will fail if no object can be found based on the subject information.

   Note: The use of the "machine" parameter is a mandatory requirement in the previous command example.
5. To verify that the certificate was published successfully, perform the following steps from a command-line prompt.

   ```
   certutil –viewstore "ldap:///cn=<dcname>,dc=<domainname>,dc=<com>?usercertificate"
   ```

   If the domain controller's computer object has no certificates in the *userCertificate* attribute, the certutil output will display an empty list in the window. If

"?userCertificate" was omitted from the command line parameters or an invalid object class was specified, an error message will appear such as the following:

```
CertUtil: -viewstore command FAILED: 0x80092009 (-2146885623)
CertUtil: Cannot find the requested object.
```

Note: It is always a good practice to verify the certificates in the requestor's Active Directory object as well. If the certificate templates have not been configured correctly and the certificate template was configured to publish certificates in Active Directory, these certificates may be published to the user account that created the certificate request. To examine certificates in the domain administrator's Active Directory user object, run the following command.

```
certutil –viewstore "ldap:///cn=Administrator,cn=Users,dc=<domainname>,dc=<com>"
```

If computer certificates are found in this object, follow the instructions to remove them in Removing Certificates from an Active Directory Computer Object.

Top of page

Removing Domain Controller Certificates

Occasionally, it may be necessary to remove or delete unwanted certificates locally in a certificate store, or remotely in Active Directory, to either correct mistakes or to perform periodic system cleanup. The following section describes the procedures to perform these activities. However, removing a certificate from a certificate store or an Active Directory object is not specific to domain controller certificates. The procedures are the same for any user or computer certificate type.

Removing Certificates from a Local Certificate Store

It may be necessary to remove certificates locally from a domain controller to ensure that only valid certificates are used by the domain controller or other applications. Expired or revoked certificates may be removed from a domain controller since the purpose of domain controller certificates is to encrypt replication traffic. Once replication has been performed, the replication data is discarded and there are no requirements to decrypt this information again. Therefore, such certificates may be safely deleted from the local machine profile.

To remove certificates from a dedicated domain controller, perform the following steps.

1. While logged on as a member of the local Administrators group, start the Microsoft Management console.
2. Add the Certificates MMC Snap-In.
3. Select Computer Account when prompted to select an account to manage.
4. In the Certificates MMC Snap-In, navigate to Personal in the left pane.
5. In the right pane, determine the domain controller certificate(s) by the template name as shown in the Certificate Templates column or select the certificate(s) by their intended purpose.
6. Delete the certificate(s) by selecting Delete on the Action menu.
7. Close the MMC Snap-In and log off.

Removing Certificates from an Active Directory Computer Object

In some cases, it may be necessary to remove certificates that are stored in an Active Directory object explicitly. Usually, this is the case if certificates have been enrolled manually. The auto-enrollment functionality in Windows XP and Windows Server 2003 can remove certificates from Active Directory objects when it determines that certificates in the Active Directory object have expired or are revoked. Also, the CA removes expired certificates when it publishes a new certificate in an object in Active Directory. For more information on the functionality performed by auto-enrollment, see the Certificate Autoenrollment in Windows Server 2003 white paper at http://www.microsoft.com/technet/prodtechnol/windowsserver2003/technologies/security/autoenro.mspx

To manually remove a certificate from a domain controller object in Active Directory, perform the following steps.

1. Log on as a domain or enterprise administrator to a computer in your domain where the Windows Server 2003 version of certutil is installed and available.

   Note: The following steps work only with the Windows Server 2003 version of certutil.exe. Thus, if you perform the following steps from a Windows 2000 domain controller, you must add a prefix to the certutil command. The prefix is the path you have copied the certutil command to. In this white paper, the *%HOMEDRIVE%\W2K3AdmPak* path is used.

2. At a command-line prompt, run the following command and press Enter.

   ```
   certutil -viewdelstore "ldap:///cn=<dcname>,ou=domain controllers,<domainname>?usercertificate"
   ```

   The command contains the distinguished name of the object and specifies the LDAP attribute explicitly. If the LDAP attribute is missing or an invalid attribute was specified, the command will fail with an "Access denied" error message.

   In the case of domain controller certificates, certificates are always stored in the *userCertificate* attribute. However, user objects may also have certificates in the *usercert* or *userSMIMEcertificate* attribute from other applications such as Outlook, which use these attributes as the preferred attribute for storing certificates.

   A window will appear displaying the array of certificates on the user object. If the specified object contains certificates in the given attribute, the certificates are shown in the window. If no certificates are available, the list in the window will be empty.

3. To remove a specific certificate, select the certificate and click OK.

   The window will close and a status message will be displayed at the command-line prompt similar to the following: "Deleted certificate <certificate subject name>" where the <certificate subject name> displays the subject name of the certificate.

4. To close the window without deleting a certificate, click Cancel.

Top of page

Troubleshooting

Despite the level of detail and documented procedures in this white paper, some scenarios and environments may encounter problems that require troubleshooting. This section contains some of the common errors and troubleshooting tips for the procedures in this white paper to assist in resolving potential problems in your environment.

The first troubleshooting tool shows the full descriptive text for an error code displayed by an application or tool. If an unexpected error appears, you can display the corresponding error message text with the following command.

```
certutil –error <Hexadecimal_Error_ID>
```

For example, to see the error text for error 0x80094800, run the following command at a command-line prompt.

```
certutil –error 0x80094800
```

Domain Controller Certificates Appear in User Objects

If a certificate template is configured for the CA to publish a certificate in Active Directory during the enrollment and issuance process, the CA will choose the requestor's Active Directory object instead of the domain controller's computer object. To correct this issue, follow the steps in Removing Certificates from an Active Directory Computer Object.

Certreq –new fails with error 0x80092023

This error may occur if the subject name was not specified in an X.500 format. You cannot specify just the raw string as the common name. You must at least add the prefix "CN=" to the string.

Certreq –submit fails with error 0x80094800

This common error occurs when the template is not available on the CA where the certificate request was submitted. To correct this issue, make sure that the template name is spelled correctly in the certificate request or as an –attrib parameter. Also, it is important to verify that you have performed the steps in **Windows Server 2003 CA Configuration**.

Certreq –submit fails with error 0x80094001

This error occurs if you have performed the steps described in Issuing Domain Controller Certificates with a Windows Server 2003 CA where the request uses the V1 *Domain Controller* certificate template and includes a subject or subject alternative name. The V1 domain controller template instructs an enterprise CA to read the subject name of the requestor from Active Directory, but this fails with manual certificate requests. An administrator's user object in Active Directory will obviously never have the appropriate subject name for a domain controller.

You must not enroll manual certificate requests with V1 certificate templates on an enterprise CA.

Certreq –submit fails with error 0x80094803

This error typically occurs when a certificate template does not allow the subject or subject alternative name to be explicitly specified in the certificate request. By default, a template only allows information to be retrieved from Active Directory when building the subject or subject alternative name in a certificate. To correct this error for offline certificate enrollment, modify the specific template by running the fixdctemplate.vbs script in Appendix 2.

To verify the mandatory attributes of a subject alternative name in a template, run the following command at a command-line prompt with the Windows Server 2003 versions of certutil. On a Windows 2000 computer, you must add a prefix to the commands. The prefix is the path you have copied these commands to. In this white paper, the *%HOMEDRIVE%\W2K3AdmPak* path is used. You must be logged on as a member of the Authenticated User group in the Active Directory forest.

```
Certutil –v –dstemplate {CertificateTemplate_commonname}
```

For example, type

```
certutil -v -dstemplate OfflineDomaincontrollerauthentication
```

The command will display the properties of the specified template. Examine the parameters following *msPKI-Certificate-Name-Flag* in the output. Parameters that are indented in the output are disabled. Ensure that you have specified all attributes in the certificate request that are not indented in the output of the template properties.

Certreq –submit fails with error 0x8009480e

This error most commonly occurs when the template is not available on the CA to which you have submitted the request. This situation may also be corrected by running the fixtemplate.vbs script in Appendix 2. See also Certreq –submit fails with error 0x80094803.

Certutil –viewstore displays an empty dialog with no certificates

This error most commonly occurs when an invalid object class was specified in the command-line parameter(s). You can use ADSIedit from the Windows Server 2003 Support Tools on the Windows Server 2003 CD-ROM to identify the correct object class and its distinguished name.

Top of page

Appendix 1: Identifying a Domain Controller GUID

Identifying the correct domain controller GUID for an SMTP replication certificate may pose a challenge for some administrators who are unfamiliar with the nuances of Active Directory and domain controller objects in the directory. To determine a specific domain controller GUID from a Windows XP or Windows Server 2003 computer joined to the Active Directory forest, perform the following steps.

Note The dsquery utility is part of the Windows Server 2003 Administration Tools Pack and is not available on Windows 2000 computers.

1. Log on to the computer with a domain account.

2. From a command-line prompt, run the following command.

```
dsquery * "CN=<hostname>,OU=Domain Controllers,DC=<yourdomain>,DC=<yourdomain>" –scope base –attr objectguid
```

You must replace the <hostname> variable with the name of the specific domain controller you want and the <yourdomain> variable with the domain name of your specific domain. For example:

```
dsquery * "CN=DC01,OU=Domain Controllers,DC=contoso,DC=com" –scope base –attr objectguid
```

The command will result in output similar to the following:

```
Objectguid
{57A8AAF4-686E-4128-8712-B6CA89FBF5BC}
```

3. Log off the computer.

Top of page

Appendix 2: Sample Scripts

This section contains two sample Visual Basic scripts that may be used to simplify or customize the various processes described in previous sections.

Reqdccert.vbs – Generates Domain Controller Certificate Requests

The reqdccert.vbs script makes it easier to create the correct INF file that is required to submit a certificate request to a Windows 2000 or Windows Server 2003 CA. Since it is fairly complicated to determine the GUID of a computer and create an ASN.1 file that contains both the GUID and the DNS name, the script generates the ASN.1 file automatically. In addition, a batch file is created that contains the certreq –submit command(s), which are required to submit the certificate request to the CA. Finally, a validation script is generated that allows you to verify the certificate(s) on a domain controller's computer object after you have published the domain controller certificate(s) in Active Directory.

If you plan to submit a certificate request to a Windows 2000 or a Windows Server 2003 stand-alone CA, run the script without any command-line parameters. If you plan to submit the certificate request to a Windows Server 2003 enterprise CA, you must specify the name of the certificate template that will be used to enroll the domain controller certificate.

Note: Since the script requires the Windows Server 2003 version of certutil for an internal encoding operation, it is highly recommended, on a Windows 2000 domain controller, to put this script in the same directory as the Windows Server 2003 version of certutil and certreq.exe.

```
Set oArgs = WScript.Arguments
Set oShell = WScript.CreateObject("WScript.Shell")
'
' Parse command line
'
if oArgs.Count < 1 then
    sTemplateName = "DomainController"
    sType = "E"
else
    if ((oArgs(0) = "-?") or (oARgs.Count < 2)) then
        Wscript.Echo "Usage: reqdccert.vbs [Templatename] [Type]"
        Wscript.Echo "[Templatename] is the name of a V2 template"
        Wscript.Echo "[Type]         can be E for Email and A for Authentication certificate"
        Wscript.Echo "If no option is specified, the DomainController certificate template is used."
        Wscript.Quit 1
    else
        sTemplateName = oArgs(0)
                sType = oArgs(1)
    end if
end if
Set oFilesystem = CreateObject("Scripting.FileSystemObject")
Set objSysInfo = CreateObject("ADSystemInfo")
Set objDC = GetObject("LDAP://" & objSysInfo.ComputerName)
sGUID = objDC.GUID
sDNShostname = objDC.DNShostname
sHostname = objDC.cn
'###############################################################################
'
' Create the ASN.1 file
'
'###############################################################################
Dim aASNsubstring(2, 5)
Const HEX_DATA_LENGTH = 1
Const ASCIIDATA = 2
Const HEXDATA = 3
Const HEX_BLOB_LENGTH = 4
Const HEX_TYPE = 5
aASNsubstring(0, ASCIIDATA) = sDNShostname
aASNsubstring(0, HEX_TYPE) = "82"
'
' Convert DNS name into Hex
'
For i = 1 to Len(aASNsubstring(0, ASCIIDATA))
    aASNsubstring(0, HEXDATA) = aASNsubstring(0, HEXDATA) & _
                                Hex(Asc(Mid(aASNsubstring(0, ASCIIDATA), i, 1)))
Next
aASNsubstring(0, HEX_DATA_LENGTH) = ComputeASN1 (Len(aASNsubstring(0, HEXDATA)) / 2)
'
' Build the ASN.1 blob for DNS name
'
```

```
sASN = aASNsubstring(0, HEX_TYPE) & _
       aASNsubstring(0, HEX_DATA_LENGTH) & _
       aASNsubstring(0, HEXDATA)
'
' Append the GUID as other name
'
if (sType = "E") then
    aASNsubstring(1, HEXDATA) = sGUID
    aASNsubstring(1, HEX_TYPE) = "A0"
    aASNsubstring(1, HEX_DATA_LENGTH) = ComputeASN1 (Len(aASNsubstring(1, HEXDATA)) / 2)
    sASN = sASN & _
           "A01F06092B0601040182371901" & _
           aASNsubstring(1, HEX_TYPE) & _
           "120410" & _
           aASNsubstring(1, HEXDATA)
end if
'
' Write the ASN.1 blob into a file
'
Set oFile = oFilesystem.CreateTextFile(sHostname & ".asn")
'
' Put sequence, total length and ASN1 blob into the file
'
oFile.WriteLine "30" & ComputeASN1 (Len(sASN) / 2) & sASN
oFile.Close
'
' Use certutil to convert the hex string into bin
'
oShell.Run "certutil -f -decodehex " & sHostname & ".asn " & _
                                       sHostname & ".bin", 0, True
'
' Use certutil to convert the bin into base64
'
oShell.Run "certutil -f -encode " & sHostname & ".bin " & _
                                    sHostname & ".b64", 0, True
'###########################################################################
'
' Create the INF file
'
'###########################################################################
Set iFile = oFilesystem.OpenTextFile(sHostname & ".b64")
Set oFile = oFilesystem.CreateTextFile(sHostname & ".inf")
oFile.WriteLine "[Version]"
oFile.WriteLine "Signature= " & Chr(34) & "$Windows NT$" & Chr(34)
oFile.WriteLine ""
oFile.WriteLine "[NewRequest]"
oFile.WriteLine "KeySpec = 1"
oFile.WriteLine "KeyLength = 1024"
oFile.WriteLine "Exportable = TRUE"
oFile.WriteLine "MachineKeySet = TRUE"
oFile.WriteLine "SMIME = FALSE"
oFile.WriteLine "PrivateKeyArchive = FALSE"
oFile.WriteLine "UserProtected = FALSE"
oFile.WriteLine "UseExistingKeySet = FALSE"
oFile.WriteLine "ProviderName = " & Chr(34) & _
                "Microsoft RSA SChannel Cryptographic Provider" & Chr(34)
oFile.WriteLine "ProviderType = 12"
oFile.WriteLine "RequestType = PKCS10"
oFile.WriteLine "KeyUsage = 0xa0"
oFile.WriteLine ""
oFile.WriteLine "[EnhancedKeyUsageExtension]"
oFile.WriteLine "OID=1.3.6.1.5.5.7.3.1"
oFile.WriteLine "OID=1.3.6.1.5.5.7.3.2"
oFile.WriteLine ";"
oFile.WriteLine "; The subject alternative name (SAN) can be included in the INF-file"
oFile.WriteLine "; for a Windows 2003 CA."
oFile.WriteLine "; You don't have to specify the SAN when submitting the request."
oFile.WriteLine ";"
oFile.WriteLine "[Extensions]"
iLine = 0
Do While iFile.AtEndOfStream <> True
    sLine = iFile.Readline
    If sLine = "-----END CERTIFICATE-----" then
        Exit Do
    end if
    if sLine <> "-----BEGIN CERTIFICATE-----" then
        if iLine = 0 then
            oFile.WriteLine "2.5.29.17=" & sLine
        else
            oFile.WriteLine "_continue_=" & sLine
        end if
        iLine = iLine + 1
    end if
Loop
oFile.WriteLine "Critical=2.5.29.17"
oFile.WriteLine ";"
oFile.WriteLine "; The template name can be included in the INF-file for any CA."
oFile.WriteLine "; You don't have to specify the template when submitting the request."
oFile.WriteLine ";"
oFile.WriteLine ";[RequestAttributes]"
oFile.WriteLine ";CertificateTemplate=" & sTemplateName
oFile.Close
iFile.Close
'###########################################################################
'
' Create the certreq.exe command-line to submit the certificate request
'
```

```
'#########################################################################
Set oFile = oFilesystem.CreateTextFile(sHostname & "-req.bat")
oFile.WriteLine "CERTREQ -attrib " _
                & Chr(34) & "CertificateTemplate:" & sTemplateName _
                & Chr(34) & " " & sHostname & ".req"
'
' The GUID structure needs to be reconstructed. The GUID is read
' as a string like f4aaa8576e6828418712b6ca89fbf5bc however the
' format that is required for the certreq command looks like
' 57a8aaf4-686e-4128-8712-b6ca89fbf5bc. The bytes are reordered
' in the following way:
'
'                          11111111112222222222333
'             Position 12345678901234567890123456789012
'                      |------|--|--|--|--------------|
' Original GUID:       f4aaa8576e6828418712b6ca89fbf5bc
'
'                          11 1 1111 1112 222222222333
'             Position 78563412 1290 5634 7890 123456789012
'                      |------- |--- |--- |--- |----------|
' Reformatted GUID:    57a8aaf4-686e-4128-8712-b6ca89fbf5bc
'
oFile.WriteLine "REM "
oFile.WriteLine "REM !!! Only valid for Windows 2003 or later versions !!!"
oFile.WriteLine "REM If you do not specify certificate extensions in the *.INF file"
oFile.WriteLine "REM they can be specified here like the following example"
oFile.WriteLine "REM "
oFile.WriteLine "REM CERTREQ -submit -attrib " _
                & Chr(34) & "CertificateTemplate:" & sTemplateName _
                & "\n" _
                & "SAN:guid=" _
                & Mid(sGUID, 7, 2) _
                & Mid(sGUID, 5, 2) _
                & Mid(sGUID, 3, 2) _
                & Mid(sGUID, 1, 2) & "-" _
                & Mid(sGUID, 11, 2) _
                & Mid(sGUID, 9, 2) & "-" _
                & Mid(sGUID, 15, 2) _
                & Mid(sGUID, 13, 2) & "-" _
                & Mid(sGUID, 17, 4) & "-" _
                & Mid(sGUID, 21, 12) _
                & "&DNS=" & sDNShostname & Chr(34) & " " & sHostname & ".req"
oFile.Close
'#########################################################################
'
' Create the certificate verification script
'
'#########################################################################
Set oFile = oFilesystem.CreateTextFile(sHostname & "-vfy.bat")
oFile.WriteLine "certutil -viewstore " & Chr(34) & objDC.distinguishedname & _
                "?usercertificate" & chr(34)
oFile.Close
'#########################################################################
'
' Compute the ASN1 string
'
'#########################################################################
Function ComputeASN1 (iStrLen)
    If Len(Hex(iStrLen)) Mod 2 = 0 then
        sLength = Hex(iStrLen)
    else
        sLength = "0" & Hex(iStrLen)
    end if
    if iStrLen > 127 then
        ComputeASN1 = Hex (128 + (Len(sLength) / 2)) & sLength
    else
        ComputeASN1 = sLength
    End If
End Function
```

FixDCtemplate.vbs

The fixDCtemplate.vbs script simplifies a change that is required for any certificate template that needs to accept the subject and subject alternative name as part of a certificate request. To change the *msPKI-Certificate-Name-Flag* in the certificate templates object, Active Directory Service Interface (ADSI) is used.

The *msPKI-Certificate-Name-Flag* attribute is actually a bit field, where the first bit determines whether the template accepts the subject and subject alternative name. Before changing the bit, the script determines if the bit is already set.

```
Set oArgs = WScript.Arguments
'
' Parse command line
'
if oArgs.Count <> 1 then
    Wscript.Echo "fixdctemplate {templatename}"
    WScript.Quit 1
end if
Set objRoot = GetObject("LDAP://rootDSE")
Set objTemplate = GetObject("LDAP://CN=" & oArgs(0) & _
                            ",CN=Certificate Templates," & _
                            "CN=Public Key Services,CN=Services," & _
                            objRoot.Get("configurationNamingContext"))
iNameFlag = objTemplate.Get("msPKI-Certificate-Name-Flag")
if iNameFlag = 1 then
    Wscript.Echo "Flag is already set"
else
```

```
        objTemplate.Put("msPKI-Certificate-Name-Flag"), 1
        objTemplate.SetInfo
        Wscript.Echo "Flag was set"
    end if
```

Top of page

Appendix 3: Certreq.exe Syntax

Certreq.exe is a command-line tool that is included in Windows 2000 Server, Windows XP, and Windows Server 2003 as well as in the Windows Server 2003 Administration Tools Pack. The version available with Windows XP and Windows Server 2003 is different from the version that is included in Windows 2000 Server. When using certreq.exe, you must distinguish between these versions.

You can perform the following tasks with certreq.exe.

- Create new certificate requests. This includes the generation of key material that corresponds to the certificate request.
- Apply policies to a certificate request. This applies to CA certificates where various constraints are required. Since you cannot include certificate policies in the certificate request that is created by a CA, you have to apply the policy to the CA certificate request explicitly.
- Submit certificate requests to a certification authority. This feature also works with remote certification authorities, so that you can submit a request via the network to a remote CA.
- Retrieve certificates from a CA. Based on the certificate request ID, you can retrieve any certificate from a CA and save it to a file.
- Accept (and install) certificates or responses. This command is intended to accept a certificate or a response that contains a certificate. Once you have created a certificate request with certreq.exe, it remains in a pending state on your computer until you install the certificate that corresponds to the request.
- Sign certificate requests. In the case where the certificate template administrator has set the number of certificate enrollment agent signatures to a value greater than zero, you can sign certificate requests with this option.

The Windows 2000 version of certreq.exe can only submit certificate requests and retrieve issued certificates. All other options require the certreq.exe version that comes with Windows XP or Windows Server 2003. For more information on the command-line syntax, run certreq,exe with the following parameters to display all available syntax and parameter information.

```
Certreq.exe -v -?
```

Creating Certificate Requests

The –new parameter with certreq.exe is used to construct new certificate requests. Certreq.exe uses an INF file as an input option that defines the certificate request parameters. When the certificate request is constructed based on the INF file, key material is also generated. A certificate request process can be divided into the following elements.

- Read the INF file.
- Create the private and public key based on the information in the INF file, and store the private key in the local key store of the user or machine profile as appropriate.
- Create the certificate request based on the information in the INF file and store the request in a Base64-encoded file or, optionally, in binary form if the –binary option is selected.

Certreq.exe INF File Structure

Since the INF file allows for a rich set of parameters and options to be specified, it is difficult to define a default template that administrators should use for all purposes. Therefore, this section describes all the options to enable you to create an INF file tailored to your specific needs.

The following key words are used to describe the INF file structure.

- A *section* is an area in the INF file that covers a logical group of keys. A section always appears in brackets in the INF file.
- A *key* is the parameter that is to the left of the equal sign.
- A *value* is the parameter that is to the right of the equal sign.

For example, a minimal INF file would look similar to the following:

```
[NewRequest]
; At least one value must be set in this section
Subject = "CN=W2K3-BO-DC.contoso2.com"
```

Note: You can remark lines by putting a semi-colon (;) in front of the line.

The following are some of the possible sections that may be added to the INF file.

[Version]
This section is optional and is not required in an INF file. If present, it must contain the Signature key.

Signature
The signature key must be equal to the fixed string "$Windows NT$".

[NewRequest]
This section is mandatory for an INF file that acts as a template for a new certificate request. If this section is missing, the following error message will be displayed.

"INF file line not found 0xe0000102 (INF: -536870654)"

This section requires at least one key with a value. If this section is empty and has no keys, the following error message will be displayed.

"Incorrect function. 0x1 (WIN32: 1)"

EncipherOnly

Syntax	EncipherOnly={Boolean}
Values	TRUE or FALSE
Default value	FALSE
Sample	`EncipherOnly = TRUE`
Supported with CA type	Windows 2000 Windows Server 2003
Can be manipulated in a pending request	Yes (through the key usage extension)

This parameter has an impact on the functional capabilities of the public-private key pair. If the value is set to TRUE, the key can exclusively be used for "encipherment" (encryption). If it is set to FALSE, the key can be used for "encipherment" and other purposes. This parameter refers only to the key type in CryptoAPI and has an indirect relationship only to the key usage extension that may be included in an issued X.509 certificate. The default value of the certificate enrollment control is set to FALSE. This setting only affects AT_KEYEXCHANGE key types and then correspondingly limits the key usage to key Encipherment, data Encipherment, or both. The digital signature and non-repudiation key usage(s) will be disallowed.

Exportable

Syntax	Exportable = {Boolean}
Values	TRUE or FALSE
Default value	FALSE
Sample	`Exportable = TRUE`
Supported with CA type	Windows 2000 Windows Server 2003
Can be manipulated in a pending request	No

This parameter is ignored when the "UseExistingKeySet" key is set to TRUE because you can set the exportable flag only when a new key is created. You cannot change this flag for an existing key. If this attribute is set to TRUE, the private key can be exported with the certificate. To ensure a high level of security, private keys should not be exportable; however, in some cases, it might be required to make the private key exportable if several computers or users must share the same private key. For example, a Web server that runs SSL and is published with Internet Security and Acceleration (ISA) to the Internet requires a certificate with an exportable key because the certificate and key must be installed on the Web server and the ISA server as well. Also, if it is required that the keys be managed and backed up, they must be exportable to provide for this functionality.

This setting correlates with the template configuration if a Windows Server 2003 enterprise CA is used. A certificate template administrator can explicitly define if private keys should be exportable.

Note: The template setting for exportable has no affect on the exportability properties of a key generated through certreq.exe -new.

KeyContainer

Syntax	KeyContainer = {Key_containerName}
Values	Random string value (GUID)
Default value	None
Sample	`KeyContainer = {C347BD28-7F69-4090-AA16-BC58CF4D749C}`
Supported with CA type	Windows 2000 Windows Server 2003
Can be manipulated in a pending request	No

It is not recommended to set this parameter for new requests where new key material is generated. The key container is automatically generated and maintained by the system.

For requests where the existing key material should be used, this value can be set to the key-container name of the existing key.

Use the certutil –key command to display the list of available key containers for the machine context.

Use the certutil –key –user command for the current user's context.

If you need to find the key container for a specific certificate, use the certutil –store my command for machine certificates.

To find the key container for a specific certificate in the current user's certificate store, use the certutil –store –user my command.

If the "UseExistingKeySet" key is set to TRUE, the key container must be set explicitly in the INF file or in the renewal certificate.

KeyLength

Syntax	KeyLength = {integer}

Values	Any valid key length that is supported by the cryptographic service provider
Default value	1024 [depends on the certificate service provider (CSP)]
Sample	`KeyLength=2048`
Supported with CA type	Windows 2000 Windows Server 2003
Can be manipulated in a pending request	No

The key length defines the length of the public and private key. The key length has an impact on the security level of the certificate. Greater key length usually provides a higher security level; however, some applications may have limitations regarding the key length.

KeySpec

Syntax	KeySpec = {integer}
Values	AT_EXCHANGE: 1 AT_SIGNATURE: 2
Default value	2
Sample	`KeySpec=1`
Supported with CA type	Windows 2000 Windows Server 2003
Can be manipulated in a pending request	No

The KeySpec determines if the key can be used for signatures, for Exchange (encryption), or for both. If the KeySpec is set to a value of 2, the EncipherOnly key is ignored.

KeyUsage

Syntax	KeyUsage = {hexadecimal_value}
Values	CERT_DIGITAL_SIGNATURE_KEY_USAGE 0x80 CERT_NON_REPUDIATION_KEY_USAGE 0x40 CERT_KEY_ENCIPHERMENT_KEY_USAGE 0x20 CERT_DATA_ENCIPHERMENT_KEY_USAGE 0x10 CERT_KEY_AGREEMENT_KEY_USAGE 0x08 CERT_KEY_CERT_SIGN_KEY_USAGE 0x04 CERT_OFFLINE_CRL_SIGN_KEY_USAGE 0x02 CERT_CRL_SIGN_KEY_USAGE 0x02 CERT_ENCIPHER_ONLY_KEY_USAGE 0x01
Default value	0xC0 for AT_SIGNATURE (default) For AT_KEYEXCHANGE, EncipherOnly = TRUE: 0x30 For AT_KEYEXCHANGE, EncipherOnly = FALSE: 0xf0
Sample	`KeyUsage=0xa0`
Supported with CA type	Windows 2000 Windows Server 2003
Can be manipulated in a pending request	Yes

The key usage defines what the certificate key should be used for. The value is a bit field that is composed of the key usage flags as they are defined in the Windows Platform Software Development Kit (SDK). (See the include file wincrypt.h.) Caution should be used in defining the correct key usage for the certificate request based on the intended application usage.

Note: The key usage extension has an indirect dependency with the extended key usage extension. For example, if you specify "Encrypting File System" as the extended key usage, you should also specify "Key Encipherment" as the key usage.

To combine multiple key usages, summarize the values of the individual key usages by performing hexadecimal arithmetic. The default key usage is set to Digital

Signature (0x80) and Non-Repudiation (0x40). The hexadecimal sum of 0x80 and 0x40 is in 0xC0.

The key usage parameter is primarily useful if certificate requests are submitted to stand-alone CAs. Only enterprise CAs have certificate key usages predefined through certificate templates. The key usage extension is set to critical by default.

MachineKeySet

Syntax	MachineKeySet = {Boolean}
Values	TRUE or FALSE
Default value	FALSE
Sample	`MachineKeySet = TRUE`
Supported with CA type	Windows 2000 Windows Server 2003
Can be manipulated in a pending request	No

This key is important when you need to create certificates that are owned by the machine and not a user. The key material that is generated is maintained in the security context of the security principal (user or computer account) that has created the request. When an administrator creates a certificate request on behalf of a computer, the key material must be created in the machine's security context and not the administrator's security context. Otherwise, the machine could not access its private key since it would be in the administrator's security context.

You must set this key to TRUE if you are creating requests for domain controllers, a Web server, or any other service that runs in the machine's security context.

PrivateKeyArchive

Syntax	PrivateKeyArchive = {Boolean}
Values	TRUE or FALSE
Default value	FALSE
Sample	`PrivateKeyArchive = TRUE`
Supported with CA type	Windows Server 2003
Can be manipulated in a pending request	No

The *PrivateKeyArchive* setting works only if the corresponding RequestType is set to "CMC" because only the Certificate Management Messages over CMS (CMC) request format allows for securely transferring the requester's private key to the CA for key archival. Only the Windows Server 2003 enterprise CA supports private key archival. The Windows Server 2003 enterprise CA must be online and accessible to retrieve the CA encryption certificate directly during this process.

ProviderName

Syntax	ProviderName = {CSP_stringname}
Values	The descriptive name of the certificate service provider
Default value	Microsoft Strong Cryptographic Provider
Sample	`ProviderName="Microsoft RSA SChannel Cryptographic Provider"`
Supported with CA type	Windows 2000 Windows Server 2003
Can be manipulated in a pending request	No

The provider name is the display name of the CSP. If you do not know the provider name of the CSP you are using, run certutil –csplist from a command line. The command will display the names of all CSPs that are available on the local system.

ProviderType

Syntax	ProviderType = {integer}
Values	The number that describes the provider type
Default value	12
Sample	`ProviderType=13`
Supported with CA type	Windows 2000 Windows Server 2003
Can be manipulated in a pending request	No

The provider type is used to select specific providers based on specific algorithm capability such as "RSA Full". If you do not know the provider type of the CSP you are using, run certutil –csplist from a command-line prompt. The command will display the provider type of all CSPs that are available on the local system.

RenewalCert

Syntax	RenewalCert={CertificateHash}
Values	The certificate hash of any certificate that is available at the computer where the certificate request is created
Default value	n/a
Sample	`RenewalCert=4EDF274BD2919C6E9EC6A522F0F3B153E9B1582D` `(must use double quotation marks if white space is used between pairs of hexadecimal values)`
Supported with CA type	Windows 2000 Windows Server 2003
Can be manipulated in a pending request	No

If you need to renew a certificate that exists on the system where the certificate request is generated, you must specify its certificate hash as the value for this key. If you do not know the certificate hash, use the Certificates MMC Snap-In and look at the certificate that should be renewed. Open the certificate properties and see the "Thumbprint" attribute of the certificate. Certificate renewal requires either a PKCS#7 or a CMC request format.

Note: You can only renew certificates that are time valid. Expired certificates cannot be renewed and must be replaced with a new certificate.

Requestername

Syntax	Requestername=samAccountName in Active Directory
Values	
Default value	n/a
Sample	`Requestername = "DOMAINNAME\username"`
Supported with CA type	Windows 2000 (with a PKCS#7 request type only) Windows Server 2003
Can be manipulated in a pending request	No

The requester name can be specified for certificate requests if the *RequestType* is set to PKCS7 or CMC. If the *RequestType* is set to PKCS10, this key will be ignored. The *Requestername* can only be set as part of the request. You cannot manipulate the *Requestername* in a pending request.

RequestType

Syntax	RequestType={string value}
Values	CMC PKCS10 PKCS10- PKCS7
Default value	PKCS10
Sample	`RequestType = CMC`
Supported with CA type	Windows 2000 (except CMC request type) Windows Server 2003
Can be manipulated in a pending request	No

The RequestType determines the standard that is used to generate and send the certificate request.

Silent

Syntax	Silent={Boolean}
Values	TRUE or FALSE
Default value	FALSE
Sample	`Silent = TRUE`
Supported with CA type	Windows 2000 Windows Server 2003

Can be manipulated in a pending request	No

By default, this option allows the CSP access to the interactive user desktop and request information such as a smart card PIN from the user. If this key is set to TRUE, the CSP must not interact with the desktop and will be blocked from displaying any user interface to the user.

SMIME

Syntax	SMIME = {Boolean}
Values	TRUE | FALSE
Default value	False (when KeySpec = AT_SIGNATURE)
Sample	`SMIME = TRUE`
Supported with CA type	Windows 2000 (if the policy module is configured in the registry to ignore the extension in the request) Windows Server 2003
Can be manipulated in a pending request	Yes

If this parameter is set to TRUE, an extension with the OID value 1.2.840.113549.1.9.15 is added to the request. The extension contains up to four OIDs, depending on the CSP capability, which refer to symmetric encryption algorithms that may be used by Secure Multipurpose Internet Mail Extensions (S/MIME) applications such as Outlook.

When an S/MIME encrypted e-mail is sent, the sender's computer does not know exactly what encryption algorithms are supported by the receiver. Thus, the weakest encryption algorithm was chosen by default to ensure a high level of interoperability. Today, the majority of computers support encryption algorithms that can handle stronger keys than 40-bit. To give the sender of an S/MIME message a hint as to which encryption algorithms might be supported by the receiver, identifiers of supported symmetric algorithms are inserted in the certificate. Alternatively, Outlook can be configured through the registry to always use the default encryption algorithm. For more information, see the Knowledge Base article at
http://support.microsoft.com/?id=307472

This functionality is supported with Outlook XP SP2 or a later version.

If the CA administrator has added the OID for S/MIME capabilities to the CA configuration parameter "*DisableExtensionList*", the CA will ignore the S/MIME capabilities in the request. For more information on how to add entries to the "*DisabledExtensionList*", see the Windows Server 2003 PKI Operations Guide at
http://www.microsoft.com/technet/prodtechnol/windowsserver2003/technologies/security/ws03pkog.mspx

Subject

Syntax	Subject={String}
Values	RDN string values
Default value	n/a
Sample	`Subject="CN=computer1.contoso.com"` `Subject="CN=John Smith,CN=Users,DC=Contoso,DC=com" x`
Supported with CA type	Windows 2000 Windows Server 2003
Can be manipulated in a pending request	Yes

Several applications rely on the subject information in a certificate. Thus, it is recommended that a value for this key be specified. If the subject is not set here, it is recommended that a subject name be included as part of the subject alternative name certificate extension.

UseExistingKeySet

Syntax	UseExistingKeySet={Boolean}
Values	TRUE or FALSE
Default value	FALSE
Sample	`UseExistingKeySet=TRUE`
Supported with CA type	Windows 2000 Windows Server 2003
Can be manipulated in a pending request	No

This parameter is used to specify that an existing key pair should be used in building a certificate request. If this key is set to TRUE, you must also specify a value for the *RenewalCert* key or the KeyContainer name. You must not set the *Exportable* key because you cannot change the properties of an existing key. In this case, no key material is generated when the certificate request is built.

UserProtected

Syntax	UserProtected={Boolean}
Values	TRUE or FALSE

Default value	FALSE
Sample	`UserProtected=TRUE`
Supported with CA type	Windows 2000 Windows Server 2003
Can be manipulated in a pending request	No

If this key is set to TRUE, a CryptoAPI password window is displayed when the key is generated during the certificate request build process. You can optionally protect the key with a password in the window or choose to display only a window when the key is used within an application. Once the key is protected with a password, you must enter this password every time the key is accessed. For more information on strong private key protection, see http://www.microsoft.com/technet/prodtechnol/windowsserver2003/technologies/security/autoenro.mspx

[EnhancedKeyUsageExtension]
The enhanced key usage extension determines for what purposes a certificate can be used. To see a list of OIDs that are supported by your CA, run the following command at a command-line prompt.

```
certutil –oid [OID string value]
```

Note: You can add custom OIDs to the CA configuration. Thus, the default list of OIDs that is supplied with a Windows Server 2003 CA can be extended according to your requirements.

For a list of object identifiers that are supported with the Microsoft PKI, see the Knowledge Base article "Object IDs Associated with Microsoft Cryptography" at http://support.microsoft.com/?id=287547

OID

Syntax	OID={OID_String}
Values	Any valid OID
Default value	n/a
Sample	`OID = 1.3.6.1.5.5.7.3.1`
Supported with CA type	Windows 2000 Windows Server 2003
Can be manipulated in a pending request	Yes

The OID is specified as a value that will be added to the extended key usage of the certificate. To specify multiple OIDs, add a new line for every OID required. For example:

```
OID = 1.3.6.1.5.5.7.3.1
OID = 1.3.6.1.5.5.7.3.2
```

[Extensions]
Instead of specifying extensions in this section, you can set extensions at the time you submit the certificate request to a Windows Server 2003 CA or a Windows 2000 CA if the request is initially made pending.

{OID}

Syntax	{OID} = {Base64Text}
Values	any valid OID that appears in the certificate
Default value	n/a

Sample	`2.5.29.17=MDmCFlcySy1CTy1EQy5jb25`
	`_continue_=Ob3NvMi5jb22gHwYJKwYBB`
	`_continue_=AGCNxkBoBIEENKgYHzc`
	`_continue_=F4dJmg+HRCPkkQ0=`

Supported with CA type	Windows 2000 Windows Server 2003
Can be manipulated in a pending request	Yes

Certificate extensions can be added to a request through this parameter. The value must be a Base64-encoded string. To split the string into several lines, use the _continue_ key word.

Critical

Syntax	Critical = {coma-separated OIDList}

Values	Any valid OID that appears in the certificate [Extensions] section
Default value	n/a
Sample value	`Critical = 2.5.29.17,2.5.29.15`
Can be manipulated in a pending request	Yes

This parameter carries the list of OIDs of certificate extensions where the extension is set to critical. A certificate extension can also be set to critical with the certutil –setextension command during the time a certificate request is pending. For more information, see the section "certutil -setextension".

[RequestAttributes]
All keys from this section can be specified as part of the certreq -submit -attrib command. For example, instead of specifying the CertificateTemplate in the INF instruction file, you could use the following command at a command-line prompt.

```
certreq -submit -attrib "CertificateTemplate:Domaincontroller" dcname.req
```

CertificateTemplate

Syntax	CertificateTemplate = {Templatename}
Values	Any valid certificate template name
Default value	n/a
Sample value	Critical = "DomainController"
Can be manipulated in a pending request	Yes

The value of this key instructs the CA as to which certificate template should be used in processing the certificate request. The name must be specified as the common name of the certificate template or the OID of the template, not the template display name. A list of valid certificate template common names can be determined with the following command.

```
certutil -templates
```

SAN

Syntax	SAN = "{tag}={value}&{tag}={value}]..."
Values	A list of valid subjectalternatename2 extension tags. *email, upn, dns, guid, url, ipaddress, oid* Alternatively, any valid OID can be specified.
Default value	n/a
Sample value	See below; should always include double quotation marks around value(s)
Can be manipulated in a pending request	Yes

The subject alternative name can be specified in various ways. A service principal name (SPN) is specified as a user principal name (UPN) value. The value is a string type by default but can have a prefix of a tag in section brackets "{}" that determines the data type of the value explicitly as exemplified in the following table.

SAN string	
`2.5.29.17={octet}MBmCF1cySzMtQk8tREMuY29udG9zbzIuY29t`	An ASN.1 string is inserted in the subject alternative name with one or several encoded extensions. The octet is a wrapper around a Base64 string encoded in binary format.
`1.2.3.4={utf8}SomeReadableText`	An ASN.1 string is inserted in the subject alternative name with one or several encoded extensions. The value is a string encoded with a UTF8 wrapper.
`2.5.29.17={asn}=3019821757...`	An ASN.1 string is inserted in the subject alternative name with one or several encoded extensions.
`email=John.Smith@contoso.com`	The user's e-mail address is inserted in the subject alternative name.
`dns="w2k3bodc.contoso.com"`	The DNS name is set to the computer's FQDN.
`dn="CN=W2K3BODC,OU=Domain Controllers,DC=contoso,DC=com"`	The distinguished name is set to the domain controllers DN.

SAN string	
url="http://www.contoso.com/default.html"	The URL is set to an http URL.
ipaddress=172.134.10.134	The IP is inserted in the subject alternative name.
oid=2.5.29.17	A given OID is inserted in the subject alternative name.
upn=John.Smith@contoso.com	The UPN name represents the user.
guid=f7c3ac41-b8ce-4fb4-aa58-3d1dc0e36b39	A GUID is inserted in the subject alternative name.

Multiple tags and values must be combined with an ampersand (&). For example, to set the UPN and the GUID, you would have to set the subject alternative name as follows at a command-line prompt.

```
-attrib "SAN:upn=John.Smith@contoso.com&guid=f7c3ac41-b8ce-4fb4-aa58-3d1dc0e36b39"
```

If you insert the subject alternative name in the INF file, the syntax would look like this.

```
SAN="upn=John.Smith@contoso.com&guid=f7c3ac41-b8ce-4fb4-aa58-3d1dc0e36b39"
```

-policy

For a detailed description and examples of this option, see the "Planning and Implementing Cross-Certification and Qualified Subordination Using Windows Server 2003" white paper at
http://www.microsoft.com/technet/prodtechnol/windowsserver2003/technologies/security/ws03qswp.mspx

-submit

This is the default option for certreq.exe. If no option is specified explicitly at the command-line prompt, certreq.exe attempts to submit a certificate request to a CA. You must specify a certificate request file when using the –submit option. If this parameter is omitted, a common File Open window is displayed where you can select the appropriate certificate request file. For more information on the available command-line parameters for this option, see Certreq –submit fails with error 0x80094800 or type certreq -submit -? from a command line.

Certificate attributes from the RequestAttributes section in the INF instruction file can be alternatively specified as a command-line parameter for Windows 2000 Server and Windows Server 2003. For a list of valid subject alternative tags that may be used with the –submit parameter, see **SAN**. It is important to separate attribute types with "\n". Use an ampersand (&) to separate each specific subject alternative name attribute value.

The following are samples of certreq.exe commands.

```
certreq -attrib "SAN:email=John.Smith@contoso.com" -submit myreq.req
certreq -submit -attrib "CertificateTemplate:Domaincontroller\nSAN:DNS=w2k3bodc.contoso.com" dcname.req dcname.cer dcname.p7b
certreq -submit -attrib "SAN:UPN=admin@contoso.com&email=foo@bar.com&guid=f7c3ac41-b8ce-4fb4-aa58-3d1dc0e36b39"
```

-retrieve

This option allows you to retrieve an issued certificate from the CA. Certificate owners or submitters can retrieve their certificate(s) from a Windows 2000 or 2003 CA. The certificate manager can retrieve all certificates from a Windows Server 2003 CA. This is not a security issue because certificates that are made available through the CA database contain only public information and only the original requestor may retrieve an issued certificate.

-accept

The final and most important step of the certificate enrollment process is to accept a certificate or response after it has been issued and retrieved from the CA. Certificates that are auto-enrolled or requested through the MMC Snap-In are accepted using the automation built into these applications. However, manual certificate requests, such as those submitted using certreq.exe, must be accepted and installed manually. The -accept parameter of certreq.exe links the previously generated private key with the issued certificate and removes the pending certificate request from the system where the certificate is requested. Accepting a CMC full response is required when archiving a private key on the CA.

-sign

Once the template administrator has configured the template to require one or more signatures on a certificate request, you must add these signatures to the request in order for the CA to process it correctly. The default Certificate Services Web enrollment pages can add only one signature to a certificate request, specifically when an Enrollment Agent certificate is used to request a smart card certificate on behalf of another user. If more than one signature is required, you must add these signatures with certreq.exe –sign from the command line to the raw certificate request, one at a time.

Top of page

Appendix 4: Certutil -setextension

This option allows an administrator to manipulate pending requests on a CA. In some cases, it is not possible to specify every possible certificate extension in the certificate request. For example, if a request for a user certificate is submitted to the CA with the Certificate Authority MMC Snap-In, additional information is not allowed in the enrollment process.

Unfortunately, to apply an extension to a pending certificate, you have to create the ASN.1-encoded extension manually. Once you have constructed the ASN.1 binary large object (BLOB) information you want, you can add the extension to the certificate request that is pending. Encoding data into ASN.1 is not straightforward, and

you have to understand the basic concept of this macro language used by certutil.exe. It is usually easier to find an existing request or certificate with the extensions you want, and then use certutil -v -dump to display the extensions as a hex-dump.

For more information on ASN.1 encoding, see Appendix 5: ASN.1 File Structure.

The -setextension option is similar to the *[Extensions]* section in the INF file used by certreq.exe. This implies that the critical flag on certificate extensions can be set for distinct OIDs and that specific value(s) can be set for these OIDs using the command-line options.

Note: With an enterprise CA, template settings will always override certificate extension conflicts that have been set manually. But this is only true for the extensions that are explicitly set through the template.

The syntax of the command line looks similar to the following example.

```
certutil.exe –setextension <RequestID> <ExtensionName> Flags {Long|Date|String|@InFile}
```

Option	Description
Request-ID	The ID of the pending request. It can be determined with the Certificate Authority MMC Snap-In.
ExtensionName	The OID of the extension that should be modified. If you are unsure what OID is appropriate, dump an existing certificate and find the OID from there.
Flags	Used to set a certificate extension to critical. flag = 0 means the extension is non-critical; flag = 1 means the extension is critical.
{Long | Date | String | @InFile}	Only the @InFile option applies. All other parameters should not be used. The @InFile can be Base64-encoded or a file in hexadecimal format. If the file contents could be interpreted as either Base64 or hexadecimal, hexadecimal is used. To force Base64 interpretation, use the "----- BEGIN -----" and "----- END -----" header. The content of the file depends on the extensions that should be manipulated.

The following examples illustrate how to use the certutil -setextension command to set the key usage of a pending certificate on the CA to critical and constrain the key usage to certificate and certificate revocation list (CRL) signing.

```
echo 03 02 01 06>ku.txt
certutil.exe -setextension <Request-ID> 2.5.29.15 1 @ku.txt
```

Top of page

Appendix 5: ASN.1 File Structure

Compiling an ASN.1 BLOB requires advanced knowledge about ASN.1 encoding. When developing your own applications, it is highly recommended that you use an encoding function such as CCertEncodeAltName to create the ASN.1 BLOB. See
http://msdn.microsoft.com/library/default.asp?url=/library/en-us/security/security/icertencodealtname.asp

Manually developing and parsing such ASN.1 data structure(s) is time-consuming and prone to error due to the overall complexity of the data encoding rules.

The following is a sample ASN.1 BLOB that was used to add a server's GUID and its DNS name to the subject alternative name extension of a certificate. The first line in both boxes is the actual ASN.1 BLOB; the lines following explain per column what the field in the BLOB actually represents. Note that the full ASN.1 BLOB is the following sequence.

30468223636B696E64657230312E6575726F70652E636F72702E6D6963726F736F66742E636F6D
A01F06092B0601040182371901A012041063303530353634346161313364326338

The following boxes explain in greater detail the breakdown of the ASN.1.

```
30468223636B696E64657230312E6575726F70652E636F72702E6D6963726F736F66742E636F6D
Sequence
  Total length of ASN.1 string
    ContextSpecificPrimitiveType
     Generalname see http://www.ietf.org/rfc/rfc2459.txt for a list of General
      Lenth of DNS name ----------------------------------------------------|
        DNS name ---------------------------------------------------------|
A01F06092B0601040182371901A012041063303530353634346161313364326338
ContextSpecificConstructedType
 Generalname see http://www.ietf.org/rfc/rfc2459.txt
  Length of othername ------------------------------------------|
    Other name OID-------|
                         ContextSpecificConstructedType
                          Generalname see http://www.ietf.org/rfc/rfc2459.txt
                           Length of datatype and othername
                             Other name datatype see
                             http://asn1.elibel.tm.fr/en/resources/tags.htm
                               Length of othername
                                  GUID --------------------------|
```

For more information about ASN.1 encoding, see the MSDN article at
http://msdn.microsoft.com/library/default.asp?url=/library/en-us/security/security/example_c_program_converting_names_from_certificates_to_asn1_and_back.asp

Top of page

Appendix 6: Encoding and Decoding with Hexadecimal, Binary, and Base64

In many cases, data that must be inserted in a certificate or certificate request must be converted from a human-readable format into a computer-readable format. The most important formats to understand when working with X.509 certificates are hexadecimal, binary, and Base64. Since it is time-consuming to perform data conversions manually, this capability is natively provided with certutil.exe.

As mentioned previously, certutil.exe comes in two versions. In the following table, V1 represents the Windows 2000 version and V2 represents the Windows Server 2003 version.

From	To Hexadecimal	To Binary	To Base64
Hexadecimal	n/a	V1: -decodehex V2: -decodehex	n/a
Binary	V1: n/a V2: -encodehex	n/a	V1: -encode V2: -encode
Base64	n/a	V1: -decode V2: -decode	n/a

The table illustrates that you cannot convert data directly from Base64 into hexidecimal, and vice versa. However, you can use binary as an intermediate format to perform this kind of conversion. The reqdccert.vbs script in Appendix 2: Sample Scripts leverages the conversion capabilities of certutil.exe to perform the work on behalf of the administrator, without requiring programming knowledge. For example, to convert an ASN.1 BLOB into a Base64 format (which is required for the certreq.exe INF instruction file), the script uses certutil.exe to automate the following commands.

```
certutil –decodehex <dcname>.asn <dcname>.bin
certutil –encode <dcname>.bin <dcname>.b64
```

Top of page

Summary

As the white paper thoroughly demonstrates, Windows Server 2003 provides a robust and flexible platform for supporting even the most complex or demanding deployment environments. Although the primary focus of the white paper is to demonstrate the procedures for supporting offline domain controller X.509 certificate enrollment in a disconnected or branch-office deployment, the procedures directly apply to all complex application environments. Administrators may apply the concepts presented to support any type of network and infrastructure services. These include directory servers (AD/AM), radius servers (IAS), web servers (IIS), and other stand-alone applications that require X.509 certificate enrollment or provisioning to provide or enable secure protocols, messaging, or application services.

Top of page

Related Links

See the following resources for further information:

- How To Configure Digital Certificates on Windows 2000 Domain Controllers for Secure LDAP and SMTP Replication at http://www.microsoft.com/technet/security/prodtech/windows2000/secmod154.mspx
- Requirements for Domain Controller Certificates from a Third-Party CA (Q291010) at http://support.microsoft.com/default.aspx?scid=kb;en-us;291010
- How to Enable LDAP over SSL with a Third-Party Certification Authority at http://support.microsoft.com/default.aspx?scid=kb;en-us;321051
- Certificate Auto-enrollment in Windows Server 2003 at http://www.microsoft.com/technet/prodtechnol/windowsserver2003/technologies/security/autoenro.mspx
- Best Practices for Implementing a Microsoft Windows Server 2003 Public Key Infrastructure at http://www.microsoft.com/technet/prodtechnol/windowsserver2003/technologies/security/ws3pkibp.mspx
- Planning and Implementing Cross-Certification and Qualified Subordination Using Windows Server 2003 at http://www.microsoft.com/technet/prodtechnol/windowsserver2003/technologies/security/ws03qswp.mspx
- Implementing and Administering Certificate Templates in Windows Server 2003 at http://www.microsoft.com/technet/prodtechnol/windowsserver2003/technologies/security/ws03crtm.mspx
- Key Archival and Management in Windows Server 2003 at http://www.microsoft.com/technet/prodtechnol/windowsserver2003/technologies/security/kyacws03.mspx
- Windows Server 2003 PKI Operations Guide at http://www.microsoft.com/technet/prodtechnol/windowsserver2003/technologies/security/ws03pkog.mspx
- Microsoft Systems Architecture (MSA) Enterprise Design for Certificate Services at http://www.microsoft.com/resources/documentation/msa/2/all/solution/en-us/msa20rak/vmhtm122.mspx

For the latest information about Windows Server 2003, see the Windows Server 2003 Web site at http://www.microsoft.com/windowsserver2003

Top of page