

**Липецкий государственный технический университет**

**Факультет автоматизации и информатики**

**Кафедра Автоматизированных систем управления**

**ЛАБОРАТОРНАЯ РАБОТА №2**

**по дисциплине «Операционная система Linux»**

**Работа с файловой системой ОС Linux**

Студентка

Фетисов В. Д.

Группа ПИ-19

Руководитель

Кургасов В. В.

Доцент и к. н.

Липецк 2021

## Оглавление

Цель работы .....	3
Задание кафедры .....	4
Ход работы.....	6
Вывод .....	26
Контрольные вопросы .....	27

## Цель работы

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

## Задание кафедры

1. Запустить виртуальную машину 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).

## Ход работы

Запустим виртуальную машину Linux Ubuntu и загрузимся пользователем root при помощи команды `sudo su`. Пример выполнения работы представлен на рисунке 1.

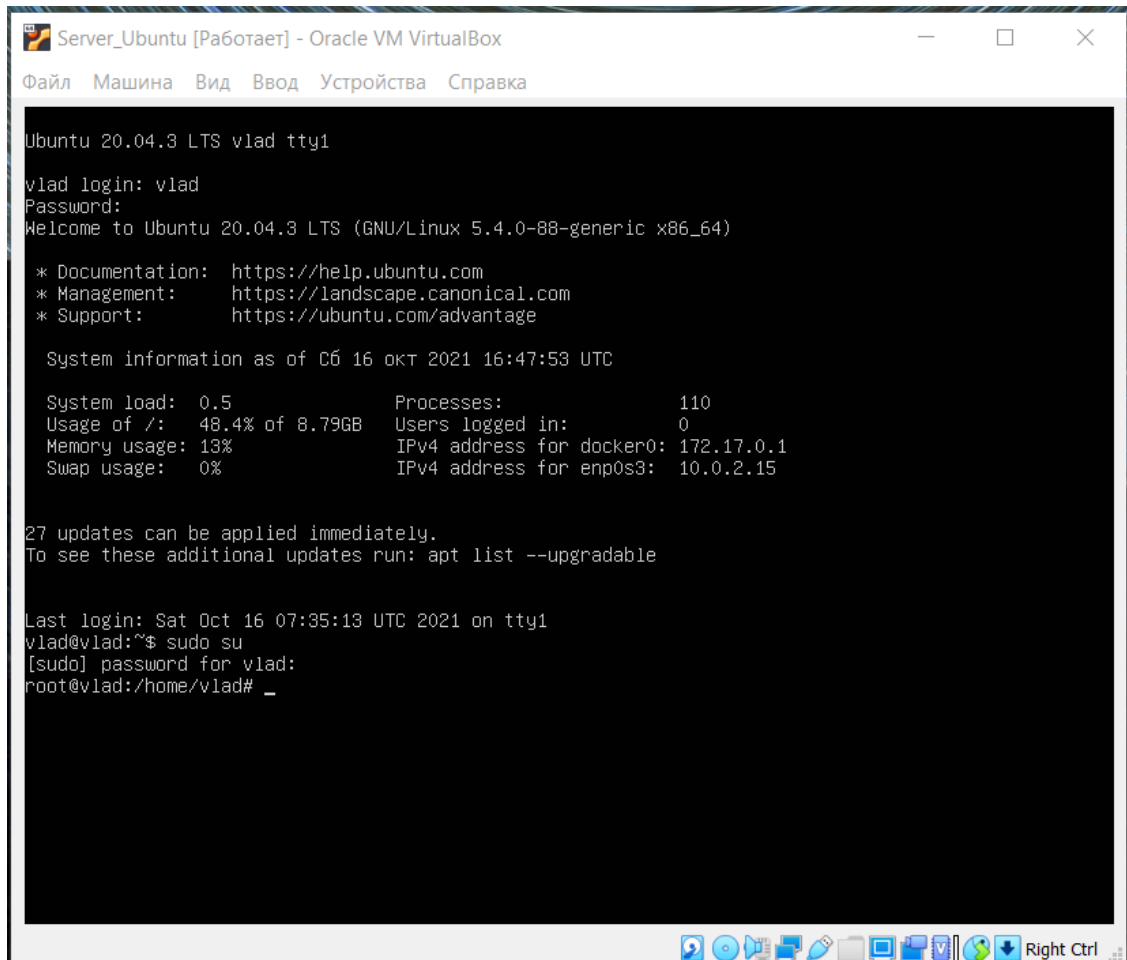
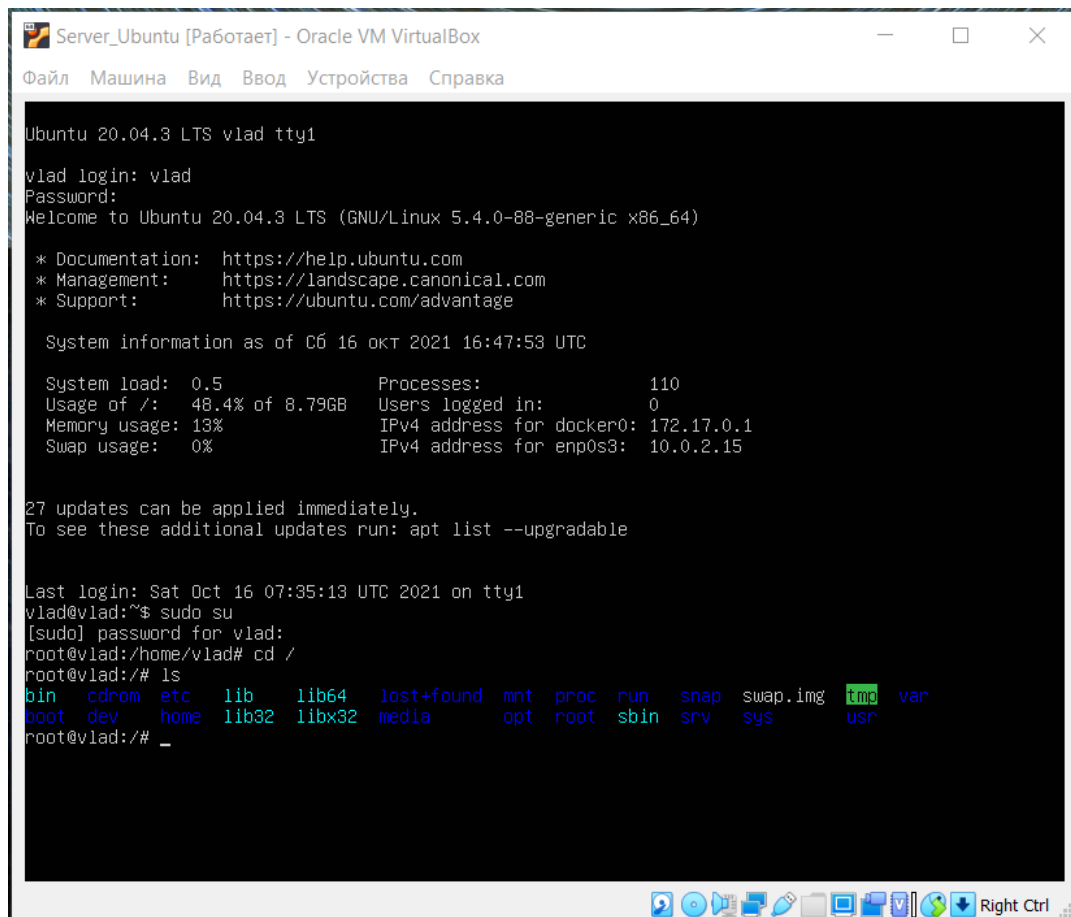


Рисунок 1 – Загрузка пользователем root (`sudo su`)

Посмотрим содержание корневой директории с помощью команды `ls` и опишем каждый из этих каталогов. Пример выполнения работы представлен на рисунке 2.



```
Server_Ubuntu [Работает] - Oracle VM VirtualBox
Файл  Машина  Вид  Ввод  Устройства  Справка

Ubuntu 20.04.3 LTS vlad tty1
vlad login: vlad
Password:
Welcome to Ubuntu 20.04.3 LTS (GNU/Linux 5.4.0-88-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

System information as of Сб 16 окт 2021 16:47:53 UTC

System load:  0.5               Processes:           110
Usage of /:   48.4% of 8.79GB   Users logged in:    0
Memory usage: 13%              IPv4 address for docker0: 172.17.0.1
Swap usage:   0%               IPv4 address for enp0s3: 10.0.2.15

27 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

Last login: Sat Oct 16 07:35:13 UTC 2021 on tty1
vlad@vlad:~$ sudo su
[sudo] password for vlad:
root@vlad:/home/vlad# cd /
root@vlad:/# ls
bin  cdrom  etc  lib  lib64  lost+found  mnt  proc  run  snap  swap.img  tmp  var
boot  dev  home  lib32  libx32  media  opt  root /sbin  srv  sys  usr
```

Рисунок 2 – Содержание корневой директории

## Описание каталогов

1. `/bin` – содержит основные утилиты, необходимые как в однопользовательском режиме, так и при обычной работе всем пользователям (пример: `cat`, `ls`, `cp`, `tail`, `ps`), исполняемые файлы, а также символичные ссылки на исполняемые файлы;
2. `/boot` – это каталог в котором находятся файлы, необходимые для загрузки системы такие как GRUB и ядра Linux. Здесь нет конфигурационных файлов, используемых загрузчиком - они находятся в каталоге `/etc` вместе с другими конфигурационными файлами. В `/boot` хранятся данные, которые используются до того, как ядро начинает исполнять программы пользователя;

3. /cdrom – это временное место, где монтируются диски CD-ROM, когда они вставляются в компьютер. Однако, стандартное место для подключаемого носителя находится внутри каталога /media;
4. /dev – содержит файлы устройств;
5. /etc – содержит конфигурационные файлы операционной системы и всех сетевых служб;
6. /home – содержит домашние каталоги всех пользователей, зарегистрированных в системе;
7. /lib – содержит основные библиотеки и модули ядра, необходимые для работы программ из /bin и /sbin;
8. /lib 64 – обычно это используется для поддержки 64-битного или 32-битного формата в системах, поддерживающих несколько форматов исполняемых файлов, и требующих библиотек с одним и тем же названием. В этом случае /lib32 и /lib64 могут быть библиотечными каталогами, а /lib - символической ссылкой на один из них;
9. /lost+found – этот каталог нужен для хранения испорченных файлов при проблемах с файловой системой, которые были восстановлены после, например, некорректного размонтирования файловой системы;
10. /media – данный каталог содержит - точки монтирования для сменных носителей, таких как CD-ROM, DVD-ROM (впервые описано в FHS2.3);
11. /mnt – точки монтирования. В современных дистрибутивах Linux этот процесс обычно происходит автоматически. При этом в каталогах /mnt или /media создается подкаталог, имя которого совпадает с именем монтируемого тома;
12. /opt – дополнительное программное обеспечение, здесь обычно размещаются установленные программы, имеющие большой дисковый объем, или вспомогательные пакеты;



13. /proc – каталог псевдофайловой системы procfs, которая используется для предоставления информации о процессах (по-другому это виртуальная файловая система, которая обеспечивает связь с ядром и монтируется в каталогу /proc). Он существует только во время работы системы в оперативной памяти компьютера. Каталог представляет интерес и с точки зрения безопасности;

14. /root – каталог пользователя root;

15. /run – хранение данных, которые были запущены приложениями;

16. /sbin – набор утилит для системного администрирования, содержит исполняемые файлы, необходимые для загрузки системы и ее восстановления в различных щекотливых ситуациях. Запускать эти утилиты имеет право только root.;

17. /snap – по умолчанию является местом, где файлы и папки из установленных пакетов snap появляются в вашей системе;

18. /srv – параметры, которые специфичные для окружения системы, чаще всего данная директория пуста;

19. /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;

20. /tmp – временные файлы. Linux, регулярно очищает этот каталог;

21. /usr – в этом каталоге хранятся все установленные пакеты программ, документация, исходный код ядра и система X Window. Все пользователи кроме суперпользователя root имеют доступ только для чтения. Может быть смонтирована по сети и может быть общей для нескольких машин;

22. /var – переменные файлы (variable), которые подвергаются наиболее частому изменению. Например, кэши различных программ; файлы блокировки для недопустимости одновременного использования одной



3. `btrfs-control` – устройство принимает некоторые вызовы `ioctl`, которые могут выполнять следующие действия с модулем файловой системы: сканирование устройства на наличие файловой системы `btrfs` (т.е. позволить файловым системам с несколькими устройствами монтировать автоматически) и регистрировать их в модуле ядра, аналогично сканированию, но также дождаться завершения процесса сканирования устройства для данной файловой системы, получение поддерживаемые функции;
4. `console` – текстовый терминал и виртуальные консоли;
5. `cpu_dma_latency` – часть интерфейса качества и обслуживания в ядре Linux;
6. `cuse` – реализация символьных устройств (`char devices`) в Linux
7. `drm_dp_aux` – канал DisplayPort AUX;
8. `ecryptfs` – POSIX- совместимая многоуровневая криптографическая файловая система в ядре Linux;
9. `fb` – устройство обеспечивает абстракцию для графического оборудования;
10. `freefall` – это решение для управления брандмауэром для многих дистрибутивов Linux, включая Ubuntu, Debian, CentOS, RHEL и Fedora;
11. `fuse` – (`filesystem in userspace` — «файловая система в пользовательском пространстве») — свободный модуль для ядер Unix подобных операционных систем, позволяет разработчикам создавать новые типы файловых систем, доступные для монтирования пользователями без привилегий (прежде всего — виртуальных файловых систем);
12. `hpet` – тип таймера, используемый в персональных компьютерах
13. `hwrng` – генератор случайных чисел;

14. i2c – шина предлагает различные преимущества, такие как экономия места на плате, уменьшение общей стоимости оборудования, а также предлагает средства упрощённой отладки;
15. kmsg – узел символьного устройства обеспечивает доступ пользователя к буферу printk ядра;
16. kvm – программное решение, обеспечивающее виртуализацию в среде Linux на платформе x86;
17. loop – это блочное устройство, которое отображает блоки данных обычного файла в файловой системе или другое блочное устройство;
18. loop-control – начиная с Linux 3.1, ядро предоставляет устройство dev/loop-control, которое позволяет приложению динамически находить свободное устройство, а также добавлять и удалять устройства loop из системы;
19. mcelog – программа mcelog декодирует машинные события (аппаратных ошибок) на x86-64, работающих под управлением 64-разрядной Linux
20. mei – это изолированный и защищенный вычислительный ресурс (сопроцессор), находящийся внутри определенных наборов микросхем Intel;
21. mem – это символьный файл устройства, в котором отображается главная память компьютера. Он может использоваться, например, для проверки (и даже исправления) системы;
22. null – специальный файл в системах класса UNIX, представляющий собой так называемое «пустое устройство»;
23. nvram – она же энергонезависимая память, применяется в современных UEFI BIOS, в отличие от старых BIOS, где для хранения использовали CMOS SRAM + батарейка;
24. port – символьное устройство для чтения и / или записи;

25. ppp – это механизм для создания и запуска IP (Internet Protocol) и других сетевых протоколов;
26. psaux – устройство мыши PS / 2;
27. ptmx – является символьным файлом с основным номером, равным 5 и вторичным номером 2, обычно имеет права доступа 0666, владелец и группа равны root. Используется для создания пары основного и подчиненного псевдотерминала;
28. random – специальные символьные псевдоустройства в некоторых UNIX-подобных системах, впервые появившиеся в ядре Linux версии ;
29. rfkill – это подсистема в ядре Linux, предоставляющая интерфейс, через который можно запрашивать, активировать и деактивировать радиопередатчики в компьютерной системе.
30. rtc – часы реального времени;
31. sda – первый жесткий диск;
32. sda – N-ый раздел первого жесткого диска;
33. sdb – второй жесткий диск;
34. sdb – N-ый раздел второго жесткого диска;
35. sg – SCSI Generic driver используется, среди прочего, для сканеров, устройств записи компакт-дисков и чтения аудио-компакт-дисков в цифровом формате;
36. snapshot – поддержка снимков устройства;
37. tpm – разрешает доступ к устройству Trusted Platform Module (tpm);
38. tty – виртуальная консоль;
39. ttyprintk – драйвер псевдо TTY, который позволяет пользователям создавать сообщения printk через вывод на устройство ttyprintk;

- 40. uhid – поддержка драйвера ввода-вывода пользовательского пространства для подсистемы HID;
- 41. uinput – поддержка драйвера уровня пользователя для ввода;
- 42. urandom – более быстрая и менее безопасная генерация случайных чисел;
- 43. userio – призван упростить жизнь разработчикам драйверов ввода, позволяя им тестировать различные устройства Serio (в основном, различные сенсорные панели на ноутбуках), не имея физического устройства перед ними;
- 44. vcs – текущее текстовое содержимое виртуальной консоли;
- 45. vcsa – текущее содержимое текстового атрибута виртуальной консоли;
- 46. vcsu – текущее текстовое содержимое виртуальной консоли (юникод);
- 47. vga\_arbiter – сканирует все устройства PCI и добавляет в арбитраж VGA. Затем арбитр включает / отключает декодирование на разных устройствах устаревших инструкций VGA;
- 48. vhci – виртуальный драйвер HCI Bluetooth;
- 49. vhost-net – ускоритель ядра хоста для virtio net;
- 50. vhost-vsock – программное устройство, поэтому нет пробного вызова, который вызывает драйвер, чтобы зарегистрировать его узел устройства misc char. Это создает проблема с курицей и яйцом: приложения в пользовательском пространстве должны открываться / dev / vhost-vsock, чтобы использовать драйвер, но файл не существует, пока модуль ядра загружен;
- 51. video – устройство видеозахвата / наложения;
- 52. zero – специальный файл в UNIX-подобных системах, представляющий собой источник нулевых байтов;

53. zfs – файловая система, разработанная компанией Sun Microsystems и обладающая такими характеристиками как возможность хранения больших объёмов данных, управления томами и множеством других.

Перейдём к директории пользователя root и посмотрим содержимое каталога на рисунке 4.

```
Server_Ubuntu [Работает] - Oracle VM VirtualBox
Файл  Машина  Вид  Ввод  Устройства  Справка

vlad@vlad:~$ sudo su
[sudo] password for vlad:
root@vlad:/home/vlad# cd /
root@vlad:/# ls
bin  cdrom  etc  lib  lib64  lost+found  mnt  proc  run  snap  swap.img  tmp  var
boot  dev  home  lib32  libx32  media  opt  root  sbin  srv  sys  usr

root@vlad:/# cd dev
root@vlad:/dev# ls
autofs          hugepages      null           stderr         tty26          tty47          ttyS0          ttyS3          vcs6
block           hwrng          nvram          stdin          tty27          tty48          ttyS1          ttyS30         vcsa
bsg            i2c-0          port           stdout         tty28          tty49          ttyS10         ttyS31         vcsa1
btfsfs-control  initctl        ppp            tty            tty29          tty5           ttyS11         ttyS4          vcsa2
bus            input          psaux          tty0           tty3           tty50          ttyS12         ttyS5          vcsa3
cdrom          kmsg           ptmx           tty1           tty30          tty51          ttyS13         ttyS6          vcsa4
char           lightningvm     pts            tty10          tty31          tty52          ttyS14         ttyS7          vcsa5
console        log            random         tty11          tty32          tty53          ttyS15         ttyS8          vcsa6
core           loop0          rfkill         tty12          tty33          tty54          ttyS16         ttyS9          vcsu
cpu            loop1          rtc            tty13          tty34          tty55          ttyS17         ubuntu-vg      vcsu1
cpu_dma_latency  loop2          rtc0           tty14          tty35          tty56          ttyS18         udmabuf       vcsu2
cuse           loop3          sda            tty15          tty36          tty57          ttyS19         uhid          vcsu3
disk           loop4          sda1           tty16          tty37          tty58          ttyS2           uinput       vcsu4
dm-0           loop5          sda2           tty17          tty38          tty59          ttyS20         urandom       vcsu5
dri            loop6          sda3           tty18          tty39          tty6           ttyS21         userio       vcsu6
dvd            loop7          sg0            tty19          tty4           tty60          ttyS22         vboxguest     vfio
ecryptfs       loop8          sg1            tty2           tty40          tty61          ttyS23         vboxuser     vga_arbiter
fbo            loop-control   sg2            tty20          tty41          tty62          ttyS24         vcs          vhci
fd             mapper         shm            tty21          tty42          tty63          ttyS25         vcs1         vhost-net
full           mcelog         snapshot       tty22          tty43          tty7           ttyS26         vcs2         vhost-vsock
fuse           mem            snd            tty23          tty44          tty8           ttyS27         vcs3         zero
hidraw0        queue          sr0            tty24          tty45          tty9           ttyS28         vcs4         zfs
hpet           net            sr1            tty25          tty46          ttyprintk      ttyS29         vcs5
```

```
root@vlad:/dev# cd /boot
root@vlad:/boot# ls
config-5.4.0-88-generic  initrd.img-5.4.0-88-generic  System.map-5.4.0-88-generic  vmlinuz.old
grub                    initrd.img.old               vmlinuz                      vmlinuz-5.4.0-88-generic
initrd.img              lost+found                   vmlinuz-5.4.0-88-generic
```

Рисунок 4 – Содержимое каталог

Посмотрим содержимое файла vmlinuz с помощью команды cat на рисунке 5.

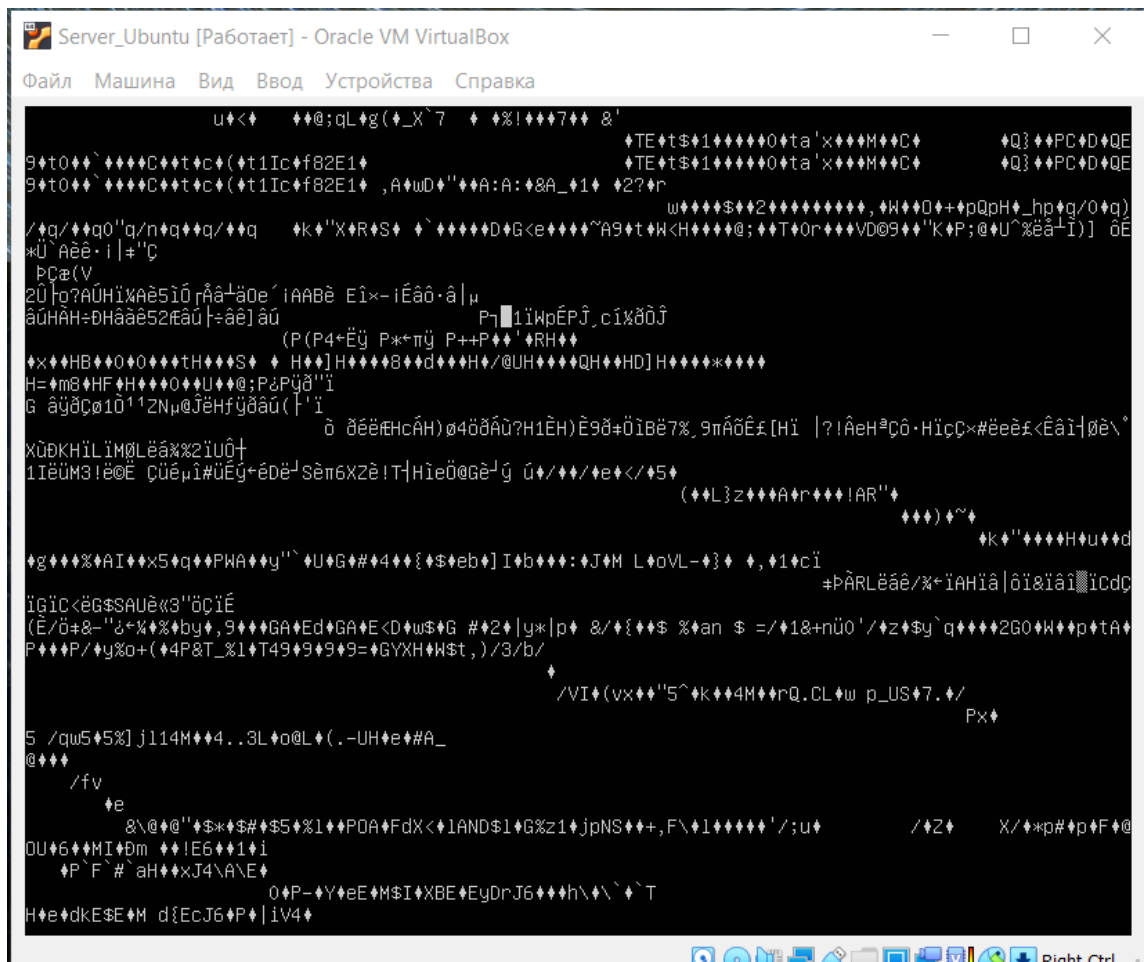


Рисунок 5 – Содержимое файла vmlinuz

Все пользователи и группы пользователей имеют полные права на файл vmlinuz. Владелец файла указан пользователь root. Создадим нового пользователя user, для этого воспользуемся командой useradd. Пример выполнения представлен на рисунке 6



```
Server_Ubuntu [Работает] - Oracle VM VirtualBox
Файл  Машина  Вид  Ввод  Устройства  Справка

root@vlad:/boot# cd /
root@vlad:/# useradd -m user
root@vlad:/# ls -l
total 1776712
drwxrwxrwx 1 root root 7 авг 24 08:41 bin -> usr/bin
drwxr-xr-x 4 root root 4096 окт 16 07:32 boot
drwxr-xr-x 2 root root 4096 окт 16 07:28 cdrom
drwxr-xr-x 20 root root 4160 окт 16 16:47 dev
drwxr-xr-x 97 root root 4096 окт 16 17:07 etc
drwxr-xr-x 4 root root 4096 окт 16 17:07 home
drwxrwxrwx 1 root root 7 авг 24 08:41 lib -> usr/lib
drwxrwxrwx 1 root root 9 авг 24 08:41 lib32 -> usr/lib32
drwxrwxrwx 1 root root 9 авг 24 08:41 lib64 -> usr/lib64
drwxrwxrwx 1 root root 10 авг 24 08:41 libx32 -> usr/libx32
drwx----- 2 root root 16384 окт 16 07:28 lost+found
drwxr-xr-x 2 root root 4096 авг 24 08:42 media
drwxr-xr-x 2 root root 4096 авг 24 08:42 mnt
drwxr-xr-x 2 root root 4096 авг 24 08:42 opt
dr-xr-xr-x 165 root root 0 окт 16 16:45 proc
drwx----- 6 root root 4096 окт 16 16:46 root
drwxr-xr-x 29 root root 900 окт 16 16:47 run
drwxrwxrwx 1 root root 8 авг 24 08:41 sbin -> usr/sbin
drwxr-xr-x 10 root root 4096 окт 16 16:46 snap
drwxr-xr-x 2 root root 4096 авг 24 08:42 srv
-rw----- 1 root root 1819279360 окт 16 07:31 swap.img
dr-xr-xr-x 13 root root 0 окт 16 16:45 sys
drwxrwxrwt 12 root root 4096 окт 16 16:48 tmp
drwxr-xr-x 15 root root 4096 авг 24 08:46 usr
drwxr-xr-x 13 root root 4096 авг 24 08:47 var
root@vlad:/# cd home
root@vlad:/home# ls -l
total 8
drwxr-xr-x 2 user user 4096 окт 16 17:07 user
drwxr-xr-x 3 vlad vlad 4096 окт 16 07:35 vlad
root@vlad:/home#
```

Рисунок 6 – Пример создания нового пользователя user

Создадим в директории пользователя user три файла 1.txt, 2.txt и 3.txt, используя команды touch, cat и текстовый редактор nano. Пример выполнения представлен на рисунках 7 и 8.

```
Server_Ubuntu [Работает] - Oracle VM VirtualBox
Файл  Машина  Вид  Ввод  Устройства  Справка

drwxr-xr-x 20 root root 4160 окт 16 16:47 dev
drwxr-xr-x 97 root root 4096 окт 16 17:07 etc
drwxr-xr-x 4 root root 4096 окт 16 17:07 home
drwxrwxrwx 1 root root 7 авг 24 08:41 lib -> usr/lib
drwxrwxrwx 1 root root 9 авг 24 08:41 lib32 -> usr/lib32
drwxrwxrwx 1 root root 9 авг 24 08:41 lib64 -> usr/lib64
drwxrwxrwx 1 root root 10 авг 24 08:41 libx32 -> usr/libx32
drwx----- 2 root root 16384 окт 16 07:28 lost+found
drwxr-xr-x 2 root root 4096 авг 24 08:42 media
drwxr-xr-x 2 root root 4096 авг 24 08:42 mnt
drwxr-xr-x 2 root root 4096 авг 24 08:42 opt
dr-xr-xr-x 165 root root 0 окт 16 16:45 proc
drwx----- 6 root root 4096 окт 16 16:46 root
drwxr-xr-x 29 root root 900 окт 16 16:47 run
drwxrwxrwx 1 root root 8 авг 24 08:41 sbin -> usr/sbin
drwxr-xr-x 10 root root 4096 окт 16 16:46 snap
drwxr-xr-x 2 root root 4096 авг 24 08:42 srv
-rw----- 1 root root 1819279360 окт 16 07:31 swap.img
dr-xr-xr-x 13 root root 0 окт 16 16:45 sys
drwxrwxrwt 12 root root 4096 окт 16 16:48 tmp
drwxr-xr-x 15 root root 4096 авг 24 08:46 usr
drwxr-xr-x 13 root root 4096 авг 24 08:47 var
root@vlad:/# cd home
root@vlad:/home# ls -l
total 8
drwxr-xr-x 2 user user 4096 окт 16 17:07 user
drwxr-xr-x 3 vlad vlad 4096 окт 16 07:35 vlad
root@vlad:/home# cd user
root@vlad:/home/user# cat > 1.txt
root@vlad:/home/user# touch 2.txt
root@vlad:/home/user# cat > 3.txt
root@vlad:/home/user# ls -l
total 0
-rw-r--r-- 1 root root 0 окт 16 17:10 1.txt
-rw-r--r-- 1 root root 0 окт 16 17:12 2.txt
-rw-r--r-- 1 root root 0 окт 16 17:12 3.txt
root@vlad:/home/user# nano_
```

Рисунок 7 – Пример работы

Просмотрим и поясним права доступа к файлам. Владелец файлов является пользователь root, он имеет полные права на файлы, остальные пользователи имеют только право на чтение.

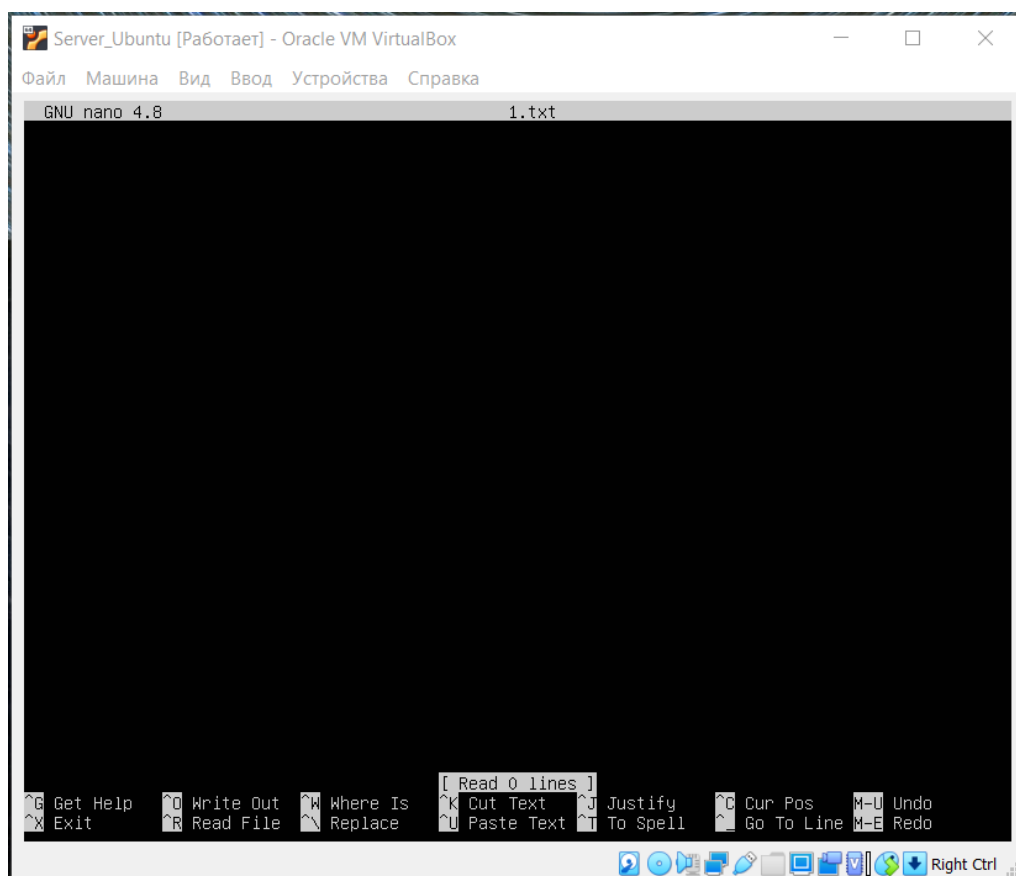


Рисунок 8 – Пример создания файла используя nano

Перейдём в директорию пользователя root. Пример выполнения представлен на рисунке 9

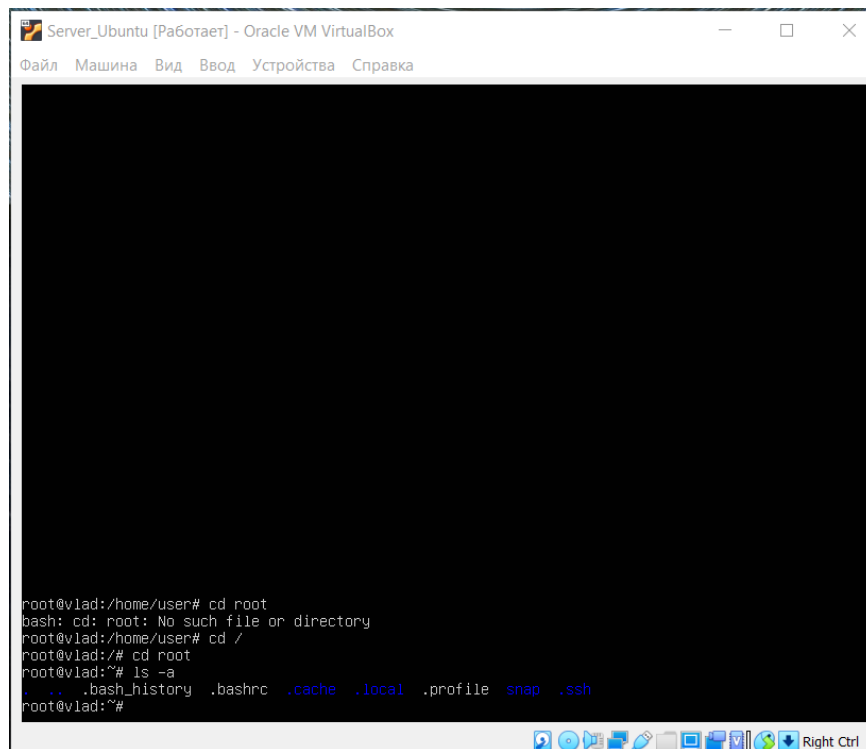


Рисунок 9 – Каталог root

Выполним следующее задание, изменим права доступа на файл 1.txt в директории пользователя user с помощью команды `chmod`.

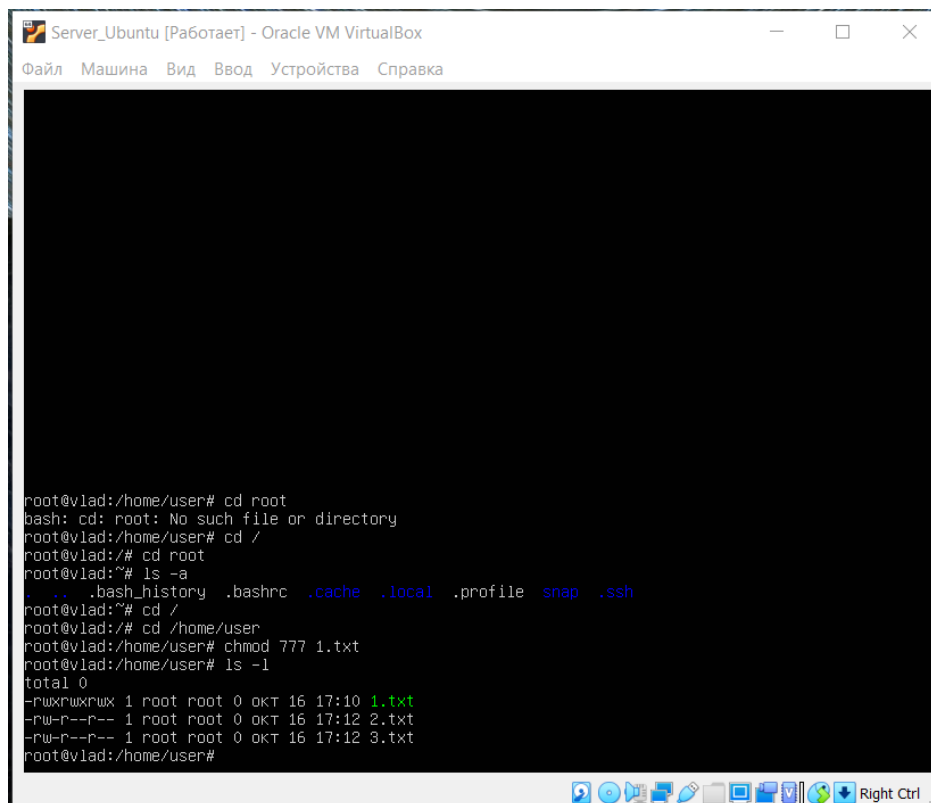
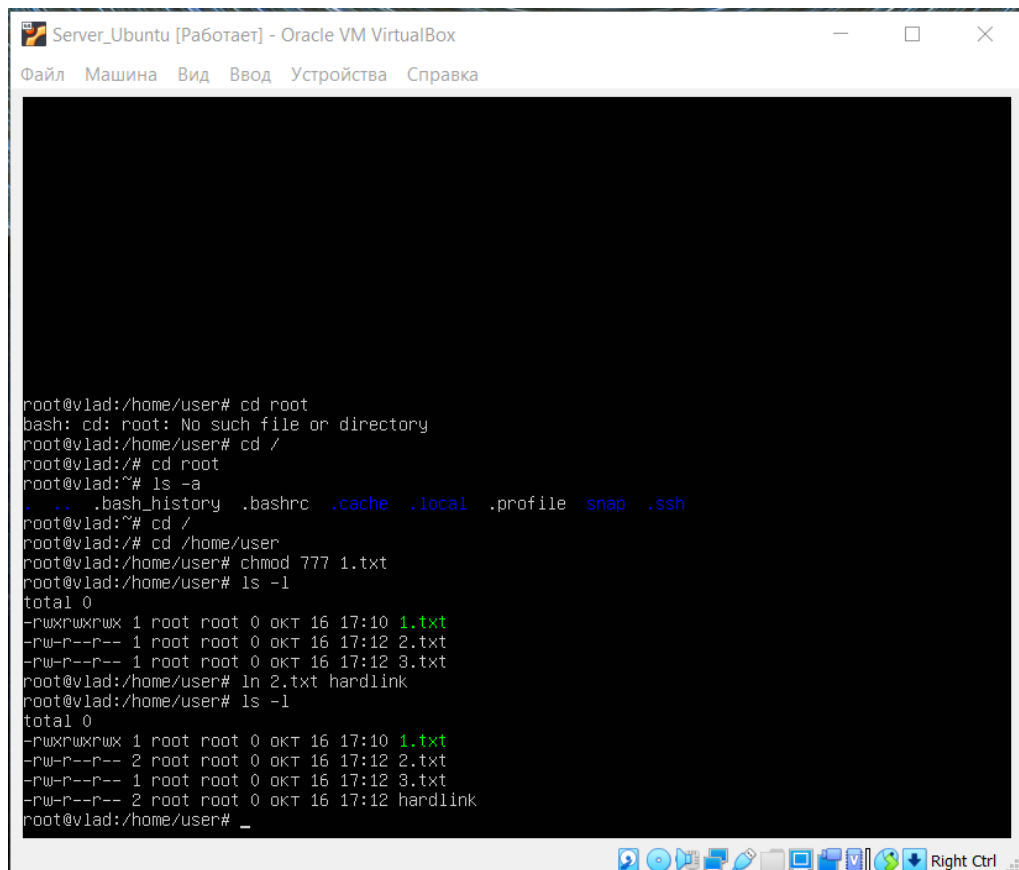


Рисунок 10 – Пример изменения прав доступа с помощью `chmod`

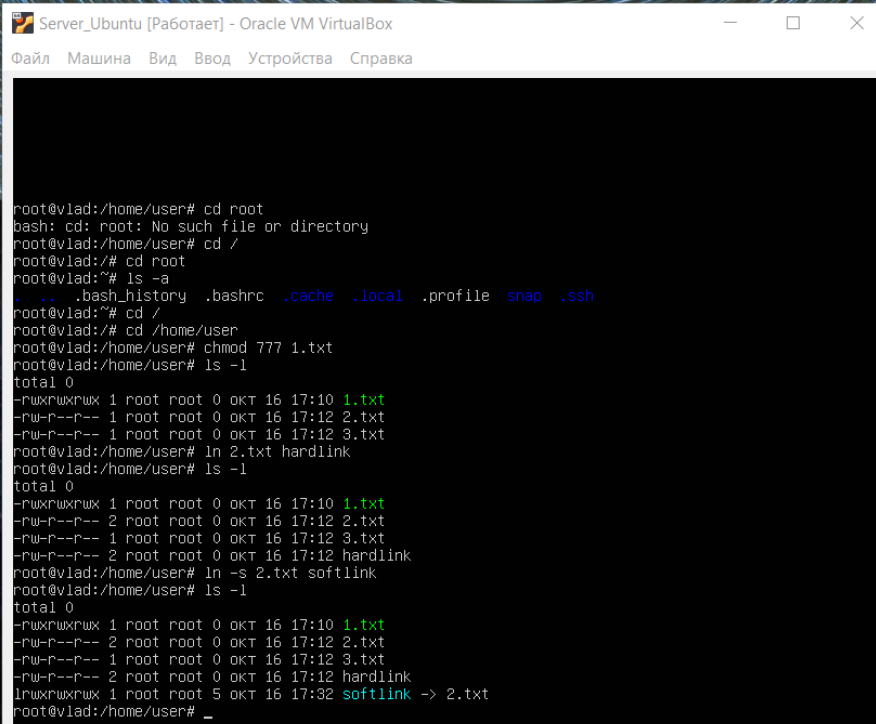
Так как после команды `chmod` было указано значение `777`, то все пользователи имеют право на чтение, изменение и исполнение файла. Далее создадим жёсткую и символическую ссылки на файл `2.txt`. Пример выполнения представлен на рисунка 11 и 12.



```
Server_Ubuntu [Работает] - Oracle VM VirtualBox
Файл  Машина  Вид  Ввод  Устройства  Справка

root@vlad:/home/user# cd root
bash: cd: root: No such file or directory
root@vlad:/home/user# cd /
root@vlad:/# cd root
root@vlad:/# ls -la
.  ..  .bash_history  .bashrc  .cache  .local  .profile  snap  .ssh
root@vlad:/# cd /
root@vlad:/# cd /home/user
root@vlad:/home/user# chmod 777 1.txt
root@vlad:/home/user# ls -l
total 0
-rwxrwxrwx 1 root root 0 окт 16 17:10 1.txt
-rw-r--r-- 1 root root 0 окт 16 17:12 2.txt
-rw-r--r-- 1 root root 0 окт 16 17:12 3.txt
root@vlad:/home/user# ln 2.txt hardlink
root@vlad:/home/user# ls -l
total 0
-rwxrwxrwx 1 root root 0 окт 16 17:10 1.txt
-rw-r--r-- 2 root root 0 окт 16 17:12 2.txt
-rw-r--r-- 1 root root 0 окт 16 17:12 3.txt
-rw-r--r-- 2 root root 0 окт 16 17:12 hardlink
root@vlad:/home/user# _
```

Рисунок 11 – Создание жёсткой ссылки (hardlink)

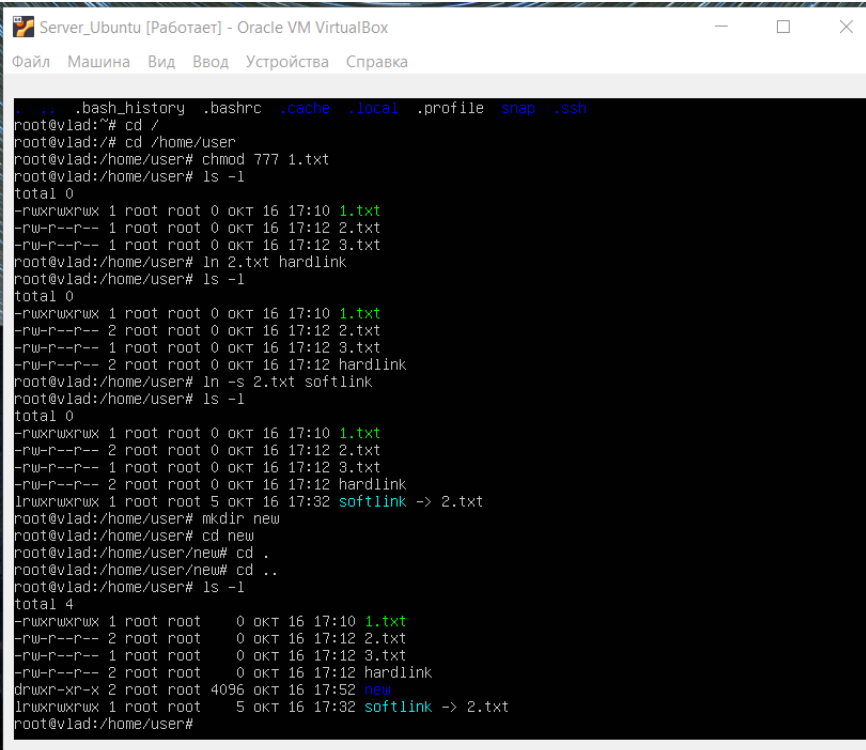


```
Server_Ubuntu [Работает] - Oracle VM VirtualBox
Файл Машина Вид Ввод Устройства Справка

root@vlad:/home/user# cd root
bash: cd: root: No such file or directory
root@vlad:/home/user# cd /
root@vlad:/# cd root
root@vlad:~# ls -a
. .bash_history .bashrc .cache .local .profile snap .ssh
root@vlad:~# cd /
root@vlad:/# cd /home/user
root@vlad:/home/user# chmod 777 1.txt
root@vlad:/home/user# ls -l
total 0
-rwxrwxrwx 1 root root 0 окт 16 17:10 1.txt
-rw-r--r-- 1 root root 0 окт 16 17:12 2.txt
-rw-r--r-- 1 root root 0 окт 16 17:12 3.txt
root@vlad:/home/user# ln 2.txt hardlink
root@vlad:/home/user# ls -l
total 0
-rwxrwxrwx 1 root root 0 окт 16 17:10 1.txt
-rw-r--r-- 2 root root 0 окт 16 17:12 2.txt
-rw-r--r-- 1 root root 0 окт 16 17:12 3.txt
-rw-r--r-- 2 root root 0 окт 16 17:12 hardlink
root@vlad:/home/user# ln -s 2.txt softlink
root@vlad:/home/user# ls -l
total 0
-rwxrwxrwx 1 root root 0 окт 16 17:10 1.txt
-rw-r--r-- 2 root root 0 окт 16 17:12 2.txt
-rw-r--r-- 1 root root 0 окт 16 17:12 3.txt
-rw-r--r-- 2 root root 0 окт 16 17:12 hardlink
lrwxrwxrwx 1 root root 5 окт 16 17:32 softlink -> 2.txt
root@vlad:/home/user# _
```

Рисунок 12 – Создание символической ссылки (softlink)

Далее нужно создать директорию new в каталоге пользователя user. Для этого используем команду mkdir.

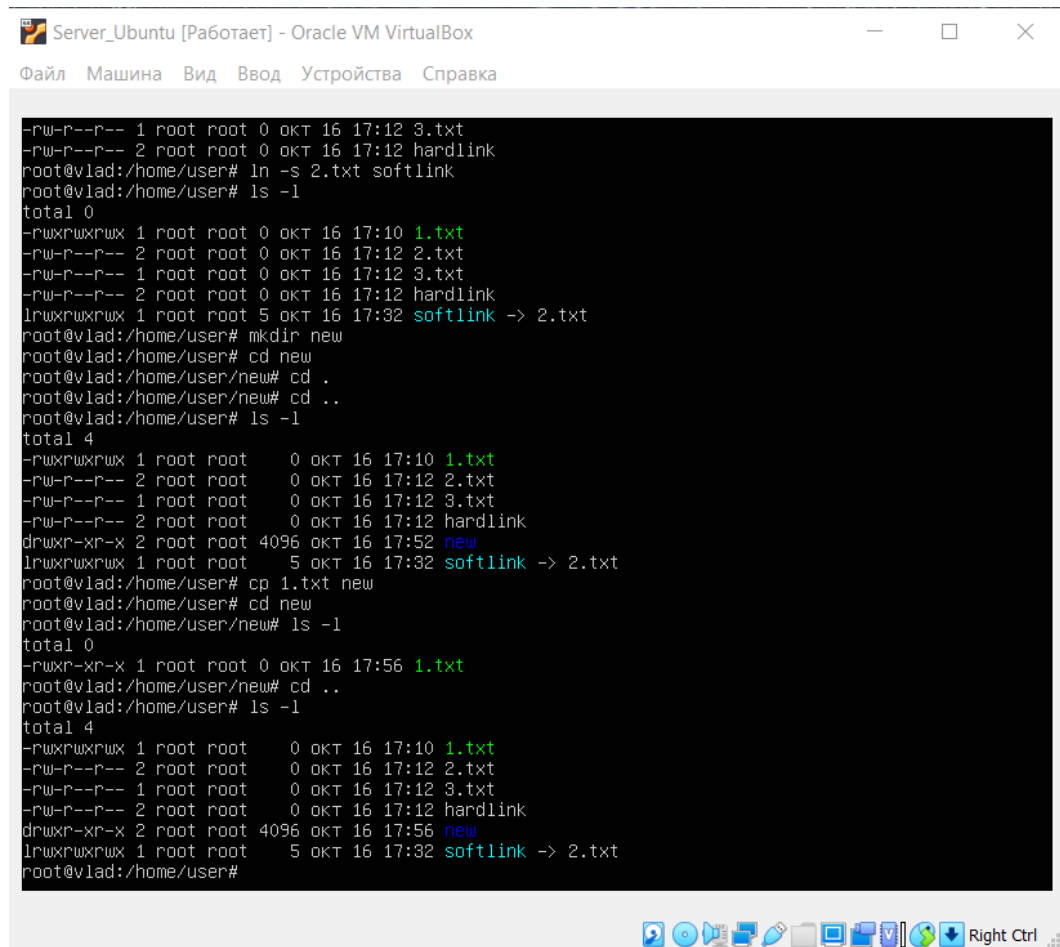


```
Server_Ubuntu [Работает] - Oracle VM VirtualBox
Файл Машина Вид Ввод Устройства Справка

root@vlad:~# cd /
root@vlad:/# cd /home/user
root@vlad:/home/user# chmod 777 1.txt
root@vlad:/home/user# ls -l
total 0
-rwxrwxrwx 1 root root 0 окт 16 17:10 1.txt
-rw-r--r-- 1 root root 0 окт 16 17:12 2.txt
-rw-r--r-- 1 root root 0 окт 16 17:12 3.txt
root@vlad:/home/user# ln 2.txt hardlink
root@vlad:/home/user# ls -l
total 0
-rwxrwxrwx 1 root root 0 окт 16 17:10 1.txt
-rw-r--r-- 2 root root 0 окт 16 17:12 2.txt
-rw-r--r-- 1 root root 0 окт 16 17:12 3.txt
-rw-r--r-- 2 root root 0 окт 16 17:12 hardlink
root@vlad:/home/user# ln -s 2.txt softlink
root@vlad:/home/user# ls -l
total 0
-rwxrwxrwx 1 root root 0 окт 16 17:10 1.txt
-rw-r--r-- 2 root root 0 окт 16 17:12 2.txt
-rw-r--r-- 1 root root 0 окт 16 17:12 3.txt
-rw-r--r-- 2 root root 0 окт 16 17:12 hardlink
lrwxrwxrwx 1 root root 5 окт 16 17:32 softlink -> 2.txt
root@vlad:/home/user# mkdir new
root@vlad:/home/user# cd new
root@vlad:/home/user/new# cd .
root@vlad:/home/user/new# cd ..
root@vlad:/home/user# ls -l
total 4
-rwxrwxrwx 1 root root 0 окт 16 17:10 1.txt
-rw-r--r-- 2 root root 0 окт 16 17:12 2.txt
-rw-r--r-- 1 root root 0 окт 16 17:12 3.txt
-rw-r--r-- 2 root root 0 окт 16 17:12 hardlink
drwxr-xr-x 2 root root 4096 окт 16 17:52 new
lrwxrwxrwx 1 root root 5 окт 16 17:32 softlink -> 2.txt
root@vlad:/home/user#
```

Рисунок 13 – Создание директории new

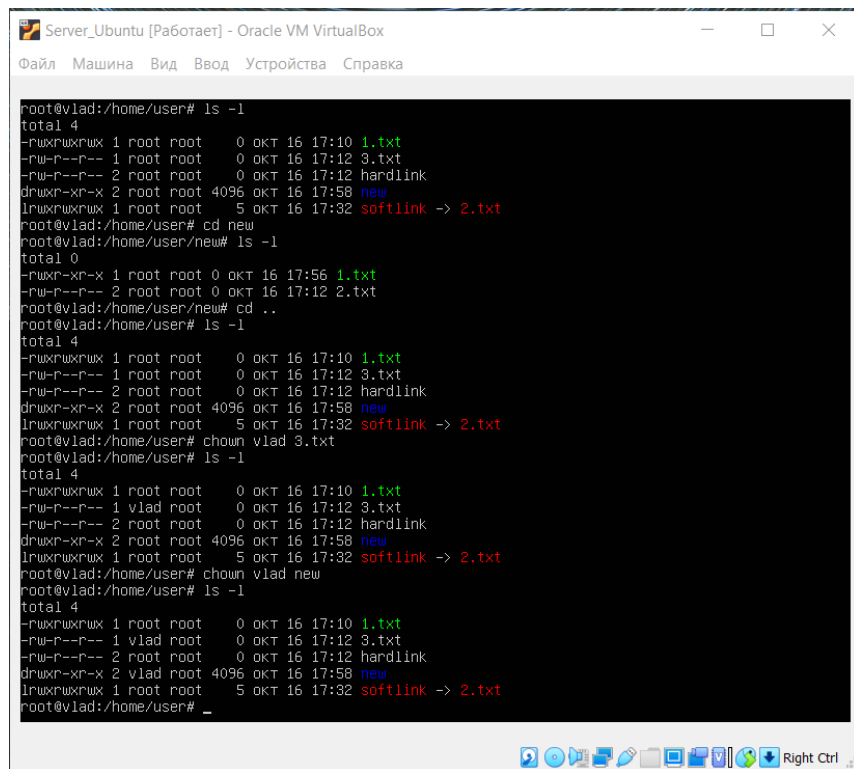
Следующим заданием нужно скопировать файл 1.txt и переместить файл 2.txt в созданную директорию new. Пример выполнения представлен на рисунках 14 и 15.



```
Server_Ubuntu [Работает] - Oracle VM VirtualBox
Файл  Машина  Вид  Ввод  Устройства  Справка

-rw-r--r-- 1 root root 0 окт 16 17:12 3.txt
-rw-r--r-- 2 root root 0 окт 16 17:12 hardlink
root@vlad:/home/user# ln -s 2.txt softlink
root@vlad:/home/user# ls -l
total 0
-rwxrwxrwx 1 root root 0 окт 16 17:10 1.txt
-rw-r--r-- 2 root root 0 окт 16 17:12 2.txt
-rw-r--r-- 1 root root 0 окт 16 17:12 3.txt
-rw-r--r-- 2 root root 0 окт 16 17:12 hardlink
lrwxrwxrwx 1 root root 5 окт 16 17:32 softlink -> 2.txt
root@vlad:/home/user# mkdir new
root@vlad:/home/user# cd new
root@vlad:/home/user/new# cd .
root@vlad:/home/user/new# cd ..
root@vlad:/home/user# ls -l
total 4
-rwxrwxrwx 1 root root 0 окт 16 17:10 1.txt
-rw-r--r-- 2 root root 0 окт 16 17:12 2.txt
-rw-r--r-- 1 root root 0 окт 16 17:12 3.txt
-rw-r--r-- 2 root root 0 окт 16 17:12 hardlink
drwxr-xr-x 2 root root 4096 окт 16 17:52 new
lrwxrwxrwx 1 root root 5 окт 16 17:32 softlink -> 2.txt
root@vlad:/home/user# cp 1.txt new
root@vlad:/home/user# cd new
root@vlad:/home/user/new# ls -l
total 0
-rwxr-xr-x 1 root root 0 окт 16 17:56 1.txt
root@vlad:/home/user/new# cd ..
root@vlad:/home/user# ls -l
total 4
-rwxrwxrwx 1 root root 0 окт 16 17:10 1.txt
-rw-r--r-- 2 root root 0 окт 16 17:12 2.txt
-rw-r--r-- 1 root root 0 окт 16 17:12 3.txt
-rw-r--r-- 2 root root 0 окт 16 17:12 hardlink
drwxr-xr-x 2 root root 4096 окт 16 17:56 new
lrwxrwxrwx 1 root root 5 окт 16 17:32 softlink -> 2.txt
root@vlad:/home/user#
```

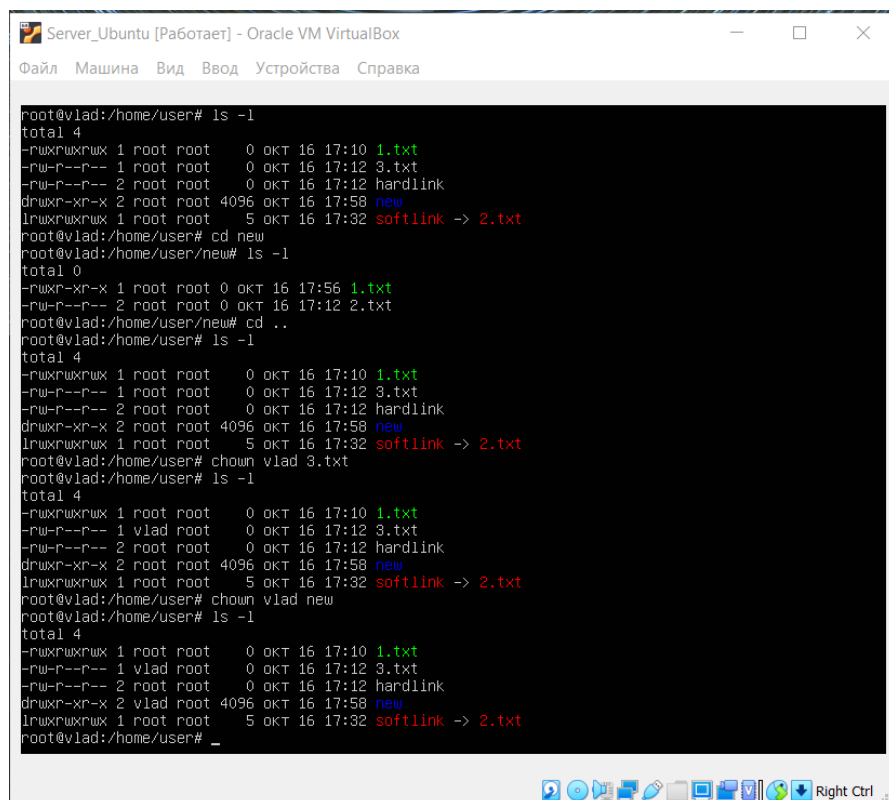
Рисунок 14 – Копирование файла



```
root@vlad:/home/user# ls -l
total 4
-rwxrwxrwx 1 root root 0 окт 16 17:10 1.txt
-rw-r--r-- 1 root root 0 окт 16 17:12 3.txt
-rw-r--r-- 2 root root 0 окт 16 17:12 hardlink
drwxr-xr-x 2 root root 4096 окт 16 17:58 new
lrwxrwxrwx 1 root root 5 окт 16 17:32 softlink -> 2.txt
root@vlad:/home/user# cd new
root@vlad:/home/user/new# ls -l
total 0
-rwxr-xr-x 1 root root 0 окт 16 17:56 1.txt
-rw-r--r-- 2 root root 0 окт 16 17:12 2.txt
root@vlad:/home/user/new# cd ..
root@vlad:/home/user# ls -l
total 4
-rwxrwxrwx 1 root root 0 окт 16 17:10 1.txt
-rw-r--r-- 1 root root 0 окт 16 17:12 3.txt
-rw-r--r-- 2 root root 0 окт 16 17:12 hardlink
drwxr-xr-x 2 root root 4096 окт 16 17:58 new
lrwxrwxrwx 1 root root 5 окт 16 17:32 softlink -> 2.txt
root@vlad:/home/user# chown vlad 3.txt
root@vlad:/home/user# ls -l
total 4
-rwxrwxrwx 1 root root 0 окт 16 17:10 1.txt
-rw-r--r-- 1 vlad root 0 окт 16 17:12 3.txt
-rw-r--r-- 2 root root 0 окт 16 17:12 hardlink
drwxr-xr-x 2 root root 4096 окт 16 17:58 new
lrwxrwxrwx 1 root root 5 окт 16 17:32 softlink -> 2.txt
root@vlad:/home/user# chown vlad new
root@vlad:/home/user# ls -l
total 4
-rwxrwxrwx 1 root root 0 окт 16 17:10 1.txt
-rw-r--r-- 1 vlad root 0 окт 16 17:12 3.txt
-rw-r--r-- 2 root root 0 окт 16 17:12 hardlink
drwxr-xr-x 2 vlad root 4096 окт 16 17:58 new
lrwxrwxrwx 1 root root 5 окт 16 17:32 softlink -> 2.txt
root@vlad:/home/user# _
```

Рисунок 15 – Перемещение файла

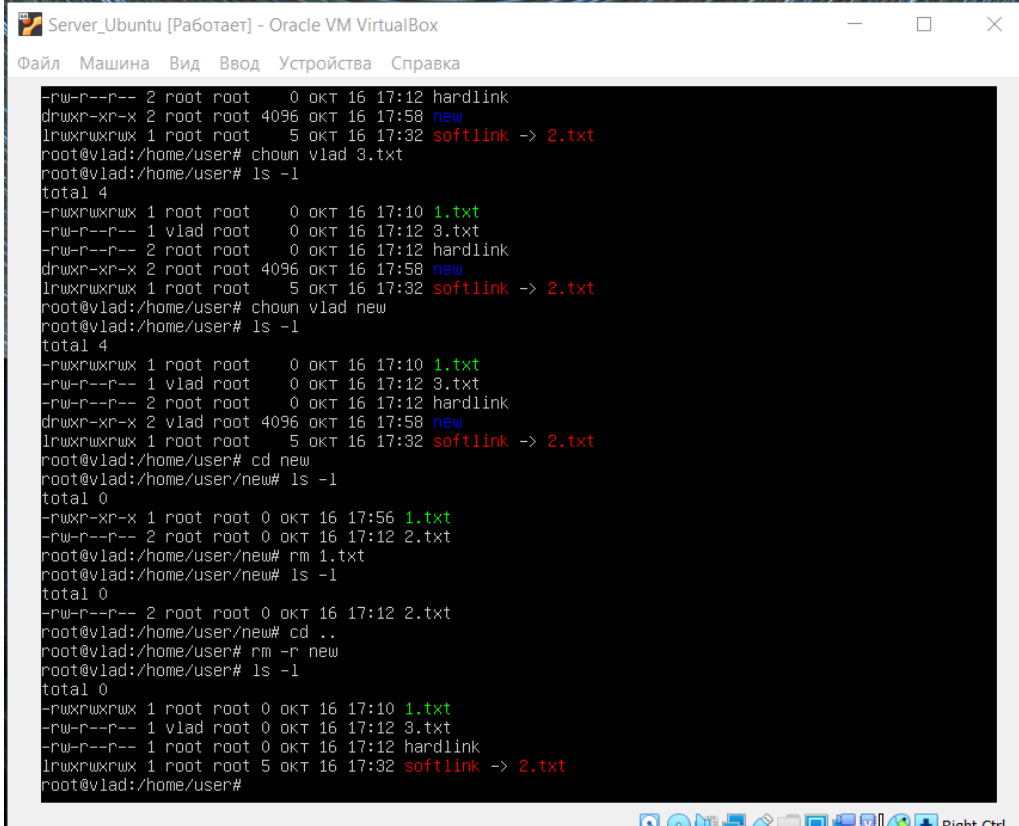
Далее нужно поменять владельцев файла 3.txt и каталога new реализуем это с помощью chown



```
root@vlad:/home/user# ls -l
total 4
-rwxrwxrwx 1 root root 0 окт 16 17:10 1.txt
-rw-r--r-- 1 root root 0 окт 16 17:12 3.txt
-rw-r--r-- 2 root root 0 окт 16 17:12 hardlink
drwxr-xr-x 2 root root 4096 окт 16 17:58 new
lrwxrwxrwx 1 root root 5 окт 16 17:32 softlink -> 2.txt
root@vlad:/home/user# cd new
root@vlad:/home/user/new# ls -l
total 0
-rwxr-xr-x 1 root root 0 окт 16 17:56 1.txt
-rw-r--r-- 2 root root 0 окт 16 17:12 2.txt
root@vlad:/home/user/new# cd ..
root@vlad:/home/user# ls -l
total 4
-rwxrwxrwx 1 root root 0 окт 16 17:10 1.txt
-rw-r--r-- 1 root root 0 окт 16 17:12 3.txt
-rw-r--r-- 2 root root 0 окт 16 17:12 hardlink
drwxr-xr-x 2 root root 4096 окт 16 17:58 new
lrwxrwxrwx 1 root root 5 окт 16 17:32 softlink -> 2.txt
root@vlad:/home/user# chown vlad 3.txt
root@vlad:/home/user# ls -l
total 4
-rwxrwxrwx 1 root root 0 окт 16 17:10 1.txt
-rw-r--r-- 1 vlad root 0 окт 16 17:12 3.txt
-rw-r--r-- 2 root root 0 окт 16 17:12 hardlink
drwxr-xr-x 2 root root 4096 окт 16 17:58 new
lrwxrwxrwx 1 root root 5 окт 16 17:32 softlink -> 2.txt
root@vlad:/home/user# chown vlad new
root@vlad:/home/user# ls -l
total 4
-rwxrwxrwx 1 root root 0 окт 16 17:10 1.txt
-rw-r--r-- 1 vlad root 0 окт 16 17:12 3.txt
-rw-r--r-- 2 root root 0 окт 16 17:12 hardlink
drwxr-xr-x 2 vlad root 4096 окт 16 17:58 new
lrwxrwxrwx 1 root root 5 окт 16 17:32 softlink -> 2.txt
root@vlad:/home/user# _
```

Рисунок 16 – Изменение владельцев файла и каталога

По заданию нужно удалить файл 1.txt из директории new, а затем удалить директорию new. Используем для этого команду rm:



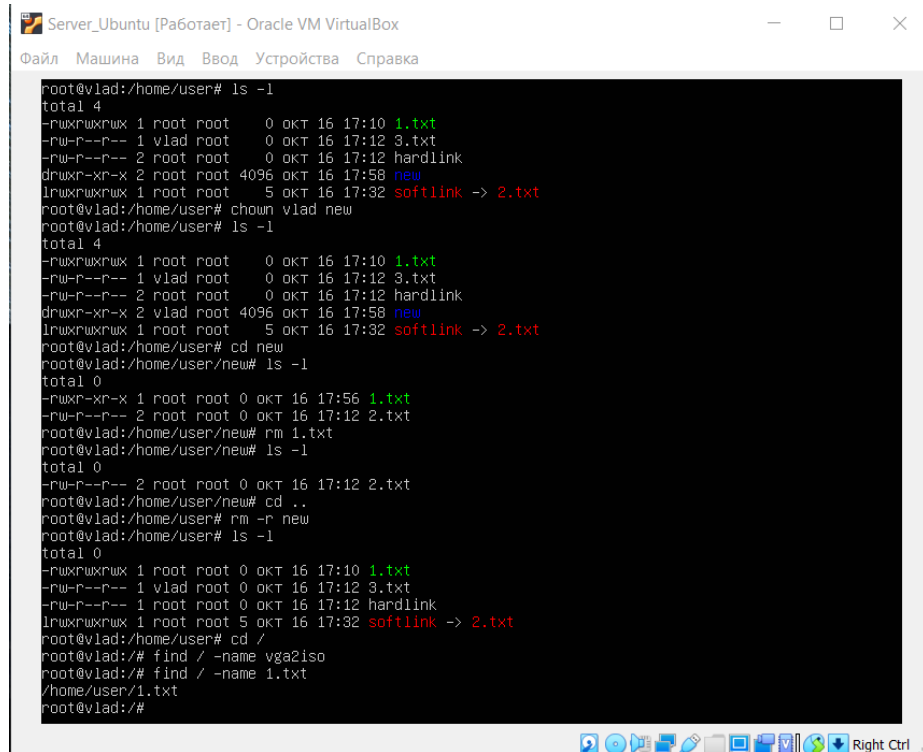
```
Server_Ubuntu [Работает] - Oracle VM VirtualBox
Файл  Машина  Вид  Ввод  Устройства  Справка

-rw-r--r-- 2 root root 0 окт 16 17:12 hardlink
drwxr-xr-x 2 root root 4096 окт 16 17:58 new
lrwxrwxrwx 1 root root 5 окт 16 17:32 softlink -> 2.txt
root@vlad:/home/user# chown vlad 3.txt
root@vlad:/home/user# ls -l
total 4
-rwxrwxrwx 1 root root 0 окт 16 17:10 1.txt
-rw-r--r-- 1 vlad root 0 окт 16 17:12 3.txt
-rw-r--r-- 2 root root 0 окт 16 17:12 hardlink
drwxr-xr-x 2 root root 4096 окт 16 17:58 new
lrwxrwxrwx 1 root root 5 окт 16 17:32 softlink -> 2.txt
root@vlad:/home/user# chown vlad new
root@vlad:/home/user# ls -l
total 4
-rwxrwxrwx 1 root root 0 окт 16 17:10 1.txt
-rw-r--r-- 1 vlad root 0 окт 16 17:12 3.txt
-rw-r--r-- 2 root root 0 окт 16 17:12 hardlink
drwxr-xr-x 2 vlad root 4096 окт 16 17:58 new
lrwxrwxrwx 1 root root 5 окт 16 17:32 softlink -> 2.txt
root@vlad:/home/user# cd new
root@vlad:/home/user/new# ls -l
total 0
-rwxr-xr-x 1 root root 0 окт 16 17:56 1.txt
-rw-r--r-- 2 root root 0 окт 16 17:12 2.txt
root@vlad:/home/user/new# rm 1.txt
root@vlad:/home/user/new# ls -l
total 0
-rw-r--r-- 2 root root 0 окт 16 17:12 2.txt
root@vlad:/home/user/new# cd ..
root@vlad:/home/user# rm -r new
root@vlad:/home/user# ls -l
total 0
-rwxrwxrwx 1 root root 0 окт 16 17:10 1.txt
-rw-r--r-- 1 vlad root 0 окт 16 17:12 3.txt
-rw-r--r-- 1 root root 0 окт 16 17:12 hardlink
lrwxrwxrwx 1 root root 5 окт 16 17:32 softlink -> 2.txt
root@vlad:/home/user#
```

Рисунок 17 – Удаление файла и директории

Последним заданием лабораторной работы является поиск файла vga2iso с использованием команды find. Осуществим эту операцию:

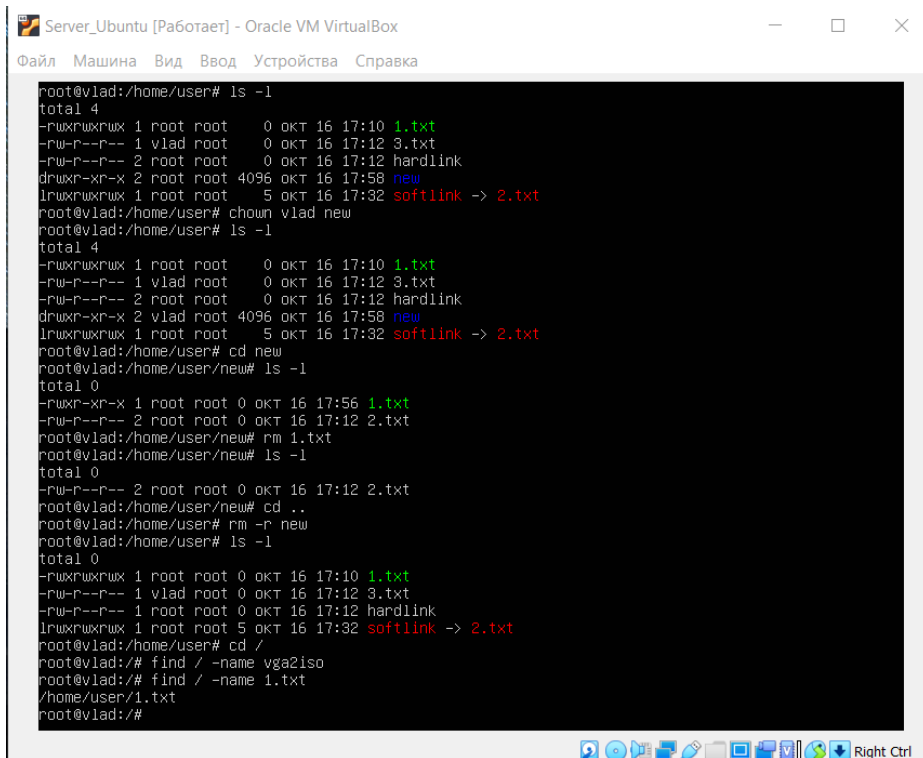




```
root@vlad:/home/user# ls -l
total 4
-rwxrwxrwx 1 root root 0 окт 16 17:10 1.txt
-rw-r--r-- 1 vlad root 0 окт 16 17:12 3.txt
-rw-r--r-- 2 root root 0 окт 16 17:12 hardlink
drwxr-xr-x 2 root root 4096 окт 16 17:58 new
lrwxrwxrwx 1 root root 5 окт 16 17:32 softlink -> 2.txt
root@vlad:/home/user# chown vlad new
root@vlad:/home/user# ls -l
total 4
-rwxrwxrwx 1 root root 0 окт 16 17:10 1.txt
-rw-r--r-- 1 vlad root 0 окт 16 17:12 3.txt
-rw-r--r-- 2 root root 0 окт 16 17:12 hardlink
drwxr-xr-x 2 vlad root 4096 окт 16 17:58 new
lrwxrwxrwx 1 root root 5 окт 16 17:32 softlink -> 2.txt
root@vlad:/home/user# cd new
root@vlad:/home/user/new# ls -l
total 0
-rwxr-xr-x 1 root root 0 окт 16 17:56 1.txt
-rw-r--r-- 2 root root 0 окт 16 17:12 2.txt
root@vlad:/home/user/new# rm 1.txt
root@vlad:/home/user/new# ls -l
total 0
-rw-r--r-- 2 root root 0 окт 16 17:12 2.txt
root@vlad:/home/user/new# cd ..
root@vlad:/home/user# rm -r new
root@vlad:/home/user# ls -l
total 0
-rwxrwxrwx 1 root root 0 окт 16 17:10 1.txt
-rw-r--r-- 1 vlad root 0 окт 16 17:12 3.txt
-rw-r--r-- 1 root root 0 окт 16 17:12 hardlink
lrwxrwxrwx 1 root root 5 окт 16 17:32 softlink -> 2.txt
root@vlad:/home/user# cd /
root@vlad:/# find / -name vga2iso
root@vlad:/# find / -name 1.txt
/home/user/1.txt
root@vlad:/#
```

Рисунок 18 – Нахождение файла vga2iso

Так как работа производится в более новой версии Ubuntu Server файла vga2iso не существует. Поэтому приведём пример работы программы поиска другого файла, а именно 1.txt.



```
root@vlad:/home/user# ls -l
total 4
-rwxrwxrwx 1 root root 0 окт 16 17:10 1.txt
-rw-r--r-- 1 vlad root 0 окт 16 17:12 3.txt
-rw-r--r-- 2 root root 0 окт 16 17:12 hardlink
drwxr-xr-x 2 root root 4096 окт 16 17:58 new
lrwxrwxrwx 1 root root 5 окт 16 17:32 softlink -> 2.txt
root@vlad:/home/user# chown vlad new
root@vlad:/home/user# ls -l
total 4
-rwxrwxrwx 1 root root 0 окт 16 17:10 1.txt
-rw-r--r-- 1 vlad root 0 окт 16 17:12 3.txt
-rw-r--r-- 2 root root 0 окт 16 17:12 hardlink
drwxr-xr-x 2 vlad root 4096 окт 16 17:58 new
lrwxrwxrwx 1 root root 5 окт 16 17:32 softlink -> 2.txt
root@vlad:/home/user# cd new
root@vlad:/home/user/new# ls -l
total 0
-rwxr-xr-x 1 root root 0 окт 16 17:56 1.txt
-rw-r--r-- 2 root root 0 окт 16 17:12 2.txt
root@vlad:/home/user/new# rm 1.txt
root@vlad:/home/user/new# ls -l
total 0
-rw-r--r-- 2 root root 0 окт 16 17:12 2.txt
root@vlad:/home/user/new# cd ..
root@vlad:/home/user# rm -r new
root@vlad:/home/user# ls -l
total 0
-rwxrwxrwx 1 root root 0 окт 16 17:10 1.txt
-rw-r--r-- 1 vlad root 0 окт 16 17:12 3.txt
-rw-r--r-- 1 root root 0 окт 16 17:12 hardlink
lrwxrwxrwx 1 root root 5 окт 16 17:32 softlink -> 2.txt
root@vlad:/home/user# cd /
root@vlad:/# find / -name vga2iso
root@vlad:/# find / -name 1.txt
/home/user/1.txt
root@vlad:/#
```

Рисунок 19 – Нахождение файла 1.txt

## Вывод

В процессе выполнения лабораторной работы была изучена файловая система ОС Linux и основные операции, а именно: просмотр директории, создание нового пользователя, различные операции с файлами (создание, перемещение, копирование, удаление, изменение прав доступа на файл), создание директории, поиск файла и изменение прав доступа на файл. Также изучены особенности установки виртуальной машины с последующим запуском в ней дистрибутива Linux Ubuntu Server.

## Контрольные вопросы

### 1. Что такое файловая система?

Файловая система – это структура, с помощью которой ядро операционной системы предоставляет пользователям (и процессам) ресурсы долговременной памяти системы, т. е. памяти на долговременных носителях информации - жестких дисках, магнитных лентах, CD-ROM и т. п. С точки зрения пользователя, файловая система — это логическая структура каталогов и файлов.

### 2. Права доступа к файлам. Назначение прав доступа.

Права доступа и информация о типе файла в UNIX-системах хранятся в индексных дескрипторах в отдельной структуре, состоящей из двух байтов. Четыре бита из этих 16-ти отведены для кодированной записи о типе файла. И, наконец, оставшиеся 9 бит определяют права доступа к файлу. Право на чтение (r) файла означает, что пользователь может просматривать содержимое файла. Но вы не сможете сохранить изменения в файле, если не имеете права на запись (w) в этот файл. Право на выполнение (x) означает, что вы можете попытаться запустить его на выполнение как исполняемую программу.

### 3. Жёсткая ссылка в Linux. Основные сведения.

Жесткая ссылка является просто другим именем для исходного файла. После создания такой ссылки ее невозможно отличить от исходного имени файла. «Настоящего» имени у файла нет, точнее, все такие имена будут настоящими. Удаление файла по любому из его имён уменьшает на единицу количество ссылок, и окончательно файл будет удален только тогда, когда это количество станет равным нулю. Поэтому удобно использовать жесткие ссылки для того, чтобы предотвратить случайное удаление важного файла.

#### 4. Команда поиска в Linux. Основные сведения.

Команда `find` может искать файлы по имени, размеру, дате создания или модификации и некоторым другим критериям. Общий синтаксис команды `find` имеет следующий вид: `find [список_каталогов] критерий_поиска`. Параметр "список\_каталогов" определяет, где искать нужный файл. Проще всего задать в качестве начального каталога поиска корневой каталог `/`, однако, в таком случае поиск может затянуться очень надолго, так как будет просматриваться вся структура каталогов, включая смонтированные файловые системы.

#### 5. Перечислите основные команды работы с каталогами.

- 1) Просмотр каталога (list): `ls -ключи путь/имя_файла`;
- 2) Узнать текущий каталог: `pwd`;
- 3) Сменить текущий каталог: `cd имя_каталога`;
- 4) Создание нового каталога: `mkdir путь/имя_каталога`;
- 5) Удаление пустого каталога: `rmdir путь/имя_каталога`