System i

Security
Digital Certificate Manager

Version 5 Release 4

System i

Security
Digital Certificate Manager

Version 5 Release 4

Note

Before using this information and the product it supports, be sure to read the information in "Notices," on page 83.

Ninth Edition (October 2006)

This edition applies to version 5, release 4, modification 0 of IBM i5/OS (product number 5722-SS1) and to all subsequent releases and modifications until otherwise indicated in new editions. This version does not run on all reduced instruction set computer (RISC) models nor does it run on CISC models.

Contents

Digital Certificate Manager (DCM)

Digital Certificate Manager allows you to manage digital certificates for your network and use Secure Sockets Layer (SSL) to enable secure communications for many applications.

A digital certificate is an electronic credential that you can use to establish proof of identity in an electronic transaction. There are an increasing number of uses for digital certificates to provide enhanced network security measures. For example, digital certificates are essential to configuring and using the Secure Sockets Layer (SSL). Using SSL allows you to create secure connections between users and server applications across an untrusted network, such as the Internet. SSL provides one of the best solutions for protecting the privacy of sensitive data, such as user names and passwords, over the Internet. Many System i™ platforms and applications, such as FTP, Telnet, HTTP Server provide SSL support to ensure data privacy.

System i provides extensive digital certificate support that allows you to use digital certificates as credentials in a number of security applications. In addition to using certificates to configure SSL, you can use them as credentials for client authentication in both SSL and virtual private network (VPN) transactions. Also, you can use digital certificates and their associated security keys to sign objects. Signing objects allows you to detect changes or possible tampering to object contents by verifying signatures on the objects to ensure their integrity.

Capitalizing on the System i support for certificates is easy when you use Digital Certificate Manager (DCM), a free feature, to centrally manage certificates for your applications. DCM allows you to manage certificates that you obtain from any Certificate Authority (CA). Also, you can use DCM to create and operate your own local CA to issue private certificates to applications and users in your organization.

Proper planning and evaluation are the keys to using certificates effectively for their added security benefits. You might review these topics to learn more about how certificates work and how you can use DCM to manage them and the applications that use them:

Related information

Secure Sockets Layer (SSL)

Object signing and signature verification

What's new for V5R4

This information describes what information is new or significantly changed for the Digital Certificate Manager topic collection.

New information for Renewing Certificates

This new information explains a step by step process for renewing existing certificates with the local CA or with an Internet CA.

• "Renewing an existing certificate" on page 53

New information for Importing Certificates

This new to information explains a step by step process for importing certificates that are located in files on your server or files from another server.

• "Importing a certificate" on page 54

Enhancements to Certificate Revocation List (CRL) and Lightweight Directory Access Protocol (LDAP) information

This information was updated to include information about on how to anonymously bind to an LDAP server for CRL processing.

- "Managing CRL locations" on page 68
- "Managing LDAP location for user certificates" on page 71
- "Certificate Revocation List (CRL) Locations" on page 7

What's new as of 14 December 2007

If you are migrating users from one system to another, you can save and restore Digital Certificate Manager (DCM) information.

- "Restoring DCM information when migrating to another system" on page 28

How to see what's new or changed

To help you see where technical changes have been made, this information uses:

- The » image to mark where new or changed information begins.
- The « image to mark where new or changed information ends.

To find other information about what's new or changed this release, see the Memo to Users.

Printable PDF

Use this page to learn how to print the entire topic as a PDF file.

To view or download the PDF version of this topic, select Digital Certificate Manager (file size is about 600 KB or about 116 pages).

Saving PDF files

To save a PDF on your workstation for viewing or printing:

1. Right-click the PDF in your browser (right-click the link above).
2. Click **Save Target As** if you are using Internet Explorer. Click **Save Link As** if you are using Netscape Communicator.
3. Navigate to the directory in which you would like to save the PDF.
4. Click **Save**.

Downloading Adobe Acrobat Reader

You need Adobe Acrobat Reader to view or print these PDFs. You can download a copy from the Adobe Web site (www.adobe.com/products/acrobat/readstep.html) .

DCM concepts

Before you start using digital certificates to enhance your system and network security policy, you need to understand what they are and what security benefits they provide.

A digital certificate is a digital credential that validates the identity of the certificate's owner, much as a passport does. The identification information that a digital certificate provides is known as the subject

distinguished name. A trusted party, called a Certificate Authority (CA), issues digital certificates to users or to organizations. The trust in the CA is the foundation of trust in the certificate as a valid credential.

A digital certificate also contains a public key which is part of a public-private key pair. A variety of security functions rely on the use of digital certificates and their associated key pairs. You can use digital certificates to configure Secure Sockets Layer (SSL) sessions to ensure private, secure communication sessions between users and your server applications. You can extend this security by configuring many SSL-enabled applications to require certificates instead of user names and passwords for more secure user authentication.

To learn more about digital certificate concepts, review these topics:

Certificate extensions

Certificate extensions are information fields that provide additional information about the certificate.

Certificate extensions provide a means of expanding the original X.509 certificate information standards. While information for some extensions is provided to extend identification information for the certificate, other extensions provide information about the cryptographic capabilities of the certificate.

Not all certificates use the extension fields to extend distinguished name and other information. The number and type of extension fields that a certificate uses vary among the Certificate Authority (CA) entities that issue certificates.

For example, the local CA that Digital Certificate Manager (DCM) provides, allows you to use the Subject Alternative Name certificate extensions only. These extensions allow you to associate a certificate with a specific IP address, a fully-qualified domain name, or e-mail address. If you intend to use the certificate to identify an System i Virtual Private Network (VPN) connection endpoint, you must provide information for these extensions.

Related concepts

"Distinguished name" on page 4
Use this information to learn about the identification characteristics of digital certificates.

Certificate renewal

The certificate renewal process that Digital Certificate Manager (DCM) uses varies based on the type of Certificate Authority (CA) that issued the certificate.

If you use the local CA to sign the renewed certificate, DCM uses the information that you provide to create a new certificate in the current certificate store and retains the previous certificate.

If you use a well-known, Internet CA to issue the certificate, you can handle the certificate renewal in one of two ways: to import the renewed certificate from a file you receive from the signing CA or to have DCM create a new public-private key pair for the certificate. DCM provides the first option in case you prefer to renew the certificate directly with the CA that issued it.

If you choose to create a new key pair, DCM handles the renewal in the same way that it handled the creation of the certificate. DCM creates a new public-private key pair for the renewed certificate and generates a Certificate Signing Request (CSR) which consists of the public key and other information that you specify for the new certificate. You can use the CSR to request a new certificate from VeriSign or any other public CA. Once you receive the signed certificate from the CA, you use DCM to import the certificate into the appropriate certificate store. The certificate store then contains both copies of the certificate, the original and the newly issued renewed certificate.

If you choose not to have DCM generate a new key pair, DCM guides you through the process of importing the renewed, signed certificate into the certificate store from an existing file that you received from the CA. The imported, renewed certificate then replaces the previous certificate.

Distinguished name

Use this information to learn about the identification characteristics of digital certificates.

Each CA has a policy to determine what identifying information the CA requires to issue a certificate. Some public Internet Certificate Authorities may require little information, such as a name and e-mail address. Other public CAs may require more information and require stricter proof of that identifying information before issuing a certificate. For example, CAs that support Public Key Infrastructure Exchange (PKIX) standards, may require that the requester verify identity information through a Registration Authority (RA) before issuing the certificate. Consequently, if you plan to accept and use certificates as credentials, you need to review the identification requirements for a CA to determine whether their requirements fit your security needs.

Distinguished name (DN) is a term that describes the identifying information in a certificate and is part of the certificate itself. A certificate contains DN information for both the owner or requestor of the certificate (called the Subject DN) and the CA that issues the certificate (called the Issuer DN). Depending on the identification policy of the CA that issues a certificate, the DN can include a variety of information. You can use Digital Certificate Manager (DCM) to operate a private Certificate Authority and issue private certificates. Also, you can use DCM to generate the DN information and key pair for certificates that a public Internet CA issues for your organization. The DN information that you can provide for either type of certificate includes:

- Certificate owner's common name
- Organization
- Organizational unit
- Locality or city
- State or province
- Country or region

When you use DCM to issue private certificates, you can use certificate extensions to provide additional DN information for the certificate, including:

- Version 4 IP address
- Fully qualified domain name
- E-mail address

Related concepts

"Certificate extensions" on page 3
Certificate extensions are information fields that provide additional information about the certificate.

Digital signatures

A digital signature on an electronic document or other object is created by using a form of cryptography and is equivalent to a personal signature on a written document.

A digital signature provides proof of the object's origin and a means by which to verify the object's integrity. A digital certificate owner "signs" an object by using the certificate's private key. The recipient of the object uses the certificate's corresponding public key to decrypt the signature, which verifies the integrity of the signed object and verifies the sender as the source.

A Certificate Authority (CA) signs certificates that it issues. This signature consists of a data string that is encrypted with the Certificate Authority's private key. Any user can then verify the signature on the certificate by using the Certificate Authority's public key to decrypt the signature.

A digital signature is an electronic signature that you or an application creates on an object by using a digital certificate's private key. The digital signature on an object provides a unique electronic binding of the identity of the signer (the owner of the signing key) to the origin of the object. When you access an object that contains a digital signature, you can verify the signature on the object to verify the source of the object as valid (for example, that an application you are downloading actually comes from an authorized source such as IBM®). This verification process also allows you to determine whether there have been any unauthorized changes to the object since it was signed.

An example of how a digital signature works

A software developer has created an i5/OS® application that he wants to distribute over the Internet as a convenient and cost-effective measure for his customers. However, he knows that customers are justifiably concerned about downloading programs over the Internet due to the increasing problem of objects that masquerade as legitimate programs but really contain harmful programs, such as viruses.

Consequently, he decides to digitally sign the application so that his customers can verify that his company is the legitimate source of the application. He uses the private key from a digital certificate that he has obtained from a well-known public Certificate Authority to sign the application. He then makes it available for his customers to download. As part of the download package he includes a copy of the digital certificate that he used to sign the object. When a customer downloads the application package, the customer can use the certificate's public key to verify the signature on the application. This process allows the customer to identify and verify the of the application, as well as ensure that the contents of the application object has not been altered since it was signed.

Related concepts

"Certificate Authority (CA)" on page 6
A Certificate Authority (CA) is a trusted central administrative entity that can issue digital certificates to users and servers.

"Cryptography" on page 9
Shared and public keys are two different types of cryptographic functions that digital certificates use to provide security.

"Public-private key pair"
Every digital certificate has a pair of associated cryptographic keys that consist of a private key and a public key.

Public-private key pair

Every digital certificate has a pair of associated cryptographic keys that consist of a private key and a public key.

Note: Signature verification certificates are an exception to this rule and have an associated public key only.

A public key is part of the owner's digital certificate and is available for anyone to use. A private key, however, is protected by and available only to the owner of the key. This limited access ensures that communications that use the key are kept secure.

The owner of a certificate can use these keys to take advantage of the cryptographic security features that the keys provide. For example, the certificate owner can use a certificate's private key to ″sign″ and encrypt data sent between users and servers, such as messages, documents, and code objects. The recipient of the signed object can then use the public key contained in the signer's certificate to decrypt the signature. Such digital signatures ensure the reliability of an object's origin and provide a means of checking the integrity of the object.

Related concepts

"Digital signatures" on page 4
A digital signature on an electronic document or other object is created by using a form of cryptography and is equivalent to a personal signature on a written document.

"Certificate Authority (CA)"
A Certificate Authority (CA) is a trusted central administrative entity that can issue digital certificates to users and servers.

Certificate Authority (CA)

A Certificate Authority (CA) is a trusted central administrative entity that can issue digital certificates to users and servers.

The trust in the CA is the foundation of trust in the certificate as a valid credential. A CA uses its private key to create a digital signature on the certificate that it issues to validate the certificate's origin. Others can use the CA certificate's public key to verify the authenticity of the certificates that the CA issues and signs.

A CA can be either a public commercial entity, such as VeriSign, or it can be a private entity that an organization operates for internal purposes. Several businesses provide commercial Certificate Authority services for Internet users. Digital Certificate Manager (DCM) allows you to manage certificates from both public CAs and private CAs.

Also, you can use DCM to operate your own private local CA to issue private certificates to systems and users. When the local CA issues a user certificate, DCM automatically associates the certificate with the user's System i user profile or other user identity. Whether DCM associates the certificate with a user profile or with a different user identity for the user depends on whether you configure DCM to work with Enterprise Identity Mapping (EIM). This ensures that the access and authorization privileges for the certificate are the same as those for the owner's user profile.

Trusted root status

The term trusted root refers to a special designation that is given to a Certificate Authority certificate. This trusted root designation allows a browser or other application to authenticate and accept certificates that the Certificate Authority (CA) issues.

When you download a Certificate Authority's certificate into your browser, the browser allows you to designate it as a trusted root. Other applications that support using certificates must also be configured to trust a CA before the application can authenticate and trust certificates that a specific CA issues.

You can use DCM to enable or disable the trust status for a Certificate Authority (CA) certificate. When you enable a CA certificate, you can specify that applications can use it to authenticate and accept certificates that the CA issues. When you disable a CA certificate, you cannot specify that applications can use it to authenticate and accept certificates that the CA issues.

Certificate Authority policy data

When you create a local Certificate Authority (CA) with Digital Certificate Manager, you can specify the policy data for the local CA. The policy data for a local CA describes the signing privileges that it has. The policy data determines:

- Whether the local CA can issue and sign user certificates.
- How long certificates that the local CA issues are valid.

Related concepts

"Digital signatures" on page 4
A digital signature on an electronic document or other object is created by using a form of cryptography and is equivalent to a personal signature on a written document.

"Public-private key pair" on page 5
Every digital certificate has a pair of associated cryptographic keys that consist of a private key and a public key.

Certificate Revocation List (CRL) Locations

A Certificate Revocation List (CRL) is a file that lists all invalid and revoked certificates for a specific Certificate Authority (CA).

CA's periodically update their CRLs and make them available for others to publish in Lightweight Directory Access Protocol (LDAP) directories. A few CAs, such as SSH in Finland, publish the CRL themselves in LDAP directories that you can access directly. If a CA publishes their own CRL, the certificate indicates this by including a CRL distribution point extension in the form of a Uniform Resource Identifier (URI).

Digital Certificate Manager (DCM) allows you to define and manage CRL location information to ensure more stringent authentication for certificates that you use or you accept from others. A CRL location definition describes the location of, and access information for, the Lightweight Directory Access Protocol (LDAP) server that stores the CRL.

When connecting to an LDAP server you need to supply a DN and password to avoid anonymously binding to an LDAP server. Binding anonymously to the server does not provide the level of authority needed to access a ″critical″ attribute such as the CRL. In such a case, DCM may validate a certificate with a revoked status because DCM is unable to obtain the correct status from the CRL. If you want to access the LDAP server anonymously, you need to use the Directory Server Web Administration Tool and select the ″Manage schema″ task to change the security class (also referred to as ″access class″) of the **certificateRevocationList** and **authorityRevocationList** attributes from ″critical″ to ″normal″.

Applications that perform certificate authentication access the CRL location, if one is defined, for a specific CA to ensure that the CA has not revoked a specific certificate. DCM allows you to define and manage the CRL location information that applications need to perform CRL processing during certificate authentication. Examples of applications and processes that may perform CRL processing for certificate authentication are: the virtual private networking (VPN) Internet Key Exchange (IKE) server, Secure Sockets Layer (SSL) enabled-applications, and the object signing process. Also, when you define a CRL location and associate it with a CA certificate, DCM performs CRL processing as part of the validating process for certificates that the specified CA issues. .

Related concepts

"Validating certificates and applications" on page 67
You can use Digital Certificate Manager (DCM) to validate individual certificates or the applications that use them. The list of things that DCM checks differs slightly depending on whether you are validating a certificate or an application.

Related tasks

"Managing CRL locations" on page 68
Digital Certificate Manager (DCM) allows you to define and manage Certificate Revocation List (CRL) location information for a specific Certificate Authority (CA) to use as part of the certificate validation process.

Certificate stores

A certificate store is a special key database file that Digital Certificate Manager (DCM) uses to store digital certificates.

The certificate store contains the certificate's private key unless you choose to use an IBM Cryptographic Coprocessor to store the key instead. DCM allows you to create and manage several types of certificate stores. DCM controls access to certificate stores through passwords in conjunction with access control of the integrated file system directory and the files that constitute the certificate store.

Certificate stores are classified based on the types of certificates that they contain. The management tasks that you can perform for each certificate store vary based on the type of certificate that the certificate store contains. DCM provides the following predefined certificate stores that you can create and manage:

local Certificate Authority (CA)
DCM uses this certificate store to store the local CA certificate and its private key if you create a local CA. You can use the certificate in this certificate store to sign certificates that you use the local CA to issue. When the local CA issues a certificate, DCM puts a copy of the CA certificate (without the private key) in the appropriate certificate store (for example, *SYSTEM) for authentication purposes. Applications use CA certificates to verify the origination of certificates that they must validate as part of the SSL negotiation to grant authorization to resources.

***SYSTEM**
DCM provides this certificate store for managing server or client certificates that applications use to participate in Secure Sockets Layer (SSL) communications sessions. System i applications (and many other software developers' applications) are written to use certificates in the *SYSTEM certificate store only. When you use DCM to create a local CA, DCM creates this certificate store as part of the process. When you choose to obtain certificates from a public CA, such as VeriSign, for your server or client applications to use, you must create this certificate store.

***OBJECTSIGNING**
DCM provides this certificate store for managing certificates that you use to digitally sign objects. Also, the tasks in this certificate store allow you to create digital signatures on objects, as well as view and verify signatures on objects. When you use DCM to create a local CA, DCM creates this certificate store as part of the process. When you choose to obtain certificates from a public CA, such as VeriSign, for signing objects, you must create this certificate store.

***SIGNATUREVERIFICATION**
DCM provides this certificate store for managing certificates that you use to verify the authenticity of digital signatures on objects. To verify a digital signature, this certificate store must contain a copy of the certificate that signed the object. The certificate store must also contain a copy of the CA certificate for the CA that issued the object signing certificate. You obtain these certificate either by exporting object signing certificates on the current system into the store or by importing certificates that you receive from the object signer.

Other System Certificate Store
This certificate store provides an alternate storage location for server or client certificates that you use for SSL sessions. Other System Certificate Stores are user-defined secondary certificate stores for SSL certificates. The Other System Certificate Store option allows you to manage certificates for applications that you or others write that use the SSL_Init API to programmatically access and use a certificate to establish an SSL session. This API allows an application to use the default certificate for a certificate store rather than a certificate that you specifically identify. Most commonly, you use this certificate store when migrating certificates from a prior release of DCM, or to create a special subset of certificates for SSL use.

Note: If you have an IBM Cryptographic Coprocessor installed on your system, you can choose other private key storage options for your certificates (with the exception of object signing certificates). You can elect to store the private key on the coprocessor itself or use the coprocessor to encrypt the private key and store it in a special key file instead of in a certificate store.

DCM controls access to certificate stores through passwords. DCM also maintains access control of the integrated file system directory and the files that constitute the certificate stores. The local Certificate Authority (CA), *SYSTEM, *OBJECTSIGNING, and *SIGNATUREVERIFICATION certificate stores must be located in the specific paths within the integrated file system, Other System Certificate stores can be located anywhere in the integrated file system.

Related concepts

"Types of digital certificates" on page 28
Use this information to learn about the different types of digital certificates and how they are used in the Digital Certificate Manager (DCM).

Cryptography

Shared and public keys are two different types of cryptographic functions that digital certificates use to provide security.

Cryptography is the science of keeping data secure. Cryptography allows you to store information or to communicate with other parties while preventing non involved parties from understanding the stored information or understanding the communication. Encryption transforms understandable text into an unintelligible piece of data (ciphertext). Decrypting restores the understandable text from the unintelligible data. Both processes involve a mathematical formula or algorithm and a secret sequence of data (the key).

There are two types of cryptography:

- In **shared or secret key (symmetric)** cryptography, one key is a shared secret between two communicating parties. Encryption and decryption both use the same key.
- In **public key (asymmetric)** cryptography, encryption and decryption each use different keys. A party has pair of keys consisting of a public key and a private key. The public key is freely distributed, typically within a digital certificate, while the private key is securely held by the owner. The two keys are mathematically related, but it is virtually impossible to derive the private key from the public key. An object, such as a message, that is encrypted with someone's public key can be decrypted only with the associated private key. Alternately, a server or user can use a private key to ″sign″ an object and the receiver can use the corresponding public key to decrypt the digital signature to verify the object's source and integrity.

Related concepts

"Digital signatures" on page 4
A digital signature on an electronic document or other object is created by using a form of cryptography and is equivalent to a personal signature on a written document.

"Secure Sockets Layer (SSL)" on page 10
The Secure Sockets Layer (SSL), originally created by Netscape, is the industry standard for session encryption between clients and servers.

IBM Cryptographic Coprocessors for System i

The cryptographic coprocessor provides proven cryptographic services, ensuring privacy and integrity, for developing secure e-business applications.

Using an IBM Cryptographic Coprocessor for the System i platform adds highly secure cryptographic processing capability to your system. If you have a cryptographic coprocessor installed and varied on for your system, you can use the cryptographic coprocessor to provide more secure key storage for your certificate private keys.

You can use the cryptographic coprocessor to store the private key for a server or client certificate and for a local Certificate Authority (CA) certificate. However, you cannot use the cryptographic coprocessor to store a user certificate private key because this key must be stored on the user's system. Also, you cannot use the coprocessor to store the private key for an object signing certificate at this time.

You can either store a certificate private key directly in the cryptographic coprocessor, or you can use the cryptographic coprocessor master key to encrypt the key and store it in a special key file. You can select these key storage options as part of the process of creating or renewing a certificate. Also, if you use the coprocessor to store a certificate's private key, you can change the coprocessor device assignment for that key.

To use the cryptographic coprocessor for private key storage, you must ensure that the coprocessor is varied on before using Digital Certificate Manager (DCM). Otherwise, DCM does not provide the option for selecting a storage location as part of the certificate creation or renewal process.

Related concepts

"Storing certificate keys on an IBM Cryptographic Coprocessor" on page 69
Review this information to learn how to use an installed coprocessor to provide more secure storage for your certificates' private keys.

Secure Sockets Layer (SSL)

The Secure Sockets Layer (SSL), originally created by Netscape, is the industry standard for session encryption between clients and servers.

SSL uses asymmetric, or public key, cryptography to encrypt the session between a server and client. The client and server applications negotiate this session key during an exchange of digital certificates. The key expires automatically after 24 hours, and the SSL process creates a different key for each server connection and each client. Consequently, even if unauthorized users intercept and decrypt a session key (which is unlikely), they cannot use it to eavesdrop on later sessions.

Related concepts

"Cryptography" on page 9
Shared and public keys are two different types of cryptographic functions that digital certificates use to provide security.

"Types of digital certificates" on page 28
Use this information to learn about the different types of digital certificates and how they are used in the Digital Certificate Manager (DCM).

Application definitions

Digital Certificate Manager (DCM) allows you to manage application definitions that will work with SSL configurations and object signing.

There are two types of application definitions that you can manage in Digital Certificate Manager (DCM):

- Client or server application definitions that use Secure Sockets Layer (SSL) communication sessions.
- Object signing application definitions that sign objects to ensure object integrity.

To use DCM to work with SSL application definitions and their certificates, the application must first be registered with DCM as an application definition so that it has a unique application ID. Application developers register SSL-enabled applications by using an API (QSYRGAP, QsyRegisterAppForCertUse) to create the application ID in DCM automatically. All IBM System i SSL-enabled applications are registered with DCM so that you can easily use DCM to assign a certificate to them so that they can establish an SSL session. Also, for applications that you write or purchase, you can define an application definition and create the application ID for it within DCM itself. You must be working in the *SYSTEM certificate store to create an SSL application definition for either a client application or a server application.

To use a certificate to sign objects, you first must define an application for the certificate to use. Unlike an SSL application definition, an object signing application does not describe an actual application. Instead, the application definition that you create might describe the type or group of objects that you intend to sign. You must be working in the *OBJECTSIGNING certificate store to create an object signing application definition.

Related concepts

"Managing applications in DCM" on page 63
Digital Certificate Manager (DCM) allows you to create application definitions and manage an application's certificate assignment. You can also define CA trust lists that applications use as the basis of accepting certificates for client authentication.

Related tasks

"Creating an application definition" on page 63
Review this topic to learn how about the two different types of applications that you can define and work with.

Validation

Digital Certificate Manager (DCM) provides tasks that allow you to validate a certificate or to validate an application to verify various properties that they each must have.

Certificate validation

When you validate a certificate, Digital Certificate Manager (DCM) verifies a number of items pertaining to the certificate to ensure the authenticity and validity of the certificate. Validating a certificate ensures that applications that use the certificate for secure communications or for signing objects are unlikely to encounter problems when using the certificate.

As part of the validation process, DCM checks that the selected certificate is not expired. DCM also checks that the certificate is not listed in a Certificate Revocation List (CRL) as revoked, if a CRL location exists for the CA that issued the certificate.

If you configure Lightweight Directory Access Protocol (LDAP) mapping to use a CRL, DCM checks the CRL when validating the certificate to make sure the certificate is not listed in the CRL. However, for the validation process to accurately check the CRL, the directory server (LDAP server) configured for LDAP mapping must contain the appropriate CRL. Otherwise, the certificate will not validate correctly. You must provide a binding DN and password to avoid having a certificate validate with a revoked status. Also, if you do not specify a DN and password when you configure LDAP mapping you will be binding anonymously to the LDAP server. An anonymous bind to an LDAP server does not provide the level of authority needed to access "critical" attributes, and the CRL is a "critical" attribute. In such a case, DCM may validate a certificate with a revoked status because DCM is unable to obtain the correct status from the CRL. If you want to access the LDAP server anonymously, you need to use the Directory Server Web Administration Tool and select the "Manage schema" task to change the security class (also referred to as "access class") of the **certificateRevocationList** and **authorityRevocationList** attributes from "critical" to "normal".

DCM also checks that the CA certificate for the issuing CA is in the current certificate store and that the CA certificate is marked as trusted. If the certificate has a private key (for example, server and client or object signing certificates), then DCM also validates the public-private key pair to ensure that the public-private key pair match. In other words, DCM encrypts data with the public key and then ensures that the data can be decrypted with the private key.

Application validation

When you validate an application, Digital Certificate Manager (DCM) verifies that there is a certificate assignment for the application and ensures that the assigned certificate is valid. Additionally, DCM ensures that if the application is configured to use a Certificate Authority (CA) trust list, that the trust list contains at least one CA certificate. DCM then verifies that the CA certificates in the application CA trust list are valid. Also, if the application definition specifies that Certificate Revocation List (CRL) processing occur and there is a defined CRL location for the CA, DCM checks the CRL as part of the validation process.

Validating an application can help alert you to potential problems that an application might have when it is performing a function that requires certificates. Such problems might prevent an application either from participating successfully in a Secure Sockets Layer (SSL) session or from signing objects successfully.

Related concepts

"Validating certificates and applications" on page 67
You can use Digital Certificate Manager (DCM) to validate individual certificates or the applications that use them. The list of things that DCM checks differs slightly depending on whether you are validating a certificate or an application.

DCM scenarios

These scenarios illustrate typical certificate implementation schemes to help you plan your own certificate implementation as part of your System i security policy. Each scenario also provides all needed configuration tasks you must perform to employ the scenario as described.

Using Digital Certificate Manager (DCM) and the System i platform allow you to use certificates to enhance your security policy in a number of different ways. How you choose to use certificates varies based on both your business objectives and your security needs.

Using digital certificates can help you improve your security in a number of ways. Digital certificates allow you to use the Secure Sockets Layer (SSL) for secure access to Web sites and other Internet services. You can use digital certificates to configure your virtual private network (VPN) connections. Also, you can use a certificate's key to digitally sign objects or to verify digital signatures to ensure the authenticity of objects. Such digital signatures ensure the reliability of an object's origin and protect the integrity of the object.

You can further augment system security by using digital certificates (instead of user names and passwords) to authenticate and authorize sessions between the server and users. Also, depending on how you configure DCM, you can use DCM to associate a user's certificate with his or her System i user profile or an Enterprise Identity Mapping (EIM) identifier. The certificate then has the same authorizations and permissions as the associated user profile.

Consequently, how you choose to use certificates can be complicated and depends on a variety of factors. The scenarios provided in this topic describe some of the more common digital certificate security objectives for secure communication within typical business contexts. Each scenario also describes all necessary system and software prerequisites and all the configuration tasks that you must perform to carry out the scenario.

Related information

Object signing scenarios

Scenario: Using certificates for external authentication

In this scenario, you lean when and how to use certificates as an authentication mechanism to protect and limit access by public users to public or extranet resources and applications.

Situation

You work for the MyCo, Inc insurance company and are responsible for maintaining different applications on your company's intranet and extranet sites. One particular application for which you are responsible is a rate-calculating application that allows hundreds of independent agents to generate quotes for their clients. Because the information that this application provides is somewhat sensitive, you want to make sure that only registered agents can use it. Further, you want to eventually provide a more secure method of user authentication to the application than your current user name and password method. You are concerned additionally that unauthorized users might capture this information when it is transmitted over an untrusted network. Also, you have concerns that different agents might share this information with each other without authorization to do so.

After some research, you decide that using digital certificates can provide you with the security that you need to protect the sensitive information entered into and retrieved from this application. The use of certificates allows you to use Secure Sockets Layer (SSL) to protect the transmission of the rate data.

Although eventually you want all agents to use a certificate to access the application, you know that your company and your agents may need some time before this goal can be achieved. In addition to the use of certificate client authentication, you plan to continue the current use of user name and password authentication because SSL protects the privacy of this sensitive data in transmission.

Based on the type of application and its users and your future goal of certificate authentication for all users, you decide to use a public certificate from a well known Certificate Authority (CA) to configure SSL for your application.

Scenario advantages

This scenario has the following advantages:

- Using digital certificates to configure SSL access to your rate calculation application ensures that the information transmitted between the server and client is protected and private.
- Using digital certificates whenever possible for client authentication provides a more secure method of identifying authorized users. Even where the use of digital certificates is not possible, client authentication by means of user name and password authentication is protected and kept private by the SSL session, making the exchange of such sensitive data more secure.
- Using *public* digital certificates to authenticate users to your applications and data in the manner that this scenario describes is a practical choice under these or similar conditions:
  - Your data and applications require varying degrees of security.
  - There is a high rate of turnover among your trusted users.
  - You provide public access to applications and data, such as an Internet Web site, or an extranet application.
  - You do not want to operate your own Certificate Authority (CA) based on administrative reasons, such as a large number of outside users who access your applications and resources.
- Using a public certificate to configure the rate calculating application for SSL in this scenario decreases the amount of configuration that users must perform to access the application securely. Most client software contains CA certificates for most well-known CAs.

Objectives

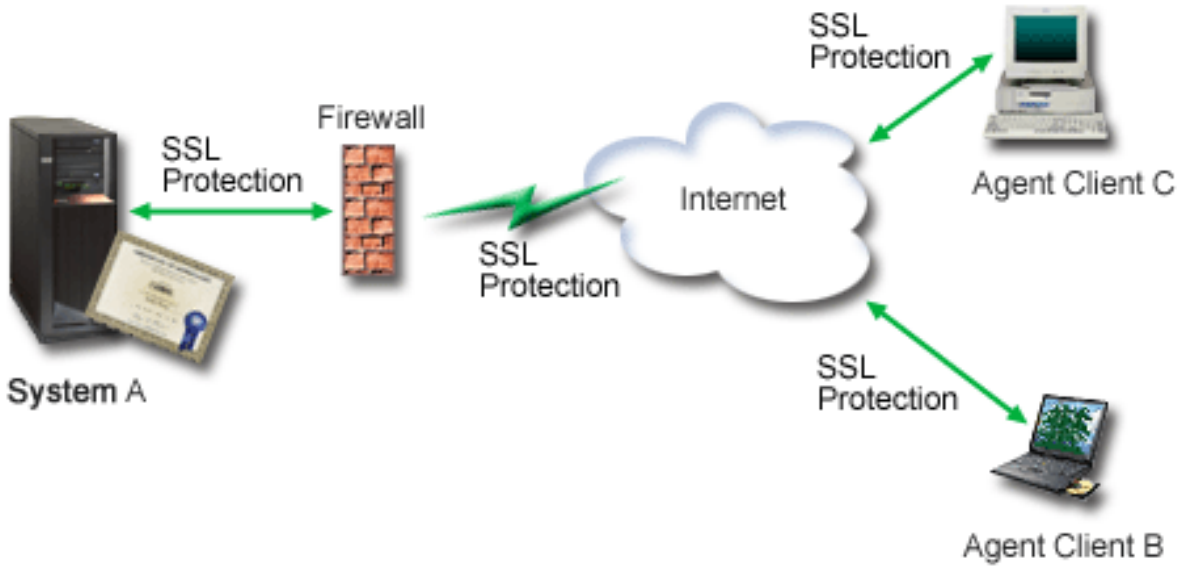
In this scenario, MyCo, Inc. wants to use digital certificates to protect the rate calculating information that their application provides to authorized public users. The company also wants a more secure method of authenticating those users who are allowed to access this application when possible.

The objectives of this scenario are as follows:

- Company public rate calculating application must use SSL to protect the privacy of the data that it provides to users and receives from users.
- SSL configuration must be accomplished with public certificates from a well-known public Internet Certificate Authority (CA).
- Authorized users must provide a valid user name and password to access the application in SSL mode. Eventually, authorized users must be able to use one of two methods of secure authentication to be granted access to the application. Agents must present either a public digital certificate from a well-known Certificate Authority (CA) or a valid user name and password if a certificate is unavailable.

Details

The following figure illustrates the network configuration in this scenario:



The figure illustrates the following information about the situation for this scenario:

Company public server – System A

- System A is the server that hosts the company's rate calculating application.
- System A runs i5/OS Version 5 Release 4 (V5R4).
- System A has Digital Certificate Manager (i5/OS option 34) and IBM HTTP Server for i5/OS (5722–DG1) installed and configured.
- System A runs the rate calculating application, which is configured such that it:
  - Requires SSL mode.
  - Uses a public certificate from a well-known Certificate Authority (CA) to authenticate itself to initialize an SSL session.
  - Requires user authentication by user name and password.
- System A presents its certificate to initiate an SSL session when Clients B and C access the rate calculating application.
- After initializing the SSL session, System A requests that Clients B and C provide a valid user name and password before allowing access to the rate calculating application.

Agent client systems – Client B and Client C

- Clients B and C are independent agents who access the rate calculating application.
- Clients B and C client software has an installed copy of the well-known CA certificate that issued the application certificate.
- Clients B and C access the rate calculating application on System A, which presents its certificate to their client software to authenticate its identity and initiate an SSL session.
- Client software on Clients B and C is configured to accept the certificate from System A for the purpose of initializing an SSL session.
- After the SSL session begins, Clients B and C must provide a valid user name and password before System A grants access to the application.

Prerequisites and assumptions

This scenario depends on the following prerequisites and assumptions:

- The rate calculating application on System A is a generic application that can be configured to use SSL. Most applications, including many System i applications, provide SSL support. SSL configuration steps vary widely among applications. Consequently, this scenario does not provide specific instructions for configuring the rate calculating application to use SSL. This scenario provides instructions for configuring and managing the certificates that are necessary for any application to use SSL.
- The rate calculating application may provide the capability of requiring certificates for client authentication. This scenario provides instructions for how to use Digital Certificate Manager (DCM) to configure certificate trust for those applications that provide this support. Because the configuration steps for client authentication vary widely among applications, this scenario does not provide specific instructions for configuring certificate client authentication for the rate calculating application.
- System A meets the "DCM set up requirements" on page 27 for installing and using Digital Certificate Manager (DCM)
- No one has previously configured or used DCM on System A.
- Whoever uses DCM to perform the tasks in this scenario must have *SECADM and *ALLOBJ special authorities for their user profile.
- System A does not have an IBM Cryptographic Coprocessor installed.

Configuration tasks

Related tasks

"Starting Digital Certificate Manager" on page 38
Use this information to learn how to access the Digital Certificate Manager (DCM) feature on your system.

Completing planning work sheets

The following planning work sheets demonstrate the information that you need to gather and the decisions you need to make to prepare the digital certificate implementation that this scenario describes. To ensure a successful implementation, you need to be able to answer `Yes` to all prerequisite items and you need to have gathered all the information requested before you perform any configuration tasks.

Table 1. Certificate implementation prerequisite planning work sheet

Prerequisite work sheet	Answers
Is your i5/OS V5R42 (5722-SS1)?	`Yes`
Is option 34 of i5/OS installed on your system?	`Yes`
Is the IBM HTTP Server for i5/OS (5722–DG1) installed on your system and Administrative server instance started?	`Yes`
Is TCP configured for your system so that you can use a Web browser and the HTTP Server Administrative server instance to access DCM?	`Yes`
Do you have *SECADM and *ALLOBJ special authorities?	`Yes`

You need to gather the following information about your digital certificate implementation to perform the necessary configuration tasks to complete the implementation:

Table 2. Certificate implementation configuration planning work sheet

Planning work sheet for System A	Answers
Will you operate your own local CA or obtain certificates for your application from a public CA?	`Obtain certificate from public CA`
Does System A host the applications that you want to enable for SSL?	`Yes`
What distinguished name information will you use for the certificate signing request (CSR) that you use DCM to create? • **Key size**: determines strength of cryptographic keys for certificate. • **Certificate label**: identifies the certificate with a unique string of characters. • **Common name**: identifies the owner of the certificate, such as a person, entity, or application; part of the Subject DN for the certificate. • **Organization unit**: identifies the organizational section or area for the application that will use this certificate. • **Organization name**: identifies your company or divisional section for the application that will use this certificate. • **Locality or city**: identifies your city or a locality designation for your organization. • **State or province**: identifies the state or province in which you will use this certificate. • **Country or region**: identifies, with a two-letter designation, the country or region in which you will use this certificate.	**Key size**: `1024`**Certificate label**: `Myco_public_cert`**Common name**: `myco_rate_server@myco.com`**Organization unit**: `Rate dept`**Organization name**: `myco`**Locality or city**: `Any_city`**State or province**: `Any`**Country or region**: `ZZ`
What is the DCM application ID for the application that you want to configure to use SSL?	`mcyo_agent_rate_app`
Will you configure the SSL-enabled application to use certificates for client authentication? If yes, which CAs do you want to add to the application's CA trust list?	`No`

Creating a server or client certificate request

1. Start DCM. Refer to Starting DCM.
2. In the navigation frame of DCM, select **Create New Certificate Store** to start the guided task and complete a series of forms. These forms guide you through the process of creating a certificate store and a certificate that your applications can use for SSL sessions.

   Note: If you have questions about how to complete a specific form in this guided task, select the question mark (**?**) at the top of the page to access the online help.
3. Select ***SYSTEM** as the certificate store to create and click **Continue**.
4. Select **Yes** to create a certificate as part of creating the *SYSTEM certificate store and click **Continue**.
5. Select **VeriSign or other Internet Certificate Authority (CA)** as the signer of the new certificate, and click **Continue** to display a form that allows you to provide identifying information for the new certificate.
6. Complete the form and click **Continue** to display a confirmation page. This confirmation page displays the certificate request data that you must provide to the public Certificate Authority (CA) that will issue your certificate. The Certificate Signing Request (CSR) data consists of the public key, distinguished name, and other information that you specified for the new certificate.

7. Carefully copy and paste the CSR data into the certificate application form, or into a separate file, that the public CA requires for requesting a certificate. You must use all the CSR data, including both the Begin and End New Certificate Request lines.

   Note: When you exit this page, the data is lost and you cannot recover it.
8. When you exit this page, the data is lost and you cannot recover it.
9. Wait for the CA to return the signed, completed certificate before you continue to the next task step for the scenario.

After the CA returns the signed completed certificate, you can configure your application to use SSL, import the certificate into the *SYSTEM certificate store, and assign it to your application to use for SSL.

Configuring applications to use SSL

When you receive your signed certificate back from the public Certificate Authority (CA), you can continue the process of enabling Secure Sockets Layer (SSL) communications for your public application. You must configure your application to use SSL before working with your signed certificate. Some applications, such as the HTTP Server forSystem i generate a unique application ID and register the ID with Digital Certificate Manager (DCM) when you configure the application to use SSL. You must know the application ID before you can use DCM to assign your signed certificate to it and complete the SSL configuration process.

How you configure your application to use SSL varies based on the application. This scenario does not assume a specific source for the rate calculating application that it describes because there are a number of ways that MyCo, Inc. might provide this application to its agents.

To configure your application to use SSL, follow the instructions that your application documentation provides. Also, you can learn more about configuring many common IBM applications to use SSL by reviewing, Secure Sockets Layer (SSL) in the i5/OS Information Center.

When you complete the SSL configuration for your application, you can configure the signed public certificate for the application so that it can initiate SSL sessions.

Importing and assigning the signed public certificate

After you configure your application to use SSL, you can use Digital Certificate Manager (DCM) to import your signed certificate and assign it to your application.

To import your certificate and assign it to your application to complete the process of configuring SSL, follow these steps:

1. Start DCM. Refer to Starting DCM.
2. In the navigation frame, click **Select a Certificate Store** and select ***SYSTEM** as the certificate store to open.
3. When the **Certificate Store and Password** page displays, provide the password that you specified for the certificate store when you created it and click **Continue**.
4. After the navigation frame refreshes, select **Manage Certificates** to display a list of tasks.
5. From the task list, select **Import certificate** to begin the process of importing the signed certificate into the *SYSTEM certificate store.

   Note: If you have questions about how to complete a specific form in this guided task, select the question mark (**?**) at the top of the page to access the online help.
6. Next, select **Assign certificate** from the **Manage Certificates** task list to display a list of certificates for the current certificate store.

7. Select your certificate from the list and click **Assign to Applications** to display a list of application definitions for the current certificate store.
8. Select your application from the list and click **Continue**. A page displays with either a confirmation message for your assignment selection or an error message if a problem occurred.

With these tasks complete, you can start your application in SSL mode and begin protecting the privacy of the data that it provides.

Starting applications in SSL mode

After you complete the process of importing and assigning the certificate to your application, you may need to end and restart your application in SSL mode. This is necessary in some cases because the application may not be able to determine that the certificate assignment exists while the application is running. Review the documentation for your application to determine whether you need to restart the application or for other specific information about starting the application in SSL mode.

If you want to use certificates for client authentication, you can now define a CA trust list for the application.

(Optional): Defining a CA trust list for an application that requires

Applications that support the use of certificates for client authentication during a Secure Sockets Layer (SSL) session must determine whether to accept a certificate as valid proof of identity. One of the criteria that an application uses for authenticating a certificate is whether the application trusts the Certificate Authority (CA) that issued the certificate.

The situation that this scenario describes does not require that the rate calculating application use certificates for client authentication, but that the application be able to accept certificates for authentication when they are available. Many applications provide client authentication certificate support; how you configure this support varies widely among applications. This optional task is provided to help you understand how to use DCM to enable certificate trust for client authentication as a foundation for configuring your applications to use certificates for client authentication.

Before you can define a CA trust list for an application, several conditions must be met:

- The application must support the use of certificates for client authentication.
- The DCM definition for the application must specify that the application use a CA trust list.

If the definition for an application specifies that the application use a CA trust list, you must define the list before the application can perform certificate client authentication successfully. This ensures that the application can validate only those certificates from CAs that you specify as trusted. If users or a client application present a certificate from a CA that is not specified as trusted in the CA trust list, the application will not accept it as a basis for valid authentication.

To use DCM to define a CA trust list for your application, complete these steps:

1. Start DCM. Refer to Starting DCM.
2. In the navigation frame, click **Select a Certificate Store** and select ***SYSTEM** as the certificate store to open.
3. When the **Certificate Store and Password** page displays, provide the password that you specified for the certificate store when you created it and click **Continue**.
4. After the navigation frame refreshes, select **Manage Certificates** to display a list of tasks.
5. From the task list, select **Set CA status** to display a list of CA certificates.

   Note: If you have questions about how to complete a specific form in this guided task, select the question mark (**?**) at the top of the page to access the online help.

6. Select one or more CA certificates from the list that your application will trust and click **Enable** to display a list of applications that use a CA trust list.
7. Select the application from the list that needs to add the selected CA to its trust list and click **OK**. A message displays at the top of the page to indicate that the applications you selected will trust the CA and the certificates that it issues.

You can now configure your application to require certificates for client authentication. Follow the instructions provided by the documentation for your application.

Scenario: Using certificates for internal authentication

In this scenario, you to learn how to use certificates as an authentication mechanism to protect and restrict which resources and applications that internal users can access on your internal servers.

Situation

You are the network administrator for a company (MyCo, Inc.) whose human resource department is concerned with such issues as legal matters and privacy of records. Company employees have requested that they be able to access their personal benefits and health care information online. The company has responded to this request by creating an internal Web site to provide this information to employees. You are responsible for administering this internal Web site, which runs on the IBM HTTP Server for i5/OS (powered by Apache).

Because employees are located in two geographically separate offices and some employees travel frequently, you are concerned about keeping this information private as it travels across the Internet. Also, you traditionally authenticate users by means of a user name and password to limit access to company data. Because of the sensitive and private nature of this data, you realize that limiting access to it based on password authentication may not be sufficient. After all, people can share, forget, and even steal passwords.

After some research, you decide that using digital certificates can provide you with the security that you need. Using certificates allows you to use Secure Sockets Layer (SSL) to protect the transmission of the data. Additionally, you can use certificates instead of passwords to more securely authenticate users and limit the human resource information that they can access.

Therefore, you decide to set up a private local Certificate Authority (CA) and issue certificates to all employees and have the employees associate their certificates with their System i user profiles. This type of private certificate implementation allows you to more tightly control access to sensitive data, as well as control the privacy of the data by using SSL. Ultimately, by issuing certificates yourself, you have increased the probability that your data remains secure and is accessible only to specific individuals.

Scenario advantages

This scenario has the following advantages:

- Using digital certificates to configure SSL access to your human resource Web server ensures that the information transmitted between the server and client is protected and private.
- Using digital certificates for client authentication provides a more secure method of identifying authorized users.
- Using *private* digital certificates to authenticate users to your applications and data is a practical choice under these or similar conditions:
  - You require a high degree of security, especially in regards to authenticating users.
  - You trust the individuals to whom you issue certificates.
  - Your users already have System i user profiles for controlling their access to applications and data.
  - You want to operate your own Certificate Authority (CA).

- Using private certificates for client authentication allows you to more easily associate the certificate with the authorized user's System i user profile. This association of certificate with a user profile allows the HTTP Server to determine the certificate owner's user profile during authentication. The HTTP Server can then swap to it and run under that user profile or perform actions for that user based on information in the user profile.

Objectives

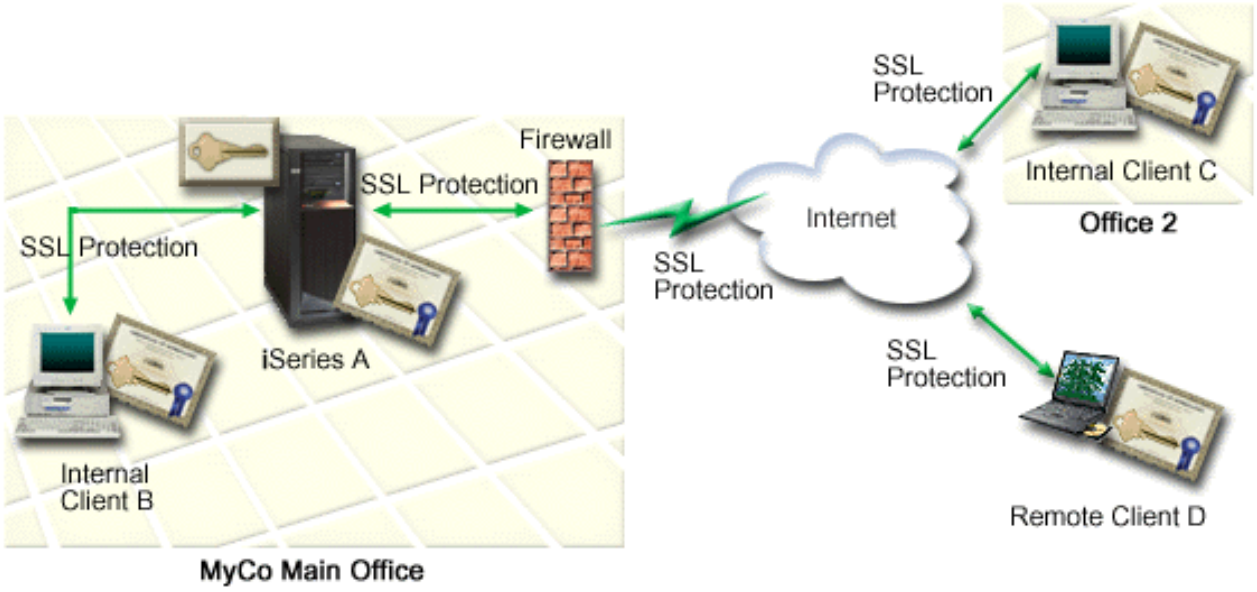
In this scenario, MyCo, Inc. wants to use digital certificates to protect the sensitive personal information that their internal human resources Web site provides to company employees. The company also wants a more secure method of authenticating those users who are allowed to access this Web site.

The objectives of this scenario are as follows:

- Company internal human resources Web site must use SSL to protect the privacy of the data that it provides to users.
- SSL configuration must be accomplished with private certificates from an internal local Certificate Authority (CA).
- Authorized users must provide a valid certificate to access the human resources Web site in SSL mode.

Details

The following figure illustrates the network configuration for this scenario:



The figure illustrates the following information about the situation for this scenario:

Company public server – System A

- System A is the server that hosts the company's rate calculating application.
- System A runs i5/OS Version 5 Release 4 (V5R4).
- System A has Digital Certificate Manager (i5/OS option 34) and IBM HTTP Server for i5/OS (5722–DG1) installed and configured.
- System A runs the rate calculating application, which is configured such that it:
  - Requires SSL mode.

  - Uses a public certificate from a well-known Certificate Authority (CA) to authenticate itself to initialize an SSL session.
  - Requires user authentication by user name and password.
- System A presents its certificate to initiate an SSL session when Clients B and C access the rate calculating application.
- After initializing the SSL session, System A requests that Clients B and C provide a valid user name and password before allowing access to the rate calculating application.

Agent client systems – Client B and Client C

- Clients B and C are independent agents who access the rate calculating application.
- Clients B and C client software has an installed copy of the well-known CA certificate that issued the application certificate.
- Clients B and C access the rate calculating application on System A, which presents its certificate to their client software to authenticate its identity and initiate an SSL session.
- Client software on Clients B and C is configured to accept the certificate from System A for the purpose of initializing an SSL session.
- After the SSL session begins, Clients B and C must provide a valid user name and password before System A grants access to the application.

Prerequisites and assumptions

This scenario depends on the following prerequisites and assumptions:

- TheIBM HTTP Server for i5/OS (powered by Apache) runs the human resource application on System A. This scenario does not provide specific instructions for configuring the HTTP Server to use SSL. This scenario provides instructions for configuring and managing the certificates that are necessary for any application to use SSL.
- The HTTP Server provides the capability of requiring certificates for client authentication. This scenario provides instructions for using Digital Certificate Manager (DCM) to configure the certificate management requirements for this scenario. However, this scenario does not provide the specific configuration steps for configuring certificate client authentication for the HTTP Server.
- The human resources HTTP Server on System A already uses password authentication.
- System A meets the requirements for installing and using Digital Certificate Manager (DCM).
- No one has previously configured or used DCM on System A.
- Whoever uses DCM to perform the tasks in this scenario must have *SECADM and *ALLOBJ special authorities for their user profile.
- System A does not have an IBM Cryptographic Coprocessor installed.

Configuration tasks

Completing planning work sheets

The following planning work sheets demonstrate the information that you need to gather and the decisions you need to make to prepare the digital certificate implementation that this scenario describes. To ensure a successful implementation, you need to be able to answer `Yes` to all prerequisite items and you need to have gathered all the information requested before you perform any configuration tasks.

Table 3. Certificate implementation prerequisite planning work sheet

Prerequisite work sheet	Answers
Is youri5/OS V5R4 (5722-SS1)?	`Yes`
Is option 34 of i5/OS installed on your system?	`Yes`

Table 3. Certificate implementation prerequisite planning work sheet (continued)

Prerequisite work sheet	Answers
Is the IBM HTTP Server for i5/OS (5722–DG1) installed on your system and Administrative server instance started?	Yes
Is TCP configured for your system so that you can use a Web browser and the HTTP Server Administrative server instance to access DCM?	Yes
Do you have *SECADM and *ALLOBJ special authorities?	Yes

You need to gather the following information about your digital certificate implementation to perform the necessary configuration tasks to complete the implementation:

Table 4. Certificate implementation configuration planning work sheet

Planning work sheet for System A	Answers
Will you operate your own local CA or obtain certificates for your application from a public CA?	Create local CA to issue certificates
Does System A host the applications that you want to enable for SSL?	Yes
What distinguished name information will you use for the local CA? • **Key size**: determines strength of cryptographic keys for certificate. • **Certificate Authority (CA) name**: identifies the CA and becomes the common name for the CA certificate and the Issuer DN for certificates that the CA issues. • **Organization unit**: identifies the organizational section or area for the application that will use this certificate. • **Organization name**: identifies your company or divisional section for the application that will use this certificate. • **Locality or city**: identifies your city or a locality designation for your organization. • **State or province**: identifies the state or province in which you will use this certificate. • **Country or region**: identifies, with a two-letter designation, the country or region in which you will use this certificate. • **Validity period of Certificate Authority**: specifies the number of days for which the Certificate Authority certificate is valid	**Key size**: 1024**Certificate Authority (CA) name**: Myco_CA@myco.com**Organization unit**: Rate dept**Organization name**: myco**Locality or city**: Any_city**State or province**: Any**Country or region**: ZZ**Validity period of Certificate Authority**: 1095
Do you want to set the policy data for the local CA to allow it to issue user certificates for client authentication?	Yes

Table 4. Certificate implementation configuration planning work sheet (continued)

Planning work sheet for System A	Answers
What distinguished name information will you use for the server certificate that the local CA issues? • **Key size**: determines strength of cryptographic keys for certificate. • **Certificate label**: identifies the certificate with a unique string of characters. • **Common name**: identifies the owner of the certificate, such as a person, entity, or application; part of the Subject DN for the certificate. • **Organization unit**: identifies the organizational section or area for the application that will use this certificate. • **Organization name**: identifies your company or divisional section for the application that will use this certificate. • **Locality or city**: identifies your city or a locality designation for your organization. • **State or province**: identifies the state or province in which you will use this certificate. • **Country or region**: identifies, with a two-letter designation, the country or region in which you will use this certificate.	**Key size**: 1024**Certificate label**: Myco_public_cert**Common name**: myco_rate_server@myco.com**Organization unit**: Rate dept**Organization name**: myco**Locality or city**: Any_city**State or province**: Any**Country or region**: ZZ
What is the DCM application ID for the application that you want to configure to use SSL?	mcyo_agent_rate_app
Will you configure the SSL-enabled application to use certificates for client authentication? If yes, which CAs do you want to add to the application's CA trust list?	YesMyco_CA@myco.com

Configuring the human resources HTTP Server to use SSL

Secure Sockets Layer (SSL) configuration for the human resources HTTP Server (powered by Apache) on System A involves a number of tasks which vary depending on how your server is configured currently.

To configure the server to use SSL, follow these steps:

1. Start the HTTP Server Administration interface.
2. To work with a specific HTTP server, select these page tabs **Manage → All Servers → All HTTP Servers** to view a list of all configured HTTP servers.
3. Select the appropriate server from the list and click **Manage Details**.
4. In the navigation frame, select **Security**.
5. Select the **SSL with Certificate Authentication** tab in the form.
6. In the **SSL** field, select Enabled.
7. In the **Server certificate application name** field, specify an application ID by which this server instance is known. Or, you can select one from the list. This application ID is in the form QIBM_HTTP_SERVER_[server_name], for example, QIBM_HTTP_SERVER_MYCOTEST. **Note:** Remember this application ID. You will need to select it again in the DCM.

When you complete the configuration for the HTTP Server to use SSL, you can use DCM to configure the certificate support that you need for SSL and client authentication.

Related information

IBM HTTP Server for i5/OS

Creating and operating a local CA

After you configure the human resources HTTP Server to use Secure Sockets Layer (SSL), you must configure a certificate for the server to use to initiate SSL. Based on the objectives for this scenario, you have chosen to create and operate a local Certificate Authority (CA) to issue a certificate to the server.

When you use Digital Certificate Manager (DCM) to create a local CA, you are guided through a process that ensures that you configure everything that you need to enable SSL for your application. This includes assigning the certificate that the local CA issues to your Web server application. Also, you add the local CA to the Web server application's CA trust list. Having the local CA in the application's trust list ensures that the application can recognize and authenticate users that present certificates that the local CA issues.

To use Digital Certificate Manager (DCM) to create and operate a local CA and issue a certificate to your human resources server application, complete these steps:

1. Start DCM. Refer to Starting DCM.
2. In the navigation frame of DCM, select **Create a Certificate Authority (CA)** to display a series of forms. These forms guide you through the process of creating a local CA and completing other tasks needed to begin using digital certificates for SSL, object signing, and signature verification.

   Note: If you have questions about how to complete a specific form in this guided task, select the question mark (**?**) button at the top of the page to access the online help.
3. Complete the forms for this guided task. In using these forms to perform all the tasks that you need to set up a working local Certificate Authority (CA), you perform the following steps:
   a. Provide identifying information for the local CA.
   b. Install the local CA certificate on your PC or in your browser so that your software can recognize the local CA and validate certificates that the local CA issues.
   c. Choose the policy data for your local CA.

      Note: Be sure to select that the local CA can issue user certificates.
   d. Use the new local CA to issue a server or client certificate that your applications can use for SSL connections.
   e. Select the applications that can use the server or client certificate for SSL connections.

      Note: Be sure to select the application ID for your human resources HTTP Server.
   f. Use the new local CA to issue an object signing certificate that applications can use to digitally sign objects. This subtask creates the *OBJECTSIGNING certificate store; this is the certificate store that you use to manage object signing certificates.

      Note: Although this scenario does not use object signing certificates, be sure to complete this step. If you cancel at this point in the task, the task ends and you must perform separate tasks to complete your SSL certificate configuration.
   g. Select the applications that will trust the local CA.

      Note: Be sure to select the application ID for your human resources HTTP Server, for example, `QIBM_HTTP_SERVER_MYCOTEST`, as one of the applications that trusts the local CA.

When you complete the certificate configuration that your Web server application requires to use SSL, you can configure the Web server to require certificates for user authentication.

Configuring client authentication for human resources Web server

You must configure the general authentication settings for the HTTP Server when you specify that the HTTP Server require certificates for authentication. You configure these settings in the same security form that you used to configure the server to use Secure Sockets Layer (SSL).

To configure the server to require certificates for client authentication, follow these steps:

1. Start the HTTP Server Administration interface.
2. Using your browser, go to the i5/OS Tasks page on your system at http://your_system_name:2001.
3. Select **IBM Web Administration for i5/OS**.
4. To work with a specific HTTP server, select these page tabs **Manage** → **All Servers** → **All HTTP Servers** to view a list of all configured HTTP servers.
5. Select the appropriate server from the list and click **Manage Details**.
6. In the navigation frame, select **Security**.
7. Select the **Authentication** tab in the form.
8. Select **Use i5/OS profile of client**.
9. In the **Authentication name or realm** field, specify a name for the authorization realm.
10. Select `Enabled` for the **Process requests using client's authority** field and click **Apply**.
11. Select the **Control Access** tab in the form.
12. Select **All authenticated users (valid user name and password)** and click **Apply**.
13. Select the **SSL with Certificate Authentication** tab in the form.
14. Ensure that `Enabled` is the selected value in the **SSL** field.
15. In the **Server certificate application name** field, ensure that the correct value is specified, for example, `QIBM_HTTP_SERVER_MYCOTEST`.
16. Select **Accept client certificate if available before making connection**. Click **OK**.

When you complete the client authentication configuration, you can restart the HTTP server in SSL mode and begin protecting the privacy of the data of the human resources application.

Related information

IBM HTTP Server for i5/OS

Starting the human resources Web server in SSL mode

You may need to stop and restart your HTTP Server to ensure that the server is able to determine that the certificate assignment exists and use it to initiate SSL sessions.

To stop and start the HTTP Server (powered by Apache) follow these steps:

1. In iSeries™ Navigator expand your **system** → **Network** → **Servers** → **TCP/IP** → **HTTP Administration**
2. Click **Start** to start the HTTP Server Administration interface.
3. Click the **Manage** tab to view a list of all configured HTTP servers.
4. Select the appropriate server from the list and click **Stop** if the server is running.
5. Click **Start** to restart the server. Refer to the online help for more information about startup parameters.

Before users can access the human resources Web application, they first must install a copy of the local CA certificate in their browser software.

Related information

HTTP Server Information Center Overview

Installing a copy of the local CA certificate in a browser

When users access a server that provides a Secure Sockets Layer (SSL) connection, the server presents a certificate to the user's client software as proof of its identity. The client software must then validate the server's certificate before the server can establish the session. To validate the server certificate, the client software must have access to a locally stored copy of the certificate for the Certificate Authority (CA) that issued the server certificate. If the server presents a certificate from a public Internet CA, the user's browser or other client software must already have a copy of the CA certificate. If, as in this scenario, the server presents a certificate from a private local CA, each user must use Digital Certificate Manager (DCM) to install a copy of the local CA certificate.

Each user (Clients B, C, and D) must complete these steps to obtain a copy of a local CA certificate:

1. Start DCM. Refer to Starting DCM.
2. In the navigation frame, select **Install local CA Certificate on Your PC** to display a page that allows you to download the local CA certificate into your browser or to store it in a file on your system.
3. Select the option to install the certificate. This option downloads the local CA certificate as a trusted root in your browser. This ensures that your browser can establish secure communications sessions with Web servers that use a certificate from this CA. Your browser will display a series of windows to help you complete the installation.
4. Click **OK** to return to the Digital Certificate Manager home page.

Now that users can access the human resources Web server in SSL mode, these users must be able to present an appropriate certificate to authenticate to the server. Consequently, they must obtain a user certificate from the local CA.

Requesting a certificate from the local CA

In earlier steps, you configured the human resources Web server to require certificates for user authentication. Now users must present a valid certificate from the local CA before they are allowed to access the Web server. Each user must use Digital Certificate Manager (DCM) to obtain a certificate by using the **Create Certificate** task. In order to obtain a certificate from the local CA, the local CA policy must allow the CA to issue user certificates.

Each user (Clients B, C, and D) must complete these steps to obtain a certificate:

1. Start DCM. Refer to Starting DCM.
2. In the navigation frame, select **Create Certificate**.
3. Select **User certificate** as the type of certificate to create. A form displays so that you can provide identifying information for the certificate.
4. Complete the form and click **Continue**.

   Note: If you have questions about how to complete a specific form in this guided task, select the question mark (**?**) at the top of the page to access the online help.
5. At this point, DCM works with your browser to create the private and public key for the certificate. Your browser may display windows to guide you through this process. Follow the browser's instructions for these tasks. After the browser generates the keys, a confirmation page displays to indicate that DCM created the certificate.
6. Install the new certificate in your browser software. Your browser may display windows to guide you through this process. Follow the instructions that the browser gives to complete this task.
7. Click **OK** to finish the task.

During processing, the Digital Certificate Manager automatically associates the certificate with your System i user profile.

With these tasks complete, only authorized users with a valid certificate can access data from the human resources Web server and that data is protected during transmission by SSL.

Planning for DCM

To use Digital Certificate Manager (DCM) to effectively manage your company's digital certificates, you must have an overall plan for how you will use digital certificates as part of your security policy.

To learn more about how to plan for using DCM and to better understand how digital certificates can fit into your security policy, review these topics:

DCM set up requirements

Review this topic to make sure you have the required options installed to run Digital Certificate Manager (DCM).

DCM is a free System i feature that allows you to centrally manage digital certificates for your applications. To use DCM successfully, ensure that you do the following:

- Install option 34 of i5/OS. This is the browser-based DCM feature.
- Install the IBM HTTP Server for i5/OS (5722–DG1) and start the Administrative server instance.
- Ensure that TCP is configured for your system so that you can use a Web browser and the HTTP Server Administrative server instance to access DCM.

Note: You will not be able to create certificates unless you install all the required products. If a required product is not installed, DCM displays an error message instructing you to install the missing component.

Backup and recovery considerations for DCM data

Use this information to learn how to ensure that important DCM data is added to your backup and recovery plan for your system.

The encrypted key database passwords that you use to access certificate stores in Digital Certificate Manager (DCM) are stored, or *stashed*, in a special security file on your system. When you use DCM to create a certificate store on your system, DCM automatically stashes the password for you. However, you need to manually ensure that DCM stashes certificate store passwords under certain circumstances.

An example of one such circumstance is when you use DCM to create a certificate for another System i model and you choose to use the certificate files on the target system to create a new certificate store. In this situation, you need to open the newly created certificate store and use the **Changepassword** task to change the password for the certificate store on the target system, which ensures that DCM stashes the new password. If the certificate store is an Other System Certificate Store, you should also specify that you want to use the **Auto login** option when you change the password.

Additionally, you must specify the **Auto login** option whenever you change or reset the password for an Other System Certificate Store.

To ensure that you have a complete backup of critical DCM data, you must do the following:

- Use the save (SAV) command to save all `.KDB` and `.RDB` files. Every DCM certificate store is comprised of two files, one with a `.KDB` extension and one with a `.RDB` extension.
- Use the save system (SAVSYS) command and the save security data (SAVSECDTA) command to save the special security file that contains the key database passwords for certificate store access. To restore the DCM password security file, use the restore user profiles (RSTUSRPRF) command and specify `*ALL` for the user profile (USRPRF) option.

Another recovery consideration concerns the use of the SAVSECDTA operation and the potential for the current certificate store passwords to become out of sync with the passwords in the saved DCM password security file. If you change the password for a certificate store after you do a SAVSECDTA operation, but before you restore the data from that operation, the current certificate store password will be out of sync with the one in the restored file.

To avoid this situation, you must use the **Change password** task (under **Manage Certificate Store** in the navigation frame) in DCM to change certificate store passwords after you restore the data from a SAVSECDTA operation to ensure that you get the passwords back in sync. However, in this situation do not use the **Reset Password** button that displays when you select a certificate store to open. When you attempt to reset the password, DCM tries to retrieve the stashed password. If the stashed password is out of sync with the current password, the reset operation will fail. If you do not change certificate store passwords often, you may want to consider doing a SAVSECDTA every time you change these passwords to ensure that you always have the most current stashed version of the passwords saved in case you ever need to restore this data.

Restoring DCM information when migrating to another system

If you are migrating all users from system A to system B, and you have already set up your Digital Certificate Manager (DCM) information on system B, follow these steps to preserve the DCM information on system B:

1. Save the user profiles and authorities by using the Save Security Data (SAVSECDTA) command on both system A and system B.
2. Restore the user profiles from system A to system B by using RSTUSRPRF USRPRF(*ALL).
3. Restore the DCM information from system B to system B by using RSTUSRPRF USRPRF(*NONE) SECDTA(*DCM).

The first Restore User Profiles (RSTUSRPRF) command replaces the DCM data from the VFYOBJ.KDB file and the stash password index on system B with the data from system A. The second RSTUSRPRF command restores the DCM data that was originally on system B back to its original state.

Related tasks

"Using a local CA to issue certificates for other System i models" on page 55
Review this information to learn how to use a private local CA on one system to issue certificates for use on other System i models.

Types of digital certificates

Use this information to learn about the different types of digital certificates and how they are used in the Digital Certificate Manager (DCM).

You can use DCM to manage the following types of certificates:

Certificate Authority (CA) certificates
A Certificate Authority certificate is a digital credential that validates the identity of the Certificate Authority (CA) that owns the certificate. The Certificate Authority's certificate contains identifying information about the Certificate Authority, as well as its public key. Others can use the CA certificate's public key to verify the authenticity of the certificates that the CA issues and signs. A Certificate Authority certificate can be signed by another CA, such as VeriSign, or can be self-signed if it is an independent entity. The local CA that you create and operate with Digital Certificate Manager is an independent entity. Others can use the CA certificate's public key to verify the authenticity of the certificates that the CA issues and signs. To use a certificate for SSL, signing objects, or verifying object signatures, you must also have a copy of the issuing CAs certificate.

Server or client certificates
A server or client certificate is a digital credential that identifies the server or client application

that uses the certificate for secure communications. Server or client certificates contain identifying information about the organization that owns the application, such as the system's distinguished name. The certificate also contains the system's public key. A server must have a digital certificate to use the Secure Sockets Layer (SSL) for secure communications. Applications that support digital certificates can examine a server's certificate to verify the identity of the server when the client accesses the server. The application can then use the authentication of the certificate as the basis for initiating an SSL-encrypted session between the client and the server. You can manage these types of certificates from the *SYSTEM certificate store only.

Object signing certificates
An object signing certificate is a certificate that you use to digitally ″sign″ an object. By signing the object, you provide a means by which you can verify both the object's integrity and the origination or ownership of the object. You can use the certificate to sign a variety of objects, including most objects in the Integrated File System and *CMD objects. You can find a complete list of signable objects in the Object signing and signature verification topic. When you use an object signing certificate's private key to sign an object, the receiver of the object must have access to a copy of the corresponding signature verification certificate in order to properly authenticate the object signature. You can manage these types of certificates from the *OBJECTSIGNING certificate store only.

Signature verification certificates
A signature verification certificate is a copy of an object signing certificate without that certificate's private key. You use the signature verification certificate's public key to authenticate the digital signature created with an object signing certificate. Verifying the signature allows you to determine the origin of the object and whether it has been altered since it was signed. You can manage these types of certificates from the *SIGNATUREVERIFICATION certificate store only.

User certificates
A user certificate is a digital credential that validates the identity of the client or user that owns the certificate. Many applications now provide support that allows you to use certificates to authenticate users to resources instead of user names and passwords. Digital Certificate Manager (DCM) automatically associates user certificates that your private CA issues with the user's System i user profile. You can also use DCM to associate user certificates that other Certificate Authorities issue with the user's System i user profile.

When you use Digital Certificate Manager (DCM) to manage your certificates, DCM organizes and stores them and their associated private keys in a certificate store based on these classifications .

Note: If you have an IBM Cryptographic Coprocessor installed on your system, you can choose other private key storage options for your certificates (with the exception of object signing certificates). You can elect to store the private key on the cryptographic coprocessor itself. Or, you can use the cryptographic coprocessor to encrypt the private key and store it in a special key file instead of in a certificate store. User certificates and their private keys, however, are stored on the user's system, either in browser software or in a file for use by other client software packages.

Related concepts

"Secure Sockets Layer (SSL)" on page 10
The Secure Sockets Layer (SSL), originally created by Netscape, is the industry standard for session encryption between clients and servers.

"Certificate stores" on page 7
A certificate store is a special key database file that Digital Certificate Manager (DCM) uses to store digital certificates.

Public certificates versus private certificates

Review this information to learn how to determine which type of certificate (public or private) best suits your business needs.

You can use certificates from a public CA or you can create and operate a private CA to issue certificates. How you choose to obtain your certificates depends on how you plan to use them. Once you decide on the type of CA to issue the certificates, you need to choose the type of certificate implementation that best suits your security needs. The choices that you have for obtaining your certificates include:

- Purchasing your certificates from a public Internet Certificate Authority (CA).
- Operating your own local CA to issue private certificates for your users and applications.
- Using a combination of certificates from public Internet CAs and your own local CA.

Which of these implementation choices you make depends on a number of factors, one of the most important being the environment in which the certificates are used. Here's some information to help you better determine which implementation choice is right for your business and security needs.

Using public certificates

Public Internet CAs issue certificates to anyone who pays the necessary fee. However, an Internet CA still requires some proof of identity before it issues a certificate. This level of proof varies, though, depending on the identification policy of the CA. You need to evaluate whether the stringency of the identification policy of the CA suits your security needs before deciding to obtain certificates from the CA or to trust the certificates that it issues. As Public Key Infrastructure for X.509 (PKIX) standards have evolved, some public CAs now provide much more stringent identification standards for issuing certificates. While the process for obtaining certificates from such PKIX CAs is more involved, the certificates the CA issues provide better assurance for securing access to applications by specific users. Digital Certificate Manager (DCM) allows you to use and manage certificates from PKIX CAs that use these new certificate standards.

You must also consider the cost associated with using a public CA to issue certificates. If you need certificates for a limited number of server or client applications and users, cost may not be an important factor for you. However, cost can be particularly important if you have a large number of *private* users that need public certificates for client authentication. In this case, you need to also consider the administrative and programming effort needed to configure server applications to accept only a specific subset of certificates that a public CA issues.

Using certificates from a public CA may save you time and resources because many server, client, and user applications are configured to recognize most of the well-known public CAs. Also, other companies and users may recognize and trust certificates that a well-known public CA issues more than those that your private local CA issues.

Using private certificates

If you create your own local CA, you can issue certificates to systems and users within a more limited scope, such as within your company or organization. Creating and maintaining your own local CA allows you to issue certificates only to those users who are trusted members of your group. This provides better security because you can control who has certificates, and therefore who has access to your resources, more stringently. A potential disadvantage of maintaining your own local CA is the amount of time and resources that you must invest. However, Digital Certificate Manager (DCM) makes this process easier for you.

When you use a local CA to issue certificates to users for client authentication, you need to decide where you want to store the user certificates. When users obtain their certificates from the local CA through DCM their certificates are stored with a user profile by default. However, you can configure DCM to work with Enterprise Identity Mapping (EIM) so that their certificates are stored in a Lightweight Directory Access Protocol (LDAP) location instead. If you prefer not to have user certificates associated or stored with a user profile in any manner, you can use APIs to programmatically issue certificates to users other than System i users.

Note: No matter which CA you use to issue your certificates, the system administrator controls which CAs will be trusted by applications on his system. If a copy of a certificate for a well-known CA can be found in your browser, your browser can be set to trust server certificates that were issued by that CA. Administrators set trust for CA certificates in the appropriate DCM certificate store, which contains copies of most well-known public CA certificates. However, if a CA certificate is not in your certificate store, your server cannot trust user or client certificates that were issued by that CA until you obtain and import a copy of the CA certificate. The CA certificate must be in the correct file format and you must add that certificate to your DCM certificate store.

You may find it helpful to review some common certificate usage scenarios to help you choose whether using public or private certificates best suits your business and security needs.

Related tasks

After you decide how you want to use certificates and which type to use, review these procedures to learn more about how to use Digital Certificate Manager to put your plan into action:

- Creating and operating a private CA describes the tasks that you must perform if you choose to operate a local CA to issue private certificates.
- Managing certificates from a public Internet CA describes the tasks that you must perform to use certificates from a well-known public CA, including a PKIX CA.
- Using a local CA on other System i models describes the tasks that you must perform if you want to use certificates from a private local CA on more than one system.

Related concepts

"Managing certificates from a public Internet CA" on page 47
Review this information to learn how to manage certificates from a public Internet CA by creating a certificate store.

"Public certificates versus private certificates" on page 29
Review this information to learn how to determine which type of certificate (public or private) best suits your business needs.

"Setting up certificates for the first time" on page 38
Use this information to learn how to get started managing certificates from a public Internet Certificate Authority (CA) or how to create and operate a private local CA to issue certificates.

"Digital certificates for signing objects" on page 36
Use this information to learn how to use certificates to ensure an object's integrity or to verify the digital signature on an object to verify its authenticity.

Related tasks

"Digital certificates and Enterprise Identity Mapping (EIM)" on page 34
Using Enterprise Identity Mapping (EIM) and Digital Certificate Mangers (DCM) together allows you to apply a certificate as the source of an EIM mapping lookup operation to map from the certificate to a target user identity associated with the same EIM identifier.

"Creating a user certificate" on page 42
Review this information to learn how your users can use the local CA to issue a certificate for client authentication.

"Creating and operating a local CA" on page 39
This information explains how to create and operate a local Certificate Authority (CA) to issue private certificates for your applications.

"Using a local CA to issue certificates for other System i models" on page 55
Review this information to learn how to use a private local CA on one system to issue certificates for use on other System i models.

Related reference

"Using APIs to programmatically issue certificates to users other than System i users" on page 45
Use this information to learn how you can use your local CA to issue private certificates to users without associating the certificate with a System i user profile.

Digital certificates for SSL secure communications

Use this information to learn how to use certificates so that your applications can establish secure communication sessions.

You can use digital certificates to configure applications to use the Secure Sockets Layer (SSL) for secure communications sessions. To establish an SSL session, your server always provides a copy of its certificate for validation by the client that requests a connection. Using an SSL connection:

- Assures the client or end-user that your site is authentic.
- Provides an encrypted communications session to ensure that data that passes over the connection remains private.

The server and client applications work together as follows to ensure data security:

1. The server application presents the certificate to the client (user) application as proof of the server's identity.
2. The client application verifies the server's identity against a copy of the issuing Certificate Authority (CA) certificate. (The client application must have access to a locally stored copy of the relevant CA certificate.)
3. The server and client applications agree on a symmetric key for encryption and use it to encrypt the communications session.
4. Optionally, the server now can require the client to provide proof of identify before allowing access to the requested resources. To use certificates as proof of identity, the communicating applications must support using certificates for user authentication.

SSL uses asymmetric key (public key) algorithms during SSL initial processing to negotiate a symmetric key that is subsequently used for encrypting and decrypting the application's data for that particular SSL session. This means that your server and the client use different session keys, which automatically expire after a set amount of time, for each connection. In the unlikely event that someone intercepts and decrypts a particular session key, that session key cannot be used to deduce any future keys.

Related concepts

"Digital certificates for user authentication"
Review this information to learn how to use certificates to provide a means of more strongly authenticating users who access System i resources.

Digital certificates for user authentication

Review this information to learn how to use certificates to provide a means of more strongly authenticating users who access System i resources.

Traditionally, users receive access to resources from an application or system based on their user name and password. You can further augment system security by using digital certificates (instead of user names and passwords) to authenticate and authorize sessions between many server applications and users. Also, you can use Digital Certificate Manager (DCM) to associate a user's certificate with that user's System i user profile or another user identity. The certificate then has the same authorizations and permissions as the associated user identity or user profile. Alternatively, you can use APIs to programmatically use your private local Certificate Authority (CA) to issue certificates to users other than System i users. These APIs provide you with the ability to issue private certificates to users when you do not want these users to have a System i user profile or other internal user identity.

A digital certificate acts as an electronic credential and verifies that the person presenting it is truly who she claims to be. In this respect, a certificate is similar to a passport. Both establish an individual's

identity, contain a unique number for identification purposes, and have a recognizable issuing authority that verifies the credential as authentic. In the case of a certificate, a CA functions as the trusted, third party that issues the certificate and verifies it as an authentic credential.

For authentication purposes, certificates make use of a public key and a related private key. The issuing CA binds these keys, along with other information about the certificate owner, to the certificate itself for identification purposes.

An increasing number of applications now provide support for using certificates for client authentication during an SSL session. Currently, these System i applications provide client authentication certificate support:

- Telnet server
- IBM HTTP Server for i5/OS (powered by Apache)
- IBM Directory Server
- iSeries Access for Windows® (including iSeries Navigator Navigator)
- FTP server

Over time, additional applications may provide client authentication certificate support; review the documentation for specific applications to determine whether they provide this support.

Certificates can provide a stronger means of authenticating users for several reasons:

- There is the possibility that an individual might forget his or her password. Therefore, users must memorize or record their user names and passwords to ensure that they remember them. As a result, unauthorized users may more readily obtain user names and passwords from authorized users. Because certificates are stored in a file or other electronic location, client applications (rather than the user) handle accessing and presenting the certificate for authentication. This ensures users are less likely to share certificates with unauthorized users unless unauthorized users have access to the user's system. Also, certificates can be installed on smart cards as an additional means of protecting them from unauthorized usage.
- A certificate contains a private key that is never sent with the certificate for identification. Instead, the system uses this key during encryption and decryption processing. Others can use the certificate's corresponding public key to verify the identity of the sender of objects that are signed with the private key.
- Many systems require passwords that are 8 characters or shorter in length, making these passwords more vulnerable to guessing attacks. A certificate's cryptographic keys are hundreds of characters long. This length, along with their random nature, makes cryptographic keys much harder to guess than passwords.
- Digital certificate keys provide several potential uses that passwords cannot provide, such as data integrity and privacy. You can use certificates and their associated keys to:
  - Assure data integrity by detecting changes to data.
  - Prove that a particular action was indeed performed. This is called nonrepudiation.
  - Ensure the privacy of data transfers by using the Secure Sockets Layer (SSL) to encrypt communication sessions.

Related concepts

"Digital certificates for SSL secure communications" on page 32
Use this information to learn how to use certificates so that your applications can establish secure communication sessions.

Related reference

"Using APIs to programmatically issue certificates to users other than System i users" on page 45
Use this information to learn how you can use your local CA to issue private certificates to users without associating the certificate with a System i user profile.

Digital certificates and Enterprise Identity Mapping (EIM)

Using Enterprise Identity Mapping (EIM) and Digital Certificate Mangers (DCM) together allows you to apply a certificate as the source of an EIM mapping lookup operation to map from the certificate to a target user identity associated with the same EIM identifier.

EIM is an @server technology that allows you to manage user identities in your enterprise, including user profiles and user certificates. A user name and password is the most common form of user identity; certificates are another form of user identity. Some applications are configured to allow users to be authenticated by means of a user certificate rather than by means of a user name and password.

You can use EIM to create mappings between user identities, which allows a user to authenticate with one user identity and access resources of another user identity without the user having to supply the needed user identity. You accomplish this in EIM by defining an association between one user identity and another user identity. User identities can be in various forms, including user certificates. You can either create individual associations between an EIM identifier and the various user identities that belong to a user represented by that EIM identifier. Or, you can create policy associations, which map a group of user identities to a single target user identity. User identities can be in various forms, including user certificates. When you create these associations, user certificates can be mapped to the appropriate EIM identifiers thereby making it easier for the certificates to be used for authentication.

To take advantage of this EIM feature for managing user certificates, you need to perform these EIM configuration tasks before performing any DCM configuration tasks:

1. Use the **EIM Configuration** wizard in **iSeries Navigator** to configure EIM.
2. Create an EIM identifier for each user that you want to have participate in EIM.
3. Create a target association between each EIM identifier and that user's user profile in the local i5/OS user registry so that any user certificates that the user assigns through DCM or creates in DCM can be mapped to the user profile. Use the EIM registry definition name for the local **i5/OS** user registry that you specified in the **EIM Configuration** wizard.

After you complete the necessary EIM configuration tasks, you must use the **Manage LDAP Location** task to configure Digital Certificate Manager (DCM) to store user certificates in a Lightweight Directory Access Protocol (LDAP) location instead of with a user profile. When you configure EIM and DCM to work together, the **Create Certificate** task for user certificates and the **Assign a user certificate** task process certificates for EIM usage rather than assigning the certificate to a user profile. DCM stores the certificate in the configured LDAP directory and uses the certificate's distinguished name (DN) information to create a source association for the appropriate EIM identifier. This allows operating systems and applications to use the certificate as the source of an EIM mapping lookup operation to map from the certificate to a target user identity associated with the same EIM identifier.

Additionally, when you configure EIM and DCM to work together you can use DCM to check user certificate expiration at the enterprise level rather than just at the system level.

Related concepts

"Public certificates versus private certificates" on page 29
Review this information to learn how to determine which type of certificate (public or private) best suits your business needs.

Related tasks

"Managing user certificates by expiration" on page 44
Digital Certificate Manager (DCM) provides certificate expiration management support to allow administrators to check the expiration dates of user certificates on the local System i model. DCM user certificate expiration management support can be used in conjunction with Enterprise Identity Mapping (EIM) so that administrators can use DCM to check user certificate expiration at the enterprise level.

"Managing LDAP location for user certificates" on page 71
Review this information to learn how to configure DCM to store user certificates in a Lightweight Directory Access Protocol (LDAP) server directory location to extend Enterprise Identity Mapping to work with user certificates.

Related information

EIM Information Center topic

Digital certificates for VPN connections

Review this information to learn how to use certificates as part of configuring a Virtual Private Network (VPN) connection.

You can use digital certificates as a means of establishing an System i VPN connection. Both endpoints of a dynamic VPN connection must be able to authenticate each other before activating the connection. Endpoint authentication is done by the Internet Key Exchange (IKE) server on each end. After successful authentication, the IKE servers then negotiate the encryption methodologies and algorithms they will use to secure the VPN connection.

One method that the IKE servers can use to authenticate each other is a pre-shared key. However, the use of a pre-shared key is less secure because you must communicate this key manually to the administrator of the other endpoint for your VPN. Consequently, there is a possibility that the key could be exposed to others during the process of communicating the key.

You can avoid this risk by using digital certificates to authenticate the endpoints instead of using a pre-shared key. The IKE server can authenticate the other server's certificate to establish a connection to negotiate the encryption methodologies and algorithms the servers will use to secure the connection.

You can use Digital Certificate Manager (DCM) to manage the certificates that your IKE server uses for establishing a dynamic VPN connection. You must first decide whether to use public certificates versus issuing private certificates for your IKE server.

Some VPN implementations require that the certificate contain alternative subject name information, such as a domain name or an e-mail address, in addition to the standard distinguished name information. When you use the local CA in DCM to issue a certificate you can specify alternative subject name information for the certificate. Specifying this information ensures that your VPN connection is compatible with other VPN implementations that may require it for authentication.

Related concepts

"Managing certificates from a public Internet CA" on page 47
Review this information to learn how to manage certificates from a public Internet CA by creating a certificate store.

Related tasks

"Creating and operating a local CA" on page 39
This information explains how to create and operate a local Certificate Authority (CA) to issue private certificates for your applications.

"Defining a CA trust list for an application" on page 65
Applications that support the use of certificates for client authentication during a Secure Sockets Layer (SSL) session must determine whether to accept a certificate as valid proof of identity. One of the criteria that an application uses for authenticating a certificate is whether the application trusts the Certificate Authority (CA) that issued the certificate.

Related information

Configuring a VPN connection

Digital certificates for signing objects

Use this information to learn how to use certificates to ensure an object's integrity or to verify the digital signature on an object to verify its authenticity.

IBM i5/OS provides support for using certificates to digitally ″sign″ objects. Digitally signing objects provides a way to verify both the integrity of the object's contents and its source of origin. Object signing support augments traditional System i model tools for controlling who can change objects. Traditional controls cannot protect an object from unauthorized tampering while the object is in transit across the Internet or other untrusted network, or while the object is stored on a system other than the System i platform. Also, traditional controls cannot always determine whether unauthorized changes to or tampering with an object has occurred. Using digital signatures on objects provides a sure means of detecting changes to the signed objects.

Placing a digital signature on an object consists of using a certificate's private key to add an encrypted mathematical summary of the data in an object. The signature protects the data from unauthorized changes. The object and its contents are not encrypted and made private by the digital signature; however, the summary itself is encrypted to prevent unauthorized changes to it. Anyone who wants to ensure that the object has not been changed in transit and that the object originated from an accepted, legitimate source can use the signing certificate's public key to verify the original digital signature. If the signature no longer matches, the data may have been altered. In such a case, the recipient can avoid using the object and can instead contact the signer to obtain another copy of the signed object.

If you decide that using digital signatures fits your security needs and policies, you need to evaluate whether you need to use public certificates versus issuing private certificates. If you intend to distribute objects to users in the general public, you might consider using certificates from a well-known public Certificate Authority (CA) to sign objects. Using public certificates ensures that others can easily and inexpensively verify the signatures that you place on objects that you distribute to them. If, however, you intend to distribute objects solely within your organization, you may prefer to use Digital Certificate Manager (DCM) to operate your own local CA to issue certificates for signing objects. Using private certificates from a local CA to sign objects is less expensive than purchasing certificates from a well-known public CA.

The signature on an object represents the system that signed the object, not a specific user on that system (although the user must have the appropriate authority to use the certificate for signing objects). You use DCM to manage the certificates that you use to sign objects and to verify object signatures. You can also use DCM to sign objects and to verify object signatures.

Related concepts

"Public certificates versus private certificates" on page 29
Review this information to learn how to determine which type of certificate (public or private) best suits your business needs.

"Digital certificates for verifying object signatures" on page 37
This information explains how to use certificates to verify the digital signature on an object to verify its authenticity.

Related tasks

"Verifying object signatures" on page 74
You can use Digital Certificate Manager (DCM) to verify the authenticity of digital signatures on objects. When you verify the signature, you ensure that the data in the object has not been changed since the object owner signed the object.

"Managing public Internet certificates for signing objects" on page 49
You can use Digital Certificate Manager (DCM) to manage public Internet certificates to digitally sign objects.

"Managing certificates for verifying object signatures" on page 51
You can use Digital Certificate Manager (DCM) to manage the signature verification certificates that you use to validate digital signatures on objects.

Digital certificates for verifying object signatures

This information explains how to use certificates to verify the digital signature on an object to verify its authenticity.

IBM i5/OS provides support for using certificates to verify digital signatures on objects. Anyone who wants to ensure that a signed object has not been changed in transit and that the object originated from an accepted source can use the signing certificate's public key to verify the original digital signature. If the signature no longer matches, the data may have been altered. In such a case, the recipient can avoid using the object and can instead contact the signer to obtain another copy of the signed object.

The signature on an object represents the system that signed the object, not a specific user on that system. As part of the process of verifying digital signatures, you must decide which Certificate Authorities you trust and which certificates you trust for signing objects. When you elect to trust a Certificate Authority (CA), you can elect whether to trust signatures that someone creates by using a certificate that the trusted CA issued. When you elect not to trust a CA, you also are electing not to trust certificates that the CA issues or signatures that someone creates by using those certificates.

Verify object restore (QVFYOBJRST) system value

If you decide to perform signature verification, one of the first important decisions you must make is to determine how important signatures are for objects being restored to your system. You control this with a system value called Verify object signatures during restore (QVFYOBJRST). The default setting for this system value allows unsigned objects to be restored, but ensures that signed objects can be restored only if the objects have a valid signature. The system defines an object as signed only if the object has a signature that your system trusts; the system ignores other, ″untrusted″ signatures on the object and treats the object as if it is unsigned.

There are several values that you can use for the QVFYOBJRST system value, ranging from ignoring all signatures to requiring valid signatures for all objects that the system restores. This system value only affects executable objects that are being restored, not save files or integrated file system files. To learn more about using this and other system values, see the System Value Finder in the i5/OS Information Center.

You use Digital Certificate Manager (DCM) to implement your certificate and CA trust decisions as well as to manage the certificates that you use to verify object signatures. You can also use DCM to sign objects and to verify object signatures.

Related concepts

"Digital certificates for signing objects" on page 36
Use this information to learn how to use certificates to ensure an object's integrity or to verify the digital signature on an object to verify its authenticity.

Related information

System Value Finder

QVFYOBJRST system value

Configuring DCM

Digital Certificate Manager (DCM) provides a browser-based user interface that you can use to manage and configure digital certificates for your applications and users. The user interface is divided into two main frames: a navigation frame and a task frame.

You use the navigation frame to select the tasks to manage certificates or the applications that use them. While some individual tasks appear directly in the main navigation frame, most tasks in the navigation frame are organized into categories. For example, **Manage Certificates** is a task category that contains a variety of individual guided tasks, such as View certificate, Renew certificate, Import certificate, and so

forth. If an item in the navigation frame is a category that contains more than one task, an arrow appears to the left of it. The arrow indicates that when you select the category link, an expanded list of tasks displays so that you may choose which task to perform.

With the exception of the **Fast Path** category, each task in the navigation frame is a guided task that takes you through a series of steps to complete the task quickly and easily. The Fast Path category provides a cluster of certificate and application management functions which allows experienced DCM users to quickly access a variety of related tasks from a central set of pages.

Which tasks are available in the navigation frame vary based on the certificate store in which you are working. Also, the category and number of tasks that you see in the navigation frame vary depending on the authorizations that your i5/OS user profile has. All tasks for operating a CA, managing the certificates that applications use, and other system level tasks are available only to System i security officers or administrators. The security officer or administrator must have *SECADM and *ALLOBJ special authorities to view and use these tasks. Users without these special authorities have access to user certificate functions only.

To learn how to configure DCM and begin using it to manage your certificates, review these topics:

If you want like more educational information about using digital certificates in an Internet environment for enhancing your system and network security, the VeriSign Web site is an excellent resource. The VeriSign Web site provides an extensive library on digital certificates topics, as well as a number of other Internet security subjects. You can access their library at the VeriSign Help Desk .

Starting Digital Certificate Manager

Use this information to learn how to access the Digital Certificate Manager (DCM) feature on your system.

Before you can use any DCM functions, you need to start it. Complete these tasks to ensure that you can start DCM successfully:

1. Install 5722 SS1 Option 34. This is Digital Certificate Manager (DCM).
2. Install 5722 DG1. This is the IBM HTTP Server for i5/OS.
3. Use iSeries Navigator to start the HTTP Server Administrative server:
   a. Start **iSeries Navigator** .
   b. Double-click your system in the main tree view.
   c. Expand **Network** > **Servers** > **TCP/IP**.
   d. Right-click **HTTP Administration**.
   e. Click **Start**.
4. Start your Web browser.
5. Using your browser, go to the System i Tasks page on your system at http://*your_system_name*:2001.
6. Select **Digital Certificate Manager** from the list of products on the System i Tasks page to access the DCM user interface.

Related concepts

"Scenario: Using certificates for external authentication" on page 12
In this scenario, you lean when and how to use certificates as an authentication mechanism to protect and limit access by public users to public or extranet resources and applications.

Setting up certificates for the first time

Use this information to learn how to get started managing certificates from a public Internet Certificate Authority (CA) or how to create and operate a private local CA to issue certificates.

The left frame of Digital Certificate Manager (DCM) is the task navigation frame. You can use this frame to select a wide variety of tasks for managing certificates and the applications that use them. Which tasks are available depends on which certificate store (if any) you work with and your user profile special authorities. Most tasks are available only if you have *ALLOBJ and *SECADM special authorities. To use DCM to verify object signatures, your user profile must also have *AUDIT special authority.

When you use Digital Certificate Manager (DCM) for the first time, no certificate stores exist. Consequently, when you initially access DCM, the navigation pane displays only these tasks and only when you have the necessary special authorities:

- Manage User Certificates.
- Create New Certificate Store.
- Create a Certificate Authority (CA). (Note: After you use this task to create a private local CA, this task no longer appears in the list.)
- Manage CRL Locations.
- Manage LDAP Location.
- Manage PKIX Request Location.
- Return to iSeries Tasks.

Even if certificate stores already exist on your system (for example, you are migrating from an earlier version of DCM), DCM displays only a limited number of tasks or task categories in the left navigation frame. Which tasks or categories DCM displays varies based on the certificate store (if any) that is open and the special authorities for your user profile.

You must first access the appropriate certificate store before you can begin working with most certificate and application management tasks. To open a specific certificate store, click **Select a Certificate Store** in the navigation frame.

The navigation frame of DCM also provides a **Secure Connection** button. You can use this button to display a second browser window to initiate a secure connection by using Secure Sockets Layer (SSL). To use this function successfully, you must first configure the IBM HTTP Server for i5/OS to use SSL to operate in secure mode. You must then start the HTTP Server in secure mode. If you have not configured and started the HTTP Server for SSL operation, you will see an error message and your browser will not start a secure session.

Getting started

Although you may want to use certificates to accomplish a number of security-related goals, what you do first depends on how you plan to obtain your certificates. There are two primary paths that you can take when you first use DCM, based on whether you intend to use public certificates versus issuing private certificates.

Related concepts

"Public certificates versus private certificates" on page 29
Review this information to learn how to determine which type of certificate (public or private) best suits your business needs.

Creating and operating a local CA

This information explains how to create and operate a local Certificate Authority (CA) to issue private certificates for your applications.

After careful review of your security needs and policies, you have decided to operate a local Certificate Authority (CA) to issue private certificates for your applications. You can use Digital Certificate Manager (DCM) to create and operate your own local CA. DCM provides you with a guided task path that takes you through the process of creating a CA and using it to issue certificates to your applications. The

guided task path ensures that you have everything you need to begin using digital certificates to configure applications to use SSL and to sign objects and verify object signatures.

Note: To use certificates with the IBM HTTP Server for i5/OS , you must create and configure your Web server before working with DCM. When you configure a Web server to use SSL, an application ID is generated for the server. You must make a note of this application ID so that you can use DCM to specify which certificate this application will use for SSL.

Do not end and restart the server until you use DCM to assign a certificate to the server. If you end and restart the *ADMIN instance of the Web server before assigning a certificate to it, the server will not start and you will not be able to use DCM to assign a certificate to the server.

To use DCM to create and operate a local CA, follow these steps:

1. Start DCM. Refer to Starting DCM.
2. In the navigation frame of DCM, select Create a Certificate Authority (CA) to display a series of forms. These forms guide you through the process of creating a local CA and completing other tasks needed to begin using digital certificates for SSL, object signing, and signature verification.

   Note: If you have questions about how to complete a specific form in this guided task, select the question mark (?) button at the top of the page to access the online help.
3. Complete all the forms for this guided task. In using these forms to perform all the tasks that you need to set up a working local Certificate Authority (CA), you:
   a. Choose how to store the private key for the local CA certificate. (This step is provided only if you have an IBM Cryptographic Coprocessor installed on your system. If your system does not have a cryptographic coprocessor, DCM automatically stores the certificate and its private key in the local Certificate Authority (CA) certificate store.)
   b. Provide identifying information for the local CA.
   c. Install the local CA certificate on your PC or in your browser so that your software can recognize the local CA and validate certificates that the CA issues.
   d. Choose the policy data for your local CA.
   e. Use the new local CA to issue a server or client certificate that your applications can use for SSL connections. (If your system has an IBM Cryptographic Coprocessor installed, this step allows you to select how to store the private key for the server or client certificate. If your system does not have a coprocessor, DCM automatically places the certificate and its private key in the *SYSTEM certificate store. DCM creates the *SYSTEM certificate store as part of this subtask.)
   f. Select the applications that can use the server or client certificate for SSL connections.

      Note: If you used DCM previously to create the *SYSTEM certificate store to manage certificates for SSL from a public Internet CA, you do not perform this or the previous step.
   g. Use the new local CA to issue an object signing certificate that applications can use to digitally sign objects. This subtask creates the *OBJECTSIGNING certificate store; this is the certificate store that you use to manage object signing certificates.
   h. Select the applications that can use the object signing certificate to place digital signatures on objects.

      Note: If you used DCM previously to create the *OBJECTSIGNING certificate store to manage object signing certificates from a public Internet CA, you do not perform this or the previous step.
   i. Select the applications that will trust your local CA.

When you finish the guided task, you have everything that you need to begin configuring your applications to use SSL for secure communications.

After you configure your applications, users that access the applications through an SSL connection must use DCM to obtain a copy of the local CA certificate. Each user must have a copy of the certificate so that the user's client software can use it to authenticate the identity of the server as part of the SSL negotiation process. Users can use DCM either to copy the local CA certificate to a file or to download the certificate into their browser. How the users store the local CA certificate depends on the client software that they use to establish an SSL connection to an application .

Also, you can use this local CA to issue certificates to applications on other System i models in your network.

To learn more about using DCM to manage user certificates and how users can obtain a copy of the local CA certificate to authenticate certificates the local CA issues, review these topics:

Related concepts

"Public certificates versus private certificates" on page 29
Review this information to learn how to determine which type of certificate (public or private) best suits your business needs.

"Digital certificates for VPN connections" on page 35
Review this information to learn how to use certificates as part of configuring a Virtual Private Network (VPN) connection.

"Managing user certificates"
You can use Digital Certificate Manager (DCM) to obtain certificates with SSL or associate existing certificates with their iSeries user profiles.

Related tasks

"Using a local CA to issue certificates for other System i models" on page 55
Review this information to learn how to use a private local CA on one system to issue certificates for use on other System i models.

"Obtaining a copy of the private CA certificate" on page 46
Review this information to learn how to obtain a copy of the private CA certificate and install it on your PC so that you can authenticate any server certificates that the CA issues.

Related reference

"Using APIs to programmatically issue certificates to users other than System i users" on page 45
Use this information to learn how you can use your local CA to issue private certificates to users without associating the certificate with a System i user profile.

Managing user certificates:

You can use Digital Certificate Manager (DCM) to obtain certificates with SSL or associate existing certificates with their iSeries user profiles.

If users access your public or internal servers through an SSL connection, they must have a copy of the Certificate Authority (CA) certificate that issued the server's certificate. They must have the CA certificate so that their client software can validate the authenticity of the server certificate to establish the connection. If your server uses a certificate from a public CA, your users' software might already possess a copy of the CA certificate. Consequently, neither you as a DCM administrator, nor your users, need take any action before they can participate in an SSL session. However, if your server uses a certificate from a private local CA, your users must obtain a copy of the local CA certificate before they can establish an SSL session with the server.

Additionally, if the server application supports and requires client authentication through certificates, users must present an acceptable user certificate to access resources that the server provides. Depending on your security needs, users can present a certificate from a public Internet CA or one that they obtain from a local CA that you operate. If your server application provides access to resources for internal users

who currently have iSeries user profiles, you can use DCM to add their certificates to their user profiles. This association ensures that users have the same access and restrictions to resources when presenting certificates as their user profile grants or denies.

Digital Certificate Manager (DCM) allows you to manage certificates that are assigned to an iSeries user profile. If you have a user profile with *SECADM and *ALLOBJ special authorities, you can manage user profile certificate assignments for yourself or for other users. When no certificate store is open, or when the local Certificate Authority (CA) certificate store is open, you can select **Manage User Certificates** in the navigation frame to access the appropriate tasks. If a different certificate store is open, user certificate tasks are integrated into the tasks under **Manage Certificates**.

Users without *SECADM and *ALLOBJ user profile special authorities can manage their own certificate assignments only. They can select **Manage User Certificates** to access tasks that allow them to view the certificates associated with their user profiles, remove a certificate from their user profiles, or assign a certificate from a different CA to their user profiles. Users, regardless of the special authorities for their user profiles, can obtain a user certificate from the local CA by selecting the **Create Certificate** task in the main navigation frame.

To learn more about how to use DCM to manage and create user certificates, review these topics:

Related tasks

"Creating and operating a local CA" on page 39
This information explains how to create and operate a local Certificate Authority (CA) to issue private certificates for your applications.

"Obtaining a copy of the private CA certificate" on page 46
Review this information to learn how to obtain a copy of the private CA certificate and install it on your PC so that you can authenticate any server certificates that the CA issues.

Creating a user certificate:

Review this information to learn how your users can use the local CA to issue a certificate for client authentication.

If you want to use digital certificates for user authentication, users must have certificates. If you use Digital Certificate Manager (DCM) to operate a private local Certificate Authority (CA), you can use the local CA to issue certificates to each user. Each user must access DCM to obtain a certificate by using the **Create Certificate** task. In order to obtain a certificate from the local CA, the CA policy must allow the CA to issue user certificates.

To obtain a certificate from the local CA, complete these steps:

1. Start DCM. Refer to Starting DCM.
2. In the navigation frame, select **Create Certificate**.
3. Select **User certificate** as the type of certificate to create. A form displays so that you can provide identifying information for the certificate.
4. Complete the form and click **Continue**.

   Note: If you have questions about how to complete a specific form in this guided task, select the question mark (**?**) at the top of the page to access the online help.
5. At this point, DCM works with your browser to create the private and public key for the certificate. Your browser may display windows to guide you through this process. Follow the browser's instructions for these tasks. After the browser generates the keys, a confirmation page displays to indicate that DCM created the certificate.
6. Install the new certificate in your browser software. Your browser may display windows to guide you through this process. Follow the instructions that the browser gives to complete this task.
7. Click **OK** to complete the task.

During processing, the Digital Certificate Manager automatically associates the certificate with your System i user profile.

If you want a certificate from another CA that a user presents for client authentication to have the same authorities as their user profile, the user can use DCM to assign the certificate to their user profile.

Related concepts

"Public certificates versus private certificates" on page 29
Review this information to learn how to determine which type of certificate (public or private) best suits your business needs.

Related tasks

"Assigning a user certificate"
You can assign a user certificate that you own to your i5/OS user profile or other user identity. The certificate may be from a private local CA on another system or from a well-known Internet CA. Before you can assign a certificate to a user identity, the issuing CA must be trusted by the server, and the certificate must not already be associated with a user profile or other user identity on the system.

"Obtaining a copy of the private CA certificate" on page 46
Review this information to learn how to obtain a copy of the private CA certificate and install it on your PC so that you can authenticate any server certificates that the CA issues.

Assigning a user certificate:

You can assign a user certificate that you own to your i5/OS user profile or other user identity. The certificate may be from a private local CA on another system or from a well-known Internet CA. Before you can assign a certificate to a user identity, the issuing CA must be trusted by the server, and the certificate must not already be associated with a user profile or other user identity on the system.

Some users may have certificates from an outside Certificate Authority (CA) or a local CA on a different iSeries system that you, as an administrator, want them to make available to Digital Certificate Manager (DCM). This allows you and the user to use DCM to manage these certificates, which are most often used for client authentication. The **Assign a user certificate** task provides a mechanism for allowing a user to create a DCM assignment for a certificate obtained from an outside CA.

When a user assigns a certificate, DCM has one of two ways of handling the assigned certificate:

- Storing the certificate locally on the System i with the user's user profile. When an LDAP location is not defined for DCM, the **Assign a user certificate** task allows a user to assign an outside certificate to an i5/OS user profile. Assigning the certificate to a user profile ensures that the certificate can be used with applications on the system that require certificates for client authentication.
- Storing the certificate in a Lightweight Directory Access Protocol (LDAP) location for use with Enterprise Identity Mapping (EIM). When there is a defined LDAP location and the System i model is configured to participate in EIM, then the **Assign a user certificate** task allows a user to store a copy of an outside certificate in the specified LDAP directory. DCM also creates a source association in EIM for the certificate. Storing the certificate in this manner allows an EIM administrator to recognize the certificate as a valid user identity that can participate in EIM.

  Note: Before a user can assign a certificate to a user identity in an EIM configuration, EIM must be configured appropriately for the user. This EIM configuration involves the creation of an EIM identifier for the user and the creation of a target association between that EIM identifier and the user profile. Otherwise, DCM cannot create a corresponding source association with the EIM identifier for the certificate.

To use the **Assign a user certificate** task, a user must meet the following requirements:

1. Have a secure session with the HTTP Server through which you are accessing DCM.

Whether you have a secure session is determined by the port number in the URL that you used to access DCM. If you used port 2001, which is the default port for accessing DCM, you do not have a secure session. Also, the HTTP Server must be configured to use SSL before you can switch to a secure session.

When the user selects this task, a new browser window displays. If the user does not have a secure session, DCM prompts the user to click **Assign a User Certificate** to start one. DCM then initiates Secure Sockets Layer (SSL) negotiations with the user's browser. As part of these negotiations, the browser may prompt the user as to whether to trust the Certificate Authority (CA) that issued the certificate that identifies the HTTP Server. Also, the browser may prompt the user as to whether to accept the server certificate itself.

2. Present a certificate for client authentication.

   Depending on the configuration settings for your browser, your browser may prompt you to select a certificate to present for authentication. If your browser presents a certificate from a CA that the system accepts as trusted, DCM displays the certificate information in a separate window. If you do not present an acceptable certificate, the server may prompt you instead for your user name and password for authentication before allowing you access.

3. Have a certificate in the browser that is not already associated with the user identity for the user who is performing the task. (Or, if DCM is configured for working in conjunction with EIM, the user must have a certificate in the browser that is not already stored in the LDAP location for DCM.)

   Once you establish a secure session, DCM attempts to retrieve an appropriate certificate from your browser so that it can associate it with your user identity. If DCM successfully retrieves one or more certificates, you can view the certificate information and choose to associate the certificate with your user profile.

   If DCM does not display information from a certificate, you were not able to provide a certificate that DCM can assign to your user identity. One of several user certificate problems may be responsible. For example, the certificates that your browser contains may be associated with your user identity already.

   Related tasks

   "Creating a user certificate" on page 42
   Review this information to learn how your users can use the local CA to issue a certificate for client authentication.

   "Troubleshooting assigning a user certificate" on page 81
   When you use the **Assign a user certificate** task, Digital Certificate Manager (DCM) displays certificate information for you to approve before registering the certificate.

   Related information

   EIM Information Center Overview

Managing user certificates by expiration:

Digital Certificate Manager (DCM) provides certificate expiration management support to allow administrators to check the expiration dates of user certificates on the local System i model. DCM user certificate expiration management support can be used in conjunction with Enterprise Identity Mapping (EIM) so that administrators can use DCM to check user certificate expiration at the enterprise level.

To take advantage of expiration management support for user certificates at the enterprise level, EIM must be configured in the enterprise and EIM must contain the appropriate mapping information for user certificates. To check the expiration of user certificates other than those associated with your own user profile, you must have *ALLOBJ and *SECADM special authorities.

Using DCM to view certificates based on their expiration allows you to determine quickly and easily which certificates are close to expiring so that certificates can be renewed in a timely fashion.

To view and manage user certificates based on their expiration dates, follow these steps:

1. Start DCM. Refer to Starting DCM..

   Note: If you have questions about how to complete a specific form while using DCM, select the question mark (**?**) at the top of the page to access the online help.
2. In the navigation frame, select **Manage User Certificates** to display a list of tasks.

   Note: If you are currently working with a certificate store, select **Manage Certificates** to display a list of tasks, then select **Check expiration**, and select **User**.
3. If your user profile has *ALLOBJ and *SECADM special authorities, you can select a method for choosing which user certificates to view and manage based on their expiration dates. (If your user profile does not have these special authorities, DCM prompts you to specify the expiration date range as described in the next step.) You can select one of the following:
   - **User profile** to view and manage user certificates that are assigned to a specific i5/OS user profile. Specify a **User profile name** and click **Continue**.

     Note: You can specify a user profile other than your own user profile only if you have *ALLOBJ and *SECADM special authorities.
   - **All user certificates** to view and to manage user certificates for all user identities.
4. In the **Expiration date range in days (1-365)** field, enter the number of days for which to view user certificates based on their expiration date and click **Continue**. DCM displays all user certificates for the specified user profile that expire between today's date and the date that matches the number of specified days. DCM also displays all user certificates that have expiration dates before today's date.
5. Select a user certificate to manage. You can choose to view certificate information details or remove the certificate from the associated user identity.
6. When you finish working with certificates from the list, click **Cancel** to exit the task.

Related tasks

"Digital certificates and Enterprise Identity Mapping (EIM)" on page 34
Using Enterprise Identity Mapping (EIM) and Digital Certificate Mangers (DCM) together allows you to apply a certificate as the source of an EIM mapping lookup operation to map from the certificate to a target user identity associated with the same EIM identifier.

"Managing certificates by expiration" on page 66
Digital Certificate Manager (DCM) provides certificate expiration management support to allow administrators to manage server or client certificates, object signing certificates, and user certificates by expiration date on the local system.

Related information

EIM Information Center Overview

Using APIs to programmatically issue certificates to users other than System i users:

Use this information to learn how you can use your local CA to issue private certificates to users without associating the certificate with a System i user profile.

In i5/OS V5R3 or later, there are two new APIs available that you can use to programmatically issue certificates to users other than System i users. In previous releases, when you used your local Certificate Authority (CA) to issue certificates to users, these certificates were automatically associated with their System i user profiles. Consequently, to use the local CA to issue a certificate to a user for client authentication, you had to provide that user with a System i user profile. Also, when users needed to obtain a certificate from a local CA for client authentication, each user had to use Digital Certificate Manager (DCM) to create the needed certificate. Therefore, each user must have a System i user profile on the system that hosts DCM and a valid sign-on to that system.

Having the certificate associated with a user profile has its advantages, especially when internal users are concerned. However, these restrictions and requirements made it less practical to use the local CA to

issue user certificates for a large number of users, especially when you do not want those users to have a System i user profile. To avoid providing user profiles to these users, you might require users to pay for a certificate from a well-known CA if you wanted to require certificates for user authentication for your applications.

These two new APIs provide support that allows you to provide an interface for creating user certificates signed by the local CA certificate for any user name. This certificate will not be associated with a user profile. The user does not need to exist on the system that hosts DCM and the user does not need to use DCM to create the certificate.

There are two APIs, one for each of the predominate browser programs, that you can call when using Net.Data® to create a program for issuing certificates to users. The application that you create must provide the Graphical User Interface (GUI) code needed to create the user certificate and to call one of the appropriate API to use the local CA to sign the certificate.

Related concepts

"Public certificates versus private certificates" on page 29
Review this information to learn how to determine which type of certificate (public or private) best suits your business needs.

"Digital certificates for user authentication" on page 32
Review this information to learn how to use certificates to provide a means of more strongly authenticating users who access System i resources.

Related tasks

"Creating and operating a local CA" on page 39
This information explains how to create and operate a local Certificate Authority (CA) to issue private certificates for your applications.

Related information

Generate and Sign User Certificate Request (QYCUGSUC) API

Sign User Certificate Request (QYCUSUC) API

Obtaining a copy of the private CA certificate:

Review this information to learn how to obtain a copy of the private CA certificate and install it on your PC so that you can authenticate any server certificates that the CA issues.

When you access a server that uses a Secure Sockets Layer (SSL) connection, the server presents a certificate to your client software as proof of its identity. Your client software must then validate the server's certificate before the server can establish the session. To validate the server certificate, your client software must have access to a locally stored copy of the certificate for the Certificate Authority (CA) that issued the server certificate. If the server presents a certificate from a public Internet CA, your browser or other client software might already have a copy of the CA certificate. If, however, the server presents a certificate from a private local CA, you must use Digital Certificate Manager (DCM) to obtain a copy of the local CA certificate.

You can use DCM to download the local CA certificate directly into your browser, or you can copy the local CA certificate into a file so that other client software can access and use it. If you use both your browser and other applications for secure communications, you may need to use both methods to install the local CA certificate. If using both methods, install the certificate in your browser before you copy and paste it into a file.

If the server application requires that you authenticate yourself by presenting a certificate from the local CA, you must download the local CA certificate into your browser before requesting a user certificate from the local CA.

To use DCM to obtain a copy of a local CA certificate, complete these steps:

1. Start DCM. Refer to Starting DCM.
2. In the navigation frame, select **Install local CA Certificate on Your PC** to display a page that allows you to download the local CA certificate into your browser or to store it in a file on your system.
3. Select a method for obtaining the local CA certificate.
   a. Select **Install certificate** to download the local CA certificate as a trusted root in your browser. This ensures that your browser can establish secure communications sessions with servers that use a certificate from this CA. Your browser will display a series of windows to help you complete the installation.
   b. Select **Copy and paste certificate** to display a page that contains a specially coded copy of the local CA certificate. Copy the text object shown on the page into your clipboard. You must later paste this information into a file. This file is used by a PC utility program (such as MKKF or IKEYMAN) to store certificates for use by client programs on the PC. Before your client applications can recognize and use the local CA certificate for authentication, you must configure the applications to recognize the certificate as a trusted root. Follow the instructions that these applications provide for using the file.
4. Click **OK** to return to the Digital Certificate Manager home page.

Related concepts

"Managing user certificates" on page 41
You can use Digital Certificate Manager (DCM) to obtain certificates with SSL or associate existing certificates with their iSeries user profiles.

Related tasks

"Creating and operating a local CA" on page 39
This information explains how to create and operate a local Certificate Authority (CA) to issue private certificates for your applications.

"Creating a user certificate" on page 42
Review this information to learn how your users can use the local CA to issue a certificate for client authentication.

Managing certificates from a public Internet CA

Review this information to learn how to manage certificates from a public Internet CA by creating a certificate store.

After careful review of your security needs and policies, you have decided that you want to use certificates from a public Internet Certificate Authority (CA), such as VeriSign. For example, you operate a public Web site and want to use the Secure Sockets Layer (SSL) for secure communication sessions to ensure the privacy of certain information transactions. Because the Web site is available to the general public, you want to use certificates that most Web browsers can recognize readily.

Or, you develop applications for external customers and want to use a public certificate to digitally sign the application packages. By signing the application package, your customers can be sure that the package came from your company and that unauthorized parties have not altered the code while it was in transit. You want to use a public certificate so that your customers can easily and inexpensively verify the digital signature on the package. You can also use this certificate to verify the signature before sending the package to your customers.

You can use the guided tasks in Digital Certificate Manager (DCM) to centrally manage these public certificates and the applications that use them for establishing SSL connections, signing objects, or verifying the authenticity of digital signatures on objects.

Manage public certificates

When you use DCM to manage certificates from a public Internet CA, you must first create a certificate store. A certificate store is a special key database file that DCM uses to store digital certificates and their associated private keys. DCM allows you to create and manage several types of certificate stores based on the types of certificates that they contain.

The type of certificate store that you create, and the subsequent tasks that you must perform for managing your certificates and the applications that use them, depends on how you plan to use your certificates.

Note: DCM also allows you to manage certificates that you obtain from a Public Key Infrastructure for X.509 (PKIX) Certificate Authority.

To learn how to use DCM to create the appropriate certificate store and manage public Internet certificates for your applications, review these topics:

Related concepts

"Public certificates versus private certificates" on page 29
Review this information to learn how to determine which type of certificate (public or private) best suits your business needs.

"Digital certificates for VPN connections" on page 35
Review this information to learn how to use certificates as part of configuring a Virtual Private Network (VPN) connection.

Related tasks

"Managing the request location for a PKIX CA" on page 70
A Public Key Infrastructure for X.509 (PKIX) Certificate Authority (CA) is a CA that issues certificates based on the newest Internet X.509 standards for implementing a public key infrastructure.

Managing public Internet certificates for SSL communications sessions:

You can use Digital Certificate Manager (DCM) to manage public Internet certificates for your applications to use for establishing secure communications sessions with the Secure Sockets Layer (SSL).

If you do not use DCM to operate your own local Certificate Authority (CA), you must first create the appropriate certificate store for managing the public certificates that you use for SSL. This is the *SYSTEM certificate store. When you create a certificate store, DCM takes you through the process of creating the certificate request information that you must provide to the public CA to obtain a certificate.

To use DCM to manage and use public Internet certificates so that your applications can establish SSL communications sessions, follow these steps:

1. Start DCM. Refer to Starting DCM.
2. In the navigation frame of DCM, select **Create New Certificate Store** to start the guided task and complete a series of forms. These forms guide you through the process of creating a certificate store and a certificate that your applications can use for SSL sessions.

   Note: If you have questions about how to complete a specific form in this guided task, select the question mark (**?**) at the top of the page to access the online help.
3. Select ***SYSTEM** as the certificate store to create and click **Continue**.
4. Select **Yes** to create a certificate as part of creating the *SYSTEM certificate store and click **Continue**.
5. Select **VeriSign or other Internet Certificate Authority (CA)** as the signer of the new certificate, and click **Continue** to display a form that allows you to provide identifying information for the new certificate.

   Note: If yoursystem has an IBM Cryptographic Coprocessor installed, DCM allows you to select how to store the private key for the certificate as the next task. If your system does not have a coprocessor, DCM automatically places the private key in the *SYSTEM certificate store. If you need help with selecting how to store the private key, see the online help in DCM.

6. Complete the form and click **Continue** to display a confirmation page. This confirmation page displays the certificate request data that you must provide to the public Certificate Authority (CA) that will issue your certificate. The Certificate Signing Request (CSR) data consists of the public key and other information that you specified for the new certificate.
7. Carefully copy and paste the CSR data into the certificate application form, or into a separate file, that the public CA requires for requesting a certificate. You must use all the CSR data, including both the Begin and End New Certificate Request lines. When you exit this page, the data is lost and you cannot recover it. Send the application form or file to the CA that you have chosen to issue and sign your certificate.

   Note: You must wait for the CA to return the signed, completed certificate before you can finish this procedure.

   To use certificates with the HTTP Server for your system, you must create and configure your Web server before working with DCM to work with the signed completed certificate. When you configure a Web server to use SSL, an application ID is generated for the server. You must make a note of this application ID so that you can use DCM to specify which certificate this application must use for SSL.

   Do not end and restart the server until you use DCM to assign the signed completed certificate to the server. If you end and restart the *ADMIN instance of the Web server before assigning a certificate to it, the server will not start and you will not be able to use DCM to assign a certificate to the server.
8. After the public CA returns your signed certificate, start DCM.
9. In the navigation frame, click **Select a Certificate Store** and select ***SYSTEM** as the certificate store to open.
10. When the Certificate Store and Password page displays, provide the password that you specified for the certificate store when you created it and click **Continue**.
11. After the navigation frame refreshes, select **Manage Certificates** to display a list of tasks.
12. From the task list, select **Import certificate** to begin the process of importing the signed certificate into the *SYSTEM certificate store. After you finish importing the certificate, you can specify the applications that must use it for SSL communications.
13. In the navigation frame, select **Manage Applications** to display a list of tasks.
14. From the task list, select **Update certificate assignment** to display a list of SSL-enabled applications for which you can assign a certificate.
15. Select an application from the list and click **Update Certificate Assignment**.
16. Select the certificate that you imported and click **Assign New Certificate**. DCM displays a message to confirm your certificate selection for the application.

    Note: Some SSL-enabled applications support client authentication based on certificates. If you want an application with this support to be able to authenticate certificates before providing access to resources, you must define a CA trust list for the application. This ensures that the application can validate only those certificates from CAs that you specify as trusted. If a user or a client application presents a certificate from a CA that is not specified as trusted in the CA trust list, the application will not accept it as a basis for valid authentication.

When you finish the guided task, you have everything that you need to begin configuring your applications to use SSL for secure communications. Before users can access these applications through an SSL session, they must have a copy of the CA certificate for the CA that issued the server certificate. If your certificate is from a well-known Internet CA, your users' client software may already have a copy of the necessary CA certificate. If users need to obtain the CA certificate, they must access the Web site for the CA and follow the directions the site provides.

Managing public Internet certificates for signing objects:

You can use Digital Certificate Manager (DCM) to manage public Internet certificates to digitally sign objects.

If you do not use DCM to operate your own local Certificate Authority (CA), you must first create the appropriate certificate store for managing the public certificates that you use for signing objects. This is the *OBJECTSIGNING certificate store. When you create a certificate store, DCM takes you through the process of creating the certificate request information that you must provide to the public Internet CA to obtain a certificate.

Also, to use a certificate to sign objects you must define an application ID. This application ID controls how much authority is required for someone to sign objects with a specific certificate and provides another level of access control beyond that which DCM provides. By default, the application definition requires a user to have *ALLOBJ special authority to use the certificate for the application to sign objects. (However, you can change the authority the application ID requires by using iSeries Navigator.)

To use DCM to manage and use public Internet certificates to sign objects, complete these tasks:

1. Start DCM. Refer to Starting DCM.
2. In the left navigation frame of DCM, select **Create New Certificate Store** to start the guided task and complete a series of forms. These forms guide you through the process of creating a certificate store and a certificate that you can use to sign objects.

   Note: If you have questions about how to complete a specific form in this guided task, select the question mark (**?**) button at the top of the page to access the online help.
3. Select ***OBJECTSIGNING** as the certificate store to create and click **Continue**.
4. Select **Yes** to create a certificate as part of creating the certificate store and click **Continue**.
5. Select **VeriSign or other Internet Certificate Authority (CA)** as the signer of the new certificate and click **Continue**. This displays a form that allows you to provide identifying information for the new certificate.
6. Complete the form and click **Continue** to display a confirmation page. This confirmation page displays the certificate request data that you must provide to the public Certificate Authority (CA) that will issue your certificate. The Certificate Signing Request (CSR) data consists of the public key and other information that you specified for the new certificate.
7. Carefully copy and paste the CSR data into the certificate application form, or into a separate file, that the public CA requires for requesting a certificate. You must use all the CSR data, including both the Begin and End New Certificate Request lines. When you exit this page, the data is lost and you cannot recover it. Send the application form or file to the CA that you have chosen to issue and sign your certificate.

   Note: You must wait for the CA to return the signed completed certificate before you can finish this procedure.
8. After the public CA returns your signed certificate, start DCM.
9. In the left navigation frame, click **Select a Certificate Store** and select ***OBJECTSIGNING** as the certificate store to open.
10. When the Certificate Store and Password page displays, provide the password that you specified for the certificate store when you created it and click **Continue**.
11. In the navigation frame, select **Manage Certificates** to display a list of tasks.
12. From the task list, select **Import certificate** to begin the process of importing the signed certificate into the *OBJECTSIGNING certificate store. After you finish importing the certificate, you can create an application definition for using the certificate to sign objects.
13. After the left navigation frame refreshes, select **Manage Applications** to display a list of tasks.
14. From the task list, select **Add application** to begin the process of creating an object signing application definition to use a certificate to sign objects.

15. Complete the form to define your object signing application and click **Add**. This application definition does not describe an actual application, but rather describes the type of objects that you plan to sign with a specific certificate. Use the online help to determine how to complete the form.
16. Click **OK** to acknowledge the application definition confirmation message and display the Manage Applications task list.
17. From the task list, select **Update certificate assignment** and click **Continue** to display a list of object signing application IDs for which you can assign a certificate.
18. Select your application ID from the list and click **Update Certificate Assignment**.
19. Select the certificate that you imported and click **Assign New Certificate**.

When you finish these tasks, you have everything that you need to begin signing objects to ensure their integrity.

When you distribute signed objects, those who receive the objects must use a OS/400® V5R1 or later version of DCM to validate the signature validate the signature on the objects to ensure that the data is unchanged and to verify the identify of the sender. To validate the signature, the receiver must have a copy of the signature verification certificate. You must provide a copy of this certificate as part of the package of signed objects.

The receiver also must have a copy of the CA certificate for the CA that issued the certificate that you used to sign the object. If you signed the objects with a certificate from a well-known Internet CA, the receiver's version of DCM might already have a copy of the necessary CA certificate. However, you might provide a copy of the CA certificate along with the signed objects if you think the receiver may not already have a copy. For example, you must provide a copy of the local CA certificate if you signed the objects with a certificate from a private local CA. For security reasons, you must provide the CA certificate in a separate package or publicly make the CA certificate available at the request of those who need it.

Related concepts

"Digital certificates for signing objects" on page 36
Use this information to learn how to use certificates to ensure an object's integrity or to verify the digital signature on an object to verify its authenticity.

Related tasks

"Verifying object signatures" on page 74
You can use Digital Certificate Manager (DCM) to verify the authenticity of digital signatures on objects. When you verify the signature, you ensure that the data in the object has not been changed since the object owner signed the object.

Managing certificates for verifying object signatures:

You can use Digital Certificate Manager (DCM) to manage the signature verification certificates that you use to validate digital signatures on objects.

To sign an object, you use a certificate's private key to create the signature. When you send the signed object to others, you must include a copy of the certificate that signed the object. You do this by using DCM to export the object signing certificate (without the certificate's private key) as a signature verification certificate. You can export a signature verification certificate to a file that you can then distribute to others. Or, if you want to verify signatures that you create, you can export a signature verification certificate into the *SIGNATUREVERIFICATION certificate store.

To validate a signature on an object, you must have a copy of the certificate that signed the object. You use the signing certificate's public key, which the certificate contains, to examine and verify the signature that was created with the corresponding private key. Therefore, before you can verify the signature on an object, you must obtain a copy of the signing certificate from whomever provided you with the signed objects.

You must also have a copy of the Certificate Authority (CA) certificate for the CA that issued the certificate that signed the object. You use the CA certificate to verify the authenticity of the certificate that signed the object. DCM provides copies of CA certificates from most well-known CAs. If, however, the object was signed by a certificate from another public CA or a private local CA, you must obtain a copy of the CA certificate before you can verify the object signature.

To use DCM to verify object signatures, you must first create the appropriate certificate store for managing the necessary signature verification certificates; this is the *SIGNATUREVERIFICATION certificate store. When you create this certificate store, DCM automatically populates it with copies of most well-known public CA certificates.

Note: If you want to be able to verify signatures that you created with your own object signing certificates, you must create the *SIGNATUREVERIFICATION certificate store and copy the certificates from the *OBJECTSIGNING certificate store into it. This is true even if you plan to perform signature verification from within the *OBJECTSIGNING certificate store.

To use DCM to manage your signature verification certificates, complete these tasks:

1. Start DCM. Refer to Starting DCM.
2. In the left navigation frame of DCM, select **Create New Certificate Store** to start the guided task and complete a series of forms.

   Note: If you have questions about how to complete a specific form in this guided task, select the question mark (**?**) button at the top of the page to access the online help.
3. Select ***SIGNATUREVERIFICATION** as the certificate store to create and click **Continue**.

   Note: If the *OBJECTSIGNING certificate store exists, at this point DCM will prompt you to specify whether to copy the object signing certificates into the new certificate store as signature verification certificates. If you want to use your existing object signing certificates to verify signatures, select **Yes** and click **Continue**. You must know the password for the *OBJECTSIGNING certificate store to copy the certificates from it.
4. Specify a password for the new certificate store and click **Continue** to create the certificate store. A confirmation page displays to indicate that the certificate store was created successfully. Now you can use the store to manage and use certificates to verify object signatures.

   Note: If you created this store so that you can verify signatures on objects that you signed, you can stop. As you create new object signing certificates, you must export them from the *OBJECTSIGNING certificate store into this certificate store. If you do not export them, you will not be able to verify the signatures that you create with them. If you created this certificate store so that you can verify signatures on objects that you received from other sources, you must continue with this procedure so that you can import the certificates that you need into the certificate store.
5. In the navigation frame, click **Select a Certificate Store** and select ***SIGNATUREVERIFICATION** as the certificate store to open.
6. When the Certificate Store and Password page displays, provide the password that you specified for the certificate store when you created it and click **Continue**.
7. After the navigation frame refreshes, select **Manage Certificates** to display a list of tasks.
8. From the task list, select **Import certificate**. This guided task guides you through the process of importing the certificates that you need into the certificate store so that you can verify the signature on the objects that you received.
9. Select the type of certificate that you want to import. Select **Signature verification** to import the certificate that you received with the signed objects and complete the import task.

   Note: If the certificate store does not already contain a copy of the CA certificate for the CA that issued the signature verification certificate, you must import the CA certificate *first*. You may

receive an error when importing the signature verification certificate if you do not import the CA certificate before importing the signature verification certificate.

You can now use these certificates to verify object signatures.

Related concepts

"Digital certificates for signing objects" on page 36
Use this information to learn how to use certificates to ensure an object's integrity or to verify the digital signature on an object to verify its authenticity.

Related tasks

"Verifying object signatures" on page 74
You can use Digital Certificate Manager (DCM) to verify the authenticity of digital signatures on objects. When you verify the signature, you ensure that the data in the object has not been changed since the object owner signed the object.

Renewing an existing certificate

The certificate renewal process that Digital Certificate Manager (DCM) uses varies based on the type of Certificate Authority (CA) that issued the certificate.

You can renew a certificate with the local CA or with an Internet CA.

Renewing a certificate from the local CA

If you use the local CA to sign the renewed certificate, DCM uses the information that you provide to create a new certificate in the current certificate store and retains the previous certificate.

To renew a certificate with a local CA follow these steps:

1. In the navigation frame, click **Select a Certificate Store**, then select the certificate store that holds the certificate you want to renew.
2. In the navigation frame, select **Manage Certificates**.
3. In the navigation frame, select **Renew certificate**.
4. Select the certificate that you want to renew and click **Renew**.
5. Select **local Certificate Authority (CA)** and click **Continue**.
6. Complete the certificate identification form. You must change the **New certificate label** field, but any other fields can remain the same.
7. Select any applications that you want the renewed certificate to use and click **Continue** to finish renewing the certificate.

   Note: You do not have to select an application to use the certificate.

Renewing a certificate from an Internet CA

If you use a well-known, Internet CA to issue the certificate, you can handle the certificate renewal in two different ways.

You can renew the certificate directly with the Internet CA and then import the renewed certificate from the file that you receive from the signing CA. Or, you can use DCM to create a new public-private key pair and Certificate Signing Request (CSR) for the certificate and then send this information to the Internet CA to obtain a new certificate. When you receive that certificate back from the CA you can then complete the renewal process.

Import and renew a certificate obtained directly from an Internet CA:

To import and renew a certificate that you obtained directly from an Internet CA follow these steps:

1. In the navigation frame, click **Select a Certificate Store**, then select the certificate store that holds the certificate you want to renew.

Note: Click on the "?" for any panel to answer any further questions you have about completing the panel.

2. In the navigation frame, select **Manage Certificates**.
3. In the navigation frame, click **Renew certificate**.
4. Select the certificate you want to renew and click **Renew**.
5. Select **VeriSign** or other **Internet Certificate Authority (CA)** and click **Continue**.
6. Select **No - Import the renewed signed certificate from an existing file**.
7. Complete the guided task to import the certificate. When you chose to renew the certificate directly with the issuing CA, that CA returns the renewed certificate to you in a file. Make sure that you specify the correct absolute path for the file where the certificate is stored on the server when you import the certificate. The file that contains the renewed certificate can be stored in any integrated file system (IFS) directory.
8. Click **OK** to finish the task.

Renew a certificate by creating a new public-private key pair and CSR for the certificate:

To renew a certificate with an Internet CA by creating a new public-private key pair and CSR for the certificate follow these steps

1. In the navigation frame, click **Select a Certificate Stor**e, then select the certificate store that holds the certificate that you want to renew.

   Note: Click on the "?" for any panel to answer any further questions you have about completing the panel.
2. In the navigation frame, select **Manage Certificates**.
3. In the navigation frame, click **Renew certificate**
4. Select the certificate you want to renew and click **Renew**.
5. Select **VeriSign** or other **Internet Certificate Authority (CA)** and click **Continue**.
6. Click **Yes - Create a new key pair for this certificate and click Continue**.
7. Complete the certificate identification form. You must change the New certificate label field, but any other fields can remain the same. Note: Click on the "?" for any panel to answer any further questions you have about completing the panel.
8. Click **OK** to finish the task.

Importing a certificate

Review this information to learn how you can use Digital Certificate Manager (DCM) to import certificates that are located in files on your server.

You can also import a certificate from another server instead of recreating the certificate on the current server. For example, on System A you used the local CA to create a certificate for your retail web application to use to initiate SSL connections. Your business has grown recently and you have installed a new System i model (System B) to host more instances of this very busy retail application. You want all instances of the retail application to use an identical certificate to identify them and initiate SSL connections. Consequently, you might decide to import both the local CA certificate and the server certificate from System A to System B rather than to use the local CA on System A to create a new, different certificate for System B to use.

Follow these steps to use DCM to import a certificate:

1. In the left-hand navigation pane, click **Select a Certificate Store** and select the certificate store that you want to import the certificate into. The certificate store that you import the certificate into must contain certificates that are the same type as the certificate that you exported on the other system. For example, if you are importing a server certificate (type) then import it into a certificate store that contains server certificates such as *SYSTEM or an Other System Certificate Store.

2. In the navigation frame, select **Manage Certificates**.
3. In the navigation frame, select **Import certificate**.
4. Select the type of certificate that you want to import and click **Continue**. The type of certificate that you are importing needs to be the same type of certificate that you exported. For example, if you exported a server certificate select to import a server certificate.

   Note: When DCM exports a certificate in pkcs12 format, the issuing CA is included in the exported certificate chain and is therefore imported automatically when the certificate itself is imported into the certificate store by DCM. However, if the certificate is not exported in pkcs12 format and you do not have the CA certificate in the certificate store to which you are importing, you need to import the issuing CA certificate before you can import the certificate.
5. Complete the guided task to import the certificate. When you import the certificate make sure that you specify the correct absolute path where the certificate is stored on the server.

Managing DCM

After you configure Digital Certificate Manager (DCM), there are a number of certificate management tasks that you will need to perform over time.

To learn how to use DCM to manage your digital certificates, review these topics:

Using a local CA to issue certificates for other System i models

Review this information to learn how to use a private local CA on one system to issue certificates for use on other System i models.

You may already be using a private local Certificate Authority (CA) on a system in your network. Now, you want to extend the use of this local CA to another system in your network. For example, you want your current local CA to issue a server or client certificate for an application on another system to use for SSL communications sessions. Or, you want to use certificates from your local CA on one system to sign objects that you store on another server.

You can accomplish this goal by using Digital Certificate Manager (DCM). You perform some of tasks on the system on which you operate the local CA and perform others on the secondary system that hosts the applications for which you want to issue certificates. This secondary system is called the target system. The tasks that you must perform on the target system depend on that system's release level.

Note: You can encounter a problem if the system on which you operate the local CA uses a cryptographic access provider product that provides stronger encryption than the target system. For OS/400 V5R2 and OS/400 V5R3 the only cryptographic access provider available is 5722–AC3, which is the strongest product available. However, in earlier releases, you were able to install other, weaker cryptographic access provider products (5722–AC1, or 5722–AC2) that provided lower levels of cryptographic function When you export the certificate (with its private key), the system encrypts the file to protect its contents. If the system uses a stronger cryptographic product than the target system, the target system cannot decrypt the file during the import process. Consequently, the import may fail or the certificate may not be usable for establishing SSL sessions. This is true even if you use a key size for the new certificate that is appropriate for use with the cryptographic product on the target system.

You can use your local CA to issue certificates to other systems, which you can then use for signing objects or have applications use for establishing SSL sessions. When you use the local CA to create a certificate for use on another system, the files that DCM creates contain a copy of the local CA certificate, as well as copies of certificates for many public Internet CAs.

The tasks that you must perform in DCM vary slightly depending on which type of certificate that your local CA issues and the release level and conditions on the target system.

Issue private certificates for use on another System i model

To use your local CA to issue certificates for use on another system, perform these steps on the system that hosts the local CA:

1. Start DCM. Refer to Starting DCM.
2. In the navigation frame, select **Create Certificate** to display a list of certificate types that you can use your local CA to create.

   Note: You do not need to open a certificate store to complete this task. These instructions assume either that you are not working within a specific certificate store or that you are working within the local Certificate Authority (CA) certificate store. A local CA must exist on this system before you can perform these tasks. If you have questions about how to complete a specific form in this guided task, select the question mark (**?**) at the top of the page to access the online help.
3. Select the type of certificate that you want the local CA to issue, and click **Continue** to start the guided task and complete a series of forms.
4. Select either to create a **server or client certificate for another System i** (for SSL sessions), or an **object signing certificate for another System i** (for use on another system).

   Note: If you are creating an object signing certificate for another system to use, that system must be running OS/400 V5R1 or later version to use the certificate. Because the target system must be at OS/400 V5R1 or later, DCM on the local host system does not prompt you to select a target release format for the new object signing certificate.
5. Complete the form and click **Continue** to display a confirmation page.

   Note: If there is an existing *OBJECTSIGNING or *SYSTEM certificate store on the target system, be sure to specify a unique certificate label and unique file name for the certificate. Specifying a unique certificate label and file name ensures that you can easily import the certificate into the existing certificate store on the target system. This confirmation page displays the names of the files that DCM created for you to transfer to the target system. DCM creates these files based on the release level of the target system that you specified. DCM automatically puts a copy of the local CA certificate into these files.

   DCM creates the new certificate in its own certificate store and generates two files for you to transfer: a certificate store file (`.KDB` extension) and a request file (`.RDB` extension).
6. Use binary File Transfer Protocol (FTP) or another method to transfer the files to the target system.

Related concepts

"Backup and recovery considerations for DCM data" on page 27
Use this information to learn how to ensure that important DCM data is added to your backup and recovery plan for your system.

"Public certificates versus private certificates" on page 29
Review this information to learn how to determine which type of certificate (public or private) best suits your business needs.

Related tasks

"Creating and operating a local CA" on page 39
This information explains how to create and operate a local Certificate Authority (CA) to issue private certificates for your applications.

Using a private certificate for SSL

You manage the certificates that your applications use for SSL sessions from the *SYSTEM certificate store in Digital Certificate Manager (DCM). If you have never used DCM on the target system to manage certificates for SSL, then this certificate store will not exist on the target system.

The tasks for using the transferred certificate store files that you created on the local Certificate Authority (CA) host system vary based on whether the *SYSTEM certificate store exists. If the *SYSTEM certificate store does not exist, you can use the transferred certificate files as a means of creating the *SYSTEM certificate store. If the *SYSTEM certificate store does exist on the target system, you can either use the transferred files as an Other System Certificate Store or import the transferred files into the existing *SYSTEM certificate store.

***SYSTEM certificate store does not exist:**
If the *SYSTEM certificate store does not exist on the system on which you want to use the transferred certificate store files, you can use the transferred certificate files as the *SYSTEM certificate store. To create the *SYSTEM certificate store and use the certificate files on your target system, follow these steps:

1. Make sure that the certificate store files (two files: one with a `.KDB` extension and one with a `.RDB` extension) that you created on the system that hosts the local CA are in the `/QIBM/USERDATA/ICSS/CERT/SERVER` directory.
2. Once the transferred certificate files are in the `/QIBM/USERDATA/ICSS/CERT/SERVER` directory, rename these files to `DEFAULT.KDB`, and `DEFAULT.RDB`. By renaming these files in the appropriate directory, you create the components that comprise the *SYSTEM certificate store for the target system. The certificate store files already contain copies of certificates for many public Internet CAs. DCM added these, as well as a copy of the local CA certificate, to the certificate store files when you created the them.

   Attention: If your target system already has a `DEFAULT.KDB` and a `DEFAULT.RDB` file in the `/QIBM/USERDATA/ICSS/CERT/SERVER` directory, the *SYSTEM certificate store currently exists on this target system. Consequently, you must not rename the transferred files as suggested. Overwriting the default files will create problems when using DCM, the transferred certificate store, and its contents. Instead, you must ensure that they have unique names and must use the transferred certificate store as an **Other System Certificate Store**. If you use the files as an Other System Certificate Store, you cannot use DCM to specify which applications will use the certificate.
3. Start DCM. You must now change the password for the *SYSTEM certificate store that you created by renaming the transferred files. Changing the password allows DCM to store the new password so that you can use all DCM certificate management functions on the certificate store.
4. In the navigation frame, click **Select a Certificate Store** and select ***SYSTEM** as the certificate store to open.
5. When the Certificate Store and Password page displays, provide the password that you specified on the *host* system for the certificate store when you created the certificate for the target system and click **Continue**.
6. In the navigation frame, select **Manage Certificate Store** and select **Change password** from the list of tasks. Complete the form to change the password for the certificate store. After you change the password, you must re-open the certificate store before you can work with the certificates in it. Next you can specify which applications will use the certificate for SSL sessions.
7. In the navigation frame, click **Select a Certificate Store** and select ***SYSTEM** as the certificate store to open.
8. When the **Certificate Store and Password** page displays, provide the new password and click **Continue**.
9. After the navigation frame refreshes, select **Manage Certificates** in the navigation frame to display a list of tasks.
10. From the task list, select **Assign certificate** to display a list of certificates in the current certificate store.
11. Select the certificate that you created on the *host* system and click **Assign to Applications** to display a list of SSL-enabled applications to which you can assign the certificate.
12. Select the applications that will use the certificate for SSL sessions and click **Continue**. DCM displays a message to confirm your certificate selection for the applications.

Note: Some SSL-enabled applications support client authentication based on certificates. An application with this support must to be able to authenticate certificates before providing access to resources. Consequently, you must define a CA trust list for the application. This ensures that the application can validate only those certificates from CAs that you specify as trusted. If users or a client application present a certificate from a CA that is not specified as trusted in the CA trust list, the application will not accept it as a basis for valid authentication.

With these tasks complete, applications on the target system can use the certificate issued by the local CA on another system. However, before you can begin using SSL for these applications, you must configure the applications to use SSL.

Before a user can access the selected applications through an SSL connection, the user must use DCM to obtain a copy of the local CA certificate from the host system. The local CA certificate must be copied to a file on the user's PC or downloaded into the user's browser, depending on the requirements of the SSL-enabled application.

***SYSTEM certificate store exists — using the files as an Other System Certificate:**

If the target system already has a *SYSTEM certificate store, you must decide how to work with the certificate files that you transferred to the target system. You can choose to use the transferred certificate files as an **Other System Certificate Store**. Or, you can choose to import the private certificate and its corresponding local CA certificate into the existing *SYSTEM certificate store.

Other System Certificate Stores are user-defined secondary certificate stores for SSL certificates. You can create and use them to provide certificates for user-written SSL-enabled applications that do not use DCM APIs to register an application ID with the DCM feature. The Other System Certificate Store option allows you to manage certificates for applications that you or others write that use the SSL_Init API to programmatically access and use a certificate to establish an SSL session. This API allows an application to use the default certificate for a certificate store rather than a certificate that you specifically identify.

IBM System i applications (and many other software developers' applications) are written to use certificates in the *SYSTEM certificate store only. If you choose to use the transferred files as an Other System Certificate Store, you cannot use DCM to specify which applications will use the certificate for SSL sessions. Consequently, you cannot configure standard System i SSL-enabled applications to use this certificate. If you want to use the certificate for System i applications, you must import the certificate from your transferred certificate store files into the *SYSTEM certificate store.

To access and work with the transferred certificate files as an Other System Certificate Store, follow these steps:

1. Start DCM.
2. In the navigation frame, click **Select a Certificate Store** and select **Other System Certificate Store** as the certificate store to open
3. When the Certificate Store and Password page displays, provide the fully qualified path and file name of the certificate store file (the one with the `.KDB` extension) that you transferred from the host system. Also provide the password that you specified on the *host* system for the certificate store when you created the certificate for the target system and click **Continue**.
4. In the navigation frame, select **Manage Certificate Store** and select **Change password** from the list of tasks. Complete the form to change the password for the certificate store.

   Note: Be sure to select the **Automatic login** option when you change the password for the certificate store. Using this option ensures that DCM stores the new password so that you can use all DCM certificate management functions on the new store.

   After you change the password, you must re-open the certificate store before you can work with the certificates in it. Next you can specify that the certificate in this store be used as the default certificate

5. In the navigation frame, click **Select a Certificate Store** and select **Other System Certificate Store** as the certificate store to open.
6. When the **Certificate Store and Password** page displays, provide the fully qualified path and file name of the certificate store file, provide the new password, and click **Continue**.
7. After the navigation frame refreshes, select **Manage Certificate Store** and select **Set default certificate** from the list of tasks.

Now that you have created and configured the Other System Certificate store, any applications that use the SSL_Init API can use the certificate in it to establish SSL sessions.

**SYSTEM certificate store exists — using the certificates in the existing *SYSTEM certificate store:*

You can use the certificates in the transferred certificate store files in an existing *SYSTEM certificate store on a system. To do so, you must import the certificates from the certificate store files into the existing *SYSTEM certificate store. However, you cannot import the certificates directly from the `.KDB` and `.RDB` files because they are not in a format that the DCM import function can recognize and use. To use the transferred certificates in an existing *SYSTEM certificate store, you must open the files as an Other System Certificate Store and export them into the *SYSTEM certificate store.

To export the certificates from the certificate store files into the *SYSTEM certificate store, complete these steps on the target system:

1. Start DCM.
2. In the navigation frame, click **Select a Certificate Store** and specify **Other System Certificate Store** as the certificate store to open.
3. When the Certificate Store and Password page displays, provide the fully qualified path and file name of the certificate store file (the one with the `.KDB` extension) that you transferred from the host system. Also provide the password that you specified on the *host* system for the certificate store when you created the certificate for the target system and click **Continue**.
4. In the navigation frame, select **Manage Certificate Store** and select **Change password** from the list of tasks. Complete the form to change the password for the certificate store. After you change the password, you must re-open the certificate store before you can work with the certificates in it.

   Note: Be sure to select the **Automatic login** option when you change the password for the certificate store. Using this option ensures that DCM stores the new password so that you can use all DCM certificate management functions on the new store. If you do not change the password and select the Automatic login option, you may encounter errors when exporting the certificates from this store into the *SYSTEM certificate store.
5. In the navigation frame, click **Select a Certificate Store** and select **Other System Certificate Store** as the certificate store to open.
6. When the **Certificate Store and Password** page displays, provide the fully qualified path and file name of the certificate store file, provide the new password, and click **Continue**.
7. After the navigation frame refreshes, select **Manage Certificates** in the navigation frame to display a list of tasks and select **Export certificate**.
8. Select **Certificate Authority (CA)** as the type of certificate to export and click **Continue**.

   Note: You must export the local CA certificate into the certificate store before you export the server or client certificate into the certificate store. If you export the server or client certificate first, you may encounter an error because the local CA certificate does not exist in the certificate store.
9. Select the local CA certificate to export and click **Export**.
10. Select **Certificate store** as the destination for the exported certificate and click **Continue**.

11. Enter *SYSTEM as the target certificate store, enter the password for the *SYSTEM certificate store, and click **Continue**. A message displays to indicate that the certificate exported successfully or to provide error information if the export process failed.
12. Now you can export the server or client certificate into the *SYSTEM certificate store. Re-select the **Export certificate** task.
13. Select **Server or client** as the type of certificate to export and click **Continue**.
14. Select the appropriate server or client certificate to export and click **Export**.
15. Select **Certificate store** as the destination for the exported certificate and click **Continue**.
16. Enter *SYSTEM as the target certificate store, enter the password for the *SYSTEM certificate store, and click **Continue**. A message displays to indicate that the certificate exported successfully or to provide error information if the export process failed.
17. Now you can assign the certificate to applications to use for SSL. Click **Select a Certificate Store** in the navigation frame and select ***SYSTEM** as the certificate store to open.
18. When the Certificate Store and Password page displays, provide the password for the *SYSTEM certificate store and click **Continue**.
19. After the navigation frame refreshes, select **Manage Certificates** to display a list of tasks.
20. From the task list, select **Assign certificate** to display a list of certificates in the current certificate store.
21. Select the certificate that you created on the *host* system and click **Assign to Applications** to display a list of SSL-enabled applications to which you can assign the certificate.
22. Select the applications that will use the certificate for SSL sessions and click **Continue**. DCM displays a message to confirm your certificate selection for the applications.

    Note: Some SSL-enabled applications support client authentication based on certificates. An application with this support must to be able to authenticate certificates before providing access to resources. Consequently, you must define a CA trust list for the application. This ensures that the application can validate only those certificates from CAs that you specify as trusted. If users or a client application present a certificate from a CA that is not specified as trusted in the CA trust list, the application will not accept it as a basis for valid authentication.

With these tasks complete, applications on the target system can use the certificate issued by the local CA on another system. However, before you can begin using SSL for these applications, you must configure the applications to use SSL.

Before a user can access the selected applications through an SSL connection, the user must use DCM to obtain a copy of the local CA certificate from the host system. The local CA certificate must be copied to a file on the user's PC or downloaded into the user's browser, depending on the requirements of the SSL-enabled application.

Using a private certificate for signing objects on a target system

You manage the certificates that you use for signing objects from the *OBJECTSIGNING certificate store in Digital Certificate Manager (DCM). If you have never used DCM on the target system to manage object signing certificates, then this certificate store will not exist on the target system.

The tasks that you must perform to use the transferred certificate store files that you created on the local CA host system vary based on whether the *OBJECTSIGNING certificate store exists. If the *OBJECTSIGNING certificate store does not exist, you can use the transferred certificate files as a means of creating the *OBJECTSIGNING certificate store. If the *OBJECTSIGNING certificate exists on the target system, you must import the transferred certificates into it.

***OBJECTSIGNING certificate store does not exist:**

The tasks that you perform to use the certificate store files that you created on the local CA host system vary based on whether you have ever used DCM on the target system to manage object signing certificates.

If the *OBJECTSIGNING certificate store does not exist on the target system with the transferred certificate store files, follow these steps:

1. Make sure that the certificate store files (two files: one with a `.KDB` extension and one with a `.RDB` extension) that you created on the system that hosts the local CA are in the `/QIBM/USERDATA/ICSS/CERT/SIGNING` directory.
2. Once the transferred certificate files are in the `/QIBM/USERDATA/ICSS/CERT/SIGNING` directory, rename the certificate files to `SGNOBJ.KDB`, and `SGNOBJ.RDB`, if necessary. By renaming these files, you create the components that comprise the *OBJECTSIGNING certificate store for the target system. The certificate store files already contain copies of certificates for many public Internet CAs. DCM added these, as well as a copy of the local CA certificate, to the certificate store files when you created them.

   Attention: If your target system already has a `SGNOBJ.KDB` and a `SGNOBJ.RDB` file in the `/QIBM/USERDATA/ICSS/CERT/SIGNING` directory, the *OBJECTSIGNING certificate store currently exists on this target system. Consequently, you must not rename the transferred files as suggested. Overwriting the default object signing files will create problems for using DCM, the transferred certificate store, and its contents. When the *OBJECTSIGNING certificate store already exists, you must use a different process to get the certificates into the existing certificate store.
3. Start DCM. You must now change the password for the *OBJECTSIGNING certificate store. Changing the password allows DCM to store the new password so that you can use all DCM certificate management functions on the certificate store.
4. In the navigation frame, click **Select a Certificate Store** and select ***OBJECTSIGNING** as the certificate store to open.
5. When the password page displays, provide the password that you specified for the certificate store when you created it on the host system and click **Continue**.
6. In the navigation frame, select **Manage Certificate Store** and select **Change password** from the list of tasks. Complete the form to change the password for the certificate store. After you change the password, you must re-open the certificate store before you can work with the certificates in it. Next you can create an application definition for using the certificate to sign objects.
7. After you re-open the certificate store, select **Manage Applications** in the navigation frame to display a list of tasks.
8. From the task list, select **Add application** to begin the process of creating an object signing application definition to use a certificate to sign objects.
9. Complete the form to define your object signing application and click **Add**. This application definition does not describe an actual application, but rather describes the type of objects that you plan to sign with a specific certificate. Use the online help to determine how to complete the form.
10. Click **OK** to acknowledge the application definition confirmation message and display the **Manage Applications** task list.
11. From the task list, select **Update certificate assignment** to display a list of object signing application IDs for which you can assign a certificate.
12. Select your application ID from the list and click **Update Certificate Assignment**.
13. Select the certificate that the local CA on the host system created and click **Assign New Certificate**.

When you finish these tasks, you have everything that you need to begin signing objects to ensure their integrity.

When you distribute signed objects, those who receive the objects must use DCM to verify the signature on the objects to ensure that the data is unchanged and to verify the identity of the sender. To validate

the signature, the receiver must have a copy of the signature verification certificate. You must provide a copy of this certificate as part of the package of signed objects.

The receiver also must have a copy of the CA certificate for the CA that issued the certificate that you used to sign the object. If you signed the objects with a certificate from a well-known Internet CA, the receiver's version of DCM will already have a copy of the necessary CA certificate. However, you must provide a copy of the CA certificate, in a separate package, along with the signed objects if necessary. For example, you must provide a copy of the local CA certificate if you signed the objects with a certificate from a local CA. For security reasons, you must provide the CA certificate in a separate package or publicly make the CA certificate available at the request of those who need it.

***OBJECTSIGNING certificate store exists:**

You can use the certificates in the transferred certificate store files in an existing *OBJECTSIGNING certificate store on a system. To do so, you must import the certificates from the certificate store files into the existing *OBJECTSIGNING certificate store. However, you cannot import the certificates directly from the `.KDB` and `.RDB` files because they are not in a format that the DCM import function can recognize and use. You can add the certificates into the existing *OBJECTSIGNING certificate store by opening the transferred files as an Other System Certificate Store on the target system. You can then export the certificates directly into the *OBJECTSIGNING certificate store. You must export a copy of both the object signing certificate itself and the local CA certificate from the transferred files.

To export the certificates from the certificate store files directly into the *OBJECTSIGNING certificate store, complete these steps on the target system:

1. Start DCM.
2. In the navigation frame, click **Select a Certificate Store** and specify **Other System Certificate Store** as the certificate store to open
3. When the Certificate Store and Password page displays, provide the fully qualified path and file name for the certificate store files. Also provide the password that you used when you created them on the host system and click **Continue**.
4. In the navigation frame, select **Manage Certificate Store** and select **Change password** from the list of tasks. Complete the form to change the password for the certificate store.

   Note: Be sure to select the **Automatic login** option when you change the password for the certificate store. Using this option ensures that DCM stores the new password so that you can use all DCM certificate management functions on the new store. If you do not change the password and select the Automatic login option, you may encounter errors when exporting the certificates from this store into the *OBJECTSIGNING certificate store.

   After you change the password, you must re-open the certificate store before you can work with the certificates in it.
5. In the navigation frame, click **Select a Certificate Store** and select **Other System Certificate Store** as the certificate store to open.
6. When the Certificate Store and Password page displays, provide the fully qualified path and file name of the certificate store file, provide the new password, and click **Continue**.
7. After the navigation frame refreshes, select **Manage Certificates** in the navigation frame to display a list of tasks and select **Export certificate**.
8. Select **Certificate Authority (CA)** as the type of certificate to export and click **Continue**.

   Note: The wording for this task assumes that when you work with an Other System Certificate Store that you are working with server or client certificates. This is because this type of certificate store is designed for use as a secondary certificate store to the *SYSTEM certificate store. However, using the export task in this certificate store is the easiest way to add the certificates from the transferred files into the existing *OBJECTSIGNING certificate store.
9. Select the local CA certificate to export and click **Export**.

Note: You must export the local CA certificate into the certificate store before you export the object signing certificate into the certificate store. If you export the object signing certificate first, you may encounter an error because the local CA certificate does not exist in the certificate store.

10. Select **Certificate store** as the destination for the exported certificate and click **Continue**.
11. Enter `*OBJECTSIGNING` as the target certificate store, enter the password for the *OBJECTSIGNING certificate store, and click **Continue**.
12. Now you can export the object signing certificate into the *OBJECTSIGNING certificate store. Re-select the **Export certificate** task.
13. Select **Server or client** as the type of certificate to export and click **Continue**.
14. Select the appropriate certificate to export and click **Export**.
15. Select **Certificate store** as the destination for the exported certificate and click **Continue**
16. Enter `*OBJECTSIGNING` as the target certificate store, enter the password for the *OBJECTSIGNING certificate store, and click **Continue**. A message displays to indicate that the certificate exported successfully or to provide error information if the export process failed.

    Note: To use this certificate to sign objects, you must now assign the certificate to an object signing application.

Managing applications in DCM

Digital Certificate Manager (DCM) allows you to create application definitions and manage an application's certificate assignment. You can also define CA trust lists that applications use as the basis of accepting certificates for client authentication.

You can use DCM, to perform various management tasks for Secure Sockets Layer (SSL) enabled applications and object signing applications. For example, you can manage which certificates your applications use for SSL communications sessions. The application management tasks that you can perform vary based on the type of application and the certificate store in which you are working. You can manage applications from the *SYSTEM or *OBJECTSIGNING certificate stores only.

While most application management tasks that DCM provides are easy to understand, a few of these tasks may not be familiar to you. For more information about these tasks, review these topics:

Related concepts

"Application definitions" on page 10
Digital Certificate Manager (DCM) allows you to manage application definitions that will work with SSL configurations and object signing.

Creating an application definition

Review this topic to learn how about the two different types of applications that you can define and work with.

There are two types of application definitions that you can work with in DCM: application definitions for server or client applications that use SSL and application definitions that you use for signing objects.

To use DCM to work with SSL application definitions and their certificates, the application must first be registered with DCM as an application definition so that it has a unique application ID. Application developers register SSL-enabled applications by using an API (QSYRGAP, QsyRegisterAppForCertUse) to create the application ID in DCM automatically. All IBM System i SSL-enabled applications are registered with DCM so that you can easily use DCM to assign a certificate to them so that they can establish an SSL session. Also, for applications that you write or purchase, you can define an application definition and create the application ID for it within DCM itself. You must be working in the *SYSTEM certificate store to create an SSL application definition for either a client application or a server application.

To use a certificate to sign objects, you first must define an application for the certificate to use. Unlike an SSL application definition, an object signing application does not describe an actual application. Instead, the application definition that you create might describe the type or group of objects that you intend to sign. You must be working in the *OBJECTSIGNING certificate store to create an object signing application definition.

To create an application definition, follow these steps:

1. Start DCM. Refer to Starting DCM.
2. Click **Select a Certificate Store** and select the appropriate certificate store. (This is either the *SYSTEM certificate store or the *OBJECTSIGNING certificate store depending on the type of application definition that you are creating.)

   Note: If you have questions about how to complete a specific form in this guided task, select the question mark (**?**) at the top of the page to access the online help.
3. When the Certificate Store and Password page displays, provide the password that you specified for the certificate store when you created it and click **Continue**.
4. In the navigation frame, select **Manage Applications** to display a list of tasks.
5. Select **Add application** from the task list to display a form for defining the application.

   Note: If you are working in the *SYSTEM certificate store, DCM will prompt you to choose whether to add a server application definition or a client application definition.
6. Complete the form and click **Add**. The information that you can specify for the application definition varies based on the type of application that you are defining. If you are defining a server application, you can also specify whether the application can use certificates for client authentication and must require client authentication. You can also specify that the application must use a CA trust list to authenticate certificates.

Related concepts

"Application definitions" on page 10
Digital Certificate Manager (DCM) allows you to manage application definitions that will work with SSL configurations and object signing.

Related information

QSYRGAP, QsyRegisterAppForCertUse API

Managing the certificate assignment for an application

You must use Digital Certificate Manager (DCM) to assign a certificate to an application before the application can perform a secure function, such as establishing a Secure Sockets Layer (SSL) session or signing an object.

To assign a certificate to an application, or to change the certificate assignment for an application, follow these steps:

1. Start DCM. Refer to Starting DCM.
2. Click **Select a Certificate Store** and select the appropriate certificate store. (This is either the *SYSTEM certificate store or the *OBJECTSIGNING certificate store depending on the type of application to which you are assigning a certificate.)

   Note: If you have questions about how to complete a specific form in this guided task, select the question mark (**?**) at the top of the page to access the online help.
3. When the Certificate Store and Password page displays, provide the password that you specified for the certificate store when you created it and click **Continue**.
4. In the navigation frame, select **Manage Applications** to display a list of tasks.
5. If you are in the *SYSTEM certificate store, select the type of application to manage. (Select either **Server** or **Client** application, as appropriate.)

6. From the task list, select **Update certificate assignment** to display a list of applications for which you can assign a certificate.
7. Select an application from the list and click **Update Certificate Assignment** to display a list of certificates that you can assign to the application.
8. Select a certificate from the list and click **Assign New Certificate**. DCM displays a message to confirm your certificate selection for the application.

   Note: If you are assigning a certificate to an SSL-enabled application that supports the use of certificates for client authentication, you must define a CA trust list for the application. This ensures that the application can validate only those certificates from CAs that you specify as trusted. If users or a client application presents a certificate from a CA that is not specified as trusted in the CA trust list, the application will not accept it as a basis for valid authentication.

When you change or remove a certificate for an application, the application may or may not be able to recognize the change if the application is running at the time you change the certificate assignment. For example, iSeries Access for Windows servers will apply any certificate changes that you make automatically. However, you may need to stop and start Telnet servers, the IBM HTTP Server for i5/OS, or other applications before these applications can apply your certificate changes.

In OS/400 V5R2 or later, you can use the Assign certificate task when you want to assign a certificate to several applications at once.

Defining a CA trust list for an application

Applications that support the use of certificates for client authentication during a Secure Sockets Layer (SSL) session must determine whether to accept a certificate as valid proof of identity. One of the criteria that an application uses for authenticating a certificate is whether the application trusts the Certificate Authority (CA) that issued the certificate.

You can use Digital Certificate Manager (DCM) to define which CAs an application can trust when performing client authentication for certificates. You manage the CAs that an application trusts through a CA trust list.

Before you can define a CA trust list for an application, several conditions must be met:

- The application must support the use of certificates for client authentication.
- The definition for the application must specify that the application use a CA trust list.

If the definition for an application specifies that the application use a CA trust list, you must define the list before the application can perform certificate client authentication successfully. This ensures that the application can validate only those certificates from CAs that you specify as trusted. If users or a client application present a certificate from a CA that is not specified as trusted in the CA trust list, the application will not accept it as a basis for valid authentication.

When you add a CA to the trust list for an application, you must ensure that the CA is enabled as well.

To define a CA trust list for an application, follow these steps:

1. Start DCM. Refer to Starting DCM.
2. Click **Select a Certificate Store** and select ***SYSTEM** as the certificate store to open.

   Note: If you have questions about how to complete a specific form in this guided task, select the question mark (**?**) at the top of the page to access the online help.
3. When the Certificate Store and Password page displays, provide the password that you specified for the certificate store when you created it and click **Continue**.
4. In the navigation frame, select **Manage Applications** to display a list of tasks.
5. From the task list, select **Define CA trust list**.

6. Select the type of application (server or client) for which you want to define the list and click **Continue**.
7. Select an application from the list and click **Continue** to display a list of CA certificates that you use to define the trust list.
8. Select the CAs that the application will trust and click **OK**. DCM displays a message to confirm your trust list selections.

   Note: You can either select individual CAs from the list or you can specify that the application will trust all or trust none of the CAs in the list. Also, you can view or validate the CA certificate before you add it to the trust list.

Related concepts

"Digital certificates for VPN connections" on page 35
Review this information to learn how to use certificates as part of configuring a Virtual Private Network (VPN) connection.

Managing certificates by expiration

Digital Certificate Manager (DCM) provides certificate expiration management support to allow administrators to manage server or client certificates, object signing certificates, and user certificates by expiration date on the local system.

Note: If you configure DCM to work with Enterprise Identity Mapping (EIM), you can manage user certificates by expiration date across the enterprise.

Using DCM to view certificates based on their expiration date allows you to determine quickly and easily which certificates are close to expiring so that certificates can be renewed in a timely fashion.

Note: Because you can use a signature verification certificate to verify object signatures even when the certificate is expired, DCM does not provide support for checking the expiration of these certificates.

To view and manage server and client certificates or object signing certificates based on their expiration dates, follow these steps:

1. Start DCM. Refer to Starting DCM.
2. In the navigation frame, click **Select a Certificate Store** and select either ***OBJECTSIGNING** or ***SYSTEM** as the certificate store to open.

   Note: If you have questions about how to complete a specific form while using DCM, select the question mark (**?**) at the top of the page to access the online help.
3. Enter the password for the certificate store and click **Continue**.
4. After the navigation frame refreshes, select **Manage Certificates** to display a list of tasks.
5. From the list of tasks, select **Check expiration**.
6. Select the type of certificate that you want to check. If you are in the *SYSTEM certificate store, select **Server or client**; if you are in the *OBJECTSIGNING certificate store, select **Object signing**.
7. In the **Expiration date range in days (1-365)** field, enter the number of days for which to view certificates based on their expiration date and click **Continue**. DCM displays all certificates that expire between today's date and the date that corresponds to the number of days that you specify. DCM also displays all certificates that have expiration dates before today's date.
8. Select a certificate that you want to manage. You can choose to view certificate information details, delete the certificate, or renew the certificate.
9. When you finish working with certificates from the list, click **Cancel** to exit.

Related tasks

"Managing user certificates by expiration" on page 44
Digital Certificate Manager (DCM) provides certificate expiration management support to allow

administrators to check the expiration dates of user certificates on the local System i model. DCM user certificate expiration management support can be used in conjunction with Enterprise Identity Mapping (EIM) so that administrators can use DCM to check user certificate expiration at the enterprise level.

Validating certificates and applications

You can use Digital Certificate Manager (DCM) to validate individual certificates or the applications that use them. The list of things that DCM checks differs slightly depending on whether you are validating a certificate or an application.

Application validation

Using DCM to validate an application definition helps prevent certificate problems for the application when it is performing a function that requires certificates. Such problems might prevent an application either from participating successfully in a Secure Sockets Layer (SSL) session or from signing objects successfully.

When you validate an application, DCM verifies that there is a certificate assignment for the application and ensures that the assigned certificate is valid. Additionally, DCM ensures that if the application is configured to use a Certificate Authority (CA) trust list, that the trust list contains at least one CA certificate. DCM then verifies that the CA certificates in the application CA trust list are valid. Also, if the application definition specifies that Certificate Revocation List (CRL) processing occur and there is a defined CRL location for the CA, DCM checks the CRL as part of the validation process.

Certificate validation

When you validate a certificate, DCM verifies a number of items pertaining to the certificate to ensure the authenticity and validity of the certificate. Validating a certificate ensures that applications that use the certificate for secure communications or for signing objects are unlikely to encounter problems when using the certificate.

As part of the validation process, DCM checks that the selected certificate is not expired. DCM also checks that the certificate is not listed in a Certificate Revocation List (CRL) as revoked, if a CRL location exists for the CA that issued the certificate. In addition, DCM checks that the CA certificate for the issuing CA is in the current certificate store and that the CA certificate is enabled and therefore trusted. If the certificate has a private key (for example, server, client, and object signing certificates), then DCM also validates the public-private key pair to ensure that the public-private key pair match. In other words, DCM encrypts data with the public key and then ensures that the data can be decrypted with the private key.

Related concepts

"Certificate Revocation List (CRL) Locations" on page 7
A Certificate Revocation List (CRL) is a file that lists all invalid and revoked certificates for a specific Certificate Authority (CA).

"Validation" on page 11
Digital Certificate Manager (DCM) provides tasks that allow you to validate a certificate or to validate an application to verify various properties that they each must have.

Assigning a certificate to applications

Digital Certificate Manager (DCM) allows you to assign a certificate quickly and easily to multiple applications. You can assign a certificate to multiple applications in the *SYSTEM or *OBJECTSIGNING certificate stores only.

To make a certificate assignment for one or more applications, follow these steps:

1. Start DCM. Refer to Starting DCM.

2. In the navigation frame, click **Select a Certificate Store** and select either ***OBJECTSIGNING** or ***SYSTEM** as the certificate store to open.

   Note: If you have questions about how to complete a specific form while using DCM, select the question mark (**?**) at the top of the page to access the online help.
3. Enter the password for the certificate store and click **Continue**.
4. After the navigation frame refreshes, select **Manage Certificates** to display a list of tasks.
5. From the list of tasks, select **Assign certificate** to display a list of certificates for the current certificate store.
6. Select a certificate from the list and click **Assign to Applications** to display a list of application definitions for the current certificate store.
7. Select one or more applications from the list and click **Continue**. A page displays with either a confirmation message for your assignment selection or an error message if a problem occurred.

Managing CRL locations

Digital Certificate Manager (DCM) allows you to define and manage Certificate Revocation List (CRL) location information for a specific Certificate Authority (CA) to use as part of the certificate validation process.

DCM, or an application that requires CRL processing, can use the CRL to determine that the CA that issued a specific certificate has not revoked the certificate. When you define a CRL location for a specific CA, applications that support the use of certificates for client authentication can access the CRL.

Applications that support the use of certificates for client authentication can perform CRL processing to ensure more stringent authentication for certificates that they accept as valid proof of identity. Before an application can use a defined CRL as part of the certificate validation process, the DCM application definition must require that the application perform CRL processing.

How CRL processing works

When you use DCM to validate a certificate or application, DCM performs CRL processing by default as part of the validation process. If there is no CRL location defined for the CA that issued the certificate that you are validating, DCM cannot perform CRL checking. However, DCM can attempt to validate other important information about the certificate, such as that the CA signature on the specific certificate is valid and that the CA that issued it is trusted.

Define a CRL location

To define a CRL location for a specific CA, follow these steps:

1. Start DCM. Refer to Starting DCM.
2. In the navigation frame, select **Manage CRL Locations** to display a list of tasks.

   Note: If you have questions about how to complete a specific form in this guided task, select the question mark (**?**) at the top of the page to access the online help.
3. Select **Add CRL location** from the task list to display a form that you can use to describe the CRL location and how DCM or the application will access the location.
4. Complete the form and click **OK**. You must give the CRL location a unique name, identify the LDAP server that hosts the CRL, and provide connection information that describes how to access the LDAP server. Now you need to associate the CRL location definition with a specific CA
5. In the navigation frame, select **Manage Certificates** to display a list of tasks.
6. Select **Update CRL location assignment** from the task list to display a list of CA certificates.

7. Select the CA certificate from the list to which you want to assign the CRL location definition that you created and click **Update CRL Location Assignment**. A list of CRL locations displays.
8. Select the CRL location from the list that you want to associate with the CA and click **Update Assignment**. A message displays at the top of the page to indicate that the CRL location has been assigned to the Certificate Authority (CA) certificate.

Note: To anonymously bind to an LDAP server for CRL processing, you must use the Directory Server Web Administration Tool and select the ″Manage schema″ task to change the security class (also referred to as ″access class″) of the certificateRevocationList and authorityRevocationList attributes from ″critical″ to ″normal″, and leave both the **Login distinguished name** field and the **Password** field blank.

Having defined a location for a CRL for a specific CA, DCM or other applications can use it when performing CRL processing. However, before CRL processing can work, the Directory Services server must contain the appropriate CRL. Also, you must configure both the Directory Server (LDAP) and client applications to use SSL, and assign a certificate to the applications in DCM.

Related concepts

“Certificate Revocation List (CRL) Locations” on page 7
A Certificate Revocation List (CRL) is a file that lists all invalid and revoked certificates for a specific Certificate Authority (CA).

Related information

IBM Directory Server for iSeries (LDAP)

Enable SSL on the Directory Server

Storing certificate keys on an IBM Cryptographic Coprocessor

Review this information to learn how to use an installed coprocessor to provide more secure storage for your certificates’ private keys.

If you have installed an IBM Cryptographic Coprocessor on your system, you can use the coprocessor to provide more secure storage for a certificate’s private key. You can use the coprocessor to store the private key for a server certificate, a client certificate, or a local Certificate Authority (CA) certificate. However, you cannot use the coprocessor for storing a user certificate private key because this key must be stored on the user’s system. Also, you cannot use the coprocessor to store the private key for an object signing certificate at this time.

You can use the coprocessor for certificate private key storage in one of two ways:

- Storing the certificate private key directly on the coprocessor itself.
- Using the coprocessor master key to encrypt the certificate private key for storage in a special key file.

You can select this key storage option as part of the process of creating or renewing a certificate. Also, if you use the coprocessor to store a certificate’s private key, you can change the coprocessor device assignment for that key.

To use the coprocessor for private key storage, you must ensure that the coprocessor is varied on before using Digital Certificate Manager (DCM). Otherwise, DCM will not provide a page for selecting a storage option as part of the certificate creation or renewal process.

If you are creating or renewing a server or client certificate, you select the private key storage option after you select the type of CA that is signing the current certificate. If you are creating or renewing a local CA, you select the private key storage option as the first step in the process.

Related concepts

"IBM Cryptographic Coprocessors for System i" on page 9
The cryptographic coprocessor provides proven cryptographic services, ensuring privacy and integrity, for developing secure e-business applications.

Related information

Cryptography overview

Using the coprocessor master key to encrypt the certificate private key

For extra security to protect access to and use of a certificate's private key, you can use the master key of an IBM Cryptographic Coprocessor to encrypt the private key and store the key in a special key file. You can select this key storage option as part of creating or renewing a certificate in Digital Certificate Manager (DCM).

Before you can use this option successfully, you must use the IBM Cryptographic Coprocessor configuration Web interface to create an appropriate keystore file. Also, you must use the coprocessor configuration Web interface to associate the keystore file with the coprocessor device description that you want to use. You can access the coprocessor configuration Web interface from the System i Tasks page.

If your system has more than one coprocessor device installed and varied on, you can choose to share the certificate's private key among multiple devices. In order for device descriptions to share the private key, all of the devices must have the same master key. The process for distributing the same master key to multiple devices is called *cloning*. Sharing the key among devices allows you to use Secure Sockets Layer (SSL) load balancing, which can improve performance for secure sessions.

Follow these steps from the **Select a Key Storage Location** page to use the coprocessor master key to encrypt the certificate's private key and store it in a special keystore file:

1. Select **Hardware encrypted** as your storage option.
2. Click **Continue**. This displays the **Select a Cryptographic Device Description** page.
3. From the list of devices, select the one that you want to use for encrypting the certificate's private key.
4. Click **Continue**. If you have more than one coprocessor device installed and varied on, the **Select Additional Cryptographic Device Descriptions** page displays.

   Note: If you do not have multiple coprocessor devices available, DCM continues to display pages for the task that you are completing, such as identifying information for the certificate that you are creating or renewing.
5. From the list of devices, select the name of one or more device descriptions with which you want to share the certificate's private key.

   Note: The device descriptions that you select must have the same master key as the device you selected on the previous page. To verify that the master key is the same on the devices, use the Master Key Verification task in the 4758 Cryptographic Coprocessor Configuration Web interface. You can access the coprocessor configuration Web interface from the System i Tasks page.
6. Click **Continue**. DCM continues to display pages for the task that you are completing, such as identifying information for the certificate that you are creating or renewing.

Related information

Cryptography overview

Managing the request location for a PKIX CA

A Public Key Infrastructure for X.509 (PKIX) Certificate Authority (CA) is a CA that issues certificates based on the newest Internet X.509 standards for implementing a public key infrastructure.

A PKIX CA requires more stringent identification before issuing a certificate; usually by requiring that an applicant provide proof of identity through a Registration Authority (RA). After the applicant supplies

the proof of identity that the RA requires, the RA certifies the applicant's identity. Either the RA or the applicant, depending on the CAs established procedure, submits the certified application to the associated CA. As these standards are adopted more widely, PKIX compliant CAs will become more widely available. You might investigate using a PKIX compliant CA if your security needs require strict access control to resources that your SSL-enabled applications provide to users. For example, Lotus® Domino® provides a PKIX CA for public use.

If you choose to have a PKIX CA issue certificates for your applications to use, you can use Digital Certificate Manager (DCM) to manage these certificates. You use DCM to configure a URL for a PKIX CA. Doing so configures Digital Certificate Manager (DCM) to provide a PKIX CA as an option for obtaining signed certificates.

To use DCM to manage certificates from a PKIX CA, you must configure DCM to use the location for the CA by following these steps:

1. Start DCM. Refer to Starting DCM.
2. In the navigation frame, select **Manage PKIX Request Location** to display a form that allows you to specify the URL for the PKIX CA or its associated RA.
3. Enter the fully qualified URL for the PKIX CA that you want to use for requesting a certificate; for example: `http://www.thawte.com` and click **Add**. Adding the URL configures DCM to add the PKIX CA as an option for obtaining signed certificates.

After you add a PKIX CA request location, DCM adds PKIX CA as an option for specifying the type of CA that you can choose for issuing a certificate when using the **Create Certificate** task.

Note: PKIX standards are outlined in Request For Comments (RFC) 2560.

Related concepts

"Managing certificates from a public Internet CA" on page 47
Review this information to learn how to manage certificates from a public Internet CA by creating a certificate store.

Managing LDAP location for user certificates

Review this information to learn how to configure DCM to store user certificates in a Lightweight Directory Access Protocol (LDAP) server directory location to extend Enterprise Identity Mapping to work with user certificates.

By default, Digital Certificate Manager (DCM) stores the user certificates that the local Certificate Authority (CA) issues with i5/OS user profiles. However, you can configure Digital Certificate Manager (DCM) in conjunction with Enterprise Identity Mapping (EIM) so that when the local Certificate Authority (CA) issues user certificates, the public copy of the certificate is stored in a specific Lightweight Directory Access Protocol (LDAP) server directory location. A combined configuration of EIM with DCM allows you to store user certificates in an LDAP directory location to make the certificates more readily available to other applications. This combined configuration also allows you to use EIM to manage user certificates as a type of user identity within your enterprise.

Note: If you want a user to store a certificate from a different CA in the LDAP location, the user must complete the **Assign a user certificate** task.

EIM is an @server technology that allows you to manage user identities in your enterprise, including i5/OS user profiles and user certificates. If you want to use EIM to manage user certificates, you need to perform these EIM configuration tasks before performing any DCM configuration tasks:

1. Use the **EIM Configuration** wizard in iSeries Navigator to configure EIM.
2. Create the X.509 registry in the EIM domain to be used for certificate associations

3. Select the Properties menu option for the Configuration folder in the EIM domain and enter the X.509 registry name.
4. Create an EIM identifier for each user that you want to have participate in EIM.
5. Create a target association between each EIM identifier and that user's user profile in the local i5/OS user registry. Use the EIM registry definition name for the local i5/OS user registry that you specified in the **EIM Configuration** wizard.

After you complete the necessary EIM configuration tasks, you must perform the following tasks to finish the overall configuration for using EIM and DCM together:

1. In DCM, use the **Manage LDAP Location** task to specify the LDAP directory that DCM will use to store a user certificate that the local CA creates. The LDAP location does not need to be on the local System i model, nor does it need to be the same LDAP server that EIM uses. When you configure the LDAP location in DCM, DCM uses the specified LDAP directory to store all user certificates that the local CA issues. DCM also uses the LDAP location to store user certificates processed by the **Assign a user certificate** task instead of storing the certificate with a user profile.
2. Run the **Convert User Certificates** (CVTUSRCERT) command. This command copies existing user certificates into the appropriate LDAP directory location. However, the command only copies certificates for a user that has had a target association created between an EIM identifier and the user profile. The command then creates a source association between each certificate and the associated EIM identifier. The command uses the certificate's subject distinguished name (DN), issuer DN, and a hash of these DNs along with the certificate's public key to define the user identity name for the source association.

Note: To anonymously bind to an LDAP server for CRL processing, you must use the Directory Server Web Administration Tool and select the ″Manage schema″ task to change the security class (also referred to as ″access class″) of the certificateRevocationList and authorityRevocationList attributes from ″critical″ to ″normal″, and leave both the **Login distinguished name** field and the **Password** field blank.

Related tasks

“Digital certificates and Enterprise Identity Mapping (EIM)” on page 34
Using Enterprise Identity Mapping (EIM) and Digital Certificate Mangers (DCM) together allows you to apply a certificate as the source of an EIM mapping lookup operation to map from the certificate to a target user identity associated with the same EIM identifier.

Related information

Convert User Certificate (CVTUSRCERT) command

Enterprise Identity Mapping (EIM)

Signing objects

Use this information to learn how to use DCM to manage certificates that you use to digitally sign objects to ensure their integrity.

There are three methods that you can use for signing objects. You can write a program that calls the Sign Object API. You can use Digital Certificate Manager (DCM) to sign objects. In OS/400 V5R2 or later, you can use iSeries Navigator Management Central feature to sign objects as you package them for distribution to other systems.

You can use the certificates that you manage in DCM to sign any object that you store in the system's integrated file system, except objects that are stored in a library. You can sign only these objects that are stored in the QSYS.LIB file system: *PGM, *SRVPGM, *MODULE, *SQLPKG, and *FILE (save file only). In OS/400 V5R2 or later, you can also sign command (*CMD) objects. You cannot sign objects that are stored on other systems.

You can sign objects with certificates that you purchase from a public Internet Certificate Authority (CA) or that you create with a private, local CA in DCM. The process of signing certificates is the same, regardless of whether you use public or private certificates.

Object signing prerequisites

Before you can use DCM (or the Sign Object API) to sign objects, you must ensure that certain prerequisite conditions are met:

- You must have created the *OBJECTSIGNING certificate store, either as part of the process of creating a local CA or as part of the process of managing object signing certificates from a public Internet CA.
- The *OBJECTSIGNING certificate store must contain at least one certificate, either one that you created by using the local CA or one that you obtained from a public Internet CA.
- You must have created an object signing application definition to use for signing objects.
- You must have assigned a certificate to the object signing application that you plan to use to sign objects.

Use DCM to sign objects

To use DCM to sign one or more objects, follow these steps:

1. Start DCM. Refer to Starting DCM.
2. In the navigation frame, click **Select a Certificate Store** and select ***OBJECTSIGNING** as the certificate store to open.

   Note: If you have questions about how to complete a specific form while using DCM, select the question mark (**?**) at the top of the page to access the online help.
3. Enter the password for the *OBJECTSIGNING certificate store and click **Continue**.
4. After the navigation frame refreshes, select **Manage Signable Objects** to display a list of tasks.
5. From the list of tasks, select **Sign an object** to display a list of application definitions that you can use for signing objects.
6. Select an application and click **Sign an Object** to view a form for specifying the location of the objects that you want to sign.

   Note: If the application that you select does not have a certificate assigned to it, you cannot use it to sign an object. You must first use the **Update certificate assignment** task under **Manage Applications** to assign a certificate to the application definition.
7. In the field provided, enter the fully qualified path and file name of the object or directory of objects that you want to sign and click **Continue**. Or, enter a directory location and click **Browse** to view the contents of the directory to select objects for signing.

   Note: You must start the object name with a leading slash or you may encounter an error. You can also use certain wildcard characters to describe the part of the directory that you want to sign. These wildcard characters are the asterisk (*), which specifies ″any number of characters,″ and the question mark (**?**), which specifies ″any single character.″ For example, to sign all the objects in a specific directory, you can enter `/mydirectory/*`; to sign all the programs in a specific library, you might enter `/QSYS.LIB/QGPL.LIB/*.PGM`. You can use these wildcard characters only in the last part of the path name; for example, `/mydirectory*/filename` results in an error message. If you want to use the Browse function to see a list of library or directory contents, you must enter the wildcard as part of the path name before clicking **Browse**.
8. Select the processing options that you want to use for signing the selected object or objects and click **Continue**.

   Note: If you choose to wait for job results, the results file displays directly in your browser. Results for the current job are appended to the end of the results file. Consequently, the file may

contain results from any previous jobs, in addition to those of the current job. You can use the date field in the file to determine which lines in the file apply to the current job. The date field is in YYYYMMDD format. The first field in the file can be either the message ID (if an error occurred during processing the object) or the date field (indicating the date on which the job processed).

9. Specify the fully qualified path and file name to use for storing job results for the object signing operation and click **Continue**. Or, enter a directory location and click **Browse** to view the contents of the directory to select a file for storing the job results. A message displays to indicate that the job was submitted to sign objects. To view the job results, see job **QOBJSGNBAT** in the job log.

Verifying object signatures

You can use Digital Certificate Manager (DCM) to verify the authenticity of digital signatures on objects. When you verify the signature, you ensure that the data in the object has not been changed since the object owner signed the object.

Signature verification prerequisites

Before you can use DCM to verify signatures on objects, you must ensure that certain prerequisite conditions are met:

- You must have created the *SIGNATUREVERIFICATION certificate store to manage your signature verification certificates.

  Note: You can perform signature verification while working within the *OBJECTSIGNING certificate store in cases where you are verifying signatures for objects that were signed on the same system. The steps that you perform to verify the signature in DCM are the same in either certificate store. However, the *SIGNATUREVERIFICATION certificate store must exist and must contain a copy of the certificate that signed the object even if you perform signature verification while working within the *OBJECTSIGNING certificate store.
- The *SIGNATUREVERIFICATION certificate store must contain a copy of the certificate that signed the objects.
- The *SIGNATUREVERIFICATION certificate store must contain a copy of the CA certificate that issued the certificate that signed the objects.

Use DCM to verify signatures on objects

To use DCM to verify object signatures, follow these steps:

1. Start DCM. Refer to Starting DCM.
2. In the navigation frame, click **Select a Certificate Store** and select ***SIGNATUREVERIFICATION** as the certificate store to open.

   Note: If you have questions about how to complete a specific form while using DCM, select the question mark (**?**) at the top of the page to access the online help.
3. Enter the password for the *SIGNATUREVERIFICATION certificate store and click **Continue**.
4. After the navigation frame refreshes, select **Manage Signable Objects** to display a list of tasks.
5. From the list of tasks, select **Verify object signature** to specify the location of the objects for which you want to verify signatures.
6. In the field provided, enter the fully qualified path and file name of the object or directory of objects for which you want to verify signatures and click **Continue**. Or, enter a directory location and click **Browse** to view the contents of the directory to select objects for signature verification.

   Note: You can also use certain wildcard characters to describe the part of the directory that you want to verify. These wildcard characters are the asterisk (*****), which specifies ″any number of characters,″ and the question mark (**?**), which specifies ″any single character.″ For example, to

sign all the objects in a specific directory, you might enter /mydirectory/*; to sign all the programs in a specific library, you might enter /QSYS.LIB/QGPL.LIB/*.PGM. You can use these wildcard characters only in the last part of the path name; for example, /mydirectory*/filename results in an error message. If you want to use the Browse function to see a list of library or directory contents, you must enter the wildcard as part of the path name before clicking **Browse**.

7. Select the processing options that you want to use for verifying the signature on the selected object or objects and click **Continue**.

   Note: If you choose to wait for job results, the results file displays directly in your browser. Results for the current job are appended to the end of the results file. Consequently, the file may contain results from any previous jobs, in addition to those of the current job. You can use the date field in the file to determine which lines in the file apply to the current job. The date field is in YYYYMMDD format. The first field in the file can be either the message ID (if an error occurred during processing the object) or the date field (indicating the date on which the job processed).

8. Specify the fully qualified path and file name to use for storing job results for the signature verification operation and click **Continue**. Or, enter a directory location and click **Browse** to view the contents of the directory to select a file for storing the job results. A message displays to indicate that the job was submitted to verify object signatures. To view the job results, see job **QOBJSGNBAT** in the job log.

You can also, use DCM to view information about the certificate that signed an object. This allows you to determine whether the object is from a source that you trust before you work with the object.

Related concepts

"Digital certificates for signing objects" on page 36
Use this information to learn how to use certificates to ensure an object's integrity or to verify the digital signature on an object to verify its authenticity.

Related tasks

"Managing public Internet certificates for signing objects" on page 49
You can use Digital Certificate Manager (DCM) to manage public Internet certificates to digitally sign objects.

"Managing certificates for verifying object signatures" on page 51
You can use Digital Certificate Manager (DCM) to manage the signature verification certificates that you use to validate digital signatures on objects.

Troubleshooting DCM

Review this information to learn how to resolve some of the more common errors that you may experience when using Digital Certificate Manager (DCM).

When you work with DCM and certificates, you may encounter errors that prevent you from accomplishing your tasks and goals. Many of the common errors or problems that you may experience fall into a number of categories, such as the following:

Troubleshooting passwords and general problems

Review this information to help you troubleshoot some of the more common password and other general problems that you may encounter while working with Digital Certificate Manager (DCM).

Problem	Possible Solution
You cannot find additional help for DCM.	In DCM, click the "?" help icon. You can also search the i5/OS Information Center and external IBM web sites on the Internet.

Problem	Possible Solution
Your password for the local Certificate Authority (CA) and *SYSTEM certificate stores do not work.	Passwords are case sensitive. Be sure the caps lock is the same as it was when you assigned the password.
You receive an error message that your password has expired when you attempt to open a certificate store.	You must change the password for the certificate store. Click the **OK** button to change the password.
Your attempt to reset the password when you used the **Select a Certificate Store** task failed.	The reset function works only if DCM has stored the password. DCM stores the password automatically when you create a certificate store. However, if you change (or reset) the password for an Other System Certificate Store, then you must select the **Automatic login** option so that DCM continues to stash the password.
	Also, if you move a certificate store from one system to another, you must change the password for the certificate store on the new system to ensure that DCM stashes it automatically. To change the password, you must supply the original password for the certificate store when you open it on the new system. You cannot use the reset password option until you have opened the store with the original password and changed the password to stash it. If the password is not changed and stashed, DCM and SSL cannot automatically recover the password when it is needed for various functions. If you are moving a certificate store that you will use as an Other System Certificate Store, you must select the **Automatic login** option when you change the password to ensure that DCM stashes the new password for this type of certificate store.
	Check the value assigned to the **Allow new digital certificates** attribute under the **Work with system security** option of the System Service Tools (SST). If this attribute is set to a value of 2 (No), then the certificate store password cannot be reset. You can view or change the value for this attribute by using the STRSST command and entering the Service Tools user ID and password. Then choose the **Work with system security** option. The Service Tools user ID is probably the QSECOFR user ID.
You cannot find a source for a CA certificate to receive it into your system.	Some CAs do not make their CA certificate readily available. If you cannot get the CA certificate from the CA, then contact your VAR since your VAR may have made special or monetary arrangements with the CA.
You cannot find the *SYSTEM certificate store.	The file location of the *SYSTEM certificate must be `/qibm/userdata/icss/cert/server/default.kdb`. If that certificate store does not exist you need to use DCM to create the certificate store. Use the **Create New Certificate Store** task.
You received an error from DCM, and the error continues to appear after you have fixed it.	Clear your browser cache. Set the cache size to 0, and end and restart the browser.
You have a Directory Server (LDAP) problem such as certificate assignments not being shown when the information about the secure application is displayed immediately after assigning a certificate. This problem occurs more often when using iSeries Navigator to get to a Netscape Communications browser. Your preference for the browser cache is set to compare the document in cache to the document on the network **Once per session**.	Change your default preference to check the caching every time.

Problem	Possible Solution
When you use DCM to import a certificate signed by an external CA such as Entrust, you receive an error message that the validity period does not contain today or does not fall within its issuer's validity period.	The system is using Generalized Time format for the validity period. Wait a day and try again. Also, verify that your system has the correct value for UTC offset (`dspsysval qutcoffset`). If you observe Daylight Savings Time, your offset might be incorrectly set.
You received a base 64 error when trying to import an Entrust certificate.	The certificate is listed as being a specific format such as PEM format. If the copy function of your browser does not work well you may copy extra material that does not belong with the certificate, such as blank spaces at the front of each line. If this is the case, then the certificate will not be the right format when you try to use it on the system. Some Web page designs cause this problem. Other Web pages are designed to avoid this problem. Be sure to compare the appearance of the original certificate to the results of the paste, since the pasted information must look the same.

Troubleshooting certificate store and key database problems

Review this information to help you troubleshoot some of the more common certificate store and key database problems you may encounter while working with Digital Certificate Manager (DCM).

Problem	Possible Solution
The system has not found the key database, or has found it to be invalid.	Check your password and file name for typographical errors. Be sure that the path is included with the file name, including the leading forward slash.

Problem	Possible Solution
Key database creation failed or Create a local CA creation fails.	Check for a file name conflict. The conflict may exist in a different file than the one for which you asked. DCM attempts to protect user data in the directories that it creates, even if those files keep DCM from successfully creating files when it needs to. Resolve this by copying all of the conflicting files to a different directory and, if possible, use DCM functions to delete the corresponding files. If you cannot use DCM to accomplish this, manually delete the files from the original integrated file system directory where they were conflicting with DCM. Ensure that you record exactly which files you move and where you move them. The copies allow you to recover the files if you find that you still need them. You need to create a new local CA after moving the following files: /QIBM/USERDATA/ICSS/CERT/CERTAUTH/DEFAULT.KDB /QIBM/USERDATA/ICSS/CERT/CERTAUTH/DEFAULT.TEMP.KDB /QIBM/USERDATA/ICSS/CERT/CERTAUTH/DEFAULT.RDB /QIBM/USERDATA/ICSS/CERT/CERTAUTH/DEFAULT.STH /QIBM/USERDATA/ICSS/CERT/CERTAUTH/DEFAULT.STH .OLD /QIBM/USERDATA/ICSS/CERT/CERTAUTH/DEFAULT.KYR /QIBM/USERDATA/ICSS/CERT/CERTAUTH/DEFAULT.POL /QIBM/USERDATA/ICSS/CERT/CERTAUTH/DEFAULT.BAK /QIBM/USERDATA/ICSS/CERT/CERTAUTH/DEFAULT.TEMP /QIBM/USERDATA/ICSS/CERT/CERTAUTH/DEFAULT.STHBAK /QIBM/USERDATA/ICSS/CERT/CERTAUTH/DEFAULT.TEMP.STH /QIBM/USERDATA/ICSS/CERT/CERTAUTH/CA.TXT /QIBM/USERDATA/ICSS/CERT/CERTAUTH/CA.BAK /QIBM/USERDATA/ICSS/CERT/CERTAUTH/CA.TMP /QIBM/USERDATA/ICSS/CERT/CERTAUTH/DEFAULT.POLTMP /QIBM/USERDATA/ICSS/CERT/CERTAUTH/DEFAULT.POLBAK /QIBM/USERDATA/ICSS/CERT/DOWNLOAD/CERTAUTH/CA.CACRT /QIBM/USERDATA/ICSS/CERT/DOWNLOAD/CERTAUTH/CA.CATMP /QIBM/USERDATA/ICSS/CERT/DOWNLOAD/CERTAUTH/CA.CABAK /QIBM/USERDATA/ICSS/CERT/DOWNLOAD/CLIENT/*.USRCRT You need to create a new *SYSTEM certificate store and system certificate after moving the following files: /QIBM/USERDATA/ICSS/CERT/SERVER/DEFAULT.KDB /QIBM/USERDATA/ICSS/CERT/SERVER/DEFAULT.BAK /QIBM/USERDATA/ICSS/CERT/SERVER/DEFAULT.RDB /QIBM/USERDATA/ICSS/CERT/SERVER/DEFAULT.STH /QIBM/USERDATA/ICSS/CERT/SERVER/DEFAULT.STH.OLD /QIBM/USERDATA/ICSS/CERT/SERVER/DEFAULT.STHBAK /QIBM/USERDATA/ICSS/CERT/SERVER/DEFAULT.TMP /QIBM/USERDATA/ICSS/CERT/SERVER/DEFAULT.TEMP.STH /QIBM/USERDATA/ICSS/CERT/SERVER/SRV.TMP /QIBM/USERDATA/ICSS/CERT/SERVER/SRV.BAK /QIBM/USERDATA/ICSS/CERT/SERVER/SRV.TXT /QIBM/USERDATA/ICSS/CERT/SERVER/SRV.SGN /QIBM/USERDATA/ICSS/CERT/SERVER/SGN.TMP /QIBM/USERDATA/ICSS/CERT/SERVER/SGN.BAK /QIBM/USERDATA/ICSS/CERT/SERVER/EXPSRV.TMP /QIBM/USERDATA/ICSS/CERT/SERVER/EXPSGN.TMP
	You may be missing a prerequisite licensed program (LPP) that DCM requires be installed. Check the list of “DCM set up requirements” on page 27 and ensure that all licensed programs are installed properly.

Problem	Possible Solution
The system does not accept a CA text file that was transferred in binary mode from another system. It does accept the file when it is transferred in American National Standard Code for Information Interchange (ASCII).	Key rings and key databases are binary and, therefore, different. You must use File Transfer Protocol (FTP) in ASCII mode for CA text files and FTP in binary mode for binary files, such as files with these extensions: `.kdb`, `.kyr`, `.sth`, `.rdb`, and so forth.
You cannot change the password of a key database. A certificate in the key database is no longer valid.	After verifying that an incorrect password is not the problem, find and delete the invalid certificate or certificates from the certificate store, and then try to change the password. If you have expired certificates in your certificate store, the expired certificates are no longer valid. Since the certificates are not valid, the password change function for the certificate store may not allow the password to be changed and the encryption process will not encrypt the private keys of the expired certificate. This keeps the password change from occurring, and the system may report that certificate store corruption is one of the reasons. You must remove the invalid (expired) certificates from the certificate store.
You need to use certificates for an Internet user and therefore need to use validation lists, but DCM does not provide functions for validation lists.	Business partners who are writing applications to use validation lists must write their code to associate the validation list with their application as expected. They must also write the code that determines when the Internet user's identity is appropriately validated so that the certificate can be added to the validation list. For more information review the i5/OS Information Center topic QsyAddVldlCertificate API. Consult the IBM HTTP Server for i5/OS documentation for help with configuring a secure HTTP server instance to use the validation list.

Troubleshooting browser problems

Review this information to help you troubleshoot some of the more common browser-related problems that you may encounter while working with Digital Certificate Manager (DCM).

Problem	Possible Solution
Microsoft® Internet Explorer does not let you select a different certificate until you start a new browser session.	Begin a new browser session for Internet Explorer.
Internet Explorer does not show all selectable client/user certificates in a browser's selection list. Internet Explorer only shows certificates, issued by the trusted CA, that you can use at the secure site.	A CA must be trusted in the key database as well as by the secure application. Be sure that you signed on to the PC for the Internet Explorer browser with the same user name as the one that put the user certificate in the browser. Get another user certificate from the system that you are accessing. The system administrator must be sure that the certificate store (key database) still trusts the CA that signed the user and system certificates.
Internet Explorer 5 receives the CA certificate, but cannot open the file or find the disk to which you saved the certificate.	This is a new browser feature for certificates that are not yet trusted by the Internet Explorer browser. You can choose the location on your PC.
You received a browser warning that the system name and system certificate do not match.	Some browsers do different things for uppercase and lowercase matching on system names. Type the URL with the same case as the system certificate shows. Or, create the system certificate with the case that matches what most users use. Unless you know what you are doing, it is best to leave the server name or system name as it was. You must also check that your domain name server is set up correctly.

Problem	Possible Solution
You started Internet Explorer with HTTPS instead of HTTP, and you received a warning of a secure and nonsecure mix of sessions.	Choose to accept and ignore the warning; a future release of Internet Explorer fixes this problem.
Netscape Communicator 4.04 for Windows converted hexadecimal values A1 and B1 to B2 and 9A in the Polish code page.	This is a browser bug that affects NLS. Use a different browser or even use the same version of this browser on a different platform, such as Netscape Communicator 4.04 for AIX®.
In a user profile, Netscape Communicator for 4.04 showed uppercase user certificate NLS characters correctly, but showed lowercase characters incorrectly.	Some national language characters that were entered correctly as one character but are not the same character when displayed later. For example, on the Windows version of Netscape Communicator 4.04, the hexadecimal values A1 and B1 were converted to B2 and 9A for the Polish code page, resulting in different NLS characters being displayed.
The browser continues to tell the user that the CA is not yet trusted.	Use DCM to set the **CA status** to **enabled** to mark the CA as trusted.
Internet Explorer requests reject the connection for HTTPS.	This is a problem with the browser function or its configuration. The browser chose not to connect to a site that is using a system certificate that might be self-signed or may not be valid for some other reason.
Netscape Communicator browser and server products employ root certificates from companies, including, but not limited to, VeriSign, as an enabling feature of SSL communications — specifically, authentication. All root certificates expire periodically. Some Netscape browser and server root certificates expired between December 25, 1999 and December 31, 1999. If you did not fix this problem on or before December 14, 1999, you will receive an error message.	Earlier versions of the browser (Netscape Communicator 4.05 or earlier) have certificates that expire. You need to upgrade the browser to the current Netscape Communicator version. Information on browser root certificates is available on many sites including `http://home.netscape.com/security/` and `http://www.verisign.com/server/cus/rootcert/webmaster.html`. Free browser downloads are available from `http://www.netcenter.com`.

Troubleshooting HTTP Server for System i problems

Review this information to help you troubleshoot HTTP Server problems you may encounter while working with Digital Certificate Manager (DCM).

Problem	Possible Solution
Hypertext Transfer Protocol Secure (HTTPS) does not work.	Be sure the HTTP Server is configured correctly for using SSL. In V5R1 or later versions the configuration file must have **SSLAppName** set by using the HTTP Server Administration interface. Also, the configuration must have a virtual host configured that uses the SSL port, with **SSL** set to **Enabled** for the virtual host. There must also be two **Listen** directives specifying two different ports, one for SSL and the other not for SSL. These are set on the **General Settings** page. Be sure the server instance is created and the server certificate is signed.

Problem	Possible Solution
The process for registering an HTTP Server instance as a secure application needs clarification.	On your system, go to the HTTP Server Administration interface to set the configuration for your HTTP Server. You first must define a virtual host to enable SSL. After you define a virtual host, you must specify that the virtual host use the SSL port defined previously on the **Listen** directive (on the **General Settings** page. Next, you must use the **SSL with Certificate Authentication** page under **Security** to enable SSL in the previously configured virtual host. All changes must be applied to the configuration file. Note that registering your instance does not automatically choose which certificates the instance will use. You must use DCM to assign a specific certificate to your application before you try to end and then restart your server instance.
You are having difficulty setting up the HTTP Server for validation lists and optional client authentication.	See the IBM HTTP Server for i5/OS documentation for options on setting up the instance.
Netscape Communicator waits for the configuration directive in the HTTP Server code to expire before allowing you to select a different certificate.	A large certificate value makes it hard to register a second certificate since the browser is still using the first one.
You are trying to get the browser to present the X.509 certificate to the HTTP Server so that you can use the certificate as input to the QsyAddVldlCertificate API.	You must use **SSLEnable** and **SSLClientAuth ON** in order to get the HTTP Server to load the HTTPS_CLIENT_CERTIFICATE environment variable. You can locate information about these APIs with the API finder topic in the i5/OS Information Center. You may also want to look at these validation list or certificate-related APIs: • QsyListVldlCertificates and QSYLSTVC • QsyRemoveVldlCertificate and QRMVVC • QsyCheckVldlCertificate and QSYCHKVC • QsyParseCertificate and QSYPARSC, and so on.
The HTTP Server takes too long to return, or times out if you request a list of the certificates in the validation list and there are more than 10,000 items.	Create a batch job that looks for and deletes certificates matching certain criteria, such as all those that have expired or are from a certain CA.
The HTTP Server will not start successfully with **SSL** set to **Enabled**, and error message `HTP8351` appears in the job log. The error log for the HTTP Server shows an error that SSL Initialization operation failed with a return code error of `107` when the HTTP Server fails.	Error `107` means the certificate has expired. Use DCM to assign a different certificate to the application; for example, QIBM_HTTP_SERVER_MY_SERVER. If the server instance that is failing to start is the *ADMIN server, then temporarily set **SSL** to Disabled so that you can use DCM on the *ADMIN server. Then use DCM to assign a different certificate to the QIBM_HTTP_SERVER_ADMIN application and try setting **SSL** to **Enable** again.

Troubleshooting assigning a user certificate

When you use the **Assign a user certificate** task, Digital Certificate Manager (DCM) displays certificate information for you to approve before registering the certificate.

If DCM is unable to display a certificate, the problem might be caused by one of these situations:

1. Your browser did not request that you select a certificate to present to the server. This may happen if the browser cached a previous certificate (from accessing a different server). Try clearing the browser's cache and try the task again. The browser will prompt you to select a certificate.
2. This may also happen if you configure your browser so that it does not display a selection list and the browser contains only one certificate from a Certificate Authority (CA) in the list of CAs that the server trusts. Check your browser configuration settings and change them, if necessary. Your browser will then prompt you to select a certificate. If you cannot present a certificate from a CA that the server is set to trust, you cannot assign a certificate. Contact your DCM administrator.

3. The certificate that you want to register is already registered with DCM.
4. The Certificate Authority that issued the certificate is not designated as trusted for the system or the application in question. Therefore, the certificate you are presenting is not valid. Contact your system administrator to determine if the CA that issued your certificate is correct. If the CA is correct, the system administrator may need to **Import** the CA certificate into the *SYSTEM certificate store. Or, the administrator may need to use the **Set CA status** task to enable the CA as trusted to correct the problem.
5. You do not have a certificate to register. You can check for user certificates in your browser to see if this is the problem.
6. The certificate that you are trying to register is expired or incomplete. You must either renew the certificate or contact the CA that issued it to resolve the problem.
7. The IBM HTTP Server for i5/OS is not correctly set up to do certificate registration using SSL and client authentication on the secure Administrative server instance. If none of the previous troubleshooting tips works, contact your system administrator to report the problem.

To **Assign a user certificate**, you must connect to Digital Certificate Manager (DCM) by using an SSL session. If you are not using SSL when you select the **Assign a user certificate** task, DCM displays a message that you must use SSL. The message contains a button so that you can connect to DCM by using SSL. If the message displays without the button, inform your system administrator of the problem. The Web server may need to be restarted to ensure that the configuration directives for using SSL are activated.

Related tasks

"Assigning a user certificate" on page 43
You can assign a user certificate that you own to your i5/OS user profile or other user identity. The certificate may be from a private local CA on another system or from a well-known Internet CA. Before you can assign a certificate to a user identity, the issuing CA must be trusted by the server, and the certificate must not already be associated with a user profile or other user identity on the system.

Related information for DCM

Review this page to find links to other resources for learning more about digital certificates, public key infrastructure, Digital Certificate Manager (DCM), and other related information.

As the use of digital certificates has become more prevalent, information resources have also become more available. Here is a small list of other resources that you can review to learn more about digital certificates and how you can use them to enhance your systems security policy:

- **VeriSign Help Desk Web site** The VeriSign Web site provides an extensive library on digital certificates topics, as well as a number of other Internet security subjects.
- **IBM eServer™ iSeries Wired Network Security: OS/400 V5R1 DCM and Cryptographic Enhancements SG24-6168** This IBM Redbook focuses on OS/400 V5R1 network security enhancements. The redbook covers many topics including how to use object signing capabilities, Digital Certificate Manager (DCM), the 4758 Cryptographic Coprocessor support for SSL, and so forth.
- **AS/400® Internet Security: Developing a Digital Certificate Infrastructure (SG24-5659)** This redbook describes what you can do with digital certificates on the OS/400 system. It explains how to set up the various servers and clients to use certificates. Further it provides information and sample code of how to use OS/400 APIs to manage and use digital certificates in user applications.
- **RFC Index Search** This Web site provides a searchable repository of Request for Comments (RFCs). RFCs describes the standards for Internet protocols, such as SSL, PKIX, and others that related to using digital certificates.

Appendix. Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106-0032, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation

Software Interoperability Coordinator, Department YBWA
3605 Highway 52 N
Rochester, MN 55901
U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this information and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement, IBM License Agreement for Machine Code, or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

All IBM prices shown are IBM's suggested retail prices, are current and are subject to change without notice. Dealer prices may vary.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

Each copy or any portion of these sample programs or any derivative work, must include a copyright notice as follows:

© (your company name) (year). Portions of this code are derived from IBM Corp. Sample Programs. © Copyright IBM Corp. _enter the year or years_. All rights reserved.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Trademarks

The following terms are trademarks of International Business Machines Corporation in the United States, other countries, or both:

- AIX
- AS/400
- Domino

| eServer
| i5/OS
| IBM
| iSeries
| Lotus
| Net.Data
| OS/400
| System i

Microsoft, Windows, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Other company, product, and service names may be trademarks or service marks of others.

Terms and conditions

Permissions for the use of these publications is granted subject to the following terms and conditions.

Personal Use: You may reproduce these publications for your personal, noncommercial use provided that all proprietary notices are preserved. You may not distribute, display or make derivative works of these publications, or any portion thereof, without the express consent of IBM.

Commercial Use: You may reproduce, distribute and display these publications solely within your enterprise provided that all proprietary notices are preserved. You may not make derivative works of these publications, or reproduce, distribute or display these publications or any portion thereof outside your enterprise, without the express consent of IBM.

Except as expressly granted in this permission, no other permissions, licenses or rights are granted, either express or implied, to the publications or any information, data, software or other intellectual property contained therein.

IBM reserves the right to withdraw the permissions granted herein whenever, in its discretion, the use of the publications is detrimental to its interest or, as determined by IBM, the above instructions are not being properly followed.

You may not download, export or re-export this information except in full compliance with all applicable laws and regulations, including all United States export laws and regulations.

IBM MAKES NO GUARANTEE ABOUT THE CONTENT OF THESE PUBLICATIONS. THE PUBLICATIONS ARE PROVIDED ″AS-IS″ AND WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, AND FITNESS FOR A PARTICULAR PURPOSE.

Printed in USA