

Липецкий государственный технический университет

Факультет автоматизации и информатики

Кафедра автоматизированных систем управления

ЛАБОРАТОРНАЯ РАБОТА №1

по дисциплине «Операционная система Linux»

Работа с файловой системой ОС Linux

Студент

Посаднев В.В.

Группа АС-18

Руководитель

Кургасов В.В.

Липецк 2020 г.

Цель работы

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

Ход работы

Для выполнения данной лабораторной работы была установлена ОС Linux на виртуальную, а именно Ubuntu 18.04. Результат запуска данной ОС изображен на рисунке 1.

```
Ubuntu 18.04.5 LTS ubuntu tty1
ubuntu login: elemabor
Password:
Last login: Sun Oct 11 15:50:17 PDT 2020 on tty1
Welcome to Ubuntu 18.04.5 LTS (GNU/Linux 4.15.0-118-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage
New release '20.04.1 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

elemabor@ubuntu:~$ _
```

Рисунок 1 – Результат запуска системы Linux

Следующим шагом необходимо авторизоваться под пользователем root, для этого необходимо выполнить команду `sudo su`. Результат успешной авторизации изображен на рисунке 2.

```
Ubuntu 18.04.5 LTS ubuntu tty1
ubuntu login: elemabor
Password:
Last login: Sun Oct 11 15:50:17 PDT 2020 on tty1
Welcome to Ubuntu 18.04.5 LTS (GNU/Linux 4.15.0-118-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage
New release '20.04.1 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

elemabor@ubuntu:~$ whoami
elemabor
elemabor@ubuntu:~$ sudo su
[sudo] password for elemabor:
root@ubuntu:/home/eleabor# whoami
root
root@ubuntu:/home/eleabor#
```

Рисунок 2 – Авторизация под пользователем root

Следующим подпунктом выполнения является приведения перечня каталогов с указанием их назначения. Для этого необходимо выполнить команду `ls -a`. Пример выполнения данной команды находится на рисунке 3.

```
root@ubuntu:/# ls -a
.   boot  home   lib      media  proc  sbin   sys   var
..  dev    initrd.img  lib64   mnt    root  srv    tmp   vmlinuz
bin  etc    initrd.img.old  lost+found  opt    run   swapfile  usr   vmlinuz.old
root@ubuntu:/# _
```

Рисунок 3 – Перечень каталогов в корневой папке

Определим для чего необходим каждый элемент корневого каталога, приведенного на рисунке 3.

bin – бинарные файлы пользователя

boot – файлы загрузчика ОС

dev – файлы устройств

etc – конфигурационные файлы

home – домашняя папка

lib – системные библиотеки

lib64 – системные библиотеки x64-разрядности

lost+found – хранение сломанных файлов при сбоях файловой системы

media – съемные носители

mnt – монтированные устройства

opt – дополнительные программы

proc – информация о процессах

root – директория суперпользователя

run – процессы

sbin – системные исполняемые файлы

srv – сервер

swapfile – файл подкачки

sys – информация о системе

tmp – временные файлы

usr – программы пользователя

var – переменные файлы

Для получения перечня файлов физических устройств на рабочем месте с указанием их назначения необходимо перейти в соответствующую папку, а именно dev. Это можно сделать с помощью команды `cd dev/`. Результат выполнения данной команды приведен на рисунке 4.

```

root@ubuntu:/# ls -la
.      boot  home      lib        media      proc  sbin      sys  var
..     dev   initrd.img lib64       mnt        root  srv       tmp  vmlinuz
bin    etc     initrd.img.old lost+found  opt       run   swapfile  usr  vmlinuz.old
root@ubuntu:/# cd dev/
root@ubuntu:/dev# ls -la
.      initctl      pts          tty19  tty44      ttyS10  ttyS8
..     input       random      tty2   tty45      ttyS11  ttyS9
agpgart  kmsg        rfkill      tty20  tty46      ttyS12  uhid
autofs   lightnvm    rtc         tty21  tty47      ttyS13  uinput
block    log         rtc0        tty22  tty48      ttyS14  urandom
bsg      loop0       sda         tty23  tty49      ttyS15  userio
btrfs-control loop1       sda1        tty24  tty5       ttyS16  vcs
bus      loop2       sg0         tty25  tty50      ttyS17  vcs1
cdrom    loop3       sg1         tty26  tty51      ttyS18  vcs2
cdrom    loop4       shm         tty27  tty52      ttyS19  vcs3
char     loop5       snapshot    tty28  tty53      ttyS2   vcs4
console  loop6       snd         tty29  tty54      ttyS20  vcs5
core     loop7       sr0         tty3   tty55      ttyS21  vcs6
cpu_dma_latency loop-control stderr      tty30  tty56      ttyS22  vcsa
cuse     mapper      stdin      tty31  tty57      ttyS23  vcsa1
disk     mcelog     stdout     tty32  tty58      ttyS24  vcsa2
dmide    mem         tty        tty33  tty59      ttyS25  vcsa3
dri      memory_bandwidth tty0       tty34  tty6       ttyS26  vcsa4
dvd      midi        tty1       tty35  tty60      ttyS27  vcsa5
encryptfs mqueue     tty10      tty36  tty61      ttyS28  vcsa6
fb0      net         tty11      tty37  tty62      ttyS29  vfiio
fd       network_latency tty12      tty38  tty63      ttyS3   vga_arbiter
full     network_throughput tty13      tty39  tty7       ttyS30  vhci
fuse     null        tty14      tty4   tty8       ttyS31  vhost-net
hidraw0  port        tty15      tty40  tty9       ttyS4   vhost-vsock
hpet     ppp         tty16      tty41  ttyprintk  ttyS5   vmci
hugepages psaux       tty17      tty42  ttyS0      ttyS6   vsock
hwrng    ptmx       tty18      tty43  ttyS1      ttyS7   zero
root@ubuntu:/dev# _

```

Рисунок 4 – Содержимое каталога устройств

Описание устройств

agpgart – модуль ядра видеопроцессора

autofs – пакет для автоматического подключения разнообразных ресурсов, к примеру переносные устройства

block – таблицы разделов внешних устройств

btrfs-control – модуль контроля файловой системы

bus – модуль для общения взаимодействия приложений между собой

cdrom – модуль для дискового устройства

console – модуль для терминала ОС

cpu_dma_latency – модуль конфигурации загрузчика

cuse – реализация символьных устройств

ecryptfs – модуль для защиты файлов безотносительно к не желаемым файловым системам

fb0 – модуль для хранения промежуточных значений

fuse – формат и настройки для предохранения файловой системы

hidraw0 – чтение уровней доступа для потоков

hpet – модуль драйвер таймера для точности вычисления

hwng – проверка и заполнение данных из расшифрованных данных

loop-x – циклические устройства

null – несуществующее устройство

random – случайные числа

rftkill – модуль который запрашивает состояние переключателей, кнопок и интерфейсов подсистемы

rtc – real-time clock

sd(a..) – SCSI-диски

snapshot – создание резервной точки остановки системы

ttyN – консоль с номером N

uhid – модуль для работы с USB HID

vcs(a..) – память виртуальной консоли

vsoc – ядро для виртуальной машины VMWare

Для выполнения следующего пункта необходимо выполнить команду `ls -a` в консоли. Результат выполнения данной команды приведен на рисунке 5.

```
root@ubuntu:~# ls -a
.  ..  .bashrc  .profile
root@ubuntu:~#
```

Рисунок 5 – Содержимое каталога root

Для доступа к файлу `vmlinuz` необходимо перейти в корневой каталог, для этого необходимо выполнить команду `cd /` и `ls -l`, следующим шагом необходимо открыть данный файл. Для текстового редактора был выбран `vim`, для открытия необходимо прописать `vim vmlinuz`. Пример выполнения данных команд изображены на рисунках 6 и 7.

```
root@ubuntu:~# ls -l
total 483888
drwxr-xr-x  2 root root    4096 Oct 11 15:56 bin
drwxr-xr-x  3 root root    4096 Oct 11 16:00 boot
drwxr-xr-x 18 root root   3920 Oct 11 22:56 dev
drwxr-xr-x 76 root root    4096 Oct 11 15:58 etc
drwxr-xr-x  3 root root    4096 Oct 11 15:46 home
lrwxrwxrwx  1 root root      34 Oct 11 15:58 initrd.img -> boot/initrd.img-4.15.0-118-generic
lrwxrwxrwx  1 root root      34 Oct 11 15:34 initrd.img.old -> boot/initrd.img-4.15.0-112-generic
drwxr-xr-x 18 root root    4096 Oct 11 15:40 lib
drwxr-xr-x  2 root root    4096 Oct 11 15:32 lib64
drwx----- 2 root root  16384 Oct 11 15:31 lost+found
drwxr-xr-x  4 root root    4096 Oct 11 15:32 media
drwxr-xr-x  2 root root    4096 Aug 10 01:41 mnt
drwxr-xr-x  2 root root    4096 Oct 11 15:50 opt
dr-xr-xr-x 149 root root      0 Oct 11 22:56 proc
drwx----- 2 root root    4096 Oct 11 15:31 root
drwxr-xr-x 18 root root    520 Oct 11 23:42 run
drwxr-xr-x  2 root root    4096 Oct 11 15:56/sbin
drwxr-xr-x  2 root root    4096 Aug 10 01:41/srv
-rw-----  1 root root 495416320 Oct 11 15:31 swapfile
dr-xr-xr-x 13 root root      0 Oct 11 23:04 sys
drwxrwxrwt 11 root root    4096 Oct 11 22:56 tmp
drwxr-xr-x 10 root root    4096 Oct 11 15:32/usr
drwxr-xr-x 11 root root    4096 Oct 11 15:32/var
lrwxrwxrwx  1 root root      31 Oct 11 15:58 vmlinuz -> boot/vmlinuz-4.15.0-118-generic
lrwxrwxrwx  1 root root      31 Oct 11 15:34 vmlinuz.old -> boot/vmlinuz-4.15.0-112-generic
root@ubuntu:~#
```

Рисунок 6 – Содержимое корневого каталога

vim. Результат создания трех файлов командой touch приведен на рисунке 9. После создания новых файлов запишем в них какие-нибудь данные с помощью текстового редактора vim, после выведем содержимое полученных файлов с помощью команды cat и название файла. Результат выполнения данных команд изображен на рисунке 10.

```
root@ubuntu:/home# cd user/
root@ubuntu:/home/user# ls
root@ubuntu:/home/user# touch 1.txt
root@ubuntu:/home/user# touch 2.txt
root@ubuntu:/home/user# touch 3.txt
root@ubuntu:/home/user# ls
1.txt 2.txt 3.txt
root@ubuntu:/home/user# _
```

Рисунок 9 – Результат создания трех файлов

```
root@ubuntu:/home/user# ls
1.txt 2.txt 3.txt
root@ubuntu:/home/user# cat 1.txt
sample text for first file
root@ubuntu:/home/user# cat 2.txt
test text for second file
root@ubuntu:/home/user# cat 3.txt
new data from third file
root@ubuntu:/home/user# ls -l
total 12
-rw-r--r-- 1 root root 27 Oct 14 05:30 1.txt
-rw-r--r-- 1 root root 26 Oct 14 05:30 2.txt
-rw-r--r-- 1 root root 25 Oct 14 05:31 3.txt
root@ubuntu:/home/user#
```

Рисунок 10 – Вывод содержимого трех текстовых файлов

Также на рисунке 10 изображен результат работы команды ls -l, которое показывает права доступа на файлы в текущем каталоге. Как видно из полученного результата владелец файлов является root и он может производить чтение-запись с этими файлами. Остальные пользователи имеют права доступа только чтение.

Для изменения прав доступа на файл 1.txt в директории пользователя user необходимо выполнить команду chmod уровень_доступа и название файла уровень доступа которого хотим изменить. Для примера выполним команду для установки прав на чтение и запись для владельца файла, группы и остальных пользователей – chmod 666 1.txt. После выполнения данной команды посмотрим на права доступа еще раз, пример выполнения данной комбинации команд изображен на рисунке 11.

```

root@ubuntu:/home/user# ls -l
total 12
-rw-r--r-- 1 root root 27 Oct 14 05:30 1.txt
-rw-r--r-- 1 root root 26 Oct 14 05:30 2.txt
-rw-r--r-- 1 root root 25 Oct 14 05:31 3.txt
root@ubuntu:/home/user# chmod 666 1.txt
root@ubuntu:/home/user# ls -l
total 12
-rw-rw-rw- 1 root root 27 Oct 14 05:30 1.txt
-rw-r--r-- 1 root root 26 Oct 14 05:30 2.txt
-rw-r--r-- 1 root root 25 Oct 14 05:31 3.txt
root@ubuntu:/home/user#

```

Рисунок 11 – Пример изменения прав доступа на файл 1.txt

Следующим шагом выполнения лабораторной работы идет создание жесткой и символической ссылки на файл 2.txt. Данные ссылки позволяют размещать один и тот же файл в нескольких директориях, если проводить аналогию с Windows, то это ярлыки на файлы. Каждая из жестких ссылок это отдельный файл которые ведут к одному участку жесткого диска. Особенностью является то, что такой файл можно перемещать между каталогами и работоспособность ссылок останется на прежнем уровне, это происходит из-за того, что для таких ссылок неважно имя. В свою очередь символические ссылки при удалении файла остаются, но они будут указывать в никуда, из-за того, что исходного файла больше нет. Выполним создание жесткой и символической ссылки на файл 2.txt и покажем результат работы на рисунке 12.

```

root@ubuntu:/home/user# ls -l
total 12
-rw-rw-rw- 1 root root 27 Oct 14 05:30 1.txt
-rw-r--r-- 1 root root 26 Oct 14 05:30 2.txt
-rw-r--r-- 1 root root 25 Oct 14 05:31 3.txt
root@ubuntu:/home/user# ln 2.txt hard_link
root@ubuntu:/home/user# ln -s 2.txt symb_link
root@ubuntu:/home/user# ls -l
total 16
-rw-rw-rw- 1 root root 27 Oct 14 05:30 1.txt
-rw-r--r-- 2 root root 26 Oct 14 05:30 2.txt
-rw-r--r-- 1 root root 25 Oct 14 05:31 3.txt
-rw-r--r-- 2 root root 26 Oct 14 05:30 hard_link
lrwxrwxrwx 1 root root 5 Oct 14 05:52 symb_link -> 2.txt
root@ubuntu:/home/user# _

```

Рисунок 12 – Создание жесткой и символической ссылки на файл 2.txt

Далее следует создать каталог new в каталоге пользователя user. Для этого выполним команду mkdir название_папки. Результат выполнения данной команды изображен на рисунке 13.

```

root@ubuntu:/home/user# ls -l
total 16
-rw-rw-rw- 1 root root 27 Oct 14 05:30 1.txt
-rw-r--r-- 2 root root 26 Oct 14 05:30 2.txt
-rw-r--r-- 1 root root 25 Oct 14 05:31 3.txt
-rw-r--r-- 2 root root 26 Oct 14 05:30 hard_link
lrwxrwxrwx 1 root root 5 Oct 14 05:52 symb_link -> 2.txt
root@ubuntu:/home/user# mkdir new
root@ubuntu:/home/user# ls -l
total 20
-rw-rw-rw- 1 root root 27 Oct 14 05:30 1.txt
-rw-r--r-- 2 root root 26 Oct 14 05:30 2.txt
-rw-r--r-- 1 root root 25 Oct 14 05:31 3.txt
-rw-r--r-- 2 root root 26 Oct 14 05:30 hard_link
drwxr-xr-x 2 root root 4096 Oct 14 05:56 new
lrwxrwxrwx 1 root root 5 Oct 14 05:52 symb_link -> 2.txt
root@ubuntu:/home/user#

```

Рисунок 13 – Создание каталога new в каталоге пользователя user

После создания каталога new в каталоге пользователя user необходимо скопировать файл 1.txt и переместить файл 2.txt в новый каталог new. Пример выполнения данной операции изображен на рисунке 14.

```

root@ubuntu:/home/user# ls -l
total 20
-rw-rw-rw- 1 root root 27 Oct 14 05:30 1.txt
-rw-r--r-- 2 root root 26 Oct 14 05:30 2.txt
-rw-r--r-- 1 root root 25 Oct 14 05:31 3.txt
-rw-r--r-- 2 root root 26 Oct 14 05:30 hard_link
drwxr-xr-x 2 root root 4096 Oct 14 05:59 new
lrwxrwxrwx 1 root root 5 Oct 14 05:52 symb_link -> 2.txt
root@ubuntu:/home/user# cp 1.txt new/1.txt
root@ubuntu:/home/user# mv 2.txt new/2.txt
root@ubuntu:/home/user# ls -l
total 16
-rw-rw-rw- 1 root root 27 Oct 14 05:30 1.txt
-rw-r--r-- 1 root root 25 Oct 14 05:31 3.txt
-rw-r--r-- 2 root root 26 Oct 14 05:30 hard_link
drwxr-xr-x 2 root root 4096 Oct 14 05:59 new
lrwxrwxrwx 1 root root 5 Oct 14 05:52 symb_link -> 2.txt
root@ubuntu:/home/user# cd new/
root@ubuntu:/home/user/new# ls -l
total 8
-rw-r--r-- 1 root root 27 Oct 14 05:59 1.txt
-rw-r--r-- 2 root root 26 Oct 14 05:30 2.txt
root@ubuntu:/home/user/new#

```

Рисунок 14 – Копирование файла 1.txt и перемещение файла 2.txt в каталог new в каталоге пользователя user

Изменим владельца файла 3.txt и каталога new. Например, изменим на владельца elemabor (основная запись пользователя). Для этого выполним команду `chown имя_пользователя название_папки_или_каталога`. В нашем случае будет выполнена команда `chown elemabor 3.txt` и `chown elemabor new`. Пример выполнения данных команд изображен на рисунке 15.

```

root@ubuntu:/home/user# ls -l
total 16
-rw-rw-rw- 1 root root 27 Oct 14 05:30 1.txt
-rw-r--r-- 1 root root 25 Oct 14 05:31 3.txt
-rw-r--r-- 2 root root 26 Oct 14 05:30 hard_link
drwxr-xr-x 2 root root 4096 Oct 14 05:59 new
lrwxrwxrwx 1 root root 5 Oct 14 05:52 symb_link -> 2.txt
root@ubuntu:/home/user# chown elemabor 3.txt
root@ubuntu:/home/user# chown elemabor new
root@ubuntu:/home/user# ls -l
total 16
-rw-rw-rw- 1 root root 27 Oct 14 05:30 1.txt
-rw-r--r-- 1 elemabor root 25 Oct 14 05:31 3.txt
-rw-r--r-- 2 root root 26 Oct 14 05:30 hard_link
drwxr-xr-x 2 elemabor root 4096 Oct 14 05:59 new
lrwxrwxrwx 1 root root 5 Oct 14 05:52 symb_link -> 2.txt
root@ubuntu:/home/user#

```

Рисунок 15 – Изменение владельца файла 3.txt и каталога new на нового владельца – elemabor

Следующим шагом необходимо удалить файл 1.txt в каталоге new и удалить каталог new. Для удаления файла 1.txt в каталоге new нам необходимо перейти в соответствующий каталог и выполнить команду `rm название_файла`. После проверим содержимое каталога. Далее выйдем из каталога на уровень выше и выполним команду `rm -rf название_каталога`. Параметр `-rf` указывает, что мы собираемся удалить каталог и файлы внутри него. Пример выполнения данного алгоритма изображен на рисунке 16.

```

root@ubuntu:/home/user# cd new/
root@ubuntu:/home/user/new# ls -l
total 8
-rw-r--r-- 1 root root 27 Oct 14 05:59 1.txt
-rw-r--r-- 2 root root 26 Oct 14 05:30 2.txt
root@ubuntu:/home/user/new# rm 1.txt
root@ubuntu:/home/user/new# ls -l
total 4
-rw-r--r-- 2 root root 26 Oct 14 05:30 2.txt
root@ubuntu:/home/user/new# cd ..
root@ubuntu:/home/user# rm -rf new/
root@ubuntu:/home/user# ls -l
total 12
-rw-rw-rw- 1 root root 27 Oct 14 05:30 1.txt
-rw-r--r-- 1 elemabor root 25 Oct 14 05:31 3.txt
-rw-r--r-- 1 root root 26 Oct 14 05:30 hard_link
lrwxrwxrwx 1 root root 5 Oct 14 05:52 symb_link -> 2.txt
root@ubuntu:/home/user# _

```

Рисунок 16 – Пример удаления файла 1.txt в каталоге new и удаление каталога new с его содержимым

Последним пунктом выполнения данной лабораторной работы является нахождение с помощью команды `find` файла `vga2iso`. Результат выполнения данной команды изображен на рисунке 17. Как видно из данного рисунка поиск файла `vga2iso` не дал результатов, файл не был найден. Поэтому в

корневой папке был создан каталог test_dir и в нем был создан файл example_search_file.txt. После этого была выполнена еще раз команда find только уже файла example_search_file.txt, при этом был выведен относительный путь до этого файла.

```
root@ubuntu:/# ls
bin  etc      initrd.img.old  lost+found  opt   run   swapfile  tmp  vmlinuz
boot home    lib             media       proc  sbin  sys       usr  vmlinuz.old
dev  initrd.img  lib64          mnt         root  srv   test_dir  var

root@ubuntu:/# find -name vga2iso
root@ubuntu:/# cd test_dir/
root@ubuntu:/test_dir# ls
example_search_file.txt
root@ubuntu:/test_dir# cd ..
root@ubuntu:/# find -name example_search_file.txt
./test_dir/example_search_file.txt
root@ubuntu:/# _
```

Рисунок 17 – Поиск с помощью команды find

Контрольные вопросы

1. Что такое файловая система?

Файловая система – это структура, с помощью которой ядро операционной системы предоставляет пользователям и процессам ресурсы постоянной памяти системы. В сравнении UNIX-подобных и Windows систем во всех UNIX-подобных системах древовидная структура растет из одного корня – корневого каталога. В то время, в Windows у нас есть разделение на диски.

2. Права доступа к файлам. Название прав доступа

Права доступа к файлам необходимы для того, чтобы разграничить доступ к каталогам и файлам посредством выделения определенных прав (например исполнение, чтение и запись). Для просмотра прав доступа необходимо выполнить команду `ls -l` и название файла или каталога. Будут показаны сообщение, описывающее права доступа. Первым символ означает, что это за файл: обычный файл, каталог или символическая ссылка. Следующие 9 символов обозначают права доступа. Первая группа из трех символов обозначают права доступа владельца файла или каталога (создателя файла или каталога). Вторая группа из трех символов обозначает права доступа на файл или директорию для системной группы. Последняя третья группа из трех символов обозначает права доступа на файл или директорию для всех остальных.

3. Жесткая ссылка в Linux. Основные сведения.

Как ранее было рассмотрено в ходе выполнения лабораторной работы, жесткая ссылка является отдельным файлом, который ведет к одному участку жесткого диска. Данный файл можно перемещать между каталогами и все ссылки на файлы останутся работоспособными. Особенности жесткой ссылки является то, что они работают только в пределах одной файловой системы, ими нельзя ссылаться на каталоги и можно перемещать, переименовывать и удалять исходный файл без вреда ссылки.

4. Команда поиска в Linux. Основные сведения.

Для поиска нужного файла используется команда `find`. Данная команда может искать по имени, размеру, даже создания и по другим критериям. Базовый синтаксис выглядит следующим образом: `find` ключ_поиска критерий_поиска. Среди ключей могут использоваться как имя файла, по которому будет происходить поиск, тип искомого (файл, каталог, ссылка и т.д.), владелец, права доступа и т.д. Также есть возможность перенаправление ввода вывода данной команды, для чтения из файла или запись в него.

5. Перечислите основные команды работы с каталогами.

Среди основных команд работы с каталогами имеются следующие: команда `pwd` – используется для того, чтобы узнать, где располагается каталог, в котором мы сейчас находимся. Просмотр содержимого любой директории осуществляется с помощью команды `ls` и её аргументов. Команда отвечающей за перемещение между каталогами является – `cd` и абсолютный или относительный путь. Для создания каталогов служит команда `mkdir`, а для удаления `rmdir` с различными аргументами в виде флагов. Перемещение и копирование каталогов происходит с помощью команд `mv` и `cp` соответственно.

Вывод

В ходе выполнения данной лабораторной работы были получены навыки работы с файловой системы ОС Linux и производить настройку доступа к файлам и каталогам. Были получены навыки обработки информации о файлах и каталогах, манипулирования файлами и каталогами (создание, удаление, копирование, перемещение и т.д.). Также научился настраивать права доступа к файлам, каталогам и использовать команду для поиска требуемых объектов.