

Using Oracle ZFS Storage Appliance iSCSI LUNs in an Oracle Linux Environment

Updated January 2014

By Andrew Ness

This article describes how to configure iSCSI LUNs in an Oracle ZFS Storage Appliance and integrate them into an Oracle Linux environment.

Contents

[Introduction](#)

[Installing the Oracle Linux iSCSI Initiator](#)

[Configuring the iSCSI Initiator to Start on Reboot](#)

[Identifying the Host IQN](#)

[Setting up CHAP Authentication](#)

[Configuring the Oracle ZFS Storage Appliance Using the Browser User Interface](#)

[Defining an iSCSI Target Group](#)

[Defining an iSCSI Initiator](#)

[Defining an iSCSI Initiator Group](#)

[Defining an Oracle ZFS Storage Appliance Project](#)

[Defining an Oracle ZFS Storage Appliance LUN](#)

[Configuring the Oracle ZFS Storage Appliance Using the Command Line Interface](#)

[Defining an iSCSI Target Group](#)

[Defining an iSCSI Initiator](#)

[Defining an iSCSI Initiator Group](#)

[Defining an Oracle ZFS Storage Appliance Project](#)

[Defining an Oracle ZFS Storage Appliance LUN](#)

[Configuring a LUN for Use by the Oracle Linux Server](#)

[Conclusion](#)

Introduction

Oracle Linux comes supplied with a software iSCSI initiator package that can be configured to integrate iSCSI LUNs presented by the Oracle ZFS Storage Appliance into the Oracle Linux environment. This article describes how to install the Oracle Linux iSCSI initiator and how to configure iSCSI LUNs for access by Oracle Linux servers (which appear as "OL" in the code examples that follow) on the Oracle ZFS Storage Appliance using either its browser user interface (BUI) or the command line interface (CLI).

The first section that follows describes how to install the Oracle Linux iSCSI Initiator on an Oracle Linux server. The following two sections describe how to configure a LUN in the Oracle ZFS Storage Appliance that can be accessed by an Oracle Linux server using iSCSI protocol. Procedures are provided for using either the BUI or the CLI of the Oracle ZFS Storage Appliance. The final section describes how to prepare the LUN for use on the Oracle Linux server.

The procedures in this article assume the following:

- The `root` password for the Oracle ZFS Storage Appliance is known.
- The IP address or hostname of the Oracle ZFS Storage Appliance is known.
- The network used by the Oracle ZFS Storage Appliance has been configured.
- The Oracle ZFS Storage Appliance has pools configured with sufficient free space available.
- The `root` password for the Oracle Linux server is known.

Installing the Oracle Linux iSCSI Initiator

The Oracle Linux iSCSI initiator package is not installed by default so must be installed manually. The package can be installed using one of these options:

- In the Oracle Linux Standard Desktop Menu system, select System > Administration > Add / Remove Software and select **iscsi initiator** to add the package.
- Use the `yum(8)` command as `root` to execute a text-based installation as shown:

```
[root@ol ~]# yum install iscsi-initiator-utils
Loaded plugins: refresh-packagekit
Setting up Install Process
Resolving Dependencies
--> Running transaction check
--> Package iscsi-initiator-utils.x86_64 0:6.2.0.872-21.0.1.el6 will be
--> installed
--> Finished Dependency Resolution

Dependencies Resolved

=====
Package                Arch      Version                Repository            Size
=====
Installing:
iscsi-initiator-utils  x86_64   6.2.0.872-21.0.1.el6  ol6_u1_base          578 k

Transaction Summary
=====
Install                1 Package(s)

Total download size: 578 k
Installed size: 2.0 M
Is this ok [y/N]: y
Downloading Packages:
iscsi-initiator-utils-6.2.0.872-21.0.1.el6.x86_64.rpm | 578 kB  00:03

Running rpm_check_debug
Running Transaction Test
Transaction Test Succeeded
Running Transaction
  Installing : iscsi-initiator-utils-6.2.0.872-21.0.1.el6.x86_64 1/1

Installed:
  iscsi-initiator-utils.x86_64 0:6.2.0.872-21.0.1.el6

Complete!
[root@ol ~]#
```

Configuring the iSCSI Initiator to Start on Reboot

Once the iSCSI Initiator package is installed, configure the package to start on reboot by completing the following steps:

1. Check that the iSCSI initiator will start when the system is rebooted:

```
[root@ol ~]# chkconfig --list iscsi
iscsi          0:off 1:off 2:off 3:off 4:off 5:off 6:off
[root@ol ~]# chkconfig --list iscsid
iscsid         0:off 1:off 2:off 3:off 4:off 5:off 6:off
[root@ol ~]#
```

This example shows that both the `iscsi` and `iscsid` services are set to `off` at all run levels, so the iSCSI Initiator will not start at system reboot.

2. If the settings at run levels 3, 4 and 5 are not set to `on` for either the `iscsi` or `iscsid` service, enable them as follows:

```
[root@ol ~]# chkconfig iscsi on
[root@ol ~]# chkconfig iscsid on
[root@ol ~]# chkconfig --list iscsi
iscsi          0:off  1:off  2:on   3:on   4:on   5:on   6:off
[root@ol ~]# chkconfig --list iscsid
iscsid        0:off  1:off  2:on   3:on   4:on   5:on   6:off
[root@ol ~]# service iscsi start
[root@ol ~]# service iscsid start
```

The `iscsi` and `iscsid` services are now set to start at system reboot.

Identifying the Host IQN

The iSCSI Qualified Name (IQN) of the host is used to identify the host to the Oracle ZFS Storage Appliance. The IQN is automatically generated when the iSCSI initiator package is installed and is stored in a file in `/etc/iscsi`. To find the IQN for the host, enter:

```
[root@ol ~]# cat /etc/iscsi/initiatorname.iscsi
InitiatorName=iqn.1988-12.com.oracle:5fa4af61c4a0
```

The IQN for the host used in this example is `iqn.1988-12.com.oracle:5fa4af61c4a0`.

Setting up CHAP Authentication

In this example, CHAP is used to authenticate communication between the host and the Oracle ZFS Storage Appliance. The iSCSI Qualified Name (IQN) is used as the CHAP name for the iSCSI initiator and a secure password is used for the CHAP secret.

To configure CHAP authentication, edit the file `/etc/iscsi/iscsid.conf` to make the following changes:

1. To activate CHAP authentication, remove the `#` character from the beginning of the following line:

```
node.session.auth.authmethod = CHAP
```

2. To set the CHAP username and password, complete the following steps:

- a. Edit the lines that set the CHAP username and password to remove the `#` character from the beginning of these lines:

```
node.session.auth.username = username
node.session.auth.password = password
```

- b. Change `username` to the IQN for the host. For this example, the username is:

```
iqn.1988-12.com.oracle:5fa4af61c4a0
```

- c. Change the password to a reasonably secure password to be used for authentication. In this example, the password `CHAPsecret11` is used.

The lines now look like this:

```
node.session.auth.username = iqn.1988-12.com.oracle:5fa4af61c4a0
node.session.auth.password = CHAPsecret11
```

3. To set the username and password for all targets, complete the following steps:

- a. Removing the `#` character in front of the following lines:

```
node.session.auth.username_in = username
node.session.auth.password_in = password
```

- b. Set the username and password as shown:

```
node.session.auth.username_in = chapuser
node.session.auth.password_in = CHAPsecret22
```

Since this username and password are used for all targets, which may include more than one Oracle ZFS Storage Appliance or devices from other manufacturers, the username should be generic rather than associated with a particular IQN. For this example, the storage administrator has provided the username `chapuser`. The CHAP secret is the reasonably secure password `CHAPsecret22`.

4. Write the changes to the file and quit the editor.

Next, the Oracle ZFS Storage Appliance must be configured to recognize the Oracle Linux host and to present iSCSI LUNs to it. To use the Oracle ZFS Storage Appliance BUI to complete these configuration steps, go to the next section, "Configuring the Oracle ZFS Storage Appliance Using the Browser User Interface." To use the CLI, go to the section "Configuring the Oracle ZFS Storage Appliance Using the Command Line Interface."

Configuring the Oracle ZFS Storage Appliance Using the Browser User Interface

As a unified storage platform, the Oracle ZFS Storage Appliance supports access to block-protocol LUNs using iSCSI and Fibre Channel protocols. This section describes how to use the Oracle ZFS Storage Appliance BUI to configure the Oracle ZFS Storage Appliance to recognize the Oracle Linux host and present iSCSI LUNs to it.

To open the Oracle ZFS Storage Appliance BUI:

1. Enter an address in the address field of a Web browser that includes the IP address or hostname of the Oracle ZFS Storage Appliance:

```
https://<ip-address or hostname>:215
```

The login dialog window shown in Figure 1 is displayed.



Figure 1. Oracle ZFS Storage Appliance login

2. Enter a **Username** and **Password** and click **LOGIN**.

Defining an iSCSI Target Group

A target group is created on the Oracle ZFS Storage Appliance to define the ports and the protocol by which the LUN to be presented to the Oracle Linux server is accessed. For this example, an iSCSI target group is created that contains the LUN as an iSCSI target that is identified by the default IQN for the Oracle ZFS Storage Appliance and presented over default appliance interfaces.

Since CHAP will be used for authentication between the storage and the host, CHAP parameters are also specified in this example.

To define an iSCSI target group on the Oracle ZFS Storage Appliance, complete these steps:

1. Click **Configuration > SAN** to display the Storage Area Network (SAN) screen as shown in Figure 2.
2. Click **iSCSI** on the right hand side of the panel and **Targets** on the left hand side of the panel as shown in Figure 2.



Figure 2. iSCSI target configuration

3. Click the + icon at the left of **Targets** to display the Create iSCSI Target dialog shown in Figure 3.
4. Enter an **Alias** for the target and select the **Initiator authentication mode** (for this example, CHAP is selected).
5. Enter the **Target CHAP name** and a **Target CHAP secret** that is different from the secret used for the iSCSI initiator. For this example, chapuser and CHAPsecret22 are used. Additionally, the network interfaces over which iSCSI LUNs will be presented should be chosen.



Figure 3. Create iSCSI Target dialog window

6. Click **OK** to confirm.
7. To create an iSCSI target group that includes the iSCSI target just defined, place the cursor over the new **iscsi-Targets** entry in the left panel. The Move icon appears to the left of the entry as shown in Figure 4.



Figure 4. Selecting the iSCSI target entry

8. Click the icon and drag it to the iSCSI Target Groups panel on the right. A new entry (highlighted in yellow) appears at the bottom of the iSCSI Target Groups column as shown in Figure 5.



Figure 5. Creating an iSCSI target group

- Move the cursor over the new target group and release the mouse button. A new iSCSI target group is created with a name `targets-n`, where n is an integer, as shown in



Figure 6.



Figure 6. Selecting the new target group for editing

- Move the cursor over the entry for the new target group. Two icons appear to the right of the target group box as shown in



Figure 6.

- Click the edit icon (✎) to display the dialog in Figure 7.
- In the **Name** field, replace the default name with the name to be used for the iSCSI target group and click **OK**. For this example, the name `iscsi-TG` is used.

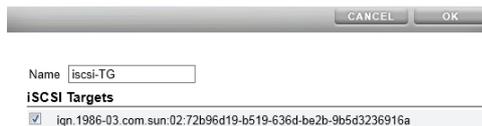


Figure 7. Renaming the iSCSI target group

- Click **APPLY**. The changes are shown in the iSCSI Targets panel on the left in Figure 8.



Figure 8. Viewing the new iSCSI target group

Defining an iSCSI Initiator

An iSCSI initiator is defined to restrict which servers have access to a particular volume. If more than one host can write to a given volume concurrently, inconsistency in file system caching between hosts can cause corruption in the on-disk image. Typically, a single initiator is given access to a volume, unless a specialized cluster file system is being used.

The iSCSI initiator serves to define the “host” from the point of view of the Oracle ZFS Storage Appliance. To identify the Oracle Linux server to the Oracle ZFS Storage Appliance, the Oracle Linux iSCSI initiator IQN must be registered with the Oracle ZFS Storage Appliance by completing the following steps:

1. Click **Configuration > SAN** to display the Storage Area Network (SAN) screen shown in Figure 9.
2. Click the **iSCSI** tab at the right and then select **Initiators** on the left panel as shown in Figure 9.



Figure 9. Selecting the iSCSI Initiators panel

3. Click the + icon to the left of **Initiators** to display the Identify iSCSI Initiator dialog shown in Figure 10.
4. Enter the **Initiator IQN** for the Oracle Linux server (see the previous section *Identifying the Host IQN* for how to obtain the Initiator IQN).
5. Enter a more meaningful symbolic name as the **Alias**.
6. If CHAP authentication has been set up (see the previous section "Setting up CHAP Authentication"), check the **Use CHAP** option. Enter the CHAP name in the **Initiator CHAP name** field and enter the password in the **Initiator CHAP secret** field as shown in Figure 10. The CHAP name and password must be identical to those defined in the host configuration.



Figure 10. Configuring a new iSCSI initiator

7. Click **OK**.

Defining an iSCSI Initiator Group

Related iSCSI initiators can be combined into logical groups to allow single commands to be executed on multiple iSCSI initiators, such as, for example, assigning LUN access to all iSCSI initiators in a group with one command. For this example, the iSCSI initiator group will contain one initiator, but in a cluster, where multiple servers are treated as a single logical entity, the initiator group may contain multiple initiators.

To create an iSCSI initiator group, complete these steps:

1. Select **Configuration > SAN** to display the Storage Area Network (SAN) screen.
2. Select the **iSCSI** tab at the right and then click **Initiators** on the left panel.

- Place the cursor over the entry for the iSCSI initiator created in the previous section. The Move icon  appears to the left of the entry as shown in Figure 11.



Figure 11. Displaying the Move icon for a new iSCSI initiator

- Click the  icon and drag it to the Initiator Groups panel on the right. A new entry (highlighted in yellow) appears at the bottom of the Initiators Groups panel as shown in Figure 12.

Move the cursor over the new entry box and release the mouse button. A new iSCSI initiator group is created with a name `initiators-n`, where *n* is an integer, as shown in Figure 13.



Figure 12. Creating an iSCSI initiator group

- Move the cursor over the entry for the new initiator group. Two icons appear to the right of the target group box as shown in Figure 13.



Figure 13. Selecting the iSCSI initiator group

- Click the edit icon  to display the dialog window in Figure 14.
- In the **Name** field, replace the default name with the name to be used for the iSCSI initiator group and click **OK**. For this example, the name `linux-servers` is used.



Figure 14. Renaming the iSCSI initiator group

- Click **APPLY** on the SAN configuration screen to confirm all the modifications, as shown in Figure 15.



Figure 15. iSCSI initiator configuration completed

Defining an Oracle ZFS Storage Appliance Project

A project can be defined in the Oracle ZFS Storage Appliance to group related volumes. A project allows inheritance of properties for file systems and LUNs presented from the project and also allows quotas and reservations to be applied.

To create a project, complete the following steps:

1. Select **Shares > Projects** to display the Projects screen as shown in Figure 16.

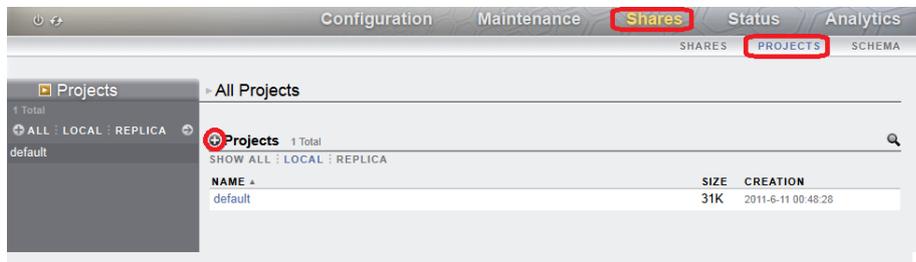


Figure 16. Viewing a project

2. Click the + icon to the left of **Projects** at the top of the left panel to display the Create Project dialog window shown in Figure 17.



Figure 17. Create Project dialog window

3. To create a new project, enter a **Name** for the project and click **APPLY**. A new project appears in the Projects list in the left panel.
4. Select the new project to view the components that comprise the project as shown in Figure 18.



Figure 18. Displaying the new project linux-project

Defining an Oracle ZFS Storage Appliance LUN

A LUN must now be created from an existing pool of storage resources, which the Oracle Linux server will access. For this example, a thin-provisioned 64 GB LUN called `DocArchive1` will be created. The LUN will be set up for an EXT3 file system.

The target group will be the iSCSI target group created in the section "Defining an iSCSI Target Group" to ensure that this LUN can be accessed using iSCSI protocol. The initiator group defined in the section "Defining an iSCSI Initiator Group" will be used to ensure that only the server(s) defined in the `linux-servers` group can access this LUN. (In this example, this group contains only one server.)

To create a target group, complete the following steps:

1. Select **Shares > Projects** to display the Projects screen.
2. In the **Projects** panel at the left, select the project. Then select **LUNs** to the right of the panel as shown in Figure 19.

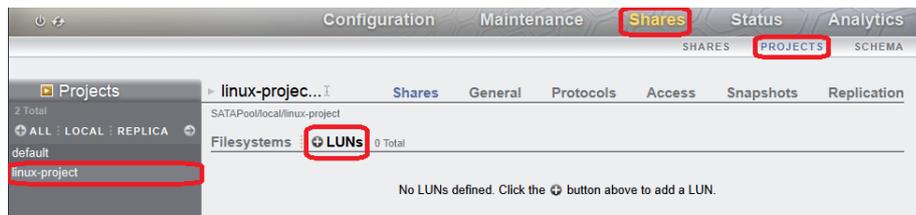


Figure 19. Viewing LUNs for the project `linux-project`

3. Click the **+** icon to the left of **LUNs** to display the Create LUN dialog shown in Figure 20.
4. Enter values as appropriate to set up the LUN. For this example, set the **Name** to `DocArchive1`, the **Volume size** to `64 GB`, and check the box next to **Thin provisioned**. Set the **Target Group** to the iSCSI target group `iscsi-TG` and the **Initiator Group** to `linux-servers`. Set the **Volume block size** to `32K` as the LUN will hold an EXT3 file system.

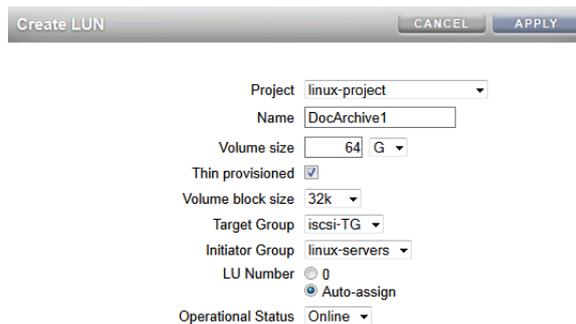


Figure 20. Create LUN dialog window

5. Click **APPLY** to create the LUN and make it available to the Oracle Linux server.

Configuring the Oracle ZFS Storage Appliance Using the Command Line Interface

As a unified storage platform, the Oracle ZFS Storage Appliance supports access to block-protocol LUNs using iSCSI and Fibre Channel protocols. This section describes how to use the Oracle ZFS Storage Appliance CLI to configure the Oracle ZFS Storage Appliance to recognize the Oracle Linux host and present iSCSI LUNs to it.

The CLI must be accessed using an SSH-enabled terminal session. On a host that has an SSH client available, enter the following:

```
user@host> ssh root@zfssa-ip-address-or-hostname
login as: root
Using keyboard-interactive authentication.
Password: XXXXX
Last login: Mon Jul XX XX:XX:XX XXXX from Linux
zfssa:>
```

Navigation follows a similar pattern in the CLI as in the BUI, with paths used in commands often corresponding to paths through tabs and screens in the BUI.

Defining an iSCSI Target Group

A target group is created on the Oracle ZFS Storage Appliance to define the ports and the protocol by which the LUN to be presented to the Oracle Linux server is accessed. For this example, an iSCSI target group is created that contains the LUN as an iSCSI target that is identified by the default IQN for the Oracle ZFS Storage Appliance and presented over default appliance interfaces.

Since CHAP will be used for authentication between the storage and the host, CHAP parameters are also specified in this example.

```
zfssa:>
configuration san targets
zfssa:configuration san targets>
iscsi

zfssa:configuration san targets iscsi>
create
zfssa:configuration san targets iscsi target (uncommitted)>
set alias=iscsi-Target
      alias = iscsi-Target (uncommitted)
zfssa:configuration san targets iscsi target (uncommitted)>
commit
zfssa:configuration san targets iscsi>
list
TARGET      ALIAS
target-000  iscsi-Target
            |
            +-> IQN
            iqn.1986-03.com.sun:02:e8589be5-144d-c9b2-89d4-f7fe4e887881

zfssa:configuration san targets iscsi>
select target-000
zfssa:configuration san targets iscsi target-000>
list
Properties:
  alias = iscsi-Target
  iqn = iqn.1986-03.com.sun:02:e8589be5-144d-c9b2-89d4-f7fe4e887881
  auth = none
  targetchapuser = (unset)
  targetchapsecret = (unset)
  interfaces = nge0

zfssa:configuration san targets iscsi target-000>
set targetchapuser=chapuser
  targetchapuser =
  iqn.1986-03.com.sun:02:e8589be5-144d-c9b2-89d4-f7fe4e887881
  (uncommitted)

zfssa:configuration san targets iscsi target-000>
set targetchapsecret=CHAPsecret22
  targetchapsecret = ***** (uncommitted)
zfssa:configuration san targets iscsi target-000>
commit
zfssa:configuration san targets iscsi target-000>
list
Properties:
  alias =
  iqn =
```

```

        auth =
        targetchapuser = \
            iqn.1986-03.com.sun:02:e8589be5-144d-c9b2-89d4-f7fe4e887881
        targetchapsecret = *****
        interfaces =

zfssa:configuration san targets iscsi target-000>
    cd ..
zfssa:configuration san targets iscsi>
    list
TARGET     ALIAS
target-000 iscsi-Target
           |
           +-> IQN
                iqn.1986-03.com.sun:02:e8589be5-144d-c9b2-89d4-f7fe4e887881

zfssa:configuration san targets iscsi>
    group
zfssa:configuration san targets iscsi groups>
    create
zfssa:configuration san targets iscsi group (uncommitted)>
    set name=iscsi-TG
                name = iscsi-TG (uncommitted)
zfssa:configuration san targets iscsi group (uncommitted)>
set targets=iqn.1986-03.com.sun:02:e8589be5-144d-c9b2-89d4-f7fe4e887881
                targets = iqn.1986-03.com.sun:02:e8589be5-144d-
c9b2-89d4-f7fe4e887881 (uncommitted)
zfssa:configuration san targets iscsi group (uncommitted)>
    commit
zfssa:configuration san targets iscsi groups>
    list
GROUP     NAME
group-000 iscsi-TG
           |
           +-> TARGETS
                iqn.1986-03.com.sun:02:e8589be5-144d-c9b2-89d4-f7fe4e887881
zfssa:configuration san targets iscsi groups>
    cd /
zfssa:>

```

Defining an iSCSI Initiator

An iSCSI initiator is defined to restrict which servers have access to a particular volume. If more than one host can write to a given volume concurrently, inconsistency in file system caching between hosts can cause corruption in the on-disk image. Typically, a single initiator is given access to a volume, unless a specialized cluster file system is being used.

The iSCSI initiator serves to define the “host” from the point of view of the Oracle ZFS Storage Appliance. To identify the Oracle Linux server to the Oracle ZFS Storage Appliance, the Linux iSCSI IQN must be registered with the Oracle ZFS Storage Appliance as shown in the following example.

```

zfssa:>
    configuration san initiators
zfssa:configuration san initiators>
    iscsi
zfssa:configuration san initiators iscsi>
    ls
Children:
                groups => Manage groups

```

```

zfssa:configuration san initiators iscsi>
    create
zfssa:configuration san initiators iscsi initiator (uncommitted)>
    set alias=OL
        alias = OL (uncommitted)
zfssa:configuration san initiators iscsi initiator (uncommitted)>
    set initiator= iqn.1988-12.com.oracle:5fa4af61c4a0
        initiator = iqn.1988-12.com.oracle:5fa4af61c4a0 (uncommitted)
zfssa:configuration san initiators iscsi initiator (uncommitted)>
    set chapuser= iqn.1988-12.com.oracle:5fa4af61c4a0
        chapuser = iqn.1988-12.com.oracle:5fa4af61c4a0 (uncommitted)
zfssa:configuration san initiators iscsi initiator (uncommitted)>
    set chapsecret=CHAPsecret11
        chapsecret = ***** (uncommitted)
zfssa:configuration san initiators iscsi initiator (uncommitted)>
    commit
zfssa:configuration san initiators iscsi> ls
Initiators:

NAME          ALIAS
initiator-000 Linux
              |
              +--> INITIATOR
                  iqn.1988-12.com.oracle:5fa4af61c4a0

Children:
              groups => Manage groups

zfssa:configuration san initiators iscsi>

```

Defining an iSCSI Initiator Group

Related iSCSI initiators can be combined into logical groups to allow single commands to be executed on multiple iSCSI initiators, such as, for example, assigning LUN access to all iSCSI initiators in a group with one command. For this example, the iSCSI initiator group will contain one initiator, but in a cluster, where multiple servers are treated as a single logical entity, the initiator group may contain multiple initiators.

```

zfssa:configuration san initiators iscsi>
    groups
zfssa:configuration san initiators iscsi groups>
    create
zfssa:configuration san initiators iscsi group (uncommitted)>
    set name=linux-servers
        name = linux-servers (uncommitted)
zfssa:configuration san initiators iscsi group (uncommitted)>
    set initiators= iqn.1988-12.com.oracle:5fa4af61c4a0
        initiators = iqn.1988-12.com.oracle:5fa4af61c4a0 (uncommitted)
zfssa:configuration san initiators iscsi group (uncommitted)>
    commit
zfssa:configuration san initiators iscsi groups >
    ls
Groups:

GROUP        NAME
group-000    linux-servers
              |
              +--> INITIATORS
                  iqn.1988-12.com.oracle:5fa4af61c4a0

```

```
zfssa:configuration san initiators iscsi groups>
    cd /
zfssa:>
```

Defining an Oracle ZFS Storage Appliance Project

A project can be defined in the Oracle ZFS Storage Appliance to group related volumes. A project allows inheritance of properties for file systems and LUNs presented from the project and also allows quotas and reservations to be applied.

```
zfssa:>
    shares
zfssa:shares>
    project linux-project
zfssa:shares linux-project (uncommitted)>
    commit
zfssa:shares>
    cd /
zfssa:>
```

Defining an Oracle ZFS Storage Appliance LUN

A LUN must now be created from an existing pool of storage resources, which the Oracle Linux server will access. For this example, a thin-provisioned 64 GB LUN called `DocArchive1` will be created. The LUN will be set up for an EXT3 file system.

The target group will be the iSCSI target group created in the section "Defining an iSCSI Target Group" to ensure that this LUN can be accessed using iSCSI protocol. The initiator group defined in the section "Defining an iSCSI Initiator Group" will be used to ensure that only the server(s) defined in the `linux-servers` group can access this LUN. (In this example, this group contains only one server.)

```
zfssa:shares>
    select linux-project
zfssa:shares linux-project>
    lun DocArchive
zfssa:shares linux-project/DocArchive (uncommitted)>
    set volsize=64G
        volsize = 64G (uncommitted)
zfssa:shares linux-project/DocArchive (uncommitted)>
    set targetgroup=iscsi-TG
        targetgroup = iscsi-TG (uncommitted)
zfssa:shares linux-project/DocArchive (uncommitted)>
    set initiatorgroup=linux-servers
        initiatorgroup = linux-servers (uncommitted)
zfssa:shares linux-project/DocArchive (uncommitted)>
    list
Properties:
    checksum = fletcher4 (inherited)
    compression = off (inherited)
    dedup = false (inherited)
    copies = 1 (inherited)
    logbias = latency (inherited)
    secondarycache = all (inherited)
    volblocksize = 8K (default)
    volsize = 64G (uncommitted)
    sparse = false (default)
    exported = true (inherited)
    targetgroup = iscsi-TG (uncommitted)
```

```

        initiatorgroup = linux-servers (uncommitted)
        lunumber = (default)
        assignednumber = (default)
        status = (default)
        fixednumber = (default)
        lunguid = (default)
        canonical_name = (default)
        nodestroy = (default)

Children:
        snapshots => Manage snapshots
        replication => Manage remote replication

zfssa:shares linux-project/DocArchive (uncommitted)>
    set volblocksize=32k
        volblocksize = 32K (uncommitted)
zfssa:shares linux-project/DocArchive (uncommitted)>
    commit
zfssa:shares linux-project>
    cd /
zfssa:shares>

```

Configuring a LUN for Use by the Oracle Linux Server

Now that the LUN is prepared and available to an authenticated iSCSI initiator, the LUN must be configured for use by the Oracle Linux server by completing the following steps:

1. Initiate an Oracle Linux iSCSI session with the Oracle ZFS Storage Appliance. Since the LUN was created prior to initiating the iSCSI session, it will be automatically enabled.

```

[root@ol ~]# service iscsi start
[root@ol ~]# iscsiadm -m discovery -t sendtargets -p zfssa
192.168.1.12:3260,2 iqn.1986-03.com.sun:02:4ec2b85a-9b5d-e02c-ee9c-
bdd55452e7de
[fe80::214:4fff:fe0f:92b4]:3260,2 iqn.1986-03.com.sun:02:4ec2b85a-9b5d-
e02c-ee9c-bdd55452e7de
[root@ol ~]# iscsiadm -m node -p zfssa -login
[root@ol ~]# tail /var/log/messages
Jul 13 14:35:37 OL kernel: scsi21 : iSCSI Initiator over TCP/IP
Jul 13 14:35:37 OL iscsid: Could not set session19 priority. READ/WRITE
throughout and
    latency could be affected.
Jul 13 14:35:38 OL kernel: scsi 21:0:0:0: Direct-Access      SUN
Sun Storage NAS  1.0
        PQ: 0 ANSI: 5
Jul 13 14:35:38 OL kernel: sd 21:0:0:0: Attached scsi generic sg2 type
0
Jul 13 14:35:38 OL kernel: sd 21:0:0:0: [sdb] 134217728 512-byte
logical blocks:
(68.7 GB/64.0 GiB)
Jul 13 14:35:38 OL kernel: sd 21:0:0:0: [sdb] Write Protect is off
Jul 13 14:35:38 OL kernel: sd 21:0:0:0: [sdb] Write cache: enabled,
read cache: enabled,
doesn't support DPO or FUA
Jul 13 14:35:38 OL kernel: sdb: unknown partition table
Jul 13 14:35:39 OL kernel: sd 21:0:0:0: [sdb] Attached SCSI disk
Jul 13 14:35:39 OL iscsid: Connection19:0 to [target: iqn.1986-
03.com.sun:02:4ec2b85a-9b5d-
e02c-ee9c-bdd55452e7de, portal: 192.168.1.12,3260] through [iface:
default] is
operational now
[root@ol ~]# ls -l /dev/sd?
brw-rw----. 1 root disk 8,  0 Jul 13 13:06 /dev/sda
brw-rw----. 1 root disk 8, 16 Jul 13 14:35 /dev/sdb

```

2. Prepare the LUN by creating a Linux partition table using the `fdisk(8)` command.

```
root@ol:~# fdisk /dev/sdb
Device contains neither a valid DOS partition table, nor Sun, SGI or
OSF disklabel
Building a new DOS disklabel with disk identifier 0xda0cb962.
Changes will remain in memory only, until you decide to write them.
After that, of course, the previous content won't be recoverable.

Warning: invalid flag 0x0000 of partition table 4 will be corrected by
w(rite)

WARNING: DOS-compatible mode is deprecated. It's strongly recommended
to switch off the mode (command 'c') and change display units to
sectors (command 'u').

Command (m for help): p

Disk /dev/sdb: 68.7 GB, 68719476736 bytes
255 heads, 63 sectors/track, 8354 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0xda0cb962

   Device Boot      Start         End      Blocks   Id  System

```

Device	Boot	Start	End	Blocks	Id	System
--------	------	-------	-----	--------	----	--------

```
Command (m for help): n
Command action
   e   extended
   p   primary partition (1-4)
P
Partition number (1-4): 1
First cylinder (1-8354, default 1): 1
Last cylinder, +cylinders or +size(K,M,G) (1-8354, default 8354): 8354

Command (m for help): w
The partition table has been altered!

Calling ioctl() to re-read partition table.
Syncing disks.
```

3. This device is now available for selection similar to an internal server disk. Examples of disk deployment include:

- Initializing the device as a physical volume for a logical volume manager (LVM) configuration
- Configuring the device as storage for a large EXT3 file system without any intervening virtualization as shown in the following example:

```
[root@ol ~]# mkfs.ext3 -L DocArchive /dev/sdb1
mke2fs 1.41.12 (17-May-2010)
Filesystem label=DocArchive
OS type: Linux
Block size=4096 (log=2)
Fragment size=4096 (log=2)
Stride=0 blocks, Stripe width=0 blocks
4194304 inodes, 16775868 blocks
838793 blocks (5.00%) reserved for the super user
First data block=0
Maximum filesystem blocks=4294967296
512 block groups
32768 blocks per group, 32768 fragments per group
```

```

8192 inodes per group
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632,
    2654208, 4096000, 7962624, 11239424

Writing inode tables: done
Creating journal (32768 blocks): done
Writing superblocks and filesystem accounting information: done

This filesystem will be automatically checked every 27 mounts or
180 days, whichever comes first.  Use tune2fs -c or -i to override.
[root@ol ~]# vi /etc/fstab
#
# /etc/fstab
# Created by anaconda on Wed Jul 13 09:19:08 2011
#
# Accessible filesystems, by reference, are maintained under
# '/dev/disk'
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more
# info
#
/dev/mapper/vg_OL-lv_root / ext4 defaults 1 1
UUID=043800a8-30b9-4ba8-9ce2-0c114273d71b /boot ext4 defaults 1 2
/dev/mapper/vg_OL-lv_swap swap swap defaults 0 0
tmpfs /dev/shm tmpfs defaults 0 0
devpts /dev/pts devpts gid=5,mode=620 0 0
sysfs /sys sysfs defaults 0 0
proc /proc proc defaults 0 0
LABEL=DocArchive /docarchive ext3 defaults 2 2
[root@ol ~]# mkdir /docarchive
[root@ol ~]# mount /docarchive
[root@ol ~]# df -k
Filesystem      1K-blocks      Used Available Use% Mounted on
/dev/mapper/vg_OL-lv_root
                29745220    3067908  25166300  11% /
tmpfs            704376         420    703956   1% /dev/shm
/dev/sda1        495844         70382   399862  15% /boot
/dev/sdb1        66050540      184268  62511100  1% /docarchive
[root@ol ~]#

```

The final two lines of the output from the `df(1)` command show that approximately 64 GB of new space is now available.

Conclusion

This article has described how to configure the software iSCSI initiator package supplied with Oracle Linux to enable an Oracle Linux server to access iSCSI LUNs presented by the Oracle ZFS Storage Appliance.



Oracle is committed to developing practices and products that help protect the environment

Using Oracle ZFS Storage Appliance iSCSI LUNs
in an Oracle Linux Environment
January 2014, Version 1.2.0
Author: Application Integration Engineering,
Andrew Ness

Oracle Corporation
World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065
U.S.A.

Worldwide Inquiries:
Phone: +1.650.506.7000
Fax: +1.650.506.7200

oracle.com

Copyright © 2012, 2014 Oracle and/or its affiliates. All rights reserved. This document is provided for information purposes only and the contents hereof are subject to change without notice. This document is not warranted to be error-free, nor subject to any other warranties or conditions, whether expressed orally or implied in law, including implied warranties and conditions of merchantability or fitness for a particular purpose. We specifically disclaim any liability with respect to this document and no contractual obligations are formed either directly or indirectly by this document. This document may not be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without our prior written permission.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark licensed through X/Open Company, Ltd. 0611

Hardware and Software, Engineered to Work Together