

Configure ASA: SSL Digital Certificate Installation and Renewal

Contents

[Introduction](#)

[Background Information](#)

[Prerequisites](#)

[Requirements](#)

[Components Used](#)

[Configure](#)

[CSR Generation](#)

[1. Configure with the ASDM](#)

[2. Configure with the ASACLI](#)

[3. Use OpenSSL to Generate the CSR](#)

[SSL Certificate Generation on the CA](#)

[Example of SSL Certificate Generation on GoDaddy CA](#)

[SSL Certificate Installation on the ASA](#)

[1.1 Installation of the Identity Certificate in PEM Format with ASDM](#)

[1.2. Installation of a PEM Certificate with the CLI](#)

[2.1 Installation of a PKCS12 Certificate with ASDM](#)

[2.2 Installation of a PKCS12 Certificate with the CLI](#)

[Verify](#)

[View Installed Certificates via ASDM](#)

[View Installed Certificates via the CLI](#)

[Verify Installed Certificate for WebVPN with a Web Browser](#)

[Renew SSL Certificate on the ASA](#)

[Frequently Asked Questions](#)

[1. What is the best way to transfer identity certificates out of one ASA onto a different ASA?](#)

[2. How to generate SSL certificates for use with VPN Load Balancing ASAs?](#)

[3. Do the certificates need to be copied from the Primary ASA to the Secondary ASA in an ASA failover pair?](#)

[4. If ECDSA keys are used, is the SSL certificate generation process different?](#)

[Troubleshoot](#)

[Troubleshoot Commands](#)

[Common Issues](#)

[Appendix](#)

[Appendix A: ECDSA or RSA](#)

[Appendix B: Use OpenSSL to Generate a PKCS12 Certificate from an Identity Certificate, CA Certificate, and Private Key](#)

[Related Information](#)

Introduction

This document describes installation of third-party trusted SSL digital certificate on the ASA for Clientless SSLVPN and AnyConnect connections.

Background Information

A GoDaddy Certificate is used in this example. Each step contains the Adaptive Security Device Manager (ASDM) procedure and the CLI equivalent.

Prerequisites

Requirements

This document requires access to a trusted third-party Certificate Authority (CA) for certificate enrollment. Examples of third-party CA vendors include, but are not limited to, Baltimore, Cisco, Entrust, Geotrust, G, Microsoft, RSA, Thawte, and VeriSign.

Before you start, verify that the ASA has the correct clock time, date, and time zone. With certificate authentication, it is recommended to use a Network Time Protocol (NTP) server to synchronize the time on the ASA. The [Cisco ASA Series General Operations CLI Configuration Guide, 9.1](#) details the steps to take in order to set up the time and date correctly on the ASA.

Components Used

This document uses an ASA 5500-X that runs software version 9.4.1 and ASDM version 7.4(1).

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, ensure that you understand the potential impact of any command.

Configure

The SSL protocol mandates that the SSL Server provide the client with a server certificate for the client to perform server authentication. Cisco does not recommend use of a self-signed certificate because of the possibility that a user could inadvertently configure a browser to trust a certificate from a rogue server. There is also the inconvenience to users to have to respond to a security warning when it connects to the secure gateway. It is recommended to use trusted third-party CAs to issue SSL certificates to the ASA for this purpose.

The lifecycle of a third-party certificate on the ASA essentially takes place with these steps:



CSR Generation

CSR generation is the first step in the lifecycle of any X.509 digital certificate.

Once the private/public Rivest-Shamir-Adleman (RSA) or Elliptic Curve Digital Signature Algorithm (ECDSA) keypair is generated ([Appendix A](#) details the difference between the use of RSA or ECDSA), a Certificate Signing Request (CSR) is created.

A CSR is a PKCS10 formatted message that contains the public key and identity information of the host which sends the request. [PKI Data Formats](#) explains the different certificate formats applicable to the ASA and Cisco IOS®.

Notes:

1. Check with the CA on the required keypair size. The CA/Browser Forum has mandated that all certificates generated by their member CAs have a minimum size of 2048 bits.
2. ASA currently does not support 4096 bit keys (Cisco bug ID [CSCut53512](#)) for SSL server authentication. However, IKEv2 does support the use of 4096 bit server certificates on the ASA 5580, 5585, and 5500-X platforms alone.
3. Use the DNS Name of the ASA in the FQDN field of the CSR in order to prevent Untrusted Certificate warnings and pass Strict Certificate check.

There are three methods to generate CSR.

- Configure with ASDM
- Configure with the ASA CLI

- Use OpenSSL to Generate the CSR

1. Configure with the ASDM

1. Navigate to **Configuration > Remote Access VPN > Certificate Management**, and choose **Identity Certificates**.
2. Click **Add**.

The screenshot shows the 'Add Identity Certificate' dialog box with the following configuration:

- Trustpoint Name:
- Import the identity certificate from a file (PKCS12 format with Certificate(s) +Private Key):
 - Decryption Passphrase:
 - File to Import From:
- Add a new identity certificate:
 - Key Pair:
 - Certificate Subject DN:
 - Generate self-signed certificate
 - Act as local certificate authority and issue dynamic certificates to TLS-Proxy
 - Enable CA flag in basic constraints extension

Buttons:

3. Define a trustpoint name in the Trustpoint Name input field.
4. Click the **Add a new identity certificate** radio button.
5. For the Key Pair, click **New**.

The screenshot shows a dialog box titled "Add Key Pair". It has a standard Windows-style title bar with a close button (X) in the top right corner. The dialog contains the following fields and options:

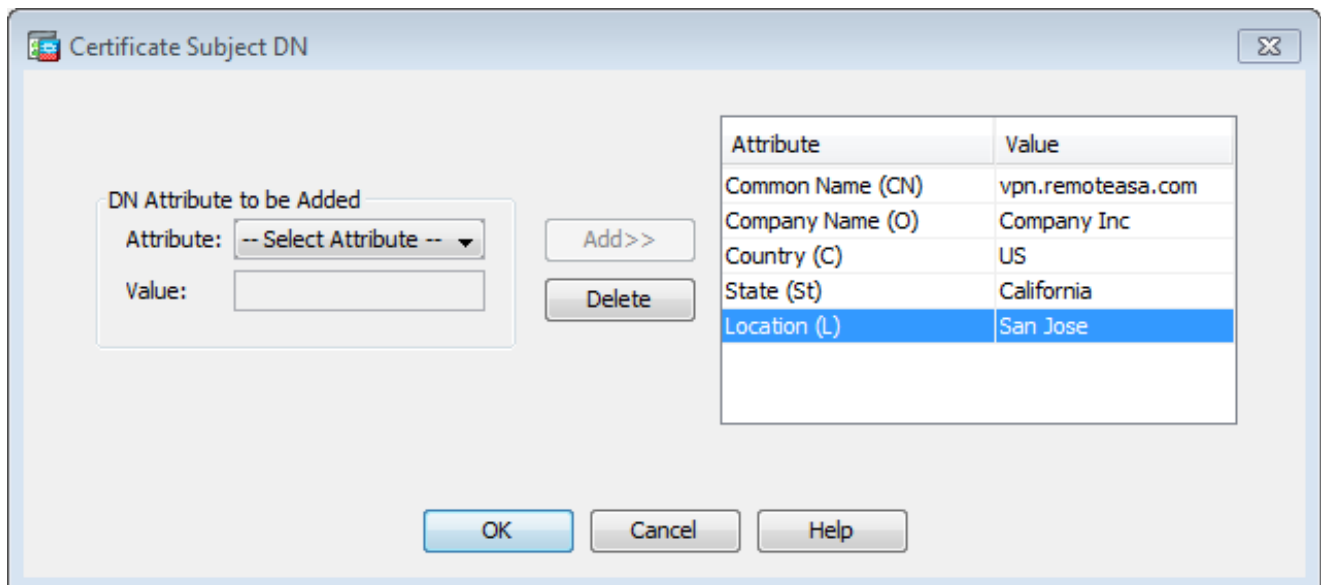
- Key Type:** Two radio buttons are present: "RSA" (which is selected) and "ECDSA".
- Name:** Two radio buttons are present: "Use default key pair name" and "Enter new key pair name:" (which is selected). To the right of the second radio button is a text input field containing the text "SSL-Keypair".
- Size:** A dropdown menu is shown with "2048" selected.
- Usage:** Two radio buttons are present: "General purpose" (which is selected) and "Special".


At the bottom of the dialog, there are three buttons: "Generate Now" (highlighted in blue), "Cancel", and "Help".

6. Choose the Key Type - RSA or ECDSA. (Refer to [Appendix A](#) to understand the differences.)
7. Click the **Enter new key pair name** radio button. Identify the key pair name for recognition purposes.
8. Choose the **Key Size**. Choose **General Purpose** for Usage with RSA.
9. Click **Generate Now**. The key pair are created.
10. To define the Certificate Subject DN, click **Select**, and configure the attributes listed in this table:

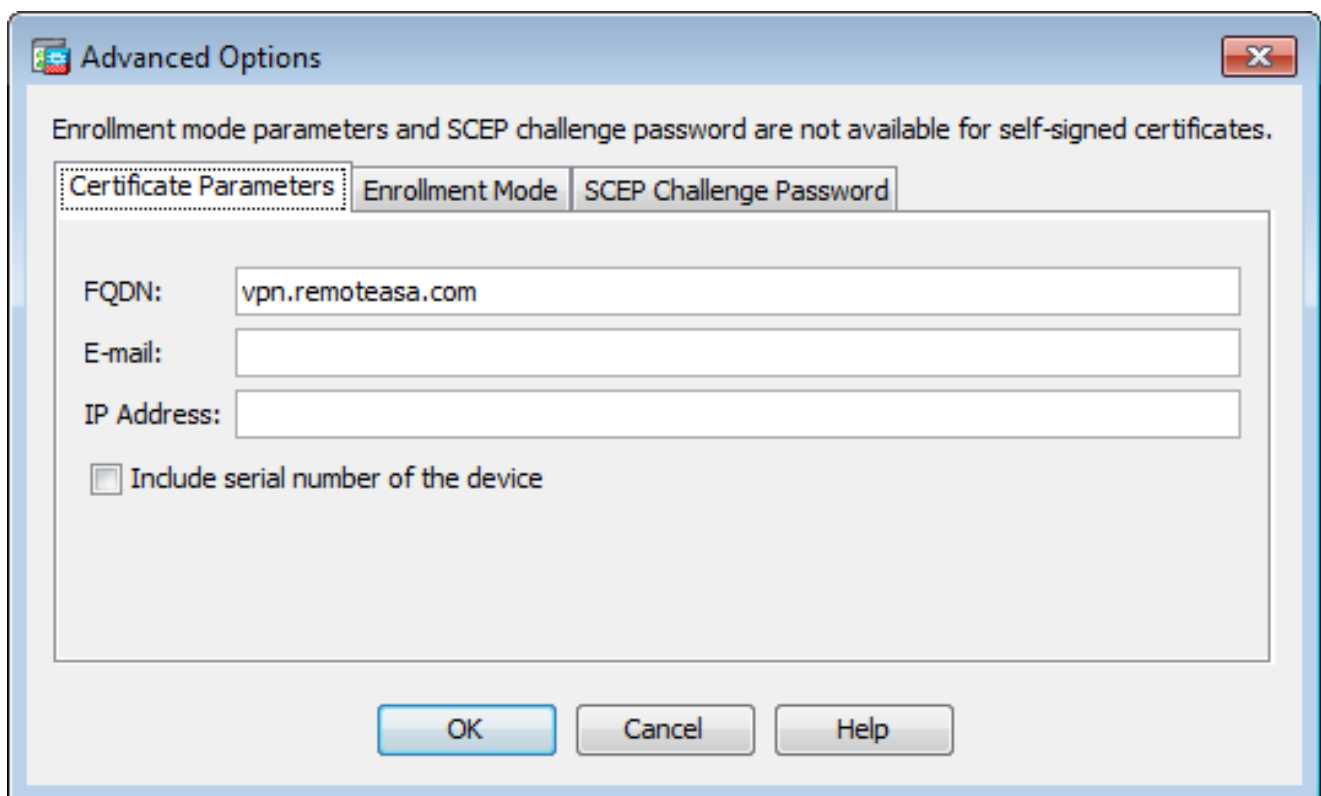
Attribute	Description
CN	FQDN (Full Qualified Domain Name) that will be used for connections to your firewall. For example, webvpn.cisco.com
OU	Department Name
O	Company Name (Avoid using Special Characters)
C	Country Code (2 Letter Code without Punctuation)
St	State (Must be spelled out completely. For example, North Carolina)
L	City
EA	Email Address

To configure these values, choose a value from the **Attribute** drop-down list, enter the value, and click **Add**.



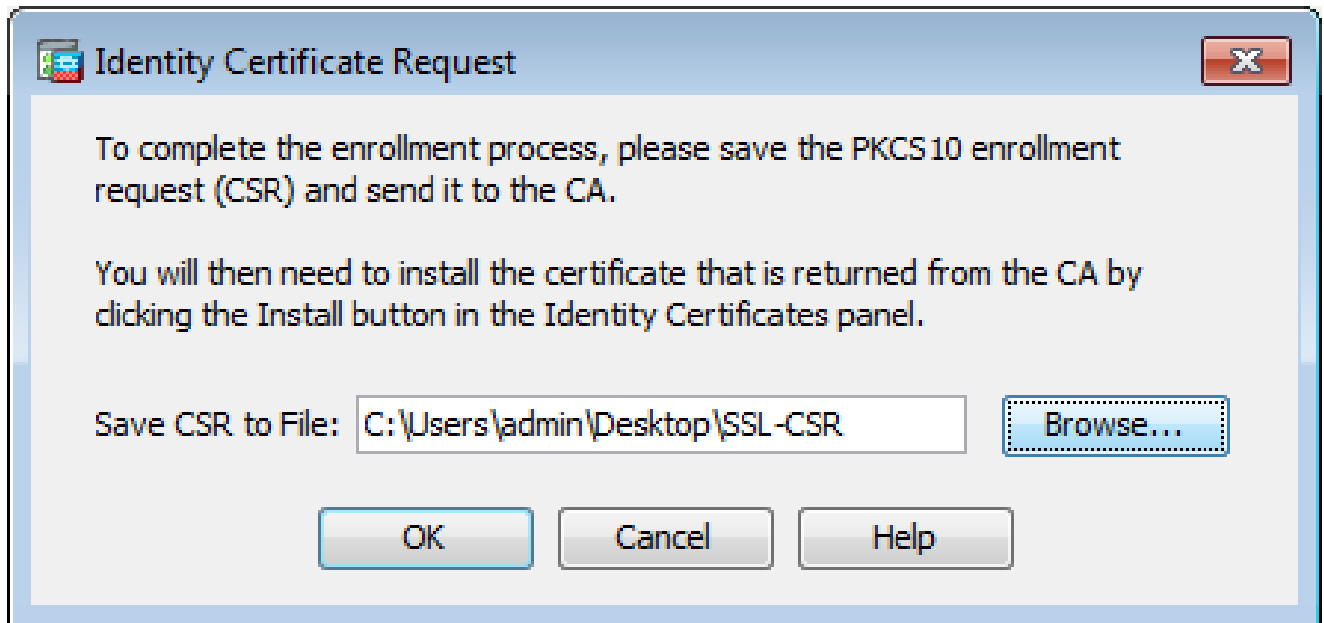
 **Note:** Some third-party vendors require particular attributes to be included before an identity certificate is issued. If unsure of the required attributes, check with the vendor for details.

11. After the appropriate values are added, click **OK**. The Add Identity Certificate dialog box appears with the CertificateSubject DN field populated.
12. Click **Advanced**.




13. In the **FQDN** field, enter the FQDN that is used to access the device from the Internet. Click **OK**.
14. Leave the Enable CA flag in basic constraints extension option checked. Certificates without the CA flag now cannot be installed on the ASA as CA certificates by default. The basic constraints extension identifies whether the subject of the certificate is a CA and the maximum depth of valid certification paths that include this certificate. Un-check the option to bypass this requirement.
15. Click **OK**, and then click **Add Certificate**. A prompt displays in order to save the CSR to a file on the local

machine.



16. Click **Browse**, choose a location in which to save the CSR, and save the file with the .txt extension.

 **Note:** When the file is saved with a .txt extension, the PKCS#10 request can be opened and viewed with a text editor (such as Notepad).

2. Configure with the ASA CLI

In the ASDM, the trustpoint is automatically created when a CSR is generated or when the CA certificate is installed. In the CLI, the trustpoint must be created manually.

```
<#root>
```

```
! Generates 2048 bit RSA key pair with label SSL-Keypair.
```

```
MainASA(config)#
```

```
crypto key generate rsa label SSL-Keypair modulus 2048
```

```
INFO: The name for the keys are: SSL-Keypair  
Keypair generation process begin. Please wait...
```

```
! Define trustpoint with attributes to be used on the SSL certificate
```

```
MainASA(config)#
```

```
crypto ca trustpoint SSL-Trustpoint
```

```
MainASA(config-ca-trustpoint)#
```

```
enrollment terminal
```

```
MainASA(config-ca-trustpoint)#
```

```
fqdn (remoteasavpn.url)
```

```
MainASA(config-ca-trustpoint)#
```



```
YkHXnFne1LQd41BgoL1Cr9+hx74XsTHGBmI1s/9T5oAX26Ym+B21/i/DP5BktIUA
8GvIY1/ypj9K049fP5ap8a10qvLtYYcCcfwrCt+0oj0rZ1YyJb3dFuMNRdAX37t
DuHN12EYNpYkjVk1wI53/5w3
-----END CERTIFICATE REQUEST-----
```


Redisplay enrollment request? [yes/no]:

no

! Displays the PKCS#10 enrollment request to the terminal. Copy this from the terminal to a text file to submit to the third party CA.

3. Use OpenSSL to Generate the CSR

OpenSSL makes use of the **OpenSSL config** file to pull the attributes to be used in the CSR generation. This process results in the generation of a CSR and a Private Key.

 **Caution:** Verify that the **Private key** that is generated is not shared with anyone else as it compromises the integrity of the certificate.

1. Ensure that OpenSSL is installed on the system that this process is run on. For Mac OSX and GNU/Linux users, this is installed by default.
2. Switch to a functional directory.

On Windows: By default, the utilities are installed in `C:\OpenSSL\bin`. Open a command prompt in this location.

On Mac OSX/Linux: Open the Terminal window in the directory needed to create the CSR.

3. Create an OpenSSL config file with a text editor with the attributes given . Once done, save the file as **openssl.cnf** in the location mentioned in the previous step (If you version 0.9.8h and later, the file is **sopenssl.cfg**)
<#root>

```
[req]
```

```
default_bits = 2048
default_keyfile = privatekey.key
distinguished_name = req_distinguished_name
req_extensions = req_ext
```

```
[req_distinguished_name]
```

```
commonName = Common Name (eg, YOUR name)
commonName_default = (asa.remotevpn.url)
```

```
countryName = Country Name (2 letter code)
countryName_default = US
```

```
stateOrProvinceName = State or Province Name (full name)
stateOrProvinceName_default = California
```

```
localityName = Locality Name (eg, city)
localityName_default = San Jose
```

```
0.organizationName = Organization Name (eg, company)
0.organizationName_default = Company Inc
```

```
[req_ext]
```

```
subjectAltName = @alt_names
```

```
[alt_names]
```

```
DNS.1 = *.remotearsa.com
```

4. Generate the CSR and Private Key with this command:

```
openssl req -new -nodes -out CSR.csr -config openssl.cnf
```

```
<#root>
```

```
# Sample CSR Generation:
```

```
openssl req -new -nodes -out CSR.csr -config openssl.cnf
```

```
Generate a 2048 bit RSA private key
```

```
.....+++
.....+++
writing new private key to 'privatekey.key'
-----
```

```
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
```

```
Common Name (eg, YOUR name) [(asa.remotevpn.url)]:
Country Name (2 letter code) [US]:
State or Province Name (full name) [California]:
Locality Name (eg, city) [San Jose]:
Organization Name (eg, company) [Company Inc]:
```

Submit the saved CSR to the third-party CA vendor. Once the certificate is issued, the CA provides the identity certificate and the CA certificate to be installed on the ASA.

SSL Certificate Generation on the CA

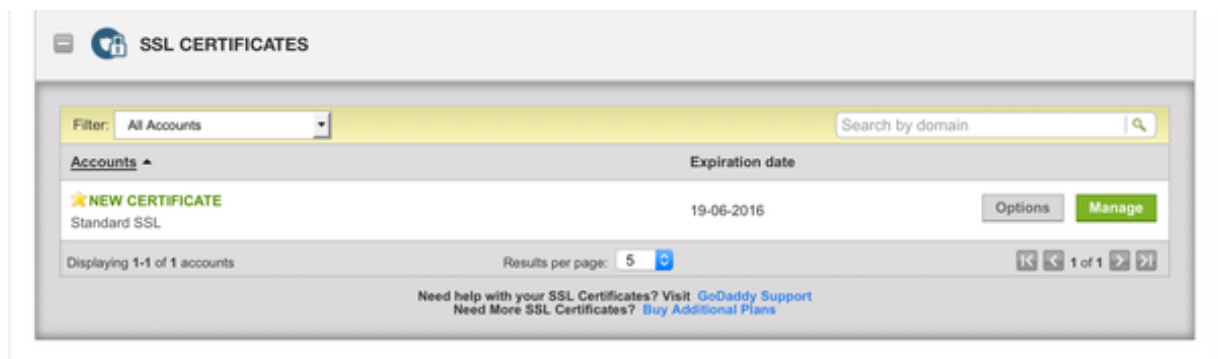
The next step is to get the CSR signed from the CA. The CA provides either a newly generated PEM encoded Identity Certificate or with a PKCS12 certificate along with the CA certificate bundle.

If the CSR is generated outside the ASA (either via OpenSSL or on the CA itself), the PEM encoded Identity Certificate with the Private Key and CA certificate are available as separate files. [Appendix B](#) provides the steps to bundle these elements together into a single PKCS12 file (.p12 or .pfx format).

In this document, the GoDaddy CA is used as an example to issue identity certificates to the ASA. This process differs in other CA vendors. Read through the CA documentation carefully before proceeding.

Example of SSL Certificate Generation on GoDaddy CA

After purchase and the initial setup phase of the SSL certificate, navigate to the GoDaddy Account and view the SSL Certificates. There must be a new certificate. Click **Manage** to proceed.



This then brings up a page to provide the CSR as seen in this image.

Based on the CSR entered, the CA determines the Domain Name to which the certificate is to be issued.

Verify that this matches the FQDN of the ASA.

Choose website

Select a domain hosted with us

Provide a certificate signing request (CSR)

Certificate Signing Request (CSR) [Learn more](#)

```
/ypj9KO49fP5ap8al0qvLtYYcCcfwrCt+OojOrZ1YyJb3dFuMNRRedAX37t
DuHNI2EYNpYkjVklwI53/5w3
-----END CERTIFICATE REQUEST-----
```

Domain Name (based on CSR):

vpn.remoteasa.com

Domain ownership

We'll send an email with a unique code to your address on file. Follow its instructions to verify you have website or DNS control over the selected domain. [More info](#)

AND

We can send domain ownership instructional emails to one or both of the following:


- Contacts listed in the domain's public WHOIS database record
- Email addresses: admin@[domain], administrator@[domain], hostmaster@[domain], postmaster@[domain], and webmaster@[domain]

[Hide advanced options](#)

Signature Algorithm [Learn more](#)

GoDaddy SHA-2

I agree to the terms and conditions of the [Subscriber Agreement](#).

 **Note:** GoDaddy and most other CAs use SHA-2 or SHA256 as the default Certificate Signature Algorithm. ASA supports the SHA-2 signature algorithm which starts from **8.2(5)** [pre-8.3 releases] and **8.4(1)** [post-8.3 releases] onwards (Cisco bug ID [CSCti30937](#)). Choose SHA-1 signature algorithm if a version older than 8.2(5) or 8.4(1) is used.

Once the request is submitted, GoDaddy verifies the request before it issues the certificate.

After the certificate request is validated, GoDaddy issues the certificate to the account.

The certificate can be then downloaded for installation on the ASA. Click **Download** on the page in order to proceed further.

The screenshot shows the GoDaddy SSL Certificate Management interface. At the top, there is a navigation bar with 'Certificates', 'Repository', 'Help', and 'Report EV Abuse'. The main heading is 'All > vpn.remoteasa.com' with 'Standard SSL Certificate' below it. Under 'Certificate Management Options', there are three buttons: 'Download', 'Revoke', and 'Manage'. To the right, there is a section for 'Display your SSL Certificate security seal' with options for 'Color' (set to 'Light') and 'Language' (set to 'English'). Below these is a 'Preview' of the seal and a 'Code' block containing a JavaScript snippet for embedding the seal. The 'Certificate Details' table is as follows:

Certificate Details	
Status	Certificate issued
Domain name	vpn.remoteasa.com
Encryption Strength	GoDaddy SHA-2
Validity Period	7/22/2015 - 7/22/2016
Serial Number	25.cd:73:a9:84:07:06:05

Choose **Other** as the Server Type and download the certificate zip bundle.

The screenshot shows the 'Download Certificate' page for 'vpn.remoteasa.com'. The page title is 'Standard SSL Certificate'. Below the heading, there is a paragraph explaining that users should download a Zip file matching their hosting server type. A link is provided for 'View Installation Instructions for the selected server.' The 'Server type' dropdown menu is open, showing options: 'Select ...', 'Apache', 'Exchange', 'IIS', 'Mac OS X', 'Tomcat', and 'Other' (which is highlighted in blue). There are 'File' and 'Cancel' buttons next to the dropdown.

The .zip file contains the identity certificate and GoDaddy CA certificate chain bundles as two separate .crt files. Proceed to SSL certificate installation to install these certificates on the ASA.

SSL Certificate Installation on the ASA

The SSL certificate can be installed on the ASA with either ASDM or CLI in two ways:

1. Import the CA and identity certificate separately in PEM formats.
2. Or import the PKCS12 file (base64 encoded for CLI) wherein Identity certificate, CA certificate, and private key are bundled in the PKCS12 file.

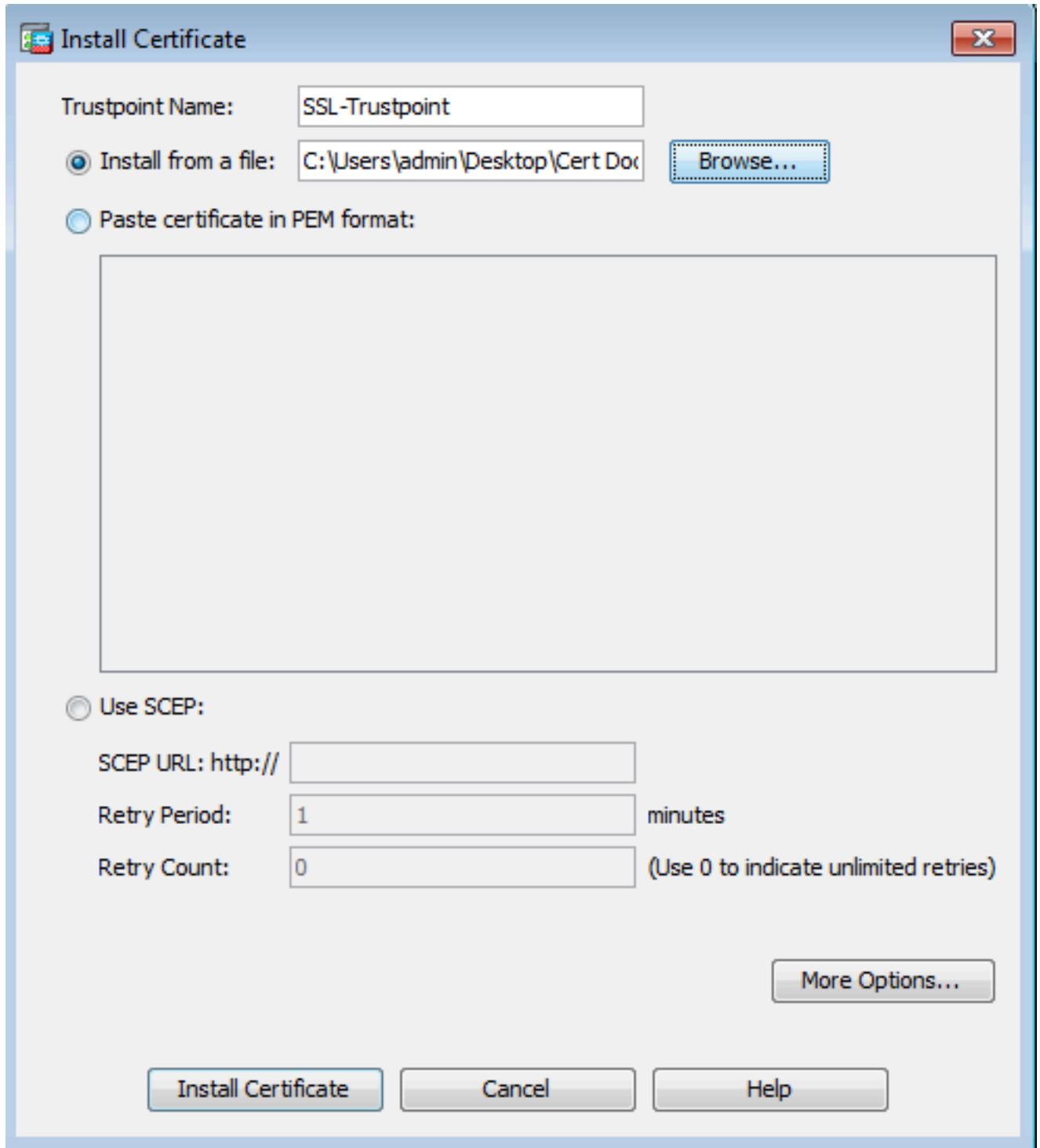


Note: If the CA provides a CA certificate chain, only install the immediate intermediate CA certificate in the hierarchy on the trustpoint used to generate the CSR. The Root CA certificate and any other intermediate CA certificates can be installed in new trustpoints.

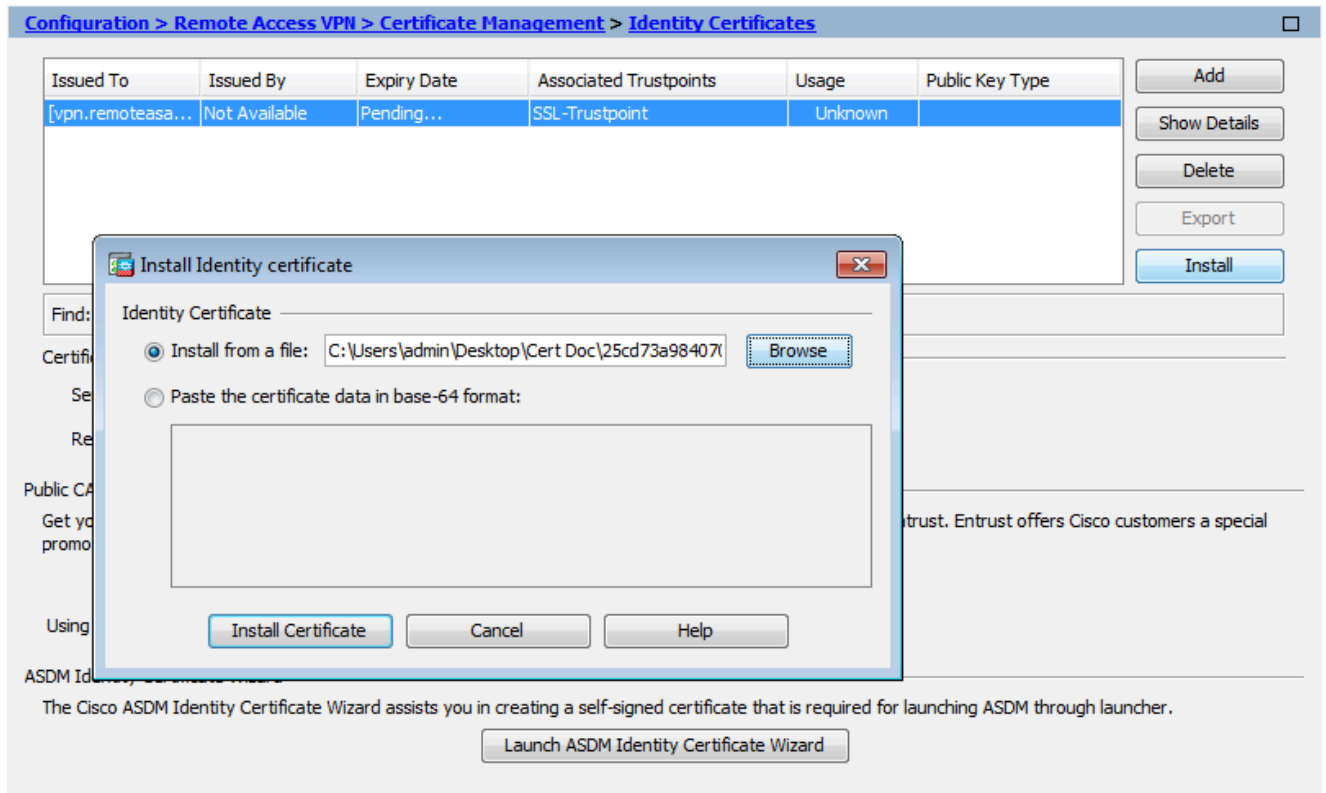
1.1 Installation of the Identity Certificate in PEM Format with ASDM

The installation steps given assume that the CA provides a PEM encoded (.pem, .cer, .crt) identity certificate and CA certificate bundle.

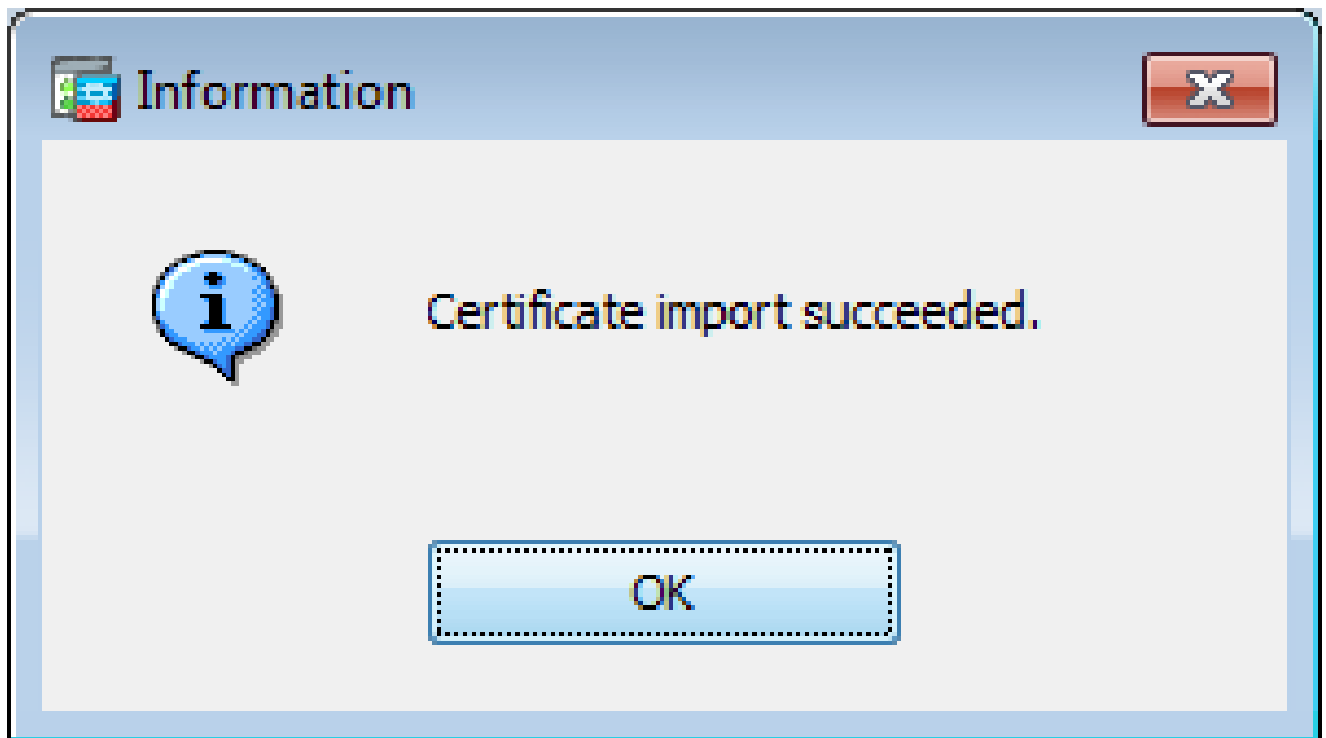
1. Navigate to **Configuration > Remote Access VPN > Certificate Management**, and choose CA Certificates.
2. The PEM encoded certificate in a text editor and copy and paste the base64 CA certificate provided by the third-party vendor into the text field.



3. Click **Install certificate**.
4. Navigate to **Configuration > Remote Access VPN > Certificate Management**, and choose Identity Certificates.
5. Select the Identity Certificate created previously. Click **Install**.
6. Either click the option **Install from a file** radio button and choose the PEM encoded Identity certificate or, open the PEM encoded certificate in a text editor and copy and paste the base64 Identity certificate provided by the third-party vendor into the text field.



7. Click **Add Certificate**.

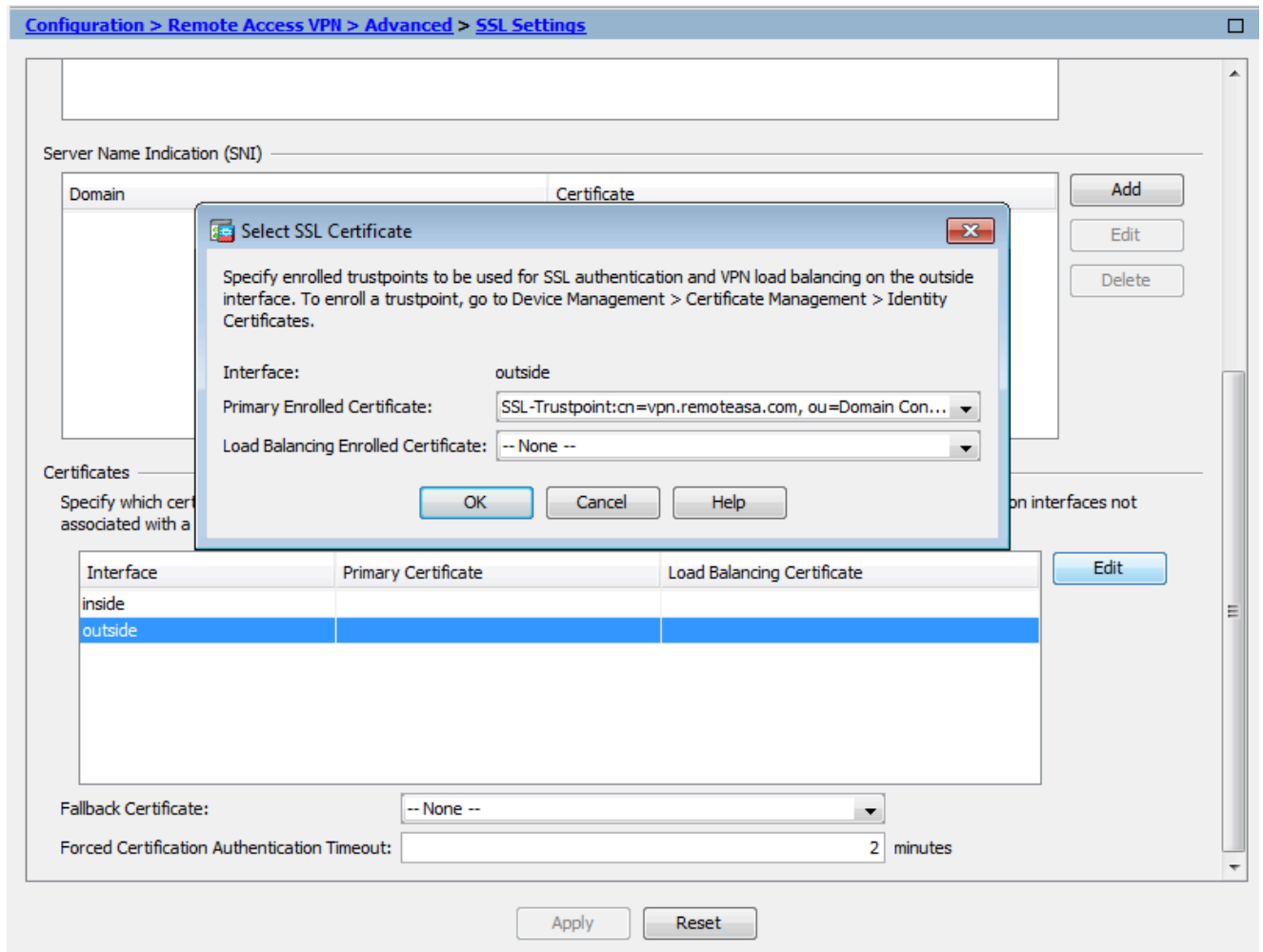


8. Navigate to **Configuration > Remote Access VPN > Advanced > SSL Settings**.

9. Under **Certificates**, select the interface that is used to terminate WebVPN sessions. In this example, the outside interface is used.

10. Click **Edit**.

11. In the **Certificate** drop-down list and choose the newly installed certificate.



12. Click **OK**.

13. Click **Apply**. The new certificate is now utilized for all WebVPN sessions that terminate on the interface specified.

1.2. Installation of a PEM Certificate with the CLI

```
<#root>
```

```
MainASA(config)#
```

```
crypto ca authenticate SSL-Trustpoint
```

Enter the base 64 encoded CA certificate.
End with the word "quit" on a line by itself

```
-----BEGIN CERTIFICATE----- MIIEDCCAuigAwIBAgIBADANBgkqhkiG9w0BAQUFADBjMQswCQYDVQQGEwJVuzEh MB8GA1UECh
```

```
!!! - Installing Next-level SubCA in the PKI hierarchy
```

```
.
```

```
!!! - Create a separate trustpoint to install the next subCA certificate (if present)  
in the hierarchy leading up to the Root CA (including the Root CA certificate)
```

```
MainASA(config)#crypto ca trustpoint SSL-Trustpoint-1
```

```
MainASA(config-ca-trustpoint)#enrollment terminal
MainASA(config-ca-trustpoint)#exit
MainASA(config)#
MainASA(config)# crypto ca authenticate SSL-Trustpoint-1
Enter the base 64 encoded CA certificate.
End with the word "quit" on a line by itself
```

```
-----BEGIN CERTIFICATE-----
MIIEFTCCA2WgAwIBAgIDG+cVMA0GCSqGSIb3DQEBCwUAMGMxCzAJBgNVBAYTA1VT
MSEwHwYDVQQKEWhUaGUgR28gRGFkZHKgR3JvdXAsIE1uYy4xMTAvBgNVBAsTKEdv
IERhZGR5IENsYXNzIDIgQ2VydG1maWNhdG1vbiBBdXRob3JpdHkwHhcNMTQwMTAx
MDcwMDAwWhcNMzEwNTMwMDcwMDAwWjCBgzELMAkGA1UEBhMCVVMxEDA0BgNVBAGT
B0FYaXpvc2VudG1maWNhdG1vbiBBdXRob3JpdHkwHhcNMTQwMTAxMDcwMDAwWjCB
Y29tLCBjb20wY29kZG1maWNhdG1vbiBBdXRob3JpdHkwHhcNMTQwMTAxMDcwMDAw
WjCBMAwGA1UdEwQFMASjBgZgZCAQAgMB0GA1UdDgQWBBQwY29kZG1maWNhdG1vbi
BBdXRob3JpdHkwHhcNMTQwMTAxMDcwMDAwWjCBMAwGA1UdEwQFMASjBgZgZCAQAg
dGhvcml0eSAtIEcyMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEA3Fi
CPH6WTT3G8kYo/eASVjPjIoMTpsUgQwE7hPHmUmFJ+r2hBt0oLTbcJjHMgGxBT4H
Tu70+k8vWTAi56sZvmvigaF88xZ1gD1Re+X5NbZ0TqmNghPktj+pA4P6or6KFWp/
3gvDthkUBcrqw6gE1DtGfDIN8wBmI5iNaW02jBEYt90yHGC00PoCjM7T3UYH3go+
6118yHz7sCtTpJJiaVE1BWEaRIGMLK1D1iPfrDqBmg4pxRyp6V0etp6eMAo5zvGI
gPtLXcwy7IViQyU0A1YnAZG003AqP26x6JyIAX2f1PnbU21gnb8s51iruF9G/M7E
GwM8CetJMVxpRrPgrWIDAQABo4IBFzCCARMwDwYDVR0TAQH/BAUwAwEB/zA0BgNV
HQ8BAf8EBAMCAQYwHQYDVR0OBBYEFdqahQcQZyi27/a9BUFuIMGU2g/eMB8GA1Ud
IwQYMBaAFNLEsNKR1EwRcbNhyz2h/t2oatTjMDQGCCsGAQUFBwEBBCgwJjAkBggr
BgEFBQcwAYYYaHR0cDovL29jc3AuZ29kYWRkeS5jb20vMDIGA1UdHwQrMCKwJ6A1
oCOGIWh0dHA6Ly9jcmwuZ29kYWRkeS5jb20vZ2Ryb290LmNybDBGBgNVHSAEPzA9
MDsGBFUdIAAwMzAxBggrBgEFBQcCARY1aHR0cHM6Ly9jZXJ0cy5nb2RlZGR5LmNv
bS9yZXBvc210b3J5LzANBgkqhkiG9w0BAQsFAAOCAQEAWQtTvZKGEacke+1bMc8d
H2xwxbhuvk679r6XU0Ewf7ooXGKUwuN+M/f7QnaF25UcjCJYdQkMiGVn0QowCcWg
0JekxS0TP7QYpgEGRJHj2kntFo1fzq3Ms3dhP8q0CkzPn1nsoX+oYggHFCJyNwq
9kIDN0zmiN/VryTyscPzfLXs4J1et01UIDyUGAZHHFIYSaRt4bNYC8nY7NmuHDK0
KHAN4v6mF56ED71XcLNa6R+gh10773z/aQvgSM03kwvIC1TErF0UZzdsyqUvMQg3
qm5vjLyb41ddJIGv15echK1srDdMZvNhkREg5L4wn3qkKQmw4TRfZHCyQFHfjDCm
rw==
-----END CERTIFICATE-----
quit
```

```
INFO: Certificate has the following attributes:
Fingerprint:      81528b89 e165204a 75ad85e8 c388cd68
Do you accept this certificate? [yes/no]: yes
```

Trustpoint 'SSL-Trustpoint-1' is a subordinate CA and holds a non self-signed certificate.

Trustpoint CA certificate accepted.

```
% Certificate successfully imported
BGL-G-17-ASA5500-8(config)#
```

!!! - Similarly create additional trustpoints (of the name "SSL-Trustpoint-n", where n is number thats incremented for every level in the PKI hierarchy) to import the CA certificates leading up to the Root CA certificate.

!!! - Importing identity certificate (import it in the first trustpoint that was created namely "SSL-Trustpoint")

```
MainASA(config)#
crypto ca import SSL-Trustpoint certificate
```


- Specify a Trustpoint name.
- Click the **Import the identity certificate from a file** radio button.
- Enter the passphrase used to create the PKCS12 file. Browse and select the PKCS12 file. Enter the certificate passphrase.

Add Identity Certificate

Trustpoint Name:

Import the identity certificate from a file (PKCS12 format with Certificate(s) +Private Key):

Decryption Passphrase:

File to Import From:

Add a new identity certificate:

Key Pair:

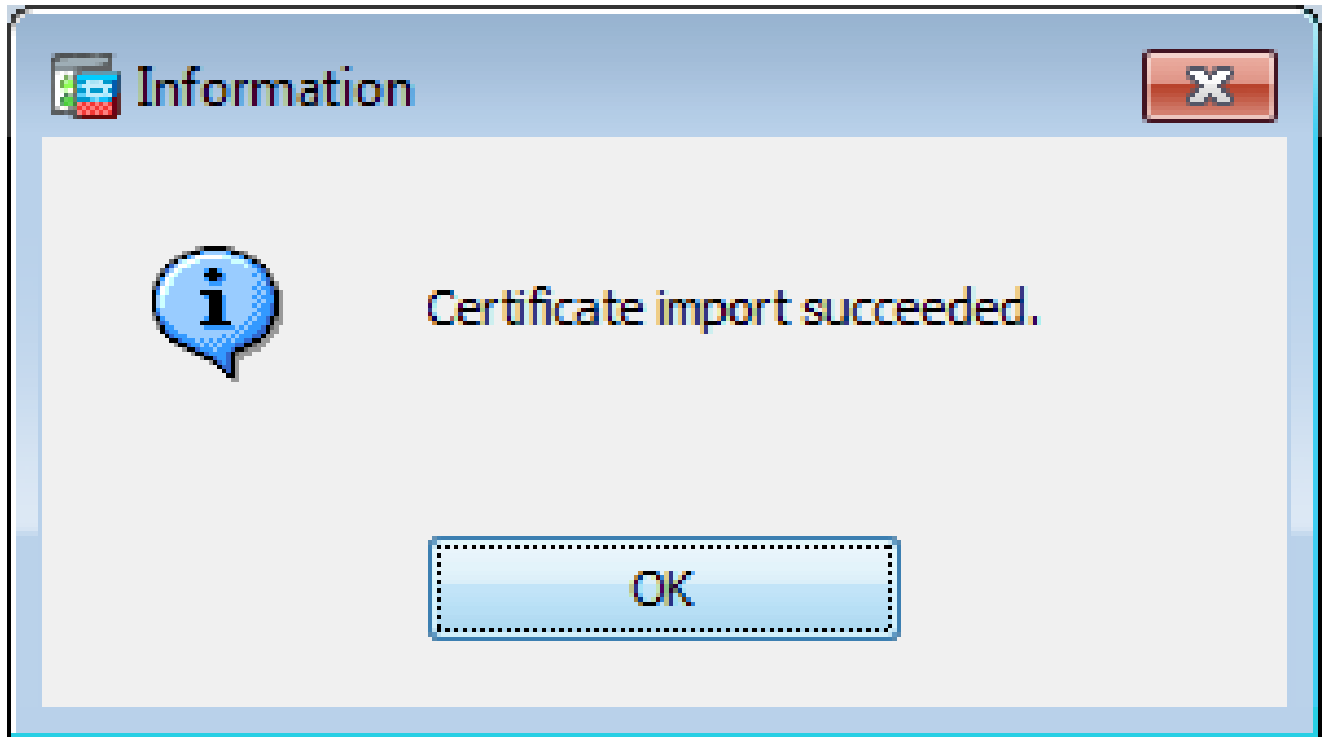
Certificate Subject DN:

Generate self-signed certificate

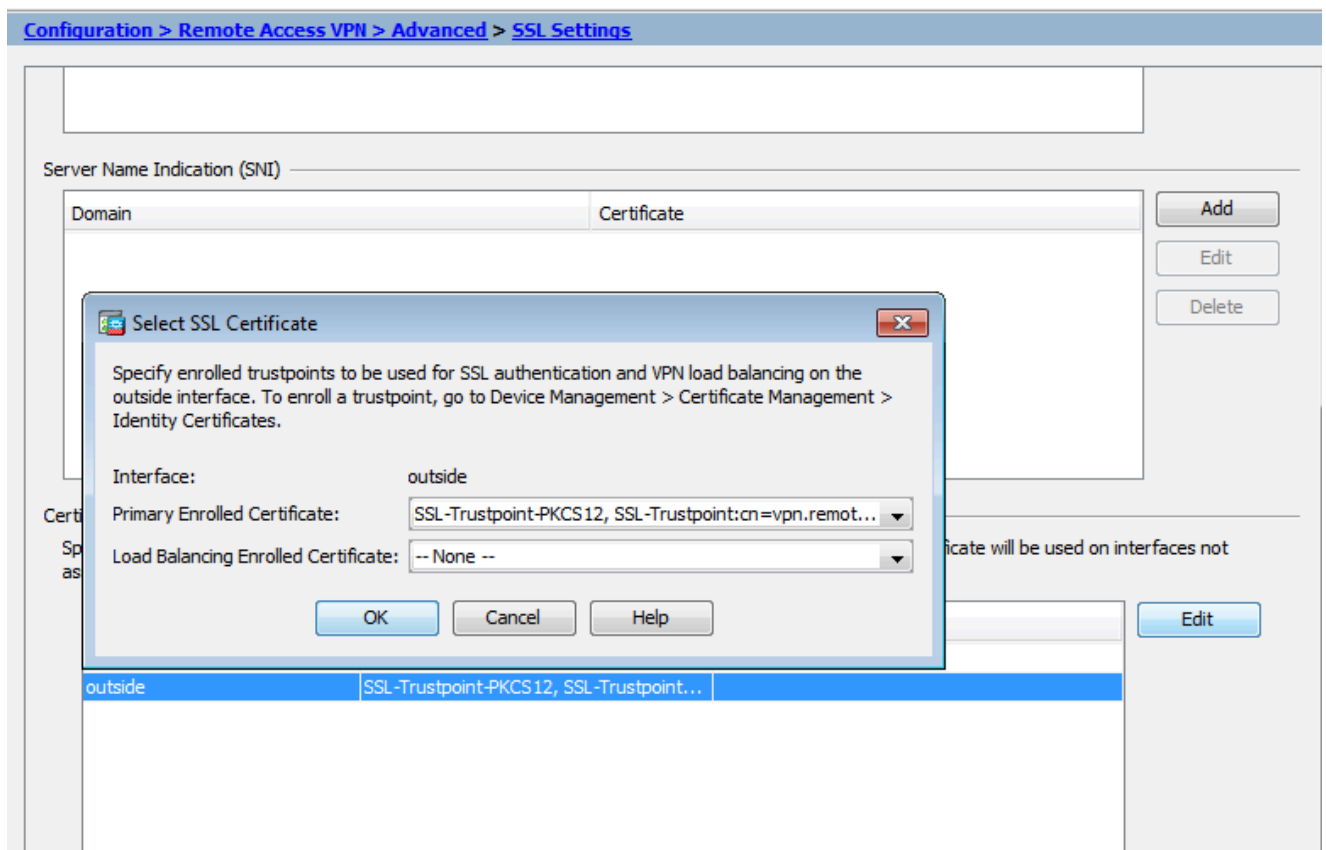
Act as local certificate authority and issue dynamic certificates to TLS-Proxy

Enable CA flag in basic constraints extension

- Click **Add Certificate**.



8. Navigate to **Configuration > Remote Access VPN > Advanced**, and choose **SSL Settings**.
9. Under **Certificates**, choose the interface that is used to terminate WebVPN sessions. In this example, the outside interface is used.
10. Click **Edit**.
11. In the Certificate drop-down list, choose the newly installed certificate.



12. Click **OK**.
13. Click **Apply**. The new certificate are now utilized for all WebVPN sessions that terminate on the

interface specified.

2.2 Installation of a PKCS12 Certificate with the CLI

```
<#root>
```

```
MainASA(config)#
```

```
crypto ca trustpoint SSL-Trustpoint-PKCS12
```

```
MainASA(config-ca-trustpoint)#
```

```
enrollment terminal
```

```
MainASA(config-ca-trustpoint)#
```

```
exit
```

```
MainASA(config)#
```

```
crypto ca import SSL-Trustpoint-PKCS12 pkcs12 cisco123
```

Enter the base 64 encoded pkcs12.

End with the word "quit" on a line by itself:

```
-----BEGIN PKCS12-----
```

```
MIISNwIBAzCCEfEGCSqGSIb3DQEHAaCCEeIEghHeMIIR2jCCEdYGCsGSIb3DQEH  
BqCCEccwghHDAgEAMIRvAYJKoZIhvcNAQcBMBsGCiqGSIb3DQEMAQMwDQIWO3D  
hDtI/uECAQGAghGQ9ospee/qtIbVZh2T8/Z+5dxRPBcStDTqyKy7q3+9ram5AZdG  
Ce9n5UCckqT4WcTjs7XZtCrUrt/LkNbmGDVhwGBmYwi0S7npgaUq0eoqiJRK+Yc7  
LN0nbho6I5WfL56/JiceAM1XDLr/IqqLg2QAAPGdN+F5vANsHse2GsAATewBDLt7  
Jy+SKfoNvvIw9QvzCiUzMjYZBANmBdMCQ13H+YQTHitT3vn2/iCD1zRSuXcqypEV  
q5e3hei00751E8TDLWm03PMvWIZqi8yzWesjcTt1Kd4FoJBZpB70/v9LntoIUOY7  
kIQM8fHb4ga8BYfbgRmG6mkMm01STbtSv1vTa19WTmdQdTycA+G5PkrryRsy3Ww1  
1kGFMhImmrnNADF7Hmzbys1VohQZ7h09iVQY9krJogoXHjmQYxG9brf0oEwxSJD  
mGDhhESh+s/WuFSV9Z9kiTXpJNZxpTASoWBQrrwm05v8ZwbjbVNJ7sVdbwpU16d+  
NNFGR7LTq08hpueeJny9eJc2yYqeAXWXQ5kL0Zo6/gBEdGtEaZBgCFK9JZ3b13A  
xqxGi fanWpNLyG611NKuNjTgbjhnEEYI2uZzU0qxn1Ka8zyXw+1zrKuJscDbkAPZ  
wKtw8K+p40zXVHhuANO6MDvffNRY1KQDtyK1inoPH5ksVSE5awkVam4+HTcqEUfa  
16LMana+4QRgSetJhU0LtsMaQfRJGkha4JLq2t+JrCAPz2osAR1TsB0jQBNq6YNj  
OuB+gGk2G18Q5N1n6K1fz0XBFLWEDBLsaBR05MAnE7wWt00+4awGYqVdmIF11kf  
XIRKAiQEr1pZ6BVPuvscNjXaaUHzufhYI2ZackasKBZ0T8/7YK3fnAaGoBCz4cHa  
o2EEQhq2aYb6YTv0+wtLEWGHZsbGZEM/u54XmsXAI7g28LGJYdfwi509KyV+Ac1V  
KzHqXZMM2BbUQCncTF5JIMiW+r62k42FdahfaQb0vJsIe/IwkAKG7y6DIQFs0hwg  
Z1PXiDbNr1k4e8L4gqumMKWg853PY+oY22rLDC7bu11CKtixIYBcvbn7dAYsI4GQ  
16xXhNu3+ie0HgbUQCcftU/mBrAOZO+bpKjWOCfqNBuYnZ6kUEdCI7GFLH9QqtM  
K7YinFLoHwTWbi3MsmqVv+Z4ttVwy7Xmiko02nMynJMP6/CNV80MxMKdC2qm+c1j  
s4Q1KcAmFsQmNp/7SIP1wnv0c6JbUmC10520U/r8ftTzn8C7WL62W79cLK4HOr7J  
sNsZnOz0J0Z/xdZT+cLTctVevKJQqMK3vMsiOuy52FkuF3HnfrmBqDkbR7yZxELG  
RCELOEDdbp8VP0+IhN1yz1q7975ScdxFSL0TvjnHGFwd14ndoqN+bLhWbdPjQWV  
13W2NCI95tmHDLGgp3P001S+rjdCEGGMg+9cpgBfFC1JocuTDIEcUbJBY8QRUNiS  
/ubyUagdzUKt1ecfb9hMLP65ZnQ93VIw/NJKbIm7b4P/1Zp/1FP5eq7LkQPaxE4/  
bQ4mHcnwrs+JGfkn19B8hJmmGoowH3p4IEvwZy7CThB3E1ejw5R4enqmrgrvHqpQe  
B7odN10FLAhd01G5BsHExlunEsEb40Q0pmKXiDDB5B001bJsr748fZ6L/LGx8A13  
<snip>
```

```
ijDqxyfQXY4zSyt1jSmWmtYA9hG5I79Sg7pnME1E9xq1D0oRGg8vgxlwiciKtLxp  
LL0ReDY31KRYv00vW0gf+tE71ST/3TKZvh0sQ/BE0V3kHnw1dejMFH+dvYAA9Y1E  
c80+tdaFBFX4B/HP46E6heP6ZSt0xAfRW1/JF41jNvUNV09VtVfR2FTyWpzZFY8A  
GG5XPIA80WF6wKEPFHICn8scY+Vot8kXxG96hwt2Cm5NQ20nVzxUZQbpKsjs/2jC  
3HVFe3UJFBsY9UxTLcPXYSIG+VeqkI8hWZp6c1TFNDLY2ELdy1Qzp1mBg2FujZa  
YuE0avjCjzBzZUG2umt55mHQnwPF+XkOujEyhGMauhGxHp4nghSzrUZrBeuL91UF
```

2mbps0cgZkzxMS/rjdNXjCmPF1oRBvKkZS1xHFRE/5ZopAhn4i7YtHQNrZ9U4RjQ
xo9cUuaJ+LNmvzE8Yg3epAMYZ16UNGQQkVQ6ME4BcjRONzW8BYgTq4+pmT1ZNq1P
X87CXCPtYrPHF57eSo+tHDINCgfqYXD6e/7r2ngfiCeUeNDZ4aV12XxvZDaU1BPP
Tx5fMARqx/Z8BdDyBJDVBjdsxmQau9HLkhPvdFG1ZIWdTe13CzKqXA5Pmpjt4q9
GnCpC53m76x9Su4ZDw6aUdBcgCTMvfaqJC9gz0bee2Wz+aRRwzSxu6tEWVZo1PEM
v0AA7po3vPek1g0nLRAwEoTTn4SdgNLWeRoxqZgkw1FC1GrotxF1so7uA+z0aMeU
Tw73reonsNdZvRAcVX3Y6UNFDyt70Ixvo1H4VLzWmOK/op62C9/eqqMwZ8zoCMPt
ENna7T+70s66SCbMmXCHwyh00tygNKZFFw/AATFyjQPMWPaxGuPN0rnB6uYcN0Hk
1BU7tF143RNIzaQQEH3XnaPvUuAA4C0FCoE3h+/tVjtfNKDvFmb6ZLZHYQmUYpyS
uhdFEpoDrJH1Vmi2tik/iqYwaz+oDqXPHQXnJhw25h9ombR4qnD+FCfwFCGtPFON
o3Qffz53C95n5jPHVMYUr0xDdpwnvzCQPdj6yQm564TwLAmiZ7uD1pqJZJe5QxHD
no1v+4MdGSfVtBq+ykFoVcaamqeaq6sKgvAVujLXXEs4KEmIgcPqATVRG49E1ndI
L01DEQyKhVoDGebAuVRBjzwAm/qxWxxFv3hrbCjPHCwEYms4Wgt/vKKRFsuWJNZf
efH1dw11tkd5dKwSvDocPT/7mSLtLJa94c6AfgxYy9z0+FTLDQwzXga7xC2krAN1
yHxR2KHN5YeRL+KDzu+u6dYoKaz+YAgw1W6KbeavALSuH4EYqcvG8hUEhp/ySiSc
RDhuygxEovIMGfES4FP5V521PyDhM3Dqwhn0vuYUmYnX8EXURkay44iwwI5HhqYJ
TptWYyO8Bdr4Wnwt5xqsZgYR6mmGeAIin7bDunsF1uBHWYF4dyK1z1tsdRNMqYQ
+W5q+QjVdrj1dwv/bMF0aqEjxeNwBRqjzccff3BxMnwvVxtgqxFvRh+DZxiJoiBG+
yx7x8np2AQ1r0METSSxbnZzfNkZKvBVMkIC6Jsm2WEVTQvoFJ8em+nem0WgTi/
hHSBzjE7RhAucnHuiFOCX0gvR1SDDqyCQbduc1QjXN0svA8Fqbea9WEH5khOPv3
pbtSL4gsf12pv8diBQkVQgiZDi8wb++7PR6ttiY65kVwrdson11/qq+xW0d3tB4/
zoH9LEMgTy9Ssz7myWrB9E00Z8BIjL1M8oMigEYrTD0c3KbyW1S9dd7QAxIU0BaX1
8J8q10ydvTBzmqcjesFH4/1NHn5Vnf0ZnNpui4uHP0XBG+K2zJUJXm6dq1AHB1E
KQFsFzPNNyave0Kk8JzQnLAPd70UU/Iksy0CGQozGBH+HSzVp1RDjrrbC342rkBj
wnI+j+/1JdwBmHdJMZCfoMZFLSI9ZBqFirdii1/NRu6jh76TQor5TnNjxIyNREJC
FE5FZnMfVhM900LaiUZff8WWCOferDMttLXb1nuxPF1+1Rk+LN1PLVptWgcxzfSr
JXrGiWjxybBB9oCOrAcq8fGatEs8WRxJyDH3Jjmn9i/G16J1mMCUF//LxAH2WQx8
Ld/qS50M2iFCffDQjxAj0K6DEN5pUebBv1Em5SOHXvyq5nxgUh4/y84CwaKjwOMQ
5tbbLM1nc7ALIj9LxZ97YiXSTyeM6oBxBfx6Rpk1kDv05m1BghSpVQiMcQ20RIkh
UVVNBsh019S3cb5wqxaWqAKBqb4h1uLGVbYWZf2mzLZ8U5U5ioiqoMBqNZbzTXp0
EqEFuatT11QvCRbcKS3xou4MAixcYUxKwEhbZA/6hd10XSBjwe7jKBV9M6w1iKab
UfoJCGTaf3sY681qrMPrbt0eewf1C02Sd9Mn+V/jvni17mxYFFUpruRq3r1LeqP
J5camfTtHwyL8N3Q/Zwp+zQeWziLA8a/iAVu/hYLR1bpF2WCK010tJqkvVmrLVLz
maZZjbJe0ft5cP/1RxbK1S6Gd5dFTEKDE15c6gWUX8RKZP6Q7iaE5hnGmQjm8Lj1
kXwF+ivox0Q8a+Gg1bVTR0c7tqW9e9/ewisV1mwvEB6Ny7TDS1oPUDHM84pY6dqi
1+0io07Ked4BySwN1Yy9yaJtBTZSCstfP+ApLiDn7pSBvvXf1aHmeNbkPOZJ+c+t
fGpUdL6V2UTXfCsOPHTC0ezA15sOHwCuPchrDIj/eGUwMS3NfS25XgcMuvnLqGVO
RzcrZ1ZIG8G0oLYwOCuzoY0D/m901001ahePyA9tmVB7HRRbytLdaW7gYeEikoCv
7qtBqJFF17ntWJ3EpQHZUCVClbHIKqjNqRbDCY7so4A1IW7kSEUGWMIUDhprE8Ks
NpvnPH2i9JrYrTeRoYUI0tL/7SATd2P0a21xz/zUwekeqd0bmVCsAgQNbB2XkrR3
XS0B52o1+63e8KDqS2zL2TZd3daDFidH1B8QB26tfbf0Aca0bJH5/dWP8ddo8UYo
Y3JqT10malxSjhaMhMqDZIqP49utW3Tcjg11YS4HEmcqtHud0ShaUysC6239j1Q
K1FwrwXT1BC5vnq5IcOMqx5zyNbfXz28969cwoMCyU6+kRw0TyF6kF7EEv6XWca
XLEwABx+tKRUKHJ673SyDMu96KMV3yZN+RtKbCjqCPVTP/3ZeIp7nCMUcj5sW9HI
N34yeI/ORCLyeGs0EiBLkucikC32LI9ik5HvImVTELQ0Uz3ceFqU/PkasjJUve6S
/n/1ZVUHbUk71xKR2bwZgEC17fIe17w1rbjP3Wbk+Er0kfYcsNRHxeTDpKPS9s
u/UsyQJiyNARG4X3iYQ1sTce/06Ycyri6GcLHAu58B02nj4Cxo1Cp1ABZ2N79HtN
/7Kh5L0pS9MwsDCHUUI8KFRtSET7TB1tIU99FdB19L64s1/shYAHbccvVWU50WhT
PdLoaErrX81Tof41IxbSZbI8grUC4KfG2sdPLJKu3HVTeQ8Lfl1bBLxfs8ZBS+Oc
v8rH1Q012kY6LsFGLehJ+/yJ/uvXORiv0ESp4EhFpFfkp+o+YcFeLUUPd+jzb62K
HFSCCbLpKCyEay80dyWkHfgy1qXmb9ud0oM050aFJyqRONjnt6pcxBRY2A6AJR5S
IIC26YNwbh0GjF9qL2FiUqnNH/7GTqPnd2qmsB6FTIwSBT6d854qN7PRt+ZXgdtQ
Ojcyt1r9qpWZpNFK8EzizwKiAYtsiEh2pzPt6YUpxRb6CXTkIzoG+Klsv2m3b8
OHyZ9a8z81/gnxrZ11s5SCTf0SU70pHWh8VAYKVHhK+MwgQrOm/2ocV32dkRBLMy
2R6P4WfHyI/+9de1x3PtIu0iv2knpxHv2fKM6sQw45F7XkmwHxjq1YRJ6vIwPTAh
MAKGBSs0AwIaBQAeffTRETzpiSHKZR+Kmen68VrTwpV7BBSQi0IesQ4n4E/bSVsd
qJSzcwh0hgICBAA=
-----END PKCS12-----
quit

INFO: Import PKCS12 operation completed successfully

!!! Link the SSL trustpoint to the appropriate interface
MainASA(config)#

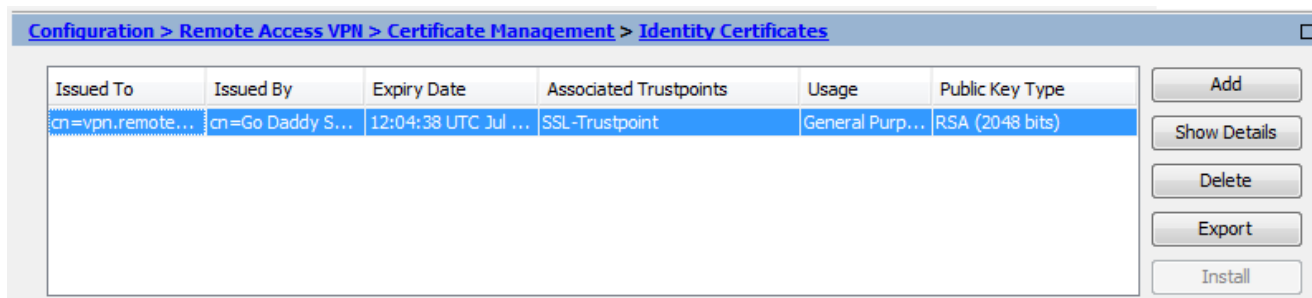
```
ssl trust-point SSL-Trustpoint-PKCS12 outside
```

Verify

Use these steps in order to verify successful installation of the third-party Vendor Certificate and use for SSLVPN connections.

View Installed Certificates via ASDM

1. Navigate to **Configuration > Remote Access VPN > Certificate Management**, and choose **Identity Certificates**.
2. The identity certificate issued by the third-party vendor appears.



The screenshot shows the ASDM interface for Identity Certificates. The breadcrumb path is Configuration > Remote Access VPN > Certificate Management > Identity Certificates. A table lists the installed certificates:

Issued To	Issued By	Expiry Date	Associated Trustpoints	Usage	Public Key Type
cn=vpn.remote...	cn=Go Daddy S...	12:04:38 UTC Jul ...	SSL-Trustpoint	General Purp...	RSA (2048 bits)

On the right side of the table, there are buttons for Add, Show Details, Delete, Export, and Install.

View Installed Certificates via the CLI

```
<#root>
```

```
MainASA(config)#
```

```
show crypto ca certificate
```

Certificate

```
Status: Available
Certificate Serial Number: 25cd73a984070605
Certificate Usage: General Purpose
Public Key Type: RSA (2048 bits)
Signature Algorithm: SHA256 with RSA Encryption
Issuer Name:
  cn=Go Daddy Secure Certificate Authority - G2
  ou=http://certs.godaddy.com/repository/
  o=GoDaddy.com\, Inc.
  l=Scottsdale
  st=Arizona
  c=US
Subject Name:
  cn=(asa.remotevpn.url)
  ou=Domain Control Validated
OCSP AIA:
  URL: http://ocsp.godaddy.com/
CRL Distribution Points:
```


[1] <http://cr1.godaddy.com/gdig2s1-96.cr1>
Validity Date:
start date: 12:04:38 UTC Jul 22 2015
end date: 12:04:38 UTC Jul 22 2016
Associated Trustpoints:

SSL-Trustpoint

CA Certificate

Status: Available
Certificate Serial Number: 07
Certificate Usage: General Purpose
Public Key Type: RSA (2048 bits)
Signature Algorithm: SHA256 with RSA Encryption
Issuer Name:
cn=Go Daddy Root Certificate Authority - G2
o=GoDaddy.com\, Inc.
l=Scottsdale
st=Arizona
c=US
Subject Name:
cn=Go Daddy Secure Certificate Authority - G2
ou=<http://certs.godaddy.com/repository/>
o=GoDaddy.com\, Inc.
l=Scottsdale
st=Arizona
c=US
OCSP AIA:
URL: <http://ocsp.godaddy.com/>
CRL Distribution Points:
[1] <http://cr1.godaddy.com/gdroot-g2.cr1>
Validity Date:
start date: 07:00:00 UTC May 3 2011
end date: 07:00:00 UTC May 3 2031
Associated Trustpoints:

SSL-Trustpoint

CA Certificate

Status: Available
Certificate Serial Number: 1be715
Certificate Usage: General Purpose
Public Key Type: RSA (2048 bits)
Signature Algorithm: SHA256 with RSA Encryption
Issuer Name:
ou=Go Daddy Class 2 Certification Authority
o=The Go Daddy Group\, Inc.
c=US
Subject Name:
cn=Go Daddy Root Certificate Authority - G2
o=GoDaddy.com\, Inc.
l=Scottsdale
st=Arizona
c=US
OCSP AIA:
URL: <http://ocsp.godaddy.com/>

CRL Distribution Points:
[1] http://crl.godaddy.com/gdroot.crl
Validity Date:
start date: 07:00:00 UTC Jan 1 2014
end date: 07:00:00 UTC May 30 2031
Associated Trustpoints:

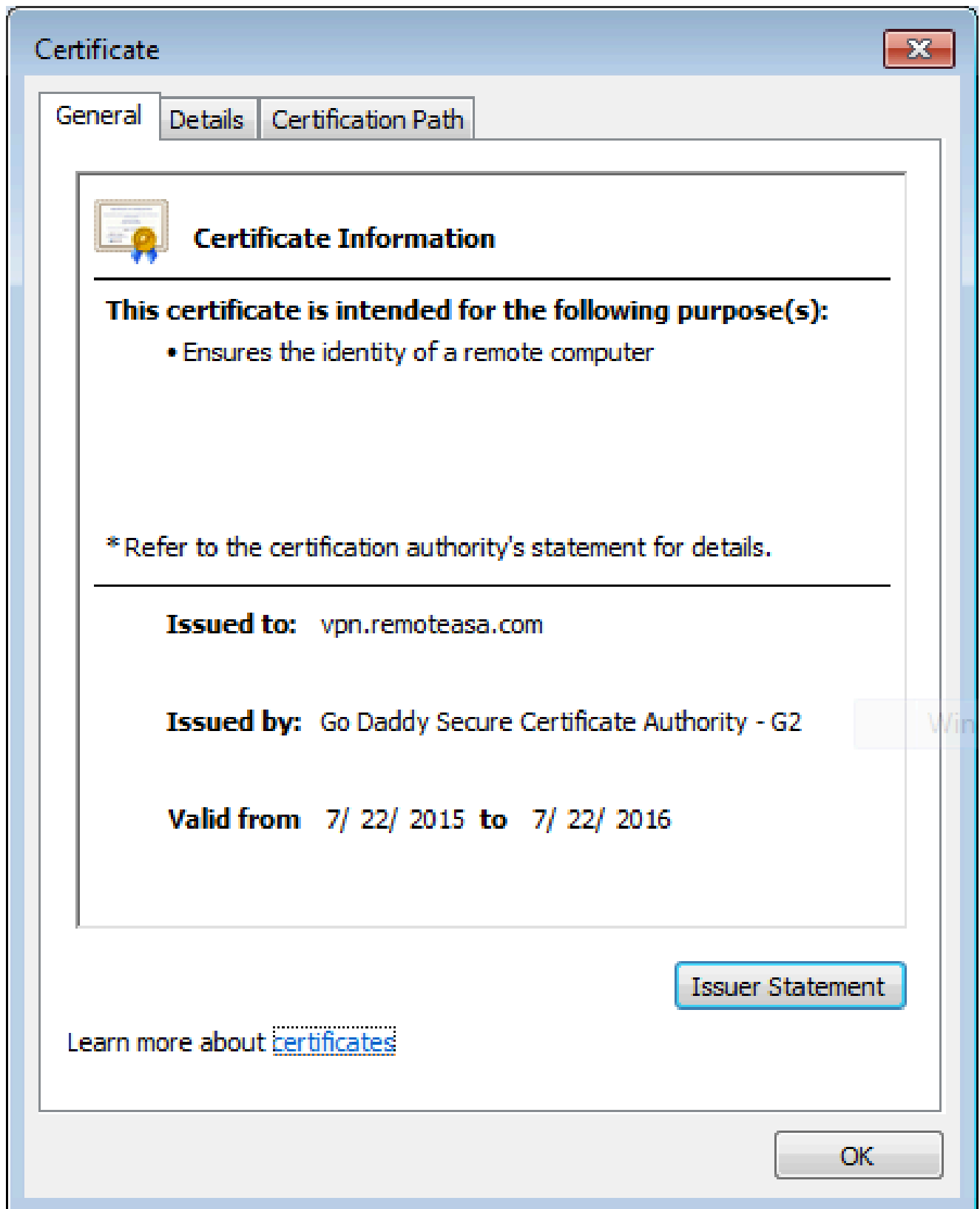
SSL-Trustpoint-1

...(and the rest of the Sub CA certificates till the Root CA)

Verify Installed Certificate for WebVPN with a Web Browser

Verify that WebVPN uses the new certificate.

1. Connect to the WebVPN interface through a web browser. Use https:// along with the FQDN used in order to request the certificate (for example, [https://\(vpn.remoteasa.com\)](https://vpn.remoteasa.com)).
2. Double-click the lock icon that appears in the lower-right corner of the WebVPN login page. The installed certificate information must appear.
3. Review the contents in order to verify that it matches the third-party vendor issued certificate.

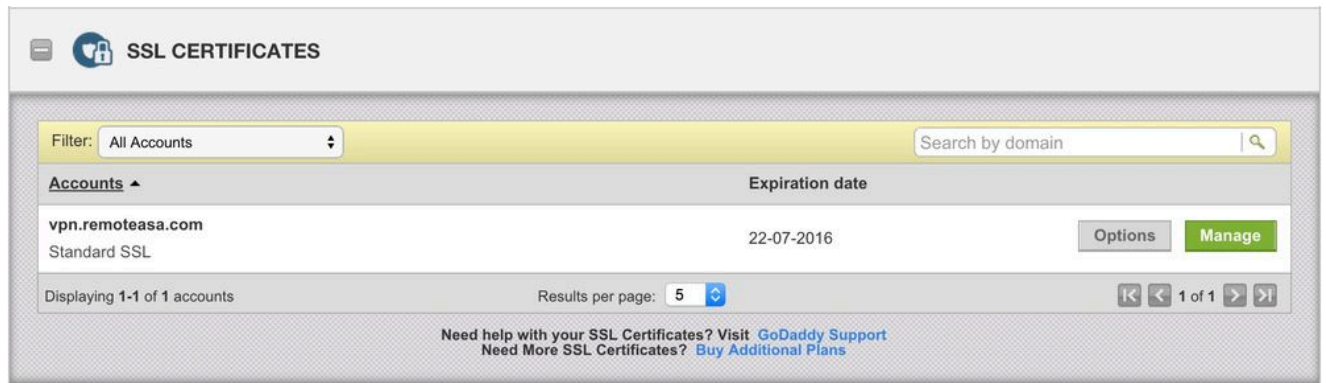


Renew SSL Certificate on the ASA

1. Regenerate the CSR either on the ASA, or with OpenSSL or on the CA with the same attributes as the old certificate. Complete the steps given in [CSR Generation](#).
2. Submit the CSR on the CA and generate a new Identity certificate in PEM format (.pem, .cer, .crt) along with the CA certificate. In the case of a PKCS12 certificate there is also a new Private key.

In the case of GoDaddy CA, the certificate can be rekeyed with a new CSR generated.

Go to the GoDaddy account and click **Manage** under SSL Certificates.



Click **View Status** for the required domain name.



Click **Manage** in order to give options to re-key the certificate.

All > vpn.remoteasa.com

Standard SSL Certificate

Certificate Management Options



Download



Revoke



Manage

Certificate Details

Status	Certificate issued
Domain name	vpn.remoteasa.com
Encryption Strength	GoDaddy SHA-2
Validity Period	7/22/2015 - 7/22/2016
Serial Number	25.cd:73:a9:84:07:06:05

Expand the option **Re-Key certificate** and add the new CSR.

vpn.remoteasa.com > Manage Certificate

Standard SSL Certificate

Use this page to submit your certificate changes for review all at once, not individually. We'll review them together so your changes happen faster.

Submitting any changes on this form will issue a new certificate and your current certificate will be revoked. You will have 72 hours to install the new certificate on your website.

Re-Key certificate *Private key lost, compromised, or stolen? Time to re-key.*

Certificate Signing Request (CSR)

Domain Name (based on CSR):
vpn.remoteasa.com

New Keys, please...
You can generate a Certificate Signing Request (CSR) by using a certificate signing tool specific to your operating system. Your CSR contains a public key that matches the private key generated at the same time.

Change the site that your certificate protects *If you want to switch your certificate from one site to another, do it here.*

Change encryption algorithm and/or certificate issuer *Upgrade your protection or change the company behind your cert.*

Save and proceed to the next step. GoDaddy issues a new certificate based on the CSR provided.

3. Install the new certificate on a new trustpoint as shown in the SSL Certificate Installation on the ASA section.

Frequently Asked Questions

1. What is the best way to transfer identity certificates out of one ASA onto a different ASA?

Export the certificate along with the keys to a PKCS12 file.

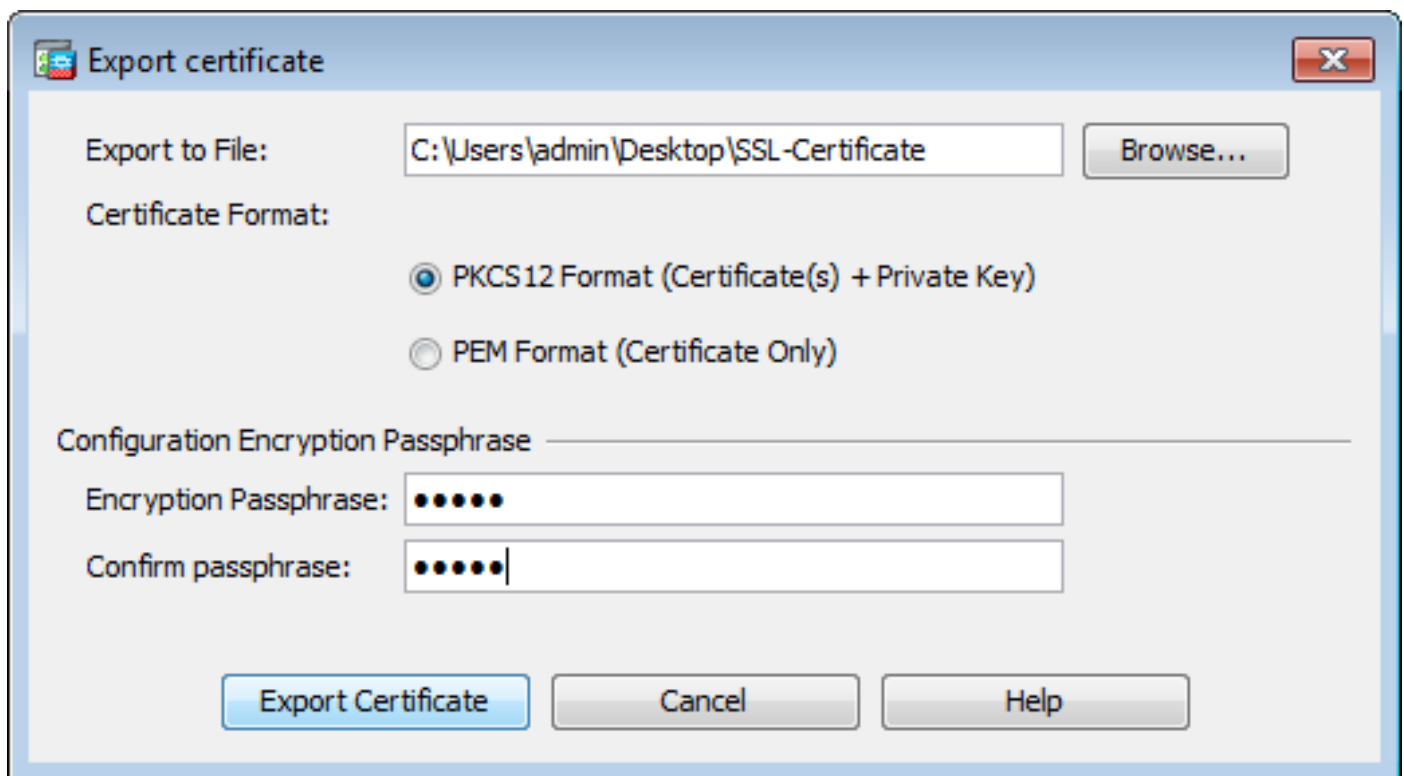
Use this command in order to export the certificate via the CLI from the original ASA:

```
<#root>
```

```
ASA(config)#
```

```
crypto ca export <trust-point-name> pkcs12 <passphrase>
```

ASDM configuration:



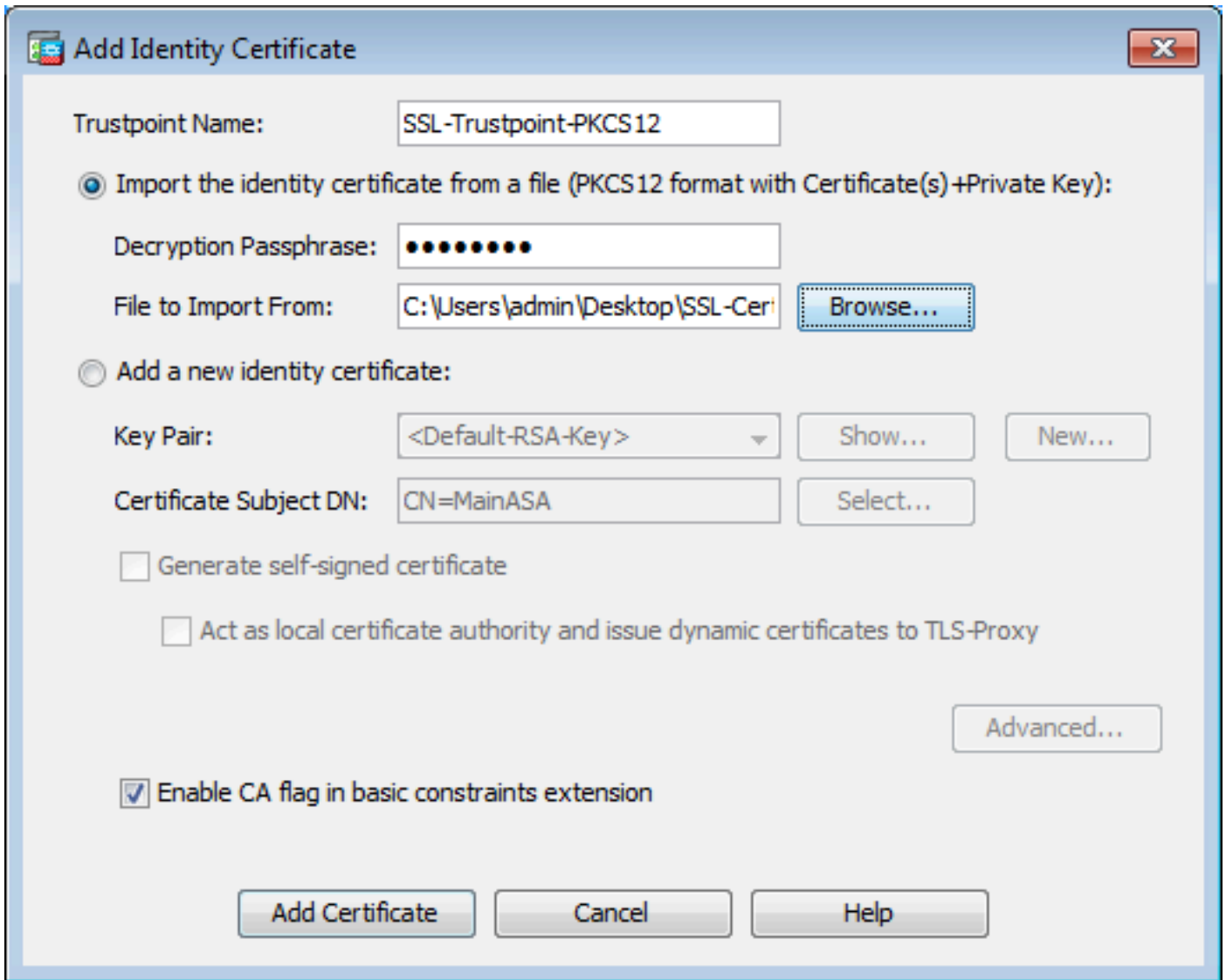
Use this command in order to import the certificate via CLI to the target ASA:

```
<#root>
```

```
ASA(config)#
```

```
crypto ca import <trust-point-name> pkcs12 <passphrase>
```

ASDM configuration:



This can also be done via the Backup/Restore feature on the ASDM with these steps:

1. Log in to the ASA via ASDM and choose **Tools > Backup Configuration**.
2. Backup All Configuration or just the Identity certificates.
3. On the target ASA, open the ASDM and choose **Tools > Restore Configuration**.

2. How to generate SSL certificates for use with VPN Load Balancing ASAs?

There are multiple methods that can be used to set up ASAs with SSL certificates for a VPN Load Balancing environment.

1. Use a single Unified Communications/Multiple Domains Certificate (UCC) which has the load-balancing FQDN as the DN and each of the ASA FQDNs as a separate Subject Alternative Name (SAN). There are several well known CAs like GoDaddy, Entrust, Comodo and others that support such certificates. When you choose this method, it is important to remember that the ASA currently does not support the creation of a CSR with multiple SAN fields. This has been documented in the enhancement Cisco bug ID [CSCso70867](#) . In this case there are two options to generate the CSR
 - a. Via the CLI or ASDM. When the CSR is submitted to the CA, add in the multiple SANs on the CA portal itself.
 - b. Use OpenSSL to generate the CSR and include the multiple SANs in the openssl.cnf file.

Once the CSR has been submitted to the CA and the certificate generated, import this PEM certificate to the ASA that generated the CSR. Once done, export and import this certificate in the PKCS12 format onto the other member ASAs.

2. Use a Wildcard certificate. This is a less secure and flexible method when compared to a UC certificate. In the case that the CA does not support UC certificates, a CSR is generated either on the CA or with OpenSSL where the FQDN is on the form of *.domain.com. Once the CSR has been submitted to the CA and the certificate generated, import the PKCS12 certificate to all the ASAs in the cluster.
3. Use a separate certificate for each of the member ASAs and the for the load-balancing FQDN. This is the least effective solution. The certificates for each of the individual ASAs can be created as shown in this document. The certificate for the VPN Loadbalancing FQDN is created on one ASA and exported and imported as a PKCS12 certificate onto the other ASAs.

3. Do the certificates need to be copied from the Primary ASA to the Secondary ASA in an ASA failover pair?

There is no need to manually copy the certificates from the Primary to Secondary ASA as the certificates are synced between the ASAs as long as Stateful Failover is configured. If on initial setup of failover, the certificates are not seen on the Standby device, issue the command **write standby** in order to force a sync.

4. If ECDSA keys are used, is the SSL certificate generation process different?

The only difference in configuration is the keypair generation step, where an ECDSA keypair is generated instead of an RSA keypair. The rest of the steps remain the same. The CLI command to generate ECDSA keys are shown here:

```
<#root>
```

```
MainASA(config)#
```

```
crypto key generate ecdsa label SSL-Keypair elliptic-curve 256
```

```
INFO: The name for the keys will be: SSL-Keypair  
Keypair generation process begin. Please wait...
```

Troubleshoot

Troubleshoot Commands

These debug commands are to be collected on the CLI in the case of an SSL Certificate Installation failure:

```
debug crypto ca 255
```

```
debug crypto ca messages 255
```

```
debug crypto ca transactions 255
```

Common Issues

Untrusted certificate warning with a valid third-party SSL certificate on the external interface on ASA with 9.4(1) and later.

Solution: This issue presents itself when an RSA keypair is used with the certificate. On ASA versions from 9.4(1) onwards, all the ECDSA and RSA ciphers are enabled by default and the strongest cipher (usually an ECDSA cipher) is used for negotiation. If this happens, the ASA presents a Self-Signed certificate instead of the currently configured RSA-based certificate. There is an enhancement in place to change the behaviour when an RSA-based certificate is installed on an interface and is tracked by Cisco bug ID [CSCuu02848](#).

Recommended Action: Disable ECDSA ciphers with these CLI commands:

```
ssl cipher tls1.2 custom "AES256-SHA:AES128-SHA:DHE-RSA-AES256-SHA:DHE-RSA-AES128-SHA:DES-CBC3-SHA:DES-CBC-SHA:RC4-SHA:RC4-MD5"
```

Or, with the ASDM, navigate to **Configuration > Remote Access VPN > Advanced**, and choose **SSL Settings**. Under the Encryption section, select **tls1.2 Cipher version** and edit it with the custom string **AES256-SHA:AES128-SHA:DHE-RSA-AES256-SHA:DHE-RSA-AES128-SHA:DES-CBC3-SHA:DES-CBC-SHA:RC4-SHA:RC4-MD5**

Appendix

Appendix A: ECDSA or RSA

The ECDSA algorithm is a part of the Elliptic curve cryptography (ECC) and uses an equation of an elliptic curve to generate a Public Key whereas the RSA algorithm uses the product of two primes plus a smaller number to generate the Public Key. This means that with ECDSA the same level of security as RSA can be achieved, but with smaller keys. This reduces computation time and increases the connection times for sites that use ECDSA certificates.

The document on [Next Generation Cryptography and the ASA](#) provides more in-depth information.

Appendix B: Use OpenSSL to Generate a PKCS12 Certificate from an Identity Certificate, CA Certificate, and Private Key

1. Verify that the OpenSSL is installed on the system that this process is run on. For Mac OSX and GNU/Linux users, this is installed by default.
2. Switch to a valid directory.

On Windows: By default, the utilities are installed in C:\Openssl\bin. Open a command prompt in this location.

On Mac OSX/Linux: Open the Terminal window in the directory needed to create the PKCS12 certificate.

3. In the directory mentioned in the previous step, save the private key (privateKey.key), identity certificate (certificate.crt) and root CA certificate chain (CACert.crt) files.

Combine the private key, identity certificate and the root CA certificate chain into a PKCS12 file. Enter a passphrase to protect your PKCS12 certificate.

```
strong> openssl pkcs12 -export -out certificate.pfx -inkey privateKey.key -in certificate.crt -cer
```

4. Convert the PKCS12 certificate generated to a Base64 encoded certificate:

```
<#root>
```

```
openssl base64 -in certificate.pfx -out certificate.p12
```

Next, import the certificate that was generated in the last step for use with SSL.

Related Information

- [ASA 9.x Configuration Guide - Configure Digital Certificates](#)
- [How to obtain a Digital Certificate from a Microsoft Windows CA with ASDM on an ASA](#)
- [Technical Support & Documentation - Cisco Systems](#)