



# 保护自定义应用程序 SnapCenter Software 5.0

NetApp  
April 30, 2024

# 目录

保护自定义应用程序 .....	1
SnapCenter 自定义插件 .....	1
为您的应用程序开发一个插件 .....	7
准备安装 SnapCenter 自定义插件 .....	29
准备数据保护 .....	50
备份自定义插件资源 .....	51
还原自定义插件资源 .....	69
克隆自定义插件资源备份 .....	74

# 保护自定义应用程序

## SnapCenter 自定义插件

### SnapCenter 自定义插件概述

您可以为使用的应用程序开发自定义插件，然后使用 SnapCenter 备份，还原或克隆这些应用程序。与其他 SnapCenter 插件一样，您的自定义插件也充当 NetApp SnapCenter 软件的主机端组件，从而实现应用程序感知型数据保护和资源管理。

安装自定义插件后，您可以使用采用 NetApp SnapMirror 技术的 SnapCenter 在另一个卷上创建备份集的镜像副本，并使用 NetApp SnapVault 技术执行磁盘到磁盘备份复制。自定义插件既可以在 Windows 环境中使用，也可以在 Linux 环境中使用。



SnapCenterCLI 不支持 SnapCenter 自定义插件命令。

NetApp 提供了存储插件，可使用 SnapCenter 中内置的自定义插件框架对 ONTAP 存储上的数据卷执行数据保护操作。

您可以从添加主机页面安装自定义插件和存储插件。

["添加主机并在远程主机上安装插件软件包。"](#)

NetApp 还提供 MySQL，MaxDB，DB2，Sybase，DPGLUE，MongoDB，ORASCPM 和 PostgreSQL 自定义插件。这些插件可从下载 ["NetApp 存储自动化商店"](#)。



SnapCenter 支持策略将涵盖对 SnapCenter 自定义插件框架，核心引擎和关联 API 的支持。支持范围不包括插件源代码以及基于自定义插件框架构建的关联脚本。

您可以参考创建自己的自定义插件 ["为您的应用程序开发一个插件"](#)。

### 您可以使用 SnapCenter 自定义插件和存储插件执行什么操作

您可以使用 SnapCenter 自定义插件执行数据保护操作。

- 自定义插件 \*
- 添加数据库，实例，文档或表空间等资源。
- 创建备份
- 从备份还原
- 克隆备份。
- 计划备份操作。
- 监控备份，还原和克隆操作。
- 查看备份，还原和克隆操作的报告。
- 存储插件 \*

您可以使用存储插件执行数据保护操作。

- 为ONTAP集群中的存储卷创建一致性组快照。
- 使用内置的脚本编写前后框架备份自定义应用程序

您可以备份 ONTAP 卷， LUN 或 qtree 。

- 使用SnapCenter策略利用现有复制关系(SnapVault或SnapMirror)将主系统上创建的快照更新到ONTAP二级系统

ONTAP主存储系统和二级存储系统可以是ONTAP FAS、AFF、全SAN阵列(ASA)、Select或Cloud ONTAP。

- 恢复完整的 ONTAP 卷， LUN 或文件。

由于产品中未内置浏览或索引功能，您应手动提供相应的文件路径。

不支持 qtree 或目录还原，但如果在 qtree 级别定义了备份范围，则只能克隆和导出 qtree 。

## SnapCenter 自定义插件功能

SnapCenter 可与插件应用程序以及存储系统上的 NetApp 技术集成。要使用自定义插件，请使用 SnapCenter 图形用户界面。

- \* 统一图形用户界面 \*

SnapCenter 界面可在插件和环境之间实现标准化和一致性。通过 SnapCenter 界面，您可以跨插件完成一致的备份，还原，恢复和克隆操作，使用集中式报告，使用信息板概览视图，设置基于角色的访问控制（ Role-Based Access Control ， RBAC ）以及监控所有插件中的作业。

- \* 自动化中央管理 \*

您可以计划备份操作，配置基于策略的备份保留以及执行还原操作。您还可以通过配置 SnapCenter 以发送电子邮件警报来主动监控您的环境。

- 无中断**NetApp**快照技术

SnapCenter将NetApp快照技术与SnapCenter自定义插件结合使用来备份资源。快照占用的存储空间最少。

使用自定义插件功能还具有以下优势：

- 支持备份，还原和克隆 workflow
- 支持 RBAC 的安全性和集中式角色委派

您还可以设置凭据，以便授权的 SnapCenter 用户具有应用程序级别的权限。

- 使用 NetApp FlexClone 技术为测试或数据提取创建节省空间的时间点资源副本

要创建克隆的存储系统需要 FlexClone 许可证。

- 在创建备份时支持ONTAP的一致性组(CG) Snapshot功能。
- 能够在多个资源主机之间同时运行多个备份

在单个操作中、如果单个主机中的资源共享同一个卷、则会整合Snapshot。

- 能够使用外部命令创建Snapshot。
- 能够在Windows环境中创建文件系统一致的快照。

## SnapCenter 自定义插件支持的存储类型

SnapCenter 在物理机和虚拟机上均支持多种存储类型。在安装 SnapCenter 自定义插件之前，您必须验证是否支持您的存储类型。

计算机	Storage type
VM主机上的物理挂载和NFS直接挂载(不支持VMDK和RDM LUN。)	FC 连接的 LUN
VM主机上的物理挂载和NFS直接挂载(不支持VMDK和RDM LUN。)	iSCSI 连接的 LUN
VM主机上的物理挂载和NFS直接挂载(不支持VMDK和RDM LUN。)	NFS-connected volumes

## 自定义插件所需的最小 ONTAP 权限

所需的最小 ONTAP 权限因用于数据保护的 SnapCenter 插件而异。

- all-access 命令： ONTAP 8.3.0 及更高版本所需的最低权限
  - event generate-autosupport-log
  - 作业历史记录显示
  - 作业停止
  - lun attribute show
  - 创建 LUN
  - LUN 删除
  - LUN 几何结构
  - lun igroup add
  - lun igroup create
  - LUN igroup 删除
  - LUN igroup 重命名
  - lun igroup show
  - LUN 映射 add-reporting-nodes

- 创建 LUN 映射
- LUN 映射删除
- LUN 映射 remove-reporting-nodes
- lun mapping show
- 修改 LUN
- LUN 移动卷
- LUN 脱机
- LUN 联机
- 调整 LUN 大小
- LUN 串行
- lun show
- 网络接口
- SnapMirror 策略附加规则
- snapmirror policy modify-rule
- snapmirror policy remove-rule
- snapmirror policy show
- SnapMirror 还原
- snapmirror show
- snapmirror show-history
- SnapMirror 更新
- snapmirror update-ls-set
- snapmirror list-destinations
- version
- 创建卷克隆
- volume clone show
- 卷克隆拆分开始
- 卷克隆拆分停止
- 卷创建
- 卷销毁
- 创建卷文件克隆
- volume file show-disk-usage
- 卷脱机
- 卷联机
- 卷修改
- 卷 qtree create

- 卷 qtree 删除
- 卷 qtree 修改
- volume qtree show
- 卷限制
- volume show
- 创建卷快照
- 卷快照删除
- 卷快照修改
- 卷快照重命名
- 卷快照还原
- 卷快照还原文件
- volume snapshot show
- 卷卸载
- Vserver CIFS
- Vserver CIFS 共享 create
- SVM CIFS 共享删除
- vserver cifs ShadowCopy show
- vserver cifs share show
- vserver cifs show
- 创建 SVM 导出策略
- SVM 导出策略删除
- 创建 SVM 导出策略规则
- vserver export-policy rule show
- vserver export-policy show
- vserver iscsi connection show
- vserver show
- 只读命令： ONTAP 8.3.0 及更高版本所需的最低权限
  - 网络接口

## 准备用于 **SnapMirror** 的存储系统，以及用于自定义插件的 **SnapVault** 复制

您可以使用采用 ONTAP SnapMirror 技术的 SnapCenter 插件在另一个卷上为备份集创建镜像副本，并使用 ONTAP SnapVault 技术执行磁盘到磁盘备份复制，以满足标准和其他监管相关要求。在执行这些任务之前，您必须在源卷和目标卷之间配置数据保护关系并初始化此关系。

SnapCenter会在完成Snapshot操作后执行SnapMirror和SnapVault更新。SnapMirror和SnapVault更新是在SnapCenter作业中执行的；请勿创建单独的ONTAP计划。



如果您是从 NetApp SnapManager 产品访问 SnapCenter 的，并且对您配置的数据保护关系感到满意，则可以跳过本节。

数据保护关系会将主存储（源卷）上的数据复制到二级存储（目标卷）。初始化此关系时，ONTAP 会将源卷上引用的数据块传输到目标卷。



SnapCenter 不支持 SnapMirror 和 SnapVault 卷之间的级联关系（\* 主卷 \* > \* 镜像 \* > \* 存储 \*）。您应使用扇出关系。

SnapCenter 支持管理版本灵活的 SnapMirror 关系。有关版本灵活的 SnapMirror 关系以及如何设置这些关系的详细信息，请参见 ["ONTAP 文档"](#)。



SnapCenter 不支持 \* 同步镜像 \* 复制。

## 定义备份策略

在创建备份作业之前定义备份策略可确保您拥有成功还原或克隆资源所需的备份。您的服务级别协议（SLA），恢复时间目标（RTO）和恢复点目标（RPO）在很大程度上决定了您的备份策略。

### 关于此任务

SLA 定义了预期的服务级别，并解决了许多与服务相关的问题，包括服务的可用性和性能。RTO 是指服务中断后必须恢复业务流程的时间。RPO 定义了必须从备份存储中恢复的文件的使用期限策略，以便在发生故障后恢复常规操作。SLA，RTO 和 RPO 有助于实施数据保护策略。

### 步骤

1. 确定应在何时备份资源。
2. 确定需要多少备份作业。
3. 确定如何为备份命名。
4. 确定是否需要一致性组快照、并确定删除一致性组快照的适当选项。
5. 确定是要使用 NetApp SnapMirror 技术进行复制，还是使用 NetApp SnapVault 技术进行长期保留。
6. 确定源存储系统和 SnapMirror 目标上的 Snapshot 的保留期限。
7. 确定是否要在备份操作之前或之后运行任何命令，并提供预处理或后处理。

## 自定义插件的备份策略

### 自定义插件资源的备份计划

确定备份计划的最关键因素是资源的更改率。备份资源的频率越高，SnapCenter 用于还原的归档日志就越少，从而可以加快还原操作的速度。

您可以每小时备份一次使用率较高的资源，而每天备份一次很少使用的资源。其他因素包括资源对组织的重要性，服务级别协议（SLA）和恢复点目标（RPO）。

SLA 定义了预期的服务级别，并解决了许多与服务相关的问题，包括服务的可用性和性能。RPO 定义了必须从



备份存储中恢复的文件的使用期限策略，以便在发生故障后恢复常规操作。SLA 和 RPO 有助于制定数据保护策略。

备份计划分为两部分，如下所示：

- 备份频率

备份频率（执行备份的频率）也称为某些插件的计划类型，属于策略配置的一部分。例如，您可以将备份频率配置为每小时，每天，每周或每月。您可以通过单击 \* 设置 \* > \* 策略 \* 来访问 SnapCenter 图形用户界面中的策略。

- 备份计划

备份计划（恰好是在执行备份时）属于资源或资源组配置。例如，如果您的资源组配置了每周备份的策略，则可以将该计划配置为每星期四晚上 10：00 进行备份您可以在 SnapCenter 图形用户界面中访问资源组计划，方法是单击 \* 资源 \*，选择相应的插件并单击 \* 视图 \* > \* 资源组 \*。

所需备份作业的数量

决定所需备份作业数量的因素包括资源大小，已用卷数，资源更改率以及您的服务级别协议（Service Level Agreement，SLA）。

您选择的备份作业数量通常取决于您放置资源的卷数量。例如，如果您将一组小型资源放在一个卷上，而将一个大型资源放在另一个卷上，则可以为小型资源创建一个备份作业，并为大型资源创建一个备份作业。

手动添加的自定义插件资源支持的还原策略类型

您必须先定义策略，然后才能使用 SnapCenter 成功执行还原操作。手动添加的自定义插件资源有两种还原策略。



您无法恢复手动添加的自定义插件资源。

完成资源还原

- 还原资源的所有卷，qtree 和 LUN



如果此资源包含卷或qtrees、则在为此类卷或qtrees选择要还原的Snapshot之后创建的Snapshot将被删除、并且无法恢复。此外，如果任何其他资源托管在同一个卷或 qtree 上，则该资源也会被删除。

文件级还原

- 从卷，qtree 或目录还原文件
- 仅还原选定的 LUN

## 为您的应用程序开发一个插件

## 概述

通过 SnapCenter 服务器，您可以将应用程序作为 SnapCenter 的插件进行部署和管理。您选择的应用程序可以插入到 SnapCenter 服务器中，以实现数据保护和管理功能。

通过 SnapCenter，您可以使用不同的编程语言开发自定义插件。您可以使用 Perl，Java，批处理或其他脚本语言开发自定义插件。

要在 SnapCenter 中使用自定义插件，必须执行以下任务：

- 按照本指南中的说明为您的应用程序创建一个插件
- 创建问题描述文件
- 导出自定义插件以将其安装在 SnapCenter 主机上
- 将此插件 zip 文件上传到 SnapCenter 服务器

所有 API 调用中的通用插件处理

对于每个 API 调用，请使用以下信息：

- 插件参数
- 退出代码
- 记录错误消息
- 数据一致性

使用插件参数

在每次进行 API 调用时，都会向插件传递一组参数。下表列出了参数的特定信息。

参数	目的
Action	确定工作流名称。例如，discover，backup，fileOrVolRestore 或 cloneVolAndLun
Resources	列出要保护的资源。资源由 UID 和类型标识。此列表将按以下格式显示给此插件：  "<UID>， <type> ; <UID>， <type>"。例如， "实例 1， 实例；实例 2\\DB1， 数据库"
应用程序名称	确定正在使用的插件。例如 DB2， MySQL。 SnapCenter 服务器内置了对所列应用程序的支持。此参数区分大小写。
app_ignore_error	(Y 或 N) 如果遇到应用程序错误，则会导致 SnapCenter 退出或不退出。如果您要备份多个数据库，而不希望单个故障停止备份操作，则此功能非常有用。

参数	目的
<resource_name>_app_instance_username	已为资源设置 SnapCenter 凭据。
<resource_name>_app_instance_password	已为资源设置 SnapCenter 凭据。
<resource_name>_<custom_param>	每个资源级别自定义密钥值均可供插件使用，并以 "<resource_name>_" 开头。例如，如果名为 MySQLDB 的资源的自定义密钥为 "master_slave"，则该密钥将作为 MySQLDB_master_slave 可用

#### 使用退出代码

此插件通过退出代码将操作状态返回给主机。每个代码都有一个特定的含义，此插件使用正确的退出代码来指示相同的含义。

下表介绍了错误代码及其含义。

退出代码	目的
0	操作成功。
99	不支持或不实施请求的操作。
100	操作失败，跳过静默并退出。默认情况下，取消静默状态为。
101.	操作失败，请继续执行备份操作。
其他	操作失败，运行 unquiesce 并退出。

#### 记录错误消息

错误消息将从插件传递到 SnapCenter 服务器。此消息包括消息，日志级别和时间戳。

下表列出了级别及其用途。

参数	目的
信息	信息性消息
警告	警告消息
error	错误消息
调试	调试消息

参数	目的
跟踪	跟踪消息

保持数据一致性

自定义插件可在执行相同工作流的操作之间保留数据。例如，插件可以在暂停结束时存储数据，可在取消静默操作期间使用。

要保留的数据将通过插件作为结果对象的一部分进行设置。它采用特定格式，并在每种插件开发模式下进行详细介绍。

## 基于 **Perl** 的开发

在使用 Perl 开发插件时，必须遵循某些约定。

- 内容必须可读
- 必须实施强制操作 `setenv`，`quiesce` 和 `unquiesce`
- 必须使用特定语法将结果传递回代理
- 这些内容应保存为 `<plugin_name>.pm` 文件

可用操作包括

- 设置
- `version`
- 暂停
- 取消静默
- `clone_pre`，`clone_post`
- `restore_pre`，`restore`
- 清理

常规插件处理

使用 **Results** 对象

每个自定义插件操作都必须定义结果对象。此对象会将消息，退出代码，`stdout` 和 `stderr` 发送回主机代理。

Results 对象：

```
my $result = {
```

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

返回结果对象：

```
return $result;
```

保持数据一致性

在执行同一工作流期间，可以在操作之间保留数据（清理除外）。这可通过密钥值对来实现。关键值数据对作为结果对象的一部分进行设置，并在同一工作流的后续操作中保留和使用。

以下代码示例将设置要保留的数据：

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

上述代码设置了两个键值对，这些键值对可在后续操作中用作输入。可以使用以下代码访问这两个键值对：

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

=== Logging error messages

每个操作都可以将消息发送回主机代理，主机代理将显示和存储内容。消息包含消息级别，时间戳和消息文本。支持多行消息。

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

使用 msgObj 通过使用收集方法捕获消息。

```
$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");
```

将消息应用于结果对象：

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

使用插件存根

自定义插件必须公开插件存根。这些方法是 SnapCenter 服务器根据工作流调用的。

插件存根	可选 / 必需	目的
设置	Required	<p>此存根用于设置环境和配置对象。</p> <p>任何环境解析或处理都应在此处完成。每次调用存根时，都会在此之前调用 setenv 存根。只有 Perl 模式插件才需要此功能。</p>
version	可选	<p>此存根用于获取应用程序版本。</p>
发现	可选	<p>此存根用于发现代理或主机上托管的实例或数据库等应用程序对象。</p> <p>在响应过程中，此插件应以特定格式返回发现的应用程序对象。只有在应用程序与适用于 Unix 的 SnapDrive 集成时，才会使用此存根。</p> <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;">  <p>支持 Linux 文件系统（Linux Flavors）。不支持 AIX/Solaris（Unix 模式）。</p> </div>

插件存根	可选 / 必需	目的
discovery_complete	可选	<p>此存根用于发现代理或主机上托管的实例或数据库等应用程序对象。</p> <p>在响应过程中，此插件应以特定格式返回发现的应用程序对象。只有在应用程序与适用于 Unix 的 SnapDrive 集成时，才会使用此存根。</p> <div style="border: 1px solid gray; padding: 5px; margin-top: 10px;">  支持 Linux 文件系统（Linux Flavors）。不支持 AIX 和 Solaris（Unix 模式）。 </div>
暂停	Required	<p>此存根负责执行静息、这意味着将应用程序置于可以创建Snapshot的状态。这在执行Snapshot操作之前调用。要保留的应用程序元数据应设置为响应的一部分、响应应在随后对相应存储Snapshot执行克隆或还原操作时以配置参数的形式返回。</p>
取消静默	Required	<p>此存根负责执行静默，这意味着将应用程序置于正常状态。创建Snapshot后会调用此命令。</p>
clone_pre	可选	<p>此存根负责执行克隆前任务。此操作假定您使用的是内置的 SnapCenter 服务器克隆接口，并在执行克隆操作时触发。</p>
clone_post	可选	<p>此存根负责执行克隆后任务。这假定您使用的是内置的 SnapCenter 服务器克隆接口，并且只有在执行克隆操作时才会触发。</p>
restore_pre	可选	<p>此存根负责执行预存储任务。此操作假定您使用的是内置的 SnapCenter 服务器还原界面，并且是在执行还原操作时触发的。</p>
还原	可选	<p>此存根负责执行应用程序还原任务。这假定您使用的是内置的 SnapCenter 服务器还原界面，并且只有在执行还原操作时才会触发。</p>

插件存根	可选 / 必需	目的
清理	可选	此存根负责在执行备份，还原或克隆操作后执行清理。清理可以在正常工作流执行期间进行，也可以在工作流出现故障时进行。您可以通过引用配置参数操作来推断调用清理时使用的工作流名称，该操作可以是 backup ， cloneVolAndLun 或 fileOrVolRestore 。配置参数 error_message 用于指示执行工作流时是否存在任何错误。如果已定义 error_message ，而不是 NULL ，则在执行工作流失败期间会调用清理。
APP_VERSION	可选	SnapCenter 使用此存根来获取此插件管理的应用程序版本详细信息。

#### 插件软件包信息

每个插件都必须具有以下信息：

```

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 = ();

```

#### 操作

您可以对自定义插件支持的各种操作进行编码，例如 setenv ， Version ， Quiesce 和 Unquiesce 。



## setenv 操作

使用 Perl 创建的插件需要执行 setenv 操作。您可以设置 ENV 并轻松访问插件参数。

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

## 版本操作

版本操作将返回应用程序版本信息。

```
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;
}
```

## 暂停操作

暂停操作会对 Resources 参数中列出的资源执行应用程序暂停操作。

```

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;
}

```

## 取消暂停操作

要取消应用程序静默，需要执行 "取消暂停" 操作。资源列表位于 Resources 参数中。

```

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;
}

```

## 原生模式

SnapCenter 支持使用非 Perl 编程或脚本语言来创建插件。这称为原生模式编程，可以是脚本或批处理文件。

本机模式插件必须遵循以下特定约定：

此插件必须是可执行的

- 对于 Unix 系统，运行代理的用户必须对此插件具有执行权限

- 对于 Windows 系统， PowerShell 插件的后缀必须为 .ps1 ，而其他 Windows 脚本的后缀必须为 .cmd 或 .bat ，并且必须可由用户执行
- 插件必须对命令行参数做出响应，例如 "-quiesce" ， "-unquiesce"
- 如果未实施某项操作或功能，插件必须返回退出代码 99
- 插件必须使用特定语法将结果传递回服务器

## 常规插件处理

### 记录错误消息

每个操作都可以将消息发送回服务器，服务器将显示和存储内容。消息包含消息级别，时间戳和消息文本。支持多行消息。

格式。

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

### 使用插件存根

SnapCenter 插件必须实施插件存根。这些方法是 SnapCenter 服务器根据特定工作流调用的。

插件存根	可选 / 必需	目的
暂停	Required	此存根负责执行暂停。它将放置 将应用程序置于可以创建Snapshot的状态。这在执行存储Snapshot操作之前调用。
取消静默	Required	此存根负责执行静默。它的位置 处于正常状态的应用程序。这在存储后称为 Snapshot操作。
clone_pre	可选	此存根负责执行克隆前任务。此操作假定您使用的是内置的 SnapCenter 克隆接口，并且只有在执行操作 "clone_vol 或 clone_LUN" 时才会触发此接口。
clone_post	可选	此存根负责执行克隆后任务。这假定您使用的是内置的 SnapCenter 克隆接口，并且只有在执行 "clone_vol 或 clone_lun" 操作时才会触发此接口。

插件存根	可选 / 必需	目的
restore_pre	可选	此存根负责执行还原前任务。这假定您使用的是内置的 SnapCenter 还原界面，并且仅在执行还原操作时触发。
还原	可选	此存根负责执行所有还原操作。此操作假定您未使用内置还原界面。它会在执行还原操作时触发。

示例

### Windows PowerShell

检查是否可以在您的系统上执行此脚本。如果无法执行此脚本，请为此脚本设置 Set-ExecutionPolicy bypass ，然后重试此操作。

```

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 模式

Java 自定义插件直接与数据库，实例等应用程序交互。

### 限制

在使用 Java 编程语言开发插件时，您应注意一些限制。

插件特征	Java 插件
复杂性	从低到中

插件特征	Java 插件
内存占用空间	最多 10-20 MB
其他库的依赖关系	用于应用程序通信的库
线程数	1.
线程运行时	不到一小时

### Java 限制的原因

SnapCenter 代理的目标是确保持续，安全且稳定可靠的应用程序集成。通过支持 Java 插件，插件可能会导致内存泄漏和其他不必要的问题。这些问题很难解决，尤其是在目标是让事情简单易用的情况下。如果插件的复杂性不是太复杂，开发人员可能会出现错误的概率就会小得多。Java 插件的风险在于，它们与 SnapCenter 代理本身在同一 JVM 中运行。当插件崩溃或内存泄漏时，它也可能对代理产生负面影响。

### 支持的方法

方法	Required	Description	何时被调用？由谁调用？
version	是的。	需要返回插件的版本。	由 SnapCenter 服务器或代理请求插件的版本。
暂停	是的。	需要对应用程序执行暂停。在大多数情况下、这意味着将应用程序置于 SnapCenter 服务器可以创建备份(例如 Snapshot) 的状态。	在 SnapCenter 服务器创建 Snapshot 副本或执行常规备份之前。
取消静默	是的。	需要对应用程序执行静默操作。在大多数情况下，这意味着将应用程序重新置于正常运行状态。	在 SnapCenter 服务器创建快照或具有之后已执行常规备份。
清理	否	负责清理插件需要清理的任何内容。	SnapCenter 服务器上的工作流完成后（成功或出现故障）。
clonePre	否	应在执行克隆操作之前执行需要执行的操作。	当用户触发 "cloneVol" 或 "cloneLun" 操作并使用内置克隆向导（GUI/CLI）时。

方法	Required	Description	何时被调用？由谁调用？
clonePost	否	应在执行克隆操作后执行需要执行的操作。	当用户触发 "cloneVol" 或 "cloneLun" 操作并使用内置克隆向导（GUI/CLI）时。
还原前	否	应在调用还原操作之前执行需要执行的操作。	用户触发还原操作时。
还原	否	负责执行应用程序的还原 / 恢复。	用户触发还原操作时。
应用程序版本	否	以检索插件管理的应用程序版本。	作为备份 / 还原 / 克隆等每个工作流程中 ASUP 数据收集的一部分。

## 教程

本节介绍如何使用 Java 编程语言创建自定义插件。

### 设置 Eclipse

1. 在 Eclipse 中创建一个新的 Java 项目 "TutorialPlug"
2. 单击 \* 完成 \*。
3. 右键单击 \* 新项目 \* → \* 属性 \* → \* Java 构建路径 \* → \* 库 \* → \* 添加外部 JAR \*
4. 导航到主机代理的 ./lib/ 文件夹，然后选择 JAR scAgent-5.0-core.jar 和 common-5.0.jar
5. 选择项目并右键单击 \* 源文件夹 \* → \* 新增 \* → \* 软件包 \*，然后创建一个名为 com.netapp.snapcreator.agent.plugin.TutorialPlugin 的新软件包
6. 右键单击新软件包并选择新建 → Java 类。
  - a. 输入名称作为 TutorialPlugin。
  - b. 单击超类浏览按钮并搜索 "\* 抽象插件"。只应显示一个结果：

```
"AbstractPlugin - com.netapp.snapcreator.agent.nextgen.plugin".
.. 单击 * 完成 *。
.. Java 类：
```

```

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;
    }
}

```

实施所需的方法

暂停，取消静默和版本是每个自定义 Java 插件必须实施的强制方法。

以下是返回插件版本的版本方法。



```

@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 plugin 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;
}

```

方法在上下文对象中传递。其中包含多个帮助程序，例如 `Logger` 和上下文存储，以及有关当前操作的信息（工作流 ID，作业 ID）。我们可以通过调用 `final Logger logger = context.getLogger()` 来获取此日志程序。`logger` 对象提供了其他日志记录框架中已知的类似方法，例如，`login`。在 `result` 对象中，您还可以指定退出代码。在此示例中，返回零，因为没有问题描述。其他退出代码可以映射到不同的故障情形。

正在使用结果对象

result 对象包含以下参数：

参数	Default	Description
配置	空配置	此参数可用于将配置参数发送回服务器。它可以是插件要更新的参数。此更改是否实际反映在 SnapCenter 服务器的配置中取决于配置中的 APP_CONF_persistency=Y 或 N 参数。
ExitCode	0	指示操作的状态。"0" 表示操作已成功执行。其他值表示错误或警告。
标准输出	空列表	这可用于将 stdout 消息传输回 SnapCenter 服务器。
标准	空列表	这可用于将 stderr 消息传输回 SnapCenter 服务器。
消息	空列表	此列表包含插件要返回到服务器的所有消息。SnapCenter 服务器会在命令行界面或图形用户界面中显示这些消息。

SnapCenter 代理可提供构建程序 ("构建程序模式")。这使得使用它们变得非常简单：

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

例如，将退出代码设置为 0，为 stdout 和 stderr 设置列表，设置配置参数，并附加要发送回服务器的日志消息。如果您不需要所有参数，请仅发送所需的参数。由于每个参数都有一个默认值，因此，如果从以下代码中删除 .withExitCode (0)，则结果不会受到影响：

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

## 版本结果

VersionResult 会向 SnapCenter 服务器通知插件版本。由于它还会从结果继承，因此它包含 config ， exitCode ， stdout ， stderr 和 messages 参数。

参数	Default	Description
major	0	插件的主要版本字段。
次要	0	插件的次要版本字段。
patch	0	插件的修补程序版本字段。
build	0	此插件的 Build version 字段。

例如：

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

## 使用上下文对象

上下文对象提供了以下方法：

上下文方法	目的
字符串 getWorkflowId ( ) ；	返回 SnapCenter 服务器在当前工作流中使用的工作流 ID 。
config getConfig ( ) ；	返回正在从 SnapCenter 服务器发送到代理的配置。

## 工作流 ID

工作流 ID 是 SnapCenter 服务器用于引用特定正在运行的工作流的 ID 。

## 配置

此对象包含（大多数）用户可在 SnapCenter 服务器的配置中设置的参数。但是，由于安全原因，其中某些参数可能会在服务器端进行筛选。以下是有关如何访问 Config 并检索参数的示例：

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

现在，"/" myParameter " 包含从 SnapCenter 服务器上的配置中读取的参数如果配置参数密钥不存在，则它将返回空字符串（""）。

导出插件

要在 SnapCenter 主机上安装此插件，必须导出此插件。

在 Eclipse 中，执行以下任务：

1. 右键单击插件的基础软件包（在我们的示例 com.netapp.snapcreator.agent.plugin.TutorialPlugin 中）。
2. 选择 \* 导出 \* → \* Java \* → \* JAR 文件 \*
3. 单击 \* 下一步 \*。
4. 在以下窗口中，指定目标 JAR 文件路径：tutorial\_plugin.jar 插件的基础类名为 TuttorialPlugin。class，必须将此插件添加到同名文件夹中。

如果插件依赖于其他库，则可以创建以下文件夹：lib/

您可以添加与插件相关的 JAR 文件（例如数据库驱动程序）。当 SnapCenter 加载此插件时，它会自动将此文件夹中的所有 JAR 文件与其关联，并将其添加到类路径中。

## SnapCenter 中的自定义插件

### SnapCenter 中的自定义插件

可以使用 SnapCenter 服务器将使用 Java，Perl 或原生模式创建的自定义插件安装在主机上，以便为应用程序启用数据保护。您必须已导出此插件，才能使用本教程中提供的操作步骤将其安装到 SnapCenter 主机上。

创建插件问题描述文件

对于创建的每个插件，您都必须具有一个问题描述文件。问题描述文件介绍了此插件的详细信息。文件名必须为 Plugin\_Descriptioner.xml。

使用插件描述符文件属性及其重要性

属性	Description
Name	插件的名称。允许使用字母数字字符。例如，DB2，MySQL，MongoDB  对于以原生模式创建的插件，请确保不提供文件扩展名。例如，如果此插件的名称是 MongoDB.sh，请将此名称指定为 MongoDB。

属性	Description
version	插件版本。可以包括主要版本和次要版本。例如， 1.0 ， 1.1 ， 2.0 ， 2.1
displayName	要在 SnapCenter 服务器中显示的插件名称。如果写入了同一插件的多个版本，请确保所有版本的显示名称相同。
插件类型	用于创建插件的语言。支持的值包括 Perl ， Java 和原生。原生插件类型包括 Unix/Linux Shell 脚本， Windows 脚本， Python 或任何其他脚本语言。
OSNAME	安装此插件的主机操作系统名称。有效值为 Windows 和 Linux 。可以在多种操作系统类型上部署一个插件，例如 Perl 类型的插件。
操作系统配置	安装了插件的主机操作系统版本。
资源名称	插件可以支持的资源类型的名称。例如，数据库，实例，集合。
父级	<p>在这种情况下， ResourceName 在层次结构上依赖于另一种资源类型，然后 Parent 确定父 ResourceType 。</p> <p>例如，在 DB2 插件中， ResourceName " 数据库 " 具有父 " 实例 " 。</p>
RequireFileSystemPlugin	是或否确定恢复选项卡是否显示在还原向导中。
ResourceRequireAuthentication	是或否确定自动发现或尚未自动发现的资源在发现存储后是否需要凭据来执行数据保护操作。
RequireFileSystemClone	是或否确定此插件是否需要为克隆 workflow 集成文件系统插件。

以下是自定义插件 DB2 的 Plugin\_descriptor.xml 文件示例：

```

<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>

```

## 创建 ZIP 文件

开发插件并创建描述符文件后，必须将插件文件和 Plugin\_descriptor.xml 文件添加到文件夹中并将其压缩。

在创建 ZIP 文件之前，您必须考虑以下事项：

- 脚本名称必须与插件名称相同。
- 对于 Perl 插件，ZIP 文件夹必须包含一个包含脚本文件的文件夹，并且描述符文件必须位于此文件夹之外。文件夹名称必须与插件名称相同。
- 对于 Perl 插件以外的插件，ZIP 文件夹必须包含描述符和脚本文件。
- 操作系统版本必须为数字。

## 示例

- DB2 插件：将 db2.pm 和 Plugin\_descriptor.xml 文件添加到 "db2.zip" 中。

- 使用 Java 开发的插件：将 JAR 文件，相关的 JAR 文件和 Plugin\_descriptor.xml 文件添加到文件夹中并将其压缩。

上传插件 ZIP 文件

您必须将此插件 ZIP 文件上传到 SnapCenter 服务器，才能在所需主机上部署此插件。

您可以使用 UI 或 cmdlet 上传此插件。

- 用户界面： \*
- 在 \* 添加 \* 或 \* 修改主机 \* 工作流向导中上传插件 ZIP 文件
- 单击 \* " 选择以上传自定义插件 "
- PowerShell： \*
- upload-SmPluginPackage cmdlet

例如， PS> Upload — SmPluginPackage — AbsolutePath c : \DB2\_1.zip

有关 PowerShell cmdlet 的详细信息，请使用 SnapCenter cmdlet 帮助或参阅 cmdlet 参考信息。

" 《 SnapCenter 软件 cmdlet 参考指南》 "。

部署自定义插件

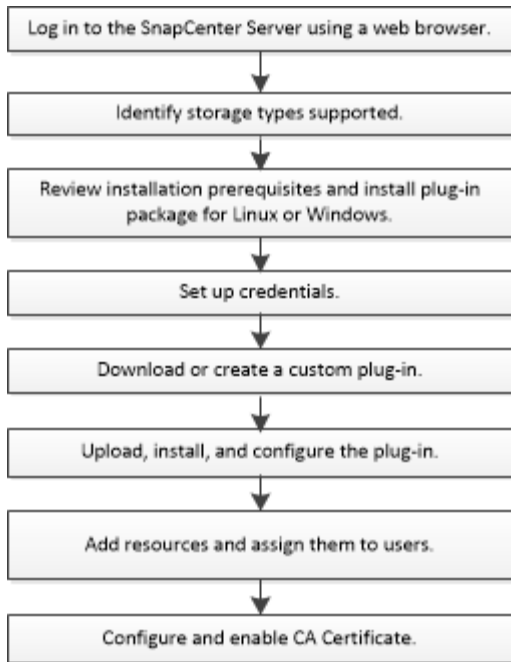
现在，在 \* 添加 \* 和 \* 修改主机 \* workflows 中，可以在所需主机上部署上传的自定义插件。您可以将多个版本的插件上传到 SnapCenter 服务器，并且可以选择要在特定主机上部署的所需版本。

有关如何上传此插件的详细信息，请参见：["添加主机并在远程主机上安装插件软件包"](#)

## 准备安装 SnapCenter 自定义插件

### SnapCenter 自定义插件的安装 workflow

如果要保护自定义插件资源，应安装和设置 SnapCenter 自定义插件。



["为您的应用程序开发一个插件"](#)

## 添加主机和安装 **SnapCenter** 自定义插件的前提条件

在添加主机并安装插件软件包之前，您必须满足所有要求。自定义插件既可以在 Windows 环境中使用，也可以在 Linux 环境中使用。

- 您必须已创建自定义插件。有关详细信息，请参见开发人员信息。

["为您的应用程序开发一个插件"](#)

- 如果要管理 MySQL 或 DB2 应用程序，必须已下载 NetApp 提供的 MySQL 和 DB2 自定义插件。
- 您必须已在 Linux 或 Windows 主机上安装 Java 1.8 或 Java 11 (64 位)。
- 在 Windows 主机上安装插件时，如果您指定的凭据不是内置的，或者用户属于本地工作组用户，则必须在主机上禁用 UAC。
- 自定义插件必须在执行添加主机操作的客户端主机上可用。

### 常规

如果使用的是 iSCSI，则 iSCSI 服务应正在运行。

### SHA512 哈希

- 对于 NetApp 提供的自定义插件，您应确保已将自定义插件文件的 SHA512 哈希添加到 `_custom_plugin_checksum_list` 文件中。
  - 对于 Linux 主机，SHA512 哈希位于：`_var/opt/snapcenter/SCC/custom_plugin_checksum_list.txt`
  - 对于 Windows 主机，SHA512 哈希位于：`_C:\Program Files\NetApp\SnapCenter Plug-in Creator\etc\custom_plugin_checksum_list.txt`



对于自定义安装路径、SHA512哈希位于\_<custom path>\NetApp\SnapCenter\SnapCenter Plug-in Creator\etc\custom\_plugin\_checksum\_list.txt

custom\_plugin\_checksum\_list是SnapCenter 在主机上安装的自定义插件的一部分。

- 对于为应用程序创建的自定义插件、您应已执行以下步骤：

- a. 已生成插件zip文件的SHA512哈希。

您可以使用等联机工具 ["SHA512哈希"](#)。

- b. 已将生成的SHA512哈希添加到custom\_plugin\_checksum\_list文件中的新行。

注释以#符号开头、用于标识哈希所属的插件。

以下是校验和文件中SHA512哈希条目的示例：

```
#ORASCPM
03721f567a1e4a1cb5569066b9a58af619ee12b1f8713108f81b696cfbdb81c25232fa63
d6e6777a2b2a1ec068bb0a93a59a8ade71587182f8bccbe81f7e0ba6
```

## Windows 主机

- 您必须具有具有本地管理员权限的域用户，并在远程主机上具有本地登录权限。
- 如果您在 SnapCenter 中管理集群节点，则必须具有对集群中所有节点具有管理权限的用户。

## Linux 主机

- 您必须已为 root 用户或非 root 用户启用基于密码的 SSH 连接。
- 您必须已在Linux主机上安装Java 1.8或Java 11 (64位)。

如果要对SnapCenter 服务器主机使用Windows Server 2019或Windows Server 2016、则必须安装Java 1.8或Java 11 (64位)。互操作性表工具（IMT）包含有关要求的最新信息。

["适用于所有操作系统的 Java 下载"](#)

["NetApp 互操作性表工具"](#)

- 您必须为非 root 用户配置 sudo 权限，才能提供对多个路径的访问权限。使用 visudo Linux 实用程序将以下行添加到 /etc/sudoers 文件中。



确保使用的是sudo 1.8.7或更高版本。

```

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 SCCMDEXECUTOR =checksum_value==
/opt/NetApp/snapcenter/scc/bin/sccCommandExecutor
LINUX_USER ALL=(ALL) NOPASSWD:SETENV: HPPLCMD, PRECHECKCMD,
CONFIGCHECKCMD, SCCMDEXECUTOR, SCCMD
Defaults: LINUX_USER !visiblepw
Defaults: LINUX_USER !requiretty

```

`_linux_user_` 是您创建的非root用户的名称。

您可以从\* ORACLE\_checksum.txt \*文件中获取\_checksum\_value\_、该文件位于\_C  
: \ProgramData\NetApp\SnapCenter\Package Repository \_。




此示例只能用作创建自己数据的参考。

## 安装适用于 Windows 的 SnapCenter 插件软件包的主机要求

在安装适用于 Windows 的 SnapCenter 插件软件包之前，您应熟悉一些基本的主机系统空间要求和规模估算要求。

项目	要求
操作系统	Microsoft Windows  有关受支持版本的最新信息，请参见 " <a href="#">NetApp 互操作性表工具</a> "。
主机上 SnapCenter 插件的最小 RAM	1 GB

项目	要求
主机上 SnapCenter 插件的最小安装和日志空间	5 GB   您应分配足够的磁盘空间并通过 logs 文件夹监控存储消耗。所需的日志空间因要保护的实体数量和数据保护操作的频率而异。如果没有足够的磁盘空间，则不会为最近运行的操作创建日志。
所需的软件包	<ul style="list-style-type: none"> <li>• Microsoft .NET Framework 4.7.2或更高版本</li> <li>• Windows Management Framework ( WMF ) 4.0 或更高版本</li> <li>• PowerShell 4.0 或更高版本</li> </ul> <p>有关受支持版本的最新信息，请参见 "<a href="#">NetApp 互操作性表工具</a>"。</p> <p>有关.NET专用的故障排除信息、请参见 "<a href="#">对于没有Internet连接的原有系统、SnapCenter 升级或安装失败。</a>"</p>

## 安装适用于 Linux 的 SnapCenter 插件软件包的主机要求

在安装适用于 Linux 的 SnapCenter 插件软件包之前，应确保主机满足要求。

项目	要求
操作系统	<ul style="list-style-type: none"> <li>• Red Hat Enterprise Linux</li> <li>• Oracle Linux</li> <li>• SUSE Linux Enterprise Server ( SLES )</li> </ul>
主机上 SnapCenter 插件的最小 RAM	1 GB
主机上 SnapCenter 插件的最小安装和日志空间	2 GB   您应分配足够的磁盘空间并通过 logs 文件夹监控存储消耗。所需的日志空间因要保护的实体数量和数据保护操作的频率而异。如果没有足够的磁盘空间，则不会为最近运行的操作创建日志。

项目	要求
所需的软件包	<p>Java 1.8 (64位) Oracle Java或OpenJDK</p> <p>如果已将 Java 升级到最新版本，则必须确保 <code>/var/opt/snapcenter/spl/etc/spl.properties</code> 上的 <code>java_home</code> 选项设置为正确的 Java 版本和路径。</p>

有关受支持版本的最新信息，请参见 ["NetApp 互操作性表工具"](#)

## 设置 SnapCenter 自定义插件的凭据

SnapCenter 使用凭据对 SnapCenter 操作的用户进行身份验证。您应创建用于安装 SnapCenter 插件的凭据以及用于对数据库或 Windows 文件系统执行数据保护操作的其他凭据。

### 开始之前

- Linux 主机

您必须设置在 Linux 主机上安装插件的凭据。

您必须为 root 用户或具有 sudo 权限的非 root 用户设置凭据，才能安装和启动插件过程。

\* 最佳实践：\* 虽然在部署主机和安装插件后，您可以为 Linux 创建凭据，但最佳实践是在添加 SVM 之后，在部署主机和安装插件之前创建凭据。

- Windows 主机

在安装插件之前，您必须设置 Windows 凭据。

您必须使用管理员权限设置凭据，包括对远程主机的管理员权限。

- 自定义插件应用程序

此插件使用在添加资源时选择或创建的凭据。如果在数据保护操作期间资源不需要凭据，则可以将凭据设置为 \* 无 \*。

### 关于此任务

如果您为各个资源组设置了凭据，并且用户名不具有完全管理员权限，则必须至少为此用户名分配资源组和备份权限。

### 步骤

1. 在左侧导航窗格中，单击 \* 设置 \*。
2. 在设置页面中，单击 \* 凭据 \*。
3. 单击 \* 新建 \*。

4. 在 \* 凭据 \* 页面中，指定配置凭据所需的信息：

对于此字段 ...	执行此操作 ...
凭据名称	输入凭据的名称。
用户名	<p>输入要用于身份验证的用户名和密码。</p> <ul style="list-style-type: none"> <li>域管理员或管理员组的任何成员</li> </ul> <p>指定要安装 SnapCenter 插件的系统上的域管理员或管理员组的任何成员。用户名字段的有效格式为：</p> <ul style="list-style-type: none"> <li><code>netbios\username</code></li> <li>域 FQDN\username_</li> </ul> <ul style="list-style-type: none"> <li>本地管理员（仅适用于工作组）</li> </ul> <p>对于属于工作组的系统，请指定要安装 SnapCenter 插件的系统上的内置本地管理员。如果用户帐户具有提升的权限或在主机系统上禁用了用户访问控制功能，则可以指定属于本地管理员组的本地用户帐户。用户名字段的有效格式为：<code>username</code></p>

对于此字段 ...	执行此操作 ...
Password	输入用于身份验证的密码。
身份验证模式	选择要使用的身份验证模式。
使用 sudo 权限	<p>如果要为非 root 用户创建凭据，请选中 * 使用 sudo 权限 * 复选框。</p> <p> 仅适用于 Linux 用户。</p>

5. 单击 \* 确定 \*。

完成凭据设置后，您可能需要在 " 用户和访问 " 页面上为用户或用户组分配凭据维护。

## 在 Windows Server 2012 或更高版本上配置 GMSA

通过 Windows Server 2012 或更高版本，您可以创建组托管服务帐户（GMSA），以便从受管域帐户自动管理服务帐户密码。

开始之前

- 您应具有 Windows Server 2012 或更高版本的域控制器。
- 您应该拥有一个 Windows Server 2012 或更高版本的主机，该主机是域的成员。

步骤

1. 创建一个 KDS 根密钥，以便为 GMSA 中的每个对象生成唯一的密码。
2. 对于每个域，从 Windows 域控制器运行以下命令：Add-KDSRootKey -EffectiveImmediately
3. 创建和配置 GMSA：
  - a. 按以下格式创建用户组帐户：

```
domainName\accountName$
.. 向组中添加计算机对象。
.. 使用刚刚创建的用户组创建 GMSA 。
```

例如：

```
New-ADServiceAccount -name <ServiceAccountName> -DNSHostName <fqdn>
-PrincipalsAllowedToRetrieveManagedPassword <group>
-ServicePrincipalNames <SPN1,SPN2,...>
.. 运行 `Get-ADServiceAccount` 命令验证服务帐户。
```

4. 在主机上配置 GMSA：

- a. 在要使用 GMSA 帐户的主机上为 Windows PowerShell 启用 Active Directory 模块。

为此，请从 PowerShell 运行以下命令：

```
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.
```

- a. 重新启动主机。
  - b. 在 PowerShell 命令提示符处运行以下命令，在主机上安装 GMSA：`Install-AdServiceAccount <GMSA>`
  - c. 运行以下命令验证您的 GMSA 帐户：`Test-AdServiceAccount <GMSA>`
5. 为主机上配置的 GMSA 分配管理权限。
  6. 通过在 SnapCenter 服务器中指定已配置的 GMSA 帐户来添加 Windows 主机。

SnapCenter 服务器将在主机上安装选定插件，并且在安装此插件期间，指定的 GMSA 将用作服务登录帐户。

## 安装 SnapCenter 自定义插件

添加主机并在远程主机上安装插件软件包

您必须使用 SnapCenterAdd Host 页面添加主机，然后安装插件软件包。这些插件会自动安装在远程主机上。您可以为单个主机或集群添加主机并安装插件软件包。

开始之前

- 您应该是分配给具有插件安装和卸载权限的角色的用户、例如 SnapCenter 管理员角色。
- 您应确保消息队列服务正在运行。
- 如果您使用的是组托管服务帐户（GMSA），则应使用管理权限配置 GMSA。

["在 Windows Server 2012 或更高版本上为自定义应用程序配置组托管服务帐户"](#)

## 关于此任务

您不能将 SnapCenter 服务器作为插件主机添加到另一个 SnapCenter 服务器。

如果在集群（WSFC）上安装插件，则这些插件将安装在集群的所有节点上。

## 步骤

1. 在左侧导航窗格中，选择 \* 主机 \*。
2. 验证顶部是否已选中 \* 受管主机 \* 选项卡。
3. 选择 \* 添加 \*。
4. 在主机页面中，执行以下操作：

对于此字段 ...	执行此操作 ...
主机类型	<p>选择主机类型：</p> <ul style="list-style-type: none"><li>• Windows</li><li>• Linux</li></ul> <p> 自定义插件可在 Windows 和 Linux 环境中使用。</p>
主机名	<p>输入主机的完全限定域名（FQDN）或 IP 地址。</p> <p>SnapCenter 取决于 DNS 的正确配置。因此，最佳做法是输入 FQDN。</p> <p>对于 Windows 环境，只有在将 IP 地址解析为 FQDN 时，不可信域主机才支持此 IP 地址。</p> <p>您可以输入独立主机的 IP 地址或 FQDN。</p> <p>如果要使用 SnapCenter 添加主机，并且此主机属于子域，则必须提供 FQDN。</p>
凭据	<p>选择您创建的凭据名称或创建新凭据。</p> <p>这些凭据必须对远程主机具有管理权限。有关详细信息，请参见有关创建凭据的信息。</p> <p>您可以通过将光标置于指定的凭据名称上方来查看有关凭据的详细信息。</p> <p> 凭据身份验证模式由您在添加主机向导中指定的主机类型决定。</p>

5. 在 \* 选择要安装的插件 \* 部分中，选择要安装的插件。
6. (可选)选择\*更多选项\*。



对于此字段 ...	执行此操作 ...
Port	<p>保留默认端口号或指定端口号。</p> <p>默认端口号为 8145。如果 SnapCenter 服务器安装在自定义端口上，则该端口号将显示为默认端口。</p> <div style="border-left: 1px solid #ccc; padding-left: 10px; margin-top: 10px;">  如果您手动安装了插件并指定了自定义端口，则必须指定相同的端口。否则，操作将失败。 </div>
安装路径	<p>自定义插件可以安装在 Windows 系统或 Linux 系统上。</p> <ul style="list-style-type: none"> <li>• 对于适用于 Windows 的 SnapCenter 插件软件包，默认路径为 C:\Program Files\NetApp\SnapCenter。</li> </ul> <p>您也可以自定义路径。</p> <ul style="list-style-type: none"> <li>• 对于适用于 Linux 的 SnapCenter 插件软件包，默认路径为 /opt/NetApp/snapcenter。</li> </ul> <p>您也可以自定义路径。</p> <ul style="list-style-type: none"> <li>• 对于 SnapCenter 自定义插件： <ul style="list-style-type: none"> <li>i. 在“Custom Plug-ins”(自定义插件)部分，选择*Browse*(浏览*)，然后选择压缩的自定义插件文件夹。</li> </ul> <p>压缩文件夹包含自定义插件代码和描述符 .xml 文件。</p> <p>对于存储插件、导航到  <code>C:\ProgramData\NetApp\SnapCenter\Package Repository</code> 并选择  <code>Storage.zip</code> 文件夹。</p> <ul style="list-style-type: none"> <li>ii. 选择*上传*。</li> </ul> <p>上传软件包之前，会验证压缩后的自定义插件文件夹中的描述符 .xml 文件。</p> <p>此时将列出上传到 SnapCenter 服务器的自定义插件。</p> <p>如果要管理 MySQL 或 DB2 应用程序，可以使用 NetApp 提供的 MySQL 和 DB2 自定义插件。MySQL 和 DB2 自定义插件可从获取 <a href="#">"NetApp 自动化商店"</a></p> </li> </ul>

对于此字段 ...	执行此操作 ...
跳过安装前检查	如果您已手动安装插件，并且不想验证主机是否满足安装插件的要求，请选中此复选框。
使用组托管服务帐户（GMSA）运行插件服务	<p>对于 Windows 主机，如果要使用组托管服务帐户（GMSA）运行插件服务，请选中此复选框。</p> <p> 按以下格式提供 GMSA 名称： domainname\accountName\$。</p> <p> GMSA 仅用作适用于 Windows 的 SnapCenter 插件服务的登录服务帐户。</p>

## 7. 选择 \* 提交 \*。

如果未选中 \* 跳过预检查 \* 复选框，则主机将通过验证以验证主机是否满足安装插件的要求。磁盘空间，RAM，PowerShell 版本，.NET 版本，位置（对于 Windows 插件）和 Java 版本（对于 Linux 插件）均已根据最低要求进行验证。如果不满足最低要求，则会显示相应的错误或警告消息。

如果此错误与磁盘空间或 RAM 相关，您可以更新位于 C : \Program Files\NetApp\SnapCenter WebApp 的 web.config 文件以修改默认值。如果此错误与其他参数相关，则必须修复问题描述。



在 HA 设置中，如果要更新 web.config 文件，则必须同时更新两个节点上的文件。

## 8. 如果主机类型为 Linux，请验证指纹，然后选择 \* 确认并提交 \*。



即使先前已将同一主机添加到 SnapCenter 并确认了指纹，也必须进行指纹验证。

## 9. 监控安装进度。

安装专用的日志文件位于 /custom\_location/snapcenter/ 日志。

使用 **cmdlet** 在多个远程主机上安装适用于 **Linux** 或 **Windows** 的 **SnapCenter** 插件软件包

您可以使用 Install-SmHostPackage PowerShell cmdlet 在多个主机上同时安装适用于 Linux 或 Windows 的 SnapCenter 插件软件包。

开始之前

添加主机的用户应具有该主机的管理权限。

步骤

1. 启动 PowerShell 。
2. 在 SnapCenter 服务器主机上，使用 Open-SmConnection cmdlet 建立会话，然后输入凭据。
3. 使用 Install-SmHostPackage cmdlet 和所需参数在多个主机上安装此插件。

有关可与 cmdlet 结合使用的参数及其说明的信息，可通过运行 `get-help command_name` 来获取。或者，您也可以参考 "《[SnapCenter 软件 cmdlet 参考指南](#)》"。

如果您已手动安装插件，并且不想验证主机是否满足安装插件的要求，则可以使用 `-skipprecheck` 选项。

#### 4. 输入远程安装的凭据。

使用命令行界面在 **Linux** 主机上安装 **SnapCenter** 自定义插件

您应使用 SnapCenter 用户界面（UI）安装 SnapCenter 自定义插件。如果您的环境不允许从 SnapCenter UI 远程安装此插件，则可以使用命令行界面（CLI）在控制台模式或静默模式下安装此自定义插件。

#### 步骤

1. 将适用于 Linux 的 SnapCenter 插件软件包安装文件（`snapcenter_linux_host_plugin.bin`）从 `C:\ProgramData\NetApp\SnapCenter\Package Repository` 复制到要安装自定义插件的主机。

您可以从安装了 SnapCenter 服务器的主机访问此路径。

2. 在命令提示符处，导航到复制安装文件的目录。

3. 安装插件：`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` 用于指定 SMCORE HTTPS 通信端口。
- `-dserver_ip` 指定 SnapCenter 服务器 IP 地址。
- `-dserver_HTTPS_port` 指定 SnapCenter 服务器 HTTPS 端口。
- `-duser_install_dir` 指定要安装适用于 Linux 的 SnapCenter 插件软件包的目录。
- `DINSTALL_LOG_name` 指定日志文件的名称。

```
/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. 使用 `Add-Smhost` cmdlet 和所需参数将主机添加到 SnapCenter 服务器。

有关可与命令结合使用的参数及其说明的信息，可通过运行 `get-help command_name` 来获取。或者，您也可以参考 "《[SnapCenter 软件 cmdlet 参考指南](#)》"。

5. 登录到 SnapCenter，然后从 UI 或使用 PowerShell cmdlet 上传自定义插件。

您可以参考从 UI 上传自定义插件 "[添加主机并在远程主机上安装插件软件包](#)" 部分。

SnapCenter cmdlet 帮助和 cmdlet 参考信息包含有关 PowerShell cmdlet 的详细信息。






" 《 SnapCenter 软件 cmdlet 参考指南》 "。

## 监控安装自定义插件的状态

您可以使用作业页面监控 SnapCenter 插件软件包的安装进度。您可能需要检查安装进度以确定安装完成的时间或是否存在问题描述。

### 关于此任务

以下图标将显示在作业页面上，并指示操作的状态：

-  正在进行中
-  已成功完成
-  失败
-  已完成，但出现警告或由于出现警告而无法启动
-  已排队

### 步骤

1. 在左侧导航窗格中，单击 \* 监控 \*。
2. 在 \* 监控 \* 页面中，单击 \* 作业 \*。
3. 在 \*Jobs\* 页中，要过滤列表以便仅列出插件安装操作，请执行以下操作：
  - a. 单击 \* 筛选器 \*。
  - b. 可选：指定开始和结束日期。
  - c. 从类型下拉菜单中，选择 \* 插件安装 \*。
  - d. 从状态下拉菜单中，选择安装状态。
  - e. 单击 \* 应用 \*。
4. 选择安装作业并单击 \* 详细信息 \* 以查看作业详细信息。
5. 在 \* 作业详细信息 \* 页面中，单击 \* 查看日志 \*。

## 配置 CA 证书

### 生成 CA 证书 CSR 文件

您可以生成证书签名请求（CSR），并导入可使用生成的 CSR 从证书颁发机构（CA）获取的证书。此证书将具有一个关联的专用密钥。

CSR 是一个编码文本块，提供给授权证书供应商以采购签名的 CA 证书。



CA证书RSA密钥长度应至少为3072位。

有关生成 CSR 的信息，请参见 ["如何生成 CA 证书 CSR 文件"](#)。



如果您拥有域（\*.domain.company.com）或系统（machine1.domain.company.com）的 CA 证书，则可以跳过生成 CA 证书 CSR 文件。您可以使用 SnapCenter 部署现有 CA 证书。

对于集群配置，CA 证书中应提及集群名称（虚拟集群 FQDN）以及相应的主机名。在获取证书之前，可以通过填写使用者替代名称（SAN）字段来更新此证书。对于通配符证书（\*.domain.company.com），此证书将隐式包含域的所有主机名。

## 导入 CA 证书

您必须使用 Microsoft 管理控制台（MMC）将 CA 证书导入到 SnapCenter 服务器和 Windows 主机插件中。

### 步骤

1. 转到 Microsoft 管理控制台（MMC），然后单击 \* 文件 \* > \* 添加 / 删除 Snapin \*。
2. 在添加或删除管理单元窗口中，选择 \* 证书 \*，然后单击 \* 添加 \*。
3. 在证书管理单元窗口中，选择 \* 计算机帐户 \* 选项，然后单击 \* 完成 \*。
4. 单击 \* 控制台根 \* > \* 证书-本地计算机 \* > \* 可信根证书颁发机构 \* > \* 证书 \*。
5. 右键单击文件夹 "可信根证书颁发机构"，然后选择 \* 所有任务 \* > \* 导入 \* 以启动导入向导。
6. 完成向导，如下所示：

在此向导窗口中 ...	执行以下操作 ...
导入私钥	选择 * 是 * 选项，导入私钥，然后单击 * 下一步 *。
导入文件格式	不进行任何更改；单击 * 下一步 *。
安全性	指定要用于导出的证书的新密码，然后单击 * 下一步 *。
正在完成证书导入向导	查看摘要，然后单击 * 完成 * 开始导入。



导入证书应与私钥捆绑在一起(支持的格式为：.pfx、.p12和\*.p7b)。

7. 对 "Personal" 文件夹重复步骤 5。

## 获取 CA 证书指纹

证书指纹是用于标识证书的十六进制字符串。指纹是使用指纹算法根据证书内容计算得出的。

### 步骤

1. 在 GUI 上执行以下操作：
  - a. 双击证书。
  - b. 在证书对话框中，单击 \* 详细信息 \* 选项卡。

- c. 滚动字段列表，然后单击 \* 缩略图 \*。
- d. 从框中复制十六进制字符。
- e. 删除十六进制数之间的空格。

例如，如果指纹为 "A9 09 50 2D D8 2a E4 14 33 E6 F8 38 86 b0 0d 42 77 A3 2a 7b"，则在删除空格后，指纹将为 "a909502dd82ae41433e6f83886b00d4277a32a7b"。

## 2. 从 PowerShell 执行以下操作：

- a. 运行以下命令以列出已安装证书的指纹，并按主题名称标识最近安装的证书。

```
Get-子项 -Path Cert : \LOCALMACHINE\My
```

- b. 复制指纹。

## 使用 Windows 主机插件服务配置 CA 证书

您应使用 Windows 主机插件服务配置 CA 证书，以激活已安装的数字证书。

在 SnapCenter 服务器以及已部署 CA 证书的所有插件主机上执行以下步骤。

### 步骤

1. 运行以下命令，删除与 SMCore 默认端口 8145 的现有证书绑定：

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

例如：

```
> netsh http delete sslcert ipport=0.0.0.0:8145  
. 运行以下命令，将新安装的证书与 Windows 主机插件服务绑定：
```

```
> $cert = "_<certificate thumbprint>_"  
> $guid = [guid]::NewGuid().ToString("B")  
> netsh http add sslcert ipport=0.0.0.0: _<SMCore Port>_ certhash=$cert  
appid="$guid"
```

例如：

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

在 Linux 主机上为 SnapCenter 自定义插件服务配置 CA 证书

您应管理自定义插件密钥库及其证书的密码，配置 CA 证书，为自定义插件信任存储配置根或中间证书，并将 CA 签名密钥对配置为使用 SnapCenter 自定义插件服务的自定义插件信任存储，以激活已安装的数字证书。

自定义插件使用文件 "keystore.jks"，该文件位于 `/opt/netapp/snapcenter/scc/etc`，同时用作其信任存储和密钥存储。

管理自定义插件密钥库的密码以及正在使用的 CA 签名密钥对的别名

步骤

1. 您可以从自定义插件代理属性文件中检索自定义插件密钥库默认密码。

它是与密钥 "keystore\_pass" 对应的值。

2. 更改密钥库密码：

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

• 将密钥库中私钥条目的所有别名的密码更改为密钥库使用的相同密码：

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

为 `agent.properties` 文件中的 `keystore_pass` 密钥更新相同的。

3. 更改密码后重新启动服务。



自定义插件密钥库的密码和专用密钥的所有关联别名密码应相同。

将根证书或中间证书配置为自定义插件信任存储

您应将不带私钥的根证书或中间证书配置为自定义插件信任存储。

步骤

1. 导航到包含自定义插件密钥库的文件夹：`/opt/netapp/snapcenter/SCC/etc`.
2. 找到文件 "keystore.jks"。
3. 列出密钥库中添加的证书：

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

4. 添加根证书或中间证书：

```
keytool -import -trustcacerts -alias myRootCA -file
/root/USERTrustRSA_Root.cer -keystore keystore.jks
. 将根证书或中间证书配置为自定义插件信任存储后重新启动服务。
```



您应先添加根 CA 证书，然后再添加中间 CA 证书。

将 CA 签名密钥对配置为自定义插件信任存储

您应将 CA 签名密钥对配置为自定义插件信任存储。

#### 步骤

1. 导航到包含自定义插件密钥库 `/opt/netapp/snapcenter/scc/` 等的文件夹
2. 找到文件 "keystore.jks" 。
3. 列出密钥库中添加的证书：

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

4. 添加同时具有私钥和公有密钥的 CA 证书。

```
keytool -importkeystore -srckeystore /root/snapcenter.ssl.test.netapp.com.pfx
-srcstoretype PKCS12 -destkeystore keystore.jks -deststoretype JKS
```

5. 列出密钥库中添加的证书。

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

6. 验证密钥库是否包含与已添加到密钥库中的新 CA 证书对应的别名。
7. 将为 CA 证书添加的私钥密码更改为密钥库密码。

默认自定义插件密钥库密码是 `agent.properties` 文件中密钥 `keystore_pass` 的值。

```
keytool -keypasswd -alias "alias_name_in_CA_cert" -keystore
keystore.jks
. 如果 CA 证书中的别名较长，并且包含空格或特殊字符（ "*" ， " ， "
） ， 请将别名更改为简单名称：
```

```
keytool -changealias -alias "long_alias_name" -destalias "simple_alias"
-keystore keystore.jks
. 从 agent.properties 文件中的 CA 证书配置别名。
```

根据密钥 `SCC_certificate_alias` 更新此值。

8. 将 CA 签名密钥对配置为自定义插件信任存储后重新启动服务。



为 **SnapCenter** 自定义插件配置证书撤销列表（**Certificate Revocation List**，**CRL**）

关于此任务

- SnapCenter 自定义插件将在预配置的目录中搜索 CRL 文件。
- SnapCenter 自定义插件的 CRL 文件的默认目录为 "opt/netapp/snapcenter/SCC/etc/CRL"。

步骤

1. 您可以使用密钥 `CRL_PATH` 修改和更新 `agent.properties` 文件中的默认目录。

您可以在此目录中放置多个 CRL 文件。系统将根据每个 CRL 验证传入的证书。

在 **Windows** 主机上为 **SnapCenter** 自定义插件服务配置 **CA** 证书

您应管理自定义插件密钥库及其证书和密码，配置 CA 证书，为自定义插件信任存储配置根或中间证书，并将 CA 签名密钥对配置为使用 SnapCenter 自定义插件服务的自定义插件信任存储，以激活已安装的数字证书。

自定义插件使用文件 `keystore.jks`，该文件位于 `C:\Program Files\NetApp\SnapCenter\SnapCenter Plug-in Creator\etc`，同时用作其信任存储和密钥存储。

管理自定义插件密钥库的密码以及正在使用的 **CA** 签名密钥对的别名

步骤

1. 您可以从自定义插件代理属性文件中检索自定义插件密钥库默认密码。

该值与 `key_keystore_pass` 键对应。

2. 更改密钥库密码：

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



如果在 Windows 命令提示符处无法识别 "keytool" 命令，请将 keytool 命令替换为其完整路径。

```
C:\Program Files\java\<JDK_version>\bin\keytool.exe -storepasswd -keystore keystore.jks
```

3. 将密钥库中私钥条目的所有别名的密码更改为密钥库使用的相同密码：

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

为 `agent.properties` 文件中的 `keystore_pass` 密钥更新相同的。

4. 更改密码后重新启动服务。



自定义插件密钥库的密码和专用密钥的所有关联别名密码应相同。

将根证书或中间证书配置为自定义插件信任存储

您应将不带私钥的根证书或中间证书配置为自定义插件信任存储。

## 步骤

1. 导航到包含自定义插件密钥库 C : \Program Files\NetApp\SnapCenter\SnapCenter Plug-in Creator\etc 的文件夹
2. 找到文件 "keystore.jks" 。
3. 列出密钥库中添加的证书:

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

4. 添加根证书或中间证书:

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

5. 将根证书或中间证书配置为自定义插件信任存储后重新启动服务。



您应先添加根 CA 证书，然后再添加中间 CA 证书。

将 CA 签名密钥对配置为自定义插件信任存储

您应将 CA 签名密钥对配置为自定义插件信任存储。

## 步骤

1. 导航到包含自定义插件密钥库 C : \Program Files\NetApp\SnapCenter\SnapCenter Plug-in Creator\etc 的文件夹
2. 找到文件 *keystore.jks* 。
3. 列出密钥库中添加的证书:

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

4. 添加同时具有私钥和公有密钥的 CA 证书。

```
keytool -importkeystore -srckeystore /root/snapcenter.ssl.test.netapp.com.pfx -srcstoretype PKCS12 -destkeystore keystore.jks -deststoretype JKS
```

5. 列出密钥库中添加的证书。

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

6. 验证密钥库是否包含与已添加到密钥库中的新 CA 证书对应的别名。
7. 将为 CA 证书添加的私钥密码更改为密钥库密码。

默认自定义插件密钥库密码是 agent.properties 文件中密钥 keystore\_pass 的值。

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

8. 从 agent.properties 文件中的 CA 证书配置别名。

根据密钥 SCC\_certificate\_alias 更新此值。

9. 将 CA 签名密钥对配置为自定义插件信任存储后重新启动服务。

为 **SnapCenter** 自定义插件配置证书撤销列表（**Certificate Revocation List**，**CRL**）

关于此任务

- 要下载相关 CA 证书的最新 CRL 文件，请参见 ["如何更新 SnapCenter CA 证书中的证书撤销列表文件"](#)。
- SnapCenter 自定义插件将在预配置的目录中搜索 CRL 文件。
- SnapCenter 自定义插件的 CRL 文件的默认目录为：`C:\Program Files\NetApp\SnapCenter\SnapCenter Plug-in Creator\etc\CRL'`。

步骤

1. 您可以使用密钥 `CRL_PATH` 修改和更新 `agent.properties` 文件中的默认目录。
2. 您可以在此目录中放置多个 CRL 文件。

系统将根据每个 CRL 验证传入的证书。

为插件启用 **CA** 证书

您应在 SnapCenter 服务器和相应插件主机中配置 CA 证书并部署 CA 证书。您应为插件启用 CA 证书验证。

开始之前

- 您可以使用 `run set-SmCertificateSettings` cmdlet 启用或禁用 CA 证书。
- 您可以使用 `Get-SmCertificateSettings` 显示插件的证书状态。





有关可与 cmdlet 结合使用的参数及其说明的信息，可通过运行 `get-help command_name` 来获取。或者，您也可以参考 ["《SnapCenter 软件 cmdlet 参考指南》"](#)。

步骤

1. 在左侧导航窗格中，单击 \* 主机 \*。
2. 在主机页面中，单击 \* 受管主机 \*。
3. 选择一个或多个插件主机。
4. 单击 \* 更多选项 \*。
5. 选择 \* 启用证书验证 \*。

完成后

受管主机选项卡主机会显示一个挂锁，挂锁的颜色表示 SnapCenter 服务器与插件主机之间的连接状态。

-  表示 CA 证书既未启用，也未分配给插件主机。
-  表示 CA 证书已成功验证。
-  表示无法验证 CA 证书。
-  表示无法检索连接信息。



如果状态为黄色或绿色，则表示数据保护操作已成功完成。

# 准备数据保护

## 使用 SnapCenter 自定义插件的前提条件

在使用 SnapCenter 自定义插件之前，SnapCenter 管理员必须安装和配置 SnapCenter 服务器并执行前提条件任务。

- 安装和配置 SnapCenter 服务器。
- 登录到 SnapCenter 服务器。
- 如果适用，可通过添加存储系统连接和创建凭据来配置 SnapCenter 环境。
- 添加主机，然后安装和上传插件。
- 如果适用，请在插件主机上安装 Java 1.7 或 Java 1.8 。
- 如果您有多个数据路径（LIF）或 DNFS 配置，则可以在数据库主机上使用 SnapCenter 命令行界面执行以下操作：
  - 默认情况下，数据库主机的所有 IP 地址都会添加到 Storage Virtual Machine（SVM）中克隆卷的 NFS 存储导出策略中。如果要使用特定的 IP 地址或限制为部分 IP 地址，请运行 `Set-PreferredHostIPsInStorageExportPolicy CLI` 。
  - 如果 SVM 中有多个数据路径（LIF），则 SnapCenter 会选择适当的数据路径（LIF）来挂载 NFS 克隆卷。但是，如果要指定特定数据路径（LIF），则必须运行 `Set-SvmPreferredDataPath CLI`。有关可与命令结合使用的参数及其说明的信息，可通过运行 `get-help command_name` 来获取。或者，您也可以参考 "《[SnapCenter 软件命令参考指南](#)》"。
- 如果要进行备份复制，请设置 SnapMirror 和 SnapVault 。
- 确保端口 9090 未被主机上的任何其他应用程序使用。

除了 SnapCenter 所需的其他端口之外，还必须保留端口 9090 以供 SnapCenter 自定义插件使用。

## 如何使用资源，资源组和策略保护自定义插件资源

在使用 SnapCenter 之前，了解与要执行的备份，克隆和还原操作相关的基本概念非常有用。您可以与不同操作的资源，资源组和策略进行交互。

- 资源通常是指使用 SnapCenter 备份或克隆的数据库，Windows 文件系统或 VM 。
- SnapCenter 资源组是主机或集群上的一组资源。

在对资源组执行操作时，您可以根据为资源组指定的计划对资源组中定义的资源执行该操作。

您可以按需备份单个资源或资源组。您还可以对单个资源和资源组执行计划的备份。

- 这些策略可指定数据保护操作的备份频率，副本保留，复制，脚本和其他特征。

创建资源组时，您可以为该组选择一个或多个策略。您也可以在为单个资源按需执行备份时选择策略。

将某个资源组视为定义了您要保护的内容，以及您希望在日期和时间保护的时间。将某个策略视为定义了要保护的 `_how_`。例如，如果要备份主机的所有数据库或所有文件系统，则可以创建一个资源组，其中包含主机中的

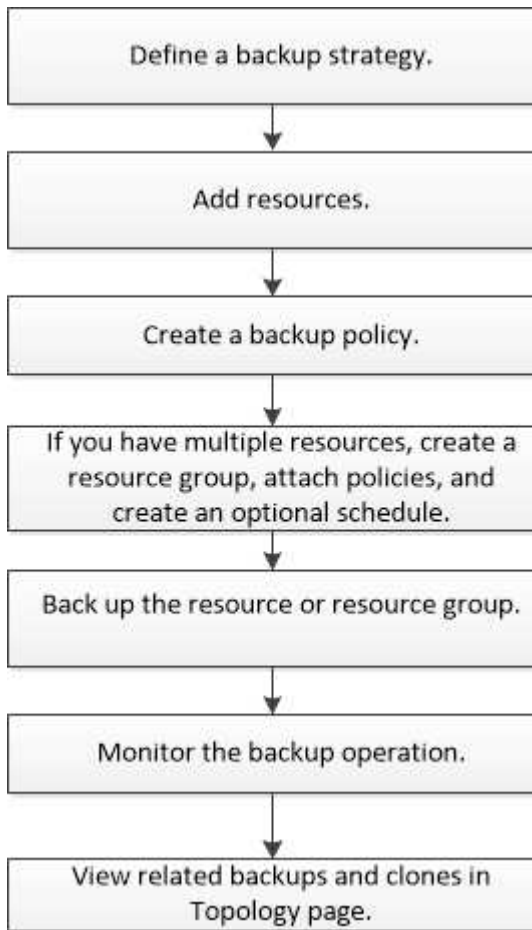
所有数据库或所有文件系统。然后，您可以将两个策略附加到资源组：每日策略和每小时策略。在创建资源组并附加策略时，您可以将资源组配置为每天执行基于文件的备份，并执行另一个计划，以便每小时执行一次基于 Snapshot 的备份。

## 备份自定义插件资源

### 备份自定义插件资源

备份工作流包括规划，确定要备份的资源，管理备份策略，创建资源组和附加策略，创建备份以及监控操作。

以下工作流显示了必须执行备份操作的顺序：



您也可以手动或在脚本中使用 PowerShell cmdlet 执行备份，还原和克隆操作。有关 PowerShell cmdlet 的详细信息，请使用 SnapCenter cmdlet 帮助或参见 "[《SnapCenter 软件 cmdlet 参考指南》](#)"

### 向 SnapCenter 自定义插件添加资源

您必须添加要备份或克隆的资源。根据您的环境，资源可能是要备份或克隆的数据库实例或集合。

开始之前

- 您必须已完成安装 SnapCenter 服务器，添加主机，创建存储系统连接和添加凭据等任务。

- 您必须拥有 "已为您的应用程序创建自定义插件"。
- 您必须已将插件上传到 SnapCenter 服务器。

#### 关于此任务

您还可以为 MySQL 和 DB2 应用程序添加资源。这些插件可从下载 "[NetApp 存储自动化商店](#)"。

#### 步骤

1. 在左侧导航窗格中，选择\*Resoles\*，然后从列表中选择相应的插件。
2. 在“资源”页面中，选择\*Add ResResource \*。
3. 在提供资源详细信息页面中，执行以下操作：

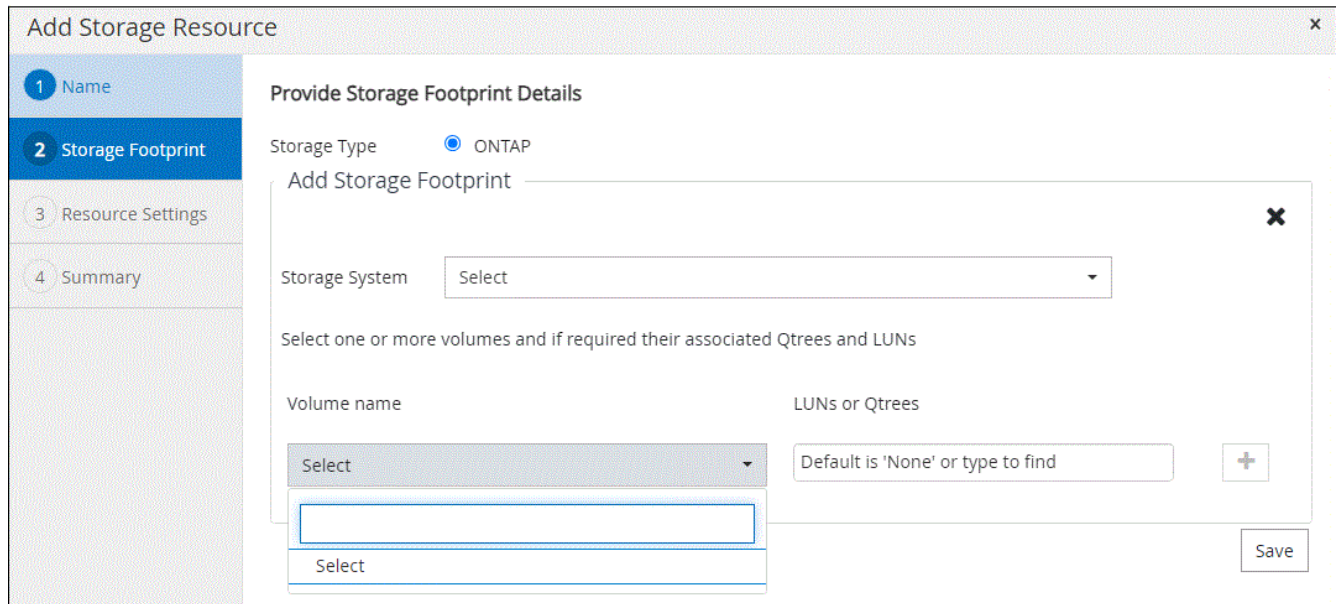
对于此字段 ...	执行此操作 ...
Name	输入资源的名称。
主机名	选择主机。
Type	选择类型。type 是用户根据插件问题描述文件定义的。例如，数据库和实例。  如果选定类型为父类型，请输入父类型的详细信息。例如，如果类型为 "数据库"，而父级为 "实例"，请输入此实例的详细信息。
凭据名称	选择 Credential 或创建新凭据。
挂载路径	输入挂载资源的挂载路径。这仅适用于 Windows 主机。

4. 在提供存储占用空间页面中，选择一个存储系统并选择一个或多个卷、LUN和qtrees，然后选择\*Save\*。

可选：选择  图标，用于从其他存储系统添加更多卷，LUN 和 qtree 。



SnapCenter自定义插件不支持自动发现资源。物理和虚拟环境的存储详细信息也不会自动发现。您必须在创建资源时提供物理和虚拟环境的存储信息。



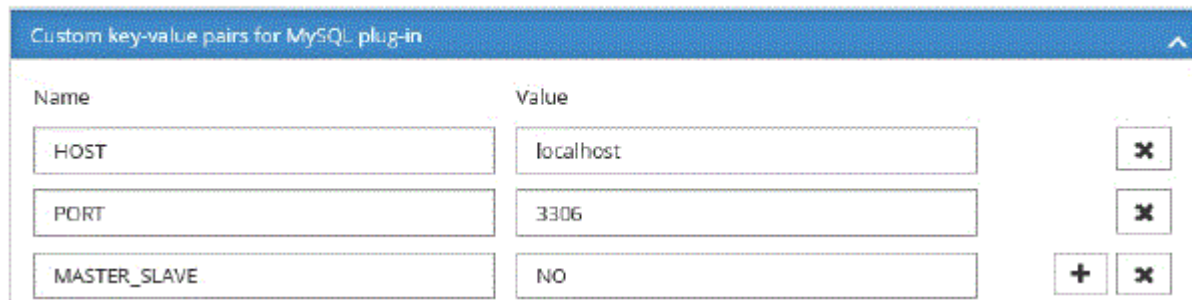
5. 在资源设置页面中，为资源提供自定义密钥值对。

如果要传递特定于资源的信息，请使用自定义密钥值对。例如，在使用 MySQL 插件时，您必须将主机指定为 `host=hostname`，`port =port-no` 用于 MySQL，而主从配置指定为 `master_slave = "Yes"` 或 `"no"`（名称是 `master_slave`，值为 `"Yes"` 或 `"no"`）。



确保单词 `host` 和 `port` 为大写。

#### Resource settings !



6. 查看摘要，然后选择\*完成\*。

#### 结果

此时将显示资源以及类型，主机或集群名称，关联的资源组和策略以及整体状态等信息。



如果在 SnapCenter 外部重命名了数据库，则必须刷新资源。

#### 完成后

如果要为其他用户提供对资产的访问权限，SnapCenter 管理员必须为这些用户分配资产。这样，用户就可以执行对分配给他们的资产具有权限的操作。

添加资源后，您可以修改资源详细信息。如果自定义插件资源具有关联的备份，则无法修改以下字段：资源名称，资源类型和主机名。

## 为自定义插件资源创建策略

在使用 SnapCenter 备份自定义插件特定资源之前，您必须为要备份的资源或资源组创建备份策略。

### 开始之前

- 您应已定义备份策略。

有关详细信息，请参见有关为自定义插件定义数据保护策略的信息。

- 您应已做好数据保护准备。

准备数据保护包括安装 SnapCenter，添加主机，创建存储系统连接和添加资源等任务。

- 应将 Storage Virtual Machine（SVM）分配给您以执行镜像或存储操作。

如果要将快照复制到镜像或存储，SnapCenter管理员必须已将源卷和目标卷的SVM分配给您。

- 您应已手动添加要保护的资源。

### 关于此任务

- 备份策略是一组规则，用于控制如何管理，计划和保留备份。此外，您还可以指定复制，脚本和应用程序设置。
- 在策略中指定选项可节省将策略重复用于其他资源组的时间。
- SnapLock
  - 如果选择了“将备份副本保留特定天数”选项，则SnapLock保留期限必须小于或等于上述保留天数。
  - 指定Snapshot锁定期限可防止删除Snapshot、直到保留期限到期为止。这样可能会导致保留的Snapshot数量超过策略中指定的数量。
  - 对于ONTAP 9.12.1及更低版本、在还原过程中从SnapLock存储快照创建的克隆将继承SnapLock存储到期时间。存储管理员应在SnapLock到期时间后手动清理克隆。



主SnapLock设置在SnapCenter备份策略中进行管理、辅SnapLock设置由ONTAP进行管理。

### 步骤

1. 在左侧导航窗格中，单击 \* 设置 \*。
2. 在设置页面中，单击 \* 策略 \*。
3. 单击 \* 新建 \*。
4. 在名称页面中，输入策略名称和问题描述。
5. 在设置页面中，执行以下步骤：
  - 通过选择 \* 按需 \*，\* 每小时 \*，\* 每日 \*，\* 每周 \* 或 \* 每月 \* 来指定计划类型。



您可以在创建资源组时为备份操作指定计划（开始日期，结束日期和频率）。这样，您可以创建共享相同策略和备份频率的资源组，但可以为每个策略分配不同的备份计划。



**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



如果您已计划凌晨 2 : 00 ，则在夏令时（ DST ）期间不会触发此计划。

- 在自定义备份设置部分中，以密钥值格式提供必须传递到插件的任何特定备份设置。您可以提供多个要传递到插件的密钥值。

6. 在\*保留\*页面中，指定备份类型和在\*备份类型\*页面中选择的计划类型的保留设置：

如果您要 ...	那么 ...
保留一定数量的快照	<p>选择*要保留的Snapshot副本总数*、然后指定要保留的Snapshot数量。</p> <p>如果快照数量超过指定数量、则会删除快照、并首先删除最早的副本。</p> <div style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;"> <p> 如果计划启用 SnapVault 复制，则必须将保留计数设置为 2 或更高。如果将保留计数设置为1、则保留操作可能会失败、因为第一个快照是SnapVault关系的参考快照、直到将较新的快照复制到目标为止。</p> </div> <div style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;"> <p> 对于 ONTAP 9.4 或更高版本上的资源，最大保留值为 1018 ，对于 ONTAP 9.3 或更早版本上的资源，最大保留值为 254 。如果将保留设置为高于底层 ONTAP 版本支持的值，则备份将失败。</p> </div>
将Snapshot保留一定天数	选择*保留Snapshot副本的时间*、然后指定要在删除快照之前保留这些快照的天数。
Snapshot副本锁定期限	<p>选择Snapshot锁定期限、然后选择天、月或年。</p> <p>SnapLock保留期限应少于100年。</p>

7. 在\*复制\*页中，指定复制设置：

对于此字段 ...	执行此操作 ...
<ul style="list-style-type: none"> <li>• 创建本地 Snapshot 副本后更新 SnapMirror *</li> </ul>	<p>选择此字段可在另一个卷上创建备份集的镜像副本（SnapMirror 复制）。</p> <p>如果ONTAP中的保护关系类型为"镜像和存储"、而您仅选择此选项、则在主系统上创建的快照不会传输到目标系统、而是会在目标系统中列出。如果从目标中选择此Snapshot来执行还原操作、则会显示以下错误消息：Secondary Location is not available for the selected vaulel/Mirrored backup。</p> <p>在二级复制期间、SnapLock到期时间会加载主SnapLock到期时间。</p> <p>单击"Topology"页面中的*Refresh*按钮可刷新从ONTAP检索到的二级和主SnapLock到期时间。</p> <p>请参见 <a href="#">"在 Topology 页面中查看与自定义插件资源相关的备份和克隆"</a>。</p>
<ul style="list-style-type: none"> <li>• 创建本地 Snapshot 副本后更新 SnapVault *</li> </ul>	<p>选择此选项可执行磁盘到磁盘备份复制（SnapVault 备份）。</p> <p>在二级复制期间、SnapLock到期时间会加载主SnapLock到期时间。单击"Topology"页面中的*Refresh*按钮可刷新从ONTAP检索到的二级和主SnapLock到期时间。</p> <p>如果仅在ONTAP中称为SnapLock存储的二级系统上配置了SnapLock，则单击"拓扑"页面中的*Refresh*按钮可刷新从ONTAP检索到的二级系统上的锁定期限。</p> <p>有关SnapLock存储的详细信息、请参见将快照提交到存储上的WORM destination</p> <p>请参见 <a href="#">"在 Topology 页面中查看与自定义插件资源相关的备份和克隆"</a>。</p>
<ul style="list-style-type: none"> <li>• 二级策略标签 *</li> </ul>	<p>选择 Snapshot 标签。</p> <p>根据您选择的Snapshot标签、ONTAP会应用与该标签匹配的二级Snapshot保留策略。</p> <div style="border: 1px solid #ccc; padding: 10px; margin-top: 10px;"> <p> 如果选择了 * 创建本地 Snapshot 副本后更新 Snapmirror*，则可以选择指定二级策略标签。但是，如果在创建本地 Snapshot 副本之后选择了 * 更新 SnapVault *，则应指定二级策略标签。</p> </div>

对于此字段 ...	执行此操作 ...
<ul style="list-style-type: none"> <li>• 错误重试计数 *</li> </ul>	输入操作停止前允许的最大复制尝试次数。



您应在ONTAP中为二级存储配置SnapMirror保留策略、以避免达到二级存储上Snapshot的最大限制。

8. 查看摘要，然后单击 \* 完成 \*。

## 在SnapCenter 中创建资源组并附加策略

资源组是指必须向其添加要备份和保护的资源容器。通过它、您可以同时备份与给定应用程序关联的所有数据。您还必须将一个或多个策略附加到资源组，才能定义要执行的数据保护作业类型。

### 步骤

1. 在左侧导航窗格中，选择\*Resoles\*，然后从列表中选择相应的插件。
2. 在资源页面中、选择新建资源组。
3. 在名称页面中，执行以下操作：

对于此字段 ...	执行此操作 ...
Name	输入资源组的名称。  注意：资源组名称不应超过250个字符。
Tags	输入一个或多个标签，以帮助您稍后搜索资源组。  例如，如果您将HR作为标签添加到多个资源组，则以后可以找到与HR标签关联的所有资源组。
对 Snapshot 副本使用自定义名称格式	选中此复选框、然后输入要用于Snapshot名称的自定义名称格式。  例如， <i>customtext_resource group_policy_hostname</i> 或 <i>resource group_hostname</i> 。默认情况下、Snapshot名称会附加一个时间戳。

4. 可选：在资源页面中，从\*Host\*下拉列表中选择主机名，并从\*Resource Type\*下拉列表中选择资源类型。

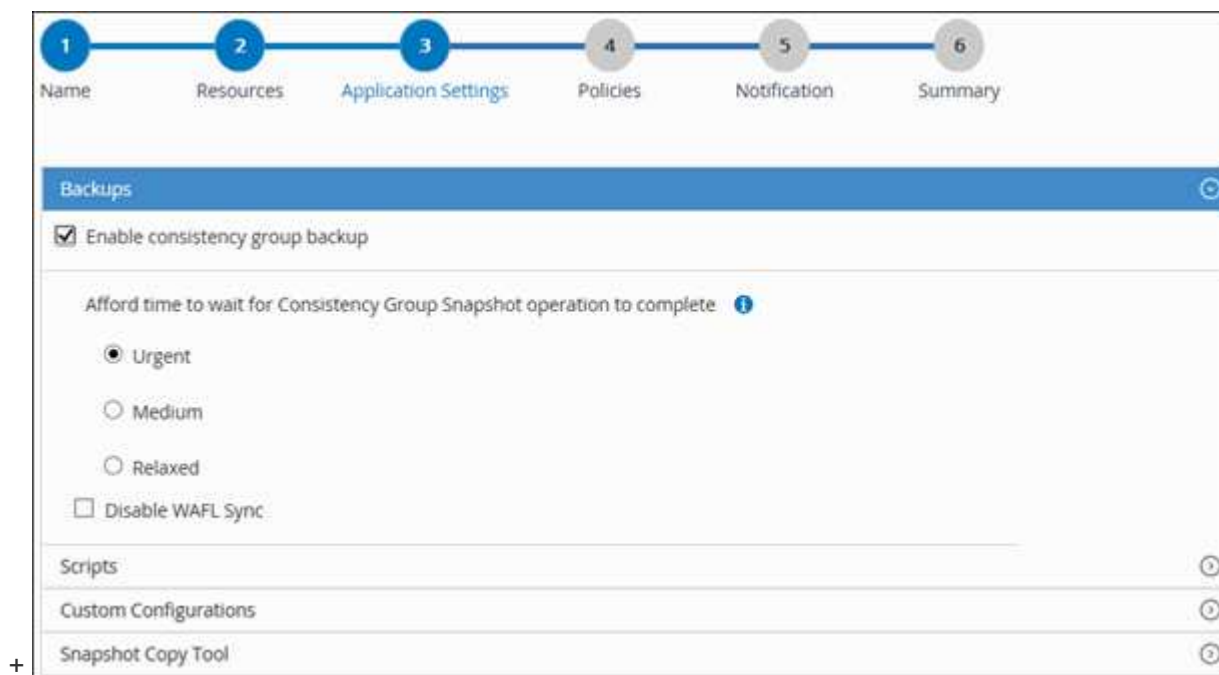
这有助于筛选屏幕上的信息。

5. 从“可用资源”部分中选择资源，然后选择右箭头将其移动到“选定资源”部分。
6. 可选：在\*Application Settings\*页中，执行以下操作：

- a. 选择备份箭头以设置其他备份选项：

启用一致性组备份并执行以下任务：

对于此字段 ...	执行此操作 ...
有时间等待一致性组 Snapshot 操作完成	选择紧急、中或宽松以指定完成Snapshot操作的等待时间。  紧急 = 5 秒，中等 = 7 秒，宽松 = 20 秒。
禁用 WAFL 同步	选择此选项可避免强制设置 WAFL 一致点。



- 选择Scripts (脚本)箭头、然后输入用于quiesce (静噪)、Snapshot (快照)和unquiesce (取消静噪)操作的pre和post命令。此外，您还可以输入在发生故障时退出之前要执行的 PRE 命令。
- 选择Custom Configurations箭头、然后输入使用此资源执行所有数据保护操作所需的自定义键值对。

参数	正在设置 ...	Description
archive_log_enable	( Y/N )	启用归档日志管理以删除归档日志。
archive_log_retention	天数	指定保留归档日志的天数。  此设置必须等于或大于 NTAP_snapshot_ 保留值。
Archive_log_DIR	change_info_directory/logs	指定归档日志所在目录的路径。

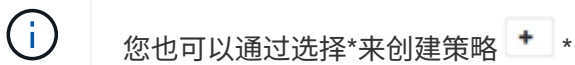
参数	正在设置 ...	Description
archive_log_EXT	file_extension	指定归档日志文件扩展名长度。  例如，如果归档日志为 log_backup_0_0_0_0.161518551942 9，而文件扩展名值为 5，则日志扩展名将保留 5 位数，即 16151。
archive_log_recursive_SE arch	( Y/N )	用于管理子目录中的归档日志。  如果归档日志位于子目录下，则应使用此参数。

c. 选择\*Snapshot副本工具\*箭头以选择用于创建Snapshot的工具：

如果您希望 ...	那么 ...
SnapCenter以使用适用于Windows的插件、并在创建快照之前将文件系统置于一致状态。对于Linux 资源，此选项不适用。	选择 * 具有文件系统一致性的 SnapCenter *。  此选项不适用于适用于 SAP HANA 数据库的 SnapCenter 插件。
SnapCenter以创建存储级别快照	选择 * 不使用文件系统一致性的 SnapCenter *。
输入要在主机上执行的命令以创建Snapshot。	选择*其他*、然后输入要在主机上执行的命令以创建Snapshot。

7. 在策略页面中，执行以下步骤：

a. 从下拉列表中选择一个或多个策略。



这些策略将在 \* 为选定策略配置计划 \* 部分中列出。

b. 在\*配置计划\*列中，选择\* + \* 表示要配置的策略。

c. 在为policy\_policy\_name\_添加计划对话框中、配置计划并选择确定。

其中policy\_name是选定策略的名称。

已配置的计划将列在 "applied Schedules" 列中。如果第三方备份计划与 SnapCenter 备份计划重叠，则不支持这些计划。

8. 从\*通知\*页面的\*电子邮件首选项\*下拉列表中，选择要发送电子邮件的场景。

您还必须指定发件人和收件人电子邮件地址以及电子邮件主题。必须在 \* 设置 \* > \* 全局设置 \* 中配置 SMTP 服务器。

9. 查看摘要，然后选择\*完成\*。

## 备份各个自定义插件资源



如果单个自定义插件资源不属于任何资源组，您可以从 " 资源 " 页面备份该资源。您可以按需备份资源，或者，如果资源已附加策略并配置了计划，则会根据计划自动进行备份。

### 开始之前

- 您必须已创建备份策略。
- 如果要备份与二级存储具有 SnapMirror 关系的资源，则分配给存储用户的 ONTAP 角色应包括 " snapmirror all " 权限。但是，如果您使用的是 " vsadmin " 角色，则不需要 " snapmirror all " 特权。

### 步骤

1. 在左侧导航窗格中，单击 \* 资源 \* ，然后从列表中选择相应的插件。
2. 在资源页面中，根据资源类型从 \* 视图 \* 下拉列表中筛选资源。

单击 ，然后选择主机名和资源类型以筛选资源。然后，您可以单击  以关闭筛选器窗格。

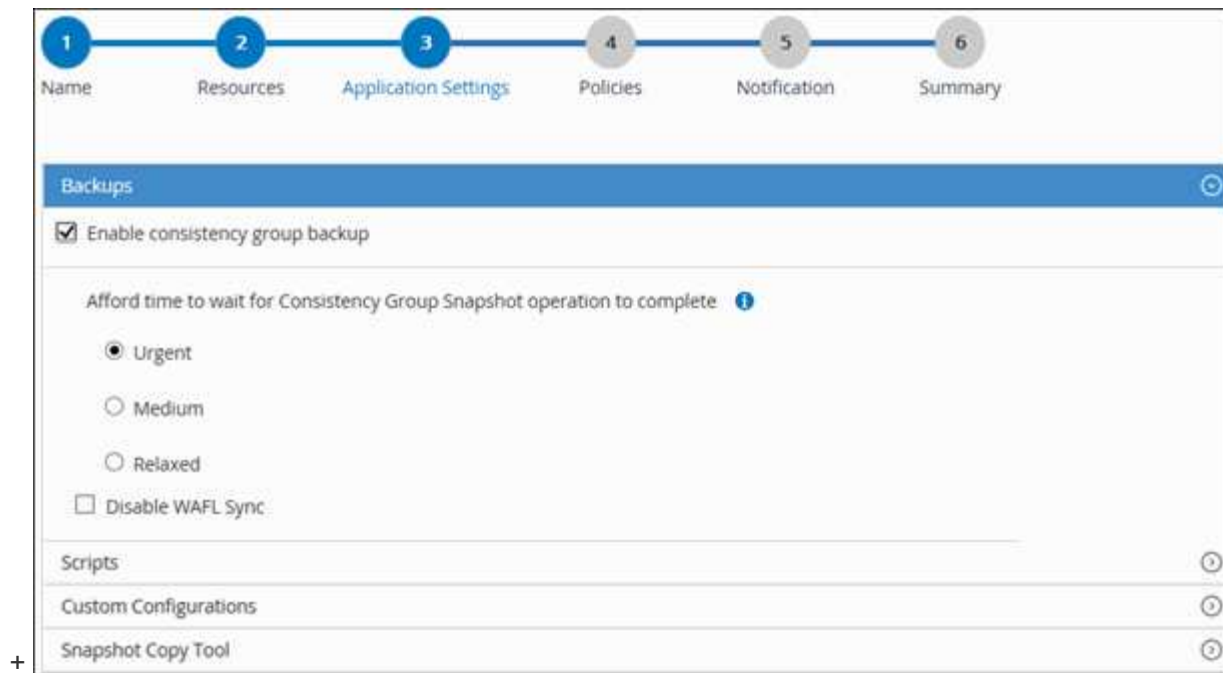
3. 单击要备份的资源。
4. 在资源页面中，如果要使用自定义名称，请选中\*Use custom name format for Snapshot copy\*复选框，然后为Snapshot名称输入自定义名称格式。

例如， *customtext\_policy\_hostname* 或 *resource\_hostname* 。默认情况下、Snapshot名称会附加一个时间戳。

5. 在应用程序设置页面中，执行以下操作：
  - a. 单击 \* 备份 \* 箭头可设置其他备份选项：

根据需要启用一致性组备份，并执行以下任务：

对于此字段 ...	执行此操作 ...
有时间等待一致性组 Snapshot 操作完成	选择紧急、中或宽松以指定完成Snapshot操作的等待时间。  紧急 = 5 秒，中等 = 7 秒，宽松 = 20 秒。
禁用 WAFL 同步	选择此选项可避免强制设置 WAFL 一致点。



- a. 单击\*Scripts\*箭头可运行用于静修、快照和取消静修操作的PRE和POST命令。您也可以在退出备份操作之前运行 PRE 命令。

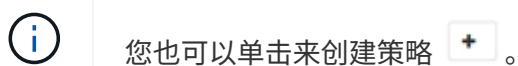
预处理脚本和后处理脚本在 SnapCenter 服务器中运行。

- b. 单击 \* 自定义配置 \* 箭头，然后输入使用此资源的所有作业所需的自定义值对。
- c. 单击\*Snapshot副本工具\*箭头选择用于创建Snapshot的工具：


如果您希望 ...	那么 ...
SnapCenter创建存储级别快照	选择 * 不使用文件系统一致性的 SnapCenter *。
SnapCenter以使用适用于Windows的插件将文件系统置于一致状态、然后创建Snapshot	选择 * 具有文件系统一致性的 SnapCenter *。
以输入命令以创建Snapshot	选择*其他*、然后输入命令创建Snapshot。

6. 在策略页面中，执行以下步骤：

- a. 从下拉列表中选择一个或多个策略。



在为选定策略配置计划部分中，将列出选定策略。

- b. 单击  在要配置计划的策略的 Configure Schedules 列中。
- c. 在添加策略 *policy\_name* 的计划对话框中，配置计划，然后单击 \* 确定 \*。

其中, *policy\_name* 是选定策略的名称。

已配置的计划将列在 "applied Schedules" 列中。

7. 在通知页面的 \* 电子邮件首选项 \* 下拉列表中, 选择要发送电子邮件的场景。

您还必须指定发件人和收件人电子邮件地址以及电子邮件主题。此外, 还必须在 \* 设置 \* > \* 全局设置 \* 中配置 SMTP。

8. 查看摘要, 然后单击 \* 完成 \*。

此时将显示资源拓扑页面。

9. 单击 \* 立即备份 \*。

10. 在备份页面中, 执行以下步骤:

- a. 如果已对资源应用多个策略, 请从 \* 策略 \* 下拉列表中选择要用于备份的策略。

如果为按需备份选择的策略与备份计划关联, 则按需备份将根据为计划类型指定的保留设置进行保留。

- b. 单击 \* 备份 \*。

11. 单击 \* 监控 \* > \* 作业 \* 以监控操作进度。

## 备份自定义插件资源的资源组



您可以从 "资源" 页面按需备份资源组。如果资源组附加了策略并配置了计划, 则会根据计划自动进行备份。

### 开始之前

- 您必须已创建附加了策略的资源组。
- 如果要将具有 SnapMirror 关系的资源备份到二级存储, 则分配给存储用户的 ONTAP 角色应包括 "snapmirror all" 特权。但是, 如果您使用的是 "vsadmin" 角色, 则不需要 "snapmirror all" 特权。

### 步骤

1. 在左侧导航窗格中, 单击 \* 资源 \* , 然后从列表中选择相应的插件。
2. 在资源页面中, 从 \* 视图 \* 列表中选择 \* 资源组 \*。

您可以通过在搜索框中输入资源组名称或单击来搜索资源组  并选择标记。然后, 您可以单击  以关闭筛选器窗格。

3. 在资源组页面中, 选择要备份的资源组, 然后单击 \* 立即备份 \*。
4. 在备份页面中, 执行以下步骤:

- a. 如果已将多个策略与资源组关联, 请从 \* 策略 \* 下拉列表中选择要用于备份的策略。

如果为按需备份选择的策略与备份计划关联, 则按需备份将根据为计划类型指定的保留设置进行保留。

- b. 单击 \* 备份 \*。



5. 单击 \* 监控 \* > \* 作业 \* 以监控操作进度。

- 在 MetroCluster 配置中，SnapCenter 可能无法在故障转移后检测到保护关系。

#### "MetroCluster 故障转移后无法检测 SnapMirror 或 SnapVault 关系"

- 如果要备份 VMDK 上的应用程序数据，并且适用于 VMware vSphere 的 SnapCenter 插件的 Java 堆大小不够大，则备份可能会失败。要增加 Java 堆大小，请找到脚本文件 `/opt/netapp/init_scripts/scvservice`。在该脚本中，`do_start` 方法命令将启动 SnapCenter VMware 插件服务。将该命令更新为以下命令：`java -JAR -Xmx8192M -Xms4096M`。

## 使用 PowerShell cmdlet 创建存储系统连接和凭据

在使用 PowerShell cmdlet 执行数据保护操作之前，您必须创建 Storage Virtual Machine (SVM) 连接和凭据。

开始之前

- 您应已准备好 PowerShell 环境以执行 PowerShell cmdlet。
- 您应具有创建存储连接所需的基础架构管理员角色权限。
- 您应确保插件安装未在进行中。

添加存储系统连接时，主机插件安装不得正在进行中，因为主机缓存可能不会更新，并且数据库状态可能会在 SnapCenter 图形用户界面中显示为 "不可用于备份" 或 "不在 NetApp 存储上"。

- 存储系统名称应是唯一的。

SnapCenter 不支持在不同集群上使用相同名称的多个存储系统。SnapCenter 支持的每个存储系统都应具有唯一的名称和唯一的管理 LIF IP 地址。

步骤

1. 使用 `Open-SmConnection` cmdlet 启动 PowerShell 连接会话。

此示例将打开 PowerShell 会话：

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

2. 使用 `Add-SmStorageConnection` cmdlet 创建与存储系统的新连接。

此示例将创建一个新的存储系统连接：

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

3. 使用 `Add-SmCredential` cmdlet 创建新凭据。

此示例将使用 Windows 凭据创建一个名为 FinanceAdmin 的新凭据：

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

有关可与 cmdlet 结合使用的参数及其说明的信息，可通过运行 `get-help command_name` 来获取。或者，您也可以参考 "《[SnapCenter 软件 cmdlet 参考指南](#)》"。

## 使用 PowerShell cmdlet 备份资源

备份资源包括与 SnapCenter 服务器建立连接，添加资源，添加策略，创建备份资源组和备份。

### 开始之前

- 您必须已准备好 PowerShell 环境以执行 PowerShell cmdlet。
- 您必须已添加存储系统连接并创建凭据。

### 关于此任务

有关可与 cmdlet 结合使用的参数及其说明的信息，可通过运行 `get-help command_name` 来获取。或者，您也可以参考 "《[SnapCenter 软件 cmdlet 参考指南](#)》"。

### 步骤

1. 使用 `Open-SmConnection` cmdlet 为指定用户启动与 SnapCenter 服务器的连接会话。

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

此时将显示用户名和密码提示符。

2. 使用 `Add-SmResources` cmdlet 添加资源。

此示例添加了以下资源：

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

3. 使用 `Add-SmPolicy` cmdlet 创建备份策略。

此示例将创建一个新的备份策略：

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

4. 使用 Add-SmResourceGroup cmdlet 向 SnapCenter 添加新资源组。

此示例将使用指定的策略和资源创建一个新的资源组：

```
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. 使用 New-SmBackup cmdlet 启动新的备份作业。

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

6. 使用 Get-SmBackupReport cmdlet 查看备份作业的状态。

此示例显示了在指定日期运行的所有作业的作业摘要报告：

```

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 :







```

## 监控自定义插件资源备份操作


您可以使用 `SnapCenterJobs` 页面监控不同备份操作的进度。您可能需要检查进度以确定完成时间或是否存在问题描述。

关于此任务


以下图标将显示在作业页面上，并指示相应的操作状态：

-  正在进行中
-  已成功完成
-  失败
-  已完成，但出现警告或由于出现警告而无法启动
-  已排队
-  已取消

## 步骤

1. 在左侧导航窗格中，单击 \* 监控 \*。
2. 在监控页面中，单击 \* 作业 \*。
3. 在作业页面中，执行以下步骤：
  - a. 单击  对列表进行筛选，以便仅列出备份操作。
  - b. 指定开始日期和结束日期。
  - c. 从 \* 类型 \* 下拉列表中，选择 \* 备份 \*。
  - d. 从 \* 状态 \* 下拉列表中，选择备份状态。
  - e. 单击 \* 应用 \* 以查看已成功完成的操作。
4. 选择备份作业，然后单击 \* 详细信息 \* 以查看作业详细信息。



但会显示备份作业状态 ，单击作业详细信息时，您可能会看到备份操作的某些子任务仍在进行中或标记有警告标志。


5. 在作业详细信息页面中，单击 \* 查看日志 \*。
  - 查看日志 \* 按钮可显示选定操作的详细日志。

## 取消自定义插件的备份操作

您可以取消已排队的备份操作。

- 您需要的内容 \*
  - 要取消操作，您必须以 SnapCenter 管理员或作业所有者的身份登录。
  - 您可以从 \* 监控 \* 页面或 \* 活动 \* 窗格取消备份操作。
  - 您不能取消正在运行的备份操作。
  - 您可以使用 SnapCenter 图形用户界面， PowerShell cmdlet 或命令行界面命令取消备份操作。
  - 对于无法取消的操作， \* 取消作业 \* 按钮处于禁用状态。
  - 如果在创建角色时选择了 \* 此角色的所有成员均可在用户 \ 组页面中查看和操作其他成员对象 \*，则可以在使用此角色时取消其他成员的排队备份操作。
  - 步骤 \*
1. 执行以下操作之一：

从 ...	Action
监控页面	<ol style="list-style-type: none"><li>a. 在左侧导航窗格中，单击 * 监控 * &gt; * 作业 *。</li><li>b. 选择操作，然后单击 * 取消作业 *。</li></ol>

从 ...	Action
活动窗格	a. 启动备份操作后，单击  以查看最近的五个操作。 b. 选择操作。 c. 在作业详细信息页面中，单击 * 取消作业 *。





此操作将被取消，并且资源将还原到先前的状态。

## 在 **Topology** 页面中查看与自定义插件资源相关的备份和克隆


在准备备份或克隆资源时，您可能会发现以图形方式显示主存储和二级存储上的所有备份和克隆非常有用。在拓扑页面中，您可以看到可用于选定资源或资源组的所有备份和克隆。您可以查看这些备份和克隆的详细信息，然后选择这些备份和克隆以执行数据保护操作。

### 关于此任务

您可以在管理副本视图中查看以下图标，以确定备份和克隆在主存储还是二级存储（镜像副本或存储副本）上可用。

-  显示主存储上可用的备份和克隆的数量。
-  显示使用 SnapMirror 技术在二级存储上镜像的备份和克隆的数量。
-  镜像存储类型卷上版本灵活的镜像备份的克隆将显示在拓扑视图中，但拓扑视图中的镜像备份计数不包括版本灵活的备份。
-  显示使用 SnapVault 技术在二级存储上复制的备份和克隆的数量。

显示的备份数包括从二级存储中删除的备份。例如，如果您使用策略创建了 6 个备份，仅保留 4 个备份，则显示的备份数为 6。

-  镜像存储类型卷上版本灵活的镜像备份的克隆将显示在拓扑视图中，但拓扑视图中的镜像备份计数不包括版本灵活的备份。

### 步骤

1. 在左侧导航窗格中，单击 \* 资源 \*，然后从列表中选择相应的插件。
2. 在资源页面中，从 \* 视图 \* 下拉列表中选择资源或资源组。
3. 从资源详细信息视图或资源组详细信息视图中选择资源。

如果资源受保护，则会显示选定资源的拓扑页面。

4. 查看摘要卡，查看主存储和二级存储上可用的备份和克隆数量的摘要。

摘要卡部分显示备份和克隆的总数。

单击刷新按钮可启动存储查询以显示准确的计数。

如果进行了启用了SnapLock的备份，则单击\*Refresh\*按钮将刷新从ONTAP检索到的主和二级SnapLock到期时间。每周计划还会刷新从ONTAP检索到的主和二级SnapLock到期时间。

如果应用程序资源分布在多个卷上、则备份的SnapLock到期时间将是为卷中的Snapshot设置的最长SnapLock到期时间。从ONTAP检索到最长的SnapLock到期时间。


按需备份后，单击\*Refresh\*按钮可刷新备份或克隆的详细信息。


5. 在管理副本视图中，单击主存储或二级存储中的 \* 备份 \* 或 \* 克隆 \* 以查看备份或克隆的详细信息。

备份和克隆的详细信息以表格式显示。

6. 从表中选择备份，然后单击数据保护图标以执行还原，克隆，重命名和删除操作。

 您不能重命名或删除二级存储系统上的备份。

 您不能重命名主存储系统上的备份。

7. 如果要删除克隆，请从表中选择此克隆，然后单击  删除克隆。

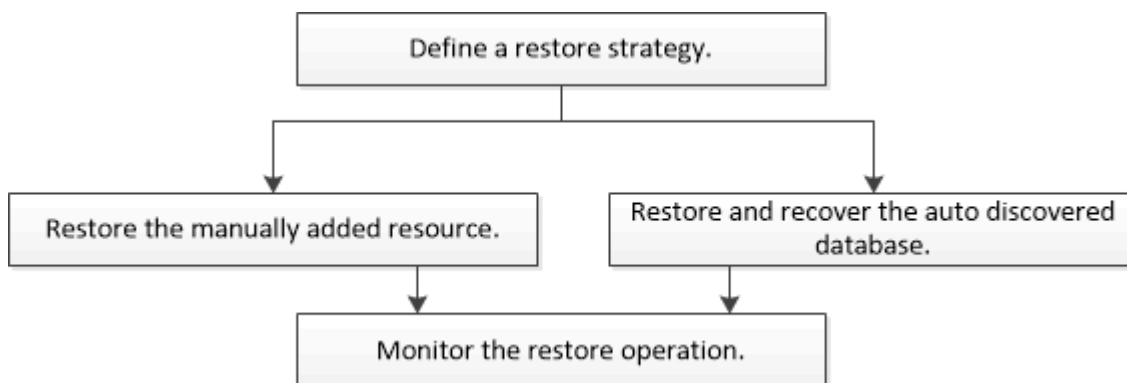
## 还原自定义插件资源

### 还原自定义插件资源

还原和恢复 workflow 包括规划，执行还原操作和监控操作。

关于此任务

以下 workflow 显示了必须执行还原操作的顺序：



您也可以手动或在脚本中使用 PowerShell cmdlet 执行备份，还原和克隆操作。有关 PowerShell cmdlet 的信息，请使用 SnapCenter cmdlet 帮助或参见 "《SnapCenter 软件 cmdlet 参考指南》"。

## 还原资源备份

您可以使用 SnapCenter 还原资源。还原操作的功能取决于您使用的插件。

### 开始之前

- 您必须已备份资源或资源组。
- 如果要将快照复制到镜像或存储、则SnapCenter管理员必须已为源卷和目标卷分配Storage Virtual Machine (SVM)。
- 您必须已取消要还原的资源或资源组当前正在进行的任何备份操作。

### 关于此任务

- 默认还原操作仅还原存储对象。只有在自定义插件提供应用程序级别的还原操作时，才能执行此功能。
- 对于ONTAP 9.12.1及更低版本、在还原过程中从SnapLock存储快照创建的克隆将继承SnapLock存储到期时间。存储管理员应在SnapLock到期时间后手动清理克隆。

### 步骤

1. 在左侧导航窗格中，单击 \* 资源 \*，然后从列表中选择相应的插件。
2. 在资源页面中，根据资源类型从 \* 视图 \* 下拉列表中筛选资源。

此时将显示资源以及类型，主机或集群名称，关联的资源组和策略以及状态等信息。




虽然备份可能针对某个资源组，但在还原时，您必须选择要还原的各个资源。


如果资源未受保护，则 *not protected* 将显示在 \* 整体状态 \* 列中。

"\* 整体状态 \*" 列中的状态 *not protected* 可能表示资源未受保护，或者资源已由其他用户备份。

3. 选择资源或选择一个资源组，然后选择该组中的一个资源。

此时将显示资源拓扑页面。

4. 在 \* 管理副本 \* 视图中，从主存储系统或二级（镜像或存储）存储系统中选择 \* 备份 \*。
5. 在主备份表中，选择要从中还原的备份，然后单击 。



Backup Name	End Date
rg1_scspr0191683001_01-05-2017_01.35.06.6463	1/5/2017 1:35:27 AM

6. 在还原范围页面中，选择 \* 完成资源 \* 或 \* 文件级别 \*。
  - a. 如果选择 \* 完成资源 \*，则会还原资源备份。

如果资源包含卷或qtrees作为存储占用空间、则此类卷或qtrees上较新的Snapshot将被删除、并且无法恢复。此外，如果任何其他资源托管在同一个卷或 qtree 上，则该资源也会被删除。



- b. 如果选择了 \* 文件级别 \* ，则可以选择 \* 所有 \* ，也可以选择卷或 qtree ，然后输入与选定卷或 qtree 相关的路径，并用逗号分隔。
    - 您可以选择多个卷和 qtree 。
    - 如果资源类型为 lun ，则会还原整个 LUN 。您可以选择多个 LUN 。+ 注意：如果选择 \* 全部 \* ，则卷， qtree 或 LUN 上的所有文件都将还原。
7. 在 \* 恢复类型 \* 页面中，执行以下步骤：选择选项以应用日志。在选择此插件之前，请确保此插件支持所有日志和日志，直到达到还原类型为止。

如果您要 ...	执行此操作 ...
还原所有日志	选择 * 所有日志 * 。确保此插件支持 * 所有日志 * 。
将所有日志还原到指定时间	选择 * 日志至 * 。确保此插件支持 * 日志至 * 。
还原资源备份	选择 * 无 * 。

- 8. 在 \* 操作前 \* 页面中，输入要在执行还原作业之前运行的还原前和卸载命令。
- 9. 在 \* 后操作 \* 页面中，输入要在执行还原作业后运行的 mount 和 post restore 命令。
- 10. 在 \* 通知 \* 页面的 \* 电子邮件首选项 \* 下拉列表中，选择要发送电子邮件的场景。
 

您还必须指定发件人和收件人电子邮件地址以及电子邮件主题。此外，还必须在 \* 设置 \* > \* 全局设置 \* 页面中配置 SMTP 。
- 11. 查看摘要，然后单击 \* 完成 \* 。
- 12. 单击 \* 监控 \* > \* 作业 \* 以监控操作进度。

## 使用 PowerShell cmdlet 还原资源

还原资源备份包括启动与 SnapCenter 服务器的连接会话，列出备份和检索备份信息以及还原备份。

您必须已准备好 PowerShell 环境以执行 PowerShell cmdlet 。

### 步骤

1. 使用 Open-SmConnection cmdlet 为指定用户启动与 SnapCenter 服务器的连接会话。

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

2. 使用 Get-SmBackup 和 Get-SmBackupReport cmdlet 检索有关要还原的一个或多个备份的信息。

此示例显示有关所有可用备份的信息：

```
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

此示例显示了有关 2015 年 1 月 29 日至 2015 年 2 月 3 日备份的详细信息：

```
PS C:\> Get-SmBackupReport -FromDate "1/29/2015" -ToDate "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. 使用 Restore-SmBackup cmdlet 从备份中还原数据。

```

Restore-SmBackup -PluginCode 'DummyPlugin' -AppObjectId
'scc54.sscore.test.com\DummyPlugin\NTP\DB1' -BackupId 269
-Confirm:$false
output:
Name                : Restore
'scc54.sscore.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         :

```

有关可与 cmdlet 结合使用的参数及其说明的信息，可通过运行 `get-help command_name` 来获取。或者，您也可以参考“《[SnapCenter 软件 cmdlet 参考指南](#)》”。



## 监控自定义插件资源还原操作





您可以使用作业页面监控不同 SnapCenter 还原操作的进度。您可能需要检查操作的进度，以确定操作何时完成或是否存在问题描述。

关于此任务


还原后状态用于描述还原操作之后资源的状况以及您可以执行的任何进一步还原操作。

以下图标将显示在作业页面上，并指示操作的状态：

-  正在进行中
-  已成功完成

-  失败
-  已完成，但出现警告或由于出现警告而无法启动
-  已排队
-  已取消

#### 步骤

1. 在左侧导航窗格中，单击 \* 监控 \*。
2. 在 \* 监控 \* 页面中，单击 \* 作业 \*。
3. 在 \* 作业 \* 页面中，执行以下步骤：
  - a. 单击  对列表进行筛选，以便仅列出还原操作。
  - b. 指定开始日期和结束日期。
  - c. 从 \* 类型 \* 下拉列表中，选择 \* 还原 \*。
  - d. 从 \* 状态 \* 下拉列表中，选择还原状态。
  - e. 单击 \* 应用 \* 以查看已成功完成的操作。
4. 选择还原作业，然后单击 \* 详细信息 \* 以查看作业详细信息。
5. 在 \* 作业详细信息 \* 页面中，单击 \* 查看日志 \*。
  - 查看日志 \* 按钮可显示选定操作的详细日志。

## 克隆自定义插件资源备份

### 克隆自定义插件资源备份

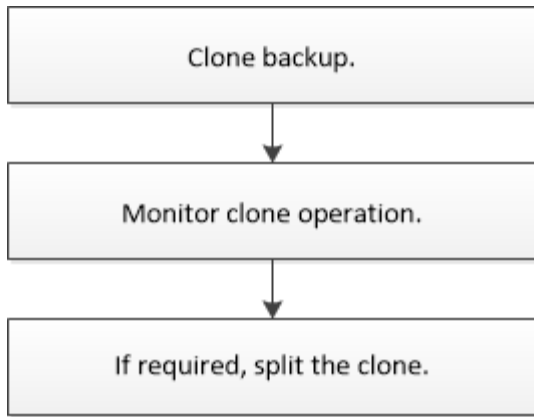
克隆 workflow 包括执行克隆操作和监控操作。

#### 关于此任务

克隆资源备份的原因如下：

- 测试在应用程序开发周期中必须使用当前资源结构和内容实施的功能
- 用于填充数据仓库时的数据提取和操作工具
- 恢复错误删除或更改的数据

以下 workflow 显示了必须执行克隆操作的顺序：



您也可以手动或在脚本中使用 PowerShell cmdlet 执行备份，还原和克隆操作。有关 PowerShell cmdlet 的详细信息，请使用 SnapCenter cmdlet 帮助或参见 "《SnapCenter 软件 cmdlet 参考指南》"。

## 从备份克隆

您可以使用 SnapCenter 克隆备份。您可以从主备份或二级备份克隆。克隆操作的功能取决于您使用的插件。

### 开始之前

- 您必须已备份资源或资源组。
- 默认克隆操作仅克隆存储对象。只有在自定义插件提供应用程序级别的克隆操作时，才能执行此功能。
- 您应确保托管卷的聚合应位于 Storage Virtual Machine (SVM) 的已分配聚合列表中。

### 关于此任务

对于 ONTAP 9.12.1 及更低版本、在还原过程中从 SnapLock 存储快照创建的克隆将继承 SnapLock 存储到期时间。存储管理员应在 SnapLock 到期时间后手动清理克隆。

### 步骤


1. 在左侧导航窗格中，单击 \* 资源 \*，然后从列表中选择相应的插件。
2. 在 \* 资源 \* 页面中，根据资源类型从 \* 视图 \* 下拉列表中筛选资源。

此时将显示资源以及类型，主机或集群名称，关联的资源组和策略以及状态等信息。

3. 选择资源或资源组。

如果选择资源组，则必须选择一个资源。

此时将显示资源或资源组拓扑页面。

4. 在管理副本视图中，从主存储系统或二级（镜像或存储）存储系统中选择 \* 备份 \*。
5. 从表中选择数据备份，然后单击 。
6. 在位置页面中，执行以下操作：

对于此字段 ...	执行此操作 ...
克隆服务器	默认情况下，系统会填充源主机。  如果要指定其他主机，请选择应挂载克隆并安装插件的主机。
克隆后缀	如果克隆目标与源相同，则必须执行此操作。  输入要附加到新克隆的资源名称的后缀。后缀可确保克隆的资源在主机上是唯一的。  例如 RS1_clone 。如果要克隆的资源与原始资源克隆到同一主机，则必须提供后缀以区分克隆的资源与原始资源；否则，操作将失败。

如果所选资源是 LUN ，并且您要从二级备份克隆，则会列出目标卷。单个源可以包含多个目标卷。

7. 在 \* 设置 \* 页面中，执行以下操作：

对于此字段 ...	执行此操作 ...
启动程序名称	输入主机启动程序名称，即 IQDN 或 WWPN 。
igroup 协议	选择 igroup protocol 。



只有当存储类型为 LUN 时，才会显示设置页面。

8. 在脚本页面中，输入克隆前或克隆后分别应在克隆操作之前或之后运行的命令。输入 mount 命令将文件系统挂载到主机。

例如：

- 预克隆命令：删除同名的现有数据库
- 克隆后命令：验证数据库或启动数据库。

Linux 计算机上卷或 qtree 的 mount 命令： `mount<vserver_name> : %<volume_name_Clone /mn>`

9. 在 \* 通知 \* 页面的 \* 电子邮件首选项 \* 下拉列表中，选择要发送电子邮件的场景。

您还必须指定发件人和收件人电子邮件地址以及电子邮件主题。

10. 查看摘要并单击 \* 完成 \* 。

11. 单击 \* 监控 \* > \* 作业 \* 以监控操作进度。

## 使用 PowerShell cmdlet 克隆备份

克隆 workflow 包括规划，执行克隆操作以及监控操作。

## 开始之前

您必须已准备好 PowerShell 环境以执行 PowerShell cmdlet。

有关 PowerShell cmdlet 的信息，请使用 SnapCenter cmdlet 帮助或参见 "《SnapCenter 软件 cmdlet 参考指南》"。

## 步骤

1. 使用 Open-SmConnection cmdlet 为指定用户启动与 SnapCenter 服务器的连接会话。

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

2. 列出可使用 Get-SmBackup 或 Get-SmResourceGroup cmdlet 克隆的备份。

此示例显示有关所有可用备份的信息：

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

BackupId	BackupName	BackupTime
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

此示例显示有关指定资源组的信息：

```
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 : {}  
HostResourceMapping : {}  
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                : {}
HostResourceMapping     : {}
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. 使用 New-SmClone cmdlet 从克隆资源组或现有备份启动克隆操作。

以下示例将使用所有日志从指定备份创建克隆：

```

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

```

### 4. 使用 Get-SmCloneReport cmdlet 查看克隆作业的状态。

此示例显示指定作业 ID 的克隆报告：

```
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          :
```

## 监控自定义插件资源克隆操作


您可以使用作业页面监控 SnapCenter 克隆操作的进度。您可能需要检查操作的进度，以确定操作何时完成或是否存在问题描述。

关于此任务

以下图标将显示在作业页面上，并指示操作的状态：

-  正在进行中
-  已成功完成
-  失败
-  已完成，但出现警告或由于出现警告而无法启动
-  已排队
-  已取消
- 步骤 \*

1. 在左侧导航窗格中，单击 \* 监控 \*。
2. 在 \* 监控 \* 页面中，单击 \* 作业 \*。

3. 在 \* 作业 \* 页面中，执行以下步骤：
  - a. 单击  对列表进行筛选，以便仅列出克隆操作。
  - b. 指定开始日期和结束日期。
  - c. 从 \* 类型 \* 下拉列表中，选择 \* 克隆 \*。
  - d. 从 \* 状态 \* 下拉列表中，选择克隆状态。
  - e. 单击 \* 应用 \* 以查看已成功完成的操作。
4. 选择克隆作业，然后单击 \* 详细信息 \* 以查看作业详细信息。
5. 在作业详细信息页面中，单击 \* 查看日志 \*。

## 版权信息

版权所有 © 2024 NetApp, Inc.。保留所有权利。中国印刷。未经版权所有者事先书面许可，本档中受版权保护的任何部分不得以任何形式或通过任何手段（图片、电子或机械方式，包括影印、录音、录像或存储在电子检索系统中）进行复制。

从受版权保护的 NetApp 资料派生的软件受以下许可和免责声明的约束：

本软件由 NetApp 按“原样”提供，不含任何明示或暗示担保，包括但不限于适销性以及针对特定用途的适用性的隐含担保，特此声明不承担任何责任。在任何情况下，对于因使用本软件而以任何方式造成的任何直接性、间接性、偶然性、特殊性、惩罚性或后果性损失（包括但不限于购买替代商品或服务；使用、数据或利润方面的损失；或者业务中断），无论原因如何以及基于何种责任理论，无论出于合同、严格责任或侵权行为（包括疏忽或其他行为），NetApp 均不承担责任，即使已被告知存在上述损失的可能性。

NetApp 保留在不另行通知的情况下随时对本文档所述的任何产品进行更改的权利。除非 NetApp 以书面形式明确同意，否则 NetApp 不承担因使用本文档所述产品而产生的任何责任或义务。使用或购买本产品不表示获得 NetApp 的任何专利权、商标权或任何其他知识产权许可。

本手册中描述的产品可能受一项或多项美国专利、外国专利或正在申请的专利的保护。

有限权利说明：政府使用、复制或公开本文档受 DFARS 252.227-7013（2014 年 2 月）和 FAR 52.227-19（2007 年 12 月）中“技术数据权利 — 非商用”条款第 (b)(3) 条规定的限制条件的约束。

本文档中所含数据与商业产品和/或商业服务（定义见 FAR 2.101）相关，属于 NetApp, Inc. 的专有信息。根据本协议提供的所有 NetApp 技术数据和计算机软件具有商业性质，并完全由私人出资开发。美国政府对这些数据的使用权具有非排他性、全球性、受限且不可撤销的许可，该许可既不可转让，也不可再许可，但仅限在与交付数据所依据的美国政府合同有关且受合同支持的情况下使用。除本文档规定的情形外，未经 NetApp, Inc. 事先书面批准，不得使用、披露、复制、修改、操作或显示这些数据。美国政府对国防部的授权仅限于 DFARS 的第 252.227-7015(b)（2014 年 2 月）条款中明确的权利。

## 商标信息

NetApp、NetApp 标识和 <http://www.netapp.com/TM> 上所列的商标是 NetApp, Inc. 的商标。其他公司和产品名称可能是其各自所有者的商标。