

Настройка подключений разного вида от инициатора до физического диска

Содержание

Содержание	1
LIO	2
На таргете	2
На инициаторе	3
NVMEOF	5
На таргете	5
На инициаторе	6
SPDK	7
Общая информация	7
На таргете	7
На инициаторе	9

LIO

Таргет lio встроен в ядро linux

Установить targetcli

apt install targetcli-fb

На таргете

Выбираем нужный диск

nvme list

```
/dev/nvme0n1 /dev/ng0n1 A074F563 WUS4BA138D5P3X1 1 3.84 TB / 3.84 TB 512 B + 0 B R2210003
/dev/nvme8n1 /dev/ng8n1 A074F681 WUS4BA138D5P3X1 1 3.84 TB / 3.84 TB 512 B + 0 B R2210003
/dev/nvme7n1 /dev/ng7n1 A074F58A WUS4BA138D5P3X1 1 3.84 TB / 3.84 TB 512 B + 0 B R2210003
/dev/nvme6n1 /dev/ng6n1 A074F70C WUS4BA138D5P3X1 1 3.84 TB / 3.84 TB 512 B + 0 B R2210003
/dev/nvme5n1 /dev/ng5n1 PHKD9113010V375AGN INTEL SSDPD21K375GA 1 375.08 GB / 375.08 GB 512 B + 0 B E2010470
/dev/nvme4n1 /dev/ng4n1 PHKD911300RU375AGN INTEL SSDPD21K375GA 1 375.08 GB / 375.08 GB 512 B + 0 B E2010470
/dev/nvme3n1 /dev/ng3n1 A074F676 WUS4BA138D5P3X1 1 3.84 TB / 3.84 TB 512 B + 0 B R2210003
/dev/nvme2n1 /dev/ng2n1 A074F5DC WUS4BA138D5P3X1 1 3.84 TB / 3.84 TB 512 B + 0 B R2210003
/dev/nvme18n1 /dev/ng18n1 A074F65B WUS4BA138D5P3X1 1 3.84 TB / 3.84 TB 512 B + 0 B R2210003
/dev/nvme17n1 /dev/ng17n1 A074F606 WUS4BA138D5P3X1 1 3.84 TB / 3.84 TB 512 B + 0 B R2210003
/dev/nvme16n1 /dev/ng16n1 PHKD912500D5375AGN INTEL SSDPD21K375GA 1 375.08 GB / 375.08 GB 512 B + 0 B E2010470
/dev/nvme10n2 /dev/ng10n2 PHKD9113005C375AGN INTEL SSDPD21K375GA 1 375.08 GB / 375.08 GB 512 B + 0 B E2010470
/dev/nvme14n1 /dev/ng14n1 A074F757 WUS4BA138D5P3X1 1 3.84 TB / 3.84 TB 512 B + 0 B R2210003
/dev/nvme13n1 /dev/ng13n1 PHKD9113005Z375AGN INTEL SSDPD21K375GA 1 375.08 GB / 375.08 GB 512 B + 0 B E2010470
/dev/nvme12n1 /dev/ng12n1 A074F621 WUS4BA138D5P3X1 1 3.84 TB / 3.84 TB 512 B + 0 B R2210003
/dev/nvme11n1 /dev/ng11n1 A074F680 WUS4BA138D5P3X1 1 3.84 TB / 3.84 TB 512 B + 0 B R2210003
/dev/nvme1n1 /dev/ng1n1 PHKD911300US375AGN INTEL SSDPD21K375GA 1 375.08 GB / 375.08 GB 512 B + 0 B E2010470
/dev/nvme0n1 /dev/ng0n1 A074F553 WUS4BA138D5P3X1 1 3.84 TB / 3.84 TB 512 B + 0 B R2210003
```

Через fdisk создаем раздел на /dev/nvme0n1

И уже к этому разделу создаем подключение через таргет

/backstores/block create storage01 /dev/nvme0n1p1

/iscsi create

```
o- backstores ..... [Targets: 1]
o- block ..... [Storage Objects: 1]
  o- storage01 ..... [/dev/nvme0n1 (3.5TiB) write-thru activated]
    o- alua ..... [ALUA Groups: 1]
      o- default_tg_pt_gp ..... [ALUA state: Active/optimized]
    o- fileio ..... [Storage Objects: 0]
    o- pscsi ..... [Storage Objects: 0]
    o- ramdisk ..... [Storage Objects: 0]
o- iscsi ..... [Targets: 1]
o- iqn.2003-01.org.linux-iscsi.node20.x8664:sn.173a5daf8572 ..... [TPGs: 1]
  o- tpg1 ..... [no-gen-acls, no-auth]
    o- acls ..... [ACLs: 1]
      o- iqn.2004-10.com.ubuntu:01:4f598a60af82 ..... [Mapped LUNs: 1]
        o- mapped_lun0 ..... [lun0 block/storage01 (rw)]
      o- luns ..... [LUNs: 1]
        o- lun0 ..... [block/storage01 (/dev/nvme0n1) (default_tg_pt_gp)]
      o- portals ..... [Portals: 1]
        o- 192.168.0.5:3260 ..... [OK]
o- loopback ..... [Targets: 0]
o- srpt ..... [Targets: 0]
o- vhost ..... [Targets: 0]
/> cd iscsi/iqn.2003-01.org.linux-iscsi.node20.x8664:sn.173a5daf8572/tpg1/
```

cd iscsi/iqn.2003-01.org.linux-iscsi.node20.x8664:sn.173a5daf8572/tpg1/

Войти внутрь и сделать настройку

```
set parameter AuthMethod=None
```

```
set attribute authentication=0
```

```
acls/ create iqn.2004-10.com.ubuntu:01:4f598a60af82
```

Указываем инициатор с которого будем подключаться

```
luns/ create /backstores/block/storage01
```

Указываем лун

```
portals/ create 192.168.0.5
```

Указываем ip где находится сервер

```
cd /
```

```
saveconfig
```

Для того чтобы очистить настройки

```
clearconfig confirm=True
```

На инициаторе

Настроиваем /etc/iscsi/iscsid.conf

```
node.session.queue_depth = 256
```

Устанавливаем количество сессий на 8, иначе будет работать в один поток, что будет хуже по производительности

```
node.session.nr_sessions = 8
```

1 поток пишет до 35kIOPS, 8 потоков показывают 280kIOPS, 16 потоков показали 330kIOPS

Минимальная настройка для /etc/multipath.conf

Мультипас нужен для того чтобы восемь сессий, которые будут созданы от подключения таргета, будут определены как отдельные диски, чтобы это исправить выставляем политику группировки path_grouping_policy multibus

```
defaults {
```

```
    path_grouping_policy  multibus
```

```
    user_friendly_names  yes
```

```
    polling_interval      5
```

```
}
```

8 сессий так и будут отображаться как отдельные диски, но у нас будет единая точка входа

/dev/mapper/mpathb

После настроек перезагружаем сервисы

systemctl restart multipathd.service

systemctl restart iscsid.service

Смотрим есть ли на таргете что-то

iscsiadm -m discovery -t st -p 192.168.0.5


Подключаемся

iscsiadm -m node -l

```
sdc                8:32    0    3.5T    0 disk
└─mpatha          253:0    0    3.5T    0 mpath
sdd                8:48    0    3.5T    0 disk
└─mpatha          253:0    0    3.5T    0 mpath
sde                8:64    0    3.5T    0 disk
└─mpatha          253:0    0    3.5T    0 mpath
sdf                8:80    0    3.5T    0 disk
└─mpatha          253:0    0    3.5T    0 mpath
sdg                8:96    0    3.5T    0 disk
└─mpatha          253:0    0    3.5T    0 mpath
sdh                8:112   0    3.5T    0 disk
└─mpatha          253:0    0    3.5T    0 mpath
sdi                8:128   0    3.5T    0 disk
└─mpatha          253:0    0    3.5T    0 mpath
sdj                8:144   0    3.5T    0 disk
└─mpatha          253:0    0    3.5T    0 mpath
```

ls /dev/mapper

```
root@node19:~/streamlit# ll /dev/mapper/
total 0
drwxr-xr-x  2 root root   100 Dec 11 14:16 ./
drwxr-xr-x 21 root root  5020 Dec 11 14:11 ../
crw-----  1 root root   10, 236 Dec  6 11:01 control
lrwxrwxrwx  1 root root    7 Dec 11 14:16 mpathb -> ../dm-0
lrwxrwxrwx  1 root root    7 Dec  6 11:01 ubuntu--vg-ubuntu--lv -> ../dm-1
```



В fio добавляем

[writetest]

filename=/dev/mapper/mpathb

```

direct=1
rw=randwrite
bs=4k
size=100%
rate_iops=300000
ioengine=libaio
iodepth=64
ramp_time=3
numjobs=8
runtime=300
name=iops-test-job

```

```

iops-test-job: (g=0): rw=randwrite, bs=(R) 4096B-4096B, (W) 4096B-4096B, (T) 4096B-4096B, ioengine=libaio, iodepth=64
...
fio-3.28
Starting 8 processes
Jobs: 8 (f=8), 0-2400000 IOPS: [w(8)][5.6%][w=1090MiB/s][w=279k IOPS][eta 04m:47s]

```

NVMEOF

Nvmeof over tcp встроен в ядро.

Для включения требуется загрузить модули на обеих взаимодействующих сторонах, после чего в `/sys/kernel/config/` будет создана папка `nvmef` с необходимой структурой внутри

```
modprobe nvmef
```

```
modprobe nvmef-tcp
```

На таргете

```
mkdir /sys/kernel/config/nvmef/ports/1
```

```
cd /sys/kernel/config/nvmef/ports/1
```

Указываем адрес сервера

```
echo 192.168.0.5 | tee -a addr_traddr > /dev/null
```

```
echo tcp | tee -a addr_trtype > /dev/null
```

Порт должен быть открыт в файерволе

```
echo 4420 | tee -a addr_trsvcid > /dev/null
echo ipv4 | tee -a addr_adrfam > /dev/null
```

```
cd /sys/kernel/config/nvmet/subsystems; mkdir test; cd test
echo -n 1 > /sys/kernel/config/nvmet/subsystems/test/attr_allow_any_host
```

Добавляем один диск

```
cd namespaces ; mkdir 1; cd 1
```

```
echo -n /dev/nvme0n1 > device_path
```

```
echo -n 1 > enable
```

```
ln -s /sys/kernel/config/nvmet/subsystems/test/ /sys/kernel/config/nvmet/ports/1/subsystems/test
```

На инициаторе

Затем переходим на сервер инициатор и подключаем диски

```
nvme connect -n test -t tcp -a 192.168.0.5 -s 4420
```

Node	SN	Model	Namespace	Usage	Format	FW Rev
/dev/nvme0n1	17faf72e81fe5dd4eecd	Linux	2	3.84 TB / 3.84 TB	512 B + 0 B	6.2.0-37
/dev/nvme0n2	17faf72e81fe5dd4eecd	Linux	3	3.84 TB / 3.84 TB	512 B + 0 B	6.2.0-37
/dev/nvme0n3	17faf72e81fe5dd4eecd	Linux	4	3.84 TB / 3.84 TB	512 B + 0 B	6.2.0-37

Fio для одного диска

```
0[| 86.6%] 4[| 71.5%] 8[| 0.0%] 12[| 0.7%]
1[| 51.4%] 5[| 38.1%] 9[| 0.0%] 13[| 0.0%]
2[| 71.5%] 6[| 67.1%] 10[| 0.0%] 14[| 0.0%]
3[| 79.9%] 7[| 19.0%] 11[| 0.0%] 15[| 0.7%]
Mem[| 21.6G/378G] Tasks: 39, 49 thr, 308 kthr; 6 running
Swap[| 496K/8.00G] Load average: 1.90 0.60 0.37
              uptime: 6 days, 05:10:10
```

Для трех дисков по 8 джобов на каждом на запись

```
0[| 70.7%] 4[| 100.0%] 8[| 0.0%] 12[| 0.0%]
1[| 0.0%] 5[| 100.0%] 9[| 0.7%] 13[| 0.0%]
2[| 75.9%] 6[| 96.0%] 10[| 0.0%] 14[| 1.3%]
3[| 100.0%] 7[| 73.2%] 11[| 0.0%] 15[| 0.0%]
Mem[| 21.7G/378G] Tasks: 39, 49 thr, 308 kthr; 8 running
Swap[| 496K/8.00G] Load average: 4.92 1.92 2.02
              uptime: 6 days, 05:05:44
```

```
Starting 8 processes
Jobs: 8 (f=1), 0-2400000 IOPS: [f(5),w(1),f(2)][100.0%][w=1290MiB/s][w=330k IOPS][eta 00m:00s]
iops-test-job: (groupid=0, jobs=8): err= 0: pid=177456: Tue Dec 12 12:46:48 2023
write: IOPS=210k, BW=822MiB/s (862MB/s)(241GiB/300002msec); 0 zone resets
slat (usec): min=2, max=17650, avg=31.27, stdev=180.70
clat (usec): min=24, max=44954, avg=2400.84, stdev=2142.13
lat (usec): min=60, max=44961, avg=2432.26, stdev=2163.82
```

```
Starting 8 processes
Jobs: 8 (f=1), 0-2400000 IOPS: [f(5),w(1),f(2)][100.0%][w=930MiB/s][w=238k IOPS][eta 00m:00s]
iops-test-job: (groupid=0, jobs=8): err= 0: pid=177402: Tue Dec 12 12:46:46 2023
  write: IOPS=220k, BW=860MiB/s (902MB/s)(252GiB/300003msec); 0 zone resets
    slat (usec): min=2, max=16978, avg=28.24, stdev=151.52
    clat (usec): min=11, max=37955, avg=2295.01, stdev=1777.14
    lat (usec): min=58, max=37961, avg=2323.43, stdev=1791.68
  clat percentiles (usec):
```

```
Starting 8 processes
Jobs: 8 (f=1), 0-2400000 IOPS: [f(7),w(1)][100.0%][w=1512MiB/s][w=387k IOPS][eta 00m:00s]
iops-test-job: (groupid=0, jobs=8): err= 0: pid=177491: Tue Dec 12 12:46:52 2023
  write: IOPS=213k, BW=834MiB/s (874MB/s)(244GiB/300001msec); 0 zone resets
    slat (usec): min=2, max=17612, avg=30.21, stdev=168.58
    clat (usec): min=2, max=42217, avg=2366.93, stdev=1987.19
    lat (usec): min=62, max=42222, avg=2397.29, stdev=2005.91
```

640k IOPS в сумме

На двух дисках получается 600k IOPS

SPDK

Общая информация

<https://spdk.io/>

Для работы spdk требуется скачать его

```
git clone https://github.com/spdk/spdk
```

При возникновении ошибок потребуется рекурсивно клонировать репозиторий

```
git clone --recurse-submodules https://github.com/spdk/spdk
```

Установка дополнительных пакетов и инструментов, подготовка перед компиляцией spdk

```
./scripts/pkgdep.sh
```

Эта библиотека подменяет собой стандартный стек протокола работы с дисками, но не имеет отношения к сетевому взаимодействию.

То есть, все настройки выполняются на стороне таргета, для инициатора эти настройки остаются невидимы

На таргете

Конфигурируем с поддержкой rdma

```
./configure --with-rdma  
make -j8
```

Запускаем настройку системы, уберем из nvme list устройства, чтобы вернуть нужно запустить этот скрипт с опцией reset
scripts/setup.sh

Запускаем таргет

```
./build/bin/nvmf_tgt &  
./scripts/rpc.py nvmf_create_transport -t RDMA -u 8192 -i 131072 -c 8192  
включаем rdma для передачи
```

```
./scripts/rpc.py bdev_nvme_attach_controller -b nvme0 -a 8e:00:00 -t pcie -x multipath  
Подключаем диск по BDF иключаем multipath(можно попробовать и без него)
```

```
./scripts/rpc.py nvmf_create_subsystem nqn.2016-06.io.spdk:cnode1 -a -s SPDK0000000000000001 -d SPDK_Controller1  
Создаем название таргета, и название контроллера, под которым будет отображаться на инициаторе в nvme list
```

```
./scripts/rpc.py nvmf_subsystem_add_ns nqn.2016-06.io.spdk:cnode1 nvme0n1  
Добавляем диск в таргет
```

```
./scripts/rpc.py nvmf_subsystem_add_listener nqn.2016-06.io.spdk:cnode1 -t rdma -a 192.168.0.5 -s 4422
```


На инициаторе

`nvme connect -t rdma -n nqn.2016-06.io.spdk:cnode1 -a 192.168.0.6 -s 4422`

Через `nvme list` можно будет увидеть добавленный диск

```
root@node19:~/spdk# nvme list
Node                               SN                               Model                               Namespace Usage                               Format                               FW Rev
-----
/dev/nvme0n1                       SPDK0000000000000001           SPDK_Controller1                   1                               3.84 TB / 3.84 TB                   512 B + 0 B                       24.01
root@node19:~/spdk#
```

Нагрузку на него можно давать через `fio`

Нагрузка с трех дисков на таргет

```
0[|||||] 0.0% 4[|||||] 0.0% 8[|||||] 0.0% 12[|||||] 0.0%
1[|||||] 0.0% 5[|||||] 0.0% 9[|||||] 0.0% 13[|||||] 0.0%
2[|||||] 0.0% 6[|||||] 0.0% 10[|||||] 0.7% 14[|||||] 0.0%
3[|||||] 0.0% 7[|||||] 0.0% 11[|||||] 0.0% 15[|||||] 0.0%
Mem[|||||]
Swap[|||||]
21.4G/378G Tasks: 34, 40 thr, 284 kthr; 2 running
490k/8.00G Load average: 1.27 1.16 1.06
Uptime: 7 days, 07:16:23
```

```
fio-3.28
Starting 8 processes
Jobs: 8 (f=1), 0-2400000 IOPS: [f(1),w(1),f(6)][100.0%][w=1641MiB/s][w=420k IOPS][eta 00m:00s]
iops-test-job: (groupid=0, jobs=8): err= 0: pid=181028: Wed Dec 13 14:12:50 2023
write: IOPS=236k, BW=921MiB/s (966MB/s)(270GiB/300002msec); 0 zone resets
slat (usec): min=2, max=9114, avg=25.45, stdev=104.76
clat (usec): min=34, max=11921, avg=2144.02, stdev=894.08
lat (usec): min=49, max=11930, avg=2169.66, stdev=899.93
clat percentiles (usec):
```

```
Starting 8 processes
Jobs: 1 (f=1), 0-300000 IOPS: [w(1),_(7)][12.6%][w=1287MiB/s][w=329k IOPS][eta 35m:00s]
iops-test-job: (groupid=0, jobs=8): err= 0: pid=180999: Wed Dec 13 14:12:48 2023
write: IOPS=246k, BW=961MiB/s (1007MB/s)(281GiB/300002msec); 0 zone resets
slat (usec): min=2, max=10893, avg=23.86, stdev=97.25
clat (usec): min=23, max=19530, avg=2056.45, stdev=830.02
lat (usec): min=31, max=19538, avg=2080.52, stdev=835.30
clat percentiles (usec):
```

```
fio-3.28
Starting 8 processes
Jobs: 8 (f=1), 0-2400000 IOPS: [w(1),f(7)][100.0%][w=893MiB/s][w=229k IOPS][eta 00m:00s]
iops-test-job: (groupid=0, jobs=8): err= 0: pid=180968: Wed Dec 13 14:12:44 2023
write: IOPS=228k, BW=892MiB/s (936MB/s)(261GiB/300002msec); 0 zone resets
slat (usec): min=2, max=9425, avg=26.39, stdev=109.19
clat (usec): min=21, max=24185, avg=2212.79, stdev=925.49
lat (usec): min=50, max=24191, avg=2239.34, stdev=931.50
clat percentiles (usec):
```

710kIOPS в сумме

Запись блоками 4к