

Centos7系列（八）ISCSI网络存储服务

原创

Mr大表哥

2017-05-17 10:34:09

评论(0)

367人阅读

博主QQ: 819594300

博客地址: <http://zpf666.blog.51cto.com/>

有什么疑问的朋友可以联系博主，博主会帮你们解答，谢谢支持！

ISCSI网络存储服务：

说明：iscsi实现网络存储，提供存储端叫 target，使用存储端叫 initiator。target 上可以提供存储空间，initiator负责连接ISCSI设备，在ISCSI设备中创建文件系统，以及存取数据，在 initiator 看上去是多了一块硬盘。

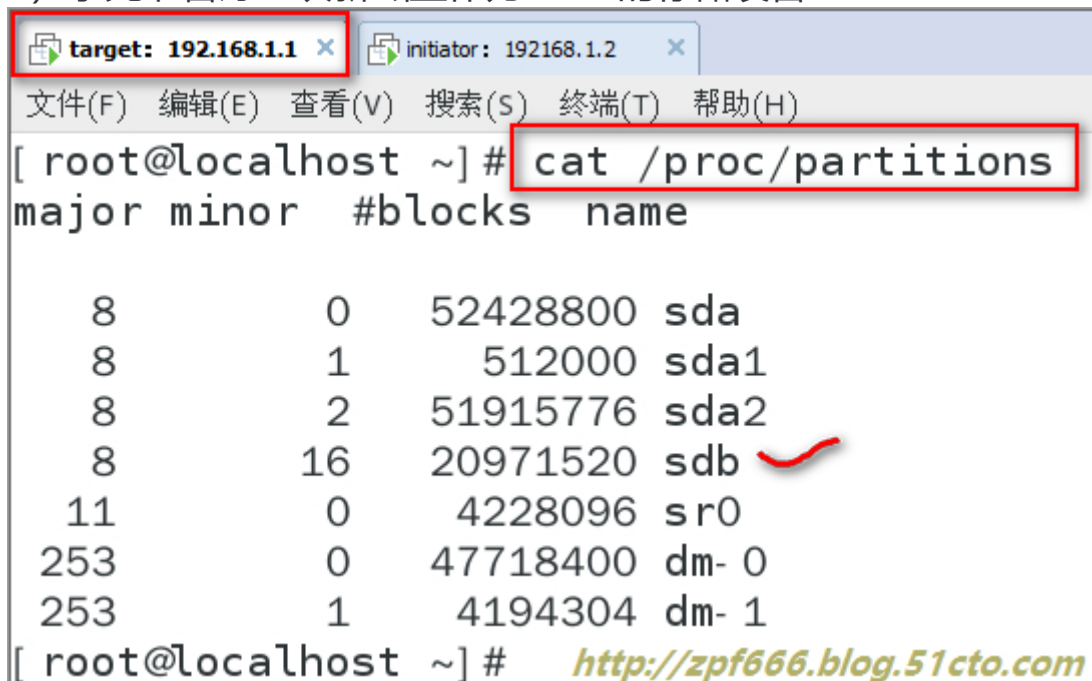
服务器端配置target，准备做为LUN发布的ISCSI的存储设备（可以是一整块磁盘、一个分区、逻辑卷或RAID阵列）。

虚拟机环境图：



实验步骤（本实验以一整块磁盘做为iscsi存储）：

1) 事先准备好一块新磁盘作为ISCSI的存储设备



2) 安装target



```
target: 192.168.1.1 x initiator: 192168.1.2 x
文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)
[ local ]
name=linux- centos7
baseurl=file:///media
enabled=1
gpgcheck=0
~
http://zpf666.blog.51cto.com
```

```
target: 192.168.1.1 x initiator: 192168.1.2 x
文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)
[ root@localhost ~] # yum -y install targetd targetcli
```

3) 启动target服务

```
target: 192.168.1.1 x initiator: 192168.1.2 x
文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)
[ root@localhost ~] # systemctl enable target;systemctl start target
[ root@localhost ~] #
```

4)设置防火墙,开启3260端口例外, 以便客户端连接

```
target: 192.168.1.1 x initiator: 192168.1.2 x
文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)
[ root@localhost ~] # firewall-cmd --permanent --add-port=3260/tcp
success
[ root@localhost ~] # firewall-cmd --reload
success
[ root@localhost ~] # firewall-cmd --list-all
public (default, active)
  interfaces: eno16777736
  sources:
  services: dhcpv6-client ssh
  ports: 3260/tcp
  masquerade: no
  forward-ports:
  icmp-blocks:
  rich rules:
[ root@localhost ~] #
```

3260是iSCSI target 的TCP监听端口, 为了initiator能连接iscsi, 需要开启该端口例外。

<http://zpf666.blog.51cto.com>

5) 执行targetcli工具

```
target: 192.168.1.1 x initiator: 192168.1.2 x
文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)
[ root@localhost ~] # targetcli
Warning: Could not load preferences file /root/.targetcli/prefs.bin.
targetcli shell version 2.1.fb41
Copyright 2011- 2013 by Datera, Inc and others.
For help on commands, type 'help'.

/>
```

<http://zpf666.blog.51cto.com>

下面正式开始做iscsi的共享储存:

基本思路:

先把准备共享的块做出来, 创建一个target, 在target上创建lun, 一个lun连接一个块

6) 创建块即给要发布的存储磁盘起个名字



```

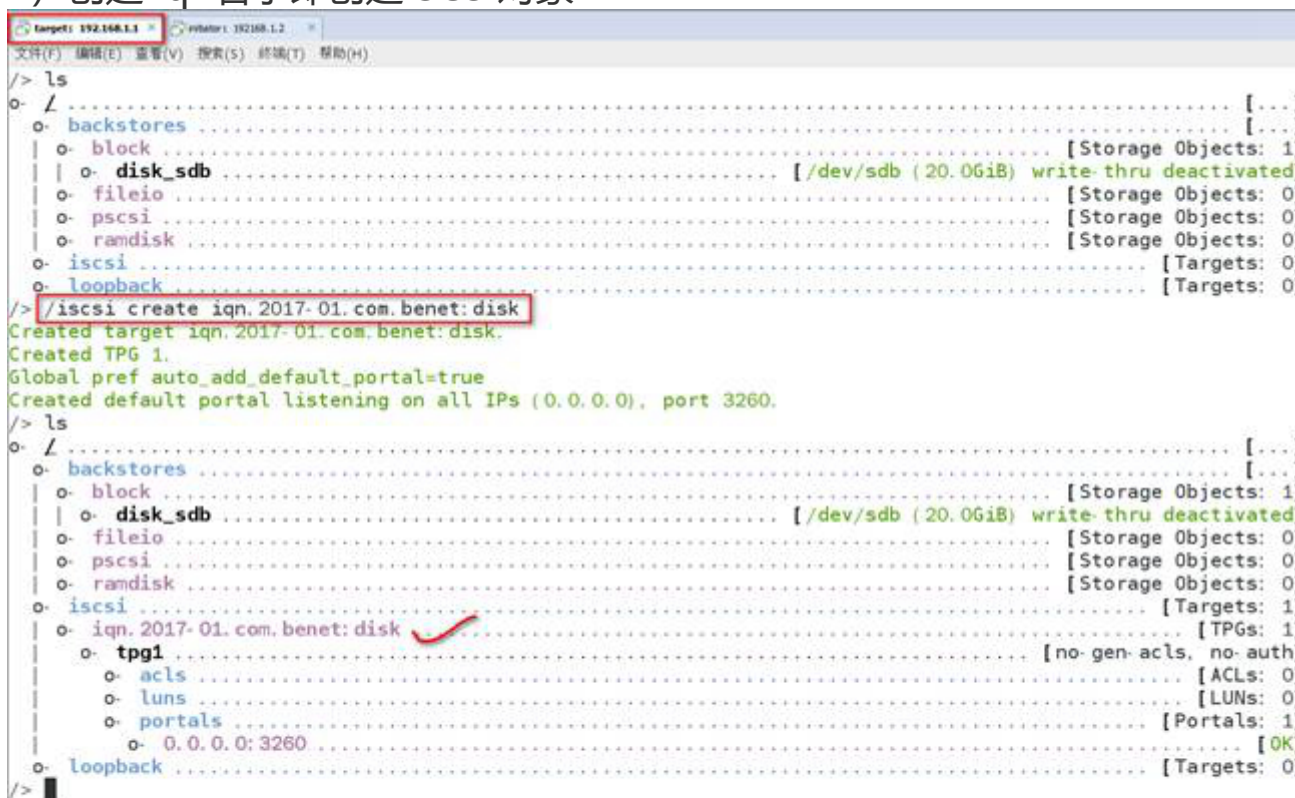
target: 192.168.1.1  initiator: 192.168.1.2
文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)

/> ls
o- /
  o- backstores
    o- block
    o- fileio
    o- pscsi
    o- ramdisk
    o- iscsi
    o- loopback
/> /backstores/block create disk_sdb /dev/sdb
Created block storage object disk_sdb using /dev/sdb.
/> ls
o- /
  o- backstores
    o- block
      o- disk_sdb
    o- fileio
    o- pscsi
    o- ramdisk
  o- iscsi
  o- loopback
  
```

disk_sdb是给要共享的/dev/sdb磁盘自定义的名字

<http://zpf666.blog.51cto.com>

7) 创建 iqn名字即创建ISCSI对象



```

target: 192.168.1.1  initiator: 192.168.1.2
文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)

/> ls
o- /
  o- backstores
    o- block
      o- disk_sdb
    o- fileio
    o- pscsi
    o- ramdisk
  o- iscsi
  o- loopback
/> /iscsi create iqn.2017-01.com.benet:disk
Created target iqn.2017-01.com.benet:disk.
Created TPG 1.
Global pref auto_add_default_portal=true
Created default portal listening on all IPs (0.0.0.0), port 3260.
/> ls
o- /
  o- backstores
    o- block
      o- disk_sdb
    o- fileio
    o- pscsi
    o- ramdisk
  o- iscsi
    o- iqn.2017-01.com.benet:disk
      o- tpg1
        o- acls
        o- luns
        o- portals
          o- 0.0.0.0:3260
  o- loopback
  
```

iqn: 英文全称是iSCSI Qualified Name, 意思是iSCSI合格的名字。

根据iSCSI定义的规范, Target的命名格式为“iqn.YYYY-mm.反向域名:识别标记”, 其中YYYY-mm表示年月, 反向域名建议为服务器FQDN的倒序, 识别标记是用户自定义的字符串(以便区分)。

这一步操作呢就是创建iscsi对象, 通俗的说就是创建一个target。(target英语的意思是目标) 就是创建一个客户端能访问的访问目标, 为客户机分配存储空间。在这里就是给访问目标起个名字。

<http://zpf666.blog.51cto.com>

使用tgtadm工具创建iSCSI对象时, 主要命令选项如下所述。

- 1) -l, --lld: 指定驱动类型, 如“-l iscsi”表示iSCSI存储。
- 2) -o, --op: 指定操作类型, 如“-o new”表示创建、“-o delete”表示删除。
- 3) -m, --mode: 指定管理目标, 如“-m target”表示iSCSI对象。
- 4) -t, --tid: 指定对象ID号, 如“-t 1”表示第一个对象。
- 5) -T, --targetname: 指定iSCSI对象的名称, 注意应符合“iqn.”规范。

<http://zpf666.blog.51cto.com>

8) 创建LUN (即逻辑单元) 并绑定块

```
target: 192.168.1.1 initiator: 192.168.1.2
文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)
/> /iscsi/iqn.2017-01.com.benet:disk/tpg1/luns create /backstores/block/disk_sdb
Created LUN 0.
/> ls
一个iscsi对象可以创建多个lun(如: lun0, lun1,...)
o- /
  o- backstores
    o- block
      o- disk_sdb [Storage Objects: 1]
    o- fileio [Storage Objects: 0]
    o- pscsi [Storage Objects: 0]
    o- ramdisk [Storage Objects: 0]
  o- iscsi
    o- iqn.2017-01.com.benet:disk [Targets: 1]
      o- tpg1 [TPGs: 1]
        o- acls [ACLs: 0]
        o- luns [LUNs: 1]
          o- lun0 [block/disk_sdb (/dev/sdb)]
        o- portals [Portals: 1]
          o- 0.0.0.0:3260 [OK]
      o- loopback [OK]
```

<http://zpf666.blog.51cto.com>

- 1) -l, --lun: 指定LUN的ID号, 如“-l 1”表示第一个LUN。
- 2) -b, --backing-store: 指定块设备位置, 如“-b /dev/vol0/lv-host21”。
- 3) -I, --initiator-address: 指定客户机地址, 如“-I 192.168.4.21”。

<http://zpf666.blog.51cto.com>

9) 设置ACL即将iSCSI对象与客户端IP或主机名绑定(initiator端必须指定ACL的名字才可以连接iscsi存储, 就相当于initiator端连接iscsi存储的一个凭证)

```
target: 192.168.1.1 initiator: 192.168.1.2
文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)
/> /iscsi/iqn.2017-01.com.benet:disk/tpg1/acls create iqn.2017-01.com.benet:client
Created Node ACL for iqn.2017-01.com.benet:client
Created mapped LUN 0.
/> ls
iqn.2017-01.com.benet:client是initiator的名字, 在initiator端是要用到的
o- /
  o- backstores
    o- block
      o- disk_sdb [Storage Objects: 1]
    o- fileio [Storage Objects: 0]
    o- pscsi [Storage Objects: 0]
    o- ramdisk [Storage Objects: 0]
  o- iscsi
    o- iqn.2017-01.com.benet:disk [Targets: 1]
      o- tpg1 [TPGs: 1]
        o- acls [ACLs: 1]
          o- iqn.2017-01.com.benet:client [Mapped LUNs: 1]
            o- mapped_lun0 [lun0 block/disk_sdb (rw)]
        o- luns [LUNs: 1]
          o- lun0 [block/disk_sdb (/dev/sdb)]
        o- portals [Portals: 1]
          o- 0.0.0.0:3260 [OK]
      o- loopback [OK]
```

<http://zpf666.blog.51cto.com>

10) 修改监听地址 (把默认的0.0.0.0改成本机的地址)


```

Target: 192.168.1.1
Initiator: 192.168.1.2

/> /iscsi/iqn.2017-01.com.benet:disk/tpg1/portals/ delete 0.0.0.0 3260
Deleted network portal 0.0.0.0:3260
/> /iscsi/iqn.2017-01.com.benet:disk/tpg1/portals/ create 192.168.1.1
Using default IP port 3260
Created network portal 192.168.1.1:3260.
/> ls
o- /
o- backstores
  o- block
    o- disk_sdb
  o- fileio
  o- pscsi
  o- ramdisk
o- iscsi
  o- iqn.2017-01.com.benet:disk
    o- tpg1
      o- acls
      o- iqn.2017-01.com.benet:client
        o- mapped_lun0
      o- luns
      o- lun0
      o- portals
        o- 192.168.1.1:3260
o- loopback

```

target: 192.168.1.1 initiator: 192168.1.2

文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)

```

/> exit 只能用exit退出，不能用bye。

```

说明：可以查看 `/etc/target/saveconfig.json` 配置文件，该配置文件保存着iSCSI的配置。



target: 192.168.1.1 x initiator: 192.168.1.2 x

文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)

[root@localhost ~]# vim /etc/target/saveconfig.json

这里面保存着我们刚才用过targetcli工具操作的一系列操作配置。

<http://sp666.blog.51cto.com>

下面是在initiator端的配置:

1) 安装iscsi-initiator-utils

target: 192.168.1.1 initiator: 192168.1.2

文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)

```
[root@localhost ~]# rpm -q iscsi-initiator-utils
iscsi-initiator-utils-6.2.0.873-35.el7.x86_64
[root@localhost ~]#
```

我们可以先看一下这个软件有没有安装，我这里是安装的，其实安装系统的时候默认这个软件就是安装的，如果没有安装，插系统盘，YUM安装即可。

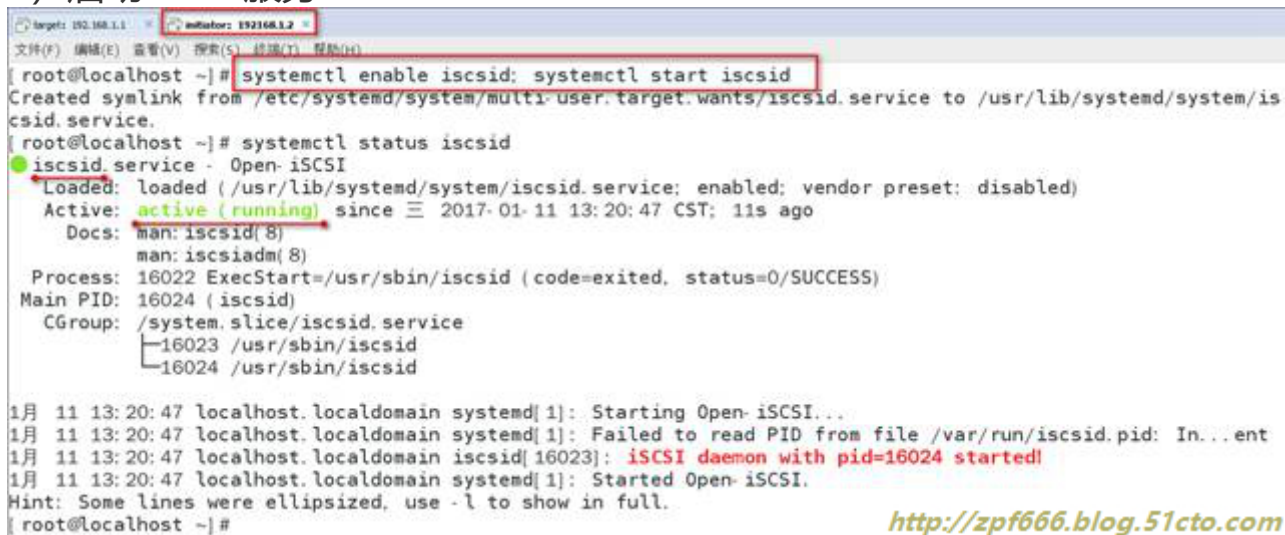
2) 给initiator起一个名字



The screenshot shows a terminal window with a menu bar at the top containing '文件(F)', '编辑(E)', '查看(V)', '搜索(S)', '终端(T)', and '帮助(H)'. Below the menu bar, the command `[root@localhost ~] # vim /etc/iscsi/initiatorname.iscsi` is entered. The terminal has two tabs: 'target: 192.168.1.1' and 'initiator: 192168.1.2', with the latter being the active tab. The command and the active tab are highlighted with red boxes.



3) 启动iscsid服务



4) 寻找 (发现) target端的存储 (即查询target端的iscsi对象)



Iscsi的用法:

1. 发现 iscsi 存储: `iscsiadm -m discovery -t st -p ISCSI_IP`
2. 查看 iscsi 发现记录 `iscsiadm -m node`
3. 删除 iscsi 发现记录 `iscsiadm -m node -o delete -T LUN_NAME -p ISCSI_IP`
4. 登录 iscsi 存储 `iscsiadm -m node -T LUN_NAME -p ISCSI_IP -l`
5. 登出 iscsi 存储 `iscsiadm -m node -T LUN_NAME -p ISCSI_IP -u`

<http://zpf666.blog.51cto.com>

5) 登陆存储 (即登陆iscsi对象)



6) 验证客户端ISCSI连接



```

[ root@localhost ~]# lsscsi
[ 0: 0: 0: 0]    disk      VMware,  VMware Virtual S 1.0    /dev/sda
[ 2: 0: 0: 0]    cd/dvd    NECVMWar VMware IDE CDR10  1.00   /dev/sr0
[ 3: 0: 0: 0]    disk      LIIO- ORG  disk_sdb          4.0    /dev/sdb
[ root@localhost ~]#

```

<http://zpf666.blog.51cto.com>

7) 格式化、创建挂载点、挂载



```

[ root@localhost ~]# mkdir /data
[ root@localhost ~]#

```

<http://zpf666.blog.51cto.com>



```

[ root@localhost ~]# mkfs.xfs /dev/sdb
meta-data=/dev/sdb              isize=512    agcount=4, agsize=1310720 blks
=                               sectsz=512   attr=2, projid32bit=1
=                               crc=1        finobt=0, sparse=0
data      =                       bsize=4096   blocks=5242880, imaxpct=25
=                               sunit=0          swidth=0 blks
naming    =version 2           bsize=4096   ascii-ci=0 ftype=1
log       =internal log        bsize=4096   blocks=2560, version=2
=                               sectsz=512     sunit=0 blks, lazy-count=1
realtime  =none                extsz=4096   blocks=0, rtextents=0
[ root@localhost ~]#

```

<http://zpf666.blog.51cto.com>



```

[ root@localhost ~]# blkid /dev/sdb >> /etc/fstab
[ root@localhost ~]# vim /etc/fstab

```

```

#
# /etc/fstab
# Created by anaconda on Wed Jan  4 23:01:26 2017
#
# 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/cl-root    /                    xfs     defaults        0 0
UUID=83e8291a-e44e-47ab-82c4-d734a2ed0064 /boot                xfs     defaults        0 0
/dev/mapper/cl-swap    swap                 swap     defaults        0 0
UUID="7954d28f-0fd5-481f-9f58-b32601138dca" /data                xfs     defaults,_netdev 0 0

```

注意红框内的东西，后面一定要跟上_netdev

<http://zpf666.blog.51cto.com>

版权声明：原创作品，如需转载，请注明出处。否则将追究法律责任