



## Configuring PKI

---

This chapter describes the Public Key Infrastructure (PKI) support on the Cisco NX-OS device. PKI allows the device to obtain and use digital certificates for secure communication in the network and provides manageability and scalability for Secure Shell (SSH).

This chapter includes the following sections:

- [Information About PKI, on page 1](#)
- [Guidelines and Limitations for PKI, on page 7](#)
- [Default Settings for PKI, on page 8](#)
- [Configuring CAs and Digital Certificates, on page 8](#)
- [Verifying the PKI Configuration, on page 26](#)
- [Configuration Examples for PKI, on page 26](#)
- [Additional References for PKI, on page 47](#)
- [Resource Public Key Infrastructure \(RPKI\), on page 48](#)
- [RPKI Configuration, on page 48](#)
- [RPKI Show Commands, on page 50](#)
- [RPKI Clear Commands, on page 51](#)
- [RPKI Debug and Event History Commands, on page 51](#)

## Information About PKI

This section provides information about PKI.

### CAs and Digital Certificates

Certificate authorities (CAs) manage certificate requests and issue certificates to participating entities such as hosts, network devices, or users. The CAs provide centralized key management for the participating entities.

Digital signatures, based on public key cryptography, digitally authenticate devices and individual users. In public key cryptography, such as the RSA encryption system, each device or user has a key pair that contains both a private key and a public key. The private key is kept secret and is known only to the owning device or user only. However, the public key is known to everybody. Anything encrypted with one of the keys can be decrypted with the other. A signature is formed when data is encrypted with a sender's private key. The receiver verifies the signature by decrypting the message with the sender's public key. This process relies on the receiver having a copy of the sender's public key and knowing with a high degree of certainty that it really does belong to the sender and not to someone pretending to be the sender.

Digital certificates link the digital signature to the sender. A digital certificate contains information to identify a user or device, such as the name, serial number, company, department, or IP address. It also contains a copy of the entity's public key. The CA that signs the certificate is a third party that the receiver explicitly trusts to validate identities and to create digital certificates.

To validate the signature of the CA, the receiver must first know the CA's public key. Typically, this process is handled out of band or through an operation done at installation. For instance, most web browsers are configured with the public keys of several CAs by default.

## Trust Model, Trust Points, and Identity CAs

The PKI trust model is hierarchical with multiple configurable trusted CAs. You can configure each participating device with a list of trusted CAs so that a peer certificate obtained during the security protocol exchanges can be authenticated if it was issued by one of the locally trusted CAs. The Cisco NX-OS software locally stores the self-signed root certificate of the trusted CA (or certificate chain for a subordinate CA). The process of securely obtaining a trusted CA's root certificate (or the entire chain in the case of a subordinate CA) and storing it locally is called *CA authentication*.

The information about a trusted CA that you have configured is called the *trust point* and the CA itself is called a *trust point CA*. This information consists of a CA certificate (or certificate chain in case of a subordinate CA) and certificate revocation checking information.

The Cisco NX-OS device can also enroll with a trust point to obtain an identity certificate to associate with a key pair. This trust point is called an *identity CA*.

## CA Certificate Hierarchy

For secure services, you typically have multiple trusted CAs. The CAs are usually installed in all the hosts as a bundle. The NX-OS PKI infrastructure does support importing certificate chain. However, with the current CLIs, one chain at a time can be installed. This procedure can be cumbersome when there are several CA chains to be installed. This requires a facility to download CA bundles that could include several intermediate and root CAs.

## Importing CA Bundle

The **crypto CA trustpoint** command binds the CA certificates, CRLs, identity certificates and key pairs to a named label. All files corresponding to each of these entities are stored in the NX-OS certstore directory (`/isan/etc/certstore`) and tagged with the trustpoint label.

To access the CA certificates, an SSL app only needs to point to the standard NX-OS cert-store and specify that as the CA path during SSL initialization. It does not need to be aware of the trustpoint label under which CAs are installed.

If clients need to bind to an identity certificate, the trustpoint label needs to be used as the binding point.

The `import pkcs` command is enhanced to install the CA certificates under a trustpoint label. This can be further enhanced to install a CA bundle. The import command structure is modified to add `pkcs7` option which is used for providing CA bundle file in `pkcs7` format.

Beginning with Cisco NX-OS Release 10.1(1), the `pkcs7` file format is supported to unpack the CA bundle and install each CA chain under its own label. The labels are formed by appending an index to the main trustpoint label.

Once installed, there is no logical binding of all CA chains to a bundle.

## Import of the CA Certificate Bundle in PKCS7 Format

To support the import of the ca certificate bundle which consists of multiple independent certificate chains, the option of 'pkcs7' is introduced in the crypto import command.

### Procedure

	Command or Action	Purpose
<b>Step 1</b>	<b>configure terminal</b> <b>Example:</b> <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
<b>Step 2</b>	<b>crypto ca import &lt;baselabel&gt; pkcs7 &lt;uri0&gt;</b> <b>force</b>	<p>There are two input arguments in the command. The source file which is the ca bundle file is given in the &lt;uri0&gt;, the input file has to be in pkcs7 format indicating that it is a cabundle file.</p> <p>Multiple certificate chains will be extracted out of the cabundle. The command will generate multiple trustpoints with ca certificate chain attached to each one. Import command generates two configurations which are global CA bundle configuration and CA bundle sub-configuration with each trustpoint generated.</p> <p>The force option removes the CA bundle and related trustpoint configurations, imports a new CA bundle with the same bundle name, and generates fresh trustpoint configurations related to the cabundle.</p>
<b>Step 3</b>	<b>crypto ca cabundle &lt;bundle-name&gt;</b>	<p>The bundle-name is same as baselabel for import case. You can use the <b>no</b> form of this command to delete the, CA bundle, trustpoints, and related certificate chains.</p> <p>After importing CA bundle under a particular baselabel name and generating all the trustpoints, if a user try to execute the import command again under the same baselabel name, it will throw error saying CA bundle already exists. The user can use force option to modify the existing CA bundle.</p> <p>Maximum number of Cabundles supported is 20.</p>
<b>Step 4</b>	<b>exit</b> <b>Example:</b>	Exits configuration mode.

	Command or Action	Purpose
	<pre>switch(config)# exit switch#</pre>	
<b>Step 5</b>	(Optional) <b>show crypto ca certificates</b>  <b>Example:</b> <pre>switch# show crypto ca certificates</pre>	Displays the CA certificates.
<b>Step 6</b>	(Optional) <b>copy running-config startup-config</b>  <b>Example:</b> <pre>switch# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

## RSA Key Pairs and Identity Certificates

You can obtain an identity certificate by generating one or more RSA key pairs and associating each RSA key pair with a trust point CA where the Cisco NX-OS device intends to enroll. The Cisco NX-OS device needs only one identity per CA, which consists of one key pair and one identity certificate per CA.

The Cisco NX-OS software allows you to generate RSA key pairs with a configurable key size (or modulus). The default key size is 512. You can also configure an RSA key-pair label. The default key label is the device fully qualified domain name (FQDN).

The following list summarizes the relationship between trust points, RSA key pairs, and identity certificates:

- A trust point corresponds to a specific CA that the Cisco NX-OS device trusts for peer certificate verification for any application (such as SSH).
- A Cisco NX-OS device can have many trust points and all applications on the device can trust a peer certificate issued by any of the trust point CAs.
- A trust point is not restricted to a specific application.
- A Cisco NX-OS device enrolls with the CA that corresponds to the trust point to obtain an identity certificate. You can enroll your device with multiple trust points which means that you can obtain a separate identity certificate from each trust point. The identity certificates are used by applications depending upon the purposes specified in the certificate by the issuing CA. The purpose of a certificate is stored in the certificate as a certificate extension.
- When enrolling with a trust point, you must specify an RSA key pair to be certified. This key pair must be generated and associated to the trust point before generating the enrollment request. The association between the trust point, key pair, and identity certificate is valid until it is explicitly removed by deleting the certificate, key pair, or trust point.
- The subject name in the identity certificate is the fully qualified domain name for the Cisco NX-OS device.
- You can generate one or more RSA key pairs on a device and each can be associated to one or more trust points. But no more than one key pair can be associated to a trust point, which means only one identity certificate is allowed from a CA.
- If the Cisco NX-OS device obtains multiple identity certificates (each from a distinct CA), the certificate that an application selects to use in a security protocol exchange with a peer is application specific.

- You do not need to designate one or more trust points for an application. Any application can use any certificate issued by any trust point as long as the certificate purpose satisfies the application requirements.
- You do not need more than one identity certificate from a trust point or more than one key pair to be associated to a trust point. A CA certifies a given identity (or name) only once and does not issue multiple certificates with the same name. If you need more than one identity certificate for a CA and if the CA allows multiple certificates with the same names, you must define another trust point for the same CA, associate another key pair to it, and have it certified.

## Multiple Trusted CA Support

The Cisco NX-OS device can trust multiple CAs by configuring multiple trust points and associating each with a distinct CA. With multiple trusted CAs, you do not have to enroll a device with the specific CA that issued the certificate to a peer. Instead, you can configure the device with multiple trusted CAs that the peer trusts. The Cisco NX-OS device can then use a configured trusted CA to verify certificates received from a peer that were not issued by the same CA defined in the identity of the peer device.

## PKI Enrollment Support

Enrollment is the process of obtaining an identity certificate for the device that is used for applications like SSH. It occurs between the device that requests the certificate and the certificate authority.

The Cisco NX-OS device performs the following steps when performing the PKI enrollment process:

- Generates an RSA private and public key pair on the device.
- Generates a certificate request in standard format and forwards it to the CA.



---

**Note** The CA administrator may be required to manually approve the enrollment request at the CA server, when the request is received by the CA.

---

- Receives the issued certificate back from the CA, signed with the CA's private key.
- Writes the certificate into a nonvolatile storage area on the device (bootflash).

## Manual Enrollment Using Cut-and-Paste

The Cisco NX-OS software supports certificate retrieval and enrollment using manual cut-and-paste. Cut-and-paste enrollment means that you must cut and paste the certificate requests and resulting certificates between the device and the CA.

You must perform the following steps when using cut and paste in the manual enrollment process:

- Create an enrollment certificate request, which the Cisco NX-OS device displays in base64-encoded text form.
- Cut and paste the encoded certificate request text in an e-mail or in a web form and send it to the CA.
- Receive the issued certificate (in base64-encoded text form) from the CA in an e-mail or in a web browser download.

- Cut and paste the issued certificate to the device using the certificate import facility.

## Multiple RSA Key Pair and Identity CA Support

Multiple identity CAs enable the device to enroll with more than one trust point, which results in multiple identity certificates, each from a distinct CA. With this feature, the Cisco NX-OS device can participate in SSH and other applications with many peers using certificates issued by CAs that are acceptable to those peers.

The multiple RSA key-pair feature allows the device to maintain a distinct key pair for each CA with which it is enrolled. It can match policy requirements for each CA without conflicting with the requirements specified by the other CAs, such as the key length. The device can generate multiple RSA key pairs and associate each key pair with a distinct trust point. Thereafter, when enrolling with a trust point, the associated key pair is used to construct the certificate request.

## Peer Certificate Verification

The PKI support on a Cisco NX-OS device can verify peer certificates. The Cisco NX-OS software verifies certificates received from peers during security exchanges for applications, such as SSH. The applications verify the validity of the peer certificates. The Cisco NX-OS software performs the following steps when verifying peer certificates:

- Verifies that the peer certificate is issued by one of the locally trusted CAs.
- Verifies that the peer certificate is valid (not expired) with respect to current time.
- Verifies that the peer certificate is not yet revoked by the issuing CA.

For revocation checking, the Cisco NX-OS software supports the certificate revocation list (CRL). A trust point CA can use this method to verify that the peer certificate has not been revoked.

## Certificate Revocation Checking

The Cisco NX-OS software can check the revocation status of CA certificates. The applications can use the revocation checking mechanisms in the order that you specify. The choices are CRL, NDCPP: OCSP for Syslog, none, or a combination of these methods.

## CRL Support

The CAs maintain certificate revocation lists (CRLs) to provide information about certificates revoked prior to their expiration dates. The CAs publish the CRLs in a repository and provide the download public URL in all issued certificates. A client verifying a peer's certificate can obtain the latest CRL from the issuing CA and use it to determine if the certificate has been revoked. A client can cache the CRLs of some or all of its trusted CAs locally and use them later if necessary until the CRLs expire.

The Cisco NX-OS software allows the manual configuration of predownloaded CRLs for the trust points, and then caches them in the device bootflash (cert-store). During the verification of a peer certificate, the Cisco NX-OS software checks the CRL from the issuing CA only if the CRL has already been cached locally and the revocation checking is configured to use the CRL. Otherwise, the Cisco NX-OS software does not perform CRL checking and considers the certificate to be not revoked unless you have configured other revocation checking methods.

## NDcPP: OCSP for Syslog

Online Certificate Status Protocol (OCSP) is a method to check certificate revocation when a peer has to retrieve this revocation information and then validate it to check the certificate revocation status. In this method, the certification revocation status is limited by the peer's ability to reach an OCSP responder through the cloud or by the certificate sender's performance in retrieving the certificate revocation-information.

When the remote syslog server shares the certificate which has an OCSP responder URL, the client sends the server certificate to an external OCSP responder (CA) server. The CA server validates this certificate and confirms if it is a valid or a revoked certificate. In this case, the client does not have to maintain the revoked certificate list locally.

## Import and Export Support for Certificates and Associated Key Pairs

As part of the CA authentication and enrollment process, the subordinate CA certificate (or certificate chain) and identity certificates can be imported in standard PEM (base64) format.

The complete identity information in a trust point can be exported to a file in the password-protected PKCS#12 standard format. It can be later imported to the same device (for example, after a system crash) or to a replacement device. The information in a PKCS#12 file consists of the RSA key pair, the identity certificate, and the CA certificate (or chain).

## Guidelines and Limitations for PKI

PKI has the following configuration guidelines and limitations:

- The maximum number of key pairs you can configure on a Cisco NX-OS device is 16.
- The maximum number of trust points you can declare on a Cisco NX-OS device is 16.
- The maximum number of identity certificates that you can configure on a Cisco NX-OS device are 16.
- The maximum number of certificates in a CA certificate chain is 10.
- The maximum number of trust points you can authenticate to a specific CA is 10.
- Configuration rollbacks do not support the PKI configuration.
- Beginning with Cisco NX-OS Release 9.3(5), Cisco NX-OS software supports NDcPP: OCSP for Syslog.
- Beginning with Cisco NX-OS Release 10.3(3)F, Elliptic Curve Cryptography (ECC) key pair support is provided to generate and import the certificate on Cisco Nexus switches.



---

**Note** If you are familiar with the Cisco IOS CLI, be aware that the Cisco NX-OS commands for this feature might differ from the Cisco IOS commands that you would use.

---

## Default Settings for PKI

This table lists the default settings for PKI parameters.

**Table 1: Default PKI Parameters**

Parameters	Default
Trust point	None
RSA key pair	None
RSA key-pair label	Device FQDN
RSA key-pair modulus	512
RSA key-pair exportable	Enabled
Revocation check method	CRL

## Configuring CAs and Digital Certificates

This section describes the tasks that you must perform to allow CAs and digital certificates on your Cisco NX-OS device to interoperate.

### Configuring the Hostname and IP Domain Name

You must configure the hostname and IP domain name of the device if you have not yet configured them because the Cisco NX-OS software uses the fully qualified domain name (FQDN) of the device as the subject in the identity certificate. Also, the Cisco NX-OS software uses the device FQDN as a default key label when you do not specify a label during key-pair generation. For example, a certificate named DeviceA.example.com is based on a device hostname of DeviceA and a device IP domain name of example.com.




---

**Caution** Changing the hostname or IP domain name after generating the certificate can invalidate the certificate.

---

#### Procedure

	Command or Action	Purpose
<b>Step 1</b>	<b>configure terminal</b>  <b>Example:</b> <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.



	Command or Action	Purpose
<b>Step 2</b>	<b>hostname</b> <i>hostname</i>  <b>Example:</b> switch(config)# hostname DeviceA	Configures the hostname of the device.
<b>Step 3</b>	<b>ip domain-name</b> <i>name</i> [ <b>use-vrf</b> <i>vrf-name</i> ]  <b>Example:</b> DeviceA(config)# ip domain-name example.com	Configures the IP domain name of the device. If you do not specify a VRF name, the command uses the default VRF.
<b>Step 4</b>	<b>exit</b>  <b>Example:</b> switch(config)# exit switch#	Exits configuration mode.
<b>Step 5</b>	(Optional) <b>show hosts</b>  <b>Example:</b> switch# show hosts	Displays the IP domain name.
<b>Step 6</b>	(Optional) <b>copy running-config startup-config</b>  <b>Example:</b> switch# copy running-config startup-config	Copies the running configuration to the startup configuration.

## Generating an RSA Key Pair

You can generate an RSA key pairs to sign and/or encrypt and decrypt the security payload during security protocol exchanges for applications. You must generate the RSA key pair before you can obtain a certificate for your device.

Beginning Cisco NX-OS Release 9.3(3), you must explicitly generate RSA key pairs before you associate the Cisco NX-OS device with a trust point CA. Prior to Cisco NX-OS Releases 9.3(3), if unavailable, the RSA key pairs would be auto generated.

### Procedure

	Command or Action	Purpose
<b>Step 1</b>	<b>configure terminal</b>  <b>Example:</b> switch# configure terminal switch(config)#	Enters global configuration mode.
<b>Step 2</b>	<b>crypto key generate rsa</b> [ <b>label</b> <i>label-string</i> ] [ <b>exportable</b> ] [ <b>modulus</b> <i>size</i> ]  <b>Example:</b>	Generates an RSA key pair. The maximum number of key pairs on a device is 16.  The label string is alphanumeric, case sensitive, and has a maximum length of 64 characters.

	Command or Action	Purpose
	<pre>switch(config)# crypto key generate rsa exportable</pre>	<p>The default label string is the hostname and the FQDN separated by a period character (.).</p> <p>Valid modulus values are 512, 768, 1024, 1536, 2048, 3072 and 4096. The default modulus size is 512.</p> <p><b>Note</b> The security policy on the Cisco NX-OS device and on the CA (where enrollment is planned) should be considered when deciding the appropriate key modulus.</p> <p>By default, the key pair is not exportable. Only exportable key pairs can be exported in the PKCS#12 format.</p> <p><b>Caution</b> You cannot change the exportability of a key pair.</p>
<b>Step 3</b>	<p><b>exit</b></p> <p><b>Example:</b></p> <pre>switch(config)# exit switch#</pre>	Exits configuration mode.
<b>Step 4</b>	<p>(Optional) <b>show crypto key mypubkey rsa</b></p> <p><b>Example:</b></p> <pre>switch# show crypto key mypubkey rsa</pre>	Displays the generated key.
<b>Step 5</b>	<p>(Optional) <b>copy running-config startup-config</b></p> <p><b>Example:</b></p> <pre>switch# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

## Generating an ECC Key Pair

You can generate an ECC key pair to sign and/or encrypt and decrypt the security payload during security protocol exchanges for applications. You must generate the ECC key pair before you can obtain a certificate for your device. The ECC keys are stronger compared to RSA keys for a given length.

Beginning Cisco NX-OS Release 10.3(3)F, you can generate an ECC key pair to associate the Cisco NX-OS device with a trust point CA.

## Procedure

	Command or Action	Purpose
<b>Step 1</b>	<b>configure terminal</b> <b>Example:</b> <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
<b>Step 2</b>	<b>crypto key generate ecc [label <i>ecc-key-label</i>]  [exportable] [modulus <i>size</i>]</b> <b>Example:</b> <pre>switch(config)# crypto key generate ecc exportable modulus 224</pre>	<p>Generates an ECC key pair. The maximum number of key pairs on a device is 16.</p> <p>The label string is alphanumeric, case sensitive, and has maximum length of 64 characters. The default label string is the hostname and the FQDN separated by a period character (.).</p> <p>Valid modulus values are 224, 384, and 521. The default modulus size is 224.</p> <p><b>Note</b> The security policy on the Cisco NX-OS device and on the CA (where enrollment is planned) should be considered when deciding the appropriate key modulus.</p> <p>By default, the key pair is not exportable. Only exportable key pairs can be exported in the PKCS#12 format.</p> <p><b>Caution</b> You cannot change the exportability of a key pair.</p>
<b>Step 3</b>	<b>no crypto key generate ecc [label  <i>ecc-key-label</i>]</b> <b>Example:</b> <pre>switch(config)# no crypto key generate ecc label label-name</pre>	Deletes the ECC key.
<b>Step 4</b>	<b>exit</b> <b>Example:</b> <pre>switch(config)# exit switch#</pre>	Exits configuration mode.
<b>Step 5</b>	(Optional) <b>show crypto key mypubkey ecc</b> <b>Example:</b> <pre>switch# show crypto key mypubkey ecc</pre>	Displays the generated ECC key.

	Command or Action	Purpose
<b>Step 6</b>	(Optional) <b>copy running-config startup-config</b>  <b>Example:</b> switch# copy running-config startup-config	Copies the running configuration to the startup configuration.

## Creating a Trust Point CA Association

You must associate the Cisco NX-OS device with a trust point CA.

### Before you begin

Generate the RSA key pair.

### Procedure

	Command or Action	Purpose
<b>Step 1</b>	<b>configure terminal</b>  <b>Example:</b> switch# configure terminal switch(config)#	Enters global configuration mode.
<b>Step 2</b>	<b>crypto ca trustpoint <i>name</i></b>  <b>Example:</b> switch(config)# crypto ca trustpoint admin-ca switch(config-trustpoint)#	Declares a trust point CA that the device should trust and enters trust point configuration mode.  <b>Note</b> The maximum number of trustpoints that can be configured is 50.
<b>Step 3</b>	<b>cabundle <i>baselabel</i></b>  <b>Example:</b> switch(config-trustpoint)# cabundle test	Groups the trustpoints under a specific CA bundle. The <b>No</b> form of this command detaches the trustpoints from the CA bundle. This command associates the trustpoints to an existing CA bundle and it does not configure any new CA bundle.
<b>Step 4</b>	<b>enrollment terminal</b>  <b>Example:</b> switch(config-trustpoint)# enrollment terminal	Enables manual cut-and-paste certificate enrollment. The default is enabled.  <b>Note</b> The Cisco NX-OS software supports only the manual cut-and-paste method for certificate enrollment.
<b>Step 5</b>	<b>rsakeypair <i>label</i></b>  <b>Example:</b>	Specifies the label of the RSA key pair to associate to this trust point for enrollment.  <b>Note</b>

	Command or Action	Purpose
	<code>switch(config-trustpoint)# rsakeypair SwitchA</code>	You can specify only one RSA key pair per CA.
<b>Step 6</b>	<b>exit</b> <b>Example:</b> <code>switch(config-trustpoint)# exit</code> <code>switch(config)#</code>	Exits trust point configuration mode.
<b>Step 7</b>	(Optional) <b>show crypto ca trustpoints</b> <b>Example:</b> <code>switch(config)# show crypto ca trustpoints</code>	Displays trust point information.
<b>Step 8</b>	(Optional) <b>copy running-config startup-config</b> <b>Example:</b> <code>switch(config)# copy running-config startup-config</code>	Copies the running configuration to the startup configuration.

**Related Topics**

[Generating an RSA Key Pair](#), on page 9

## Configuring Certificate Mapping Filters

You can configure mapping filters to validate the CA certificates that are used for authentication. The mapping filters are used to match the CA certificate against a username.

Cisco NX-OS supports the following certificate mapping filters:

- `%username%`—Substitutes the user's login name.
- `%hostname%`—Substitutes the peer hostname.

**Before you begin**

Configure a cert-store for certificate authentication.

**Procedure**

	Command or Action	Purpose
<b>Step 1</b>	<b>configure terminal</b> <b>Example:</b> <code>switch# configure terminal</code> <code>switch(config)#</code>	Enters global configuration mode.
<b>Step 2</b>	<b>crypto certificatemap mapname map-name</b> <b>Example:</b>	Creates a new filter map.

	Command or Action	Purpose
	<pre>switch(config)# crypto certificatemap mapname filtermap1</pre>	
<b>Step 3</b>	<p><b>filter</b> [<b>subject-name</b> <i>subject-name</i>   <b>altname-email</b> <i>e-mail-ID</i>   <b>altname-upn</b> <i>user-principal-name</i>]</p> <p><b>Example:</b></p> <pre>switch(config-certmap-filter)# filter altname-upn %username%@cisco.com</pre>	<p>Configures one or more certificate mapping filters within the filter map. These certificate field attributes are supported in the filters: The validation passes if the certificate passes all of the filters configured in the map.</p> <ul style="list-style-type: none"> <li>• <b>subject-name</b>—The required subject name in the LDAP distinguished name (DN) string format. For example: <pre>filter subject-name CN=%username%</pre> <p>or</p> <pre>filter subject-name /C=IN/ST=KA/L=BLR/O=CISCO/OU=ABC/CN=%username%</pre> </li> <li>• <b>altname-email</b>—The e-mail address that must be present in the certificate as a subject alternative name. For example: <pre>filter altname-email %username%@cisco.com</pre> </li> <li>• <b>altname-upn</b>—The principal name that must be present in the certificate as a subject alternative name. For example: <pre>filter altname-upn %username%@%hostname%</pre> </li> </ul> <p>The validation passes if the certificate passes all of the filters configured in the map.</p>
<b>Step 4</b>	<p><b>exit</b></p> <p><b>Example:</b></p> <pre>switch(config-certmap-filter)# exit switch(config)#</pre>	Exits certificate mapping filter configuration mode.
<b>Step 5</b>	<p>(Optional) <b>crypto cert ssh-authorize</b> [<b>default</b>   <i>issuer-CAname</i>] [<b>map</b> <i>map-name1</i> [<i>map-name2</i>]]</p> <p><b>Example:</b></p> <pre>switch(config)# crypto cert ssh-authorize default map filtermap1</pre>	<p>Configures a certificate mapping filter for the Secure Shell (SSH) protocol. You can use the default filter map for SSH authorization or specify the issuer of the CA certificate. If you do not use the default map, you can specify one or two filter maps for authorization.</p> <p>If you specify the issuer of the CA certificate, the certificate bound to the user account is validated as successful if it passes one of the configured maps.</p>
<b>Step 6</b>	<p>(Optional) <b>show crypto certificatemap</b></p> <p><b>Example:</b></p>	Displays the certificate mapping filters.

	Command or Action	Purpose
	<code>switch(config)# show crypto certificatemap</code>	
<b>Step 7</b>	(Optional) <b>show crypto ssh-auth-map</b>  <b>Example:</b> <code>switch(config)# show crypto ssh-auth-map</code>	Displays the mapping filters configured for SSH authentication.
<b>Step 8</b>	(Optional) <b>copy running-config startup-config</b>  <b>Example:</b> <code>switch(config)# copy running-config startup-config</code>	Copies the running configuration to the startup configuration.

## Authenticating the CA

The configuration process of trusting a CA is complete only when the CA is authenticated to the Cisco NX-OS device. You must authenticate your Cisco NX-OS device to the CA by obtaining the self-signed certificate of the CA in PEM format, which contains the public key of the CA. Because the certificate of the CA is self-signed (the CA signs its own certificate) the public key of the CA should be manually authenticated by contacting the CA administrator to compare the fingerprint of the CA certificate.



**Note** The CA that you are authenticating is not a self-signed CA when it is a subordinate CA to another CA, which itself may be a subordinate to yet another CA, and so on, finally ending in a self-signed CA. This type of CA certificate is called the *CA certificate chain* of the CA being authenticated. In this case, you must input the full list of the CA certificates of all the CAs in the certification chain during the CA authentication. The maximum number of certificates in a CA certificate chain is 10.

### Before you begin

Create an association with the CA.

Obtain the CA certificate or CA certificate chain.

### Procedure

	Command or Action	Purpose
<b>Step 1</b>	<b>configure terminal</b>  <b>Example:</b> <code>switch# configure terminal switch(config)#</code>	Enters global configuration mode.
<b>Step 2</b>	<b>crypto ca authenticate name pemfile uri0</b>  <b>Example:</b> <code>switch(config)# crypto ca authenticate admin-ca input (cut &amp; paste) CA certificate</code>	Prompts you to cut and paste the certificate of the CA. Use the same name that you used when declaring the CA.  Also validates and attaches the CA chain directly to the specified trust point.

	Command or Action	Purpose
	<pre>(chain) in PEM format; end the input with a line containing only END OF INPUT : -----BEGIN CERTIFICATE----- MIIChCCoYqAwIBAgIQWDSiayOZPESR1LjK0ZejANBgkqhkiG9w0BAQIEADBB kDEMB4CSqSIB3QIEAFRFRWlHmRzUBjaXNjby5jb20uY2AueGMEAVTAk0 MRlWFAVDQIEWILYXUyRha2EjejaQBgNBAcICUhhbichbG9yZTEOMAwGAUE CMFQ2IzY28uZAFBgMEAsTgr6lcHNb3JhZ2UjejaQBgNBAcICUFWXUjUSF QIEFw0NlAIMDMjQvadaFw0NzAIMDMjUIMichMIGMSwHjKkZihvnd AQIEFHhWfuzG-IQQpc2NlmlNblEDVAKAUEBhMCSUkEjejaQBgNBAcICU crrhdGFYIESMFAAUEBwMQInFuzZFs3JIMQ4DAMDQKwDwAnjzeZIMEB AUECM4m0c3RvcrifZIESMFAAUEBwMQKcharfhtENBFWDDkKcZihvnd AQIEBQADS4wSAJFAW/7o3HXUEANBstHHzhNcdM87ypzwocSNZXMpeFXX CzyEgixLZASFLUOQIhIM8rO/4ljEBRwMkysCwEFAaOBzCB/DALBjWkQE EPMCAwMDwMDwF0TAQH/EPUwEe/zacBjNMQIEFQUjyRdMbzQMRUOQRQ GysWdEwawMDwF0BQWjAucCygk0caHFDovL3vzSO0C9DXURW5y2s IOWXUjUSJMNEMjYDawO6gILYzmlsZibzLkcc3NLIIP4ENlcrF8bnJv bGrcQcharfhtJlWQElit3UEMAGCSsCAQIEBjC/AQIDAgFAAOCSSqSIB3QIE BQIAOEFAH6Qh8E399Iw#KqG0gNlEqNthQARtOeEjyut/MGPKzsf9Ea NBG7E0on66zexOEOfG1Vs6mXp1//w== -----END CERTIFICATE----- END OF INPUT Fingerprint(s) : MD5 Fingerprint=65:84:9A:27:D5:71:03:33:9C:12:23:92:38:6F:78:12 Do you accept this certificate? [yes/no]: yes</pre>	<p>The maximum number of trust points that you can authenticate to a specific CA is 10.</p> <p><b>Note</b> For subordinate CA authentication, the Cisco NX-OS software requires the full chain of CA certificates ending in a self-signed CA because the CA chain is needed for certificate verification as well as for PKCS#12 format export.</p>
<b>Step 3</b>	<p><b>exit</b></p> <p><b>Example:</b></p> <pre>switch(config)# exit switch#</pre>	Exits configuration mode.
<b>Step 4</b>	<p>(Optional) <b>show crypto ca trustpoints</b></p> <p><b>Example:</b></p> <pre>switch# show crypto ca trustpoints</pre>	Displays the trust point CA information.
<b>Step 5</b>	<p>(Optional) <b>copy running-config startup-config</b></p> <p><b>Example:</b></p> <pre>switch# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

**Related Topics**

[Creating a Trust Point CA Association](#), on page 12

## Configuring Certificate Revocation Checking Methods

During security exchanges with a client (for example, an SSH user), the Cisco NX-OS device performs the certificate verification of the peer certificate sent by the client. The verification process may involve certificate revocation status checking.

You can configure the device to check the CRL downloaded from the CA. Downloading the CRL and checking locally does not generate traffic in your network. However, certificates can be revoked between downloads and your device would not be aware of the revocation.



**Before you begin**

Authenticate the CA.

Ensure that you have configured the CRL if you want to use CRL checking.

**Procedure**

	<b>Command or Action</b>	<b>Purpose</b>
<b>Step 1</b>	<b>configure terminal</b>  <b>Example:</b> switch# configure terminal switch(config)#	Enters global configuration mode.
<b>Step 2</b>	<b>crypto ca trustpoint <i>name</i></b>  <b>Example:</b> switch(config)# crypto ca trustpoint admin-ca switch(config-trustpoint)#	Specifies a trust point CA and enters trust point configuration mode.
<b>Step 3</b>	<b>revocation-check {crl [none]   none}</b>  <b>Example:</b> switch(config-trustpoint)# revocation-check none	Configures the certificate revocation checking methods. The default method is <b>crl</b> .  The Cisco NX-OS software uses the certificate revocation methods in the order that you specify.
<b>Step 4</b>	<b>exit</b>  <b>Example:</b> switch(config-trustpoint)# exit switch(config)#	Exits trust point configuration mode.
<b>Step 5</b>	(Optional) <b>show crypto ca trustpoints</b>  <b>Example:</b> switch(config)# show crypto ca trustpoints	Displays the trust point CA information.
<b>Step 6</b>	(Optional) <b>copy running-config startup-config</b>  <b>Example:</b> switch(config)# copy running-config startup-config	Copies the running configuration to the startup configuration.

**Related Topics**

[Authenticating the CA](#), on page 15

[Configuring a CRL](#), on page 23

## Generating Certificate Requests

You must generate a request to obtain identity certificates from the associated trust point CA for each of your device's RSA key pairs. You must then cut and paste the displayed request into an e-mail or in a website form for the CA.

### Before you begin

Create an association with the CA.

Obtain the CA certificate or CA certificate chain.

### Procedure

	Command or Action	Purpose
<b>Step 1</b>	<b>configure terminal</b> <b>Example:</b> <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
<b>Step 2</b>	<b>crypto ca enroll name</b> <b>Example:</b> <pre>switch(config)# crypto ca enroll admin-ca Create the certificate request .. Create a challenge password. You will need to verbally provide this password to the CA Administrator in order to revoke your certificate. For security reasons your password will not be saved in the configuration. Please make a note of it. Password:nbv123 The subject name in the certificate will be: DeviceA.cisco.com Include the switch serial number in the subject name? [yes/no]: no Include an IP address in the subject name [yes/no]: yes ip address:172.22.31.162 The certificate request will be displayed... -----BEGIN CERTIFICATE REQUEST----- MIIBgQCARQAWhDEAMBGAUUEAARMIhYXMTMS5jaXNjby5jb20wczBwDQYK KzZlbnVpQEBBAGj0AMIGAgEBLBy1UAJZNC7jUJDAsMNIgZ2t8r14IKY UUGAnY4q38vEMZSILU4JfZwbbkDKTysnjuCGjbtwj0Ehw/55IT9y P2NU8omqShrVZgC7ysN/PyMfKgzhbVpj+hzarGzHtG99XITq4WbMS0zXv8S VzH0VBAgMBAQglZABjki9wBCCxCMGDMJMLZMDGCSgSIB3DQET DjgMCwQIDRFAH/BswGIRMIhYXMTMS5jaXNjby5jb22HKwH6LwDQYK KzZlbnVpQEBBAGjEAKIGKERQ8rj0sKZM56fZhg6Jtdz3Gc89GlfWgt PftuNwLE/pw8HayfQlZT3ecgNw12bl5L33MF2bktExiTGU188rIOjgIXmjjas 8e23hDn5VBrklwAgWkMLANUZFUqjpfingENIZacJUS8ZqfUvtdkYtk0- -----END CERTIFICATE REQUEST-----</pre>	Generates a certificate request for an authenticated CA. <b>Note</b> You must remember the challenge password. It is not saved with the configuration. You must enter this password if your certificate needs to be revoked.
<b>Step 3</b>	<b>exit</b> <b>Example:</b>	Exits trust point configuration mode.

	Command or Action	Purpose
	<code>switch(config-trustpoint)# exit</code> <code>switch(config)#</code>	
<b>Step 4</b>	(Optional) <b>show crypto ca certificates</b>  <b>Example:</b> <code>switch(config)# show crypto ca certificates</code>	Displays the CA certificates.
<b>Step 5</b>	(Optional) <b>copy running-config startup-config</b>  <b>Example:</b> <code>switch(config)# copy running-config startup-config</code>	Copies the running configuration to the startup configuration.

**Related Topics**

[Creating a Trust Point CA Association](#), on page 12

## Installing Identity Certificates

You can receive the identity certificate from the CA by e-mail or through a web browser in base64 encoded text form. You must install the identity certificate from the CA by cutting and pasting the encoded text.

**Before you begin**

Create an association with the CA.

Obtain the CA certificate or CA certificate chain.

**Procedure**

	Command or Action	Purpose
<b>Step 1</b>	<b>configure terminal</b>  <b>Example:</b> <code>switch# configure terminal</code> <code>switch(config)#</code>	Enters global configuration mode.
<b>Step 2</b>	<b>crypto ca import name certificate</b>  <b>Example:</b> <code>switch(config)# crypto ca import admin-ca certificate</code> input (cut & paste) certificate in PEM format: -----BEGIN CERTIFICATE----- MIEFADCA6GgWIEAgIPCj0oQAAAAcDANBjckidGwOEAQLEADCEKDEgM4G CSqSStb3DQETARFRWlhmRzUBjaNjyo5jbc0xCAIBjMERYTAKlOMRIWEAYD VQJIBwILXUjXPa2EeEjAQBjMBAICUUhndhG9yZIEOMwGALUECHMQ2Lz Y2&Ez4RBJMEAsIGr6ldhN03JfZZUeEjAQBjMBAITURWYXUySEIQIAeFw0 NIEMLTWzAANDaRw0NjBMTWmE5jANDBaBwGjAYBjMERMIEVZlZ2FzLIEH Y2lZ2BjY29MIGFAOCSSqSStb3DQEBAPAAAGNADCEIKBgQC/GNACtjQ.41C dQlWgkjSICqLrK5aStNQ.rjQpzoKsZPEXjFZbiyeCIE8ylndWwSE08rJ47 glx42/sI9IRtb/8uLU/cj9jSSFK56ka7wWA8rDz8jMChIM4WlaY/cp2433	Prompts you to cut and paste the identity certificate for the CA named admin-ca.  The maximum number of identify certificates that you can configure on a device is 16.

	Command or Action	Purpose
	<pre>x7Rif6V0GurFzEgsl7/EIash9LxLwITDQPB04ICEzCCAg9wQDVFORQH/EEBw GIRfMhYMMMS5jaXNjby5jb2ZHEkWHGiwHQDMDFOBBMEKCIi+2ssqWEfigpF kHvnlVyc9jngMIHMBgNMSMEgQwgcGAFcc08aD06wjTEANjskUBoLEfwo0rGV pIGIMICQMSwHgjKcZllwcvPQkEhPhWfRZGLQGnc2WlnWbIEIMAKGALUE BfMCSY4EjAQbNfEgIUtharfnrGPhYTESMEGALUEBxMQrfLzZFS03JIMQ4w DkYDQQkEwDaxNjocZIMEGALUEBxMKnI0c3RvcrfZTESMEGALUEBxMQkEh crfHfENBjAFYnkHdQZIE9UEIwMERlAGsCALLUHRMGLwLcPsoCgGfGh0GfP6 Ly9zc2UtMDgVQ2VyeEUAar8sbC9EaGfYonfElmjEDQ55jamwMKAuoCyGfrZpbG06 Ly9z4HnzS0wCEMDXUJrV5tb2s4ERWxUuSUjMNBInjkdOBigYfWwBEGUf AQEFfjBMDsGCCsGAUEBzCh9ocHRwOi8wczNlLlIAPALNlcrfEbnJkbGwc3Nl LlIAP40fWxUuSUjMNBInjkdA9BggBgfFBQwA0ixZmlsZT0vL1kcc3NlLlIAP4 XENlcrfEbnJkbGwc3NlLlIAP40fWxUuSUjMNBInjkdANBjgphkdG9wQEAUF AANEADbcEGSe7GNLh8a0IwBNm24U69ZSJDrcOdZUUIgqrIdjpcPyejtsyflv E36cIZu4WSExREqxbTk8ycx7V5o= -----END CERTIFICATE-----</pre>	
<b>Step 3</b>	<pre>exit Example: switch(config)# exit switch#</pre>	Exits configuration mode.
<b>Step 4</b>	<pre>(Optional) show crypto ca certificates Example: switch# show crypto ca certificates</pre>	Displays the CA certificates.
<b>Step 5</b>	<pre>(Optional) copy running-config startup-config Example: switch# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

**Related Topics**

[Creating a Trust Point CA Association](#), on page 12

## Ensuring Trust Point Configurations Persist Across Reboots

You can ensure that the trustpoint configuration persists across Cisco NX-OS device reboots.

The trust point configuration is a normal Cisco NX-OS device configuration that persists across system reboots only if you copy it explicitly to the startup configuration. The certificates, key pairs, and CRL associated with a trust point are automatically persistent if you have already copied the trust point configuration in the startup configuration. Conversely, if the trust point configuration is not copied to the startup configuration, the certificates, key pairs, and CRL associated with it are not persistent since they require the corresponding trust point configuration after a reboot. Always copy the running configuration to the startup configuration to ensure that the configured certificates, key pairs, and CRLs are persistent. Also, save the running configuration after deleting a certificate or key pair to ensure that the deletions permanent.

The certificates and CRL associated with a trust point automatically become persistent when imported (that is, without explicitly copying to the startup configuration) if the specific trust point is already saved in startup configuration.

We recommend that you create a password-protected backup of the identity certificates and save it to an external server.



**Note** Copying the configuration to an external server does include the certificates and key pairs.

### Related Topics

[Exporting Identity Information in PKCS 12 Format](#), on page 21

## Exporting Identity Information in PKCS 12 Format

You can export the identity certificate along with the RSA key pair and CA certificate (or the entire chain in the case of a subordinate CA) of a trust point to a PKCS#12 file for backup purposes. You can import the certificate and RSA key pair to recover from a system crash on your device or when you replace the supervisor modules.



**Note** You can use only the `bootflash:filename` format when specifying the export URL.

### Before you begin

Authenticate the CA.

Install an identity certificate.

### Procedure

	Command or Action	Purpose
<b>Step 1</b>	<b>configure terminal</b> <b>Example:</b> <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
<b>Step 2</b>	<b>crypto ca export name pkcs12 bootflash:filename password</b> <b>Example:</b> <pre>switch(config)# crypto ca export admin-ca pkcs12 bootflash:adminid.p12 nbv123</pre>	Exports the identity certificate and associated key pair and CA certificates for a trust point CA. The password is alphanumeric, case sensitive, and has a maximum length of 128 characters.
<b>Step 3</b>	<b>exit</b> <b>Example:</b> <pre>switch(config)# exit switch#</pre>	Exits configuration mode.
<b>Step 4</b>	<b>copy bootflash:filename scheme://server/ [url /]filename</b> <b>Example:</b> <pre>switch# copy bootflash:adminid.p12 tftp:adminid.p12</pre>	Copies the PKCS#12 format file to a remote server.  For the <i>scheme</i> argument, you can enter <b>tftp:</b> , <b>ftp:</b> , <b>scp:</b> , or <b>sftp:</b> . The <i>server</i> argument is the address or name of the remote server, and the

	Command or Action	Purpose
		<p><i>url</i> argument is the path to the source file on the remote server.</p> <p>The <i>server</i>, <i>url</i>, and <i>filename</i> arguments are case sensitive.</p>

### Related Topics

[Generating an RSA Key Pair](#), on page 9

[Authenticating the CA](#), on page 15

[Installing Identity Certificates](#), on page 19

## Importing Identity Information in PKCS 12 or PKCS 7 Format

You can import the certificate and RSA key pair to recover from a system crash on your device or when you replace the supervisor modules.



**Note** You can use only the `bootflash:filename` format when specifying the import URL.

### Before you begin

Ensure that the trust point is empty by checking that no RSA key pair is associated with it and no CA is associated with the trust point using CA authentication.

### Procedure

	Command or Action	Purpose
<b>Step 1</b>	<p><code>copy scheme:// server/[url /]filename</code>  <code>bootflash:filename</code></p> <p><b>Example:</b></p> <pre>switch# copy tftp:adminid.p12 bootflash:adminid.p12</pre>	<p>Copies the PKCS#12 format file from the remote server.</p> <p>For the <i>scheme</i> argument, you can enter <b>tftp:</b>, <b>ftp:</b>, <b>scp:</b>, or <b>sftp:</b>. The <i>server</i> argument is the address or name of the remote server, and the <i>url</i> argument is the path to the source file on the remote server.</p> <p>The <i>server</i>, <i>url</i>, and <i>filename</i> arguments are case sensitive.</p>
<b>Step 2</b>	<p><code>configure terminal</code></p> <p><b>Example:</b></p> <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
<b>Step 3</b>	<p><code>crypto ca import name [pkcs12   pkcs7]</code>  <code>bootflash:filename</code></p> <p><b>Example:</b></p>	Imports the identity certificate and associated key pair and CA certificates for trust point CA.

	Command or Action	Purpose
	<code>switch(config)# crypto ca import admin-ca pkcs12 bootflash:adminid.p12 nbv123</code>	
<b>Step 4</b>	<b>exit</b> <b>Example:</b> <code>switch(config)# exit</code> <code>switch#</code>	Exits configuration mode.
<b>Step 5</b>	(Optional) <b>show crypto ca certificates</b> <b>Example:</b> <code>switch# show crypto ca certificates</code>	Displays the CA certificates.
<b>Step 6</b>	(Optional) <b>copy running-config startup-config</b> <b>Example:</b> <code>switch# copy running-config startup-config</code>	Copies the running configuration to the startup configuration.

## Configuring a CRL

You can manually configure CRLs that you have downloaded from the trust points. The Cisco NX-OS software caches the CRLs in the device bootflash (cert-store). During the verification of a peer certificate, the Cisco NX-OS software checks the CRL from the issuing CA only if you have downloaded the CRL to the device and you have configured certificate revocation checking to use the CRL.

### Before you begin

Ensure that you have enabled certificate revocation checking.

### Procedure

	Command or Action	Purpose
<b>Step 1</b>	<b>copy <i>scheme</i>:[//<i>server</i>/[<i>url</i> /]]<i>filename</i></b> <b>bootflash:<i>filename</i></b> <b>Example:</b> <code>switch# copy tftp:adminca.crl</code> <code>bootflash:adminca.crl</code>	Downloads the CRL from a remote server.  For the <i>scheme</i> argument, you can enter <b>tftp:</b> , <b>ftp:</b> , <b>scp:</b> , or <b>sftp:</b> . The <i>server</i> argument is the address or name of the remote server, and the <i>url</i> argument is the path to the source file on the remote server.  The <i>server</i> , <i>url</i> , and <i>filename</i> arguments are case sensitive.
<b>Step 2</b>	<b>configure terminal</b> <b>Example:</b> <code>switch# configure terminal</code> <code>switch(config)#</code>	Enters global configuration mode.

	Command or Action	Purpose
<b>Step 3</b>	<b>crypto ca crl request <i>name</i> bootflash:<i>filename</i></b> <b>Example:</b> switch(config)# crypto ca crl request admin-ca bootflash:adminca.crl	Configures or replaces the current CRL with the one specified in the file.
<b>Step 4</b>	<b>exit</b> <b>Example:</b> switch(config)# exit switch#	Exits configuration mode.
<b>Step 5</b>	(Optional) <b>show crypto ca crl <i>name</i></b> <b>Example:</b> switch# show crypto ca crl admin-ca	Displays the CA CRL information.
<b>Step 6</b>	(Optional) <b>copy running-config startup-config</b> <b>Example:</b> switch# copy running-config startup-config	Copies the running configuration to the startup configuration.

## Deleting Certificates from the CA Configuration

You can delete the identity certificates and CA certificates that are configured in a trust point. You must first delete the identity certificate, followed by the CA certificates. After deleting the identity certificate, you can disassociate the RSA key pair from a trust point. You must delete certificates to remove expired or revoked certificates, certificates that have compromised (or suspected to be compromised) key pairs, or CAs that are no longer trusted.

### Procedure

	Command or Action	Purpose
<b>Step 1</b>	<b>configure terminal</b> <b>Example:</b> switch# configure terminal switch(config)#	Enters global configuration mode.
<b>Step 2</b>	<b>crypto ca trustpoint <i>name</i></b> <b>Example:</b> switch(config)# crypto ca trustpoint admin-ca switch(config-trustpoint)#	Specifies a trust point CA and enters trust point configuration mode.
<b>Step 3</b>	<b>delete ca-certificate</b> <b>Example:</b> switch(config-trustpoint)# delete ca-certificate	Deletes the CA certificate or certificate chain.



	Command or Action	Purpose
<b>Step 4</b>	<b>delete certificate [force]</b>  <b>Example:</b> <pre>switch(config-trustpoint)# delete certificate</pre>	Deletes the identity certificate.  You must use the <b>force</b> option if the identity certificate you want to delete is the last certificate in a certificate chain or only identity certificate in the device. This requirement ensures that you do not mistakenly delete the last certificate in a certificate chain or only the identity certificate and leave the applications (such as SSH) without a certificate to use.
<b>Step 5</b>	<b>exit</b>  <b>Example:</b> <pre>switch(config-trustpoint)# exit switch(config)#</pre>	Exits trust point configuration mode.
<b>Step 6</b>	(Optional) <b>show crypto ca certificates [name]</b>  <b>Example:</b> <pre>switch(config)# show crypto ca certificates admin-ca</pre>	Displays the CA certificate information.
<b>Step 7</b>	(Optional) <b>copy running-config startup-config</b>  <b>Example:</b> <pre>switch(config)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

## Deleting RSA Key Pairs from a Cisco NX-OS Device

You can delete the RSA key pairs from a Cisco NX-OS device if you believe the RSA key pairs were compromised in some way and should no longer be used.



**Note** After you delete RSA key pairs from a device, ask the CA administrator to revoke your device's certificates at the CA. You must supply the challenge password that you created when you originally requested the certificates.

### Procedure

	Command or Action	Purpose
<b>Step 1</b>	<b>configure terminal</b>  <b>Example:</b> <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.

	Command or Action	Purpose
<b>Step 2</b>	<b>crypto key zeroize rsa label</b>  <b>Example:</b> switch(config)# crypto key zeroize rsa MyKey	Deletes the RSA key pair.
<b>Step 3</b>	<b>exit</b>  <b>Example:</b> switch(config)# exit switch#	Exits configuration mode.
<b>Step 4</b>	(Optional) <b>show crypto key mypubkey rsa</b>  <b>Example:</b> switch# show crypto key mypubkey rsa	Displays the RSA key pair configuration.
<b>Step 5</b>	(Optional) <b>copy running-config startup-config</b>  <b>Example:</b> switch# copy running-config startup-config	Copies the running configuration to the startup configuration.

**Related Topics**

[Generating Certificate Requests](#), on page 18

## Verifying the PKI Configuration

To display PKI configuration information, perform one of the following tasks:

Command	Purpose
<b>show crypto key mypubkey rsa</b>	Displays information about the RSA public keys generated on the Cisco NX-OS device.
<b>show crypto ca certificates</b>	Displays information about CA and identity certificates.
<b>show crypto ca crl</b>	Displays information about CA CRLs.
<b>show crypto ca trustpoints</b>	Displays information about CA trust points.

## Configuration Examples for PKI

This section shows examples of the tasks that you can use to configure certificates and CRLs on Cisco NX-OS devices using a Microsoft Windows Certificate server.



**Note** You can use any type of certificate server to generate digital certificates. You are not limited to using the Microsoft Windows Certificate server.

## Configuring Certificates on a Cisco NX-OS Device

To configure certificates on a Cisco NX-OS device, follow these steps:

### Procedure

**Step 1** Configure the device FQDN.

```
switch# configure terminal  
Enter configuration commands, one per line. End with CNTL/Z.  
switch(config)# hostname Device-1  
Device-1(config)#
```

**Step 2** Configure the DNS domain name for the device.

```
Device-1(config)# ip domain-name cisco.com
```

**Step 3** Create a trust point.

```
Device-1(config)# crypto ca trustpoint myCA  
Device-1(config-trustpoint)# exit  
Device-1(config)# show crypto ca trustpoints  
trustpoint: myCA; key:  
revokation methods:  crl
```

**Step 4** Create an RSA key pair for the device.

```
Device-1(config)# crypto key generate rsa label myKey exportable modulus 1024  
Device-1(config)# show crypto key mypubkey rsa  
key label: myKey  
key size: 1024  
exportable: yes
```

**Step 5** Associate the RSA key pair to the trust point.

```
Device-1(config)# crypto ca trustpoint myCA  
Device-1(config-trustpoint)# rsakeypair myKey  
Device-1(config-trustpoint)# exit  
Device-1(config)# show crypto ca trustpoints  
trustpoint: myCA; key: myKey  
revokation methods:  crl
```

**Step 6** Download the CA certificate from the Microsoft Certificate Service web interface.

**Step 7** Authenticate the CA that you want to enroll to the trust point.

```
Device-1(config)# crypto ca authenticate myCA  
input (cut & paste) CA certificate (chain) in PEM format;
```

```

end the input with a line containing only END OF INPUT :
-----BEGIN CERTIFICATE-----
MIIC4jCCAoygAwIBAgIQBWDSiay0GZRPRI1jK0ZejaNBgkqhkiG9w0BAQUFADCB
kDEgMB4GCSqGSIb3DQEJARYRYW1hbmRrZUBjaXNjby5jb20xMzA1MjEwMjEwMjEw
MRIwEAYDVQQIEw1LYXJuYXRha2ExEjAQBGNVBAcTCUJhbmdbG9yZTEOMAwGA1UE
ChMFQ21zY28xZzARBGNVBAStCm5ldHN0b3JhZ2UxEjAQBGNVBAMTCUFWYXJuYSBD
QTAEFw0wNTA1MDMyMjQ2MzdaFw0wNzA1MDMyMjU1MTdaMIGQMSAwHgYJKoZIhvcN
AQkBFhFhbWVuzGt1QGNpc2NvLmNvbTElMAkGA1UEBHMCSU4xEjAQBGNVBAcTCUth
cm5hdGFrYTESMBAGA1UEBxMJQmFuZ2Fsb3JlMQ4wDAYDVQQKEwVdaxNjbjETMBEG
A1UECzMKbmV0c3RvcnFnZTESMBAGA1UEAxMJQXBhcm5hIENBMFwwDQYJKoZIhvcN
AQEBBQADSwAwSAJBAMW/7b3+DXJPANBsIHHz1uNccNM87ypyzwuoSNZXOMpeRXXI
OzyBAgiXT2ASFuUowQ1iDM8rO/41jf8RxxvYKvysCAwEAAaOBvzCBvDALBGNVHQ8E
BAMCAcYwDwYDVR0TAQH/BAUwAwEB/zAdBgNVHQ4EFgQUUjyR0mbrCNMRU2OyRhQ
GgsWbHEwawYDVR0fBGQwYjAuoCygKoYoahrOcdovL3NzZS0wOC9DZXJ0RW5yb2xs
L0FwYXJuYXUyMENBLmNybDAwOC6gLIYqZmlsZTovL1xccc3N1LTA4XEN1cnRfbnJv
bGxcQXBhcm5hJT1wQ0EuY3JsbGAGCSsGAQQBQjJcVAQDAGgEAMA0GCSqGSIb3DQEB
BQUAA0EAHv6UQ+8nE399Tww+KaGr0gONIJaNgLh0AFcT0rEyuyt/WYGPzksF9EA
NBG7E0oN66zex0EOEFG1Vs6mXp1//w==
-----END CERTIFICATE-----
END OF INPUT
Fingerprint(s): MD5 Fingerprint=65:84:9A:27:D5:71:03:33:9C:12:23:92:38:6F:78:12
Do you accept this certificate? [yes/no]:y

```

```

Device-1(config)# show crypto ca certificates
Trustpoint: myCA
CA certificate 0:
subject= /emailAddress=admin@yourcompany.com/C=IN/ST=Karnataka/
L=Bangalore/O=Yourcompany/OU=netstorage/CN=Aparna CA
issuer= /emailAddress=admin@yourcompany.com/C=IN/ST=Karnataka/
L=Bangalore/O=Yourcompany/OU=netstorage/CN=Aparna CA
serial=0560D289ACB419944F4912258CAD197A
notBefore=May 3 22:46:37 2005 GMT
notAfter=May 3 22:55:17 2007 GMT
MD5 Fingerprint=65:84:9A:27:D5:71:03:33:9C:12:23:92:38:6F:78:12
purposes: sslserver sslclient ike

```

## Step 8 Generate a request certificate to use to enroll with a trust point.

```

Device-1(config)# crypto ca enroll myCA
Create the certificate request ..
Create a challenge password. You will need to verbally provide this
password to the CA Administrator in order to revoke your certificate.
For security reasons your password will not be saved in the configuration.
Please make a note of it.
Password: nbv123
The subject name in the certificate will be: Device-1.cisco.com
Include the switch serial number in the subject name? [yes/no]: no
Include an IP address in the subject name [yes/no]: yes
ip address: 10.10.1.1
The certificate request will be displayed...
-----BEGIN CERTIFICATE REQUEST-----
MIIBqzCCARQCAQAwHDEaMBGGA1UEAxMRVnVnYXNjby5jb20wZz8wDQYJ
KoZIhvcNAQEBBQADgY0AMIGJAoGBAL8Y1UAJ2NC7jUJ1DVaSMQnIgj2kt8r141KY
0JC6ManNy4qxk8VeMXZSiLJ4JgTzKWdxBLDkTTysnjucXGvjw+wbj0hEhv/y51T9y
P2NJ8ornqShrvfZgC7ysN/PyMwKcgzhhVpj+rargZvHtGJ91XTq4WoVksCzXv8S
VqyH0vEvAgMBAAGgTzAVBgkqhkiG9w0BCQcxCBMGBmJ2MTIzMDYGCsqGSIb3DQEJ
DjEpMCcwJQYDVR0RAQH/BSswGYIRVnVnYXNjby5jb22HBKwWH6IwDQYJ
KoZIhvcNAQEBBQADgYEAKT60KER6Qo8nj0sDXZVHsfJZh6K6JtDz3Gkd99G1FWgt
PfrNcWUE/pw6HayfQ12T3ecgNwe12d15133YBF2bktExiI6U188nTOjglXMjja8
8a23bNDpNsM8rklwA6hWkrVL8NUZEFJxqbjfngPNTZacJCUS6ZqKCMetbKytUx0=
-----END CERTIFICATE REQUEST-----

```

**Step 9** Request an identity certificate from the Microsoft Certificate Service web interface.

**Step 10** Import the identity certificate.

```
Device-1(config)# crypto ca import myCA certificate
input (cut & paste) certificate in PEM format:
-----BEGIN CERTIFICATE-----
MIIEADCCA6ggAwIBAgIKCj00oQAAAAAdDANBgkqhkiG9w0BAQUFADCBkDEgMB4G
CSqGSIB3DQEJARYRYW1hbmRrZUBjaXNjby5jb20xCzAJBgNVBAYTAK1OMRIWEAYD
VQIQIEWLlYXJuYXRha2ExEjAQBGNVBAcTCUJhbmdhbG9yZTEOMAwwG1UEChMFQ2lZ
Y28xEzARBGNVBAsTCm5ldHN0b3JhZ2UxEjAQBGNVBAMTCUFwYXJ5SBDQTAeFw0w
NTExMTIwMzAyNDBaFw0wNjExMTIwMzEyNDBaMBwxGjAYBgNVBAMTEVZlZ2FzLTFE
Y2lZyY28uY29tMIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQC/GNVAcDjQu41C
dQ1WkjkjSICdpLfk5eJSmNCQujGpzcKsZPFXjF2UoieCYE8ylncWyw5E08rJ47
glxr42/sI9IRIb/8udU/cj9jSSfKK56koa7xWYA8rDfz8jMcNIM4W1aY/q2q4Gb
x7Ri.fdv06uFqFZEgs17/Elash9LxLwIDAQABo4ICEzCCA8wJQYDVR0RAQH/BBsw
GYIRVmnVnYXMTMS5jaXNjby5jb22HBKwWH6IwHQYDVR0OBBYEFKCLi+2sspWEfgrR
bhWmlVyo9jngMIHMBGNVHSMEGcQwgcGAFCCo8kaDG6wjTEVNjSkYUBoLFmxxoYGW
pIGTMIGQMSAwHgYJKoZiIhvcNAQkBFhFhbWFuZGt1QGNpc2NvLmNvbTELMakGA1UE
BhMCSU4xEjAQBGNVBAcTCUthcm5hdGFryTESMBAGA1UEBxMJQmFuZ2Fsb3JlMQ4w
DAYDVQQKEWVdaXNjbzETMBEGA1UECXMKBmV0c3RvcmluZTEsMBAGA1UEAxMJQXBh
cm5hIENBghAFYNKJrLQZLE9JEiWMrR16MGsGA1UdHwRkMGIlwLqAsocqGKGh0dHA6
Ly9zc2UtdGvQ2VydEVucm9sbC9BcGFybmlmBDQ55jcmwwMKAuoCyGKmZpbGU6
Ly9cXHNzZS0wOFxDZlJ0RW5yb2xsXEFwYXJ5SUYMENBLmNybDcBiGyIKwYBBQUH
AQEEfjB8MDsGCCsGAQUFBzAChi9odHRwOi8vc3NlLTA4L0N1cnRFbnJvbGwvc3Nl
LTA4X0FwYXJ5SUYMENBLmNydDA9BggrBgEFBQcwoAoYxZmlsZTovL1xccc3NlLTA4
XEN1cnRFbnJvbGxccc3NlLTA4X0FwYXJ5SUYMENBLmNydDANBgkqhkiG9w0BAQUF
AANBADbGBGsbse7GNLh9xeOTWBNbm24U69ZSuDDcOcuZUUTgrpnTqVpPyejtsyflw
E36cIZu4WsExREqxbTk8ycx7V5o=
-----END CERTIFICATE-----
Device-1(config)# exit
Device-1#
```

**Step 11** Verify the certificate configuration.

**Step 12** Save the certificate configuration to the startup configuration.

### Related Topics

[Downloading a CA Certificate](#), on page 29

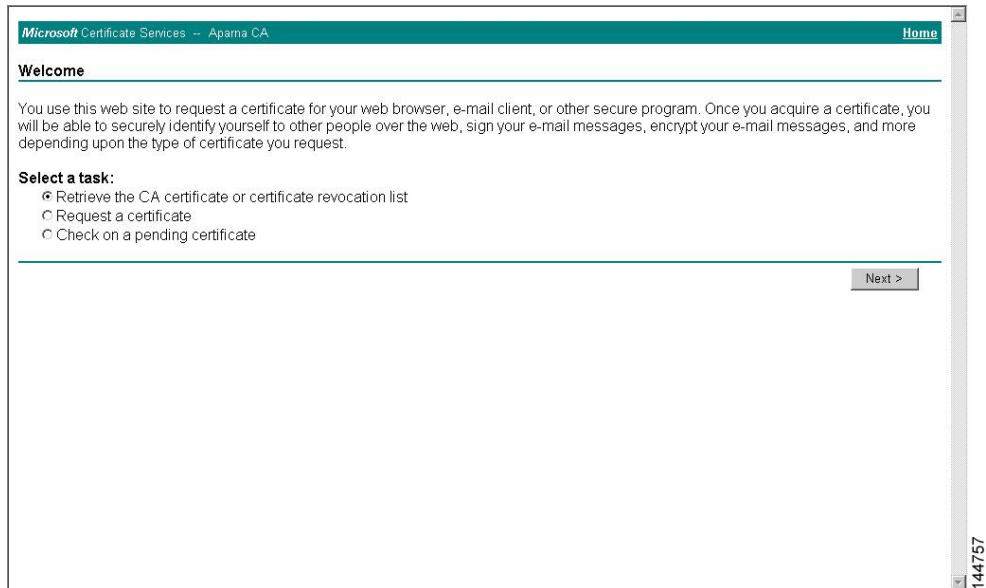
[Requesting an Identity Certificate](#), on page 33

## Downloading a CA Certificate

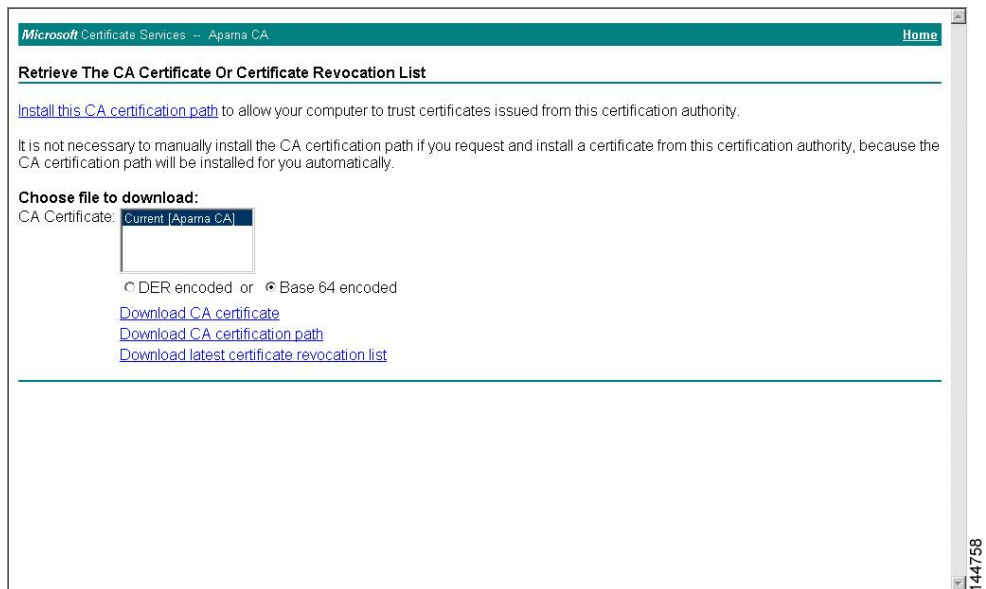
To download a CA certificate from the Microsoft Certificate Services web interface, follow these steps:

## Procedure

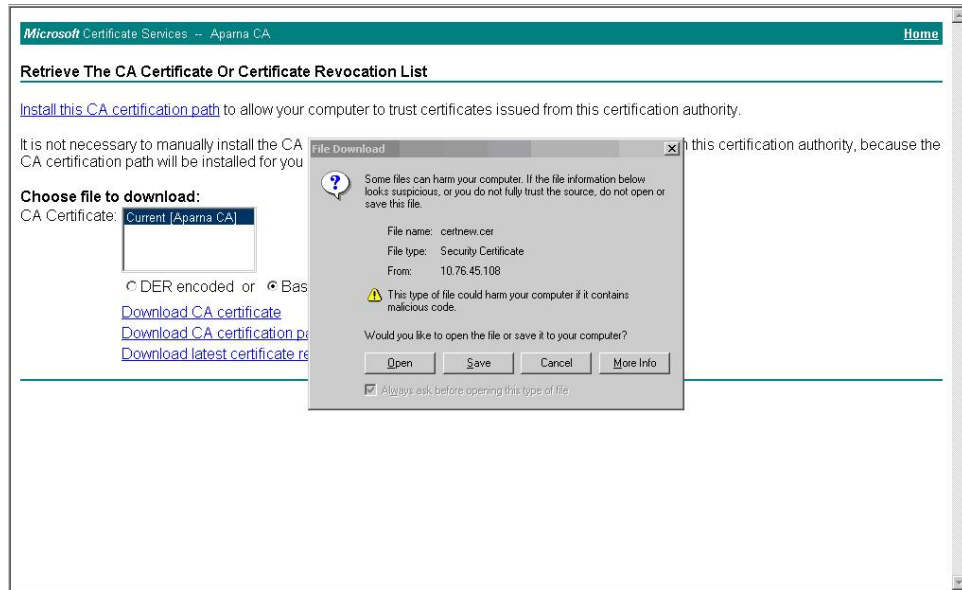
- Step 1** From the Microsoft Certificate Services web interface, click **Retrieve the CA certificate or certificate revocation task** and click **Next**.



- Step 2** From the display list, choose the CA certificate file to download from the displayed list. Then click **Base 64 encoded** and click **Download CA certificate**.

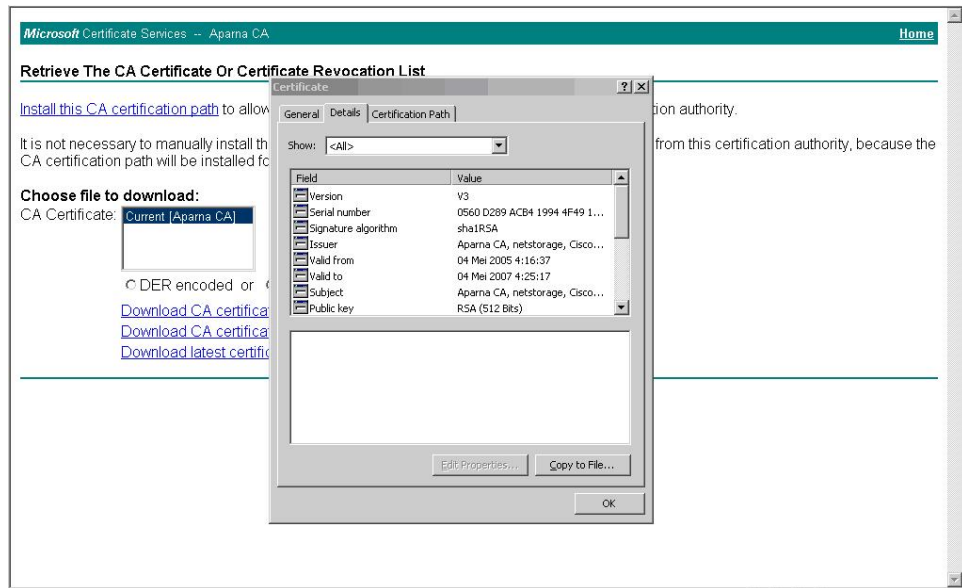


**Step 3** Click **Open** in the File Download dialog box.



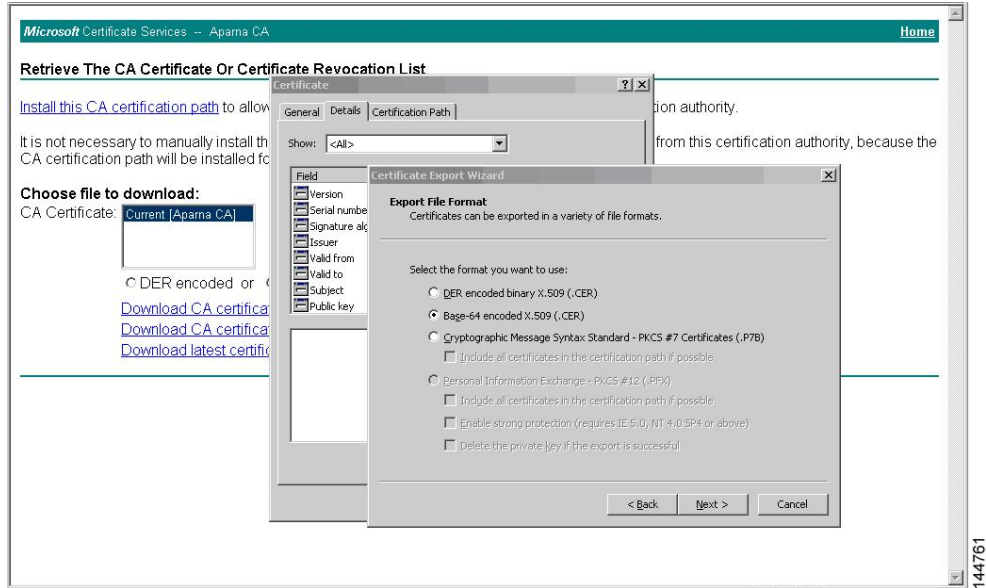
144759

**Step 4** In the Certificate dialog box, click **Copy to File** and click **OK**.



144760

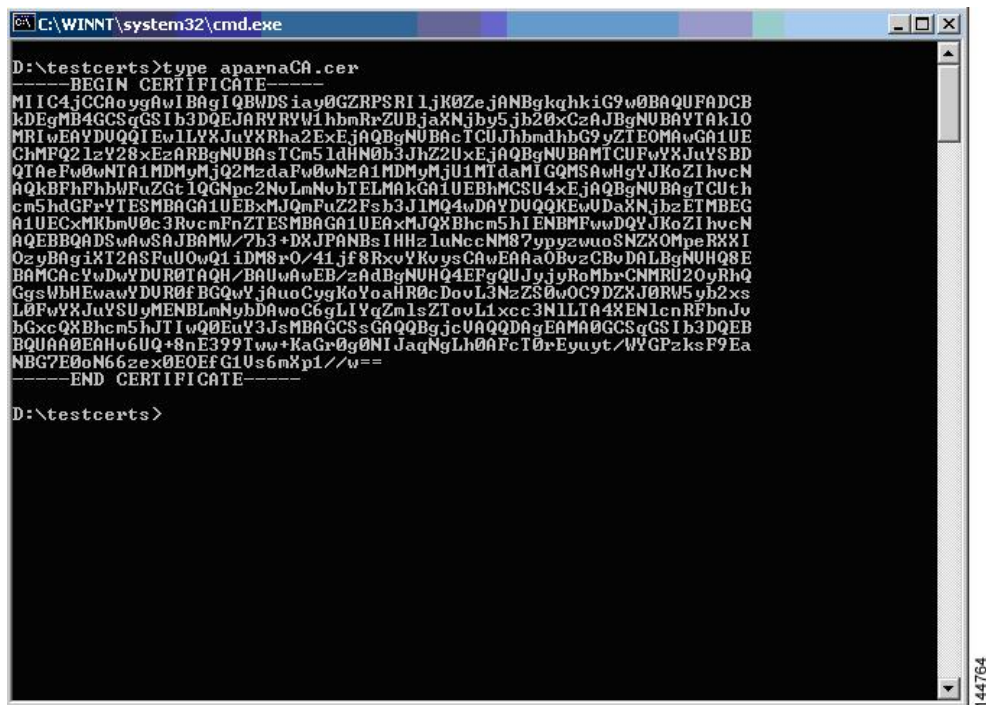
**Step 5** From the Certificate Export Wizard dialog box, choose the **Base-64 encoded X.509 (CER)** and click **Next**.



**Step 6** In the File name: text box on the Certificate Export Wizard dialog box, enter the destination file name and click **Next**.

**Step 7** In the Certificate Export Wizard dialog box, click **Finish**.

**Step 8** Enter the Microsoft Windows **type** command to display the CA certificate stored in Base-64 (PEM) format.



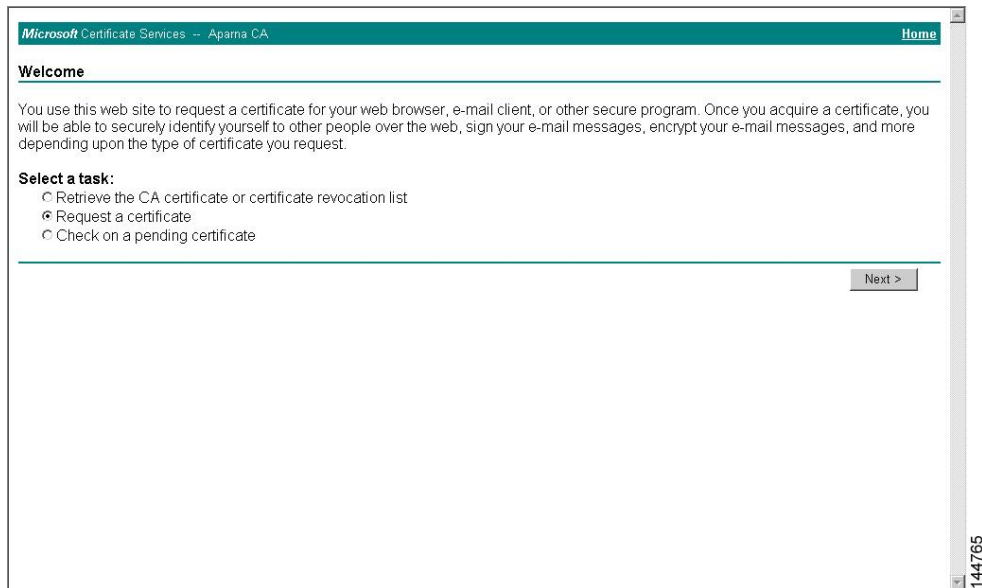


# Requesting an Identity Certificate

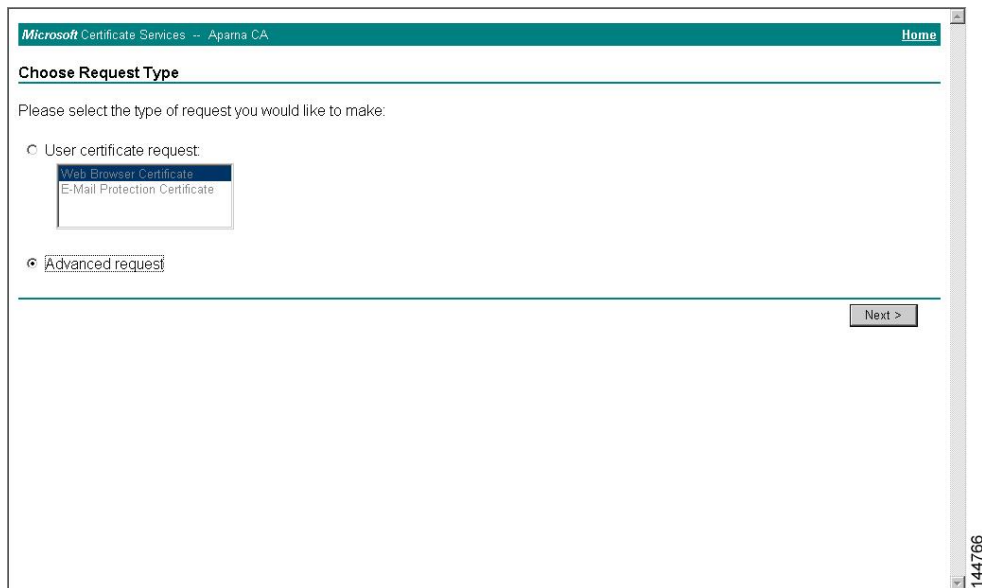
To request an identify certificate from a Microsoft Certificate server using a PKCS#12 certificate signing request (CRS), follow these steps:

## Procedure

**Step 1** From the Microsoft Certificate Services web interface, click **Request a certificate** and click **Next**.



**Step 2** Click **Advanced request** and click **Next**.



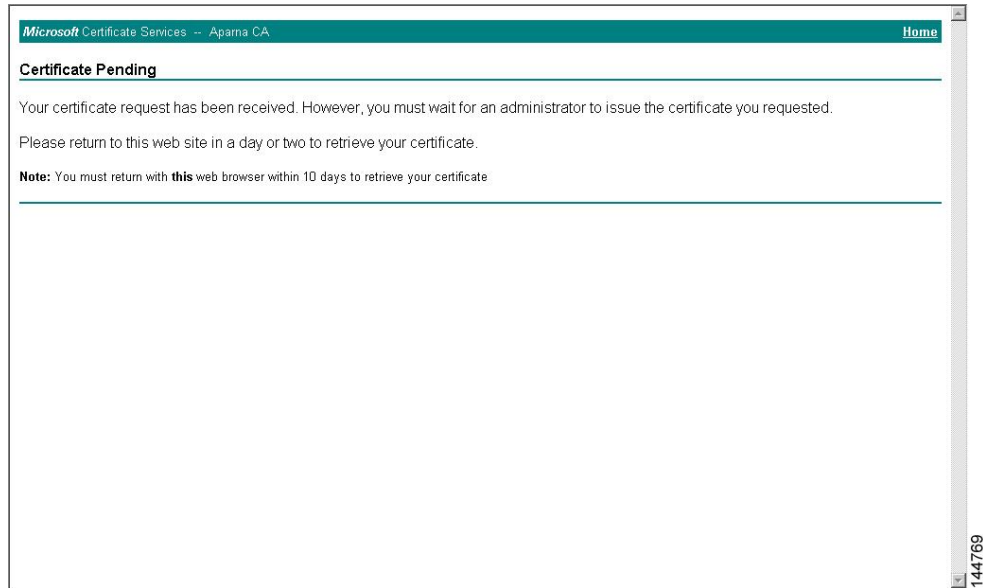
**Step 3**

Click **Submit a certificate request using a base64 encoded PKCS#10 file or a renewal request using a base64 encoded PKCS#7 file** and click **Next**.

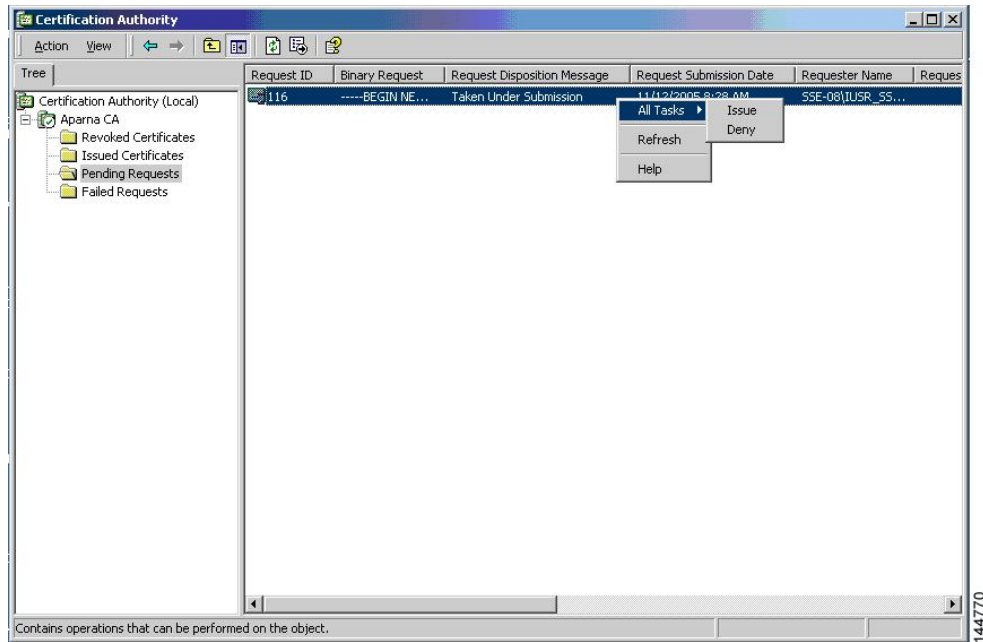
**Step 4**

In the **Saved Request** text box, paste the base64 PKCS#10 certificate request and click **Next**. The certificate request is copied from the Cisco NX-OS device console.

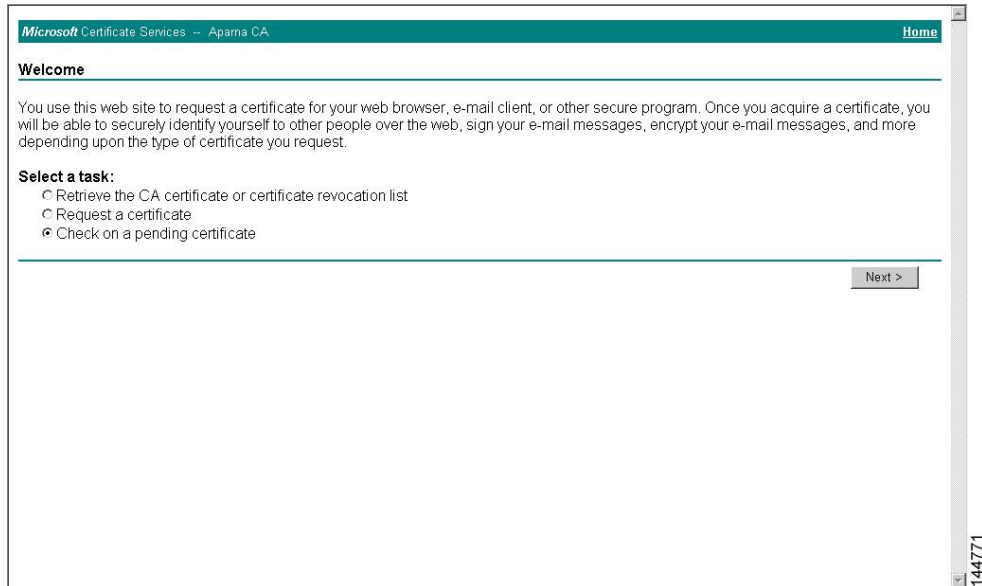
**Step 5** Wait one or two days until the certificate is issued by the CA administrator.



**Step 6** Note that the CA administrator approves the certificate request.

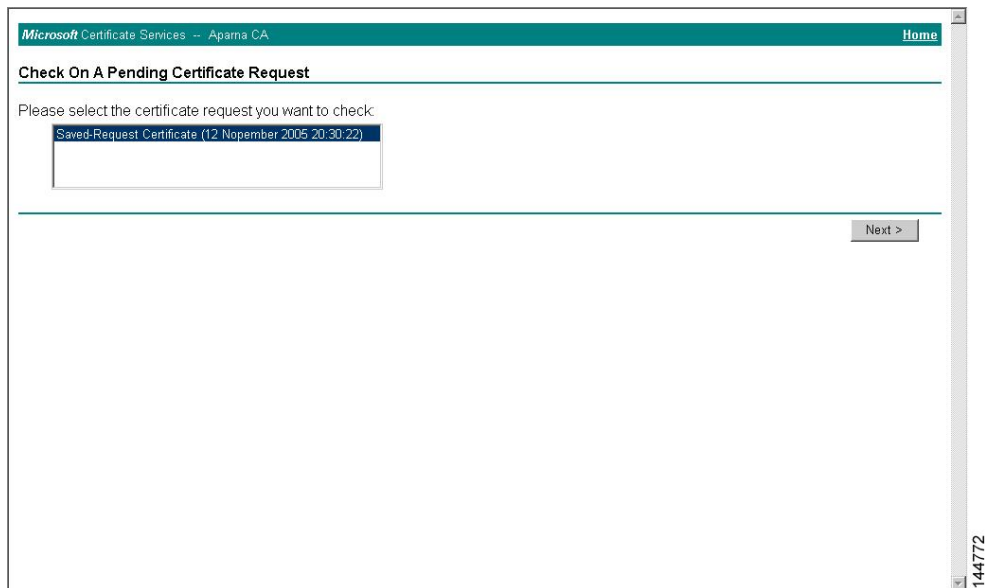


**Step 7** From the Microsoft Certificate Services web interface, click **Check on a pending certificate** and click **Next**.



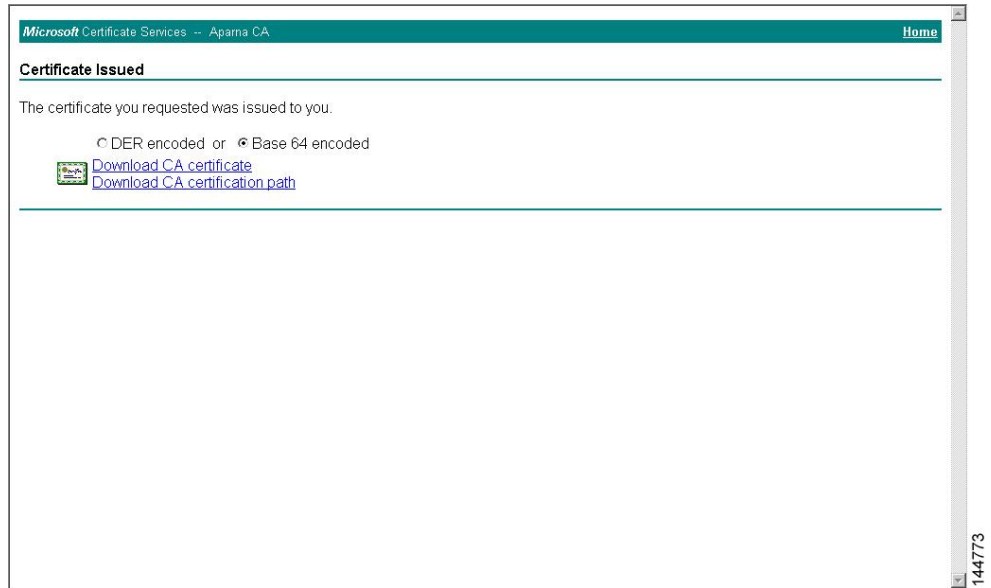
The screenshot shows the Microsoft Certificate Services web interface for the 'Aparna CA'. The page title is 'Microsoft Certificate Services -- Aparna CA' and there is a 'Home' link in the top right. The main heading is 'Welcome'. Below this, a paragraph explains the site's purpose: 'You use this web site to request a certificate for your web browser, e-mail client, or other secure program. Once you acquire a certificate, you will be able to securely identify yourself to other people over the web, sign your e-mail messages, encrypt your e-mail messages, and more depending upon the type of certificate you request.' Under the heading 'Select a task:', there are three radio button options: 'Retrieve the CA certificate or certificate revocation list', 'Request a certificate', and 'Check on a pending certificate'. The 'Check on a pending certificate' option is selected. A 'Next >' button is located at the bottom right of the main content area. A vertical scrollbar on the right side of the page shows the value '144771'.

**Step 8** Choose the certificate request that you want to check and click **Next**.

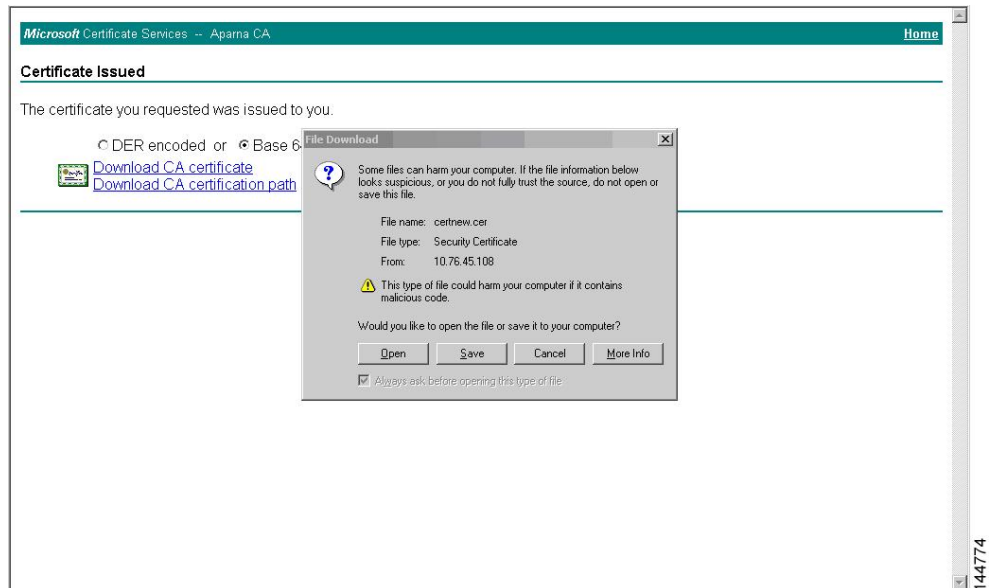


The screenshot shows the Microsoft Certificate Services web interface for the 'Aparna CA'. The page title is 'Microsoft Certificate Services -- Aparna CA' and there is a 'Home' link in the top right. The main heading is 'Check On A Pending Certificate Request'. Below this, a paragraph says 'Please select the certificate request you want to check:'. There is a list box containing one item: 'Saved-Request Certificate (12 November 2005 20:30:22)'. A 'Next >' button is located at the bottom right of the main content area. A vertical scrollbar on the right side of the page shows the value '144772'.

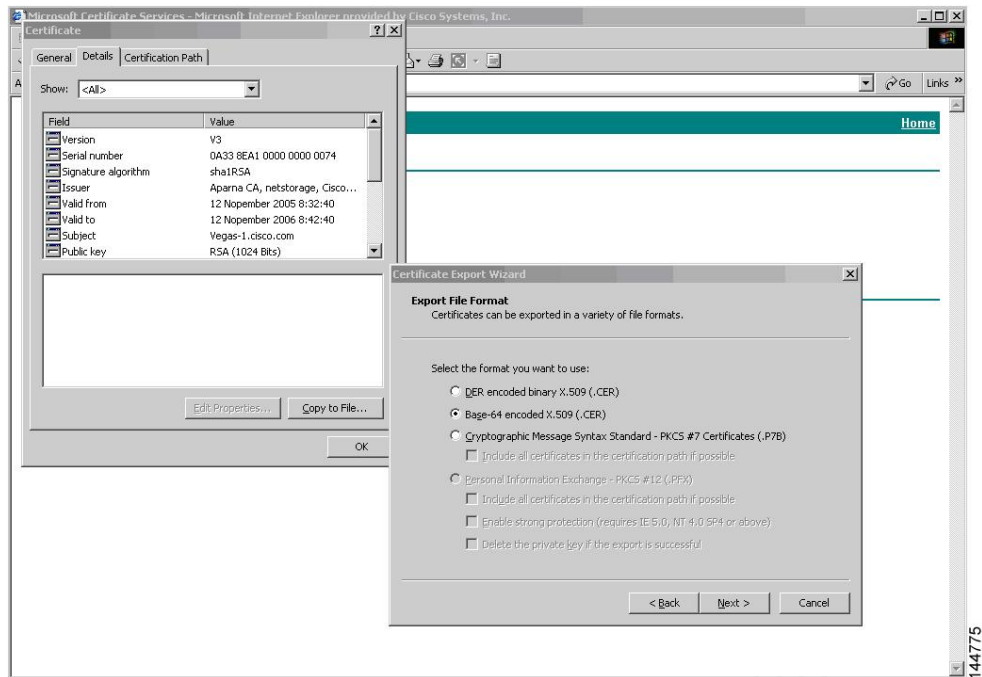
**Step 9** Click **Base 64 encoded** and click **Download CA certificate**.



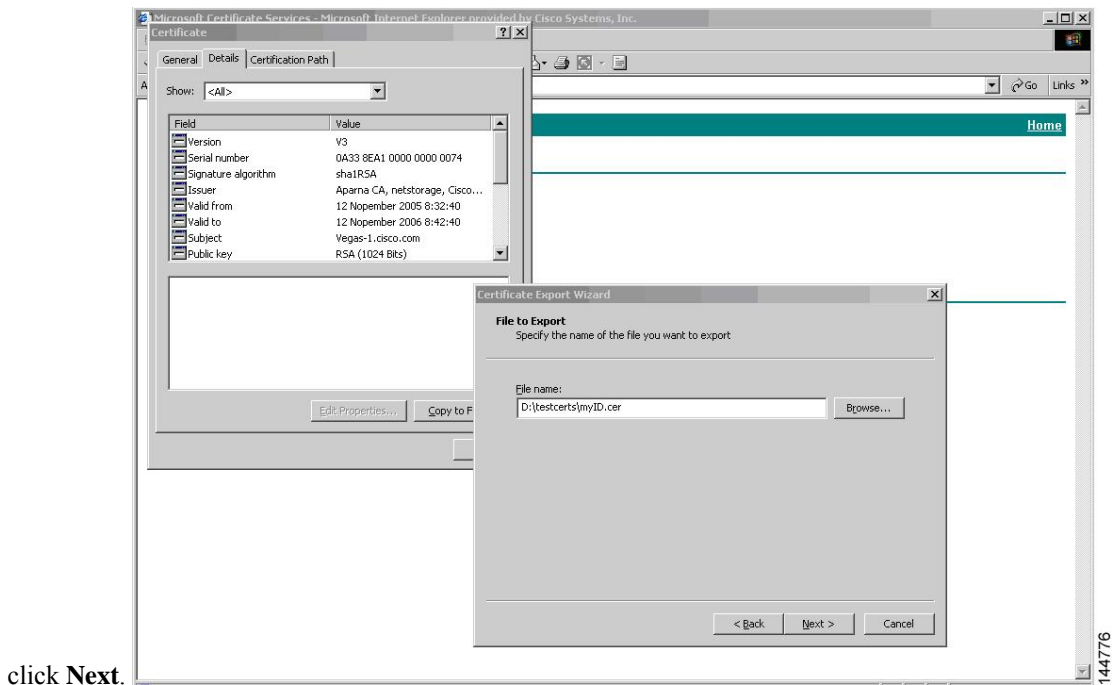
**Step 10** In the File Download dialog box, click **Open**.



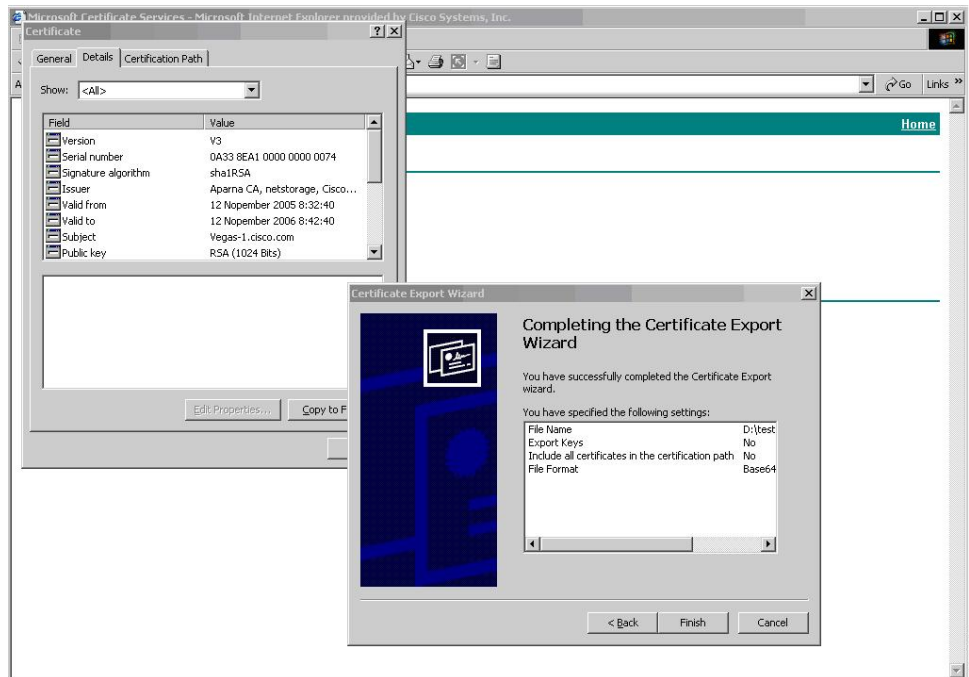
**Step 11** In the Certificate box, click **Details** tab and click **Copy to File...**. In the Certificate Export Dialog box, click **Base-64 encoded X.509 (.CER)**, and click **Next**.



**Step 12** In the File name: text box on the Certificate Export Wizard dialog box, enter the destination file name and

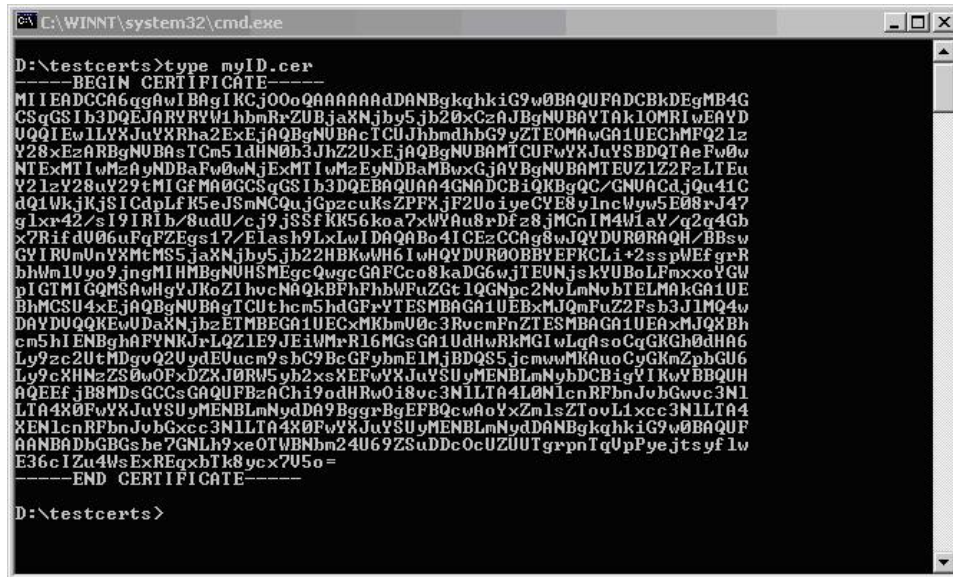


click **Next**.



**Step 13** Click **Finish**.

**Step 14** Enter the Microsoft Windows **type** command to display the identity certificate in base64-encoded format.



**Related Topics**

- [Generating Certificate Requests](#), on page 18
- [Configuring Certificates on a Cisco NX-OS Device](#), on page 27

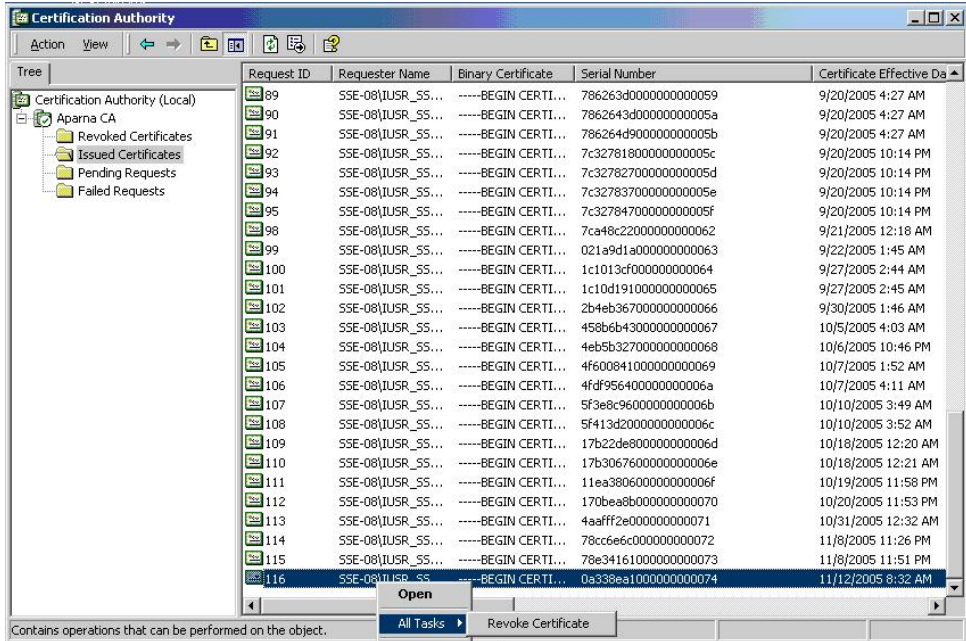
## Revoking a Certificate

To revoke a certificate using the Microsoft CA administrator program, follow these steps:

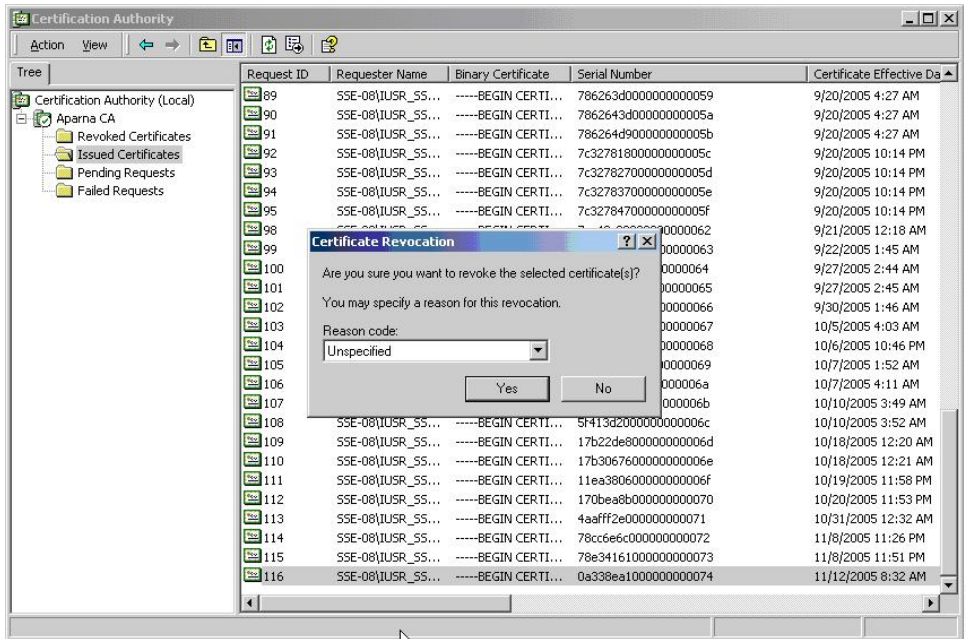
Procedure

**Step 1** From the Certification Authority tree, click **Issued Certificates** folder. From the list, right-click the certificate that you want to revoke.

**Step 2** Choose **All Tasks > Revoke Certificate**.

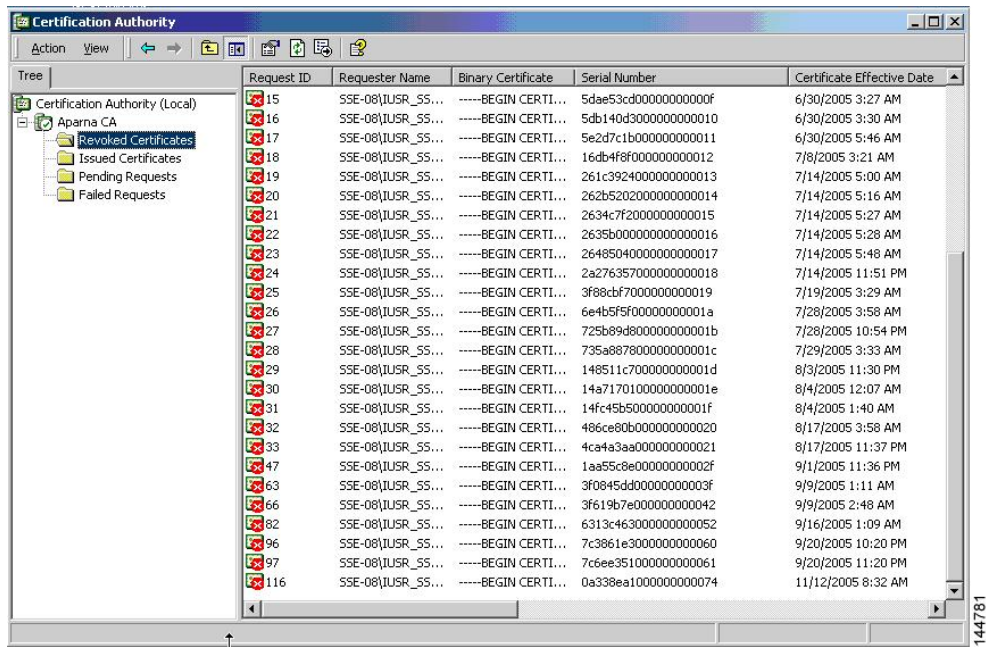


**Step 3** From the Reason code drop-down list, choose a reason for the revocation and click **Yes**.





**Step 4** Click the **Revoked Certificates** folder to list and verify the certificate revocation.

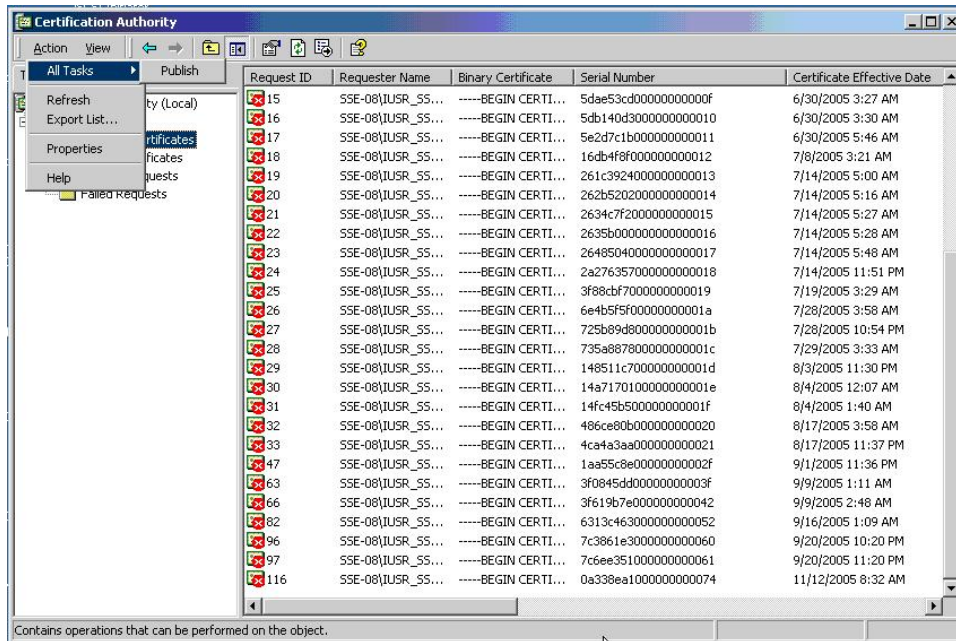


## Generating and Publishing the CRL

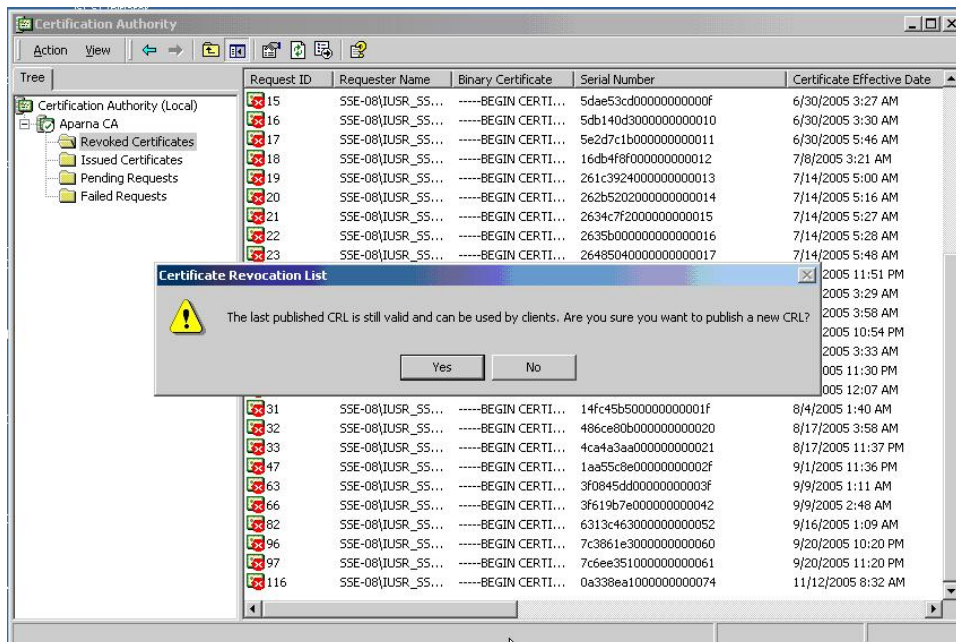
To generate and publish the CRL using the Microsoft CA administrator program, follow these steps:

Procedure

**Step 1** From the Certification Authority screen, choose **Action > All Tasks > Publish**.



**Step 2** In the Certificate Revocation List dialog box, click **Yes** to publish the latest CRL.

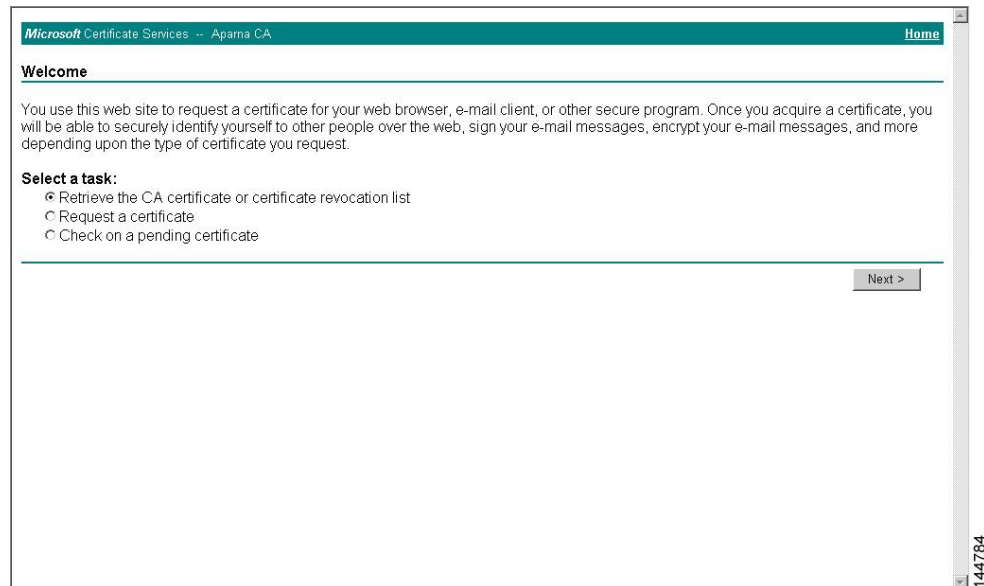


# Downloading the CRL

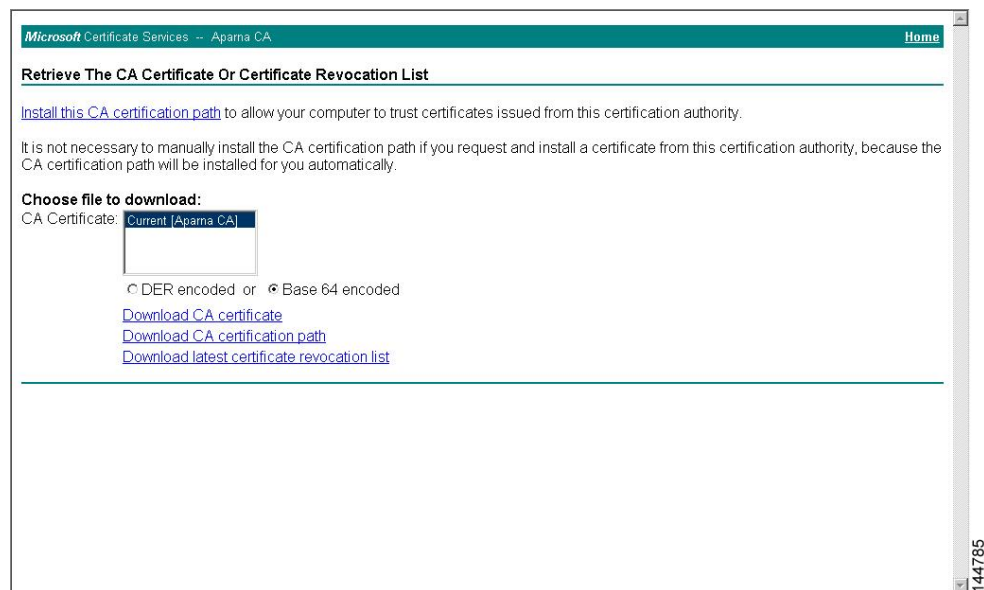
To download the CRL from the Microsoft CA website, follow these steps:

## Procedure

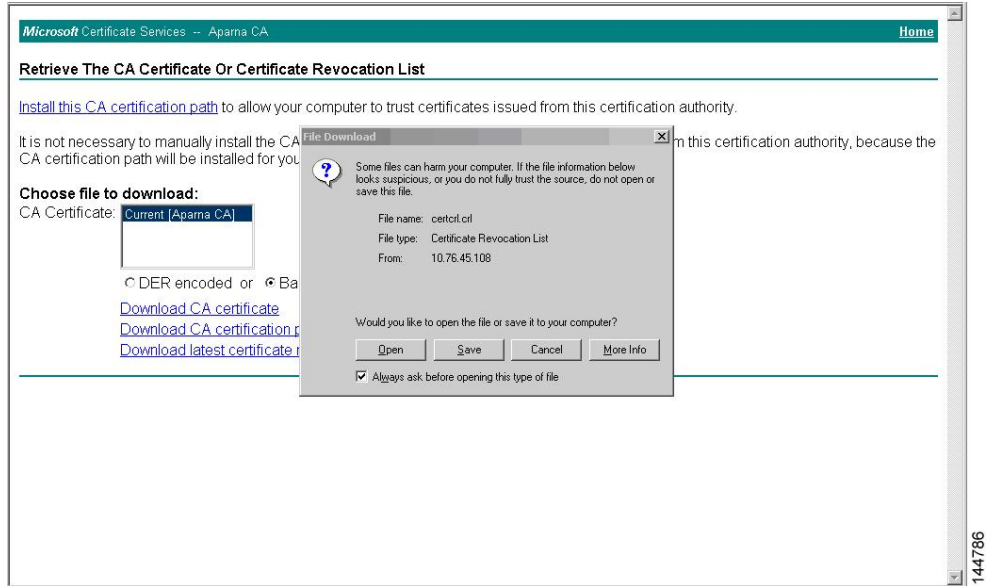
- Step 1** From the Microsoft Certificate Services web interface, click **Retrieve the CA certificate or certificate revocation list** and click **Next**.



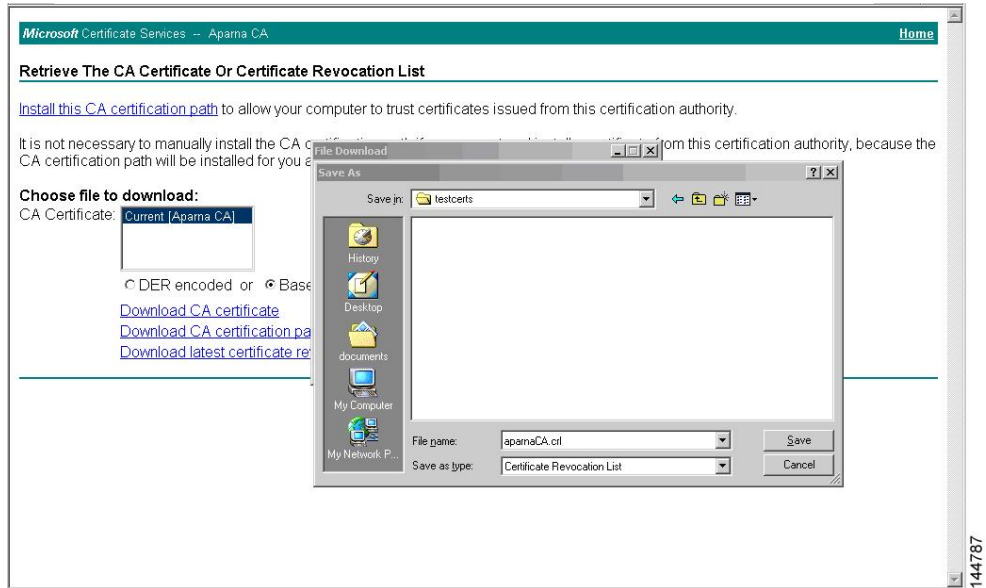
- Step 2** Click **Download latest certificate revocation list**.



**Step 3** In the File Download dialog box, click **Save**.



**Step 4** In the Save As dialog box, enter the destination file name and click **Save**.



**Step 5** Enter the Microsoft Windows **type** command to display the CRL.

```

C:\WINNT\system32\cmd.exe
D:\testcerts>type aparnaCA.crl
-----BEGIN X509 CRL-----
MIIGBTCCBa8CAQEWdQVJKoZIHvCNAQEFBQAwZAXIDAeBgkqhkiG9w0BCQEWEMFt
YW5ka2UAY21zY28uY29tMQswCQYDUQGEwJITjESMBAGA1UECBMJS2FybmF0YVth
MRIwEAYDUQOHw1CYW5nYWxvcmluXDJjAMBgNURBAoIBUNpc2NvMRRwEQYDUQOLEWpu
ZXRzdG9yYVdlbMRIwEAYDUQODEwLBCGFybmEgQ00EXDTA1MTExMjA0MzYwNFoXDTA1
MTEyOTExMTEyYWNFowggSxMBsCCmEBCAFAAAAAAAAAIXDTA1MDg4NjI1xNTI1xOUowGwIK
IN5TgaAAAAAAAAxcNMDUwODE2MjE1MjE1WjAbaGppM/CtCAAAAAAAAAEFw0wNTA4MTYy
MTUyNDFAmBScCCmXpnsIAAAAAAAAAUXDTA1MDg4NjI1xNTI1MlowGwIKbM993AAAAAAAA
BhcNMDUwNjA4MDAxMjA0WjAbaGppwzE/AAAAAAAAHFw0wNTA4MTYyMTUzMTUaMBSc
Ck2bERYAAAAAAAAgXDTA1MDg4NjI1xNTMxNUowKQIKUggCAAAAAAAAAACRcNMDUwNjI3
MjM0NDA2WjAMMAoGA1UdFQDDCgEChCkCCINjxUYAAAAAAAAoXDTA1MDYyMzIzNDcy
MlowDDAKBgnUHRUEawBAjAbaGppT0Rc8AAAAAAAAALFw0wNTA3MDQxODAwMDFAmAw
CgYDUROUBAMKAQYwGwIKWR56zgAAAAAAAAADBCNMDUwODE2MjE1MzE1WjAbaGppdP9Uu
AAAAAAAAANFw0wNTA2MjkyMjA3MjUaMAwwCgYDUROUBAMKAQEWGwIKKXat3EwAAAAA
DhcNMDUwNzE0MDAzMzU2WjAbaGppdr1PNAAAAAAAAAFw0wNTA4MTYyMTUzMTUaMBSc
C12xQNMAAAAAAAAABAxDTA1MDg4NjI1xNTMxNUowKQIKXii18GwAAAAAAAAERcNMDUwNzA2
MjE1WjEwWjAMMAoGA1UdFQDDCgEFMBScCChbbT48AAAAAAAAIBXDTA1MDg4NjI1xNTMx
NUowGwIKJhw5JAAAAAAAAEXcNMDUwODE2MjE1MzE1WjAbaGppMzI1CAAAAAAAAAFw0w
NTA3MTQwMDMzMTBaMBScCciY0x/IAAAAAAAAABUXDTA1MDc4NDAwMzI0NUowGwIKJjW
AAAAAAAAAFhcNMDUwNzE0MDAzMTUxWjAbaGppSFBAAAAAAAAAXFw0wNTA3MTQwMDM4
MjUaMBScCcionY1cAAAAAAAABgXDTA1MDg4NjI1xNTMxNUowGwIKP4jL9wAAAAAAAAGRcN
MDUwODE2MjE1MzE1WjAbaGppuS19fAAAAAAAAaFw0wNTA4MTYyMTUzMTUaMBScCnJb
idgAAAAAAAABsXDTA1MDg4NjI1xNTMxNUowGwIKc1q1eAAAAAAAAAHBcNMDUwODE2MjE1
MzE1WjAbaGpUhhRHAAAAAAAAAdFw0wNTA4MTYyMTUzMTUaMBScCChSnFwEAAAAAAAAAB4X
DTA1MDg4NjI1xNTMxNUowGwIKFPxftQAAAAAAAAHxcNMDUwODE3MTgzMDQyWjAbaGppI
b0GLAAAAAAAAAgFw0wNTA4MTc1ODMwNDNaMBScCkyko6oAAAAAAAAACEXDTA1MDg4NzE4
MzA0M1owGwIKGGuCjgAAAAAAAAALxcNMDUwOTA1MTcwNzA2WjAbaGp0/CEXAAAAAAAA/
Fw0wNTA5MDg4MDI0MzJhMBScCj9hm34AAAAAAAAEIXDTA1MDg4ODI1xNDAA00FowGwIK
YxPEYwAAAAAAAAUhcNMDUwOTE5MTczNzE4WjAbaGpp8OGHjAAAAAAAABgFw0wNTA5MjAx
NzUyNTZaMBScCcxu41EAAAAAAAAGEXDTA1MDk4MDE4NTIzMFowGwIKCj00oQAAAAAAAA
dBcNMDUxMTEyMDQzNDQyYUa1MDMwHwYDUROUjBBgwFoAUJyJyRoMbrCNMRU20vRhQ
GgsWbhEwEAYJKoYBBAGCNxUBBAMCAQAwDQYJKoZIhvcNAQEFBQADQALy91DCshi
HoCUBm9NgwYjJjEjEjU168CuaacFP3rkM8YyZYpu1c32R/UuU6a5xgrAC/SbsEa
nXpJt5xYJNdY
-----END X509 CRL-----
D:\testcerts>

```

## Related Topics

[Configuring Certificate Revocation Checking Methods](#), on page 16

## Importing the CRL

To import the CRL to the trust point corresponding to the CA, follow these steps:

### Procedure

**Step 1** Copy the CRL file to the Cisco NX-OS device bootflash.

```
Device-1# copy tftp:aparnaCA.crl bootflash:aparnaCA.crl
```

**Step 2** Configure the CRL.

```
Device-1# configure terminal
Device-1(config)# crypto ca crl request myCA bootflash:aparnaCA.crl
Device-1(config)#
```

**Step 3** Display the contents of the CRL.

```
Device-1(config)# show crypto ca crl myCA
Trustpoint: myCA
```

```

CRL:
Certificate Revocation List (CRL):
  Version 2 (0x1)
  Signature Algorithm: sha1WithRSAEncryption
  Issuer: /emailAddress=admin@yourcompany.com/C=IN/ST=Karnatak
Yourcompany/OU=netstorage/CN=Aparna CA
  Last Update: Nov 12 04:36:04 2005 GMT
  Next Update: Nov 19 16:56:04 2005 GMT
  CRL extensions:
    X509v3 Authority Key Identifier:
      keyid:27:28:F2:46:83:1B:AC:23:4C:45:4D:8E:C9:18:50:1
      1.3.6.1.4.1.311.21.1:
        ...
Revoked Certificates:
  Serial Number: 611B09A1000000000002
    Revocation Date: Aug 16 21:52:19 2005 GMT
  Serial Number: 4CDE464E000000000003
    Revocation Date: Aug 16 21:52:29 2005 GMT
  Serial Number: 4CFC2B42000000000004
    Revocation Date: Aug 16 21:52:41 2005 GMT
  Serial Number: 6C699EC2000000000005
    Revocation Date: Aug 16 21:52:52 2005 GMT
  Serial Number: 6CCF7DDC000000000006
    Revocation Date: Jun  8 00:12:04 2005 GMT
  Serial Number: 70CC4FFF000000000007
    Revocation Date: Aug 16 21:53:15 2005 GMT
  Serial Number: 4D9B1116000000000008
    Revocation Date: Aug 16 21:53:15 2005 GMT
  Serial Number: 52A80230000000000009
    Revocation Date: Jun 27 23:47:06 2005 GMT
    CRL entry extensions:
      X509v3 CRL Reason Code:
        CA Compromise
  Serial Number: 5349AD4600000000000A
    Revocation Date: Jun 27 23:47:22 2005 GMT
    CRL entry extensions:
      X509v3 CRL Reason Code:
        CA Compromise
  Serial Number: 53BD173C00000000000B
    Revocation Date: Jul  4 18:04:01 2005 GMT
    CRL entry extensions:
      X509v3 CRL Reason Code:
        Certificate Hold
  Serial Number: 591E7ACE00000000000C
    Revocation Date: Aug 16 21:53:15 2005 GMT
  Serial Number: 5D3FD52E00000000000D
    Revocation Date: Jun 29 22:07:25 2005 GMT
    CRL entry extensions:
      X509v3 CRL Reason Code:
        Key Compromise
  Serial Number: 5DAB771300000000000E
    Revocation Date: Jul 14 00:33:56 2005 GMT
  Serial Number: 5DAE53CD00000000000F
    Revocation Date: Aug 16 21:53:15 2005 GMT
  Serial Number: 5DB140D3000000000010
    Revocation Date: Aug 16 21:53:15 2005 GMT
  Serial Number: 5E2D7C1B000000000011
    Revocation Date: Jul  6 21:12:10 2005 GMT
    CRL entry extensions:
      X509v3 CRL Reason Code:
        Cessation Of Operation
  Serial Number: 16DB4F8F000000000012
    Revocation Date: Aug 16 21:53:15 2005 GMT
  Serial Number: 261C3924000000000013

```

```

    Revocation Date: Aug 16 21:53:15 2005 GMT
  Serial Number: 262B5202000000000014
    Revocation Date: Jul 14 00:33:10 2005 GMT
  Serial Number: 2634C7F2000000000015
    Revocation Date: Jul 14 00:32:45 2005 GMT
  Serial Number: 2635B000000000000016
    Revocation Date: Jul 14 00:31:51 2005 GMT
  Serial Number: 26485040000000000017
    Revocation Date: Jul 14 00:32:25 2005 GMT
  Serial Number: 2A276357000000000018
Revocation Date: Aug 16 21:53:15 2005 GMT
  Serial Number: 3F88CBF7000000000019
    Revocation Date: Aug 16 21:53:15 2005 GMT
  Serial Number: 6E4B5F5F00000000001A
    Revocation Date: Aug 16 21:53:15 2005 GMT
  Serial Number: 725B89D800000000001B
    Revocation Date: Aug 16 21:53:15 2005 GMT
  Serial Number: 735A887800000000001C
    Revocation Date: Aug 16 21:53:15 2005 GMT
  Serial Number: 148511C700000000001D
    Revocation Date: Aug 16 21:53:15 2005 GMT
  Serial Number: 14A7170100000000001E
    Revocation Date: Aug 16 21:53:15 2005 GMT
  Serial Number: 14FC45B500000000001F
    Revocation Date: Aug 17 18:30:42 2005 GMT
  Serial Number: 486CE80B000000000020
    Revocation Date: Aug 17 18:30:43 2005 GMT
  Serial Number: 4CA4A3AA000000000021
    Revocation Date: Aug 17 18:30:43 2005 GMT
  Serial Number: 1AA55C8E00000000002F
    Revocation Date: Sep  5 17:07:06 2005 GMT
  Serial Number: 3F0845DD00000000003F
    Revocation Date: Sep  8 20:24:32 2005 GMT
  Serial Number: 3F619B7E000000000042
    Revocation Date: Sep  8 21:40:48 2005 GMT
  Serial Number: 6313C463000000000052
    Revocation Date: Sep 19 17:37:18 2005 GMT
  Serial Number: 7C3861E3000000000060
    Revocation Date: Sep 20 17:52:56 2005 GMT
  Serial Number: 7C6EE351000000000061
    Revocation Date: Sep 20 18:52:30 2005 GMT
  Serial Number: 0A338EA1000000000074  <-- Revoked identity certificate
    Revocation Date: Nov 12 04:34:42 2005 GMT
  Signature Algorithm: sha1WithRSAEncryption
  0b:cb:dd:43:0a:b8:62:1e:80:95:06:6f:4d:ab:0c:d8:8e:32:
  44:8e:a7:94:97:af:02:b9:a6:9c:14:fd:eb:90:cf:18:c9:96:
  29:bb:57:37:d9:1f:d5:bd:4e:9a:4b:18:2b:00:2f:d2:6e:c1:
  1a:9f:1a:49:b7:9c:58:24:d7:72

```

**Note**

The identity certificate for the device that was revoked (serial number 0A338EA1000000000074) is listed at the end.

## Additional References for PKI

This section includes additional information related to implementing PKI.

## Related Documents for PKI

Related Topic	Document Title
Cisco NX-OS Licensing	<i>Cisco NX-OS Licensing Guide</i>
VRF configuration	<i>Cisco Nexus 9000 Series NX-OS Unicast Routing Configuration Guide</i>

## Standards for PKI

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

## Resource Public Key Infrastructure (RPKI)

RPKI is a globally distributed database that contains information mapping BGP (internet) prefixes to their authorized origin-AS numbers. To validate the origin-AS of BGP paths, routers running BGP can connect to RPKI caches.

The RPKI-Cache-to-Router connectivity can be many-to-many, one RPKI cache can provide origin-AS validation data to multiple routers and one router can be connected to multiple RPKI caches. A router connects to RPKI caches to download information to build a special RPKI database that can be used by BGP to validate the origin-AS numbers for the internet routing table.

The RPKI database is a set of Route-Origin-Attestation (ROA) objects aggregated from the different RPKI caches to which BGP connects. ROA objects provide a mapping between a BGP prefix-block, and an AS number authorized to originate that block.

## RPKI Configuration

RPKI configuration is categorized as:

- commands for connecting to RPKI Caches.
- commands for marking incoming prefixes with RPKI validation state.
- commands for using RPKI validation state in BGP best-path computation.
- commands for dropping out or manipulating prefixes with specific validation states using route-map.

## Commands for connecting to RPKI caches

RPKI cache configuration is done in a new `rpki-cache` submode under the `router-bgp` submode. This is like configuring BGP peers under the default VRF. The submode is entered by using the `"rpki cache <IP address>"` command. When you enter the submode, various parameters for the RPKI cache can be configured.



```

router bgp 100
  rpki cache 147.28.0.11
    description      A description to identify the cache
    shutdown         Shutdown the cache
    transport tcp port Transport port on which cache is listening
    vrf              Vrf in which RPKI cache is reachable
    refresh-interval Specify periodic wait time between cache poll attempts
    retry-interval   Specify wait time before retrying failed serial or reset query
    expiry-interval  Specify how long to use current data while unable to perform successful
query

```




---

**Note** Unless transport TCP port is explicitly configured, BGP will connect to RPKI cache on RPKI-RTR port 323. Unless explicitly configured, all intervals will be determined as suggested by the RPKI Cache in End of Data PDU.

---

## Commands for marking incoming prefixes with RPKI validation state

There are knobs that control the behavior of RPKI prefix validation processing. These knobs can be configured at the address-family level.

- **origin-as validate** - Configured at the address-family level enables eBGP path validation against ROA database. By default, this is disabled.




---

**Note** This command has no bearing on iBGP paths. The iBGP paths are not validated against ROA database. The only way to mark path validation state on iBGP paths is receiving the BGP Prefix Origin Validation State Extended Community, and is done by default without configuring any command.

---

- **origin-as validate signal ibgp** - Configured at the address-family level enables the iBGP signalling of validity state through BGP Prefix Origin Validation State Extended Community.

## Commands for using RPKI validation state in BGP best-path-computation

There are commands to control the behavior of RPKI prefix validation processing. These commands can be configured at the address-family level.

- **bestpath origin-as use-validity** - Configured at the address-family level enables the validity states of BGP paths to affect the path's preference in the BGP bestpath process. By default, this is disabled.
- **bestpath origin-as allow invalid** - Configured at the address-family level allows all "invalid" paths to be considered for BGP bestpath computation (all such paths are not bestpath candidates if best-path origin-as validate is configured). By default, this is disabled.

## Commands for dropping out or manipulating prefixes with specific validation states using route-map

The following is the command for dropping out or manipulating prefixes with specific validation states using route-map:

```
route-map sample1 permit 10
  match rpki {not-found | invalid | valid}
```

The parameters of the match rpki command are described as follows:

- `not-found` - This origin-AS is unknown in the RPKI database.
- `invalid` - This is an invalid origin-AS in the RPKI database.
- `valid` - This is a valid origin-AS in the RPKI database.

This match clause is relevant for inbound route-maps only.

For iBGP learnt paths, the incoming BGP Prefix Origin Validation State Extended Community in the update will be compared against this route-map clause.

For eBGP learnt paths, the validation state obtained by ROA database lookup will be compared against this route-map clause.

While prefixes marked as validation-state invalid are rendered ineffective by not being considered for best-path computation in BGP, an administrator may decide to drop such prefixes altogether to save system memory. The following inbound route-map is recommended for this purpose:

```
route-map sample deny 10
match rpki invalid
route-map sample permit 20
```

## RPKI Show Commands

To display RPKI configuration information, perform one of the following tasks:

Command	Purpose
<code>show bgp rpki summary</code>	Displays an overview of RPKI statistics including the number of RPKI caches.

Command	Purpose
<code>show bgp rpki table {ipv4   ipv6} {IP address/masklength}</code>	<p>Displays information about the current RPKI ROA database. With no options specified, the command shows the IPv4 ROA database. With the IPv6 option (show bgp rpki table ipv6), the command shows the IPv6 ROA database. ROAs that are received from a cache that is temporarily down (due to connectivity issues, for example) are displayed with (*). These ROAs will be removed from the RPKI database if the cache session does not establish within the purge-time for that cache.</p> <p>If an ROA prefix-block is specified after the table show command (for example, show bgp rpki table 67.21.36.0/24 max 24), then that specific ROA entry is displayed in detail, if the ROA exists.</p> <p><b>Note</b> One ROA (IP address/min-max) can have multiple origin ASs and can be sourced from multiple caches.</p>
<code>show bgp rpki cache {IP address}</code>	<p>Displays a summary listing of all the caches that are configured and their parameters, such as <b>show bgp summary</b>.</p> <p>If a cache IP address is specified with the previous command, then detailed information is shown for that cache.</p>
<code>show bgp {ipv4 unicast   ipv6 unicast} origin-as validity-state {valid   invalid   unknown}</code>	<p>Displays information about BGP. This command has new options to filter the BGP table output based on path (validation_state). Specify a validity state (valid, invalid, or unknown) with this command to filter the relevant information from the BGP table, and only the BGP paths matching that validity-state are displayed.</p>

## RPKI Clear Commands

The following is the RPKI Clear command:

- **clear bgp rpki cache \*** - This command resets the transport sessions of all configured RPKI caches and immediately purges the RPKI database of all IPv4 and IPv6 ROAs received from all caches.

## RPKI Debug and Event History Commands

The following are the RPKI Debug and Event History commands:

- **debug bgp rpki** - This command turns on debugging for all RPKI related operations excluding prefix-validation. This includes debugging events such as RPKI cache connectivity, protocol state-machine for the RPKI caches, and RPKI database events such as ROA insertion or deletion.
- **sh bgp event-history rpki** - This command dumps high level information about RPKI.