

Отчёт по лабораторной работе №7

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

Седохин Даниил Алексеевич

Содержание

1	Цель работы	4
2	Выполнение лабораторной работы	5
3	Контрольные вопросы	20
4	Выводы	22

Список иллюстраций

2.1	Копирование файлов и каталогов	6
2.2	Перемещение и переименование файлов и каталогов	8
2.3	Изменение прав доступа	10
2.4	Копирование файла /usr/include/sys/io.h	11
2.5	Создание директории, перемещение и переименование	11
2.6	Создание файла abc1 и каталога	11
2.7	Перемещение файлов	12
2.8	Создание и перемещение ~/newdir	12
2.9	Определение команды chmod	13
2.10	/etc/passwd	14
2.11	Упражнения лабораторной работы	15
2.12	mount	16
2.13	fsck	17
2.14	mskf	18
2.15	kill	19

1 Цель работы

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

2 Выполнение лабораторной работы

- 1) Копирование файла в текущем каталоге. Скопировать файл ~/abc1 в файл april и в файл may:

1 cd

2 touch abc1

3 cp abc1 april

4 cp abc1 may

- 1.1) Копирование нескольких файлов в каталог. Скопировать файлы april и may в каталог monthly:

1 mkdir monthly

2 cp april may monthly

- 1.2) Копирование файлов в произвольном каталоге. Скопировать файл monthly/may в файл с именем june:

1 cp monthly/may monthly/june

2 ls monthly

Опция i в команде cp выведет на экран запрос подтверждения о перезаписи файла. Для рекурсивного копирования каталогов, содержащих файлы, используется команда cp с опцией r.

Примеры

- 1.3) Копирование каталогов в текущем каталоге. Скопировать каталог monthly в каталог monthly.00:

1 mkdir monthly.00

2 cp -r monthly monthly.00

1.4) Копирование каталогов в произвольном каталоге. Скопировать каталог monthly.00 в каталог /tmp

1 cp -r monthly.00 /tmp (рис. 2.1).

```
[sdaniil@sdaniil ~]$ cd
[sdaniil@sdaniil ~]$ touch abc1
[sdaniil@sdaniil ~]$ cp abc1 april
[sdaniil@sdaniil ~]$ cp abc1 may
[sdaniil@sdaniil ~]$ ls
abc1          lab.reprot    sf_work       Изображения
april         LICENSE      work          Музыка
Downloads     may          Видео        Общедоступные
git-extended  media        Документы    'Рабочий стол'
lab.pres      newdir       Загрузки     Шаблоны
[sdaniil@sdaniil ~]$ mkdir monthly
[sdaniil@sdaniil ~]$ cp april may monthly
[sdaniil@sdaniil ~]$ ls monthly
april may
[sdaniil@sdaniil ~]$ cp monthly/may monthly/june
[sdaniil@sdaniil ~]$ ls monthly
april june may
[sdaniil@sdaniil ~]$ mkdir monthly.00
[sdaniil@sdaniil ~]$ cp -r monthly monthly.00
[sdaniil@sdaniil ~]$ ls monthly.00/
monthly
[sdaniil@sdaniil ~]$ cp -r monthly.00 /tmp
[sdaniil@sdaniil ~]$ ls /tmp
monthly.00
sddm-auth-b097837d-3843-45cc-b712-f7a3ec89f495
sddm--sIRmrm
systemd-private-d8d26cc26d34468292e3831e63882383-chronyd.servic
```

Рис. 2.1: Копирование файлов и каталогов

2) Перемещение и переименование файлов и каталогов

Команды mv и mvdir предназначены для перемещения и переименования

файлов и каталогов.

Формат команды mv:

1 mv [-опции] старый_файл новый_файл

Примеры:

1. Переименование файлов в текущем каталоге. Изменить название файла april на july в домашнем каталоге:

1 cd

2 mv april july

2. Перемещение файлов в другой каталог. Переместить файл july в каталог monthly.00:

1 mv july monthly.00

2 ls monthly.00

Результат:

1 april july june may

Если необходим запрос подтверждения о перезаписи файла, то нужно использовать опцию i.

3. Переименование каталогов в текущем каталоге. Переименовать каталог monthly.00 в monthly.01

1 mv monthly.00 monthly.01

4. Перемещение каталога в другой каталог. Переместить каталог monthly.01 в каталог reports:

1 mkdir reports

2 mv monthly.01 reports

5. Переименование каталога, не являющегося текущим. Переименовать ка-

талог reports/monthly.01 в reports/monthly:

1 mv reports/monthly.01 reports/monthly (рис. 2.2).

```
[sdaniil@sdaniil ~]$ cd
[sdaniil@sdaniil ~]$ mv april july
[sdaniil@sdaniil ~]$ ls
abc1          LICENSE      sf_work      Музыка
Downloads     may          work         Общедоступные
git-extended  media       Видео       'Рабочий стол'
july          monthly     Документы   Шаблоны
lab.pres      monthly.00  Загрузки
lab.reprot    newdir     Изображения
[sdaniil@sdaniil ~]$ mv july monthly.00
[sdaniil@sdaniil ~]$ ls monthly.00
july  monthly
[sdaniil@sdaniil ~]$ mv monthly.00 monthly.01
[sdaniil@sdaniil ~]$ mkdir reports
[sdaniil@sdaniil ~]$ mv monthly.01 reports
[sdaniil@sdaniil ~]$ ls
abc1          LICENSE      reports      Загрузки      Шаблоны
Downloads     may          sf_work     Изображения
git-extended  media       work        Музыка
lab.pres      monthly     Видео       Общедоступные
lab.reprot    newdir     Документы   'Рабочий стол'
[sdaniil@sdaniil ~]$ mv reports/monthly.01 reports/monthly
[sdaniil@sdaniil ~]$ ls
abc1          LICENSE      reports      Загрузки      Шаблоны
Downloads     may          sf_work     Изображения
git-extended  media       work        Музыка
lab.pres      monthly     Видео       Общедоступные
lab.reprot    newdir     Документы   'Рабочий стол'
[sdaniil@sdaniil ~]$
```

Рис. 2.2: Перемещение и переименование файлов и каталогов

3)1. Требуется создать файл ~/may с правом выполнения для владельца:

1 cd

2 touch may

3 ls -l may

4 chmod u+x may

5 ls -l may

2. Требуется лишить владельца файла ~/may права на выполнение:

1 chmod u-x may

2 ls -l may

3. Требуется создать каталог monthly с запретом на чтение для членов группы и всех остальных пользователей:

1 cd

2 mkdir monthly

3 chmod g-r, o-r monthly

4. Требуется создать файл ~/abc1 с правом записи для членов группы:

1 cd

2 touch abc1

3 chmod g+w abc1 (рис. 2.3).

```

[sdaniil@sdaniil ~]$ cd
[sdaniil@sdaniil ~]$ touch may
[sdaniil@sdaniil ~]$ ls -l may
-rw-r--r--. 1 sdaniil sdaniil 0 map 23 13:02 may
[sdaniil@sdaniil ~]$ chmod u+x may
[sdaniil@sdaniil ~]$ ls -l may
-rwxr--r--. 1 sdaniil sdaniil 0 map 23 13:02 may
[sdaniil@sdaniil ~]$ chmod u-x may
[sdaniil@sdaniil ~]$ ls -l may
-rw-r--r--. 1 sdaniil sdaniil 0 map 23 13:02 may
[sdaniil@sdaniil ~]$ cd
[sdaniil@sdaniil ~]$ mkdir monthly
mkdir: невозможно создать каталог «monthly»: Файл существует
[sdaniil@sdaniil ~]$ chmod g-r, o-r monthly
chmod: неверный режим: «g-r,»
По команде «chmod --help» можно получить дополнительную информа-
цию.
[sdaniil@sdaniil ~]$ rm -R monthly
[sdaniil@sdaniil ~]$
[sdaniil@sdaniil ~]$ cd
[sdaniil@sdaniil ~]$ chmod go-r monthly
chmod: невозможно получить доступ к 'monthly': Нет такого файла
или каталога
[sdaniil@sdaniil ~]$ mkdir monthly
[sdaniil@sdaniil ~]$ chmod go-r monthly
[sdaniil@sdaniil ~]$ cd
[sdaniil@sdaniil ~]$ touch abc1
[sdaniil@sdaniil ~]$ chmod g+w abc1
[sdaniil@sdaniil ~]$

```

Рис. 2.3: Изменение прав доступа

- 4) Скопируем файл /usr/include/sys/io.h в домашний каталог и назовем его equipment. Если файла io.h нет, то используем любой другой файл в каталоге /usr/include/sys/ вместо него. (рис. 2.4)

```
[sdaniil@sdaniil ~]$ cp /usr/include/sys/io.h equipment
[sdaniil@sdaniil ~]$ ls
abc1          LICENSE      sf_work      Музыка
Downloads     may         work         Общедоступные
equipment     media      Видео       'Рабочий стол'
git-extended  monthly    Документы   Шаблоны
lab.pres      newdir     Загрузки
lab.reprot    reports    Изображения
[sdaniil@sdaniil ~]$
```

Рис. 2.4: Копирование файла /usr/include/sys/io.h

5) В домашнем каталоге создадим директорию ~/ski.plases.

Переместм файл equipment в каталог ~/ski.plases.

Переименуем файл ~/ski.plases/equipment в ~/ski.plases/equiplist. (рис. 2.5).

```
[sdaniil@sdaniil ~]$ mkdir ski.plases
[sdaniil@sdaniil ~]$ mv equipment ski.plases
[sdaniil@sdaniil ~]$ mv ski.plases/equipment ski.plases/equipli
st
[sdaniil@sdaniil ~]$ ls ski.plases
equiplist
[sdaniil@sdaniil ~]$
```

Рис. 2.5: Создание директории, перемещение и переименование

6) Создадим в домашнем каталоге файл abc1 и скопируем его в каталог ~/ski.plases, назовем его equiplist2. Создадим каталог с именем equipment в каталоге ~/ski.plases. (рис. 2.6).

```
[sdaniil@sdaniil ~]$ touch abc1
[sdaniil@sdaniil ~]$ cp abc1 ski.plases/equiplist2
```

Рис. 2.6: Создание файла abc1 и каталога

- 7) Переместим файлы ~/ski.plases/equiplist и equiplist2 в каталог ~/ski.plases/equipment.
(рис. 2.7).

```
[sdaniil@sdaniil ~]$ mkdir ski.plases/equipment
[sdaniil@sdaniil ~]$ mv ski.plases/equiplist ski.plases/equipli
st2 ski.plases/equipment/
[sdaniil@sdaniil ~]$
```

Рис. 2.7: Перемещение файлов

- 8) Создадим и переместим каталог ~/newdir в каталог ~/ski.plases и назовем его plans. (рис. 2.8).

```
[sdaniil@sdaniil ~]$ mkdir newdir
mkdir: невозможно создать каталог «newdir»: Файл существует
[sdaniil@sdaniil ~]$ mv newdir ski.plases/plans
[sdaniil@sdaniil ~]$
```

Рис. 2.8: Создание и перемещение ~/newdir

- 9) Определим опции команды chmod, необходимые для того, чтобы присвоить перечисленным ниже файлам выделенные права доступа, считая, что в начале таких прав нет:

drwxr-r- ... australia

drwx-x-x ... play

-r-xr-r- ... my_os

-rw-rw-r- ... feathers (рис. 2.9).

```
[sdaniil@sdaniil ~]$ mkdir australia
[sdaniil@sdaniil ~]$ mkdir play
[sdaniil@sdaniil ~]$ touch my_os
[sdaniil@sdaniil ~]$ touch feathers
[sdaniil@sdaniil ~]$ chmod 744 australia/
[sdaniil@sdaniil ~]$ chmod 711 play/
[sdaniil@sdaniil ~]$ chmod 544 my_os
[sdaniil@sdaniil ~]$ chmod 664 feathers
[sdaniil@sdaniil ~]$
```

Рис. 2.9: Определение команды chmod

10) Просмотрим содержимое файла /etc/passwd (рис. 2.10).

```
[sdaniil@sdaniil ~]$ cat /etc/passwd
root:x:0:0:Super User:/root:/bin/bash
bin:x:1:1:bin:/bin:/usr/sbin/nologin
daemon:x:2:2:daemon:/sbin:/usr/sbin/nologin
adm:x:3:4:adm:/var/adm:/usr/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/usr/sbin/nologin
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/var/spool/mail:/usr/sbin/nologin
operator:x:11:0:operator:/root:/usr/sbin/nologin
games:x:12:100:games:/usr/games:/usr/sbin/nologin
ftp:x:14:50:FTP User:/var/ftp:/usr/sbin/nologin
nobody:x:65534:65534:Kernel Overflow User:/:/usr/sbin/nologin
dbus:x:81:81:System Message Bus:/:/usr/sbin/nologin
tss:x:59:59:Account used for TPM access:/:/usr/sbin/nologin
systemd-coredump:x:998:998:systemd Core Dumper:/:/usr/sbin/nologin
systemd-network:x:192:192:systemd Network Management:/:/usr/sbin/nologin
systemd-oom:x:997:997:systemd Userspace OOM Killer:/:/usr/sbin/nologin
systemd-resolve:x:193:193:systemd Resolver:/:/usr/sbin/nologin
systemd-timesync:x:996:996:systemd Time Synchronization:/:/usr/
```

Рис. 2.10: /etc/passwd

11) Скопируем файл ~/feathers в файл ~/file.old.

Переместим файл ~/file.old в каталог ~/play.

Скопируем каталог ~/play в каталог ~/fun.

Переместим каталог ~/fun в каталог ~/play и назовем его games.

Лишим владельца файла ~/feathers права на чтение.

В результате попытки просмотра файла нам отказано в доступе ~/feathers

Дадим владельцу файла ~/feathers право на чтение. Лишим владельца ката-

лога ~/play права на выполнение. Перейдем в каталог ~/play. В результате нам отказано в доступе. Дадим владельцу каталога ~/play право на выполнение. (рис. 2.11).

```
[sdaniil@sdaniil ~]$ cp feathers file.old
[sdaniil@sdaniil ~]$ mv file.old play/
[sdaniil@sdaniil ~]$ cp -r play/ fun/
[sdaniil@sdaniil ~]$ mv fun play/games
[sdaniil@sdaniil ~]$ chmod u-r feathers
[sdaniil@sdaniil ~]$ cp feathers play/
cp: невозможно открыть 'feathers' для чтения: Отказано в доступе
[sdaniil@sdaniil ~]$ chmod u+r feathers
[sdaniil@sdaniil ~]$ chmod u-x play/
[sdaniil@sdaniil ~]$ cd play/
bash: cd: play/: Отказано в доступе
[sdaniil@sdaniil ~]$ chmod u+x play/
[sdaniil@sdaniil ~]$
```

Рис. 2.11: Упражнения лабораторной работы

12) Прочитаем man по командам mount, fsck, mskf, kill. (рис. 2.12 2.13 2.14 2.15).

```
MOUNT(8)                      System Administration                      MOUNT(8)

NAME
    mount - mount a filesystem

SYNOPSIS
    mount [-h|-V]

    mount [-l] [-t fstype]

    mount -a [-fFnrsvw] [-t fstype] [-O optlist]

    mount [-fnrsvw] [-o options] device|mountpoint

    mount [-fnrsvw] [-t fstype] [-o options] device
mountpoint

    mount --bind|--rbind|--move olddir newdir

    mount
    --make-[shared|slave|private|unbindable|rshared|rslave|r
private|runbindable]
    mountpoint

DESCRIPTION
```

Рис. 2.12: mount

NAME

`fsck` - check and repair a Linux filesystem

SYNOPSIS

```
fsck [-lsAVRTMNP] [-r [fd]] [-C [fd]] [-t fstype]  
[filesystem...] [--] [fs-specific-options]
```

DESCRIPTION

`fsck` is used to check and optionally repair one or more Linux filesystems. `filesystem` can be a device name (e.g., `/dev/hdc1`, `/dev/sdb2`), a mount point (e.g., `/`, `/usr`, `/home`), or a filesystem label or UUID specifier (e.g., `UUID=8868abf6-88c5-4a83-98b8-bfc24057f7bd` or `LABEL=root`). Normally, the `fsck` program will try to handle filesystems on different physical disk drives in parallel to reduce the total amount of time needed to check all of them.

Рис. 2.13: `fsck`

MKFS(8)	System Administration	MKFS(8)
NAME mkfs - build a Linux filesystem		
SYNOPSIS mkfs [options] [-t <u>type</u>] [<u>fs-options</u>] <u>device</u> [<u>size</u>]		
DESCRIPTION <p>This mkfs frontend is deprecated in favour of filesystem specific mkfs.<type> utils.</p> <p>mkfs is used to build a Linux filesystem on a device, usually a hard disk partition. The <u>device</u> argument is either the device name (e.g., <u>/dev/hda1</u>, <u>/dev/sdb2</u>), or a regular file that shall contain the filesystem. The <u>size</u> argument is the number of blocks to be used for the filesystem.</p> <p>The exit status returned by mkfs is 0 on success and 1 on failure.</p> <p>In actuality, mkfs is simply a front-end for the various filesystem builders (<u>mkfs.fstype</u>) available under Linux. The filesystem-specific builder is searched for via your PATH environment setting only.</p>		

Рис. 2.14: mskf

NAME

kill - terminate a process

SYNOPSIS

```
kill [-signal|-s signal|-p] [-q value] [-a]
[--timeout milliseconds signal] [--] pid|name...
```

```
kill -l [number] | -L
```

DESCRIPTION

The command **kill** sends the specified signal to the specified processes or process groups.

If no signal is specified, the **TERM** signal is sent. The default action for this signal is to terminate the process. This signal should be used in preference to the **KILL** signal (number 9), since a process may install a handler for the **TERM** signal in order to perform clean-up steps before terminating in an orderly fashion. If a process does not terminate after a **TERM** signal has been sent, then the **KILL** signal may be used; be aware that the latter signal cannot be caught, and so does not give the target process the opportunity to perform any clean-up before terminating.

Рис. 2.15: kill

3 Контрольные вопросы

1. На моем компьютере используются несколько файловых систем. Например, на жестком диске есть файловая система NTFS, которая обладает хорошей поддержкой различных функций и контроля доступа к файлам. Также есть файловая система FAT32, которая поддерживается многими устройствами, но имеет ограничения на размер файлов и разделов. Кроме того, используется файловая система ext4, распространенная в Linux и обладающая хорошей производительностью и надежностью.
2. Общая структура файловой системы включает в себя директории, файлы и разделы. На первом уровне структуры обычно располагаются директории, такие как “home” (домашние папки пользователей), “bin” (папка с исполняемыми файлами), “etc” (конфигурационные файлы), “var” (переменные данные), “tmp” (временные файлы).
3. Для того чтобы содержимое файловой системы стало доступно операционной системе, необходимо выполнить операцию монтирования (mount), которая подключит файловую систему к дереву каталогов операционной системы.
4. Основными причинами нарушения целостности файловой системы могут быть сбои в питании, ошибки записи/чтения данных, вирусы или повреждения файлов сторонними программами. Для устранения повреждений

файловой системы можно использовать инструменты проверки и восстановления данных, такие как `fsck` в Linux.

5. Файловая система создается путем форматирования раздела на диске, то есть установки определенного типа файловой системы на этот раздел. Например, при создании файловой системы `ext4` на разделе используется команда `mkfs.ext4`.
6. Для просмотра текстовых файлов в Linux часто используются команды `cat`, `less`, `more`, `head`, `tail`. Например, команда `cat` выводит содержимое файла на экран, команда `less` позволяет просматривать файл постранично.
7. Основные возможности команды `cp` в Linux включают копирование файлов и директорий с сохранением их атрибутов и разрешений, создание копий файлов, а также возможность рекурсивного копирования для директорий.
8. Команда `mv` в Linux используется для перемещения файлов и директорий из одного места в другое. При перемещении файлов с помощью `mv`, информация о файле остается неизменной, но путь к нему меняется.
9. Права доступа определяют, какие операции пользователи и группы могут выполнять с файлами и директориями. Права доступа могут быть изменены с помощью команды `chmod` в Linux, позволяющей устанавливать различные разрешения на чтение, запись и выполнение для владельца файла, группы и других пользователей.

4 Выводы

Я ознакомился с файловой системой Linux, её структурой, именами и содержанием каталогов. Приобрел практические навыки по применению команд для работы с файлами и каталогами, по управлению процессами (и работами), по проверке использования диска и обслуживанию файловой системы.