



Certificate Signing Request (CSR) Generation for a Third-Party Certificate on Cisco Prime Infrastructure

This appendix describes how to generate a Certificate Signing Request (CSR) to obtain a third-party certificate with Cisco Prime Infrastructure and how to import the certificate into Prime Infrastructure. It contains the following sections:

- [Prerequisites, page C-15](#)
- [Components Used, page C-15](#)
- [Certificate Signing Request \(CSR\), page C-16](#)
- [Generating a Certificate, page C-16](#)
- [Importing a Certificate, page C-17](#)
- [Importing a Certificate and a Key, page C-17](#)
- [Importing Signed Certificates, page C-18](#)
- [Viewing the list of Certificates, page C-18](#)
- [Deleting Certificates, page C-19](#)
- [Related Publications, page C-19](#)
- [Troubleshooting, page C-19](#)

Prerequisites

Ensure that you meet these requirements before you attempt this configuration:

- Knowledge of how to install and configure Prime Infrastructure for basic operation
- Knowledge of self-signed and digital certificates, and other security mechanisms related to Public Key Infrastructure (PKI)

Components Used

The information in this document is based on these software and hardware versions:

- Prime Infrastructure Release 1.1.0.58

For more information about the supported hardware, see the Prime Infrastructure release notes at the following URL:

http://www.cisco.com/en/US/docs/wireless/ncs/1.1/release/notes/NCS_RN1.1.html

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, make sure that you understand the potential impact of any command.

Certificate Signing Request (CSR)

A certificate is an electronic document that you use to identify a server, a company, or some other entity and to associate that identity with a public key.

A self-signed certificate is an identity certificate that is signed by its own creator. That is, the person who created the certificate also signed off on its legitimacy.

Certificates can be self-signed or can be attested by a digital signature from a certificate authority (CA).

CAs are entities that validate identities and issue certificates. The certificate issued by the CA binds a particular public key to the name of the entity that the certificate identifies, such as the name of a server or device. Only the public key that the certificate certifies works with the corresponding private key possessed by the entity that the certificate identifies. Certificates help prevent the use of fake public keys for impersonation.

A CSR is a message that an applicant sends to a CA to apply for a digital identity certificate. Before a CSR is created, the applicant first generates a key pair, which keeps the private key secret. The CSR contains information that identifies the applicant, such as a directory name in the case of an X.509 certificate, and the public key chosen by the applicant. The corresponding private key is not included in the CSR, but is used to digitally sign the entire request.

The CSR can be accompanied by other credentials or proofs of identity required by the certificate authority, and the certificate authority can contact the applicant for further information. For the most part, a third-party CA company, such as Entrust or VeriSign, requires a CSR before the company can create a digital certificate.

CSR generation is independent of the device on which you plan to install an external certificate. Therefore, a CSR and a private key file can be generated on any individual machine which supports CSR generation. CSR generation is not switch-dependent or appliance-dependent in this case.

This appendix describes how to generate a CSR for a third-party certificate using the Cisco Prime Infrastructure.

Generating a Certificate

To generate a certificate, enter the following command:

```
ncs key genkey -newdn -csr csrfilename repository repositoryname
```

-newdn	Generates a new RSA key and self-signed certificate with domain information.
-csr	Generates new CSR certificate file.
repository	Repository command.

<i>csrfilename</i>	CSR filename.
<i>repositoryname</i>	Location where the files should be backed up to. Up to 80 alphanumeric characters.

This generates a new key/self-signed certificate pair, and output the CSR to the specified file. The `newdn` flag causes it to prompt for the distinguished name fields for the certificate. It is important to specify the final hostname that will be used to access Prime Infrastructure in the CN field of the DN to avoid browser warnings.

This example shows how to generate new rsa key and certificate files in the Prime Infrastructure server:

```
admin# ncs key genkey -newdn -csr csrfile.cert repository ncs-sftp-repo
Prime Infrastructure server is running
Changes will take affect on the next server restart
Enter the domain name of the server: <server name>
Enter the name of your organizational unit: <organizational unit>
Enter the name of your organization: <organization>
Enter the name of your city or locality: <city>
Enter the name of your state or province: <state>
Enter the two letter code for your country: <country code>
Generating RSA key
Writing certificate signing request to /opt/CSConcs/migrate/restore/test
INFO: no staging url defined, using local space.      rval:2
```

Importing a Certificate

To import a CA certificate to a trust store in Prime Infrastructure, use the Prime Infrastructure **key importcert** command.

```
ncs key importcert aliasname ca-cert-filename repository repositoryname
```

<i>aliasname</i>	A short name given for this CA certificate.
<i>ca-cert-filename</i>	CA certificate filename.
<i>repositoryname</i>	The repository name configured in Prime Infrastructure where the ca-cert-filename is hosted.

This example shows how to apply the CA certificate file to a trust store in Prime Infrastructure server:

```
admin# ncs key importcert alias1 cacertfile repository ncs-sftp-repo
```



Note

After entering this command, enter the **ncs stop** and **ncs start** commands to restart the Prime Infrastructure server to make the changes take effect.

Importing a Certificate and a Key

To import an RSA key and signed certificate to Prime Infrastructure, use Prime Infrastructure **key importkey** command.

```
ncs key importkey key-filename cert-filename repository repositoryname
```

<i>key-filename</i>	RSA private key file name.
<i>cert-filename</i>	Certificate file name.
<i>repositoryname</i>	The repository name configured in Prime Infrastructure where the key-file and cert-file is hosted.

This example shows how to apply the new RSA key and certificate files to Prime Infrastructure server.

```
admin# ncs key importkey keyfile certfile repository ncs-sftp-repo
```

**Note**

After applying this command, enter **ncs stop** and **ncs start** command to restart Prime Infrastructure server to make changes into effect.

Importing Signed Certificates

To apply an RSA key and signed certificate to Prime Infrastructure, use Prime Infrastructure **key importsignedcert** command.

```
ncs key importsignedcert signed-cert-filename repository repositoryname
```

This example shows how to apply signed certificate files to Prime Infrastructure server:

```
admin# ncs key importsingedcert signed-certfile repository ncs-sftp-repo
```

**Note**

After applying this command, enter **ncs stop** and **ncs start** command to restart Prime Infrastructure server to make changes into effect.

Viewing the list of Certificates

To list all the CA certificates that exist in Prime Infrastructure trust store, use Prime Infrastructure **key listcacerts** command.

```
ncs key listcacerts
```

This example shows how to list all the CA certificates exist in Prime Infrastructure trust store:

```
admin# ncs key listcacerts
```

```
Certificate utnuserfirsthardwareca from CN=UTN-USERFirst-Hardware,
OU=http://www.example.com, O=The USERTRUST Network, L=Salt Lake City, ST=UT, C=US
Certificate gtecybertrust5ca from CN=GTE CyberTrust Root 5, OU="GTE CyberTrust Solutions,
Inc.", O=GTE Corporation, C=US
Certificate equifaxsecureebusinessca1 from CN=Equifax Secure eBusiness CA-1, O=Equifax
Secure Inc., C=US
```

```
Certificate thawtepersonalfreemailca from EMAILADDRESS=email@example.com, CN=Thawte
Personal Freemail CA, OU=Certification Services Division, O=Thawte Consulting, L=Cape
Town, ST=Western Cape, C=ZA
Certificate addtrustclass1ca from CN=AddTrust Class 1 CA Root, OU=AddTrust TTP Network,
O=AddTrust AB, C=SE
Certificate aolrootca1 from CN=America Online Root Certification Authority 1, O=America
Online Inc., C=US
Certificate geotrustuniversalca from CN=GeoTrust Universal CA, O=GeoTrust Inc., C=US
Certificate digicertglobalrootca from CN=DigiCert Global Root CA, OU=www.example.com,
O=DigiCert Inc, C=US
Certificate certumtrustednetworkca from CN=Certum Trusted Network CA, OU=Certum
Certification Authority, O=Unizeto Technologies S.A., C=PL
Certificate swissignsilver2ca from CN=SwissSign Silver CA - G2, O=SwissSign AG, C=CH
```

Deleting Certificates

To delete CA certificates that exist in Prime Infrastructure trust store, use Prime Infrastructure key `deletecacert` command.

```
ncs key deletecacert aliasname
```

This example shows how to delete CA certificates exist in Prime Infrastructure trust store:

```
admin# ncs key deletecacert certumtrustednetworkca
Deleting certificate from trust store
```

Related Publications

For more information about Prime Infrastructure commands, see the following URL:

<http://www.cisco.com/en/US/docs/wireless/ncs/1.1/command/reference/cli11.html>

Troubleshooting

There is currently no specific troubleshooting information available for this configuration.

