

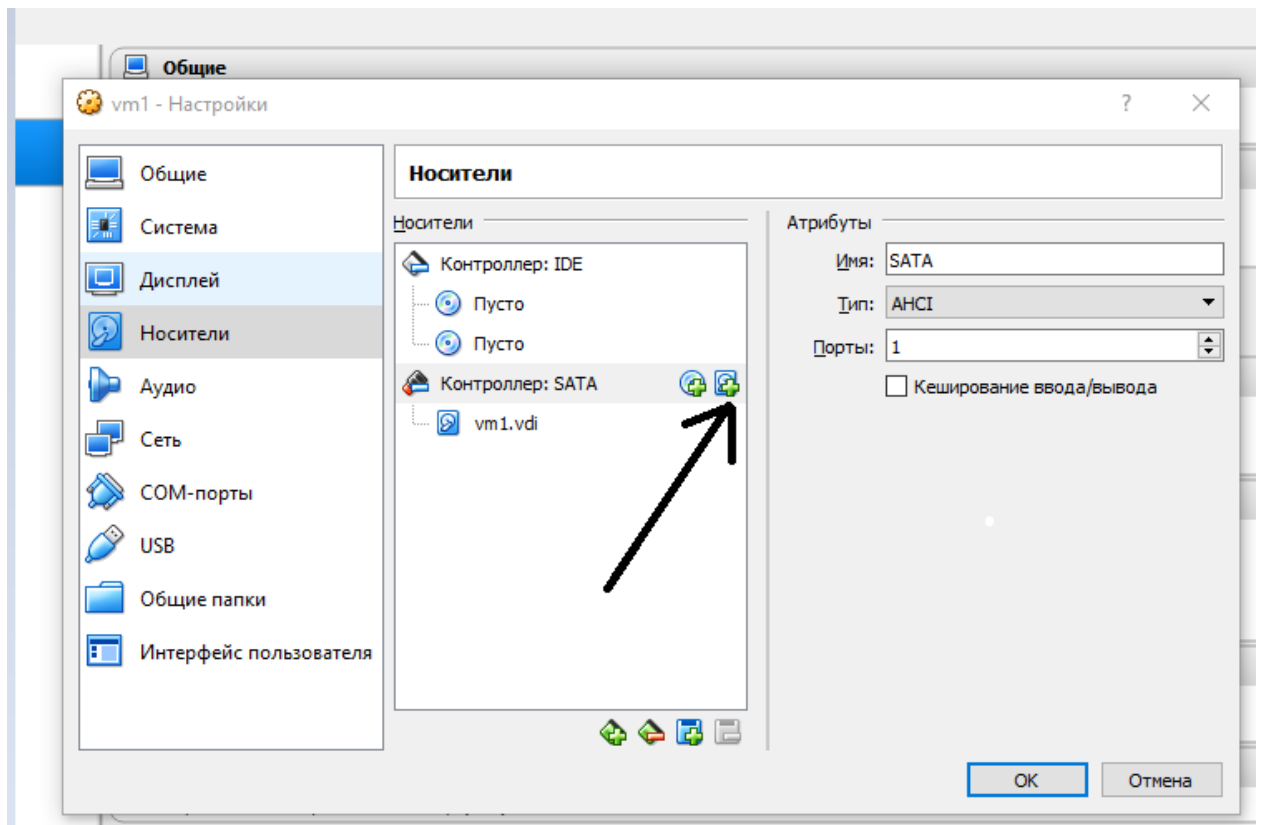
Лабораторная работа №2  
«Файловые системы и файлы»

Выполнил:

Юсупов М.Р. гр. 11-002

## Файловые системы.

Заходим в Virtual Box, выбираем нашу машину и нажимаем «Настройки». Выбираем раздел «Носители» и кликаем на контроллер SATA и добавляем виртуальный жесткий диск (скрин 1.)



Выбираем стандартные настройки, соглашаясь на 10 ГБ, тип VDI и Динамический тип.

Теперь запускаем машину и для просмотра всех доступных дисков используем команду `sudo fdisk -l`

```
Device      Start       End   Sectors  Size Type
/dev/sda1   2048        4095    2048     1M BIOS boot
/dev/sda2   4096 20969471 20965376 10G Linux filesystem

Disk /dev/sdb: 10 GiB, 10737418240 bytes, 20971520 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
```

Нам будет интересен **/dev/sdb**, мы будем с ним работать. Теперь зайдем на выбранный диск для того, чтобы создать таблицу разделов GPT: `sudo fdisk /dev/sdb`. Теперь выбираем команду **g** для создания GPT

```
Command (m for help): g
Created a new GPT disklabel (GUID: 3020DEB1-F155-B24C-9A27-B756D30F3276).
```

Далее создадим разделы, вызвав команду **n**, согласившись с настройками по умолчанию, кроме последнего пункта, где нам нужно отступить 4G

```
Command (m for help): n
Partition number (1-128, default 1):
First sector (2048-20971486, default 2048):
Last sector, +/-sectors or +/-size{K,M,G,T,P} (2048-20971486, default 20971486): +4G
Created a new partition 1 of type 'Linux filesystem' and of size 4 GiB.
```

Таким же образом создаем второй раздел, но соглашаемся со всеми настройками по умолчанию.

```
Command (m for help): n
Partition number (2-128, default 2):
First sector (8390656-20971486, default 8390656):
Last sector, +/-sectors or +/-size{K,M,G,T,P} (8390656-20971486, default 20971486):
Created a new partition 2 of type 'Linux filesystem' and of size 6 GiB.
```

Удостоверимся в правильном формировании разделов

```
disk /dev/sdb: 10 GiB, 10737418240 bytes, 20971520 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: gpt
disk identifier: 3020DEB1-F155-B24C-9A27-B756D30F3276

   device      Start      End  Sectors  Size  Type
/dev/sdb1      2048    8390655  8388608    4G  Linux filesystem
/dev/sdb2    8390656 20971486 12580831    6G  Linux filesystem
osp@vm01:~$ _
```

**Форматирование разделов.**

Для первого раздела форматируем командой `mkfs.ext4 /dev/sdb1`

```
osp@vm01:~$ sudo mkfs.ext4 /dev/sdb1
mke2fs 1.46.5 (30-Dec-2021)
Creating filesystem with 1048576 4k blocks and 262144 inodes
Filesystem UUID: 23912a1b-ebad-4d5a-a292-ee4285a052ec
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376, 294912, 819200, 884736

Allocating group tables: done
Writing inode tables: done
Creating journal (16384 blocks): done
Writing superblocks and filesystem accounting information: done
```

Для второго раздела форматируем командой `mkfs.ext4 /dev/sdb2`

```
osp@vm01:~$ sudo mkfs.ext2 /dev/sdb2
mke2fs 1.46.5 (30-Dec-2021)
Creating filesystem with 1572608 4k blocks and 393216 inodes
Filesystem UUID: d6bf9ecc-8df9-4adc-8825-80cb4876c495
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376, 294912, 819200, 884736

Allocating group tables: done
Writing inode tables: done
Writing superblocks and filesystem accounting information: done
```

Теперь зарезервируем для root пространство

`tune2fs -m 5 /dev/sdb1`

```
osp@vm01:~$ sudo tune2fs -m 5 /dev/sdb1
tune2fs 1.46.5 (30-Dec-2021)
Setting reserved blocks percentage to 5% (52428 blocks)
osp@vm01:~$
```

На втором разделе вообще не должно быть такого зарезервированного пространства:

`tune2fs -m 0 /dev/sdb2`

```
osp@vm01:~$ sudo tune2fs -m 0 /dev/sdb2
tune2fs 1.46.5 (30-Dec-2021)
Setting reserved blocks percentage to 0% (0 blocks)
osp@vm01:~$
```

## Монтирование

Для начало создадим директории для монтирования:

`mkdir /media`

```
mkdir /media/docs
```

```
mkdir /mnt
```

```
mkdir /mnt/work
```

Теперь монтируем:

```
osp@vm01:~$ sudo mount /dev/sdb1 /media/docs
osp@vm01:~$ sudo mount /dev/sdb2 /mnt/work
osp@vm01:~$
```

Теперь нам нужно, чтобы монтирование происходило автоматически при запуске системы, для этого отредактируем файл `/etc/fstab`:

```
nano /etc/fstab
```

```
GNU nano 6.2 /etc/fstab
# /etc/fstab: static file system information.
#
# Use 'blkid' to print the universally unique identifier for a
# device; this may be used with UUID= as a more robust way to name devices
# that works even if disks are added and removed. See fstab(5).
#
# <file system> <mount point> <type> <options> <dump> <pass>
# / was on /dev/sda2 during curtin installation
/dev/disk/by-uuid/b45878e3-0551-4c47-b816-10c203c72ec4 / ext4 defaults 0 1
/swap.img none swap sw 0 0
/dev/sdb1 /media/docs ext4 defaults 0 0
/dev/sdb2 /mnt/work ext2 defaults 0 0
```

Здесь мы настраиваем устройство, путь монтирования и файловую систему.

## Пользователи и группы.

Создадим необходимые группы пользователей `developers`, `managers` и `writers`

```
osp@vm01:~$ sudo groupadd developers
osp@vm01:~$ sudo groupadd managers
osp@vm01:~$ sudo groupadd writers
osp@vm01:~$
```

Далее создадим необходимых пользователей:

woody и buzz

```
osp@vm01:~$ sudo useradd woody
osp@vm01:~$ sudo useradd buzz
```

Добавим пользователей в группу к developers:

```
osp@vm01:~$ sudo usermod -a -G developers woody
osp@vm01:~$ sudo usermod -a -G developers buzz
```

\*Флаги означают, что мы добавляем пользователей в группы.

Далее такие же махинации проводим с managers и writers:

```
osp@vm01:~$ sudo useradd potato
osp@vm01:~$ sudo useradd slinky
osp@vm01:~$ sudo usermod -a -G managers potato
osp@vm01:~$ sudo usermod -a -G managers slinky
osp@vm01:~$ sudo useradd rex
osp@vm01:~$ sudo useradd sid
osp@vm01:~$ sudo usermod -a -G writers rex
osp@vm01:~$ sudo usermod -a -G writers sid
```

## Директории и файлы.

Создаем директорию manuals и устанавливаем владельцем пользователя rex

```
osp@vm01:~$ sudo mkdir /media/docs/manuals
osp@vm01:~$ sudo chown rex /media/docs/manuals
```

Теперь установим группу-владельца writers:

```
osp@vm01:~$ sudo chgrp writers /media/docs/manuals
```

И устанавливаем права доступа:

```
osp@vm01:~$ sudo chmod u=rwx,g=rws,o=rx /media/docs/manuals
```

Далее по аналогии остальные директории, приписанные в лабораторной работе:

```
osp@vm01:~$ sudo mkdir /media/docs/reports
osp@vm01:~$ sudo chown potato /media/docs/reports
osp@vm01:~$ sudo chgrp managers /media/docs/reports
osp@vm01:~$ sudo chmod u=rwx,g=rws,o= /media/docs/reports
osp@vm01:~$ sudo mkdir /media/docs/todo
osp@vm01:~$ sudo chown woody /media/docs/todo
osp@vm01:~$ sudo chgrp developers /media/docs/todo
osp@vm01:~$ sudo chmod u=rwx,g=rx,o=rx /media/docs/todo
```

Проделаем такие же команды для директории /mnt/work:

```
osp@vm01:~$ sudo mkdir /mnt/work/writers
osp@vm01:~$ chown rex /mnt/work/writers
chown: changing ownership of '/mnt/work/writers': Operation not permitted
osp@vm01:~$ sudo chown rex /mnt/work/writers
osp@vm01:~$ sudo chgrp writers /mnt/work/writers
osp@vm01:~$ sudo chmod u=rwx,g=rws,o= /mnt/work/writers
osp@vm01:~$ sudo mkdir /mnt/work/managers
osp@vm01:~$ sudo chown potato /mnt/work/managers
osp@vm01:~$ sudo chgrp managers /mnt/work/managers
osp@vm01:~$ sudo chmod u=rwx,g=rws,o= /mnt/work/managers
osp@vm01:~$ sudo mkdir /mnt/work/developers
osp@vm01:~$ sudo chown woody /mnt/work/developers
osp@vm01:~$ sudo chgrp developers /mnt/work/developers
osp@vm01:~$ sudo chmod u=rwx,g=rws,o= /mnt/work/developers
```

Осталось создать ссылки в /mnt/work/developer:

```
osp@vm01:~$ sudo ln -s /media/docs/manuals /mnt/work/developers/docs
osp@vm01:~$ sudo ln -s /media/docs/todo /mnt/work/developers/todo
osp@vm01:~$
```