Protect Custom applications

SnapCenter Software 4.8

NetApp
September 26, 2025

Table of Contents

Protect Custom applications

SnapCenter Custom Plug-ins

SnapCenter Custom Plug-ins overview

You can develop custom plug-ins for applications that you use and then use SnapCenter to backup, restore, or clone these applications. Like other SnapCenter plug-ins, your custom plug-ins act as host-side components of the NetApp SnapCenter Software, enabling application-aware data protection and management of resources.

When Custom Plug-ins are installed, you can use SnapCenter with NetApp SnapMirror technology to create mirror copies of backup sets on another volume and use NetApp SnapVault technology to perform disk-to-disk backup replication. The Custom Plug-ins can be used in both Windows and Linux environments.

ⓘ SnapCenterCLI does not support SnapCenter Custom Plug-ins commands.

NetApp provides the Storage plug-in to perform data protection operations of the data volume on the ONTAP storage using the custom plug-in framework built into SnapCenter.

You can install the custom plug-in and storage plug-in from the Add Host page.

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

NetApp also provides MySQL, MAXDB, DB2, SYBASE, DPGLUE, MongoDB, ORASCPM, and PostgreSQL custom plug-ins.

ⓘ SnapCenter support policy will cover support for SnapCenter custom plug-in framework, core engine, and the associated APIs. Support will not cover the plug-in source code and the associated scripts built on the custom plug-in framework.

You can create your own custom plug-ins by referring to Develop a plug-in for your application.

What you can do with the SnapCenter Custom Plug-ins and Storage plug-in

You can use the SnapCenter Custom Plug-ins for data protection operations.

Custom plug-in

- Add resources such as databases, instances, documents, or tablespaces.
- Create backups.
- Restore from backups.
- Clone backups.
- Schedule backup operations.
- Monitor backup, restore, and clone operations.
- View reports for backup, restore, and clone operations.

Storage plug-in

You can use the storage plug-in for data protection operations.

- Take consistency group Snapshot copies of the storage volumes across ONTAP clusters.
- Backup custom applications using the built in pre and post scripting framework

  You can backup ONTAP volume, LUN, or a Qtree.

- Update Snapshot copies taken on the primary to an ONTAP secondary, leveraging the existing replication relationship (SnapVault/SnapMirror/unified replication) using SnapCenter policy

  ONTAP primary and secondary can be ONTAP FAS, AFF, Select, or Cloud ONTAP.

- Recover complete ONTAP volume, LUN, or files.

  You should provide the respective file path manually as the browse or indexing features are not built into the product.

  Qtree or directory restore is not supported but you can clone and export only the Qtree if the backup scope is defined at a Qtree level.

SnapCenter Custom Plug-ins features

SnapCenter integrates with the plug-in application and with NetApp technologies on the storage system. To work with Custom Plug-ins, you use the SnapCenter graphical user interface.

- **Unified graphical user interface**

  The SnapCenter interface provides standardization and consistency across plug-ins and environments. The SnapCenter interface enables you to complete consistent backup, restore, recovery, and clone operations across plug-ins, use centralized reporting, use at-a-glance dashboard views, set up role-based access control (RBAC), and monitor jobs across all plug-ins.

- **Automated central administration**

  You can schedule backup operations, configure policy-based backup retention, and perform restore operations. You can also proactively monitor your environment by configuring SnapCenter to send email alerts.

- **Nondisruptive NetApp Snapshot copy technology**

  SnapCenter uses NetApp Snapshot copy technology with the SnapCenter Custom Plug-ins to back up resources. Snapshot copies consume minimal storage space.

Using the Custom Plug-ins feature also offers the following benefits:

- Support for backup, restore, and clone workflows
- RBAC-supported security and centralized role delegation

  You can also set the credentials so that the authorized SnapCenter users have application-level permissions.

- Creation of space-efficient and point-in-time copies of resources for testing or data extraction by using NetApp FlexClone technology

  A FlexClone license is required on the storage system where you want to create the clone.

- Support for the consistency group (CG) Snapshot copy feature of ONTAP as part of creating backups.
- Capability to run multiple backups simultaneously across multiple resource hosts

  In a single operation, Snapshot copies are consolidated when resources in a single host share the same volume.

- Capability to create Snapshot copy using external commands.
- Capability to create file system consistent Snapshot copies in Windows environments.

Storage types supported by SnapCenter Custom Plug-ins

SnapCenter supports a wide range of storage types on both physical and virtual machines. You must verify the support for your storage type before installing SnapCenter Custom Plug-ins.

Machine	Storage type
Physical and NFS direct mounts on the VM hosts (VMDKs and RDM LUNs are not supported.)	FC-connected LUNs
Physical and NFS direct mounts on the VM hosts (VMDKs and RDM LUNs are not supported.)	iSCSI-connected LUNs
Physical and NFS direct mounts on the VM hosts (VMDKs and RDM LUNs are not supported.)	NFS-connected volumes

Minimum ONTAP privileges required for custom plug-in

The minimum ONTAP privileges that are required vary according to the SnapCenter plug-ins you are using for data protection.

All-access commands: Minimum privileges required for ONTAP 8.3.0 and later
event generate-autosupport-log
job history show job stop

All-access commands: Minimum privileges required for ONTAP 8.3.0 and later
lun attribute show lun create lun delete lun geometry lun igroup add lun igroup create lun igroup delete lun igroup rename lun igroup show lun mapping add-reporting-nodes lun mapping create lun mapping delete lun mapping remove-reporting-nodes lun mapping show lun modify lun move-in-volume lun offline lun online lun resize lun serial lun show
network interface

All-access commands: Minimum privileges required for ONTAP 8.3.0 and later
snapmirror policy add-rule snapmirror policy modify-rule snapmirror policy remove-rule snapmirror policy show snapmirror restore snapmirror show snapmirror show-history snapmirror update snapmirror update-ls-set snapmirror list-destinations
version

All-access commands: Minimum privileges required for ONTAP 8.3.0 and later
volume clone create volume clone show volume clone split start volume clone split stop volume create volume destroy volume file clone create volume file show-disk-usage volume offline volume online volume modify volume qtree create volume qtree delete volume qtree modify volume qtree show volume restrict volume show volume snapshot create volume snapshot delete volume snapshot modify volume snapshot rename volume snapshot restore volume snapshot restore-file volume snapshot show volume unmount

All-access commands: Minimum privileges required for ONTAP 8.3.0 and later
vserver cifs vserver cifs share create vserver cifs share delete vserver cifs shadowcopy show vserver cifs share show vserver cifs show vserver export-policy create vserver export-policy delete vserver export-policy rule create vserver export-policy rule show vserver export-policy show vserver iscsi connection show vserver show

Read-only commands: Minimum privileges required for ONTAP 8.3.0 and later
network interface

Prepare storage systems for SnapMirror and SnapVault replication for custom plug-ins

You can use a SnapCenter plug-in with ONTAP SnapMirror technology to create mirror copies of backup sets on another volume, and with ONTAP SnapVault technology to perform disk-to-disk backup replication for standards compliance and other governance-related purposes. Before you perform these tasks, you must configure a data-protection relationship between the source and destination volumes and initialize the relationship.

SnapCenter performs the updates to SnapMirror and SnapVault after it completes the Snapshot copy operation. SnapMirror and SnapVault updates are performed as part of the SnapCenter job; do not create a separate ONTAP schedule.

> ⓘ If you are coming to SnapCenter from a NetApp SnapManager product and are satisfied with the data protection relationships you have configured, you can skip this section.

A data protection relationship replicates data on primary storage (the source volume) to secondary storage (the destination volume). When you initialize the relationship, ONTAP transfers the data blocks referenced on the source volume to the destination volume.

ⓘ SnapCenter does not support cascade relationships between SnapMirror and SnapVault volumes (**Primary** > **Mirror** > **Vault**). You should use fanout relationships.

SnapCenter supports the management of version-flexible SnapMirror relationships. For details about version-flexible SnapMirror relationships and how to set them up, see the ONTAP documentation.

ⓘ SnapCenter does not support **sync_mirror** replication.

Define a backup strategy

Defining a backup strategy before you create your backup jobs ensures that you have the backups that you require to successfully restore or clone your resources. Your service-level agreement (SLA), recovery time objective (RTO), and recovery point objective (RPO) largely determine your backup strategy.

About this task

An SLA defines the level of service that is expected and addresses many service-related issues, including the availability and performance of the service. RTO is the time by which a business process must be restored after a disruption in service. RPO defines the strategy for the age of the files that must be recovered from backup storage for regular operations to resume after a failure. SLA, RTO, and RPO contribute to the data protection strategy.

Steps

1. Determine when you should back up your resources.
2. Decide how many backup jobs you require.
3. Decide how to name your backups.
4. Decide if you want Consistency Group Snapshot copies and decide on appropriate options for deleting Consistency Group Snapshot copies.
5. Decide whether you want to use NetApp SnapMirror technology for replication or NetApp SnapVault technology for long term retention.
6. Determine the retention period for the Snapshot copies on the source storage system and the SnapMirror destination.
7. Determine if you want to run any commands before or after the backup operation and provide a prescript or postscript.

Backup strategy for custom plug-ins

Backup schedules of custom plug-in resources

The most critical factor in determining a backup schedule is the rate of change for the resource. The more often you back up your resources, the fewer archive logs SnapCenter has to use for restoring, which can result in faster restore operations.

You might back up a heavily used resource every hour, while you might back up a rarely used resource once a day. Other factors include the importance of the resource to your organization, your service-level agreement (SLA) and your recovery point objective (RPO).

SLA defines the level of service expected and addresses many service-related issues, including the availability and performance of service. RPO defines the strategy for the age of the files that must be recovered from backup storage for regular operations to resume after a failure. SLA and RPO contribute to the data protection strategy.

Backup schedules have two parts, as follows:

- Backup frequency

  Backup frequency (how often backups are to be performed), also called schedule type for some plug-ins, is part of a policy configuration. For example, you might configure the backup frequency as hourly, daily, weekly or monthly. You can access policies in the SnapCenter GUI by clicking **Settings** > **Policies**.

- Backup schedules

  Backup schedules (exactly when backups are to be performed) are part of a resource or resource group configuration. For example, if you have a resource group that has a policy configured for weekly backups, you might configure the schedule to back up every Thursday at 10:00 p.m. You can access resource group schedules in the SnapCenter GUI by clicking **Resources**, then selecting the appropriate plug-in, and clicking **View** > **Resource Group**.

Number of backup jobs needed

Factors that determine the number of backup jobs that you need include the size of the resource, the number of volumes used, the rate of change of the resource, and your Service Level Agreement (SLA).

The number of backup jobs that you choose typically depends on the number of volumes on which you placed your resources. For example, if you placed a group of small resources on one volume and a large resource on another volume, you might create one backup job for the small resources and one backup job for the large resource.

Types of restore strategies supported for manually added custom plug-in resources

You must define a strategy before you can successfully perform restore operations using SnapCenter. There are two types of restore strategies for manually added custom plug-in resources.

> ⓘ You cannot recover manually added custom plug-in resources.

Complete resource restore

- Restores all volumes, qtrees, and LUNs of a resource

> ⓘ If the resource contains volumes or qtrees, the Snapshot copies taken after the Snapshot copy selected for restore on such volumes or qtrees are deleted and cannot be recovered. Also, if any other resource is hosted on the same volumes or qtrees, then that resource is also deleted.

File level restore

- Restores files from volumes, qtrees, or directories
- Restores only the selected LUNs

Develop a plug-in for your application

Overview

The SnapCenter Server enables you to deploy and manage your applications as plug-ins to SnapCenter. Applications of your choice can be plugged into the SnapCenter Server for data protection and management capabilities.

SnapCenter enables you to develop custom plug-ins using different programming languages. You can develop a custom plug-in using Perl, Java, BATCH, or other Scripting languages.

To use custom plug-ins in SnapCenter, you must perform the following tasks:

- Create a plug-in for your application using the instructions in this guide
- Create a description file
- Export the custom plug-in to install it on the SnapCenter host
- Upload the plug-in zip file into SnapCenter Server

Generic plug-in handling in all API calls

For every API call, use the following information:

- Plug-in parameters
- Exit codes
- Log error messages
- Data consistency

Use Plug-in parameters

A set of parameters are passed to the plug-in as part of every API call made. The following table lists the specific information for the parameters.

Parameter	Purpose
ACTION	Determines the workflow name. For example, discover, backup, fileOrVolRestore or cloneVolAndLun
RESOURCES	Lists resources to be protected. A resource is identified by UID and Type. The list is presented to the plug-in in the following format: “<UID>,<TYPE>;<UID>,<TYPE>”. For example, “Instance1,Instance;Instance2\\DB1,Database”

Parameter	Purpose
APP_NAME	Determines which plug-in is being used. For example, DB2, MYSQL. SnapCenter Server has built-in support for the listed applications. This parameter is case sensitive.
APP_IGNORE_ERROR	(Y or N) This causes SnapCenter to exit or not exit when an application error is encountered. This is useful when you are backing up multiple databases and do not want a single failure to stop the backup operation.
<RESOURCE_NAME>__APP_INSTANCE_USERNAME	SnapCenter credential is set for the resource.
<RESOURCE_NAME>_APP_INSTANCE_PASSWORD	SnapCenter credential is set for the resource.
<RESOURCE_NAME>_<CUSTOM_PARAM>	Every Resource level custom key value is available to plug-ins prefixed with “<RESOURCE_NAME>_”. For example, if a custom key is “MASTER_SLAVE” for a resource named “MySQLDB”, then it will be available as MySQLDB_MASTER_SLAVE

Use exit codes

The plug-in returns the status of the operation back to the host by means of exit codes. Each code has a specific meaning and the plug-in uses the right exit code to indicate the same.

The following table depicts error codes and their meaning.

Exit code	Purpose
0	Successful operation.
99	Requested operation is not supported or implemented.
100	Failed operation, skip unquiesce, and exit. Unquiesce is by default.
101	Failed operation, continue with backup operation.
other	Failed operation, run unquiesce, and exit.

Log error messages

The error messages are passed from the plug-in to the SnapCenter Server. The message includes the message, log level, and time stamp.

The following table lists levels and their purposes.

Parameter	Purpose
INFO	informational message
WARN	warning message
ERROR	error message
DEBUG	debug message
TRACE	trace message

Preserve data consistency

Custom plug-ins preserve data between operations of the same workflow execution. For example, a plug-in can store data at the end of quiesce, which can be used during unquiesce operation.

The data to be preserved is set as part of result object by plug-in. It follows a specific format and is described in detail under each style of plug-in development.

PERL-based development

You must follow certain conventions while developing the plug-in using PERL.

- Contents must be readable
- Must implement mandatory operations setENV, quiesce, and unquiesce
- Must use a specific syntax to pass results back to the agent
- The contents should be saved as <PLUGIN_NAME>.pm file

Available operations are

- setENV
- version
- quiesce
- unquiesce
- clone_pre, clone_post
- restore_pre, restore
- cleanup

General plug-in handling

Using results object

Every custom plug-in operation must define the results object. This object sends messages, exit code, stdout, and stderr back to the host agent.

Results object:

```
my $result = {
```

```
      exit_code => 0,
      stdout => "",
      stderr => "",
};
```

Returning the results object:

```
return $result;
```

Preserving data consistency

It is possible to preserve data between operations (except cleanup) as part of same workflow execution. This is done using key-value pairs. The key-value pairs of data are set as part of result object and are retained and available in the subsequent operations of same workflow.

The following code sample sets the data to be preserved:

```
my $result = {
  exit_code => 0,
  stdout => "",
  stderr => "",
};
  $result->{env}->{‘key1’} = ‘value1’;
  $result->{env}->{‘key2’} = ‘value2’;
  ….
  return $result
```

The above code sets two key-value pairs, which are available as input in the subsequent operation. The two key-value pairs are accessible using the following code:

```
sub setENV {
    my ($self, $config) = @_;
    my $first_value = $config->{‘key1’};
    my $second_value = $config->{‘key2’};
    …
}
```

```
=== Logging error messages
```

Each operation can send messages back to the host agent, which displays and stores the content. A message contains the message level, a timestamp, and a message text. Multiline messages are supported.

```
Load the SnapCreator::Event Class:
my $msgObj = new SnapCreator::Event();
my @message_a = ();
```

Use the msgObj to capture a message by using the collect method.

```
$msgObj->collect(\@message_a, INFO, "My INFO Message");
$msgObj->collect(\@message_a, WARN, "My WARN Message");
$msgObj->collect(\@message_a, ERROR, "My ERROR Message");
$msgObj->collect(\@message_a, DEBUG, "My DEBUG Message");
$msgObj->collect(\@message_a, TRACE, "My TRACE Message");
```

Apply messages to the results object:

```
$result->{message} = \@message_a;
```

Using plug-in stubs

Custom plug-ins must expose plug-in stubs. These are methods that the SnapCenter Server calls, based on a workflow.

Plug-in Stub	Optional/Required	Purpose
setENV	required	This stub sets the environment and the configuration object. Any environment parsing or handling should be done here. Each time a stub is called, the setENV stub is called just before. It is only required for PERL-style plug-ins.
Version	Optional	This stub is used to get application version.

Plug-in Stub	Optional/Required	Purpose
Discover	Optional	This stub is used to discover application objects like instance or database hosted on the agent or host. The plug-in is expected to return discovered application objects in specific format as part of the response. This stub is only used in case the application is integrated with SnapDrive for Unix. ⓘ Linux file system (Linux Flavors) is supported. AIX/Solaris (Unix Flavors) are not supported.
discovery_complete	Optional	This stub is used to discover application objects like instance or database hosted on the agent or host. The plug-in is expected to return discovered application objects in specific format as part of the response. This stub is only used in case the application is integrated with SnapDrive for Unix. ⓘ Linux file system (Linux flavors) is supported. AIX and Solaris (Unix flavors) are not supported.
Quiesce	required	This stub is responsible for performing a quiesce, which means placing application into a state where you can create a Snapshot copy. This is called before Snapshot copy operation. The metadata of application to be retained should be set as part of response, which shall be returned during subsequent clone or restore operations on corresponding storage Snapshot copy in the form of configuration parameters.

Plug-in Stub	Optional/Required	Purpose
Unquiesce	required	This stub is responsible for performing a unquiesce, which means placing application into a normal state. This is called after you create a Snapshot copy.
clone_pre	optional	This stub is responsible for performing preclone tasks. This assumes you are using the built-in SnapCenter Server cloning interface and is triggered when performing clone operation.
clone_post	optional	This stub is responsible for performing post clone tasks. This assumes you are using the built-in SnapCenter Server cloning interface and is triggered only when performing clone operation.
restore_pre	optional	This stub is responsible for performing prerestore tasks. This assumes you are using the built-in SnapCenter Server restore interface and is triggered while performing restore operation.
Restore	optional	This stub is responsible for performing application restore tasks. This assumes you are using the built-in SnapCenter Server restore interface and is only triggered when performing restore operation.

Plug-in Stub	Optional/Required	Purpose
Cleanup	optional	This stub is responsible for performing cleanup after backup, restore, or clone operations. Cleanup can be during normal workflow execution or in the event of a workflow failure. You can infer the workflow name under which cleanup is called by referring to configuration parameter ACTION, which can be backup, cloneVolAndLun, or fileOrVolRestore. The configuration parameter ERROR_MESSAGE indicates if there was any error while executing the workflow. If ERROR_MESSAGE is defined and NOT NULL, then cleanup is called during workflow failure execution.
app_version	Optional	This stub is used by SnapCenter to get application version detail managed by the plug-in.

Plug-in package information

Every plug-in must have following information:

```
package MOCK;
our @ISA = qw(SnapCreator::Mod);
=head1 NAME
MOCK - class which represents a MOCK module.
=cut
=head1 DESCRIPTION
MOCK implements methods which only log requests.
=cut
use strict;
use warnings;
use diagnostics;
use SnapCreator::Util::Generic qw ( trim isEmpty );
use SnapCreator::Util::OS qw ( isWindows isUnix getUid
createTmpFile );
use SnapCreator::Event qw ( INFO ERROR WARN DEBUG COMMENT ASUP
CMD DUMP );
my $msgObj = new SnapCreator::Event();
my %config_h = ();
```

Operations

You can code various operations like setENV, Version, Quiesce, and Unquiesce, which are supported by the custom plug-ins.

setENV operation

The setENV operation is required for plug-ins created using PERL. You can set the ENV and can easily access plug-in parameters.

```
sub setENV {
    my ($self, $obj) = @_;
    %config_h = %{$obj};
    my $result = {
      exit_code => 0,
      stdout => "",
      stderr => "",
    };
    return $result;
}
```

Version operation

The version operation returns the application version information.

```
sub version {
  my $version_result = {
    major => 1,
    minor => 2,
    patch => 1,
    build => 0
  };
  my @message_a = ();
  $msgObj->collect(\@message_a, INFO, "VOLUMES
$config_h{'VOLUMES'}");
  $msgObj->collect(\@message_a, INFO,
"$config_h{'APP_NAME'}::quiesce");
  $version_result->{message} = \@message_a;
  return $version_result;
}
```

Quiesce operations

Quiesce operation performs application quiesce operation on resources listed in the RESOURCES parameter.

```
sub quiesce {
  my $result = {
      exit_code => 0,
      stdout => "",
      stderr => "",
  };
  my @message_a = ();
  $msgObj->collect(\@message_a, INFO, "VOLUMES
$config_h{'VOLUMES'}");
  $msgObj->collect(\@message_a, INFO,
"$config_h{'APP_NAME'}::quiesce");
  $result->{message} = \@message_a;
  return $result;
}
```

Unquiesce operation

Unquiesce operation is required to unquiesce the application. The list of resources is available in the RESOURCES parameter.

```
sub unquiesce {
  my $result = {
      exit_code => 0,
      stdout => "",
      stderr => "",
  };
  my @message_a = ();
  $msgObj->collect(\@message_a, INFO, "VOLUMES
$config_h{'VOLUMES'}");
  $msgObj->collect(\@message_a, INFO,
"$config_h{'APP_NAME'}::unquiesce");
  $result->{message} = \@message_a;
  return $result;
}
```

NATIVE style

SnapCenter supports non-PERL programming or scripting languages to create plug-ins. This is known as NATIVE style programming, which can be script or BATCH file.

The NATIVE-style plug-ins must follow certain conventions given below:

The plug-in must be executable

- For Unix systems, the user who runs the agent must have execute privileges on the plug-in

- For Windows systems, PowerShell plug-ins must have the suffix .ps1, other windows scripts must have either .cmd or .bat suffix and must be executable by the user
- The plug-ins must react to command-line argument like "-quiesce", "-unquiesce"
- The plug-ins must return exit code 99 incase an operation or function is not implemented
- The plug-ins must use a specific syntax to pass results back to the server

General plug-in handling

Logging error messages

Each operation can send messages back to the server, which displays and stores the content. A message contains the message level, a timestamp, and a message text. Multiline messages are supported.

Format:

```
SC_MSG#<level>#<timestamp>#<message>
SC_MESSAGE#<level>#<timestamp>#<message>
```

Using plug-in stubs

SnapCenter plug-ins must implement plug-in stubs. These are methods that the SnapCenter Server calls based on a specific workflow.

Plug-in Stub	Optional/Required	Purpose
quiesce	required	This stub is responsible for performing a quiesce. It places the application into a state where we can create a Snapshot copy. This is called before storage Snapshot copy operation.
unquiesce	required	This stub is responsible for performing a unquiesce. It places the application in a normal state. This is called after storage Snapshot copy operation.
clone_pre	optional	This stub is responsible for performing pre clone tasks. This assumes that you are using the built-in SnapCenter cloning interface and also is only triggered while performing action "clone_vol or clone_lun".

Plug-in Stub	Optional/Required	Purpose
clone_post	Optional	This stub is responsible for performing post clone tasks. This assumes you are using the built-in SnapCenter cloning interface and also is only triggered while performing "clone_vol or clone_lun" operations.
restore_pre	Optional	This stub is responsible for performing pre restore tasks. This assumes you are using the built-in SnapCenter restore interface and is only triggered while performing restore operation.
restore	optional	This stub is responsible for performing all restore actions. This assumes you are not using built-in restore interface. It is triggered while performing restore operation.

Examples

Windows PowerShell

Check if the script can be executed on your system. If you cannot execute the script, set Set-ExecutionPolicy bypass for the script and retry the operation.

```
if ($args.length -ne 1) {
  write-warning "You must specify a method";
  break;
}
function log ($level, $message) {
  $d = get-date
  echo "SC_MSG#$level#$d#$message"
}
function quiesce {
  $app_name = (get-item env:APP_NAME).value
  log "INFO" "Quiescing application using script $app_name";
  log "INFO" "Quiescing application finished successfully"
}
function unquiesce {
  $app_name = (get-item env:APP_NAME).value
  log "INFO" "Unquiescing application using script $app_name";
  log "INFO" "Unquiescing application finished successfully"
}
  switch ($args[0]) {
    "-quiesce" {
    quiesce;
}
"-unquiesce" {
    unquiesce;
}
default {
    write-error "Function $args[0] is not implemented";
    exit 99;
  }
}
exit 0;
```

Java style

A Java custom plug-in interacts directly with an application like database, instance and so on.

Limitations

There are certain limitations that you should be aware of while developing a plug-in using Java programing language.

Plug-in characteristic	Java plug-in
Complexity	Low to Medium

Plug-in characteristic	Java plug-in
Memory footprint	Up to 10-20 MB
Dependencies on other libraries	Libraries for application communication
Number of threads	1
Thread runtime	Less than an hour

Reason for Java limitations

The goal of the SnapCenter Agent is to ensure continuous, safe, and robust application integration. By supporting Java plug-ins, it is possible for plug-ins to introduce memory leaks and other unwanted issues. Those issues are hard to tackle, especially when the goal is to keep things simple to use. If a plug-in’s complexity is not too complex, it is much less likely that the developers would have introduced the errors. The danger of Java plug-in is that they are running in the same JVM as the SnapCenter Agent itself. When the plug-in crashes or leaks memory, it may also impact the Agent negatively.

Supported methods

Method	Required	Description	Called when and by whom?
Version	Yes	Needs to return the version of the plug-in.	By the SnapCenter Server or agent to request the version of the plug-in.
Quiesce	Yes	Needs to perform a quiesce on the application. In most cases, this means putting the application into a state where the SnapCenter Server can create a backup (for example, a Snapshot copy).	Before the SnapCenter Server creates a Snapshot(s) copy or performs a backup in general.
Unquiesce	Yes	Needs to perform an unquiesce on the application. In most cases, this means putting the application back into a normal operation state.	After the SnapCenter Server has created a Snapshot copy or has performed a backup in general.
Cleanup	No	Responsible for cleaning up anything that the plug-in needs to clean up.	When a workflow on the SnapCenter Server finish (successfully or with a failure).

Method	Required	Description	Called when and by whom?
clonePre	No	Should perform actions that need to happen before a clone operation is performed.	When a user triggers a "cloneVol" or "cloneLun" action and uses the built-in cloning wizard (GUI/CLI).
clonePost	No	Should perform actions that need to happen after a clone operation was performed.	When a user triggers a "cloneVol" or "cloneLun" action and uses the built-in cloning wizard (GUI/CLI).
restorePre	No	Should perform actions that need to happen before the restore operation is called.	When a user triggers a restore operation.
Restore	No	Responsible for performing a restore/recovery of application.	When a user triggers a restore operation.
appVersion	No	To retrieve application version managed by the plug-in.	As part of ASUP data collection in every workflow like Backup/Restore/Clone.

Tutorial

This section describes how to create a custom plug-in using the Java programming language.

Setting up eclipse

1. Create a new Java Project "TutorialPlugin" in Eclipse
2. Click **Finish**
3. Right click the **new project → Properties → Java Build Path → Libraries → Add External JARs**
4. Navigate to the ../lib/ folder of host Agent and select jars scAgent-5.0-core.jar and common-5.0.jar
5. Select the project and right click the **src folder → New → Package** and create a new package with the name com.netapp.snapcreator.agent.plugin.TutorialPlugin
6. Right-click on the new package and select New → Java Class.
    a. Enter name as TutorialPlugin.
    b. Click the superclass browse button and search for "*AbstractPlugin". Only one result should show up:

    ```
    "AbstractPlugin - com.netapp.snapcreator.agent.nextgen.plugin".
    ```

c. Click **Finish**.

d. Java class:

```
package com.netapp.snapcreator.agent.plugin.TutorialPlugin;
import
com.netapp.snapcreator.agent.nextgen.common.result.Describe
Result;
import
com.netapp.snapcreator.agent.nextgen.common.result.Result;
import
com.netapp.snapcreator.agent.nextgen.common.result.VersionR
esult;
import
com.netapp.snapcreator.agent.nextgen.context.Context;
import
com.netapp.snapcreator.agent.nextgen.plugin.AbstractPlugin;
public class TutorialPlugin extends AbstractPlugin {
  @Override
  public DescribeResult describe(Context context) {
    // TODO Auto-generated method stub
    return null;
  }
  @Override
  public Result quiesce(Context context) {
    // TODO Auto-generated method stub
    return null;
  }
  @Override
  public Result unquiesce(Context context) {
    // TODO Auto-generated method stub
    return null;
  }
  @Override
  public VersionResult version() {
    // TODO Auto-generated method stub
    return null;
  }
}
```

Implementing the required methods

Quiesce, unquiesce, and version are mandatory methods that each custom Java plug-in must implement.

The following is a version method to return the version of the plug-in.

```
@Override
public VersionResult version() {
    VersionResult versionResult = VersionResult.builder()
                                           .withMajor(1)
                                           .withMinor(0)
                                           .withPatch(0)
                                           .withBuild(0)
                                           .build();
    return versionResult;
}
```

```
Below is the implementation of quiesce and unquiesce method. These will be
interacting with   the application, which is being protected by SnapCenter
Server. As this is just a tutorial, the
application part is not explained, and the focus is more on the
functionality that SnapCenter   Agent provides the following to the plug-
in developers:
```

```
@Override
  public Result quiesce(Context context) {
    final Logger logger = context.getLogger();
    /*
      * TODO: Add application interaction here
    */
```

```
logger.error("Something bad happened.");
logger.info("Successfully handled application");
```

```
    Result result = Result.builder()
                    .withExitCode(0)
                    .withMessages(logger.getMessages())
                    .build();
    return result;
}
```

The method gets passed in a Context object. This contains multiple helpers, for example a Logger and a Context Store, and also the information about the current operation (workflow-ID, job-ID). We can get the logger by calling final Logger logger = context.getLogger();. The logger object provides similar methods known from other logging frameworks, for example, logback. In the result object, you can also specify the exit code. In this example, zero is returned, since there was no issue. Other exit codes can map to different failure scenarios.

Using result object

The Result object contains the following parameters:

Parameter	Default	Description
Config	Empty config	This parameter can be used to send config parameters back to the server. It can be parameters that the plug-in wants to update. Whether this change is actually reflected in the config on the SnapCenter Server is dependent on the APP_CONF_PERSISTENCY=Y or N parameter in the config.
exitCode	0	Indicates the status of the operation. A "0" means the operation was executed successfully. Other values indicate errors or warnings.
Stdout	Empty List	This can be used to transmit stdout messages back to the SnapCenter Server.
Stderr	Empty List	This can be used to transmit stderr messages back to the SnapCenter Server.
Messages	Empty List	This list contains all the messages that a plug-in wants to return to the server. The SnapCenter Server displays those messages in the CLI or GUI.

The SnapCenter Agent provides Builders (Builder Pattern) for all its result types. This makes using them very straightforward:

```
Result result = Result.builder()
                    .withExitCode(0)
                    .withStdout(stdout)
                    .withStderr(stderr)
                    .withConfig(config)
                    .withMessages(logger.getMessages())
                    .build()
```

For example, set exit code to 0, set lists for Stdout and Stderr, set config parameters and also append the log messages that will be sent back to the server. If you do not need all the parameters, send only the ones that are needed. As each parameter has a default value, if you remove .withExitCode(0) from the code below, the result is unaffected:

```
Result result = Result.builder()
                      .withExitCode(0)
                      .withMessages(logger.getMessages())
                      .build();
```

VersionResult

The VersionResult informs the SnapCenter Server the plug-in version. As it also inherits from Result, it contains the config, exitCode, stdout, stderr, and messages parameters.

Parameter	Default	Description
Major	0	Major version field of the plug-in.
Minor	0	Minor version field of the plug-in.
Patch	0	Patch version field of the plug-in.
Build	0	Build version field of the plug-in.

For example:

```
VersionResult result = VersionResult.builder()
                                  .withMajor(1)
                                  .withMinor(0)
                                  .withPatch(0)
                                  .withBuild(0)
                                  .build();
```

Using the Context Object

The context object provides the following methods:

Context method	Purpose
String getWorkflowId();	Returns the workflow id that is being used by the SnapCenter Server for the current workflow.
Config getConfig();	Returns the config that is being send from the SnapCenter Server to the Agent.

Workflow-ID

The workflow-ID is the id that the SnapCenter Server uses to refer to a specific running workflow.

Config

This object contains (most) of the parameters that a user can set in the config on the SnapCenter Server. However, due to security reasons, some of those parameters may get filtered on the server side. Following is an example on how to access to the Config and retrieve a parameter:

```
final Config config = context.getConfig();
String myParameter =
config.getParameter("PLUGIN_MANDATORY_PARAMETER");
```

""// myParameter" now contains the parameter read from the config on the SnapCenter Server If a config parameter key doesn’t exist, it will return an empty String ("").

Exporting the plug-in

You must export the plug-in to install it on the SnapCenter host.

In Eclipse perform the following tasks:

1. Right click on the base package of the plug-in (in our example com.netapp.snapcreator.agent.plugin.TutorialPlugin).
2. Select **Export → Java → Jar File**
3. Click **Next**.
4. In the following window, specify the destination jar file path: tutorial_plugin.jar The plug-in’s base class is named TutorialPlugin.class, the plug-in must be added to a folder with the same name.

If your plug-in depends on additional libraries, you can create the following folder: lib/

You can add jar files, on which the plug-in is dependent (for example, a database driver). When SnapCenter loads the plug-in, it automatically associates all the jar files in this folder with it and adds them to the classpath.

Custom plug-in in SnapCenter

Custom plug-in in SnapCenter

The custom plug-in created using Java, PERL, or NATIVE style can be installed on the host using SnapCenter Server to enable data protection of your application. You must have exported the plug-in to install it on the SnapCenter host using the procedure provided in this tutorial.

Creating a plug-in description file

For every plug-in created, you must have a description file. The description file describes the details of the plug-in. The name of the file must be Plugin_descriptor.xml.

Using plug-in descriptor file attributes and its significance

Attribute	Description
Name	Name of the plug-in. Alpha numeric characters are allowed. For example, DB2, MYSQL, MongoDB For plug-ins created in NATIVE style, ensure that you do not provide the extension of the file. For example, if the plug-in name is MongoDB.sh, specify the name as MongoDB.
Version	Plug-in version. Can include both major and minor version. For example, 1.0, 1.1, 2.0, 2.1
DisplayName	The plug-in name to be displayed in SnapCenter Server. If multiple versions of the same plug-in are written, ensure that the display name is the same across all versions.
PluginType	Language used to create the plug-in. Supported values are Perl, Java and Native. Native plug-in type includes Unix/Linux shell scripts, Windows scripts, Python or any other scripting language.
OSName	The host OS name where the plug-in is installed. Valid values are Windows and Linux. It is possible for a single plug-in to be available for deployment on multiple OS types, like PERL type plug-in.
OSVersion	The host OS version where plug-in is installed.
ResourceName	Name of resource type that the plug-in can support. For example, database, instance, collections.
Parent	In case, the ResourceName is hierarchically dependent on another Resource type, then Parent determines the parent ResourceType. For instance, DB2 plug-in, the ResourceName “Database” has a parent “Instance”.
RequireFileSystemPlugin	Yes or No. Determines if the recovery tab is displayed in the restore wizard.
ResourceRequiresAuthentication	Yes or No. Determines if the resources, which are auto discovered or have not been auto discovered need credentials to perform the data protection operations after discovering the storage.
RequireFileSystemClone	Yes or No. Determines if the plug-in requires FileSystem plug-in integration for clone workflow.

An example of the Plugin_descriptor.xml file for custom plug-in DB2 is as follows:

```
<Plugin>
<SMSServer></SMSServer>
<Name>DB2</Name>
<Version>1.0</Version>
<PluginType>Perl</PluginType>
<DisplayName>Custom DB2 Plugin</DisplayName>
<SupportedOS>
<OS>
<OSName>windows</OSName>
<OSVersion>2012</OSVersion>
</OS>
<OS>
<OSName>Linux</OSName>
<OSVersion>7</OSVersion>
</OS>
</SupportedOS>
<ResourceTypes>
<ResourceType>
<ResourceName>Database</ResourceName>
<Parent>Instance</Parent>
</ResourceType>
<ResourceType>
<ResourceName>Instance</ResourceName>
</ResourceType>
</ResourceTypes>
<RequireFileSystemPlugin>no</RequireFileSystemPlugin>
<ResourceRequiresAuthentication>yes</ResourceRequiresAuthentication>
<SupportsApplicationRecovery>yes</SupportsApplicationRecovery>
</Plugin>
```

Creating a ZIP file

After a plug-in is developed and a descriptor file is created, you must add the plug-in files and the Plugin_descriptor.xml file to a folder and zip it.

You must consider the following before creating a ZIP file:

- The script name must be same as the plug-in name.
- For PERL plug-in, the ZIP folder must contain a folder with the script file and the descriptor file must be outside this folder. The folder name must be the same as the plug-in name.
- For plug-ins other than the PERL plug-in, the ZIP folder must contain the descriptor and the script files.
- The OS version must be a number.

Examples:

- DB2 plug-in: add DB2.pm and Plugin_descriptor.xml file to “DB2.zip”.
- Plug-in developed using Java: add jar files, dependent jar files, and Plugin_descriptor.xml file to a folder and zip it.

Uploading the plug-in ZIP file

You must upload the plug-in ZIP file to SnapCenter Server so that the plug-in is available for deployment on the desired host.

You can upload the plug-in using the UI or cmdlets.

UI:

- Upload the plug-in ZIP file as part of **Add** or **Modify Host** workflow wizard
- Click **“Select to upload custom plug-in”**

PowerShell:

- Upload-SmPluginPackage cmdlet

  For example, PS> Upload-SmPluginPackage -AbsolutePath c:\DB2_1.zip

  For detailed information about PowerShell cmdlets, use the SnapCenter cmdlet help or see the cmdlet reference information.

SnapCenter Software Cmdlet Reference Guide.

Deploying the custom plug-ins

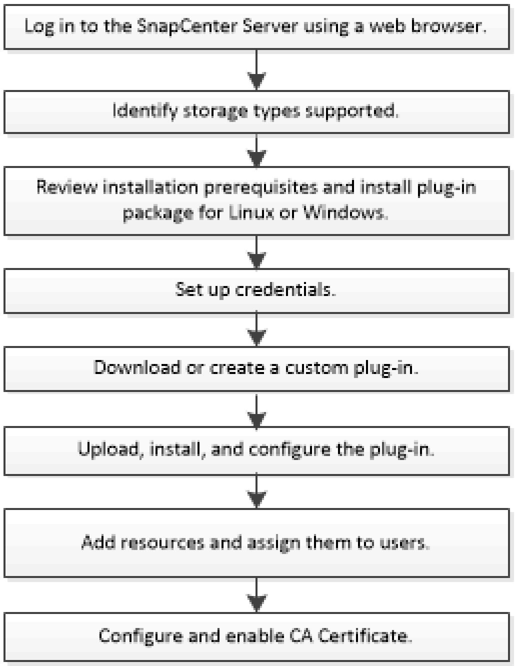
The uploaded custom plug-in is now available for deployment on the desired host as part of the **Add** and **Modify Host** workflow. You can have multiple version of plug-ins uploaded to the SnapCenter Server and you can select the desired version to deploy on a specific host.

For more information on how to upload the plug-in see, Add hosts and install plug-in packages on remote hosts

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.


Log in to the SnapCenter Server using a web browser.
Identify storage types supported.
Review installation prerequisites and install plug-in package for Linux or Windows.
Set up credentials.
Download or create a custom plug-in.
Upload, install, and configure the plug-in.
Add resources and assign them to users.
Configure and enable CA Certificate.

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 or Java 11 (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 should be running.

SHA512 hash

- For custom plug-ins provided by NetApp, you should ensure that you have added the SHA512 hash of the custom plug-in file to the *custom_plugin_checksum_list* file.
  - For Linux host, the SHA512 hash is located at */var/opt/snapcenter/scc/custom_plugin_checksum_list.txt*
  - For Windows host, the SHA512 hash is located at *C:\Program Files\NetApp\SnapCenter Plug-in Creator\etc\custom_plugin_checksum_list.txt*

For custom installation path, the SHA512 hash is located at *<custom path>\NetApp\SnapCenter\Snapcenter Plug-in Creator\etc\custom_plugin_checksum_list.txt*

The custom_plugin_checksum_list is part of the custom plug-in installation on the host by SnapCenter.

- For custom plug-ins created for your application, you should have performed the following steps:
  1. Generated the SHA512 hash of the plug-in zip file.

     You can use online tools like SHA512 hash.
  2. Added the generated SHA512 hash to the custom_plugin_checksum_list file in a new line.

     The comments start with # symbol to identify the plug-in to which the hash belongs.

     Following is an example of an entry of SHA512 hash in the checksum file:

     ```
     #ORASCPM
     03721f567a1e4a1cb5569066b9a58af619ee12b1f8713108f81b696cfbdb81c25232f
     a63d6e6777a2b2a1ec068bb0a93a59a8ade71587182f8bccbe81f7e0ba6
     ```

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 or Java 11 (64-bit), on your Linux host.

  If you are using Windows Server 2019 or Windows Server 2016 for the SnapCenter Server host, you must install Java 1.8 or Java 11 (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.

  Ensure that you are using Sudo version 1.8.7 or later.

```
Cmnd_Alias HPPLCMD = sha224:checksum_value== /home/
LINUX_USER/.sc_netapp/snapcenter_linux_host_plugin.bin,
/opt/NetApp/snapcenter/spl/installation/plugins/uninstall,
/opt/NetApp/snapcenter/spl/bin/spl, /opt/NetApp/snapcenter/scc/bin/scc
Cmnd_Alias PRECHECKCMD = sha224:checksum_value== /home/
LINUX_USER/.sc_netapp/Linux_Prechecks.sh
Cmnd_Alias CONFIGCHECKCMD = sha224:checksum_value==
/opt/NetApp/snapcenter/spl/plugins/scu/scucore/configurationcheck/Config
_Check.sh
Cmnd_Alias SCCMD = sha224:checksum_value==
/opt/NetApp/snapcenter/spl/bin/sc_command_executor
Cmnd_Alias SCCCMDEXECUTOR =checksum_value==
/opt/NetApp/snapcenter/scc/bin/sccCommandExecutor
LINUX_USER ALL=(ALL) NOPASSWD:SETENV: HPPLCMD, PRECHECKCMD,
CONFIGCHECKCMD, SCCCMDEXECUTOR, SCCMD
Defaults: LINUX_USER !visiblepw
Defaults: LINUX_USER !requiretty
```

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

You can obtain the *checksum_value* from the **oracle_checksum.txt** file, which is located at *C:\ProgramData\NetApp\SnapCenter\Package Repository*.

ⓘ The example should be used only as a reference for creating your own data.

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 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 ⓘ 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 4.7.2 or later • Windows Management Framework (WMF) 4.0 or later • PowerShell 4.0 or later For the latest information about supported versions, see the NetApp Interoperability Matrix Tool. For .NET specific troubleshooting information, see SnapCenter upgrade or install fails for legacy systems that do not have internet connectivity.

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 • Oracle Linux • 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 ⓘ 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 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

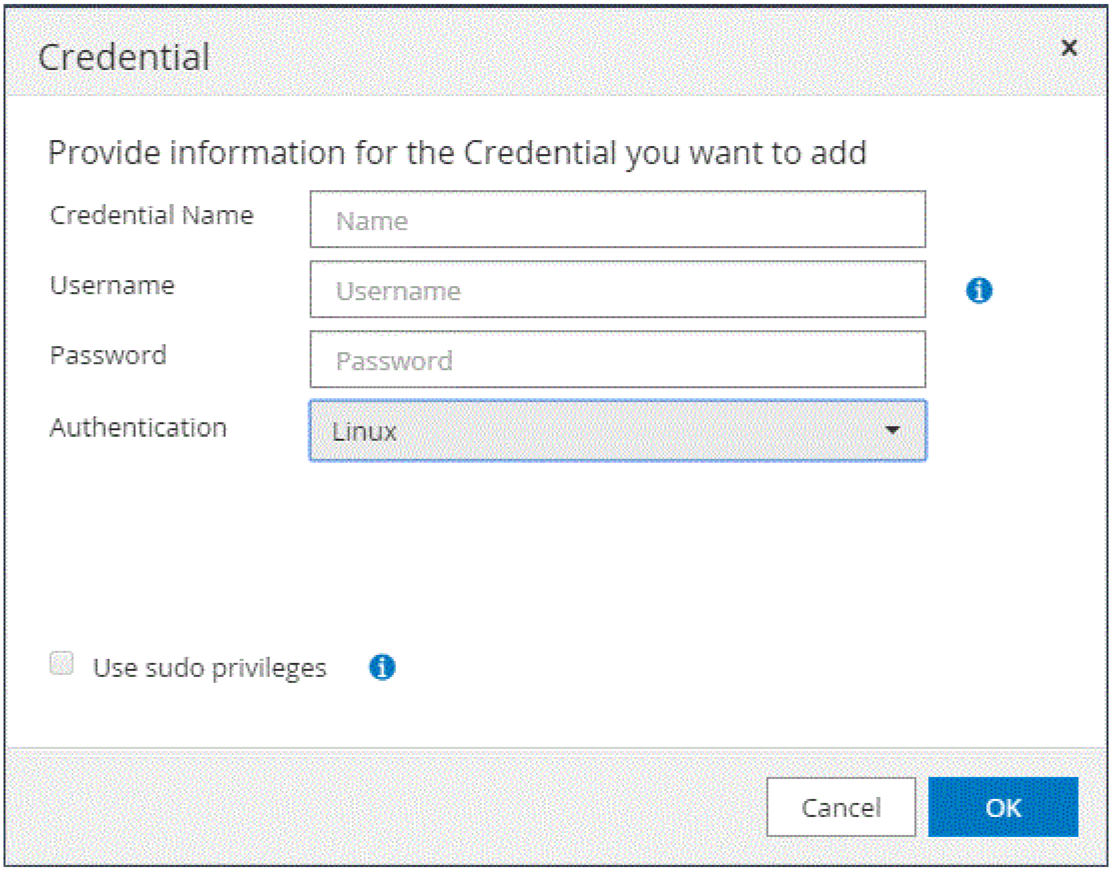
The plug-in uses the credentials that are selected or created while adding a resource. If a resource does not require credentials during data protection operations, you can set the credentials as **None**.

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**.



Credential

Provide information for the Credential you want to add

Authentication: Linux

□ Use sudo privileges

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. • Domain administrator or any member of the administrator group 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: ◦ *NetBIOS\UserName* ◦ *Domain FQDN\UserName* • Local administrator (for workgroups only) 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. ⓘ 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 should 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: • Windows • Linux ⓘ 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. SnapCenter depends on the proper configuration of the DNS. Therefore, the best practice is to enter the FQDN. For Windows environments, the IP address is supported for untrusted domain hosts only if it resolves to the FQDN. You can enter the IP addresses or FQDN of a stand-alone host. 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. The credentials must have administrative rights on the remote host. For details, see the information about creating credentials. You can view details about the credentials by positioning your cursor over the credential name that you specified. ⓘ 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. 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. ⓘ 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. • For the SnapCenter Plug-ins Package for Windows, the default path is C:\Program Files\NetApp\SnapCenter. Optionally, you can customize the path. • For SnapCenter Plug-ins Package for Linux, the default path is /opt/NetApp/snapcenter. Optionally, you can customize the path. • For the SnapCenter Custom Plug-ins: i. In the Custom Plug-ins section, click **Browse**, and select the zipped custom plug-in folder. The zipped folder contains the custom plug-in code and the descriptor .xml file. For Storage Plug-in, navigate to *C:\ProgramData\NetApp\SnapCenter\Package Repository* and select `Storage.zip` folder. ii. Click **Upload**. The descriptor .xml file in the zipped custom plug-in folder is validated before the package is uploaded. The custom plug-ins that are uploaded to the SnapCenter Server are listed. If you want to manage MySQL or DB2 applications, you can use the MySQL and DB2 custom plug-ins that are provided by NetApp.
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. Provide the gMSA name in the following format: domainName\accountName$. 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, and *.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”
> $guid = [guid]::NewGuid().ToString("B")
> netsh http add sslcert ipport=0.0.0.0: _<SMCore Port>_ certhash=$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.

Prepare for data protection

Prerequisites for using the SnapCenter Custom Plug-ins

Before you use SnapCenter Custom Plug-ins, the SnapCenter administrator must install and configure the SnapCenter Server and perform the prerequisite tasks.

- Install and configure SnapCenter Server.
- Log in to SnapCenter Server.
- Configure the SnapCenter environment by adding storage system connections and creating credentials, if applicable.
- Add hosts, and install and upload the plug-ins.
- If applicable, install Java 1.7 or Java 1.8 on the plug-in host.
- If you have multiple data paths (LIFs) or a dNFS configuration, you can perform the following using the SnapCenter CLI on the database host:
  - By default, all the IP addresses of the database host are added to the NFS storage export policy in storage virtual machine (SVM) for the cloned volumes. If you want to have a specific IP address or restrict to a subset of the IP addresses, run the Set-PreferredHostIPsInStorageExportPolicy CLI.
  - If you have multiple data paths (LIFs) in SVMs, SnapCenter chooses the appropriate data path (LIF) for mounting the NFS cloned volume. However, if you want to specify a specific data path (LIF), you must run the Set-SvmPreferredDataPath CLI. 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 Command Reference Guide.

- Set up SnapMirror and SnapVault, if you want backup replication.
- Ensure that port 9090 is not used by any other application on the host.

  Port 9090 must be reserved for use by SnapCenter Custom Plug-ins in addition to the other ports required by SnapCenter.

How resources, resource groups, and policies are used for protecting custom plug-in resources

Before you use SnapCenter, it is helpful to understand basic concepts related to the backup, clone, and restore operations you want to perform. You interact with resources, resource groups, and policies for different operations.

- Resources are typically databases, Windows file systems, or VMs that you back up or clone with SnapCenter.
- A SnapCenter resource group, is a collection of resources on a host or cluster.

  When you perform an operation on a resource group, you perform that operation on the resources defined in the resource group according to the schedule you specify for the resource group.

  You can back up on demand a single resource or a resource group. You also can perform scheduled backups for single resources and resource groups.

- The policies specify the backup frequency, copy retention, replication, scripts, and other characteristics of data protection operations.

  When you create a resource group, you select one or more policies for that group. You can also select a policy when you perform a backup on demand for a single resource.

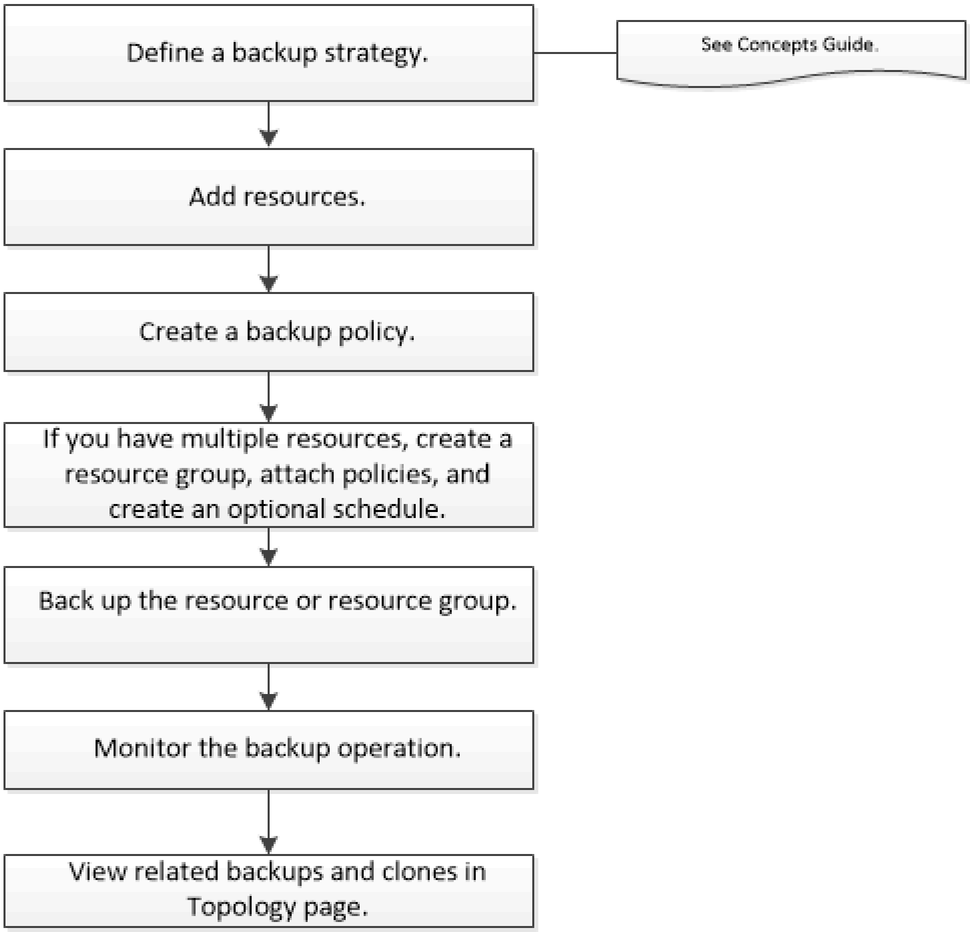
Think of a resource group as defining *what* you want to protect and when you want to protect it in terms of day and time. Think of a policy as defining *how* you want to protect it. If you are backing up all databases or backing up all file systems of a host, for example, you might create a resource group that includes all the databases or all the file systems in the host. You could then attach two policies to the resource group: a daily policy and an hourly policy. When you create the resource group and attach the policies, you might configure the resource group to perform a File-Based backup daily and another schedule that performs Snapshot based backup hourly.

Back up custom plug-in resources

Back up custom plug-in resources

The backup workflow includes planning, identifying the resources for backup, managing backup policies, creating resource groups and attaching policies, creating backups, and monitoring the operations.

The following workflow shows the sequence in which you must perform the backup operation:


Define a backup strategy.
See Concepts Guide.
Add resources.
Create a backup policy.
If you have multiple resources, create a resource group, attach policies, and create an optional schedule.
Back up the resource or resource group.
Monitor the backup operation.
View related backups and clones in Topology page.

You can also use PowerShell cmdlets manually or in scripts to perform backup, restore, and clone operations. For detailed information about PowerShell cmdlets, use the SnapCenter cmdlet help or see the SnapCenter Software Cmdlet Reference Guide

Add resources to SnapCenter Custom Plug-ins

You must add the resources that you want to back up or clone. Depending on your environment, resources might be either database instances or collections that you want to back up or clone.

Before you begin

- You must have completed tasks such as installing the SnapCenter Server, adding hosts, creating storage system connections, and adding credentials.
- You must have created a custom plug-in for your application.
- You must have uploaded the plug-ins to SnapCenter Server.

Steps

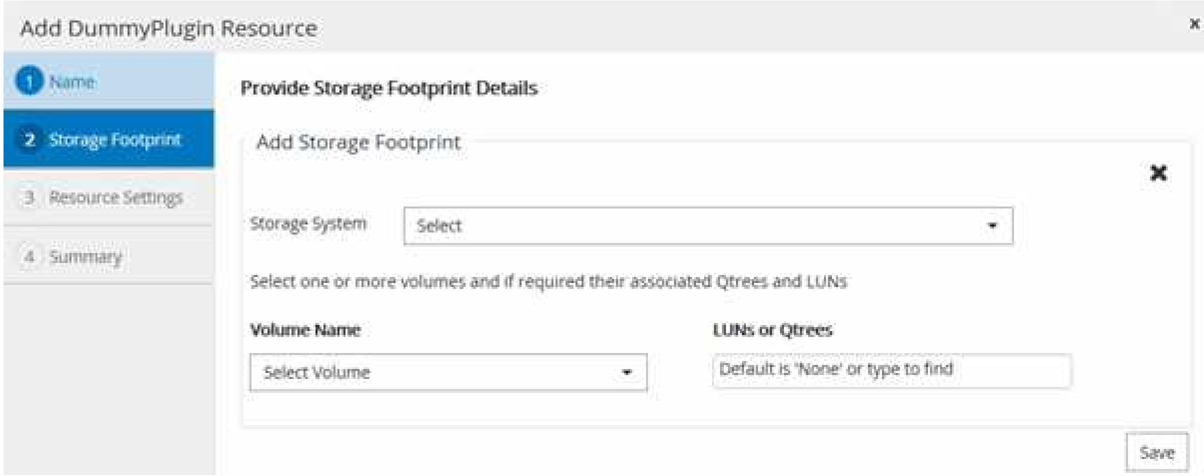
1. In the left navigation pane, select **Resources**, and then select the appropriate plug-in from the list.
2. In the Resources page, select **Add Resource**.
3. In the Provide Resource Details page, perform the following actions:

For this field…	Do this…
Name	Enter the name of the resource.
Host name	Select the host.
Type	Select the type. Type is user defined as per the plug-in description file. For example, database and instance. In case the type selected has a parent, enter the details of the parent. For example, if the type is Database and the parent is Instance, enter the details of the Instance.
Credential name	Select Credential or create a new credential.
Mount Paths	Enter the mount paths where the resource is mounted. This is applicable only for a Windows host.

4. In the Provide Storage Footprint page, select a storage system and choose one or more volumes, LUNs, and qtrees, and then select **Save**.

   Optional: Select the + icon to add more volumes, LUNs, and qtrees from other storage systems.

   > ⓘ SnapCenter Custom Plug-ins does not support automatic discovery of the resources and the storage details for physical and virtual environments. You must provide the storage information for physical and virtual environments while creating the resources.



Add DummyPlugin Resource

Provide Storage Footprint Details

Add Storage Footprint

Storage System: Select

Select one or more volumes and if required their associated Qtrees and LUNs

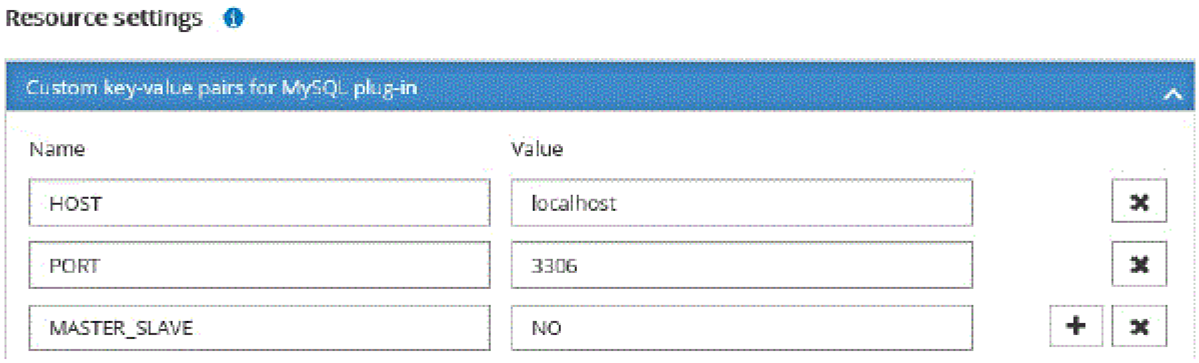
Volume Name: Select Volume

LUNs or Qtrees: Default is 'None' or type to find

5. In the Resource Settings page, provide custom key-value pairs for the resource.

   Use the custom key-value pairs if you want to pass resource-specific information. For example, when you are using the MySQL plug-in, you must specify a HOST as HOST=hostname, PORT =port-no used for MySQL and master-slave configuration as MASTER_SLAVE = “YES” or “NO” (name is MASTER_SLAVE and value is “YES” or “NO”).

Ensure that the words HOST and PORT are in uppercase.



Resource settings

Custom key-value pairs for MySQL plug-in

Name	Value
HOST	localhost
PORT	3306
MASTER_SLAVE	NO

6. Review the summary, and then select **Finish**.

Result

The resources are displayed along with information such as type, host or cluster name, associated resource groups and policies, and overall status.

You must refresh the resources if the databases are renamed outside of SnapCenter.

After you finish

If you want to provide access to the assets to other users, the SnapCenter administrator must assign assets to those users. This enables users to perform the actions for which they have permissions on the assets that are assigned to them.

After adding the resources, you can modify the resource details. If a custom plug-in resource has backups associated with it, the following fields cannot be modified: resource name, resource type, and host name.

Create policies for custom plug-in resources

Before you use SnapCenter to back up custom plug-in specific resources, you must create a backup policy for the resource or resource group that you want to back up.

What you will need

- You should have defined your backup strategy.

  For details, see the information about defining a data protection strategy for custom plug-ins.

- You should have prepared for data protection.

  Preparing for data protection includes tasks such as installing SnapCenter, adding hosts, creating storage system connections, and adding resources.

- The storage virtual machines (SVMs) should be assigned to you for mirror or vault operations.

  The SnapCenter administrator must have assigned the SVMs for both the source and destination volumes to you if you are replicating Snapshot copies to a mirror or vault.

- You should have manually added the resources that you want to protect.

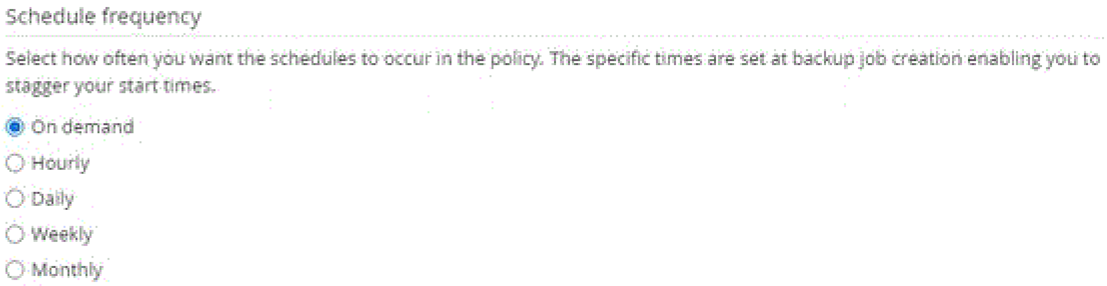
About this task

- A backup policy is a set of rules that governs how you manage, schedule, and retain backups. Additionally, you can specify replication, script, and application settings.
- Specifying options in a policy saves time when you want to reuse the policy for another resource group.

Steps

1. In the left navigation pane, click **Settings**.
2. In the Settings page, click **Policies**.
3. Click **New**.
4. In the Name page, enter the policy name and description.
5. In the Settings page, perform the following steps:
   - Specify the schedule type by selecting **On demand**, **Hourly**, **Daily**, **Weekly**, or **Monthly**.

     You can specify the schedule (start date, end date, and frequency) for the backup operation while creating a resource group. This enables you to create resource groups that share the same policy and backup frequency, but enables you to assign different backup schedules to each policy.

     Schedule frequency

     Select how often you want the schedules to occur in the policy. The specific times are set at backup job creation enabling you to stagger your start times.

     ● On demand
     ○ Hourly
     ○ Daily
     ○ Weekly
     ○ Monthly

     If you have scheduled for 2:00 a.m., the schedule will not be triggered during daylight saving time (DST).

   - In the Custom backup settings section, provide any specific backup settings that has to be passed to the plug-in in key-value format. You can provide multiple key-values to be passed to the plug-in.
6. In the Retention page, specify the retention settings for the backup type and the schedule type selected in the Backup Type page:

If you want to…	Then…
Keep a certain number of Snapshot copies	Select **Total Snapshot copies to keep**, and then specify the number of Snapshot copies that you want to keep. If the number of Snapshot copies exceeds the specified number, the Snapshot copies are deleted with the oldest copies deleted first. ⓘ You must set the retention count to 2 or higher if you plan to enable SnapVault replication. If you set the retention count to 1, the retention operation might fail because the first Snapshot copy is the reference Snapshot copy for the SnapVault relationship until a newer Snapshot copy is replicated to the target. ⓘ The maximum retention value is 1018 for resources on ONTAP 9.4 or later, and 254 for resources on ONTAP 9.3 or earlier. Backups will fail if retention is set to a value higher than what the underlying ONTAP version supports.
Keep the Snapshot copies for a certain number of days	Select **Keep Snapshot copies for**, and then specify the number of days for which you want to keep the Snapshot copies before deleting them.

7. In the Replication page, specify the replication settings:

For this field…	Do this…
Update SnapMirror after creating a local Snapshot copy	Select this field to create mirror copies of the backup sets on another volume (SnapMirror replication). If the protection relationship in ONTAP is of type Mirror and Vault and if you select only this option, Snapshot copy created on the primary will not be transferred to the destination, but will be listed in the destination. If this Snapshot copy is selected from the destination to perform a restore operation, then the following error message is displayed: Secondary Location is not available for the selected vaulted/mirrored backup.

For this field…	Do this…
Update SnapVault after creating a local Snapshot copy	Select this option to perform disk-to-disk backup replication (SnapVault backups).
Secondary policy label	Select a Snapshot label. Depending on the Snapshot copy label that you select, ONTAP applies the secondary Snapshot copy retention policy that matches the label. If you have selected **Update SnapMirror after creating a local Snapshot copy**, you can optionally specify the secondary policy label. However, if you have selected **Update SnapVault after creating a local Snapshot copy**, you should specify the secondary policy label.
Error retry count	Enter the maximum number of replication attempts that can be allowed before the operation stops.

> You should configure SnapMirror retention policy in ONTAP for the secondary storage to avoid reaching the maximum limit of Snapshot copies on the secondary storage.

8. Review the summary, and then click **Finish**.

Create resource groups and attach policies in Snapcenter

A resource group is the container to which you must add resources that you want to back up and protect. It enables you to back up all the data that is associated with a given application simultaneously and is required for any data protection job. You must also attach one or more policies to the resource group to define the type of data protection job that you want to perform.

Steps

1. In the left navigation pane, select **Resources**, and then select the appropriate plug-in from the list.
2. In the Resources page, select New Resource Group.
3. In the Name page, perform the following actions:

For this field…	Do this…
Name	Enter a name for the resource group. Note: The resource group name should not exceed 250 characters.

For this field…	Do this…
Tags	Enter one or more labels that will help you later search for the resource group. For example, if you add HR as a tag to multiple resource groups, you can later find all resource groups associated with the HR tag.
Use custom name format for Snapshot copy	Select this check box, and enter a custom name format that you want to use for the Snapshot copy name. For example, *customtext_resource group_policy_hostname or resource group_hostname*. By default, a timestamp is appended to the Snapshot copy name.

4. Optional: In the Resources page, select a host name from the **Host** drop-down list and resource type from the **Resource Type** drop-down list.

   This helps to filter information on the screen.

5. Select the resources from the **Available Resources** section, and then select the right arrow to move them to the **Selected Resources** section.
6. Optional: In the Application Settings page, do the following:
   a. Select the Backups arrow to set additional backup options:

      Enable consistency group backup and perform the following tasks:

For this field…	Do this…
Afford time to wait for Consistency Group Snapshot operation to complete	Select Urgent, Medium, or Relaxed to specify the wait time for Snapshot copy operation to complete. Urgent = 5 seconds, Medium = 7 seconds, and Relaxed = 20 seconds.
Disable WAFL Sync	Select this to avoid forcing a WAFL consistency point.

   b. Select the Scripts arrow and enter the pre and post commands for quiesce, Snapshot copy, and unquiesce operations. You can also enter the pre commands to be executed before exiting in the event of a failure.
   c. Select the Custom Configurations arrow and enter the custom key-value pairs required for all data protection operations using this resource.

Parameter	Setting	Description
ARCHIVE_LOG_ENABLE	(Y/N)	Enables the archive log management to delete the archive logs.
ARCHIVE_LOG_RETENTION	number_of_days	Specifies the number of days the archive logs are retained. This setting must be equal to or greater than NTAP_SNAPSHOT_ RETENTIONS.
ARCHIVE_LOG_DIR	change_info_directory/logs	Specifies the path to the directory that contains the archive logs.
ARCHIVE_LOG_EXT	file_extension	Specifies the archive log file extension length. For example, if the archive log is log_backup_0_0_0_0.16151855 1942 9 and if the file_extension value is 5, then the extension of the log will retain 5 digits, which is 16151.
ARCHIVE_LOG_RECURSIVE_ SE ARCH	(Y/N)	Enables the management of archive logs within subdirectories. You should use this parameter if the archive logs are located under subdirectories.

d. Select the **Snapshot Copy Tool** arrow to select the tool to create Snapshot copies:

If you want…	Then…
SnapCenter to use the plug-in for Windows and put the file system into a consistent state before creating a Snapshot copy. For Linux resources, this option is not applicable.	Select SnapCenter with File System Consistency. This option is not applicable for SnapCenter Plug-in for SAP HANA Database.
SnapCenter to create a storage level Snapshot copy	Select SnapCenter without File System Consistency.
To enter the command to be executed on the host to create Snapshot copies.	Select Other, and then enter the command to be executed on the host to create a Snapshot copy.

7. In the Policies page, perform the following steps:
    a. Select one or more policies from the drop-down list.

        > You can also create a policy by selecting +.

        The policies are listed in the **Configure schedules for selected policies** section.

    b. In the **Configure Schedules** column, select + for the policy you want to configure.
    c. In the Add schedules for policy *policy_name* dialog box, configure the schedule and select OK.

        Where policy_name is the name of the policy that you have selected.

        The configured schedules are listed in the Applied Schedules column. Third party backup schedules are not supported when they overlap with SnapCenter backup schedules.

8. From the **Email preference** drop-down list on the Notification page, select the scenarios in which you want to send the emails.

    You must also specify the sender and receiver email addresses, and the subject of the email. The SMTP server must be configured in **Settings** > **Global Settings**.

9. Review the summary, and then select **Finish**.

Back up individual custom plug-in resources

If an individual custom plug-in resource is not part of any resource group, you can back up the resource from the Resources page. You can back up the resource on demand, or, if the resource has a policy attached and a schedule configured, then backups occur automatically according to the schedule.

What you will need

- You must have created a backup policy.
- If you want to back up a resource that has a SnapMirror relationship with a secondary storage, the ONTAP role assigned to the storage user should include the “snapmirror all” privilege. However, if you are using the “vsadmin” role, then the “snapmirror all” privilege is not required.

Steps

1. In the left navigation pane, click **Resources**, and then select the appropriate plug-in from the list.
2. In the Resources page, filter resources from the **View** drop-down list based on resource type.

    Click ▼, and then select the host name and the resource type to filter the resources. You can then click ▼ to close the filter pane.

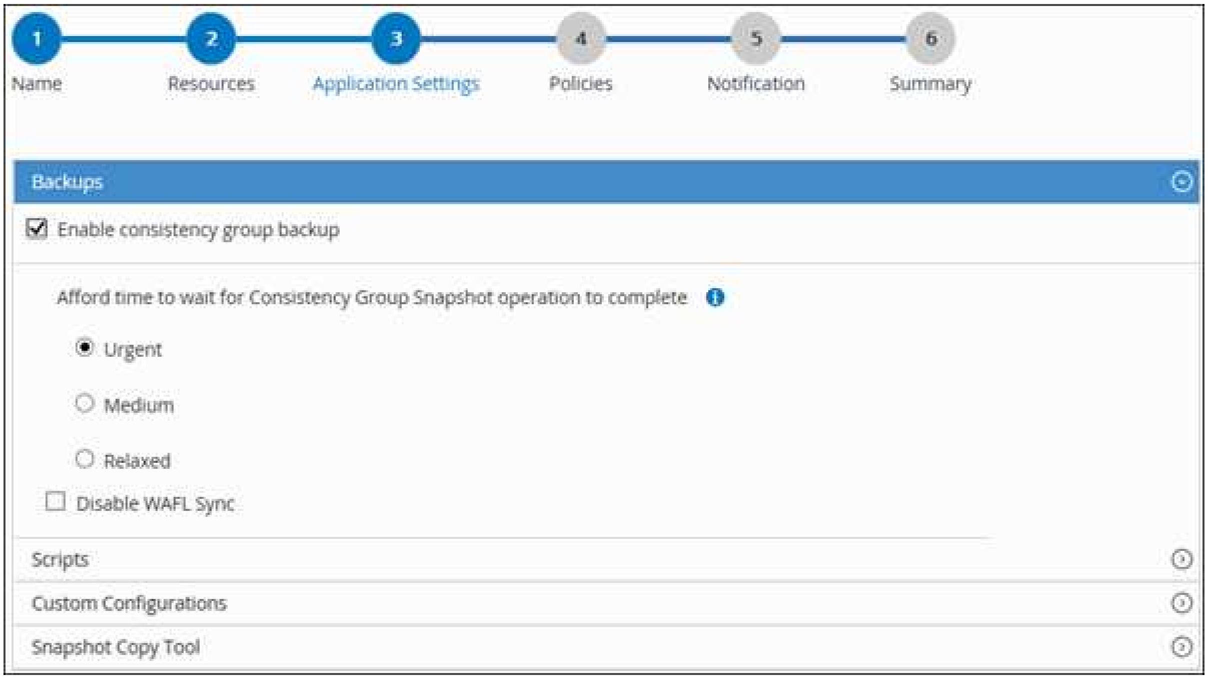
3. Click the resource that you want to back up.
4. In the Resource page, if you want to use a custom name, select the **Use custom name format for Snapshot copy** check box , and then enter a custom name format for the Snapshot copy name.

    For example, *customtext_policy_hostname* or *resource_hostname*. By default, a timestamp is appended to the Snapshot copy name.

5. In the Application Settings page, do the following:
   a. Click the **Backups** arrow to set additional backup options:

      Enable consistency group backup, if needed, and perform the following tasks:

For this field…	Do this…
Afford time to wait for Consistency Group Snapshot operation to complete	Select Urgent, Medium, or Relaxed to specify the wait time for Snapshot copy operation to complete. Urgent = 5 seconds, Medium = 7 seconds, and Relaxed = 20 seconds.
Disable WAFL Sync	Select this to avoid forcing a WAFL consistency point.



☑ Enable consistency group backup

Afford time to wait for Consistency Group Snapshot operation to complete

● Urgent

○ Medium

○ Relaxed

□ Disable WAFL Sync

   b. Click the **Scripts** arrow to run pre and post commands for quiesce, Snapshot copy, and unquiesce operations. You can also run pre commands before exiting the backup operation.

      Prescripts and postscripts are run in the SnapCenter Server.

   c. Click the **Custom Configurations** arrow, and then enter the custom value pairs required for all jobs using this resource.
   d. Click the **Snapshot Copy Tool** arrow to select the tool to create Snapshot copies:

If you want…	Then…
SnapCenter to take a storage level Snapshot copy	Select **SnapCenter without File System Consistency**.
SnapCenter to use the plug-in for Windows to put the file system into a consistent state and then take a Snapshot copy	Select **SnapCenter with File System Consistency**.
To enter the command to create a Snapshot copy	Select **Other**, and then enter the command to create a Snapshot copy.

6. In the Policies page, perform the following steps:
   a. Select one or more policies from the drop-down list.

      > ⓘ You can also create a policy by clicking + .

      In the Configure schedules for selected policies section, the selected policies are listed.

   b. Click + in the Configure Schedules column for the policy for which you want to configure a schedule.
   c. In the Add schedules for policy *policy_name* dialog box, configure the schedule, and then click **OK**.

      Where, *policy_name* is the name of the policy that you have selected.

      The configured schedules are listed in the Applied Schedules column.

7. In the Notification page, from the **Email preference** drop-down list, select the scenarios in which you want to send the emails.

   You must also specify the sender and receiver email addresses, and the subject of the email. SMTP must also be configured in **Settings** > **Global Settings**.

8. Review the summary, and then click **Finish**.

   The resources topology page is displayed.

9. Click **Back up Now**.
10. In the Backup page, perform the following steps:
    a. If you have applied multiple policies to the resource, from the **Policy** drop-down list, select the policy that you want to use for backup.

       If the policy selected for the on-demand backup is associated with a backup schedule, the on-demand backups will be retained based on the retention settings specified for the schedule type.

    b. Click **Backup**.
11. Monitor the operation progress by clicking **Monitor** > **Jobs**.

Back up resource groups of custom plug-in resources

You can back up a resource group on demand from the Resources page. If a resource group has a policy attached and a schedule configured, then backups occur automatically according to the schedule.

What you will need

- You must have created a resource group with a policy attached.
- If you want to back up a resource that has a SnapMirror relationship to secondary storage, the ONTAP role assigned to the storage user should include the “snapmirror all” privilege. However, if you are using the “vsadmin” role, then the “snapmirror all” privilege is not required.

Steps

1. In the left navigation pane, click **Resources**, and then select the appropriate plug-in from the list.
2. In the Resources page, select **Resource Group** from the **View** list.

   You can search the resource group either by entering the resource group name in the search box or by clicking ▼ and selecting the tag. You can then click ▼ to close the filter pane.

3. In the Resource Groups page, select the resource group that you want to back up, and then click **Back up Now**.
4. In the Backup page, perform the following steps:
   a. If you have associated multiple policies with the resource group, from the **Policy** drop-down list, select the policy that you want to use for backup.

      If the policy selected for the on-demand backup is associated with a backup schedule, the on-demand backups will be retained based on the retention settings specified for the schedule type.

   b. Click **Backup**.
5. Monitor the operation progress by clicking **Monitor** > **Jobs**.
   - In MetroCluster configurations, SnapCenter might not be able to detect a protection relationship after a failover.

     Unable to detect SnapMirror or SnapVault relationship after MetroCluster failover

   - If you are backing up application data on VMDKs and the Java heap size for the SnapCenter Plug-in for VMware vSphere is not large enough, the backup might fail. To increase the Java heap size, locate the script file /opt/netapp/init_scripts/scvservice. In that script, the `do_start method` command starts the SnapCenter VMware plug-in service. Update that command to the following: `Java -jar -Xmx8192M -Xms4096M`.

Create a storage system connection and a credential using PowerShell cmdlets

You must create a storage virtual machine (SVM) connection and a credential before using PowerShell cmdlets to perform data protection operations.

What you will need

- You should have prepared the PowerShell environment to execute the PowerShell cmdlets.

- You should have the required permissions in the Infrastructure Admin role to create storage connections.
- You should ensure that the plug-in installations are not in progress.

  Host plug-in installations must not be in progress while adding a storage system connection because the host cache might not be updated and databases status might be displayed in the SnapCenter GUI as “Not available for backup” or “Not on NetApp storage”.
- Storage system names should be unique.

  SnapCenter does not support multiple storage systems with the same name on different clusters. Each storage system that is supported by SnapCenter should have a unique name and a unique management LIF IP address.

Steps

1. Initiate a PowerShell connection session by using the Open-SmConnection cmdlet.

   This example opens a PowerShell session:

   ```
   PS C:\> Open-SmConnection
   ```

2. Create a new connection to the storage system by using the Add-SmStorageConnection cmdlet.

   This example creates a new storage system connection:

   ```
   PS C:\> Add-SmStorageConnection -Storage test_vs1 -Protocol Https
   -Timeout 60
   ```

3. Create a new credential by using the Add-SmCredential cmdlet.

   This example creates a new credential named FinanceAdmin with Windows credentials:

   ```
   PS C:> Add-SmCredential -Name FinanceAdmin -AuthMode Windows
   -Credential sddev\administrator
   ```

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.

Back up resources using PowerShell cmdlets

Backing up a resource includes establishing a connection with the SnapCenter Server, adding resources, adding a policy, creating a backup resource group, and backing up.

What you will need

- You must have prepared the PowerShell environment to execute the PowerShell cmdlets.

- You must have added the storage system connection and created a credential.

About this task

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. Initiate a connection session with the SnapCenter Server for a specified user by using the Open-SmConnection cmdlet.

   ```
   Open-smconnection  -SMSbaseurl  https:\\snapctr.demo.netapp.com:8146\
   ```

   The username and password prompt is displayed.

2. Add resources by using the Add-SmResources cmdlet.

   This example adds resources:

   ```
   Add-SmResource -HostName '10.232.206.248' -PluginCode 'DB2'
   -ResourceName NONREC1 -ResourceType Database -StorageFootPrint ( @
   {"VolumeName"="DB2_NONREC1DB";"LunName"="DB2_NONREC1DB";"Vserver"="vserv
   er_scauto_secondary"}) -Instance db2inst1
   ```

3. Create a backup policy by using the Add-SmPolicy cmdlet.

   This example creates a new backup policy:

   ```
   Add-SMPolicy -PolicyName 'db2VolumePolicy' -PolicyType 'Backup'
   -PluginPolicyType DB2  -description 'VolumePolicy'
   ```

4. Add a new resource group to SnapCenter by using the Add-SmResourceGroup cmdlet.

   This example creates a new resource group with the specified policy and resources:

   ```
   Add-SmResourceGroup -ResourceGroupName
   'Verify_ManualBackup_DatabaseLevel_MultipleVolume_unix' -Resources @(@
   {"Host"="10.232.206.248";"Uid"="db2inst2\NONREC"},@{"Host"="10.232.206.2
   48";"Uid"="db2inst1\NONREC"}) -Policies db2ManualPolicy
   ```

5. Initiate a new backup job by using the New-SmBackup cmdlet.

```
New-SMBackup -DatasetName
Verify_ManualBackup_DatabaseLevel_MultipleVolume_unix -Policy
db2ManualPolicy
```

6. View the status of the backup job by using the Get-SmBackupReport cmdlet.

   This example displays a job summary report of all jobs that were run on the specified date:

```
PS C:\> Get-SmBackupReport -JobId 351
Output:
BackedUpObjects           : {DB1}
FailedObjects             : {}
IsScheduled               : False
HasMetadata               : False
SmBackupId                : 269
SmJobId                   : 2361
StartDateTime             : 10/4/2016 11:20:45 PM
EndDateTime               : 10/4/2016 11:21:32 PM
Duration                  : 00:00:46.2536470
CreatedDateTime           : 10/4/2016 11:21:09 PM
Status                    : Completed
ProtectionGroupName       : Verify_ASUP_Message_windows
SmProtectionGroupId       : 211
PolicyName                : test2
SmPolicyId                : 20
BackupName                : Verify_ASUP_Message_windows_scc54_10-04-
2016_23.20.46.2758
VerificationStatus        : NotVerified
VerificationStatuses      :
SmJobError                :
BackupType                : SCC_BACKUP
CatalogingStatus          : NotApplicable
CatalogingStatuses        :
ReportDataCreatedDateTime :
```

Monitor custom plug-in resources backup operations

You can monitor the progress of different backup operations by using the SnapCenterJobs page. You might want to check the progress 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 corresponding state of the operations:

- ▶ In progress
- ✓ Completed successfully
- ✖ Failed
- ⚠ Completed with warnings or could not start due to warnings
- ↺ Queued
- ⊘ Canceled

Steps

1. In the left navigation pane, click **Monitor**.
2. In the Monitor page, click **Jobs**.
3. In the Jobs page, perform the following steps:
    a. Click ▼ to filter the list so that only backup operations are listed.
    b. Specify the start and end dates.
    c. From the **Type** drop-down list, select **Backup**.
    d. From the **Status** drop-down, select the backup status.
    e. Click **Apply** to view the operations completed successfully.
4. Select a backup job, and then click **Details** to view the job details.

    > ⓘ Though the backup job status displays ✓ , when you click on job details you might see that some of the child tasks of the backup operation are still in progress or marked with warning signs.

5. In the Job Details page, click **View logs**.

    The **View logs** button displays the detailed logs for the selected operation.

Cancel backup operations for custom plug-ins

You can cancel backup operations that are queued.

What you will need

- You must be logged in as the SnapCenter Admin or job owner to cancel operations.
- You can cancel a backup operation from either the **Monitor** page or the **Activity** pane.
- You cannot cancel a running backup operation.
- You can use the SnapCenter GUI, PowerShell cmdlets, or CLI commands to cancel the backup operations.
- The **Cancel Job** button is disabled for operations that cannot be canceled.
- If you selected **All members of this role can see and operate on other members objects** in Users\Groups page while creating a role, you can cancel the queued backup operations of other members while using that role.

Steps

1. Perform one of the following actions:

From the…	Action
Monitor page	a. In the left navigation pane, click **Monitor** > **Jobs**. b. Select the operation, and then click **Cancel Job**.
Activity pane	a. After initiating the backup operation, click ⌃ on the Activity pane to view the five most recent operations. b. Select the operation. c. In the Job Details page, click **Cancel Job**.

The operation is canceled, and the resource is reverted to the previous state.

View custom plug-in resource related backups and clones in the Topology page

When you are preparing to back up or clone a resource, you might find it helpful to view a graphical representation of all backups and clones on the primary and secondary storage. In the Topology page, you can see all of the backups and clones that are available for the selected resource or resource group. You can view the details of those backups and clones, and then select them to perform data protection operations.

About this task

You can review the following icons in the Manage Copies view to determine whether the backups and clones are available on the primary or secondary storage (Mirror copies or Vault copies).

- displays the number of backups and clones that are available on the primary storage.
- displays the number of backups and clones that are mirrored on the secondary storage using SnapMirror technology.

  > ⓘ Clones of a backup of a version-flexible mirror on a mirror-vault type volume are displayed in the topology view but the mirror backup count in the topology view does not include the version-flexible backup.

- displays the number of backups and clones that are replicated on the secondary storage using SnapVault technology.

  The number of backups displayed includes the backups deleted from the secondary storage. For example, if you have created 6 backups using a policy to retain only 4 backups, the number of backups displayed are 6.

> Clones of a backup of a version-flexible mirror on a mirror-vault type volume are displayed in the topology view but the mirror backup count in the topology view does not include the version-flexible backup.

Steps

1. In the left navigation pane, click **Resources**, and then select the appropriate plug-in from the list.
2. In the Resources page, either select the resource or resource group from the **View** drop-down list.
3. Select the resource either from the resource details view or from the resource group details view.

   If the resource is protected, the topology page of the selected resource is displayed.

4. Review the Summary card to see a summary of the number of backups and clones available on the primary and secondary storage.

   The Summary Card section displays the total number of backups and clones.

   Clicking the refresh button starts a query of the storage to display an accurate count.

5. In the Manage Copies view, click **Backups** or **Clones** from the primary or secondary storage to see details of a backup or clone.

   The details of the backups and clones are displayed in a table format.

6. Select the backup from the table, and then click the data protection icons to perform restore, clone, rename, and delete operations.

   > You cannot rename or delete backups that are on the secondary storage system.

   > You cannot rename the backups that are on the primary storage system.

7. If you want to delete a clone, then select the clone from the table and click 🗑 to delete the clone.

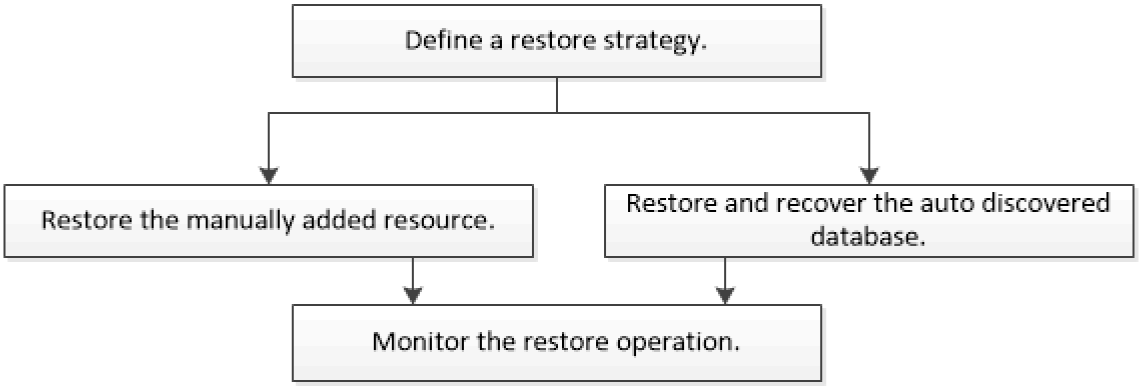
Restore custom plug-in resources

Restore custom plug-in resources

The restore and recovery workflow includes planning, performing the restore operations, and monitoring the operations.

About this task

The following workflow shows the sequence in which you must perform the restore operation:


Define a restore strategy.
Restore the manually added resource.
Restore and recover the auto discovered database.
Monitor the restore operation.

You can also use PowerShell cmdlets manually or in scripts to perform backup, restore, and clone operations. For information about PowerShell cmdlets, use the SnapCenter cmdlet help or see the SnapCenter Software Cmdlet Reference Guide.

Restore a resource backup

You can use SnapCenter to restore resources. The capabilities of the restore operations depends upon the plug-in that you use.

What you will need

- You must have backed up the resource or resource groups.
- The SnapCenter administrator must have assigned you the storage virtual machines (SVMs) for both the source volumes and destination volumes if you are replicating Snapshot copies to a mirror or vault.
- You must have cancelled any backup operation that is currently in progress for the resource or resource group you want to restore.

About this task

The default restore operation only restores storage objects. Restore operations at the application level can only be performed if the custom plug-in provides that capability.

Steps

1. In the left navigation pane, click **Resources**, and then select the appropriate plug-in from the list.
2. In the Resources page, filter resources from the **View** drop-down list based on resource type.

   The resources are displayed along with information such as type, host or cluster name, associated resource groups and policies, and status.

   > ⓘ Although a backup might be for a resource group, when you restore, you must select the individual resources you want to restore.

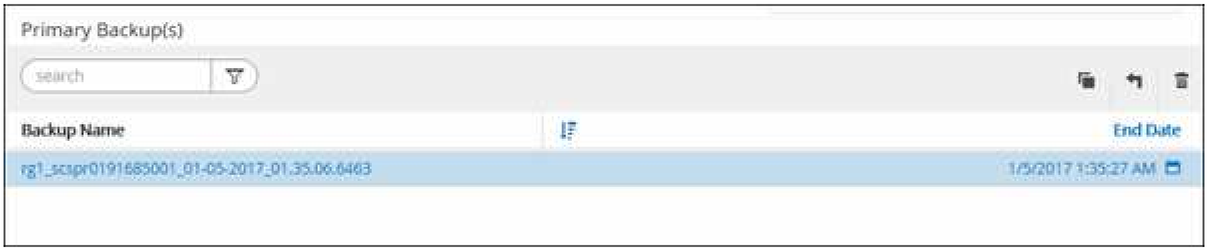
   If the resource is not protected, *Not protected* is displayed in the **Overall Status** column.

   The status *Not protected* in the **Overall Status** column can mean either that the resource is not protected, or that the resource was backed up by a different user.

3. Select the resource or select a resource group and then select a resource in that group.

The resource topology page is displayed.

4. From the **Manage Copies** view, select **Backups** either from the primary or secondary (mirrored or vaulted) storage systems.
5. In the Primary backup(s) table, select the backup that you want to restore from, and then click ↰.



Primary Backup(s)

Backup Name	End Date
rg1_scspr0191685001_01-05-2017_01.35.06.6463	1/5/2017 1:35:27 AM

6. In the Restore Scope page, select either **Complete Resource** or **File Level**.
   a. If you selected **Complete Resource**, the resource backup is restored.

      If the resource contains volumes or qtrees as Storage Footprint, then newer Snapshot copies on such volumes or qtrees are deleted and cannot be recovered. Also, if any other resource is hosted on same volumes or qtrees, then that resource is also deleted.

   b. If you selected **File Level**, then you can either select **All**, or select volumes or qtrees, and then enter the path related to the volumes or qtrees that are selected separated by commas.
      - You can select multiple volumes and qtrees.
      - If resource type is LUN, entire LUN is restored. You can select multiple LUNs.
        NOTE: If you select **All**, all the files on the volumes, qtrees, or LUNs are restored.
7. In the Recovery Type page, perform the following steps: select option to apply logs. Make sure your plugin supports All logs and Logs until restore type before selecting it.

If you want to…	Do this…
Restore all logs	Select **All logs**. Ensure that the plug-in supports **All logs**.
Restore all logs till the specified time	Select **Logs until**. Ensure that the plug-in supports **Logs until**.
Restore the resource backup	Select **None**.

8. In the Pre ops page, enter pre restore and unmount commands to run before performing a restore job.
9. In the Post ops page, enter mount and post restore commands to run after performing a restore job.
10. In the Notification page, from the **Email preference** drop-down list, select the scenarios in which you want to send the emails.

    You must also specify the sender and receiver email addresses, and the subject of the email. SMTP must also be configured in the **Settings** > **Global Settings** page.

11. Review the summary, and then click **Finish**.

12. Monitor the operation progress by clicking **Monitor** > **Jobs**.

Restore resources using PowerShell cmdlets

Restoring a resource backup includes initiating a connection session with the SnapCenter Server, listing the backups and retrieving backup information, and restoring a backup.

You must have prepared the PowerShell environment to execute the PowerShell cmdlets.

Steps

1. Initiate a connection session with the SnapCenter Server for a specified user by using the Open-SmConnection cmdlet.

   ```
   Open-smconnection  -SMSbaseurl  https:\\snapctr.demo.netapp.com:8146/
   ```

2. Retrieve the information about the one or more backups that you want to restore by using the Get-SmBackup and Get-SmBackupReport cmdlets.

   This example displays information about all available backups:

   ```
    C:\PS>PS C:\> Get-SmBackup

   BackupId                      BackupName                    BackupTime
   BackupType
   --------                      ----------                    ----------
   ----------
     1               Payroll Dataset_vise-f6_08... 8/4/2015    11:02:32 AM
   Full Backup
     2               Payroll Dataset_vise-f6_08... 8/4/2015    11:23:17 AM
   ```

   This example displays detailed information about the backup from January 29th 2015 to February 3rd, 2015:

```
PS C:\> Get-SmBackupReport -FromDateTime "1/29/2015" -ToDateTime
"2/3/2015"

   SmBackupId          : 113
   SmJobId             : 2032
   StartDateTime       : 2/2/2015 6:57:03 AM
   EndDateTime         : 2/2/2015 6:57:11 AM
   Duration            : 00:00:07.3060000
   CreatedDateTime     : 2/2/2015 6:57:23 AM
   Status              : Completed
   ProtectionGroupName : Clone
   SmProtectionGroupId : 34
   PolicyName          : Vault
   SmPolicyId          : 18
   BackupName          : Clone_SCSPR0019366001_02-02-2015_06.57.08
   VerificationStatus  : NotVerified

   SmBackupId          : 114
   SmJobId             : 2183
   StartDateTime       : 2/2/2015 1:02:41 PM
   EndDateTime         : 2/2/2015 1:02:38 PM
   Duration            : -00:00:03.2300000
   CreatedDateTime     : 2/2/2015 1:02:53 PM
   Status              : Completed
   ProtectionGroupName : Clone
   SmProtectionGroupId : 34
   PolicyName          : Vault
   SmPolicyId          : 18
   BackupName          : Clone_SCSPR0019366001_02-02-2015_13.02.45
   VerificationStatus  : NotVerified
```

3. Restore data from the backup by using the Restore-SmBackup cmdlet.

```
Restore-SmBackup -PluginCode 'DummyPlugin' -AppObjectId
'scc54.sccore.test.com\DummyPlugin\NTP\DB1' -BackupId 269
-Confirm:$false
output:
Name                  : Restore
'scc54.sccore.test.com\DummyPlugin\NTP\DB1'
Id                    : 2368
StartTime             : 10/4/2016 11:22:02 PM
EndTime               :
IsCancellable         : False
IsRestartable         : False
IsCompleted           : False
IsVisible             : True
IsScheduled           : False
PercentageCompleted : 0
Description           :
Status                : Queued
Owner                 :
Error                 :
Priority              : None
Tasks                 : {}
ParentJobID           : 0
EventId               : 0
JobTypeId             :
ApisJobKey            :
ObjectId              : 0
PluginCode            : NONE
PluginName            :
```

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.

Monitor custom plug-in resources restore operations

You can monitor the progress of different SnapCenter restore operations by using the Jobs page. You might want to check the progress of an operation to determine when it is complete or if there is an issue.

About this task

Post-restore states describe the conditions of the resource after a restore operation and any further restore actions that you can take.

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
- ⊘ Canceled

Steps

1. In the left navigation pane, click **Monitor**.
2. In the Monitor page, click **Jobs**.
3. In the Jobs page, perform the following steps:
   a. Click ▼ to filter the list so that only restore operations are listed.
   b. Specify the start and end dates.
   c. From the **Type** drop-down list, select **Restore**.
   d. From the **Status** drop-down list, select the restore status.
   e. Click **Apply** to view the operations that have been completed successfully.
4. Select the restore job, and then click **Details** to view the job details.
5. In the Job Details page, click **View logs**.

   The **View logs** button displays the detailed logs for the selected operation.

> ⓘ After the volume based restore operation, the backup metadata is deleted from the SnapCenter repository but the backup catalog entries remain in SAP HANA catalog. Though the restore job status displays ✓ , you should click on job details to see the warning sign of some of the child tasks. Click on the warning sign and delete the indicated backup catalog entries.

Clone custom plug-in resource backups

Clone custom plug-in resource backups

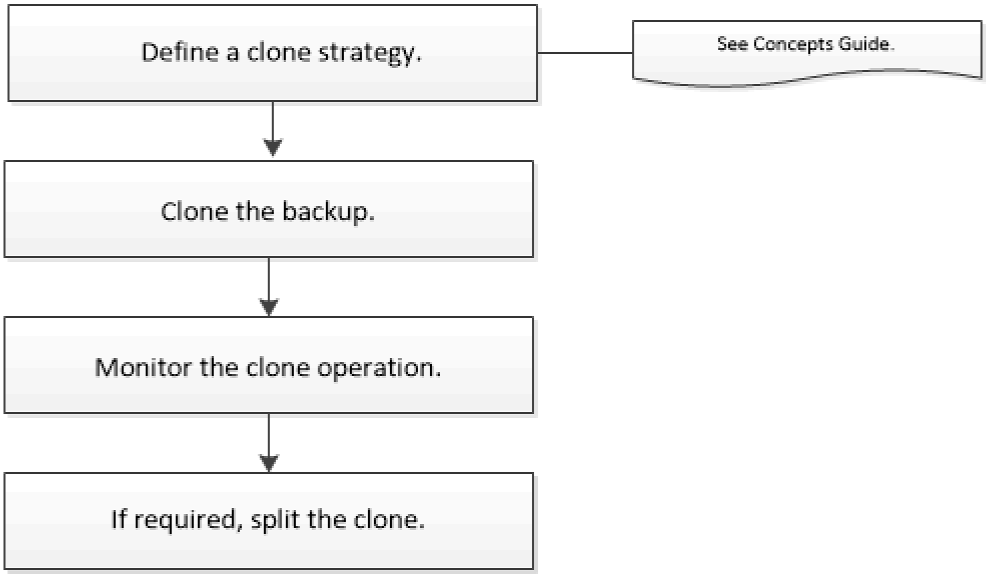
The clone workflow includes performing the clone operation and monitoring the operation.

About this task

You might clone resource backups for the following reasons:

- To test functionality that has to be implemented using the current resource structure and content during application development cycles
- For data extraction and manipulation tools when populating data warehouses
- To recover data that was mistakenly deleted or changed

The following workflow shows the sequence in which you must perform the clone operation:



You can also use PowerShell cmdlets manually or in scripts to perform backup, restore, and clone operations. For detailed information about PowerShell cmdlets, use the SnapCenter cmdlet help or see the SnapCenter Software Cmdlet Reference Guide.

Clone from a backup

You can use SnapCenter to clone a backup. You can clone from primary or secondary backup. The capabilities of the clone operations depends upon the plug-in that you use.

What you will need

- You must have backed up the resources or resource group.
- The default clone operation only clones storage objects. Clone operations at the application level can only be performed if the custom plug-in provides that capability.
- You should ensure that the aggregates hosting the volumes should be in the assigned aggregates list of the storage virtual machine (SVM).

Steps

1. In the left navigation pane, click **Resources**, and then select the appropriate plug-in from the list.
2. In the Resources page, filter resources from the **View** drop-down list based on resource type.

   The resources are displayed along with information such as type, host or cluster name, associated resource groups and policies, and status.

3. Select the resource or resource group.

   You must select a resource if you select a resource group.

   The resource or resource group topology page is displayed.

4. From the Manage Copies view, select **Backups** either from the primary or secondary (mirrored or vaulted) storage systems.
5. Select the data backup from the table, and then click ⧉.

6. In the Locations page, perform the following:

For this field…	Do this…
Clone server	By default, the source host is populated. If you want to specify a different host, select the host on which the clone should be mounted and the plug-in is installed.
Clone suffix	This is mandatory when the clone destination is the same as the source. Enter a suffix that will be appended to the newly cloned resource name. The suffix ensures that the cloned resource is unique on the host. For example, rs1_clone. If you are cloning to the same host as the original resource, you must provide a suffix to differentiate the cloned resource from the original resource; otherwise, the operation fails.

   If the resource selected is a LUN and if you are cloning from a secondary backup, then the destination volumes are listed. Single source can have multiple destination volumes.

7. In the Settings page, perform the following:

For this field…	Do this…
Initiator name	Enter the host initiator name, which is either a IQDN or WWPN.
Igroup protocol	Select Igroup protocol.

   ⓘ Settings page is displayed only if the storage type is LUN.

8. In the Scripts page, enter the commands for pre clone or post clone that should be run before or after the clone operation, respectively. Enter the mount command to mount a file system to a host.

   For example:

   - Pre clone command: delete existing databases with the same name
   - Post clone command: verify a database or start a database.

     Mount command for a volume or qtree on a Linux machine: mount<VSERVER_NAME>:%<VOLUME_NAME_Clone /mnt>

9. In the Notification page, from the **Email preference** drop-down list, select the scenarios in which you want to send the emails.

   You must also specify the sender and receiver email addresses, and the subject of the email.

10. Review the summary and click **Finish**.
11. Monitor the operation progress by clicking **Monitor** > **Jobs**.

Clone backups using PowerShell cmdlets

The clone workflow includes planning, performing the clone operation, and monitoring the operation.

What you will need

You must have prepared the PowerShell environment to execute the PowerShell cmdlets.

For information about PowerShell cmdlets, use the SnapCenter cmdlet help or see the SnapCenter Software Cmdlet Reference Guide.

Steps

1. Initiate a connection session with the SnapCenter Server for a specified user by using the Open-SmConnection cmdlet.

```
Open-SmConnection  -SMSbaseurl  https:\\snapctr.demo.netapp.com:8146/
```

2. List the backups that can be cloned using the Get-SmBackup or Get-SmResourceGroup cmdlet.

   This example displays information about all available backups:

```
 C:\PS>PS C:\> Get-SmBackup

BackupId                      BackupName                    BackupTime
BackupType
--------                      ----------                    ----------
----------
   1              Payroll Dataset_vise-f6_08... 8/4/2015    11:02:32 AM
Full Backup
   2              Payroll Dataset_vise-f6_08... 8/4/2015    11:23:17 AM
```

   This example displays information about a specified resource group:

```
PS C:\> Get-SmResourceGroup

    Description                 :
    CreationTime                : 10/10/2016 4:45:53 PM
    ModificationTime            : 10/10/2016 4:45:53 PM
    EnableEmail                 : False
    EmailSMTPServer             :
    EmailFrom                   :
```

```
    EmailTo                     :
    EmailSubject                :
    EnableSysLog                : False
    ProtectionGroupType         : Backup
    EnableAsupOnFailure         : False
    Policies                    : {}
    HostResourceMaping          : {}
    Configuration               : SMCoreContracts.SmCloneConfiguration
    LastBackupStatus            : Completed
    VerificationServer          :
    EmailBody                   :
    EmailNotificationPreference : Never
    VerificationServerInfo      :
    SchedulerSQLInstance        :
    CustomText                  :
    CustomSnapshotFormat        :
    SearchResources             : False
    ByPassCredential            : False
    IsCustomSnapshot            :
    MaintenanceStatus           : Production
    PluginProtectionGroupTypes  : {SMSQL}
    Tag                         :
    IsInternal                  : False
    EnableEmailAttachment       : False
    VerificationSettings        : {}
    Name                        : NFS_DB
    Type                        : Group
    Id                          : 2
    Host                        :
    UserName                    :
    Passphrase                  :
    Deleted                     : False
    Auth                        : SMCoreContracts.SmAuth
    IsClone                     : False
    CloneLevel                  : 0
    Hosts                       :
    StorageName                 :
    ResourceGroupNames          :
    PolicyNames                 :

    Description                 :
    CreationTime                : 10/10/2016 4:51:36 PM
    ModificationTime            : 10/10/2016 5:27:57 PM
    EnableEmail                 : False
    EmailSMTPServer             :
    EmailFrom                   :
```

```
    EmailTo                     :
    EmailSubject                :
    EnableSysLog                : False
    ProtectionGroupType         : Backup
    EnableAsupOnFailure         : False
    Policies                    : {}
    HostResourceMaping          : {}
    Configuration               : SMCoreContracts.SmCloneConfiguration
    LastBackupStatus            : Failed
    VerificationServer          :
    EmailBody                   :
    EmailNotificationPreference : Never
    VerificationServerInfo      :
    SchedulerSQLInstance        :
    CustomText                  :
    CustomSnapshotFormat        :
    SearchResources             : False
    ByPassRunAs                 : False
    IsCustomSnapshot            :
    MaintenanceStatus           : Production
    PluginProtectionGroupTypes  : {SMSQL}
    Tag                         :
    IsInternal                  : False
    EnableEmailAttachment       : False
    VerificationSettings        : {}
    Name                        : Test
    Type                        : Group
    Id                          : 3
    Host                        :
    UserName                    :
    Passphrase                  :
    Deleted                     : False
    Auth                        : SMCoreContracts.SmAuth
    IsClone                     : False
    CloneLevel                  : 0
    Hosts                       :
    StorageName                 :
    ResourceGroupNames          :
    PolicyNames                 :
```

3. Initiate a clone operation from a clone resource group or an existing backup using the New-SmClone cmdlet.

   This example creates a clone from a specified backup with all logs:

```
New-SmClone -BackupName Verify_delete_clone_on_qtree_windows_scc54_10-
04-2016_19.05.48.0886 -Resources
@{"Host"="scc54.sccore.test.com";"Uid"="QTREE1"}  -
CloneToInstance scc54.sccore.test.com -Suffix '_QtreeCloneWin9'
-AutoAssignMountPoint -AppPluginCode 'DummyPlugin' -initiatorname
'iqn.1991-
05.com.microsoft:scc54.sccore.test.com' -igroupprotocol 'mixed'
```

4. View the status of the clone job by using the Get-SmCloneReport cmdlet.

   This example displays a clone report for the specified job ID:

```
PS C:\> Get-SmCloneReport -JobId 186

    SmCloneId           : 1
    SmJobId             : 186
    StartDateTime       : 8/3/2015 2:43:02 PM
    EndDateTime         : 8/3/2015 2:44:08 PM
    Duration            : 00:01:06.6760000
    Status              : Completed
    ProtectionGroupName : Draper
    SmProtectionGroupId : 4
    PolicyName          : OnDemand_Clone
    SmPolicyId          : 4
    BackupPolicyName    : OnDemand_Full_Log
    SmBackupPolicyId    : 1
    CloneHostName       : SCSPR0054212005.mycompany.com
    CloneHostId         : 4
    CloneName           : Draper__clone__08-03-2015_14.43.53
    SourceResources     : {Don, Betty, Bobby, Sally}
    ClonedResources     : {Don_DRAPER, Betty_DRAPER, Bobby_DRAPER,
Sally_DRAPER}
    SmJobError          :
```

Monitor custom plug-in resource clone operations

You can monitor the progress of SnapCenter clone operations by using the Jobs page. You might want to check the progress of an operation 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
- ⊘ Canceled

Steps

1. In the left navigation pane, click **Monitor**.
2. In the Monitor page, click **Jobs**.
3. In the Jobs page, perform the following steps:
   a. Click ▼ to filter the list so that only clone operations are listed.
   b. Specify the start and end dates.
   c. From the **Type** drop-down list, select **Clone**.
   d. From the **Status** drop-down list, select the clone status.
   e. Click **Apply** to view the operations that are completed successfully.
4. Select the clone job, and then click **Details** to view the job details.
5. In the Job Details page, click **View logs**.

Copyright information

Trademark information