

Липецкий государственный технический университет

Отчет по Лабораторной работе № 1 по дисциплине «Операционная система Linux» на тему «Работа с файловой системой»

Студент

Руководитель

доцент, к.п.н.

учёная степень, учёное звание

подпись, дата

подпись, дата

Елфимова Д.А.

фамилия, инициалы

Кургасов В.В.

фамилия, инициалы

Липецк 2019 г.

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 директорию 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).

Оглавление

1. Ход работы	4
1.1. Запуск виртуальной машины и загрузка пользователем root	4
1.2. Перечень каталогов с указанием их назначения	5
1.3. Перечень файлов каталога физических устройств	8
1.4. Директорий пользователя root	9
1.5. Новый пользователь user	11
1.6. Директории пользователя root	12
1.7. Изменение прав доступа	12
1.8. Жесткая и символическая ссылки	12
1.9. Работа с каталогом new	14
1.10Поиск файлов	15
Заключение	17

1. Ход работы

1.1. Запуск виртуальной машины и загрузка пользователем root

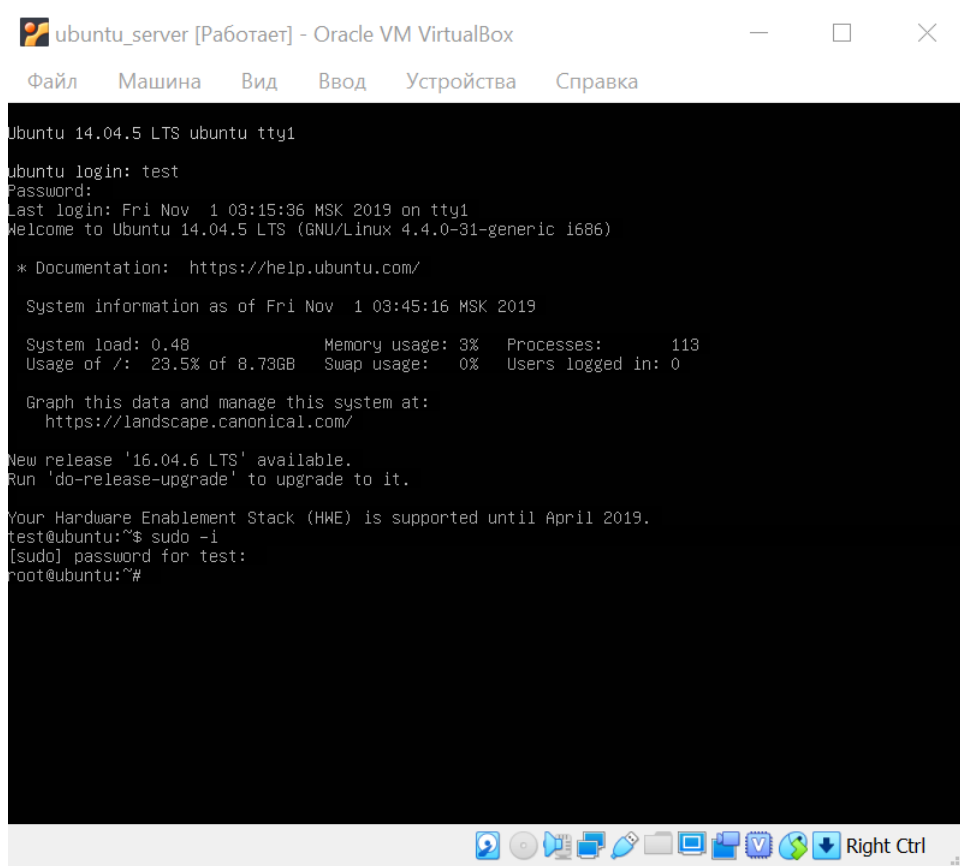



Рисунок 1. Запуск вирт.машины и загрузка пользователем root

1.2. Перечень каталогов с указанием их назначения



Имя	Размер	Время правки
/bin	4096	дек. 19 2018
/boot	4096	дек. 19 2018
/dev	4220	окт. 22 09:51
/etc	4096	окт. 22 09:51
/home	4096	февр. 6 2018
/lib	4096	апр. 10 2018
/lost+found	16384	февр. 6 2018
/media	4096	февр. 6 2018
/mnt	4096	апр. 11 2014
/opt	4096	авг. 3 2016
/proc	0	окт. 22 09:51
/root	4096	дек. 19 2018
/run	680	окт. 22 09:53
/sbin	12288	дек. 19 2018
/srv	4096	авг. 3 2016
/sys	0	окт. 22 09:51
/tmp	4096	окт. 22 09:59
/usr	4096	февр. 6 2018
/var	4096	февр. 6 2018
@initrd.img	32	февр. 6 2018
@vmlinuz	29	февр. 6 2018

Имя	Размер	Время правки
..	-BVERX-	февр. 6 2018
./cache	4096	марта 6 2018
./config	4096	марта 6 2018
./local	4096	марта 6 2018
./first	4096	окт. 22 10:02
.bash_history	2106	окт. 22 10:03
.bash_logout	220	февр. 6 2018
.bashrc	3637	февр. 6 2018
.nano.swp	12288	окт. 22 09:49
.profile	675	февр. 6 2018
.selected_editor	66	марта 12 2018
.viminfo	1063	окт. 22 09:54
1	29957	нояб. 11 2018
123	1679	дек. 7 2018
123.pub	393	дек. 7 2018
dump.sql	3018	дек. 19 2018
mc	29957	нояб. 11 2018
screenrc	3663	марта 21 2012

Рисунок 2. Каталоги

/bin - В этом каталоге хранятся основные команды, необходимые пользователю для работы в системе. Например, такие как командные оболочки и команды файловой системы (ls, cp и т.д.). Каталог /bin обычно не изменяется после установки. Если изменяется, то обычно лишь при обновлениях пакетов программ, предоставленных разработчиками операционной системы.

/boot - В этом каталоге хранятся файлы, используемые загрузчиком ОС — Linux LOader (LILO). Этот каталог так же практически не изменяется после установки.

/dev - В этом каталоге размещены описания устройств системы. В Linux всё рассматривается, как файл, даже различные устройства, такие как последовательные порты, жёсткие диски и сканеры. Для получения доступа к определённому устройству, необходимо чтобы существовал специальный файл, называемый device node. Все эти файлы находятся в каталоге /dev. Аналогично устроено большинство UNIX-подобных операционных систем.

/etc - Этот каталог содержит файлы настроек: всё, от конфигурационных файлов системы X Window, базы данных пользователей и до стартовых сценариев.

/home - В этом каталоге размещены домашние каталоги пользователей. Linux является многопользовательской системой и каждому пользователю присваивается имя и уникальный каталог для персональных файлов. Этот каталог называется "home"(домашним) каталогом пользователя.

/lib - В этом каталоге находятся системные библиотеки, необходимые для основных программ: библиотека C, динамический загрузчик, библиотека ncurses, модули ядра и другое.

/lib/modules - Подгружаемые модули для ядра (например, сетевые драйверы или поддержка дополнительных файловых систем).

/lost+found - В этом каталоге сохраняются восстановленные части файловой системы. При загрузке системы происходит проверка файловых систем на наличие ошибок. Для исправления ошибок файловой системы запускается программа fsck.

/mnt - Этот каталог предоставляется как временная точка монтирования для жёстких дисков, дискет, компакт-дисков или отключаемых устройств.

/media Директория, используемая для автоматического монтирования различных устройств USB-накопителей, CD-ROM и т.д. При вставке в привод CD-ROM диска он автоматически подключается во вложенный каталог этой директории.

/opt - В этом каталоге размещаются дополнительные пакеты программ. Особенность Linux в том, что все пакеты программ, устанавливаются в этот каталог, например /opt/<программный пакет>. В последствии если этот пакет больше не будет нужен, то достаточно всего лишь удалить соответствующий каталог. В дистрибутивах SlackWare некоторые программы изначально поставляются в каталоге /opt (например, KDE - в /opt/kde).

/proc - Это специальный каталог не входящий в файловую систему. Каталог /proc представляет собой виртуальную файловую систему, которая предоставляет доступ к информации ядра. Различная информа-

ция, которую ядро может сообщить пользователям, находится в "файлах" каталога `/proc`. Например, в файле `/proc/modules` находится список загруженных модулей ядра. А в файле `/proc/cpuinfo` — информация о процессоре компьютера.

`/root` - Это домашний каталог администратора, вместо `/home/root`. Это потому, что каталог `/home` может находиться в разделе, отличном от корневого (`/`) и если по какой-то причине `/home` не может быть подключён, то пользователь `root` вынужден будет войти в систему, чтобы решить проблему. И если его домашний каталог на другом диске, то это усложнит вход в систему.

`/run` Каталог, содержащий PID файлы процессов. Размещен в `TMPSFS`, а поэтому после перезагрузки все файлы теряются.

`/sbin` - В этом каталоге хранятся основные программы, выполняемые пользователем `root` а так же программы, выполняемые в процессе загрузки. Обычные пользователи не могут пользоваться этими программами.

`/srv` Параметры, которые специфичные для окружения системы. Чаще всего данная директория пуста.

`/sys` — это директория, к которой примонтирована виртуальная файловая система `sysfs`, которая добавляет в пространство пользователя информацию ядра Linux о присутствующих в системе устройствах и драйверах. В версии ядра ниже 2.6 не использовалась.

`/tmp` - Временное хранилище данных. Все пользователи имеют права чтения и записи в этом каталоге.

`/usr` - Это один из самых больших каталогов в системе. Практически всё остальное расположено здесь. Программы, документация, исходный код ядра и система X Window. Именно в этот каталог, чаще всего, устанавливаются программы.

`/var` - В этом каталоге хранятся системные лог-файлы, кэш-файлы и файлы-замки программ. Это каталог для часто меняющихся данных.

1.3. Перечень файлов каталога физических устройств

```
test@ubuntu:~$ ls /dev
autofs      log          ram11        sr0           tty28        tty51        ttyS16       urandom
block       loop0        ram12        stderr        tty29        tty52        ttyS17       userio
bsg         loop1        ram13        stdin         tty3         tty53        ttyS18       vboxguest
btrfs-control loop2        ram14        stdout        tty30        tty54        ttyS19       vboxuser
bus         loop3        ram15        tty           tty31        tty55        ttyS2         vcs
cdrom       loop4        ram2         tty0          tty32        tty56        ttyS20       vcs1
char        loop5        ram3         tty1          tty33        tty57        ttyS21       vcs2
console     loop6        ram4         tty10         tty34        tty58        ttyS22       vcs3
core        loop7        ram5         tty11         tty35        tty59        ttyS23       vcs4
cpu         loop-control ram6         tty12         tty36        tty6         ttyS24       vcs5
cpu_dma_latency mapper       ram7         tty13         tty37        tty60        ttyS25       vcs6
cuse        mcelog       ram8         tty14         tty38        tty61        ttyS26       vcs7
disk        mem          ram9         tty15         tty39        tty62        ttyS27       vcsa
dri         memory_bandwidth random       tty16         tty4         tty63        ttyS28       vcsa1
ecryptfs    net          rfc0         tty17         tty40        tty7         ttyS29       vcsa2
fb0         network_latency rtc          tty18         tty41        tty8         ttyS3         vcsa3
fd          network_throughput rtd0         tty19         tty42        tty9         ttyS30       vcsa4
full        null         sda          tty2          tty43        ttyprintk    ttyS31       vcsa5
fuse        port         sda1         tty20         tty44        ttyS0        ttyS4        vcsa6
hidraw0     ppp          sda2         tty21         tty45        ttyS1        ttyS5        vcsa7
hpet        psaux        sda5         tty22         tty46        ttyS10       ttyS6        vflo
hwrng       ptmx         sg0          tty23         tty47        ttyS11       ttyS7        vga_arbiter
l2c-0       pts          sg1          tty24         tty48        ttyS12       ttyS8        vhci
input       ram0         shm          tty25         tty49        ttyS13       ttyS9        vhost-net
kmsg        ram1         snapshot     tty26         tty5         ttyS14       uhid         zero
lightnvm    ram10        snd          tty27         tty50        ttyS15       uinput
```

Рисунок 3. Каталог физических устройств

Все подключаемые к компьютеру устройства (жесткие и съемные диски, терминал, принтер, модем и т. д.), представляются файлами. Если, например, надо вывести на экран какую-то информацию, то система как бы производит запись в файл `/dev/tty01`.

Описание основных специальных файлов

`/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

`/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

`/dev/cua` Специальные устройства для работы с модемами

`/dev/null` 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

```

root@ubuntu:/# ls -l
итого 80
drwxr-xr-x  2 root root 4096 дек. 19 2018 bin
drwxr-xr-x  3 root root 4096 дек. 19 2018 boot
drwxr-xr-x 17 root root 4220 окт. 22 09:51 dev
drwxr-xr-x 100 root root 4096 окт. 22 09:51 etc
drwxr-xr-x  3 root root 4096 февр. 6 2018 home
drwxrwxrwx  1 root root  32 февр. 6 2018 initrd.img -> boot/initrd.img-4.4.0-31-generic
drwxr-xr-x 22 root root 4096 апр. 10 2018 lib
drwx----- 2 root root 16384 февр. 6 2018 lost+found
drwxr-xr-x  3 root root 4096 февр. 6 2018 media
drwxr-xr-x  2 root root 4096 апр. 11 2014 mnt
drwxr-xr-x  2 root root 4096 авг. 3 2016 opt
dr-xr-xr-x 111 root root  0 окт. 22 09:51 proc
drwx----- 5 root root 4096 дек. 19 2018 root
drwxr-xr-x 20 root root 680 окт. 22 09:53 run
drwxr-xr-x  2 root root 12288 дек. 19 2018 sbin
drwxr-xr-x  2 root root 4096 авг. 3 2016 srv
dr-xr-xr-x 13 root root  0 окт. 22 09:51 sys
drwxrwxrwt  5 root root 4096 окт. 22 10:17 tmp
drwxr-xr-x 10 root root 4096 февр. 6 2018 usr
drwxr-xr-x 12 root root 4096 февр. 6 2018 var
drwxrwxrwx  1 root root  29 февр. 6 2018 vmlinuz -> boot/vmlinuz-4.4.0-31-generic

```

Рисунок 6. Права доступа к файлу vmlinuz

Информация о правах доступа:

- — — - нет прав;
- — *x* - разрешено только выполнение файла, как программы, но не изменение и не чтение;
- *w* — - разрешена только запись и изменение файла;
- *wx* - разрешено изменение и выполнение, но в случае с каталогом, вы не можете посмотреть его содержимое;
- r* — — - права только на чтение;
- r* — *x* - только чтение и выполнение, без права на запись;
- rw* — - права на чтение и запись, но без выполнения;
- rw* *x* - все права;
- — *s* - установлен SUID или SGID бит, первый отображается в поле для владельца, второй для группы;
- — *t* - установлен sticky-bit, а значит пользователи не могут удалить этот файл.

В данном случае владельцем файла является пользователь root и группа root. Права доступа представлены цепочкой символов `lrwxrwxrwx`. Эти символы можно условно разделить на 4 группы. Первая группа, состоящая из единственного символа, определяет тип файла. Этот символ может принимать такие значения:

- - обычный файл;
- *d* = - каталог;
- *b* = - файл блочного устройства;
- *c* = - файл символьного устройства;

- s = - доменное гнездо (socket);
- p = - именованный канал (pipe);
- l = - символическая ссылка (link).

Далее следуют три группы по три символа, которые и определяют права доступа к файлу соответственно для владельца файла, для группы пользователей, которая сопоставлена данному файлу, и для всех остальных пользователей системы.

1.5. Новый пользователь user

```
root@ubuntu:~# sudo adduser user
Добавляется пользователь «user» ...
Добавляется новая группа «user» (1002) ...
Добавляется новый пользователь «user» (1002) в группу «user» ...
Домашний каталог «/home/user» уже существует. Не копируется из «/etc/skel».
Введите новый пароль UNIX:
Повторите ввод нового пароля UNIX:
passwd: password updated successfully
Changing the user information for user
Enter the new value, or press ENTER for the default
  Full Name []:
  Room Number []:
  Work Phone []:
  Home Phone []:
  Other []:
Данная информация корректна? [Y/n] y
```

Рисунок 7. Создание пользователя user

```
root@ubuntu:~# cat /home/user/1.txt
Hello, linux!
root@ubuntu:~# nano /home/user/2.txt_
```

Рисунок 8. Создание файлов 1.txt и 2.txt

```
root@ubuntu:/home/user# touch -t 200001010000.00 3.txt
root@ubuntu:/home/user# stat 3.txt
  файл: «3.txt»
  Размер: 0          Блоков: 0          Блок В/В: 4096   пустой обычный файл
Устройство: 801h/2049d  Inode: 407814      Ссылки: 1
Доступ: (0644/-rw-r--r--)  Uid: (   0/   root)   Gid: (   0/   root)
Доступ: 2000-01-01 00:00:00.000000000 +0300
Модифицирован: 2000-01-01 00:00:00.000000000 +0300
Изменён: 2019-10-22 12:03:15.857639082 +0300
Создан: -
```

Рисунок 9. Создание файла 3.txt

```
root@ubuntu:/home/user# ls -l
итого 8
-rw-r--r-- 1 root root 14 окт. 22 11:53 1.txt
-rw-r--r-- 1 root root 25 окт. 22 11:58 2.txt
-rw-r--r-- 1 root root 0 янв.  1 2000 3.txt
```

Рисунок 10. Права доступа к файлам

1.6. Директории пользователя root

```
root@ubuntu:/home# cd /root
root@ubuntu:~# cd /root
root@ubuntu:~# su test
test@ubuntu:/root$ cd /home
test@ubuntu:/home$ cd /root
bash: cd: /root: Отказано в доступе
test@ubuntu:/home$
```

Рисунок 11. Результат

1.7. Изменение прав доступа

```
root@ubuntu:/home/user# ls -l 1.txt
-rw-r--r-- 1 root root 14 окт. 22 11:53 1.txt
root@ubuntu:/home/user# chmod 777 1.txt
root@ubuntu:/home/user# ls -l 1.txt
-rwxrwxrwx 1 root root 14 окт. 22 11:53 1.txt
```

Рисунок 12. Изменение прав доступа

Каждый номер в этой команде представляет собой права для одного из типов пользователей (владелец, группа и другие). Возьмем номер 7. Теперь, используя объяснение числовых значений ($r=4, w=2, x=1$), единственный вариант для получения числа 7 – сложение чисел 4, 2 и 1, получаем $4+2+1=7$. Проще говоря, это означает ВСЕ права доступа (чтение, запись, выполнение – rwX). Первый номер устанавливает права доступа для владельца файла. Второй номер устанавливает права доступа для группы владельца. Третий номер дает группе другие. Третья часть в команде – это имя файла для которого мы изменяем права доступа.

1.8. Жесткая и символическая ссылки

Символические ссылки

Символические ссылки более всего похожи на обычные ярлыки. Они содержат адрес нужного файла в вашей файловой системе. Когда вы пытаетесь открыть такую ссылку, то открывается целевой файл или папка. Главное ее отличие от жестких ссылок в том, что при удалении целевого файла ссылка останется, но она будет указывать в никуда, поскольку файла на самом деле больше нет. Вот основные особенности символических ссылок:

Могут ссылаться на файлы и каталоги;

После удаления, перемещения или переименования файла становятся недействительными;

Права доступа и номер inode отличаются от исходного файла;
При изменении прав доступа для исходного файла, права на ссылку останутся неизменными;
Можно ссылаться на другие разделы диска;
Содержат только имя файла, а не его содержимое.

Жесткие ссылки

Этот тип ссылок реализован на более низком уровне файловой системы. Файл размещен только в определенном месте жесткого диска. Но на это место могут ссылаться несколько ссылок из файловой системы. Каждая из ссылок - это отдельный файл, но ведут они к одному участку жесткого диска. Файл можно перемещать между каталогами, и все ссылки останутся рабочими, поскольку для них неважно имя. Рассмотрим особенности: Работают только в пределах одной файловой системы;

Нельзя ссылаться на каталоги;

Имеют ту же информацию inode и набор разрешений что и у исходного файла;

Разрешения на ссылку изменяться при изменении разрешений файла;

Можно перемещать и переименовывать и даже удалять файл без вреда ссылке.

Если удалить исходный файл, а потом запустить ссылку, то вы получите ошибку, что такого файла не существует, потому что мы действительно удалили исходный файл. Если вы удалите ссылку, то исходный файл останется на месте.

Для проверки некоторых свойств создадим отдельный файл, для которого будут созданы ссылки, после чего файл удалим.

```

root@ubuntu:/home/user# cd /home/user/ttest
root@ubuntu:/home/user/ttest# echo "Ссылки" source
Ссылки source
root@ubuntu:/home/user/ttest# cat source
cat: source: Нет такого файла или каталога
root@ubuntu:/home/user/ttest# echo "Ссылки" > source
root@ubuntu:/home/user/ttest# cat source
Ссылки
root@ubuntu:/home/user/ttest# cd ..
root@ubuntu:/home/user# ln -s /home/user/ttest/source softlink
root@ubuntu:/home/user# cat softlink
Ссылки
root@ubuntu:/home/user# ls -li
итого 12
407797 -rwxrwxrwx 1 root root 14 окт. 22 11:53 1.txt
407812 -rwxrwxrwx 1 root root 25 окт. 22 11:58 2.txt
407814 -rwxrwxrwx 1 root root 0 янв. 1 2000 3.txt
407816 lrwxrwxrwx 1 root root 23 окт. 22 12:21 softlink -> /home/user/ttest/source
417835 drwxr-xr-x 2 root root 4096 окт. 22 12:19 ttest

```

Рисунок 13. Символическая ссылка

```

root@ubuntu:/home/user# ln ttest/source hardlink
root@ubuntu:/home/user# cat hardlink
Ссылки
root@ubuntu:/home/user# ls -li
итого 16
407797 -rwxrwxrwx 1 root root 14 окт. 22 11:53 1.txt
407812 -rwxrwxrwx 1 root root 25 окт. 22 11:58 2.txt
407814 -rwxrwxrwx 1 root root 0 янв. 1 2000 3.txt
407815 -rw-r--r-- 2 root root 13 окт. 22 12:19 hardlink
407816 lrwxrwxrwx 1 root root 23 окт. 22 12:21 softlink -> /home/user/ttest/source
417835 drwxr-xr-x 2 root root 4096 окт. 22 12:19 ttest

```

Рисунок 14. Жесткая ссылка

```

root@ubuntu:/home/user# rm /home/user/ttest/source
root@ubuntu:/home/user# ls -li
итого 16
407797 -rwxrwxrwx 1 root root 14 окт. 22 11:53 1.txt
407812 -rwxrwxrwx 1 root root 25 окт. 22 11:58 2.txt
407814 -rwxrwxrwx 1 root root 0 янв. 1 2000 3.txt
407815 -rw-r--r-- 1 root root 13 окт. 22 12:19 hardlink
407816 lrwxrwxrwx 1 root root 23 окт. 22 12:21 softlink -> /home/user/ttest/source
417835 drwxr-xr-x 2 root root 4096 окт. 22 12:27 ttest
root@ubuntu:/home/user# cat hardlink
Ссылки
root@ubuntu:/home/user# cat softlink
cat: softlink: Нет такого файла или каталога

```

Рисунок 15. Удаление файла

1.9. Работа с каталогом new

```

root@ubuntu:~# mkdir /home/user/new
root@ubuntu:~# cd /home/user
root@ubuntu:/home/user# ls -li
итого 16
407817 -rwxr-xr-x 1 root root 14 окт. 22 12:35 1.txt
407814 -rwxrwxrwx 1 test root 0 янв. 1 2000 3.txt
407815 -rw-r--r-- 1 root root 13 окт. 22 12:19 hardlink
417836 drwxr-xr-x 2 root root 4096 нояб. 1 04:48 new
407816 lrwxrwxrwx 1 root root 23 окт. 22 12:21 softlink -> /home/user/ttest/source
417835 drwxr-xr-x 2 root root 4096 окт. 22 12:27 ttest

```

Рисунок 16. Создать каталог new в каталоге пользователя user

```

root@ubuntu:/home/user# cp 1.txt /home/user/new
root@ubuntu:/home/user# mv 2.txt /home/user/new
root@ubuntu:/home/user# ls -li
итого 16
407817 -rwxr-xr-x 1 root root 14 окт. 22 12:35 1.txt
407814 -rwxrwxrwx 1 root root 0 янв. 1 2000 3.txt
407815 -rw-r--r-- 1 root root 13 окт. 22 12:19 hardlink
417836 drwxr-xr-x 2 root root 4096 окт. 22 12:38 new
407816 lrwxrwxrwx 1 root root 23 окт. 22 12:21 softlink -> /home/user/ttest/source
417835 drwxr-xr-x 2 root root 4096 окт. 22 12:27 ttest
root@ubuntu:/home/user# cd new
root@ubuntu:/home/user/new# ls -li
итого 8
407797 -rwxr-xr-x 1 root root 14 окт. 22 12:38 1.txt
407812 -rwxrwxrwx 1 root root 25 окт. 22 11:58 2.txt

```

Рисунок 17. Скопировать файл 1.txt в каталог new. Переместить файл 2.txt в каталог new

```

root@ubuntu:/home/user# chown test new
root@ubuntu:/home/user# chown test 3.txt
root@ubuntu:/home/user# chown test new
root@ubuntu:/home/user# ls -li
итого 16
407817 -rwxr-xr-x 1 root root 14 окт. 22 12:35 1.txt
407814 -rwxrwxrwx 1 test root 0 янв. 1 2000 3.txt
407815 -rw-r--r-- 1 root root 13 окт. 22 12:19 hardlink
417836 drwxr-xr-x 2 test root 4096 окт. 22 12:38 new
407816 lrwxrwxrwx 1 root root 23 окт. 22 12:21 softlink -> /home/user/ttest/source
417835 drwxr-xr-x 2 root root 4096 окт. 22 12:27 ttest

```

Рисунок 18. Изменить владельца файла 3.txt и каталога new.

```

test@ubuntu:/home/user$ cd /home/user/new
test@ubuntu:/home/user/new$ rm 1.txt
rm: удалить защищенный от записи обычный файл «1.txt»? y
test@ubuntu:/home/user/new$ ls -li
итого 4
407812 -rwxrwxrwx 1 root root 25 окт. 22 11:58 2.txt

```

Рисунок 19. Удалить файл 1.txt в каталоге new.

```

root@ubuntu:~# rm -r /home/user/new
root@ubuntu:~# ls -li /home/user
итого 12
407817 -rwxr-xr-x 1 root root 14 окт. 22 12:35 1.txt
407814 -rwxrwxrwx 1 test root 0 янв. 1 2000 3.txt
407815 -rw-r--r-- 1 root root 13 окт. 22 12:19 hardlink
407816 lrwxrwxrwx 1 root root 23 окт. 22 12:21 softlink -> /home/user/ttest/source
417835 drwxr-xr-x 2 root root 4096 окт. 22 12:27 ttest
root@ubuntu:~# _

```

Рисунок 20. Удалить каталог new.

1.10. Поиск файлов

Найти, используя команду `find`, файл `vga2iso` не получилось, поэтому был проведен поиск по всем файлам, которые начинаются на "vga"

```

root@ubuntu:~# find / -name "vga*"
/sys/devices/virtual/misc/vga_arbiter
/sys/class/misc/vga_arbiter
/dev/vga_arbiter
/usr/src/linux-headers-4.4.0-31/include/linux/vgaarb.h
/usr/src/linux-headers-4.4.0-31/include/linux/vga_switcheroo.h
/usr/src/linux-headers-4.4.0-31/include/asm-generic/vga.h
/usr/src/linux-headers-4.4.0-31/include/video/vga.h
/usr/src/linux-headers-4.4.0-31/drivers/gpu/vga
/usr/src/linux-headers-4.4.0-31/arch/ia64/include/asm/vga.h
/usr/src/linux-headers-4.4.0-31/arch/x86/include/asm/vga.h
/usr/src/linux-headers-4.4.0-31/arch/arm/include/asm/vga.h
/usr/src/linux-headers-4.4.0-31/arch/sparc/include/asm/vga.h
/usr/src/linux-headers-4.4.0-31/arch/m32r/include/asm/vga.h
/usr/src/linux-headers-4.4.0-31/arch/microblaze/include/asm/vga.h
/usr/src/linux-headers-4.4.0-31/arch/mn10300/include/asm/vga.h
/usr/src/linux-headers-4.4.0-31/arch/sh/include/asm/vga.h
/usr/src/linux-headers-4.4.0-31/arch/tile/include/asm/vga.h
/usr/src/linux-headers-4.4.0-31/arch/s390/include/asm/vga.h
/usr/src/linux-headers-4.4.0-31/arch/powerpc/include/asm/vga.h
/usr/src/linux-headers-4.4.0-31/arch/frv/include/asm/vga.h
/usr/src/linux-headers-4.4.0-31/arch/alpha/include/asm/vga.h
/usr/src/linux-headers-4.4.0-31/arch/xtensa/include/asm/vga.h
/usr/src/linux-headers-4.4.0-31/arch/mips/include/asm/vga.h
/usr/src/linux-headers-4.4.0-31/arch/m68k/include/asm/vga.h
/usr/src/linux-headers-4.4.0-31-generic/include/config/fb/vga16.h
/usr/src/linux-headers-4.4.0-31-generic/include/config/vga
/usr/src/linux-headers-4.4.0-31-generic/include/config/vgastate.h
/usr/src/linux-headers-4.4.0-31-generic/include/config/vfio/pci/vga.h
/usr/lib/grub/i386-pc/vga.mod
/usr/lib/grub/i386-pc/vga_text.mod
/boot/grub/i386-pc/vga.mod
/boot/grub/i386-pc/vga_text.mod
/lib/i386-linux-gnu/plymouth/renderers/vga16fb.so
/lib/modules/4.4.0-31-generic/kernel/drivers/video/fbdev/vga16fb.ko
/lib/modules/4.4.0-31-generic/kernel/drivers/video/vgastate.ko

```

Рисунок 21. Поиск файлов

Заключение

В ходе данной лабораторной работы были приобретены навыки использования виртуальной машины, а также освоены основные моменты работы в терминале ОС Linux, изучены способы работы с файлами и каталогами и настройки прав на доступ к файлам и каталогам.