[MS-CERSOD]:
Certificate Services Overview Document

Intellectual Property Rights Notice for Open Specifications Documentation

- **Technical Documentation.** Microsoft publishes Open Specifications documentation for protocols, file formats, languages, standards as well as overviews of the interaction among each of these technologies.
- **Copyrights.** This documentation is covered by Microsoft copyrights. Regardless of any other terms that are contained in the terms of use for the Microsoft website that hosts this documentation, you may make copies of it in order to develop implementations of the technologies described in the Open Specifications and may distribute portions of it in your implementations using these technologies or your documentation as necessary to properly document the implementation. You may also distribute in your implementation, with or without modification, any schema, IDL's, or code samples that are included in the documentation. This permission also applies to any documents that are referenced in the Open Specifications.
- **No Trade Secrets.** Microsoft does not claim any trade secret rights in this documentation.
- **Patents.** Microsoft has patents that may cover your implementations of the technologies described in the Open Specifications. Neither this notice nor Microsoft's delivery of the documentation grants any licenses under those or any other Microsoft patents. However, a given Open Specification may be covered by Microsoft Open Specification Promise or the Community Promise. If you would prefer a written license, or if the technologies described in the Open Specifications are not covered by the Open Specifications Promise or Community Promise, as applicable, patent licenses are available by contacting iplg@microsoft.com.
- **Trademarks.** The names of companies and products contained in this documentation may be covered by trademarks or similar intellectual property rights. This notice does not grant any licenses under those rights.
- **Fictitious Names.** The example companies, organizations, products, domain names, e-mail addresses, logos, people, places, and events depicted in this documentation are fictitious. No association with any real company, organization, product, domain name, email address, logo, person, place, or event is intended or should be inferred.

Reservation of Rights. All other rights are reserved, and this notice does not grant any rights other than specifically described above, whether by implication, estoppel, or otherwise.

Tools. The Open Specifications do not require the use of Microsoft programming tools or programming environments in order for you to develop an implementation. If you have access to Microsoft programming tools and environments you are free to take advantage of them. Certain Open Specifications are intended for use in conjunction with publicly available standard specifications and network programming art, and assumes that the reader either is familiar with the aforementioned material or has immediate access to it.

This document provides an overview of the Certificate Services Overview Document Protocol Family. It is intended for use in conjunction with the Microsoft Protocol Technical Documents, publicly

available standard specifications, network programming art, and Microsoft Windows distributed systems concepts. It assumes that the reader is either familiar with the aforementioned material or has immediate access to it.

A Protocol Family System Document does not require the use of Microsoft programming tools or programming environments in order to implement the Protocols in the System. Developers who have access to Microsoft programming tools and environments are free to take advantage of them.

Abstract

The Certificate Services System consists of a set of protocols that are used for certificate enrollment, certificate policy, and remote administration of certificate services. It includes the certificate enrollment protocols: [MS-WCCE], [MS-ICPR], and [MS-WSTEP], certificate enrollment policy protocols: [MS-XCEP] and [MS-CRTD], and certificate remote administration protocol: [MS-CSRA]. The Certificate Services System operates in two modes: Standalone and Enterprise.

This document describes the intended functionality of the Certificate Services System and how the protocols in this system interact. It provides examples of some of the common user scenarios. It does not restate the processing rules and other details that are specific for each protocol. These details are described in the protocol specifications for each of the protocols and data structures that make up this system.

Revision Summary

Date	Revision History	Revision Class	Comments
06/17/2011	1.0	New	Released new document.

Contents

1 Introduction

The Certificate Services System provides a set of customizable services for issuing and managing certificates used in software security systems that are employing public key technologies.

Certificates are used to bind the identity of a person, device, or service to a corresponding private key.

The Certificate Services System provides the following technologies:

- Issuing certificates to requestors: Verifying the information in a certificate request, verifying the identity of the certificate requestor, and issuing certificates.
- Managing certificate lifetime and certificate renewal.
- Revoking certificates and verifying revocation status.

1.1 Conceptual Overview

A public key infrastructure (PKI) supports public key cryptography within and between organizations. A PKI consists of digital certificates, **key pairs**, a certificate authority (CA), and other registration authorities.

1.1.1 Public Key Cryptography

Public key cryptography allows one **entity** to prove its identity to another and exchange encrypted information without having to exchange private encryption keys. In this form of cryptography, an entity has a key pair consisting of a private key and a public key. The public key is freely exchanged with other parties. The public key can be used to encrypt information to be sent to the owner of the key pair, and the key-pair owner can use the private key to decrypt the information. The owner of the key pair can also use the private key to digitally sign documents; anyone else can use the public key to verify that the signature is authentic.

1.1.2 Certificates

A certificate is a digital statement issued by a CA that vouches for the identity of the certificate holder; a certificate binds a public key and a collection of **attributes** to the certificate holder. The certificate can be freely shared with other entities.

Certificates are electronic representations of users, computers, network devices, or services, issued by a Certificate Authority (CA), that are associated with a public and private key pair.

1.1.3 Certificate Authority

A Certificate Authority (CA) is an entity that issues digital certificates. A CA verifies the identity of a certificate requestor before a certificate can be issued. After validating the identity of a requestor, the CA issues the requested type of certificate. A CA also manages certificate revocation.

A CA issues certificates and confirms to other entities that the certificate is valid. People, computers, and applications, collectively described as end entities within this document, can all be issued certificates from the CA. Certificate holders can use the private key to digitally encrypt data, to digitally sign documents, and to identify themselves.

An entity requests a new certificate or a renewal of an existing certificate from the CA. Policy will normally define whether a CA automatically issues the certificate or queues the request for manual review by a **CA Administrator**. The CA typically requires authentication before processing the request. The CA can support different policies for each kind of certificate. For example, it might automatically issue certificates to be used for signing and encrypting email messages but only allow smart card authentication certificates to be issued by CA administrators who have visually verified the user's identity.

Policy-controlling certificate issuance can be restricted in two ways; the administrator decides to control certificate issuance either manually or automatically. Under the manual policy algorithm, the administrator would typically approve or deny each request in the queue. When certificate templates are used, the requestor is granted a certificate if the requestor has Enroll permissions on the corresponding template. The template can also specify additional constraints around the issuance of certificates.

While certificates are not required for normal **client** or server functionality in an out-of-the-box installation, there are a variety of systems or components that might utilize or rely on digital certificates for their operation, depending upon their configuration. In situations where other systems or components use certificates, there is no requirement that these certificates be provided through the implementation of the system specified in this document. In some cases, there might be other systems or components that will attempt to interact directly with this system, if available, in order to obtain certificates.

1.1.4 Certificate Revocation Lists

End entities will normally evaluate certificates for validity when making trust decisions and will no longer trust the certificate if it is presented after the expiration date. In order to invalidate a previously issued certificate, prior to its expiration, an administrator can revoke it. This might be desired, for example, when an employee leaves the organization or when the private key has been compromised. The CA maintains a list of revoked certificates that it makes available publicly at a location specified in all of the certificates it issues. This list is known as the certificate revocation list (CRL). Entities that are required to verify the validity of a certificate can download the CRL and determine if the certificate is in it.

1.1.5 Basic Certificate Enrollment

The **certificate enrollment** is the process by which an **end entity** obtains the certificate from the certificate issuer. The following diagram depicts the basic certificate enrollment process.

Figure 1: Basic certificate enrollment

The individual steps are described herein.

1. The enrollment client generates a certificate request. The certificate request contains the public key of the key pair, along with any other information required by the certificate template or configured by the user. The certificate request is signed by the private key of the key pair and is sent by the enrollment client to the certificate issuer.

2. The certificate issuer validates the certificate request and if the request is valid, then issues the requested certificate to the user; otherwise, it denies the request, or causes the request to be pending until a certificate manager manually approves or denies it.

1.2 Glossary

The following terms are defined in [MS-GLOS]:

Active Directory
Active Directory domain
attribute
certificate
certificate authority (CA)
certificate revocation list (CRL)
certificate template
client
digital certificate
digital signature
directory
Distributed Component Object Model (DCOM)
domain controller (DC)
encryption
enhanced key usage (EKU)
enroll/enrollment
end entity (EE)
enterprise certificate authority
exchange certificate
Group Policy
interface
key
key archival
key exchange
keyholder
key recovery agent (KRA)
Lightweight Directory Access Protocol (LDAP)
object
object identifier (OID)
private key
public key
public key infrastructure (PKI)
registration authority (RA)
root certificate
remote procedure call (RPC)
revocation
root CA
schema
standalone CA
trust

The following terms are defined in [MS-WCCE]:

CA exit algorithm
CA policy algorithm
Cryptographic Message Syntax (CMS)
Enroll On Behalf Of (EOBO)

issuance
key length
key pair

The following terms are defined in [MS-XCEP]:

certificate enrollment policy

The following terms are specific to this document:

CA Administrator: A human operator who is responsible for managing the **CA System**.

CA certificates: CA certificates are **certificates** that are issued by one CA to another CA. These CA certificates become a part of the certificate trust hierarchy (the certificate path from the **end entity** certificates to the trusted root CA certificate).

CA System: The system that implements the protocols and data structures specified in [MS-WCCE], [MS-CSRA], [MS-CRTD], and [MS-ICPR].

CEP: The **certificate enrollment policy** as defined in [MS-XCEP].

certificate enrollment: The process of acquiring a **digital certificate** from a certificate authority. This certificate and its associated **private key** establish a trusted identity for an **entity** that is using the **public key**-based services and applications.

Component: The principal computational elements and data stores that execute in a system.

Entity: A unit that is part of the system such as a component or an element.

LDAP: In this document, the term LDAP always refers to the Lightweight Directory Access Protocol (LDAP) profile specified in [MS-ADTS] section 3.1.1.3.

PKI Administrators: PKI Administrators are responsible for implementing the company's policy by defining CEPs and setting up servers that provide certificates to clients.

policy server endpoint: A collection of information about a policy server, such as the protocol it supports, its Uniform Resource Identifier (URI), and authentication to be used when accessing the server.

The following protocol abbreviations are used in this document:

CRTD: Certificate Templates Structure Specification

DCOM: Distributed Component Object Model (DCOM) Remote Protocol Specification

GPREG: Group Policy: Registry Extension Encoding

HTTP: Hypertext Transfer Protocol

WCCE: Windows Client Certificate Enrollment Protocol Specification

WSTEP: WS-Trust Enrollment Extensions

XCEP: X.509 Certificate Enrollment Policy Protocol Specification

CSRA: Certificate Services Remote Administration Protocol Specification

1.3 References

References to Microsoft Open Specification documents do not include a publishing year because links are to the latest version of the documents, which are updated frequently. References to other documents include a publishing year when one is available.

[HOWARD] Howard, M., "Writing Secure Code", Microsoft Press, 2002, ISBN: 0735617228.

[MS-ADSOD] Microsoft Corporation, "Active Directory System Overview", to be published 2011.

[MS-ADTS] Microsoft Corporation, "Active Directory Technical Specification".

[MS-AUTHSOD] Microsoft Corporation, "Authentication Services Subsystem Overview Document".

[MS-CRTD] Microsoft Corporation, "Certificate Templates Structure".

[MS-CSRA] Microsoft Corporation, "Certificate Services Remote Administration Protocol Specification".

[MS-DCOM] Microsoft Corporation, "Distributed Component Object Model (DCOM) Remote Protocol Specification".

[MS-DRSR] Microsoft Corporation, "Directory Replication Service (DRS) Remote Protocol Specification".

[MS-GLOS] Microsoft Corporation, "Windows Protocols Master Glossary".

[MS-GPSOD] Microsoft Corporation, "Group Policy System Overview", to be published 2011.

[MS-GPREG] Microsoft Corporation, "Group Policy: Registry Extension Encoding".

[MS-ICPR] Microsoft Corporation, "ICertPassage Remote Protocol Specification".

[MS-WCCE] Microsoft Corporation, "Windows Client Certificate Enrollment Protocol Specification".

[MS-WSTEP] Microsoft Corporation, "WS-Trust X.509v3 Token Enrollment Extensions".

[MS-XCEP] Microsoft Corporation, "X.509 Certificate Enrollment Policy Protocol Specification".

[MSFT-ARCHIVE] Microsoft Corporation, "Key Archival and Management in Windows Server 2003", December 2004, http://technet2.microsoft.com/WindowsServer/en/Library/296f87df-06c3-4e27-89ff-5283cb76fb811033.mspx

[X509] ITU-T, "Information Technology - Open Systems Interconnection - The Directory: Public-Key and Attribute Certificate Frameworks", Recommendation X.509, August 2005, http://www.itu.int/rec/T-REC-X.509/en

Note There is a charge to download the specification.

[RFC5280] Cooper, D., Santesson, S., Farrell, S., et al., "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile", RFC 5280, May 2008, http://www.ietf.org/rfc/rfc5280.txt

2 Functional Architecture

2.1 Overview

The primary components of the Certificate Services System architecture in Enterprise mode is as follows:

Certificate Authority (CA) server: The **CA** can operate in one of two modes, as a Standalone CA or as an Enterprise CA.

The following are additional requirements when the CA operates in Enterprise CA mode:

- The CA server is a member of the domain.
- The CA server uses Active Directory service to store the policy, authentication, and other related information that required.
- Optionally, the CA Server depends on the Group Policy service as the configuration store for the **Policy Server endpoints**.

Enrollment Clients: Clients can enroll the certificates by using one of the two methods: Direct enrollment and WSTEP enrollment. The enrollment clients can be different types, see section 2.1.2 for more details.

Policy Server: Enrollment clients contact the policy server to get the policy information consisting of the types of certificates it can enroll for, which enrollment servers to contact to enroll for them, and what type of authentication to use for each service. The policy server can be an XCEP server or domain controller; the Direct enrollment clients always use the domain controller as the Policy Server; WSTEP enrollment clients use the XCEP server as Policy Server.

The clients must first be configured with information about which policy server(s) to contact and how to authenticate to them; this information can be configured through either Group Policy or local configuration.

XCEP Server: Hosts the enrollment policy web services and allows the enrollment clients to retrieve the certificate enrollment policies (**CEP**) by using the XCEP protocol.

WSTEP Server: Hosts the enrollments' Web services and allows the enrollment clients to enroll the certificates by using the WSTEP protocol.

CA Admin Clients: Using CA Admin clients, the Administrators performs the CA administration remotely.

Domain Controller: Enrollment clients and CA servers in Enterprise mode primarily depend on the Active Directory and optionally on Group Policy server as described above.

The following figure depicts the Certificate Services System Functional architecture in Enterprise mode and the protocols involved to communicate both with and within the system, and other systems. The classification and purpose of the member protocols are described in section 2.2.

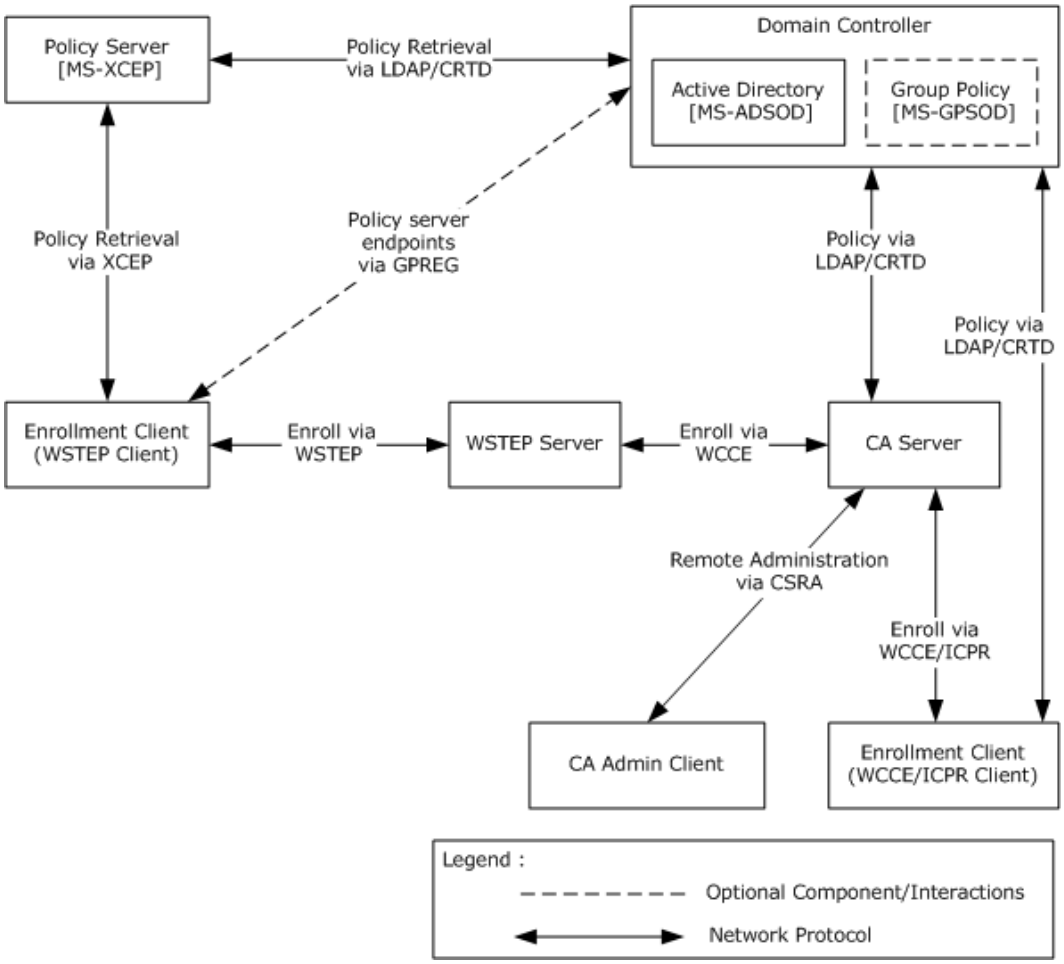

Figure 2: Certificate Services System functional architecture in Enterprise mode

2.1.1 System Purpose

The purpose of the Certificate Services System is to issue and manage certificates. The Certificate Services System includes the protocols used for submitting **certificate** requests to the CA, for CA server side processing of these requests and for remote administration of the CA. The certificates themselves do not generally contain sensitive information and are often publicly available. The certificates can be used for different purposes and are typically stored in a variety of methods and locations. Certificates have a certain lifetime and will eventually face expiration. Certificate autoenrollment automates certificate enrollment and renewal for computer certificates.

2.1.2 System Components

The Certificate Services System has two main components:

- Certificate Authority
- Enrollment Client

The following sections provide an overview of these components.

2.1.2.1 Certificate Authority

The Certificate Authority is the core component of the Certificate Services System. The CA implements the Windows Client Certificate Enrollment Protocol (WCCE), the ICertPassage Remote Protocol (ICPR), and the Certificate Services Remote Administration Protocol (CSRA) to enable certificate enrollment and CA administration. It also makes use of policy and exit algorithms to facilitate the more complex enrollment processes.

This section describes the interfaces and modes of the CA.

2.1.2.1.1 Certificate Authority Interfaces

The following diagram shows the components that interact with the Certificate Authority.

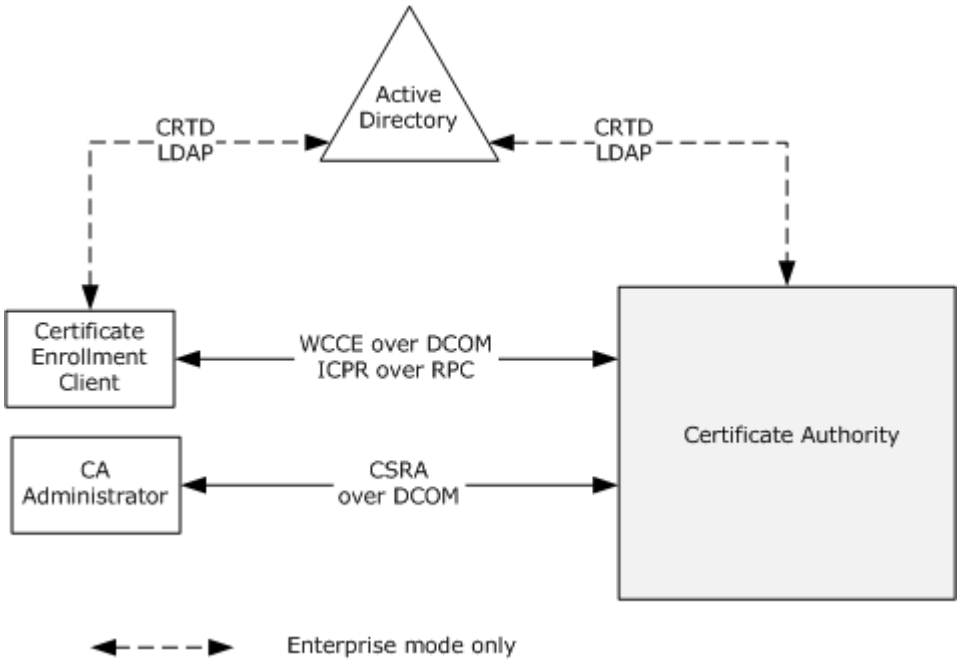

Figure 3: Certificate Authority Interfaces

End users use their computers to obtain new certificates, to renew existing certificates, and to obtain information about the CA. Administrators use their client computers to connect to external interfaces to manage the CA remotely.

2.1.2.1.2 Certificate Authority (CA) Modes

The CA consists of two distinct groups. One group of components is responsible for the certificate enrollment and the other for **CA system** administration. The two groups communicate through shared data and the interaction between them is defined in [MS-WCCE] and [MS-CSRA].

The protocols and their interaction with shared data are shown in the following diagrams.

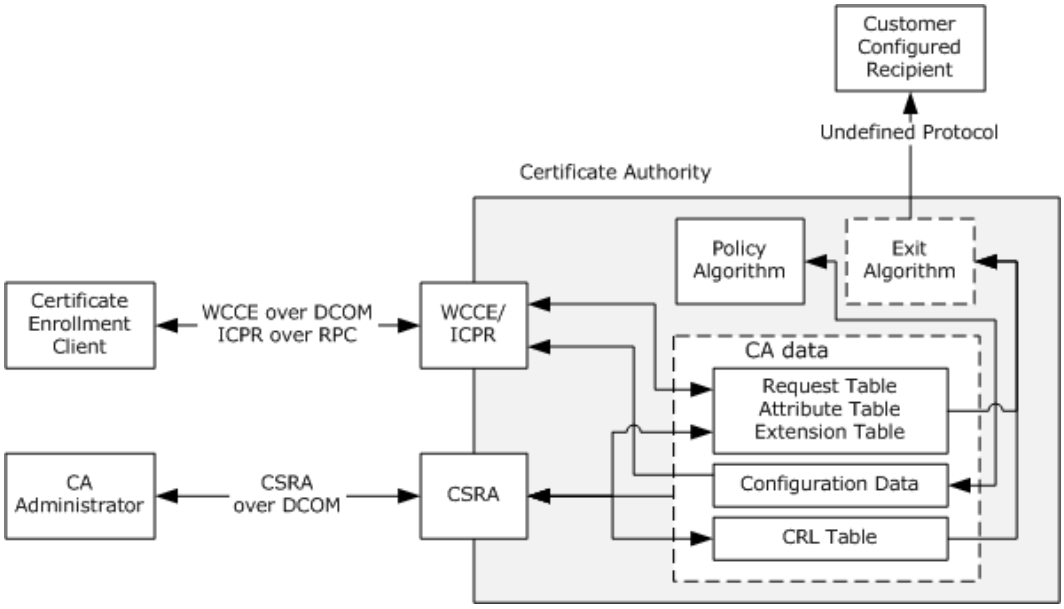

Figure 4: Certificate Authority (CA) in standalone mode

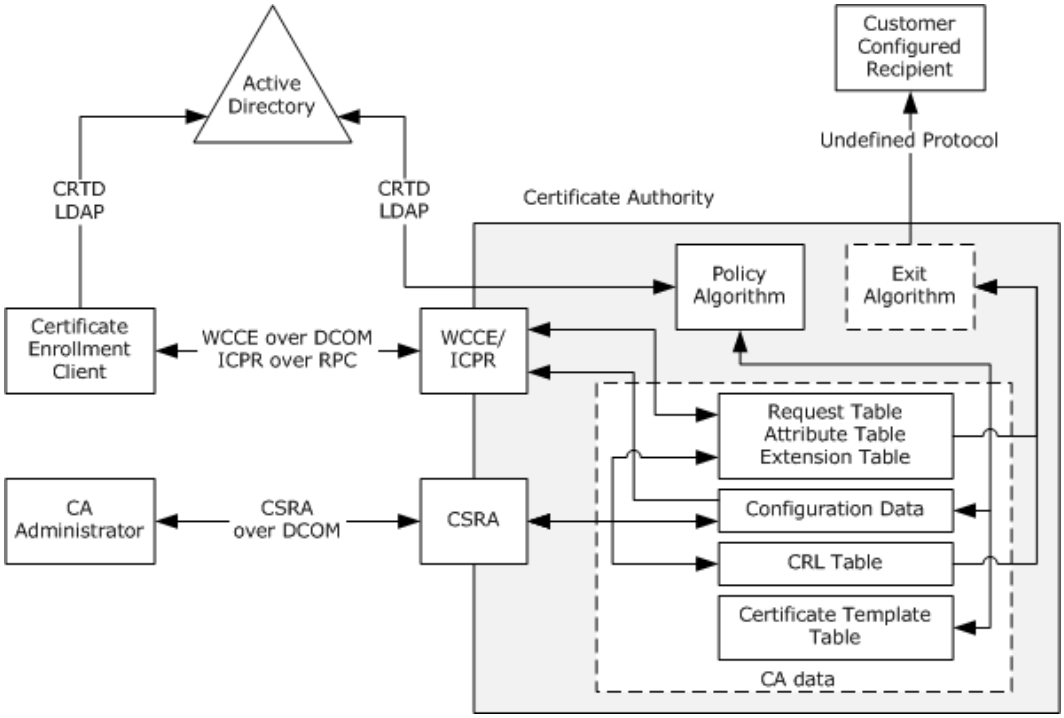

Figure 5: Certificate Authority (CA) in enterprise mode

Policy Algorithm

The CA policy algorithm is a required component of the system. Requests for new and renewed certificates are subject to the policy algorithm. It determines whether a certificate request is to be fulfilled, denied, or set to pending administrator approval. For example, a system implementing the

enterprise CA functionality that is specified in [MS-WCCE] section 3.2.2 must verify that the requester has Enroll permission on the requested certificate template. The policy algorithm and rules for its implementation are defined in [MS-WCCE] sections 3.2.1.4.2.1.4.4 and 3.2.2.6.2.1.4.

Exit Algorithm

The CA exit algorithm is an optional internal component responsible for request post-processing, which may include communicating via another protocol. For example, the CA could send email notifications to the end entity and system administrator when a new certificate is generated. The exit algorithm and rules for its implementation are defined in [MS-WCCE] section 3.2.1.4.2.1.4.8.

CA Data Storage

The method used for data storage is independent of the protocols and interfaces described in this document. The implementer can use a general purpose database, files stored in the operating system's native file system, or whatever is preferred. The data that must be stored is shown in the preceding figures.

The following tables constitute the CA data.

Request Element Tables

Request Table: Described in [MS-WCCE] section 3.2.1.1.1 and [MS-CSRA] section 3.1.1.1.

Attribute Table: Described in [MS-CSRA] section 3.1.1.2.

Extension Table: Described in [MS-CSRA] section 3.1.1.3.

The preceding tables are all related to certificate request processing and their purpose is to hold the history and data associated with all requests processed by the CA. All the tables are commonly maintained by the CA and their data is persistent across all states of the CA, including system and server restarts, as well as backup and recoveries.

Configuration Data

Configuration Data: Described in [MS-CSRA] sections 3.1.1.6, 3.1.1.7, 3.1.1.8, 3.1.1.9, and 3.1.1.10, and in [MS-WCCE] section 3.2.1.1.4.

The purpose of the configuration data elements is to provide information about the operational behavior of the CA. This data is persistent across all states of the CA. It is most commonly accessed by CA Administrators using CSRA, but can also be read by enrollment clients using WCCE.

Server implementations of the Certificate Services Remote Administration Protocol, the Windows Client Certificate Enrollment Protocol, and the ICertPassage Remote Protocol in combination use the same list of configuration data elements across the implementations.

CRL Table

Certificate Revocation List (CRL) Table: Described in [MS-CSRA] section 3.1.1.4 and [MS-WCCE] section 3.2.1.1.3.

The purpose of this table is to store certificate revocation information. This data is persistent across all states of the CA. It is typically accessed and manipulated by CA Administrators via the Certificate Services Remote Administration Protocol.

Certificate Template Table

Certificate Template Replica Table: Described in [MS-WCCE] section 3.2.2.3.1.

The CA maintains a replica of all certificate template data stored in Active Directory. This data is accessed by the policy algorithm and is used in processing enrollment requests. The data is persistent across all states of the CA.

2.1.2.2 Enrollment Client

There are a variety of enrollment client types and their behavior regarding the handling of the certificate requests and the resulting issued certificate, can be different.

The common enrollment client types include:

- Autoenrollment
- User Enrollment Tools
- Registration Authority Application
- Direct Enrollment Applications

Autoenrollment

Autoenrollment is normally performed without user input. In some cases, user input during the enrollment process might be required, such as in the case of smart card usage and PIN input. However, the enrollment process itself is not triggered by the user. Issued certificates are received from the CA and are then stored within a local certificate store on the client system.

User Enrollment Tools

User Enrollment Tools will typically work in a similar fashion, but are user-initiated and can involve further user input during the enrollment process.

Registration Authority Application

Registration Authority (RA) Applications are typically used in situations where a higher level of assurance is required for the certificate enrollment process. Often, their use involves an enrollment agent. Most RA applications are used in order to process or submit certificate requests for other users and therefore do not keep or install issued certificates for their own use. Instead, the issued certificates are transferred to the end entity in some fashion.

Direct Enrollment Applications

Direct Enrollment Applications are those that might be written by third parties to interact directly with the CA for certificate enrollment. How these applications handle the certificates after they have been issued is completely dependent upon the design and development of the application.

In all cases, the involvement of the CA system ends with the CA providing the issued certificate to the Enrollment client. How the client handles the certificate after that point is independent of the CA.

2.1.2.2.1 Certificate Enrollment Methods

There are two methods for certificate enrollment: **DCOM**-based certificate enrollment (Direct enrollment) and Web services-based certificate enrollment (WSTEP enrollment).

DCOM-based Certificate Enrollment

DCOM-based certificate enrollment uses WCCE for certificate requests. When a CA is operating in Enterprise CA mode, it uses the **LDAP** profile specified in [MS-ADTS] section 3.1.1.3 to obtain a CEP

from a **domain controller (DC)**. The CEP is expressed via certificate templates that are data structures specified in [MS-CRTD] and Certificate Authority (CA) information.

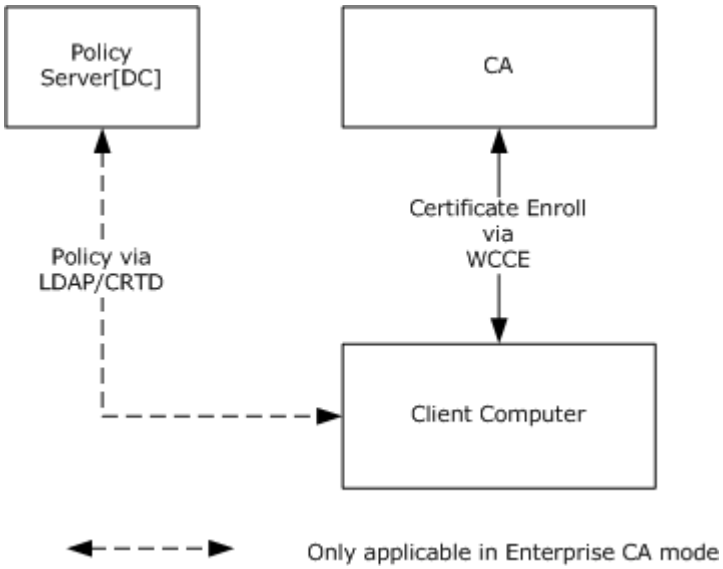

Figure 6: DCOM-based certificate enrollment

A client computer starts by discovering a policy server. In case of DCOM-based enrollment, the policy server is always a domain controller, discovered as specified in [MS-ADTS] section 7.3.

Web Services-Based Certificate Enrollment

Web services-based certificate enrollment (as shown in the following figure) uses the WSTEP protocol for certificate requests. It uses XCEP to retrieve the CEP.

For the use of XCEP/WSTEP, the Web service address has to be configured out-of-band (for example, manually or by Group Policy).

Certificate enrollment clients can use Group Policy, specifically the GPREG protocol, to obtain policy server endpoints that were configured by the administrator in the enterprise environment. Clients can also use a local configuration store that contains policy server endpoints specific to a particular client. The following figure illustrates this concept.

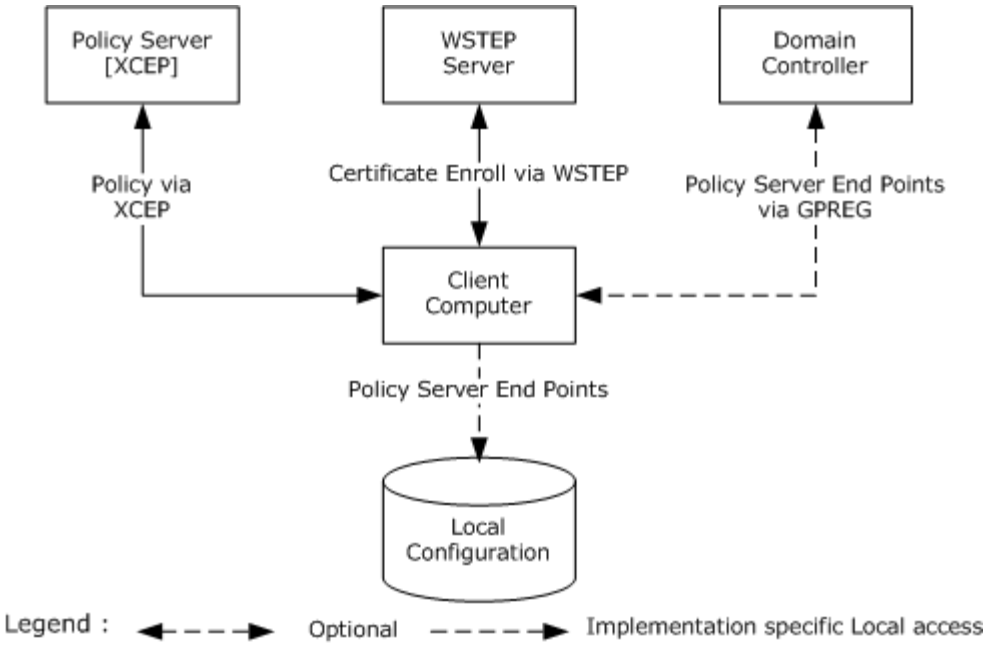

Figure 7: Web services-based Certificate enrollment

Based on an organization's security policies, it is possible for the client to use both methods to enroll for certificates. The following diagram shows an example of one such possible deployment.

In this case, the client computer is a member of a domain where a **PKI administrator** has configured a CEP by defining some templates and installing an enterprise CA, XCEP server, and WSTEP server. The client computer discovers available CEP servers through Group Policy. Also, the administrator of the client computer itself needs to obtain a certificate for this computer from a third party so the computer can be configured with the policy server endpoint of the third party server. The client computer can now request certificates based on both policies.

Considering that any client can be configured to work with multiple CEPs that have multiple policy server endpoints, can define multiple certificate templates, and are used by multiple issuers, it is clear that enrolling for certificates manually can be a difficult task. The job of autoenrollment is to traverse all of the CEPs and enroll them for certificates as required.

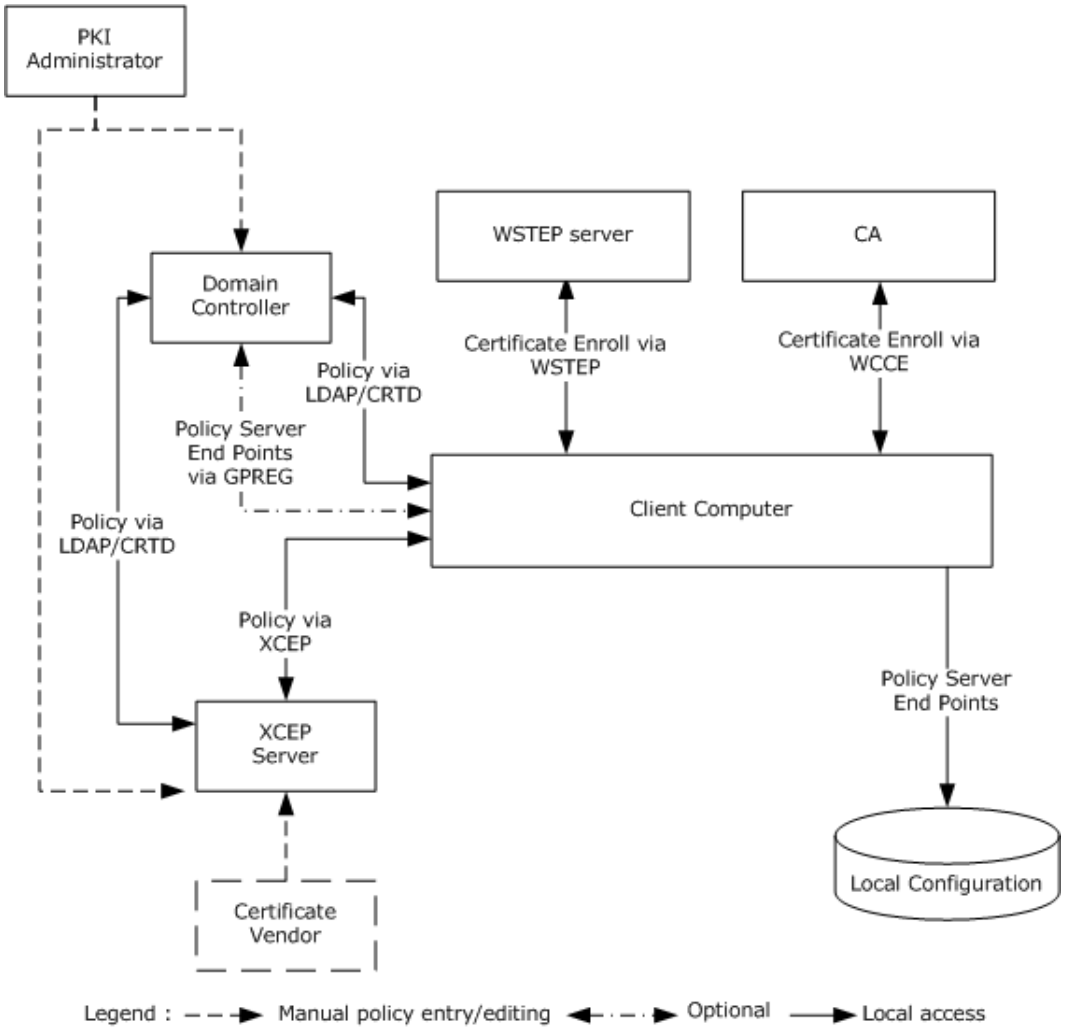

Figure 8: Deployment of Certificate Enrollment

2.1.2.2.2 Autoenrollment in a Domain Environment

This section describes the system components of the autoenrollment client that is joined to a domain and how the external entities to the certificate enrollment services system influence the behavior of the autoenrollment process.

As shown in the following diagram, the autoenrollment process access two local data stores (certificate/key storage and local configuration) and communicates with the XCEP server, WSTEP server, CA system, and domain controller. The autoenrollment process examines local certificate storage and renews an already issued certificate or enrolls for new certificates as required, based on a pre-defined policy, encoded in the form of CEPs.

In the case of DCOM-based certificate enrollment, the autoenrollment process gets the certificate templates and CA information from the domain controller, whereas in the case of the Web services certificate enrollment, the Group Policy client on the Enrollment client computer gets the Policy Server endpoints information from the domain controller through the Group Policy: Registry Extension Encoding [MS-GPREG] and updates the local configuration; then the autoenrollment

process fetches the Policy Server endpoint URL information in an implemention-specific way and connects to the XCEP server to download the CEP. Depending on the available CEP and certificates currently present on the system local certificate/key storage, autoenrollment submits requests and persists newly enrolled or renewed certificates in the local certificate storage. In the case of DCOM-based certificate enrollment, autoenrollment submits requests to the CA, whereas in the case of Web services certificate enrollment, it submits the requests to the WSTEP server.

The local certificate/key storage can be read or modified by other systems in an implementation-specific way, but the autoenrollment process makes no assumptions about how or even if this happens. Local configuration is modified by the computer administrator through the use of an administration tool such as a Group Policy client.

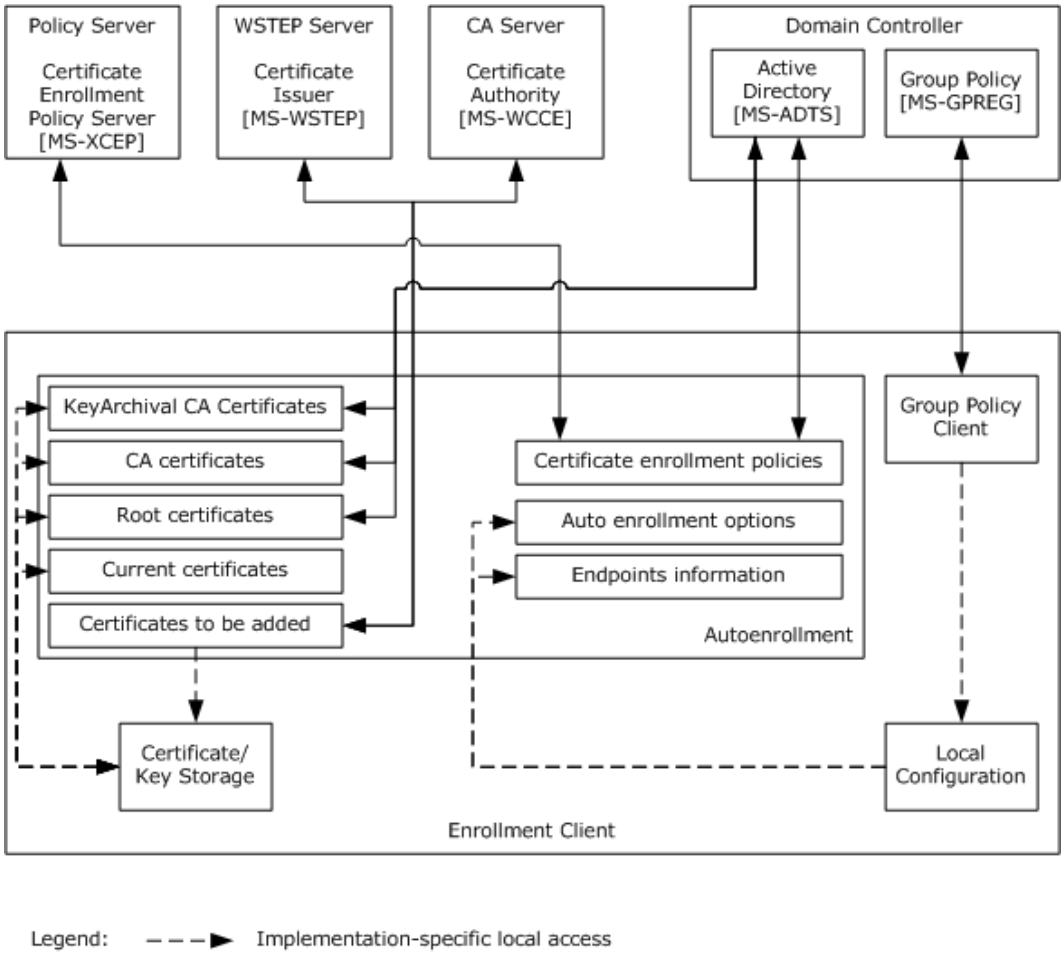

Figure 9: The autoenrollment process executing on a computer that is joined to a domain

Certificate Storage: This entity provides some implementation-specific persisted local certificate storage that can be logically organized into groups of certificates.

Key Storage: This entity provides some implementation-specific persisted local private key storage where it could store private keys associated with the certificates it is requesting.

Local configuration: This entity provides the configuration options and policy server endpoint information.

As depicted in the preceding figure, Autoenrollment uses the Certificate's data which contains lists of certificates. The Certificate's data defines these lists:

Key Archival CA certificates: A list of **CA certificates** in local certificate storage that are allowed to perform key archival. These certificates are used to validate the CA exchange certificates.

CA certificates: A list of intermediate certificate authority (CA) certificates in local certificate storage that are used to validate end entity certificates.

Root certificates: A list of the root certificates in local certificate storage that are used to validate end entity certificates.

Current certificates: A list of the current end entity certificates.

Certificates to be added: A list of certificates that are to be added to the Local certificate storage.

Certificate enrollment policies: The Certificate enrollment policies provide the information about certificate templates and issuers. The autoenrollment client has a list of zero or more instances of the Certificate enrollment policy.

Autoenrollment options: Autoenrollment options specify options that control autoenrollment behavior.

Endpoints information: The EndPoints information contains information about the policy server endpoints that are used to obtain information about CEPs. Endpoints information can be initialized either through Group Policy or local configuration information to facilitate communication with CEP servers.

2.1.3 Applicability

The Certificate Services System is applicable to an environment in which clients benefit from the capability to interact with the CA in order to enroll or manage X509 certificates. In particular, the autoenrollment client is applicable in envrionments where the workload of provising certificate is large enough to warrant automation.

2.1.4 Relevant Standards

Relevant standards are the Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile, as specified in [RFC5280]. This specification is one part of a family of standards for the X.509 Public Key Infrastructure (PKI) for the Internet. This specification profiles the format and semantics of certificates and **certificate revocation lists (CRLs)** for the Internet **public key infrastructure (PKI)**.

2.2 Protocol Summary

The table below provides a comprehensive list of the Member Protocols of the Certificate Services System.

Protocol name	Description	Short name
Windows Client Certificate Enrollment Protocol	A DCOM-based protocol. Responsible for certificate enrollment, it enables clients to request various services from a CA, such as certificate enrollment and property retrieval.	[MS-WCCE]
Certificate Services	Responsible for CA administration, enables administrative tools to	[MS-

Protocol name	Description	Short name
Remote Administration Protocol	configure the state and policy of a CA on a server.	CSRA]
ICertPassage Remote Protocol	This protocol is a subset of the WCCE Protocol used for certificate enrollment over RPC by clients that do not support DCOM.	[MS-ICPR]
Certificate Templates Structure	Certificate templates are stored in **Active Directory** and are used when the CA operates as an enterprise CA. They contain details about requesting and issuing certificates. Policy algorithms on the CA use certificate templates to determine how to respond to certificate requests. [MS-CRTD] defines attributes that are accessed by using the Lightweight Directory Access Protocol (LDAP).	[MS-CRTD]
X.509 Certificate Enrollment Policy Protocol	A SOAP-based Protocol that enables client to retrieve enrollment policies.	[MS-XCEP]
WS-Trust Enrollment Extensions	A SOAP-based Protocol that provides web services-based certificate enrollment, renewal and pending certificate retrieval. The WS-Trust profile enables X.509 certificate enrollment.	[MS-WSTEP]

The Member Protocols are grouped according to their primary purpose.

Certificate Enrollment Protocols:

Protocols in the following table enable certificate enrollment.

Protocol name	Description	Short name
Windows Client Certificate Enrollment Protocol	A DCOM-based protocol. Responsible for certificate enrollment, it enables clients to request various services from a CA, such as certificate enrollment and property retrieval.	[MS-WCCE]
ICertPassage Remote Protocol	This protocol is a subset of the WCCE Protocol used for certificate enrollment over RPC by clients that do not support DCOM.	[MS-ICPR]
WS-Trust Enrollment Extensions	A SOAP-based protocol that provides Web services-based certificate enrollment, renewal, and pending certificate retrieval. The WS-Trust profile enables X.509 certificate enrollment.	[MS-WSTEP]

Certificate Services Administration Protocols:

Protocols in the following table enable remote administration of the certificate services.

Protocol name	Description	Short name
Certificate Services Remote Administration Protocol	Responsible for CA administration, it enables administrative tools to configure the state and policy of a CA on a server.	[MS-CSRA]

Certificate Enrollment Policy (CEP) Protocols:

Protocols in the following table enable **certificate enrollment policy**.

Protocol name	Description	Short name
X.509 Certificate Enrollment Policy Protocol	A SOAP-based Protocol that enables the client to retrieve enrollment policies.	[MS-XCEP]
Certificate Templates Structure	Certificate templates are stored in Active Directory and are used when the CA operates as an enterprise CA. They contain details about requesting and issuing certificates. Policy algorithms on the CA use certificate templates to determine how to respond to certificate requests. [MS-CRTD] defines attributes that are accessed by using the Lightweight Directory Access Protocol (LDAP).	[MS-CRTD]

2.3 Environment

The following sections identify the context in which the system exists. This includes the systems that use the interfaces provided by this system of protocols, other systems that depend on this system, and, as appropriate, how components of the system communicate.

2.3.1 Dependencies on This System

None.

2.3.2 Dependencies on Other Systems/Components

This System depends on the following systems and components in the CA Enterprise mode:

- The Active Directory for the storage and retrieval of certificate templates [MS-ADSOD].
- The Group Policy server for the policy server endpoints information through the GPREG protocol [MS-GPSOD].

2.4 Assumptions and Preconditions

The Certificate Services System has the following assumptions, regardless of the mode:

- The transport protocols, RPC, DCOM and SOAP, are available if the CA is to be accessed over a network.
- The authentication protocols NTLM and SSL/TLS are available for authentication.

These additional assumptions apply when running in Enterprise CA mode:

- Active Directory is available for the storage and retrieval of certificate templates.
- The Kerberos authentication protocol is available for the authentication and message security.

An autoenrollment client makes these assumptions about the policies it consumes:

- When the autoenrollment client processes CEP endpoint information via Group Policy, it assumes that:
  - The Group Policy data that initializes **EndPoint** flags that describes the options for CEP does not differ between separate Uniform Resource Identifiers (URIs) that belong to the same

policy. The group policy data is configured using the Group Policy Administrative tool as specified in [MS-GPSOD].

- The Group Policy data does not contain instructions that identify more than one CEP as the default. The group policy data is configured by using the Group Policy Administrative tool as specified in [MS-GPSOD].

- Autoenrollment assumes that no single CEP defines more than one template that has the same CA CommonName.

2.5 Use Cases

2.5.1 Actors

The actors that participate in certificate services are:

End Entity: An End Entity is a keyholder (person or computer) to whose key or name a particular certificate refers.

Enrollment Agent: An entity that submits requests on behalf of another End Entity. An Enrollment Agent is typically authorized by the CA to enroll for certificates that End Entities themselves might not be able to. The policy enforcement for those certified End Entities is thus assumed to be done by the Enrollment Agent.

CA Administrator: A person who is responsible for management of the CA System, such as system configuration, and managing pending requests for certificates.

2.5.2 Use Case Summary Diagrams

There are two main use cases for the CA System:

- Enroll for a Certificate

- Administer the CA

The Enroll for a Certificate use case is the most important use case for this system. In its simplest form, it allows a caller (either an End Entity or Enrollment Agent) to request a certificate from a CA (see the examples in sections 3.1 and 3.2). Upon successful completion of the use case, the End Entity receives a certificate signed by the CA.

Common variations of the certificate enrollment use case are as follows:

- Certificate renewal is when an End Entity already possesses a valid certificate and uses the private key associated with that certificate to sign a renewal request for a new certificate of the same type.

- Enrollment on behalf of another user introduces an Enrollment Agent who acts as a cosigner for the certificate request to provide a higher level of control in the enrollment process.

- Autoenrollment removes the burden on the server administrator to enroll and renew the certificates by automatically certificate enrollment and renew the certificates.

- Certificate enrollment with CA Administrator approval interrupts the automatic flow of the certificate enrollment to allow the administrator to modify the request itself, modify the resulting certificate, or approve or deny the request.

The Administer the CA use cases include generic functions such as editing the CA configuration, as well as more specific functions such as revoking certificates or recovering escrowed private keys from a CA.

The primary CA administration use cases are:

- Edit CA configuration settings
- Revoke a certificate
- Recover an archived certificate and key

2.5.3 Use Case Descriptions

2.5.3.1 Enroll for a Certificate

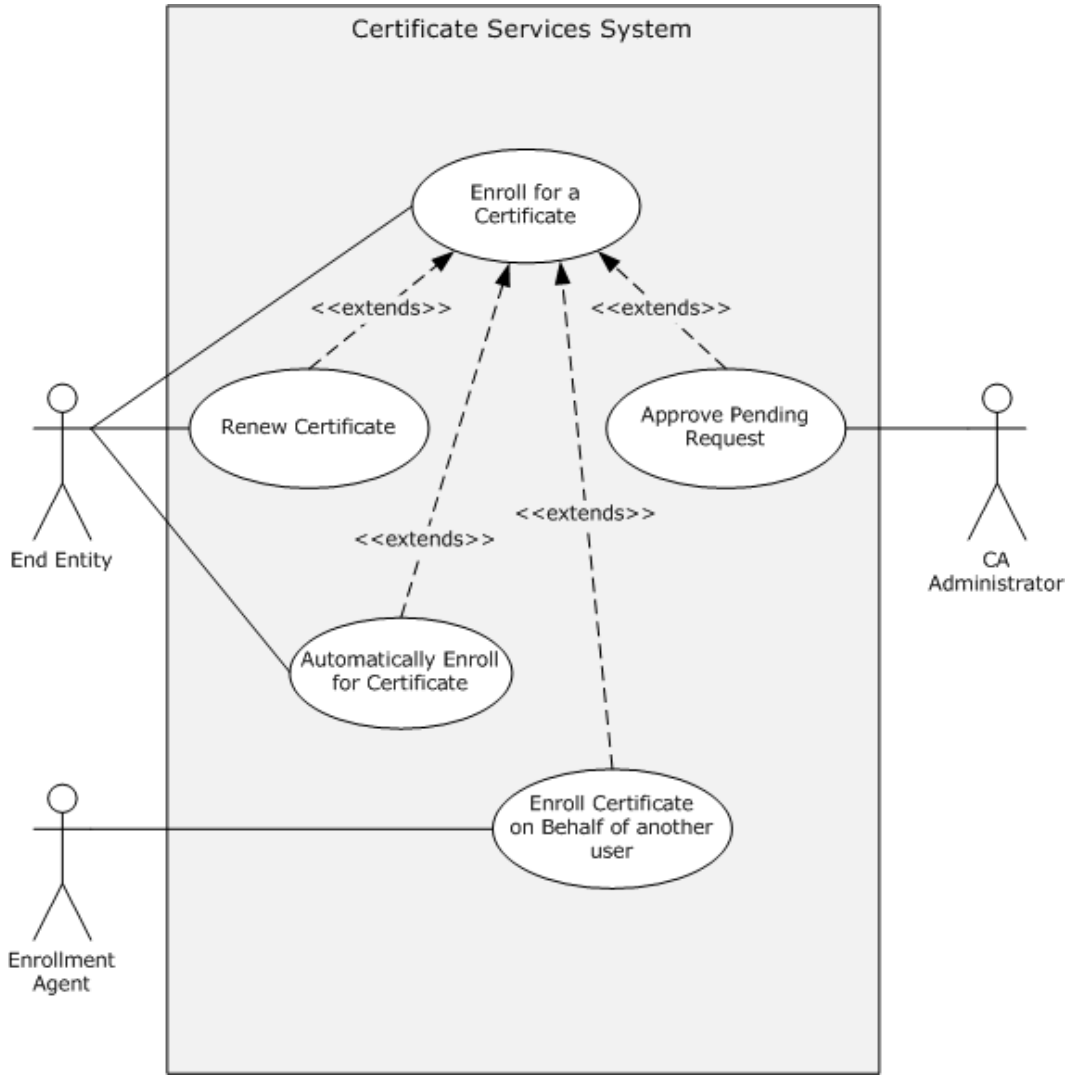

Figure 10: Enroll for a certificate

This use case allows a caller (either an End Entity, Enrollment Agent, or Autoenrollment client) to request a certificate from a CA. Upon successful completion of the use case, the End Entity receives a certificate signed by the CA.

Goal: To enroll for a certificate so that the End Entity is issued a certificate.

Context of Use: An End Entity can use a certificate for any number of different reasons and scenarios. When a certificate is required, a caller generates a certificate request and submits a certificate request to the CA as specified in [MS-WCCE]. The certificate enrollment can either be a new enrollment or a renewal. In the renewal case, an existing certificate is used to sign a request for a new certificate of the same type before being submitted to the CA. Depending upon the scenario, the caller might be an Enrollment Agent or Autoenrollment client, rather than the End Entity. In the Enroll on Behalf Of use case, a certificate request is signed by an Enrollment Agent before being submitted to the CA. Autoenrollment use case automatically handles certificate enrollment and the re-enrollment of expired certificates, which relieves the administrator from this task.

Direct Actor: The direct actor of this use case is the End Entity.

Primary Actors: The primary actors of this use case are the same as the direct actor, with the possible inclusion of an Enrollment Agent.

Supporting Actors: The CA Administrator could be a supporting actor in this use case.

Stakeholders and Interests:

- The primary interest of the End Entity is to submit certificate requests and receive certificates.
- The primary interest of the Enrollment Agent is to submit certificate request to the CA and receive certificates on behalf of the End Entity.
- The primary interest of the CA Administrator is approving pending certificate requests so that the CA can issue them.
- The primary interest of the Autoenrollment client is to submit the End Entity's certificate requests to the CA and to receive certificates automatically.

Preconditions: The End Entity, and possibly the Enrollment Agent and CA Administrator, require access to the CA.

Minimal Guarantees: The minimal guarantee is that End Entity gets the error message that provides the reason why the certificate request was not issued.

Success Guarantees: The CA System guarantees that it will be able to issue certificates when permitted by its policy algorithm.

Trigger: The certificate enrollment process is triggered when the CA receives a certificate request.

Main Success Scenario:

1. When the trigger occurs, the CA makes a decision on whether the certificate can be issued based on its policy.
2. The CA constructs a certificate based on the certificate request and its policy.
3. The CA signs the certificate and returns it to the client.

Extensions:

- Depending upon the configuration of the system, a CA Administrator might be involved in the certificate enrollment decision process. When the certificate request is held in a pending state by the CA, it requires CA Administrator approval before issuance, as specified in [MS-CSRA]. In the case of a request requiring administrator approval, the CA will hold the request in a pending state until a CA Administrator approves the request. Once approved, the certificate will be issued.

Post-conditions: The End Entity received the required certificate from CA.

2.5.3.2 CA Administration

The CA Administration use cases include generic functions such as editing the CA configuration, as well as more specific functions such as revoking certificates or recovering escrowed private keys from a CA. There are three primary CA Administration use cases:

- Edit CA configuration settings
- Recover an Archived Certificate and Key
- Revoke a certificate

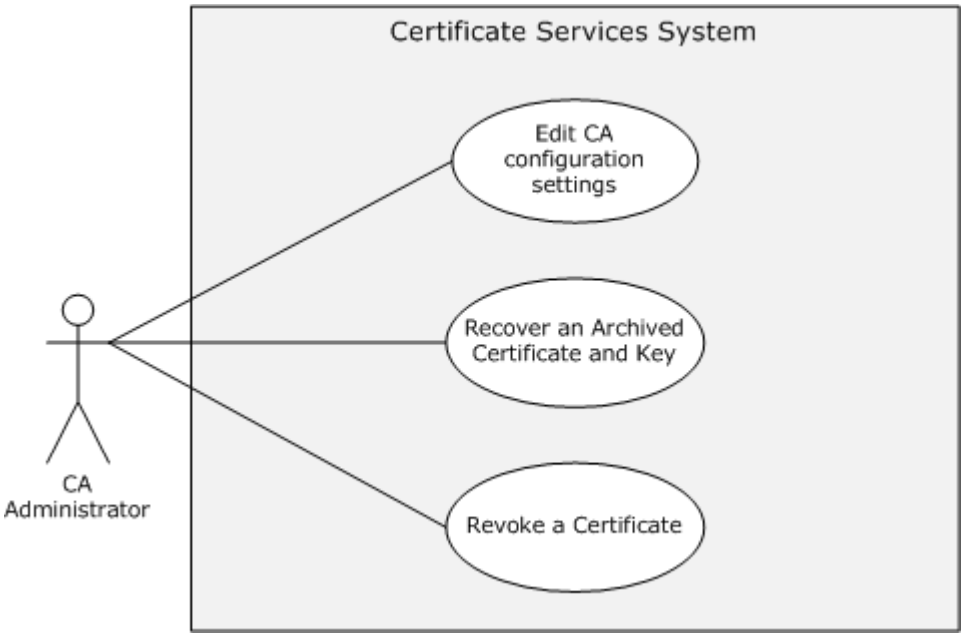

Figure 11: CA Administration use cases

2.5.3.2.1 Edit CA Configuration Settings - CA Administrator

Goal: To edit configuration settings on the CA. The goal of this use case is for the CA Administrator to be able to define and edit various configuration settings on the CA that affect behavior and policy around the issuance of certificates.

Context of Use: When a CA server is put into service, there is a variety of configuration settings that need to be defined by the CA Administrator in order for the CA operation to be in line with the requirements and desires of the enterprise or organization that has deployed it. In order to be able to define and edit these configuration settings and CA properties, the CA Administrator often is

required to be able to administer the CA remotely and can do so by using the interfaces defined in [MS-CSRA].

Direct Actor: The direct actor is the CA Administrator.

Primary Actor: The primary actor is the same as the direct actor.

Supporting Actors: None.

Stakeholders and Interests:

- The primary interest of the CA Administrator is to ensure the CA is configured and is working properly.
- The primary interest of the End Entity is assurance that the CA is configured correctly so that it can issue certificates as expected.

Preconditions: The CA Administrator requires access to the CA.

Minimal Guarantees: The minimal guarantee is that CA Administrator gets the error message that provides the reason why the Editing of CA configuration failed.

Success Guarantees:

- The CA System guarantees that configuration settings made by the CA Administrator will be maintained.
- The CA System guarantees that End Entities will be able to obtain certificates when requested in accordance with defined policy and configuration settings.

Trigger: The CA Administrator triggers all CA administration operations.

Main Success Scenario:

1. When the trigger occurs, the CA responds to connection attempts from the CA Administrator.
2. The CA Administrator then defines or edits configuration settings or CA properties as desired.

Extensions: None.

Post-conditions: The configuration settings or CA properties are updated on CA as desired.

2.5.3.2.2 Recover an Archived Certificate and Key

Goal: To recover a certificate and its private key that have been archived within the CA database (see the example in section 3.6).

Context of Use: Key archival and recovery is typically used in encryption scenarios. In the event an encryption certificate and/or its private key are unavailable for decryption, the ability to recover them will provide the ability to decrypt data that was encrypted using the certificate and its public key.

Key archival behavior is defined by a flag set within the certificate template as specified in [MS-CRTD] section 2.27. When an Enterprise CA issues a certificate based on a template with the key archival flag, the issued certificate and its corresponding private key are archived within the CA database as defined in [MS-WCCE] section 1.3.2.1.

Direct Actor: The direct actor is the CA Administrator.

Primary Actor: The primary actor is the same as the direct actor.

Supporting Actors: None.

Stakeholders and Interests:

- The primary interest of the CA Administrator is to ensure access to the CA and archived key material.
- The primary interest of the End Entity is ability to decrypt previously encrypted data using the retrieved private key.

Preconditions:

- The CA has been configured with one or more key recovery agent (KRA) certificates.
- An archived certificate has been issued by the CA and the archived material exists within the CA's database.
- The CA Administrator has access to any required KRA certificates and their private keys.

Minimal Guarantees: The minimal guarantee is that End Entity gets the error message that provides the reason why the recovering of an Archived Certificate and Key failed.

Success Guarantees:

- The CA System guarantees that archived certificates and private keys can be retrieved from the CA database.
- The CA System guarantees that archived material retrieved from the CA database can be decrypted using the associated KRA certificates and private keys.

Trigger: The CA Administrator triggers the recovery operation.

Main Success Scenario:

1. When the trigger occurs, archived certificate and key information is retrieved from the CA database by the requesting client.
2. Once retrieved, this information is then decrypted by using the associated KRA certificates and private keys.
3. The recovered certificate and private key are then available to be restored to the End Entity for use.

Extensions: None.

Post-conditions: The End Entity recovered the archived certificate and its private key.

2.5.3.2.3 Revoke a Certificate

Goal: To revoke a previously issued certificate and to publish a list of revoked certificates.

Context of Use: In the event it is desired to invalidate a previously issued certificate for any number of different reasons, such as a compromise, the CA Administrator can revoke the certificate and include this certificate within a CRL that can be referenced by any entity consuming the certificate and attempting to validate it.

Direct Actor: The direct actor is the CA Administrator.

Primary Actor: The primary actor is the same as the direct actor.

Supporting Actors: There are no supporting actors in this use case.

Stakeholders and Interests:

- The primary interest of the CA Administrator is to ensure that certificate is revoked and a new CRL is published.
- The primary interest of the End Entity is assurance that the revoked certificates referencing them are no longer valid.

Other application and system administrators might rely upon or use the End Entity's certificate for a variety of purposes, for assurance that certificates are valid for their intended purpose.

Preconditions:

- The CA has previously issued a certificate.
- The CA Administrator can provide the serial number of the certificate to be revoked.

Minimal Guarantees: The minimal guarantee is that CA administrator gets the error message that provides the reason why revocation of certificate failed.

Success Guarantees: The CA system guarantees that a certificate is revoked and added to a CRL.

Trigger: The CA Administrator requests a certificate revocation.

Main Success Scenario:

1. When the trigger occurs, the CA revokes the certificate.
2. The CA Administrator then invokes the CA to create and publish the CRL so the revoked status can be discovered by interested parties.

Extensions: None.

Post-conditions: Upon successful completion of the use case, the certificate is revoked and a new CRL is published with the latest information on the status of the certificate.

2.6 Versioning, Capability Negotiation, and Extensibility

There is no capability negotiation that is associated with this system. Any deviations from a specific version's implementation of these protocol specifications are documented in the respective protocol document. Capability negotiations between client and server implementations of these protocols are specified in the System Versioning and Capability Negotiation sections in their respective technical documents (TDs).

Three aspects of the system have multiple versions.

2.6.1 Interface Versions

There are multiple versions of the interfaces specified in [MS-WCCE] and [MS-CSRA]. The versioning rules for those interfaces are defined in section 1.7 of [MS-WCCE] and [MS-CSRA].

2.6.2 Client and Server Modes

The CA can operate in one of two modes: as a Standalone CA or as an Enterprise CA. The Standalone CA is specified in [MS-WCCE] section 3.2.1 and the Enterprise CA in [MS-WCCE] section 3.2.2. On client computers, these two modes correspond to the basic enrollment mode as specified in [MS-WCCE] section 3.1.1 and the enrollment based on certificate templates mode as specified in [MS-WCCE] section 3.1.2.

2.6.3 Certificate Template Versions

Certificate templates have three different versions as specified in [MS-CRTD] section 2.16. The processing rules for the client and server for each version of the certificate templates are specified in [MS-WCCE] sections 3.1.2 and 3.2.2.

2.7 Error Handling

The system does not define any errors beyond those described in the specifications of the member protocols, as listed in section 2.2.

Section 3 of the member protocol specifications describes the errors relevant to each protocol.

2.8 Coherency Requirements

There are several areas where coherency is important for the CA system:

- Access to the same type of information: When multiple clients attempt to perform operations that affect the same tables of the ADM (for example, submitting a new request that gets recorded in the Request table), the implementation must provide the record-level coherency for that table.
- Access to the same information: When multiple clients (possibly clients of different protocols) access that same datum of the ADM, the implementation must provide the datum-level coherency for that table.

Autoenrollment should have a timer that allows it to periodically execute to keep the local certificate storage current. Autoenrollment should execute at least twice a day<1>.

2.9 Security

This section documents systemwide security issues that are not otherwise described in the Technical Documents (TDs) for the Member Protocols. It does not duplicate what is already in the Member Protocol TDs unless there is some unique aspect that applies to the system as a whole.

Security is paramount for a CA because it stores sensitive data and issues certificates that are used for processes that can involve the organization's most important data. Therefore, it is critical that the system implementation be robust and resistant to attack. The security considerations include protection of the CA's signing and key exchange keys, protection of the requestor's data and private key that is being archived, and enforcement of certificate issuance policies that have been configured. These considerations call for the implementation of suitable protection for the storage of the CA's data, suitable protection of key recovery procedures, and the use of certificate templates for policy enforcement.

The CA serves as the foundation for authentication, authorization, encryption, and digital signatures. In other words, it is the cornerstone for many of an organization's information security capabilities. A good implementation of this system will include robust protection of data that is stored locally and transmitted to remote clients. [MS-CSRA] section 5 and [MS-WCCE] section 5 discuss security issues specific to the individual protocols.

2.9.1 Internal Security

There are several internal areas of the CA that have notable security considerations. This section discusses these in greater detail.

2.9.1.1 CA Signing Key

The CA uses its signing key to sign all certificates that it issues and all the CRLs that it publishes. This key is bound to the CA signing certificate. Therefore, there are several important properties to consider:

Strength of the key

Acceptable algorithms and **key lengths** are to be specified by enterprise security policy.

Lifetime of the key

The CA signing keys are long-lived keys that exceed the lifetime of the certificates that they sign because, when that key expires, all certificates signed with that key are no longer considered valid by others.

Key storage

If the CA signing key is compromised, certificates that were signed with that key can no longer be trusted, because an attacker could issue certificates that appear to originate from that CA.

CA signing certificate revocation

Organizations must have a documented process to handle the compromise of CA signing keys. For example, if the CA is subordinate to another CA, then it would make sense to revoke the compromised certificate on the parent CA and publish a new CRL. An even more severe situation occurs when the signing key of a **root CA** is compromised. In this situation, the only way to stop it from being **trusted** is to reconfigure all of the client computers to no longer trust it.

2.9.1.2 CA Data

Attackers could interfere with CA operations or tamper with certificate **revocation** information if they were able to access the CA data defined in section 2.1.3.3, Abstract Data Model. Therefore, it is a good idea to implement strong controls to protect this data and ensure that only authorized administrators are able to manage it.

Much of the data stored in the database is provided by the caller requesting a certificate. This caller could actually be an attacker. Therefore, it is recommended that each incoming request be validated before it is processed by the system. That is, a CA might inspect each incoming request to ensure that each field within the request is formatted correctly and that it does not exceed a reasonable size [HOWARD].

2.9.1.3 Certificate Templates

[MS-WCCE] section 5.1.11 describes data consistency considerations for the **certificate templates**. Additionally, it is reasonable to restrict write access to a certificate template to the administrators. Certificate templates define a policy by which certificates are issued. Therefore, an attacker who can modify certificate templates could potentially obtain certificates that would otherwise have been unobtainable.

2.9.1.4 Certificates for Special Roles

Although not required by the protocol, it is recommended to restrict the use of certificates that are issued for **KRAs** and enrollment agents by requiring explicit CA administrator approval. These certificates have special purposes in some of the scenarios for this system, as illustrated in examples.

2.9.1.5 Caller Authentication

As specified in [MS-CSRA] section 1.4 and in [MS-WCCE] section 2.1, the CA depends on a component implementing the server role of DCOM authentication to identify the caller of the DCOM interfaces that it implements.

2.9.2 External Security

There are several external areas of the CA that have notable security considerations. These external areas are discussed in this section.

2.9.2.1 Private Key Archival

There are several considerations for key archival. These considerations include transporting the private key from the client to the CA, storing the private key on the server, and recovering lost keys. Note that, while message formats and specific processing rules are described in [MS-CSRA] and [MS-WCCE], only security considerations are discussed here. [MS-WCCE] section 5.1.10 also addresses security considerations for the key archival.

2.9.2.2 CA Exchange Certificate

The public key in the CA exchange certificate can be used to encrypt end entities' private keys when requests for new certificates are sent to the CA (see [MS-WCCE] section 3.1.1.4.3.4). The concerns for key length that were presented for the CA signing key apply to this key as well. However, the lifetime of this private key might be shorter than the lifetime of the CA signing key. Also, this private key is not required to extend the validity of the certificates that the CA issues.

If this key is compromised, all of the certificates and private keys that were processed using the key can no longer be trusted since an attacker who possesses the private key could intercept and decrypt the end entity's private key.

Key storage considerations are the same as for the CA signing key. These certificates can be revoked and not used by the CA if they are compromised.

Storage and transmission of the Exchange public key is important because an attacker might generate its own key pair and if it could substitute its public key for a CA's Exchange public key, the client would be induced into encrypting a private key using that key for which the attacker has the private key.

2.9.2.3 Archived Key Storage

Neither the protocols nor the CA mandates any particular protection mechanism for the private keys archived by a CA. When choosing an algorithm and key sizes for the key protection, it is recommended that an implementer consider the lifetime of the key that is being protected and document its strength to set expectations for the clients of the system. For more information about the key archival and recovery process on the Microsoft Windows® platform, see [MSFT-ARCHIVE].

2.9.2.4 Key Recovery Agent Certificates

KRA certificates and the private keys associated with them can be used to protect and recover end entities' private keys. The CA does not need to possess the KRA's private key to archive keys, so the storage responsibility for KRA keys is solely on KRAs themselves. However, the CA Administrator who defines policies about what types of KRA certificates are issued and configured on the CA can ensure that they are appropriate for this purpose.

The KRA public key needs to be protected from tampering and especially replacement, because an attacker that could substitute its own public key for the KRA public key would potentially have access to all private keys encrypted under the KRA public key.

2.9.2.5 Transport Security

The CA uses the DCOM and **RPC** protocols for transport. Both DCOM and RPC provide authentication, data integrity, and **encryption** capabilities. Although those modes are not required by the CA protocols themselves, it is strongly recommended to both authenticate and encrypt at the DCOM and RPC layers.

2.9.2.6 Privacy

The CA stores data that was submitted by the client when the certificate was requested. Some of these data can be considered private by law in many jurisdictions, so it is important to provide access protection to the CA database for compliance.

2.10 Additional Considerations

None.

3 Examples

This section provides a series of examples illustrating the use of the Certificate Services System.

The examples are:

- Enrollment from a Standalone CA (Basic Enrollment)
- Enrollment from an Enterprise CA (Template-based Enrollment)
- Enrollment in the Domain Environment with XCEP/WSTEP Protocols
- Enrollment with CA Administrator Approval
- Enroll on Behalf of Request and Renewal
- Private Key Archival and Recovery
- Certificate Revocation
- Certificate Denied by Policy Algorithm
- Certificate Denied Due to Out-of-Sync Certificate Templates

3.1 Example 1: Enrollment from a Standalone CA (Basic Enrollment)

This example demonstrates the Enroll for a certificate use case described in section 2.5.3.1.

The goal of this example is to enroll for a certificate. The simplest case of certificate enrollment is basic enrollment. In this example, the caller creates a PKCS#10 request by populating its fields as the caller chooses. The caller then uses an implementation that has a WCCE client component to submit the request to the WCCE server (the CA).

Basic enrollment consists of a single message exchange between the client and the server where a client sends a certificate request to a server, which then issues the requested certificate.

Initial System State and Prerequisites

The example described in this section applies under the following conditions:

- The client implements the basic enrollment mode (in [MS-WCCE] section 3.1.1).
- The server implements the Standalone CA mode (in [MS-WCCE] section 3.2.1) and Standalone CA role configured on the server to issue the certificates.

Sequence

Figure 12: Basic enrollment

The message flow represented in the preceding figure is as follows:

1. The End Entity, using a WCCE client component, creates a PKCS#10 request and submits it to the CA as specified in [MS-WCCE] section 3.1.1.4.3.1.1.

2. The CA responds by issuing a certificate as specified in [MS-WCCE] section 3.2.1.4.2.1.4.1.1.

Final System state

- The End Entity has the issued certificate from CA.
- The CA-WCCE Server stores the request fields in the Request table as specified in [MS-WCCE] sections 3.2.1.4.2.1.4.3 and 3.2.1.4.2.1.4.4, along with the status of the certificate request and the End Entity details.

3.2 Example 2: Enrollment from an Enterprise CA (Template-based Enrollment)

This example demonstrates the Enroll for a certificate use case described in section 2.5.3.1.

This example builds on the example in section 3.1 by introducing an Enterprise CA. An Enterprise CA uses certificate templates for all certificate enrollments. Certificate templates, as defined in [MS-CRTD], contain data for requesting and issuing certificates. Policy algorithms use certificate templates to determine how to respond to certificate requests. In this example, the caller creates a certificate request PKCS#10, as specified in [MS-WCCE] section 3.1.1.4.3.1.1, based on the certificate template. The Enterprise CA then uses the template information to decide whether to issue the certificate, and if it does, how to construct the certificate.

Initial System State and Prerequisites

This example of certificate enrollment is based on the following assumptions:

- The End Entity operates in the client mode specified in [MS-WCCE] section 3.1.2 and the server implements the Enterprise CA mode as specified in [MS-WCCE] section 3.2.2.
- The Enterprise CA role is configured on the server to issue the certificates.
- The certificate templates are stored in Active Directory as specified by [MS-CRTD].

Sequence

The sequence of the steps for this example is organized into the following sections:

A. Query for available certificate templates from the Active Directory Active Directory server.

B. Request for a certificate.

A. Query for available certificate templates from the Active Directory server

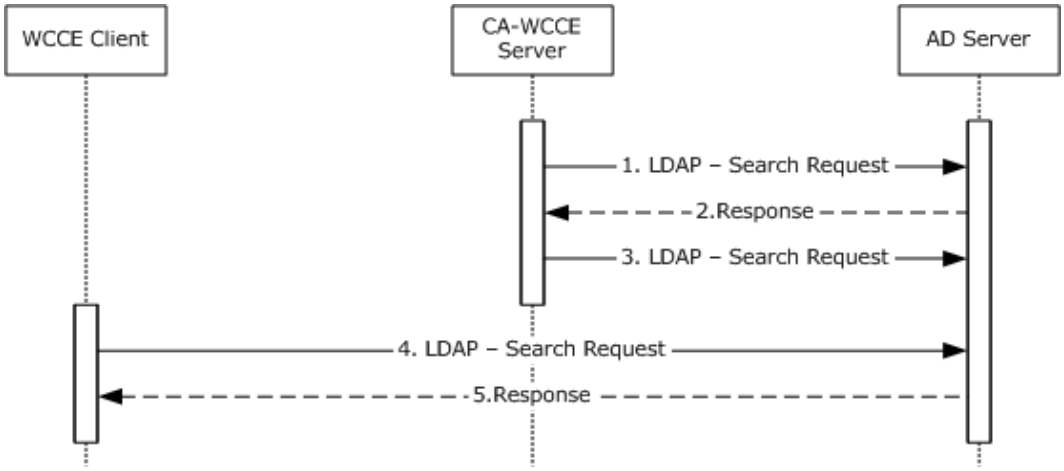

Figure 13: Query for available certificate templates from the Active Directory server

1. Upon startup, the CA-WCCE server requests the Active Directory server for certificate template data via an LDAP search request as described in [MS-WCCE] section 3.2.2.1.

2. The Active Directory server processes the request and responds with certificate template data in the format specified in [MS-WCCE] section 3.2.2.1.1.

3. The CA-WCCE server registers itself to receive change notifications, as specified in [MS-ADTS] section 3.1.1.3.4.1.9, when an attribute of a certificate template is being modified in order to stay up to date with any changes and avoid having to retrieve the templates for each request.

4. The WCCE client requests for the certificate templates from the Active Directory server via an LDAP search request as described in [MS-WCCE] section 3.2.2.1.

5. The Active Directory server responds with certificate templates in the format specified in [MS-WCCE] section 3.2.2.1.

B. Request for a certificate

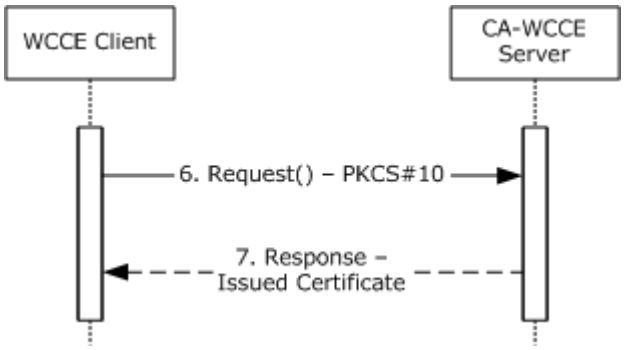

Figure 14: Request for a certificate

6. The End Entity, using the WCCE client component, creates a PKCS#10 request based on one of the certificate templates and submits it to the CA by calling the Request method specified in [MS-WCCE] section 3.1.2.4.2.

7. The CA checks the policy defined in the certificate template and concludes that it is appropriate to issue the certificate (see [MS-WCCE] section 3.2.2.6.2.1.4). The CA constructs a new certificate as defined by the certificate template (see [MS-WCCE] section 3.2.2.6.2.1.4) and sends a new certificate to the client.

Final System State

- The End Entity has the issued certificate from CA.
- The CA-WCCE Server stores the request fields in the Request table as specified in [MS-WCCE] sections 3.2.1.4.2.1.4.3 and 3.2.1.4.2.1.4.4 with the status of the certificate request and also the End Entity details.

3.3 Example 3: Enrollment in The Domain Environment with the XCEP/WSTEP Protocols

This example demonstrates the Enroll for a certificate use case described in section 2.5.3.1.

Initial System State and Prerequisites

This example is based on the following assumptions:

- The client computer, XCEP server, WSTEP server, and CA server exist in the same domain.
- The client computer is configured with Policy Server's endpoint information in its local configuration and the local configuration has the policy server endpoints information to locate the XCEP server.
- The WSTEP server operates in the client mode specified in [MS-WCCE] section 3.1.2 and the CA server implements the Enterprise CA mode specified in [MS-WCCE] section 3.2.2.
- The Enterprise CA role is properly configured on the CA Server to issue the certificates.

Sequence

The process and specific message flow in this example are as follows:

A. Query for certificate enrollment policies.

B. Request for a certificate.

A. Query for certificate enrollment policies

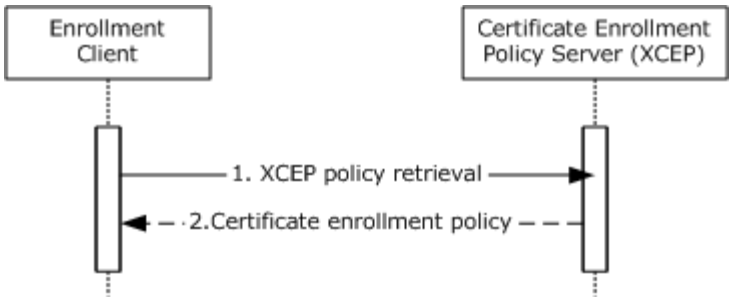

Figure 15: Query for certificate enrollment policies

1. The enrollment client sends a GetPolicies request [MS-XCEP] section 3.1.4.1 message to an XCEP server to retrieve the certificate enrollment policy information.

2. The XCEP server responds with the certificate enrollment policy (CEP) information.

B. Request for a certificate

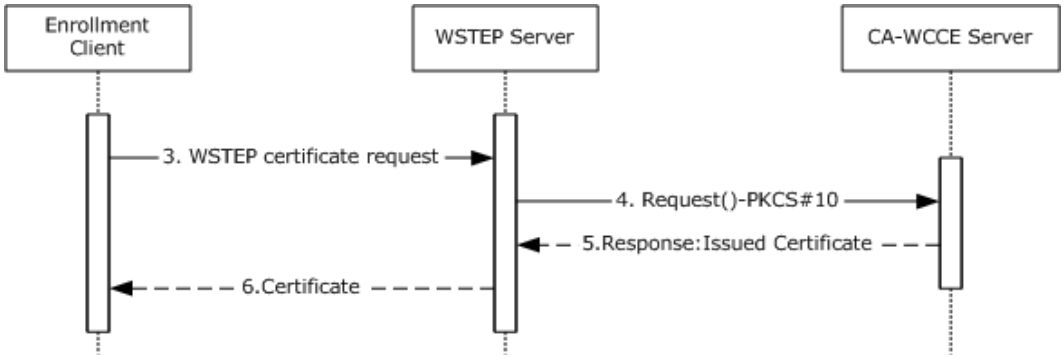

Figure 16: Request for certificate

3. Based on the CEP received from the XCEP server, the Enrollment client, using WSTEP client component, creates a RequestSecurityToken request as described in [MS-WSTEP], and submits the request to the WSTEP server.

4. The WSTEP server, using a WCCE client component, creates a PKCS #10 request and submits it to the CA as specified in [MS-WCCE] section 3.1.1.4.3.1.1.

5. The CA-WCCE server responds by issuing a certificate as specified in [MS-WCCE] section 3.2.1.4.2.1.4.1.1.

6. The WSTEP server responds with a newly issued certificate to its requested enrollment client.

Final System State

- The client computer has the issued certificate from the CA.

- The CA-WCCE Server stores the request fields in the Request table as specified in [MS-WCCE] sections 3.2.1.4.2.1.4.3 and 3.2.1.4.2.1.4.4, along with the status of the certificate request and the End Entity details.

3.4 Example 4: Enrollment with CA Administrator Approval

This example demonstrates the Enroll For a Certificate and Approve Pending Request use cases described in section 2.5.3.1.

This example builds on the example in section 3.2 by introducing a CA Administrator who modifies and approves the certificate request before the certificate is issued. One possible context for this scenario is where the certificate that is being requested requires a higher level of scrutiny before it can be issued, or requires input from someone other than the requestor.

Initial System State and Prerequisites

This example of certificate enrollment is based on the following additional assumption, in addition to the ones that are described in the example in section 3.2:

A certificate template has been defined in Active Directory that has the **msPKI-Enrollment-Flag CT_FLAG_PEND_ALL_REQUESTS** bit set (see [MS-WCCE] section 3.2.2.6.2.1.4.5.6).

Sequence

The sequence of the steps for this example is organized into the following sections:

A. Query for available certificate templates from the Active Directory server

B. Request for a certificate

C. Approve the pending certificate request

D. Get the issued certificate

A. Query for available certificate templates from the Active Directory server

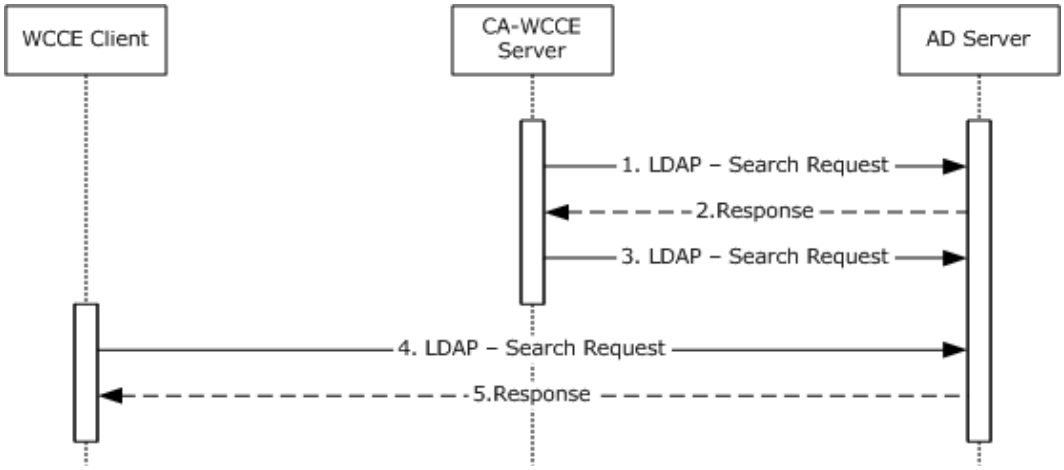

Figure 17: Query for available certificate templates

1. Upon startup, the CA-WCCE server requests the certificate template data from the Active Directory server via an LDAP search request as described in [MS-WCCE] section 3.2.2.1.

2. The Active Directory server processes the request and responds with the certificate template data in the format specified in [MS-WCCE] section 3.2.2.1.1.

3. The CA-WCCE server registers itself with the Active Directory server to receive change notifications, as specified in [MS-ADTS] section 3.1.1.3.4.1.9, when an attribute of a certificate template is being modified in order to stay up to date with any changes and avoid retrieving the templates for each request.

4. The WCCE client requests the certificate templates from the Active Directory server via an LDAP search request as described in [MS-WCCE] section 3.2.2.1.

5. The Active Directory server responds with certificate templates in the format specified in [MS-WCCE] section 3.2.2.1.

B. Request for a certificate

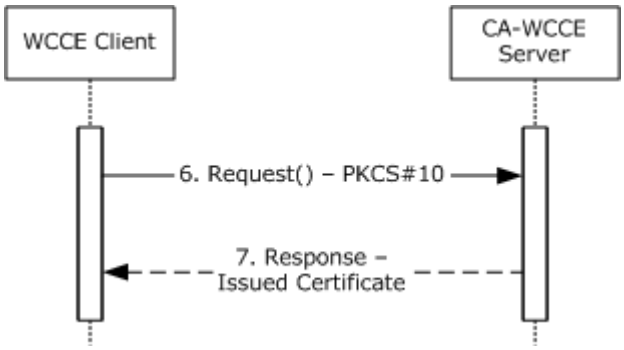

Figure 18: Request for a certificate

6. The End Entity, using the WCCE client, creates a PKCS#10 request based on one of the certificate templates and submits it to the CA by calling the Request method specified in [MS-WCCE] section 3.1.2.4.2.

7. The CA checks the certificate template and because the msPKI-Enrollment-Flag has the CT_FLAG_PEND_ALL_REQUESTS bit set (see [MS-WCCE] section 3.2.2.6.2.1.4.5.6), it records this request in its database and informs the client that the request's status is set to pending (see [MS-WCCE] section 3.2.1.4.2.1).

C. Approve the pending certificate request

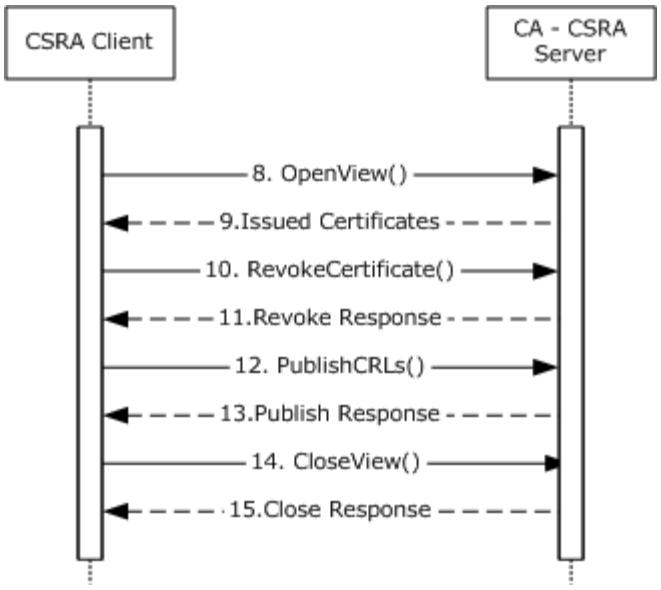

Figure 19: Approve the pending Certificate request

8. The CA Administrator, using an implementation that has a CSRA client component, queries the CA database to obtain information about pending requests by calling the OpenView method as specified in [MS-CSRA] section 3.1.4.1.12.

9. The CA-CSRA server responds with the list of the pending requests.

10. The CA Administrator sends the SetExtension method to add the certificate extension on the pending request which must be approved from the list returned in step 9.

11. The CA-CSRA server adds the requested certificate extensions as specified in [MS-CSRA] section 3.1.4.1.1 and returns a success response message.

12. The CA Administrator sends the ResubmitRequest method to approve the request.

13. The CA-CSRA server processes the request as specified [MS-CSRA] section 3.1.4.1.3, and returns the disposition as an issued certificate.

14. The CA Administrator requests the CA to close the CA database view by calling the CloseView method.

15. The CA-CSRA processes the CloseView method as specified in [MS-CSRA] section 3.1.4.1.14 and returns a success response message.

D. Get the issued certificate

Figure 20: Request for the issued certificate

16. After the certificate has been approved by the CA Administrator, the caller of the certificate, by using the WCCE client, requests the issued certificate from the CA by calling the Request method as specified in [MS-WCCE] section 3.1.1.4.3.5.

17. The CA processes the request and returns the issued certificate to the WCCE client.

Final System State

- The End Entity has the issued certificate request from the CA.
- The CA-WCCE server store the request fields in the Request table as specified in [MS-WCCE] sections 3.2.1.4.2.1.4.3 and 3.2.1.4.2.1.4.4, along with the status of the certificate request and the End Entity details.
- The CA-CSRA server has updated the extension table.

3.5 Example 5: Enroll on Behalf of Request and Renewal

This example demonstrates the Enroll for a certificate, Enroll Certificate on Behalf of User and Renew Certificate use cases described in section 2.5.3.1.

This example builds on the example in section 3.2 by introducing a cosigner for the certificate request. In this example, the enrollment agent creates and signs the initial certificate request. The enrollment agent then submits the signed request to the CA. The CA returns the issued certificate to the enrollment agent, who then provides the issued certificate to the end entity via an out-of-band process.

Later, when a certificate must be renewed, the end entity creates a renewal request and signs it with the key associated with its current certificate. The certificate template is configured to allow renewals when the request is signed by the end entity's existing valid certificate that is based on the same template. The end entity submits the renewal request to the CA. If the certificate used for the signature is still valid, the CA automatically renews the certificate and returns the issued certificate to the client.

Smart card certificates are typically provisioned in the following manner: The smart card user might visit an enrollment agent in person so that their identity can be verified. The enrollment agent can then submit the certificate request on their behalf. The end entity, however, is allowed to renew their certificate without again requiring the involvement of the enrollment agent. By signing the renewal request with its existing valid certificate, it is providing evidence that identity has already been verified.

Initial System State and Prerequisites

This example has the following additional assumption, in addition to ones that are described in the example in section 3.2:

- A certificate template has been defined in Active Directory that has the msPKI-RA-Application-Policies attribute set with enhanced key usage (EKU), for example, 1.3.6.1.4.1.311.20.2.1, Certificate Request Agent. The certificate template also has 0x00000040 (CT_FLAG_PREVIOUS_APPROVAL_VALIDATE_REENROLLMENT) bit set on the **msPKI-Enrollment-Flag** field.
- The enrollment agent has a certificate containing an EKU with the same object identifier (OID) as defined in the previous template's **msPKI-RA-Application-Policies** attribute.

Sequence

The sequence of the steps for this example is organized into the following sections:

A. Query for available certificate templates from the Active Directory server

B. Request for a certificate on behalf of another user

C. Query for available templates and renew the certificate on behalf of another user

A. Query for available certificate templates from the Active Directory server

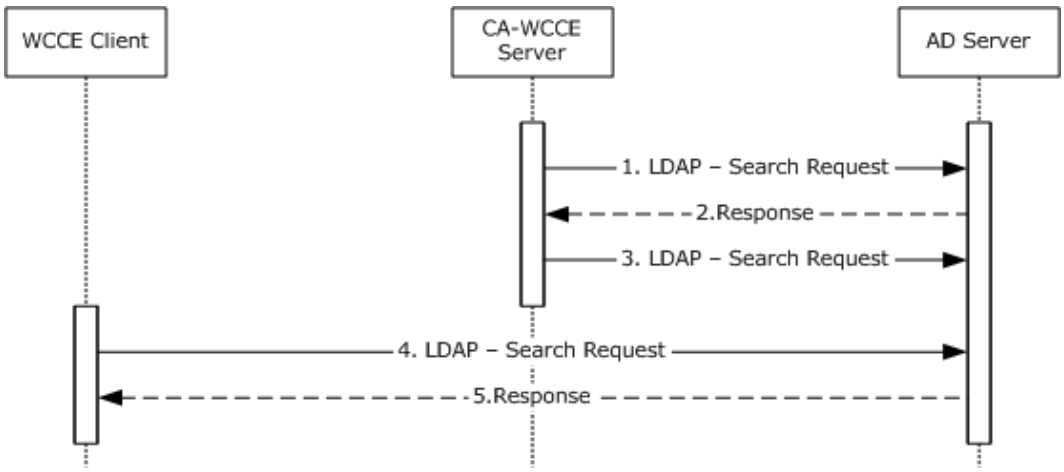

Figure 21: Query for available certificate templates

1. Upon startup, the CA-WCCE server requests the certificate template data from the Active Directory server via an LDAP search request as described in [MS-WCCE] section 3.2.2.1.

2. The Active Directory server processes the request and responds with certificate template data in the format specified in [MS-WCCE] section 3.2.2.1.1.

3. The CA-WCCE server registers itself to receive change notifications, as specified in [MS-ADTS] section 3.1.1.3.4.1.9, when an attribute of a certificate template is being modified in order to stay up to date with any changes and avoid retrieving the templates for each request.

4. The enrollment agent, using a WCCE client component, requests the certificate templates from the Active Directory server via an LDAP search.

5. Active Directory processes the request and returns the certificate templates.

B. Request for certificate on behalf of another user

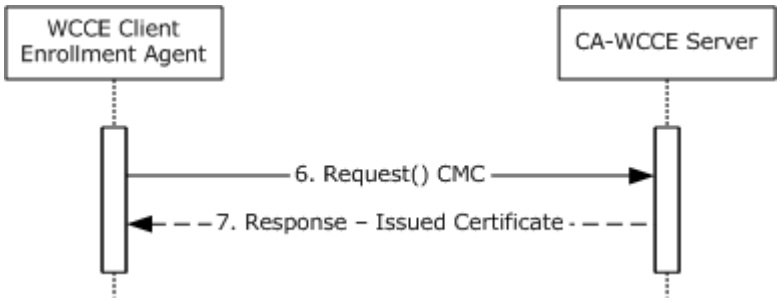

Figure 22: Request for certificate on behalf of another user

6. The enrollment agent generates a Cryptographic Message Syntax (CMS) structure with an embedded CMC request on behalf of another user, as specified in [MS-WCCE] section 3.1.1.4.3.3, and submits it to the CA by calling the Request method as specified in [MS-WCCE] section 3.1.2.4.2.

7. The CA determines that the certificate template that corresponds to the request requires the enrollment agent's signature. It validates the signature and verifies that the certificate associated with the signature has the required EKUs as specified in [MS-WCCE] section 3.2.2.6.2.1.2.1.2. When validation is complete, the CA issues the certificate and sends it to the enrollment agent.

The enrollment agent then transfers the new certificate to the end entity via an out-of-band process. The process for this communication is not defined within this document.

C. Query for certificate templates and renew certificate

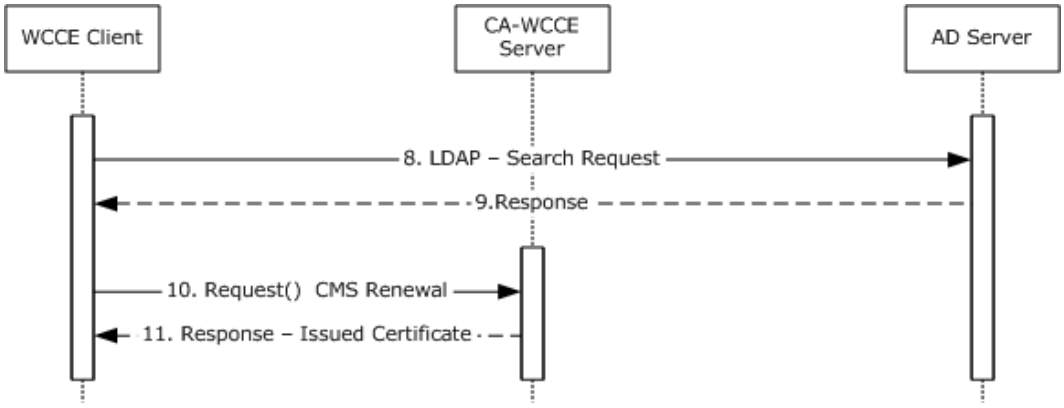

Figure 23: Query for certificate templates and renew the certificate

8. When it is time to renew the certificate, the End Entity uses a WCCE client component to retrieve the certificate templates from the Active Directory server via an LDAP search request.

9. The Active Directory server validates the request and returns the certificate templates.

10. The client creates a CMS renewal request and sends it to the CA as specified in [MS-WCCE] section 3.1.1.4.3.2.

11. When the CA receives the request, it checks the certificate template for the msPKI-Enrollment-Flag and confirms that the 0x00000040 (CT_FLAG_PREVIOUS_APPROVAL_VALIDATE_REENROLLMENT) bit is set, therefore allowing the use of the previous certificate to sign the request as specified in [MS-WCCE] section 3.2.2.6.2.1.2.3. The CA issues the renewed certificate and sends it to the End Entity.

Final System State

The End entity has the renewed certificate.

- The CA-WCCE Server store the request fields in the Request table as specified in [MS-WCCE] sections 3.2.1.4.2.1.4.3 and 3.2.1.4.2.1.4.4, along with certificate status and the requested End Entity details.

3.6 Example 6: Private Key Archival and Recovery

This example is the combination of two separate use cases. The first is the Enroll for a Certificate - End Entity use case, section 2.5.3.1, and the second is the Recover Archived Certificate and Key - CA Administrator use case, section 2.5.3.2. This example builds on the example in section 3.2 by introducing private key archival and recovery. A CA Administrator configures the CA to be able to archive private keys. An end entity enrolls for an encryption certificate, based upon a template that has been configured for archival. Later, a CA Administrator recovers the archived certificate and a private key from the CA database.

Initial System State and Prerequisites

This example of Key Archival and Recovery is based on the following additional assumption in addition to ones that are described in the example in section 3.2:

- The certificate template has been configured in Active Directory with the msPKI-Private-Key-Flag attribute with the 0x00000001 (CT_FLAG_REQUIRE_PRIVATE_KEY_ARCHIVAL) bit as specified in [MS-CRTD].
- A CA Administrator is acting as a KRA and has obtained a KRA encryption certificate.

Sequence

The process and specific message flow in this example are as follows:

A. Query for available certificate templates from Active Directory server

B. Configure CA to use KRA certificate

C. Archive private key

D. Recover private key

A. Query for available Certificate Templates from Active Directory server

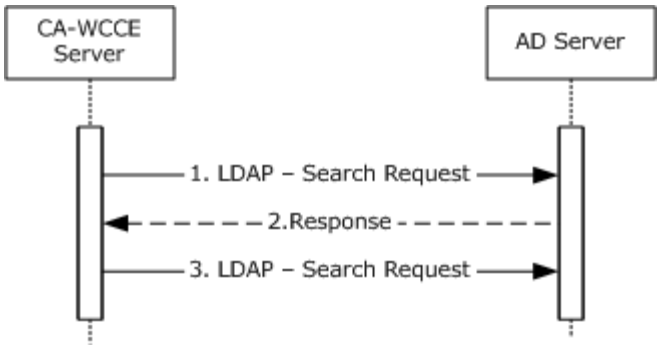

Figure 24: Query for certificate templates from Active Directory server

1. Upon startup, the CA-WCCE server requests the Active Directory server for certificate template data via an LDAP search request as described in [MS-WCCE] section 3.2.2.1.

2. The Active Directory server processes the request and responds with certificate template data in the format specified in [MS-WCCE] section 3.2.2.1.1.

3. The CA-WCCE server registers itself to receive change notifications, as specified in [MS-ADTS] section 3.1.1.3.4.1.9, when an attribute of a certificate template is being modified in order to stay up to date with any changes and avoid having to retrieve the templates for each request.

B. Configure CA to use KRA Certificate

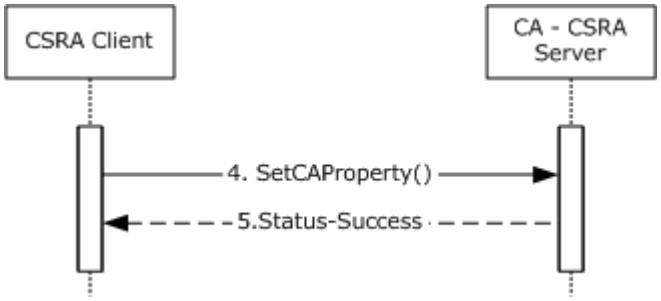

Figure 25: Configure CA to use KRA certificate

4. The CA Administrator configures the CA by using the CSRA client to use the KRA certificates when archiving private keys. This is accomplished by calling the **SetCAProperty** method and setting the 0x0000001a (CR_PROP_KRACERT) and 0x00000018 (CR_PROP_KRACERTUSEDCOUNT) properties as specified in [MS-CSRA] section 3.1.4.2.3.

5. The CA-CSRA server processes the request as per processing rules specified in [MS-CSRA] section 3.1.4.2.3 and return the success response message.

C. Archive Private Key

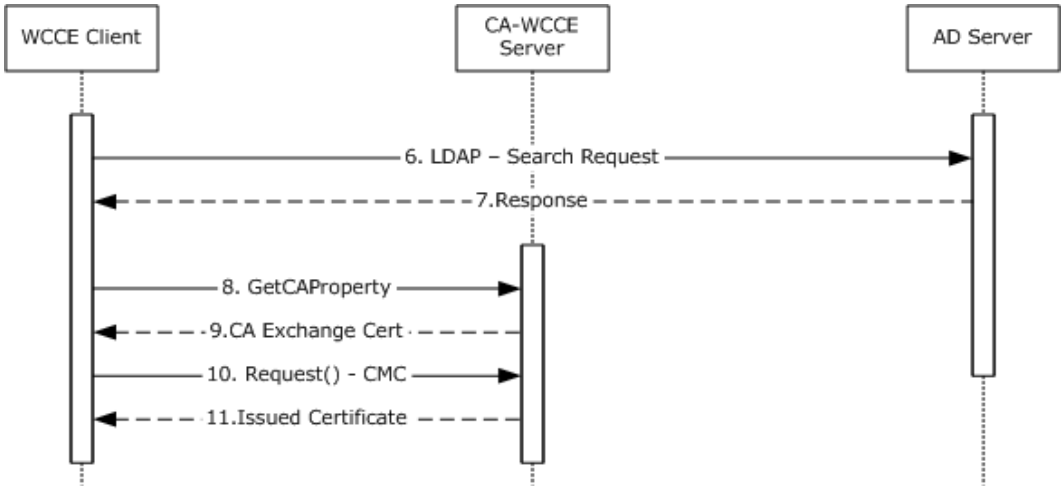

Figure 26: Query for Certificate templates and Archive private key

6. The end entity, using a WCCE client component, requests the certificate templates from the Active Directory server via an LDAP search request.

7. The Active Directory server returns the certificate templates in an LDAP search response.

8. Because the template specified by the caller has the 0x00000001 (CT_FLAG_REQUIRE_PRIVATE_KEY_ARCHIVAL) bit set on the msPKI-Private-Key-Flag attribute, the client calls GetCAProperty method to retrieve the CA exchange certificate.

9. The CA returns the exchange certificate and this exchange certificate is required to construct a certificate request with the private key archival information as specified in [MS-WCCE] section 3.1.1.4.3.4.

10. The client constructs a CMS request that includes CMC request and sends it to the CA as specified in [MS-WCCE] section 3.1.1.4.3.4.

11. When the CA receives the request, it issues a new certificate and archives the private key by using the KRA certificate that the administrator configured in step 3 as specified in [MS-WCCE] section 1.3.2.1.

D. Recover Private Key

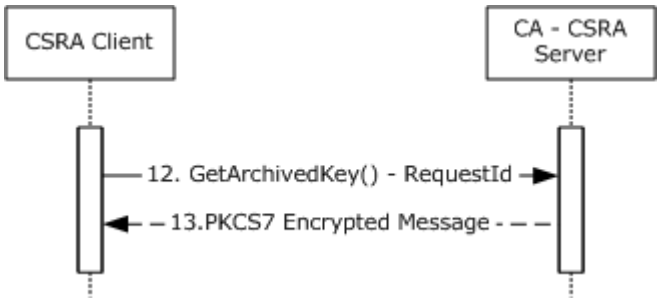

Figure 27: Recover the archived private key

12. Later, when the CA Administrator must recover the archived private key by using a CSRA client component, he calls the **GetArchivedKey** method to retrieve it as specified in [MS-CSRA] section 3.1.4.2.9.

13. The CA-CSRA server processes the request and it returns the archived private key to the CSRA client.

Final System State

- Private key of the requested certificate is archived on the CA.
- The CSRA administrator has the private key of the archived certificate.

3.7 Example 7: Certificate Revocation

The goal of this example is as detailed in the Revoke a Certificate - CA Administrator use case. This example builds on the example in section 3.2 by adding the process of revoking a previously issued certificate. A CA Administrator is able to revoke certificates for different reasons, such as private key compromise or cessation of operation.

Initial System State and Prerequisites

This example of Certificate Revocation is based on the following additional assumption as well as the ones that are described in the example in section 3.2:

- The Config_CA_CDP_Publish_To_Base and Config_CA_CDP_Publish_To_delta datum of the CA's ADM (see [MS-WCCE] section 3.2.1.1.4) are configured with valid paths.

Sequence

The process and specific message flow in this example are as follows:

A. Query for available certificate templates from the Active Directory server

B. Request for a certificate

C. Revoke the certificate

A. Query for Available Certificate Templates from the Active Directory server

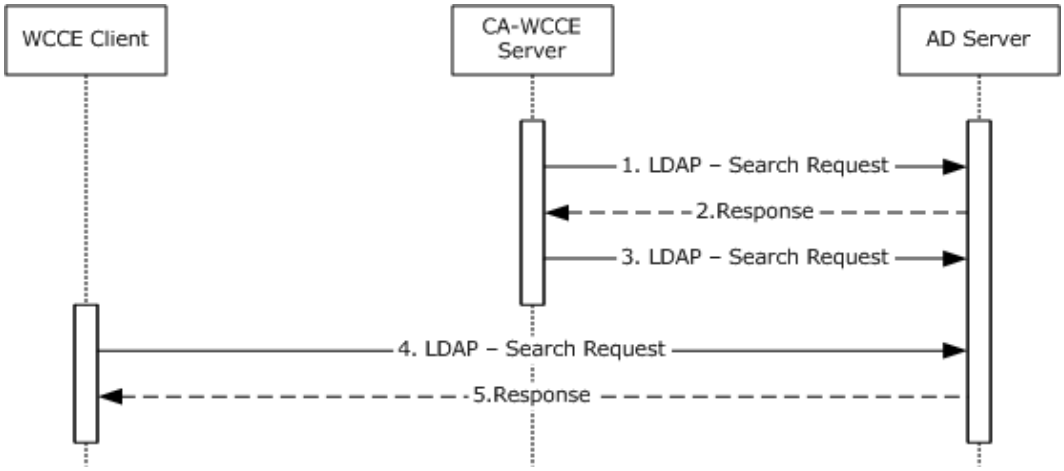

Figure 28: Query for available certificate templates from Active Directory server

1. Upon startup, the CA-WCCE server requests the certificate template data from the Active Directory server via an LDAP search request as described in [MS-WCCE] section 3.2.2.1.

2. The Active Directory server processes the request and responds with certificate template data in the format specified in [MS-WCCE] section 3.2.2.1.1.

3. The CA-WCCE server registers itself with the Active Directory server to receive change notifications, as specified in [MS-ADTS] section 3.1.1.3.4.1.9, when an attribute of a certificate template is being modified in order to stay up to date with any changes and avoid retrieving the templates for each request.

4. The End Entity, using a WCCE client, requests the certificate templates from the Active Directory server via an LDAP search request as described in [MS-WCCE] section 3.2.2.1.

5. The Active Directory server responds with the certificate templates in the format specified in [MS-WCCE] section 3.2.2.1.

B. Request for certificate

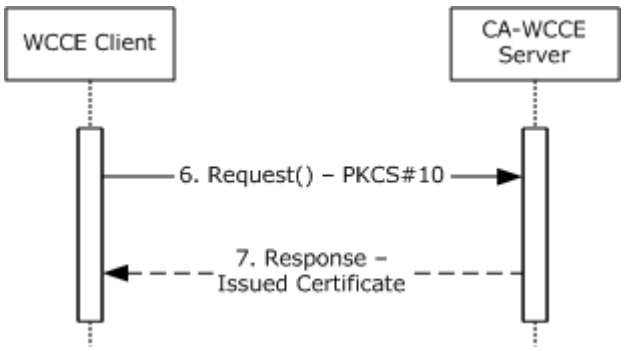

Figure 29: Request for certificate

6. The End Entity, using the WCCE client, creates a PKCS#10 request based on one of the certificate templates and submits it to the CA by calling the Request method specified in [MS-WCCE] section 3.1.2.4.2.

7. The CA checks the policy defined in the certificate template and concludes that it is appropriate to issue the certificate (see [MS-WCCE] section 3.2.2.6.2.1.4). The CA constructs a new certificate as defined by the certificate template (see [MS-WCCE] section 3.2.2.6.2.1.4) and sends a new certificate to the client.

C. Revoke the certificate

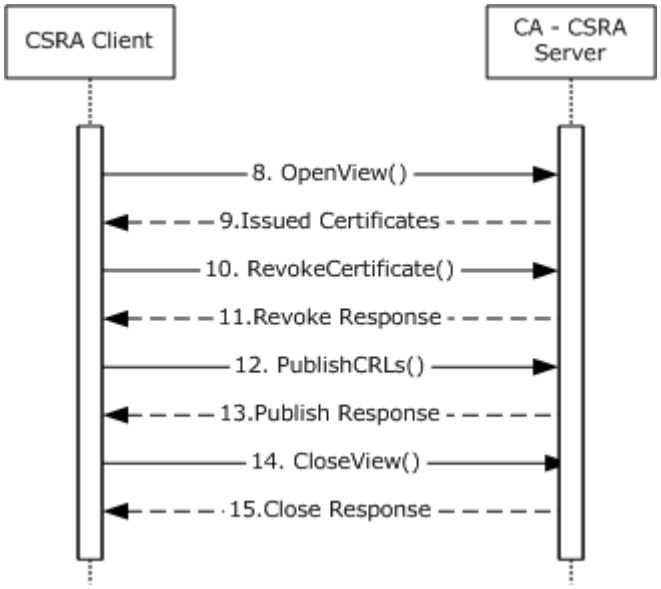

Figure 30: Revoke the certificate

8. Some point later, the issued certificate may need to be revoked. The CA Administrator, using a CSRA client component, queries the CA database to obtain the serial number of the certificate to be revoked by calling the **OpenView** method as specified in [MS-CSRA] section 3.1.4.1.12.

9. The CA returns the serial numbers of the certificates to the CSRA client.

10. The CA Administrator revokes the certificate by calling the RevokeCertificate method with the serial number of the intended certificate as specified in [MS-CSRA] section 3.1.4.1.8.

11. The CA revokes the requested certificate and returns a success response message to the CSRA client.

12. The CA Administrator instructs the CA to publish the CRL by calling the **PublishCRLs** method as specified in [MS-CSRA] section 3.1.4.2.1.

13. The CA publishes the CRLs on the configured publishing path.

14. The CA Administrator closes the CA database view by calling the **CloseView** method as specified in [MS-CSRA] section 3.1.4.1.14.

15. The CA validates the request and closes the database view.

Final System State

- The CA-WCCE Server store the request fields in the Request table as specified in [MS-WCCE] sections 3.2.1.4.2.1.4.3 and 3.2.1.4.2.1.4.4, along with the status of the certificate request and the End Entity details.
- The CA-CSRA server has updated the CRL table.

3.8 Example 8: Certificate Denied by the Policy Algorithm

This example represents a failure scenario for the Enroll for a Certificate - End Entity use case described in section 2.5.3.1.

This example builds on the example in section 3.2.

Initial System State and Prerequisites

This example is based on the following additional assumption, in addition to ones that are described in the example in section 3.2:

- The caller does not have permission to enroll.

Sequence

The process and specific message flow in this example are as follows:

A. Query for available certificate templates from the Active Directory server

B. Request for a certificate

A. Query for certificate templates from the Active Directory server

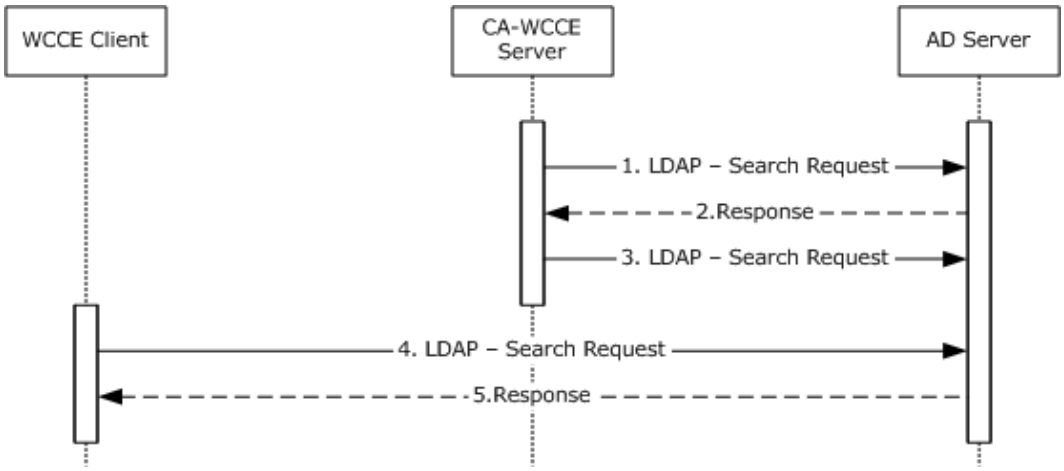

Figure 31: Query for available certificate templates from Active Directory server

1. Upon startup, the CA-WCCE server requests certificate template data from the Active Directory server via an LDAP search request as described in [MS-WCCE] section 3.2.2.1.

2. The Active Directory server processes the request and responds with certificate template data in the format specified in [MS-WCCE] section 3.2.2.1.1.

3. The CA-WCCE server registers itself with the Active Directory server to receive change notifications, as specified in [MS-ADTS] section 3.1.1.3.4.1.9, when an attribute of a certificate template is being modified in order to stay up-to-date with any changes and avoid retrieving the templates for each request.

4. The End Entity, using the WCCE client, requests the certificate templates from the Active Directory server via an LDAP search request as described in [MS-WCCE] section 3.2.2.1.

5. The Active Directory server responds with certificate templates in the format specified in [MS-WCCE] section 3.2.2.1.

B. Request for a certificate

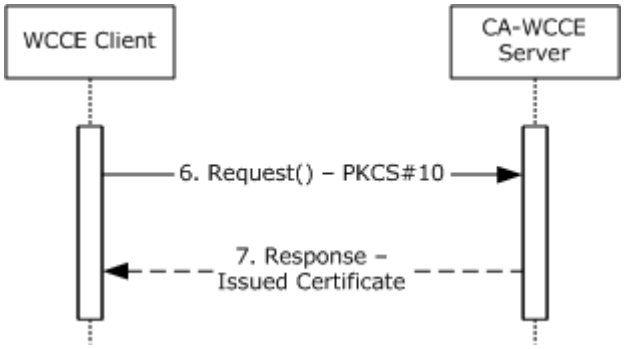

Figure 32: Request for a certificate

6. The End entity, using the WCCE client component, creates a PKCS#10 request based on one of the certificate templates and submits it to the CA by calling the Request method specified in [MS-WCCE] section 3.1.2.4.2.

7. When the CA receives the request, the policy algorithm is checked to see if it is to be issued. The CA examines ntSecurityDescriptor of the certificate template that corresponds to the request to determine if the caller has the permissions required to enroll for that template as specified in [MS-WCCE] section 3.2.2.6.2.1.4.3, and [MS-CRTD] section 2.5. In this example, the caller does not have permission, so the error 0x80094012L (CERTSRV_E_TEMPLATE_DENIED) is returned.

Final System State

- The CA-WCCE Server store the request fields in the Request table as specified in [MS-WCCE] sections 3.2.1.4.2.1.4.3 and 3.2.1.4.2.1.4.4 with the status of the certificate request and also the End Entity details.

3.9 Example 9: Certificate Denied Due to Out-of-Sync Certificate Templates

This example represents another failure scenario for the Enroll for a Certificate - End Entity use case described in section 2.5.3.1. This example builds on the example in section 3.2 and illustrates a situation where two Active Directory servers are out of sync, resulting in a version mismatch between the certificate templates used by client and server. Due to this mismatch, the server rejects the request. At a later time, after the directory has synchronized, the client submits another request that results in the certificate being issued.

Initial System state and Prerequisites

This example is based on the following additional assumptions in addition to ones that are described in the example in section 3.2:

- There is more than one Active Directory server on this network that replicates periodically.
- Active Directory replication occurs as discussed in [MS-DRSR].
- A CA Administrator has the appropriate security permissions to make modifications to certificate templates stored within Active Directory. Modifications made to Active Directory are performed as specified within [MS-ADTS].

Sequence

The process and specific message flow in this example are as follows:

A. Query for available certificate templates from Active Directory server

B. Modify certificate templates with new policies

C. Request for a certificate

D. DRSR Directory replication

E. Request for a certificate

A. Query for Certificate Templates from Active Directory server

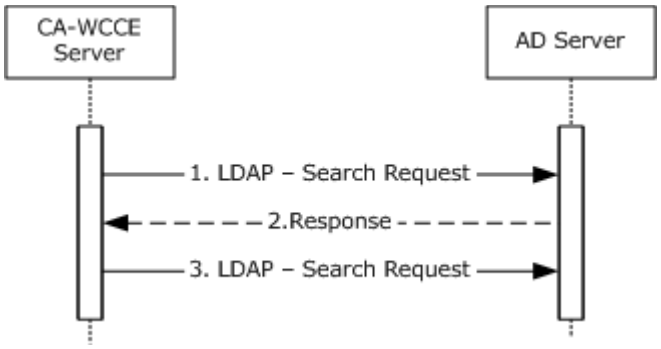

Figure 33: Query for certificate templates from Active Directory server

1. Upon startup, the CA-WCCE server requests the Active Directory server for certificate template data via an LDAP search request as described in [MS-WCCE] section 3.2.2.1.

2. The Active Directory server processes the request and responds with certificate template data in the format specified in [MS-WCCE] section 3.2.2.1.1.

3. The CA-WCCE server registers itself to receive change notifications, as specified in [MS-ADTS] section 3.1.1.3.4.1.9, when an attribute of a certificate template is being modified in order to stay up-to-date with any changes and avoid having to retrieve the templates for each request.

B. Modify Certificate templates with new policies

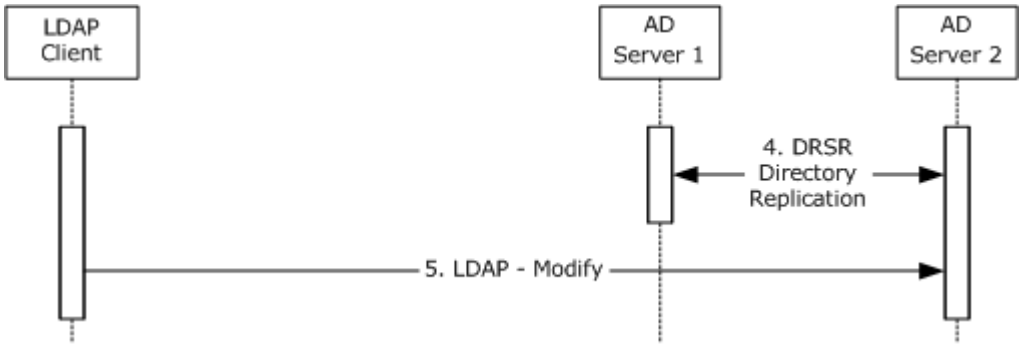

Figure 34: Modify certificate templates after Directory replication

4. At some later time, the two Active Directory servers replicate their information between each other as specified in [MS-DRSR].

5. After the replication, a CA Administrator using an LDAP client modifies some of the certificate templates with new policies on Active Directory Server 2. Modifications to Active Directory are performed as detailed in [MS-ADTS].

C. Request for a Certificate

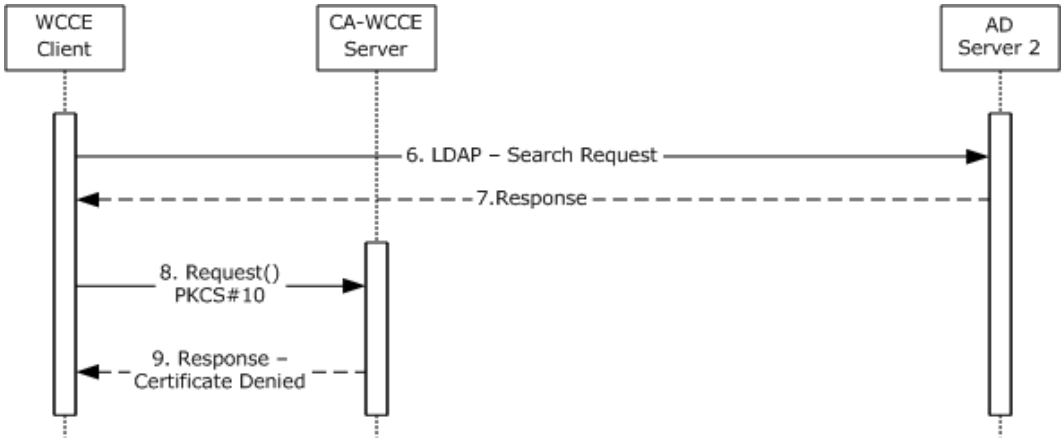

Figure 35: Request for a certificate

6. The WCCE client requests the certificate templates from the Active Directory server via an LDAP search request as described in [MS-WCCE] section 3.2.2.1.

7. The Active Directory server responds with certificate templates in the format specified in [MS-WCCE] section 3.2.2.1.

8. The client creates a PKCS#10 request based on one of the certificate templates and submits it to the CA by calling the Request method specified in [MS-WCCE] section 3.1.2.4.2.

9. As described in [MS-WCCE] section 3.2.2.6.2.1.4.2, the CA's policy algorithm verifies the certificate template version. The changes made on Active Directory Server 2 have not yet replicated to Active Directory Server 1. Because the CA has not been notified of the change to the template and the CA's certificate template instance is of an older version, the CA rejects a request, and replies with error (**CERTSRV_E_BAD_TEMPLATE_VERSION**).

D. DRSR Directory Replication

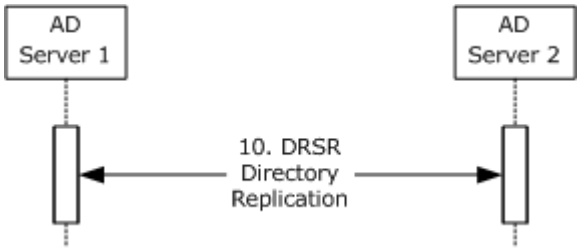

Figure 36: DRSR Directory Replication

10. At some later time, the two Active Directory servers replicate their information between each other as specified in [MS-DRSR].

E. Request for a certificate

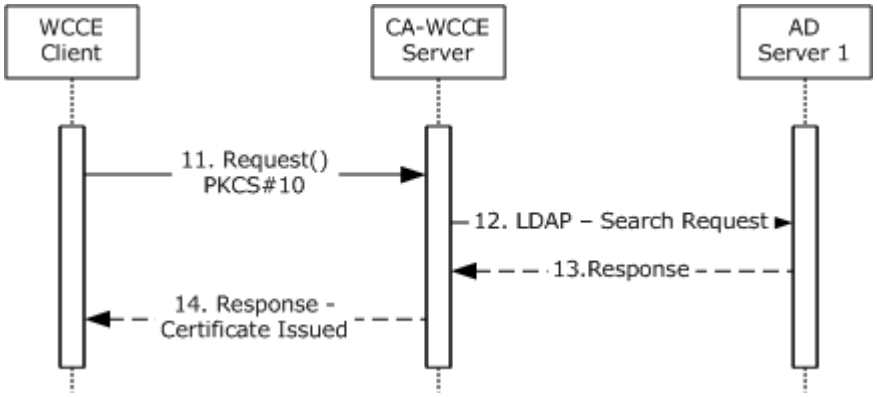

Figure 37: Request for a certificate

11. Some time later, the client attempts again the request for the same certificate in the same way as step 8.

12. This time the CA has registered the change in Active Directory because it has registered for asynchronous notifications in step 2.

13. The CA retrieves the updated certificate template data from the Active Directory Server 1 as specified in [MS-WCCE] section 3.2.2.1.1.

14. The CA checks the policy defined by the certificate and issues the certificate. The CA constructs the new certificate as defined by the certificate template (see [MS-WCCE] section 3.2.2.6.2.1.4) and returns the new certificate to the client.

Final System state

- The WCCE server stores the request fields in the Request table as specified in [MS-WCCE] sections 3.2.1.4.2.1.4.3 and 3.2.1.4.2.1.4.4.

4 Microsoft Implementations

The information in this specification is applicable to the following versions of Microsoft Windows®:

- Microsoft Windows® 2000 operating system
- Windows® XP operating system
- Windows Server® 2003 operating system
- Windows Server® 2003 R2 operating system
- Windows Vista® operating system
- Windows Server® 2008 operating system
- Windows® 7 operating system
- Windows Server® 2008 R2 operating system

Exceptions, if any, are noted in the following section.

4.1 Product Behavior

<1> Section 2.8: Only Windows Vista, Windows 7, and Windows Server 2008 R2 execute, based on this timer.

5 Change Tracking

No table of changes is available. The document is either new or has had no changes since its last release.

6 Index

T

U

V