**Липецкий государственный технический университет**

Факультет автоматизации и информатики

Кафедра Автоматизированных систем управления

ЛАБОРАТОРНАЯ РАБОТА №1

по курсу "Операционная система Linux"

Работа с файловой системой

Курбатова К.А.

|  |
| --- |
| Студент |
| Группа АИ-17-1 | Кургасов В.В. |  |
| Руководитель  Доцент |  |  |

Липецк 2019г.

Цель работы

Приобрести опыт работы с файлами и каталогами в ОС Linux, настройки прав на доступ к файлам и каталогам.

Задание

1. Запустить виртуальную машину с Linux Ubuntu.

2. Загрузиться пользователем root (sudo su).

3. Ознакомиться со структурой системных каталогов ОС Linux на рабочем месте. Изучить стандарт (2.1. Filesystem Hierarchy Standard)

4. Привести в отчете перечень каталогов с указанием их назначения.

5. Просмотреть содержимое каталога файлов физических устройств. В отчете привести перечень файлов физических устройств на рабочем месте с указанием назначения файлов.

6. Перейти в директорий пользователя root. Просмотреть содержимое каталога. Просмотреть содержимое файла vmlinuz. Просмотреть и пояснить права доступа к файлу vmlinuz.

7. Создать нового пользователя user

8. Создать в директории пользователя user три файла 1.txt, 2.txt и 3.txt, используя команды touch, cat и текстовый редактор (на выбор vi/nano). Просмотреть и пояснить права доступа к файлам.

9. Перейти в директории пользователя root. В отчете описать результат.

10. Изменить права доступа на файл 1.txt в директории пользователя user.

11. Создать жесткую и символическую ссылки на файл 2.txt. Просмотреть результаты.

12. Создать каталог new в каталоге пользователя user.

13. Скопировать файл 1.txt в каталог new.

14. Переместить файл 2.txt в каталог new.

15. Изменить владельца файла 3.txt и каталога new.

16. Удалить файл 1.txt в каталоге new.

17. Удалить каталог new.

18. Найти, используя команду find, файл vga2iso (или другой файл по заданию преподавателя).

Ход работы

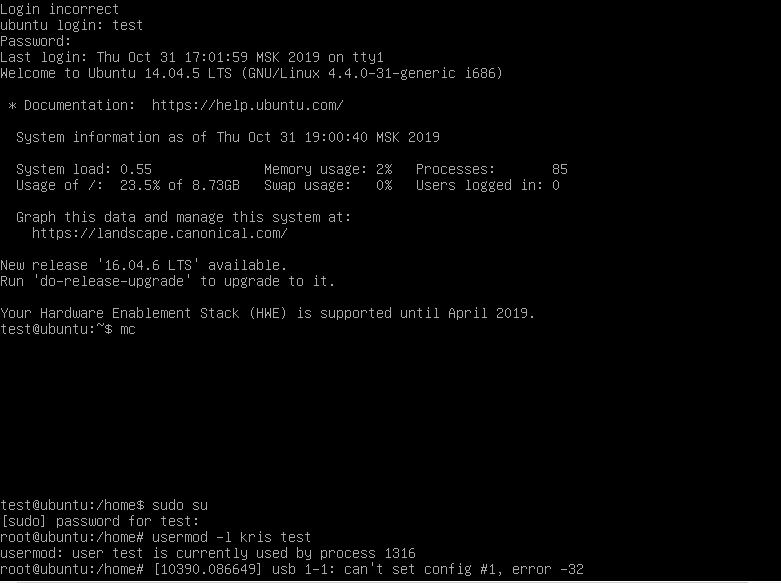


Рисунок 1 – Запущенная виртуальная машина.



Рисунок 2 – Пользователь root.

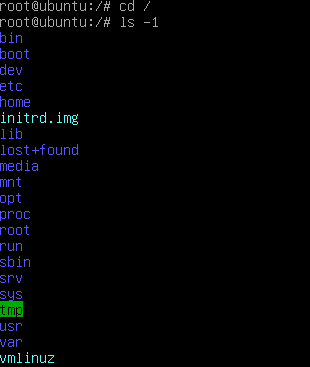
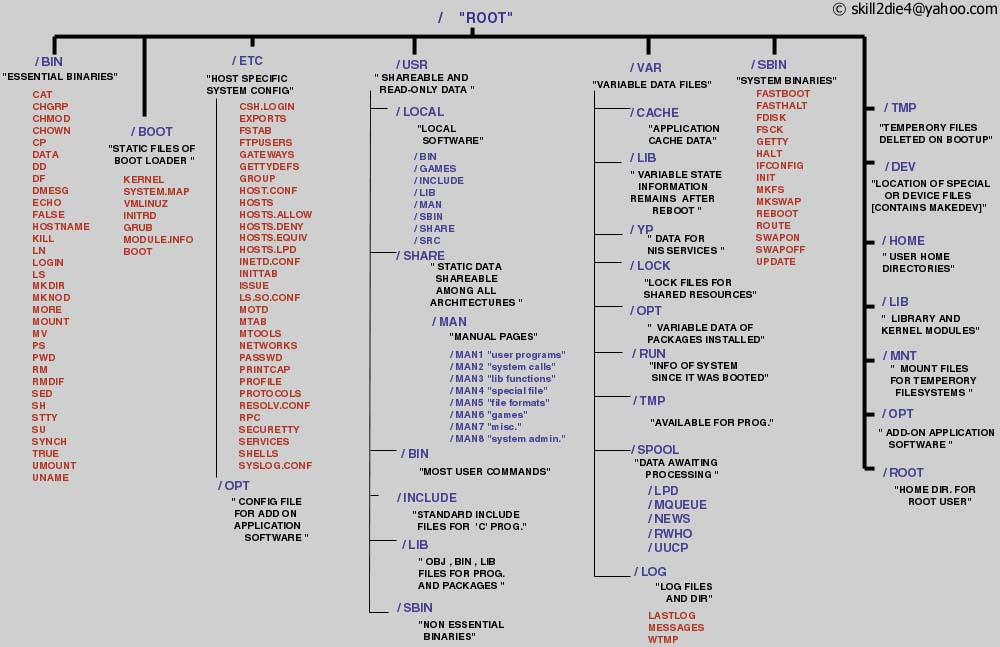


Рисунок 3 – Перечень каталогов в корневой директории

Стандарт иерархии файловой системы

Рисунок 4 - Filesystem Hierarchy Standard, «стандарт иерархии файловой системы»

Перечень каталогов с указанием их назначения

/“ROOT” – корень каталог, содержащий всю файловую систему;

/BIN – основные [утилиты](https://ru.wikipedia.org/wiki/%D0%A3%D1%82%D0%B8%D0%BB%D0%B8%D1%82%D0%B0), необходимые как в однопользовательском режиме, так и при обычной работе всем [пользователям](https://ru.wikipedia.org/wiki/%D0%9F%D0%BE%D0%BB%D1%8C%D0%B7%D0%BE%D0%B2%D0%B0%D1%82%D0%B5%D0%BB%D1%8C);

/BOOT – загрузочные файлы;

/ETC – общесистемные конфигурационные файлы (настройки программ);

/ETC/OPT – файлы конфигураций

/USR - Вторичная иерархия для данных [пользователя](https://ru.wikipedia.org/wiki/%D0%9F%D0%BE%D0%BB%D1%8C%D0%B7%D0%BE%D0%B2%D0%B0%D1%82%D0%B5%D0%BB%D1%8C). Содержит большинство пользовательских [приложений](https://ru.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BC%D0%BF%D1%8C%D1%8E%D1%82%D0%B5%D1%80%D0%BD%D0%B0%D1%8F_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B0) и [утилит](https://ru.wikipedia.org/wiki/%D0%A3%D1%82%D0%B8%D0%BB%D0%B8%D1%82%D0%B0), используемых в многопользовательском режиме;

/USR/LOCAL - третичная иерархия для данных, специфичных для данного хоста. Обычно содержит такие подкаталоги, как bin, lib, share;

/USR/SHARE - архитектурно-независимые общие данные;

/USR/SHARE/MAN – справочные страницы;

/USR/BIN - дополнительные программы для всех пользователей, не являющиеся необходимыми в однопользовательском режиме;

/USR/INCLUDE - стандартные [заголовочные файлы](https://ru.wikipedia.org/wiki/%D0%97%D0%B0%D0%B3%D0%BE%D0%BB%D0%BE%D0%B2%D0%BE%D1%87%D0%BD%D1%8B%D0%B9_%D1%84%D0%B0%D0%B9%D0%BB);

/USR/LIB - библиотеки для программ, находящихся в /usr/bin и /usr/sbin;

/USR/SBIN - дополнительные системные программы;

/VAR - изменяемые файлы, такие как файлы регистрации, временные почтовые файлы;

/VAR/CASHE – кэш приложений;

/VAR/LIB - информация о состоянии. Постоянные данные, изменяемые программами в процессе работы (например, базы данных, метаданные пакетного менеджера и др.);

/VAR/YP – данные для сервисов NIS( Network Information Services (Сетевые Информационные Службы));

/VAR/LOCK - файлы блокировки, указывающие на занятость некоторого ресурса ;

/VAR/OPT – переменные данные для пакетов, установленных в /opt;

/VAR/RUN - информационная система с момента её загрузки (устарело);

/VAR/TMP - временные файлы, которые должны быть сохранены между перезагрузками;

/VAR/SPOOL - задачи, ожидающие обработки;

/VAR/LOG - различные файлы регистрации;

/TMP – временные файлы;

/DEV – основные файлы устройств;

/HOME - содержит [домашние каталоги пользователей](https://ru.wikipedia.org/wiki/%D0%94%D0%BE%D0%BC%D0%B0%D1%88%D0%BD%D0%B8%D0%B9_%D0%BA%D0%B0%D1%82%D0%B0%D0%BB%D0%BE%D0%B3), которые в свою очередь содержат персональные настройки и данные пользователя. Часто размещается на отдельном разделе;

/LIB - основные библиотеки, необходимые для работы программ из /bin и /sbin;

/MNT - содержит временно монтируемые файловые системы;

/OPT- дополнительное [программное обеспечение](https://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%BD%D0%BE%D0%B5_%D0%BE%D0%B1%D0%B5%D1%81%D0%BF%D0%B5%D1%87%D0%B5%D0%BD%D0%B8%D0%B5);

/ROOT- домашний каталог пользователя root.

Перечень файлов физических устройств и их назначение

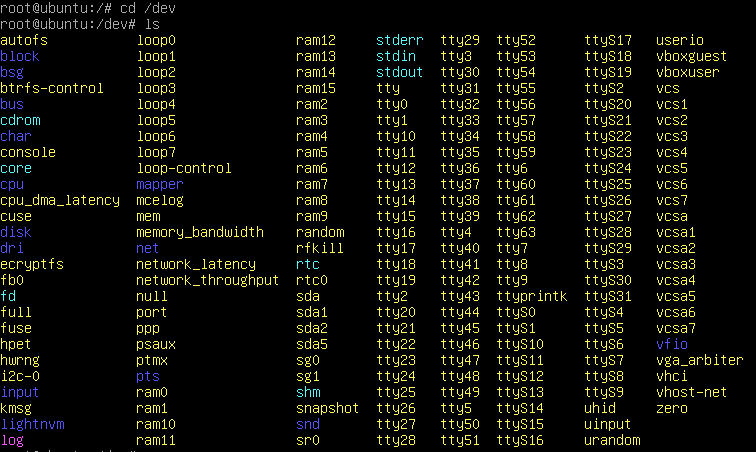


Рисунок 5 – Перечень фалов физических устройств из директории /dev

Основные специальные файлы

|  |  |
| --- | --- |
| Имя | Значения |
| /dev/console | Системная консоль, т. е. монитор и клавиатура, физически подключенные к компьютеру |
| /dev/hd | Жесткие диски с IDE-интерфейсом. Устройство /dev/hda1 соответствует первому разделу на первом жестком диске (/dev/hda), т. е. на диске, подключенном как Primary Master |
| /dev/sd | Жесткие диски с SCSI-интерфейсом |
| /dev/fd | Файлы дисководов для гибких дисков. Первому дисководу соответствует /dev/fd0, второму /dev/fd1 |
| /dev/tty | Файлы поддержки пользовательских консолей. Название сохранилось с тех пор, когда к системе UNIX подключались телетайпы в качестве терминалов. В Linux эти файлы устройств обеспечивают работу виртуальных консолей (переключаться между которыми можно с помощью ‹Alt›+‹F1› - ‹Alt›+‹F6›) |
| /dev/pty | Файлы поддержки псевдо-терминалов. Применяются для удаленных рабочих сессий с использованием telnet |
| /dev/ttS | Файлы, обеспечивающие работу с последовательными портами. /dev/ttS0 соответствует COM1 в MS-DOS, /dev/ttS1 - COM2. Если ваша мышь подключается через последовательный порт, то /dev/mouse является символической ссылкой на соответствующий /dev/ttSN |
| /dev/cua | Специальные устройства для работы с модемами |
| /dev/null | Это устройство - просто черная дыра. Все, что записывается в /dev/null, навсегда потеряно. На это устройство можно перенаправить вывод ненужных сообщений. Если /dev/null используется как устройство ввода, то оно ведет себя как файл нулевой длины |

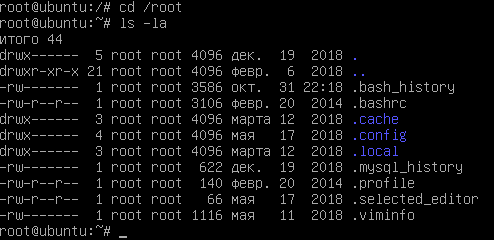


Рисунок 6 – Содержимое каталога /root

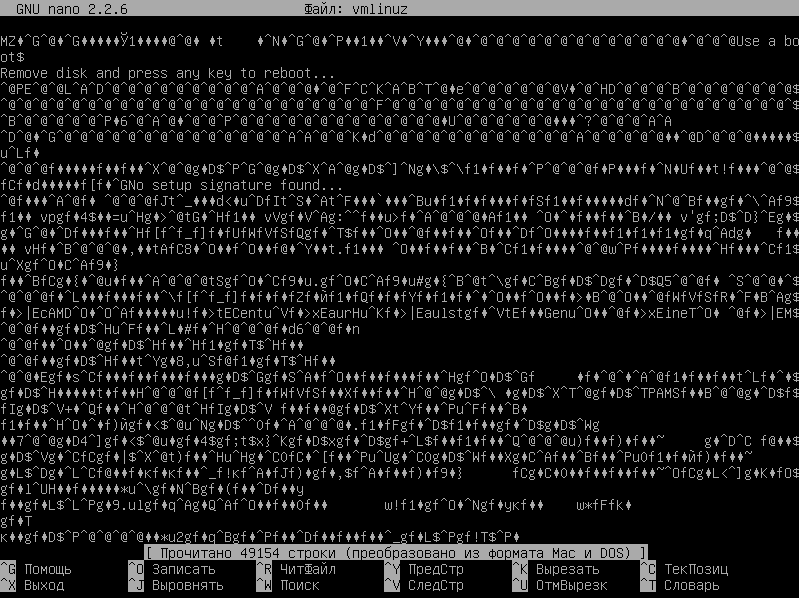


Рисунок 7 – Содержимое файла vmlinuz



Рисунок 8 – Доступы к файлу vmlinuz

Доступы у ссылки :

«rwx», что расшифровывается, как чтение, запись и выполнение. В данном случае это доступно пользователю, группе и остальным.

Создание нового пользователя user

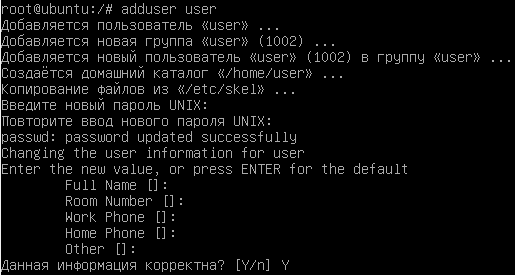


Рисунок 9 – Создание нового пользователя



Рисунок 10 – Вход за него в систему

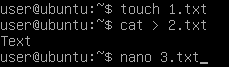


Рисунок 11 – Создание с помощью touch, cat, nano файлов в директории нового пользователя

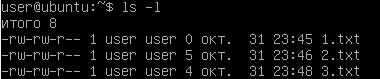


Рисунок 12 – Доступы к созданным файлом

Пояснение к правам доступа файлов 1.txt, 2.txt, 3.txt:

Права доступа к чтению и записи имеют пользователь и группа, все остальные же имеют только право на чтение.



Рисунок 13 – Попытка зайти в директорию /root пользователем user.

У пользователя нет доступа к директории /root.

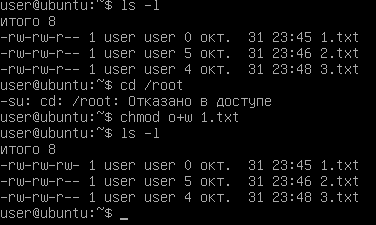


Рисунок 14 – Изменение прав доступа у файла 1.txt.

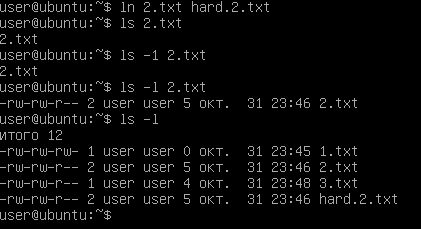


Рисунок 15 – Создание жёсткой ссылки на файл 2.txt

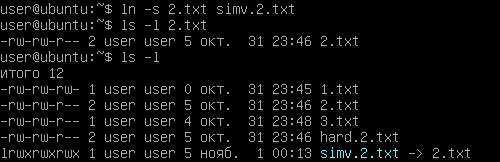


Рисунок 16 – Создание символической ссылки на файл 2.txt



Рисунок 17 – Создание директории new

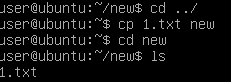


Рисунок 18 – Копирование файла 1.txt в директорию new

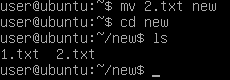


Рисунок 19 – Перемещение файла 2.txt в директорию new

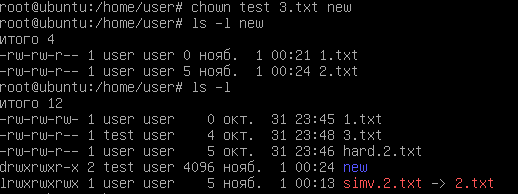


Рисунок 20 – Смена пользователя у файла 3.txt и директории new



Рисунок 21 – Удаление файла 1.txt в каталоге new



Рисунок 22 – Удаление каталога new



Рисунок 23 – Поиск файла vga2iso, файл не найден



Рисунок 24 – Поиск файла 3.txt, файл найден

Контрольные вопросы

1. Что такое файловая система?

**Файловая система**- часть операционной системы, которая обеспечивает чтение и запись файлов на дисковых носителях информации. Файловая система устанавливает физическую и логическую структуру файлов, правила их создания и управления ими, а также сопутствующие данные файла и идентификацию. Конкретная файловая система определяет размер имени файла, максимальный возможный размер файла.

2. Права доступа к файлам. Назначение прав доступа.

В индексном дескрипторе каждого файла записаны имя так называемого владельца файла и группы, которая имеет права на этот файл. Первоначально, при создании файла его владельцем объявляется тот пользователь, который этот файл создал. Точнее — тот пользователь, от чьего имени запущен процесс, создающий файл. Группа тоже назначается при создании файла — по идентификатору группы процесса, создающего файл. Владельца и группу файла можно поменять в ходе дальнейшей работы с помощью команд chown и chgrp

Право на чтение (r) файла означает, что пользователь может просматривать содержимое файла с помощью различных команд просмотра, например, командой more или с помощью любого текстового редактора. Но, подредактировав содержимое файла в текстовом редакторе, вы не сможете сохранить изменения в файле на диске, если не имеете права на запись (w) в этот файл. Право на выполнение (x) означает, что вы можете загрузить файл в память и попытаться запустить его на выполнение как исполняемую программу. Конечно, если в действительности файл не является программой (или скриптом shell), то запустить этот файл на выполнение не удастся, но, с другой стороны, даже если файл действительно является программой, но право на выполнение для него не установлено, то он тоже не запустится.

Для изменения прав доступа к файлу используется команда chmod. Ее можно использовать в двух вариантах. В первом варианте вы должны явно указать, кому какое право даете или кого этого права лишаете:

[user]$ chmod wXp имя-файла

где вместо символа w подставляется

* либо символ u (т. е. пользователь, который является владельцем);
* либо g (группа);
* либо o (все пользователи, не входящие в группу, которой принадлежит данный файл);
* либо a (все пользователи системы, т. е. и владелец, и группа, и все остальные).

Вместо X ставится:

* либо + (предоставляем право);
* либо – (лишаем соответствующего права);
* либо = (установить указанные права вместо имеющихся),

Вместо p — символ, обозначающий соответствующее право:

* r (чтение);
* w (запись);
* x (выполнение).

Вот несколько примеров использования команды chmod:

[user]$ chmod a+x file\_name

предоставляет всем пользователям системы право на выполнение данного файла.

[user]$ chmod go-rw file\_name

удаляет право на чтение и запись для всех, кроме владельца файла.

[user]$ chmod ugo+rwx file\_name

дает всем права на чтение, запись и выполнение.

3. Жесткая ссылка в Linux. Основные сведения.

Жесткие ссылки можно создавать только для файлов, но не для директорий. Исключение составляют специальные записи каталогов, указывающие на саму директорию и на ее родительскую директорию (. и ..) и представляющие собой жесткие ссылки, необходимые для подсчета числа поддиректорий. Жесткие ссылки можно использовать только в пределах одной файловой системы, поскольку они являются указателями на дескрипторы inode, которые уникальны только в пределах отдельной файловой системы. Если файл имеет несколько жестких ссылок, то он удаляется только тогда, когда удаляется последняя ссылка, указывающая на его inode, и счетчик ссылок сбрасывается до 0.

4. Команда поиска в Linux. Основные сведения.

Find - это одна из наиболее важных и часто используемых утилит системы Linux. Это команда для поиска файлов и каталогов на основе специальных условий.

Команда find имеет такой синтаксис:

find [папка] [параметры] критерий шаблон [действие]

Папка - каталог в котором будем искать

Параметры - дополнительные параметры, например, глубина поиска, и т д

Критерий - по какому критерию будем искать: имя, дата создания, права, владелец и т д.

Шаблон - непосредственно значение, по которому будем отбирать файлы.

5. Перечислите основные команды работы с каталогами.

Для просмотра содержимого каталога используется команда ls. Название этой команды является сокращением слов list files.

Просмотреть содержимое любой директории можно следующими способами:

ls имя\_директории

Либо введя команду:

ls /путь \_к \_директории

В результате выполнения команды отображаются файлы и каталоги, находящиеся внутри данного каталога: dir1 file1 file2 file3.

Чтобы показать детализированное представление (права, список владельцев файлов или папок, размер, дату последнего модифицирования) файлов и директорий в текущей директории, можно использовать флаг -l: ls –l. Для просмотра списка всех файлов, включая скрытые файлы и каталоги, вы можете добавить флаг -a: ls –a.

Для перехода в домашний каталог используется команда cd. Название этой команды является сокращением слов change directory.

Для перехода в домашний каталог пользователя user используется команда:

cd ~user.

Каждый абсолютный путь должен начинаться с косой черты - символа слэш “/”.

Для перехода в директорию уровнем выше используется команда:

cd ..

Для перехода в директорию двумя уровнями выше:

cd ../..

Создание каталогов выполняется с помощью команды вида:

mkdir имя\_директории.

Для удаления директорий используется команда rmdir имя\_директории или rm –r имя директории. Название этой команды является сокращением слов remove directory.

Переименовать файл можно с помощью команды:

mv имя\_файла новое\_имя

Для того чтобы переместить файл, используется команда:

mv имя\_файла путь/

Чтобы скопировать файл file1 и назвать его file2, используется команда:

cp file1 file2

Для того чтобы копировать директорию dir1 в директорию dir2, используется команда:

cp -a dir1/ dir2/

Скопировать файл с именем file1 в директорию с именем dir1, можно с помощью команды:

cp file1 dir1/

Вывод

Я приобрела опыт работы с файлами и каталогами в ОС Linux, ознакомилась с настройками прав на доступ к файлам и каталогам и созданием пользователей.