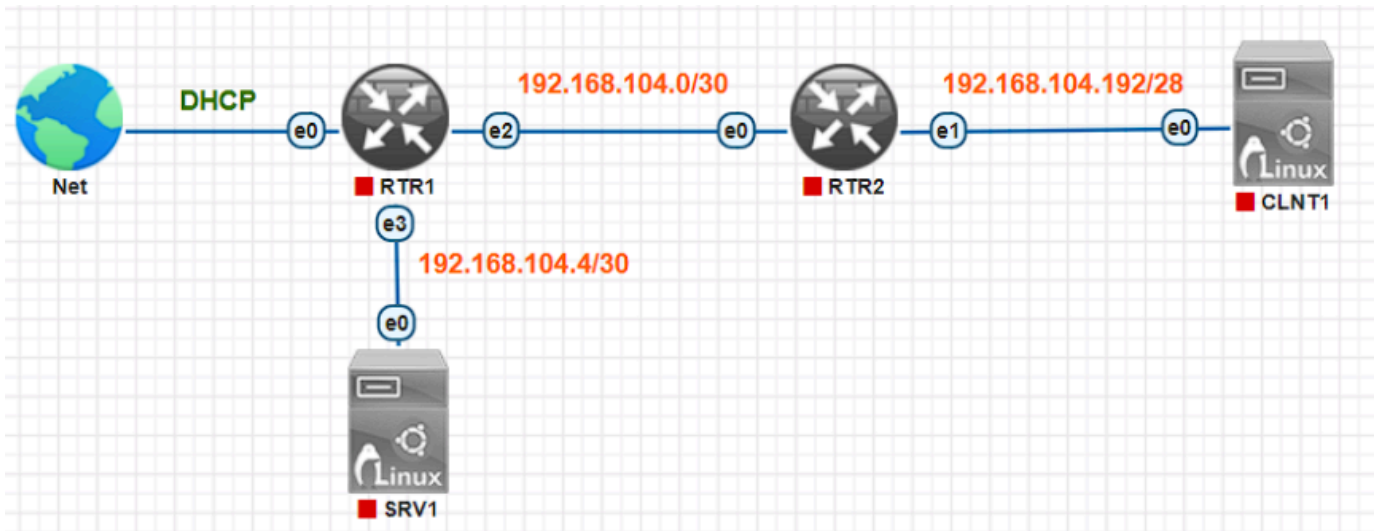


# 11-PR-RAID

## Задание



## Базовая настройка

1. Настройте имена устройств в соответствии с топологией
2. Настройте часовой пояс Баку

```
hostnamectl set-hostname SRV,CLNT  
timedatectl set-timezone Asia/Baku
```

## Настройка RAID массивов

### 1. Установите службу **mdadm** на **SRV**

1. При установке будет так же установлен **postfix**, и предложит произвести преднастройку.
2. В этот момент необходимо пропустить нажатием **Enter** всех вопросов интерфейса

```
apt install mdadm -y
```

### 2. Изучите список дисков доступных на **SRV**

1. Для просмотра списка доступных дисков используется команда: **lsblk**

```

root@SRV:~# lsblk
NAME        MAJ:MIN RM   SIZE RO TYPE MOUNTPOINTS
fd0          2:0    1     4K  0 disk
sda          8:0    0    10G  0 disk
├─sda1       8:1    0     9G  0 part /
├─sda2       8:2    0     1K  0 part
└─sda5       8:5    0    975M  0 part [SWAP]
sdb          8:16   0   93.9M  0 disk
sr0         11:0    1 414.1M  0 rom  /mnt/repo
vda         254:0    0    10G  0 disk
vdb         254:16   0    10G  0 disk
vdc         254:32   0    10G  0 disk
vdd         254:48   0    10G  0 disk

```

2. В данном выводе мы наблюдаем:

1. Диск **sda** имеющий разделы sda1, sda2, sda5
  1. sda1 - примонтирован к корневому разделу
  2. sda5 - представляет собой SWAP
2. **sdb** - диск с загрузчиком (НЕ ЗАДЕЙСТВУЙТЕ ЕГО)
3. **sr0** - подключенный к вашей VM образ диска, содержащий локальный репозиторий
4. Диски **vd[abcd]** - не используемые в настоящий момент диски размером по 10G
  1. Именно с ними мы и будем дальше работать

### 3. Создайте RAID массив нулевого уровня

1. Создать RAID массив нулевого уровня:

```

mdadm --create -v /dev/md0 --level=0 --raid-devices=2 /dev/vd[ab]
# mdadm: chunk size defaults to 512K
# mdadm: Defaulting to version 1.2 metadata
# mdadm: array /dev/md0 started.

```

2. Выполните снова `lsblk`

```

vda         254:0    0    10G  0 disk
└─md0        9:0    0    20G  0 raid0
vdb         254:16   0    10G  0 disk
└─md0        9:0    0    20G  0 raid0

```

3. Выполните просмотр статуса работы RAID массива: `cat /proc/mdstat`

```
root@SRV:~# cat /proc/mdstat
Personalities : [raid0]
md0 : active raid0 vdb[1] vda[0]
      20953088 blocks super 1.2 512k chunks
```

## 4. Создайте RAID массив первого уровня

1. Создать RAID массив первого уровня:

```
mdadm --create -v /dev/md1 --level=1 --raid-devices=2 /dev/vd[cd]
# mdadm: Note: this array has metadata at the start and
#   may not be suitable as a boot device.  If you plan to
#   store '/boot' on this device please ensure that
#   your boot-loader understands md/v1.x metadata, or use
#   --metadata=0.90
# mdadm: size set to 10476544K
# Continue creating array? yes
# mdadm: Defaulting to version 1.2 metadata
# mdadm: array /dev/md1 started.
```

2. Выполните снова `lsblk`

```
vdc      254:32    0    10G  0 disk
└─md1     9:1      0    10G  0 raid1
vdd      254:48    0    10G  0 disk
└─md1     9:1      0    10G  0 raid1
```

3. Выполните просмотр статуса работы RAID массива: `cat /proc/mdstat`

```
root@SRV:~# cat /proc/mdstat
Personalities : [raid0] [raid1]
md1 : active raid1 vdd[1] vdc[0]
      10476544 blocks super 1.2 [2/2] [UU]
      [=====>.....]  resync = 31.2% (3274240/10476544) finish=1.6min
      speed=73736K/sec

# md0 : active raid0 vdb[1] vda[0]
#       20953088 blocks super 1.2 512k chunks
#
# unused devices: <none>
```

4. В данном случае идет процесс синхронизации диска

## 6. Форматирование дисков

После создания дисковых массивов, для того что б их использовать необходимо их разметить. Для этого мы используем утилиту: `mkfs`, которая позволяет производить форматирование в различные файловые системы.

1. Выполните команды:

```
mkfs.ext4 /dev/md0
mkfs.ext4 /dev/md1
# mke2fs 1.47.0 (5-Feb-2023)
# Discarding device blocks: done
# Creating filesystem with 2619136 4k blocks and 655360 inodes
# Filesystem UUID: 296b1331-d69c-408c-815c-0be1f7ca07d7
# Superblock backups stored on blocks:
#       32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632
#
# Allocating group tables: done
# Writing inode tables: done
# Creating journal (16384 blocks): done
# Writing superblocks and filesystem accounting information: don
```

2. После форматирования дисков каждый получает свой уникальный идентификатор: **UUID**. Для просмотра которого выполните команду: `blkid`:

```
/dev/md0: UUID="8bf01415-1b07-4465-88ba-8a83d476f56b" BLOCK_SIZE="4096"
TYPE="ext4"
/dev/md1: UUID="296b1331-d69c-408c-815c-0be1f7ca07d7" BLOCK_SIZE="4096"
TYPE="ext4"
```

```
root@SRV:~# blkid | grep md1
[ 575.530530] I/O error, dev fd0, sector 0 op 0x0:(READ) flags 0x0 phys_seg 1 prio class 2
/dev/md127: UUID="ef533862-55e2-443b-af51-481786918588" BLOCK_SIZE="4096" TYPE="ext4"
/dev/md126: UUID="fa568ae8-dcc7-4a97-9248-81b7c1cd5259" BLOCK_SIZE="4096" TYPE="ext4"
root@SRV:~#
```

## 7. Временное монтирование дисков

Для использования созданных нами дисковых массивов необходимо подключить их к файловой системе. Этот процесс именуется **монтированием**. Для его выполнения используется команда **mount**, а для действия в обратном направлении, команда **umount**.

Выполненные в ручном режиме, данные команды выполняют свою задачу только в рамках текущего запуска системы, разделы и диски примонтированные автоматически, после перезагрузки будут снова примонтированы. А те что не были примонтированы ранее, так же вернуться к этому состоянию.

1. Создайте каталоги в которые будут подключены наши дисковые массивы:

```
cd /mnt  
mkdir raid1 raid2
```

2. Выполните монтирование массивов в каталоги:

```
mount /dev/md0 /mnt/raid1  
mount /dev/md1 /mnt/raid2
```

3. Выполните команду: `lsblk`

```
vda      254:0      0      10G  0 disk  
└─md0     9:0      0      20G  0 raid0 /mnt/raid1  
vdb      254:16    0      10G  0 disk  
└─md0     9:0      0      20G  0 raid0 /mnt/raid1  
vdc      254:32    0      10G  0 disk  
└─md1     9:1      0      10G  0 raid1 /mnt/raid2  
vdd      254:48    0      10G  0 disk  
└─md1     9:1      0      10G  0 raid1 /mnt/raid2
```

4. Как видно. Наши массивы были примонтированы в каталоги raid1 и raid2

5. Выполните команду: `mount`

```
/dev/md0 on /mnt/raid1 type ext4 (rw,relatime,stripe=256)  
/dev/md1 on /mnt/raid2 type ext4 (rw,relatime)
```

6. Тут отображается что

1. массивы примонтированы
2. имеют формат ext4
3. диск позволяет выполнять операции чтения (r) и записи (w)

7. Пришло время все вернуть на круги своя

8. Выполните команды:

```
umount /mnt/raid1
umount /mnt/raid2
```

9. Посмотрим статус монтирования: `lsblk`

```
vda      254:0      0      10G  0 disk
└─md0     9:0      0      20G  0 raid0
vdb      254:16     0      10G  0 disk
└─md0     9:0      0      20G  0 raid0
vdc      254:32     0      10G  0 disk
└─md1     9:1      0      10G  0 raid1
vdd      254:48     0      10G  0 disk
└─md1     9:1      0      10G  0 raid1
```

10. Видим что диски отмонтировались

## 8. Перманентное монтирование по имени диска

[https://wiki.archlinux.org/title/Fstab\\_\(Русский\)](https://wiki.archlinux.org/title/Fstab_(Русский))

Для выполнения постоянного монтирования в linux действует **fstab**. В этот конфигурационный файл записываются инструкции по монтированию, которые должны быть выполнены при старте системы.

1. Добавим данные о монтировании дискового массива по его имени:

1. Откройте файл `/etc/fstab`
2. И добавьте туда следующую запись:

```
/dev/md0      /mnt/raid1    ext4    defaults    0          0
```

2. Она означает следующее:

1. RAID md0 будет примонтирован
2. в каталог **/mnt/raid1**
3. будет ожидаться файловая система **ext4**
4. параметры должны быть использованы по умолчанию
5. первый ноль нужен команде **dump**, которая узнает по нему нужно ли делать резервную копию данных. (У нас её нет, значит отключаем)
6. второй ноль использует **fsck**, которая проверяет целостность файловой системы. 1 - ставится для корневой ФС, 2 - для другой ФС которую вы хотите проверять

## 8. Перманентное монтирование по UUID

1. Второй диск подключим с использованием UUID:

1. Сделайте копию файла fstab: `cp /etc/fstab ~/`

1. Если вы его вдруг сломаете, то сможете потом восстановить

2. Выполните команду:

```
blkid | grep md1 >> /etc/fstab
```

2. Команда выполнит следующее:

1. запустит утилиту **blkid**

2. отфильтрует её вывод, оставив только строки содержащие **md1**

3. перенаправит вывод на **дозапись** в файл `/etc/fstab`

4. ВНИМАНИЕ!!! Если вы поставите не две >> а одну >, то вы ПЕРЕЗАПИШИТЕ содержимое fstab.

3. Теперь приведем переданную в fstab строку в следующий вид:

```
UUID=296b1331-d69c-408c-815c-0be1f7ca07d7 /mnt/raid2    ext4    defaults
0    0
```

```
#md1
UUID="fa568ae8-dcc7-4a97-9248-81b7c1cd5259" /mnt/raid2  ext4    defaults    0    0
#md0
UUID="ef533862-55e2-443b-af51-481786918588" /mnt/raid1  ext4    defaults    0    0
```

4. Вы великолепны! Осталось чуть-чуть.

5. Выполните команду: `mount -a`

6. Выполните команду: `lsblk`

1. Изучите вывод. Выполнилось ли мониторинг?

```
root@SRV:~# lsblk
NAME        MAJ:MIN RM   SIZE RO TYPE  MOUNTPOINTS
fd0          2:0    1     4K  0 disk
sda          8:0    0    10G  0 disk
├─sda1       8:1    0     9G  0 part  /
├─sda2       8:2    0      1K  0 part
└─sda5       8:5    0   975M  0 part  [SWAP]
sdb          8:16   0   93.9M  0 disk
sr0         11:0    1  414.1M  0 rom   /mnt/repo
vda         254:0    0    10G  0 disk
├─md127      9:127   0    20G  0 raid0 /mnt/raid1
├─vdb        254:16   0    10G  0 disk
├─md127      9:127   0    20G  0 raid0 /mnt/raid1
├─vdc        254:32   0    10G  0 disk
├─md126      9:126   0    10G  0 raid1 /mnt/raid2
├─vdd        254:48   0    10G  0 disk
└─md126      9:126   0    10G  0 raid1 /mnt/raid2
```