**Практическая работа №13**

**Тема:** Работа с дисками и томами. Управление дисковыми ресурсами.

**Количество часов**: 2

**Цель:** Изучить теоретические сведения и получить практический опыт по управлению дисками и томами, а также файлами и папками в ALT Linux.

**Задание(я):**

1. Изучить теоретические сведения.

2. Создать раздел или логический диск.

3. Выполнить форматирование основного тома.

4. Выполнить расширение основного тома.

5. Выполнить дефрагментацию тома.

6. Выполнить предоставление общего доступа к папке или диску.

*Выводы.*

**Методические указания к выполнению:**

**Выполнение задания 1**

Управление дисками и томами включает: создание и форматирование разделов, логических дисков и томов; задание дисковых квот для ограничения использования диска; дефрагментацию томов для повышения быстродействия файловой системы и исправление ошибок и поврежденных секторов файловой системы на жестком диске. Семейство операционных систем Linux предоставляет множество средств, которые можно использовать для эффективного управления дисками и томами на новой или существующей системах. Эти средства включают программу «Управление дисками», программу «Дефрагментация диска», дисковые квоты и контроль ошибок.

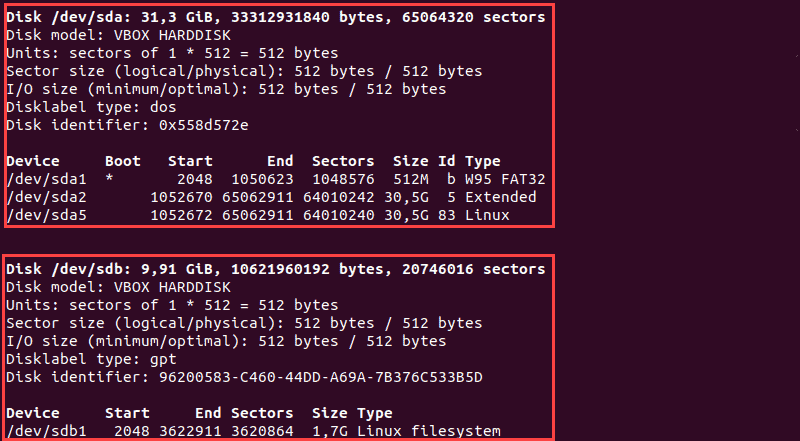
Некоторыми наиболее распространенными задачами являются создание разделов или логических дисков, форматирование основных томов, расширение основных томов и дефрагментация томов. Также можно управлять дисками и томами из командной строки.

Некоторыми наиболее распространенными задачами являются общий доступ к папке или диску, включение теневых копий общих папок и изменение параметров для теневых копий общих папок. Можно также управлять файлами и папками из командной строки.

**Выполнение задания 2**

Перед началом выполнения задания следует убедиться существует ли диск в системе. Иногда наличие устройства в BIOS может быть недостаточным. Доступные накопители проверяем командой:

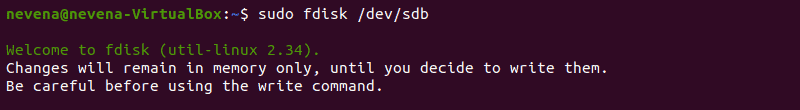
**sudo fdisk –l**



Выберите диск для хранения, на котором вы хотите создать разделы, выполнив следующую команду:

**sudo fdisk /dev/sdb**

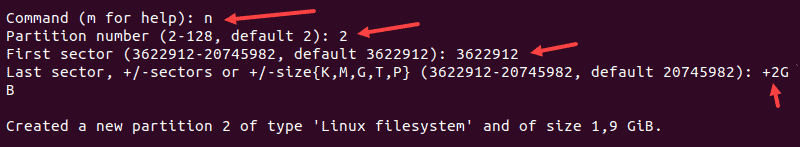
Диск /dev/sdbstorage открыт:



Теперь создадим новый раздел.

1. Запустите команду n, чтобы создать новый раздел.
2. Выберите номер раздела, набрав номер по умолчанию (2).
3. После этого вас попросят указать начальный и конечный сектор вашего жесткого диска. Лучше всего ввести в этом разделе номер по умолчанию (3622912).
4. Последний запрос связан с размером раздела. Вы можете выбрать несколько секторов или установить размер в мегабайтах или гигабайтах. Введите + 2 GB, чтобы установить размер раздела 2 ГБ.

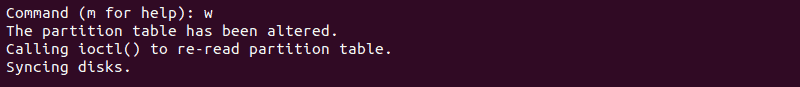
Появится сообщение, подтверждающее создание раздела.



Запись на диск

Система создала раздел, но изменения не записываются на диск.

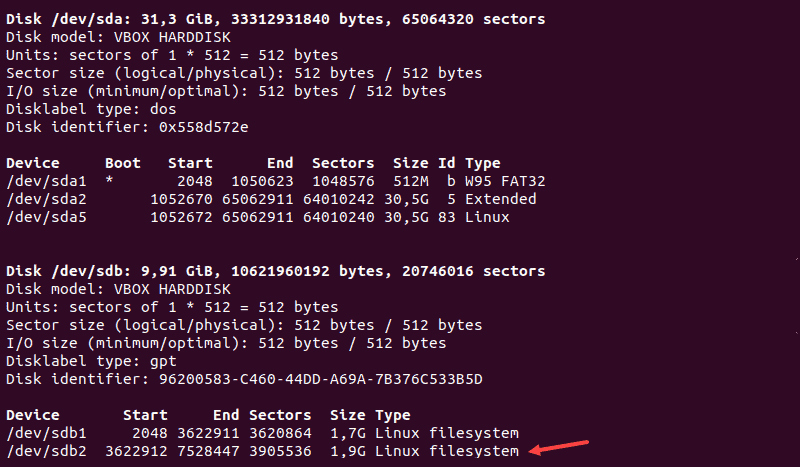
1. Чтобы записать изменения на диск, выполните команду **w**:



2. Убедитесь, что раздел создан, выполнив следующую команду:

**sudo fdisk -l**

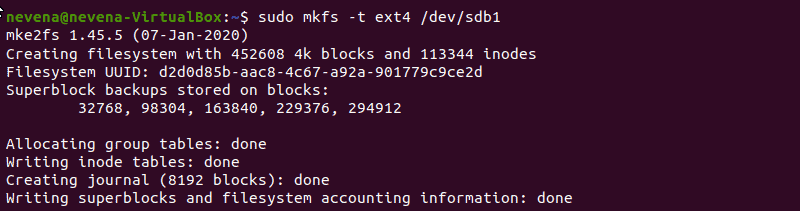
Как видите, раздел /dev/sdb2 создан:



**Выполнения задания 3**

После создания раздела с помощью команды **parted** или **fdisk** отформатируйте его перед использованием, выполнив следующую команду:

**sudo mkfs -t ext4 /dev/sdb1**



Смонтировать раздел

Чтобы начать взаимодействие с диском, создайте точку монтирования (mount point) и смонтируйте к ней раздел.

1. Создайте точку монтирования, выполнив следующую команду:

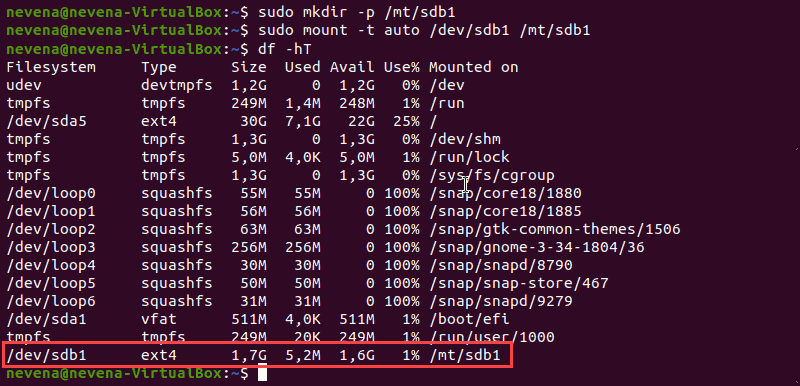
**sudo mkdir -p /mt/sdb1**

2. После этого смонтируйте раздел, введя:

**sudo mount -t auto /dev/sbd1 /mt/sdb1**

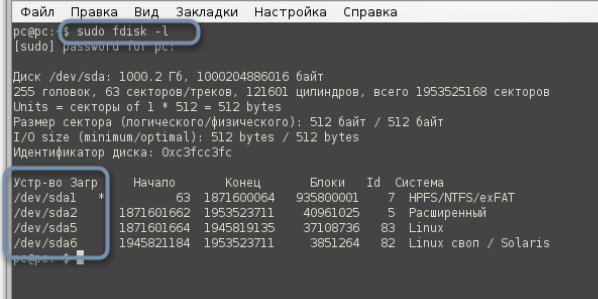
Терминал не распечатывает вывод, если команды выполнены успешно.

3. Убедитесь, что раздел смонтирован, с помощью команды **df hT**:



**Выполнение задания 4**

Если вы не знаете, под какими метками располагается сам диск и его логические тома, используйте команду **sudo fdisk -l**, отобразив список со всеми секторами.



1. Используйте команду **umount /dev/sdb2** для размонтирования раздела. Замените его название на необходимое и учитывайте, что корневой том размонтировать так просто нельзя, понадобится загрузка с **LiveCD** (загрузочная флэшка с ОС Linux).

2. Далее укажите **fdisk /dev/sdb**, подключившись тем самым к указанному диску.

3. Посмотрите все разделы, введя **p** и нажав на **Enter**.

4. За удаление отвечает **d**; после активации этого атрибута понадобится ввести цифру раздела, который вы хотите удалить (при этом форматирования не происходит, все данные сохраняются).

5. Последовательно активируйте **n** (новый раздел), **p** (основной) и **2** (замените на номер тома). При появлении уведомления о создании первого и последнего сектора подтверждайте операцию нажатием на **Enter**.

6. Проверьте список томов еще раз через **p**, а затем нажмите **w** для сохранения изменений.

Осталось только указать файловой системе, что размер тома был расширен до максимального, поскольку пока отображается старое значение. Для этого сначала определите используемую файловую систему через **df -T**. Введите одну из трех следующих команд, отталкиваясь от полученной информации:

Для файловой системы ext2/ext3/ext4:

**resize2fs /dev/vg\_centos/lv\_root**

Для файловой системы XFS:

**xfs\_growfs /dev/sda2**

Для файловой системы Reiserfs:

**resize\_reiserfs /dev/sdb**

В командах номера разделов указаны в качестве примеров, вы же заменяете их на свои. Пример команды для монтирования диска, если ранее он был отключен:

**mount /dev/sda1 /mnt**

Сделать это нужно уже после внесения всех изменений.

Для уменьшения размера раздела используйте последовательность команд:

**umount /dev/vdb1**

**fsck -f /dev/vdb1**

**resize2fs /dev/vdb1 100M**

**fdisk /dev/vdb**

После ввода последней снова используйте последовательную активацию букв так, как это было показано в предыдущей инструкции, чтобы создать новый раздел. Кроме того, не забудьте смонтировать его обратно.

**Выполнения задания 5**

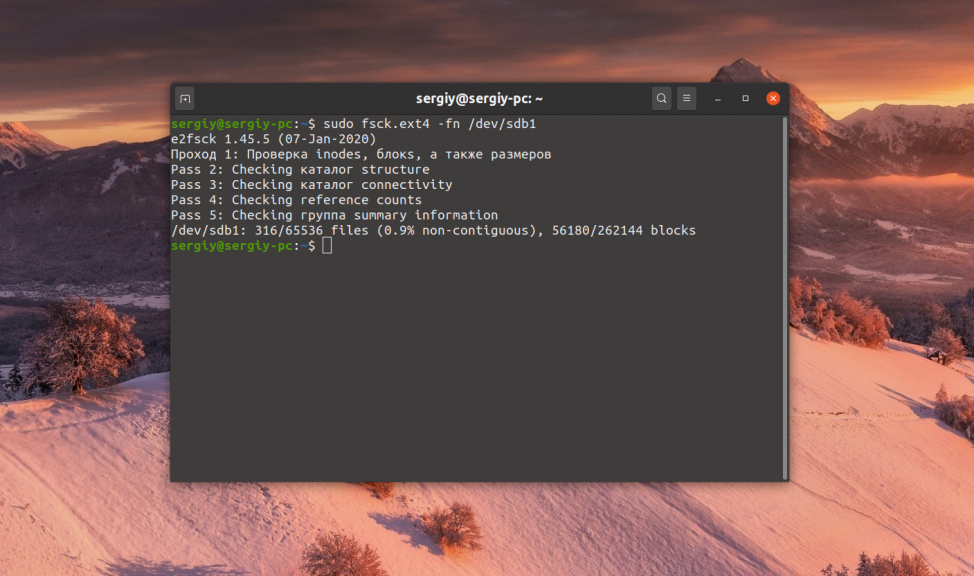
Первым делом надо понять, что же такое фрагментация. Жесткий диск состоит из множества маленьких секторов, в каждом из которых хранится небольшая частичка данных. Если файл большой он разделяется на несколько секторов. Когда вы записываете файл в файловых системах FAT или NTFS, под него выделяется определённое количество секторов, а после него записываются уже следующие файлы, поэтому если вы захотите обновить файл и добавить в него информацию, то для этого уже будут использоваться секторы в другой части диска. Если таких операций будет много, то файл может оказаться раскиданным по всему диску и при чтении диску придется очень часто перемещать читающую головку, что замедлит производительность. Это и есть фрагментация.

Но Ext4 большой фрагментации не подвержена, потому что файловая система выделяет много места после записанного файла и использует его. А даже если возникает фрагментация, то она пытается переместить файл в свободное место. Здесь фрагментация может возникнуть только если файловая система переполнена и то это возникает довольно редко.

Надо ещё отметить, что для SSD дисков дефрагментация не нужна. Там нет движущихся частей, а поэтому фрагментация не замедляет их работу, зато дефрагментация уменьшает срок службы.

Чтобы посмотреть какая сейчас фрагментация диска можно использовать утилиту fsck. Ей надо передать параметр -f. Чтобы в файловую систему не вносились никакие изменения используйте опцию -n:

**sudo fsck -fn /dev/sdb1**



Здесь /dev/sdb1 - это раздел, который надо проверить. Фрагментация указывается в скобках и измеряется в процентах, обычно она очень маленькая и не достигает даже одного процента. Не рекомендуется выполнять проверку на примонтированной системе, иначе вы получите кучу ложных ошибок, однако если с файловой системой не будут выполнятся никакие действия, то вы ничего не повредите. Для дефрагментации нам понадобится утилита e4defrag, находящаяся в папке e2fsprogs. Если у вас этот набор утилит ещё не установлен, установите его командой:

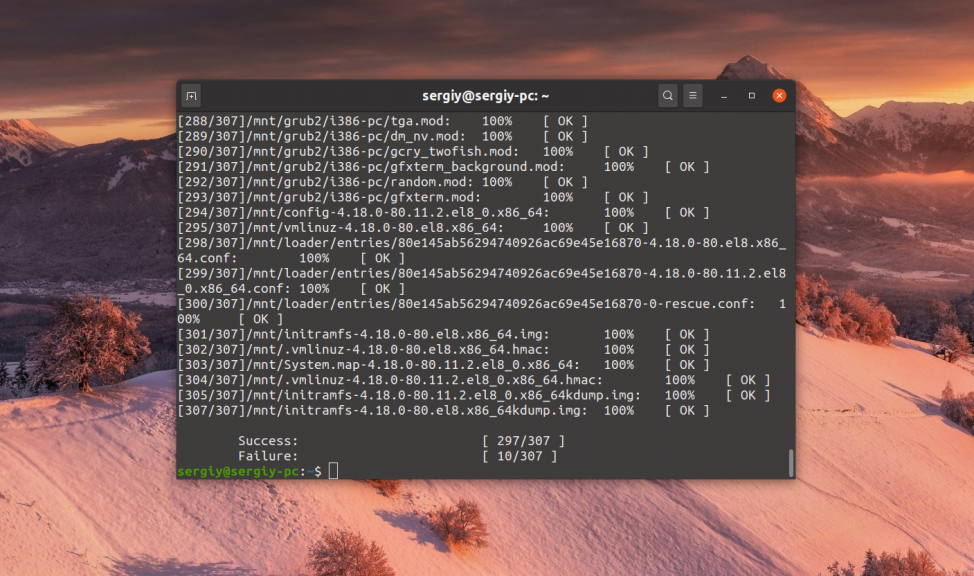
**sudo apt install e2fsprogs**

Теперь можете дефрагментировать нужный раздел. Например, тот же /dev/sdb1. Но перед дефрагментацией раздел надо примонтировать:

**sudo mount /dev/sdb1 /mnt**

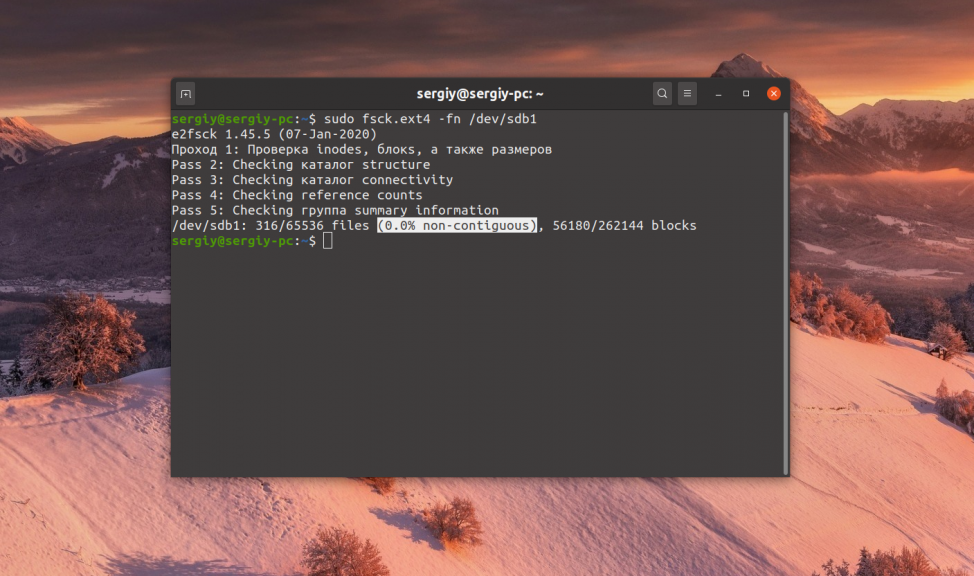
Затем дефрагментация Ext4:

**sudo e4defrag /dev/sdb1**



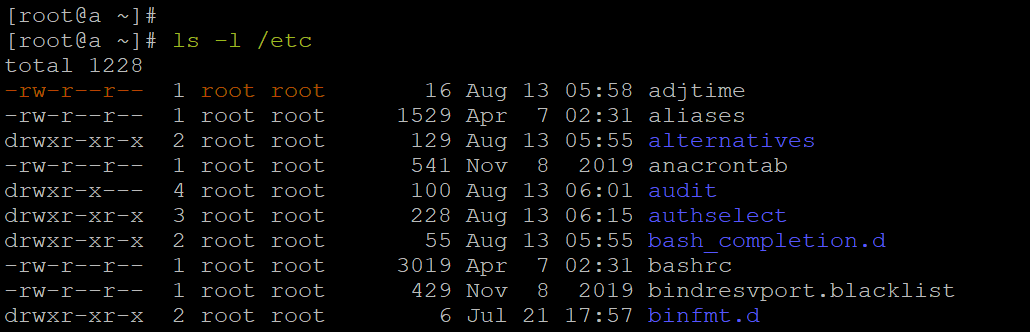
После этого можно снова проверить фрагментацию. Если есть возможность, лучше отмонтировать перед этим раздел. Как видите, теперь фрагментация 0%:

**sudo fsck -fn /dev/sdb1**



**Выполнение задания 6**

Общая схема прав одинакова и для файлов, и для папок. Можно увидеть, как система отображает права доступа, в выводе команды **ls -l**, которая показывает содержимое директории:



Первые 10 символов содержат информацию о правах к файлу или каталогу.

|  |  |
| --- | --- |
| - | тип файла |
| rw- | права пользователя-владельца |
| r-- | права пользователей группы-владельца |
| r-- | права всех остальных пользователей |

Существуют следующие операции с файлом: **чтение (r)**, **запись (w)** или выполнение **(x) — для исполняемых** **файлов**. Для директорий параметры те же, но обозначают немного другое: **просмотр** директории (r), **создание** папок / файлов (w) внутри директории, **переход** в директорию (x).

Каждый из этих уровней доступа можно выразить в восьмеричной системе с помощью числового значения: **4 (r), 2 (w), 1 (x)**. Вот так мы и получаем общую схему прав:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| a – любые пользователи | | | | | | | | |
| u – права пользователя | | | g – права группы | | | o – права всех остальных | | |
| r | w | x | r | w | x | r | w | x |
| 4 | 2 | 1 | 4 | 2 | 1 | 4 | 2 | 1 |

Для отображения уровня прав помимо формата rwxrwxrwx используется упомянутый восьмеричный формат. Для этого достаточно сложить все уровни прав по категориям:

rwxrwxrwx = (4+2+1), (4+2+1), (4+2+1) = 777

Для работы непосредственно с правами используется команда chmod:

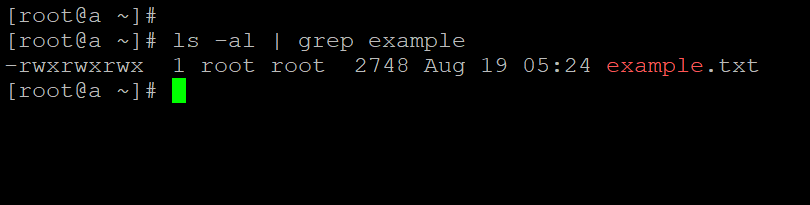
**chmod [настройки прав] [файл или папка]**

При работе с директориями опция -R позволит изменить права на все вложенные файлы и папки.

Настройки прав в chmod можно определять двумя способами:

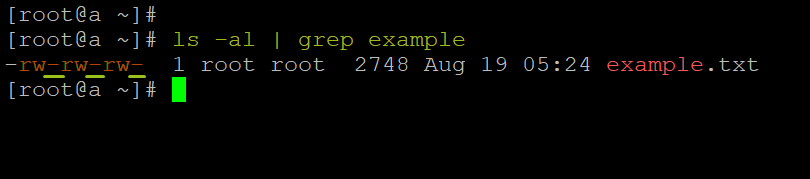
1. Указав **категорию** (u — пользователь-владелец, g — группа-владелец, o — другие пользователи, a — все пользователи), **модификатор** (+, - , =) и, соответственно, нужные **права** (r, w, x).

Например, представим, что у нас есть файл example.txt с максимальным уровнем прав для всех категорий пользователей:



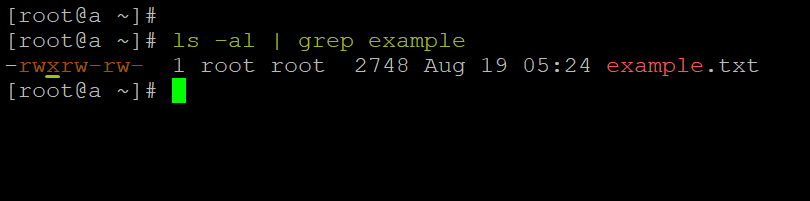
Допустим, мы не хотим, чтобы кто-то в принципе мог запускать этот файл на выполнение. В таком случае нам нужно убрать параметр «x» из прав всех категорий пользователей сразу. Это можно сделать так:

**chmod a-x example.txt**



Представим, что потом мы решили вернуть владельцу права на запуск файла. То есть нам нужно добавить параметр «x» в категорию пользователя-владельца:

**chmod u+x example.txt**



**Вопросы для самоконтроля:**

1. Объясните понятие «Дефрагментация диска».

2. Для чего предназначена команда sudo fdisk –l?

3. Что включает в себя управление дисками?

4. Что такое фрагментация?

5. Где могут быть созданы основные и дополнительные разделы и логические диски?

6. Форматирование каких разделов нельзя выполнить?

7. Какие существуют операции с файлом?

8. На каких накопителях нельзя выполнять дефрагментацию диска?

**Список литературы и ссылки на Интернет-ресурсы, содержащие информацию по теме:**