# Prepare to install SnapCenter Custom Plug-ins

## SnapCenter Software 4.6

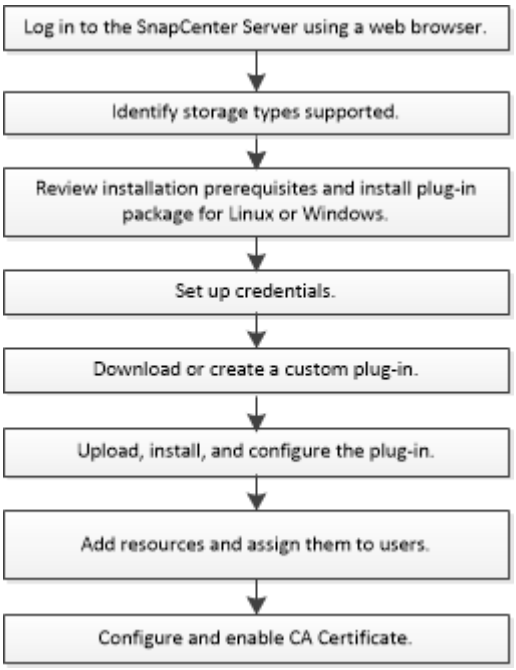
NetApp
November 24, 2022

# Table of Contents

# Prepare to install SnapCenter Custom Plug-ins

## Installation workflow of SnapCenter Custom Plug-ins

You should install and set up SnapCenter Custom Plug-ins if you want to protect custom plug-in resources.




Develop a plug-in for your application

## Prerequisites for adding hosts and installing SnapCenter Custom Plug-ins

Before you add a host and install the plug-ins packages, you must complete all the requirements. The Custom Plug-ins can be used in both Windows and Linux environments.

- You must have created a custom plug-in. For details, see the developer information.

  Develop a plug-in for your application

- If you want to manage MySQL or DB2 applications, you must have downloaded the MySQL and DB2 Custom Plug-ins that are provided by NetApp.
- You must have installed Java 1.8, 64-bit on your Linux or Windows host.
- When installing a plug-in on a Windows host, if you specify a credential that is not built-in, or if the user belongs to a local workgroup user, you must disable UAC on the host.
- The Custom Plug-ins must be available on the client host from where the add host operation is performed.

### General

If you are using iSCSI, the iSCSI service must be running.

### Windows hosts

- You must have a domain user with local administrator privileges with local login permissions on the remote host.
- If you manage cluster nodes in SnapCenter, you must have a user with administrative privileges to all the nodes in the cluster.

### Linux hosts

- You must have enabled the password-based SSH connection for the root or non-root user.
- You must have installed Java 1.8 64-bit, on your Linux host.

  If you are using Windows 2019 or Windows 2016 for the SnapCenter Server host, you must install Java 1.8, 64-bit. The Interoperability Matrix Tool (IMT) contains the latest information about requirements.

  Java Downloads for All Operating Systems

  NetApp Interoperability Matrix Tool

- You must configure sudo privileges for the non-root user to provide access to several paths. Add the following lines to the /etc/sudoers file by using the visudo Linux utility. For example,

  ```
  Cmnd_Alias SCCMD = /opt/NetApp/snapcenter/scc/bin/scc <non_root_user>
  ALL=(ALL) NOPASSWD:SETENV: SCCMD
  ```

  non_root_user is the name of the non-root user that you created.

# Host requirements to install SnapCenter Plug-ins Package for Windows

Before you install the SnapCenter Plug-ins Package for Windows, you should be familiar with some basic host system space requirements and sizing requirements.

| Item | Requirements |
| --- | --- |
| Operating Systems | Microsoft Windows<br><br>For the latest information about supported versions, see the NetApp Interoperability Matrix Tool. |
| Minimum RAM for the SnapCenter plug-in on host | 1 GB |

| Item | Requirements |
| --- | --- |
| Minimum install and log space for the SnapCenter plug-in on host | 5 GB<br>You should allocate sufficient disk space and monitor the storage consumption by the logs folder. The log space required varies depending on the number of the entities to be protected and the frequency of data protection operations. If there is no sufficient disk space, the logs will not be created for the recently run operations. |
| Required software packages | • Microsoft .NET Framework 4.5.2 or later<br>• Windows Management Framework (WMF) 4.0 or later<br>• PowerShell 4.0 or later<br>For the latest information about supported versions, see the NetApp Interoperability Matrix Tool. |

# Host requirements for installing the SnapCenter Plug-ins Package for Linux

You should ensure that the host meets the requirements before installing the SnapCenter Plug-ins Package for Linux.

| Item | Requirements |
| --- | --- |
| Operating systems | • Red Hat Enterprise Linux<br>• Oracle Linux<br>• SUSE Linux Enterprise Server (SLES) |
| Minimum RAM for the SnapCenter plug-in on host | 1 GB |

| Item | Requirements |
| --- | --- |
| Minimum install and log space for the SnapCenter plug-in on host | 2 GB<br><br>You should allocate sufficient disk space and monitor the storage consumption by the logs folder. The log space required varies depending on the number of the entities to be protected and the frequency of data protection operations. If there is no sufficient disk space, the logs will not be created for the recently run operations. |
| Required software packages | Java 1.8 (64-bit)Oracle Java or OpenJDK flavors<br><br>If you have upgraded JAVA to the latest version, you must ensure that the JAVA_HOME option located at /var/opt/snapcenter/spl/etc/spl.properties is set to the correct JAVA version and the correct path. |

For the latest information about supported versions, see the NetApp Interoperability Matrix Tool

# Set up credentials for SnapCenter Custom Plug-ins

SnapCenter uses credentials to authenticate users for SnapCenter operations. You should create credentials for installing SnapCenter plug-ins and additional credentials for performing data protection operations on databases or Windows file systems.

**What you will need**

- Linux hosts

  You must set up credentials for installing plug-ins on Linux hosts.

  You must set up the credentials for the root user or for a non-root user who has sudo privileges to install and start the plug-in process.

  **Best Practice:** Although you are allowed to create credentials for Linux after deploying hosts and installing plug-ins, the best practice is to create credentials after you add SVMs, before you deploy hosts and install plug-ins.

- Windows hosts

  You must set up Windows credentials before installing plug-ins.

  You must set up the credentials with administrator privileges, including administrator rights on the remote host.

- Custom Plug-ins applications

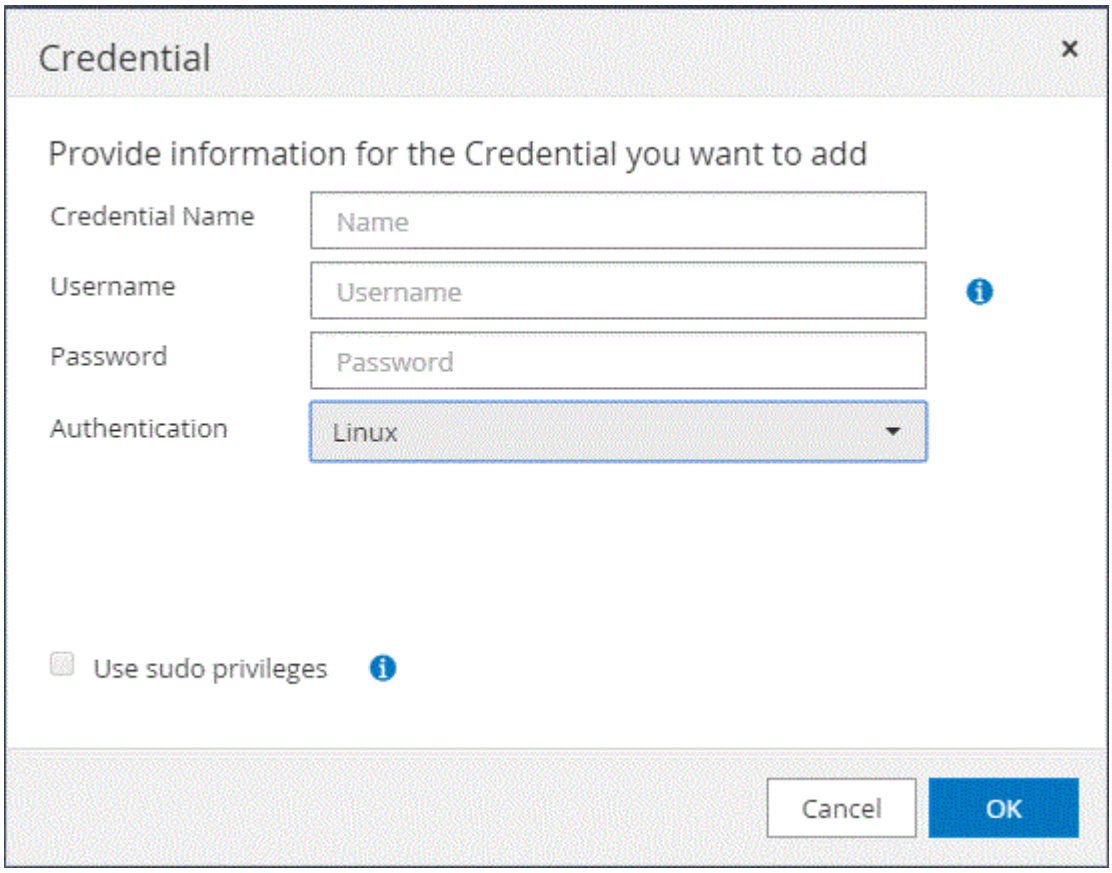
The plug-in uses the credentials that are selected or created while adding a resource. If a resource does not require credentials during data protection operations, you can set the credentials as **None**.

### About this task

If you set up credentials for individual resource groups and the username does not have full admin privileges, you must assign at least the resource group and backup privileges to the username.

### Steps

1. In the left navigation pane, click **Settings**.
2. In the Settings page, click **Credential**.
3. Click **New**.




4. In the Credential page, specify the information required for configuring credentials:

| For this field… | Do this… |
| --- | --- |
| Credential name | Enter a name for the credentials. |

| For this field… | Do this… |
| --- | --- |
| User name | Enter the user name and password that are to be used for authentication.<br>• Domain administrator or any member of the administrator group<br>Specify the domain administrator or any member of the administrator group on the system on which you are installing the SnapCenter plug-in. Valid formats for the Username field are:<br>◦ *NetBIOS\UserName*<br>◦ *Domain FQDN\UserName*<br>• Local administrator (for workgroups only)<br>For systems that belong to a workgroup, specify the built-in local administrator on the system on which you are installing the SnapCenter plug-in. You can specify a local user account that belongs to the local administrators group if the user account has elevated privileges or the User Access control feature is disabled on the host system. The valid format for the Username field is: *UserName* |
| Password | Enter the password used for authentication. |
| Authentication Mode | Select the authentication mode that you want to use. |
| Use sudo privileges | Select the **Use sudo privileges** check box if you are creating credentials for a non-root user.<br>Applicable to Linux users only. |

5. Click **OK**.

After you finish setting up the credentials, you might want to assign credential maintenance to a user or group of users on the User and Access page.

# Configure gMSA on Windows Server 2012 or later

Windows Server 2012 or later enables you to create a group Managed Service Account (gMSA) that provides automated service account password management from a managed domain account.

**What you will need**

- You should have a Windows Server 2012 or later domain controller.
- You should have a Windows Server 2012 or later host, which is a member of the domain.

**Steps**

1. Create a KDS root key to generate unique passwords for each object in your gMSA.
2. For each domain, run the following command from the Windows domain controller: Add-KDSRootKey -EffectiveImmediately
3. Create and configure your gMSA:
   a. Create a user group account in the following format:

      ```
      domainName\accountName$
      ```

   b. Add computer objects to the group.
   c. Use the user group you just created to create the gMSA.

      For example,

      ```
      New-ADServiceAccount -name <ServiceAccountName> -DNSHostName <fqdn>
      -PrincipalsAllowedToRetrieveManagedPassword <group>
      -ServicePrincipalNames <SPN1,SPN2,…>
      ```

   d. Run `Get-ADServiceAccount` command to verify the service account.
4. Configure the gMSA on your hosts:
   a. Enable the Active Directory module for Windows PowerShell on the host where you want to use the gMSA account.

      To do this, run the following command from PowerShell:

```
PS C:\> Get-WindowsFeature AD-Domain-Services

Display Name                            Name                Install
State
------------                            ----
-------------
[ ] Active Directory Domain Services   AD-Domain-Services  Available


PS C:\> Install-WindowsFeature AD-DOMAIN-SERVICES

Success Restart Needed Exit Code      Feature Result
------- -------------- ---------      --------------
True    No             Success        {Active Directory Domain
Services, Active ...
WARNING: Windows automatic updating is not enabled. To ensure that
your newly-installed role or feature is
automatically updated, turn on Windows Update.
```

b. Restart your host.

c. Install the gMSA on your host by running the following command from the PowerShell command prompt: `Install-AdServiceAccount <gMSA>`

d. Verify your gMSA account by running the following command: `Test-AdServiceAccount <gMSA>`

5. Assign the administrative privileges to the configured gMSA on the host.
6. Add the Windows host by specifying the configured gMSA account in the SnapCenter Server.

   SnapCenter Server will install the selected plug-ins on the host and the specified gMSA will be used as the service log on account during the plug-in installation.

# Install the SnapCenter Custom Plug-ins

## Add hosts and install plug-in packages on remote hosts

You must use the SnapCenterAdd Host page to add hosts, and then install the plug-in packages. The plug-ins are automatically installed on the remote hosts. You can add a host and install the plug-in packages either for an individual host or for a cluster.

### What you will need

- You must be a user that is assigned to a role that has the plug-in install and uninstall permissions, such as the SnapCenter Admin role.
- You should ensure that the message queueing service is running.
- If you are using group Managed Service Account (gMSA), you should configure gMSA with administrative privileges.

Configure group Managed Service Account on Windows Server 2012 or later for custom applications

**About this task**

You cannot add a SnapCenter Server as a plug-in host to another SnapCenter Server.

If you install plug-ins on a cluster (WSFC), the plug-ins are installed on all of the nodes of the cluster.

**Steps**

1. In the left navigation pane, click **Hosts**.
2. Verify that the **Managed Hosts** tab is selected at the top.
3. Click **Add**.
4. In the Hosts page, perform the following actions:

| For this field… | Do this… |
| --- | --- |
| Host Type | Select the host type:<br>• Windows<br>• Linux<br>The custom plug-ins can be used in both Windows and Linux environments. |
| Host name | Enter the fully qualified domain name (FQDN) or the IP address of the host.<br>SnapCenter depends on the proper configuration of the DNS. Therefore, the best practice is to enter the FQDN.<br>For Windows environments, the IP address is supported for untrusted domain hosts only if it resolves to the FQDN.<br>You can enter the IP addresses or FQDN of a stand-alone host.<br>If you are adding a host by using SnapCenter and the host is part of a subdomain, you must provide the FQDN. |

| For this field… | Do this… |
| --- | --- |
| Credentials | Either select the credential name that you created, or create new credentials.<br><br>The credentials must have administrative rights on the remote host. For details, see the information about creating credentials.<br><br>You can view details about the credentials by positioning your cursor over the credential name that you specified.<br><br>The credentials authentication mode is determined by the host type that you specify in the Add Host wizard. |

5. In the **Select Plug-ins to Install** section, select the plug-ins to install.
6. (Optional) Click **More Options**.

| For this field… | Do this… |
| --- | --- |
| Port | Either retain the default port number, or specify the port number.<br><br>The default port number is 8145. If the SnapCenter Server was installed on a custom port, that port number will be displayed as the default port.<br><br>If you manually installed the plug-ins and specified a custom port, you must specify the same port. Otherwise, the operation fails. |

| For this field… | Do this… |
| --- | --- |
| Installation Path | The custom plug-ins can be installed on either a Windows system or a Linux system.<br><br>• For the SnapCenter Plug-ins Package for Windows, the default path is C:\Program Files\NetApp\SnapCenter.<br><br>Optionally, you can customize the path.<br><br>• For SnapCenter Plug-ins Package for Linux, the default path is /opt/NetApp/snapcenter.<br><br>Optionally, you can customize the path.<br><br>• For the SnapCenter Custom Plug-ins:<br><br>i. In the Custom Plug-ins section, click **Browse**, and select the zipped custom plug-in folder.<br><br>The zipped folder contains the custom plug-in code and the descriptor .xml file.<br><br>For Storage Plug-in, navigate to *C:\ProgramData\NetApp\SnapCenter\Package Repository* and select `Storage.zip` folder.<br><br>ii. Click **Upload**.<br><br>The descriptor .xml file in the zipped custom plug-in folder is validated before the package is uploaded.<br><br>The custom plug-ins that are uploaded to the SnapCenter Server are listed.<br><br>If you want to manage MySQL or DB2 applications, you can use the MySQL and DB2 custom plug-ins that are provided by NetApp. The MySQL and DB2 custom plug-ins are available at the NetApp Automation Store |
| Skip preinstall checks | Select this check box if you already installed the plug-ins manually and you do not want to validate whether the host meets the requirements for installing the plug-in. |

| For this field… | Do this… |
| --- | --- |
| Use group Managed Service Account (gMSA) to run the plug-in services | For Windows host, select this check box if you want to use group Managed Service Account (gMSA) to run the plug-in services.<br>Provide the gMSA name in the following format: domainName\accountName$.<br>gMSA will be used as a log on service account only for SnapCenter Plug-in for Windows service. |

7. Click **Submit**.

   If you have not selected the **Skip prechecks** checkbox, the host is validated to verify whether the host meets the requirements for installing the plug-in. The disk space, RAM, PowerShell version, .NET version, location (for Windows plug-ins), and Java version (for Linux plug-ins) are validated against the minimum requirements. If the minimum requirements are not met, appropriate error or warning messages are displayed.

   If the error is related to disk space or RAM, you can update the web.config file located at C:\Program Files\NetApp\SnapCenter WebApp to modify the default values. If the error is related to other parameters, you must fix the issue.

   In an HA setup, if you are updating web.config file, you must update the file on both nodes.

8. If host type is Linux, verify the fingerprint, and then click **Confirm and Submit**.

   Fingerprint verification is mandatory even if the same host was added earlier to SnapCenter and the fingerprint was confirmed.

9. Monitor the installation progress.

   The installation-specific log files are located at /custom_location/snapcenter/logs.

## Install SnapCenter Plug-in Packages for Linux or Windows on multiple remote hosts by using cmdlets

You can install the SnapCenter Plug-in Packages for Linux or Windows on multiple hosts simultaneously by using the Install-SmHostPackage PowerShell cmdlet.

### What you will need

The user adding a host should have the administrative rights on the host.

### Steps

1. Launch PowerShell.

2. On the SnapCenter Server host, establish a session using the Open-SmConnection cmdlet, and then enter your credentials.
3. Install the plug-in on multiple hosts using the Install-SmHostPackage cmdlet and the required parameters.

   The information regarding the parameters that can be used with the cmdlet and their descriptions can be obtained by running *Get-Help command_name*. Alternatively, you can also refer to the SnapCenter Software Cmdlet Reference Guide.

   You can use the -skipprecheck option when you have installed the plug-ins manually and do not want to validate whether the host meets the requirements to install the plug-in.

4. Enter your credentials for remote installation.

## Install the SnapCenter Custom Plug-ins on Linux hosts by using the command-line interface

You should install the SnapCenter Custom Plug-ins by using the SnapCenter user interface (UI). If your environment does not allow remote installation of the plug-in from the SnapCenter UI, you can install the custom plug-ins either in console mode or in silent mode by using the command-line interface (CLI).

### Steps

1. Copy the SnapCenter Plug-ins Package for Linux installation file (snapcenter_linux_host_plugin.bin) from C:\ProgramData\NetApp\SnapCenter\Package Repository to the host where you want to install the custom plug-ins.

   You can access this path from the host where the SnapCenter Server is installed.

2. From the command prompt, navigate to the directory where you copied the installation file.
3. Install the plug-in: `path_to_installation_bin_file/snapcenter_linux_host_plugin.bin -i silent -DPORT=port_number_for_host -DSERVER_IP=server_name_or_ip_address -DSERVER_HTTPS_PORT=port_number_for_server`

   - -DPORT specifies the SMCore HTTPS communication port.
   - -DSERVER_IP specifies the SnapCenter Server IP address.
   - -DSERVER_HTTPS_PORT specifies the SnapCenter Server HTTPS port.
   - -DUSER_INSTALL_DIR specifies the directory where you want to install the SnapCenter Plug-ins Package for Linux.
   - DINSTALL_LOG_NAME specifies the name of the log file.

   ```
   /tmp/sc-plugin-installer/snapcenter_linux_host_plugin.bin -i silent
   -DPORT=8145 -DSERVER_IP=scserver.domain.com -DSERVER_HTTPS_PORT=8146
   -DUSER_INSTALL_DIR=/opt
   -DINSTALL_LOG_NAME=SnapCenter_Linux_Host_Plugin_Install_2.log
   -DCHOSEN_FEATURE_LIST=CUSTOM
   ```

4. Add the host to the SnapCenter Server using the Add-Smhost cmdlet and the required parameters.

The information regarding the parameters that can be used with the command and their descriptions can be obtained by running *Get-Help command_name*. Alternatively, you can also refer to the SnapCenter Software Cmdlet Reference Guide.

5. Log in to SnapCenter and upload the custom plug-in from the UI or by using PowerShell cmdlets.

   You can upload the custom plug-in from the UI by referring to Add hosts and install plug-in packages on remote hosts section.

   The SnapCenter cmdlet help and the cmdlet reference information contain more information about PowerShell cmdlets.

   SnapCenter Software Cmdlet Reference Guide.

## Monitor the status of installing custom plug-ins

You can monitor the progress of SnapCenter plug-in package installation by using the Jobs page. You might want to check the progress of installation to determine when it is complete or if there is an issue.

**About this task**

The following icons appear on the Jobs page and indicate the state of the operation:

- In progress
- Completed successfully
- Failed
- Completed with warnings or could not start due to warnings
- Queued

**Steps**

1. In the left navigation pane, click **Monitor**.
2. In the Monitor page, click **Jobs**.
3. In the Jobs page, to filter the list so that only plug-in installation operations are listed, do the following:
   a. Click **Filter**.
   b. Optional: Specify the start and end date.
   c. From the Type drop-down menu, select **Plug-in installation**.
   d. From the Status drop-down menu, select the installation status.
   e. Click **Apply**.
4. Select the installation job and click **Details** to view the job details.
5. In the Job Details page, click **View logs**.

# Configure CA Certificate

## Generate CA Certificate CSR file

You can generate a Certificate Signing Request (CSR) and import the certificate that can be obtained from a Certificate Authority (CA) using the generated CSR. The certificate will have a private key associated with it.

CSR is a block of encoded text that is given to an authorized certificate vendor to procure the signed CA certificate.

For information to generate a CSR, see How to generate CA Certificate CSR file.

If you own the CA certificate for your domain (*.domain.company.com) or your system (machine1.domain.company.com), you can skip generating the CA Certificate CSR file. You can deploy the existing CA certificate with SnapCenter.

For cluster configurations, the cluster name (virtual cluster FQDN), and the respective host names should be mentioned in the CA certificate. The certificate can be updated by filling the Subject Alternative Name (SAN) field before procuring the certificate. For a wild card certificate (*.domain.company.com), the certificate will contain all the hostnames of the domain implicitly.

## Import CA certificates

You must import the CA certificates to the SnapCenter Server and the Windows host plug-ins using the Microsoft management console (MMC).

### Steps

1. Go to the Microsoft management console (MMC), and then click **File** > **Add/Remove Snapin**.
2. In the Add or Remove Snap-ins window, select **Certificates** and then click **Add**.
3. In the Certificates snap-in window, select the **Computer account** option, and then click **Finish**.
4. Click **Console Root** > **Certificates – Local Computer** > **Trusted Root Certification Authorities** > **Certificates**.
5. Right-click on the folder “Trusted Root Certification Authorities”, and then select **All Tasks** > **Import** to start the import wizard.
6. Complete the wizard, as follows:

| In this wizard window… | Do the following… |
|---|---|
| Import Private Key | Select the option **Yes**, import the private key, and then click **Next**. |
| Import File Format | Make no changes; click **Next**. |
| Security | Specify the new password to be used for the exported certificate, and then click **Next**. |
| Completing the Certificate Import Wizard | Review the summary, and then click **Finish** to start the import. |

Importing certificate should be bundled with the private key (supported formats are: *.pfx, *.p12, *.p7b).

7. Repeat Step 5 for the “Personal” folder.

## Get the CA certificate thumbprint

A certificate thumbprint is a hexadecimal string that identifies a certificate. A thumbprint is calculated from the content of the certificate using a thumbprint algorithm.

**Steps**

1. Perform the following on the GUI:
   a. Double-click the certificate.
   b. In the Certificate dialog box, click the **Details** tab.
   c. Scroll through the list of fields and click **Thumbprint**.
   d. Copy the hexadecimal characters from the box.
   e. Remove the spaces between the hexadecimal numbers.

      For example, if the thumbprint is: "a9 09 50 2d d8 2a e4 14 33 e6 f8 38 86 b0 0d 42 77 a3 2a 7b", after removing the spaces, it will be: "a909502dd82ae41433e6f83886b00d4277a32a7b".

2. Perform the following from PowerShell:
   a. Run the following command to list the thumbprint of the installed certificate and identify the recently installed certificate by the subject name.

      *Get-ChildItem -Path Cert:\LocalMachine\My*

   b. Copy the thumbprint.

## Configure CA certificate with Windows host plug-in services

You should configure the CA certificate with Windows host plug-in services to activate the installed digital certificate.

Perform the following steps on the SnapCenter Server and all the plug-in hosts where CA certificates are already deployed.

**Steps**

1. Remove the existing certificate binding with SMCore default port 8145, by running the following command:

   `> netsh http delete sslcert ipport=0.0.0.0:_<SMCore Port>`

   For example:

   ```
   > netsh http delete sslcert ipport=0.0.0.0:8145
   ```

2. Bind the newly installed certificate with the Windows host plug-in services, by running the following

commands:

```
> $cert = "<certificate thumbprint>"

> $guid = [guid]::NewGuid().ToString("B")

> netsh http add sslcert ipport=0.0.0.0: <SMCore Port> certhash=$cert
appid="$guid"
```

For example:

```
> $cert = "a909502dd82ae41433e6f83886b00d4277a32a7b"
> $guid = [guid]::NewGuid().ToString("B")
> netsh http add sslcert ipport=0.0.0.0:8145 certhash=$cert
appid="$guid"
```

## Configure the CA Certificate for the SnapCenter Custom Plug-ins service on Linux host

You should manage the password of the custom plug-ins keystore and its certificate, configure the CA certificate, configure root or intermediate certificates to the custom plug-ins trust-store, and configure CA signed key pair to custom plug-ins trust-store with SnapCenter Custom Plug-ins service to activate the installed digital certificate.

Custom plug-ins uses the file 'keystore.jks', which is located at */opt/NetApp/snapcenter/scc/etc* both as its trust-store and key-store.

### Manage password for custom plug-in keystore and alias of the CA signed key pair in use

**Steps**

1. You can retrieve custom plug-in keystore default password from custom plug-in agent property file.

   It is the value corresponding to the key 'KEYSTORE_PASS'.

2. Change the keystore password:

   ```
   keytool -storepasswd -keystore keystore.jks
   ```

3. Change the password for all aliases of private key entries in the keystore to the same password used for the keystore:

   ```
   keytool -keypasswd -alias "alias_name_in_cert" -keystore keystore.jks
   ```

   Update the same for the key KEYSTORE_PASS in *agent.properties* file.

4. Restart the service after changing the password.

Password for custom plug-in keystore and for all the associated alias password of the private key should be same.

### Configure root or intermediate certificates to custom plug-in trust-store

You should configure the root or intermediate certificates without the private key to custom plug-in trust-store.

**Steps**

1. Navigate to the folder containing the custom plug-in keystore: /opt/NetApp/snapcenter/scc/etc.
2. Locate the file 'keystore.jks'.
3. List the added certificates in the keystore:

   ```
   keytool -list -v -keystore keystore.jks
   ```

4. Add a root or intermediate certificate:

   ```
   keytool -import -trustcacerts -alias myRootCA -file
   /root/USERTrustRSA_Root.cer -keystore keystore.jks
   ```

5. Restart the service after configuring the root or intermediate certificates to custom plug-in trust-store.

You should add the root CA certificate and then the intermediate CA certificates.

### Configure CA signed key pair to custom plug-in trust-store

You should configure the CA signed key pair to the custom plug-in trust-store.

**Steps**

1. Navigate to the folder containing the custom plug-in keystore /opt/NetApp/snapcenter/scc/etc.
2. Locate the file 'keystore.jks'.
3. List the added certificates in the keystore:

   ```
   keytool -list -v -keystore keystore.jks
   ```

4. Add the CA certificate having both private and public key.

   ```
   keytool -importkeystore -srckeystore /root/snapcenter.ssl.test.netapp.com.pfx
   -srcstoretype pkcs12 -destkeystore keystore.jks -deststoretype JKS
   ```

5. List the added certificates in the keystore.

   ```
   keytool -list -v -keystore keystore.jks
   ```

6. Verify that the keystore contains the alias corresponding to the new CA certificate, which was added to the keystore.
7. Change the added private key password for CA certificate to the keystore password.

Default custom plug-in keystore password is the value of the key KEYSTORE_PASS in agent.properties file.

```
keytool -keypasswd -alias "alias_name_in_CA_cert" -keystore keystore.jks
```

8. If the alias name in the CA certificate is long and contains space or special characters ("*",","), change the alias name to a simple name:

```
keytool -changealias -alias "long_alias_name" -destalias "simple_alias"
-keystore keystore.jks
```

9. Configure the alias name from CA certificate in agent.properties file.

   Update this value against the key SCC_CERTIFICATE_ALIAS.

10. Restart the service after configuring the CA signed key pair to custom plug-in trust-store.

#### Configure certificate revocation list (CRL) for SnapCenter Custom Plug-ins

#### About this task

- SnapCenter Custom Plug-ins will search for the CRL files in a pre-configured directory.
- Default directory for the CRL files for SnapCenter Custom Plug-ins is ' opt/NetApp/snapcenter/scc/etc/crl'.

#### Steps

1. You can modify and update the default directory in agent.properties file against the key CRL_PATH.

   You can place more than one CRL file in this directory. The incoming certificates will be verified against each CRL.

## Configure the CA Certificate for the SnapCenter Custom Plug-ins service on Windows host

You should manage the password of the custom plug-ins keystore and its certificate, configure the CA certificate, configure root or intermediate certificates to the custom plug-ins trust-store, and configure CA signed key pair to custom plug-ins trust-store with SnapCenter Custom Plug-ins service to activate the installed digital certificate.

Custom plug-ins uses the file *keystore.jks*, which is located at *C:\Program Files\NetApp\SnapCenter\Snapcenter Plug-in Creator\etc* both as its trust-store and key-store.

#### Manage password for custom plug-in keystore and alias of the CA signed key pair in use

#### Steps

1. You can retrieve custom plug-in keystore default password from custom plug-in agent property file.

   It is the value corresponding to the key *KEYSTORE_PASS*.

2. Change the keystore password:

   *keytool -storepasswd -keystore keystore.jks*

   If the "keytool" command is not recognized on the Windows command prompt, replace the keytool command with its complete path.

   *C:\Program Files\Java\<jdk_version>\bin\keytool.exe" -storepasswd -keystore keystore.jks*

3. Change the password for all aliases of private key entries in the keystore to the same password used for the keystore:

   *keytool -keypasswd -alias "alias_name_in_cert" -keystore keystore.jks*

   Update the same for the key KEYSTORE_PASS in *agent.properties* file.

4. Restart the service after changing the password.

   Password for custom plug-in keystore and for all the associated alias password of the private key should be same.

### Configure root or intermediate certificates to custom plug-in trust-store

You should configure the root or intermediate certificates without the private key to custom plug-in trust-store.

**Steps**

1. Navigate to the folder containing the custom plug-in keystore *C:\Program Files\NetApp\SnapCenter\Snapcenter Plug-in Creator\etc*
2. Locate the file 'keystore.jks'.
3. List the added certificates in the keystore:

   *keytool -list -v -keystore keystore.jks*

4. Add a root or intermediate certificate:

   *keytool -import -trustcacerts -alias myRootCA -file /root/USERTrustRSA_Root.cer -keystore keystore.jks*

5. Restart the service after configuring the root or intermediate certificates to custom plug-in trust-store.

You should add the root CA certificate and then the intermediate CA certificates.

### Configure CA signed key pair to custom plug-in trust-store

You should configure the CA signed key pair to the custom plug-in trust-store.

**Steps**

1. Navigate to the folder containing the custom plug-in keystore *C:\Program Files\NetApp\SnapCenter\Snapcenter Plug-in Creator\etc*
2. Locate the file *keystore.jks*.

3. List the added certificates in the keystore:

   *keytool -list -v -keystore keystore.jks*

4. Add the CA certificate having both private and public key.

   *keytool -importkeystore -srckeystore /root/snapcenter.ssl.test.netapp.com.pfx -srcstoretype pkcs12 -destkeystore keystore.jks -deststoretype JKS*

5. List the added certificates in the keystore.

   *keytool -list -v -keystore keystore.jks*

6. Verify that the keystore contains the alias corresponding to the new CA certificate, which was added to the keystore.
7. Change the added private key password for CA certificate to the keystore password.

   Default custom plug-in keystore password is the value of the key KEYSTORE_PASS in agent.properties file.

   *keytool -keypasswd -alias "alias_name_in_CA_cert" -keystore keystore.jks*

8. Configure the alias name from CA certificate in *agent.properties* file.

   Update this value against the key SCC_CERTIFICATE_ALIAS.

9. Restart the service after configuring the CA signed key pair to custom plug-in trust-store.

**Configure certificate revocation list (CRL) for SnapCenter Custom Plug-ins**

**About this task**

- To download the latest CRL file for the related CA certificate see How to update certificate revocation list file in SnapCenter CA Certificate.
- SnapCenter Custom Plug-ins will search for the CRL files in a pre-configured directory.
- Default directory for the CRL files for SnapCenter Custom Plug-ins is *'C:\Program Files\NetApp\SnapCenter\Snapcenter Plug-in Creator\ etc\crl'*.

**Steps**

1. You can modify and update the default directory in *agent.properties* file against the key CRL_PATH.
2. You can place more than one CRL file in this directory.

   The incoming certificates will be verified against each CRL.

## Enable CA Certificates for plug-ins

You should configure the CA certificates and deploy the CA certificates in the SnapCenter Server and the corresponding plug-in hosts. You should enable the CA certificate validation for the plug-ins.

**What you will need**

- You can enable or disable the CA certificates using the run *Set-SmCertificateSettings* cmdlet.

- You can display the certificate status for the plug-ins using the *Get-SmCertificateSettings*.

The information regarding the parameters that can be used with the cmdlet and their descriptions can be obtained by running *Get-Help command_name*. Alternatively, you can also refer to the SnapCenter Software Cmdlet Reference Guide.

**Steps**

1. In the left navigation pane, click **Hosts**.
2. In the Hosts page, click **Managed Hosts**.
3. Select single or multiple plug-in hosts.
4. Click **More options**.
5. Select **Enable Certificate Validation**.

**After you finish**

The Managed Hosts tab host displays a padlock and the color of the padlock indicates the status of the connection between SnapCenter Server and the plug-in host.

- indicates that the CA certificate is neither enabled nor assigned to the plug-in host.
- indicates that the CA certificate is successfully validated.
- indicates that the CA certificate could not be validated.
- indicates that the connection information could not be retrieved.

> When the status is yellow or green, the data protection operations completes successfully.

## Copyright information



## Trademark information