Linux® Unified Host Utilities 7.1

Using Linux® Hosts with ONTAP storage

August 2016 | 215-11446_A0
doccomments@netapp.com

NetApp®

Contents

Overview of the supported Linux environments

The Host Utilities support several Linux environments. These environments consist of different versions of the Linux operating system using protocols such as Fibre Channel (FC), FCoE, or iSCSI. Some environments also use Veritas Storage Foundation.

For details about the environments, including the supported operating system versions and Veritas Storage Foundation versions, see the Interoperability Matrix. The Linux Host Utilities documentation is available on the NetApp Support Site.

The following table summarizes key aspects of the main environments:

Linux Environment	Notes
Red Hat Enterprise Linux, Oracle Linux, SUSE Linux Enterprise Server KVM and Xen, Citrix XenServer, Red Hat Enterprise Linux KVM, SUSE Linux Enterprise Server, and Red Hat Enterprise Virtualization Hypervisor	• This environment works with features provided by the Linux operating system. • Multipathing: DM-Multipath • Volume management: Native Linux volume managers • Protocols: FC, FCoE, and iSCSI • ALUA (Asymmetric Logical Unit Access): For systems using clustered Data ONTAP, ALUA is supported with FC, FCoE, and iSCSI. For systems using Data ONTAP operating in 7-Mode, ALUA is supported with FC and FCoE only. • Setup issues: ◦ If you are using HBAs, set them up before you install the Host Utilities. ◦ If you are using multipathing, set up DM-Multipath. ◦ **(Red Hat Enterprise Linux 6 update 2 or later)** If you want to use space reclamation, you must use this version of the operating system. • Configuration issues: ◦ If you are using a Hypervisor Virtual hard Disk (HVD), make sure the LUNs are properly aligned for best performance. **Oracle Linux:** This guide provides instructions and examples using Red Hat Enterprise Linux and SUSE Linux Enterprise Server. In most cases, Oracle Linux uses the same setup procedures as Red Hat Enterprise Linux. To simplify this guide, it uses "Red Hat" to refer to both systems using Red Hat Enterprise Linux and systems using Oracle Linux. If Oracle Linux requires a different procedure, that procedure is included. **Note:** Ensure that the kernel and dm-multipath versions are as per Interoperability Matrix Tool. If not, install the versions from Oracle ULN (Oracle Unbreakable Linux Network). **Note:** The examples in this documentation apply to current Linux distributions. If you have an earlier version of Linux, you must check the documentation for previous versions of the Linux Host Utilities. For example, if you use SUSE Linux Enterprise Server 9, see the *Linux Host Utilities Installation and Setup Guide* for those instructions and examples.

Linux Environment	**Notes**
Veritas Storage Foundation	• This environment works with features provided by Veritas Storage Foundation • Multipathing: Veritas Dynamic Multipathing (VxDMP) • Linux operating systems: Red Hat Enterprise Linux, Oracle Linux, or SUSE Linux Enterprise Server • Volume management: Veritas Volume Manager (VxVM) • Protocols: Fibre Channel (FC) and, with certain versions of Veritas Storage Foundation, iSCSI • ALUA: Supported with Veritas 5.1 and FC • Setup issues: ◦ Set up the HBAs before you install the Host Utilities. ◦ If you have Veritas Storage Foundation 5.0 or earlier, you must install the NetApp Array Support Library (ASL) and the Array Policy Module (APM).

Improving I/O performance on Red Hat Enterprise Linux hosts

You can improve I/O performance on Red Hat Enterprise Linux 6 and 7 by using the `tuned-adm`command in the **`enterprise-storage`** profile. You can use the **`virtual-guest`** profile for Red Hat Enterprise Linux as a guest virtual machine. Using these settings can result in marked improvement in throughput and latency on clustered Data ONTAP.

About this task

You can use the **`enterprise-storage`** profile for configurations where LUNs from enterprise storage arrays are used. It enables a number of performance fine-tuning measures such as switching the I/O scheduler to **`deadline`** from the default **`cfq`**, as well as increasing the read-ahead, setting the CPU governor to performance mode, and so on.

The man page for the `tuned-adm` command contains syntax, default value, and other usage.

Steps

1. Verify that the currently active profile is **`default`**:

   `tuned-adm active`

   ```
   Current active profile: default
   Service tuned: enabled, running
   Service ktune: enabled, running
   ```

2. Set the profile to **`enterprise-storage`**:

   `tuned-adm profile enterprise storage`

   #### Example

   The output changes based on the number of LUNs presented to the host, and might differ from what this example shows:

   ```
   Stopping tuned:                                        [  OK  ]
   Switching to profile 'enterprise-storage'
   Applying deadline elevator: dm-0 dm-1 dm-2 dm-3 sda sdb sdc[  OK  ]
   sdf sdg sdh sdi sdj sdk
   Applying ktune sysctl settings:
   /etc/ktune.d/tunedadm.conf:                                [  OK  ]
   Calling '/etc/ktune.d/tunedadm.sh start':                  [  OK  ]
   Applying sysctl settings from /etc/sysctl.conf
   Starting tuned:                                            [  OK  ]
   ```

3. Verify that the profile was changed:

   `tuned-adm active`

   ```
   Current active profile: enterprise-storage
   Service tuned: enabled, running
   Service ktune: enabled, running
   ```

(iSCSI) How to configure iSCSI for Linux

The iSCSI initiator software comes with the operating system distribution. Its components include a kernel module, which is already compiled into the Linux kernel, and user space packages. For it to work properly, you must perform configuration tasks.

These tasks include the following:

- Record the host's initiator node name.
  You must supply the node name when you set up igroups on the storage system. Once the igroup is created, you map LUNs to it so that all the hosts associated with that igroup can discover the LUNs.
  It is a good practice to change the initiator node name from the default name to one that includes the host name. That way the name is easier to use.
- If you want to use multipathing, you must edit the iSCSI configuration file to set it up.
- If you want to use CHAP, you must edit the iSCSI configuration file to set it up.
- You must set up target discovery so that the host can access LUNs on the storage system.
- Configure the initiator with the IP address for each storage system using either static, ISNS, or dynamic discovery.

The following sections explain how to perform these tasks.

Getting the iSCSI initiator node name

You must supply the iSCSI initiator node name when you set up igroups. To make this easier, you can write down the node name. Before you record the node name, you can change the random string of numbers in the end to something such as the host name which will make the node name easier to use.

About this task

Each node name must be unique. Do not change multiple node names to the same name.

If you decide to change the node name, you can only change the *`RandomNumber`* portion of the name that appears at the end. The first part of a node name starts with `iqn` and uses a standard format that you cannot change. For example, if your node name is `iqn.2005-03.com.RedHat:012345`, you could change the last six numbers to a word or phrase so that the name becomes `iqn.2005-03.com.RedHat:Toaster`.

Steps

1. Use a text editor to open the file containing the node names:

If you are using...	Open the following file...
Red Hat Enterprise Linux 7 series, 6 series, and 5 series, SUSE Linux Enterprise Server 10, 11, or 12 series	`/etc/iscsi/initiatorname.iscsi`
Red Hat Enterprise Linux 4 series	`/etc/initiatorname.iscsi`

2. If you want to change the default name, edit the line in the file containing the node name.

You can only replace the *RandomNumber* portion of the name, and any changes must follow these naming rules:

- A node name can be up to 223 characters.
- Uppercase characters are always mapped to lowercase characters.
- A node name can contain alphabetic characters (a to z), numbers (0 to 9), and the following three special characters (the underscore character (_) is not supported):

  . - :

Note: If the node name does not exist, you can create one by adding a line to the file containing node names. Use the same format shown below for modifying a node name.

If you are using...	Then...
Red Hat Enterprise Linux 7 series, 6 series, and 5 series, Oracle VM, and Red Hat Enterprise Virtualization	Locate the node name (`InitiatorName=iqn.2005-03.com.RedHat:`*RandomNumber*) you want to modify, and then change the *RandomNumber* part of the name to a unique value. For example: **InitiatorName=iqn.2005-03.com.RedHat:linux-host1**
Red Hat Enterprise Linux 4 series	Locate the node name (`InitiatorName=iqn.1987-05.com.cisco:`*RandomNumber*) you want to modify, and then change the *RandomNumber* part of the name to a unique value. For example: **InitiatorName=iqn.1987-05.com.cisco:linux-host1**
SUSE Linux Enterprise Server 10, 11, or 12 series	Locate the node name (`InitiatorName=iqn.1996-04.de.suse:`*RandomNumber*) you want to modify, and then change the *RandomNumber* part of the name to a unique value. For example: **InitiatorName=iqn.1996-04.de.suse:linux-host1**
Citrix XenServer 6 series	The `initiatorname.iscsi` file in `/etc/iscsi` is not present in the earlier version of Citrix XenServer releases. To set the IQN, use the following command: `xe host-param-set uuid=UUID_of_the_host other-config:iscsi_iqn=IQN_In_the_Standard_format.` For example: **xe host-param-set uuid=163c53be-8de5-4035-8770-fbe012ab1f56 other-config:iscsi_iqn=iqn.2013-06.com.example:netapp123**

3. Write down the node name so that you can easily enter it when you configure the storage system.
4. If you modified the file, save it before closing it.

Setting the timeout values to enable multipathing with iSCSI

If you use multipathing, you need to edit the timeout value in the iSCSI configuration file.

Step

1. Edit the following file to provide the correct timeout value for your Host Utilities environment (DM-Multipath or Veritas Storage Foundation):

If you are using...	Enter the following values...
Red Hat Enterprise Linux 7 series, 6 series, or 5 series, SUSE Linux Enterprise Server 10, 11, or 12 series, SUSE Linux Enterprise Server KVM and XEN, and Red Hat Enterprise Linux KVM.	Edit `/etc/iscsi/iscsid.conf` to include the following value: • DM-Multipath environments: **`node.session.timeo.replacement_timeout = 5`** • Veritas Storage Foundation environments: **`node.session.timeo.replacement_timeout = 120`**
Oracle VM 3 series	Edit `/etc/iscsi/iscsid.conf` to include the following value: **`node.session.timeo.replacement_timeout = 5`**
Red Hat Enterprise Linux 4 series	Edit `/etc/iscsi.conf` to include the following DM-Multipath value: **a.** Remove the comment indicator from the `ConnFailTimeout` line in the Session Timeout Settings section. **b.** Set the value of `ConnFailTimeout` to **`5`**
Citrix XenServer 6 series	**For existing Storage Repositories**: **a.** Edit `/etc/iscsi/iscsid.conf` to include the following value for DM-Multipath environments: **`node.session.timeo.replacement_timeout = 5`** **b.** Detach and then reattach SRs from XenCenter or by using the `XenServer xe` CLI. This will reflect the new iSCSI timeout settings for the existing SRs. **For new Storage Repositories**: **a.** Edit `/etc/iscsi/iscsid.conf` to include the following value for DM-Multipath environments: **`node.session.timeo.replacement_timeout = 5`** **b.** Create the new SR. New as well as existing SRs will be updated with the new iSCSI timeout settings.
Red Hat Enterprise Virtualization Hypervisor 6.2 and later	**a.** Edit `/etc/iscsi/iscsid.conf` to include the following value for DM-Multipath environments: **`node.session.timeo.replacement_timeout = 5`** **b.** Make sure that your `iscsid.conf` file is persistent across reboots: **`persist /etc/iscsi/iscsid.conf`**

Setting up CHAP for Red Hat Linux 5, 6, and 7 and SUSE Linux 10, 11, and 12 series for iSCSI

You can use the CHAP protocol on hosts running Red Hat Enterprise Linux 5, 6, and 7 series and SUSE Linux Enterprise Server 10, 11, and 12 series to provide enhanced security. To set up CHAP,

you must add CHAP user names and passwords to the `/etc/iscsi/iscsid.conf` file and then use the `iscsi security` command to set up the same user names and passwords on the storage system.

Steps

1. Open the `/etc/iscsi/iscsid.conf` file with a text editor.
2. Enable CHAP authentication:

   `node.session.auth.authmethod = CHAP`

   The default is **`None`**.
3. Provide a CHAP user name and password for the target to use when authenticating the initiator.

   You must remove the comment indicators and supply values for the options *`username`* and *`password`* in the following configuration entries:

   - `node.session.auth.username = username`
   - `node.session.auth.password = password`
4. Provide a CHAP user name and password for the initiator to use when authenticating the target.

   You must remove the comment indicators and supply values for the options *`username_in`* and *`password_in`* in the following configuration entries:

   - `node.session.auth.username_in = username_in`
   - `node.session.auth.password_in = password_in`
5. For a successful session discovery, enable discovery CHAP authentication by supplying the passwords in the `discovery.sendtargets.auth.` options.

   The user name and password must match for both session and discovery on the host. Make sure that you use the same user names and passwords that you used when you set up CHAP on the storage system with the `iscsi security` command.

   - `discovery.sendtargets.auth.authmethod = CHAP`
   - `discovery.sendtargets.auth.username = username`
   - `discovery.sendtargets.auth.password = password`
   - `discovery.sendtargets.auth.username_in = username_in`
   - `discovery.sendtargets.auth.password_in = password_in`

(iSCSI) Setting up CHAP for Red Hat Enterprise Linux 4 series

You can use the CHAP protocol on hosts running Red Hat Enterprise Linux 4 series to provide enhanced security. To set up CHAP, you need edit the `/etc/iscsi.conf` file to add CHAP user names and passwords. To complete the setup, you must use the `iscsi security` command to set up the same user names and passwords on the storage system.

Steps

1. Open the `/etc/iscsi.conf` file with a text editor.
2. Add CHAP user names and passwords to the storage system's `DiscoveryAddress` section. Use a white space or tab to indent the CHAP settings.

You can set up CHAP as either unidirectional authentication or bidirectional authentication.

- For unidirectional authentication, you should define only the `OutgoingUsername` and `OutgoingPassword`.
  Use the `OutgoingUsername` and `OutgoingPassword` for the storage system's inbound user name and password (*`inname`* and *`inpassword`*).
- For bidirectional authentication, you should define both sets of user names/passwords: outgoing and incoming.
  Use `IncomingUsername` and `IncomingPassword` of the host as the storage system's outbound user name and password (*`outname`* and *`outpassword`*).

Note: Ensure that you use the same user names and passwords when you set up CHAP on the storage system with the `iscsi security` command.

If you want to configure global CHAP—that is, the same user name and password for all the targets—ensure that the CHAP settings are mentioned before the `DiscoveryAddress`.

Example

```
DiscoveryAddress=192.168.10.20
      OutgoingUsername=username_out
      OutgoingPassword=password_out
      IncomingUsername=username_in
      IncomingPassword=password_in
```

3. Configure the storage system as a target by adding the following line for any one iSCSI-enabled interface on each storage system that you used for iSCSI LUNs:

   `DiscoveryAddress=`*`storage_system_IPaddress`*

   `storage_system_IPaddress` is the IP address of an Ethernet interface on the storage system. You should specify an interface that is used for iSCSI communication.

   Example: This example specifies two targets. You now need to edit the sections under the targets to add the user names and passwords.

```
DiscoveryAddress=192.168.10.100
DiscoveryAddress=192.168.10.20
```

Starting the iSCSI service

After you edit the iSCSI configuration file, you must start the iSCSI service so that the host can discover the targets and access LUNs. If the iSCSI service is running, you must restart it.

Step

1. Start the iSCSI service at the Linux host command prompt:

If you are using...	Enter...
Red Hat Enterprise Linux 7 series and SUSE Linux Enterprise 12 series	`systemctl start iscsid`

If you are using...	Enter...
Red Hat Enterprise Linux 6 series **Note:** You must execute this command the first time you start the iSCSI service on a host running Red Hat Enterprise Linux 6 series and Oracle Linux 6 series. If you execute `/etc/init.d/iscsi start` without previously executing `service iscsid force-start`, you get an error message.	`service iscsid force-start`
Red Hat Enterprise Linux 5 or 4 series and Oracle Linux 5 series, Red Hat Enterprise Virtualization, Oracle Linux, and Oracle VM	`/etc/init.d/iscsi start`
SUSE Linux Enterprise Server 10 or 11 series	`/etc/init.d/open-iscsi start`

Citrix discourages the use of the iscsiadm tool. The native XAPI stack accomplishes the tasks of starting and stopping the iscsi service, automatic login on boot, and other iSCSI operations.

Methods for setting up target discovery with software initiators on iSCSI

You need to configure the iSCSI initiator to discover the target so that the host can access LUNs on the target. The method you use to do this depends on your version of the operating system.

- If you are using Red Hat Enterprise Linux 5, 6 and 7 series you can use the `iscsiadm` utility.
- If you are using Red Hat Enterprise Linux 4 series you should modify the `/etc/iscsi.conf` file.
- If you are using SUSE Linux Enterprise Server 10,11, and 12 series you can use either the `iscsiadm` utility or YaST2.

Note: If you are using RHEV, Oracle VM, and Citrix XenServer, you can use the Management GUI for setting up target discovery.

The following sections provide instructions for setting up targets on Red Hat Enterprise Linux 5, 6, and 7 series; Red Hat Enterprise Linux 4 series; and SUSE Linux Enterprise Server 10, 11, and 12 series. If you are using SUSE Linux Enterprise Server 9 series, see Recommended Host Settings for Linux Unified Host Utilities 7.0 at *mysupport.netapp.com*.

Discovering the iSCSI target by using the iscsiadm utility on Red Hat 5, 6, 7 and SUSE 10, 11, and 12

You can use the `iscsiadm` utility to manage (update, delete, insert, and query) the persistent database on Red Hat Enterprise 5, 6, or 7 series and SUSE Linux Enterprise Server 10, 11, or 12

series. The utility enables you to perform a set of operations on iSCSI nodes, sessions, connections, and discovery records.

Steps

1. Discover the iSCSI target:

   iscsiadm --mode discovery --op update --type sendtargets --portal *targetIP*

   targetIP is the IP address of the target.

   The host discovers the target specified by the *targetIP* variable. The `iscsiadm` utility displays each target it discovers on a separate line. It stores the values associated with the target in an internal persistent database.

2. Create all needed devices:

   iscsiadm --mode node -l all

   The initiator logs in to the discovered nodes that are maintained in the iSCSI database.

3. See all of the active iSCSI sessions:

   iscsiadm --mode session

Example

The following sample output shows that 1 is the record ID:

```
tcp: [1] 10.10.10.10:3260,1040 iqn.1992-08.com.netapp:sn.
5d35f5e7ed9711e3ba53123478563412:vs.10 (non-flash)
```

(iSCSI) Setting up target discovery on Red Hat Enterprise Linux 4 series

When you are using Red Hat Enterprise Linux 4 series, you can set up target discovery by editing the `/etc/iscsi.conf` file and adding the IP addresses of the storage systems you want to use as targets.

Steps

1. Open the `/etc/iscsi.conf` file with a text editor.
2. Configure the storage system as a target by adding the following line for any one iSCSI-enabled interface on each storage system that you used for iSCSI LUNs:

   DiscoveryAddress=*storage_system_IPaddress*

   storage_system_IPaddress is the IP address of an Ethernet interface on the storage system. You should specify an interface that is used for iSCSI communication.

Example

The following lines set up the storage systems with the IP addresses 192.168.10.100 and 192.168.10.20 as targets:

```
DiscoveryAddress=192.168.10.100
DiscoveryAddress=192.168.10.20
```

Discovering targets by using YaST2 on SUSE 10, 11, and 12 series on iSCSI

If you are running SUSE Linux Enterprise Server 10, 11, or 12 series you can use YaST2 to discover and configure iSCSI connections. By using YaST2, you can enable the iSCSI initiator at boot time,

add new targets to the system, and discover iSCSI targets in the network. You can also view the iSCSI targets that are currently connected.

About this task

For more information about using YaST2, see the SUSE Linux Enterprise Server 10 series documentation available at the Novell web site.

Steps

1. Enter the following command:

   `yast2`

2. Click **Network Services > iSCSI Initiator > Discovered Targets > Discovery** in the **YaST2** window.
3. If necessary, change the port number to 3260.
4. Enter the IP address of the iSCSI target.
5. If you have an authentication mechanism in place, enter the credentials.
6. Click **Next** to start the discovery.
7. After discovering the targets, use **Login** to activate the target.
8. Enter the authentication credentials required for using the selected iSCSI target.
9. Click **Next** to finish the configuration.

   The target now appears in **Connected Targets**.

10. Click **Connected Targets**.
11. Change the startup option to Manual or Automatic, depending on your requirement, by using the **Toggle Start-Up** button for all the discovered targets.

Related information

Novell web site

Configuring the iSCSI service to start automatically

You can configure the iSCSI service to start automatically at system boot.

Step

1. From the Linux host command prompt, configure the iSCSI service to start automatically:

If you are using...	Enter the following command...
Red Hat Enterprise Linux 7 series and the SUSE Linux Enterprise Server 12 series	`systemctl enable iscsi` `systemctl enable iscsid`
Red Hat Enterprise Linux 6 and 5 series	`chkconfig iscsi on`
SUSE Linux Enterprise Server 10 or 11 series	`chkconfig open-iscsi on`

Configuring manual or automatic node login with iSCSI

When you are running Red Hat Enterprise Linux 7, 6, or 5 series or SUSE Linux Enterprise Server 12, 11, or 10 series, you can specify whether the system automatically logs in to an iSCSI node at startup or whether you must manually log it in to the node.

Before you begin

The iSCSI service must be running when the logins are supposed to occur.

About this task

If you set your login mode to **`manual`**, you must log in to the nodes manually the next time the system starts up. If you set your login mode to **`automatic`**, the system logs in to the nodes automatically when it starts up.

Note: When you are running Red Hat Enterprise Linux 4 series, all sessions are logged in automatically when you start the iSCSI service.

Setting the login mode affects only nodes that are discovered after the value is set.

Step

1. Set the login mode for a specific portal on a target, for all the portals on a target, or for all targets and their ports:

To set the login mode for...	Do the following...
A specific port on a target	Enter the command with the applicable format for your system, including the *`targetname`* and whether the login will be **`manual`** or **`automatic`**: • **`iscsiadm --mode node -T targetname -p ip:port -o update -n node.startup -v manual|automatic`** • **`iscsiadm --mode node -T targetname -p ip:port -o update -n node.conn[0].startup -v manual|automatic`** For more information about the `iscsiadm` options, see the man page.
All the ports on a target	Enter the command with the applicable format for your system, including the *`targetname`* and whether the login will be **`manual`** or **`automatic`**: • **`iscsiadm --mode node -T targetname -o update -n node.startup -v manual|automatic`** • **`iscsiadm --mode node -T targetname -o update -n node.conn[0].startup -v manual|automatic`** For more information about the `iscsiadm` options, see the man page.
All the targets	a. Modify the following line of the `/etc/iscsi/iscsid.conf` file to specify either **`manual`** or **`automatic`**: **`node.startup = manual|automatic`** b. Rediscover the iSCSI target. c. Restart the iSCSI service.

Configuring the storage system

You need to configure the storage system so that the protocol you are using can access it. For example, if your environment uses FC, you must supply the host's WWPN and make sure the storage system is using a supported cfmode. For environments using the iSCSI protocol, you must supply the names of the initiator nodes and, if you are using CHAP, set up the user names and passwords on the storage system.

Before you begin

Ensure that you have the host's WWPNs that you recorded when you installed the FC HBAs or the iSCSI initiator nodes that you recorded.

Note: If you are using Red Hat Enterprise Linux 6.0 or later, NetApp recommends that you run Data ONTAP 8.0.1 or later on the storage systems connected to the host. That way you can use the Block Limits VPD page (0xb0) information. Using the Block Limits VPD page helps maintain block alignment, which leads to enhanced I/O performance on the NetApp LUN.

About this task

This checklist steps you through the requirements for making sure your storage system is correctly set up.

Steps

1. Ensure that the protocol you are using (FC or iSCSI) is licensed and the service is running.
2. **(iSCSI)** If you want to use CHAP authentication, use the `iscsi security` command or the FilerView interface to configure the CHAP user name and password on the storage system.

   Ensure that you use the same user names and passwords that you supplied when you set up CHAP on the host.
3. Create an igroup that includes the Linux host.

   Note: You can use the `sanlun fcp show adapter -c` command to get the information necessary to create an igroup on the controller. The information supplied by this command can only be used to create an igroup if the controller is running Data ONTAP operating in 7-Mode.
4. Create and map the LUNs to the igroup.

   Ensure that you specify the LUN operating system type and igroup type as `linux`.
5. Optionally, enable ALUA.

   For clustered Data ONTAP, ALUA works with the FC, FCoE, and iSCSI protocols. For Data ONTAP operating in 7-Mode, ALUA works with the FC and FCoE protocols. For information about which versions of Linux support ALUA, see the section *Linux configurations that support ALUA*.

   Note: ALUA must be enabled if you are using clustered Data ONTAP.
6. If you are using clustered Linux hosts, ensure that the igroup contains either the WWPNs or the initiator names of all the hosts in the cluster that need access to the mapped LUN.

DM-Multipath configuration

You can configure DM-Multipath for use in multipathing in environments that use native Linux solutions. With DM-Multipath, you can configure multiple I/O paths between a host and storage controllers into a single device. If one path fails, DM-Multipath reroutes I/Os to the remaining paths.

Note: If you are running Veritas Storage Foundation, you need to use VxDMP as your multipathing solution.

When you have multiple paths to a LUN, Linux creates a SCSI device for each path. This means that a single LUN might appear as`/dev/sdd and /dev/sdf` if there are two paths to it. To make it easy to keep track of the LUNs, DM-Multipath creates a single device in `/dev/mapper/` for each LUN that includes all the paths. For example, `/dev/mapper/360a98000433468525634447 17a513571` is the multipath device that is created on top of `/dev/sdd` and `/dev/sdf`.

When you are using DM-Multipath, you should create a file system for each LUN and then mount the LUN using the device in `/dev/mapper/`.

Note: To create a file system on a LUN, use `/dev/mapper/`*`device`* on a Linux host console.

`device` is the multipath device name of the LUN in the `/dev/mpath/` directory.

You also use the DM-Multipath's configuration file to specify whether ALUA is being used and if the hardware handler should be enabled for ALUA.

When DM-Multipath is running, it automatically checks the paths. As a result, if you look at the output of a command such as `lun stats -o`, you see a small amount of FC partner path traffic listed under the operations per second. On average, this is usually about 4 kb of operations per path per LUN every 20 seconds, which is the default time period. This is expected behavior for DM-Multipath.

Verifying the required multipathing packages

It is a good practice to verify that you have the multipathing packages that enable you to use DM-Multipath. These packages are part of your Linux distribution.

Steps

1. Use the `rpm -q` command to display information about the name and version of the DM-Multipath package that you have installed.

If you are using...	Enter the following command...
Red Hat Enterprise Linux	**`rpm -q device-mapper`** **`rpm -q device-mapper-multipath`**
SUSE Linux Enterprise Server	**`rpm -q device-mapper`** **`rpm -q multipath-tools`**

2. If you do not have the required packages, get a copy of your operating system RPM and install the multipathing package.

Related information

Red Hat Web site - http://www.redhat.com/software/rhn/
Novell Web site - http://www.novell.com

Editing the DM-Multipath configuration file

For DM-Multipath to function properly, you must edit the `/etc/multipath.conf` configuration file.

Steps

1. If the `/etc/multipath.conf` file exists, edit it to include the sections needed for your system.
2. If you do not have the `/etc/multipath.conf` file, copy the sample configuration file for your operating system.

   The following sections provide sample configuration files for several versions of Red Hat Enterprise Linux and SUSE Linux Enterprise Server.
3. Specify the devices that you want to exclude (blacklist).

   You should exclude all of the devices that do not correspond to LUNs configured on the storage system that are mapped to your Linux host. That is, you should exclude the devices that are not displayed by the `sanlun lun show` command.

   Note: You **must** supply information that is specific to your system. Otherwise, you will encounter problems.

   a. In the `blacklist` section of the configuration file, enter the WWID of all non-NetApp SCSI devices installed on your host.

      You can get the WWID by running the `scsi_id` command on a device.

      #### Example

      For example, assume that `/dev/sda` is a local SCSI drive. To obtain the WWID on systems running Red Hat Enterprise Linux 7 or 6 series or SUSE Linux Enterprise Server 12 and 11, enter **`/lib/udev/scsi_id -gud /dev/`*`sda`***.

      To obtain the WWID on systems running other Linux operating systems, enter **`scsi_id -gus /block/sda`**.

      In both cases, the output looks similar to the following:

      `SIBM-ESXSMAW3073NC_FDAR9P66067W`

      To exclude that device, enter `SIBM-ESXSMAW3073NC_FDAR9P66067W` in the `blacklist` section of the configuration file:

      ```
      blacklist
      {
        wwid IBM-ESXSMAW3073NC_FDAR9P66067W
        devnode "^hd[a-z]"
        devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
        devnode "^cciss.*"
      }
      ```

   b. Exclude other devices by using the devnode:

      `devnode "^hd[a-z]"`

      `devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"`

      `devnode "^cciss.*"`

Example

On Red Hat Enterprise Linux 4 hosts, the `blacklist` section might appear as the following:

```
devnode_blacklist
{
devnode "^hd[a-z]"
devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
devnode ^cciss.*"
wwid SIBM-ESXSMAW3073NC_FDAR9P66067WJ
}
```

4. Make sure that you use the correct settings based on whether you are using ALUA.

   If you are using ALUA, you must specify the ALUA callout program. If you are not using ALUA, you must specify the Data ONTAP callout program. The following table provides information about the values that you must supply.

   Note: If you are using clustered Data ONTAP, you must have ALUA enabled.

If you are running...	Without ALUA, use the value	Without ALUA, use the value
SUSE Linux Enterprise Server 10 SP2 and later, SUSE Linux Enterprise Server 11, 12, and later, or Red Hat Enterprise Linux 6.0 and later	Set `prio` to: **`"alua"`**	Set `prio` to: **`"ontap"`**
Any other Red Hat Enterprise Linux operating system	Set `prio_callout` to: **`/sbin/mpath_prio_alua`**	Set `prio_callout` to: **`/sbin/mpath_prio_ontap`**
SUSE Linux Enterprise Server 10 SP1 and SP2	~	Set `prio_callout` to: **`/sbin/mpath_prio_ontap`**
All supported Linux operating systems that support ALUA	Set `hardware_handler` to: **`"1 alua"`**	Set `hardware_handler` to: **`"0"`**

   Note: ALUA is supported in Red Hat Enterprise Linux 5 Update 1 or later, SUSE Linux Enterprise Server 10 SP2 or later, and SUSE Linux Enterprise Server 11 or later.

5. Save the changes.

Starting DM-Multipath

You can start DM-Multipath manually to configure LUNs to work with it.

Steps

1. To start the multipath daemon, enter the `start` command.

If you are using...	Enter the following command...
Red Hat Enterprise Linux 7 series	**`# systemctl start multipathd`**

If you are using...	Enter the following command...
Red Hat Enterprise Linux 6 and 5 series, Red Hat Enterprise Virtualization, and Oracle VM	`# /etc/init.d/multipathd start`
SUSE Linux Enterprise Server	`# /etc/init.d/boot.multipath start` `# /etc/init.d/multipathd start`

2. To configure the DM-Multipath devices, run the following command:

   ```
   # multipath
   ```

3. Perform the following steps to enable multipath using Xen center or Xen CLI:

   a. Enter maintenance mode:

      ```
      xe host-disable uuid=<host UUID>
      ```

   b. Enable multipath:

      ```
      xe host-param-set other-config:multipathing=true uuid=host_uuid
      ```

   c. xe host-param-set other-config:

      ```
      multipathhandle=dmp uuid=host_uuid
      ```

   d. Release maintenance mode:

      ```
      # xe host-enable uuid=<host UUID>
      ```

Configuring DM-Multipath to start automatically while booting

You can configure DM-Multipath to start automatically while booting. Depending on your version of Linux, you must enter different commands to complete the configuration.

Step

1. Add the multipath service to the boot sequence on the Linux host console:

If you are using...	Enter the following commands...
Red Hat Enterprise Linux 7 series and SUSE Linux Enterprise Server 12 series	`systemctl start multipathd` `systemctl enable multipathd`

Red Hat Enterprise Linux 6 series and 5 series	`chkconfig --add multipathd`
	`chkconfig multipathd on`
	Note: You should reboot the host if you are configuring a SAN boot LUN on the host.

SUSE Linux Enterprise Server 11 and 10 series	`chkconfig --add boot.multipath`
	`chkconfig --add multipathd`
	`chkconfig boot.multipath on`
	`chkconfig multipathd on`

Note: Hypervisors Oracle VM, Red Hat Enterprise Virtualization, and Citrix XenServer management stack ensure automatic start of multipath service.

Verifying the DM-Multipath configuration

You can use the `multipath` command on the Linux host to view and verify the DM-Multipath configuration. You can change the amount of configuration detail that is displayed by using the `-v` parameter.

Steps

1. To view the DM-Multipath configuration, enter

   `multipath -v3 -d -ll`

   The `-d` (dry run) parameter prevents the command from updating the multipath maps.

2. Verify that the multipathd is running:

   - For Red Hat Enterprise Linux 6 and 5 series, use the `/etc/init.d/multipathd status` command.
   - For Red Hat Enterprise Linux 7 series, use the `systemctl status multipathd` command.

3. To determine whether multipathd is working correctly on your system, enter the `multipathd show config` command. This command displays the values currently being used for the `multipath.conf` file. You can then confirm that multipathd is using the values you specified.

4. View a list of the multipath devices, including which `/dev/sd`*x* devices are used:

   `multipath -ll`

   The `multipath -ll` command output varies slightly across Linux versions, as shown in the following examples.

Example

Red Hat Enterprise Linux 5.11:

```
# multipath -ll
3600a0980417577656624444342565 57a dm-11 NETAPP,LUN C-Mode
[size=10G][features=3 queue_if_no_path pg_init_retries 50]
[hwhandler=1 alua][rw]
\_ round-robin 0 [prio=50][active]
 \_ 7:0:6:3  sdco 69:192 [active][ready]
 \_ 7:0:1:3  sdbu 68:128 [active][ready]
 \_ 8:0:14:3 sdbi 67:192 [active][ready]
 \_ 8:0:13:3 sdbe 67:128 [active][ready]
\_ round-robin 0 [prio=10][enabled]
 \_ 8:0:7:3  sdag 66:0   [active][ready]
 \_ 8:0:15:3 sdbm 68:0   [active][ready]
 \_ 7:0:3:3  sdcc 69:0   [active][ready]
 \_ 7:0:15:3 sddy 128:0  [active][ready]
```

Example

Red Hat Enterprise Linux 6.6:

```
# multipath -ll
3600a0980383034586b3f4644694d6a51 dm-14 NETAPP,LUN C-Mode
size=5.0G features='4 queue_if_no_path pg_init_retries 50
retain_attached_hw_handle' hwhandler='1 alua' wp=rw
|-+- policy='round-robin 0' prio=50 status=active
| |- 0:0:10:11 sdxq  128:512  active ready running
| |- 1:0:5:11  sdxn  71:720   active ready running
| |- 0:0:2:11  sdcq  69:224   active ready running
```

```
| `- 1:0:8:11  sdafa 68:768   active ready running
`-+- policy='round-robin 0' prio=10 status=enabled
  |- 0:0:8:11  sdrg  133:416  active ready running
  |- 1:0:10:11 sdaie 129:800  active ready running
  |- 0:0:11:11 sdaau 133:544  active ready running
  `- 1:0:11:11 sdajt 131:944  active ready running
```

Example

FC on Red Hat Enterprise Linux 7.0:

```
# multipath -ll
3600a09804d542d4d71244635712f4a4a dm-20 NETAPP  ,LUN C-Mode
size=5.0G 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
| |- 4:0:3:21 sdeo 129:0   active ready running
| |- 4:0:7:21 sdnj 71:336  active ready running
| |- 5:0:5:21 sdui 66:672  active ready running
| `- 5:0:7:21 sdxm 71:704  active ready running
`-+- policy='service-time 0' prio=10 status=enabled
  |- 4:0:5:21 sdhs 134:32  active ready running
  |- 4:0:6:21 sdks 67:256  active ready running
  |- 5:0:4:21 sdst 8:528   active ready running
  `- 5:0:6:21 sdvx 69:560  active ready running
```

Example

SLES11 SP3:

```
# multipath -ll
3600a09804d542d4d71244635712f4a4a dm-20 NETAPP  ,LUN C-Mode
size=5.0G 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
| |- 4:0:3:21 sdeo 129:0   active ready running
| |- 4:0:7:21 sdnj 71:336  active ready running
| |- 5:0:5:21 sdui 66:672  active ready running
| `- 5:0:7:21 sdxm 71:704  active ready running
`-+- policy='service-time 0' prio=10 status=enabled
  |- 4:0:5:21 sdhs 134:32  active ready running
  |- 4:0:6:21 sdks 67:256  active ready running
  |- 5:0:4:21 sdst 8:528   active ready running
  `- 5:0:6:21 sdvx 69:560  active ready running
```

Example

FC on Red Hat Enterprise Virtualization Hypervisor 6.2 with ALUA enabled:

```
# multipath -ll
360a98000316b5a776b3f2d7035505a6f dm-0 NETAPP,LUN
size=60G features='3 queue_if_no_path pg_init_retries 50'
hwhandler='1 alua' wp=rw
|-+- policy='round-robin 0' prio=50 status=active
| |- 4:0:0:0 sda 8:0   active ready running
| `- 5:0:0:0 sdc 8:32  active ready running
`-+- policy='round-robin 0' prio=10 status=enabled
  |- 4:0:1:0 sdb 8:16  active ready running
  `- 5:0:1:0 sdd 8:48  active ready running
```

5. You can verify that the NetApp recommended DM-Multipath settings are currently in use by entering the following commands:

- For RHEL6 and 7 hosts, use the `multipathd show config` command.
  - For RHEL5 hosts, use the `multipathd -k"show config"` command.
- For SLES11 hosts, use the `multipathd show config` command.

For details about the commands, see the man pages.

6. If you want to view the `/dev/mapper` devices:

   `ls -l /dev/mapper/`

   Example

   The following example shows the output of the `ls -l dev/mapper` command:

```
total 0
brw------- 1 root root 253, 1 Sep 20 17:09
360a98000486e5363693444646a2f656c
brw------- 1 root root 253, 0 Sep 20 17:09
360a98000486e5372635a44646a505643
lrwxrwxrwx 1 root root     16 Sep 12 10:16 control -> ../device-mapper
```

Stopping DM-Multipath

When you want to stop DM-Multipath on the Linux host, you should stop the affected services.

Steps

1. Unmount all the file systems on the LUNs.
2. Flush the DM-Multipath devices:

   `multipath -F`

3. Stop the multipath daemon:

If you are using...	Enter the following command...
Red Hat Enterprise Linux 7 series and SUSE Linux Enterprise Server 12 series	**`systemctl stop multipathd`**
Red Hat Enterprise Linux 6 series and 5 series	**`/etc/init.d/multipathd stop`**
SUSE Linux Enterprise Server 11 and 10 series	**`/etc/init.d/multipathd stop`** and **`/etc/init.d/boot.multipath stop`**

Veritas Dynamic Multipath configuration

When you are using Veritas Dynamic Multipathing (VxDMP), you must perform some configuration tasks to enable multipathing.

There are several tunable parameters that you might need to set, depending on your version of Linux. These parameters might include:

- `vxdmpadm start restore interval`
- `vxdmpadm settune dmp_lun_retry_timeout`
- `vxdmpadm settune dmp_path_age`

In addition, if you are running either Red Hat Enterprise Linux 6 series, Red Hat Enterprise Linux 7 series, or SUSE Linux Enterprise Server 11 series, you must modify the `/etc/udev/rules.d/40-rport.rules` file.

You must have the Array Support Library (ASL) and the Array Policy Module (APM) that Symantec provides for NetApp storage systems installed. The amount of work you must do to set up the ASL and APM depends on your version of Veritas Storage Foundation.

Related concepts

(Veritas) VxDMP restore daemon and LUN retries tunable configuration on page 28

(Veritas) VxDMP restore daemon and LUN retries tunable configuration

It is a good practice to set the Veritas restore daemon values for the restore policy and the polling interval and the VxDMP tunable `dmp_lun_retry_timeout` to the Host Utilities recommended values.

The restore policy and the polling interval settings determine how frequently the Veritas daemon checks paths between the host and the storage system. At the time this document was produced, the Host Utilities recommended the following restore daemon settings:

- A restore policy of "disabled"
- A polling interval of "60"

Check the *Release Notes* to see if these recommendations have changed.

The tunable `dmp_lun_retry_timeout` tells the VxDMP to continue retrying I/O requests to a LUN where all the paths to the disk have failed. Setting this value provides for more robust handling of temporary path failures that are quickly restored. The recommended value for this tunable is 300, which means that the VxDMP continues to retry paths to the LUN until either the I/O succeeds or 300 seconds have elapsed.

Setting the Veritas restore daemon and LUN retry editable values

To access LUNs by using VxDMP, you should configure the restore daemon and then verify the configuration. If you have Veritas Storage Foundation 5.1, you should also set the VxDMP tunable `dmp_lun_retry_timeout` command.

Steps

1. Set the value for the VxDMP restore daemon to an interval of 60:

`vxdmptune dmp_restore_interval=60`

On reboot, this value takes effect and remains persistent. Alternatively, you can use the following commands to configure the daemon value:

`vxdmpadm stop restore`

`vxdmpadm start restore interval=60`

This value takes effect immediately; however, it is not persistent on reboot. You must reset the value each time you reboot the system.

2. Verify that the daemon value has been successfully configured:

   `vxdmpadm stat restored`

   For details about configuring Veritas Volume Manager, see the *Veritas Volume Manager Administrator's Guide* for Linux that is shipped along with the software.

3. Set the value of the `dmp_lun_retry_timeout` to the appropriate interval:

   - If you are using Veritas Storage Foundation 5.1 up to 6.0), set the value to **`300`**:

     `vxdmpadm settune dmp_lun_retry_timeout=300`
     The editable value changes immediately.

   - If you are using Veritas Storage Foundation 6 series and InfoScale 7 series, set the value to **`60`**
     The editable value changes immediately.

4. For Veritas Storage Foundation 5.1 SP1 and later, and InfoScale 7.0 series, set the value of the `dmp_path_age` to an interval of **`120`**:

   `vxdmpadm settune dmp_path_age=120`

   The editable value changes immediately.

5. Verify that the value has been successfully configured:

   `vxdmpadm gettune`

   This command displays the current values for all the Veritas DMP editables.

Configuring Red Hat Enterprise Linux 6 series and Red Hat Enterprise Linux 7 series to support Veritas Storage Foundation

If you are using Red Hat Enterprise Linux 6 series or Red Hat Enterprise Linux 7 series, you must configure it to support Veritas Storage Foundation.

About this task

If you are running Red Hat Enterprise Linux 6 series or Red Hat Enterprise Linux 7 series, you must perform some additional steps and ensure that the value of the `IOFENCE timeout` parameter is correct.

Steps

1. Log in to the host as the root user.

2. Create the file `/etc/udev/rules.d/40-rport.rules` with the following content line:

   ```
   KERNEL=="rport-*", SUBSYSTEM=="fc_remote_ports", ACTION=="add", RUN
   +="/bin/sh -c 'echo 20 > /sys/class/fc_remote_ports/%k/
   fast_io_fail_tmo;echo 864000 > /sys/class/fc_remote_ports/%k/
   dev_loss_tmo'"
   ```

3. Check the value of the `IOFENCE timeout` parameter and ensure that it is set to `30000`.

   The `IOFENCE timeout` parameter specifies the amount of time in milliseconds that it takes clients to respond to an IOFENCE message before the system halts. When clients receive an IOFENCE message, they must unregister from the GAB driver within the number of milliseconds specified by the `IOFENCE timeout` parameter, or the system halts. The default value for this parameter is `15000` milliseconds or 15 seconds.

   Example

   To check the value of this parameter, enter the command `gabconfig -l` on the host. The following is an example of the type of output this command produces:

```
# gabconfig -l
GAB Driver Configuration
Driver state         : Configured
Partition arbitration: Disabled
Control port seed    : Enabled
Halt on process death: Disabled
Missed heartbeat halt: Disabled
Halt on rejoin       : Disabled
Keep on killing      : Disabled
Quorum flag          : Disabled
Restart              : Enabled
Node count           : 2
Send queue limit     : 128
Recv queue limit     : 128
IOFENCE timeout (ms) : 15000
Stable timeout (ms)  : 5000
```

4. If the value of the `IOFENCE timeout` parameter is not `30000`, correct it:

   `gabconfig -f 30000`

   This value is not persistent across reboots, so you must check it each time you boot the host and reset it if necessary.

Configuring SUSE Linux Enterprise Server 11 to support Veritas Storage Foundation

If you are using SUSE Linux Enterprise Server 11 or later, you must configure it to support Veritas Storage Foundation.

Before you begin

You must have reviewed Symantec TechNote 124725 for the latest information about using SUSE Linux Enterprise Server 11 with Veritas Storage Foundation.

Steps

1. Log in to the host as the root user.
2. Install SUSE Linux Enterprise Server 11 with kernel version 2.6.27.45-0.1.1 or later from Novell.
3. Create the file `/etc/udev/rules.d/40-rport.rules` with the following content line:

   `KERNEL=="rport-*", SUBSYSTEM=="fc_remote_ports", ACTION=="add",RUN +="/bin/sh -c 'echo 20 > /sys/class/fc_remote_ports/%k/ fast_io_fail_tmo;echo 864000 >/sys/class/fc_remote_ports/%k/ dev_loss_tmo'"`

4. Reboot the host.

Related information

Novell Web site - http://www.novell.com

Symantec TechNote for setting up SUSE Linux Enterprise Server 11 - http://www.symantec.com/business/support/index?page=content&id=TECH124725

The Veritas Array Support Library and Array Policy Module

Symantec provides both the Array Support Library (ASL) and the Array Policy Module (APM) for NetApp storage systems. These are Symantec products; therefore, Symantec provides technical support if you encounter a problem using them.

To determine which version of the ASL and APM you need for this version of the Host Utilities, check the NetApp Interoperability Matrix. After you know which version you need, go to the Symantec Web site and download the ASL and APM.

Note: Starting with Veritas Storage Foundation 5.1 and later, InfoScale 7.0 series, ALUA is supported on FC.

(Veritas) What the ASL is

The ASL is a Data ONTAP-qualified library that provides information about storage array attributes and multipathing configurations to the Device Discovery Layer (DDL) and Veritas Dynamic Multipathing (DMP) components of Veritas Volume Manager (VxVM).

The ASL provides enclosure-based naming, where the name of the disk is based on the logical name of its enclosure, disk array, or a Storage Virtual Machine (SVM, formerly known as Vserver). The ASL provides specific vendor and model information to DMP and VxVM, instead of referring to them as JBOD or raw devices.

Note: You cannot use storage systems simultaneously as JBOD and vendor arrays. If you install the ASL, storage systems cannot be configured in VxVM as JBOD. They are reported as storage arrays, unless you explicitly exclude them by using the `vxddladm exclude array` command.

(Veritas) ASL array type

The ASL reports information about the multipathing configuration to the DDL as an Active/Active (A/A), ALUA, or an Active/Passive Concurrent (A/P-C) disk array type.

- Active/Active (A/A)
  There are multiple active paths to a storage system, and simultaneous I/O is supported on each path. If a path fails, I/O is distributed across the remaining paths.
- Active/Passive Concurrent (A/P-C)
  An A/P-C array is a variant of the A/P array type that supports concurrent I/O and load balancing by having multiple primary paths to LUNs. Failover to the secondary (passive) path occurs only if all the active primary paths fail.
- ALUA
  A LUN in an ALUA-enabled array can be accessed through both controllers, by using optimized and non-optimized paths. The array notifies the host of path options, their current state, and state changes. Using this information, the host can determine which paths are optimized. Failover to the non-optimized path occurs only if all the optimized paths fail.

For more information about system management, see the *Veritas Volume Manager Administrator's Guide*.

(Veritas) What the APM is

The APM is a kernel module that defines I/O error handling, failover path selection, and other failover behavior for a specific array.

The NetApp APM is customized to optimize I/O error handling and failover path selection for the NetApp environment. After the ASL discovers the storage array as a NetApp array, the ASL instructs VxDMP to use the NetApp APM to handle I/O error processing and path failures for the NetApp storage array.

Installing the ASL and APM software to support Veritas Storage Foundation

If you are using Veritas Storage Foundation for multipathing, you should install and configure the Symantec Array Support Library (ASL) and Array Policy Module (APM) for storage systems.

Before you begin

- You must have verified that your configuration meets the system requirements.
  For more information, see the NetApp Interoperability Matrix.
- You must have downloaded the ASL and APM software.
  You can obtain the ASL and APM from the Symantec Web site. For a direct link to the ASL and APM on the Symantec Web site, see the NetApp Interoperability Matrix.

Note: In Veritas Storage Foundation 5.1 series and later, InfoScale 7.0 series, the NetApp ASL and APM are included in the Veritas Storage Foundation product.

About this task

Only one version of the NetApp ASL and APM package can be installed on the host at any given time.

Steps

1. Log in to the Linux host.
2. If you already have the NetApp storage configured as JBOD in your VxVM configuration, remove the JBOD support for: NetApp

   `# vxddladm rmjbod vid=NETAPP`
3. Install the ASL and APM according to the instructions provided by Symantec:
4. If your host is connected to a NetApp storage system, verify the installation by following these steps:
   a. Run the following command:

      `# vxdmpadm listenclosure all`

      The output shows the model name of the storage device if you are using enclosure-based naming with VxVM.

Example

The `vxdmpadm listenclosure all` command shows the Enclosure Type as FAS3170 in this example:

```
# vxdmpadm listenclosure all
ENCLR_NAME ENCLR_TYPE ENCLR_SNO STATUS     ARRAY_TYPE  LUN_COUNT
============================================================
disk       Disk       DISKS     CONNECTED  Disk         1
fas31700   FAS3170    80010081  CONNECTED  A/A-NETAPP  15
fas31701   FAS3170    80010082  CONNECTED  A/A-NETAPP  15
```

(Veritas) Removing the ASL and APM

If you do not require the ASL and APM, you can remove them from the host.

About this task

You do not need to stop any volumes created on unaffected disk arrays, such as disk arrays from other vendors, before removing ASL and APM support. This is also true for arrays or disks in the OTHER_DISKS category. The OTHER_DISKS category includes local non-FC attached disks. Volumes created on these arrays remain accessible because they do not have multipath support.

Steps

1. Log in to the host as the root user.
2. Stop all I/O to LUNs configured on storage controllers.

   Note: In a Storage Foundation RAC cluster, you should also stop clustering on a node before you remove the ASL and APM.

3. Use the `rpm` command to remove the ASL package. This command has the format: `rpm -ev` *`asl_rpm_name`*

   ### Example

   The following command line removes a previous version of the ASL:

If you are using...	Enter the following command...
Red Hat Enterprise Linux	**`rpm -ev VRTSNTAPasl-5.0-3.0_RHEL5`**
SUSE Linux Enterprise Server	**`rpm -ev VRTSNTAPasl-5.0-3.0_SLES10`**

4. Use the `rpm` command to remove the APM package. This command has the format: `rpm -ev` *`apm_rpm_name`*.

   ### Example

   The following command line removes a previous version of the APM:

If you are using...	Enter the following command...
Red Hat Enterprise Linux	**`rpm -ev VRTSNTAPapm-5.0-1.0_RHEL5`**
SUSE Linux Enterprise Server	**`rpm -ev VRTSNTAPapm-5.0-1.0_SLES10`**

(Veritas) Information about ASL error messages

The ASL works silently and seamlessly with the VxVM DDL. If an error, misconfiguration, or malfunction occurs, messages from the library are logged to the console by using the host's logging facility.

The following table lists the importance and severity of these messages:

Message severity	**Definition**	**Action required**
Error	Indicates that an ERROR status is being returned from the ASL to the VxVM DDL that prevents the device (LUN) from being used. The device might still appear in the vxdisk list, but it is not usable.	Call Symantec technical support for help.
Warning	Indicates that an UNCLAIMED status is being returned. Unless claimed by a subsequent ASL, dynamic multipathing is disabled. No error is being returned, but the device (LUN) might not function as expected.	Call Symantec technical support for help.
Info	Indicates that a CLAIMED status is being returned. The device functions fully, with VxDMP enabled, but the results seen by the user might be other than what is expected. For example, the enclosure name might change.	Call Symantec technical support for help.

Methods for working with LUNs in native Linux environments

The method you use when working with LUNs often varies depending on your environment--whether you are using multipathing, which protocol you are using, whether you have an HBA, hardware iSCSI initiator, or a software iSCSI initiator, and whether you are using Veritas Storage Foundation. In some cases, the method also varies depending on your version of the Linux operating system.

Note: If you are using Veritas Storage Foundation, see the section on accessing LUNs with VxDMP. The sections here focus on working with LUNs in a native Linux environment.

The sections that follow provide information about the tools you need to use to work LUNs as well as what actions you should take when working with LUNs in your environment. For example, if you do not have multipathing enabled, it is a good practice to provide persistent identification for the LUNs. Or, if you are using the iSCSI software initiator, you can use either the `sanlun` or `iscsiadm` command to view LUNs.

As you work with LUNs, remember that the host cannot distinguish multiple LUNs from multiple paths to the same LUN without multipathing software. As a result:

- If you have more than one path from the host to a LUN, you should use DM-Multipath.
- If you are not using multipathing software, you should limit each LUN to a single path.

For information about the supported configurations for DM-Multipath, see the NetApp Interoperability Matrix.

Discovering new LUNs in FC and hardware iSCSI environments

When you are using an FC or hardware iSCSI environment, you can use the `rescan` script to discover the LUNs you have created and mapped to the Linux host.

Before you begin

You must have a copy of the `rescan` script. The `rescan` script is available with the sg3_utils package. In addition, the `rescan` script is available with Red Hat Enterprise Linux 5 Update 4 or later; Red Hat Enterprise Linux 6 and 7 series; SUSE Linux Enterprise Server 10 SP2 or later; and SUSE Linux Enterprise Server 11 and 12 series. For earlier versions, you can use the vendor-specific `rescan` scripts, which are available on their web sites. See the HBA vendor-specific documentation.

Step

1. Enter one of the following commands:

If you are using Red Hat Linux Enterprise	Enter...
5 and 6 series, up to 6.5	**`/usr/bin/rescan-scsi-bus.sh`**
6.6 and later and 7 series	**`/usr/bin/rescan-scsi-bus.sh -a`**

Discovering new LUNs on Red Hat and SUSE with iSCSI and multipathing

When you are running Red Hat Enterprise Linux 5, 6, and 7 series or SUSE Linux Enterprise Server 10, 11, or 12 series with DM-Multipath and the software iSCSI initiator, you can discover new LUNs by rescanning the iSCSI service on the host. Rescanning the service displays all the newly created LUNs that have been mapped to the host.

Before you begin

You must have a copy of the `rescan` script. The `rescan` script is available with the sg3_utils package. In addition, the `rescan` script is available with Red Hat Enterprise Linux 5 Update 4 or later, Red Hat Enterprise Linux 6, 7 series, SUSE Linux Enterprise Server 10 SP2 or later, SUSE Linux Enterprise Server 11, 12 series.

About this task

You cannot view new LUNs until the operating system discovers them. After the LUNs have been discovered, they are automatically added to the DM-Multipath configuration.

Note: You can use the `rescan` script to discover new LUNs in environments that are using multipathing and those that are not.

Steps

1. To discover a new LUN on a system running DM-Multipath, enter one of the following commands:

If you want to...	Enter the following command...
Obtain the list of all the current sessions	**`iscsiadm -m session`**
Rescan a specific session	**`iscsiadm -m session --sid=N --rescan`** *N* is the specific session ID.
Rescan all the sessions	**`iscsiadm -m session --rescan`**
Rescan using the SCSI `rescan` script	**`/usr/bin/rescan-scsi-bus.sh`**

2. To verify that the new LUNs have been discovered, use the `sanlun` command or the `iscsiadm` command.

Discovering new LUNs on Red Hat Enterprise Linux 4 with DM-Multipath and software iSCSI

When you are running Red Hat Enterprise Linux 4 series with DM-Multipath and the software iSCSI initiator, you can discover new LUNs by reloading the iSCSI service on the host. Reloading the service displays all the newly created LUNs that have been mapped to the host.

About this task

You cannot view new LUNs until the operating system discovers them. After the LUNs have been discovered, they are automatically added to the DM-Multipath configuration.

Steps

1. On the Linux host, reload the iSCSI service:

   ```
   /etc/init.d/iscsi reload
   ```

2. Use the `sanlun` or `iscsi-ls` command to verify that the new LUNs have been discovered.

Viewing a list of LUNs

Whether your environment uses multipathing or not, you have several options for viewing LUNs that are mapped to the host. The `sanlun lun show all` command works for all environments and protocols. If you are using an iSCSI software initiator, you also have a choice between using the `iscsiadm` command or the `iscsi` command, depending on your version of Linux.

Step

1. To view the list of LUNs mapped to your host, run the appropriate command for your system environment.

   The following table summarizes the commands and the environments that support them. For more information on the commands and their options, see the man pages.

   Note: You can use the `sanlun` command to display LUN information and the `iscsiadm` command to view iSCSI information.

If you are using...	Enter one of the following commands...
All environments	**`sanlun lun show all`**
(iSCSI software initiator) Red Hat Enterprise Linux 5 or 6 series	**`iscsiadm --mode session --sid=N -P 3`**
(iSCSI software initiator) Red Hat Enterprise Linux 4 series	**`iscsi-ls -l`**
(iSCSI software initiator) SUSE Linux Enterprise Server 10 or 11 series	• **(SUSE Linux Enterprise Server 10 SP2):** **`iscsiadm --mode session --sid=N -P 3`** • **(SUSE Linux Enterprise Server 10 SP1):** **`iscsiadm --mode session --sid=N -i`**

   The sections that follow contain examples of the type of output these commands produce with different protocols and operating systems.

Examples of sanlun, iscsiadm, iscsi output when used to view LUNs

You can use either the `sanlun` command, the `iscsiadm` command, or the `iscsi` command to view the LUNs configured on your Linux host. The examples in this section show the type of output you would see if you run one of these commands on your Linux operating system in an environment with DM-Multipath enabled or one with it disabled.

The tool you use depends on your version of Linux and what you want to view as well as whether you have DM-Multipath enabled. The `sanlun` command displays the host device names and the LUNs to which they are mapped. The `iscsiadm` command lists the available storage systems and LUNs. The `iscsi-ls -l` command lists storage system node names, IP addresses, and available LUNs.

The following sections provide examples of the type of output you see if you run one of these commands in a specific environment; for example with iSCSI and DM-Multipath on Red Hat Enterprise Linux 5 series.

- FC with DM-Multipath running `sanlun`
- iSCSI with multipathing running `sanlun`
- (Red Hat Linux) Software iSCSI with DM-Multipath running `iscsiadm`
- (Red Hat 5, 6) Software iSCSI without multipathing running `iscsiadm`
- (SUSE Linux 10, 11) Software iSCSI with DM-Multipath running `iscsiadm`
- (SUSE Linux 10, 11) Software iSCSI without multipathing running `iscsiadm`
- (Red Hat 4) Software iSCSI without multipathing running `iscsi-ls`

Example of using sanlun to view LUNs running FC with DM-Multipath

This example shows sample output from the `sanlun lun show all` command when it is issued in a Host Utilities environment that is running the FC protocol with DM-Multipath on a storage system running clustered Data ONTAP.

```
# sanlun lun show all
controller(7mode)/             device          host                  lun
vserver(Cmode) lun-pathname    filename        adapter    protocol   size    mode
--------------------------------------------------------------------------------
vs_data28_2    /vol/vol1/lun1     /dev/sdgf       host1      FCP        3g      C
vs_data28_2    /vol/vol2/lun2     /dev/sdge       host1      FCP        3g      C
vs_data28_2    /vol/vol3/lun3     /dev/sdgd       host1      FCP        3g      C
```

This example shows sample output from the `sanlun lun show all` command when it is issued in a Host Utilities environment that is running the FC protocol with DM-Multipath on a storage system running Data ONTAP operating in 7-Mode.

```
# sanlun lun show all
controller(7mode)/               device      host                  lun
vserver(Cmode)  lun-pathname     filename    adapter    protocol   size    mode
--------------------------------------------------------------------------------
fas3040-201-25   /vol/vol1/lun1   /dev/sdu    host1      FCP        5g      7
fas3040-201-25   /vol/vol2/lun2   /dev/sdt    host1      FCP        5g      7
fas3040-201-25   /vol/vol3/lun3   /dev/sds    host1      FCP        5g      7
```

Example of using sanlun to view LUNs running iSCSI with DM-Multipath

This example shows sample output from the `sanlun lun show all` command when it is issued in a Host Utilities environment that is running the iSCSI protocol with DM-Multipath on a storage system running clustered Data ONTAP.

```
# sanlun lun show all
controller(7mode)/             device       host                  lun
vserver(Cmode) lun-pathname    filename     adapter    protocol   size    mode
--------------------------------------------------------------------------------
vs_data78_1    /vol/vol1/lun1     /dev/sdcx     host29     iSCSI      3g      C
vs_data78_0    /vol/vol2/lun2     /dev/sdcw     host20     iSCSI      3g      C
vs_data79_1    /vol/vol3/lun3     /dev/sdck     host14     iSCSI      3g      C
```

This example shows sample output from the `sanlun lun show all` command when it is issued in a Host Utilities environment that is running the iSCSI protocol with DM-Multipath on a storage system running Data ONTAP operating in 7-Mode.

```
# sanlun lun show all
controller(7mode)/              device      host                  lun
vserver(Cmode)    lun-pathname  filename    adapter    protocol   size    mode
--------------------------------------------------------------------------
fas3040-201-24   /vol/vol1/lun1  /dev/sdb    host6      iSCSI      10m     7
fas3040-201-24   /vol/vol2/lun2  /dev/sdb    host6      iSCSI      10m     7
fas3040-201-24   /vol/vol3/lun3  /dev/sdb    host6      iSCSI      10m     7
```

Example of using iscsiadm to view LUNs running iSCSI with DM-Multipath on RHEL series system

This example shows sample output from the `iscsiadm` command when it is issued in a Host Utilities environment that is running the iSCSI protocol and DM-Multipath on a Red Hat Enterprise Linux 5 or 6 series system.

Note: This example lists the available storage systems and LUNs for a session with a specific session ID. To view the details of all the sessions, you use the `iscsiadm -m session -P 3` command.

```
# iscsiadm -m session  -P 3 -r 2
Target: iqn.1992-08.com.netapp:sn.101183016
        Current Portal: 10.72.199.71:3260,1001
        Persistent Portal: 10.72.199.71:3260,1001
                **********
                Interface:
                **********
                Iface Name: default
                Iface Transport: tcp
                Iface Initiatorname: iqn.1994-05.com.redhat:5e3e11e0104d
                Iface IPaddress: 10.72.199.119
                Iface HWaddress: default
                Iface Netdev: default
                SID: 2
                iSCSI Connection State: LOGGED IN
                iSCSI Session State: LOGGED_IN
                Internal iscsid Session State: NO CHANGE
                ************************
                Negotiated iSCSI params:
                ************************
                HeaderDigest: None
                DataDigest: None
                MaxRecvDataSegmentLength: 131072
                MaxXmitDataSegmentLength: 65536
                FirstBurstLength: 65536
                MaxBurstLength: 65536
                ImmediateData: Yes
                InitialR2T: No
                MaxOutstandingR2T: 1
                ************************
                Attached SCSI devices:
                ************************
                Host Number: 4  State: running
                scsi4 Channel 00 Id 0 Lun: 0
                        Attached scsi disk sdc      State: running
                scsi4 Channel 00 Id 0 Lun: 1
                        Attached scsi disk sde      State: running
                scsi4 Channel 00 Id 0 Lun: 2
                        Attached scsi disk sdg      State: running
                scsi4 Channel 00 Id 0 Lun: 3
                        Attached scsi disk sdi      State: running
                scsi4 Channel 00 Id 0 Lun: 4
                        Attached scsi disk sdk      State: running
                scsi4 Channel 00 Id 0 Lun: 5
                        Attached scsi disk sdm      State: running
```

```
                scsi4 Channel 00 Id 0 Lun: 6
                        Attached scsi disk sdp      State: running
                scsi4 Channel 00 Id 0 Lun: 7
                        Attached scsi disk sdq      State: running
```

Example of using iscsiadm to view LUNs running iSCSI without multipathing on RHEL series system

This example shows sample output from the `iscsiadm` command when it is issued in a Host Utilities environment that is running the iSCSI protocol without multipathing on a Red Hat Enterprise Linux 5 or 6 series system. This example uses a specific session ID.

Note: This example lists the available storage systems and LUNs for a session with a specific session ID. To view the details of all the sessions, you use the `iscsiadm -m session -P 3` command.

```
# iscsiadm -m session  -P 3 -r 2
Target: iqn.1992-08.com.netapp:sn.101183016
        Current Portal: 10.72.199.71:3260,1001
        Persistent Portal: 10.72.199.71:3260,1001
                **********
                Interface:
                **********
                Iface Name: default
                Iface Transport: tcp
                Iface Initiatorname: iqn.1994-05.com.redhat:5e3e11e0104d
                Iface IPaddress: 10.72.199.119
                Iface HWaddress: default
                Iface Netdev: default
                SID: 2
                iSCSI Connection State: LOGGED IN
                iSCSI Session State: LOGGED_IN
                Internal iscsid Session State: NO CHANGE
                ************************
                Negotiated iSCSI params:
                ************************
                HeaderDigest: None
                DataDigest: None
                MaxRecvDataSegmentLength: 131072
                MaxXmitDataSegmentLength: 65536
                FirstBurstLength: 65536
                MaxBurstLength: 65536
                ImmediateData: Yes
                InitialR2T: No
                MaxOutstandingR2T: 1
                ************************
                Attached SCSI devices:
                ************************
                Host Number: 4  State: running
                scsi4 Channel 00 Id 0 Lun: 0
                        Attached scsi disk sdc      State: running
                scsi4 Channel 00 Id 0 Lun: 1
                        Attached scsi disk sde      State: running
                scsi4 Channel 00 Id 0 Lun: 2
                        Attached scsi disk sdg      State: running
                scsi4 Channel 00 Id 0 Lun: 3
                        Attached scsi disk sdi      State: running
                scsi4 Channel 00 Id 0 Lun: 4
                        Attached scsi disk sdk      State: running
                scsi4 Channel 00 Id 0 Lun: 5
                        Attached scsi disk sdm      State: running
                scsi4 Channel 00 Id 0 Lun: 6
                        Attached scsi disk sdp      State: running
                scsi4 Channel 00 Id 0 Lun: 7
                        Attached scsi disk sdq      State: running
```

Example of using iscsiadm command running iSCSI with DM-Multipath on SLES series system

This example shows sample output from the `iscsiadm` command when it is issued in a Host Utilities environment that is running the iSCSI protocol and DM-Multipath on a SUSE Linux Enterprise Server 10 or 11 series.

Note: This example lists the available storage systems and LUNs for a specific session. To view the details of all the sessions, you use the `iscsiadm -m session -P 3` command.

```
# iscsiadm -m session --sid=0 -P 3
iSCSI Transport Class version 2.0-724
iscsiadm version 2.0-868
Target: iqn.1992-08.com.netapp:sn.101180456
        Current Portal: 10.72.199.144:3260,1001
        Persistent Portal: 10.72.199.144:3260,1001
                **********
                Interface:
                **********
                Iface Name: default
                Iface Transport: tcp
                Iface Initiatorname: iqn.1996-04.de.suse:lnx.200.109
                Iface IPaddress: 10.72.200.109
                Iface HWaddress: default
                Iface Netdev: default
                SID: 1
                iSCSI Connection State: LOGGED IN
                iSCSI Session State: LOGGED_IN
Internal iscsid Session State: NO CHANGE
                ************************
                Negotiated iSCSI params:
                ************************
                HeaderDigest: CRC32C
                DataDigest: None
                MaxRecvDataSegmentLength: 131072
                MaxXmitDataSegmentLength: 65536
                FirstBurstLength: 65536
                MaxBurstLength: 65536
                ImmediateData: Yes
                InitialR2T: No
                MaxOutstandingR2T: 1
                ************************
                Attached SCSI devices:
                ************************
                Host Number: 47 State: running
                scsi47 Channel 00 Id 0 Lun: 2
                   Attached scsi disk sdj       State: running
                scsi47 Channel 00 Id 0 Lun: 1
                   Attached scsi disk sdf       State: running
                scsi47 Channel 00 Id 0 Lun: 0
                   Attached scsi disk sdb       State: running
```

Example of using iscsiadm command running iSCSI without multipathing on SLES series system

This example shows sample output from the `iscsiadm` command when it is issued in a Host Utilities environment that is running the iSCSI protocol without multipathing on a SUSE Linux Enterprise Server 10 or 11 series.

Note: This example lists the available storage systems and LUNs for a specific session. To view the details of all the sessions, you use the `iscsiadm -m session -P 3` command.

```
# iscsiadm -m session --sid=N -P 3
iSCSI Transport Class version 2.0-724
iscsiadm version 2.0-868
Target: iqn.1992-08.com.netapp:sn.101180456
```

```
        Current Portal: 10.72.199.144:3260,1001
        Persistent Portal: 10.72.199.144:3260,1001
                **********
                Interface:
                **********
                Iface Name: default
                Iface Transport: tcp
                Iface Initiatorname:iqn.1996-04.de.suse:lnx.200.109
                Iface IPaddress: 10.72.200.109
                Iface HWaddress: default
                Iface Netdev: default
                SID: 1
                iSCSI Connection State: LOGGED IN
                iSCSI Session State: LOGGED_IN
                Internal iscsid Session State: NO CHANGE
                ************************
                Negotiated iSCSI params:
                ************************
                HeaderDigest: CRC32C
                DataDigest: None
                MaxRecvDataSegmentLength: 131072
                MaxXmitDataSegmentLength: 65536
                FirstBurstLength: 65536
                MaxBurstLength: 65536
                ImmediateData: Yes
                InitialR2T: No
                MaxOutstandingR2T: 1
                ************************
                Attached SCSI devices:
                ************************
                Host Number: 47 State: running
                scsi47 Channel 00 Id 0 Lun: 2
                   Attached scsi disk sdj       State: running
                scsi47 Channel 00 Id 0 Lun: 1
                   Attached scsi disk sdf       State: running
                scsi47 Channel 00 Id 0 Lun: 0
                   Attached scsi disk sdb       State: running
```

Example of using iscsi-ls command running iSCSI without multipathing on RHEL series system

This example shows sample output from the `iscsi-ls -l` command when it is issued in a Host Utilities environment that is running the iSCSI protocol without multipathing on a Red Hat Enterprise Linux 4 series system.

```
# /sbin/iscsi-ls -l
*******************************************************************************
        SFNet iSCSI Driver Version ... 3.6.2 (27-Sep-2004 )
*******************************************************************************
TARGET NAME             : iqn.1992-08.com.netapp:sn.33604646
TARGET ALIAS            :
HOST NO                 : 0
BUS NO                  : 0
TARGET ID               : 0
TARGET ADDRESS          : 10.60.128.100:3260
SESSION STATUS          : ESTABLISHED AT Mon Jan  3 10:05:14 2005
NO. OF PORTALS          : 1
PORTAL ADDRESS 1        : 10.60.128.100:3260,1
SESSION ID              : ISID 00023d000001 TSID 103

DEVICE DETAILS :
--------------
LUN ID : 0
  Vendor: NETAPP Model: LUN Rev: 0.2
  Type:   Direct-Access                    ANSI SCSI revision: 04
  page83 type3: 60a980004f6443745359763759367733
  page83 type1:4e45544150502020204c554e204f6443745359763759367733000000000000000
  page80: 4f6443745359763759367733
  Device: /dev/sdb
LUN ID : 1
  Vendor: NETAPP Model: LUN  Rev: 0.2
  Type:   Direct-Access                    ANSI SCSI revision: 04
```

```
   page83 type3: 60a980004f644374535976426253674b
   page83 type1:4e45544150502020204c554e204f64437453597642625367 4b00000000000000
   page80: 4f644374535976426253674b
   Device: /dev/sdc
**************************************************************************
```

Enabling device persistence for newly discovered LUNs

The Linux Host sees LUNs on the storage system as SCSI devices. When you have multiple LUNs, you must be able to persistently identify each LUN across system reboots. This means you should make sure each LUN has a unique file system label and then mount the file system using that label.

Before you begin

Discover the LUNs.

About this task

The issue of persistent identification occurs because the Linux operating system assigns a unique device name, such as `/dev/sda`, to each LUN as it discovers the LUN. If you reboot or restart the iSCSI service, these names might change because the order in which the operating system discovers LUNs cannot be predicted.

When you use multipathing, DM-Multipath automatically creates persistent devices for each LUN in the `/dev/mapper/` directory on the Linux host.

Step

1. Mount the new file system by adding an entry to `/etc/fstab`.

If you are running ...	Enter the following line ...
DM-Multipath with Red Hat Enterprise Linux	*`device mount_point type`* `_netdev,defaults 0 0`
DM-Multipath with SUSE Linux Enterprise Server	*`device mount_point type`* `_netdev,defaults nofail,rw 0 0`

`device` is the name of the device in the `/dev/mapper/` directory. You can create a file system directly on a multipath device in `/dev/mapper/`. You do not have to create a partition or label on the multipath device.

`mount_point` is the mount point you created for the file system.

`type` is the file system type, such as ext2 or ext3.

`_netdev` is used for any network-dependent devices such as iSCSI. It is only used in iSCSI environments and lets you add iSCSI mount point devices to `/etc/fstab`.

Removing an unmapped LUN

When you no longer need or want it, you can use the SCSI `rescan` script to remove an unmapped LUN from the host.

Before you begin

The LUN must not be in use and must be unmapped.

You must have a copy of the `rescan` script. The `rescan` script is available with the sg3_utils package. In addition, the `rescan` script is available with Red Hat Enterprise Linux 5 Update 4 or later and Red Hat Enterprise Linux 6 or later, SUSE Linux Enterprise Server 10 SP2 or later, and

SUSE Linux Enterprise Server 11, 12, or later. For earlier versions, use the vendor-specific `rescan` scripts, which are available on their web sites. See the documentation for your HBA.

Step

1. Remove the LUN:

   `/usr/bin/rescan-scsi-bus.sh –r`

(Veritas) LUN access when using VxDMP

If you configured your system correctly for Veritas Storage Foundation, you can use VxDMP for multipathing and VxVM to manage the LUNs. You can still use commands such as `sanlun` to display information about the LUN paths.

There are some things you should check before you start working with LUNs in a Veritas Storage Foundation environment:

- If you are using iSCSI, make sure you are using a version of Veritas Storage Foundation that supports that protocol.
  See the Interoperability Matrix, which is available at *mysupport.netapp.com/matrix*.
- Make sure you have set the HBA driver parameters correctly for your system setup.
  Having the correct values for these parameters ensures that the multipathing and storage system failover work correctly.
- If you configured VxDMP, multipath devices are created for all the LUNs that are discovered by the HBA driver.
  Each time an HBA driver is started, it scans the storage system and discovers all mapped LUNs.
- Make sure you set the VxDMP restore daemon to the correct values.
  These values ensure that Veritas Storage Foundation works efficiently and correctly.
- When you use Veritas Storage Foundation, the VxVM manages the LUNs.
  This means that, in addition to using tools such as `sanlun` and `iscsadm` to display information about the LUNs, you can also use the VxVM interface to display information about the VxVM devices.

Discovering new LUNs on Vertias with FC

After you create a new LUN and map it to the Linux host, you can discover the LUN by using the SCSI rescan script.

Before you begin

You must have a copy of the `rescan` script.

The `rescan` script is available with the sg3_utils package. In addition, the `rescan` script is available with Red Hat Enterprise Linux 5 Update 4 or later, Red Hat Enterprise Linux 6 and 7 series, SUSE Linux Enterprise Server 10 SP2 or later, and SUSE Linux Enterprise Server 11 and 12 series.

For earlier versions, use the vendor-specific `rescan` scripts, which are available on their web sites. See HBA vendor-specific documentation.

Steps

1. Rescan the operating system:

   `/usr/bin/rescan-scsi-bus.sh`

2. Initiate a rescan of the operating system device tree from the Veritas Volume Manager:

   `vxdisk scandisks`

Discovering new LUNs for Red Hat 5, 6, 7 or SUSE 10, 11 and 12 while using Veritas and iSCSI

When you are running Red Hat Enterprise Linux 5, 6, or 7 series or SUSE Linux Enterprise Server 10, 11, or 12 series and the software iSCSI initiator, you can discover new LUNs by rescanning the iSCSI service on the host. Rescanning the service displays all the newly created LUNs that have been mapped to the host.

Before you begin

You must have a copy of the `rescan` script. The `rescan` script is available with the sg3_utils package. In addition, the `rescan` script is available with Red Hat Enterprise Linux 5 Update 4, Red Hat Linux 6 and 7 or later, SUSE Linux Enterprise Server 10 SP2 or later, and SUSE Linux Enterprise Server 11 and 12 or later. For earlier versions, you can use the vendor-specific `rescan` scripts, which are available on their web sites. See HBA vendor-specific documentation.

About this task

You cannot view new LUNs until after the operating system discovers them. After the LUNs have been discovered, they are automatically added to the VxDMP configuration.

Steps

1. Discover new LUNs by entering the following commands.

If you want to...	Enter...
Obtain the list of all the current sessions	**`iscsiadm -m session`**
Rescan a specific session	**`iscsiadm -m session --sid=`*`N`*` --rescan`** *`N`* is the specific session ID.
Rescan all the sessions	**`iscsiadm -m session --rescan`**
Rescan using the SCSI `rescan` script	**`/usr/bin/rescan-scsi-bus.sh`**
Rescan Veritas devices	**`vxdisk scandisks`**

2. Verify that the new LUNs have been discovered by using the `iscsiadm` command for an iSCSI setup or the`sanlun` command for other protocol hosts.

Viewing LUNs mapped to the host and on VxVM disks

Regardless of the protocol your environment uses, you have several options for viewing LUNs that are mapped to the host. The `sanlun` command works for all protocols. If you are using an iSCSI software initiator, you also have a choice between using the `iscsiadm` command or the `iscsi` command, depending on your version of Linux. You can also use `vxdisk list` to see the LUNs on VxVM disks.

Steps

1. To view a list of LUNs mapped to your host, run the appropriate command for your system environment:

If you are using...	Enter one of the following commands...
(FC, iSCSI hardware initiator) All versions of Linux	`sanlun lun show all`
(iSCSI software initiator) Red Hat Enterprise Linux 5, 6, or 7 series	• `sanlun lun show all` • `iscsiadm --mode session --sid=N -P 3`
(iSCSI software initiator) SUSE Linux Enterprise Server 10, 11, or 12 series	• `sanlun lun show all` • **(SUSE Linux Enterprise Server 10 SP2):** `iscsiadm --mode session --sid=N -P 3` • **(SUSE Linux Enterprise Server 10 SP1):** `iscsiadm --mode session --sid=N -i`

2. To view the LUNs on the VxVM disks, enter the `vxdisk list` command.

Examples of sanlun, iscsiadm, and iscsi output when used to view LUNs on Veritas

You can use either the `sanlun` command, the `iscsiadm` command, or the `iscsi` command to view the LUNs configured on your Linux host. The examples in this section show the type of output you see if you run one of these commands on your Linux operating system in an environment running VxDMP.

The tool you use depends on your version of Linux and what you would like to view. The `sanlun` command displays the host device names and the LUNs to which they are mapped. The `iscsiadm` command lists the available storage systems and LUNs.

The following sections provide examples of the type of output you see if you run one of these commands in a specific environment: for example, with iSCSI and DM-Multipath on Red Hat Enterprise Linux 5 series:

- FC running `sanlun`
- FC running `vxdisk`
- (Red Hat Linux) Software iSCSI running `sanlun`
- (SUSE Linux 10 and 11) Software iSCSI running `sanlun`
- (Red Hat Linux) Software iSCSI running `iscsiadm`
- (SUSE Linux 10 and 11) Software iSCSI running `iscsiadm`

Note: The output in the following examples has been modified to better fit the screen.

Example of using sanlun to view LUNs running Data ONTAP operating in 7-Mode with FC

This example shows sample output from the `sanlun lun show all` command when it is issued in a Host Utilities environment that is running Data ONTAP operating in 7-Mode with FC and Veritas Storage Foundation.

Note: With the Linux Host Utilities 6.0 release, the output format of the sanlun utility changed. The format is not backward-compatible when using LUNs mapped for Data ONTAP operating in 7-Mode.

```
# sanlun lun show all
controller(7mode)/                device     host                 lun
vserver(Cmode)    lun-pathname    filename   adapter    protocol   size    mode
------------------------------------------------------------------------------
fas3040-201-25    /vol/vol1/lun1      /dev/sdu   host1      FCP        5g      7
fas3040-201-25    /vol/vol2/lun2   /dev/sdt   host1      FCP        5g      7
fas3040-201-25    /vol/vol3/lun3   /dev/sds   host1      FCP        5g      7
```

If you execute the `sanlun lun show all` command in a Data ONTAP operating in 7-Mode FC environment, you get output similar to the following:

```
# sanlun lun show -p

                    ONTAP Path: fas6030-201-71:/vol/vol1/lun1
                           LUN: 12
                      LUN Size: 7g
           Controller_CF_State: Cluster Enabled
            Controller Partner: fas6030-201-70
                          Mode: 7
                     DMP NODE: fas60300_9
            Multipath Provider: Veritas
--------- ---------- ------- ------------ ------------------------------------
host      controller                      controller
path      path       /dev/   host         target
state     type       node    adapter      port
--------- ---------- ------- ------------ ------------------------------------
up        secondary  sdn     host3        2a
up        primary    sdbg    host3        2a
up        primary    sdes    host4        2b
up        secondary  sdcz    host4        2b
```

Example of using vxdisk to view LUNs running FC

This example shows sample output from the `vxdisk list` command when it is issued in a Host Utilities environment that is running the FC protocol. The `vxdisk list` command displays the LUNs on the VxVM disks:

```
# vxdisk list
DEVICE      TYPE         DISK       GROUP   STATUS
fas20200_0  auto:cdsdisk data_dg01  data_dg online thinrclm shared
fas20200_1  auto:cdsdisk data_dg02  data_dg online thinrclm shared
fas20200_2  auto:cdsdisk data_dg113 data_dg online thinrclm shared
fas20200_3  auto:cdsdisk data_dg180 data_dg online thinrclm shared
```

Example of using sanlun to view LUNs running Data ONTAP operating in 7-Mode with iSCCI and Veritas

This example shows sample output from the `sanlun lun show all` command when it is issued in a Host Utilities environment that is running Data ONTAP operating in 7-Mode with iSCSI and Veritas Storage Foundation.

Note: With the Linux Host Utilities 6.0 release, the output format of the sanlun utility changed. The format is not backward-compatible when using LUNs mapped for Data ONTAP operating in 7-Mode.

```
# sanlun lun show all
controller(7mode)/              device      host                 lun
vserver(Cmode)   lun-pathname    filename    adapter    protocol   size    mode
------------------------------------------------------------------------------
fas3040-201-24   /vol/vol1/lun1     /dev/sdb    host6      iSCSI      10m     7
fas3040-201-24   /vol/vol2/lun2     /dev/sdb    host6      iSCSI      10m     7
fas3040-201-24   /vol/vol3/lun3     /dev/sdb    host6      iSCSI      10m     7
```

If you execute the `sanlun lun show all` command in a Data ONTAP operating in 7-Mode iSCSI environment, you get output similar to the following:

```
# ./sanlun lun show -p

                    ONTAP Path: fas6030-201-70:/vol/vol1/lun1
                           LUN: 21
                      LUN Size: 7g
           Controller_CF_State: Cluster Enabled
            Controller Partner: fas6030-201-71
                          Mode: 7
                      DMP NODE: fas60300_44
            Multipath Provider: Veritas
--------- ---------- ------- ------------ --------------------------------
host      controller                      controller
path      path       /dev/   host         target
state     type       node    adapter      port
--------- ---------- ------- ------------ --------------------------------
up        iscsi          sdaq    host6        10.72.201.24
up        iscsi          sdbp    host7        10.72.201.25
```

Example of using iscsiadm to view LUNs running iSCSI and Veritas on RHEL 5 series system

This example shows sample output from the `iscsiadm` command when it is issued in a Host Utilities environment that is running the iSCSI protocol and Veritas Storage Foundation on a Red Hat Enterprise Linux 5 series system.

Note: This example lists the available storage systems and LUNs for a session with a specific session ID. To view the details of all the sessions, use the `iscsiadm -m session -P 3` command.

```
# iscsiadm -m session  -P 3 -r 2
Target: iqn.1992-08.com.netapp:sn.101183016
        Current Portal: 10.72.199.71:3260,1001
        Persistent Portal: 10.72.199.71:3260,1001
                **********
                Interface:
                **********
                Iface Name: default
                Iface Transport: tcp
                Iface Initiatorname: iqn.1994-05.com.redhat:5e3e11e0104d
                Iface IPaddress: 10.72.199.119
                Iface HWaddress: default
                Iface Netdev: default
                SID: 2
                iSCSI Connection State: LOGGED IN
                iSCSI Session State: Unknown
                Internal iscsid Session State: NO CHANGE
                ************************
                Negotiated iSCSI params:
                ************************
                HeaderDigest: None
                DataDigest: None
                MaxRecvDataSegmentLength: 131072
                MaxXmitDataSegmentLength: 65536
                FirstBurstLength: 65536
                MaxBurstLength: 65536
                ImmediateData: Yes
                InitialR2T: No
                MaxOutstandingR2T: 1
                ************************
                Attached SCSI devices:
                ************************
                Host Number: 4  State: running
                scsi4 Channel 00 Id 0 Lun: 0
                        Attached scsi disk sdc      State: running
                scsi4 Channel 00 Id 0 Lun: 1
                        Attached scsi disk sde      State: running
                scsi4 Channel 00 Id 0 Lun: 2
                        Attached scsi disk sdg      State: running
                scsi4 Channel 00 Id 0 Lun: 3
                        Attached scsi disk sdi      State: running
                scsi4 Channel 00 Id 0 Lun: 4
                        Attached scsi disk sdk      State: running
                scsi4 Channel 00 Id 0 Lun: 5
                        Attached scsi disk sdm      State: running
```

```
                scsi4 Channel 00 Id 0 Lun: 6
                        Attached scsi disk sdp      State: running
                scsi4 Channel 00 Id 0 Lun: 7
                        Attached scsi disk sdq      State: running
```

Example of using iscsiadm command running iSCSI and Veritas on SLES 10 and 11 series system

This example shows sample output from the `iscsiadm` command when it is issued in a Host Utilities environment that is running the iSCSI protocol and Veritas Storage Foundation on a SUSE Linux Enterprise Server 10 or 11 system.

Note: This example lists the available storage systems and LUNs for a specific session. To view the details of all the sessions, use the `iscsiadm -m session -P 3` command.

```
# iscsiadm -m session --sid=N -P 3
iSCSI Transport Class version 2.0-724
iscsiadm version 2.0-868
Target: iqn.1992-08.com.netapp:sn.101180456
        Current Portal: 10.72.199.144:3260,1001
        Persistent Portal: 10.72.199.144:3260,1001
                **********
                Interface:
                **********
                Iface Name: default
                Iface Transport: tcp
                Iface Initiatorname:iqn.1996-04.de.suse:lnx.200.109
                Iface IPaddress: 10.72.200.109
                Iface HWaddress: default
                Iface Netdev: default
                SID: 1
                iSCSI Connection State: LOGGED IN
                iSCSI Session State: Unknown
                Internal iscsid Session State: NO CHANGE
                ************************
                Negotiated iSCSI params:
                ************************
                HeaderDigest: CRC32C
                DataDigest: None
                MaxRecvDataSegmentLength: 131072
                MaxXmitDataSegmentLength: 65536
                FirstBurstLength: 65536
                MaxBurstLength: 65536
                ImmediateData: Yes
                InitialR2T: No
                MaxOutstandingR2T: 1
                ************************
                Attached SCSI devices:
                ************************
                Host Number: 47 State: running
                scsi47 Channel 00 Id 0 Lun: 2
                   Attached scsi disk sdj       State: running
                scsi47 Channel 00 Id 0 Lun: 1
                   Attached scsi disk sdf       State: running
                scsi47 Channel 00 Id 0 Lun: 0
                   Attached scsi disk sdb       State: running
```

(Veritas) Displaying multipathing information for VxDMP

You can use the `sanlun` command to display multipathing information for VxDMP.

Step

1. Enter the following command on the Linux host console:

   `sanlun lun show -p`

(Veritas) Examples of sanlun output for VxDMP

When you use the `sanlun` command to display multipathing information for VxDMP, the command output varies depending on the protocol you are using.

The following sections provide examples of the type of output produced when you run the `sanlun` command in the following VxDMP environments:

- FC
- Software iSCSI

FC example of using sanlun to display VxDMP information

In the following example, the `sanlun lun show -p` command is issued in a Host Utilities environment that is running the FC protocol with Veritas Storage Foundation and Data ONTAP operating in 7-Mode. The output from the command shows that VxDMP is configured.

```
# sanlun lun show -p
                    ONTAP Path: fas6030-201-71:/vol/vol1/lun1
                           LUN: 12
                      LUN Size: 7g
           Controller_CF_State: Cluster Enabled
            Controller Partner: fas6030-201-70
                          Mode: 7
                      DMP NODE: fas60300_9
            Multipath Provider: Veritas
--------- ---------- ------- ------------
-----------------------------------------
host      controller                      controller
path      path       /dev/   host         target
state     type       node    adapter      port
--------- ---------- ------- ------------
-----------------------------------------
up        secondary  sdn     host3        2a
up        primary    sdbg    host3        2a
up        primary    sdes    host4        2b
up        secondary  sdcz    host4        2b
```

Software iSCSI example of using sanlun to display VxDMP information

In the following example, the `sanlun lun show -p` command is issued in a Host Utilities environment that is running the iSCSI protocol with a software initiator Veritas Storage Foundation and Data ONTAP operating in 7-Mode. The output from the command shows that VxDMP is configured.

```
# sanlun lun show -p

                    ONTAP Path: fas6030-201-70:/vol/vol1/lun1
                           LUN: 21
                      LUN Size: 7g
           Controller_CF_State: Cluster Enabled
            Controller Partner: fas6030-201-71
                          Mode: 7
                      DMP NODE: fas60300_44
            Multipath Provider: Veritas
--------- ---------- ------- ------------
-----------------------------------------
host      controller                      controller
path      path       /dev/   host         target
state     type       node    adapter      port
```

```
--------- ---------- ------- ------------
-----------------------------------------
up        iscsi          sdaq    host6        10.72.201.24
up        iscsi          sdbp    host7        10.72.201.25
```

Removing an unmapped LUN

You can use the SCSI `rescan` script to remove an unmapped LUN from the host. This is required to remove stall entries of LUNs on the host and remove confusion between mapped and unmapped LUNs on the host.

Before you begin

The LUN must be unmapped and must not be in use.

You must have a copy of the `rescan` script. The `rescan` script is available with the sg3_utils package. In addition, the `rescan` script is available with Red Hat Enterprise Linux 5 Update 4, Red Hat Enterprise Linux 6.0 or later, SUSE Linux Enterprise Server 10 SP2 or later, and SUSE Linux Enterprise Server 11 and 12 or later. For earlier versions, you can use the vendor-specific `rescan` scripts, which are available on their web sites. See the HBA vendor-specific documentation.

Step

1. Remove the unmapped LUN:

   `/usr/bin/rescan-scsi-bus.sh -r`

Displaying available paths using VxVM on Veritas

You can use the VxVM management interface to display information about the VxVM devices and see which devices are managed by the VxVM The `vxdisk` command displays information about device, type, disk, group, and status.

Steps

1. View all VxVM devices:

   `vxdisk list`

   Note: For Veritas Storage Foundation 5.0 MP1 and MP2, the ASL displays the enclosure-based naming disk objects in uppercase.

   For Veritas Storage Foundation 5.0 MP3, Veritas Storage Foundation 5.1 and later and InfoScale 7.0 series, the default behavior of the ASL is to display the enclosure-based naming disk objects in lowercase.

   You can change the enclosure names to uppercase by using the `vxddladm set namingscheme=ebn lowercase=no` command.

Example

The output of the `vxdisk list` command is similar to the following:

```
# vxdisk list
DEVICE         TYPE            DISK         GROUP             STATUS
disk_0       auto:none       -            -        online invalid
fas31700_0   auto:cdsdisk    data_dg01    data_dg  online thinrclm shared
fas31700_1   auto:cdsdisk    data_dg02    data_dg  online thinrclm shared
fas31700_2   auto:cdsdisk    data_dg08    data_dg  online thinrclm shared
fas31700_3   auto:cdsdisk    data_dg09    data_dg  online thinrclm shared
```

```
fas31700_4   auto:cdsdisk    data_dg10    data_dg  online thinrclm shared
fas31700_5   auto:cdsdisk    data_dg11    data_dg  online thinrclm shared
fas31700_6   auto:cdsdisk    data_dg12    data_dg  online thinrclm shared
fas31700_7   auto:cdsdisk    data_dg13    data_dg  online thinrclm shared
```

2. On the host console, display path information for the device you want:

   `vxdmpadm getsubpaths dmpnodename=`*`device`*

   `device` is the name listed under the output of the `vxdisk list` command.

   #### Example

   The output of the `vxdmpadm getsubpaths dmpnodename=`*`device`* command is similar to the following:

```
vxdmpadm getsubpaths dmpnodename=fas31700_1
NAME  STATE[A]   PATH-TYPE[M] CTLR-NAME ENCLR-TYPE  ENCLR-NAME  ATTRS
=================================================================
sdb   ENABLED(A)    -          c7        FAS3170     fas31700       -
sdq   ENABLED(A)    -          c8        FAS3170     fas31700       -
```

3. To obtain path information for a host HBA:

   `vxdmpadm getsubpaths ctlr=`*`controller_name`*

   `controller_name` is the controller displayed under “CTRL-NAME” in the output of the `vxdmpadm getsubpaths dmpnodename` command.

   The output displays information about the paths to the storage system (whether the path is a primary or secondary path). The output also lists the storage system that the device is mapped to.

   #### Example

   The output of the `vxdmpadm getsubpaths ctlr=` *`controller_name`* command is similar to the following:

```
#vxdmpadm getsubpaths ctlr=c7

NAME  STATE[A]   PATH-TYPE[M] DMPNODENAME  ENCLR-TYPE  ENCLR-NAME  ATTRS
=================================================================
sdj   ENABLED(A)    -          fas31700_0   FAS3170    fas31700    -
sdb   ENABLED(A)    -          fas31700_1   FAS3170    fas31700    -
sdl   ENABLED(A)    -          fas31700_10  FAS3170    fas31700    -
sdm   ENABLED(A)    -          fas31700_11  FAS3170    fas31700    -
sdn   ENABLED(A)    -          fas31700_12  FAS3170    fas31700    -
```

(FC) Setting up a SAN boot LUN on Red Hat Enterprise Linux

You can set up a SAN boot LUN to work in a Red Hat Enterprise Linux environment that is using the FC protocol.

Before you begin

Verify that your system setup supports SAN boot LUNs. See the Interoperability Matrix.

Steps

1. Create a LUN on the storage system and map it to the host. This LUN will be the SAN boot LUN.

   You should ensure the following:

   - The SAN boot LUN is mapped to the host.
   - Multiple paths to the LUN are available.
   - The LUN is visible to the host during the boot process.

2. Enable the BIOS of the HBA port to which the SAN boot LUN is mapped.

   For information about how to enable the HBA BIOS, see your HBA vendor-specific documentation.

3. Configure the paths to the HBA boot BIOS as primary, secondary, tertiary, and so on, on the boot device.

   For more information, see your vendor-specific documentation.

4. Save and exit.

5. Reboot the host.

6. Install the operating system on the SAN boot LUN.

   Note: For Red Hat Enterprise Linux 5 series, you must specify Boot Option as `linux mpath` during the operating system installation. When you specify `linux mpath`, you can see the multipath devices (`/dev/mapper/mpath`*x*) as installation devices.

7. Install the Host Utilities.

8. Configure DM-Multipath.

(FC) Setting up a SAN boot LUN on SUSE Linux Enterprise Server

You can set up a SAN boot LUN to work in a SUSE Linux Enterprise Server environment that is using the FC protocol.

Before you begin

Verify that your system setup supports SAN boot LUNs. See the NetApp Interoperability Matrix.

Steps

1. Create a LUN on the storage system and map it to the host. This LUN will be the SAN boot LUN.

   You should ensure the following:

   - The SAN boot LUN is mapped to the host.

     Note: If you are running a SAN boot installation using a version of SUSE Linux Enterprise Server prior to 11 SP2, then only one primary path to the LUN is available. SUSE Linux Enterprise Server 11 SP2 supports multiple paths to the LUN.

   - The LUN is visible to the host during the boot process.

2. Enable the BIOS of the HBA port to which the SAN boot LUN is mapped.

   For information about how to enable the HBA BIOS, see your HBA documentation.

3. Install the operating system on the SAN boot LUN.
4. Install the Host Utilities.
5. Configure DM-Multipath.

(FC) Configuring the root partition with DM-Multipath on SUSE Linux Enterprise Server

You can configure the root partition with DM-Multipath on your SUSE Linux Enterprise Server host.

Steps

1. Configure DM-Multipath support.
2. Install the Linux Host Utilities.
3. Enable all paths to the LUN on the storage controller.
4. Re-create the initrd by running the following command:

If you are using...	Enter the following command...
SUSE Linux Enterprise Server 11	`mkinitrd -f multipath`
SUSE Linux Enterprise Server 10 SP2	`mkinitrd -f mpath`

Note: You must keep a backup of the initrd image.

5. Reboot the host.

   Note: Because all paths to the SAN boot LUN are now available on the host, you should configure all paths in the HBA boot BIOS as primary boot device, secondary boot device, and so on.

(iSCSI) SAN boot configuration for iSCSI hardware, software initiators

When you set up a SAN boot device, you use a SAN-attached disk, such as a LUN, as a root device for a host. A SAN boot LUN can be implemented either by an iSCSI HBA or a network interface card (NIC) and software iSCSI stack.

- Software iSCSI
  For a software initiator to implement a SAN boot device, you can have the root device on an iSCSI LUN, and you can use any of the following options to load the kernel:
  - A host's locally attached disk (for storing kernel and initrd images)
  - A Preboot Execution Environment (PXE) Server
- Hardware iSCSI
  If the SAN boot LUN uses an iSCSI HBA, then, because the protocol stack runs on the HBA, it is ready to communicate with the storage system and discover a LUN when it starts up.
  You can have both the boot device and root device on an iSCSI LUN.

Note: Not all operating systems supported by the Host Utilities work with iSCSI SAN boot LUNs. For example, Oracle VM does not support creating a SAN boot LUN that uses software iSCSI.

(Hardware iSCSI) Configuring SAN boot on Red Hat Enterprise Linux

When you are running Red Hat Enterprise Linux, you can configure SAN boot LUN to use an iSCSI hardware initiator.

Steps

1. Create a LUN on the storage system and map it to the host. This will be the SAN boot LUN.

   You should ensure that the SAN boot LUN is mapped, and multiple paths to the SAN boot LUN are available on the host. You should also ensure that the SAN boot LUN is visible to the host during the boot process.

2. Set the `Initiator IP Settings` and `Initiator iSCSI Name` in **Host Adapter Settings**.

3. Set the `Primary and Alternate Target IP` and `iSCSI Name and Adapter Boot Mode` to `Manual` in **iSCSI Boot Settings**.

   For information, see your HBA vendor-specific documentation.

4. After making changes to the HBA BIOS, save and exit.

5. Reboot the host.

6. Install the operating system on the boot LUN and follow the installation prompts to complete the installation.

   Note: You should specify Boot Option as `linux mpath` during the operating system installation. When you specify `linux mpath`, you can see the multipath devices (`/dev/mapper/mpath`*x*) as installation devices.

(Native multipathing) Using sanlun to display DM-Multipath information

When you are using DM-Multipath, you can use the `sanlun` command to confirm that DM-Multipath is set up.

Step

1. Enter the following command on the Linux host console:

   `sanlun lun show -p`

   You can also use the `sanlun lun show all` command to display more information about your LUN setup, such as whether you are using LUNs mapped with clustered Data ONTAP or Data ONTAP operating in 7-Mode.

   Note: Check the Interoperability Matrix to determine if clustered Data ONTAP is supported with your Host Utilities environment.

(Native multipathing) Examples of sanlun output containing DM-Multipath information

When you use the `sanlun` command to confirm that DM-Multipath is set up, the command output varies depending on the protocol you are using and whether you are using clustered Data ONTAP or Data ONTAP operating in 7-Mode.

The following sections provide examples of the type of output produced when you run the `sanlun` command in the following environments:

- Clustered Data ONTAP with FC
- Data ONTAP operating in 7-Mode with FC
- Clustered Data ONTAP with iSCSI
- Data ONTAP operating in 7-Mode iSCSI

Clustered Data ONTAP with FC: Example of using sanlun to display DM-Multipath information

The following examples show the output from the `sanlun lun show -p` command and the `sanlun lun show all` command in a Host Utilities environment that is running clustered Data ONTAP with FC and DM-Multipath.

The first example uses the `sanlun lun show -p` command. The output from the command shows that DM-Multipath (Multipath Provider: Native) is configured.

```
# sanlun lun show -p
                    ONTAP Path: vs_data28_2:/vol/vol1/lun1
                           LUN: 2
                      LUN Size: 3g
                          Mode: C
                   Host Device: 3600a09803246664c422b2d51674f7470
              Multipath Policy: round-robin 0
            Multipath Provider: Native
--------- ---------- ------- ------------ ----------------------------------
host      vserver
path      path       /dev/   host         vserver
state     type       node    adapter      LIF
--------- ---------- ------- ------------ ----------------------------------
up        primary    sdfo    host0        lif1
up        primary    sdfk    host0        lif2
up        primary    sdga    host1        lif3
```

```
up        primary    sdge    host1        lif4
up        secondary  sdgm    host1        lif5
up        secondary  sdgj    host0        lif6
up        secondary  sdfw    host0        lif7
up        secondary  sdgq    host1        lif8
```

This example uses the `sanlun lun show all` command. The output shows that the LUNs are mapped to clustered Data ONTAP operating an environment using FC.

```
# sanlun lun show all
controller(7mode)/            device          host                  lun
vserver(Cmode) lun-pathname    filename        adapter    protocol   size    mode
--------------------------------------------------------------------------------
vs_data28_2    /vol/vol1/lun1     /dev/sdgf       host1      FCP        3g      C
vs_data28_2    /vol/vol2/lun2     /dev/sdge       host1      FCP        3g      C
vs_data28_2    /vol/vol3/lun3     /dev/sdgd       host1      FCP        3g      C
```

Data ONTAP operating in 7-Mode with FC: Example of using sanlun to display DM-Multipath information

The following examples show the output from the `sanlun lun show -p` command and the `sanlun lun show all` in a Host Utilities environment that is running Data ONTAP operating in 7-Mode with FC and DM-Multipath.

Note: With the Linux Host Utilities 6.0 release, the output format of the sanlun utility has changed. The format no longer maintains backward compatibility when using LUNs mapped for Data ONTAP operating in 7-Mode.

The first example uses the `sanlun lun show -p` command. The output from the command shows that DM-Multipath (Multipath Provider: Native) is configured.

```
# sanlun lun show -p
                    ONTAP Path: fas3040-201-25:/vol/vol1/lun1
                          LUN: 27
                       LUN Size: 5g
            Controller CF State: Cluster Enabled
             Controller Partner: fas3040-201-24
                          Mode: 7
                   Host Device: 360a98000486e2f65686f647876573362
              Multipath Policy: round-robin 0
             Multipath Provider: Native
--------- ---------- ------- ------------ ----------------------------------
host      controller                      controller
path      path       /dev/   host         target
state     type       node    adapter      port
--------- ---------- ------- ------------ ----------------------------------
up        primary    sdbq    host0        2a
up        primary    sdeu    host0        1a
up        primary    sdgj    host1        2a
up        primary    sdjn    host1        1a
up        secondary  sdab    host0        2a
up        secondary  sddf    host0        1a
up        secondary  sdhy    host1        2a
up        secondary  sdlc    host1        1a
```

This example uses the `sanlun lun show all` command. The output shows that the LUNs are mapped to Data ONTAP operating in 7-Mode in an environment using FC.

```
# sanlun lun show all
controller(7mode)/              device      host                lun
vserver(Cmode)  lun-pathname    filename    adapter    protocol   size    mode
--------------------------------------------------------------------------------
fas3040-201-25   /vol/vol1/lun1   /dev/sdu    host1      FCP        5g      7
fas3040-201-25   /vol/vol2/lun2   /dev/sdt    host1      FCP        5g      7
fas3040-201-25   /vol/vol3/lun3   /dev/sds    host1      FCP        5g      7
```

Clustered Data ONTAP with iSCSI: Example of using sanlun to display DM-Multipath information

The following examples show the output from the `sanlun lun show -p` command and the `sanlun lun show all` in a Host Utilities environment that is running clustered Data ONTAP with iSCSI and DM-Multipath. The output is the same regardless of whether you are using a software iSCSI initiator or hardware iSCSI initiator.

The first example uses the `sanlun lun show -p` command. The output from the command shows that DM-Multipath (Multipath Provider: Native) is configured.

```
# sanlun lun show -p
                     ONTAP Path: vs_data28_2:/vol/vol1/lun1
                            LUN: 2
                       LUN Size: 3g
                           Mode: C
                    Host Device: 3600a09803246664c422b2d51674f7470
              Multipath Policy: round-robin 0
            Multipath Provider: Native
--------- ---------- ------- ------------ ------------------------------------
host      vserver
path      path       /dev/   host         vserver
state     type       node    adapter      LIF
--------- ---------- ------- ------------ ------------------------------------
up        primary    sdfo    host0        lif1
up        primary    sdfk    host0        lif2
up        primary    sdga    host1        lif3
up        primary    sdge    host1        lif4
up        secondary  sdgm    host1        lif5
up        secondary  sdgj    host0        lif6
up        secondary  sdfw    host0        lif7
up        secondary  sdgq    host1        lif8
```

This example uses the `sanlun lun show all` command. The output shows that the LUNs are mapped to clustered Data ONTAP in an environment using iSCSI.

```
# sanlun lun show all
controller(7mode)/              device       host                  lun
vserver(Cmode) lun-pathname     filename     adapter    protocol   size    mode
--------------------------------------------------------------------------------
vs_data78_1    /vol/vol1/lun1     /dev/sdcx     host29     iSCSI      3g      C
vs_data78_0    /vol/vol2/lun2     /dev/sdcw     host20     iSCSI      3g      C
vs_data79_1    /vol/vol3/lun3     /dev/sdck     host14     iSCSI      3g      C
```

Data ONTAP operating in 7-Mode iSCSI: Example of using sanlun to display DM-Multipath information

The following examples show the output from the `sanlun lun show -p` command and the `sanlun lun show all` in a Host Utilities environment that is running Data ONTAP operating in 7-Mode with iSCSI and DM-Multipath. The output is the same regardless of whether you are using a software iSCSI initiator or hardware iSCSI initiator.

The first example uses the `sanlun lun show -p` command. The output from the command shows that DM-Multipath (Multipath Provider: Native) is configured.

```
# sanlun lun show -p
                     ONTAP Path: f3170-201-37:/vol/vol1/lun1
                            LUN: 5
                       LUN Size: 6g
            Controller CF State: Cluster Enabled
             Controller Partner: f3170-201-36
                           Mode: 7
                    Host Device: 360a98000572d5a74526f6374342f5658
              Multipath Policy: round-robin 0
            Multipath Provider: Native
--------- ---------- ------- ------------ --------------------------------
host      controller                      controller
path      path       /dev/   host         target
```

```
state     type       node    adapter      port
--------- ---------- ------- ------------ --------------------------------
up        iscsi      sdh     host8        10.72.201.37
up        iscsi      sdp     host10       192.168.100.37
```

This example uses the `sanlun lun show all` command. The output shows that the LUNs are mapped to Data ONTAP operating in 7-Mode in an environment using iSCSI.

```
# sanlun lun show all
controller(7mode)/              device      host                  lun
vserver(Cmode)   lun-pathname    filename    adapter    protocol   size    mode
------------------------------------------------------------------------------
fas3040-201-24   /vol/vol1/lun1  /dev/sdb    host6      iSCSI      10m     7
fas3040-201-24   /vol/vol2/lun2  /dev/sdb    host6      iSCSI      10m     7
fas3040-201-24   /vol/vol3/lun3  /dev/sdb    host6      iSCSI      10m     7
```

(Software iSCSI) Configuring SAN boot on Red Hat Enterprise Linux 5 or 6 series

You can configure SAN boot on a Red Hat Enterprise Linux 5 or 6 series host using software iSCSI.

Before you begin

Check the Interoperability Matrix.

Steps

1. When you initiate the installation, specify the Boot Option as `linux mpath` and press Enter.
2. Continue with the installation until you reach the storage configuration page. Click **Advanced storage configuration**.
3. Select **Add iSCSI target** and click **Add drive**.
4. Enter the Target IP address and the iSCSI initiator name.

   Note: You should ensure that you associate this IQN with the correct privileges on the storage controller.

5. On the storage controller, create an igroup with the initiator name that you provided in Step 4.
6. Create a LUN on the storage system on which you intend to create root partition, and map it to the igroup.
7. Return to the host screen.
8. Click **Add Target** in the **Configure iSCSI Parameters** window.

   When you add the target, the target portal is discovered.

   Note: You should ensure that multiple target portals are discovered, because the Red Hat installer does not identify the iSCSI device as a multipathed device unless it has more than one path.

9. To discover more target portals, repeat Step 2 through Step 8.

   You should now see a multipathed iSCSI device listed in the drives section.

   Note: If the iSCSI multipathed device is not listed, you should check the configuration.

10. Select a partitioning layout as `Create custom layout` and Click **Next**.

You can now proceed with the installation process and enter choices until you reach the Installation Summary page.

11. At the storage devices selection screen, select the iSCSI multipathed device from the list of allowable drives where you want to install the root file system.
12. Create the root file system on the selected device and select the mount point as `/`.
13. Create a SWAP partition.

    Note: You can create a SWAP partition on the same LUN that contains the root partition or on a different LUN.

    If you are using the software suspend functionality, you should ensure that the SWAP partition is on a local disk.

14. Create the /boot partition.

    You can create a /boot partition on a locally attached disk or use a PXE server to load the kernel boot image.

15. Click **Next** and follow the installation prompts to complete the installation.

Configuring SAN boot on Red Hat Enterprise Linux 7 series

You can configure SAN boot on a Red Hat Enterprise Linux 7 series host using software iSCSI.

Before you begin

Check the Interoperability Matrix.

Steps

1. Initiate the installation.
2. Continue with the installation until you reach the installation summary page.
3. Click **Installation Destination.**
4. Click **Add a disk**
5. Click **Add iSCSI target**
6. Enter the target IP address and the iSCSI initator name.
7. On the storage controller, create an igroup with the initiator name that you provided in Step 6.
8. Create a LUN on the storage system on which you intend to create a root partition, and map it to the igroup.
9. Return to the host screen.
10. Click **Start Discovery** to discover target portals.
11. Select all the discovered iSCSI sessions and click on **Log In**.
12. Navigate to **Multipath Devices** to select iSCSI multipath SAN boot LUN and click **Done**.
13. Select the **I will configure partitioning** option and click **Done**.
14. In the **Manual Partitioning** window, create the root file system on the selected device and select the mount point as /.

15. Create a SWAP partition.

    Note: You can create a SWAP partition on the same LUN that contains the root partition or on a different LUN.

    If you are using the software suspend funcitonality, you should ensure that the SWAP partition is on a local disk.

16. Create the /boot partition on the locally attached disk or use a PXE server to load the kernel boot image.

17. Click **Done** and follow the installation prompts to complete the installation

Related information

Interoperability Matrix Tool: mysupport.netapp.com/matrix

(Software iSCSI) Configuring SAN boot on SUSE Linux Enterprise Server

You can configure a SAN boot LUN on SUSE Linux Enterprise Server. Doing this requires multiple steps and close attention to the requirements in each step.

Before you begin

Verify that your system setup supports SAN boot LUNs. See the NetApp Interoperability Matrix.

Steps

1. Log in to the storage system console or the Web interface of the storage system.
2. When you initiate the installation, specify Boot Option as follows: `linux withiscsi=1 netsetup=1`
3. In the **iSCSI Initiator Overview** page, select the **Service** tab and enter the Target IP address and the **iSCSI initiator** name.

   Note: You should ensure that you associate this IQN with the appropriate privileges on the storage controller.

4. On the storage controller, create an igroup with the initiator name that you provided in the previous step.
5. Create a LUN on the storage system on which you can create the root partition, and map it to the igroup.
6. Return to the host screen. Select the **Connected Targets** tab and click **Add**.
7. On the **iSCSI Initiator Discovery** page, perform the following steps:
   a. Specify the IP address of the storage system.
   b. Specify the port. The default is 3260.
   c. Specify the credentials if you are using an authentication mechanism.
   d. Click **Next**.
8. In the list of storage systems that are discovered, click **Connect** for each one. You might also have to do this for the authentication credentials also.

Note: During the installation, you should enable only one path to the root LUN.

Click **Next**.

9. Verify that the value for Connected is true for all the targets and click **Next**.

   The Connected Targets pane lists all the targets.

10. Set the Start-up mode to `onboot` by using the **Toggle Start-up** button, and click **Finish**.
11. In the **Installation Settings** page, select the **Expert** tab.
12. Click **Partitioning**.
13. Select the `Create Custom Partition Setup` option.

    Note: You can view the list of local disks and LUNs.

14. Select the `Custom Partitioning (for experts)` option.
15. In the **Expert Partitioner** page, select the LUN where you want to install the root file system.
16. Create the root file system on the selected LUN and select the mount point as `/`.
17. Click the **Fstab Options** button.
18. Select the `Mount by Device ID` option.
19. Ensure that you have the _netdev, nofail keyword in the Arbitrary Option Value text box, and click **OK**.
20. In the **Create a Primary Partition** page, click **OK**.
21. Create a SWAP partition.

    Note: You can create a SWAP partition on the same LUN that contains the root partition or on a different LUN.

    If you are using the software suspend functionality, you should ensure that the SWAP partition is on a local disk.

22. Create the /boot partition.

    You can create a /boot partition on a locally attached disk or use a PXE server to load the kernel boot image.

23. After you return to the **Expert Partitioner** page, review the configuration. Click **Finish**.
24. In the **Installation Settings** page, click the **Booting** tab.
25. Select **Default Label**.
26. Click **Edit**.
27. For the `Optional Kernel Command Line Parameter`, ensure that all references to installer arguments are removed.

    The parameter should look similar to the following:
    ```
    resume=/dev/sda1 splash=silent showopts
    ```

28. Click **OK**.
29. Click the **Boot Loader Installation** tab.
30. In the **Boot Loader Location** pane, select the `Boot from Master Boot Record` option.

Click **Finish**. Doing this returns you to the Installation Settings page.

31. Review the configuration settings and click **Accept**. The **Confirm Installation** page is displayed.
32. Click **Install** and follow the prompts to complete the installation.

(Software iSCSI) Configuring multipathing for a SAN boot LUN using SUSE Linux Enterprise Server

When you set up a SAN boot LUN for SUSE Linux Enterprise Server, you install the operating system on a SCSI device. After you do that, you should set up multipathing. Multipathing must be configured on the root partition of the SAN boot LUN.

Before you begin

Your system setup must support SAN boot LUNs, as listed in the Interoperability Matrix.

Steps

1. Enable additional paths to the NetApp LUN (root LUN).
2. Use YaST2 to change the startup mode for all the iSCSI sessions:

For SUSE Linux Enterprise Server...	Set the startup mode to...
11 or 12 series	**`Onboot`**
10 SP2	**`automatic`**

3. Modify the `/etc/sysconfig/network/ifcfg-`*`x`* file so that the value of STARTMODE reads **`nfsroot`**.
4. Change the value of the session re-establishment timeout for iSCSI sessions fetching the SAN boot LUN:

   `iscsiadm -m node -T` ***`targetname`*** **`-p ip:port -o update -n node.session.timeo.replacement_timeout -v 5`**

5. Create a new initrd image with the root partition on multipath enabled:

For SUSE Linux Enterprise Server...	Enter the command...
11 or 12 series	**`mkinitrd -f "iscsi multipath"`**
10 series	**`mkinitrd -f "iscsi mpath"`**

6. Change the startup mode of the iSCSI sessions fetching the SAN boot LUN:

   `iscsiadm -m node -T` ***`targetname`*** **`-p ip:port -o update -n node.startup -v onboot`**

   `iscsiadm -m node -T` ***`targetname`*** **`-p ip:port -o update -n node.conn[0].startup -v onboot`**

7. Reboot the host.

   The system boots with multipathing enabled on the root device.

8. Verify that multipathing is enabled on the root device by running the `mount` command and ensuring that the root partition is on a DM-Multipath device.

Configuring SAN boot in a Veritas environment

You can set up a SAN boot LUN to work in a Veritas Storage Foundation environment.

Before you begin

SAN boot LUNs must be supported with your version of Veritas Storage Foundation and your Linux operating system. See the NetApp Interoperability Matrix.

About this task

When you are working in a Veritas Storage Foundation environment, the steps you must perform to set up a SAN boot LUN are essentially the same for both Red Hat Enterprise Linux and SUSE Linux Enterprise Server.

Steps

1. Create a LUN on the storage system.

   This LUN will be the SAN boot LUN.

2. Map the LUN to the host.

3. Ensure that only one primary path is available to the LUN.

4. Ensure that only one SAN boot LUN is available to the host.

5. Enable the boot BIOS of the HBA port to which the SAN boot LUN is mapped.

   It is best to enable the `spinup delay` option for the HBA port.

   For information about how to enable the boot BIOS, see the HBA vendor-specific documentation.

6. After performing the appropriate changes to the HBA BIOS and ensuring that the SAN boot LUN is visible, install the operating system on the SAN boot LUN.

   Before installing the operating system, see the section on rootability in the *Veritas Volume Manager Administrator's Guide* for Linux that is shipped along with the software for partitioning information.

   Note: The Red Hat Enterprise Linux 4 Update 4 distribution does not include HBA drivers for 4-Gb and 8-Gb QLogic cards; therefore, you must use the device driver kit provided by QLogic. For more information about the supported drivers, see the NetApp Interoperability Matrix.

   Note: When you install the SUSE Linux Enterprise Server operating system, you must ensure that GRUB is installed in the Master Boot Record. You can do this from the Expert tab in the software package selection screen during installation.

7. After installing the operating system, reboot the host.

   The host boots from the SAN boot LUN on the storage system through a primary path.

8. Install the Linux Host Utilities.

9. If you are using HBA drivers acquired from an OEM, install the supported versions of the drivers.

10. Verify the HBA settings.

11. Install Veritas Storage Foundation and any appropriate patches or fixes for it.

12. Configure the `vxdmp restore` daemon by setting it to an interval of 60:

    `vxdmptune dmp_restore_interval 60`

    On reboot, this value takes effect and remains persistent across system reboots.

13. For Veritas Storage Foundation 5.1 and later, set the Veritas DMP LUN retries tunable to a value of 300:

    `vxdmpadm settune dmp_lun_retry_timeout=300`

    The new value takes effect immediately.

14. For Veritas Storage Foundation 6 series and InfoScale 7 series, set the Veritas DMP LUN retries tunable to a value of 60:

    `vxdmpadm settune dmp_lun_retry_timeout=60`

    The new value takes effect immediately.

15. (For Veritas Storage Foundation 5.1 SP1 and later, InfoScale 7 series, set the value of the dmp_path_age to an interval of 120 by entering the following command:

    `vxdmpadm settune dmp_path_age=120`

    The new value takes effect immediately.

16. Enable persistence by entering the following command:

    `vxddladm set namingscheme=osn persistence=yes`

    You must enable persistence before you can encapsulate the root disk.

17. Encapsulate the root disk for use in VxVM:

    `vxdiskadm`

    For the detailed steps, see the section on encapsulating the disk in the *Veritas Volume Manager Administrator's Guide* for Linux that is shipped along with the software.

18. Reboot the host after encapsulation.

19. Verify the encapsulation:

    `vxprint`

    This command displays the rootvol and swapvol volumes under the corresponding disk group.

20. Configure the paths to the HBA boot BIOS as primary, secondary, tertiary, and so on, on the boot device.

    For more information, see the respective HBA vendor-specific documentation.

Support for host virtualization

The Host Utilities support virtualization, including products based on technologies such as Kernel-based Virtual Machine (KVM) and Xen. The products derived from KVM are Red Hat Enterprise Linux KVM, SUSE Linux Enterprise Server KVM, and RHEV. Products derived from Xen are Citrix XenServer, SUSE Linux Enterprise Server Xen, and Oracle VM.

Server virtualization is a method of dividing computer resources into multiple, isolated environments. In a virtual ecosystem, a host operating system runs one or more guest virtual machines in a simulated environment. Each guest virtual machine (VM) has access to all of the host's hardware. You can configure guest VMs in both full and paravirtualized modes.

Full virtualization includes a virtualization translation layer (VMM) that ensures that all guest system calls are translated to the native hypervisor's format. Using this approach, all guest operating systems run completely unmodified. Commercial products based on the full virtualization concept are RHEL-KVM, RHEV, and Oracle VirtualBox. KVM is a full virtualization solution for Linux on x86 hardware that contains virtualization extensions (Intel VT or AMD-V). It consists of a loadable kernel module (kvm.ko) that provides the core virtualization infrastructure, and a processor-specific module (kvm-intel.ko or kvm-amd.ko). KVM is supported with Red Hat Enterprise Linux 5 Update 4 or later and Red Hat Enterprise Linux 6.0 or later.

Note: KVM leverages the Linux kernel's infrastructure and extends it to serve as a full-fledged hypervisor. Thus, any limitations that apply to the Linux kernel also apply to the KVM hypervisor.

Note: If you are using Oracle VM hypervisor, see the sections for configuring FC, iSCSI, and multipathing. In this guide, sections that refer to Red Hat Enterprise Linux also apply to Oracle Linux. Both operating systems use the same instructions for the tasks featured in this guide.

Paravirtualization support involves providing a paravirtualized network driver, a paravirtualized block I/O device (disk) driver, and a balloon driver to affect the operation of the guest virtual memory manager and provide CPU optimization for Linux guests. Both KVM and Oracle VM provide limited paravirtualization support for guest VMs. This approach to virtualization includes a slim hypervisor kernel running on the bare metal host hardware. Each guest OS must be ported to the hypervisor's kernel interface. The most renowned project utilizing the paravirtualization concept is Xen. Different commercial products based on Xen include OVM, Citrix XenServer, and SLES XEN.

Note: LUHU iSCSI best practices utilize the iSCSI software initiator from within the guest operating system. This applies to all the supported hypervisors.

Oracle VM is a server virtualization solution that consists of Oracle VM Server (OVS). This is a self-contained virtualization environment designed to provide a lightweight, secure, server-based platform for running virtual machines. OVS is based on an updated version of the underlying Xen hypervisor technology. Oracle VM Manager provides the user interface to manage OVS and guests.

Hypervisor VHD requires alignment for best performance

A virtual hard disk (VHD) is partitioned with a master boot record that is used by a Linux virtual machine. It must be aligned with the underlying LUN for best performance.

If the data block boundaries of a disk partition do not align with the block boundaries of the underlying LUN, the storage system often has to complete two block reads or writes for every operating system block read or write. The additional block reads and writes caused by the misalignment can lead to serious performance problems.

The misalignment is caused by the location of the starting sector for each partition defined by the master boot record. Partitions created by Linux usually are not aligned with underlying NetApp LUNs.

A workaround for misaligned Linux guests is available at the NetApp Linux Community Program site at *http://linux.netapp.com/tools/fix-alignment*.

Related information

Technical report TR-3747

Supported Linux and Data ONTAP features

The Host Utilities support a number of features and configurations for Linux hosts and storage systems running Data ONTAP.

Some of the supported features include

- SAN booting
- The Linux Device Mapper Multipathing
- Volume management and multipathing with Veritas Storage Foundation
- Host and storage virtualization
- ALUA

Note: Your specific environment can affect what the Host Utilities support.

Protocols and configurations supported by Host Utilities

The Host Utilities provide support for Fibre Channel, Fibre Channel over Ethernet (FCoE), and iSCSI connections to the storage system using direct-attached, fabric-attached, and Ethernet network configurations.

These protocols enable the host to access data on storage systems. The storage systems are targets that have storage target devices called LUNs.

Protocols enable hosts to access LUNs to store and retrieve data.

For more information about using the protocols with your storage system, see the *Data ONTAP SAN Administration Guide for 7-Mode* for your version of Data ONTAP.

For more details on supported topologies, including diagrams, see the *Data ONTAP SAN Configuration Guide for 7-Mode* for your version of Data ONTAP.

The sections that follow provide high-level information about these protocols.

The FC protocol

The FC protocol requires one or more supported host bus adapters (HBAs) in the host. Each HBA port is an initiator that uses FC to access the LUNs on the storage system. The HBA port is identified by a worldwide port name (WWPN).

You need to make a note of the WWPN so that you can supply it when you create an initiator group (igroup). To enable the host to access the LUNs on the storage system using the FC protocol, you must create an igroup on the storage system and provide the WWPN as an identifier. Then, when you create the LUN, you map it to that igroup. This mapping enables the host to access that specific LUN.

The Linux Host Utilities support the FC protocol with fabric-attached SAN and direct-attached configurations:

- Fabric-attached SAN
  The Host Utilities support two variations of fabric-attached SANs:
  - A single-port FC connection from the HBA to the storage system through a single switch
    A host is cabled to a single FC switch that is connected by cable to redundant FC ports on a high-availability storage system. A fabric-attached single-path host has one HBA.

    - A dual-port FC connection from the HBA to the storage system through dual switches
      The redundant configuration avoids the single point of failure of a single-switch configuration.

- Direct-attached
  A single host with a direct FC connection from the host HBA to stand-alone or high-availability storage system configurations.

  Note: You should use redundant configurations with two FC switches for high availability in production environments. However, direct FC connections and switched configurations using a single-zoned switch might be appropriate for less critical business applications.

The FCoE protocol

Fibre Channel over Ethernet (FCoE) is a new model for connecting hosts to storage systems. Like the traditional FC protocol, FCoE maintains existing FC management and controls, but it uses a 10-gigabit Ethernet network as the hardware transport.

Setting up an FCoE connection requires one or more supported converged network adapters (CNAs) in the host, connected to a supported data center bridging (DCB) Ethernet switch. The CNA is a consolidation point and effectively serves as both an HBA and an Ethernet adapter.

In general, you can configure and use FCoE connections the same way you use traditional FC connections.

The iSCSI protocol

The iSCSI protocol is implemented on both the host and the storage system.

On the host, the iSCSI protocol is implemented over either the host's standard Ethernet interfaces or an HBA.

On the storage system, the iSCSI protocol can be implemented over the storage system's standard Ethernet interface using one of the following:

- A software driver that is integrated into Data ONTAP.
- (Data ONTAP 7.1 and later) An iSCSI target HBA or an iSCSI TCP/IP offload engine (TOE) adapter.

The connection between the initiator and target uses a standard TCP/IP network. The storage system listens for iSCSI connections on TCP port 3260.

You need to make a note of the iSCSI node name so that you can supply it when you create an igroup.

SAN booting

SAN booting is the general term for booting a Linux host from a storage system LUN instead of an internal hard disk. SAN booting uses a SAN-attached disk, such as a LUN configured on a storage controller, as a boot device for a host.

Note: SAN booting is not supported with Oracle VM.

A SAN boot LUN provides the following advantages:

- No maintenance and servicing costs for hard drives
  You can remove the hard drives from your servers and use the SAN for booting needs.
- Consolidated and centralized storage, because the host uses the SAN
- Lower cost
  The hardware and operating costs are lowered.

- Greater reliability
  Systems without the disks are less prone to failure.
- Quick server swaps
  If a server fails, systems without the disks can be swapped.
- Better disaster recovery
  Site duplication is simplified.

For information about the configurations that are supported for SAN boot LUNs, see the Interoperability Matrix.

Support for Linux Device Mapper Multipathing

You can use Linux Device Mapper Multipathing (DM-Multipath) to provide I/O failover and path load sharing for multipathed block devices.

The software module for DM-Multipath is included in the standard Red Hat Enterprise Linux and SUSE Linux Enterprise Server distributions. You should install additional user space packages as part of the setup process.

For information on setting up DM-Multipath, see section *DM-Multipath configuration*.

Volume management and multipathing with Veritas Storage Foundation

If your environment uses Veritas Storage Foundation, you can use the tools it supplies to work with LUNs and provide multipathing. These tools include Veritas Volume Manager (VxVM) and Dynamic Multipathing (VxDMP).

With VxVM, you can manage the LUNs that you create on the storage systems. You can use VxDMP to provide the multipathing solution for your system.

For more information about Veritas Volume Manager, see the *Veritas Volume Manager Administrator's Guide* for Linux that is shipped along with the software.

To determine which versions of Veritas are supported, see the Interoperability Matrix.

Linux configurations that support ALUA

The Linux Host Utilities support ALUA (asymmetric logical unit access) on hosts running Red Hat Enterprise Linux or SUSE Linux Enterprise Server and a version of Data ONTAP that supports ALUA.

Note: ALUA is also known as Target Port Group Support (TPGS).

ALUA defines a standard set of SCSI commands for discovering path priorities to LUNs on SANs. When you have the host and storage controller configured to use ALUA, it automatically determines which target ports provide optimized and unoptimized access to LUNs.

Note: If you are using clustered DATA ONTAP, ALUA is supported with the FC, FCoE, and iSCSI protocols and you must use it. If you are using DATA ONTAP operating in 7-Mode, ALUA is supported only with the FC and FCoE FC protocols.

ALUA is automatically enabled when you set up your storage for FC.

The following configurations support ALUA:

Host Utilities Version	Host requirements	Data ONTAP versions
Host Utilities 4.0 and later	• Red Hat Enterprise Linux 6 or later • Red Hat Enterprise Linux 5 Update 1 or later • SUSE Linux Enterprise Server 10 SP2 or later • SUSE Linux Enterprise Server 11 or later **Note:** Veritas Storage Foundation 5.1 or later support ALUA with the FC protocol.	7.2.4 or later

Troubleshooting

If you encounter problems while running the Host Utilities on FC, iSCSI, or Veritas Storage Foundation, you can check the sections that follow for troubleshooting tips.

LUNs not recognized by DM-Multipath

LUNs are mapped to the Linux host and are correctly reported by the `sanlun lun show all` command. However, the LUNs are not reported by the `multipath -ll` command. Check to make sure the LUNs are not blacklisted (excluded) from the DM-Multipath support configuration in the `/etc/multipath.conf` file.

Enter the following command on the Linux host to display the devices that are blacklisted:

```
multipath -v3 -d | grep blacklist
```

If any devices show up as being blacklisted, check the devnode_blacklist or blacklist section of the `/etc/multipath.conf` file. Ensure that all the entries are correctly specified.

If the devices are not blacklisted, but are still not recognized by the `multipath` command, regenerate the multipath maps by entering the following command:

```
multipath -v3
```

For more information, see bug number 228744 on Bugs Online, which is available on the NetApp Support Site.

(FC) LUNs not visible on the Linux host

FC LUNs appear as local disks to the host. If the storage system LUNs are not available as disks on the hosts, try rescanning the HBAs by reloading the drivers. If the LUNs still do not appear, verify the configuration settings.

Configuration settings	What you should do
Driver version	Verify that the version of the external HBA driver you are running is supported. You can check the up-to-date system requirements in the NetApp Interoperability Matrix.
Cabling	Verify that the cables between the host and the storage system are properly connected.
Zoning	Verify that the zoning is correct.
HBA Setting	Verify the recommended HBA settings.
FC service status	Verify that the FC service is licensed and started on the storage system. For more information, see the *Data ONTAP SAN Administration Guide for 7-Mode*.
HBA World Wide Port Names	Verify that you are using the correct initiator WWPNs in the igroup configuration. On the storage system, use the `igroup show` command to display the WWPNs of the initiators in the storage system's igroups. On the host, use the HBA tools to display the initiator WWPN. The initiator WWPNs configured in the igroup and on the host should match.

Configuration settings	What you should do
Initiator login	Verify that the initiator is logged in to the storage system by entering the `fcp show initiator` command on the storage system console. If the initiator is configured and logged in to the storage system, the storage system console displays the initiator WWPN.
LUN mappings	Verify that the LUNs are mapped to an igroup. On the storage system, use one of the following commands: • `lun show -m` displays all the LUNs and the igroups to which they are mapped. • `lun show -g igroup-name` displays the LUNs mapped to a specific igroup.
System requirements	Verify that the components of your configuration are supported. Verify that you have the correct host Operating System (OS) Service Pack level, initiator version, Data ONTAP version, and other system requirements. You can check the up-to-date system requirements in the NetApp Interoperability Matrix.

FC troubleshooting

The troubleshooting sections that follow provide tips for dealing with problems that might occur when you are running the FC protocol with the Host Utilities.

(FC) Warnings displayed during Host Utilities installation

Occasionally, you might see warnings during the Host Utilities installation.

These warnings might be similar to the following:

- `Warning: libnl.so (32-bit) library not found, some sanlun commands may not work. Refer Linux Host Utilities Installation and Setup Guide for more details`
- `Warning: libHBAAPI.so (32-bit) library not found, some sanlun commands may not work. Refer Linux Host Utilities Installation and Setup Guide for more details`
- `Warning: libdevmapper.so (32-bit) library not found, some sanlun commands may not work. Refer Linux Host Utilities Installation and Setup Guide for more details`

To avoid these warnings, make sure you install the packages that provide the libraries before you install the Host Utilities software. For more information, see the information on installing and configuring QLogic and Emulex HBAs.

(FC) Linux hosts with QLogic initiators fail to detect LUNs

A Linux host with QLogic initiators fails to detect LUNs mapped to it when configured to autonegotiate port topology and speed. This can cause the host to miss a few LUNs during the HBA scan. This behavior persists even after rebooting the host.

About this task

In hosts using QLogic initiators, the host initiator fails to establish a consistent topology and speed, even though both are set to autonegotiate. When using QLogic initiators on a Linux host, ensure that you set the following settings for port topology and speed using the QConvergeConsole CLI package.

Steps

1. Set the value of QLogic HBA port topology to Point to Point Only while connecting to a storage controller operating in SSI, Standby or Partner CF modes, or to Loop Only if the storage system is operating in Dual Fabric or Mixed CF modes.
   a. Run the following command:

      `/opt/QLogic_Corporation/QConvergeConsoleCLI/quacli`

   b. Select **Main Menu** > **Configure HBA Settings**.
   c. For each of the WWPNs listed, select **ConnectionOptions** and set it to Point to Point Only or Loop Only, as required.
2. Set the value of the QLogic port speed to the highest speed possible, depending on its maximum and the maximum of the switch or target port to which it is connected.
   a. Run the following command:

      `/opt/QLogic_Corporation/QConvergeConsoleCLI/quacli`

   b. Select **Main Menu** > **Configure HBA Settings**.
   c. For each of the WWPNs listed, select the **Data Rate** option and set it to the specified speed.

(FC) The SAN booted root on a DM-Multipath host freezes during FC path faults

The SAN booted root on a DM-Multipath host triggers a freeze during FC path faults on Red Hat Enterprise Linux 5 Update 1.

For SAN boot support on Red Hat Enterprise Linux 5 Update 1, you have to download the appropriate errata multipathing package from the Red Hat Network Web site. For more information, see NetApp Interoperability Matrix.

(FC) Poor performance due to misalignment of partitioned LUNs

If you experience poor performance on some or all FC LUNs that have partitions, it might be due to an alignment mismatch between the LUN block boundaries and the underlying WAFL boundaries.

This problem occurs only with partitions and only with certain combinations of disk geometry parameters.

You can resolve the issue by using the `fdisk` command to manually enter the `disk geometry` parameters when creating partitions. For specific steps, see bug number 156121 on Bugs Online, which is available on the NetApp Support Site..

(FC) sanlun command displays error when HBA libHBAAPI.so is missing

The `sanlun` command uses the HBAAPI software provided by both the host operating system and the HBA vendor's API plugins to gather information about the HBAs, including their WWPNs. If this software is not available, the `sanlun` command returns an error message.

Error messages similar to the following indicate that the operating system's `libHBAAPI` library is not installed:

- `Unable to locate /usr/lib/libHBAAPI.so library`
- `Unable to locate /usr/lib64/libHBAAPI.so library`

Error messages similar to the following indicate that the HBA vendor API plug-in is not installed:

- `No supported adapters are present`
- `Unable to load HBA control library`

To avoid this problem, make sure you have the correct management software package for your HBA and host architecture installed:

- For QLogic HBAS, install the QLogic QConvergeConsole CLI package.
- For Emulex HBAs, install the Emulex OneCommand Manager core application (CLI) package.

Note: If you are using a different HBA that is supported by the Host Utilities, you must install the management software for that HBA.

These packages are available from the HBA vendor.

(FC) LUNs are unavailable after a storage system failover or giveback

The host requires specific driver, HBA, and FC switch zoning settings to maintain LUN access across a storage system takeover, giveback, and reboot. You should confirm that you have set the required driver and HBA settings. Also, verify that you have correctly zoned your FC switches.

For multipath LUNs, you can restore access to the LUNs by restarting the multipathing daemon and updating the multipathing maps by running the following commands:

- **`/etc/init.d/multipathd restart`**
- **`multipath -v3`**

(FC) SCSI rescan issues on SUSE Linux Enterprise Server hosts

On a SUSE Linux Enterprise Server host connected to a storage controller, the presence of dummy LUNs causes issues during a SCSI rescan. This issue is not yet resolved in SUSE Linux Enterprise Server 9 and SUSE Linux Enterprise Server 10.

Dummy LUNs are those that show up as "UNKNOWN" in `/proc/scsi/scsi/` even when no LUNs are mapped to the host. These LUNs have a generic device node (/dev/sg* entry), but no corresponding disk node (/dev/sd* entry). Due to these dummy LUNs, subsequent SCSI rescans on the host fail to recognize newly added LUNs, unless preceded by a write to the sysfs delete attribute of the dummy LUN.

Warning messages displayed when using LVM with multipathing software

When you use LVM in a multipathing environment, it displays a warning message.

In a default configuration, LVM scans all attached disks and identifies the disks that contain physical volumes. In environments that use multipathing software, each path to a particular LUN is registered

as a different SCSI device. LVM detects this and chooses one of the devices. LVM then displays a warning message.

To avoid this problem, you should modify the paths to the `preferred_names` parameter in the`/etc/lvm/lvm.conf` file.

The following is an example of a how the `preferred_names` parameter line should look:

```
preferred_names = [ "^/dev/mapper/*" ]
```

After you make the change, perform a rescan (`pvscan` and `vgscan`) to ensure all devices are properly displayed.

iSCSI troubleshooting

Sometimes you might encounter a problem while running iSCSI. The sections that follow provide tips for resolving any issues that might occur.

(iSCSI) LVM devices are not automatically mounted during system boot on SUSE Linux Enterprise Server 11

Currently, the volume groups created on iSCSI devices are not automatically scanned when iSCSI LUNs are discovered. Therefore, during the system boot, the volume groups that were created on the iSCSI devices are unavailable.

To overcome this problem, manually mount the logical volumes by using the following command:

`/sbin/mount -a`

(iSCSI) LVM devices are not automatically mounted during system boot on SUSE Linux Enterprise Server 10

Currently, the volume groups created on iSCSI devices are not automatically scanned when iSCSI LUNs are discovered. Therefore, during the system boot, the volume groups that were created on the iSCSI devices are unavailable. To overcome this problem, a helper script is provided in `/usr/share/doc/packages/lvm2/ lvm-vg-to-udev-rules.sh`.

You can use this script to generate `udev` rules for iSCSI Logical Volume Manager (LVM) devices, which can be automatically mounted during system boot.

Example:

- Run the script `/bin/bash /usr/share/doc/packages/lvm2/lvm-vg-to-udev-rules.sh/dev/vgname/`*`lvname`*. *`lvname`* is the name of the logical volume that is created on top of volume group vgname.
- Add the generated rules to the `/etc/udev/rules.d/85-mount-fstab-lvm.rules` file.

After completing the preceding steps for each logical volumes, the logical volumes can be automatically mounted by adding entries in `/etc/fstab`.

Multipathd occasionally fails to update the path status of DM-Multipath devices

Occasionally, multipathd does not update the path status of DM-Multipath devices after running I/O faults.

Set both `HOTPLUG_USE_HWSCAN` and `HOTPLUG_USE_SUBFS` to `no` in the `/etc/sysconfig/hotplug` file.

Multipathd fails occasionally because it fails to start "event checker" for some of the DM-Multipath devices. Because of this failure, multipathd is unable to keep track of the path up or down status for those devices.

(iSCSI) Poor performance due to misalignment of partitioned LUNs

If you experience poor performance on some or all LUNs that have partitions, the alignment between the LUN block boundaries and the underlying WAFL boundaries might not match. Manually enter the disk geometry parameters when creating partitions with the `fdisk` command.

This problem occurs only with partitions and certain combinations of disk geometry parameters. For more details, see bug number 156121 on Bugs Online, which is available on the NetApp Support Site.

Modify the filter in the `/etc/lvm/lvm.conf` file to scan only the multipath device and not the SCSI device underneath: `filter = [ "r|/dev/sd*|", "r|/dev/disk/.*|", "r|/dev/block/.*|", "a/.*/" ]`

iSCSI: LUNs not visible on the Linux host

iSCSI LUNs appear as local disks to the host. If the storage system LUNs are not available as disks on the hosts, verify the configuration settings.

Configuration setting	What you should do
Cabling	Verify that the cables between the host and the storage system are properly connected.
Network connectivity	Verify that there is TCP/IP connectivity between the host and the storage system. From the storage system command line: • Ping the host interfaces that are used for iSCSI. • Ping the storage system interfaces that are used for iSCSI.
iSCSI service status	Verify that the iSCSI service is licensed and started on the storage system. For more information, see the *Data ONTAP SAN Administration Guide for 7-Mode*.
Initiator login	Verify that the initiator is logged in to the storage system by entering the `iscsi show initiator` command on the storage system console. If the initiator is configured and logged in to the storage system, the storage system console displays the initiator node name and the target portal group to which it is connected. If the command output shows that no initiators are logged in, check the initiator configuration on the host. Verify that the storage system is configured as a target of the initiator.
iSCSI node names	Verify that you are using the correct initiator node names in the igroup configuration. On the storage system, use the `igroup show` command to display the node name of the initiators in the storage system's igroups. On the host, use the initiator tools and commands to display the initiator node name. The initiator node names configured in the igroup and on the host should match.

Configuration setting	What you should do
LUN mappings	Verify that the LUNs are mapped to an igroup. On the storage system, use one of the following commands: • `lun show -m` displays all the LUNs and the igroups to which they are mapped. • `lun show -g igroup-name` displays the LUNs mapped to a specific igroup.
System requirements	Verify that the components of your configuration are supported. Verify that you have the correct host Operating System (OS) Service Pack level, initiator version, Data ONTAP version, and other system requirements. You can check the up-to-date system requirements in the NetApp Interoperability Matrix.
Jumbo frames	If you are using jumbo frames in your configuration, ensure that the jumbo frames are enabled on all the devices in the network path: the host Ethernet NIC, the storage system, and any switches.
Firewall settings	Verify that the iSCSI port (3260) is open in the firewall rule.

Veritas Storage Foundation troubleshooting

Sometimes you might encounter a problem while running Veritas Storage Foundation with the Host Utilities. The sections that follow provide tips for resolving any issues that might occur.

(Veritas) Error while stopping the fencing driver

You might get an error when you stop the fencing driver.

Steps

1. Stop the Veritas cluster service by using `hastop` on all the nodes.
2. Try stopping the fencing driver again.
3. If the fencing driver does not stop, remove the name of the coordinator diskgroup from `/etc/vxfendg`.
4. In the `/etc/vxfenmode` file, make the following change:

   `vxfen_mode=disabled`
5. Stop the fencing driver by using the following command:

   `/etc/init.d/vxfen stop`

   Note: It is best to halt all the nodes except the last one in the cluster.

(Veritas) Secondary paths identified incorrectly

Secondary paths are identified incorrectly as primary paths while configuring VxDMP. In such a scenario, you should verify that the ASL for the storage system was installed correctly and that no error messages occurred during installation.

(Veritas) Enclosure-based naming not reflected on NetApp storage

The VxVM enclosure-based naming feature is enabled, but the VxDMP devices do not contain a storage model string. In such a scenario, verify that the ASL for the storage system was installed correctly and that no error messages occurred during installation.

(Veritas) Setting up a LUN for SAN booting fails

The process to set up a LUN for SAN booting fails if you have multiple paths configured to the boot LUN before installing the operating system. You should verify that the host can see only one path to the LUN.

(Veritas) Encapsulation of the root disk fails

Encapsulation of the root disk fails after SAN boot installation. Ensure that you follow the recommended partitioning scheme.

For more information on partitioning scheme, see the *Veritas Volume Manager Administrator's Guide*.

Installing the nSANity data collection program

Download and install the `nSANity Diagnostic and Configuration Data Collector` program when instructed to do so by your technical support representative.

About this task

The nSANity program replaces the diagnostic programs included in previous versions of the Host Utilities. The nSANity program runs on a Windows or Linux system with network connectivity to the component from which you want to collect data.

Steps

1. Log in to the NetApp Support Site and search for "nSANity".
2. Follow the instructions to download the Windows zip or Linux tgz version of the nSANity program, depending on the workstation or server you want to run it on.
3. Change to the directory to which you downloaded the zip or tgz file.
4. Extract all of the files and follow the instructions in the `README.txt` file. Also be sure to review the `RELEASE_NOTES.txt` file for any warnings and notices.

After you finish

Run the specific nSANity commands specified by your technical support representative.

Sample configuration file for Red Hat Enterprise Linux 7 series

You can use an empty `/etc/multipath.conf` file for FC, FCoE, or iSCSI configurations, as well as ALUA and non-ALUA configurations. You can also add blacklisting information for the local disks in the file, if required.

When you use a blacklist section, you must *replace* the information in the following example with information for your system.

Sample configuration file for Red Hat Enterprise Linux 7 series with and without ALUA enabled

```
# All data under blacklist must be specific to your system.
blacklist {
wwid < wwid_of_the_local_disk>
}
```

Sample configuration files for Red Hat Enterprise Linux 6 series

All versions of Red Hat Enterprise Linux 6 series use a DM-Multipath configuration file, but there might be slight variations in the file based on which Red Hat update you have installed.

You can use the sample Red Hat Enterprise Linux 6 series configuration files shown here to create your own `multipath.conf` file. When you create your file, keep the following in mind:

Red Hat Enterprise Linux 6 series notes	Explanation
Blacklist section	You **must** provide information that is specific to your system in the blacklist section. Any names shown in the sample files are examples and will not work with your system.
Blacklisting the local device	If you want to blacklist the local disk, you must put the WWID of the local disk in the blacklist section. You do not need to add other devnode lines because DM-Multipath adds them by default.
SAN boot LUNs and the `user_friendly_names` parameter	You mustset the `user_friendly_names` parameter to **no**. There have been reports of problems when this parameter is set to **yes**.

SAN boot LUNs on Red Hat Enterprise Linux 6 series and `user_friendly_names` parameter

If you create a SAN boot LUN and the installer sets the `user_friendly_names` parameter to **yes**, you **must** perform the following steps:

1. For all versions except for Red Hat Enterprise Linux 6.4 series, change the `user_friendly_names` parameter to **no**.
   For Red Hat Enterprise Linux 6.4 series, create an empty `multipath.conf` file; all the settings for both RHEL 6.4 with and without ALUA are automatically updated by default.
2. Make a backup of initrd-image.
3. Re-create the initrd-image using the `mkinitrd` command.
   Red Hat Enterprise Linux 6 series and later use either the command `mkinitrd -f /boot/initrd-"`uname -r`".img `uname -r`` or the command `dracut -f`.
4. Change the root dm-multipath device name to the WWID-based device name in all of the locations that refer to the device, such as `/etc/fstab` and `/boot/grub/device.map`.
   For example, suppose the name of the root device is `/dev/mapper/mpatha` and the WWID of the device is 360a98000486e2f66426f583133796572. You must re-create the initrd-image, and then change the device name to `/dev/mapper/360a98000486e2f66426f583133796572` in `/etc/fstab`, `/boot/grub/device.map`, and any other place that that refers to device `/dev/mapper/mpatha`.
5. Append the following parameter value to the kernel for ALUA and non-ALUA to work:
   `rdloaddriver=scsi_dh_alua`

```
Example
kernel /vmlinuz-2.6.32-358.6.1.el6.x86_64 ro root=/dev/mapper/
vg_ibmx355021082-lv_root rd_NO_LUKS rd_LVM_LV=vg_ibmx355021082/
lv_root LANG=en_US.UTF-8 rd_LVM_LV=vg_ibmx355021082/lv_swap rd_NO_MD
SYSFONT=latarcyrheb-sun16 crashkernel=auto KEYBOARDTYPE=pc
KEYTABLE=us rd_NO_DM rhgb quiet rdloaddriver=scsi_dh_alua
```

6. Reboot the host.

Red Hat Enterprise Linux 6 with ALUA enabled sample configuration file

The following sample file shows values that you might supply when your host is running Red Hat Enterprise Linux 6 with ALUA enabled. Remember that if you use the blacklist section, you must *replace* the sample information with information for your system.

```
defaults {
        user_friendly_names   no
        max_fds               max
        flush_on_last_del     yes
        queue_without_daemon  no
}
# All data under blacklist must be specific to your system.
blacklist {
devnode "^hd[a-z]"
devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
devnode "^cciss.*"
}
devices {
    device {
       vendor                 "NETAPP"
       product                "LUN"
       path_grouping_policy   group_by_prio
       features               "3 queue_if_no_path pg_init_retries 50"
       prio                   "alua"
       path_checker           tur
       failback               immediate
       path_selector          "round-robin 0"
       hardware_handler       "1 alua"
       rr_weight              uniform
       rr_min_io              128
       getuid_callout         "/lib/udev/scsi_id -g -u -d /dev/%n"
}
}
```

Red Hat Enterprise Linux 6 without ALUA enabled sample configuration file

The following file provides an example of the values you need to supply when your host is running Red Hat Enterprise Linux 6 and does not have ALUA enabled.

Note: Unless you are running the iSCSI protocol and Data ONTAP operating in 7-Mode, you should have ALUA enabled.

Remember: If you use the blacklist section, you **must replace** the sample information with information for your system.

```
defaults {
        user_friendly_names   no
        max_fds               max
        flush_on_last_del     yes
        queue_without_daemon  no
}
# All data under blacklist must be specific to your system.
blacklist {
devnode "^hd[a-z]"
devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
```

```
devnode "^cciss.*"
}
devices {
    device {
       vendor                  "NETAPP"
       product                 "LUN"
       path_grouping_policy    group_by_prio
       features                "3 queue_if_no_path pg_init_retries 50"
       prio                    "ontap"
       path_checker            tur
       failback                immediate
       path_selector           "round-robin 0"
       hardware_handler        "0"
       rr_weight               uniform
       rr_min_io               128
       getuid_callout          "/lib/udev/scsi_id -g -u -d /dev/%n"
}
}
```

Red Hat Enterprise Linux 6 update 1 with ALUA enabled sample configuration file

The following file provides an example of the values you need to supply when your host is running Red Hat Enterprise Linux 6 update 1 with ALUA enabled.

Remember: If you use the blacklist section, you **must replace** the sample information with information for your system.

```
defaults {
        user_friendly_names   no
        max_fds               max
        flush_on_last_del     yes
        queue_without_daemon  no
        dev_loss_tmo          2147483647
        fast_io_fail_tmo      5
}
# All data under blacklist must be specific to your system.
blacklist {
devnode "^hd[a-z]"
devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
devnode "^cciss.*"
}
devices {
    device {
       vendor                  "NETAPP"
       product                 "LUN"
       path_grouping_policy    group_by_prio
       features                "3 queue_if_no_path pg_init_retries 50"
       prio                    "alua"
       path_checker            tur
       failback                immediate
       path_selector           "round-robin 0"
       hardware_handler        "1 alua"
       rr_weight               uniform
       rr_min_io               128
       getuid_callout          "/lib/udev/scsi_id -g -u -d /dev/%n"
}
}
```

Red Hat Enterprise Linux 6 update 1 without ALUA enabled sample configuration file

The following file provides an example of the values you need to supply when your host is running Red Hat Enterprise Linux 6 update 1 and does not have ALUA enabled.

Note: Unless you are running the iSCSI protocol and Data ONTAP operating in 7-Mode, you should have ALUA enabled.

Remember: If you use the blacklist section, you **must replace** the sample information with information for your system.

```
defaults {
        user_friendly_names   no
        max_fds               max
        flush_on_last_del     yes
        queue_without_daemon  no
        dev_loss_tmo          2147483647
        fast_io_fail_tmo      5
}
# All data under blacklist must be specific to your system.
blacklist {
devnode "^hd[a-z]"
devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
devnode "^cciss.*"
}
devices {
    device {
       vendor                 "NETAPP"
       product                "LUN"
       path_grouping_policy   group_by_prio
       features               "3 queue_if_no_path pg_init_retries 50"
       prio                   "ontap"
       path_checker           tur
       failback               immediate
       path_selector          "round-robin 0"
       hardware_handler       "0"
       rr_weight              uniform
       rr_min_io              128
       getuid_callout         "/lib/udev/scsi_id -g -u -d /dev/%n"
}
}
```

Red Hat Enterprise Linux 6 update 2 with ALUA enabled sample configuration file

The following file provides an example of the values you need to supply when your host is running Red Hat Enterprise Linux 6 update 2 with ALUA enabled.

Remember: If you use the blacklist section, you **must replace** the sample information with information for your system.

```
defaults {
        user_friendly_names   no
        max_fds               max
        flush_on_last_del     yes
        queue_without_daemon  no
        dev_loss_tmo          infinity
        fast_io_fail_tmo      5
}
# All data under blacklist must be specific to your system.
blacklist {
devnode "^hd[a-z]"
devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
devnode "^cciss.*"
}
devices {
    device {
       vendor                 "NETAPP"
       product                "LUN"
       path_grouping_policy   group_by_prio
       features               "3 queue_if_no_path pg_init_retries 50"
       prio                   "alua"
       path_checker           tur
       failback               immediate
       path_selector          "round-robin 0"
       hardware_handler       "1 alua"
       rr_weight              uniform
```

```
        rr_min_io               128
        getuid_callout          "/lib/udev/scsi_id -g -u -d /dev/%n"
}
}
```

Red Hat Enterprise Linux 6 update 2 without ALUA enabled sample configuration file

The following file provides an example of the values you need to supply when your host is running Red Hat Enterprise Linux 6 update 2 and does not have ALUA enabled.

Note: Unless you are running the iSCSI protocol and Data ONTAP operating in 7-Mode, you should have ALUA enabled.

Remember: If you use the blacklist section, you **must replace** the sample information with information for your system.

```
defaults {
        user_friendly_names   no
        max_fds               max
        flush_on_last_del     yes
        queue_without_daemon  no
        dev_loss_tmo          infinity
        fast_io_fail_tmo      5
}
# All data under blacklist must be specific to your system.
blacklist {
devnode "^hd[a-z]"
devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
devnode "^cciss.*"
}
devices {
    device {
       vendor                  "NETAPP"
       product                 "LUN"
       path_grouping_policy    group_by_prio
       features                "3 queue_if_no_path pg_init_retries 50"
       prio                    "ontap"
       path_checker            tur
       failback                immediate
       path_selector           "round-robin 0"
       hardware_handler        "0"
       rr_weight               uniform
       rr_min_io               128
       getuid_callout          "/lib/udev/scsi_id -g -u -d /dev/%n"
}
}
```

Red Hat Enterprise Linux 6 update 3 with ALUA enabled sample configuration file

The following file provides an example of the values you need to supply when your host is running Red Hat Enterprise Linux 6 update 3 with ALUA enabled. By default, the hardware table sets the rest of the parameters.

```
devices {
    device {
       vendor                  "NETAPP"
       product                 "LUN.*"
       prio                    "alua"
       hardware_handler        "1 alua"
 }
}
```

Red Hat Enterprise Linux 6 update 3 without ALUA enabled sample configuration file

When you are not using ALUA, you only need to list any devices that must be blacklisted. All the other parameter values are set by the hardware table.

Note: Unless you are running the iSCSI protocol and Data ONTAP operating in 7-Mode, you should have ALUA enabled.

If you are using a SAN boot LUN and must blacklist the local disk, you must supply the WWID of the local disk. You do not need to add other devnode information. DM-Multipath adds that information by default.

Remember: When you use the blacklist section, you **must replace** the sample information with information for your system.

```
# All data under blacklist must be specific to your system.
blacklist {
     wwid < wwid_of_the_local_disk>
}
```

Red Hat Enterprise Linux 6 update 4, 5, 6, 7, and 8 with and without ALUA enabled sample configuration file

Refer to the steps listed in section “Red Hat Enterprise Linux 6 series, SAN boot LUNs, and user_friendly_names parameter” for setting the parameters, and then continue to the steps 1 and 2.

Remember: When you use the blacklist section, you **must replace** the sample information with information for your system.

```
# All data under blacklist must be specific to your system.
blacklist {
     wwid < wwid_of_the_local_disk>
}
```

1. Append the following parameter value to the kernel for ALUA and non-ALUA to work, and then reboot:

   `rdloaddriver=scsi_dh_alua`

   Example

   ```
   kernel /vmlinuz-2.6.32-358.6.1.el6.x86_64 ro root=/dev/mapper/
   vg_ibmx355021082-lv_root rd_NO_LUKS rd_LVM_LV=vg_ibmx355021082/
   lv_root LANG=en_US.UTF-8 rd_LVM_LV=vg_ibmx355021082/lv_swap rd_NO_MD
   SYSFONT=latarcyrheb-sun16 crashkernel=auto KEYBOARDTYPE=pc
   KEYTABLE=us rd_NO_DM rhgb quiet rdloaddriver=scsi_dh_alua
   ```

2. Verify the output of the `cat /proc/cmdline` command to ensure that the setting is complete.

Sample configuration file for Red Hat Enterprise Linux 5

All versions of Red Hat Enterprise Linux 5 series use a DM-Multipath configuration file, but there might be slight variations in the file based on which Red Hat update you have installed. You can replace your current file with the sample file, and then change the values to ones that are appropriate for your system.

You can use the sample Red Hat Enterprise Linux 5 series configuration files shown here to create your own `multipath.conf` file. When you create your file, keep the following in mind:

Red Hat Enterprise Linux 5 series notes	Explanation
Blacklist section	You **must** provide information that is specific to your system in the blacklist section. Any names shown in the sample files are examples and will not work with your system.
SAN boot LUNs and the `user_friendly_names` parameter	NetApp recommends that you set the `user_friendly_names` parameter to **no**. There have been reports of problems when this parameter is set to **yes**.
Red Hat Enterprise Linux 5 prior to Update 6	If you are using a version of Red Hat Enterprise Linux 5 series prior to update 6, check the *Recommended Host Settings for Linux Host Utilities* to see whether there are any parameter values specific to that version.

SAN boot LUNs on Red Hat Enterprise Linux 5 series and user_friendly_names parameter

If you create a SAN boot LUN and the installer sets the `user_friendly_names` parameter to **yes**, you **must** perform the following steps:

1. Change the `user_friendly_names` parameter to **no**.
2. Make a backup of initrd-image.
3. Re-create the initrd-image by using the `mkinitrd` command.
   Use the following command line: `mkinitrd -f /boot/initrd-"`uname -r`".img `uname -r``
4. Change the root dm-multipath device name to the WWID-based device name in all the locations that refer to the device, such as `/etc/fstab` and `/boot/grub/device.map`.
   For example, suppose the name of the root device is `/dev/mapper/mpatha` and the WWID of the device is 360a98000486e2f66426f583133796572. You must re-create the initrd-image, and then change the device name to `/dev/mapper/360a98000486e2f66426f583133796572` in `/etc/fstab` and `/boot/grub/device.map`, as well as any other place that refers to the `/dev/mapper/mpatha` device.
5. Reboot the host.

Red Hat Enterprise Linux 5 update 11,10, 9, 8 or update 7 with ALUA enabled sample configuration file

The following file provides an example of the values you need to supply when your host is running Red Hat Enterprise Linux 5 with update 10, 9, 8 or update 7 and has ALUA enabled.

Note: Both Red Hat Enterprise Linux 5 update 11, 10, 9, 8 and Red Hat Enterprise Linux 5 update 7 use the same values in the DM-Multipath configuration file, so this file can apply to either version.

Remember: If you use the blacklist section, you **must replace** the sample information with information for your system.

```
defaults {
        user_friendly_names   no
        queue_without_daemon  no
        flush_on_last_del     yes
        max_fds               max
        pg_prio_calc           avg
}
# All data under blacklist must be specific to your system.
blacklist {
devnode "^hd[a-z]"
devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
devnode "^cciss.*"
}
devices {
device {
       vendor                 "NETAPP"
       product                "LUN"
       path_grouping_policy   group_by_prio
       features               "3 queue_if_no_path pg_init_retries 50"
       prio_callout           "/sbin/mpath_prio_alua /dev/%n"
       path_checker           tur
       path_selector          "round-robin 0"
       failback               immediate
       hardware_handler       "1 alua"
       rr_weight              uniform
       rr_min_io              128
       getuid_callout         "/sbin/scsi_id -g -u -s /block/%n"
}
}
```

Red Hat Enterprise Linux 5 update 11,10, 9, 8, or update 7 without ALUA enabled sample configuration file

The following file provides an example of the values you need to supply when your host is running Red Hat Enterprise Linux 5 with update 11,10, 9, 8 or update 7 and does not have ALUA enabled.

Note: Unless you are running the iSCSI protocol and Data ONTAP operating in 7-Mode, you should have ALUA enabled.

Remember: If you use the blacklist section, you **must replace** the sample information with information for your system.

```
defaults {
        user_friendly_names   no
        queue_without_daemon  no
        flush_on_last_del     yes
        max_fds               max
        pg_prio_calc           avg
}
# All data under blacklist must be specific to your system.
blacklist {
devnode "^hd[a-z]"
devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
```

```
devnode "^cciss.*"
}
devices {
device {
       vendor                  "NETAPP"
       product                 "LUN"
       path_grouping_policy    group_by_prio
       features                "3 queue_if_no_path pg_init_retries 50"
       prio_callout            "/sbin/mpath_prio_ontap /dev/%n"
       path_checker            tur
       path_selector           "round-robin 0"
       failback                immediate
       hardware_handler        "0"
       rr_weight               uniform
       rr_min_io               128
       getuid_callout          "/sbin/scsi_id -g -u -s /block/%n"
}
}
```

Red Hat Enterprise Linux 5 update 6 with ALUA enabled sample configuration file

The following file provides an example of the values you need to supply when your host is running Red Hat Enterprise Linux 5 with update 6 and has ALUA enabled:

Remember: If you use the blacklist section, you **must replace** the sample information with information for your system.

```
defaults {
        user_friendly_names   no
        queue_without_daemon  no
        flush_on_last_del     yes
        max_fds               max
        pg_prio_calc             avg
}
# All data under blacklist must be specific to your system.
blacklist {
devnode "^hd[a-z]"
devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
devnode "^cciss.*"
}
devices {
device {
       vendor                  "NETAPP"
       product                 "LUN"
       path_grouping_policy    group_by_prio
       features                "1 queue_if_no_path"
       prio_callout            "/sbin/mpath_prio_alua /dev/%n"
       path_checker            directio
       path_selector           "round-robin 0"
       failback                immediate
       hardware_handler        "1 alua"
       rr_weight               uniform
       rr_min_io               128
       getuid_callout          "/sbin/scsi_id -g -u -s /block/%n"
}
}
```

Red Hat Enterprise Linux 5 update 6 without ALUA enabled sample configuration file

The following file provides an example of the values you need to supply when your host is running Red Hat Enterprise Linux 5 with update 6 and does not have ALUA enabled.

Note: Unless you are running the iSCSI protocol and Data ONTAP operating in 7-Mode, you should have ALUA enabled.

Remember: If you use the blacklist section, you **must replace** the sample information with information for your system.

```
defaults {
        user_friendly_names   no
        queue_without_daemon  no
        flush_on_last_del     yes
        max_fds               max
        pg_prio_calc             avg
}
# All data under blacklist must be specific to your system.
blacklist {
devnode "^hd[a-z]"
devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
devnode "^cciss.*"
}
devices {
device {
       vendor                  "NETAPP"
       product                 "LUN"
       path_grouping_policy    group_by_prio
       features                "1 queue_if_no_path"
       prio_callout            "/sbin/mpath_prio_ontap /dev/%n"
       path_checker            directio
       path_selector           "round-robin 0"
       failback                immediate
       hardware_handler        "0"
       rr_weight               uniform
       rr_min_io               128
       getuid_callout          "/sbin/scsi_id -g -u -s /block/%n"
}
}
```

Sample configuration file for Red Hat Enterprise Linux 4

All versions of Red Hat Enterprise Linux 4 series use a DM-Multipath configuration file, but there might be slight variations in the file based on which Red Hat update you have installed. You can replace your current file with this sample file and change the values to ones that are appropriate for your system.

Configuration Notes for DM-Multipath

Please review the following configuration notes before you set up your configuration file to enable DM-Multipath on Red Hat Enterprise Linux 4 series:

Red Hat Enterprise Linux 4 series version	Parameter Notes
4 Update 7 or later	• Use the options `flush_on_last_del` and `max_fds` • Set `pathchecker` to `directio`
(iSCSI only) 4 Update 7 and earlier	In environments running iSCSI, set `path_grouping_policy` to `multibus`
4 Update 7 and earlier	Place `rr_min_io` in the default section, not the device section, of the multipath.conf file and set its value to `128`.
4 Update 6 and earlier	Set `pathchecker` to `readsector0 .`

Red Hat Enterprise Linux 4 Update 9 sample configuration file

The following file provides an example of the values you need to supply when your host is running Red Hat Enterprise Linux 4 Update 9.

Remember: If you use the blacklist section, you must **replace** the sample information with information for your system.

```
defaults
{
   user_friendly_names   no
   queue_without_daemon  no
   max_fds               max
   flush_on_last_del     yes
}
# All data under blacklist must be specific to your system.
devnode_blacklist
{

   devnode "^hd[a-z]"
   devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
   devnode "^cciss.*"
}
devices
{
   device
   {
     vendor                "NETAPP"
     product               "LUN"
     getuid_callout        "/sbin/scsi_id -g -u -s /block/%n"
     prio_callout          "/sbin/mpath_prio_ontap /dev/%n"
```

```
        features                "1 queue_if_no_path"
        hardware_handler        "0"
        path_grouping_policy    group_by_prio
        path_selector           "round-robin 0"
        failback                immediate
        rr_weight               uniform
        rr_min_io               128
        path_checker            directio
        }
}
```

Sample configuration file for Oracle Linux Unbreakable Enterprise Kernel and Red Hat Compatible Kernel

All versions of Oracle Linux use a DM-Multipath configuration file, but there might be slight variations in the file based on which Oracle update you have installed. You can replace your current file with the sample file, and then change the values to ones that are appropriate for your system.

- For Unbreakable Enterprise Kernel, the kernel and DM-Multipath versions must be supported, per the Interoperability Matrix (IMT).
  If they are not supported, the versions must be installed from Oracle ULN (Unbreakable Linux Network).
- For Red Hat Compatible Kernel and multipath packages, you must use the RHEL-based setting.

Sample configuration file for Oracle Linux Unbreakable Enterprise Kernel

All versions of Oracle Linux Unbreakable Enterprise Kernel (UEK) 5 series use a DM-Multipath configuration file, but there might be slight variations in the file based on which UEK update you have installed. You can replace your current file with the sample file, and then change the values to ones that are appropriate for your system.

You can use the sample Oracle Linux (UEK) 5 series configuration files to create your own `multipath.conf` file. When you create your file, keep the following in mind:

Oracle Linux Unbreakable Enterprise Kernel Series notes	Explanation
Blacklist section	You **must** provide information that is specific to your system in the blacklist section. Any names shown in the sample files are examples and will not work with your system.
SAN boot LUNs and the `user_friendly_names` parameter	NetApp recommends that you set the `user_friendly_names` to **no**. There have been reports of problems when this parameter is set to **`yes`**

SAN boot LUNs on Oracle Linux 5 series and the user_friendly_names parameter

1. Change the `user_friendly_names` parameter to **no**.
2. Make a backup of initrd-image.
3. Re-create the initrd-image by using the `mkinitrd` command:
   `mkinitrd -f /boot/initrd-"`uname -r`".img `uname -r``
4. Change the root dm-multipath device name to the WWID-based device name in all of the locations that refer to the device, such as `/etc/fstab` and `/boot/grub/device.map`.
   For example, suppose the name of the root device is `/dev/mapper/mpatha` and the WWID of the device is 360a98000486e2f66426f583133796572. You must re-create the initrd-image and then change the device name to `/dev/mapper/360a98000486e2f66426f583133796572` in `/etc/fstab` and `/boot/grub/device.map`, as well as any other place that refers to device `/dev/mapper/mpatha`.
5. Reboot the host.

Oracle Linux 5 (UEK) update 11, 10, 9, and 8 with ALUA enabled sample configuration file

The following screen illustration shows sample values supplied when your host is running Oracle Linux 5 (UEK) with update 11, 10, 9, or 8 and has ALUA enabled. For Oracle Linux 5 (UEK) update 11, 10, 9, and 8, use the same values in the DM-Multipath configuration file, so that this file can apply to all versions.

Note: If you use the blacklist section, you *must replace* the sample information with information for your system.

```
defaults {
    queue_without_daemon     no
    flush_on_last_del     yes
    max_fds     max
    user_friendly_names     no
}
blacklist {
    wwid (35000c50072648313)
    devnode "^cciss.*"
    devnode "^hd[a-z]"
    devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
}
devices {
device {
    hardware_handler     "1 alua"
    prio     "alua"
    product     "LUN.*"
    vendor     "NETAPP"
}
}
```

Oracle Linux 5 (UEK) update 11, 10, 9, and 8 without ALUA enabled sample configuration file

The following file provides an example of the values you need to supply when your host is running Oracle Linux 5 (UEK) with update 11, 10, 9, and 8 and does not have ALUA enabled.

Note: Unless you are running the iSCSI protocol and Data ONTAP operating in 7-Mode, you should have ALUA enabled.

Remember: If you use the blacklist section, you **must replace** the sample information with information for your system.

```
defaults {
    queue_without_daemon     no
    flush_on_last_del     yes
    max_fds     max
    user_friendly_names     no
}
blacklist {
    wwid (35000c50072648313)
    devnode "^cciss.*"
    devnode "^hd[a-z]"
    devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
}
devices {
device {
    hardware_handler     "0"
    prio     "ontap"
    product     "LUN.*"
    vendor     "NETAPP"
}
}
```

Oracle Linux 6 series (Unbreakable Enterprise Kernel) with and without ALUA enabled sample configuration file

For Oracle Linux 6 series Unbreakable Enterprise Kernel, follow Red Hat 6 series sample DM Multipath configuration file.

Oracle Linux 7 series (Unbreakable Enterprise Kernel) with and without ALUA enabled sample configuration file

Remember: When you use a blacklist section, you must replace the sample information with information for your system.

```
# All data under blacklist must be specific to your system.
blacklist {
wwid < wwid_of_the_local_disk>
}
```

If you create a SAN boot LUN, you must perform the following steps:

1. For Oracle Linux 7 series (Unbreakable Enterprise Kernel), create an empty multipath.conf file; all the settings for Oracle Linux 7 series (Unbreakable Enterprise Kernel) with and without ALUA are automatically updated by default
2. Make a backup of initrd-image.
3. Re-create the initrd-image using the command `dracut -f`
4. Change the root dm-multipath device name to the WWID-based device name in all of the locations that refer to the device, such as `/etc/fstab and /boot/grub/device.map`. For example, suppose the name of the root device is `/dev/mapper/mpatha` and the WWID of the device is 360a98000486e2f66426f583133796572. You must re-create the initrd-image, and then change the device name to `/dev/mapper/360a98000486e2f66426f583133796572` in `/etc/fstab`, `/boot/grub/device.map`,
   and any other place that that refers to device `/dev/mapper/mpatha`.
5. Append the following parameter value to the kernel for ALUA and non-ALUA to work: `rdloaddriver=scsi_dh_alua`

   ```
   Example:
   kernel /vmlinuz-3.8.13-68.1.2.el6uek.x86_64 ro root=/dev/mapper/
   vg_ibmx3550m421096-lv_root rd_NO_LUKS
   rd_LVM_LV=vg_ibmx3550m421096/lv_root LANG=en_US.UTF-8 rd_NO_MD
   SYSFONT=latarcyrheb-sun16
   crashkernel=256M  KEYBOARDTYPE=pc KEYTABLE=us rd_LVM_LV=vg_ibmx3550m421096/
   lv_swap rd_NO_DM
   rhgb quiet rdloaddriver=scsi_dh_alua
   ```

6. Reboot the host.

Sample configuration file for Oracle Linux Red Hat Compatible Kernel

All versions of Oracle Linux Red Hat Compatible Kernel (RHCK) 5 series use a DM-Multipath configuration file, but there might be slight variations in the file based on which RHCK update you have installed. You can replace your current file with the sample file, and then change the values to ones that are appropriate for your system.

You can use the sample Oracle Linux 5 series (RHCK) series configuration files shown here to create your own `multipath.conf` file. When you create your file, keep the following in mind:

Oracle Linux Red Hat Compatible Kernel 5 series notes	Explanation
Blacklist Section	You **must** provide information of your system in the blacklist section. Any names shown in the sample files are examples and will not work with your system.
SAN boot LUNs and the `user_friendly_names` parameter	NetApp recommends that you set the `user_friendly_names` parameter to **no**. There have been reports of problems when this parameter is set to **yes**.
Oracle Linux 5 prior to Update 6	If you are using a version of Oracle Linux 5 series prior to update 6, check the *Recommended Host Settings for Linux Host Utilities* to see whether there are any parameter values specific to that version.

SAN boot LUNs on Oracle Linux 5 series (RHCK) and user_friendly_names parameter

If you create a SAN boot LUN and the installer sets the `user_friendly_names` parameter to **yes**, you must perform the following steps:

1. Change the `user_friendly_names` parameter to **no**.
2. Make a backup of initrd-image.
3. Re-create the initrd-image by using the `mkinitrd` command:

   ```
   mkinitrd -f /boot/initrd-"`uname -r`".img `uname -r`
   ```

4. Change the root dm-multipath device name to the WWID-based device name in all of the locations that refer to the device, such as `/etc/fstab` and `/boot/grub/device.map`.

   For example, suppose the name of the root device is `/dev/mapper/mpatha` and the WWID of the device is 360a98000486e2f66426f583133796572. You must re-create the initrd-image, and then change the device name to `/dev/mapper/360a98000486e2f66426f583133796572` in `/etc/fstab` and `/boot/grub/device.map`, as well as any other place that refers to device `/dev/mapper/mpatha`.
5. Reboot the host.

Oracle Linux Red Hat Compatible Kernel 5 update 11, 10, 9, 8 and 7 with ALUA enabled sample configuration file

The following file provides an example of the values you need to supply when your host is running Oracle Linux 5 with update 11, 10, 9, 8 and 7 and has ALUA enabled.

> **Note:** For Oracle Linux 5 (Red Hat Compatible Kernel) update 11, 10, 9, 8 and 7 use the same values in the DM- Multipath configuration file, so this file can apply to all versions.

Remember: If you use the blacklist section, you **must replace** the sample information with information for your system.

```
defaults {
user_friendly_names no
queue_without_daemon no
flush_on_last_del yes
max_fds max
pg_prio_calc avg
}
# All data under blacklist must be specific to your system.
```

```
blacklist {
devnode "^hd[a-z]"
devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
devnode "^cciss.*"
}
devices {
device {
vendor "NETAPP"
product "LUN"
path_grouping_policy group_by_prio
features "3 queue_if_no_path pg_init_retries 50"
prio_callout "/sbin/mpath_prio_alua /dev/%n"
path_checker tur
path_selector "round-robin 0"
failback immediate
hardware_handler "1 alua"
rr_weight uniform
rr_min_io 128
getuid_callout "/sbin/scsi_id -g -u -s /block/%n"
}
}
```

Oracle Linux 5 (RHCK) update 11, 10, 9, 8, and 7 without ALUA enabled sample configuration file

The following file provides an example of the values you need to supply when your host is running Oracle Linux 5 (RHCK) with update 11,10, 9, 8, and 7 and does not have ALUA enabled.

Note: For Oracle Linux 5 (RHCK) update 11, 10, 9, 8 and 7 use the same values in the DM-Multipath configuration file, so this file can apply to all version.

Remember: If you use the blacklist section, you **must replace** the sample information with information for your system.

```
defaults {
user_friendly_names no
queue_without_daemon no
flush_on_last_del yes
max_fds max
pg_prio_calc avg
}
# All data under blacklist must be specific to your system.
blacklist {
devnode "^hd[a-z]"
devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
devnode "^cciss.*"
}
devices {
device {
vendor "NETAPP"
product "LUN"
path_grouping_policy group_by_prio
features "3 queue_if_no_path pg_init_retries 50"
prio_callout "/sbin/mpath_prio_ontap /dev/%n"
path_checker tur
path_selector "round-robin 0"
failback immediate
hardware_handler "0"
rr_weight uniform
rr_min_io 128
getuid_callout "/sbin/scsi_id -g -u -s /block/%n"
}
}
```

Oracle Linux 5 update 6 (RHCK) with ALUA enabled sample configuration file

The following file provides an example of the values you need to supply when your host is running Oracle Linux 5 update 6 (RHCK) and has ALUA enabled:

Remember: If you use the blacklist section, you **must replace** the sample information with information for your system.

Oracle Linux 5 update 6 (RHCK) without ALUA enabled sample configuration file

The following file provides an example of the values you need to supply when your host is running Oracle Linux 5 update 6 (RHCK) with update 6 and does not have ALUA enabled.

Note: Unless you are running the iSCSI protocol and Data ONTAP operating in 7-Mode, you should have ALUA enabled.

Remember: If you use the blacklist section, you **must replace** the sample information with information for your system

```
defaults {
user_friendly_names no
queue_without_daemon no
flush_on_last_del yes
max_fds max
pg_prio_calc avg
}
# All data under blacklist must be specific to your system.
blacklist {
devnode "^hd[a-z]"
devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
devnode "^cciss.*"
}
devices {
device {
vendor "NETAPP"
product "LUN"
path_grouping_policy group_by_prio
features "3 queue_if_no_path pg_init_retries 50"
prio_callout "/sbin/mpath_prio_alua /dev/%n"
path_checker tur
path_selector "round-robin 0"
failback immediate
hardware_handler "1 alua"
rr_weight uniform
rr_min_io 128
getuid_callout "/sbin/scsi_id -g -u -s /block/%n"
}
}
```

Oracle Linux 6 series (RHCK) with and without ALUA enabled sample configuration file

For Oracle Linux 6 series (RHCK), Please follow Red Hat 6 series sample DM-Multipath configuration file.

Oracle Linux 7 series (RHCK) with and without ALUA enabled sample configuration file

For Oracle Linux 7 series (RHCK), Please follow Red Hat 7 series sample DM-Multipath configuration file.

Sample configuration file for Red Hat Enterprise Virtualization Hypervisor

All versions of Red Hat Enterprise Virtualization Hypervisor (RHEV) use a DM-Multipath configuration file, but there might be slight variations in the file. You can replace your current file with this sample file and change the values to ones that are appropriate for your system.

You can use the sample Red Hat Enterprise Virtualization Hypervisor configuration files to create your own `multipath.conf` file. When you create your file, keep the following in mind:

Red Hat Enterprise Virtualization Hypervisor	**Explanation**
Blacklist section	You **must** provide information that is specific to your system in the blacklist section. Any names shown in the sample files are examples and will not work with your system.
wwid <Dev ID>	Use the actual device ID. Run the `/lib/udev/scsi_id -g -u -d /dev/<sd_device>` command on the SCSI drive to be blacklisted.

Revision and private tags	The first and second lines of the file should be the revision and private tags:
	`# RHEV REVISION`
	`# RHEV PRIVATE`
	These tags tell the VDSM not to overwrite the configuration file. If there is no tag, then the configuration file will be overwritten with the defaults after server reboot/update.

`multipath.conf` file and persistence	You **must** set the `multipath.conf` file to persist across reboots by entering the following command **`persist /etc/multipath.conf`**

DM-Multipath configuration file requires command to persist across reboots

Red Hat Enterprise Virtualization Hypervisor 6.2 with ALUA enabled sample configuration file

The following file provides an example of the values you need to supply when your host is running Red Hat Enterprise Virtualization Hypervisor 6.2 with ALUA enabled:

Remember: If you use the blacklist section, you **must replace** the sample information with information for your system.

```
# RHEV REVISION 0.7
# RHEV PRIVATE
defaults {
        user_friendly_names   no
        max_fds               max
        flush_on_last_del     yes
        queue_without_daemon  no
        dev_loss_tmo          infinity
```

```
        fast_io_fail_tmo      5
}
# All data under blacklist must be specific to your system.
blacklist {
devnode "^hd[a-z]"
wwid "<wwid_of_the_local_disk>"
devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
devnode "^cciss.*"
}
devices {
    device {
       vendor                  "NETAPP"
       product                 "LUN"
       path_grouping_policy    group_by_prio
       features                "3 queue_if_no_path pg_init_retries 50"
       prio                    "alua"
       path_checker            tur
       failback                immediate
       path_selector           "round-robin 0"
       hardware_handler        "1 alua"
       rr_weight               uniform
       rr_min_io               128
       getuid_callout          "/lib/udev/scsi_id -g -u -d /dev/
%n"
          }
}
```

Red Hat Enterprise Virtualization Hypervisor 6.2 without ALUA enabled sample configuration file

The following file provides an example of the values you need to supply when your host is running Red Hat Enterprise Virtualization Hypervisor 6.2 and does not have ALUA enabled:

> **Note:** Unless you are running the iSCSI protocol and Data ONTAP operating in 7-Mode, you should have ALUA enabled.

Remember: If you use the blacklist section, you **must replace** the sample information with information for your system.

```
# RHEV REVISION 0.7
# RHEV PRIVATE
defaults {
        user_friendly_names   no
        max_fds               max
        flush_on_last_del     yes
        queue_without_daemon  no
        dev_loss_tmo          infinity
        fast_io_fail_tmo      5
}
# All data under blacklist must be specific to your system.
blacklist {
devnode "^hd[a-z]"
wwid <wwid_of_the_local_disk>"
devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
devnode "^cciss.*"
}
devices {
    device {
       vendor                  "NETAPP"
       product                 "LUN"
       path_grouping_policy    group_by_prio
       features                "3 queue_if_no_path pg_init_retries 50"
       prio                    "ontap"
       path_checker            tur
       failback                immediate
       path_selector           "round-robin 0"
       hardware_handler        "0"
       rr_weight               uniform
```

```
        rr_min_io               128
        getuid_callout          "/lib/udev/scsi_id -g -u -d /dev/
%n"
           }
}
```

Red Hat Enterprise Virtualization Hypervisor 6.3 with ALUA enabled sample configuration file

The following file provides an example of the values you need to supply when your host is running Red Hat Enterprise Virtualization Hypervisor 6.3 with ALUA enabled:

Remember: If you use the blacklist section, you **must replace** the sample information with information for your system.

```
# RHEV REVISION 0.8
# RHEV PRIVATE
defaults {
        user_friendly_names   no
        max_fds               max
        flush_on_last_del     yes
        queue_without_daemon  no
        dev_loss_tmo          infinity
        fast_io_fail_tmo      5
}
# All data under blacklist must be specific to your system.
blacklist {
wwid <wwid_of_the_local_disk>
}
devices {
    device {
       vendor                 "NETAPP"
       product                "LUN.*"
       prio                   "alua"
       hardware_handler       "1 alua"
    }
}
```

Red Hat Enterprise Virtualization Hypervisor 6.3 without ALUA enabled sample configuration file

The following file provides an example of the values you need to supply when your host is running Red Hat Enterprise Virtualization Hypervisor 6.3 without ALUA enabled:

Remember: If you use the blacklist section, you **must replace** the sample information with information for your system.

```
# RHEV REVISION 0.8
# RHEV PRIVATE
defaults {
        user_friendly_names   no
        max_fds               max
        flush_on_last_del     yes
        queue_without_daemon  no
        dev_loss_tmo          infinity
        fast_io_fail_tmo      5
}
# All data under blacklist must be specific to your system.
blacklist {
wwid <wwid_of_the_local_disk>
}
devices {
    device {
       vendor                 "NETAPP"
       product                "LUN.*"
```

```
        prio                    "ontap"
        hardware_handler        "0"
    }
}
```

Red Hat Enterprise Virtualization Hypervisor 6.4 with and without ALUA enabled sample configuration file

In Red Hat Enterprise Virtualization Hypervisor 6.4, `dm-multipath` can automatically apply ALUA and non-ALUA settings after you run `multipath.conf`; you must specify **`rdloaddriver=scsi_dh_alua`** in the kernel command line as described below.

Note: Unless you are running the iSCSI protocol and Data ONTAP operating in 7-Mode, you should have ALUA enabled

Remember: If you use the blacklist section, you **must replace** the sample information with information for your system.

1. Modify the`/etc/multipath.conf` file.

```
# RHEV REVISION 0.9
# RHEV PRIVATE

# All data under blacklist must be specific to your system.
blacklist {
wwid < wwid_of_the_local_disk>
}
```

2. Remount the initramfs partition as read-write.

```
mount -o remount,rw /dev/initramfs/live
```

3. Append `rloaddriver=scsi_dh_alua` to the kernel command line.

```
kernel /vmlinuz0 root=live:LABEL=Root ro rootfstype=auto
             rootflags=ro crashkernel=128M elevator=deadline quiet
rd_NO_LVM
             max_loop=256 rhgb rd_NO_LUKS rd_NO_MD rd_NO_DM
rdloaddriver=scsi_dh_alua
```

4. Modify the initramfs to read-only.

```
mount –o remount,ro /dev/.initramfs/live
```

5. Reboot the host.

Red Hat Enterprise Virtualization Hypervisor 6.5 with and without ALUA enabled sample configuration file

In Red Hat Enterprise Virtualization Hypervisor 6.5, `dm-multipath` can automatically apply ALUA and non-ALUA settings with the `multipath.conf`; you must specify

`rdloaddriver=scsi_dh_alua`
in the kernel command line as described below.

Note: Unless you are running iSCSI protocol and Data ONTAP operating in 7-Mode, you should have ALUA enabled.

Remember: If you use the blacklist section, you **must replace** the sample information with information from your system.

1. Modify the `/etc/multpath.conf` file.

```
# RHEV REVISION 1.0
# RHEV PRIVATE

# All data under blacklist must be specific to your system.
blacklist {
wwid < wwid_of_the_local_disk>
}
```

2. Remount the initramfs partition as read-write.

```
mount -o remount,rw /dev/initramfs/live
```

3. Append `rloaddriver=scsi_dh_alua` to the kernel command line.

```
nel /vmlinuz0 root=live:LABEL=Root ro rootfstype=auto
              rootflags=ro crashkernel=128M elevator=deadline quiet
rd_NO_LVM
              max_loop=256 rhgb rd_NO_LUKS rd_NO_MD rd_NO_DM
rdloaddriver=scsi_dh_alua
```

4. Modify the initramfs to read-only.

```
mount –o remount,ro /dev/.initramfs/live
```

5. Reboot the host.

Sample configuration file for Oracle VM 3 series

All versions of Oracle VM use a DM-Multipath configuration file, but there might be slight variations in the file based on which Oracle update you have installed. You can replace your current file with the sample file, and then change the values to ones that are appropriate for your system.

You can use the sample Oracle VM configuration files to create your own `multipath.conf` file.

When you create your file, keep the following in mind:

Oracle VM 3.0.1, 3.1.1, and 3.2 series	Parameter notes
Blacklist section	You **must** provide information that is specific to your system in the blacklist section. Any names shown in the sample files are examples and will not work with your system.
wwid <Dev ID>	Use the actual device ID. Run the `/lib/udev/scsi_id -gus /block/%n` command on the SCSI drive to be blacklisted.
Multipath settings and heartbeat settings	Multipath settings and heartbeat settings must be the same on all hypervisors in a server pool.
SAN boot support	SAN booting is supported only with Oracle VM 3.1.1 and the FC protocol.
SAN boot LUNs and the `user_friendly_names` parameter	NetApp recommends that you set the `user_friendly_names` parameter to **no**. There have been reports of problems when this parameter is set to **yes**.

Oracle VM, SAN boot LUNs, and user_friendly_names parameter

If you create a SAN boot LUN and the installer sets the `user_friendly_names` parameter to **yes**, you **must** perform the following steps:

1. Change the `user_friendly_names` parameter to **no**.
2. Make a backup of initrd-image.
3. Re-create the initrd-image by using the `mkinitrd` command.
   You could use the following command line: `mkinitrd -f /boot/initrd-"`uname -r`".img `uname -r``
4. Change the root dm-multipath device name to the WWID-based device name in all the locations that refer to the device, such as `/etc/fstab` and `/boot/grub/device.map`.
   For example, suppose the name of the root device is `/dev/mapper/mpatha` and the WWID of the device is 360a98000486e2f66426f583133796572. You must re-create the initrd-image, and then change the device name to `/dev/mapper/360a98000486e2f66426f583133796572` in `/etc/fstab` and `/boot/grub/device.map`, as well as any other place that refers to the `/dev/mapper/mpatha` device.
5. Reboot the host.

Oracle VM 3.0.1 with ALUA enabled sample configuration file

The following file provides an example of the values you need to supply when your host is running Oracle VM 3.0.1 with ALUA enabled.

Note: In addition to providing a DM-Multipath configuration file, you must also set the `O2CB_HEARTBEAT_THRESHOLD` timeout. For more information, see *(Oracle VM) Configuring the O2CB_HEARTBEAT_THRESHOLD* on page 112.

Remember: If you use the blacklist section, you **must replace** the sample information with information for your system.

```
defaults {
        user_friendly_names   no
        max_fds               max
        flush_on_last_del     no
        queue_without_daemon  no
}
# All data under blacklist must be specific to your system.
blacklist {
devnode "^hd[a-z]"
wwid "<wwid_of_the_local_disk>"
devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
devnode "^cciss.*"
}
devices {
    device {
       vendor                  "NETAPP"
       product                 "LUN"
       path_grouping_policy    group_by_prio
       features                "1 queue_if_no_path"
       prio                    "alua"
       path_checker            directio
       no_path_retry           "queue"
       failback                immediate
       hardware_handler        "1 alua"
       rr_weight               uniform
       rr_min_io               128
       getuid_callout          "/lib/udev/scsi_id -gus /block/%n"
            }
}
```

Oracle VM 3.0.1 without ALUA enabled sample configuration file

The following file provides an example of the values you need to supply when your host is running Oracle VM 3.0.1 and ALUA is not enabled.

Note: In addition to providing a DM-Multipath configuration file, you must also set the `O2CB_HEARTBEAT_THRESHOLD` timeout. For more information, see *(Oracle VM) Configuring the O2CB_HEARTBEAT_THRESHOLD* on page 112. Also, unless you are running the iSCSI protocol and Data ONTAP operating in 7-Mode, you should have ALUA enabled.

Remember: If you use the blacklist section, you **must replace** the sample information with information for your system.

```
defaults {
        user_friendly_names   no
        max_fds               max
        flush_on_last_del     no
        queue_without_daemon  no
}
# All data under blacklist must be specific to your system.
blacklist {
devnode "^hd[a-z]"
wwid "<wwid_of_the_local_disk>"
devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
devnode "^cciss.*"
}
devices {
    device {
```

```
        vendor                  "NETAPP"
        product                 "LUN"
        path_grouping_policy    group_by_prio
        features                "1 queue_if_no_path"
        prio                    "ontap"
        path_checker            directio
        no_path_retry           "queue"
        failback                immediate
        hardware_handler        "0"
        rr_weight               uniform
        rr_min_io               128
        getuid_callout          "/lib/udev/scsi_id -gus /block/%n"
             }
}
```

Oracle VM 3.1.1 with ALUA enabled sample configuration file

The following file provides an example of the values you need to supply when your host is running Oracle VM 3.1.1 with ALUA enabled.

Note: In addition to providing a DM-Multipath configuration file, you must also set the `O2CB_HEARTBEAT_THRESHOLD` timeout. For more information, see *(Oracle VM) Configuring the O2CB_HEARTBEAT_THRESHOLD* on page 112.

```
defaults {
        user_friendly_names   no
        max_fds               max
        flush_on_last_del     no
        queue_without_daemon  yes
}
# All data under blacklist must be specific to your system.
blacklist {
devnode "^hd[a-z]"
wwid "<wwid_of_the_local_disk>"
devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
devnode "^cciss.*"
}
devices {
    device {
       vendor                  "NETAPP"
       product                 "LUN.*"
       path_grouping_policy    group_by_prio
       features                "3 queue_if_no_path pg_init_retries 50"
       prio                    "alua"
       path_checker            tur
       no_path_retry           "queue"
       failback                immediate
       hardware_handler        "1 alua"
       rr_weight               uniform
       rr_min_io               128
       getuid_callout          "/lib/udev/scsi_id -gus /block/%n"
            }
}
```

Oracle VM 3.1.1 without ALUA enabled sample configuration file

The following file provides an example of the values you need to supply when your host is running Oracle VM and ALUA is not enabled.

Note: In addition to providing a DM-Multipath configuration file, you must also set the `O2CB_HEARTBEAT_THRESHOLD` timeout. For more information, see *(Oracle VM) Configuring the O2CB_HEARTBEAT_THRESHOLD* on page 112. Also, unless you are running the iSCSI protocol and Data ONTAP operating in 7-Mode, you should have ALUA enabled.

```
defaults {
        user_friendly_names   no
        max_fds               max
        flush_on_last_del     no
        queue_without_daemon  yes
}
# All data under blacklist must be specific to your system.
blacklist {
devnode "^hd[a-z]"
wwid "<wwid_of_the_local_disk>"
devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
devnode "^cciss.*"
}
devices {
    device {
       vendor                  "NETAPP"
       product                 "LUN.*"
       path_grouping_policy    group_by_prio
       features                "3 queue_if_no_path pg_init_retries 50"
       prio                    "ontap"
       path_checker            tur
       no_path_retry           "queue"
       failback                immediate
       hardware_handler        "0"
       rr_weight               uniform
       rr_min_io               128
       getuid_callout          "/lib/udev/scsi_id -gus /block/%n"
            }
}
```

Oracle VM 3.2 series with ALUA enabled sample configuration file

The following file provides an example of the values you need to supply when your host is running Oracle VM and ALUA is enabled.

Note: In addition to providing a DM-Multipath configuration file, you must also set the `O2CB_HEARTBEAT_THRESHOLD` timeout. For more information, see *(Oracle VM) Configuring the O2CB_HEARTBEAT_THRESHOLD* on page 112. Also, unless you are running the iSCSI protocol and Data ONTAP operating in 7-Mode, you should have ALUA enabled.

```
defaults {
        user_friendly_names   no
        max_fds               max
        flush_on_last_del     no
        queue_without_daemon  no
}
# All data under blacklist must be specific to your system.
blacklist {
devnode "^hd[a-z]"
wwid "<wwid_of_the_local_disk>"
devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
devnode "^cciss.*"
}
devices {
    device {
       vendor                  "NETAPP"
       product                 "LUN.*"
       path_grouping_policy    group_by_prio
       features                "3 queue_if_no_path pg_init_retries 50"
       prio                    "alua"
       path_checker            tur
       no_path_retry           "queue"
       failback                immediate
       hardware_handler        "1 alua"
       rr_weight               uniform
```

```
        rr_min_io               128
        getuid_callout          "/lib/udev/scsi_id -gus /block/%n"
             }
}
```

Oracle VM 3.2 series without ALUA enabled sample configuration file

The following file provides an example of the values you need to supply when your host is running Oracle VM and ALUA is not enabled.

Note: In addition to providing a DM-Multipath configuration file, you must also set the `O2CB_HEARTBEAT_THRESHOLD` timeout. For more information see, *(Oracle VM) Configuring the O2CB_HEARTBEAT_THRESHOLD* on page 112. Also, unless you are running the iSCSI protocol and Data ONTAP operating in 7-Mode, you should have ALUA enabled.

```
defaults {
        user_friendly_names   no
        max_fds               max
        flush_on_last_del     no
        queue_without_daemon  no
}
# All data under blacklist must be specific to your system.
blacklist {
devnode "^hd[a-z]"
wwid "<wwid_of_the_local_disk>"
devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
devnode "^cciss.*"
}
devices {
    device {
       vendor                  "NETAPP"
       product                 "LUN.*"
       path_grouping_policy    group_by_prio
       features                "3 queue_if_no_path pg_init_retries 50"
       prio                    "ontap"
       path_checker            tur
       no_path_retry           "queue"
       failback                immediate
       hardware_handler        "0"
       rr_weight               uniform
       rr_min_io               128
       getuid_callout          "/lib/udev/scsi_id -gus /block/%n"
            }
}
```

Oracle VM 3.3 and 3.4 series with and without ALUA enabled:

You can use an empty `/etc/multipath.conf` file for FC, FCoE, or iSCSI configurations, as well as ALUA and non-ALUA configurations. You can also add blacklisting information for the local disks in the file, if required.

When you use a blacklist section, you must replace the information in the following example with information for your system.

```
# All data under blacklist must be specific to your system.blacklist
{wwid <wwid_of_the_local_disk>}
```

1. Append the following parameter value to the kernel for ALUA and non-ALUA to work. Add to the file `/boot/grub/grub.conf` “rdloaddriver=scsi_dh_alua”

   For example, the module /vmlinuz-3.8.13-68.3.3.el6uek.x86_64 ro root=UUID=45f0d586-e59e-4892-b718-08084c78d0fb rd_NO_LUKS rd_NO_LVM LANG=en_US.UTF-8 rd_NO_MD SYSFONT=latarcyrheb-sun16 KEYBOARDTYPE=pc KEYTABLE=us rd_NO_DM rhgb quiet rdloaddriver=scsi_dh_alua

2. Reboot the host.

(Oracle VM) Configuring the O2CB_HEARTBEAT_THRESHOLD

When you run Oracle VM with multipathing in a clustered host environment, you need to configure the `O2CB_HEARTBEAT_THRESHOLD` to 65.

About this task

This timeout value works with the I/O layer and sets the maximum amount of time in seconds that a node waits for an I/O operation. In the heartbeat system file, the node writes to the block every two seconds. The block offset is equal to its global node number. This means that node 0 writes to the first block, node 1 to the second block, and so on. All the nodes also read the heartbeat sysfile every two seconds. As long as the timestamp keeps changing, that node is considered "alive."

Steps

1. On each hypervisor, run the command:

   ```
   service o2cb configure
   ```

2. Set `O2CB_HEARTBEAT_THRESHOLD` to 65 by entering:

   ```
   O2CB_HEARTBEAT_THRESHOLD = 65
   ```

   Heartbeat settings and multipath settings should be exactly same on all the hypervisors in a server pool.

3. Either restart the O2CB service or reboot your system.

Sample configuration files for SUSE Linux Enterprise Server 12 series

All versions of SUSE Linux Enterprise Server use a DM-Multipath configuration file, but there might be slight variations in the file based on which version and update of SUSE Linux Enterprise server you have. If you are running SUSE Linux Enterprise Server 12 series, you can replace your current file with one of the sample configuration files, and then change the values to ones that are appropriate for your system.

SUSE Linux Enterprise 12 series with and without ALUA enabled sample configuration

Note: With SUSE Linux Enterprise 12 series, you do not need the `/etc/multipath.conf` file to configure DM-Multipath on NetApp LUNs

Remember: To blacklist a device, you must add the following sample information in the `/etc/multipath.conf` file:

```
# All data under blacklist must be specific to your system.
blacklist {
wwid < wwid_of_the_local_disk>
}
```

Sample configuration files for SUSE Linux Enterprise Server 11 series

All versions of SUSE Linux Enterprise Server use a DM-Multipath configuration file, but there might be slight variations in the file based on which version and update of SUSE Linux Enterprise server you have. If you are running SUSE Linux Enterprise Server 11 series, you can replace your current file with one of the sample configuration files and then change the values to ones that are appropriate for your system.

You can use the sample SUSE Linux Enterprise Server 11 series configuration files shown here to create your own `multipath.conf` file. When you create your file, keep the following in mind:

SUSE Linux Enterprise Server 11 series notes	Explanation
Blacklist section	You **must** provide information of your system in the blacklist section. Any names shown in the sample files are examples and will not work with your system.

SUSE Linux Enterprise Server 11, SP1 with ALUA enabled sample configuration file

The following file provides an example of the values you need to supply when your host is running either SUSE Linux Enterprise Server 11 or 11 SP1 with ALUA.

Remember: If you use the blacklist section, you must **replace** the sample information with information for your system.

```
defaults
{
  user_friendly_names    no
  max_fds                max
  flush_on_last_del      yes
}
# All data under blacklist must be specific to your system.
blacklist
{

  devnode    "^hd[a-z]"
  wwid       "<wwid_of_the_local_disk>"
  devnode    "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
  devnode    "^cciss.*"
}
devices
{
   device
    {
      vendor                "NETAPP"
      product               "LUN"
      getuid_callout        "/lib/udev/scsi_id -g -u -d /dev/%n"
      prio                  "alua"
      features              "1 queue_if_no_path"
      hardware_handler      "1 alua"
      path_grouping_policy  group_by_prio
      path_selector         "round-robin 0"
      failback              immediate
      rr_weight             uniform
      rr_min_io             128
      path_checker          tur
    }
}
```

SUSE Linux Enterprise Server 11, 11 SP1 without ALUA enabled sample configuration file

The following file provides an example of the values you need to supply when your host is running SUSE Linux Enterprise Server 11 or 11 SP1 and ALUA is not enabled.

Note: Unless you are running the iSCSI protocol and Data ONTAP operating in 7-Mode, you should have ALUA enabled.

Remember: If you use the blacklist section, you must **replace** the sample information with information for your system.

```
defaults
{
  user_friendly_names    no
  max_fds                max
  flush_on_last_del      yes
}
# All data under blacklist must be specific to your system.
blacklist
{

  devnode    "^hd[a-z]"
  wwid       "<wwid_of_the_local_disk>"
  devnode    "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
  devnode    "^cciss.*"
}
devices
{
   device
    {
      vendor                 "NETAPP"
      product                "LUN"
      getuid_callout         "/lib/udev/scsi_id -g -u -d /dev/%n"
      prio                   "ontap"
      features               "1 queue_if_no_path"
      hardware_handler       "0"
      path_grouping_policy   group_by_prio
      path_selector          "round-robin 0"
      failback               immediate
      rr_weight              uniform
      rr_min_io              128
      path_checker           tur
    }
}
```

SUSE Linux Enterprise Server 11 SP2 with ALUA enabled sample configuration file

The following file provides an example of the values you need to supply when your host is running SUSE Linux Enterprise Server 11 SP2 KVM and Xen with ALUA.

Note: This configuration file applies to both SLES 11 SP2 KVM and Xen also.

Remember: If you use the blacklist section, you must **replace** the sample information with information for your system.

```
defaults
{
  user_friendly_names    no
  max_fds                max
  queue_without_daemon   no
  flush_on_last_del      yes
}
# All data under blacklist must be specific to your system.
blacklist
{
```

```
  devnode    "^hd[a-z]"
  wwid       "<wwid_of_the_local_disk>"
  devnode    "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
  devnode    "^cciss.*"
}
devices
{
   device
    {
      vendor                "NETAPP"
      product               "LUN"
      getuid_callout        "/lib/udev/scsi_id -g -u -d /dev/%n"
      prio                  "alua"
      features              "3 queue_if_no_path pg_init_retries 50"
      hardware_handler      "1 alua"
      path_grouping_policy  group_by_prio
      failback              immediate
      rr_weight             uniform
      rr_min_io             128
      path_checker          tur
    }
}
```

SUSE Linux Enterprise Server 11 SP2 without ALUA enabled sample configuration file

The following file provides an example of the values you need to supply when your host is running SUSE Linux Enterprise Server 11 SP2 KVM and Xen when ALUA is not enabled.

Note: Unless you are running the iSCSI protocol and Data ONTAP operating in 7-Mode, you should have ALUA enabled.

Note: This configuration file applies to both SLES 11 SP2 KVM and Xen also.

Remember: If you use the blacklist section, you must **replace** the sample information with information for your system.

```
defaults
{
  user_friendly_names    no
  max_fds                max
  queue_without_daemon   no
  flush_on_last_del      yes
}
# All data under blacklist must be specific to your system.
blacklist
{

  devnode    "^hd[a-z]"
  wwid       "<wwid_of_the_local_disk>"
  devnode    "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
  devnode    "^cciss.*"
}
devices
{
   device
    {
      vendor                "NETAPP"
      product               "LUN"
      getuid_callout        "/lib/udev/scsi_id -g -u -d /dev/%n"
      prio                  "ontap"
      features              "3 queue_if_no_path pg_init_retries 50"
      hardware_handler      "0"
      path_grouping_policy  group_by_prio
      failback              immediate
      rr_weight             uniform
```

```
      rr_min_io               128
      path_checker            tur
    }
}
```

SUSE Linux Enterprise 11 SP3, SP4 with and without ALUA enabled sample configuration file

Note: With SUSE Linux Enterprise 11 SP3, SP4 you do not need the `/etc/multipath.conf` file to configure DM-Multipath on NetApp LUNs.

Remember: To blacklist a device, you must add the following sample information in the `/etc/multipath.conf` file.

```
# All data under blacklist must be specific to your system.
blacklist {
wwid <wwid_of_the_local_disk>
}
```

Sample configuration file for SUSE Linux Enterprise Server 10

All versions of SUSE Linux Enterprise Server use a DM-Multipath configuration file, but there might be slight variations in the file based on which version and update of SUSE Linux Enterprise server you have. If you are running SUSE Linux Enterprise Server 10, you can replace your current file with this sample configuration files and then change the values to ones that are appropriate for your system.

Review the following configuration notes before you set up your configuration file to enable DM-Multipath on SUSE Linux Enterprise Server 10:

SUSE Linux Enterprise Server 10 series notes	Explanation
Blacklist section	You **must** provide information of your system in the blacklist section. Any names shown in the sample files are examples and will not work with your system.
SUSE Linux Enterprise Server 10 SP3 and earlier	If you are using a version of SUSE Linux Enterprise Server 10 series prior to SP4, check the *Recommended Host Settings for Linux Host Utilities* to see if there are any parameter values specific to that version.

SUSE Linux Enterprise Server 10 SP4 with ALUA enabled sample configuration file

The following file provides an example of the values you need to supply when your host is running SUSE Linux Enterprise Server 10 SP4 and has ALUA enabled.

Remember: If you use the blacklist section, you must **replace** the sample information with information for your system.

```
defaults
{
  user_friendly_names    no
  max_fds                max
  flush_on_last_del      yes
}
# All data under blacklist must be specific to your system.
blacklist
{

   devnode "^hd[a-z]"
   devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
   devnode "^cciss.*"
}
devices
{
    device
     {
      vendor                "NETAPP"
      product               "LUN"
      getuid_callout        "/sbin/scsi_id -g -u -s /block/%n"
      prio                  "alua"
      features              "1 queue_if_no_path"
      hardware_handler      "1 alua"
      path_grouping_policy  group_by_prio
      path_selector         "round-robin 0"
      failback              immediate
```

```
      rr_weight             uniform
      rr_min_io             128
      path_checker          tur
      }
}
```

SUSE Linux Enterprise Server 10 SP4 without ALUA enabled sample configuration file

The following file provides an example of the values you need to supply when your host is running SUSE Linux Enterprise Server 10 SP4 and does not have ALUA enabled.

Note: Unless you are running the iSCSI protocol and Data ONTAP operating in 7-Mode, you should have ALUA enabled.

Remember: If you use the blacklist section, you must **replace** the sample information with information for your system.

```
defaults
{
  user_friendly_names    no
  max_fds                max
  flush_on_last_del      yes
}
# All data under blacklist must be specific to your system.
blacklist
{

   devnode "^hd[a-z]"
   devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
   devnode "^cciss.*"
}
devices
{
    device
     {
      vendor                "NETAPP"
      product               "LUN"
      getuid_callout        "/sbin/scsi_id -g -u -s /block/%n"
      prio                  "ontap"
      features              "1 queue_if_no_path"
      hardware_handler      "0"
      path_grouping_policy  group_by_prio
      path_selector         "round-robin 0"
      failback              immediate
      rr_weight             uniform
      rr_min_io             128
      path_checker          tur
      }
}
```

Sample Configuration for Citrix XenServer

All versions of Citrix XenServer use a DM-Multipath configuration file, but there might be slight variations in the file based on which Citrix update you have installed. You can replace your current file with sample file and change the values to the ones that are appropriate for your system.

You can use the sample Citrix XenServer configuration files shown here to create your own `multipath.conf` file.

When you create your file, keep the following in mind:

Citrix XenServer 6 series	Parameter notes
Blacklist section	You **must** provide information of your system in the blacklist section. Any names shown in the sample files are examples and will not work with your system.
wwid <Dev ID>	Use the actual device ID. Run the `/sbin/scsi_id -g -u -s /block/%n` command on the SCSI drive to be blacklisted.
SAN boot support	SAN booting is supported only with Citrix XenServer 6 series and the FC/FCoE protocol.
SAN boot LUNs and the `user_friendly_names` parameter	NetApp recommends that you set the `user_friendly_names` parameter to `no`. There have been reports of problems when this parameter is set to `yes`.

Citrix XenServer, SAN boot LUNs, and `user_friendly_names` parameter

The recommended value for the `user_friendly_names` parameter is `no`. There have been reports of problems when this parameter is set to `yes`.

If you create a SAN boot LUN and the installer sets the `user_friendly_names` parameter to `yes`, you must perform the following steps.

1. Change the `user_friendly_names` parameter to `no`.
2. Make a backup of initrd-image.
3. Place the above multipath.conf file in the `/etc` directory.
4. Run the following command:
   ```
   service multipathd reload
   multipath -v3.
   ```
   Note: **`multipath -ll`** should reflect the changes on the SAN Boot LUN as per the changed multipath.conf file.
5. Re-create the initrd-image using the command `mkinitrd`.
   You could use the following command line:
   ```
   mkinitrd -f /boot/initrd-"`uname -r`".img `uname -r`
   ```
6. Reboot the host.

Note: Currently SAN Boot LUN does not reflect `queue_if_no_path` feature. To overcome this limitation, perform the step mentioned below:

- Run `multipath` command after boot.

  Note: To avoid running `multipath` command on every boot, add `/sbin/multipath` to `/etc/rc.d/rc.local file`.

  - Running multipath forces the unused LUNs (on which there is no Storage Repository) to get added to the XAPI Layer.
  - To remove the scsi_device ID of those unused LUNs from `mpathutil` use the following command: `# /opt/xensource/sm/mpathutil.py remove <scsi_device-id>`. Skip this step if you have no unused LUNs (where no SR is created).

Citrix XenServer 6.0 with ALUA enabled sample configuration file

The following file provides an example of the values you need to supply when your host is running Citrix XenServer 6.0 with ALUA enabled.

Remember: If you use the blacklist section, you must **replace** the sample information with the information from your system.

```
defaults{
        user_friendly_names   no
        max_fds               max
        flush_on_last_del     no
        queue_without_daemon  no
}
# All data under blacklist must be specific to your system.
blacklist {
devnode "^hd[a-z]"
wwid    "<wwid_of_the_local_disk>"
devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
devnode "^cciss.*"
}
devices {
    device {
       vendor                  "NETAPP"
       product                 "LUN"
       path_grouping_policy    group_by_prio
       features                "1 queue_if_no_path"
       prio_callout            "/sbin/mpath_prio_alua /dev/%n"
       path_checker            directio
       failback                immediate
       hardware_handler        "1 alua"
       rr_weight               uniform
       rr_min_io               128
       getuid_callout          "/sbin/scsi_id -g -u -s /block/%n"
            }
}
```

Citrix XenServer 6.0 without ALUA enabled sample configuration file

The following file provides an example of the values you need to supply when your host is running Citrix XenServer 6.0 and ALUA is not enabled.

Remember: If you use the blacklist section, you must **replace** the sample information with information for your system.

```
defaults {
        user_friendly_names   no
        max_fds               max
        flush_on_last_del     no
        queue_without_daemon  no
}
# All data under blacklist must be specific to your system.
blacklist {
```

```
devnode "^hd[a-z]"
wwid    "<wwid_of_the_local_disk>"
devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
devnode "^cciss.*"
}
devices {
    device {
       vendor                  "NETAPP"
       product                 "LUN"
       path_grouping_policy    group_by_prio
       features                "1 queue_if_no_path"
       prio callout            "/sbin/mpath_prio_ontap /dev/%n"
       path_checker            directio
       failback                immediate
       hardware_handler        "0"
       rr_weight               uniform
       rr_min_io               128
       getuid_callout          "/sbin/scsi_id -g -u -s /block/%n"
            }
}
```

Citrix XenServer 6.0.2 with ALUA enabled sample configuration file

The following file provides an example of the values you need to supply when your host is running Citrix XenServer 6.0.2 with ALUA enabled.

```
defaults {
        user_friendly_names   no
        max_fds               max
        flush_on_last_del     no
        queue_without_daemon  no
}
# All data under blacklist must be specific to your system.
blacklist {
devnode "^hd[a-z]"
wwid    "<wwid_of_the_local_disk>"
devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
devnode "^cciss.*"
}
devices {
    device {
       vendor                  "NETAPP"
       product                 "LUN"
       path_grouping_policy    group_by_prio
       features                "1 queue_if_no_path"
       prio callout            "/sbin/mpath_prio_alua /dev/%n"
       path_checker            tur
       failback                immediate
       hardware_handler        "1 alua"
       rr_weight               uniform
       rr_min_io               128
       getuid_callout          "/sbin/scsi_id -g -u -s /block/%n"
            }
}
```

Citrix XenServer 6.0.2 without ALUA enabled sample configuration file

The following file provides an example of the values you need to supply when your host is running Citrix XenXerver 6.0.2 and ALUA is not enabled.

Note: Unless you are running the iSCSI protocol and Data ONTAP operating in 7-Mode, you should have ALUA enabled.

```
defaults {
        user_friendly_names   no
        max_fds               max
        flush_on_last_del     no
        queue_without_daemon  no
}
# All data under blacklist must be specific to your system.
blacklist {
devnode "^hd[a-z]"
wwid    "<wwid_of_the_local_disk>"
devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
devnode "^cciss.*"
}
devices {
    device {
       vendor                  "NETAPP"
       product                 "LUN"
       path_grouping_policy    group_by_prio
       features                "1 queue_if_no_path"
       prio callout            "/sbin/mpath_prio_ontap /dev/%n"
       path_checker            tur
       failback                immediate
       hardware_handler        "0"
       rr_weight               uniform
       rr_min_io               128
       getuid_callout          "/sbin/scsi_id -g -u -s /block/%n"
            }
}
```

Citrix XenServer 6.1 with ALUA enabled sample configuration file

The following file provides an example of the values you need to supply when your host is running Citrix XenServer 6.1 with ALUA enabled.

Note: Remember, if you use the blacklist section, you must replace the sample information with information from your system.

```
defaults {
        user_friendly_names   no
        queue_without_daemon  no
        flush_on_last_del     no
        max_fds               max
}
blacklist {
devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
devnode "^hd[a-z]"
devnode "^cciss.*"
}
devices {
        device {
                vendor "NETAPP"
                product "LUN"
                path_grouping_policy group_by_prio
                features "1 queue_if_no_path"
                prio_callout "/sbin/mpath_prio_alua /dev/%n"
                path_checker tur
                failback immediate
                hardware_handler "1 alua"
                rr_weight uniform
                rr_min_io   128
                getuid_callout "/sbin/scsi_id -g -u -s /block/%n"
       }
}
```

Citrix XenServer 6.1 without ALUA enabled sample configuration file

The following file provides an example of the values you need to supply when your host is running Citrix XenServer 6.1 without ALUA enabled.

Note: Remember: If you use the blacklist section, you must replace the sample information with information for your system.

```
defaults {
        user_friendly_names   no
        queue_without_daemon  no
        flush_on_last_del     no
        max_fds               max
}
blacklist {
devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
devnode "^hd[a-z]"
devnode "^cciss.*"
}
devices {
        device {
                vendor "NETAPP"
                product "LUN"
                path_grouping_policy group_by_prio
                features "1 queue_if_no_path"
                prio_callout "/sbin/mpath_prio_ontap /dev/%n"
                path_checker tur
                failback immediate
                hardware_handler "0"
                rr_weight uniform
                rr_min_io   128
                getuid_callout "/sbin/scsi_id -g -u -s /block/%n"
       }
}
```

Citrix XenServer 6.2 with ALUA enabled sample configuration file

The following file provides an example of the values you need to supply when your host is running Citrix XenServer 6.2 with ALUA enabled.

Note: Remember: If you use the blacklist section, you must replace the sample information with information from your system.

```
defaults {
         flush_on_last_del              no
         dev_loss_tmo                   30
         fast_io_fail_tmo               off
}
blacklist {
   wwid     device_id_of the_device_to_be_blacklisted
   devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
   devnode "^hd[a-z]"
   devnode "^cciss.*"
}
devices {
      device {
            vendor                  "NETAPP"
            product                 "LUN.*"
            prio                    "alua"
            hardware_handler        "1 alua"
       }
}
```

Citrix XenServer 6.2 without ALUA enabled sample configuration file.

The following file provides an example of the values you need to supply when your host is running Citrix XenServer 6.2 without ALUA enabled.

```
defaults {
         flush_on_last_del              no
         dev_loss_tmo                   30
         fast_io_fail_tmo               off
}
blacklist {
   wwid   device_id_of the_device_to_be_blacklisted
   devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
   devnode "^hd[a-z]"
   devnode "^cciss.*"
}
devices {
      device {
            vendor                  "NETAPP"
            product                 "LUN.*"
            prio                    "ontap"
            hardware_handler         "0"

        }
}
```

Citrix XenServer 6.5 with and without ALUA enabled sample configuration file.

The following file provides an example of the values you need to supply when your host is running Citrix XenServer 6.5 with and without ALUA enabled.

Note: If you use the blacklist section, you must replace the sample information with information from your system.

```
blacklist {
wwid device_id_of the_device_to_be_blacklisted
devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
devnode "^hd[a-z]"
devnode "^cciss.*"
}
devices {
                device {
                vendor                  "NETAPP"
                product                 "LUNS.*"
                dev_loss_tmo            30
                fast_io_fail_tmo        off
                flush_on_last_del       no
        }
}
```

Citrix XenServer 7.0 with and without ALUA enabled sample Configuration file:

You can use an empty /etc/multipath.conf file for FC, FCoE, or iSCSI configurations, as well as ALUA and non-ALUA configurations. You can also add blacklisting information for the local disks in the file, if required.

Note: If you use the blacklist section, you must replace the sample information with information from your system.

```
# All data under blacklist must be specific to your system.
blacklist {
wwid <wwid_of_the_local_disk>
}
```

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

Index

N

O

P

R

S

T

U

V

Y