Configure and Enroll a Cisco IOS Router to Another Cisco IOS Router Configured as a CA Server

Document ID: 50282

Contents

Introduction

This document describes how to configure a Cisco IOS® router as a Certificate Authority (CA) server. Additionally, it illustrates how to enroll another Cisco IOS router to obtain a root and ID certificate for IPsec authentication from the CA server.

Prerequisites

Requirements

There are no specific requirements for this document.

Components Used

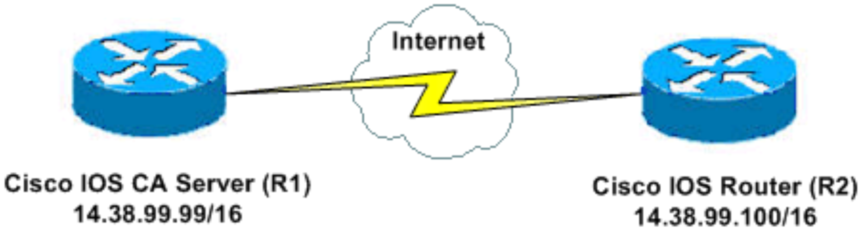
The information in this document is based on these software and hardware versions:

- Two Cisco 2600 Series Routers that run Cisco IOS Software Release 12.3(4)T3.

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.

Network Diagram

This document uses this network setup:



Conventions

Refer to the Cisco Technical Tips Conventions for more information on document conventions.

Generate and Export the RSA Key Pair for the Certificate Server

The first step is to generate the RSA key pair that the Cisco IOS CA server uses. On the router (R1), generate the RSA keys as this output shows:

```
R1(config)#crypto key generate rsa general-keys label cisco1 exportable
The name for the keys will be: cisco1
Choose the size of the key modulus in the range of 360 to 2048 for your
  General Purpose Keys. Choosing a key modulus greater than 512 may take
  a few minutes.

How many bits in the modulus [512]:
% Generating 512 bit RSA keys ...[OK]

R1(config)#
*Jan 22 09:51:46.116: %SSH-5-ENABLED: SSH 1.99 has been enabled
```

Note: You must use the same name for the key pair (*key−label*) that you plan to use for the certificate server (via the **crypto pki server** *cs−label* command covered later).

Export the Generated Key Pair

Export the keys to non−volatile RAM (NVRAM) or TFTP (based on your configuration). In this example, NVRAM is used. Based on your implementation, you might want to use a separate TFTP server in order to store your certificate information.

```
R1(config)#crypto key export rsa cisco1 pem url nvram: 3des cisco123

% Key name: cisco1
   Usage: General Purpose Key
Exporting public key...
Destination filename [cisco1.pub]?
Writing file to nvram:cisco1.pub
Exporting private key...
Destination filename [cisco1.prv]?
Writing file to nvram:cisco1.prv
R1(config)#
```

If you use a TFTP server, you can re−import the generated key pair as this command shows:

```
crypto key import rsa key−label pem [usage−keys] {terminal | url url} [exportable] passphra
```

Note: If you do not want the key to be exportable from your certificate server, import it back to the certificate server after it has been exported as a non−exportable key pair. This way, the key cannot be taken off again.

Verify the Generated Key Pair

Issue the **show crypto key mypubkey rsa** command in order to verify the generated key pair.

The Output Interpreter Tool (registered customers only) (OIT) supports certain **show** commands. Use the OIT to view an analysis of **show** command output.

```
R1#show crypto key mypubkey rsa
% Key pair was generated at: 09:51:45 UTC Jan 22 2004
Key name: cisco1
 Usage: General Purpose Key
 Key is exportable.
 Key Data:
  305C300D 06092A86 4886F70D 01010105 00034B00 30480241 00CC2DC8 ED26163A
  B3642376 FAA91C2F 93A3825B 3ABE6A55 C9DD3E83 F7B2BD56 126E0F11 50552843
  7F7CA4DA 3EC3E2CE 0F42BD6F 4C585385 3C43FF1E 04330AE3 37020301 0001
% Key pair was generated at: 09:51:54 UTC Jan 22 2004
Key name: cisco1.server
 Usage: Encryption Key
 Key is exportable.
 Key Data:
  307C300D 06092A86 4886F70D 01010105 00036B00 30680261 00EC5578 025D3066
  72149A35 32224BC4 3E41DD68 38B08D39 93A1AA43 B353F112 1E56DA42 49741698
  EBD02905 FE4EC392 7174EEBF D82B4475 2A2D7DEC 83E277F8 AEC590BE 124E00E1
  C1607433 5C7BC549 D532D18C DD0B7AE3 AECDDE9C 07AD84DD 89020301 0001
```

Enable the HTTP Server on the Router

The Cisco IOS CA Server only supports enrollments done via Simple Certificate Enrollment Protocol (SCEP). Consequently, in order to make this possible, the router must run the built−in Cisco IOS HTTP server. Use the **ip http server** command in order to enable it:

```
R1(config)#ip http server
```

Enable and Configure the CA Server on the Router

Complete these steps:

1. It is very important to remember that the certificate server must use the same name as the key pair you just manually generated.

   The label matches the generated key pair label:

   ```
   R1(config)#crypto pki server cisco1
   ```

   After you have enabled a certificate server, you can use the preconfigured default values or specify values via CLI for the functionality of the certificate server.
2. The **database url** command specifies the location where all database entries for the CA server are written out. If this command is not specified, all database entries are written to Flash.

```
R1(cs−server)#database url nvram:
```

Note: If you use a TFTP server, the URL needs to be **tftp://<ip_address>/directory**.

3. Configure the database level:

   ```
   R1(cs−server)#database level minimum
   ```

   This command controls what type of data is stored in the certificate enrollment database:

   - **Minimum** Enough information is stored only to continue issuing new certificates without conflict. The default value.
   - **Names** In addition to the information given in the minimal level, the serial number and subject name of each certificate.
   - **Complete** In addition to the information given in the minimal and names levels, each issued certificate is written to the database.

   Note: The **complete** keyword produces a large amount of information. If it is issued, you should also specify an external TFTP server in which to store the data via the **database url** command.
4. Configure the CA issuer name to the specified DN−string. On this example, the CN (Common Name) of cisco1.cisco.com, L (Locality) of RTP, and C (Country) of US are used:

   ```
   R1(cs−server)#issuer−name CN=cisco1.cisco.com L=RTP C=US
   ```
5. Specify the lifetime, in days, of a CA certificate or a certificate.

   Valid values range from *1 day to 1825 days*. The default CA certificate lifetime is three years and the default certificate lifetime is one year. The maximum certificate lifetime is *one month less* than the lifetime of the CA certificate. For example:

   ```
   R1(cs−server)#lifetime ca−certificate 365
   R1(cs−server)#lifetime certificate 200
   ```
6. Define the lifetime, in hours, of the CRL that is used by the certificate server. The maximum lifetime value is **336 hours** (two weeks). The default value is **168 hours** (one week).

   ```
   R1(cs−server)#lifetime crl 24
   ```
7. Define a Certificate−Revocation−List Distribution Point (CDP) to use in the certificates that are issued by the certificate server.

   The URL must be an HTTP URL. For example, our server had an IP address of 172.18.108.26:

   ```
   R1(cs−server)#cdp−url http://172.18.108.26/cisco1cdp.cisco1.crl
   ```
8. Issue the **no shutdown** command in order to enable the CA server:

   ```
   R1(cs−server)#no shutdown
   ```

   Note: Issue this command only after you have completely configured your certificate server.

Configure and Enroll the Second IOS Router (R2) to the Certificate Server

Follow this procedure.

1. Configure a hostname, a domain−name, and generate the RSA keys on R2.

   Use the **hostname** command in order to configure the hostname of the router to be R2:

   ```
   Router(config)#hostname R2
   ```

```
R2(config)#
```

Notice that the hostname of the router changed immediately after you entered the **hostname** command.

Use the **ip domain−name** command in order to configure the domain name on the router:

```
R2(config)#ip domain−name cisco.com
```

Use the **crypto key generate rsa** command in order to generate the R2 key pair:

```
R2(config)#crypto key generate rsa
The name for the keys will be: R2.cisco.com
Choose the size of the key modulus in the range of 360 to 2048 for your
General Purpose Keys. Choosing a key modulus greater than 512 may take
a few minutes.

How many bits in the modulus [512]:
% Generating 512 bit RSA keys ...[OK]
```

2. Use these commands in global configuration mode in order to declare to the CA that your router should use (Cisco IOS CA in this example) and specify characteristics for the trustpoint CA:

   ```
   crypto ca trustpoint cisco
    enrollment retry count 5
    enrollment retry period 3
    enrollment url http://14.38.99.99:80
    revocation−check none
   ```

   Note: The **crypto ca trustpoint** command unifies the existing **crypto ca identity** command and **crypto ca trusted−root** command, thereby providing combined functionality under a single command.
3. Use the **crypto ca authenticate cisco** command (cisco is the trustpoint label) in order to retrieve the root certificate from the CA server:

   ```
   R2(config)#crypto ca authenticate cisco
   ```
4. Use the **crypto ca enroll cisco** command (cisco is the trustpoint label) in order to enroll and generate:

   ```
   R2(config)#crypto ca enroll cisco
   ```

   After successfully enrolling to the Cisco IOS CA server, you should see the issued certificates by using the command **show crypto ca certificates**. This is the output of the command. The command displays the detailed certificate information, which correspond with the parameters configured in the Cisco IOS CA server:

   ```
   R2#show crypto ca certificates
   Certificate
     Status: Available
     Certificate Serial Number: 02
     Certificate Usage: General Purpose
     Issuer:
       cn=cisco1.cisco.com
       l=RTP
       c=US
     Subject:
       Name: R2.cisco.com
       hostname=R2.cisco.com
     CRL Distribution Point:
    http://172.18.108.26/cisco1cdp.cisco1.crl
   Validity Date:
       start date: 15:41:11 UTC Jan 21 2004
   ```

```
    end   date: 15:41:11 UTC Aug 8 2004
    renew date: 00:00:00 UTC Jan 1 1970
  Associated Trustpoints: cisco

CA Certificate
  Status: Available
  Certificate Serial Number: 01
  Certificate Usage: Signature
  Issuer:
    cn=cisco1.cisco.com
    l=RTP
    c=US
  Subject:
    cn=cisco1.cisco.com
    l=RTP
    c=US
  Validity Date:
    start date: 15:39:00 UTC Jan 21 2004
    end   date: 15:39:00 UTC Jan 20 2005
  Associated Trustpoints: cisco
```

5. Enter this command in order to save the key to persistent Flash memory:

```
hostname(config)#write memory
```

6. Enter this command in order to save the configuration:

```
hostname#copy run start
```

Verify

Use this section to confirm that your configuration works properly.

The Output Interpreter Tool (registered customers only) (OIT) supports certain **show** commands. Use the OIT to view an analysis of **show** command output.

- **show crypto ca certificates** Displays certificates.
- **show crypto key mypubkey rsa** Displays the key pair.

```
!% Key pair was generated at: 09:28:16 EST Jan 30 2004
  !Key name: ese−ios−ca
  ! Usage: General Purpose Key
  ! Key is exportable.
  ! Key Data:
  !  30819F30 0D06092A 864886F7 0D010101 05000381 8D003081 89028181 00AF2198
  !  C56F1A8F 5AC501FF ADFB1489 1F503F91 CA3C3FA3 9FB2C150 FFCBF815 2AA73060
  !  E79AF510 E292C171 C6804B45 0CAAD4AF 5834AB85 B204208B 3960D20D 9B51AF7B
  !  ACF12D3D F5BC6EAE 77186AE9 1471F5A4 443CE5B5 1336EC33 5FEB3398 002C15EE
  !  9F8FD331 83490D8A 983FBBE1 9E72A130 121A3B97 A3ACD147 C37DA3D6 77020301 0001
  !% Key pair was generated at: 09:28:17 EST Jan 30 2004
  !Key name: ese−ios−ca.server
  ! Usage: Encryption Key
  ! Key is exportable.
  ! Key Data:
  !  307C300D 06092A86 4886F70D 01010105 00036B00 30680261 0096456A 01AEC6A5
  !  0049CCA7 B41B675E 5317328D DF879CAE DB96A739 26F2A03E 09638A7A 99DFF8E9
  !  18F7635D 6FB6EE27 EF93B3DE 336C148A 6A7A91CB 6A5F7E1B E0084174 2C22B3E2
  !  3ABF260F 5C4498ED 20E76948 9BC2A360 1C799F8C 1B518DD8 D9020301 0001
```

- **crypto pki server ese−ios−ca info crl** Displays the certificate revocation list (CRL).

```
! Certificate Revocation List:
  !    Issuer: cn=ese−ios−ca,ou=ESE,o=Cisco Systems Inc,l=Raleigh,st=NC
  !    This Update: 09:58:27 EST Jan 30 2004
  !    Next Update: 09:58:27 EST Jan 31 2004
```

```
            !    Number of CRL entries: 0
            !    CRL size: 300 bytes
```

- **crypto pki server ese−ios−ca info requests** Displays pending enrollment requests.

```
          ! Enrollment Request Database:
            ! ReqID  State      Fingerprint                            SubjectName
            ! --------------------------------------------------------------
```

- **show crypto pki server** Displays the current public key infrastructure (PKI) server state.

```
          ! Certificate Server status: enabled, configured
            !    Granting mode is: manual
            !    Last certificate issued serial number: 0x1
            !    CA certificate expiration timer: 10:58:20 EDT Jun 21 2005
            !    CRL NextUpdate timer: 09:58:26 EST Jan 31 2004
            !    Current storage dir: nvram:
            !    Database Level: Names - subject name data written as .cnm
```

- **crypto pki server** *cs−label* **grant { all |** *tranaction−id* **}** Grants all or specific SCEP requests.
- **crypto pki server** *cs−label* **reject { all |** *tranaction−id* **}** Rejects all or specific SCEP requests.
- **crypto pki server cs−label password generate [** *minutes* **]** Generates a one−time password (OTP) for an SCEP request (minutes − length of time (in minutes) that the password is valid. The valid range is from 1 to 1440 minutes. The default is 60 minutes.

  Note: Only one OTP is valid at a time. If a second OTP is generated, the previous OTP is no longer valid.
- **crypto pki server** *cs−label* **revoke** *certificate−serial−number* Revokes a certificate based on its serial number.
- **crypto pki server** *cs−label* **request** *pkcs10* **{url** *url* **|** *terminal***} [pem]** Manually adds either the base64 or PEM PKCS10 certificate enrollment request to the request database.
- **crypto pki server** *cs−label* **info crl** Displays information regarding the status of the current CRL.
- **crypto pki server** *cs−label* **info request** Displays all outstanding certificate enrollment requests.

See the Verify the Generated Key Pair section of this document for additional verification information.

Troubleshoot

Refer to IP Security Troubleshooting − Understanding and Using debug Commands for troubleshooting information.

Note: In many situations, you can solve the problems when you delete and re−define the CA server.

Related Information

- **IPsec Negotiation/IKE Protocols**
- **Technical Support & Documentation − Cisco Systems**

Updated: Jan 14, 2008 Document ID: 50282