

# Партиции, файловые системы, монтирование

Лабораторная работа №14

---

Казначеев С.И.

25 ноября 2025

Российский университет дружбы народов, Москва, Россия

## Информация

---

::::::::::: {.columns align=center} :: {.column width="70%"}  
:::

- Казначеев Сергей Ильич
- Студент
- Российский университет дружбы народов
- [1132240693@pfur.ru] :: {.column width="30%"}  
::

## Цель работы

---

Получить навыки создания разделов на диске и файловых систем. Получить навыки монтирования файловых систем.

## Задание

---

1. Добавьте два диска на виртуальной машине (раздел 14.4.1).
2. Продемонстрируйте навыки создания разделов MBR с помощью fdisk (раздел 14.4.2).
3. Продемонстрируйте навыки создания логических разделов с помощью fdisk (раз- дел 14.4.3).
4. Продемонстрируйте навыки создания раздела подкачки с помощью fdisk (раз- дел 14.4.4).
5. Продемонстрируйте навыки создания разделов GPT с помощью gdisk (раздел 14.4.5).
6. Продемонстрируйте навыки форматирования файловой системы XFS (раздел 14.4.6).
7. Продемонстрируйте навыки форматирования файловой системы EXT4 (раздел 14.4.7).
8. Продемонстрируйте навыки ручного монтирования файловых систем (раздел 14.4.8).
9. Продемонстрируйте навыки монтирования файловых систем с помощью /etc/fstab (раздел 14.4.9).
10. Выполните задание для самостоятельной работы (раздел 14.5).

# Выполнение лабораторной работы

Для начала мы добавили к нашей виртуальной машине два диска размером 512МБ

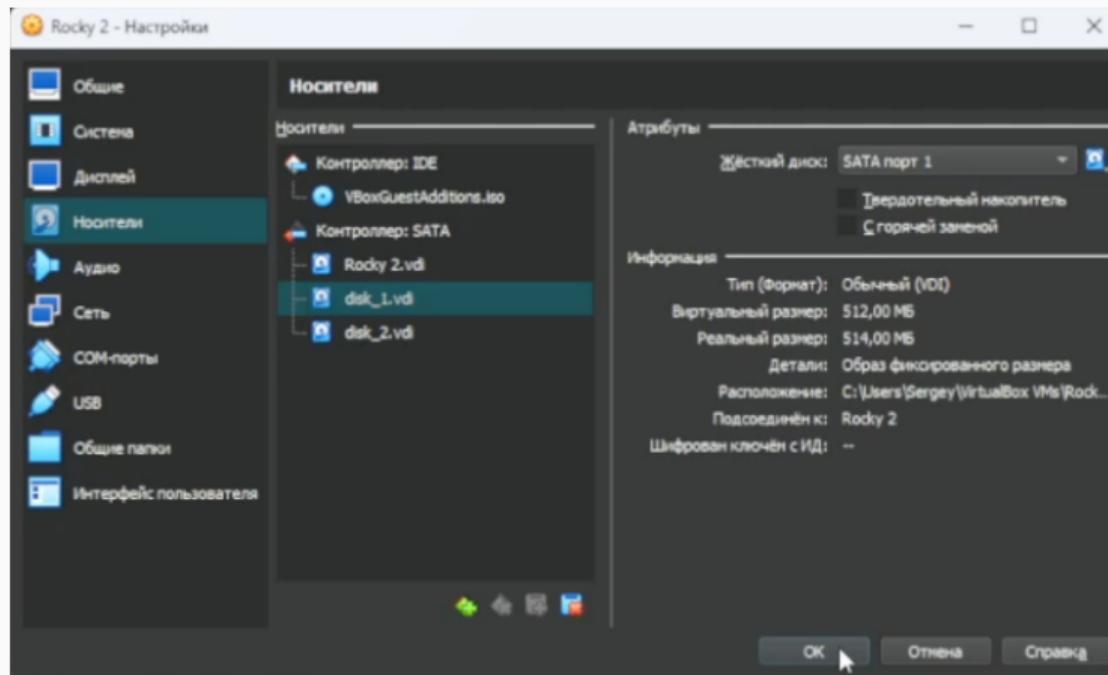


Рис. 1: 1

## Переход в супер пользователя

---

После чего запустили виртуальную машину и перешли в супер пользователя

```
[sikaznacheev@localhost ~]$ su -  
Пароль:
```

Рис. 2: 2

## Просмотр перечень разделов

Затем с помощью fdisk просмотрели перечень разделов на всех имеющихся в системе устройствах жестких дисков

```
[root@localhost ~]# fdisk --list
Диск /dev/sda: 40 GiB, 42949672960 байт, 83886080 секторов
Disk model: VBOX HARDDISK
Единицы: секторов по 1 * 512 = 512 байт
Размер сектора (логический/физический): 512 байт / 512 байт
Размер I/O (минимальный/оптимальный): 512 байт / 512 байт
Тип метки диска: dos
Идентификатор диска: 0x98bd7434

Устр-во   Загрузочный   начало     Конец   Секторы Размер Идентификатор Тип
/dev/sda1    *           2048  2099199  2097152    16          83 Linux
/dev/sda2        2099200 83886079 81786880     396         8e Linux LVM
```

```
Диск /dev/sdb: 512 MiB, 536870912 байт, 1048576 секторов
Disk model: VBOX HARDDISK
Единицы: секторов по 1 * 512 = 512 байт
Размер сектора (логический/физический): 512 байт / 512 байт
Размер I/O (минимальный/оптимальный): 512 байт / 512 байт
```

```
Диск /dev/sdc: 512 MiB, 536870912 байт, 1048576 секторов
Disk model: VBOX HARDDISK
Единицы: секторов по 1 * 512 = 512 байт
Размер сектора (логический/физический): 512 байт / 512 байт
Размер I/O (минимальный/оптимальный): 512 байт / 512 байт
```

```
Диск /dev/mapper/rl-root: 35,05 GiB, 37635489792 байт, 73506816 секторов
Единицы: секторов по 1 * 512 = 512 байт
Размер сектора (логический/физический): 512 байт / 512 байт
```

## Делаем разметку диска

После чего сделаем разметку диска `/dev/sdb` с помощью утилиты `fdisk` и просмотрим справку команд

```
[root@localhost ~]# fdisk /dev/sdb
Добро пожаловать в fdisk (util-linux 2.37.4).
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 0xc2e285a3.
```

Команда (m для справки): m

Справка:

### DOS (MBR)

- a переключение флага загрузки
- b редактирование вложенной метки диска BSD
- c переключение флага dos-совместимости

### Общие

- d удалить раздел
- F показать свободное неразмеченоное пространство
- l список известных типов разделов
- n добавление нового раздела
- p вывести таблицу разделов
- t изменение типа раздела
- v проверка таблицы разделов
- i вывести информацию о разделе

### Разное

- m вывод этого меню
- u изменение единиц измерения экрана/содержимого
- x дополнительная функциональность (только для экспертов)

### Сценарий

- I загрузить разметку из файла сценария sfdisk
- O записать разметку в файл сценария sfdisk

## Просмотр текущего распределения

Далее нажмем p чтобы просмотреть текущее распределение пространства диска, далее введем n , чтобы добавить новый раздел выберем раздел p, чтобы создать основной раздел, после чего изменим раздел используя для изменений t и поставим 83 Linux

```
Команда (m для справки): p
Диск /dev/sdb: 512 MiB, 536870912 байт, 1048576 секторов
Disk model: VBOX HARDDISK
Единицы: секторов по 1 * 512 = 512 байт
Размер сектора (логический/физический): 512 байт / 512 байт
Размер I/O (минимальный/оптимальный): 512 байт / 512 байт
Тип метки диска: dos
Идентификатор диска: 0xc2e285a3

Команда (m для справки): n
Тип раздела
    p  основной (0 primary, 0 extended, 4 free)
    e  расширенный (контейнер для логических разделов)
Выберите (по умолчанию - p):p
Номер раздела (1-4, default 1): 1
Первый сектор (2048-1048575, default 2048):
Last sector, +/-sectors or +/-size{K,M,G,T,P} (2048-1048575, default 1048575): +1

Создан новый раздел 1 с типом 'Linux' и размером 100 MiB.

Команда (m для справки): t
Выбранный раздел 1
Hex code or alias (type L to list all):
```

## Запись изменений

---

После всех проделанных действий запишем изменения с помощью w

Команда (m для справки): w

Таблица разделов была изменена.

Вызывается ioctl() для перечитывания таблицы разделов.

Синхронизируются диски.

Рис. 6: 6

## Сравнение двух команд

Далее сравним вывод команды `fdisk -l /dev/sdb` и `cat /proc/partitions`

```
[root@localhost ~]# fdisk -l /dev/sdb
Диск /dev/sdb: 512 MiB, 536870912 байт, 1048576 секторов
Disk model: VBOX HARDDISK
Единицы: секторов по 1 * 512 = 512 байт
Размер сектора (логический/физический): 512 байт / 512 байт
Размер I/O (минимальный/оптимальный): 512 байт / 512 байт
Тип метки диска: dos
Идентификатор диска: 0xc2e285a3

Устр-во   Загрузочный начало   Конец Секторы Размер Идентификатор Тип
/dev/sdb1                  2048 206847  204800  100M          0 Пустой
[root@localhost ~]# cat /proc/partitions
major minor #blocks  name

      8        0   41943040  sda
      8        1    1048576  sda1
      8        2   40893440  sda2
      8       16     524288  sdb
      8       17    102400  sdb1
      8       32     524288  sdc
     11        0     53704  sr0
    253        0   36753408  dm-0
    253        1   4136960  dm-1
```

Рис. 7: 7

## Запись изменений в таблицу

И запишем изменения в таблицу разделов ядра

```
[root@localhost ~]# partprobe /dev/sdb  
[root@localhost ~]# █
```

Рис. 8: 8

## Создание логических разделов

Затем откроем второй терминал войдем в супер пользователя

```
[sikaznacheev@localhost ~]$ su -  
Пароль:
```

Рис. 9: 9

## Создание логических разделов

И запустим fdisk /dev/sdb, затем введем p,чтобы добавить новый раздел и введя e создадим расширенный раздел,и примем первый сектор и второй нажам на enter

```
[root@localhost ~]# fdisk /dev/sdb

Добро пожаловать в fdisk (util-linux 2.37.4).
Изменения останутся только в памяти до тех пор, пока вы не решите записать их.
Будьте внимательны, используя команду write.

Команда (m для справки): n
Тип раздела
    p  основной (1 primary, 0 extended, 3 free)
    e  расширенный (контейнер для логических разделов)
Выберите (по умолчанию - p):e
Номер раздела (2-4, default 2):
Первый сектор (206848-1048575, default 206848):
Last sector, +/-sectors or +/-size{K,M,G,T,P} (206848-1048575, default 1048575):

Создан новый раздел 2 с типом 'Extended' и размером 411 MiB.

Команда (m для справки): q
```

Рис. 10: 10

## Создание логических разделов

После чего из интерфейса fdisk снова нажмем `n`, а в последнем секторе введем `+101M` после чего запишем изменения на диск

```
[root@localhost ~]# fdisk /dev/sdb

Добро пожаловать в fdisk (util-linux 2.37.4).
Изменения останутся только в памяти до тех пор, пока вы не решите записать их.
Будьте внимательны, используя команду write.

Команда (т для справки): n
Тип раздела
    p  основной (1 primary, 0 extended, 3 free)
    e  расширенный (контейнер для логических разделов)
Выберите (по умолчанию - p):e
Номер раздела (2-4, default 2): 2
Первый сектор (206848-1048575, default 206848):
Last sector, +/-sectors or +/-size{K,M,G,T,P} (206848-1048575, default 1048575):

Создан новый раздел 2 с типом 'Extended' и размером 411 MiB.

Команда (т для справки): n
Все пространство для логических разделов задействовано.
Добавление логического раздела 5
Первый сектор (208896-1048575, default 208896):
Last sector, +/-sectors or +/-size{K,M,G,T,P} (208896-1048575, default 1048575): +101M

Создан новый раздел 5 с типом 'Linux' и размером 101 MiB.

Команда (т для справки): w
```

## Проверка данных

Обновим страницу

```
[root@localhost ~]# partprobe /dev/sdb
```

Рис. 12: 12

## Просмотр информации о добавленных разделах

Затем просмотрим информацию о добавленных разделах

```
[root@localhost ~]# cat /proc/partitions
major minor #blocks name

        8          0   41943040 sda
        8          1   1048576 sda1
        8          2   40893440 sda2
        8         16    524288 sdb
        8         17   102400 sdb1
        8         18        1 sdb2
        8         21   103424 sdb5
        8         32    524288 sdc
       11          0     53704 sr0
      253          0   36753408 dm-0
      253          1   4136960 dm-1

[root@localhost ~]# fdisk --list /dev/sdb
Диск /dev/sdb: 512 MiB, 536870912 байт, 1048576 секторов
Disk model: VBOX HARDDISK
Единицы: секторов по 1 * 512 = 512 байт
Размер сектора (логический/физический): 512 байт / 512 байт
Размер I/O (минимальный/оптимальный): 512 байт / 512 байт
Тип метки диска: dos
Идентификатор диска: 0xc2e285a3

Устройство   Загрузочный начало   Конец Секторы Размер Идентификатор Тип
/dev/sdb1           2048  206847  204800  100M          0 Пустой
/dev/sdb2          206848 1048575  841728  411M          5 Расширенный
/dev/sdb5          208896 415743  206848  101M         83 Linux
```

## Создание раздела подкачки

После всех проделанных действий, снова открываем новый терминал, переходим в супер пользователя и запускаем fdisk /dev/sdb, но теперь на последнем секторе введим +100M и изменим тип раздела на 82, и запишем изменения.

```
[root@localhost ~]# fdisk /dev/sdb
Добро пожаловать в fdisk (util-linux 2.37.4).
Изменения останутся только в памяти до тех пор, пока вы не решите записать их.
Будьте внимательны, используя команду write.

Команда (m для справки): n
Все пространство для логических разделов задействовано.
Добавление логического раздела 6
Первый сектор (417792-1048575, default 417792):
Last sector, +/-sectors or +/-size{K,M,G,T,P} (417792-1048575, default 1048575): +100M

Создан новый раздел 6 с типом 'Linux' и размером 100 MiB.

Команда (m для справки): t
Номер раздела (1,2,5,6, default 6): 6
Hex code or alias (type L to list all): 82

Тип раздела 'Linux' изменен на 'Linux swap / Solaris'.

Команда (m для справки): w
Таблица разделов была изменена.
Вызывается ioctl() для перечитывания таблицы разделов.
Синхронизируются диски.
```

## Обновлением разделов ядра

Обновим таблицу разделов ядра

```
[root@localhost ~]# partprobe /dev/sdb
```

Рис. 15: 15

# Просмотр информации о добавленных разделах

И просмотрим информацию о добавленных разделах

```
[root@localhost ~]# cat /proc/partitions
major minor #blocks name

     8        0   41943040 sda
     8        1   1048576 sda1
     8        2   40893440 sda2
     8       16    524288 sdb
     8       17   102400 sdb1
     8       18        1 sdb2
     8       21   103424 sdb5
     8       22   102400 sdb6
     8       32    524288 sdc
    11        0    53704 sr0
  253        0   36753408 dm-0
  253        1   4136960 dm-1

[root@localhost ~]# fdisk --list /dev/sdb
Диск /dev/sdb: 512 MiB, 536870912 байт, 1048576 секторов
Disk model: VBOX HARDDISK
Единицы: секторов по 1 * 512 = 512 байт
Размер сектора (логический/физический): 512 байт / 512 байт
Размер I/O (минимальный/оптимальный): 512 байт / 512 байт
Тип метки диска: dos
Идентификатор диска: 0xc2e285a3

Устр-во   Загрузочный начало   Конец Секторы Размер Идентификатор Тип
/dev/sdb1            2048  206847  204800   100M          0 Пустой
/dev/sdb2            206848 1048575  841728   411M          5 Расширенный
/dev/sdb5            208896 415743  206848   101M         83 Linux
/dev/sdb6            417792 622591  204800   100M         82 Linux swap / Solaris
[root@localhost ~]# mkswap /dev/sdb6
Setting up swapspace version 1, size = 100 MiB (104853504 bytes)
```

## Отформатируем раздел подкачки

После чего отформатируем раздел подкачки, после чего снова включим и для просмотра размера подкачки введем free -m

```
[root@localhost ~]# mkswap /dev/sdb6
Setting up swapspace version 1, size = 100 MiB (104853504 bytes)
без метки, UUID=349d51ec-f278-4371-a4da-9536d6e80c5b
[root@localhost ~]# swapon /dev/sdb6
[root@localhost ~]# free -m
              total        used         free       shared  buff/cache   available
Mem:          7680        1427        4915          20        1606        6252
Swap:         4139           0        4139
```

Рис. 17: 17

## Создание разделов GPT с помощью gdisk

Затем снова открываем новый терминал и переходим в супер пользователя, затем просматриваем таблицу на втором диске

```
[sikaznacheev@localhost ~]$ su -
Пароль:
[root@localhost ~]# gdisk -l /dev/sdc
GPT fdisk (gdisk) version 1.0.7

Partition table scan:
  MBR: not present
  BSD: not present
  APM: not present
  GPT: not present

Creating new GPT entries in memory.
Disk /dev/sdc: 1048576 sectors, 512.0 MiB
Model: VBOX HARDDISK
Sector size (logical/physical): 512/512 bytes
Disk identifier (GUID): 5EC004AF-7F7E-433D-9A17-CE60DCAD5175
Partition table holds up to 128 entries
Main partition table begins at sector 2 and ends at sector 33
First usable sector is 34, last usable sector is 1048542

```

# Создание разделов GPT с помощью gdisk

После чего введя n, чтобы добавить новый раздел выберем номер по умолчанию, в последнем секторе укажем +100M и установим тип раздела 8300

```
Command (? for help): n
Partition number (1-128, default 1):
First sector (34-1048542, default = 2048) or (+/-)size(KMGTP):
Last sector (2048-1048542, default = 1048542) or (+/-)size(KMGTP): +100M
Current type is 8300 (Linux filesystem)
Hex code or GUID (L to show codes, Enter = 8300): l
Type search string, or <Enter> to show all codes:
0700 Microsoft basic data          0701 Microsoft Storage Replica
0702 ArcOS Type 1                  0c01 Microsoft reserved
2700 Windows RE                   3000 ONIE boot
3001 ONIE config                 3900 Plan 9
4100 PowerPC PReP boot           4200 Windows LDM data
4201 Windows LDM metadata         4202 Windows Storage Spaces
7501 IBM GPFS                      7f00 ChromeOS kernel
7f01 ChromeOS root                7f02 ChromeOS reserved
8200 Linux swap                   8300 Linux filesystem
8301 Linux reserved               8302 Linux /home
8303 Linux x86 root (/)           8304 Linux x86-64 root (/)
8305 Linux ARM64 root (/)          8306 Linux /srv
8307 Linux ARM32 root (/)          8308 Linux dm-crypt
8309 Linux LUKS                     830a Linux IA-64 root (/)
830b Linux x86 root verity        830c Linux x86-64 root verity
830d Linux ARM32 root verity      830e Linux ARM64 root verity
830f Linux IA-64 root verity      8310 Linux /var
8311 Linux /var/tmp                8312 Linux user's home
8313 Linux x86 /usr                8314 Linux x86-64 /usr
8315 Linux ARM32 /usr              8316 Linux ARM64 /usr
8317 Linux IA-64 /usr              8318 Linux x86 /usr verity
Press the <Enter> key to see more codes, q to quit:
8319 Linux x86-64 /usr verity      831a Linux ARM32 /usr verity
831b Linux ARM64 /usr verity       831c Linux IA-64 /usr verity
8400 Intel Rapid Start             8401 SPD block device
8500 Container Linux /usr           8501 Container Linux resizable rootfs
8502 Container Linux /OEM customization 8503 Container Linux root on RAID
8e00 Linux LVM                      a000 Android bootloader
a001 Android bootloader 2            a002 Android boot 1
a003 Android recovery 1             a004 Android misc
a005 Android metadata              a006 Android system 1
a007 Android cache                 a008 Android data
a009 Android persistent            a00a Android factory
a00b Android fastboot/tertiary     a00c Android OEM
a00d Android vendor                a00e Android config
a00f Android factory (alt)         a010 Android meta
a011 Android EXT                  a012 Android SBL1
```

## Просмотр нового раздела

Теперь раздел создан и отобразим введя `r`

```
Command (? for help): p
Disk /dev/sdc: 1048576 sectors, 512.0 MiB
Model: VBOX HARDDISK
Sector size (logical/physical): 512/512 bytes
Disk identifier (GUID): 27D10953-05C4-4704-B4C7-615E97A53405
Partition table holds up to 128 entries
Main partition table begins at sector 2 and ends at sector 33
First usable sector is 34, last usable sector is 1048542
Partitions will be aligned on 2048-sector boundaries
Total free space is 843709 sectors (412.0 MiB)

Number  Start (sector)    End (sector)  Size       Code  Name
      1              2048          206847   100.0 MiB  8300  Linux filesystem

Command (? for help): w

Final checks complete. About to write GPT data. THIS WILL OVERWRITE EXISTING
PARTITIONS!!

Do you want to proceed? (Y/N): y
OK; writing new GUID partition table (GPT) to /dev/sdc.
The operation has completed successfully.
```

## Обновление таблицы

---

После чего обновим таблицу разделов

```
[root@localhost ~]# partprobe /dev/sdc
```

Рис. 21: 21

# Просмотр информации

И просмотрим информацию о добавленных разделах

```
[root@localhost ~]# cat /proc/partitions
major minor #blocks name

      8        0   41943040 sda
      8        1   1048576 sda1
      8        2   40893440 sda2
      8       16    524288 sdb
      8       17   102400 sdb1
      8       18        1 sdb2
      8       21   103424 sdb5
      8       22   102400 sdb6
      8       32    524288 sdc
      8       33   102400 sdc1
     11        0    53704 sr0
  253        0   36753408 dm-0
  253        1   4136960 dm-1
[root@localhost ~]# gdisk -l /dev/sdc
GPT fdisk (gdisk) version 1.0.7

Partition table scan:
  MBR: protective
  BSD: not present
  APM: not present
  GPT: present

Found valid GPT with protective MBR; using GPT.
Disk /dev/sdc: 1048576 sectors, 512.0 MiB
Model: VBOX HARDDISK
Sector size (logical/physical): 512/512 bytes
Disk identifier (GUID): 27D10953-05C4-4704-B4C7-615E97A53405
Partition table holds up to 128 entries
Main partition table begins at sector 2 and ends at sector 33
First usable sector is 34, last usable sector is 1048542
Partitions will be aligned on 2048-sector boundaries
```

# Форматирование файловой системы XFS

После чего откроем новый терминал, затем перейдем в супер пользователя и создадим файловую систему XFS для sdb1, и установим файловой системы в xfsdis, и создадим файловую систему уже для sdb5

```
[sikaznacheev@localhost ~]$ su -
Пароль:
[root@localhost ~]# mkfs.xfs /dev/sdb1
Filesystem should be larger than 300MB.
Log size should be at least 64MB.
Support for filesystems like this one is deprecated and they will not be supported in future releases.
meta-data=/dev/sdb1          isize=512    agcount=4, agsize=6400 blks
                           =         sectsz=512   attr=2, projid32bit=1
                           =         crc=1     finobt=1, sparse=1, rmapbt=0
                           =         reflink=1  bigtime=1 inobtcount=1 nrext64=0
data          =             bsize=4096   blocks=25600, imaxpct=25
                           =             sunit=0    swidth=0 blks
naming        =version 2      bsize=4096   ascii-ci=0, ftype=1
log           =internal log   bsize=4096   blocks=1368, version=2
                           =             sectsz=512   sunit=0 blks, lazy-count=1
realtime      =none          extsz=4096   blocks=0, rtextents=0
[root@localhost ~]# xfs_admin -L xfsdisk /dev/sdb1
writing all SBs
new label = "xfsdisk"
[root@localhost ~]# mkfs.ext4 /dev/sdb5
mke2fs 1.46.5 (30-Dec-2021)
Creating filesystem with 103424 1k blocks and 25896 inodes
Filesystem UUID: a19c5caf-3f50-4f2d-8dac-927ed9637b61
Superblock backups stored on blocks:
          8193, 24577, 40961, 57345, 73729

Allocating group tables: done
Writing inode tables: done
Creating journal (4096 blocks): done
```

## Установка параметров монтирования

Для установки параметров монтирования по умолчанию для файловой системы используем команду:

```
tune2fs -o acl,user_xattr /dev/sdb5
```

```
[root@localhost ~]# tune2fs -o acl,user_xattr /dev/sdb5
tune2fs 1.46.5 (30-Dec-2021)
```

Рис. 24: 24

# Форматирование файловой системы EXT4

После всех проделанных действий открываем новый терминал переходим в супер пользователя и создадим точку для монтироавния для раздела /mnt/tmp, после чего используем команду mount /dev/sdb5 /mnt/tmp, чтобы смонтировать файловую систему и проверим с помощью команды mount

```
root@localhost ~]# mkdir -p /mnt/tmp
root@localhost ~]# mount /dev/sdb5 /mnt/tmp
root@localhost ~]# mount
proc on /proc type proc (rw,nosuid,nodev,noexec,relatime)
sysfs on /sys type sysfs (rw,nosuid,nodev,noexec,relatime,seclabel)
devtmpfs on /dev type devtmpfs (rw,nosuid,nodev,seclabel,stripe=4096k,nr_inodes=974811,mode=755,inode64)
securityfs on /sys/kernel/security type security (rw,nosuid,nodev,noexec,relatime)
devpts on /dev/pts type devpts (rw,nosuid,nodev,seclabel,ptmxmode=000)
tmpfs on /dev/shm type tmpfs (rw,nosuid,nodev,seclabel,inode64)
dvsync on /dev/pv type dvsync (rw,nosuid,nodev,seclabel,gid=5,mode=620,ptmxmode=000)
nfs on /dev/nfs type nfs (rw,nosuid,nodev,seclabel,vers=3,secattr=1572964k,nr_inodes=819200,mode=755,inode64)
squashfs on /sys/fs/cgroup type squashfs (ro,nosuid,nodev,noexec,relatime,seclabel,nsdelegate4,memory_recursiveprot)
pstore on /sys/fs/pstore type pstore (rw,nosuid,nodev,noexec,relatime,seclabel)
apf on /sys/fs/bpf type bpf (rw,nosuid,nodev,noexec,relatime,seclabel)
apf on /sys/fs/bpf type bpffs (rw,nosuid,nodev,noexec,relatime,seclabel,attr2,inode64,logbufs=8,logbsize=32k,noquota)
/dev/mapper/r1-root on / type xfs (rw,relatime,seclabel,attr2,inode64,logbufs=8,logbsize=32k,noquota)
selinuxfs on /sys/fs/selinux type selinuxfs (rw,nosuid,noexec,relatime)
systemd-1 on /proc/sys/fs/binfmt_misc type autofs (rw,relatime,fd=20,pgno=8,minproto=5,maxproto=5,direct,pipe_ino=16788)
cracefs on /sys/kernel/tracing type tracefs (rw,nosuid,nodev,noexec,relatime,seclabel)
hugetlbfs on /dev/hugepages type hugetlbfs (rw,relatime,seclabel,pagesize=2M)
queue on /dev/queue type queue (rw,nosuid,nodev,noexec,relatime,seclabel)
debugfs on /sys/kernel/debug type debugfs (rw,nosuid,nodev,noexec,relatime,seclabel)
fusectl on /sys/fs/fuse/connections type fusectl (rw,nosuid,nodev,noexec,relatime)
configfs on /sys/kernel/config type configfs (rw,nosuid,nodev,noexec,relatime)
none on /run/credentials/systemd-tmpfiles-setup-dev.service type ramfs (ro,nosuid,nodev,noexec,relatime,seclabel,mode=700)
none on /run/credentials/systemd-tmpfiles-setup type ramfs (ro,nosuid,nodev,noexec,relatime,seclabel,mode=700)
none on /run/credentials/systemd-tmpfiles-setup type ramfs (ro,nosuid,nodev,noexec,relatime,seclabel,mode=700)
none on /run/credentials/systemd-tmpfiles-setup.type ramfs (ro,nosuid,nodev,noexec,relatime,seclabel,mode=700)
tmpfs on /run/user/1000 type tmpfs (rw,nosuid,nodev,relatime,seclabel,stripe=786448k,nr_inodes=196611,mode=700)
gvfsd-fuse on /run/user/1000/gvfs type fuse_gvfsd-fuse (rw,nosuid,nodev,relatime,user_id=1000,group_id=1000,gid=1000,inode64)
portals on /run/user/1000/doc type fuse.portal (rw,nosuid,nodev,relatime,user_id=1000,group_id=1000)
/dev/sdb5 on /mnt/tmp type ext4 (rw,relatime,seclabel)
```

Рис. 25: 25

# Форматирование файловой системы EXT4

Для того чтобы отмонтировать раздел используем команду umount /dev/sdb5 и проверим что раздел отмонтирован

```
[root@localhost ~]# umount /dev/sdb5
[root@localhost ~]# mount
proc on /proc type proc (rw,nosuid,nodev,noexec,relatime)
sysfs on /sys type sysfs (rw,nosuid,nodev,noexec,relatime,seclabel)
devtmpfs on /dev type devtmpfs (rw,nosuid,nodev,seclabel,size=4096k,nr_inodes=974811,mode=755,inode64)
securityfs on /sys/kernel/security type securityfs (rw,nosuid,nodev,noexec,relatime)
tmpfs on /dev/shm type tmpfs (rw,nosuid,nodev,seclabel,inode64)
devpts on /dev/pts type devpts (rw,nosuid,noexec,relatime,seclabel,gid=50,mode=020,ptmxmode=080)
tmpfs on /run type tmpfs (rw,nosuid,nodev,seclabel,size=1572964k,nr_inodes=819288,mode=755,inode64)
cgroup2 on /sys/fs/cgroup type cgroup (rw,nosuid,nodev,noexec,relatime,seclabel,nsdelegate,memory_recursiveprot)
ostree on /sys/fs/pstore type pstore (rw,nosuid,nodev,noexec,relatime,seclabel)
cpuf on /sys/fs/bpf type bpf (rw,nosuid,nodev,noexec,relatime,mode=700)
/dev/mapper/r1-root on / type xfs (rw,relatime,seclabel,attr2,inode64,logbufs=8,logbsize=32k,noquota)
selinuxfs on /sys/fs/selinux type selinuxfs (rw,nosuid,nodev,noexec,relatime,seclabel)
sysfs on /run type sysfs (rw,nosuid,nodev,noexec,relatime,seclabel)
tracefs on /sys/kernel/tracing type tracefs (rw,nosuid,nodev,noexec,relatime,seclabel)
hugepages on /dev/hugepages type hugepages (rw,relatime,seclabel,pagesize=2M)
aqueue on /dev/aqueue type aqueue (rw,nosuid,nodev,noexec,relatime,seclabel)
debugfs on /sys/kernel/debug type debugfs (rw,nosuid,nodev,noexec,relatime,seclabel)
fusect on /sys/fs/fuse/connections type fuseactl (rw,nosuid,nodev,noexec,relatime)
configfs on /sys/kernel/config type configfs (rw,nosuid,nodev,noexec,relatime)
none on /run/credentials/systemd-tmpfiles-setup-dev.service type ramfs (ro,nosuid,nodev,noexec,relatime,seclabel,mode=700)
none on /run/credentials/systemd-tmpfiles-setup.service type ramfs (ro,nosuid,nodev,noexec,relatime,seclabel,mode=700)
/dev/dax0 on /boot type xfs (rw,relatime,seclabel,attr2,inode64,logbufs=8,logbsize=32k,noquota)
none on /run/credentials/systemd-tmpfiles-setup.service.type ramfs (ro,nosuid,nodev,noexec,relatime,seclabel,mode=700)
tmpfs on /run/user/1000 type tmpfs (rw,nosuid,nodev,relatime,seclabel,size=786448k,nr_inodes=196612,mode=700,uid=1000,gid=1000,inode64)
gvfsd-fuse on /run/user/1000/gvfs type fuse.gvfsd-fuse (rw,nosuid,nodev,relatime,user_id=1000,group_id=1000)
/dev/sr0 on /run/media/stkaznacheevy/Box_GAS_7.0.26 type iso9660 (ro,nosuid,nodev,relatime,nojoliet,check=s,map=n,blocksize=2048,uid=1000,gid=1000,fmode=4000,uhelper=udisks2)
portal on /run/user/1000/doc type fuse.portal (rw,nosuid,nodev,relatime,user_id=1000,group_id=1000)
```

Рис. 26: 26

## Ручное монтирование файловых систем

После чего откроем новый терминал и перейдем в супер пользователя, создадим точку монтируания для раздела /dev/sdb1 и просмотрим информацию об индикаторах блочных устройств

```
[sikaznacheev@localhost ~]$ su -
Пароль:
[root@localhost ~]# mkdir -p /mnt/data
[root@localhost ~]# blkid
/dev/mapper/rl-swap: UUID="b322b360-62cd-49ca-ac68-7eff1b955eba" TYPE="swap"
/dev/sdb5: UUID="a19c5caf-3f50-4f2d-8dac-927ed9637b61" TYPE="ext4" PARTUUID="c2e285a3-05"
/dev/sdb1: LABEL="xfsdisk" UUID="9d36309a-c546-4919-8083-a22b15c7ff43" TYPE="xfs" PARTUUID="c2e285a3-01"
/dev/sdb6: UUID="349d51ec-f278-4371-a4da-9536d6e80c5b" TYPE="swap" PARTUUID="c2e285a3-06"
/dev/sr0: UUID="2025-04-11-13-30-24-73" LABEL="VBox_GAs_7.0.26" TYPE="iso9660"
/dev/mapper/rl-root: UUID="c5cdc57d-568f-49ad-8b76-e02295642aa5" TYPE="xfs"
/dev/sdc1: PARTLABEL="Linux filesystem" PARTUUID="29cedd07-296f-425f-9bdd-1e0211e9955d"
/dev/sda2: UUID="HdDn94-f0jS-CdEt-gY7n-FKn9-7gRd-ANNGGN" TYPE="LVM2_member" PARTUUID="98bd7434-02"
/dev/sdal: UUID="ca930193-d4b4-45d3-a74c-f8d48aad4d88" TYPE="xfs" PARTUUID="98bd7434-01"
[root@localhost ~]# blkid /dev/sdb1
/dev/sdb1: LABEL="xfsdisk" UUID="9d36309a-c546-4919-8083-a22b15c7ff43" TYPE="xfs" PARTUUID="c2e285a3-01"
```

Рис. 27: 27

После чего введем blkid /dev/sdb1, скопируем UUID для устройства /dev/sdb1, и изменим файл /etc/fstab

```
/dev/mapper/rl-root      /          xfs      defaults
UUID=9d36309a-c546-4919-8083-a22b15c7ff43 /mnt/data
/dev/mapper/rl-swap       none      swap      defaults
```

Рис. 28: 28

# Ручное монтирование файловых систем

Далее монтируем все что указано в /etc/fstab и проверяем это

```
[root@localhost ~]# mount -a
mount: (hint) your fstab has been modified, but systemd still uses
      the old version; use 'systemctl daemon-reload' to reload.
[root@localhost ~]# df -h
Файловая система    Размер Использовано   Дост Использовано% Смонтировано в
/devtmpfs           4,0M      0  4,0M      0% /dev
tmpfs              3,8G      0  3,8G      0% /dev/shm
tmpfs              1,6G     1,3M  1,5G      1% /run
/dev/mapper/rl-root 35G      13G  23G      36% /
/dev/sda1            960M    600M  361M     63% /boot
tmpfs              769M    124K  768M      1% /run/user/1000
/dev/sr0              53M     53M      0  100% /run/media/sikaznacheev/VBox_GAs_7.0.26
/dev/sdb1             95M     6,0M   89M      7% /mnt/data
```

Рис. 29: 29

## Самостоятельная часть 1

Откроем новый терминал, перейдем в супер пользователя, затем запустим gdisk для диска /dev/sdc, после чего создадим первый раздел (для ext4) и второй (для swap), после чего запишем изменения и выйдем

```
[sikaznacheev@localhost ~]$ su -
Пароль:
[root@localhost ~]# gdisk /dev/sdc
GPT fdisk (gdisk) version 1.0.7

Problem opening /dev/sbc for reading! Error is 2.
The specified file does not exist!
[root@localhost ~]# gdisk /dev/sdc
GPT fdisk (gdisk) version 1.0.7

Partition table scan:
  MBR: protective
  BSD: not present
  APM: not present
  GPT: present

Found valid GPT with protective MBR; using GPT.

Command (? for help): n
Partition number (2-128, default 2):
First sector (34-1048542, default = 206848) or {+-}size{KMGTP}:
Last sector (206848-1048542, default = 1048542) or {+-}size{KMGTP}: +100M
Current type is 8300 (Linux filesystem)
Hex code or GUID (L to show codes, Enter = 8300):
Changed type of partition to 'Linux filesystem'

Command (? for help): n
Partition number (3-128, default 3):
First sector (34-1048542, default = 411648) or {+-}size{KMGTP}:
Last sector (411648-1048542, default = 1048542) or {+-}size{KMGTP}: +100M
Current type is 8300 (Linux filesystem)
Hex code or GUID (L to show codes, Enter = 8300): 8200
Changed type of partition to 'Linux swap'

Command (? for help): p
Disk /dev/sdc: 1048576 sectors, 512.0 MiB
Model: VBOX HARDDISK
Sector size (logical/physical): 512/512 bytes
```

## Самостоятельная часть 2

Далее обновим таблицу разделов ядра и проверим созданные разделы

```
[root@localhost ~]# partprobe /dev/sdc
[root@localhost ~]# gdisk -l /dev/sdc
GPT fdisk (gdisk) version 1.0.7

Partition table scan:
  MBR: protective
  BSD: not present
  APM: not present
  GPT: present

Found valid GPT with protective MBR; using GPT.
Disk /dev/sdc: 1048576 sectors, 512.0 MiB
Model: VBOX HARDDISK
Sector size (logical/physical): 512/512 bytes
Disk identifier (GUID): 27D10953-05C4-4704-B4C7-615E97A53405
Partition table holds up to 128 entries
Main partition table begins at sector 2 and ends at sector 33
First usable sector is 34, last usable sector is 1048542
Partitions will be aligned on 2048-sector boundaries
Total free space is 434109 sectors (212.0 MiB)

Number  Start (sector)    End (sector)  Size       Code  Name
      1              2048        206847  100.0 MiB  8300  Linux filesystem
      2          206848        411647  100.0 MiB  8300  Linux filesystem
      3          411648        616447  100.0 MiB  8200  Linux swap
[root@localhost ~]# cat /proc/partitions
major minor #blocks name

     8        0   41943040  sda
     8        1   1048576  sda1
     8        2   40893440  sda2
     8       16    524288  sdb
     8       17   102400  sdb1
     8       18        1  sdb2
     8       21   103424  sdb5
     8       22   102400  sdb6
     8       32    524288  sdc
     8       33   102400  sdc1
```

## Самостоятельная часть 3

После чего создаем точку монтирования для раздела ext4 и форматируем второй раздел как swap, после чего узнаем UUID разделов sdc1 и sdc2

```
[root@localhost ~]# mkfs.ext4 /dev/sdc1
mke2fs 1.46.5 (30-Dec-2021)
Creating filesystem with 102400 1k blocks and 25584 inodes
Filesystem UUID: a3c8d222-5289-4a9f-aa3d-0b2ddff1cbee
Superblock backups stored on blocks:
    8193, 24577, 40961, 57345, 73729

Allocating group tables: done
Writing inode tables: done
Creating journal (4096 blocks): done
Writing superblocks and filesystem accounting information: done

[root@localhost ~]# mkswap /dev/sdc2
Setting up swapspace version 1, size = 100 MiB (104853504 bytes)
без метки, UUID=daddb5a0-8b58-45f3-9e64-539a59367797
[root@localhost ~]# mkdir -p /mnt/data/ex
[root@localhost ~]# blkid /dev/sdc1
/dev/sdc1: UUID="a3c8d222-5289-4a9f-aa3d-0b2ddff1cbee" TYPE="ext4" PARTLABEL="Linux filesystem" PARTUUID="29cedd07-296f-425f-9bdd-1e0211e9955d"
[root@localhost ~]# blkid /dev/sdc2
/dev/sdc2: UUID="daddb5a0-8b58-45f3-9e64-539a59367797" TYPE="swap" PARTLABEL="Linux filesystem" PARTUUID="5cc98d1e-af29-4ff5-a5fc-a79b38656e20"
[root@localhost ~]# nano /etc/fstab
[root@localhost ~]# nano /etc/fstab
```

Рис. 32: 32

## Самостоятельная часть 4

Монтируем и активируем swap раздел для sdc2, проверяем монтирование, проверяем swap

```
[root@localhost ~]# swapon /dev/sdc2
[root@localhost ~]# df -h /mnt/data-ex
Файловая система Размер Использовано Дост Использовано% Смонтировано в
/dev/mapper/rl-root 35G 13G 23G 36% /
[root@localhost ~]# nano /etc/fstab
[root@localhost ~]# mount -a
mount: (hint) your fstab has been modified, but systemd still uses
      the old version; use 'systemctl daemon-reload' to reload.
[root@localhost ~]# df -h /mnt/data-ex
Файловая система Размер Использовано Дост Использовано% Смонтировано в
/dev/sdc1 89M 14K 82M 1% /mnt/data-ex
[root@localhost ~]# swapon /dev/sdc2
swapon: /dev/sdc2: swapon failed: Устройство или ресурс занят
[root@localhost ~]# df -h /mnt/data-ex
Файловая система Размер Использовано Дост Использовано% Смонтировано в
/dev/sdc1 89M 14K 82M 1% /mnt/data-ex
[root@localhost ~]# free -h
      total        used         free       shared  buff/cache   available
Mem:    7,5Gi     1,4Gi    4,7Gi    20Mi    1,7Gi    6,1Gi
Swap:   4,1Gi      0B    4,1Gi
[root@localhost ~]# swapon --show
NAME      TYPE      SIZE USED PRIOS
/dev/dm-1 partition 3,9G  0B   -2
/dev/sdb6 partition 100M  0B   -3
/dev/sdc2 partition 100M  0B   -4
[root@localhost ~]#
```

Рис. 33: 33

## Контрольные вопросы

---

1. Какой инструмент используется для создания разделов GUID?

Ответ - gdisk для разделов GUID

## Контрольные вопросы 2

---

2. Какой инструмент применяется для создания разделов MBR?

Ответ - fdisk - для разделов MBR

## Контрольные вопросы 3

---

3. Какой файл используется для автоматического монтирования разделов во время загрузки?

Ответ - /etc/fstab - для автоподключения при загрузке

## Контрольные вопросы 4

---

4. Какой вариант монтирования целесообразно выбрать, если необходимо, чтобы файловая система не была автоматически примонтирована во время загрузки?

Ответ - использовать опцию монтирования noauto

## Контрольные вопросы 5

---

5. Какая команда позволяет форматировать раздел с типом 82 с соответствующей файловой системой?

Ответ - тип 82 это Linux swap

## Контрольные вопросы 6

---

6. Вы только что добавили несколько разделов для автоматического монтирования при загрузке. Как можно безопасно проверить, будет ли это работать без реальной перезагрузки?

Ответ - проверить без перезагрузки sudo mount -a

## Контрольные вопросы 7

---

7. Какая файловая система создаётся, если вы используете команду mkfs без какой либо спецификации файловой системы?

Ответ - по умолчанию ext2

## Контрольные вопросы 8

---

8. Как форматировать раздел EXT4?

Ответ - mkfs.ext4 /deb/...

## Контрольные вопросы 9

---

9. Как найти UUID для всех устройств на компьютере?

Ответ - командой blkid

## Выводы

---

В ходе выполнения лабораторной работы я получил навыки создания разделов на диске и файловых систем и навыки монтирования файловых систем.