# Internet Small Computer System Interface (ISCI)

In the **iSCSI** world, you’ve got two types of agents:

* an **iSCSI** **target** provides some storage (here called **server**),
* an**iSCSI** **initiator** uses this available storage (here called **client**).

Creates 2 IQN(iscsi Qualified Name) – One for Server and One for Client.

|  |  |
| --- | --- |
| **SERVER** | **CLIENT** |
| Package: targetcliv | Package: iscsi-utils |
| Service: target | Service: iscsi, iscsid |
| Firewall Port: 3260/tcp |  |
|  | Initiator: /etc/iscsi/initiatorname |

**Tool:** >> target

Most of the **target** configuration is done interactively through the **targetcli** command. This command uses a directory tree to access the different objects.

## Server Side

Install the following packages:

# **yum install -y targetcli**

Activate the **target** service at boot:

# **systemctl enable target**

Start the **target** service:

# **systemctl start target**

**Firewall**

# **firewall-cmd --permanent –add-port=3260/tcp**

# **firewall-cmd reload**

**getent services 3260**

**Create Disk**

# **fdisk /dev/vdb**

**n >> 5GB >> /dev/vdb1**

Execute the targetcli command:

**# targetcli**

**/> ls**

**o- backstores .................................................**

**o- block .......................................................**

**o- fileio............................................................**

**o- psccsi ......................................................**

**o- ramdisk ......................................................**

**o- iscsi .................................................**

**o- loopback .................................................**

Now use an existing logical disk (**/dev/vdb1**) as a block-type backing store for storage object “**data**”

/> cd backstores/block

/backstores/block> create dev=/dev/vdb1 name=data

Create a target

/backstores/block> **cd /iscsi**

/iscsi> **create wwn=iqn.2017-09.com.example.com:server4**

Created target iqn.**2017-09.com.example.com:server4**.

Created TPG 1.

Below **tpg1**, three objects have been defined:

* **acls** (**a**ccess **c**ontrol **l**ists: restrict access to resources),
* **luns** (**l**ogical **u**nit **n**umber: define exported resources),
* **portals** (define ways to reach the exported resources; consist in pairs of IP addresses and ports).

o- tpg1 ........................................ [no-gen-acls, no-auth]

o- acls ................................................... [ACLs: 0]

o- luns ................................................... [LUNs: 0]

o- portals ............................................. [Portals: 1]

Create ACL for client machine (It’s the IQN which clients use to connect).

/iscsi> **cd /iscsi/iqn.2017-09.com.example.com:server4/tpg1/acls**

/iscsi/iqn.20...sk1/tpg1/acls> **wwn=iqn.2017-09.com.example.com:desktop4**

Create a LUN under the target, The LUN should use the previously mentioned backing storage object named “data”

/iscsi/iqn.20 > **cd /iscsi/iqn.2017-09.com.example.com:server4/tpg1/luns**

/iscsi/iqn.20...sk1/tpg1/luns> **create /backstores/block/data**

Created LUN 0.

Create a portal with server IP

/iscsi/iqn.20.. > **cd /iscsi/iqn.2017-09.com.example.com:server4/tpg1/portals**

/iscsi/iqn.20...sk1/tpg1/portals> **create ip\_address=172.25.4.11**

**Save Configuration**

/iscsi/iqn.20...sk1/tpg1/portals> **cd /**

/> **saveconfig**

/> **exit**

# **systemctl restart target**

## Client Side

Install the following packages:

# **yum install -y iscsi\***

Activate the services at boot:

# **systemctl enable isci iscsid**

Start the services:

# **systemctl isci iscsid**

[root@node1 ~]# vi /etc/iscsi/initiatorname.iscsi

**InitiatorName=iqn.2017-09.com.example.com:desktop4**

Discover the target using below command.

[root@node1 ~]# iscsiadm --mode discoverydb --type sendtargets --portal 172.25.4.11 --discovery

[root@node1 ~]# systemctl restart iscsid.service

Login to the discovered target.

[root@node1 ~]# iscsiadm --mode node --targetname **iqn.2017-09.com.example.com:server4** --pprtal 172.25.4.11:3260 --login

Mount

[root@node1 ~]# lsblk

[root@node1 ~]# fdisk /dev/sda

[root@node1 ~]# mkfs.xfs /dev/sda1

[root@node1 ~]# blkid

[root@node1 ~]# mkdir /iscsi

[root@node1 ~]# vim /etc/fstab

UUID=3c... /iscsi xfs defaults,\_netdev 0 0

[root@node1 ~]# mount -a