AVI
Networks®

Certificate Management Integration for CSR Automation

Avi Technical Reference (v17.2)

Certificate Management Integration for CSR Automation

view online

Avi Vantage supports automation of the process for requesting and installing a certificate signed by a certificate authority (CA). This feature handles initial certificate registration as well as renewal of certificates based on certificate expiration.

To accomplish this, a Templates > Security > Certificate Management Profile object is used. Creating an instance of this object, an individual certificate management profile, provides a way to configure a path to a certificate script, along with the set of parameters the script needs (CSR, Common Name, and others) to integrate with a certificate management service within the customer's internal network. The script itself is left opaque by design to accommodate the various certificate management services different customers may have.

As a part of the SSL certificate configuration, the Avi Vantage user needs only to select CSR, fill in the necessary fields for the certificate, and select the certificate management profile to which this certificate is bound. The Avi Controller will then use the CSR and the script to obtain the certificate and also renew the certificate upon expiration. As a part of the renewal process, a new key pair is generated and a certificate corresponding to this is obtained from the certificate management service.

Without the addition of this automation, the process for sending the CSR to the external CA, then installing the signed certificate and keys, must be performed by the Avi Vantage user.

Note: Release 16.2 only supported use of Python scripts for this feature, as illustrated in the sample shown below. With release 16.2.2, automated CSR workflow for SafeNet HSM is supported.

Configuring Certificate Management Integration

To configure certificate management integration:

1. Prepare a Python script that defines a certificate_request() method. The method must accept the following input as a dictionary:
    - CSR
    - Hostname for the Common Name field
    - Parameters defined in the certificate management profile
2. Create a certificate management profile that calls the script.

Prepare the Script

The script must use the `def certificate_request` command. The following example could be adapted:

```
def certificate_request(csr, common_name, args_dict):
    """
    Check if a token exists that can be used:
    If not, authenticate against the service with the provided credentials.
    Invoke the certificate request and get back a valid certificate.
    Inputs:
      @csr : Certificate signing request string. This is a multi-line string output like what
             you get from openssl.
```

```
        @common_name: Common name of the subject.
        @args_dict: Dictionary of the key value pairs from the certificate management profile.
      """
```

The specific parameter values to be passed to the script are specified within the certificate management profile.

Sensitive Parameters Are Hidden

For parameters that are sensitive (for example, passwords), the values can be hidden. Marking a parameter sensitive prevents its value from being displayed in the web interface or being passed by the API.

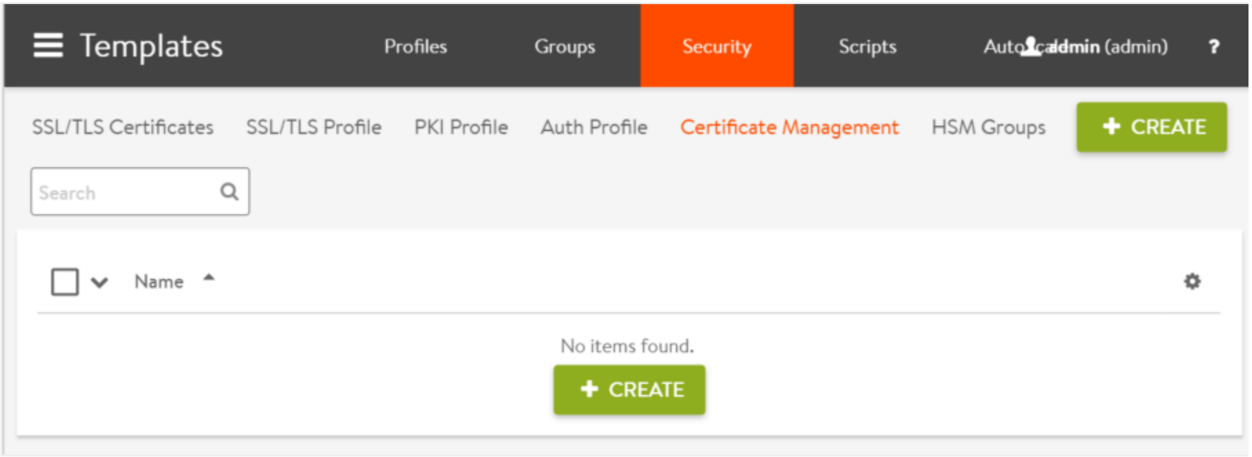
Dynamic Parameter Values Are Assigned During CSR Creation

The value for a certificate management parameter can be assigned within the profile or within individual CSRs.

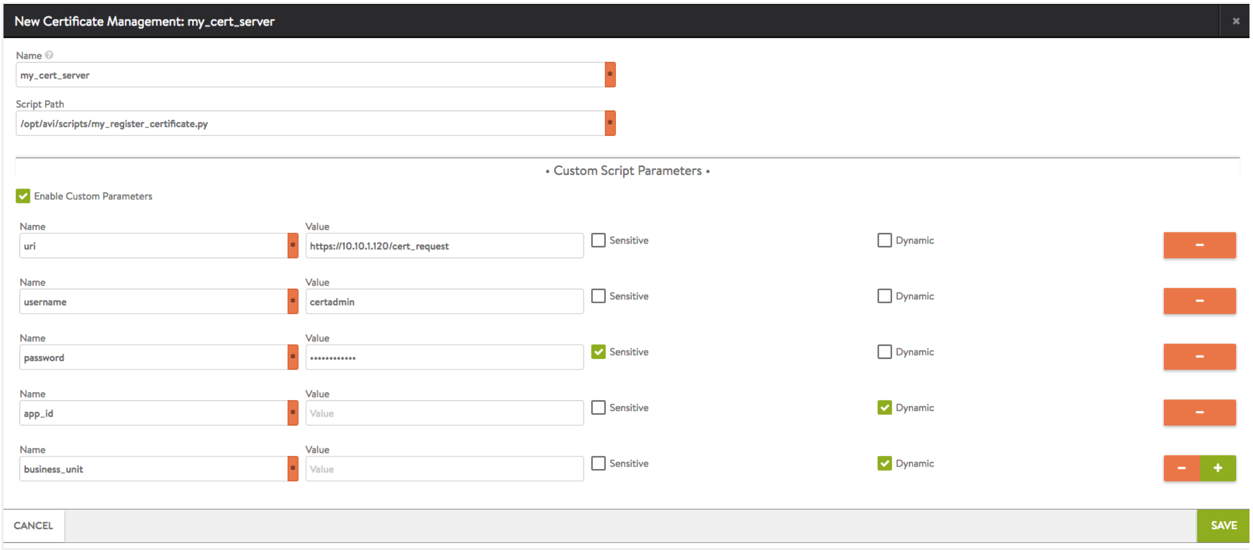
- If the parameter value is assigned within the profile, the value applies to all CSRs generated using this profile.
- To dynamically assign a parameter's value, indicate within the certificate management profile that the parameter is dynamic. This leaves the parameter's value unassigned. In this case, the dynamic parameter's value instead is assigned when creating an individual CSR using the profile. The parameter value apples only to that CSR.

Create the Certificate Management Profile

1. Navigate to Templates > Certificates > Security Management, and click Create.



2. Enter a name for the profile.
3. Enter the location (URL) of the script file.
4. If the profile will need to pass some parameter values to the script, select (check) Enable Custom Parameters, and enter their names and values.



New Certificate Management: my_cert_server

Name: my_cert_server

Script Path: /opt/avi/scripts/my_register_certificate.py

• Custom Script Parameters •

☑ Enable Custom Parameters

Name	Value		
uri	https://10.10.1.120/cert_request	□ Sensitive	□ Dynamic
username	certadmin	□ Sensitive	□ Dynamic
password	••••••••••••	☑ Sensitive	□ Dynamic
app_id	Value	□ Sensitive	☑ Dynamic
business_unit	Value	□ Sensitive	☑ Dynamic

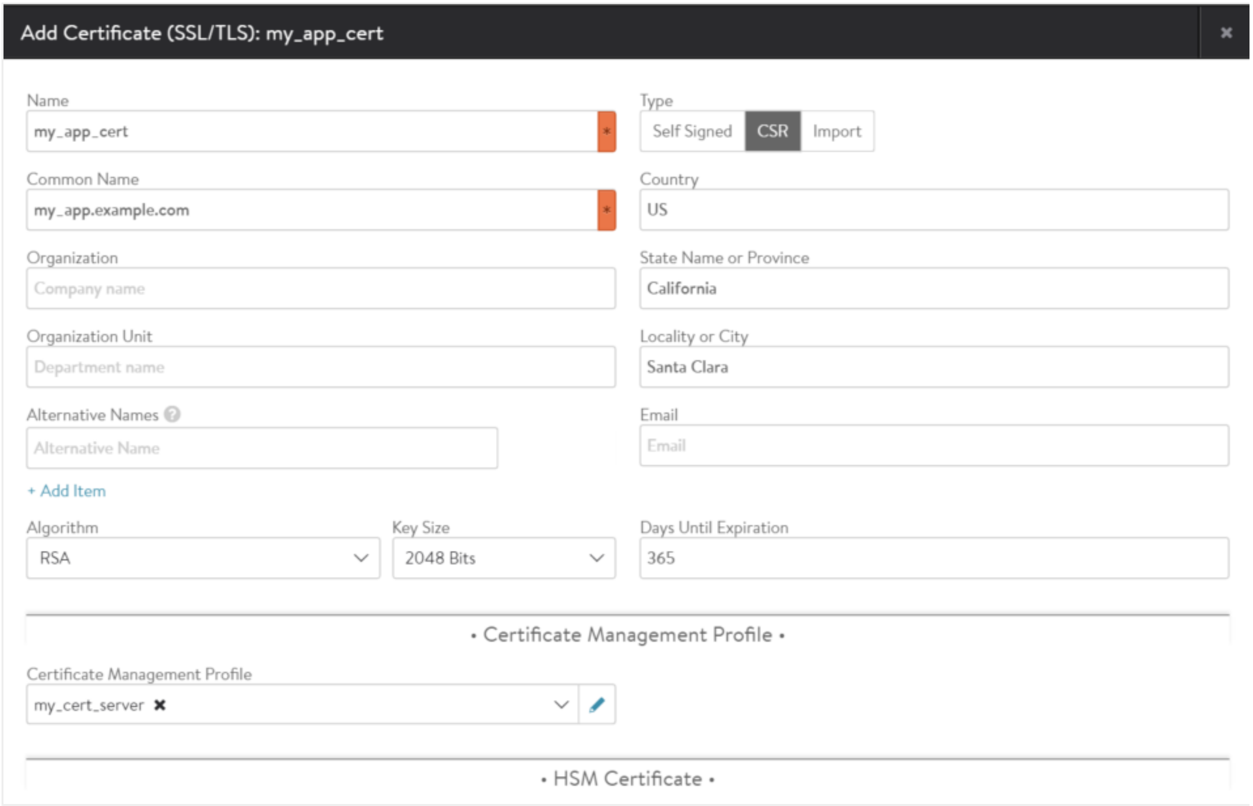
In this example, the location (URL) of the CA service and the login credentials for the service, will be passed to the script.For parameters that are sensitive (for example, passwords), select the Sensitive checkbox. Marking a parameter sensitive prevents its value from being displayed in the web interface or being passed by the API.For parameters that are to be dynamically assigned during CSR creation, select the Dynamic checkbox. This leaves the parameter unassigned within the profile.

5. Click Save.

Use the Certificate Management Profile To Get Signed Certificates

After adding the script and creating the certificate management profile, the profile can be used to easily obtain and install CA-signed certificates.

1. Navigate to Templates > Certificates > Security Management, and click Create.
2. Click CSR.
3. In the Certificate Management Profile section, select the profile configured in the previous section from the pull-down menu.



Add Certificate (SSL/TLS): my_app_cert

Name: my_app_cert

Type: Self Signed | CSR | Import

Common Name: my_app.example.com

Country: US

Organization: Company name

State Name or Province: California

Organization Unit: Department name

Locality or City: Santa Clara

Alternative Names: Alternative Name

Email: Email

+ Add Item

Algorithm: RSA

Key Size: 2048 Bits

Days Until Expiration: 365

• Certificate Management Profile •

Certificate Management Profile: my_cert_server

• HSM Certificate •

4. Click on Generate.

The Avi Controller generates a key pair and CSR, executes the script to request the CA-signed certificate from the Avi Vantage PKI service, and saves the signed certificate in persistent storage.

Automatic Certificate Renewal

Users may choose to customize when certificate expiry notifications are sent; refer to the Customizing Notification of Certificate Expiration article. If the certificate management profile is configured for a certificate, a renewal is attempted in the last-but-one interval. By default, Avi Controller generates events 30 days, 7 days, and 1 day before expiry. In this setting, certificate renewal will be attempted 7 days before expiry.