1. Узнайте о [sparse](https://ru.wikipedia.org/wiki/%D0%A0%D0%B0%D0%B7%D1%80%D0%B5%D0%B6%D1%91%D0%BD%D0%BD%D1%8B%D0%B9_%D1%84%D0%B0%D0%B9%D0%BB) (разряженных) файлах.

Последовательность нулей в файлах заменяется на информацию об этих нулях.

Частая запись в файлы приводит к фрагментации.

1. Могут ли файлы, являющиеся жесткой ссылкой на один объект, иметь разные права доступа и владельца? Почему?

В Linux каждый файл имеет уникальный идентификатор - индексный дескриптор (inode). Это число, которое однозначно идентифицирует файл в файловой системе. Жесткая ссылка и файл, для которой она создавалась имеют одинаковые inode. Поэтому жесткая ссылка имеет те же права доступа, владельца и время последней модификации, что и целевой файл. Различаются только имена файлов. Фактически жесткая ссылка это еще одно имя для файла.

1. Сделайте vagrant destroy на имеющийся инстанс Ubuntu. Замените содержимое Vagrantfile следующим:

Vagrant.configure("2") do |config|

config.vm.box = "bento/ubuntu-20.04"

config.vm.provider :virtualbox do |vb|

lvm\_experiments\_disk0\_path = "/tmp/lvm\_experiments\_disk0.vmdk"

lvm\_experiments\_disk1\_path = "/tmp/lvm\_experiments\_disk1.vmdk"

vb.customize ['createmedium', '--filename', lvm\_experiments\_disk0\_path, '--size', 2560]

vb.customize ['createmedium', '--filename', lvm\_experiments\_disk1\_path, '--size', 2560]

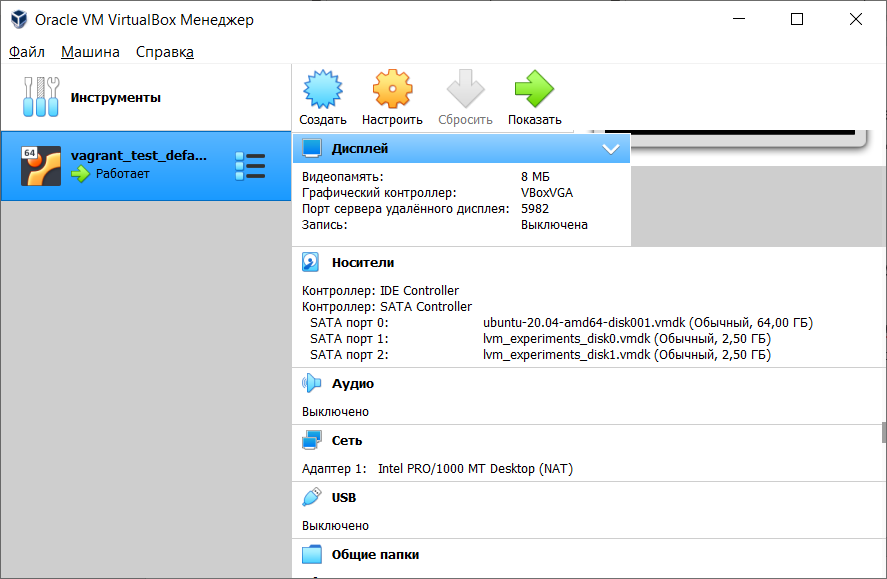
vb.customize ['storageattach', :id, '--storagectl', 'SATA Controller', '--port', 1, '--device', 0, '--type', 'hdd', '--medium', lvm\_experiments\_disk0\_path]

vb.customize ['storageattach', :id, '--storagectl', 'SATA Controller', '--port', 2, '--device', 0, '--type', 'hdd', '--medium', lvm\_experiments\_disk1\_path]

end

end

Данная конфигурация создаст новую виртуальную машину с двумя дополнительными неразмеченными дисками по 2.5 Гб.



1. Используя fdisk, разбейте первый диск на 2 раздела: 2 Гб, оставшееся пространство.

vagrant@vagrant:~$ sudo fdisk /dev/sdb

Welcome to fdisk (util-linux 2.34).

Changes will remain in memory only, until you decide to write them.

Be careful before using the write command.

Device does not contain a recognized partition table.

Created a new DOS disklabel with disk identifier 0x60344eb9.

Command (m for help): m

Help:

DOS (MBR)

a toggle a bootable flag

b edit nested BSD disklabel

c toggle the dos compatibility flag

Generic

d delete a partition

F list free unpartitioned space

l list known partition types

n add a new partition

p print the partition table

t change a partition type

v verify the partition table

i print information about a partition

Misc

m print this menu

u change display/entry units

x extra functionality (experts only)

Script

I load disk layout from sfdisk script file

O dump disk layout to sfdisk script file

Save & Exit

w write table to disk and exit

q quit without saving changes

Create a new label

g create a new empty GPT partition table

G create a new empty SGI (IRIX) partition table

o create a new empty DOS partition table

s create a new empty Sun partition table

Command (m for help): p

Disk /dev/sdb: 2.51 GiB, 2684354560 bytes, 5242880 sectors

Disk model: VBOX HARDDISK

Units: sectors of 1 \* 512 = 512 bytes

Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes

Disklabel type: dos

Disk identifier: 0x60344eb9

Command (m for help): n

Partition type

p primary (0 primary, 0 extended, 4 free)

e extended (container for logical partitions)

Select (default p): p

Partition number (1-4, default 1): 1

First sector (2048-5242879, default 2048):

Last sector, +/-sectors or +/-size{K,M,G,T,P} (2048-5242879, default 5242879): 2G

Value out of range.

Last sector, +/-sectors or +/-size{K,M,G,T,P} (2048-5242879, default 5242879): +2G

Created a new partition 1 of type 'Linux' and of size 2 GiB.

Command (m for help): n

Partition type

p primary (1 primary, 0 extended, 3 free)

e extended (container for logical partitions)

Select (default p): p

Value out of range.

p primary (1 primary, 0 extended, 3 free)

e extended (container for logical partitions)

Select (default p): p

Partition number (2-4, default 2): 2

First sector (4196352-5242879, default 4196352):

Last sector, +/-sectors or +/-size{K,M,G,T,P} (4196352-5242879, default 5242879):

Created a new partition 2 of type 'Linux' and of size 511 MiB.

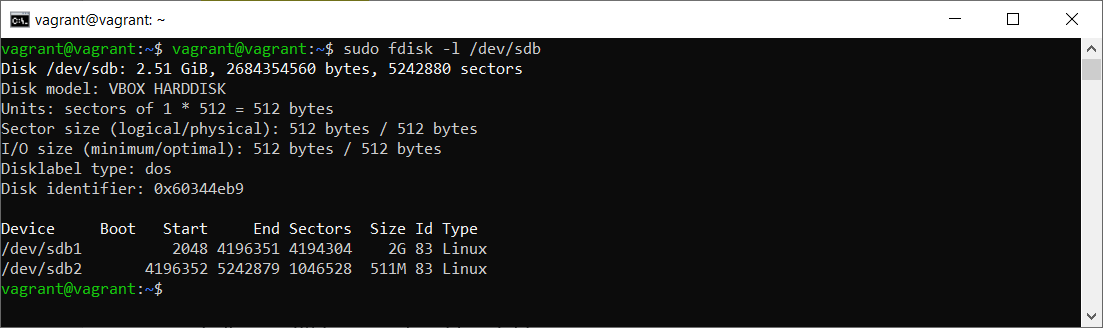
Command (m for help): w

The partition table has been altered.

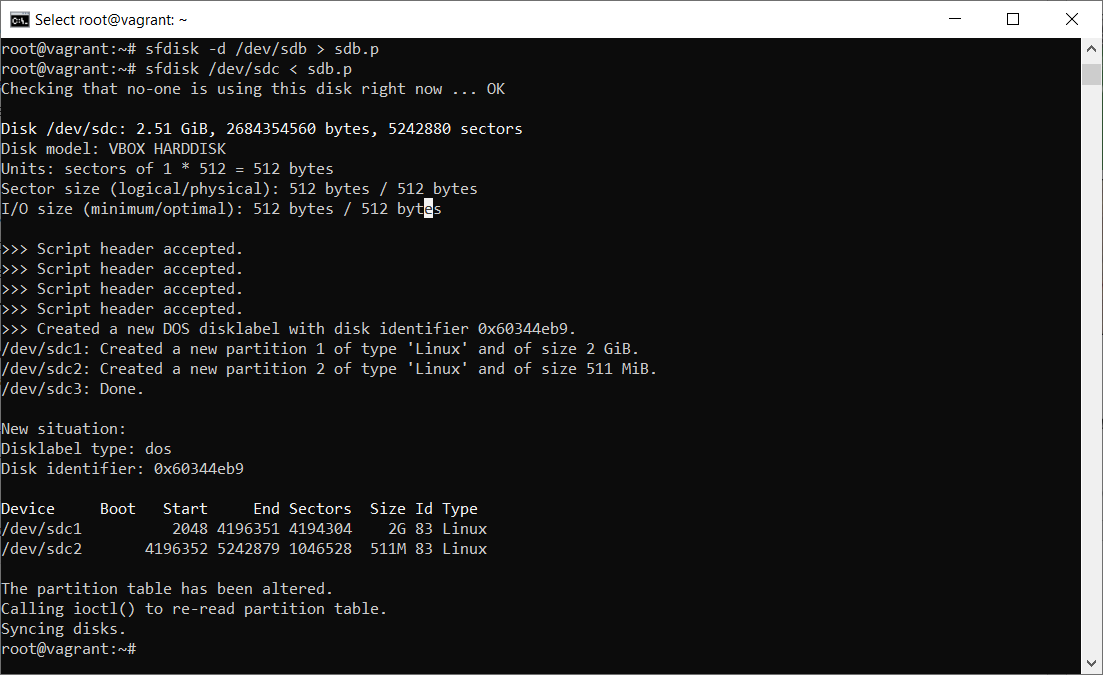
Calling ioctl() to re-read partition table.

Syncing disks.

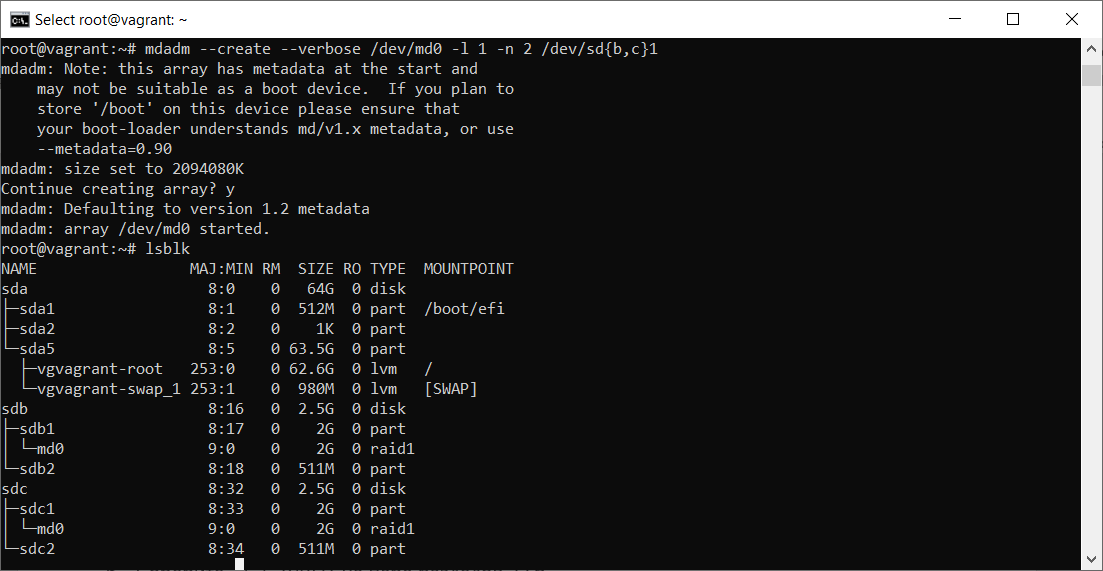
На удивление понятный и интуитивный интерактив )))



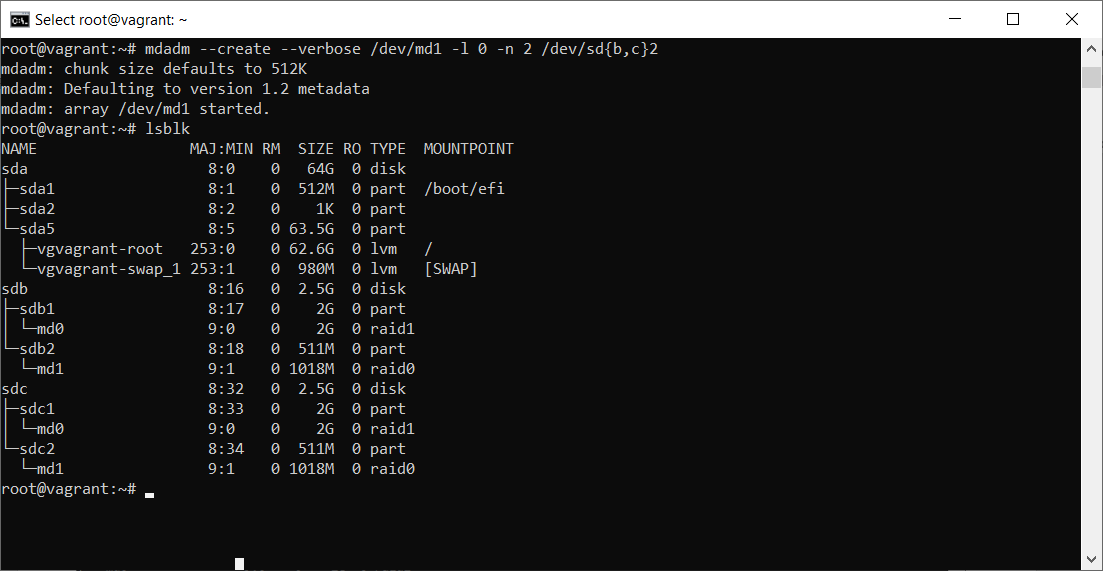
1. Используя sfdisk, перенесите данную таблицу разделов на второй диск.



1. Соберите mdadm RAID1 на паре разделов 2 Гб.



1. Соберите mdadm RAID0 на второй паре маленьких разделов.



1. Создайте 2 независимых PV на получившихся md-устройствах.

root@vagrant:~# pvcreate /dev/md1 /dev/md0

Physical volume "/dev/md1" successfully created.

Physical volume "/dev/md0" successfully created.

1. Создайте общую volume-group на этих двух PV.

root@vagrant:~# vgcreate test /dev/md0 /dev/md1

Volume group "test" successfully created

1. Создайте LV размером 100 Мб, указав его расположение на PV с RAID0.

root@vagrant:~# lvcreate -L 100M test /dev/md1

Logical volume "lvol0" created.

1. Создайте mkfs.ext4 ФС на получившемся LV.

root@vagrant:~# mkfs.ext4 /dev/test/lvol0

mke2fs 1.45.5 (07-Jan-2020)

Creating filesystem with 25600 4k blocks and 25600 inodes

Allocating group tables: done

Writing inode tables: done

Creating journal (1024 blocks): done

Writing superblocks and filesystem accounting information: done

1. Смонтируйте этот раздел в любую директорию, например, /tmp/new.

root@vagrant:~# mkdir /tmp/new

root@vagrant:~# mount /dev/test/lvol0 /tmp/new

root@vagrant:~# df -h

Filesystem Size Used Avail Use% Mounted on

udev 448M 0 448M 0% /dev

tmpfs 99M 712K 98M 1% /run

/dev/mapper/vgvagrant-root 62G 1.5G 57G 3% /

tmpfs 491M 0 491M 0% /dev/shm

tmpfs 5.0M 0 5.0M 0% /run/lock

tmpfs 491M 0 491M 0% /sys/fs/cgroup

/dev/sda1 511M 4.0K 511M 1% /boot/efi

/vagrant 238G 168G 71G 71% /vagrant

tmpfs 99M 0 99M 0% /run/user/1000

/dev/mapper/test-lvol0 93M 72K 86M 1% /tmp/new

1. Поместите туда тестовый файл, например wget [https://mirror.yandex.ru/ubuntu/ls-lR.gz -O /tmp/new/test.gz](https://mirror.yandex.ru/ubuntu/ls-lR.gz%20-O%20/tmp/new/test.gz).

root@vagrant:~# wget https://mirror.yandex.ru/ubuntu/ls-lR.gz -O /tmp/new/test.gz

--2021-04-21 05:47:27-- https://mirror.yandex.ru/ubuntu/ls-lR.gz

Resolving mirror.yandex.ru (mirror.yandex.ru)... 213.180.204.183, 2a02:6b8::183

Connecting to mirror.yandex.ru (mirror.yandex.ru)|213.180.204.183|:443... connected.

HTTP request sent, awaiting response... 200 OK

Length: 20258789 (19M) [application/octet-stream]

Saving to: ‘/tmp/new/test.gz’

/tmp/new/test.gz 100%[===========================>] 19.32M 9.89MB/s in 2.0s

2021-04-21 05:47:29 (9.89 MB/s) - ‘/tmp/new/test.gz’ saved [20258789/20258789]

1. Прикрепите вывод lsblk.

root@vagrant:~# lsblk

NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT

sda 8:0 0 64G 0 disk

├─sda1 8:1 0 512M 0 part /boot/efi

├─sda2 8:2 0 1K 0 part

└─sda5 8:5 0 63.5G 0 part

├─vgvagrant-root 253:0 0 62.6G 0 lvm /

└─vgvagrant-swap\_1 253:1 0 980M 0 lvm [SWAP]

sdb 8:16 0 2.5G 0 disk

├─sdb1 8:17 0 2G 0 part

│ └─md0 9:0 0 2G 0 raid1

└─sdb2 8:18 0 511M 0 part

└─md1 9:1 0 1018M 0 raid0

└─test-lvol0 253:2 0 100M 0 lvm /tmp/new

sdc 8:32 0 2.5G 0 disk

├─sdc1 8:33 0 2G 0 part

│ └─md0 9:0 0 2G 0 raid1

└─sdc2 8:34 0 511M 0 part

└─md1 9:1 0 1018M 0 raid0

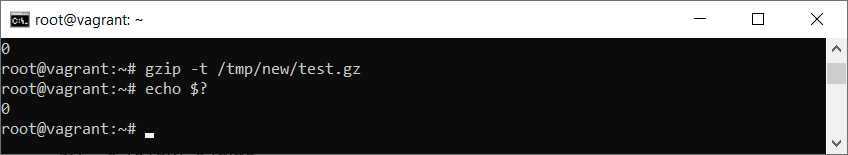
└─test-lvol0 253:2 0 100M 0 lvm /tmp/new

1. Протестируйте целостность файла:

root@vagrant:~# gzip -t /tmp/new/test.gz

root@vagrant:~# echo $?

0



1. Используя pvmove, переместите содержимое PV с RAID0 на RAID1.

root@vagrant:~# pvmove /dev/md1 /dev/md0

/dev/md1: Moved: 36.00%

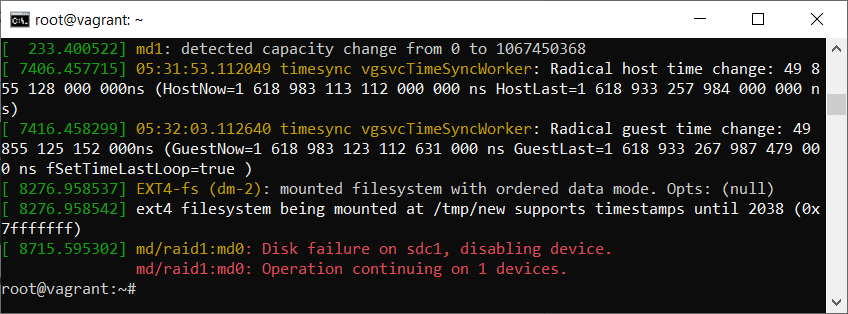
/dev/md1: Moved: 100.00%

1. Сделайте --fail на устройство в вашем RAID1 md.

root@vagrant:~# mdadm /dev/md0 --fail /dev/sdc1

mdadm: set /dev/sdc1 faulty in /dev/md0

1. Подтвердите выводом dmesg, что RAID1 работает в деградированном состоянии.

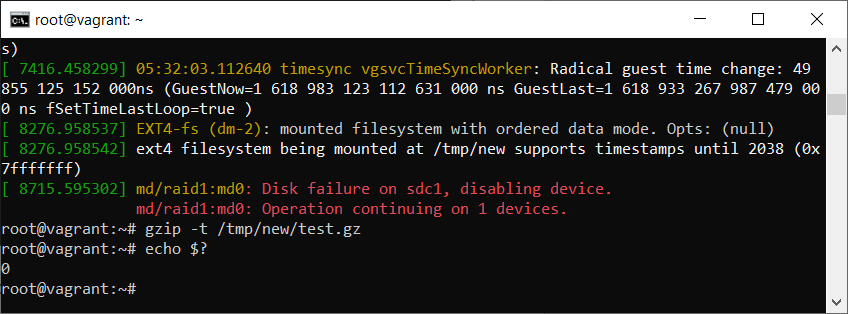


1. Протестируйте целостность файла, несмотря на "сбойный" диск он должен продолжать быть доступен:

root@vagrant:~# gzip -t /tmp/new/test.gz

root@vagrant:~# echo $?

0



1. Погасите тестовый хост, vagrant destroy.