



# Creating SSL Certificates to Use with SD-AVC

- [Summary, on page 1](#)
- [Using a Certificate Signed by a Certification Authority, on page 2](#)
- [Using a Self-signed SSL Certificate Created with Keytool, on page 2](#)
- [Using a Self-signed SSL Certificate Created with OpenSSL, on page 4](#)

## Summary

### Create certificate to be signed by certification authority

	Task	Where to find...
1	Create certificate keys.	See <a href="#">Using a Certificate Signed by a Certification Authority, on page 2</a> .
2	Generate a certificate signing request (CSR).	
3	Send the CSR file to be signed by the certification authority.	
4	Install the signed certificate in the SD-AVC Dashboard.	See "Serviceability Page" in <a href="#">Using SD-AVC</a> .

### Create self-signed certificate

	Task	Where to find...
1	Create self-signed certificate keys.	See <a href="#">Using a Self-signed SSL Certificate Created with Keytool, on page 2</a> . See <a href="#">Using a Self-signed SSL Certificate Created with OpenSSL, on page 4</a> .
2	Install the signed certificate in the SD-AVC Dashboard.	See "Serviceability Page" in <a href="#">Using SD-AVC</a> .

# Using a Certificate Signed by a Certification Authority

You can use the **keytool** or **OpenSSL** command line utilities to create a certificate to be signed by a certification authority, and used with Cisco SD-AVC.

## Using Keytool

1. Create certificate keys.

### Example:

```
keytool -genkey -alias sdavc_alias -keyalg RSA -sigalg SHA1withRSA -keysize 2048 -keystore mykeystore.keystore
```

2. Generate a certificate signing request (CSR).

### Example:

```
keytool -certreq -alias sdavc_alias -keyalg RSA -sigalg SHA1withRSA -file mycsrfile.csr -keystore mykeystore.keystore
```

The command produces a CSR file called **mycsrfile.csr**.

3. Send the CSR file to be signed by the certification authority.
4. Install the signed certificate in the SD-AVC Dashboard. See "Serviceability Page" in [Using SD-AVC](#).

## Using OpenSSL

1. Create certificate keys.

### Example:

```
openssl genrsa -des3 -out server.key 2048
```

2. Generate a certificate signing request (CSR).

### Example:

```
openssl req -new -key server.key -sha256 -out server.csr
```

3. Send the CSR file to be signed by the certification authority.
4. Install the signed certificate in the SD-AVC Dashboard. See "Serviceability Page" in [Using SD-AVC](#).

# Using a Self-signed SSL Certificate Created with Keytool

You can use the **keytool** command line utility to create a self-signed certificate, and use the certificate with Cisco SD-AVC.

This utility creates certificates in [Java KeyStore](#) (JKS) format.

The example shows how to create a self-signed certificate and how to display the details of the certificate. Details such as alias are required when configuring SD-AVC to use the certificate.



**Note** Keytool is not a Cisco product. The brief guidelines provided here are for convenience. Complete information is available online.

### Creating and Installing the SSL Certificate

This example shows the command, followed by interactive input. It creates a certificate with:

- **Alias:** abc\_ssl
- **Passphrase:** 123456

#### 1. Create certificate keys.

```
keytool -genkey -keyalg RSA -alias abc_ssl -keystore my_keystore.jks -storepass 123456
-validity 360 -keysize 2048
What is your first and last name?
[Unknown]: hostname.cisco.com
What is the name of your organizational unit?
[Unknown]: dev
What is the name of your organization?
[Unknown]: cisco
What is the name of your City or Locality?
[Unknown]: san-jose
What is the name of your State or Province?
[Unknown]: ca
What is the two-letter country code for this unit?
[Unknown]: us
Is CN=hostname.cisco.com, OU=dev, O=cisco, L=san-jose, ST=ca, C=us correct? (type "yes"
or "no")
[no]: yes

Enter key password for <abc_ssl>:
(RETURN if same as keystore password):
```

#### 2. Install the signed certificate in the SD-AVC Dashboard. See "Serviceability Page" in [Using SD-AVC](#).

### Viewing the Certificate Details

View the certificate details. Note that the output includes the alias name (which may be a default value, or a specified custom alias name, as in this example), and keystore type (jks in this example).

```
1. keytool -list -v -keystore my_keystore.jks
Enter keystore password:

Keystore type: jks
Keystore provider: IBMJCE

Your keystore contains 1 entry

Alias name: abc_ssl
Creation date: Apr 30, 2019
Entry type: keyEntry
Certificate chain length: 1
Certificate[1]:
Owner: CN=hostname.cisco.com, OU=dev, O=cisco, L=san-jose, ST=ca, C=us
Issuer: CN=hostname.cisco.com, OU=dev, O=cisco, L=san-jose, ST=ca, C=us
Serial number: 5cc899de
Valid from: 4/30/19 9:54 PM until: 4/24/20 9:54 PM
```

```

Certificate fingerprints:
    MD5:  38:B7:B4:28:43:48:11:88:C5:B1:E0:47:79:26:CD:A7
    SHA1: 7C:60:01:35:26:67:40:64:65:D0:E2:B5:2B:30:1F:7D:5E:16:44:C3
    SHA256:
42:82:63:BF:CF:87:95:B7:5A:FA:38:12:45:F9:88:D5:FD:00:68:A8:96:28:63:32:0C:D4:E5:A0:86:68:25:53

    Signature algorithm name: SHA256withRSA
    Version: 3

```

## Using a Self-signed SSL Certificate Created with OpenSSL

You can use the **OpenSSL** command line utility to create a self-signed certificate, and use the certificate with Cisco SD-AVC.

This utility creates certificates in numerous formats.

The example shows how to create a certificate and how to display the details of the certificate. Details such as alias/friendlyName, are required when configuring SD-AVC to use the certificate.




---

**Note** OpenSSL is not a Cisco product. The brief guidelines provided here are for convenience. Complete information is available online.

---

### Creating and Installing the SSL Certificate

This example shows the command, followed by interactive input. It creates and exports a certificate with:

- **Alias/friendlyName:** abc\_ssl
- **Output filename:** my\_cakey.pem

#### 1. Create certificate keys.

```

openssl req -newkey rsa:2048 -x509 -keyout my_cakey.pem -out my_cacert.pem -days 3650
Generating a 2048 bit RSA private key
.....+++
...+++
writing new private key to 'my_cakey.pem'
Enter PEM pass phrase:
Verifying - Enter PEM pass phrase:
-----
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.
-----
Country Name (2 letter code) [AU]:us
State or Province Name (full name) [Some-State]:
Locality Name (eg, city) []:city
Organization Name (eg, company) [Internet Widgits Pty Ltd]:
Organizational Unit Name (eg, section) []:
Common Name (e.g. server FQDN or YOUR name) []:hostname.cisco.com
Email Address []:anyEmail@cisco.com

openssl pkcs12 -export -in my_cacert.pem -inkey my_cakey.pem -out my_identity.p12 -name

```

```
"abc_ssl"  
Enter pass phrase for my_cakey.pem:  
Enter Export Password:  
Verifying - Enter Export Password:
```

2. Convert the format.

```
openssl pkcs12 -export -in my_cacert.pem -inkey my_cakey.pem -out my_identity.p12 -name  
"abc_ssl"  
Enter pass phrase for my_cakey.pem:  
Enter Export Password:  
Verifying - Enter Export Password:
```

3. Install the signed certificate in the SD-AVC Dashboard. See "Serviceability Page" in [Using SD-AVC](#).

### Viewing the Certificate Details

View the certificate details. Note that this command provides the alias/friendlyName, which may be a default value, or a specified custom alias name, as in this example.

```
1. openssl pkcs12 -info -in my_identity.p12  
Enter Import Password:  
MAC Iteration 2048  
MAC verified OK  
PKCS7 Encrypted data: pbeWithSHA1And40BitRC2-CBC, Iteration 2048  
Certificate bag  
Bag Attributes  
    localKeyID: 2E 12 BE F7 56 D3 1D C0 39 9A 52 29 AD 18 3A 95 05 AA A5 86  
    friendlyName: abc_ssl
```

