### **Step-by-Step Guide for Setting up iSCSI on Linux**

iSCSI (Internet Small Computer Systems Interface) allows block-level access to storage over a TCP/IP network. It's commonly used for connecting servers to storage devices, such as SANs (Storage Area Networks). Here’s a comprehensive guide to setting up an iSCSI target and initiator on a Linux machine.

### **Pre-requisites:**

* Two Linux machines (one will act as the **iSCSI Target** and the other as the **iSCSI Initiator**).
* A root or sudo user to execute commands.
* At least one disk or partition available on the **iSCSI Target** machine.
* **Target machine**: The machine providing the storage.
* **Initiator machine**: The machine requesting and using the storage.

## **Step 1: Install iSCSI Software**

On both the **Target** and **Initiator** machines, you need to install the necessary iSCSI tools.

### **On Ubuntu/Debian:**

Install iSCSI target and initiator utilities:  
  
 sudo apt update

sudo apt install targetcli initiator-utils open-iscsi

### **On CentOS/RHEL:**

Install iSCSI target and initiator utilities:  
  
 sudo yum install targetcli iscsi-initiator-utils

## **Step 2: Set Up the iSCSI Target (Server)**

### **1. Configure the iSCSI target (storage server):**

On the **iSCSI target** machine, we will configure and share the storage.

1. **Create a disk or partition** to share. If you don't have a partition yet, you can create one (example /dev/sdb):

List available disks:  
  
 sudo lsblk

* + Use fdisk or parted to create a partition on /dev/sdb (if not already partitioned).

**Start the target configuration using targetcli**:  
  
 sudo targetcli

1. This will start the interactive targetcli configuration interface.

**Create a new iSCSI target** (replace iqn.2025-01.com.example:storage.target01 with your own unique IQN):  
  
 /> cd /backstores/block

/backstores/block> create block\_storage /dev/sdb

1. This creates a backstore (virtual block storage) from the /dev/sdb disk.

**Create an iSCSI target with a unique IQN** (the iSCSI Qualified Name):  
  
 /backstores/block> cd /iscsi

/iscsi> create iqn.2025-01.com.example:storage.target01

1. The IQN format is iqn.YYYY-MM.com.example:targetname. Make sure to use a unique identifier.

**Map the block storage to the iSCSI target**:  
  
 /iscsi/iqn.2025-01.com.example:storage.target01> cd /tpg1/luns

/tpg1/luns> create /backstores/block/block\_storage

**Configure access control**:  
  
 To allow any host to access the target (this is just an example—be sure to limit access in a production environment):  
  
 /tpg1> set attribute authentication=0

/tpg1> set attribute deal\_with\_unmapped\_luns=0

If you want to restrict access by initiator, you can use:  
  
 /tpg1> set attribute authentication=1

**Enable the iSCSI target**:  
  
 /tpg1> set attribute enable\_target=1

**Save the configuration and exit**:  
  
 /> exit

**Start the target service** (this is required to allow iSCSI initiators to connect):  
  
 sudo systemctl start target

sudo systemctl enable target

## **Step 3: Set Up the iSCSI Initiator (Client)**

### **1. Discover and Connect to the iSCSI Target**

On the **iSCSI initiator** machine (client), perform the following steps:

**Start the iSCSI service**:  
  
 sudo systemctl start iscsid

sudo systemctl enable iscsid

**Discover available iSCSI targets**:  
  
 Use the following command to discover the target server’s available iSCSI targets (replace 192.168.1.100 with the IP address of the target server):  
  
 sudo iscsiadm --mode discovery --type sendtargets --portal 192.168.1.100

1. This will list the available iSCSI targets from the server.

**Log in to the discovered target**:  
  
 Now that you’ve discovered the target, log in to it (replace the iqn.2025-01.com.example:storage.target01 with your actual IQN):  
  
 sudo iscsiadm --mode node --targetname iqn.2025-01.com.example:storage.target01 --portal 192.168.1.100 --login

**Verify the connection**:  
  
 After logging in, you can check the connection status using:  
  
 sudo iscsiadm --mode session --show

1. This should show the active iSCSI session.

## **Step 4: Format and Mount the iSCSI Disk**

**Check if the iSCSI disk is available** (it should appear as /dev/sdb or /dev/sdc, etc.):  
  
 sudo lsblk

**Create a filesystem on the iSCSI disk** (let’s assume it shows up as /dev/sdb):  
  
 sudo mkfs.ext4 /dev/sdb

**Create a mount point**:  
  
 sudo mkdir /mnt/iscsi

**Mount the iSCSI disk**:  
  
 sudo mount /dev/sdb /mnt/iscsi

**Verify the mount**:  
  
 df -h

1. This should show the mounted iSCSI disk.

## **Step 5: Make the iSCSI Mount Persistent**

To ensure that the iSCSI disk mounts automatically after a reboot, add the iSCSI disk to /etc/fstab.

**Get the UUID of the iSCSI disk**:  
  
 sudo blkid /dev/sdb

**Edit the /etc/fstab file**:  
  
 sudo nano /etc/fstab

**Add the following line** (replace UUID with your disk's actual UUID):  
  
 UUID=<your-disk-uuid> /mnt/iscsi ext4 \_netdev 0 0

1. The \_netdev option ensures the disk is mounted after the network is up.

## **Step 6: Monitor and Manage iSCSI**

To monitor the iSCSI connection and manage it:

**Check iSCSI sessions**:  
  
 sudo iscsiadm --mode session --show

**Log out of the iSCSI target**:  
  
 sudo iscsiadm --mode node --targetname iqn.2025-01.com.example:storage.target01 --portal 192.168.1.100 --logout

**Remove the iSCSI target**:  
  
 sudo iscsiadm --mode node --targetname iqn.2025-01.com.example:storage.target01 --delete

## **Summary of Commands:**

### **On the iSCSI Target (Server):**

Install required packages:  
 sudo apt install targetcli

Create a backstore:  
 /backstores/block> create block\_storage /dev/sdb

Create an iSCSI target:  
 /iscsi> create iqn.2025-01.com.example:storage.target01

Map the backstore to the target:  
 /tpg1/luns> create /backstores/block/block\_storage

Enable the target:  
 /tpg1> set attribute enable\_target=1

Start the target service:  
 sudo systemctl start target

sudo systemctl enable target

### **On the iSCSI Initiator (Client):**

Start the iSCSI service:  
 sudo systemctl start iscsid

sudo systemctl enable iscsid

Discover available iSCSI targets:  
 sudo iscsiadm --mode discovery --type sendtargets --portal 192.168.1.100

Log in to the iSCSI target:  
 sudo iscsiadm --mode node --targetname iqn.2025-01.com.example:storage.target01 --portal 192.168.1.100 --login

Create a filesystem on the disk:  
 sudo mkfs.ext4 /dev/sdb

Mount the iSCSI disk:  
 sudo mount /dev/sdb /mnt/iscsi

This guide should help you configure both the iSCSI target and initiator for a full iSCSI setup on Linux. If you have any issues or need further clarification, feel free to ask!