iSCSI on Red Hat Enterprise Linux: Targets, Initiators and SANs

John VanDyk
ENT/NREM Business Center
Iowa State University
February 8, 2013
This presentation goes over iSCSI basics and introduces the open-iscsi tools on Red Hat Enterprise Linux 6.

It is a “how do I understand basic iSCSI tools” presentation. It is not a “why should I use iSCSI” presentation.
• History
• Terminology
• Server-side tools
• Client-side tools
• Demo

(The Demo was bringing up two Linux VMs, exposing a 5GB secondary drive on VM1 as a target and connecting to it from VM2.)
SCSI’s been around for a long time.
A hand-waving general timeline of SCSI-related things.
iSCSI terminology is rich and varied.
Node: an addressable, discoverable entity
Nodes have globally unique iSCSI qualified names

`iqn.date.naming authority.string`
iSCSI Names

\texttt{iqn.date.naming\ authority.string}

\texttt{iqn}

\texttt{iqn.2013-02}

\texttt{iqn.2013-02.edu.iastate.ent}

\texttt{iqn.2013-02.edu.iastate.ent:storage1}

Building an iSCSI Qualified Name. If you’re feeling adventurous you could use IEEE EUI-64 identifiers in hexadecimal (see RFC3721 section 1.2).
Nodes have addresses

IPv4 address:port
198.51.100.1:3260

IPv6 address:port
[1080::8:800:200C:417A]:3260

hostname:port
storage123.example.com:3260
So now we’ve established that both initiators and targets are kinds of iSCSI nodes.
Each node can have an alias as well. This is just a friendly name. Note there’s no port on the initiator’s address, because the connection is outgoing from the initiator. Just like your webserver runs on port 80 but the client computer accessing it can use different ports.
Aliases might be used in UIs. In the example above, a message like "you have connected to BioData" is a lot friendlier than the alternative.
Multiple target nodes may reside inside one physical computer.
Each target may have more than one LUN.
yum install scsi-target-utils

To set up an iSCSI target, you need the iSCSI target utilities.
# rpm -ql scsi-target-utils

/etc/rc.d/init.d/tgtd
/etc/sysconfig/tgtd
/etc/tgt/targets.conf

/usr/sbin/tgt-admin
/usr/sbin/tgt-setup-lun
/usr/sbin/tgtadm
/usr/sbin/tgtd
/usr/sbin/tgtimg
# rpm -ql scsi-target-utils

/etc/rc.d/init.d/tgtd
/etc/sysconfig/tgtd
/etc/tgt/targets.conf

/usr/sbin/tgt-admin
/usr/sbin/tgt-setup-lun
/usr/sbin/tgtadm
/usr/sbin/tgtd
/usr/sbin/tgtdimg
# rpm -ql scsi-target-utils

/etc/rc.d/init.d/tgtd
/etc/sysconfig/tgtd
/etc/tgt/targets.conf

/usr/sbin/tgt-admin
/usr/sbin/tgt-setup-lun
/usr/sbin/tgtadm
/usr/sbin/tgtd
/usr/sbin/tgtimg

Here is where your target configuration lives
# rpm -ql scsi-target-utils

/etc/rc.d/init.d/tgtd
/etc/sysconfig/tgtd
/etc/tgt/targets.conf

/usr/sbin/tgt-admin
/usr/sbin/tgt-setup-lun
/usr/sbin/tgtadm
/usr/sbin/tgtd
/usr/sbin/tgtimg

Perl script.
Reads targets.conf and executes tgtadm commands
# rpm -ql scsi-target-utils

/etc/rc.d/init.d/tgtd
/etc/sysconfig/tgtd
/etc/tgt/targets.conf

/usr/sbin/tgt-admin
/usr/sbin/tgt-setup-lun
/usr/sbin/tgtadm
/usr/sbin/tgtd
/usr/sbin/tgtimg

bash script. Makes entries in targets.conf from the command line
# rpm -ql scsi-target-utils

/etc/rc.d/init.d/tgtd
/etc/sysconfig/tgtd
/etc/tgt/targets.conf

/usr/sbin/tgt-admin
/usr/sbin/tgt-setup-lun
/usr/sbin/tgtadm
/usr/sbin/tgtd
/usr/sbin/tgtimg

The main utility for working with SCSI targets
# rpm -ql scsi-target-utils

/etc/rc.d/init.d/tgtd
/etc/sysconfig/tgtd
/etc/tgt/targets.conf

/usr/sbin/tgt-admin
/usr/sbin/tgt-setup-lun
/usr/sbin/tgtadm
/usr/sbin/tgtd
/usr/sbin/tgtimg

SCSI target daemon
# rpm -ql scsi-target-utils

/etc/rc.d/init.d/tgtd
/etc/sysconfig/tgtd
/etc/tgt/targets.conf

/usr/sbin/tgt-admin
/usr/sbin/tgt-setup-lun
/usr/sbin/tgtadm
/usr/sbin/tgtd
/usr/sbin/tgtimg

Creates and dumps image files for tape, dvd+r or disk
Find the 5GB drive we want to share

```bash
# dmesg | grep sd
```

```
sd 2:0:1:0: [sdb] 10485760
   512-byte logical blocks:
   (5.36 GB/5.00 GiB)
```

If you do not have static set of devices you are urged to use udev rules, as `/dev/sdb` might be `/dev/sdc` next time. A gotcha here too is LVM rules which may quietly mark a device as “in use” (see http://www.sysarchitects.com/iscsi_target_on_rhel6)
Minimal configuration in /etc/tgt/targets.conf

```xml
<target iqn.2013-02.edu.iastate.ent:biodata>
  backing-store /dev/sdb
  vendor_id "VMWare"
  # Only an initiator at this IP
  # may talk to this target.
  initiator-address 172.16.1.20
</target>
```

We could have used direct-store instead of backing-store and not defined the vendor_id, because direct-store queries the device and passes along what it finds. Sadly this threw an error on VMWare Fusion where I was running the demo so I used backing-store instead.
Start iSCSI daemon (tgtd)

# service tgtd start
Show Targets

# tgtadm --lld iscsi --op show --mode target
Target 1: iqn.2013-02.edu.iastate.ent:biodata
LUN: 1

Type: disk
SCSI ID: IET 00010001
SCSI SN: beaf11
Size: 5369 MB, Block size: 512
Online: Yes
Removable media: No
Prevent removal: No
Readonly: No
Backing store type: rdwr
Backing store path: /dev/sdb
Backing store flags:
Add Firewall Rule for iSCSI

```
 iptables -I INPUT 4 \
   -p tcp \
   -s x.x.x.x \
   --dport 3260 \
   -m state \
   --state NEW \
   -j ACCEPT
```

`x.x.x.x` is the IP address of the initiator that wants to connect to this target.
Now, on the Initiator (client that will connect)

```
yum install iscsi-initiator-utils
```

On the initiator, you need the iSCSI initiator utilities in order to connect to the target.
# rpm -ql scsi-initiator-utils \
grep | “interesting”

/etc/iscsi/iscsid.conf
/etc/rc.d/init.d/iscsi
/etc/rc.d/init.d/iscsid

/sbin/iscsiad
/sbin/iscsid
# rpm -ql scsi-initiator-utils \ grep | “interesting”

/etc/iscsi/iscsid.conf
/etc/rc.d/init.d/iscsi
/etc/rc.d/init.d/iscsid

/sbin/iscsiadm
/sbin/iscsid

Handle automatic login to targets
# rpm -ql scsi-initiator-utils \
  grep | "interesting"

/etc/iscsi/iscsid.conf
/etc/rc.d/init.d/iscsi
/etc/rc.d/init.d/iscsid

/sbin/iscsiadm
/sbin/iscsid

Start iscsid daemon

Handles iSCSI sessions
# rpm -ql scsi-initiator-utils \
grep | “interesting”

/etc/iscsi/iscsid.conf
/etc/rc.d/init.d/iscsi
/etc/rc.d/init.d/iscsid

/sbin/iscsiadm
/sbin/iscsid

Administration utility for iSCSI initiator
Define our initiator’s name

# cat /etc/iscsi/initiatorname.iscsi

InitiatorName=iqn.2013-02.edu.iastate.ent:client0
Start iscsi services

# service iscsi start
# service iscsid start
# iscsiadm -m discoverydb \
-t sendtargets \
-p 172.16.198.3:3260 \
--discover
The ,1 on the end of the IP:port is the target portal group tag. In this case, it is 1 since there is only a single target portal group on our server. If we had another IP:port with a different set of targets that would be a second target portal group.
The initiator has stored a record of this discovery.

```
# cat /var/lib/iscsi/nodes/iqn.2013-02.edu.iastate.ent:biodata/172.16.198.3,3620,1/default
```
node.name = iqn.2013-02.edu.iastate.ent:biodata
node.tpgt = 1
node.startup = automatic
node.leading_login = No
iface.iscsi_ifacename = default
iface.transport_name = tcp
iface.vlan_id = 0
iface.vlan_priority = 0
iface.iface_num = 0
iface.mtu = 0
iface.port = 0
node.discovery_address = 172.16.198.3
...
Restarting iscsi service or rebooting connects the device


kernel: sd 3:0:0:1: [sdb] 10485760 512-byte logical blocks: (5.36 GB/5.00 GiB)
kernell: sd 3:0:0:1: [sdb] Write Protect off
kernel: sdb:
kernel: sd 3:0:0:1: [sdb] Attached SCSI disk

It works!
Hullo! A local device, ready for partitioning, formatting, and general frolic!
10,000-foot overview of iSCSI SAN. Two switches for redundant paths from initiator to target.
Two subnets, two paths.
If you cannot afford 10G ethernet then gigabit may be a bottleneck.
SAN: a Storage Area Network

172.16.1.1
172.16.1.2
172.16.1.20
172.16.1.21
172.16.2.20
172.16.2.21

172.16.2.1
172.16.2.2
A target can be made available at multiple addresses, creating multiple routes from the initiator to the target. These can be used for failover or performance.
References:
Hufferd, John L. 2002. *iSCSI: The Universal Storage Connection*


/usr/share/doc/iscsi-initiator-utils-6.2.0.872/README

/usr/share/doc/scsi-target-utils-1.0.24/README