Foreign LUN import

Enterprise applications

NetApp
April 25, 2024

Table of Contents

Foreign LUN import

Oracle migration with FLI - planning

The procedures to migrate SAN resources using FLI are documented in NetApp TR-4380: SAN Migration Using Foreign LUN Import.

From a database and host point of view, no special steps are required. After the FC zones are updated and the LUNs become available on ONTAP, the LVM should be able to read the LVM metadata from the LUNs. Also, the volume groups are ready for use with no further configuration steps. In rare cases, environments might include configuration files that were hard-coded with references to the prior storage array. For example, a Linux system that included `/etc/multipath.conf` rules that referenced a WWN of a given device must be updated to reflect the changes introduced by FLI.

Reference the NetApp Compatibility Matrix for information on supported configurations. If your environment is not included, contact your NetApp representative for assistance.

This example shows the migration of both ASM and LVM LUNs hosted on a Linux server. FLI is supported on other operating systems, and, although the host-side commands might differ, the principles are the same, and the ONTAP procedures are identical.

Identify LVM LUNs

The first step in preparation is to identify the LUNs to be migrated. In the example shown here, two SAN-based file systems are mounted at `/orabin` and `/backups`.

```
[root@host1 ~]# df -k
Filesystem                      1K-blocks      Used Available Use%
Mounted on
/dev/mapper/rhel-root            52403200   8811464  43591736  17% /
devtmpfs                         65882776         0  65882776   0% /dev
...
fas8060-nfs-public:/install     199229440 119368128  79861312  60%
/install
/dev/mapper/sanvg-lvorabin       20961280  12348476   8612804  59%
/orabin
/dev/mapper/sanvg-lvbackups      73364480  62947536  10416944  86%
/backups
```

The name of the volume group can be extracted from the device name, which uses the format (volume group name)-(logical volume name). In this case, the volume group is called `sanvg`.

The `pvdisplay` command can be used as follows to identify the LUNs that support this volume group. In this case, there are 10 LUNs that make up the `sanvg` volume group.

```
[root@host1 ~]# pvdisplay -C -o pv_name,pv_size,pv_fmt,vg_name
  PV                                                PSize   VG
  /dev/mapper/3600a0980383030445424487556574266  10.00g sanvg
  /dev/mapper/3600a0980383030445424487556574267  10.00g sanvg
  /dev/mapper/3600a0980383030445424487556574268  10.00g sanvg
  /dev/mapper/3600a0980383030445424487556574269  10.00g sanvg
  /dev/mapper/3600a098038303044542448755657426a  10.00g sanvg
  /dev/mapper/3600a098038303044542448755657426b  10.00g sanvg
  /dev/mapper/3600a098038303044542448755657426c  10.00g sanvg
  /dev/mapper/3600a098038303044542448755657426d  10.00g sanvg
  /dev/mapper/3600a098038303044542448755657426e  10.00g sanvg
  /dev/mapper/3600a098038303044542448755657426f  10.00g sanvg
  /dev/sda2                                       278.38g rhel
```

Identify ASM LUNs

ASM LUNs must also be migrated. To obtain the number of LUNs and LUN paths from sqlplus as the sysasm user, run the following command:

```
SQL> select path||' '||os_mb from v$asm_disk;
PATH||''||OS_MB
--------------------------------------------------------------------------------
------
/dev/oracleasm/disks/ASM0 10240
/dev/oracleasm/disks/ASM9 10240
/dev/oracleasm/disks/ASM8 10240
/dev/oracleasm/disks/ASM7 10240
/dev/oracleasm/disks/ASM6 10240
/dev/oracleasm/disks/ASM5 10240
/dev/oracleasm/disks/ASM4 10240
/dev/oracleasm/disks/ASM1 10240
/dev/oracleasm/disks/ASM3 10240
/dev/oracleasm/disks/ASM2 10240
10 rows selected.
SQL>
```

FC network changes

The current environment contains 20 LUNs to be migrated. Update the current SAN so that ONTAP can access the current LUNs. Data is not migrated yet, but ONTAP must read configuration information from the current LUNs to create the new home for that data.

At a minimum, at least one HBA port on the AFF/FAS system must be configured as an initiator port. In addition, the FC zones must be updated so that ONTAP can access the LUNs on the foreign storage array. Some storage arrays have LUN masking configured, which limits which WWNs can access a given LUN. In

such cases, LUN masking must also be updated to grant access to the ONTAP WWNs.

After this step is completed, ONTAP should be able to view the foreign storage array with the `storage array show` command. The key field it returns is the prefix that is used to identify the foreign LUN on the system. In the example below, the LUNs on the foreign array `FOREIGN_1` appear within ONTAP using the prefix of `FOR-1`.

Identify foreign array

```
Cluster01::> storage array show -fields name,prefix
name          prefix
------------- ------
FOREIGN_1     FOR-1
Cluster01::>
```

Identify foreign LUNs

The LUNs can be listed by passing the `array-name` to the `storage disk show` command. The data returned is referenced multiple times during the migration procedure.

```
Cluster01::> storage disk show -array-name FOREIGN_1 -fields disk,serial
disk     serial-number
-------- -------------
FOR-1.1  800DT$HuVWBX
FOR-1.2  800DT$HuVWBZ
FOR-1.3  800DT$HuVWBW
FOR-1.4  800DT$HuVWBY
FOR-1.5  800DT$HuVWB/
FOR-1.6  800DT$HuVWBa
FOR-1.7  800DT$HuVWBd
FOR-1.8  800DT$HuVWBb
FOR-1.9  800DT$HuVWBc
FOR-1.10 800DT$HuVWBe
FOR-1.11 800DT$HuVWBf
FOR-1.12 800DT$HuVWBg
FOR-1.13 800DT$HuVWBi
FOR-1.14 800DT$HuVWBh
FOR-1.15 800DT$HuVWBj
FOR-1.16 800DT$HuVWBk
FOR-1.17 800DT$HuVWBm
FOR-1.18 800DT$HuVWBl
FOR-1.19 800DT$HuVWBo
FOR-1.20 800DT$HuVWBn
20 entries were displayed.
Cluster01::>
```

Register foreign array LUNs as import candidates

The foreign LUNs are initially classified as any particular LUN type. Before data can be imported, the LUNs must be tagged as foreign and therefore a candidate for the import process. This step is completed by passing the serial number to the `storage disk modify` command, as shown in the following example. Note that this process tags only the LUN as foreign within ONTAP. No data is written to the foreign LUN itself.

```
Cluster01::*> storage disk modify {-serial-number 800DT$HuVWBW} -is
-foreign true
Cluster01::*> storage disk modify {-serial-number 800DT$HuVWBX} -is
-foreign true
...
Cluster01::*> storage disk modify {-serial-number 800DT$HuVWBn} -is
-foreign true
Cluster01::*> storage disk modify {-serial-number 800DT$HuVWBo} -is
-foreign true
Cluster01::*>
```

Create volumes to host migrated LUNs

A volume is needed to host the migrated LUNs. The exact volume configuration depends on the overall plan to leverage ONTAP features. In this example, the ASM LUNs are placed into one volume and the LVM LUNs are placed in a second volume. Doing so allows you to manage the LUNs as independent groups for purposes such as tiering, creation of snapshots, or setting QoS controls.

Set the `snapshot-policy `to `none`. The migration process can include a great deal of data turnover. Therefore, there might be a large increase in space consumption if snapshots are created by accident because unwanted data is captured in the snapshots.

```
Cluster01::> volume create -volume new_asm -aggregate data_02 -size 120G
-snapshot-policy none
[Job 1152] Job succeeded: Successful
Cluster01::> volume create -volume new_lvm -aggregate data_02 -size 120G
-snapshot-policy none
[Job 1153] Job succeeded: Successful
Cluster01::>
```

Create ONTAP LUNs

After the volumes are created, the new LUNs must be created. Normally, the creation of a LUN requires the user to specify such information as the LUN size, but in this case the foreign-disk argument is passed to the command. As a result, ONTAP replicates the current LUN configuration data from the specified serial number. It also uses the LUN geometry and partition table data to adjust LUN alignment and establish optimum performance.

In this step, serial numbers must be cross-referenced against the foreign array to make sure that the correct foreign LUN is matched to the correct new LUN.

```
Cluster01::*> lun create -vserver vserver1 -path /vol/new_asm/LUN0 -ostype
linux -foreign-disk 800DT$HuVWBW
Created a LUN of size 10g (10737418240)
Cluster01::*> lun create -vserver vserver1 -path /vol/new_asm/LUN1 -ostype
linux -foreign-disk 800DT$HuVWBX
Created a LUN of size 10g (10737418240)
...
Created a LUN of size 10g (10737418240)
Cluster01::*> lun create -vserver vserver1 -path /vol/new_lvm/LUN8 -ostype
linux -foreign-disk 800DT$HuVWBn
Created a LUN of size 10g (10737418240)
Cluster01::*> lun create -vserver vserver1 -path /vol/new_lvm/LUN9 -ostype
linux -foreign-disk 800DT$HuVWBo
Created a LUN of size 10g (10737418240)
```

Create import relationships

The LUNs have now been created but are not configured as a replication destination. Before this step can be taken, the LUNs must first be placed offline. This extra step is designed to protect data from user errors. If ONTAP allowed a migration to be performed on an online LUN, it would create a risk that a typographical error could result in overwriting active data. The extra step of forcing the user to first take a LUN offline helps verify that the correct target LUN is used as a migration destination.

```
Cluster01::*> lun offline -vserver vserver1 -path /vol/new_asm/LUN0
Warning: This command will take LUN "/vol/new_asm/LUN0" in Vserver
         "vserver1" offline.
Do you want to continue? {y|n}: y
Cluster01::*> lun offline -vserver vserver1 -path /vol/new_asm/LUN1
Warning: This command will take LUN "/vol/new_asm/LUN1" in Vserver
         "vserver1" offline.
Do you want to continue? {y|n}: y
...
Warning: This command will take LUN "/vol/new_lvm/LUN8" in Vserver
         "vserver1" offline.
Do you want to continue? {y|n}: y
Cluster01::*> lun offline -vserver vserver1 -path /vol/new_lvm/LUN9
Warning: This command will take LUN "/vol/new_lvm/LUN9" in Vserver
         "vserver1" offline.
Do you want to continue? {y|n}: y
```

After the LUNs are offline, you can establish the import relationship by passing the foreign LUN serial number to the `lun import create` command.

```
Cluster01::*> lun import create -vserver vserver1 -path /vol/new_asm/LUN0
-foreign-disk 800DT$HuVWBW
Cluster01::*> lun import create -vserver vserver1 -path /vol/new_asm/LUN1
-foreign-disk 800DT$HuVWBX
...
Cluster01::*> lun import create -vserver vserver1 -path /vol/new_lvm/LUN8
-foreign-disk 800DT$HuVWBn
Cluster01::*> lun import create -vserver vserver1 -path /vol/new_lvm/LUN9
-foreign-disk 800DT$HuVWBo
Cluster01::*>
```

After all import relationships are established, the LUNs can be placed back online.

```
Cluster01::*> lun online -vserver vserver1 -path /vol/new_asm/LUN0
Cluster01::*> lun online -vserver vserver1 -path /vol/new_asm/LUN1
...
Cluster01::*> lun online -vserver vserver1 -path /vol/new_lvm/LUN8
Cluster01::*> lun online -vserver vserver1 -path /vol/new_lvm/LUN9
Cluster01::*>
```

Create initiator group

An initiator group (igroup) is part of the ONTAP LUN masking architecture. A newly created LUN is not accessible unless a host is first granted access. This is done by creating an igroup that lists either the FC WWNs or iSCSI initiator names that should be granted access. At the time this report was written, FLI was supported only for FC LUNs. However, converting to iSCSI postmigration is a simple task, as shown in Protocol Conversion.

In this example, an igroup is created that contains two WWNs that correspond to the two ports available on the host's HBA.

```
Cluster01::*> igroup create linuxhost -protocol fcp -ostype linux
-initiator 21:00:00:0e:1e:16:63:50 21:00:00:0e:1e:16:63:51
```

Map new LUNs to host

Following igroup creation, the LUNs are then mapped to the defined igroup. These LUNs are available only to the WWNs included in this igroup. NetApp assumes at this stage in the migration process that the host has not been zoned to ONTAP. This is important because if the host is simultaneously zoned to the foreign array and the new ONTAP system, then there is a risk that LUNs bearing the same serial number could be discovered on each array. This situation could lead to multipath malfunctions or damage to data.

```
Cluster01::*> lun map -vserver vserver1 -path /vol/new_asm/LUN0 -igroup
linuxhost
Cluster01::*> lun map -vserver vserver1 -path /vol/new_asm/LUN1 -igroup
linuxhost
...
Cluster01::*> lun map -vserver vserver1 -path /vol/new_lvm/LUN8 -igroup
linuxhost
Cluster01::*> lun map -vserver vserver1 -path /vol/new_lvm/LUN9 -igroup
linuxhost
Cluster01::*>
```

Oracle migration with FLI - cutover

Some disruption during a foreign LUN import is unavoidable because of the need to change the FC network configuration. However, the disruption does not have to last much longer than the time required to restart the database environment and update FC zoning to switch the host FC connectivity from the foreign LUN to ONTAP.

This process can be summarized as follows:

1. Quiesce all LUN activity on the foreign LUNs.
2. Redirect host FC connections to the new ONTAP system.
3. Trigger the import process.
4. Rediscover the LUNs.
5. Restart the database.

You do not need to wait for the migration process to complete. As soon as the migration for a given LUN begins, it is available on ONTAP and can serve data while the data copy process continues. All reads are passed through to the foreign LUN, and all writes are synchronously written to both arrays. The copy operation is very fast and the overhead of redirecting FC traffic is minimal, so any impact on performance should be transient and minimal. If there is concern, you can delay restarting the environment until after the migration process is complete and the import relationships have been deleted.

Shut down database

The first step in quiescing the environment in this example is to shut down the database.

```
[oracle@host1 bin]$ . oraenv
ORACLE_SID = [oracle] ? FLIDB
The Oracle base remains unchanged with value /orabin
[oracle@host1 bin]$ sqlplus / as sysdba
SQL*Plus: Release 12.1.0.2.0
Copyright (c) 1982, 2014, Oracle.  All rights reserved.
Connected to:
Oracle Database 12c Enterprise Edition Release 12.1.0.2.0 - 64bit
Production
With the Partitioning, Automatic Storage Management, OLAP, Advanced
Analytics
and Real Application Testing options
SQL> shutdown immediate;
Database closed.
Database dismounted.
ORACLE instance shut down.
SQL>
```

Shut down grid services

One of the SAN-based file systems being migrated also includes the Oracle ASM services. Quiescing the underlying LUNs requires dismounting the file systems, which in turn means stopping any processes with open files on this file system.

```
[oracle@host1 bin]$ ./crsctl stop has -f
CRS-2791: Starting shutdown of Oracle High Availability Services-managed
resources on 'host1'
CRS-2673: Attempting to stop 'ora.evmd' on 'host1'
CRS-2673: Attempting to stop 'ora.DATA.dg' on 'host1'
CRS-2673: Attempting to stop 'ora.LISTENER.lsnr' on 'host1'
CRS-2677: Stop of 'ora.DATA.dg' on 'host1' succeeded
CRS-2673: Attempting to stop 'ora.asm' on 'host1'
CRS-2677: Stop of 'ora.LISTENER.lsnr' on 'host1' succeeded
CRS-2677: Stop of 'ora.evmd' on 'host1' succeeded
CRS-2677: Stop of 'ora.asm' on 'host1' succeeded
CRS-2673: Attempting to stop 'ora.cssd' on 'host1'
CRS-2677: Stop of 'ora.cssd' on 'host1' succeeded
CRS-2793: Shutdown of Oracle High Availability Services-managed resources
on 'host1' has completed
CRS-4133: Oracle High Availability Services has been stopped.
[oracle@host1 bin]$
```

Dismount file systems

If all the processes are shut down, the umount operation succeeds. If permission is denied, there must be a process with a lock on the file system. The `fuser` command can help identify these processes.

```
[root@host1 ~]# umount /orabin
[root@host1 ~]# umount /backups
```

Deactivate volume groups

After all file systems in a given volume group are dismounted, the volume group can be deactivated.

```
[root@host1 ~]# vgchange --activate n sanvg
  0 logical volume(s) in volume group "sanvg" now active
[root@host1 ~]#
```

FC network changes

The FC zones can now be updated to remove all access from the host to the foreign array and establish access to ONTAP.

Start import process

To start the LUN import processes, run the `lun import start` command.

```
Cluster01::lun import*> lun import start -vserver vserver1 -path
/vol/new_asm/LUN0
Cluster01::lun import*> lun import start -vserver vserver1 -path
/vol/new_asm/LUN1
...
Cluster01::lun import*> lun import start -vserver vserver1 -path
/vol/new_lvm/LUN8
Cluster01::lun import*> lun import start -vserver vserver1 -path
/vol/new_lvm/LUN9
Cluster01::lun import*>
```

Monitor import progress

The import operation can be monitored with the `lun import show` command. As shown below, the import of all 20 LUNs is underway, which means that data is now accessible through ONTAP even though the data copy operation still progresses.

```
Cluster01::lun import*> lun import show -fields path,percent-complete
vserver   foreign-disk path              percent-complete
--------- ------------ ----------------- ----------------
vserver1  800DT$HuVWB/ /vol/new_asm/LUN4 5
vserver1  800DT$HuVWBW /vol/new_asm/LUN0 5
vserver1  800DT$HuVWBX /vol/new_asm/LUN1 6
vserver1  800DT$HuVWBY /vol/new_asm/LUN2 6
vserver1  800DT$HuVWBZ /vol/new_asm/LUN3 5
vserver1  800DT$HuVWBa /vol/new_asm/LUN5 4
vserver1  800DT$HuVWBb /vol/new_asm/LUN6 4
vserver1  800DT$HuVWBc /vol/new_asm/LUN7 4
vserver1  800DT$HuVWBd /vol/new_asm/LUN8 4
vserver1  800DT$HuVWBe /vol/new_asm/LUN9 4
vserver1  800DT$HuVWBf /vol/new_lvm/LUN0 5
vserver1  800DT$HuVWBg /vol/new_lvm/LUN1 4
vserver1  800DT$HuVWBh /vol/new_lvm/LUN2 4
vserver1  800DT$HuVWBi /vol/new_lvm/LUN3 3
vserver1  800DT$HuVWBj /vol/new_lvm/LUN4 3
vserver1  800DT$HuVWBk /vol/new_lvm/LUN5 3
vserver1  800DT$HuVWBl /vol/new_lvm/LUN6 4
vserver1  800DT$HuVWBm /vol/new_lvm/LUN7 3
vserver1  800DT$HuVWBn /vol/new_lvm/LUN8 2
vserver1  800DT$HuVWBo /vol/new_lvm/LUN9 2
20 entries were displayed.
```

If you require an offline process, delay rediscovering or restarting services until the `lun import show` command indicates that all migration is successful and complete. You can then complete the migration process as described in Foreign LUN Import—Completion.

If you require an online migration, proceed to rediscover the LUNs in their new home and bring up the services.

Scan for SCSI device changes

In most cases, the simplest option to rediscover new LUNs is to restart the host. Doing so automatically removes old stale devices, properly discovers all new LUNs, and builds associated devices such as multipathing devices. The example here shows a wholly online process for demonstration purposes.

Caution: Before restarting a host, make sure that all entries in `/etc/fstab` that reference migrated SAN resources are commented out. If this is not done and there are problems with LUN access, the OS might not boot. This situation does not damage data. However, it can be very inconvenient to boot into rescue mode or a similar mode and correct the `/etc/fstab` so that the OS can be booted to enable troubleshooting.

The LUNs on the version of Linux used in this example can be rescanned with the `rescan-scsi-bus.sh` command. If the command is successful, each LUN path should appear in the output. The output can be difficult to interpret, but, if the zoning and igroup configuration was correct, many LUNs should appear that include a `NETAPP` vendor string.

```
[root@host1 /]# rescan-scsi-bus.sh
Scanning SCSI subsystem for new devices
Scanning host 0 for  SCSI target IDs  0 1 2 3 4 5 6 7, all LUNs
 Scanning for device 0 2 0 0 ...
OLD: Host: scsi0 Channel: 02 Id: 00 Lun: 00
      Vendor: LSI      Model: RAID SAS 6G 0/1  Rev: 2.13
      Type:   Direct-Access                    ANSI SCSI revision: 05
Scanning host 1 for  SCSI target IDs  0 1 2 3 4 5 6 7, all LUNs
 Scanning for device 1 0 0 0 ...
OLD: Host: scsi1 Channel: 00 Id: 00 Lun: 00
      Vendor: Optiarc  Model: DVD RW AD-7760H  Rev: 1.41
      Type:   CD-ROM                           ANSI SCSI revision: 05
Scanning host 2 for  SCSI target IDs  0 1 2 3 4 5 6 7, all LUNs
Scanning host 3 for  SCSI target IDs  0 1 2 3 4 5 6 7, all LUNs
Scanning host 4 for  SCSI target IDs  0 1 2 3 4 5 6 7, all LUNs
Scanning host 5 for  SCSI target IDs  0 1 2 3 4 5 6 7, all LUNs
Scanning host 6 for  SCSI target IDs  0 1 2 3 4 5 6 7, all LUNs
Scanning host 7 for  all SCSI target IDs, all LUNs
 Scanning for device 7 0 0 10 ...
OLD: Host: scsi7 Channel: 00 Id: 00 Lun: 10
      Vendor: NETAPP   Model: LUN C-Mode       Rev: 8300
      Type:   Direct-Access                    ANSI SCSI revision: 05
 Scanning for device 7 0 0 11 ...
OLD: Host: scsi7 Channel: 00 Id: 00 Lun: 11
      Vendor: NETAPP   Model: LUN C-Mode       Rev: 8300
      Type:   Direct-Access                    ANSI SCSI revision: 05
 Scanning for device 7 0 0 12 ...
...
OLD: Host: scsi9 Channel: 00 Id: 01 Lun: 18
      Vendor: NETAPP   Model: LUN C-Mode       Rev: 8300
      Type:   Direct-Access                    ANSI SCSI revision: 05
 Scanning for device 9 0 1 19 ...
OLD: Host: scsi9 Channel: 00 Id: 01 Lun: 19
      Vendor: NETAPP   Model: LUN C-Mode       Rev: 8300
      Type:   Direct-Access                    ANSI SCSI revision: 05
0 new or changed device(s) found.
0 remapped or resized device(s) found.
0 device(s) removed.
```

Check for multipath devices

The LUN discovery process also triggers the recreation of multipath devices, but the Linux multipathing driver is known to have occasional problems. The output of `multipath -ll` should be checked to verify that the output looks as expected. For example, the output below shows multipath devices associated with a `NETAPP` vendor string. Each device has four paths, with two at a priority of 50 and two at a priority of 10. Although the exact output can vary with different versions of Linux, this output looks as expected.

Reference the host utilities documentation for the version of Linux you use to verify that the `/etc/multipath.conf` settings are correct.

```
[root@host1 /]# multipath -ll
3600a098038303558735d493762504b36 dm-5 NETAPP  ,LUN C-Mode
size=10G features='4 queue_if_no_path pg_init_retries 50
retain_attached_hw_handle' hwhandler='1 alua' wp=rw
|-+- policy='service-time 0' prio=50 status=active
| |- 7:0:1:4  sdat 66:208 active ready running
| `- 9:0:1:4  sdbn 68:16  active ready running
`-+- policy='service-time 0' prio=10 status=enabled
  |- 7:0:0:4  sdf  8:80   active ready running
  `- 9:0:0:4  sdz  65:144 active ready running
3600a098038303558735d493762504b2d dm-10 NETAPP  ,LUN C-Mode
size=10G features='4 queue_if_no_path pg_init_retries 50
retain_attached_hw_handle' hwhandler='1 alua' wp=rw
|-+- policy='service-time 0' prio=50 status=active
| |- 7:0:1:8  sdax 67:16  active ready running
| `- 9:0:1:8  sdbr 68:80  active ready running
`-+- policy='service-time 0' prio=10 status=enabled
  |- 7:0:0:8  sdj  8:144  active ready running
  `- 9:0:0:8  sdad 65:208 active ready running
...
3600a098038303558735d493762504b37 dm-8 NETAPP  ,LUN C-Mode
size=10G features='4 queue_if_no_path pg_init_retries 50
retain_attached_hw_handle' hwhandler='1 alua' wp=rw
|-+- policy='service-time 0' prio=50 status=active
| |- 7:0:1:5  sdau 66:224 active ready running
| `- 9:0:1:5  sdbo 68:32  active ready running
`-+- policy='service-time 0' prio=10 status=enabled
  |- 7:0:0:5  sdg  8:96   active ready running
  `- 9:0:0:5  sdaa 65:160 active ready running
3600a098038303558735d493762504b4b dm-22 NETAPP  ,LUN C-Mode
size=10G features='4 queue_if_no_path pg_init_retries 50
retain_attached_hw_handle' hwhandler='1 alua' wp=rw
|-+- policy='service-time 0' prio=50 status=active
| |- 7:0:1:19 sdbi 67:192 active ready running
| `- 9:0:1:19 sdcc 69:0   active ready running
`-+- policy='service-time 0' prio=10 status=enabled
  |- 7:0:0:19 sdu  65:64  active ready running
  `- 9:0:0:19 sdao 66:128 active ready running
```

Reactivate LVM volume group

If the LVM LUNs have been properly discovered, the `vgchange --activate y` command should succeed.

This is a good example of the value of a logical volume manager. A change in the WWN of a LUN or even a serial number is unimportant because the volume group metadata is written on the LUN itself.

The OS scanned the LUNs and discovered a small amount of data written on the LUN that identifies it as a physical volume belonging to the `sanvg volumegroup`. It then built all of the required devices. All that is required is to reactivate the volume group.

```
[root@host1 /]# vgchange --activate y sanvg
  Found duplicate PV fpCzdLTuKfy2xDZjai1NliJh3TjLUBiT: using
/dev/mapper/3600a098038303558735d493762504b46 not /dev/sdp
  Using duplicate PV /dev/mapper/3600a098038303558735d493762504b46 from
subsystem DM, ignoring /dev/sdp
  2 logical volume(s) in volume group "sanvg" now active
```

Remount file systems

After the volume group is reactivated, the file systems can be mounted with all of the original data intact. As discussed previously, the file systems are fully operational even if data replication is still active in the back group.

```
[root@host1 /]# mount /orabin
[root@host1 /]# mount /backups
[root@host1 /]# df -k
Filesystem                      1K-blocks      Used Available Use%
Mounted on
/dev/mapper/rhel-root            52403200   8837100  43566100  17% /
devtmpfs                         65882776         0  65882776   0% /dev
tmpfs                             6291456        84   6291372   1%
/dev/shm
tmpfs                            65898668      9884  65888784   1% /run
tmpfs                            65898668         0  65898668   0%
/sys/fs/cgroup
/dev/sda1                          505580    224828    280752  45% /boot
fas8060-nfs-public:/install      199229440 119368256  79861184  60%
/install
fas8040-nfs-routable:/snapomatic   9961472     30528   9930944   1%
/snapomatic
tmpfs                            13179736        16  13179720   1%
/run/user/42
tmpfs                            13179736         0  13179736   0%
/run/user/0
/dev/mapper/sanvg-lvorabin       20961280  12357456   8603824  59%
/orabin
/dev/mapper/sanvg-lvbackups      73364480  62947536  10416944  86%
/backups
```

Rescan for ASM devices

The ASMlib devices should have been rediscovered when the SCSI devices were rescanned. Rediscovery can be verified online by restarting ASMlib and then scanning the disks.

This step is only relevant to ASM configurations where ASMlib is used.

Caution: Where ASMlib is not used, the `/dev/mapper` devices should have been automatically recreated. However, the permissions might not be correct. You must set special permissions on the underlying devices for ASM in the absence of ASMlib. Doing so is usually accomplished through special entries in either the `/etc/multipath.conf` or `udev` rules, or possibly in both rule sets. These files might need to be updated to reflect changes in the environment in terms of WWNs or serial numbers to make sure that the ASM devices still have the correct permissions.

In this example, restarting ASMlib and scanning for disks show the same 10 ASM LUNs as the original environment.

```
[root@host1 /]# oracleasm exit
Unmounting ASMlib driver filesystem: /dev/oracleasm
Unloading module "oracleasm": oracleasm
[root@host1 /]# oracleasm init
Loading module "oracleasm": oracleasm
Configuring "oracleasm" to use device physical block size
Mounting ASMlib driver filesystem: /dev/oracleasm
[root@host1 /]# oracleasm scandisks
Reloading disk partitions: done
Cleaning any stale ASM disks...
Scanning system for ASM disks...
Instantiating disk "ASM0"
Instantiating disk "ASM1"
Instantiating disk "ASM2"
Instantiating disk "ASM3"
Instantiating disk "ASM4"
Instantiating disk "ASM5"
Instantiating disk "ASM6"
Instantiating disk "ASM7"
Instantiating disk "ASM8"
Instantiating disk "ASM9"
```

Restart grid services

Now that the LVM and ASM devices are online and available, the grid services can be restarted.

```
[root@host1 /]# cd /orabin/product/12.1.0/grid/bin
[root@host1 bin]# ./crsctl start has
```

Restart database

After the grid services have been restarted, the database can be brought up. It might be necessary to wait a few minutes for the ASM services to become fully available before trying to start the database.

```
[root@host1 bin]# su - oracle
[oracle@host1 ~]$ . oraenv
ORACLE_SID = [oracle] ? FLIDB
The Oracle base has been set to /orabin
[oracle@host1 ~]$ sqlplus / as sysdba
SQL*Plus: Release 12.1.0.2.0
Copyright (c) 1982, 2014, Oracle.  All rights reserved.
Connected to an idle instance.
SQL> startup
ORACLE instance started.
Total System Global Area 3221225472 bytes
Fixed Size                  4502416 bytes
Variable Size            1207962736 bytes
Database Buffers         1996488704 bytes
Redo Buffers               12271616 bytes
Database mounted.
Database opened.
SQL>
```

Oracle migration with FLI - completion

From a host point of view, the migration is complete, but I/O is still served from the foreign array until the import relationships are deleted.

Before deleting the relationships, you must confirm that the migration process is complete for all LUNs.

```
Cluster01::*> lun import show -vserver vserver1 -fields foreign-
disk,path,operational-state
vserver   foreign-disk path              operational-state
--------- ------------ ----------------- -----------------
vserver1 800DT$HuVWB/ /vol/new_asm/LUN4 completed
vserver1 800DT$HuVWBW /vol/new_asm/LUN0 completed
vserver1 800DT$HuVWBX /vol/new_asm/LUN1 completed
vserver1 800DT$HuVWBY /vol/new_asm/LUN2 completed
vserver1 800DT$HuVWBZ /vol/new_asm/LUN3 completed
vserver1 800DT$HuVWBa /vol/new_asm/LUN5 completed
vserver1 800DT$HuVWBb /vol/new_asm/LUN6 completed
vserver1 800DT$HuVWBc /vol/new_asm/LUN7 completed
vserver1 800DT$HuVWBd /vol/new_asm/LUN8 completed
vserver1 800DT$HuVWBe /vol/new_asm/LUN9 completed
vserver1 800DT$HuVWBf /vol/new_lvm/LUN0 completed
vserver1 800DT$HuVWBg /vol/new_lvm/LUN1 completed
vserver1 800DT$HuVWBh /vol/new_lvm/LUN2 completed
vserver1 800DT$HuVWBi /vol/new_lvm/LUN3 completed
vserver1 800DT$HuVWBj /vol/new_lvm/LUN4 completed
vserver1 800DT$HuVWBk /vol/new_lvm/LUN5 completed
vserver1 800DT$HuVWBl /vol/new_lvm/LUN6 completed
vserver1 800DT$HuVWBm /vol/new_lvm/LUN7 completed
vserver1 800DT$HuVWBn /vol/new_lvm/LUN8 completed
vserver1 800DT$HuVWBo /vol/new_lvm/LUN9 completed
20 entries were displayed.
```

Delete import relationships

When the migration process is complete, delete the migration relationship. After you have done so, I/O is served exclusively from the drives on ONTAP.

```
Cluster01::*> lun import delete -vserver vserver1 -path /vol/new_asm/LUN0
Cluster01::*> lun import delete -vserver vserver1 -path /vol/new_asm/LUN1
...
Cluster01::*> lun import delete -vserver vserver1 -path /vol/new_lvm/LUN8
Cluster01::*> lun import delete -vserver vserver1 -path /vol/new_lvm/LUN9
```

Deregister foreign LUNs

Finally, modify the disk to remove the `is-foreign` designation.

```
Cluster01::*> storage disk modify {-serial-number 800DT$HuVWBW} -is
-foreign false
Cluster01::*> storage disk modify {-serial-number 800DT$HuVWBX} -is
-foreign false
...
Cluster01::*> storage disk modify {-serial-number 800DT$HuVWBn} -is
-foreign false
Cluster01::*> storage disk modify {-serial-number 800DT$HuVWBo} -is
-foreign false
Cluster01::*>
```

Oracle migration with FLI - protocol conversion

Changing the protocol used to access a LUN is a common requirement.

In some cases, it is part of an overall strategy to migrate data to the cloud. TCP/IP is the protocol of the cloud, and changing from FC to iSCSI allows easier migration into various cloud environments. In other cases, iSCSI might be desirable to leverage the decreased costs of an IP SAN. On occasion, a migration might use a different protocol as a temporary measure. For example, if a foreign array and ONTAP based LUNs cannot coexist on the same HBAs, you can use iSCSI LUNs long enough to copy data from the old array. You can then convert back to FC after the old LUNs are removed from the system.

The following procedure demonstrates conversion from FC to iSCSI, but the overall principles apply to a reverse iSCSI to FC conversion.

Install iSCSI initiator

Most operating systems include a software iSCSI initiator by default, but if one is not included, it can be easily installed.

```
[root@host1 /]# yum install -y iscsi-initiator-utils
Loaded plugins: langpacks, product-id, search-disabled-repos,
subscription-
              : manager
Resolving Dependencies
--> Running transaction check
---> Package iscsi-initiator-utils.x86_64 0:6.2.0.873-32.el7 will be
updated
--> Processing Dependency: iscsi-initiator-utils = 6.2.0.873-32.el7 for
package: iscsi-initiator-utils-iscsiuio-6.2.0.873-32.el7.x86_64
---> Package iscsi-initiator-utils.x86_64 0:6.2.0.873-32.0.2.el7 will be
an update
--> Running transaction check
---> Package iscsi-initiator-utils-iscsiuio.x86_64 0:6.2.0.873-32.el7 will
be updated
---> Package iscsi-initiator-utils-iscsiuio.x86_64 0:6.2.0.873-32.0.2.el7
```

```
will be an update
--> Finished Dependency Resolution
Dependencies Resolved
================================================================================
===
 Package                         Arch   Version              Repository
Size
================================================================================
===
Updating:
 iscsi-initiator-utils           x86_64 6.2.0.873-32.0.2.el7 ol7_latest 416
k
Updating for dependencies:
 iscsi-initiator-utils-iscsiuio x86_64 6.2.0.873-32.0.2.el7 ol7_latest  84
k
Transaction Summary
================================================================================
===
Upgrade  1 Package (+1 Dependent package)
Total download size: 501 k
Downloading packages:
No Presto metadata available for ol7_latest
(1/2): iscsi-initiator-utils-6.2.0.873-32.0.2.el7.x86_6 | 416 kB   00:00
(2/2): iscsi-initiator-utils-iscsiuio-6.2.0.873-32.0.2. |  84 kB   00:00
--------------------------------------------------------------------------------
---
Total                                           2.8 MB/s | 501 kB
00:00Cluster01
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Updating   : iscsi-initiator-utils-iscsiuio-6.2.0.873-32.0.2.el7.x86
1/4
  Updating   : iscsi-initiator-utils-6.2.0.873-32.0.2.el7.x86_64
2/4
  Cleanup    : iscsi-initiator-utils-iscsiuio-6.2.0.873-32.el7.x86_64
3/4
  Cleanup    : iscsi-initiator-utils-6.2.0.873-32.el7.x86_64
4/4
rhel-7-server-eus-rpms/7Server/x86_64/productid       | 1.7 kB     00:00
rhel-7-server-rpms/7Server/x86_64/productid           | 1.7 kB     00:00
  Verifying  : iscsi-initiator-utils-6.2.0.873-32.0.2.el7.x86_64
1/4
  Verifying  : iscsi-initiator-utils-iscsiuio-6.2.0.873-32.0.2.el7.x86
2/4
```

```
  Verifying  : iscsi-initiator-utils-iscsiuio-6.2.0.873-32.el7.x86_64
3/4
  Verifying  : iscsi-initiator-utils-6.2.0.873-32.el7.x86_64
4/4
Updated:
  iscsi-initiator-utils.x86_64 0:6.2.0.873-32.0.2.el7
Dependency Updated:
  iscsi-initiator-utils-iscsiuio.x86_64 0:6.2.0.873-32.0.2.el7
Complete!
[root@host1 /]#
```

Identify iSCSI initiator name

A unique iSCSI initiator name is generated during the installation process. On Linux, it is located in the `/etc/iscsi/initiatorname.iscsi` file. This name is used to identify the host on the IP SAN.

```
[root@host1 /]# cat /etc/iscsi/initiatorname.iscsi
InitiatorName=iqn.1992-05.com.redhat:497bd66ca0
```

Create new initiator group

An initiator group (igroup) is part of the ONTAP LUN masking architecture. A newly created LUN is not accessible unless a host is first granted access. This step is accomplished by creating an igroup that lists either the FC WWNs or iSCSI initiator names that require access.

In this example, an igroup is created that contains the iSCSI initiator of the Linux host.

```
Cluster01::*> igroup create -igroup linuxiscsi -protocol iscsi -ostype
linux -initiator iqn.1994-05.com.redhat:497bd66ca0
```

Shut down environment

Before changing the LUN protocol, the LUNs must be fully quiesced. Any database on one of the LUNs being converted must be shut down, file systems must be dismounted, and volume groups must be deactivated. Where ASM is used, make sure that the ASM disk group is dismounted and shut down all grid services.

Unmap LUNs from FC network

After the LUNs are fully quiesced, remove the mappings from the original FC igroup.

```
Cluster01::*> lun unmap -vserver vserver1 -path /vol/new_asm/LUN0 -igroup
linuxhost
Cluster01::*> lun unmap -vserver vserver1 -path /vol/new_asm/LUN1 -igroup
linuxhost
...
Cluster01::*> lun unmap -vserver vserver1 -path /vol/new_lvm/LUN8 -igroup
linuxhost
Cluster01::*> lun unmap -vserver vserver1 -path /vol/new_lvm/LUN9 -igroup
linuxhost
```

Remap LUNs to IP network

Grant access to each LUN to the new iSCSI-based initiator group.

```
Cluster01::*> lun map -vserver vserver1 -path /vol/new_asm/LUN0 -igroup
linuxiscsi
Cluster01::*> lun map -vserver vserver1 -path /vol/new_asm/LUN1 -igroup
linuxiscsi
...
Cluster01::*> lun map -vserver vserver1 -path /vol/new_lvm/LUN8 -igroup
linuxiscsi
Cluster01::*> lun map -vserver vserver1 -path /vol/new_lvm/LUN9 -igroup
linuxiscsi
Cluster01::*>
```

Discover iSCSI targets

There are two phases to iSCSI discovery. The first is to discover the targets, which is not the same as discovering a LUN. The `iscsiadm` command shown below probes the portal group specified by the `-p argument` and stores a list of all IP addresses and ports that offer iSCSI services. In this case, there are four IP addresses that have iSCSI services on the default port 3260.

This command can take several minutes to complete if any of the target IP addresses cannot be reached.

```
[root@host1 ~]# iscsiadm -m discovery -t st -p fas8060-iscsi-public1
10.63.147.197:3260,1033 iqn.1992-
08.com.netapp:sn.807615e9ef6111e5a5ae90e2ba5b9464:vs.3
10.63.147.198:3260,1034 iqn.1992-
08.com.netapp:sn.807615e9ef6111e5a5ae90e2ba5b9464:vs.3
172.20.108.203:3260,1030 iqn.1992-
08.com.netapp:sn.807615e9ef6111e5a5ae90e2ba5b9464:vs.3
172.20.108.202:3260,1029 iqn.1992-
08.com.netapp:sn.807615e9ef6111e5a5ae90e2ba5b9464:vs.3
```

Discover iSCSI LUNs

After the iSCSI targets are discovered, restart the iSCSI service to discover the available iSCSI LUNs and build associated devices such as multipath or ASMlib devices.

```
[root@host1 ~]# service iscsi restart
Redirecting to /bin/systemctl restart  iscsi.service
```

Restart environment

Restart the environment by reactivating volume groups, remounting file systems, restarting RAC services, and so on. As a precaution, NetApp recommends that you reboot the server after the conversion process is complete to be certain that all configuration files are correct and all stale devices are removed.

Caution: Before restarting a host, make sure that all entries in `/etc/fstab` that reference migrated SAN resources are commented out. If this step is not taken and there are problems with LUN access, the result can be an OS that does not boot. This issue does not damage data. However, it can be very inconvenient to boot into rescue mode or a similar mode and correct `/etc/fstab` so that the OS can be booted to allow troubleshooting efforts to begin.

Copyright information

Trademark information