NetApp Cloud Volumes ONTAP 9

SVM Disaster Recovery Preparation Express Guide

June 2018 | 215-12839_B0
doccomments@netapp.com

NetApp®

Contents

Deciding whether to use the SVM Disaster Recovery Preparation Express Guide 3
Configurations replicated in an SVM disaster recovery relationship 4
SVM disaster recovery preparation workflow 7
Preparing the destination cluster 8
Creating cluster peering 9
Creating a destination SVM 11
Creating the SVM peer relationship 11
Different subnet: Creating a SnapMirror policy 12
Creating a SnapMirror relationship 12
Excluding volumes from replication 14
CIFS only: Creating a CIFS server 14
Initializing the destination SVM 15
Different subnet: Configuring NAS LIFs 16
Configuring the network and protocols for data access on the destination SVM 17
Configuring the network and NAS protocols 18
Configuring the network and SAN protocols 18
Monitoring the SnapMirror relationship status 19
Configuration example for Cloud Volumes ONTAP in AWS 20
Where to find additional information 22
Copyright information 23
Trademark information 24
How to send comments about documentation and receive update notifications 25

Deciding whether to use the SVM Disaster Recovery Preparation Express Guide

This guide describes how cluster administrators can quickly prepare a data-serving Storage Virtual Machine (SVM) for disaster recovery. You can create and configure a destination SVM in the destination cluster, and then replicate data and configuration from the source SVM to the destination SVM. SVM disaster recovery is the asynchronous mirroring of SVM data and configuration.

Note: OnCommand Cloud Manager does not support SVM disaster recovery. Cloud Manager functionality is limited in the destination system.

You should use this guide if you want to create and configure a destination SVM for disaster recovery in the following situations:

- You are a cluster administrator.
- The source SVM does not contain data protection (DP) volumes and transition (TDP) volumes.
- You are using the ONTAP command-line interface.
- You want to use best practices, not explore every available option.
- You do not want to read a lot of conceptual background.

If these assumptions are not correct for your situation, you should use the following:

- Volume-level disaster recovery by using SnapMirror technology

  You can set up volume-level disaster recovery directly from OnCommand Cloud Manager.

- Volume-level backup by using SnapVault technology

  You can also set up volume-level backups directly from OnCommand Cloud Manager.

- *NetApp Documentation: OnCommand Workflow Automation (current releases)*

  OnCommand Workflow Automation enables you to run prepackaged workflows that automate management tasks such as the workflows described in Express Guides.

Configurations replicated in an SVM disaster recovery relationship

When you set up the SVM disaster recovery relationship, the value that you select for the `identity-preserve` option of the `snapmirror create` command determines the configurations that are replicated in the destination SVM.

If you set the `-identity-preserve` option to **`true`**, all the configuration details except the SAN configuration are replicated. If the source cluster and destination cluster are in different network subnets, you can choose not to replicate the NAS LIFs on the destination SVM.

If you set the `-identity-preserve` option to **`false`**, only a subset of the configuration details— those that are not associated with the network configuration—is replicated.

The following table lists the configuration details that are replicated when the `-identitypreserve` option is set to **`true`** and when this option is set to **`false`**.

Configuration		**Replicated if the `-identity-preserve` option is set to `true` and if a SnapMirror policy with the `-discardconfigs network` option is used**	**Replicated if the `-identity-preserve` option is set to false**
CIFS	CIFS server	Yes	No
CIFS policy	Local groups and local user	Yes	Yes
	Privilege	Yes	Yes
	Shadow copy	Yes	Yes
	BranchCache	Yes	Yes
	Server options	Yes	Yes
	Server security	Yes	No
	Home directory, share	Yes	Yes
	Symlink	Yes	Yes
	Fpolicy policy, Fsecurity policy, and Fsecurity NTFS	Yes	Yes
	Name mapping and group mapping	Yes	Yes
	Audit information	Yes	Yes

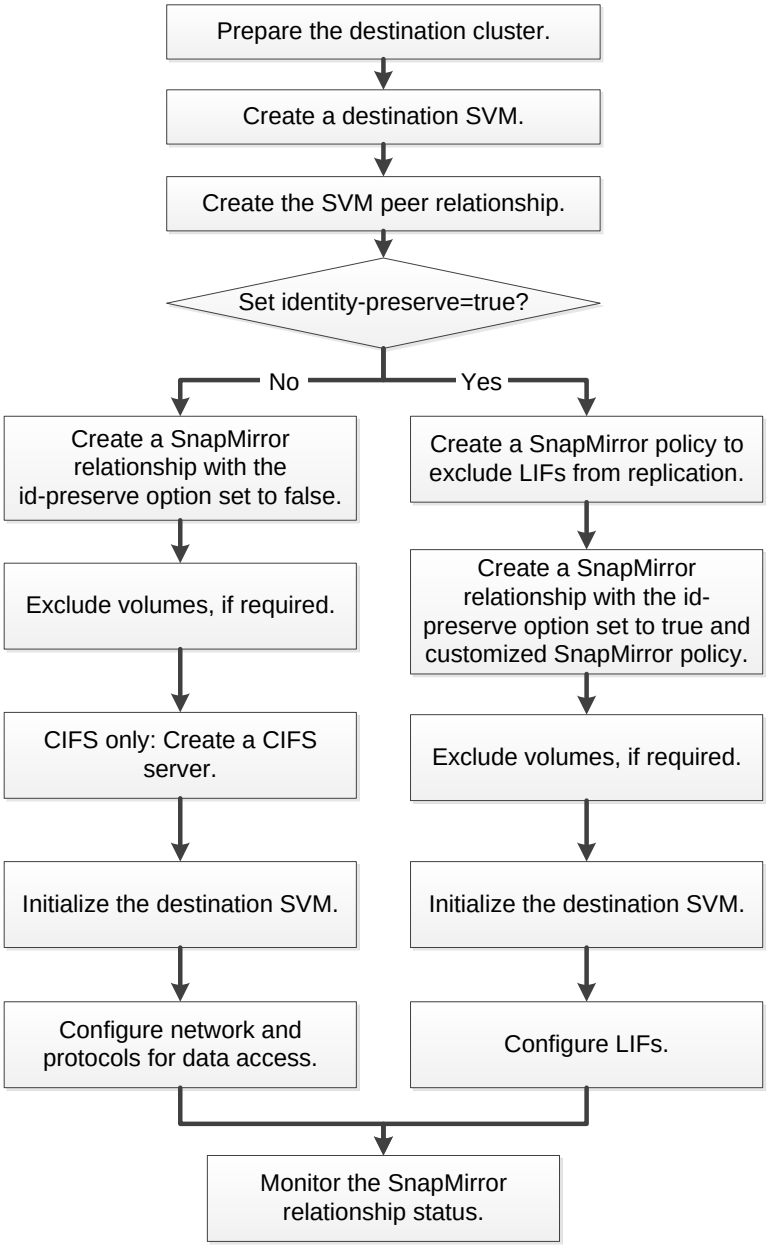
Configuration		Replicated if the `-identity-preserve` option is set to `true` and if a SnapMirror policy with the `-discardconfigs network` option is used	Replicated if the `-identity-preserve` option is set to `False`
NFS	Export policies	Yes	No
	Export policy rules	Yes	No
	NFS server	Yes	No
Network	NAS LIFs	No	No
	LIF Kerberos configuration	No	No
	SAN LIFs	No	No
	Firewall policies	Yes	No
	Routes	No	No
	Broadcast domain	No	No
	Subnet	No	No
	IPspace	No	No
RBAC	Security certificates	Yes	No
	Login user, public key, role, and role configuration	Yes	Yes
	SSL	Yes	No
Name services	DNS and DNS hosts	Yes	No
	UNIX user and UNIX group	Yes	Yes
	Kerberos realm and Kerberos keyblocks	Yes	No
	LDAP and LDAP client	Yes	No
	Netgroup	Yes	No
	NIS	Yes	No
	Web and web access	Yes	No

Configuration		Replicated if the `-identity-preserve` option is set to `true` and if a SnapMirror policy with the `-discardconfigs network` option is used	Replicated if the `-identity-preseve` option is set to `false`
Volume	Object	Yes	Yes
	Snapshot copies, Snapshot policy, and autodelete policy	Yes	Yes
	Efficiency policy	Yes	Yes
	Quota policy and quota policy rule	Yes	Yes
	Recovery queue	Yes	Yes
Root volume	Namespace	Yes	Yes
	User data	No	No
	Qtrees	No	No
	Quotas	No	No
	File-level QoS	No	No
	Attributes: state of the root volume, space guarantee, size, autosize, and total number of files	No	No
Storage QoS	QoS policy group	Yes	Yes
Fibre Channel (FC)		No	No
iSCSI		No	No
LUNs	Object	Yes	Yes
	igroups	No	No
	portsets	No	No
SNMP	v3 users	Yes	No

Note: Cluster-level objects such as aggregates are not replicated.

SVM disaster recovery preparation workflow

Preparing the SVM for disaster recovery involves preparing the destination cluster, creating the destination SVM, creating the SVM peer relationship, creating a SnapMirror relationship, initializing the destination SVM, configuring the destination SVM for data access, and monitoring the SnapMirror relationship status.



In the cloud, there is no option to assign the IP address from the ONTAP side. You must first request the cloud provider to assign you an IP address. Therefore, selecting the same subnet does not work in most cases.

Note: OnCommand Cloud Manager does not support SVM disaster recovery. Cloud Manager functionality is limited in the destination system.

Preparing the destination cluster

Before you create and configure the destination SVM, you must verify that the cluster peer relationship between the source and destination clusters is healthy and prepare the destination cluster with required licenses, custom schedules, and sufficient free space.

Steps

1. Verify that the source and destination clusters are peered and the peer cluster is available by using the `cluster peer show` command.

   Example

```
destination_cluster::> cluster peer show
Peer Cluster Name   Cluster Serial Number Availability   Authentication
------------------- --------------------- -------------- --------------
source_cluster        1-80-000011          Available      absent
```

2. Install the licenses of the features and protocols used by the source SVM on the destination cluster:

   a. Identify the licensed features and protocols on the source cluster by using the `license show` command.

      Example

```
source_cluster::> license show
  (system license show)
Serial Number: 1-80-000011
Owner: source_cluster
Package           Type    Description           Expiration
----------------- ------- --------------------- ----------
Base              site    Cluster Base License   -
NFS               site    NFS License            -
CIFS              site    CIFS License           -
iSCSI             site    iSCSI License          -
FCP               site    FCP License            -
SnapMirror        site    SnapMirror License     -
FlexClone         site    FlexClone License      -
7 entries were displayed.
```

   b. Verify that the licenses of the required features and protocols are installed on the destination cluster by using the `system license show` command.

      Example

```
destination_cluster::> system license show
Serial Number: 1-80-000011
Owner: source_cluster
Package        Type    Description           Expiration
-------------- ------- --------------------- -----------
Base           site    Cluster Base License    -
NFS            site    NFS License             -
CIFS           site    CIFS License            -
SnapMirror     site    SnapMirror License      -
FlexClone      site    FlexClone License       -
5 entries were displayed.
```

   c. If any required feature or protocol is not licensed on the destination cluster, then add the license by using the `system license add` command.

      Example

      The following example adds an FCP license on the destination cluster:

```
destination_cluster::> system license add -license-code
xxxxxxxxxxxxxxxxxxxxxxxxxxxx
License for package "FCP" installed successfully.
```

3. On the destination cluster, create the same custom schedules as on the source cluster:

   a. Identify the custom schedules on the source cluster by using the `job schedule cron show` command.

   Example

```
source_cluster::> job schedule cron show

Name                Description
----------------    ----------------------------------------------------
5min                @:00,:05,:10,:15,:20,:25,:30,:35,:40,:45,:50,:55
8hour               @2:15,10:15,18:15
daily               @0:10
hourly              @:05
weekly              Sun@0:15
5 entries were displayed.
```

   b. On the destination cluster, create the same custom schedules by using the `job schedule cron create` command.

   You must specify the exact job names and schedule as that on the source cluster because job schedules are case-sensitive.

   Example

```
destination_cluster::> job schedule cron create -name weekly -dayofweek
"Sunday" -hour 0 -minute 15
```

4. Ensure that the destination cluster has at least one non-root aggregate with a minimum free space of 10 GB for configuration replication.

   The best practice is to have at least two non-root aggregates with a minimum free space of 10 GB.

   a. Verify that the non-root aggregate has a minimum free space of 10 GB by using the `storage aggregate show` command.

   Example

```
destination_cluster::> storage aggregate show

Aggregate     Size Available Used% State   #Vols  Nodes                  RAID Status
--------- -------- --------- ----- ------- ------ ----------------       -----------
aggr0       6.04GB    1.15GB   81% online    2    destination_cluster-01     raid_dp,
                                                                           normal
aggr1       5.14GB    2.47GB   52% online    15   destination_cluster-02     raid_dp,
                                                                           normal
```

   b. If there is no non-root aggregate with a minimum free space of 10 GB, create a non-root aggregate 10 GB in size by using the `storage aggregate create` command.

   Example

```
destination_cluster::> storage aggregate create -aggregate aggr3 -nodes
destination_cluster_01 -diskcount 20 -disksize 10
```

Creating cluster peering

You can create a cluster peer relationship using a set of intercluster designated logical interfaces to make information about one cluster available to the other cluster for use in cluster peering applications.

Before you begin

Configure the intercluster network.

Steps

1. Create the cluster peer relationship from the local cluster by using the `cluster peer create` command, and then notify the administrator of the remote cluster of your peer request.

   Unilaterally creating a cluster peer relationship requires the login credentials of the remote cluster administrator. Alternatively, you can create a cluster peer relationship without exchanging credentials by having both cluster administrators issue the `cluster peer create` command from their respective clusters.

   When the remote cluster administrator issues the reciprocal cluster peer request, ONTAP creates the peer relationship.

   #### Example

   In the following example, cluster01 is peered with a remote cluster named cluster02. Cluster02 is a two-node cluster that has one intercluster LIF per node. The IP addresses of the intercluster LIFs created in cluster02 are 192.168.2.203 and 192.168.2.204. These IP addresses are used to create the cluster peer relationship.

   ```
   cluster01::> cluster peer create -peer-addrs 192.168.2.203,192.168.2.204
   ```

2. Display the cluster peer relationship by using the cluster peer show command with the `-instance` parameter.

   #### Example

   ```
   cluster01::> cluster peer show –instance
   Peer Cluster Name: cluster02
   Remote Intercluster Addresses: 192.168.2.203,192.168.2.204
   Availability: Available
   Remote Cluster Name: cluster02
   Active IP Addresses: 192.168.2.203,192.168.2.204
   Cluster Serial Number: 1-80-000013
   ```

3. Preview the health of the cluster peer relationship by using the `cluster peer health show` command.

   ```
   cluster01::> cluster peer health show
   Node       cluster-Name                Node-Name
                Ping-Status               RDB-Health Cluster-Health  Avail…
   ---------- --------------------------- ---------  --------------- --------
   cluster01-01
              cluster02                   cluster02-01
                Data: interface_reachable
                ICMP: interface_reachable true       true            true
                                          cluster02-02
                Data: interface_reachable
                ICMP: interface_reachable true       true            true

   cluster01-02
              cluster02                   cluster02-01
                Data: interface_reachable
                ICMP: interface_reachable true       true            true
                                          cluster02-02
                Data: interface_reachable
                ICMP: interface_reachable true       true            true
   ```

Creating a destination SVM

For protecting the data and configuration information on the source SVM, you must create a destination SVM on the destination cluster.

Before you begin

- The cluster must have at least one non-root aggregate with sufficient space.
- If you want to assign IPspace, you must have created the IPspace.

About this task

A destination SVM can be used for only one source SVM, and the destination SVM cannot be the source of any other SVM disaster recovery relationship.

Steps

1. Create the destination SVM by using the `vserver create` command with the subtype **`dpdestination`**.

   You must use the fully qualified domain name (FQDN) of the SVM or another convention that ensures unique SVM names across clusters.

   If you do not assign an IPspace, the SVM is created in the default IPspace.

   #### Example

   The following command creates a destination SVM in the default IPspace:

   ```
   destination_cluster::> vserver create -vserver dvs1 -subtype dp-destination
   [Job 383] Job succeeded:
   Vserver creation completed
   ```

2. Verify the status of the newly created SVM by using the `vserver show` command.

   #### Example

   ```
   destination_cluster::> vserver show
                                        Admin        Operational  Root
   Vserver       Type    Subtype         State        State        Volume     Aggregate
   -----------   ------- ----------     -----------   -----------  ---------- ----------
   dvs1          data    dp-destination   running     stopped       -          -
   ```

Result

The destination SVM is created without a root volume and is in the `stopped` state.

Creating the SVM peer relationship

You must create an intercluster SVM peer relationship between the source and the destination SVMs to provide an infrastructure for SVM disaster recovery.

About this task

The cluster administrator of the peered cluster must authenticate the SVM peer relationship.

Steps

1. On the destination cluster, create the SVM peer relationship by using the `vserver peer create` command.

Example

```
destination_cluster::> vserver peer create -vserver dvs1 -peer-vserver vs1 -
applications snapmirror -peer-cluster source_cluster
```

The SVM peer relationship is in the `initiated` state.

2. On the source cluster, accept the SVM peer relationship by using the `vserver peer accept` command.

Example

```
source_cluster::> vserver peer accept -vserver vs1 -peer-vserver dvs1 Info: [Job
371] 'vserver peer accept' job queued
```

3. Verify that the SVM peer relationship is in the `peered` state.

Example

```
destination_cluster::> vserver peer show
            Peer        Peer         Peering
Vserver     Vserver     State        Applications
----------- ----------- ------------ -----------------
dvs1         vs1        peered       snapmirror
```

Different subnet: Creating a SnapMirror policy

If the source and destination SVMs are in different network subnets and you do not want to replicate the LIFs, you must create a SnapMirror policy with the `-discard-configs network` option.

About this task

You must perform this task on the destination cluster.

Steps

1. Create a SnapMirror policy to exclude the LIFs from replication by using the `snapmirror policy create` command.

Example

```
destination_cluster::> snapmirror policy create -vserver dvs1 -policy
exclude_LIF -type async-mirror -discard-configs network
```

2. Verify that the new SnapMirror policy is created by using the `snapmirror policy show` command.

After you finish

You must use the newly created SnapMirror policy when creating the SnapMirror relationship.

Creating a SnapMirror relationship

You must create a SnapMirror relationship between the source SVM and the destination SVM for disaster recovery. You can choose to replicate data and all or a subset of the SVM configuration information when creating the SnapMirror relationship.

Before you begin

- The CIFS audit consolidation path must be on a non-root volume.

- The SVM root volume must not have any qtrees.

About this task

You must perform this task from the destination cluster.

Steps

1. Create a SnapMirror relationship between the source SVM and the destination SVM by using the `snapmirror create` command.

   You can specify the SnapMirror policy and schedule when creating the SnapMirror relationship. The SnapMirror schedule is applicable to all the volumes and configuration of the source SVM.

   You can specify the source SVM and the destination SVM as either paths or SVM names. If you want to specify the source and destination as paths, then the SVM name must be followed by a colon.

   - Replicate the data and all the configuration information by setting the `-identity-preserve` option to **`true`**.

     The following command creates the SVM disaster recovery relationship with the SVM names as the `-destination-path` and `-source-path` parameters and uses the SnapMirror policy **`exclude_LIF`** to exclude LIFs from replication:

     ```
     destination_cluster::> snapmirror create -source-path vs1: -destination-path
     dvs1: type DP -throttle unlimited -policy exclude_LIF -schedule hourly -
     identity-preserve true
     ```

   - Replicate the data and a subset of the configuration information by setting the `-identitypreserve` option to **`false`**.

     The following command creates the SVM disaster recovery relationship with the SVM names as the `-destination-path` and `-source-path` parameters:

     ```
     destination_cluster::> snapmirror create -source-path vs2: -destination-path
     dvs2: -type DP -throttle unlimited -policy DPDefault -schedule hourly -
     identity-preserve false
     ```

     The following command creates the SVM disaster recovery relationship with the SVM names as the `-destination-vserver` and `-source-vserver` parameters:

     ```
     destination_cluster::> snapmirror create -source-vserver vs2 -destination-
     vserver dvs2
     -type DP -throttle unlimited -policy DPDefault -schedule hourly -identity-
     preserve false
     ```

2. Verify that the SnapMirror relationship is established and is in the **`Uninitialized`** state by using the `snapmirror show` command.

   To view the detailed status of the relationship, you can use the `-instance` option.

   #### Example

   ```
   destination_cluster::> snapmirror show
                                                                       Progress
   Source        Destination  Mirror        Relationship   Total             Last
   Path    Type  Path         State         Status         Progress   Healthy Updated
   ------- ---- ------------ -------        -------------- --------- -------  --------
   vs1:     DP   dvs1:       Uninitialized   Idle           -         true     -

   destination_cluster::> snapmirror show -instance

                         Source Path:  vs1:
                         Destination Path: dvs1:
                        Relationship Type: DP
   ```

```
           Relationship Group Type: vserver
                 SnapMirror Schedule: -
              SnapMirror Policy Type: async-mirror
                     SnapMirror Policy: DPDefault
                  ......
                  .....

                   Total Transfer Bytes: -
 Total Transfer Time in Seconds: -
```

Excluding volumes from replication

By default, all RW data volumes of the source SVM are replicated. If you do not want to protect all the volumes that are on the source SVM, you can select the volumes that must be excluded from the replication by modifying the `-vserver-dr-protection` option of those volumes.

About this task

You must perform this task from the source cluster and for each of the volumes that must be excluded.

All the unprotected volumes as well as their namespace child volumes and clone child volumes are excluded from replication.

Steps

1. Exclude a volume from the replication by using the `volume modify` command.

   #### Example

   ```
   source_cluster::> volume modify -vserver vs1 -volume test_vol1 vserver-dr-
   protection unprotected
   ```

2. Verify that the status of the modified volumes is `unprotected` by using the `volume show fields vserver-dr-protection` command.

   #### Example

   ```
   source_cluster::> volume show -fields vserver-dr-protection
   Vserver   Volume    Vserver DR Protection
   -------   ------    --------------------
   vs1       root_vol    unprotected
   vs1       test_vol1   unprotected
   vs1       vol2             protected
   ```

CIFS only: Creating a CIFS server

If the source SVM has CIFS configuration, and you chose to set `identity-preserve` to **`false`**, you must create a CIFS server for the destination SVM. A CIFS server is required for some CIFS configurations, such as shares during initialization of the SnapMirror relationship.

Steps

1. Start the destination SVM by using the `vserver start` command.

   #### Example

   ```
   destination_cluster::> vserver start -vserver dvs1 [Job 30] Job succeeded: DONE
   ```

2. Verify that the destination SVM is in the `running` state and the subtype is `dp-destination` by using the `vserver show` command.

Example

```
destination_cluster::> vserver show
                                    Admin      Operational Root
Vserver  Type    Subtype            State      State       Volume     Aggregate
-------- ------- ----------        ---------- ----------- ---------- ----------
dvs1     data    dp-destination    running    running       -          -
```

3. Create a LIF by using the `network interface create` command.

Example

```
destination_cluster::>network interface create -vserver dvs1 -lif NAS1 -role
data -dataprotocol cifs -home-node destination_cluster-01 -home-port a0a-101  -
address 192.0.2.128 netmask 255.255.255.128
```

4. Create a route by using the `network route create` command.

Example

```
destination_cluster::>network route create -vserver dvs1 -destination 0.0.0.0/0
-gateway 192.0.2.1
```

Network and LIF management

5. Configure DNS by using the `vserver services dns create` command.

Example

```
destination_cluster::>vserver services dns create -domains mydomain.example.com
-vserver
dvs1 -name-servers 192.0.2.128 -state enabled
```

6. Add the preferred domain controller by using the `vserver cifs domain preferred-dc add` command.

Example

```
destination_cluster::>vserver cifs domain preferred-dc add -vserver dvs1 -
preferred-dc 192.0.2.128 -domain mydomain.example.com
```

7. Create the CIFS server by using the `vserver cifs create` command.

Example

```
destination_cluster::>vserver cifs create -vserver dvs1 -domain
mydomain.example.com -cifs-server CIFS1
```

CIFS management

8. Stop the destination SVM by using the `vserver stop` command.

Example

```
destination_cluster::> vserver stop -vserver dvs1 [Job 46] Job succeeded: DONE
```

Initializing the destination SVM

You must initialize the destination SVM for the baseline transfer of data and configuration details from the source SVM.

Before you begin

- The source SVM root volume must not contain any other data apart from metadata because the other data is not replicated.

  Root volume metadata such as volume junctions, symbolic links, and directories leading to the junction's symbolic links are replicated.

- The destination SVM must be in the `stopped` state.

About this task

You cannot use tape seeding to initialize the SnapMirror relationship between the source and destination SVMs.

Steps

1. Use the `snapmirror initialize` command to perform a baseline transfer from the source to the destination SVM.

   Example

   ```
   destination_cluster::> snapmirror initialize dvs1:
   ```

2. Use the `snapmirror show` command to verify that the status of the SnapMirror relationship is in the `Snapmirrored` state.

   For viewing the detailed status of the relationship, you can use the `-instance` option.

   Example

   ```
   destination_cluster::> snapmirror show
                                                                       Progress
   Source            Destination  Mirror  Relationship  Total             Last
   Path        Type  Path         State   Status        Progress  Healthy Updated
   ----------- ---- ------------ ------- -------------- --------- ------- --------
   vs1:         DP   dvs1:        Snapmirrored
                                       Idle           -         true    -

   destination_cluster::> snapmirror show -instance

                          Source Path:  vs1:
                          Destination Path: dvs1:
                         Relationship Type: DP
               Relationship Group Type: vserver
                      SnapMirror Schedule: -
                  SnapMirror Policy Type: async-mirror
                          SnapMirror Policy: DPDefault
                      ......
                      .....

                       Total Transfer Bytes: -
     Total Transfer Time in Seconds: -
   ```

   You must not associate the destination SVM with any other source SVM, because only one destination SVM can be used to protect one source SVM.

Different subnet: Configuring NAS LIFs

If the source and destination SVMs are in different network subnets, you must configure NAS LIFs on the destination SVM for data access when a disaster occurs.

About this task

This procedure provides the high-level steps that are required to configure LIFs on the destination SVM. Detailed information about configuring LIFs is available in the Network Management Guide.

Network and LIF management

AWS Multiple IP Addresses

Steps

1. Assign a new IP address to the Cloud Volumes ONTAP EC2 Instance from the AWS Console or CLI.
2. Create data LIFs by using the `network interface create` command and the IP address we got from the cloud Provider.
3. Create routes for the data LIFs by using the `network route create` command.

To assign a secondary private IPv4 address to a network interface in AWS, complete the following steps:

1. Open the Amazon EC2 console at `https://console.aws.amazon.com/ec2/`.
2. In the navigation pane, select Network Interfaces, and then select the network interface attached to the instance.
3. Select Actions, Manage IP Addresses.
4. Under IPv4 Addresses, select Assign new IP, and then click Yes, Update.
5. Copy the new assigned IP address (You will use it when later when creating the LIF).

To assign a secondary private IPv4 to an existing instance using the command line, use one of the following commands. For more information about these command line interfaces, see Accessing Amazon EC2.

- *assign-private-ip-addresses (AWS CLI)*
- *Register-EC2PrivateIpAddress (AWS Tools for Windows PowerShell)*

Configuring the network and protocols for data access on the destination SVM

If you chose to set the `identity-preserve` option to **`false`** or if the source SVM has SAN configuration, you must configure the network and protocols on the destination SVM for data access when a disaster occurs.

Before you begin

The destination SVM must be started and in the **`running`** state.

About this task

You must configure the SAN network and protocols if the source SVM is configured for SAN protocols because SAN network configuration is not replicated.

Choices

- Configuring the network and NAS protocols on page 18
- Configuring the network and SAN protocols on page 18

Configuring the network and NAS protocols

If the source SVM has NAS configuration, you must configure the network and NAS protocols on the destination SVM for data access in the event of a disaster.

About this task

This procedure provides the high-level steps that are required to complete the network and NAS protocols configuration on the destination SVM. Detailed information about these steps are available in other ONTAP documentation.

- *Network and LIF management*
- *NFS management*
- *CIFS management*

Steps

1. Start the destination SVM by using the `vserver start` command.
2. Create data LIFs by using the `network interface create` command.
3. Create routes for the data LIFs by using the `network route create` command.
4. Configure name services such as LDAP, NIS, and DNS by using the `vserver services` command.
5. Configure CIFS, NFS, or both the protocols by using the `vserver cifs create` and `vserver nfs create` commands.
6. If the source SVM has CIFS configuration, stop the destination SVM by using the `vserver stop` command.

After you finish

You can set up read-only access for NFS clients from the destination SVM.

Configuring the network and SAN protocols

If the source SVM has SAN configuration, you must configure the network and SAN protocols on the destination SVM for data access in the event of a disaster.

About this task

This procedure provides the high-level steps that are required to complete the network and SAN protocols configuration on the destination SVM. Detailed information about these steps are available in other clustered Data ONTAP documentation.

- *Network and LIF management*
- *SAN administration*

Steps

1. Start the destination SVM by using the `vserver start` command.
2. Create data LIFs by using the `network interface create` command.
3. Create igroups for the LUNs by using the `lun igroup create` command.
4. Map the LUNs to the igroups by using the `lun mapping create` command.
5. Configure iSCSI, FC, or both the protocols by using the `vserver iscsi create` and `vserver fcp create` commands.

After you finish

You can set up read-only access for SAN hosts from the destination SVM.

Monitoring the SnapMirror relationship status

You can monitor the status of the SnapMirror relationship between the source and the destination SVMs to verify that the updates are occurring per the schedule.

About this task

SNMP is not supported for monitoring the SnapMirror relationships between the source and destination SVMs.

Step

1. Use the `snapmirror show -instance` command to view the details of the SnapMirror relationship status.

Example

```
destination_cluster::> snapmirror show -instance

                            Source Path: vs1:
                         Destination Path: dvs1:
                        Relationship Type: DP
                  Relationship Group Type: vserver
                      SnapMirror Schedule: -
                   SnapMirror Policy Type: async-mirror
                        SnapMirror Policy: DPDefault
                                Mirror State: Snapmirrored
                       Relationship Status: Idle
                       ..
                       ..
                     Snapshot Checkpoint: -
                          Newest Snapshot: vserverdr.4eb1f1aa-
e4ba-11e3-9b97-005056af93d7.2014-05-26_095857
                Newest Snapshot Timestamp: 05/26 09:58:57
                          Exported Snapshot: vserverdr.4eb1f1aa-
e4ba-11e3-9b97-005056af93d7.2014-05-26_095857
              Exported Snapshot Timestamp: 05/26 09:58:57
                                Healthy: true
                                Unhealthy Reason:
                        ...
                        ...
                      Last Transfer Type: update
                      Last Transfer Error: -
                      Last Transfer From: vs1:
               Last Transfer End Timestamp: 05/26 10:05:24
                        ...
                        ...
             Lag Time: 2:0:15
             Identity Preserve Vserver DR: true
```

- If any clone parent or clone child volumes are moved by using the `vol move` command, then you must move the corresponding volume at the destination SVM.

Configuration example for Cloud Volumes ONTAP in AWS

These commands work for the specific use case—configuring Cloud Volumes ONTAP for SVM disaster recovery for another Cloud Volumes ONTAP system in the same VPC.

Cluster Peering

```
source_cluster::>       set -confirmation off
source_cluster::>       net interface  show -role intercluster
destination_cluster::>  set -confirmation off
destination_cluster::>  cluster peer create -peer-addrs [sourceInterclusterIPs]
destination_cluster::>  net interface  show -role intercluster
source_cluster::>       cluster peer create -peer-addrs [destinationInterclusterIPs]
source_cluster::>       cluster peer show
```

vServer Destination Creation and Peering

```
source_cluster::>       vserver show
destination_cluster::>  cluster peer show
destination_cluster::>  vserver show
destination_cluster::>  net int show
destination_cluster::>  vserver create -vserver [destinationVSERVER] -subtype dp-
destination
destination_cluster::>  vserver peer create -vserver [destinationVSERVER] -peer-
vserver [sourceVSERVER] -applications snapmirror -peer-cluster [sourceCLUSTER]
source_cluster::>       vserver peer accept -vserver [sourceVSERVER] -peer-vserver
[destinationVSERVER]
```

SVM-DR Creation

```
destination_cluster::> snapmirror policy create -vserver [destinationVSERVER] -
policy exclude_LIF -type async-mirror -discard-configs network
destination_cluster::> snapmirror create -source-path [sourceVSERVER]: -destination-
path [destinationVSERVER]: -throttle unlimited -identity-preserve true -policy
exclude_LIF -schedule 5min
destination_cluster::> snapmirror initialize -destination-path [destinationVSERVER]
```

Assign a new IP address to the Cloud Volumes ONTAP EC2 Instance

1. Open the Amazon EC2 console at `https://console.aws.amazon.com/ec2/`
2. In the navigation pane, select Network Interfaces, and then select the network interface attached to the instance.
3. Choose Actions > Manage IP Addresses.
4. Under IPv4 Addresses, select Assign New IP, and then click Yes, Update.
5. Copy the new assigned IP address (You will use it when later creating the LIF).

Create a LIF in the destination SVM

```
destination_cluster::>  net int create -vserver [destinationVSERVER] -lif
[destinationVSERVER]_data_lif -role data -data-protocol cifs,nfs,fcache -home-node
[destinationNODE] -home-port e0b -address [destinationVserverIP1] -netmask-length 25
-status-admin up
destination_cluster::>  net int create -vserver [destinationVSERVER] -lif
[destinationVSERVER]_mgmt_lif -role data -data-protocol cifs,nfs,fcache -home-node
[destinationNODE] -home-port e0b -address [destinationVserverIP2] -netmask-length 25
-status-admin up
destination_cluster::>  route create -vserver [destinationVSERVER] -destination
0.0.0.0/0 -gateway [gatewayIP] -metric 20
```

Activate SVM-DR

```
destination_cluster::>    vserver stop [destinationVSERVER]
source_cluster::>         vserver stop [sourceVSERVER]
destination_cluster::>    snapmirror update [destinationVSERVER]
```

```
destination_cluster::>    snapmirror break [destinationVSERVER]
destination_cluster::>    vserver start [destinationVSERVER]
```

Where to find additional information

Additional documentation is available to help you activate the destination Storage Virtual Machine (SVM) to test the disaster recovery setup or when a disaster occurs. You can also learn more about how to recreate and reactivate the source SVM after the disaster.

Reference guides

You can activate the destination SVM, and recreate and reactivate the source SVM by using the following documentation:

- *SVM disaster recovery express activation*

You can use the `snapmirror` commands to manage the SVM disaster recovery relationships.

- *ONTAP 9 commands*

For conceptual information about SVM disaster recovery, use the following documentation:

- *ONTAP concepts*

Copyright information

Trademark information

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.

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