

## 02. Модуль 2. Задание 2

### Задание

#### Сконфигурируйте файловое хранилище:

- ▶ При помощи трёх дополнительных дисков, размером 1Гб каждый, на HQ-SRV сконфигурируйте дисковый массив уровня 5
- ▶ Имя устройства – md0, конфигурация массива размещается в файле /etc/mdadm.conf
- ▶ Обеспечьте автоматическое монтирование в папку /raid5
- ▶ Создайте раздел, отформатируйте раздел, в качестве файловой системы используйте ext4
- ▶ Настройте сервер сетевой файловой системы(nfs), в качестве папки общего доступа выберите /raid5/nfs, доступ для чтения и записи для всей сети в сторону HQ-CLI
- ▶ На HQ-CLI настройте автомонтирование в папку /mnt/nfs
- ▶ Основные параметры сервера отметьте в отчёте

### Реализация



Пример создания программного RAID5 массива.

RAID массив будем размещать на разделы, это дает больше гибкости и безопасности.

#### Установка mdadm

##### Установим утилиту mdadm

```
1 | # dnf install mdadm -y
```

##### Выводим список подключенных дисков и их имена.

```
1 | # lsblk
```

```
[root@hq-srv ~]#  
[root@hq-srv ~]#  
[root@hq-srv ~]# lsblk  
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINTS  
sda          8:0    0   12G  0 disk  
├─sda1       8:1    0    1M  0 part  
├─sda2       8:2    0    1G  0 part /boot  
├─sda3       8:3    0   1,2G  0 part [SWAP]  
└─sda4       8:4    0   9,8G  0 part /  
sdb          8:16   0    1G  0 disk  
sdc          8:32   0    1G  0 disk  
sdd          8:48   0    1G  0 disk  
zram0       252:0    0   1,9G  0 disk [SWAP]  
[root@hq-srv ~]#
```

## Создание таблицы разделов



**fdisk** - общее название системных утилит для управления разделами жёсткого диска.

На каждом диске ( `/dev/sdb`, `/dev/sdc`, `/dev/sdd` ) создадим по 1 разделу (заяв все место)

Таблица разделов на `/dev/sdb`

Для разбиения диска, запускаем **fdisk** с именем устройства:

```
1 | # fdisk /dev/sdb
```

Командная строка изменится, и откроется диалоговое **fdisk** для ввода команд.



Посмотреть доступные команды можно выполнив команду `m`: **Command (m for help): m**

Будем использовать таблицу разделов **GPT**

Вводим **g** чтобы создать новую пустую таблицу разделов **GPT** :

```
1 | Command (m for help): g
```

Вводим команду **n** чтобы создать новый раздел:

1 | Command (m for help): n

Будет предложено ввести номер раздела. Жмем «Enter», чтобы использовать значение по умолчанию (1)

Далее необходимо указать первый сектор, используем значение по умолчанию. Жмем «Enter»

При следующем запросе необходимо ввести последний сектор, Жмем «Enter», так как используем весь диск

Сохраняем изменения, запустив команду **w** (Жмем **w**)

```
[root@hq-srv ~]#  
[root@hq-srv ~]#  
[root@hq-srv ~]# fdisk /dev/sdb  
  
Добро пожаловать в fdisk (util-linux 2.38.1).  
Изменения останутся только в памяти до тех пор, пока вы не решите записать их.  
Будьте внимательны, используя команду write.  
  
Устройство не содержит стандартной таблицы разделов.  
Создана новая метка DOS с идентификатором 0xa5cf1b64.  
  
Команда (m для справки): g  
Created a new GPT disklabel (GUID: A5405480-C8E9-5A49-98AD-69A7EA963632).  
  
Команда (m для справки): n  
Номер раздела (1-128, default 1):  
Первый сектор (2048-2097118, default 2048):  
Last sector, +/-sectors or +/-size{K,M,G,T,P} (2048-2097118, default 2095103):  
  
Создан новый раздел 1 с типом 'Linux filesystem' и размером 1022 MiB.  
  
Команда (m для справки): w
```

Команда запишет таблицу на диск и выйдет из меню **fdisk**.

Ядро прочитает таблицу разделов устройства без перезагрузки системы.



Аналогично создаем разделы на других дисках **/dev/sdc**, **/dev/sdd**

```
[root@hq-srv ~]#
[root@hq-srv ~]#
[root@hq-srv ~]# lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINTS
sda          8:0    0   12G  0 disk
├─sda1       8:1    0    1M  0 part
├─sda2       8:2    0    1G  0 part /boot
├─sda3       8:3    0   1,2G  0 part [SWAP]
└─sda4       8:4    0   9,8G  0 part /
sdb          8:16   0    1G  0 disk
└─sdb1       8:17   0 1022M  0 part
sdc          8:32   0    1G  0 disk
└─sdc1       8:33   0 1022M  0 part
sdd          8:48   0    1G  0 disk
└─sdd1       8:49   0 1022M  0 part
zram0       252:0    0   1,9G  0 disk [SWAP]
[root@hq-srv ~]#
```

## Создание программного RAID5

Для сборки массива применяем следующую команду:

```
1 | # mdadm --create --verbose /dev/md0 -l 5 -n 3 /dev/sdb1 /dev/sdc1 /dev/sdd1
```

где:

/dev/md0 — устройство RAID, которое появится после сборки;

-l 5 — уровень RAID;

-n 3 — количество дисков, из которых собирается массив;

/dev/sdb1 /dev/sdc1 /dev/sdd1 — сборка выполняется из разделов дисков sdb, sdc, sdd.

Система задаст контрольный вопрос, хотим ли мы продолжить и создать RAID — нужно ответить **y** :

Мы должны увидеть что-то на подобие:

```
[root@hq-srv ~]# mdadm --create --verbose /dev/md0 -l 5 -n 3 /dev/sdb1 /dev/sdc1 /dev/sdd1
mdadm: layout defaults to left-symmetric
mdadm: layout defaults to left-symmetric
mdadm: chunk size defaults to 512K
mdadm: size set to 1044480K
mdadm: Defaulting to version 1.2 metadata
mdadm: array /dev/md0 started.
[root@hq-srv ~]# _
```

Вводим команду:

```
1 | # lsblk
```

и видим информацию о том, что у дисков `sdb`, `sdc`, `sdd` появился раздел `md0` :

```
[root@hq-srv ~]#
[root@hq-srv ~]#
[root@hq-srv ~]# lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE  MOUNTPOINTS
sda          8:0    0   12G  0 disk
├─sda1       8:1    0    1M  0 part
├─sda2       8:2    0    1G  0 part  /boot
├─sda3       8:3    0   1,2G  0 part  [SWAP]
└─sda4       8:4    0   9,8G  0 part  /
sdb          8:16   0    1G  0 disk
├─sdb1       8:17   0  1022M  0 part
└─md0        9:0    0    2G  0 raid5
sdc          8:32   0    1G  0 disk
├─sdc1       8:33   0  1022M  0 part
└─md0        9:0    0    2G  0 raid5
sdd          8:48   0    1G  0 disk
├─sdd1       8:49   0  1022M  0 part
└─md0        9:0    0    2G  0 raid5
zram0       252:0   0   1,9G  0 disk  [SWAP]
[root@hq-srv ~]# _
```

Посмотреть состояние всех RAID можно командой:

```
1 | # cat /proc/mdstat
```

```
[root@hq-srv ~]#
[root@hq-srv ~]#
[root@hq-srv ~]# cat /proc/mdstat
Personalities : [raid6] [raid5] [raid4]
md0 : active raid5 sdd1[3] sdc1[1] sdb1[0]
      2088960 blocks super 1.2 level 5, 512k chunk, algorithm 2 [3/3] [UUU]

unused devices: <none>
[root@hq-srv ~]# _
```

Следующая команда покажет детальную информацию о массиве:

```
1 | # mdadm -D /dev/md0
```

## Создание файла `mdadm.conf`

В файле `mdadm.conf` находится информация о RAID-массивах и компонентах, которые в них входят. Для его создания выполняем следующие команды.

Сохраняем конфигурацию массива:

```
1 | # mdadm --detail --scan --verbose >> /etc/mdadm.conf
```

## Монтирование файловой системы массива

Когда разделы созданы их необходимо отформатировать и смонтировать их в дереве каталогов системы.

Отформатируем разделы в `ext4`

```
1 | # mkfs.ext4 /dev/md0
```

В корне системы создадим директорию `/raid5`

```
1 | # mkdir /raid5
```

Примотируем к RAID директорию `/raid5` командой:

```
1 | # mount /dev/md0 /raid5
```

Результат проверяем командой:

```
1 | # lsblk -f
2 |
3 | # ls -l /raid5
```

Чтобы разделы монтировались при загрузке системы, добавляем в `fstab`

Открываем `fstab` :

```
1 | # nano /etc/fstab
```

И добавляем строки

```
1 | /dev/md0    /raid5    ext4    defaults    0    0
```

Проверяем примонтированные разделы:

```
1 | # df -h
```

## Сервер nfs

На **HQ-SRV** устанавливаем компонент для серверной части **NFS** :

### Установка

```
1 | # dnf install nfs-utils nfs4-acl-tools
```

Создаем папку общего доступа **/raid5/nfs**

```
1 | # mkdir /raid5/nfs
```

Задаем права на общую папку

```
1 | # chmod -R 777 /raid5/nfs
```

Открываем на редактирование файл **/etc/exports** который содержит информацию о каталогах, экспортируемых с сервера.:

```
1 | # nano /etc/exports
```

Создаем шару из каталога **/raid5/nfs** , которая будет доступна для всех узлов сети **192.168.100.65/28** .

Добавим в него строку:

```
1 | /raid5/nfs      192.168.100.65/28(rw,sync,no_root_squash,subtree_check)
2 |
3 | или
4 |
5 | /raid5/nfs      192.168.100.65/28(rw)
```

#### где:

**rw** — разрешить чтение и запись.

**sync / async** — осинхронный/асинхронный режим доступа. Опция **sync** указывает, что сервер должен отвечать на запросы только после записи на диск изменений, выполненных этими запросами. Опция **async** указывает серверу не ждать записи информации на диск, что повышает производительность, но снижает надежность.

**root\_squash / no\_root\_squash** — подменять запросы от **root** на анонимные или не делать этого.



`subtree_check` / `no_subtree_check` — проверять попытки пользователя выйти за пределы экспортированной папки или отключить данную проверку.

Добавляем в автозагрузку и запускаем **NFS** сервер

```
1 | # systemctl enable --now nfs-server.service
```



Выполняем экспорт данных (перечитаем конфигурационный файл `/etc/exports`, чтобы сервер начал отдавать настроенные шары):

```
1 | # exportfs -a
```

## Автомонтирование

На **HQ-CLI** устанавливаем компонент для клиентской части **NFS** :

```
1 | # dnf install nfs-utils
```

Создаем директорию для автомонтирования `/mnt/nfs`

```
1 | # mkdir /mnt/nfs
```

Пробуем примонтировать шару:

```
1 | # mount -t nfs 192.168.100.2:/raid5/nfs /mnt/nfs
```

подключимся к серверу с IP-адресом `192.168.100.2`; каталог, который монтируем `/raid5/nfs` (который настроен в файле `exports`); монтирование будет выполнено в каталог `/mnt/nfs`.

Для автоматического монтирования на **HQ-CLI** после перезагрузки, используем `fstab`.

Открываем файл `fstab` :

```
1 | # nano /etc/fstab
```

И добавляем строку:

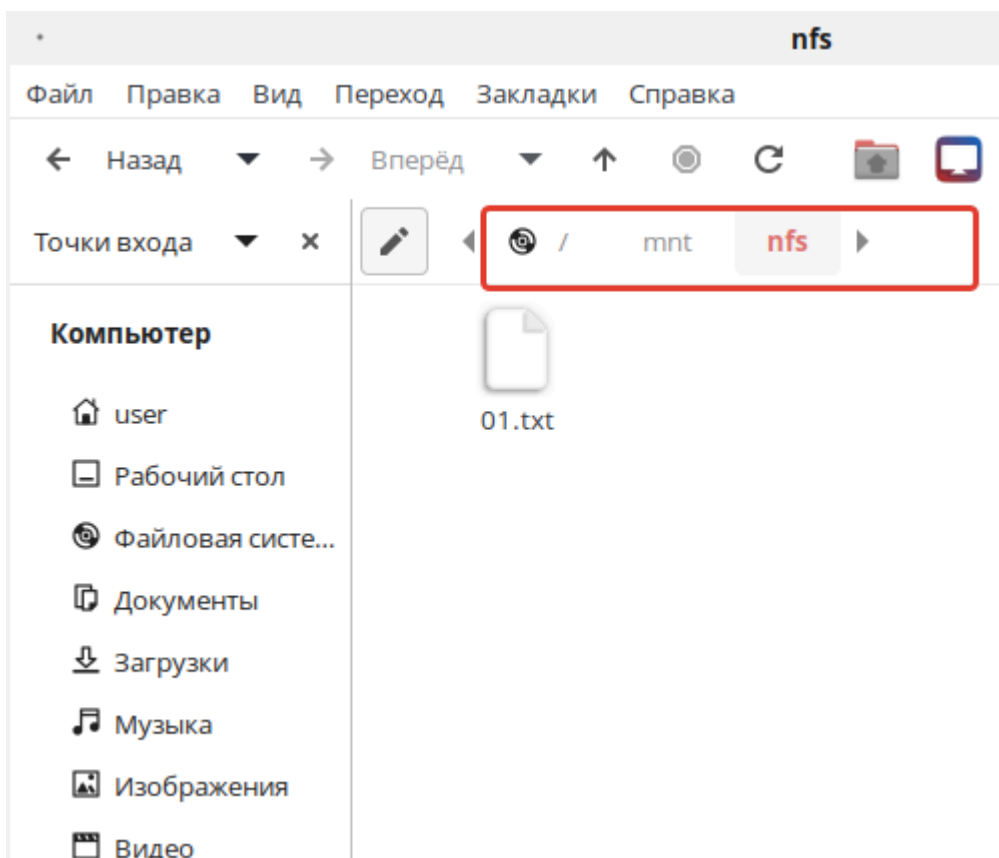


1 | 192.168.100.2:/raid5/nfs      /mnt/nfs      nfs      auto      0 0

где **192.168.100.2** — адрес сервера NFS; **/mnt/nfs** — каталог, куда будет примонтирована шара.



### Проверяем



```
user@hq-ci:~  
[user@hq-ci ~]$ df -h  
Файловая система    Размер  Использовано  Дост  Использовано%  Смонтировано в  
devtmpfs             4,0М      0             0%    0% /dev  
tmpfs                985М     18М          967М    2% /dev/shm  
tmpfs                394М     3,3М          391М    1% /run  
/dev/sda4            14G       8,9G          3,6G   72% /  
tmpfs                985М     340K          985М    1% /tmp  
/dev/sda2            974М     260М          648М   29% /boot  
tmpfs                197М     136K          197М    1% /run/user/12882  
01104  
tmpfs                197М     120K          197М    1% /run/user/1000  
192.168.100.2:/raid5/nfs 2,0G      0             1,9G    0% /mnt/nfs  
[user@hq-ci ~]$
```

Содержимое доступно в соответствии с Всеобщее достояние, от Кабинет 2.20. | Powered by [Wiki.js](#)