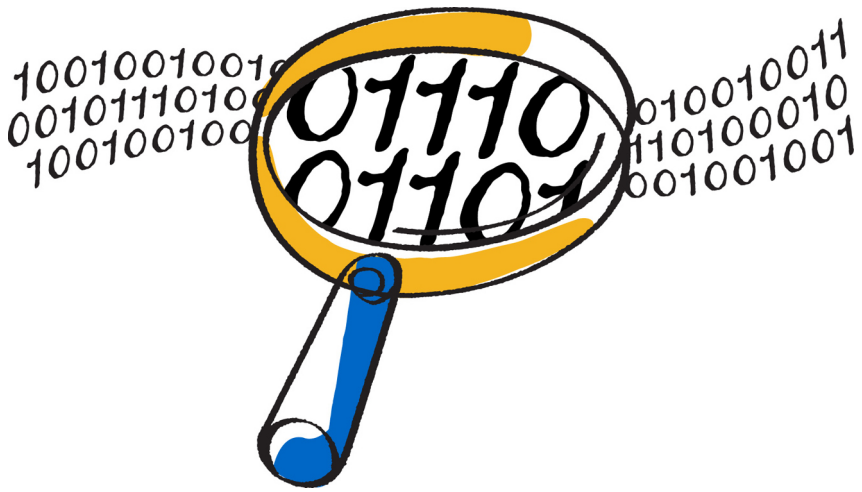
Virtual Storage Console 6.1 for VMware vSphere®

Reference Manual for APIs and PowerShell Cmdlets



NetApp, Inc.
495 East Java Drive
Sunnyvale, CA 94089
U.S.

Telephone: +1 (408) 822-6000
Fax: +1 (408) 822-4501
Support telephone: +1 (888) 463-8277
Web: www.netapp.com
Feedback: doccomments@netapp.com

Part number: 215-10306_A0_ur001
October 2015

Contents

APIs and cmdlets

Virtual Storage Console for VMware vSphere supports both programmable application interfaces (APIs) and PowerShell cmdlets. APIs work with backup and restore operations, and the cmdlets support other core VSC operations such as cloning and provisioning.

You can use these features to write applications and scripts that communicate with the VSC server at a command line level instead of a graphical user interface level.

Note: The cmdlets do not work with the standard roles that VSC provides for role-based access control (RBAC). To use the cmdlets, you must be logged in with vCenter administrator credentials.

Creating custom user accounts for backup and restore operations

You can override the default authentication method by creating a custom user account using the VSC CLI, which enables you to log in with user credentials other than your Windows credentials. A non-root or non-administrator account might be required to access a specific storage system.

You must create a custom storage system account with a new storage system role, group, and user as described in the following table.

Item	Description
Role	The new role must allow VSC to access the storage system data through its APIs.
Group	A storage system maintains groups as a collection of roles. The group you create must contain your new role.
User	A user account that VSC uses to access a storage system must be a member of a group that contains a role. You can create this user and assign a password to it, which enables you to add a storage system to VSC with the assigned user name and password.

For more information about how to manage users on your storage system, see your storage system's administrator guide.

Using the VSC CLI to create a custom user account

You can use the VSC CLI to create a custom user account. The user credentials for a custom user account provide the same access to commands and features as an administrator who logs in using the default Windows credentials authentication method.

Steps

1. Double-click the **VSC CLI** desktop icon or navigate to **Start > All Programs > NetApp > Virtual Storage Console > VSC CLI**.
2. Enter the following command:

   ```
   smvi servercredential set
   ```

3. Specify a user name and a password for this user account.

Using the VSC CLI to create a custom user account for a storage system

You can use the command line of your storage system to create a custom storage system account with a new role, group, and user for the storage system.

About this task

The following steps are performed from the command line of the storage system that VSC needs to access.

Steps

1. Create a role named api-access with the minimum configuration required for VSC to access the storage system:

   ```
   useradmin role add api-access -a api-*,login-http-admin,cli-ifconfig
   ```

2. Create a group named api-group that contains the api-access role:

   ```
   useradmin group add api-group -r api-access
   ```

3. Create a user named smvi-user as a member of the api-group group:

   ```
   useradmin user add smvi-user -g api-group
   ```

4. Set the user password by running the `passwd` command as root.

   The storage system prompts you for the account name that you want to change, followed by the new password for this account.

VSC CLI commands

The VSC for VMware vSphere command-line interface (CLI), which is labeled "VSC CLI" on your Windows desktop, provides you the benefits of a command-based view of the user interface. You can use this CLI to perform specific Backup and Recovery tasks, such as creating or deleting a backup of a virtual machine or datastore, as well as mounting a backup.

You should keep in mind the following information about the commands that you see in the interface:

- Virtual Storage Console commands are case-sensitive.
- There are no privilege levels; any user with a valid user name and password can run all commands.

For some commands, the following two parameters control the amount of output displayed:

verbose

This optional parameter provides detailed output when displaying information.

quiet

This optional parameter stops any output from displaying.

Note: Even with the quiet parameter specified, failed commands still display their failure messages.

Launching the VSC CLI

You can use either of two methods to launch the VS for VMware vSphere command-line interface (CLI), which is labeled "VSC CLI" on your Windows desktop. The first time you launch the VSC CLI, the application uses your Windows user credentials to grant you server access. Subsequent launches use stored credentials, speeding your access to the server.

About this task

When you issue your first CLI command, the CLI prompts you for your password and then runs the command. If the command succeeds, the CLI caches your user credentials and stores the information locally in an encrypted format.

Step

1. Double-click the **VSC CLI** icon or navigate to **Start > All Programs > NetApp > VSC CLI**.

   An alternative method to using your Windows user credentials is to use the `smvi servercredential set` command to create custom user credentials that allow you to log in to the server.

smvi backup create

The `smvi backup create` command creates a backup copy of a virtual machine or datastore. You can also perform this operation using the VSC for VMware vSphere user interface.

Syntax

```
smvi backup create    [-id {name | id} [name | id ...]    [-backup-name
{backup name}]    [-server {server name}]    [-include-independent] [-
```

```
exclude-datastores {name | id} [name | id ...] [-scripts {script name}]
[-vmware-snapshot] [-no-vmware-snapshot] [-update-mirror] [-allow-vault]
[-quiet]    [-verbose]    [-user]    [-help]
```

Parameters

[-id {*name* | *id*}] [*name* | *id* ...]

This mandatory parameter specifies the name or identification of the datastore or virtual machine that you are backing up. You can specify names or identifications of multiple datastores or virtual machines.

[-backup-name {*backup name*}]

This optional parameter specifies a backup copy name. After adding the flag, add a name for the backup copy. If you specify no name with this flag, the command fails. If you specify a name that contains only spaces, VSC for VMware vSphere automatically generates a name. If you specify a name that contains both spaces and other characters, VSC for VMware vSphere removes all leading and trailing spaces from the name.

[-server {*server name*}]

This optional parameter specifies the name of the SnapManager for Virtual Infrastructure server to which you are sending this command. The default value is localhost.

[-include-independent]

This optional parameter specifies that datastores containing only independent disks for a virtual machine are included in the backup job.

[-exclude-datastores {*name* | *id*}] [*name* | *id* ...]

This optional parameter specifies the name or identification of the datastores or virtual machines to be excluded from the backup job.

[-scripts {*script name*}]

This optional parameter specifies the name of the scripts to run with this backup job.

[-vmware-snapshot]

This optional parameter creates VMware snapshots of virtual machines during a backup operation. If you specify this parameter along with [`-no-vmware-snapshot`], or if you do not specify either one of the parameters, it prevents the creation of any VMware snapshots.

[-no-vmware-snapshot]

This optional parameter prevents the creation of VMware snapshots of virtual machines during a backup operation.

[-update-mirror]

This optional parameter initiates a SnapMirror image on the secondary storage.

[-allow-vault]

This optional parameter initiates a SnapVault image on the secondary storage.

[-quiet]

This optional parameter stops any output from displaying.

[-verbose]

This optional parameter provides detailed output when displaying information.

[-user]

Add this optional parameter if you want to log in to the SnapManager for Virtual Infrastructure server with different credentials than you are currently logged in with.

[-help]

This optional parameter displays help for this command.

Example: Creating a backup copy from a virtual machine

The following example creates a backup copy from a virtual machine named testInd, without specifying a backup name:

```
smvi backup create -id testInd

[18:19] Starting backup request
[18:19] Backing up datastore(s)([DS_vee_cmode_03-04
(4fd6aad6-579ff358-676a-e41 f13ba10f6), nfs_cmode_test1 (netfs://
10.60.189.79///vol_nfs_test1)])
[18:19] Backing up the following virtual machine(s) ([testInd])
[18:20] Creating storage snapshots for all datastores/virtual
machines that are being backed up.
[18:20] Storing logs for backup_43e100cdf8182d1ee79bae431f1c608f in
file .\repos itory\logs\unscheduled
\backup_backup_43e100cdf8182d1ee79bae431f1c608f.xml
[18:20] Backup backup_43e100cdf8182d1ee79bae431f1c608f of
datastores/virtual machines is complete.
SMVICLI-0100: Command completed successfully
```

smvi backup delete

The `smvi backup delete` command removes a virtual machine or datastore backup copy. You can also perform this operation using the VSC for VMware vSphere user interface.

Syntax

```
smvi backup delete    -backup-name {backup name}    [-server {server
name}]    [-quiet]    [-verbose]    [-noprompt]    [-user]    [-help]
```

Description

When you delete the most recent backup associated with a backup job, then the Last Run Status value displayed for that backup job in the Schedule Backup Jobs window is that of the most recent remaining undeleted backup copy associated with the backup job.

Parameters

[-backup-name {*backup name*}]

This mandatory parameter specifies the backup copy you want to delete. After adding the flag, add the name of the backup copy.

[-server {*server name*}]

This optional parameter specifies the name of the SnapManager for Virtual Infrastructure server to which you are sending this command. The default value is localhost.

[-quiet]

This optional parameter stops any output from displaying.

[-verbose]

This optional parameter provides detailed output when displaying information.

[-noprompt]

This optional parameter disables the default prompt that asks for confirmation when deleting a backup.

[-user]

Add this optional parameter if you want to log in to the SnapManager for Virtual Infrastructure server with different credentials than you are currently logged in with.

[-help]

This optional parameter displays help for this command.

Example: Deleting a backup copy

The following example deletes a backup copy named new-one:

```
smvi backup delete -backup-name new-one

Are you sure you want to proceed and remove backup named 'new-one'?
[yes|NO] y
[15:15] Removed backup with name "new-one"
SMVICLI-0100: Command completed successfully
```

smvi backup list

The `smvi backup list` command displays information, such as the file path on a storage system to the Snapshot copy, about all of the created and saved backups within a virtual machine or datastore. You can also perform this operation using the VSC for VMware vSphere user interface.

Syntax

```
smvi backup list    [-id {name | id} [name | id ...]]    [-mounted] [-
failed] [-recent] [-with-vmware-snapshot] [-sfr-mounted]        [-server
{server name}]    [-user]    [-help]
```

Parameters

[-id {*name* | *id*} [*name* | *id* ...]]

This mandatory parameter specifies the name or identification of the datastores or virtual machines that you want to list.

[-mounted]

This optional parameter lists all mounted backups.

[-failed]

This optional parameter lists all failed backups. The default list is only successful backups.

[-recent]

This optional parameter lists the most recent backup.

[-with-vmware-snapshot]

This optional parameter lists the backups that were taken with a VMware snapshot.

[-sfr-mounted]

This optional parameter lists the backups that were mounted for SFR.

[-server {*server name*}]

This optional parameter specifies the name of the SnapManager for Virtual Infrastructure server to which you are sending this command. The default value is localhost.

[-user]

Add this optional parameter if you want to log in to the SnapManager for Virtual Infrastructure server with different credentials than you are currently logged in with.

[-help]

This optional parameter displays help for this command.

Example: Listing backups in a datastore

The following example lists all of the backups within a datastore named data-store1:

```
smvi backup list -id data-store01

Id Name Date Entities Mounted VMware Snapshot Snapshot Name
-------- -------------------- --------------------
-------------------- -------- --------------- -------------
   backup_sch_1_20090122233100 Jan 22, 2009 23:31 vmfs_vm1 No
10.72.248.38:/vol/kas1_102_iscsi:smvi_backup_sch_1
20090122233100_36d2d99a-9ee0-4841-80c0-846698463e78_kas_sw_iscsi_ds1
```

smvi backup mount

The `smvi backup mount` command mounts a backup so that you can verify its contents.

Syntax

```
smvi backup mount    -backup-name {backup name}    -esx-server {esx
server name}    [-server {server name}]    [-quiet]    [-verbose]    [-
user]    [-help]
```

Privilege level

Note: To mount a VMFS datastore backup, the supplied ESX server must have SAN or iSAN access to the storage system, including required FC zoning or iSCSI discovery. To mount an NFS datastore backup, the supplied ESX server must be in the NFS export list of the original datastore.

Parameters

[-backup-name {*backup name*}]

This mandatory parameter specifies the backup you want to mount. After adding the flag, add the name of the backup.

[-esx-server {*esx server name* | *IP address*}]

This mandatory parameter specifies the name or IP address of the ESX server. This information describes where the backup resides on an ESX server.

Note: The server name is the name of the ESX server as viewed through the vSphere Client. This name might differ from the ESX server's host name or IP address.

[-server {*server name*}]

This optional parameter specifies the name of the SnapManager for Virtual Infrastructure server to which you are sending this command. The default value is localhost.

[-quiet]

This optional parameter stops any output from displaying.

[-verbose]

This optional parameter provides detailed output when displaying information.

[-user]

Add this optional parameter if you want to log in to the SnapManager for Virtual Infrastructure server with different credentials than you are currently logged in with.

[-help]

This optional parameter displays help for this command.

Example: Mounting a backup

The following example mounts a backup named vmfs2_vm1 on an ESX server with the IP address of 123.12.1.23:

```
smvi backup mount -backup-name vmfs2_vm1 -esx-server 123.12.1.23

[12:12] Starting mount request
SMVICLI-0100: Command completed successfully
```

smvi backup rename

The `smvi backup rename` command changes the name of a backup. Changing the name of a backup also changes the name on the corresponding storage Snapshot copy on the associated NetApp storage system. You can also perform this operation using the VSC for VMware vSphere user interface.

Syntax

```
smvi backup rename -backup-name {backup name} -new-backup-name {new name} [-server {server name}] [-user] [-help]
```

Parameters

[-backup-name {*backup name*}]

This mandatory parameter specifies the backup you want to rename. After adding the flag, add the name of the backup.

[-new-backup-name {*new name*}]

This mandatory parameter specifies the new name of the backup. After adding the flag, add a new name for the backup.

[-server-name {*server name*}]

This optional parameter specifies the name of the SnapManager for Virtual Infrastructure server to which you are sending this command.

[-user]

Add this optional parameter if you want to log in to the SnapManager for Virtual Infrastructure server with different credentials than you are currently logged in with.

[-help]

This optional parameter displays help for this command.

Example: Renaming a backup

The following example renames a backup named vmfs2-vm1 to volume-2:

```
smvi backup rename -backup-name vmfs2-vm1 -new-backup-name volume-2

[15:52] Backup "vmfs2-vm1" has been renamed to "volume-2"
SMVICLI-0100: Command completed successfully
```

smvi backup restore

The `smvi backup restore` command enables you to restore a virtual machine or datastore from a backup copy. You can also perform this operation using the VSC for VMware vSphere user interface.

Syntax

```
smvi backup restore    -id {name | id}    [-esx-server {esx server
name]    [-backup-name {backup name}]    [-vmdk {hard disk name}] [-
server {server name}]  [-restart-vm]     [-quiet]    [-verbose]    [-
noprompt]    [-user]    [-help]
```

Parameters

[-id {*name* | *id*}]

This mandatory parameter specifies the name or identification of the datastore or virtual machine that you are restoring.

[-esx-server {*esx server name* | *IP address*}]

This mandatory parameter specifies the name or IP address of the ESX server. The parameter is required when restoring a VMFS datastore, or a virtual machine that resides on a VMFS datastore, as well as when restoring an NFS virtual machine. The server name is the name of the ESX server as viewed through the vSphere Client. This name might differ from the host name or IP address of the ESX server.

[-backup-name {*backup name*}]

This optional parameter specifies which backup to restore. After adding the flag, you can add the name of the backup. If not specified, the latest available backup for the specified datastore or virtual machine is restored.

[-vmdk {*hard disk name*}]

This optional parameter specifies which hard disks are to be restored.

[-server {*server name*}]

This optional parameter specifies the name of the SnapManager for Virtual Infrastructure server to which you are sending this command. The default value is localhost.

[-restart-vm]

This optional parameter restarts the virtual machine after the restore operation.

[-quiet]

This optional parameter stops any output from displaying.

[-verbose]

This optional parameter provides detailed output when displaying information.

[-noprompt]

By default, a prompt appears, asking for confirmation when restoring a backup. This optional parameter disables the prompt.

[-user]

Add this optional parameter if you want to log in to the SnapManager for Virtual Infrastructure server with different credentials than you are currently logged in with.

[-help]

This optional parameter displays help for this command.

Example: Restoring a backup

The following example restores a virtual machine named nfs1-vm1 from a backup named backup-411:

```
smvi backup restore -id nfs1_vm1 -backup-name backup-411

Are you sure you want to proceed with this operation? [yes|NO] y
[11:04] Starting restore request
[11:04] [WARN] No active mounts found for datastore vmfs_ds1
(47ab69d8-e7c72da0-d6c5-001a6412251d)
[11:05] Restoring nfs virtual machine on folder 'nfs1_vm1'
[11:07] Reloading virtual machine
[11:07] Restore is complete
SMVICLI-0100: Command completed successfully
```

smvi backup unmount

The `smvi backup unmount` command unmounts a mounted virtual machine or datastore backup copy.

Syntax

```
smvi backup unmount -backup-name {backup name}    [-server {server
name}]    [-quiet]    [-verbose]    [-user]    [-help]
```

Description

Note: You must unmount a mounted backup copy to delete the backup copy or any of its preceding Snapshot copies.

Parameters

[-backup-name {*backup name*}]

This mandatory parameter specifies which backup copy to unmount. After adding the flag, add the name of the backup copy.

[-server {*server name*}]

This optional parameter specifies the name of the SnapManager for Virtual Infrastructure server to which you are sending this command. The default value is localhost.

[-quiet]

This optional parameter stops any output from displaying.

[-verbose]

This optional parameter provides detailed output when displaying information.

[-user]

Add this optional parameter if you want to log in to the SnapManager for Virtual Infrastructure server with different credentials than you are currently logged in with.

[-help]

This optional parameter displays help for this command.

Example: Unmounting a backup copy

The following example unmounts a backup copy named vmfs2_db:

```
smvi backup unmount -backup-name vmfs-2-db

[11:55] Starting unmount request
[11:55] Unmount is complete
SMVICLI-0100: Command completed successfully
```

smvi discover datastores

The `smvi discover datastores` command lists the datastores that are managed by the current vCenter Server and that reside on the storage systems currently assigned to your SnapManager for Virtual Infrastructure server.

Syntax

```
smvi discover datastores     [-help]
```

Parameters

[-help]

This optional parameter displays help for this command.

Example: Listing the datastores

The following example lists all the datastores managed by the current vCenter Server that reside on storage systems assigned to SnapManager for Virtual Infrastructure:

```
smvi discover datastores

Password for NETAPP\vanib: *********
Datacenter: Aladdin
Datastore: nfs_datastore6
NFS: 172.17.170.21:/vol/nfs_vol6
Datastore: nfs_datastore7
NFS: 172.17.170.21:/vol/nfs_vol7
Datastore: nfs_datastore7 (Backup test1)
NFS: 172.17.170.21:/vol/
nfs_vol7_mount_33e49878c5e74363825e84652a724aef
Datastore: nfs_datastore7 (Backup test0)
NFS: 172.17.170.21:/vol/
nfs_vol7_mount_90a6b1e7d6f948beaa6735af9692b3d4
Datastore: nfs_datastore7 (Backup backup_fgfdgfdgf_20080707134801)
NFS: 172.17.170.21:/vol/
nfs_vol7_mount_e50fc0eda0674cfbbf200f87f83ba8eb
Datastore: nfs_datastore8
NFS: 172.17.170.21:/vol/nfs_vol8
Datastore: nfs_datastore8 (Backup
backup_7d8597b0dfffd5c81806728dd45aea48)
NFS: 172.17.170.21:/vol/
nfs_vol8_mount_e7df47fbde00446cb6b589c821adc4dd
Datastore: vmfs_datastore5
LUN: 172.17.170.21:/vol/vmfs_vol5/lun5 Partition: 1
LUN: 172.17.170.21:/vol/vmfs_vol6/lun6 Partition: 1
Datastore: vmfs_datastore2
LUN: 172.17.170.21:/vol/vmfs_vol2/vmfs_lun2 Partition: 1
Datastore: vmfs_datastore3
LUN: 172.17.170.21:/vol/vmfs_vol3/lun3 Partition: 1
Datastore: vmfs_datastore4
LUN: 172.17.170.21:/vol/vmfs_vol4/lun4 Partition: 1
Datastore: vmfs_datastore7
LUN: 172.17.170.21:/vol/vmfs_vol7/qtree_vol7/lun7 Partition: 1
Datastore: snap-00000002-vmfs_datastore
LUN: 172.17.170.21:/vol/vmfs_vol1/vmfs_lun1 Partition: 1
Datastore: vmfs7_testAJ_1
LUN: 172.17.170.21:/vol/volaj1/lun1 Partition: 1
Datastore: vmfs7_testAJ-2
LUN: 172.17.170.21:/vol/volaj1/lun2 Partition: 1
Datacenter: Bellagio
```

smvi filerestore add-portgroup

The `smvi filerestore add-portgroup` command assigns virtual machines to a port group. You can also perform this operation using the VSC for VMware vSphere user interface.

Syntax

```
smvi filerestore add-portgroup [-name{port group name}] [-server{server
name}]     [-user] [-verbose]        [-help]
```

Parameters

`[-name {`*port group name*`}]`

This mandatory parameter specifies the name of the port group, or network, that is used to enable or disable administrator-assisted file-level restore operations.

`[-server {`*server name*`}]`

This optional parameter specifies the name of the SnapManager for Virtual Infrastructure server to which you are sending this command. The default value is localhost.

[-user]

Add this optional parameter if you want to log in to the SnapManager for Virtual Infrastructure server with different credentials than you are currently logged in with.

[-verbose]

This optional parameter provides detailed output when displaying information.

[-help]

This optional parameter displays help for this command.

smvi filerestore delete-portgroup

The `smvi filerestore delete-portgroup` command removes the port group and disables file restore sessions for the virtual machines assigned to the port group. You can also perform this operation using the VSC for VMware vSphere user interface.

Syntax

```
smvi filerestore delete-portgroup    [-name{port group name}]    [-
server {server name}]    [-user]    [-verbose]        [-help]
```

Parameters

[-name {*port group name*}]

This mandatory parameter specifies the name of the port group, or network, that is used to enable or disable administrator-assisted file-level restore operations.

[-server {*server name*}]

This optional parameter specifies the name of the SnapManager for Virtual Infrastructure server to which you are sending this command. The default value is localhost.

[-user]

Add this optional parameter if you want to log in to the SnapManager for Virtual Infrastructure server with different credentials than you are currently logged in with.

[-verbose]

This optional parameter provides detailed output when displaying information.

[-help]

This optional parameter displays help for this command.

smvi notification list

The `smvi notification list` command displays information about the alert notification.

Syntax

```
smvi notification list    [-server {server name}]    [-user] [-
verbose]    [-help]
```

Parameters

[-server {*server name*}]

This optional parameter specifies the name of the SnapManager for Virtual Infrastructure server to which you are sending this command. The default value is localhost.

[-user]

Add this optional parameter if you want to log in to the SnapManager for Virtual Infrastructure server with different credentials than you are currently logged in with.

[-verbose]

This optional parameter provides detailed output when displaying information.

[-help]

This optional parameter displays help for this command.

smvi notification set

The `smvi notification set` command displays information about the alert notification.

Syntax

```
smvi notification set    [-smtp server {dns name | ip address}]    [-
from {from email address}]    [-to {to email address}]        [-server
{server name}]    [-user] [-verbose]    [-help]
```

Parameters

[-smtp server {*dns name* | *ip address*}]

This mandatory parameter specifies the name or IP address of the SMTP server that handles the test notification e-mail.

[from {*from email address*}]

This mandatory parameter specifies the sender e-mail address.

[to {*to email address*}]

This mandatory parameter specifies the comma-separated list of recipient e-mail addresses.

[-server {*server name*}]

This optional parameter specifies the name of the SnapManager for Virtual Infrastructure server to which you are sending this command. The default value is localhost.

[-user]

Add this optional parameter if you want to log in to the SnapManager for Virtual Infrastructure server with different credentials than you are currently logged in with.

[-verbose]

This optional parameter provides detailed output when displaying information.

[-help]

This optional parameter displays help for this command.

smvi notification test

The `smvi notification test` command displays information about the test notification.

Syntax

```
smvi notification test    [-server {server name}]    [-user] [-
verbose]    [-help]
```

Parameters

[-server {*server name*}]

This optional parameter specifies the name of the SnapManager for Virtual Infrastructure server to which you are sending this command. The default value is localhost.

[-user]

Add this optional parameter if you want to log in to the SnapManager for Virtual Infrastructure server with different credentials than you are currently logged in with.

[-verbose]

This optional parameter provides detailed output when displaying information.

[-help]

This optional parameter displays help for this command.

smvi restoreagent set

The `smvi restoreagent set` command sets the default installation URL of the restore agent. You can also perform this operation using the VSC for VMware vSphere user interface.

Syntax

```
smvi restoreagent set    [-url]    [-server {server name}]    [-user]
[-verbose]    [-help]
```

Parameters

[-url]

This mandatory parameter provides an URL that points to a customer location for the restore agent installer.

[-server {*server name*}]

This optional parameter specifies the name of the SnapManager for Virtual Infrastructure server to which you are sending this command. The default value is localhost.

[-user]

Add this optional parameter if you want to log in to the SnapManager for Virtual Infrastructure server with different credentials than you are currently logged in with.

[-verbose]

This optional parameter provides detailed output when displaying information.

[-help]

This optional parameter displays help for this command.

smvi servercredential delete

The `smvi servercredential delete` command deletes a user account created by the `smvi servercredential set` command.

Syntax

```
smvi servercredential delete    -username {user name}    [-help]
```

Description

Note: You cannot run this command from a remote host.

Parameters

[-username {*user name*}]

This mandatory parameter specifies the internal user account that you want to delete.

[-help]

This optional parameter displays help for this command.

Example: Deleting a user account

The following example deletes the olduser2 user account:

```
smvi servercredential delete -username olduser2
SMVICLI-0100: Command completed successfully
```

smvi servercredential list

The `smvi servercredential list` command displays a user account created by the `smvi servercredential set` command.

Syntax

```
smvi servercredential list    [-help]
```

Description

Note: You cannot run this command from a remote host.

Parameters

[-help]

This optional parameter displays help for this command.

Example: Listing the server credentials

The following example lists the current SnapManager for Virtual Infrastructure server credentials:

```
smvi servercredential list

Username
-----------------------------------------------------
administrator
```

smvi servercredential set

The `smvi servercredential set` command adds a user account for Backup and Recovery capability to use for authentication instead of your Windows user credentials.

Syntax

```
smvi servercredential set    [-help]
```

Description

Note: You cannot run this command from a remote host.

Parameters

`[-help]`

This optional parameter displays help for this command.

Example: Adding a user account

The following example adds a user account named administrator and sets a seven character password:

```
smvi servercredential set

Username: administrator
Password: *******
SMVICLI-0100: Command completed successfully
```

smvi storagesystem list

The `smvi storagesystem list` command lists the added NetApp storage systems.

Syntax

```
               smvi storagesystem list               [-server {server
name}]             [-user]               [-verbose]               [-
help]
```

Parameters

[-server {*server name*}]

This optional parameter specifies the name of the SnapManager for Virtual Infrastructure server to which you are sending this command. The default value is localhost.

[-user]

Add this optional parameter if you want to log in to the SnapManager for Virtual Infrastructure server with different credentials than you are currently logged in with.

[-verbose]

This optional parameter provides detailed output when displaying information.

[-help]

This optional parameter displays help for this command.

Example: Listing the storage systems

The following example lists the NetApp storage systems that reside in the local SnapManager for Virtual Infrastructure server; in this case, a single storage system with an IP address of 123.17.170.21:

```
smvi storagesystem list

Name                                IP Address
----------------------------------- ---------------
123.17.170.21                       123.17.170.21
```

smvi version

The `smvi version` command displays the version of the VSC for VMware vSphere CLI and the SnapManager for Virtual Infrastructure server.

Syntax

```
smvi version    [-server {server name}]    [-help]
```

Parameters

[-server {*server name*}]

This optional parameter specifies the name of the SnapManager for Virtual Infrastructure server to which you are sending this command. The default value is localhost.

[-help]

This optional parameter displays help for this command.

Example: Displaying the version

The following example displays the VSC for VMware vSphere CLI and server version:

```
smvi version
SnapManager for Virtual Infrastructure CLI Rballys.4N_120127_0000
(Build: 120127
```

```
)
SnapManager for Virtual Infrastructure Server Rballys.
4N_120127_0000 (Build: 120
127)
```

VSC cmdlets

You can use PowerShell cmdlets to create scripts and applications that perform key Virtual Storage Console for VMware vSphere tasks without going through the VSC GUI. After you create an application that uses the cmdlets, you execute it from the command line.

Required credentials

You can only use the cmdlets if you have vCenter admnistrator credentials. The cmdlets do not work with the standard roles that VSC provides for role-based access control (RBAC).

Setting up the cmdlets

The cmdlet software is included as a separate .zip file in the VSC software package that you download from the NetApp Support Site. This file contains a VSC folder. To use the cmdlets, you extract the contents of the .zip file containing the cmdlets into a Windows PowerShell directory in your documents directory. If you are not familiar with setting up cmdlets, here is a high-level overview. If you need additional information, see the Windows PowerShell documentation.

1. Extract the .zip file to `$Home\Documents\WindowsPowerShell\Modules`.
   For example, you might extract the cmdlets to the following directory:
   `C:\Users\Administrator\My Documents\WindowsPowerShell\Modules`
2. Open a Windows PowerShell window and enter the following command to unblock the files:
   `gci -R $Home\Documents\WindowsPowerShell\Modules\VSC | Unblock-File`
3. Open a new Windows PowerShell window and run the following command:
   `Import-Module VSC`

Common parameters

All the VSC cmdlets support the common parameters:

- Verbose
- Debug
- ErrorAction
- ErrorVariable
- WarningAction
- WarningVariable
- OutBuffer
- OutVariable

For more information, see *about_CommonParameters* (*http://go.microsoft.com/fwlink/?LinkID=113216*).

Add-VscDatastore

You can use the `Add-VscDatastore` cmdlet to add a Virtual Storage Console for VMware vSphere datastore to the storage system you specify.

Syntax

```
Add-VscDatastore [-storageSystemId] <String> [-datastoreName]
  <String> [-hostName] <String> [-datastoreProtocol]
  <String> [-containerName] <String> [-isThin] <Boolean>
  [-datastoreSizeGB] <Int64> [[-blockSizeMB] [<Int32>]]
  [[-policyName] <String>] [[-svm] <String>]
  [<CommonParameters>]
```

Parameters

`-storageSystemId` <*`String`*>

Provides the ID of the storage system used to provision the new datastore.

Attribute	Value
Required?	true
Position?	1
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

`-datastoreName` <*`String`*>

Provides the name of the datastore that you are adding.

Attribute	Value
Required?	true
Position?	2
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

`-hostName` <*`String`*>

Provides the name of the ESX host.

Attribute	Value
Required?	true
Position?	3
Default value	N/A
Accept pipeline input?	false

Attribute	Value
Accept wildcard characters?	false

`-datastoreProtocol <String>`

Specifies which protocol the datastore uses; for example, NFS.

Attribute	Value
Required?	true
Position?	4
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

`-containerName <String>`

Provides the name of the container. If you are using NFS, the container is an aggregate. If you are using VMFS, the container is the name of an existing volume.

Attribute	Value
Required?	true
Position?	5
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

`-isThin <Boolean>`

Specifies whether the datastore uses thin provisioning:

- $true: Thin provisioning is used.
- $false: Thin provisioning is **not** used.

Attribute	Value
Required?	true
Position?	6
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

`-datastoreSizeGB <Int64>`

Specifies in gigabytes the size of the datastore being added.

Attribute	Value
Required?	true

Attribute	Value
Position?	7
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

`-blockSizeMB <Int32>`

Specifies in megabytes the size of the block size that the datastore being added is using.

Attribute	Value
Required?	false
Position?	8
Default value	1 block
Accept pipeline input?	false
Accept wildcard characters?	false

`-policyName <String>`

Specifies the policy name. To use this option, you must have VASA Provider for clustered Data ONTAP installed so that you can use its storage capability profile feature.

Attribute	Information
Required?	false
Position?	9
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

`-vFilerContext <String>`

Specifies the vFiler unit name.

Attribute	Information
Required?	false
Position?	named
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

`-svm <String>`

Specifies the Storage Virtual Machine (SVM) name.

Attribute	Information
Required?	false

Attribute	Information
Position?	named
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

Outputs

Creates the ID number of the task.

Example that uses parameter names

```
C:\PS>Add-VscDatastore -storageSystemId
11870a5c-984d-11e1-9c8f-123478563412 -svm manhar_vs1_21a -datastoreName
    nfsCmdletCan -hostName 10.777.10.777 -datastoreProtocol NFS -
containerName manhar_aggr1 -isThin $TRUE
    -datastoreSizeGB 5 -blockSizeMB 1
```

```
PS C:\Source\VSC_Commandlets\Tests> Add-VscDatastore -storageSystemId
11870a5c-984d-11e1-9c8f-123478563412 -svm
    manhar_vs1_21a -datastoreName nfsCmdletCan -hostName 10.777.10.777 -
datastoreProtocol NFS -containerName
    manhar_aggr1 -isThin $TRUE -datastoreSizeGB 5 -blockSizeMB 1
    9
```

Add-VscStorageSystem

You can use the `Add-VscStorageSystem` cmdlet to add a storage system to Virtual Storage Console for VMware vSphere. You can use the cmdlet to provide the user name/password credentials required to log onto that storage system and then perform tasks on it. These credentials support the role-based access control (RBAC) that you set up within Data ONTAP.

You can also specify whether VSC uses Transport LayerSecurity to communicate with the storage system.

Syntax

```
Add-VscStorageSystem [-ipAddress] <String>
   [[-Credential] [<PSCredential>]] [-useTLS] <Boolean>
   [-port] <Int32> [<CommonParameters>]
```

Parameters

`-ipAddress <String>`

Provides the IP address of the storage system you want to add.

Attribute	Value
Required?	true
Position?	1
Default value	N/A

Attribute	Value
Accept pipeline input?	true (`ByPropertyName`)
Accept wildcard characters?	false

`-Credential <PSCredential>`

Provides the System.Management.Automation.PSCredential object that contains the credentials that alllow you to log onto the storage system.

Attribute	Value
Required?	true
Position?	4
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

`-useTLS [<Boolean>]`

Specifies whether the datastore uses TLS to communicate with the storage system:

- $true: TLS is used
- $false: TLS is **not** used.

Attribute	Value
Required?	true
Position?	4
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

`-port <Int32>`

Provides the number of the management port that the storage system connects to.

Attribute	Value
Required?	true
Position?	5
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

Outputs

Returns information contained in the storageSystemDetails object.

Example that uses parameter names

```
C:\PS>Add-VscStorageSystem -ipAddress 10.777.10.777 -password abc123 -
port 80 -username root -useTLS $False
```

```
PS C:\Source\VSC_Commandlets\PesterTests> Add-VscStorageSystem -
ipAddress 10.777.10.777 -password abc123 -port 80
    -username root -useTLS $False

    adapter                   :
    aggregate                 :
    availableAggregateNames   :
    availableInterfaceNames   :
    availableVolumeNames      :
    cfMode                    :
    cfStatus                  :
    clusterAdmin              : False
    clusterNode               : False
    clusterVserver            : False
    createVolumeWrapper       : False
    deleteVolumeIfLastLun     : False
    flagSet                   : 1
    flagSetSpecified          : True
    interfaces                :
    ipAddressList             : {10.777.10.788}
    luns                      : {}
    model                     :
    multistore                : False
    partner                   :
    password                  : 999123
    port                      : 80
    portSpecified             : True
    qtree                     : {}
    spaceReserveNewVolumes    : False
    storageSystemGroupingName :
    thinProvisionVolumeClones : False
    updating                  : False
    usableAggregateNames      :
    usableInterfaceNames      :
    usableVolumeNames         :
    useTLS                    : False
    username                  : root
    vaaiCapable               :
    volumeWrapperBuffer       : 0
    volumes                   : {}
    clusteredOntap            : False
    id                        : 24e7b300-a2d7-4c1e-be3a-c75d45b429c1
    ipAddress                 : 10.61.123.12
    name                      : -unknown-
    overallStatus             : -1
    overallStatusSpecified    : True
    overallStatusReason       :
    storageSystemProtocol     : VSC.VscApi.storageSystemProtocol
    supportedProtocols        : 8
    totalAllocated            : 0
    totalAllocatedSpecified   : True
    totalCapacity             : 0
    totalCapacitySpecified    : True
    totalFree                 : 0
    totalFreeSpecified        : True
    totalUsed                 : 0
    totalUsedSpecified        : True
    type                      : SEVEN_MODE_STORAGE_SYSTEM
    typeSpecified             : True
    version                   :
```

Connect-VscServer

You can use the `Connect-VscServer` cmdlet to establish a connection to the VSC server. You must have administrator privileges to use this cmdlet.

Syntax

```
Connect-VscServer [-Server] <String> [[-Port] <UInt16>]
  [[-Credential] [<PSCredential>]] [<CommonParameters>]
```

Parameters

`-server` `<String>`

Provides the IP address of the VSC server.

Attribute	Value
Required?	true
Position?	1
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

`-Port` `<UInt16>`

Provides the port number that the VSC server uses.

Attribute	Value
Required?	false
Position?	4
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

`-Credential` `[<PSCredential>]`

Providers the System.Management.Automation.PSCredential object that contains the credentials required to log into the vCenter server.

Attribute	Value
Required?	false
Position?	4
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

Outputs

Prints Connected if a connection to the VSC server is made successfully.

Example that uses parameter names

```
C:\PS>Connect-VscServer -server "10.777.10.777"
```

```
PS C:\Source\VSC_Commandlets\Tests\mhc> Connect-VscServer -server
"10.777.10.777"
```

Disconnect-VscServer

You can use the `Disconnect-VscServer` cmdlet to disconnect from the VSC server. You do not need to specify any parameters for this cmdlet.

Syntax

```
Disconnect-VscServer [<CommonParameters>]
```

Outputs

Prints `Disconnected` if the connection to the VSC server is successfully disconnected. Prints `Failed`
if the connection continues.

Example that uses parameter names

```
C:\PS>Disconnect-VscServer
```

```
PS C:\Source\VSC_Commandlets\Tests\mhc> Disconnect-VscServer
    True
```

Get-VscConnectionBroker

You can use the `Get-VscConnectionBroker` cmdlet to get information about all of the connection brokers configured in Virtual Storage Console for VMware vSphere.You do not need to enter any parameters for this cmdlet.

Syntax

```
Get-VscConnectionBroker [<CommonParameters>]
```

Outputs

Displays the connection broker information that is contained in the connectionBroker object.

Example that uses parameter names

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

```
PS C:\Source\VSC_Commandlets\Tests> Get-VscConnectionBroker

    connectionBrokerType : VMWARE_VIEW_5_0
    connectionName       :
    domain               : world.lab.com
    host                 : 10.777.10.777
    password             : wsqa
    username             : administrator
```

Get-VscDatastore

You can use the `Get-VscDatastore` cmdlet to get information about all the datastores on the host system that can be mounted.

Syntax

```
Get-VscDatastore [-hostName] <String> [<CommonParameters>]
```

Parameters

`-hostName <String>`

Provides the name of the host.

Attribute	Value
Required?	true
Position?	1
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

Outputs

Displays information about each of the datastores on the host that is mountable.

Example that uses parameter names

```
 C:\PS>Get-VscDatastore -hostName vmshe-hp380-02.lab.com
```

```
PS C:\Source\VSC_Commandlets\Tests> Get-VscDatastore -hostName vmshe-
hp380-02.lab.com


    autoGrow                      : False
    clusterName                   :
    containerName                 :
    datastoreName                 :
    datastoreNames                :
    datastoreOffsetGroup          : 0
    datastoreOffsetGroupSpecified : False
    datastoreType                 : NFS
    dedupeEnabled                 : False
    excludedFromScan              : False
    filerId                       : 0151753861-0
    freeSharedSpace               : 0
```

```
    freeSpace                       : 1073442816
    growIncrement                   : 0
    hostIp                          : 10.777.10.777
    hostNames                       :
    lastScanned                     : 1/1/0001 12:00:00 AM
    lastScannedSpecified            : False
    lun                             :
    mappedToQtreeWithQuota          : False
    maxGrowSize                     : 0
    mor                             : Datastore:datastore-16
    name                            : testAG
    numberOfDatastores              : 0
    optimized                       : False
    optimizedSpecified              : False
    protocol                        : NFS
    provisioningTargetStoragePod    :
    storagePod                      :
    storageSystemId                 :
    thinProvision                   : False
    totalSpace                      : 1073741824
    uuid                            :
    virtualMachines                 :
    vmfsBlockSizeMB                 : 0
    vmfsBlockSizeMBSpecified        : False
    vmfsMajorVersion                : 0
    vmfsVersion                     :
    volume                          : VSC.VscApi.volume
    wrapperVol                      : False

    autoGrow                        : False
    clusterName                     :
    containerName                   :
    datastoreName                   :
    datastoreNames                  :
    datastoreOffsetGroup            : 0
    datastoreOffsetGroupSpecified   : False
    datastoreType                   : NFS
    dedupeEnabled                   : False
    excludedFromScan                : False
    filerId                         : 0151753861-0
    freeSharedSpace                 : 0
    freeSpace                       : 42936819712
    growIncrement                   : 0
    hostIp                          : 10.777.10.777
    hostNames                       :
    lastScanned                     : 1/1/0001 12:00:00 AM
    lastScannedSpecified            : False
    lun                             :
    mappedToQtreeWithQuota          : False
    maxGrowSize                     : 0
    mor                             : Datastore:datastore-17
    name                            : AGsourceds
    numberOfDatastores              : 0
    optimized                       : False
    optimizedSpecified              : False
    protocol                        : NFS
    provisioningTargetStoragePod    :
    storagePod                      :
    storageSystemId                 :
    thinProvision                   : False
    totalSpace                      : 42949672960
    uuid                            :
    virtualMachines                 :
    vmfsBlockSizeMB                 : 0
    vmfsBlockSizeMBSpecified        : False
    vmfsMajorVersion                : 0
    vmfsVersion                     :
    volume                          : VSC.VscApi.volume
    wrapperVol                      : False
```

Get-VscHostList

The `Get-VscHostList` cmdlet returns the list of hosts that have been added to the VSC server.You do not need to specify any parameters for this cmdlet.

Syntax

```
Get-VscHostList [<CommonParameters>]
```

Outputs

Displays information about each host that has been added to the VSC server.

Example that uses parameter names

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

```
PS C:\Source\VSC_Commandlets\Tests\mhc> Get-VscHostList


    adapterSettings      : True
    hgBuildVersion       :
    hgVmotionCompatible  : False
    ipAddress            : 10.777.10.777
    monitorStatus        : unknown
    moref                : HostSystem:host-9
    mpioSettings         : True
    name                 : 10.777.10.777
    nfsSettings          : False
    password             :
    patchCompatible      : False
    skipped              : False
    statusReason         :
    username             :
    version              : 5.1.0
    virtualMachineMorefs :
    qFullSettings        : True
```

Get-VscNasDetails

The `Get-VscNasDetails` cmdlet returns details about the specified NAS datastore.

Syntax

```
Get-VscNasDetails [-datastoreName] <String> [<CommonParameters>]
```

`-datastoreName` <*`String`*>

Provides the name of the datastore about which you want to get information.

Attribute	Value
Required?	true
Position?	1

Attribute	Value
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

Outputs

Displays information about the NAS datastore you specified.

Example that uses parameter names

```
C:\PS>Get-VscNasDetails -datastoreName AGNFSAUTOUPDATE
```

```
PS C:\Source\VSC_Commandlets\Tests> Get-VscNasDetails -datastoreName
AGNFSAUTOUPDATE


    aggrfree                          : 0
    aggrfreeSpecified                 : False
    aggrsize                          : 0
    aggrsizeSpecified                 : False
    aggrstatus                        :
    aggrusage                         : 0
    aggrusageSpecified                : False
    aggrused                          : 0
    aggrusedSpecified                 : False
    anonymoususer                     :
    clusterparent                     :
    datastore                         :
    datastorecapacity                 : 0
    datastorecapacitySpecified        : False
    datastorefree                     : 0
    datastorefreeSpecified            : False
    datastorestatus                   :
    datastoreusage                    : 0
    datastoreusageSpecified           : False
    datastoreused                     : 0
    datastoreusedSpecified            : False
    dedupeData                        :
    exportpath                        :
    ipaddress                         :
    moref                             :
    mountedhosts                      :
    name                              :
    partner                           :
    qtreeactualpath                   :
    qtreesecurity                     :
    qtreestatus                       :
    readonlyhosts                     :
    readwritehosts                    :
    rootaccesshosts                   :
    vmfspath                          :
    volaggregate                      :
    volautogrowenabled                : False
    volautogrowenabledSpecified       : False
    volautogrowinc                    : 0
    volautogrowincSpecified           : False
    volautogrowmax                    : 0
    volautogrowmaxSpecified           : False
    volcapacity                       : 0
    volcapacitySpecified              : False
    volconvertucode                   :
    volcreateucode                    :
    volfractionalreserve              :
```

```
    volfree                              : 0
    volfreeSpecified                     : False
    volguarantee                         :
    volname                              :
    volnoatimeupdate                     :
    volreserve                           :
    volsnapautodeletestate               :
    volsnapautodeletetrigger             :
    volsnapblocksreserved                : 0
    volsnapblocksreservedSpecified       : False
    volsnapreserveusedactual             : 0
    volsnapreserveusedactualSpecified    : False
    volstatus                            :
    voltype                              :
    volusage                             : 0
    volusageSpecified                    : False
    volused                              : 0
    volusedSpecified                     : False
```

Get-VscProvisionableStorageSystem

You can use the `Get-VscProvisionableStorageSystem` cmdlet to return a list of all the storage systems managed by Virtual Storage Console for VMware vSphere that you can use for provisioning storage.

Syntax

```
Get-VscProvisionableStorageSystem [-datastoreType] <String>
   [-protocol] <String> [<CommonParameters>]
```

Parameters

`-datastoreType` *<String>*

Provides the type of datastore system; for example, NFS.

Attribute	Value
Required?	true
Position?	2
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

`-protocol` *<String>*

Specifies whether the storage system uses the ISCSI or FCP protocol.

- iscsi
- fcp

Attribute	Value
Required?	true
Position?	3

Attribute	Value
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

Outputs

Displays the provisioningTargetStorageSystem object, which contains information about each of the storage systems and whether you can provision storage on it.

Example that uses parameter names

```
C:\PS>Get-VscProvisionableStorageSystem -datastoreType NFS -protocol
iscsi
```

```
PS C:\Source\VSC_Commandlets\Tests> Get-VscProvisionableStorageSystem -
datastoreType NFS -protocol iscsi


    actuallyOpsMgr             : False
    children                   : {vfilerx2xaf14b}
    cloningErrors              :
    cloningWarnings            :
    errors                     :
    ipAddress                  : 10.777.10.777
    name                       : afiler-14b
    passthroughContext         :
    password                   :
    port                       : 0
    provisionable              : True
    ssl                        : False
    storageSystemType          : CLUSTER
    storageSystemTypeSpecified : False
    storageSystemTypeAsString  :
    username                   :
    version                    :
    warnings                   :

    actuallyOpsMgr             : False
    children                   : {vmshe-fas3070-08-vFiler01}
    cloningErrors              :
    cloningWarnings            :
    errors                     :
    ipAddress                  : vmshe-fas3070-08.lab.com
    name                       : vmshe-fas3070-08
    passthroughContext         :
    password                   :
    port                       : 0
    provisionable              : True
    ssl                        : False
    storageSystemType          : CLUSTER
    storageSystemTypeSpecified : False
    storageSystemTypeAsString  :
    username                   :
    version                    :
    warnings                   :

    actuallyOpsMgr             : False
    children                   : {testvs, newtestvs}
    cloningErrors              :
    cloningWarnings            :
    errors                     : {VSC.VscApi.provisioningMessage}
    ipAddress                  : 10.00.0.22
    name                       : wilsdcluster1
```

```
    passthroughContext         :
    password                   :
    port                       : 0
    provisionable              : False
    ssl                        : False
    storageSystemType          : CLUSTER
    storageSystemTypeSpecified : False
    storageSystemTypeAsString  :
    username                   :
    version                    :
    warnings                   :
```

Get-VscProvisioningPolicyNames

The `Get-VscProvisioningPolicyNames` cmdlet returns the policy names that can be used when you provision storage.

Syntax

```
Get-VscProvisioningPolicyNames [<CommonParameters>]
```

Outputs

Displays the policy names, if any, that can be used for provisioning storage.

Example that uses parameter names

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

```
PS C:\Source\VSC_Commandlets\Tests> Get-VscProvisioningPolicyNames
    None
```

Get-VscSanDetails

The `Get-VscSanDetails` cmdlet returns details about the SAN datastore you specify.

Syntax

```
Get-VscSanDetails [-datastoreName] <String> [<CommonParameters>]
```

Parameters

`-datastoreName` <*`String`*>

Provides the name of the datastore about which you want to get information.

Attribute	Value
Required?	true
Position?	1
Default value	N/A
Accept pipeline input?	false

Attribute	Value
Accept wildcard characters?	false

Outputs

Displays the sanDetails object, which contains information about the datastore you specified.

Example that uses parameter names

```
C:\PS>Get-VscSanDetails -datastoreName AGNFSAUTOUPDATE
```

```
PS C:\Source\VSC_Commandlets\Tests> Get-VscSanDetails -datastoreName
AGNFSAUTOUPDATE

    aggrfree                          : 0
    aggrfreeSpecified                 : False
    aggrsize                          : 0
    aggrsizeSpecified                 : False
    aggrstatus                        :
    aggrusage                         : 0
    aggrusageSpecified                : False
    aggrused                          : 0
    aggrusedSpecified                 : False
    aluacapable                       :
    clusterparent                     :
    datastore                         :
    datastorecapacity                 : 0
    datastorecapacitySpecified        : False
    datastorefree                     : 0
    datastorefreeSpecified            : False
    datastoreusage                    : 0
    datastoreusageSpecified           : False
    datastoreused                     : 0
    datastoreusedSpecified            : False
    dedupeData                        :
    durablename                       :
    igroupname                        :
    igrouptype                        :
    initiators                        :
    ipaddress                         :
    luncapacity                       : 0
    luncapacitySpecified              : False
    lunfree                           : 0
    lunfreeSpecified                  : False
    lunid                             : 0
    lunidSpecified                    : False
    lunpath                           :
    luntype                           :
    lunusage                          : 0
    lunusageSpecified                 : False
    lunused                           : 0
    lunusedSpecified                  : False
    moref                             :
    multistatealua                    : False
    multistatealuaSpecified           : False
    name                              :
    partner                           :
    portsetname                       :
    protocol                          :
    serial                            :
    srenabled                         : False
    srenabledSpecified                : False
    status                            : False
    statusSpecified                   : False
    storagecapacity                   : 0
    storagecapacitySpecified          : False
```

```
    storagestatus                       : False
    storagestatusSpecified              : False
    thinprov                            : False
    thinprovSpecified                   : False
    volaggregate                        :
    volautogrowenabled                  : False
    volautogrowenabledSpecified         : False
    volautogrowinc                      : 0
    volautogrowincSpecified             : False
    volautogrowmax                      : 0
    volautogrowmaxSpecified             : False
    volcapacity                         : 0
    volcapacitySpecified                : False
    volfractionalreserve                :
    volfree                             : 0
    volfreeSpecified                    : False
    volguarantee                        :
    volname                             :
    volreserve                          :
    volsnapautodeletestate              :
    volsnapautodeletetrigger            :
    volsnapblocksreserved               : 0
    volsnapblocksreservedSpecified      : False
    volsnapreserveusedactual            : 0
    volsnapreserveusedactualSpecified : False
    volstatus                           :
    voltype                             :
    volusage                            : 0
    volusageSpecified                   : False
    volused                             : 0
    volusedSpecified                    : False
```

Get-VscStorageSystem

The `Get-VscStorageSystem` cmdlet retrieves details about storage systems form the VSC server. When you specify an IP address with the cmdlet, you can get details about a specific storage system. Otherwise, the cmdlet returns information about all the storage systems.

Syntax

```
Get-VscStorageSystem [[-detailed] [<Boolean>]] [<CommonParameters>]
```

```
Get-VscStorageSystem [-ipAddress] <String> [<CommonParameters>]
```

`-detailed` [<*`Boolean`*>]

Specifies how much information is returned:

- $true: Returns detailed information.
- $false: (Default) Returns standard information.

Attribute	Value
Required?	false
Position?	2
Default value	$false
Accept pipeline input?	false
Accept wildcard characters?	false

`--ipAddress <String>`

Specifies the storage system IP address.

Attribute	Value
Required?	true
Position?	1
Default value	NA
Accept pipeline input?	false
Accept wildcard characters?	false

Outputs

Returns information from the storageSystem[ ] object. If the `-detailed` parameter is set to $false, it returns standard information about all the storage systems. If the `-detailed` parameter is set to $true, it returns detailed storage system information.

Examples that use parameter names

Example that returns information about all storage systems.

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

```
PS C:\Source\VSC_Commandlets\PesterTests> Get-VscStorageSystem


    clusteredOntap          : False
    id                      : 1789376160-0
    ipAddress               : 10.777.10.777
    name                    : stl2240-2-1
    overallStatus           : 3
    overallStatusSpecified  : True
    overallStatusReason     : The system's global status is normal.
    storageSystemProtocol   : VSC.VscApi.storageSystemProtocol
    supportedProtocols      : 6
    totalAllocated          : 3540701843456
    totalAllocatedSpecified : True
    totalCapacity           : 4200890867712
    totalCapacitySpecified  : True
    totalFree               : 2780550631424
    totalFreeSpecified      : True
    totalUsed               : 1420340236288
    totalUsedSpecified      : True
    type                    : SEVEN_MODE_STORAGE_SYSTEM
    typeSpecified           : True
    version                 : 8.2

    clusteredOntap          : False
    id                      : 1789385161-0
    ipAddress               : 10.777.10.788
    name                    : stl2240-2-2
    overallStatus           : 3
    overallStatusSpecified  : True
    overallStatusReason     : The system's global status is normal.
    storageSystemProtocol   : VSC.VscApi.storageSystemProtocol
    supportedProtocols      : 6
    totalAllocated          : 3699240730624
    totalAllocatedSpecified : True
    totalCapacity           : 4200890867712
    totalCapacitySpecified  : True
    totalFree               : 3552740777984
    totalFreeSpecified      : True
```

```
    totalUsed               : 648150089728
    totalUsedSpecified      : True
    type                    : SEVEN_MODE_STORAGE_SYSTEM
    typeSpecified           : True
    version                 : 8.2
```

Example that returns information about a storage system identified by the IP address supplied.

```
C:\PS>Get-VscStorageSystemById -ipAddress "10.61.167.86"
```

```
PS C:\Source\VSC_Commandlets\PesterTests> Get-VscStorageSystem -
ipAddress "10.777.10.788"


    adapter                   :
    aggregate                 :
    availableAggregateNames   :
    availableInterfaceNames   :
    availableVolumeNames      :
    cfMode                    :
    cfStatus                  : ERROR
    clusterAdmin              : False
    clusterNode               : False
    clusterVserver            : False
    createVolumeWrapper       : False
    deleteVolumeIfLastLun     : False
    flagSet                   : 1
    flagSetSpecified          : True
    interfaces                : {10.777.10.788}
    ipAddressList             : {10.777.10.788}
    luns                      : {}
    model                     : FAS2240-2
    multistore                : False
    partner                   :
    password                  : 999123
    port                      : 80
    portSpecified             : True
    qtree                     : {}
    spaceReserveNewVolumes    : False
    storageSystemGroupingName :
    thinProvisionVolumeClones : False
    updating                  : False
    usableAggregateNames      :
    usableInterfaceNames      :
    usableVolumeNames         :
    useTLS                    : False
    username                  : root
    vaaiCapable               : enabled
    volumeWrapperBuffer       : 0
    volumes                   : {}
    clusteredOntap            : False
    id                        : 1789385161-0
    ipAddress                 : 10.777.10.788
    name                      : stl2240-2-2
    overallStatus             : 3
    overallStatusSpecified    : True
    overallStatusReason       : The system's global status is normal.
    storageSystemProtocol     : VSC.VscApi.storageSystemProtocol
    supportedProtocols        : 6
    totalAllocated            : 3699240730624
    totalAllocatedSpecified   : True
    totalCapacity             : 4200890867712
    totalCapacitySpecified    : True
    totalFree                 : 3552740777984
    totalFreeSpecified        : True
    totalUsed                 : 648150089728
```

```
    totalUsedSpecified        : True
    type                      : SEVEN_MODE_STORAGE_SYSTEM
    typeSpecified             : True
    version                   : 8.2
```

Get-VscStorageSystemId

You can use the `Get-VscStorageSystemId` cmdlet to return the storage system ID for the storage system you specify. This is a unique identifier and is different from the storage system IP address.

Syntax

```
Get-VscStorageSystemId [-ipAddress] <String> [<CommonParameters>]
```

Parameters

`-ipAddress<`*`String`*`>`

Provides the IP address of the storage system for which you want to get a system ID.

Attribute	Value
Required?	true
Position?	0
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

Outputs

Returns the string containing the storage system ID.

Example that uses parameter names

```
C:\PS>Get-VscStorageSystemId -ipAddress 10.777.10.777
```

```
PS C:\Source\VSC_Commandlets\Tests\mhc> Get-VscStorageSystemId -
ipAddress 10.777.10.777
    1789385161-0
```

Get-VscStorageSystemPrivileges

The `Get-VscStorageSystemPrivileges` cmdlet returns the role-based access control (RBAC) privileges associated with the user name/password credentials used to log on to the storage system.

You can get this information by specifying either the ID for storage system or the credentials required to log on to the storage system.

Syntax

```
Get-VscStorageSystemPrivileges [-storageSystemId] <String>
   [<CommonParameters>]
```

```
Get-VscStorageSystemPrivileges [-ipAddress] <String>
   [[-Credential] [<PSCredential>]] [-connectPort] <Int32>
   [-useTLS] <Boolean> [<CommonParameters>]
```

Parameters

`-storageSystemId <String>`

Provides the ID of the storage system.

Attribute	Value
Required?	true
Position?	1
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

`-ipAddress <String>`

Provides the IP address of the storage system.

Attribute	Value
Required?	true
Position?	1
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

`-Credential [<PSCredential>]`

Provides the System.Management.Automation.PSCredential object that contains the credentials for the storage system.

Attribute	Value
Required?	false
Position	4
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

-connectPort <*Int32*>

Provides the port number used to connect to the storage system.

Attribute	Value
Required?	true
Position	4
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

-useTLS [<*Boolean*>]

Specifies whether to use Transport Layer Security (TLS) when accessing the storage system:

- $true: Use a TLS connection.
- $false: Do **not** use TLS.

Attribute	Value
Required?	true
Position	5
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

Outputs

Returns the storageSystemPrivilegeOpinionBean object that contains the RBAC privileges associated with that storage system.

Examples that use parameter names

Example based on providing credentials to access the storage system.

```
C:\PS>Get-VscStorageSystemPrivileges -connectPort 80 -ipAddress
10.777.10.777
    -password 999123 -username root -useTLS $False
```

```
PS C:\Source\VSC_Commandlets\Tests\mhc> Get-VscStorageSystemPrivileges -
connectPort 80
    -ipAddress 10.777.10.777 -password 999123 -username root -useTLS
$False

    allowedRoles    : {Discovery, Create-Clones, Create-Storage, Modify-
Storage...}
    directVfiler    : False
    directVserver   : False
    disallowedRoles :
    totalBean       : VSC.VscApi.storageSystemPrivilegeInfoBean
    username        : root
```

Example based on providing storage system ID.

```
C:\PS>Get-VscStorageSystemPrivileges -storageSystemId 1789385161-0
```

```
PS C:\Source\VSC_Commandlets\Tests\mhc> Get-VscStorageSystemPrivileges
    -storageSystemId 1789385161-0

    allowedRoles    : {Discovery, Create-Clones, Create-Storage, Modify-
Storage...}
    directVfiler    : False
    directVserver   : False
    disallowedRoles :
    totalBean       : VSC.VscApi.storageSystemPrivilegeInfoBean
    username        : root
```

Get-VscTask

You can use the `Get-VscTask` cmdlet to retrieve information about the status of a task. If you do not enter a value for `-taskID`, this cmdlet returns information about all tasks. If you specify a task ID, the cmdlet only returns information about that task.

Syntax

```
Get-VscTask [<CommonParameters>]
```

```
Get-VscTask [[-taskId] <Int64>] [<CommonParameters>]
```

Parameters

`-taskId` <*`Int64`*>

Provides the ID of task.

Attribute	Value
Required?	false
Position?	1
Default value	N/A
Accept pipeline input?	true (By value)
Accept wildcard characters?	false

Outputs

Returns information about the status of the task.

Examples that use parameter names

Example that returns information about all current tasks.

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

```
PS C:\Source\VSC_Commandlets\Tests\mhc> Get-VscTask


    errorMessage                    :
    exceptionName                   :
```

```
    exceptionStack                     :
    id                                 : 1
    idSpecified                        : True
    operationName                      : UpdateAllStorageSystems
    operationNameSpecified             : True
    status                             : COMPLETE
    statusSpecified                    : True
    statusMessage                      :
    taskExpirationTime                 : 7/10/2014 4:47:02 PM
    taskExpirationTimeSpecified        : True
    taskStatusExpirationTime           : 1/1/0001 12:00:00 AM
    taskStatusExpirationTimeSpecified : False

    errorMessage                       :
    exceptionName                      :
    exceptionStack                     :
    id                                 : 2
    idSpecified                        : True
    operationName                      : UpdateAllStorageSystems
    operationNameSpecified             : True
    status                             : COMPLETE
    statusSpecified                    : True
    statusMessage                      :
    taskExpirationTime                 : 7/10/2014 4:47:04 PM
    taskExpirationTimeSpecified        : True
    taskStatusExpirationTime           : 1/1/0001 12:00:00 AM
    taskStatusExpirationTimeSpecified : False
```

Example that returns information about a specific task.

```
C:\PS>Get-VscTask -taskId 20
```

```
PS C:\Source\VSC_Commandlets\Tests\mhc> Get-VscTask -taskId 20


    errorMessage                       :
    exceptionName                      :
    exceptionStack                     :
    id                                 : 20
    idSpecified                        : True
    operationName                      : UpdateAllStorageSystems
    operationNameSpecified             : True
    status                             : COMPLETE
    statusSpecified                    : True
    statusMessage                      :
    taskExpirationTime                 : 7/12/2014 2:03:46 PM
    taskExpirationTimeSpecified        : True
    taskStatusExpirationTime           : 1/1/0001 12:00:00 AM
    taskStatusExpirationTimeSpecified : False
```

Invoke-VscSpaceReclaim

You can use the `Invoke-VscSpaceReclaim` cmdlet to start a space reclamation task on a virtual machine that is in Virtual Storage Console for VMware vSphere.

Syntax

```
Invoke-VscSpaceReclaim [-virtualMachineName] <String>
[<CommonParameters>]
```

Parameters

`-virtualMachineName <String>`

Provides the name of the virtual machine where the space reclaim operation will occur.

Attribute	Value
Required?	true
Position?	1
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

Outputs

Returns the string containing the task ID.

Example that uses parameter names

This example returns the task ID, which is 1.

```
C:\PS>Invoke-VscSpaceReclaim -virtualMachineName "AGClone1"
```

```
PS C:\Source\VSC_Commandlets\Tests> Invoke-VscSpaceReclaim -
virtualMachineName "AGClone1"

    1
```

Mount-VscDatastore

You can use the `Mount-VscDatastore` cmdlet to mount the datastore onto a host you specify.

Syntax

```
Mount-VscDatastore [-hostName] <String> [-datastoreName] <String>
[<CommonParameters>]
```

Parameters

`-hostName <String>`

Provides the name of the host on which you want to mount the datastore.

Attribute	Value
Required?	true
Position?	1
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

`-datastoreName <String>`

Provides the name of the datastore that you want to mount.

Attribute	Value
Required?	true
Position?	2
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

Outputs

Returns a boolean value of True if the operation succeeds and False if it fails.

Example that uses parameter names

```
C:\PS>Mount-VscDatastore -hostName 10.777.10.777 -datastoreName nfsCmdlet
```

```
PS C:\Source\VSC_Commandlets\Tests> Mount-VscDatastore -hostName
10.777.10.777
    -datastoreName nfsCmdlet
    True
```

New-VscClone

The `New-VscClone` cmdlet creates a clone of an existing virtual machine.

There are two parameter sets for this cmdlet. If your storage systems are running Data ONTAP operating in 7-mode, you should use the parameter set that contains the optional `vFilerContext` parameter. If your storage system is running clustered Data ONTAP, you should use the parameter set that contains the optional `svm` parameter, which specifies the name of a Storage Virtual Machine (SVM).

Note: Unlike most of the cmdlets, this cmdlet does not support positional parameters. You must enter the parameter names.

Syntax

Parameter set with `vFilerContext` parameter:

```
New-VscClone -virtualMachineName <String> [-advancedCloning]
   [-connectionBrokerType [<String>]] -parentStorageSystemId <String>
   -destStorageSystemId <String> -datastoreName <String>
   -destinationHostName <String> [-diskFormat <String>]
   -memorySizeInMb <Int32> -newCloneNamesArray <String[]> [-powerOn]
   [-upgradeHardwareVersion] -virtualProcessors <Int32>
   [-vFilerContext <String>] [-custSpecification <String>]
   [-poolName <String>] [-vmsInPool <Int32>]
   [-specifiedDestinationVmFolder <String>]
   [-viewServerConnectionBrokerType <String>] [-connectionName <String>]
   [-domain <String>] [-host <String>] [[-Credential] [<PSCredential>]]
    [<CommonParameters>]
```

Parameter set with `svm` parameter:

```
New-VscClone -virtualMachineName <String> [-advancedCloning]
   [-connectionBrokerType [<String>]] -parentStorageSystemId <String>
   -destStorageSystemId <String> -datastoreName <String>
   -destinationHostName <String> [-diskFormat <String>]
   -memorySizeInMb <Int32> -newCloneNamesArray <String[]> [-powerOn]
   [-upgradeHardwareVersion] -virtualProcessors <Int32>
   [-svm <String>] [-custSpecification <String>] [-poolName <String>]
   [-vmsInPool <Int32>] [-specifiedDestinationVmFolder <String>]
   [-viewServerConnectionBrokerType <String>] [-connectionName <String>]
   [-domain <String>] [-host <String>] [[-Credential] [<PSCredential>]]
   [<CommonParameters>]
```

`-virtualMachineName <String>`

Provides the name of the virtual machine that you want to clone.

Attribute	Value
Required?	true
Position?	named
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

`-advancedCloning`

Include this parameter if you want to use advanced cloning.

Attribute	Value
Required?	false
Position?	named
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

`-connectionBrokerType <String>`

Specifies the type of connection broker. This can be one of the following values:

- VMWARE_VIEW_4_5
- VMWARE_VIEW_4_6
- VMWARE_VIEW_5_0
- VMWARE_VIEW_5_1
- VMWARE_VIEW_5_2
- XEN_DESKTOP_4_0
- XEN_DESKTOP_5_0
- NONE

Attribute	Value
Required?	true
Position?	named
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

`-parentStorageSystemId <String>`

Provides the ID of the parent storage system where the virtual machine being cloned resides.

Attribute	Value
Required?	true
Position?	named
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

`-destStorageSystemId <String>`

Provides the ID of storage system where the cloned virtual machine will reside.

Attribute	Value
Required?	true
Position?	named
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

`-datastoreName <String>`

Provides the name of the datastore.

Attribute	Value
Required?	true
Position?	named
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

`-destinationHostName <String>`

Provides the name of the destination host.

Attribute	Value
Required?	true
Position?	named
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

`-diskFormat <String>`

Provides the format of the disk. The default format is SAME.

Attribute	Value
Required?	true
Position?	named
Default value	SAME
Accept pipeline input?	false
Accept wildcard characters?	false

`-memorySizeInMb <Int32>`

Provides the size of the memory in MB.

Attribute	Value
Required?	true
Position?	named
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

`-newCloneNamesArray <String[]>`

Provides an array that contains the names of the new clones.

Attribute	Value
Required?	true
Position?	named
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

`-powerOn`

Include this parameter if you want the system to power on the virtual machines after the cloning operation completes. If you do not include this parameter, the system does not power on the new virtual machines.

Attribute	Value
Required?	false
Position?	named
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

`-upgradeHardwareVersion`

Upgrades the hardware version of the virtual machine clone if the destination host supports a later version.

Attribute	Value
Required?	false
Position?	named
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

`-virtualProcessors <Int32>`

Specifies the number of virtual CPUs for the virtual machines.

Attribute	Value
Required?	true
Position?	named
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

`-vFilerContext <String>`

Provides the name of the vFiler unit.

Attribute	Value
Required?	false
Position?	named
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

`-svm <String>`

Provides the name of the SVM.

Attribute	Value
Required?	false
Position?	named
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

-custSpecification <*String*>

Applies a VMware specification to the new virtual machines. Refer to your VMware documentation for information about customization specifications.

Attribute	Value
Required?	false
Position?	named
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

-specifiedDestinationVmFolder<*String*>

Specifies the name of the folder where the cloned virtual machine will be placed.

Attribute	Value
Required?	false
Position?	named
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

-connectionName<*String*>

Specifies the name of the connection.

Attribute	Value
Required?	false
Position?	named
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

-domain <*String*>

Specifies the name of the domain.

Attribute	Value
Required?	false
Position?	named
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

`-host <String>`

Provides the name of the host.

Attribute	Value
Required?	false
Position?	named
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

`-Credential [<PSCredential>]`

Provides the System.Management.Automation.PSCredential object that contains the credentials needed to log on to the vCenter Server.

Attribute	Value
Required?	false
Position?	4
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

Example that uses parameter names

```
C:\PS>New-VscClone -virtualMachineName VM1-MVC2
    -parentStorageSystemId 11870a5c-984d-11e1-9c8f-123478563412
    -destStorageSystemId 11870a5c-984d-11e1-9c8f-123478563412
    -svm vs1_21a -datastoreName migrateNFS
    -destinationHostName 10.777.10.777 -diskFormat SAME -memorySizeInMb
1024
    -newCloneNamesArray @("newCloneCmdletPsCan") -virtualProcessors 1
```

```
PS C:\Source\VSC_Commandlets\Tests> New-VscClone -virtualMachineName VM1-
MVC2
    -parentStorageSystemId 11870a5c-984d-11e1-9c8f-123478563412
    -destStorageSystemId 11870a5c-984d-11e1-9c8f-123478563412
    -svm vs1_21a -datastoreName migrateNFS
    -destinationHostName 10.777.10.777 -diskFormat SAME -memorySizeInMb
```

```
1024
    -newCloneNamesArray @("newCloneCmdletPsCan") -virtualProcessors 1

    21
```

Remove-VscConnectionBroker

You can use the `Remove-VscConnectionBroker` cmdlet to remove a connection broker.

Syntax

```
Remove-VscConnectionBroker [-connectionBrokerType] <String>
   [-domain] <String> [-host] <String> [[-Credential] [<PSCredential>]]
   [<CommonParameters>]
```

Parameters

`-connectionBrokerType` *`<String>`*

Provides the type of the connection broker that you are removing.

Acceptable values are:

- VMWARE_VIEW_4_5
- VMWARE_VIEW_4_6
- VMWARE_VIEW_5_0
- VMWARE_VIEW_5_1
- VMWARE_VIEW_5_2
- XEN_DESKTOP_4_0
- XEN_DESKTOP_5_0
- NONE

Attribute	Value
Required?	true
Position?	1
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

`-domain` *`<String>`*

Provides the name of the VMware domain where the connection broker is running.

Attribute	Value
Required?	true
Position?	2
Default value	N/A

Attribute	Value
Accept pipeline input?	false
Accept wildcard characters?	false

`-host <String>`

Provides the IP address or host name of the machine running the connection broker.

Attribute	Value
Required?	true
Position?	3
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

`-Credential [<PSCredential>]`

Provides the System.Management.Automation.PSCredential object that contains the credentials needed to connect to the connection broker.

Attribute	Value
Required?	false
Position?	4
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

Outputs

Returns a boolean value to indicate whether the connection broker was successfully removed.

Remove-VscDatastore

The `Remove-VscDatastore` cmdlet removes the specified datastore from the VSC server. The datastore no longer exists after you run this cmdlet.

Syntax

```
Remove-VscDatastore [-datastoreName] <String> [-storageSystemId]
<String>
   [<CommonParameters>]
```

Parameters

`-datastoreName <String>`

Provides the name of the datastore that you want to destroy.

Attribute	Value
Required?	true
Position?	1
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

`-storageSystemId <String>`

Provides the ID of the storage system where the datastore resides.

Attribute	Value
Required?	true
Position?	2
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

Example that uses parameter names

```
C:\PS>Remove-VscDatastore -datastoreName AGrr4vmfs -storageSystemId
0151743235-0
```

```
PS C:\Source\VSC_Commandlets\Tests> Remove-VscDatastore -datastoreName
AGrr4vmfs -storageSystemId 0151743235-0
    84
```

Remove-VscStorageSystem

You can use the `Remove-VscStorageSystem` cmdlet to remove the storage system from the VSC server.

Syntax

```
Remove-VscStorageSystem [-storageSystemId] <String> [<CommonParameters>]
```

Parameters

`-storageSystemId <String>`

Provides the ID of the storage system you want to remove.

Attribute	Value
Required?	true
Position?	1
Default value	N/A

Attribute	Value
Accept pipeline input?	true (ByPropertyName)
Accept wildcard characters?	false

Outputs

Returns a boolean value to indicate whether the storage system was successfully removed.

Example that uses parameter names

```
C:\PS>Remove-VscStorageSystem -storageSystemId "1789385161-0"
```

```
PS C:\Source\VSC_Commandlets\PesterTests> Remove-VscStorageSystem -
storageSystemId "1789385161-0"
    True
```

Set-VscDatastoreSize

You can use the `Set-VscDatastoreSize` cmdlet to resize the datastore to the size you specify.

Syntax

```
Set-VscDatastoreSize [<CommonParameters>]
```

```
Set-VscDatastoreSize [-datastoreName] <String> [-storageSystemId]
<String>
   [-newSizeGB] <Double> [-forVMFS] [<CommonParameters>]
```

```
Set-VscDatastoreSize [-datastoreName] <String> [-storageSystemId]
<String>
   [-newSizeGB] <Double> [<CommonParameters>]
```

Parameters

`-datastoreName` <*`String`*>

Provides the name of the datastore that you want to resize.

Attribute	Value
Required?	true
Position?	1
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

`-storageSystemId` <*`String`*>

Provides the ID of the storage system where the datastore resides.

Attribute	Value
Required?	true
Position?	2
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

`-newSizeGB <Double>`

Provides the value of the new size in GB.

Attribute	Value
Required?	true
Position?	3
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

`-forVMFS`

This parameter is required if the datastore is a VMFS datastore.

Attribute	Value
Required?	false
Position?	4
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

Outputs

Returns a boolean value to indicate whether the datastore was successfully resized.

Example that uses parameter names

```
C:\PS>Set-VscDatastoreSize -datastoreName vmfsCmdlet -storageSystemId
27319dc1-dffa-11e1-ba77-123478563412
    -newSizeGB 10 –forVMFS
```

```
PS C:\Source\VSC_Commandlets\Tests> Set-VscDatastoreSize -datastoreName
vmfsCmdlet -storageSystemId
    27319dc1-dffa-11e1-ba77-123478563412 -newSizeGB 10 –forVMFS
    True
```

Set-VscDefaultCredentials

You can use the `Set-VscDefaultCredentials` cmdlet to set the default storage system connection details. Virtual Storage Console for VMware vSphere uses this information to log into storage systems.

Syntax

```
Set-VscDefaultCredentials [[-Credential] [<PSCredential>]]
   [-useTLS] [<Boolean>] [-port] <Int32> [<CommonParameters>]
```

Parameters

`-Credential [<PSCredential>]`

Provides the System.Management.Automation.PSCredential object that contains the credentials needed to log into storage systems.

Attribute	Value
Required?	true
Position?	4
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

`-useTLS [<Boolean>]`

Specifies whether VSC uses Transport Layer Security (TLS) to communicate with the storage system:

- $true: VSC uses TLS.
- $false: VSC does **not** use TLS.

Attribute	Value
Required?	true
Position?	3
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

`-port <Int32>`

Specifies the port number that VSC uses when connecting to the storage system.

Attribute	Value
Required?	true
Position?	4

Attribute	Value
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

Outputs

Returns a boolean value to indicate whether the default credentials for the storage system were successfully set up.

Example that uses parameter names

```
C:\PS>Set-VscDefaultCredentials -password 999123 -port 80 -username root
-useTLS $False
```

```
PS C:\Source\VSC_Commandlets\Tests\mhc>  Set-VscDefaultCredentials -
password 999123 -port 80 -username root
    -useTLS $False
    True
```

Test-VscDatastoreName

You can use the `Test-VscDatastoreName` cmdlet to determine whether the datastore you specify exists on the host. Unlike other cmdlets, this cmdlet does not accept positional parameters.

Syntax

```
Test-VscDatastoreName -hostName <String> -datastoreName <String>
[<CommonParameters>]
```

Parameters

`-hostName <String>`

Providers the name of the host that you are checking to see if the datastore exists on it. This host must be registered in the vCenter Server Inventory.

Attribute	Value
Required?	true
Position?	named
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

`-datastoreName <String>`

Provides the name of the datastore.

Attribute	Value
Required?	true
Position?	named
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

Outputs

Returns the boolean value True if the datastore exists on the specified host and false if it does not.

Example that uses parameter names

```
C:\PS>Test-VscDatastoreName -hostName vmshe-hp380-02.lab.com -
datastoreName AGcmsnnfs1
```

```
PS C:\Source\VSC_Commandlets\Tests> Test-VscDatastoreName -hostName
vmshe-hp380-02.lab.com
    -datastoreName AGcmsnnfs1
    True
```

Test-VscPolicyBasedProvisioning

You can use the `Test-VscPolicyBasedProvisioning` cmdlet to return determine whether policy-based provisioning is supported. You must have VASA Provider installed to use policy-based provisioning.

Syntax

```
Test-VscPolicyBasedProvisioning [<CommonParameters>]
```

Outputs

Returns True if policy-based provisioning is supported and false if it is not supported.

Example that uses parameter names

```
C:\PS>Test-VscPolicyBasedProvisioning
```

```
PS C:\Source\VSC_Commandlets\Tests> Test-VscPolicyBasedProvisioning
    False
```

Update-VscConnectionBroker

You can use the `Update-VscConnectionBroker` cmdlet to update the connection broker information.

Syntax

```
Update-VscConnectionBroker [-connectionBrokerType] <String>
   [-connectionName] <String> [-domain] <String> [-host] <String>
   [[-Credential] [<PSCredential>]] [<CommonParameters>]
```

Parameters

`-connectionBrokerType <String>`

Specifies the type of connection broker. Acceptable values for this parameter are:

- VMWARE_VIEW_4_5
- VMWARE_VIEW_4_6
- VMWARE_VIEW_5_0
- VMWARE_VIEW_5_1
- VMWARE_VIEW_5_2
- XEN_DESKTOP_4_0
- XEN_DESKTOP_5_0
- NONE

Attribute	Value
Required?	true
Position?	1
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

`-connectionName <String>`

Provides the name of the connection.

Attribute	Value
Required?	true
Position?	2
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

`-domain <String>`

Provides the VMware domain of the connection.

Attribute	Value
Required?	true
Position?	3
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

`-host <String>`

Provides the IP address or domain name of the connection broker.

Attribute	Value
Required?	true
Position?	4
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

`-Credential [<PSCredential>]`

Provides the System.Management.Automation.PSCredential object that contains the credentials required to access the connection broker.

Attribute	Value
Required?	false
Position?	5
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

Outputs

Returns True if the connection broker is successfully updated and false the update operation fails.

Update-VscStorageSystem

You can use the `Update-VscStorageSystem` cmdlet to update storage systems. You can specify a specific storage system or update them all. If you update all of them, you can perform the operation asychronously.

Syntax

```
Update-VscStorageSystem [[-async] [<Boolean>]] [<CommonParameters>]
```

```
Update-VscStorageSystem [-storageSystemId] <String> [-ipAddress]
<String>
   [[-Credential] [<PSCredential>]] [-useTLS] [<Boolean>] [-port] <Int32>
   [<CommonParameters>]
```

Parameters

`-async [<Boolean>]`

Specifies whether to update all the storage systems asynchronously:

- $true: The storage systems are updated asynchronously. As its output, the cmdlet returns the task ID for this operation.
- $false: The command runs and storageSystemDetails are returned for each storage system that is updated.

Attribute	Value
Required?	false
Position?	1
Default value	false
Accept pipeline input?	false
Accept wildcard characters?	false

`-storageSystemId <String>`

Provides the ID of the storage system you want to update.

Attribute	Value
Required?	true
Position?	1
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

`-ipAddress <String>`

Provides the IP address of the storage system you want to update.

Attribute	Value
Required?	true
Position?	2
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

`-Credential [<PSCredential>]`

Provides the System.Management.Automation.PSCredential object that contains the credentials required to connect to the storage system.

Attribute	Value
Required?	false
Position?	4
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

`-useTLS [<Boolean>]`

Specifies whether the VSC uses Transport Layer Security (TLS) to communicate with the storage system:

- $true: VSC uses TLS.
- $false: VSC does **not** use TLS.

Attribute	Value
Required?	true
Position?	5
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

`-port <Int32>`

Provides number of the port that VSC uses to connect to the storage system.

Attribute	Value
Required?	true
Position?	6
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

Outputs

Returns information from the storageSystemDetails object.

Examples that use parameter names

Updates all storage systems asynchronously.

```
C:\PS>Update-VscStorageSystem -async $true
```

```
PS C:\Source\VSC_Commandlets\Tests> Update-VscStorageSystem -async $true
    69
```

Updates all storage systems.

```
C:\PS>Update-VscStorageSystem
```

```
PS C:\Source\VSC_Commandlets\Tests> Update-VscStorageSystem

    errorMessage                     :
    exceptionName                    :
    exceptionStack                   :
    id                               : 68
    idSpecified                      : True
    operationName                    : UpdateAllStorageSystems
    operationNameSpecified           : True
    status                           : STARTED
    statusSpecified                  : True
    statusMessage                    :
    taskExpirationTime               : 7/14/2014 3:52:44 PM
    taskExpirationTimeSpecified      : True
    taskStatusExpirationTime         : 1/1/0001 12:00:00 AM
    taskStatusExpirationTimeSpecified : False
```

Updates a storage system based on its ID and IP address.

```
C:\PS>Update-VscStorageSystem -storageSystemId 1789385161-0 -ipAddress
10.777.10.777
    -username root -password 999123! -useTLS $false -port 80
```

Wait-VscTime

You can use the `Wait-VscTime` cmdlet to specify the amount of time in seconds that the VSC server will wait. This cmdlet returns a long taskId value that you can pass into `Get-VscTask`.

Syntax

```
Wait-VscTime [-seconds] <Int32> [<CommonParameters>]
```

Parameters

`-seconds <Int32>`

Specifies the number of seconds to wait.

Attribute	Value
Required?	true
Position?	1
Default value	N/A
Accept pipeline input?	false
Accept wildcard characters?	false

Outputs

Returns the ID of the task.

Example that uses parameter names

```
C:\PS>Wait-VscTime -seconds 5
```

```
PS C:\Source\VSC_Commandlets\Tests\mhc> Wait-VscTime -seconds 5
    32
```

Copyright information

Copyright © 1994–2015 NetApp, Inc. All rights reserved. Printed in the U.S.

No part of this document covered by copyright may be reproduced in any form or by any means—graphic, electronic, or mechanical, including photocopying, recording, taping, or storage in an electronic retrieval system—without prior written permission of the copyright owner.

Software derived from copyrighted NetApp material is subject to the following license and disclaimer:

THIS SOFTWARE IS PROVIDED BY NETAPP "AS IS" AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, WHICH ARE HEREBY DISCLAIMED. IN NO EVENT SHALL NETAPP BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

NetApp reserves the right to change any products described herein at any time, and without notice. NetApp assumes no responsibility or liability arising from the use of products described herein, except as expressly agreed to in writing by NetApp. The use or purchase of this product does not convey a license under any patent rights, trademark rights, or any other intellectual property rights of NetApp.

The product described in this manual may be protected by one or more U.S. patents, foreign patents, or pending applications.

RESTRICTED RIGHTS LEGEND: Use, duplication, or disclosure by the government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.277-7103 (October 1988) and FAR 52-227-19 (June 1987).

Trademark information

NetApp, the NetApp logo, Go Further, Faster, AltaVault, ASUP, AutoSupport, Campaign Express, Cloud ONTAP, Clustered Data ONTAP, Customer Fitness, Data ONTAP, DataMotion, Fitness, Flash Accel, Flash Cache, Flash Pool, FlashRay, FlexArray, FlexCache, FlexClone, FlexPod, FlexScale, FlexShare, FlexVol, FPolicy, GetSuccessful, LockVault, Manage ONTAP, Mars, MetroCluster, MultiStore, NetApp Insight, OnCommand, ONTAP, ONTAPI, RAID DP, RAID-TEC, SANtricity, SecureShare, Simplicity, Simulate ONTAP, Snap Creator, SnapCenter, SnapCopy, SnapDrive, SnapIntegrator, SnapLock, SnapManager, SnapMirror, SnapMover, SnapProtect, SnapRestore, Snapshot, SnapValidator, SnapVault, StorageGRID, Tech OnTap, Unbound Cloud, and WAFL and other names are trademarks or registered trademarks of NetApp, Inc., in the United States, and/or other countries. All other brands or products are trademarks or registered trademarks of their respective holders and should be treated as such. A current list of NetApp trademarks is available on the web at *http://www.netapp.com/us/legal/netapptmlist.aspx*.

How to send comments about documentation and receive update notifications

You can help us to improve the quality of our documentation by sending us your feedback. You can receive automatic notification when production-level (GA/FCS) documentation is initially released or important changes are made to existing production-level documents.

If you have suggestions for improving this document, send us your comments by email to *doccomments@netapp.com*. To help us direct your comments to the correct division, include in the subject line the product name, version, and operating system.

If you want to be notified automatically when production-level documentation is released or important changes are made to existing production-level documents, follow Twitter account @NetAppDoc.

You can also contact us in the following ways:

- NetApp, Inc., 495 East Java Drive, Sunnyvale, CA 94089 U.S.
- Telephone: +1 (408) 822-6000
- Fax: +1 (408) 822-4501
- Support telephone: +1 (888) 463-8277

Index

O

P

R

S

T

U

V

W