

Hello World!

班门弄斧 SCSI TARGET with qemu/kvm

SCSI Target

SCSI 子系统使用的是客户机 - 服务器 (C/S) 模型。

Initiator 和 Target 是 SCSI 的术语。Initiator 指的是 SCSI 会话的发起方，它向 Target 请求 LUN (Logical Unit Number ，可认为是逻辑块设备) ，并将数据的读写指令发送给 Target。Target 指的是接受 SCSI 会话的一方，它接收来自 Initiator 的指令，为 Initiator 提供 LUN ，并实现对 LUN 的读写。如果用客户端 - 服务器的模型来考虑，Initiator 即是客户端，Target 是服务器。在 SAN 中，主机一般都是 Initiator ，存储设备则是 Target。

SCSI Target 子系统可以让一台计算机扮演一个 SCSI 存储设备来工作，响应其他 initiator 发起的请求。这样，通过操作 target，我们就可以灵活地定制存储设备。

最常见的 SCSI target 子系统的实现是 iSCSI 服务器，它使用标准的 TCP/IP 来封装 SCSI 指令，通过网络来提供一个 SCSI 设备。

2.6.38 以前的 Linux 内核的 SCSI 子系统使用 STGT 来实现 SCSI target。STGT 于 2006 年末引入，它在内核中有一个库，来配合内核中的 target 驱动工作。而所有的 target 处理都在用户空间完成，这可能回带来一些性能瓶颈

Linux-IO(LIO) Target

2010 年底，LIO 项目进入了 2.6.38 主线，取代原有 STGT 项目，成为新的内核态 SCSI target。

它以软件的方式在内核态在实现了对各种 SCSI Target 的模拟，支持 SAN 技术中所有流行的存储协议包括 FibreChannel(scsi over FC)、FCoE(FibreChannel over ethernet)、iSCSI(scsi over IP) 等，还能为本机生成模拟的 SCSI 设备，以及为虚拟机提供基于 virtio 的模拟 SCSI 设备，使用户能够使用相对廉价的 Linux 系统实现 SCSI、SAN 的各种功能，而不用购买昂贵的专业设备。

它以内核模块的方式提供不同的 SCSI 传输协议和后端存储的支持，主要包括了 target_core.ko 和其他若干可选的驱动模块，模拟了通用的 SCSI 设备，并且实现了 SCSI 协议的基本命令集标准的 SPC-3 和 4，提供 scsi target 服务。

LIO 后来改名为 TCM(Target Core Module)。

用户态普遍使用的管理工具是 targetcli 和 rtslib。其中 targetcli 程序用于配置、管理 LIO target，rtslib 则提供 Python 编程接口。通过该工具可以配置相应的后端存储，并根据不同的协议导出为不同形式的 scsi target。

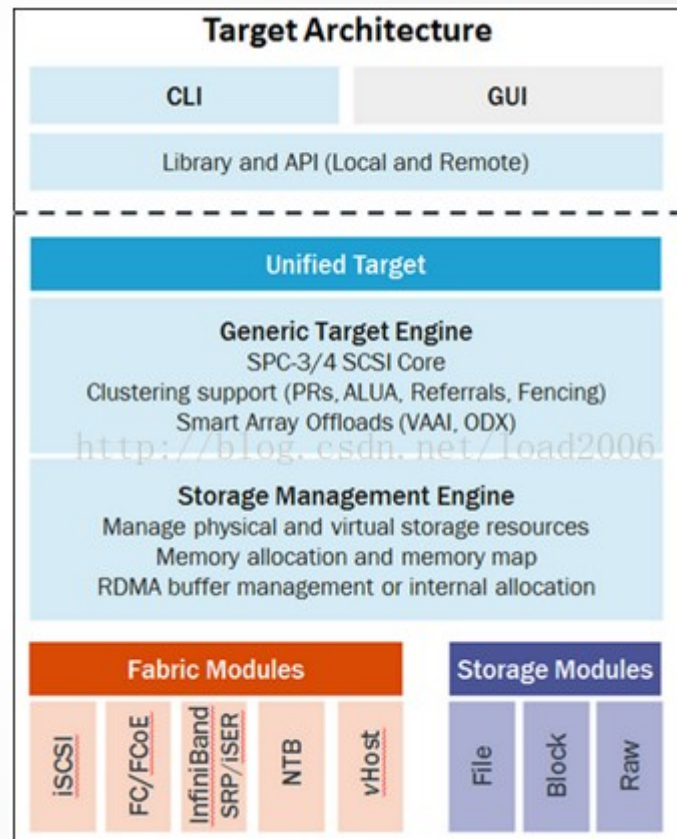
Linux-IO(LIO) Target

LIO 通过各种各样的 backstore 对象，来导入后端存储。

通过连接上各种前端 FabricModules，来导出模拟的 SCSI 设备。

无论是 Fibre Channel、FCoE、iSCSI 还是 vhost（即 virtio），都只是一种 Fabric 技术，我们可以将其理解成传输协议，而在这些传输协议中传输的 SCSI 命令则总是由核心的 SCSI 设备处理的。要支持新的 Fabric 技术，并不需要修改 Generic Target Engine 和后端存储模块，只需要按照该 Fabric 技术的规范实现对 SCSI 命令的传输就可以了。比如用于网络的 FCoE fabric module(tcm_fc.ko) 就是 SCSI over FC + FibreChannel over ethernet;

用于单机的 loop scsi fabric module(tcm_loop.ko) 相当于把本地设备经过 Lio 映射之后再给本地主机使用，而 vhost scsi fabric module(**vhost_scsi.ko** AKA tcm_vhost) 就是把 SCSI 命令放在 virtio 队列中传输。



TCMU (TCM in Userspace)

TCM 中，backstores 可以是文件，块设备，RAM 等，实现全部在内核态实现。

针对新的用户态 target 的需求，如将带有压缩、加密等特性的 Gluster 的 GLFS，Ceph 的 RBD 或 qemu 支持的存储，作为 backstore，为 TCM 增加这些支持会很繁琐和困难。因为 TCM 是纯内核态代码。

于是人们把这类 backstore 的实现放在了用户态，利用已有的 rbd 或 glfs 等的用户态库，为 TCM 创建用户态 backstore 的机制叫做 TCMU，也就是让用户态进程来扮演 target。TCMU 机制的实现在 `target_core_user.ko`

TCMU 使用了已有的 UIO 子系统。UIO 子系统允许在用户态开发设备驱动，这个概念十分贴近 TCMU 的使用案例

TCMU Runner

为了快速和灵活，TCMU 内核模块接口被设计的较复杂，为了便于开发，于是便有了 tcmu runner(含 libtcmu)，作用于 target_core_user.ko 和 userspace backstore 实现之间，帮助用户进程处理繁琐的 TCMU 接口细节，如将 UIO, netlink 以及 DBus 等处理转化为易用的 C API.

使用这些 API 的模块称为 TCMU handlers. 编写 handler 的方式有两种：

1. tcmu-runner plugin. Say /usr/lib64/tcmu-runner/*.so

```
MiWiFi-R3L-srv:~ # ll /usr/lib64/tcmu-runner/  
total 56  
-rwxr-xr-x 1 root root 14408 Nov  1 02:01 handler_glfs.so  
-rwxr-xr-x 1 root root 26792 Nov  1 02:01 handler_iow.so  
-rwxr-xr-x 1 root root 10240 Nov  1 02:01 handler_rbd.so
```

2. libtcmu. Say qemu-tcmu

```
MiWiFi-R3L-srv:~ # ldd /usr/bin/qemu-tcmu  
linux-vdso.so.1 (0x00007fff40493000)  
libz.so.1 => /lib64/libz.so.1 (0x00007fd17aca4000)  
libaio.so.1 => /lib64/libaio.so.1 (0x00007fd17aaa2000)  
libbz2.so.1 => /usr/lib64/libbz2.so.1 (0x00007fd17a885000)  
libtcmu.so.1 => /usr/lib64/libtcmu.so.1 (0x00007fd17a670000)  
libpthread-2.0.so.0 => /usr/lib64/libpthread-2.0.so.0 (0x00007fd17a670000)
```

为了管理 TCMU 的 backstores, targetcli-fb 是必需的。

qemu-tcmu example

```
# systemctl start targetcli.service
# systemctl start tcmu-runner.service
# lsmod | grep target
target_core_user      32768      0
uio                   16384      1 target_core_user
target_core_mod       401408     1 target_core_user
configfs              45056     2 target_core_mod
# ll /sys/class/uio/
total 0
```

```
MiWiFi-R3L-srv:/opt/vms # targetcli
targetcli shell version 2.1.fb47
Copyright 2011-2013 by Datera, Inc and others.
For help on commands, type 'help'.

/> ls
o- / ..... [..]
  o- backstores ..... [..]
    | o- block ..... [Storage Objects: 0]
    | o- fileio ..... [Storage Objects: 0]
    | o- pscsi ..... [Storage Objects: 0]
    | o- ramdisk ..... [Storage Objects: 0]
    | o- user:glfs ..... [Storage Objects: 0]
    | o- user:qcow ..... [Storage Objects: 0]
    | o- user:rbd ..... [Storage Objects: 0]
  o- iscsi ..... [Targets: 0]
  o- loopback ..... [Targets: 0]
  o- vhost ..... [Targets: 0]
  o- xen_pvscsi ..... [Targets: 0]
/>
```

qemu-tcmu example

```
MiWiFi-R3L-srv:/opt/vms # mkdir /opt/vms/tumbleweed
MiWiFi-R3L-srv:/opt/vms # qemu-img create -f qcow2 tumbleweed/disk0.qcow2 30G
Formatting 'tumbleweed/disk0.qcow2', fmt=qcow2 size=32212254720 encryption=off
MiWiFi-R3L-srv:/opt/vms # qemu-img info tumbleweed/disk0.qcow2
image: tumbleweed/disk0.qcow2
file format: qcow2
virtual size: 30G (32212254720 bytes)
disk size: 196K
cluster_size: 65536
Format specific information:
    compat: 1.1
    lazy refcounts: false
    refcount bits: 16
    corrupt: false
```

qemu-tcmu /opt/vms/tumbleweed/disk0.qcow2 &

```
MiWiFi-R3L-srv:/opt/vms # targetcli ls
o- / ..... [....]
  o- backstores ..... [....]
    | o- block ..... [Storage Objects: 0]
    | o- fileio ..... [Storage Objects: 0]
    | o- pscsi ..... [Storage Objects: 0]
    | o- ramdisk ..... [Storage Objects: 0]
    | o- user:glfs ..... [Storage Objects: 0]
    | o- user:qcow ..... [Storage Objects: 0]
    | o- user:qemu ..... [Storage Objects: 0]
    | o- user:rbd ..... [Storage Objects: 0]
  o- iscsi ..... [Targets: 0]
  o- loopback ..... [Targets: 0]
  o- vhost ..... [Targets: 0]
  o- xen_pvscsi ..... [Targets: 0]
MiWiFi-R3L-srv:/opt/vms #
```


qemu-tcmu example

```
//Create the qemu-tcmu target  
# targetcli /backstores/user:qemu create mydisk0 30G @drive
```

```
MiWiFi-R3L-srv:/opt/vms # ll /dev/uio*  
crw----- 1 root root 243, 0 Jan 17 10:26 /dev/uio0  
MiWiFi-R3L-srv:/opt/vms #
```

```
MiWiFi-R3L-srv:/opt/vms # targetcli ls  
o- / .....  
  o- backstores .....  
    o- block ..... [Storage Objects: 0]  
    o- fileio ..... [Storage Objects: 0]  
    o- pscsi ..... [Storage Objects: 0]  
    o- ramdisk ..... [Storage Objects: 0]  
    o- user:glfs ..... [Storage Objects: 0]  
    o- user:qcow ..... [Storage Objects: 0]  
    o- user:qemu ..... [Storage Objects: 1]  
      o- mydisk0 ..... [@drive (30.0GiB) deactivated]  
        o- alua ..... [ALUA Groups: 1]  
          o- default_tg_pt_gp ..... [ALUA state: Active/optimized]  
    o- user:rbd ..... [Storage Objects: 0]  
  o- iscsi ..... [Targets: 0]  
  o- loopback ..... [Targets: 0]  
  o- vhost ..... [Targets: 0]  
  o- xen_pvscsi ..... [Targets: 0]
```

qemu-tcmu example

```
//Create iscsi target
# IQN=iqn.2018-01.suse.lma
# targetcli /iscsi create wwn=$IQN
# targetcli /iscsi/$IQN/tpg1 set attribute authentication=0 \
generate_node_acls=1 demo_mode_write_protect=0 prod_mode_write_protect=0
```

```
MiWiFi-R3L-srv:/opt/vms # targetcli /iscsi create wwn=$IQN
Created target iqn.2018-01.suse.lma.
Created TPG 1.
Global pref auto_add_default_portal=true
Created default portal listening on all IPs (0.0.0.0), port 3260.
MiWiFi-R3L-srv:/opt/vms # targetcli ls
o- / ..... [..]
  o- backstores ..... [..]
    o- block ..... [Storage Objects: 0]
    o- fileio ..... [Storage Objects: 0]
    o- pscsi ..... [Storage Objects: 0]
    o- ramdisk ..... [Storage Objects: 0]
    o- user:glfs ..... [Storage Objects: 0]
    o- user:qcow ..... [Storage Objects: 0]
    o- user:qemu ..... [Storage Objects: 1]
    o- mydisk0 ..... [drive (30.0GiB) deactivated]
      o- alua ..... [ALUA Groups: 1]
        o- default_tg_pt_gp ..... [ALUA state: Active/optimized]
    o- user:rbt ..... [Storage Objects: 0]
  o- iscsi ..... [Targets: 1]
    o- iqn.2018-01.suse.lma ..... [TPGs: 1]
      o- tpg1 ..... [no-gen-acls, no-auth]
        o- acls ..... [ACLs: 0]
        o- luns ..... [LUNs: 0]
        o- portals ..... [Portals: 1]
          o- 0.0.0.0:3260 ..... [OK]
  o- loopback ..... [Targets: 0]
  o- vhost ..... [Targets: 0]
  o- xen_pvscsi ..... [Targets: 0]
```

qemu-tcmu example

//Export it as an iscsi LUN

targetcli /iscsi/\$IQN/tpg1/luns create storage_object=/backstores/user:qemu/mydisk0

```
MiWiFi-R3L-srv:/opt/vms # targetcli /iscsi/$IQN/tpg1/luns create storage_object=/backstores/user:qemu/mydisk0
Created LUN 0.
MiWiFi-R3L-srv:/opt/vms # targetcli ls
o- / ..... [..]
  o- backstores ..... [..]
    o- block ..... [Storage Objects: 0]
    o- fileio ..... [Storage Objects: 0]
    o- pscsi ..... [Storage Objects: 0]
    o- ramdisk ..... [Storage Objects: 0]
    o- user:glfs ..... [Storage Objects: 0]
    o- user:qcow ..... [Storage Objects: 0]
    o- user:qemu ..... [Storage Objects: 1]
      o- mydisk0 ..... [drive (30.0GiB) activated]
        o- alua ..... [ALUA Groups: 1]
          o- default_tg_pt_gp ..... [ALUA state: Active/optimized]
    o- user:rbt ..... [Storage Objects: 0]
  o- iscsi ..... [Targets: 1]
    o- iqn.2018-01.suse.lma ..... [TPGs: 1]
      o- tpg1 ..... [gen-acls, no-auth]
        o- acls ..... [ACLs: 0]
        o- luns ..... [LUNs: 1]
          o- lun0 ..... [user/mydisk0 (default_tg_pt_gp)]
        o- portals ..... [Portals: 1]
          o- 0.0.0.0:3260 ..... [OK]
  o- loopback ..... [Targets: 0]
  o- vhost ..... [Targets: 0]
  o- xen_pvscsi ..... [Targets: 0]
MiWiFi-R3L-srv:/opt/vms #
```

qemu-tcmu example

```
MiWiFi-R3L-srv:/opt/vms # iscsiadm -m discovery -t sendtargets -p 192.168.31.225
192.168.31.225:3260,1 iqn.2018-01.suse.lma
MiWiFi-R3L-srv:/opt/vms #
```

```
MiWiFi-R3L-srv:/opt/vms # lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda          8:0    0 232.9G  0 disk
├─sda1       8:1    0   549M  0 part
├─sda2       8:2    0   56.6G  0 part
├─sda3       8:3    0     1K  0 part
├─sda5       8:5    0     2G  0 part [SWAP]
└─sda6       8:6    0 163.8G  0 part /
sr0         11:0    1  1024M  0 rom
MiWiFi-R3L-srv:/opt/vms #
```

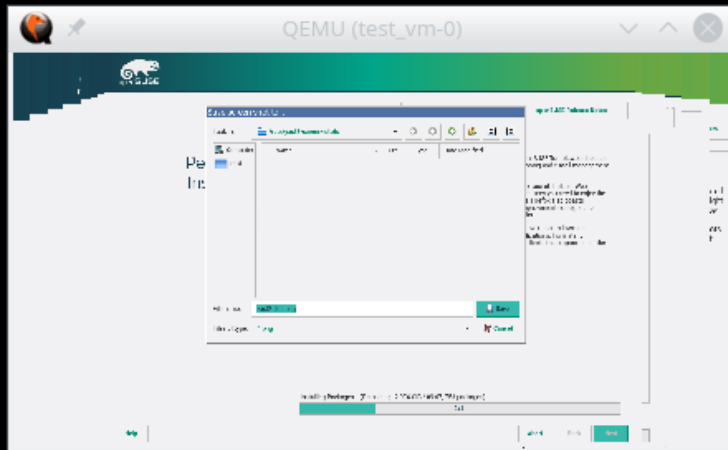
```
MiWiFi-R3L-srv:/opt/vms # cat tcmu-iscsi.sh
#!/bin/bash
/usr/bin/qemu-kvm \
-name 'test_vm' \
-m 2048 \
-nofeatures \
-monitor stdio \
-device cirrus-vga,id=video0 \
-display sdl \
-device ich9-usb-uhci1,id=usb1 \
-device usb-tablet \
-iscsi initiator-name=iqn.2018-01.asia.suse:01:76eea061d1b5 \
-drive file=/opt/isos/openSUSE-Tumbleweed-DVD-x86_64-20171218.iso,if=none,id=drive-cdrom0,format=raw \
-device ide-cd,bus=ide.0,unit=1,drive=drive-cdrom0,id=cdrom0,bootindex=1 \
-drive file=iscsi://192.168.31.225:3260/iqn.2018-01.suse.lma/0,if=none,id=drive-disk0 \
-device virtio-blk,drive=drive-disk0,id=disk0
```

qemu-tcmu example

```
(qemu) info block
drive-cdrom0 (#block189): /opt/isos/openSUSE-Tumbleweed-DVD-x86_64-20171218.iso (raw)
  Removable device: not locked, tray closed
  Cache mode:      writeback

drive-disk0 (#block354): iscsi://192.168.31.225:3260/iqn.2018-01.suse.lma/0 (raw)
  Cache mode:      writethrough
(qemu) [scsi/tcmu.c:0098] handle cmd: 0x28
[scsi/tcmu.c:0143] read at 0
[scsi/tcmu.c:0069] aio cb
[scsi/tcmu.c:0098] handle cmd: 0x28
[scsi/tcmu.c:0143] read at 0
```

```
MiWiFi-R3L-srv:/opt/vms # qemu-img info tumbleweed/disk0.qcow2
image: tumbleweed/disk0.qcow2
file format: qcow2
virtual size: 30G (32212254720 bytes)
disk size: 1.1G
cluster_size: 65536
Format specific information:
  compat: 1.1
  lazy refcounts: false
  refcount bits: 16
  corrupt: false
MiWiFi-R3L-srv:/opt/vms #
```



qemu-tcmu example

```
MiWiFi-R3L-srv:~ # iscsiadm -m node -T iqn.2018-01.suse.lma -p 192.168.31.225 -l
Logging in to [iface: default, target: iqn.2018-01.suse.lma, portal: 192.168.31.225,3260] (multiple)
Login to [iface: default, target: iqn.2018-01.suse.lma, portal: 192.168.31.225,3260] successful.
MiWiFi-R3L-srv:~ #
```

```
MiWiFi-R3L-srv:/opt/vms # lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda          8:0    0 232.9G  0 disk
├─sda1       8:1    0   549M  0 part
├─sda2       8:2    0   56.6G  0 part
├─sda3       8:3    0     1K  0 part
├─sda5       8:5    0     2G  0 part [SWAP]
└─sda6       8:6    0 163.8G  0 part /
sdb          8:16   0   30G  0 disk
├─sdb1       8:17   0     2G  0 part
└─sdb2       8:18   0   28G  0 part
sr0         11:0    1 1024M  0 rom
MiWiFi-R3L-srv:/opt/vms #
```

qemu-kvm ...

-device virtio-scsi-pci,id=scsi0 \

-drive file=/dev/sdb,format=raw,if=none,id=drive-scsi0000 \

-device scsi-hd,bus=scsi0.0,channel=0,scsi-id=0,lun=0,drive=drive-scsi0000,id=scsi0000

TCM vhost target

vhost 是专门为虚拟化设计的功能，它在 host 内核中实现一个支持 virtio 规范的 SCSI Target, Initiator 则是 guest 端的 vhost-scsi 控制器

TCM vhost example

```
MiWiFi-R3L-srv:/var/log/libvirt/qemu # targetcli /backstores/fileio create mydisk1 /opt/vms/disk1.raw 8G
Created fileio mydisk1 with size 8589934592
MiWiFi-R3L-srv:/var/log/libvirt/qemu #
MiWiFi-R3L-srv:/var/log/libvirt/qemu #
MiWiFi-R3L-srv:/var/log/libvirt/qemu # qemu-img info /opt/vms/disk1.raw
image: /opt/vms/disk1.raw
file format: raw
virtual size: 8.0G (8589934592 bytes)
disk size: 0
MiWiFi-R3L-srv:/var/log/libvirt/qemu #
```

```
# targetcli /vhost create wwn=naa.5001405162d5cf9d
```

```
# targetcli /vhost/naa.5001405162d5cf9d/tpg1/luns create /backstores/fileio/mydisk1
```


TCM vhost example

```
MiWiFi-R3L-srv:/opt/vms # targetcli ls
o- / ..... [....]
  o- backstores ..... [....]
    o- block ..... [Storage Objects: 0]
    o- fileio ..... [Storage Objects: 1]
      o- mydisk1 ..... [/opt/vms/disk1.raw (8.0GiB) write-back activated]
        o- alua ..... [ALUA Groups: 1]
          o- default_tg_pt_gp ..... [ALUA state: Active/optimized]
    o- pscsi ..... [Storage Objects: 0]
    o- ramdisk ..... [Storage Objects: 0]
    o- user:glfs ..... [Storage Objects: 0]
    o- user:qcow ..... [Storage Objects: 0]
    o- user:qemu ..... [Storage Objects: 1]
      o- mydisk0 ..... [@drive (30.0GiB) activated]
        o- alua ..... [ALUA Groups: 1]
          o- default_tg_pt_gp ..... [ALUA state: Active/optimized]
    o- user:rbd ..... [Storage Objects: 0]
  o- iscsi ..... [Targets: 1]
    o- iqn.2018-01.suse.lma ..... [TPGs: 1]
      o- tpg1 ..... [gen-acls, no-auth]
        o- acls ..... [ACLs: 0]
        o- luns ..... [LUNs: 1]
          o- lun0 ..... [user/mydisk0 (default_tg_pt_gp)]
        o- portals ..... [Portals: 1]
          o- 0.0.0.0:3260 ..... [OK]
    o- loopback ..... [Targets: 0]
  o- vhost ..... [Targets: 1]
    o- naa.5001405162d5cf9d ..... [TPGs: 1]
      o- tpg1 ..... [naa.5001405ec52c92e1, no-gen-acls]
        o- acls ..... [ACLs: 0]
        o- luns ..... [LUNs: 1]
          o- lun0 ..... [fileio/mydisk1 (/opt/vms/disk1.raw) (default_tg_pt_gp)]
    o- xen_pvscsi ..... [Targets: 0]
MiWiFi-R3L-srv:/opt/vms #
```

qemu-kvm ... -device **vhost-scsi-pci**,id=vhost-scsi0,**wwpn**=naa.5001405162d5cf9d