Отчёт по лабораторной работе № 5

Анализ файловой системы Linux. Команды для работы с файлами и каталогами

Нати Франшиску Бунда

Содержание

# 1 Цель работы

Ознакомление с файловой системой Linux, её структурой, именами и содержанием каталогов. Приобретение практических навыков по применению команд для работы с файлами и каталогами, по управлению процессами (и работами), по проверке использования диска и обслуживанию файловой системы.

# 2 Задание

* Ознакомиться и разобрать на практике основные команды для работы с файловой системой Linux, её структурой, именами и содержанием каталогов
* Выполнить упражнения
* Ответить на контрольные вопросы

# 3 Выполнение лабораторной работы

## 3.1 Команды для работы с файлами и каталогами

Для создания текстового файла воспользовались командой touch (рис. [[1](#fig:001)])

Figure 1: touch

Figure 1: touch

Для просмотра файла небольшого размера можно воспользовались командой cat.(рис. [[2](#fig:002)])



Figure 2: cat

Для просмотра файла постранично воспользовались командой less. (рис. [[3](#fig:003)]), (рис. [[4](#fig:004)])

Figure 3: less

Figure 3: less

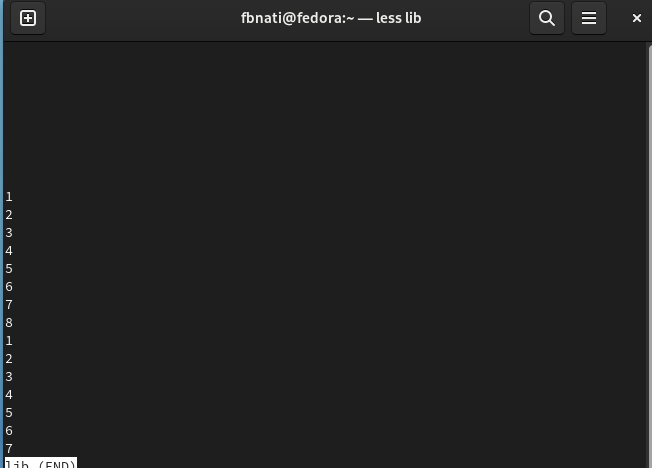


Figure 4: less

Для просмотра первых 10 страниц файла воспользовались командой head. (рис. [[5](#fig:005)])

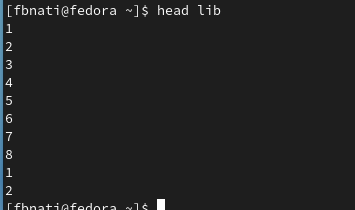


Figure 5: head

Для просмотра последних 10 страниц файла воспользовались командой tail. (рис. [[6](#fig:006)])

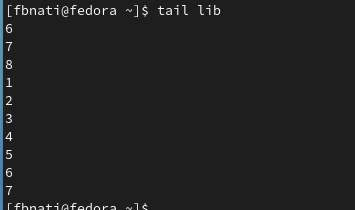


Figure 6: tail

## 3.2 Копирование, перемещение и переименование файлов и каталогов

Выполнили следующие действия, зафиксировав в отчёте по лабораторной работе используемые при этом команды и результаты их выполнения:

1. Скопировали файл /usr/include/sys/io.h в домашний каталог и назвали его equipment. (рис. [[7](#fig:007)]), (рис. [[8](#fig:008)])

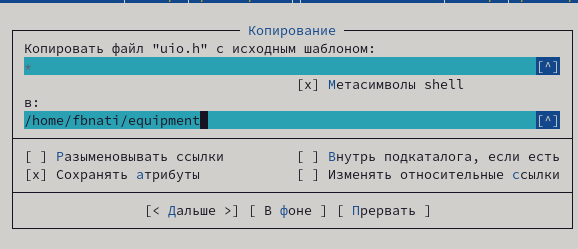


Figure 7: equipment

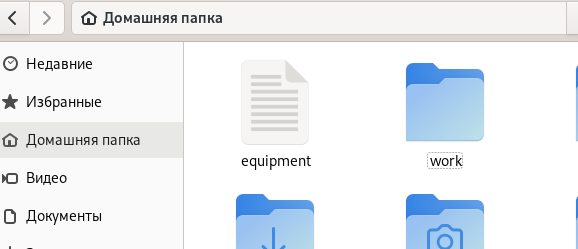


Figure 8: equipment

1. В домашнем каталоге создали директорию ~/ski.plases. (рис. [[9](#fig:009)]), (рис. [[10](#fig:010)])

Figure 9: ski.plases.

Figure 9: ski.plases.

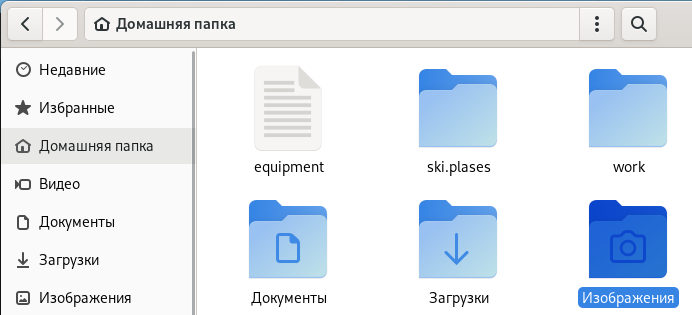


Figure 10: ski.plases.

1. Переместили файл equipment в каталог ~/ski.plases.(рис. [[11](#fig:011)]), (рис. [[12](#fig:012)])

Figure 11: Файл equipment

Figure 11: Файл equipment

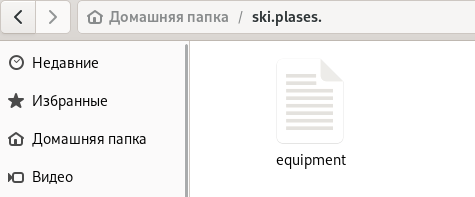


Figure 12: Файл equipment

1. Переименовали файл ~/ski.plases/equipment в ~/ski.plases/equiplist. (рис. [[13](#fig:013)]), (рис. [[14](#fig:014)])

Figure 13: equiplist

Figure 13: equiplist

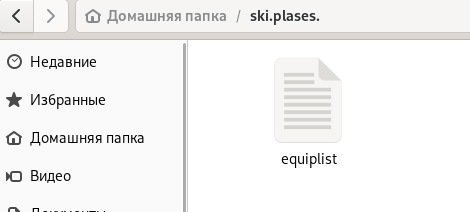


Figure 14: equiplist

1. Создали в домашнем каталоге файл abc1 и скопировали его в каталог ~/ski.plases, назвали его equiplist2. (рис. [[15](#fig:015)]), (рис. [[16](#fig:016)])

Figure 15: equiplist2

Figure 15: equiplist2

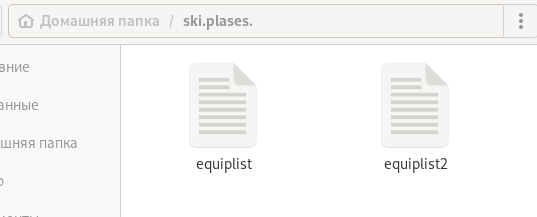


Figure 16: equiplist2

1. Создали каталог с именем equipment в каталоге ~/ski.plases. (рис. [[17](#fig:017)]), (рис. [[18](#fig:018)])

Figure 17: equipment

Figure 17: equipment

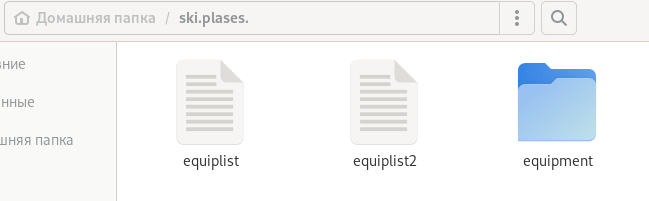


Figure 18: equipment

1. Переместите файлы ~/ski.plases/equiplist и equiplist2 в каталог ~/ski.plases/equipment. (рис. [[19](#fig:019)]), (рис. [[20](#fig:020)])

Figure 19: ~/ski.plases/equipment

Figure 19: ~/ski.plases/equipment

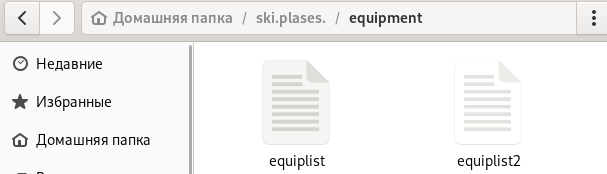


Figure 20: ~/ski.plases/equipment

1. Создали и переместили каталог ~/newdir в каталог ~/ski.plases и назвали его plans. (рис. [[21](#fig:021)]), (рис. [[22](#fig:022)]), (рис. [[23](#fig:023)])

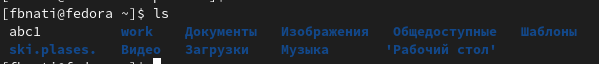


Figure 21: newdir

Figure 22: newdir

Figure 22: newdir

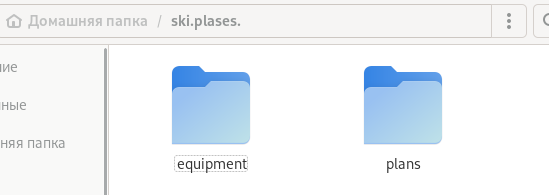


Figure 23: plans

## 3.3 Права доступа

Создали нужные файлы. Определили опции команды chmod, необходимые для того, чтобы присвоить перечисленным ниже файлам выделенные права доступа, считая, что в начале таких прав нет. (рис. [[24](#fig:024)]) 1. drwxr–r– … australia 2. drwx–x–x … play 3. -r-xr–r– … my\_os 4. -rw-rw-r– … feathers

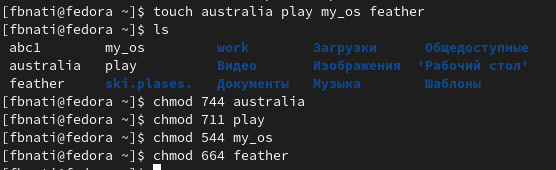


Figure 24: chmod

## 3.4 Выполнение упражнений

Проделали приведённые ниже упражнения, записывая в отчёт по лабораторной работе используемые при этом команды:

1. Просмотрели содержимое файла /etc/passwd. (рис. [[25](#fig:025)])

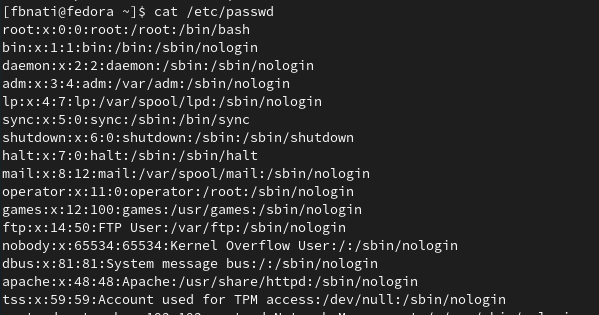


Figure 25: Содержимое файла /etc/passwd

1. Скопировали файл ~/feathers в файл ~/file.old. (рис. [[26](#fig:026)])

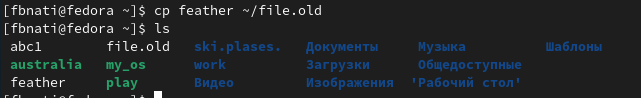


Figure 26: file.old

1. Переместили файл ~/file.old в каталог ~/play. (рис. [[27](#fig:027)])

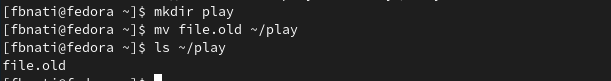


Figure 27: play

1. Скопировали каталог ~/play в каталог ~/fun. (рис. [[28](#fig:028)])

Figure 28: fun

Figure 28: fun

1. Переместили каталог ~/fun в каталог ~/play и назвали его games. (рис. [[29](#fig:029)])

Figure 29: games

Figure 29: games

1. Лишили владельца файла ~/feather права на чтение. (рис. [[30](#fig:030)])

Figure 30: feather

Figure 30: feather

1. Попытались просмотреть файл ~/feather командой cat, получили отказ в доступе. (рис. [[31](#fig:031)])

Figure 31: feather

Figure 31: feather

1. Попытались скопировать файл ~/feather, получили отказ в доступе. (рис. [[32](#fig:032)])

Figure 32: feather

Figure 32: feather

1. Дали владельцу файла ~/feathers право на чтение. (рис. [[33](#fig:033)])

Figure 33: feather

Figure 33: feather

1. Лишили владельца каталога ~/play права на выполнение. (рис. [[34](#fig:034)])

Figure 34: play

Figure 34: play

1. Попытались перейти в каталог ~/play, получили отказ в доступе. (рис. [[35](#fig:035)])

Figure 35: play

Figure 35: play

1. Дали владельцу каталога ~/play право на выполнение. (рис. [[36](#fig:036)])

Figure 36: play

Figure 36: play

## 3.5 Команда man

Прочитали man по командам mount, fsck, mkfs, kill и кратко их охарактеризовали.

Команда mount используется для монтирования файловой системы в Linux, её также можно использовать для монтирования образа ISO, монтирования удалённой файловой системы Linux и многого другого. (рис. [[37](#fig:037)])

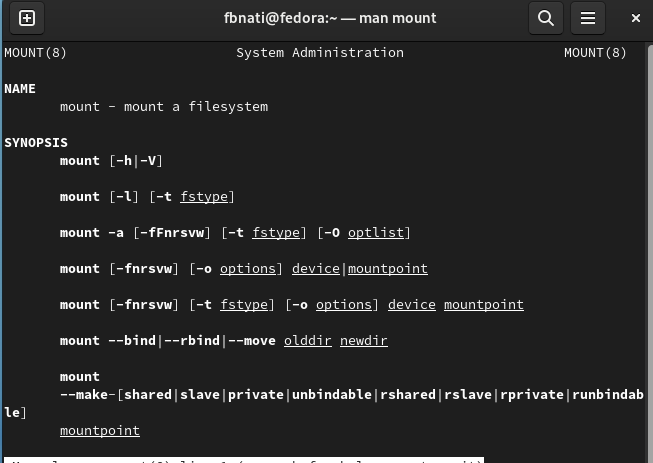


Figure 37: mount

fsck используется для проверки и, при необходимости, для восстановления файловых систем Linux, она также может печатать тип файловой системы на указанных разделах диска. (рис. [[38](#fig:038)])

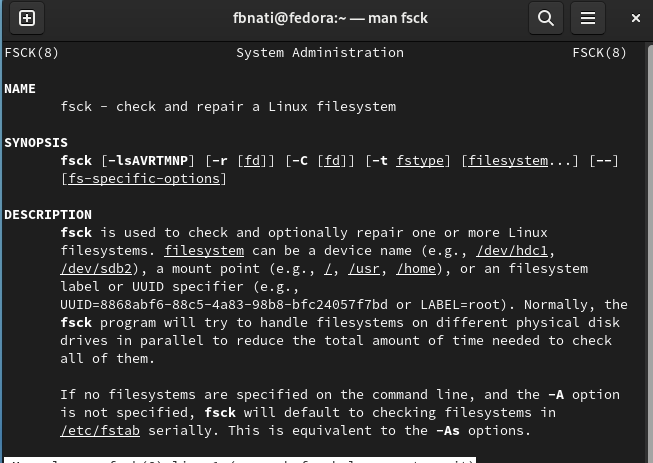


Figure 38: fsck

Буквы в mkfs означают «make file system» (создать файловую систему). Команда обычно используется для управления устройствами хранения в Linux. Отвечает за создание файловых систем. (рис. [[39](#fig:039)])

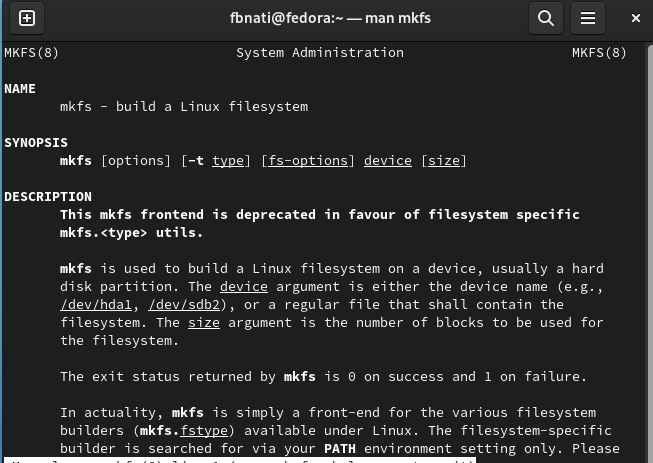


Figure 39: mkfs

Команда kill предназначена для посылки сигнала процессу. По умолчанию, если мы не указываем какой сигнал посылать, посылается сигнал SIGTERM (от слова termination — завершение). SIGTERM указывает процессу на то, что необходимо завершиться. (рис. [[40](#fig:040)])

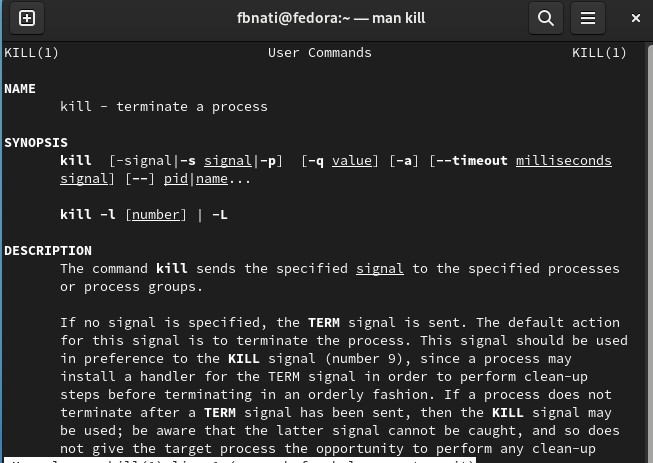


Figure 40: kill

# 4 Выводы

В ходе выполнения лабораторной работы ознакомились с файловой системой Linux, её структурой, именами и содержанием каталогов. Приобрели практические навыки по применению команд для работы с файлами и каталогами, по управлению процессами (и работами), по проверке использования диска и обслуживанию файловой системы.

# 5 Ответы на контрольные вопросы

1. Дайте характеристику каждой файловой системе, существующей на жёстком диске компьютера, на котором вы выполняли лабораторную работу. (рис. [[41](#fig:041)])

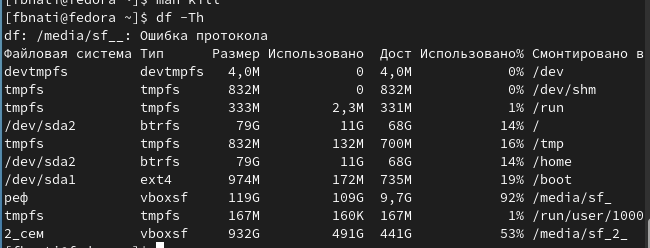


Figure 41: kill

1. Приведите общую структуру файловой системы и дайте характеристику каждой директории первого уровня этой структуры.

* / — root каталог. Содержит в себе всю иерархию системы;
* /bin — здесь находятся двоичные исполняемые файлы. Основные общие команды, хранящиеся отдельно от других программ в системе (прим.: pwd, ls, cat, ps);
* /boot — тут расположены файлы, используемые для загрузки системы (образ initrd, ядро vmlinuz);
* /dev — в данной директории располагаются файлы устройств (драйверов). С помощью этих файлов можно взаимодействовать с устройствами. К примеру, если это жесткий диск, можно подключить его к файловой системе. В файл принтера же можно написать напрямую и отправить задание на печать;
* /etc — в этой директории находятся файлы конфигураций программ. Эти файлы позволяют настраивать системы, сервисы, скрипты системных демонов;
* /home — каталог, аналогичный каталогу Users в Windows. Содержит домашние каталоги учетных записей пользователей (кроме root). При создании нового пользователя здесь создается одноименный каталог с аналогичным именем и хранит личные файлы этого пользователя;
* /lib — содержит системные библиотеки, с которыми работают программы и модули ядра;
* /lost+found — содержит файлы, восстановленные после сбоя работы системы. Система проведет проверку после сбоя и найденные файлы можно будет посмотреть в данном каталоге;
* /media — точка монтирования внешних носителей. Например, когда вы вставляете диск в дисковод, он будет автоматически смонтирован в директорию /media/cdrom;
* /mnt — точка временного монтирования. Файловые системы подключаемых устройств обычно монтируются в этот каталог для временного использования;
* /opt — тут расположены дополнительные (необязательные) приложения. Такие программы обычно не подчиняются принятой иерархии и хранят свои файлы в одном подкаталоге (бинарные, библиотеки, конфигурации);
* /proc — содержит файлы, хранящие информацию о запущенных процессах и о состоянии ядра ОС;
* /root — директория, которая содержит файлы и личные настройки суперпользователя;
* /run — содержит файлы состояния приложений. Например, PID-файлы или UNIX-сокеты;
* /sbin — аналогично /bin содержит бинарные файлы. Утилиты нужны для настройки и администрирования системы суперпользователем;
* /srv — содержит файлы сервисов, предоставляемых сервером (прим. FTP или Apache HTTP);
* /sys — содержит данные непосредственно о системе. Тут можно узнать информацию о ядре, драйверах и устройствах;
* /tmp — содержит временные файлы. Данные файлы доступны всем пользователям на чтение и запись. Стоит отметить, что данный каталог очищается при перезагрузке;
* /usr — содержит пользовательские приложения и утилиты второго уровня, используемые пользователями, а не системой. Содержимое доступно только для чтения (кроме root). Каталог имеет вторичную иерархию и похож на корневой;
* /var — содержит переменные файлы. Имеет подкаталоги, отвечающие за отдельные переменные. Например, логи будут храниться в /var/log, кэш в /var/cache, очереди заданий в /var/spool/ и так далее.

1. Какая операция должна быть выполнена, чтобы содержимое некоторой файловой системы было доступно операционной системе? (Монтирование тома)
2. Назовите основные причины нарушения целостности файловой системы. Как устранить повреждения файловой системы?

Отсутствие синхронизации между образом файловой системы в памяти и ее данными на диске в случае аварийного останова может привести к появлению следующих ошибок:

* Один блок адресуется несколькими mode (принадлежит нескольким файлам).
* Блок помечен как свободный, но в то же время занят (на него ссылается onode).
* Блок помечен как занятый, но в то же время свободен (ни один inode на него не ссылается).
* Неправильное число ссылок в inode (недостаток или избыток ссылающихся записей в каталогах).
* Несовпадение между размером файла и суммарным размером адресуемых inode блоков.
* Недопустимые адресуемые блоки (например, расположенные за пределами файловой системы).
* “Потерянные” файлы (правильные inode, на которые не ссылаются записи каталогов).
* Недопустимые или неразмещенные номера inode в записях каталогов.

1. Как создаётся файловая система?

* mkfs - позволяет создать файловую систему Linux.

1. Дайте характеристику командам для просмотра текстовых файлов.

* Cat - выводит содержимое файла на стандартное устройство вывода

1. Приведите основные возможности команды cp в Linux.

* Cp – копирует или перемещает директорию, файлы.

1. Приведите основные возможности команды mv в Linux.

* Mv - переименовать или переместить файл или директорию

1. Что такое права доступа? Как они могут быть изменены?

* Права доступа к файлу или каталогу можно изменить, воспользовавшись командой chmod. Сделать это может владелец файла (или каталога) или пользователь с правами администратора.