

Отчёт по лабораторной работе №7

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

Джаллох Ишмаил

Содержание

1	Цель работы	4
2	Задание	5
3	Выполнение лабораторной работы	6
3.1	Команды для работы с файлами и каталогами	6
4	Выводы	14
4.1	Ответы на контрольные вопросы	14

Список иллюстраций

3.1	Создание файлов abc1, april и may	6
3.2	Создание monthly	6
3.3	копирование файла may	6
3.4	копирование каталога monthly	7
3.5	копирование каталога monthly.00	7
3.6	Перемещение файла july	7
3.7	Перемещение и переименование каталога monthly.00	7
3.8	переименование каталога reports/monthly.01	7
3.9	Создание equipment	7
3.10	Проверка создания ski.plases	8
3.11	Перемещение файла equipment	8
3.12	Переименование файла /equipment	8
3.13	Создание каталога equipment, перемещение файлов	8
3.14	Создание и перемещение каталога ~/newdir	8
3.15	Создание australia	9
3.16	Удаление права	9
3.17	Изменение прав	9
3.18	Проверка изменений	9
3.19	Изменение прав к файлу feathers	9
3.20	команда cat	10
3.21	Перемещение и копирование файлов и каталогов	10
3.22	Перемещение каталога fun	10
3.23	Лишение права на чтение	10
3.24	Посмотра файла feathers	10
3.25	Лишение права на выполнение	11
3.26	Название рисунка	11
3.27	mount	11
3.28	fsck	12
3.29	mkfs	12
3.30	Kill	13

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 Задание

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

3 Выполнение лабораторной работы

3.1 Команды для работы с файлами и каталогами

Создаю файл abc1 с помощью touch и копирую его с новыми именами april и may используя cp:

```
ijalloh@ijalloh:~$ cd
ijalloh@ijalloh:~$ touch abc1
ijalloh@ijalloh:~$ cp abc1 april
ijalloh@ijalloh:~$ cp abc1 may
```

Рис. 3.1: Создание файлов abc1, april и may

Создаю каталог monthly и копирую april и may в нем используя cp. Проверяю с ls:

```
ijalloh@ijalloh:~$ mkdir monthly
ijalloh@ijalloh:~$ cp april may monthly
ijalloh@ijalloh:~$ ls
abc1  Desktop  git-extended  may    Pictures  texput.log
april  Documents go            monthly Public    Videos
bin    Downloads LICENSE      Music  Templates work
ijalloh@ijalloh:~$ ls monthly/
april may
ijalloh@ijalloh:~$
```

Рис. 3.2: Создание monthly

В каталоге monthly копирую файл may с именем june используя cp:

```
ijalloh@ijalloh:~$ cp monthly/may monthly/june
ijalloh@ijalloh:~$ ls monthly
april june may
ijalloh@ijalloh:~$
```

Рис. 3.3: копирование файла may

Копирую каталог monthly в каталог monthly.00 с помощью опции cp -r:

```
ijalloh@ijalloh:~$ mkdir monthly.00
ijalloh@ijalloh:~$ cp -r monthly monthly.00
```

Рис. 3.4: копирование каталога monthly

Копирую каталог monthly.00 в каталог /tmp:

```
ijalloh@ijalloh:~$ cp -r monthly.00 /tmp
ijalloh@ijalloh:~$
```

Рис. 3.5: копирование каталога monthly.00

Изменяю название файла april на july в домашнем каталоге и перемещаю файл july в каталог monthly.00:

```
ijalloh@ijalloh:~$ mv april july
ijalloh@ijalloh:~$ mv july monthly.00
ijalloh@ijalloh:~$ ls monthly.00
july  monthly
ijalloh@ijalloh:~$
```

Рис. 3.6: Перемещение файла july

Переименовываю каталог monthly.00 в monthly.01. Перемещаю каталог monthly.01 в каталог reports:

```
ijalloh@ijalloh:~$ mv monthly.00 monthly.01
ijalloh@ijalloh:~$ mkdir reports
ijalloh@ijalloh:~$ mv monthly.01 reports
ijalloh@ijalloh:~$
```

Рис. 3.7: Перемещение и переименование каталога monthly.00

Переименовываю каталог reports/monthly.01 в reports/monthly:

```
ijalloh@ijalloh:~$ mv reports/monthly.01 reports/monthly
ijalloh@ijalloh:~$
```

Рис. 3.8: переименование каталога reports/monthly.01

Копирую файл /usr/include/sys/io.h в домашний каталог и назову его equipment:

```
ijalloh@ijalloh:~$ cp /usr/include/sys/io.h equipment
ijalloh@ijalloh:~$ ls
abcl1  Documents  git-extended  may  Pictures  Templates  work
bin    Downloads  go            monthly  Public    texput.log
Desktop  equipment  LICENSE      Music   reports  Videos
ijalloh@ijalloh:~$
```

Рис. 3.9: Создание equipment

В домашнем каталоге создаю директорию ~/ski.plases с помощью mkdir:

```
ijalloh@ijalloh:~$ mkdir ski.please
ijalloh@ijalloh:~$ ls
abc1      Documents  git-extended  may      Pictures  ski.please  Videos
bin       Downloads  go            monthly  Public    Templates  work
```

Рис. 3.10: Проверка создания ski.plases

Перемещаю файл equipment в каталог ~/ski.plases:

```
ijalloh@ijalloh:~$ mv equipment ski.please
ijalloh@ijalloh:~$ ls ski.please
equipment
ijalloh@ijalloh:~$
```

Рис. 3.11: Перемещение файла equipment

Переименую файл ~/ski.plases/equipment в ~/ski.plases/equiplist и копирую abc1 в каталог ~/ski.plases, назову его equiplist2:

```
ijalloh@ijalloh:~$ mv ski.please/equipment ski.please/equiplist
ijalloh@ijalloh:~$ cp abc1 ski.please/equiplist2
```

Рис. 3.12: Переименование файла /equipment

Создаю каталог с именем equipment в каталоге ~/ski.plases и перемещаю файлы ~/ski.plases/equiplist и equiplist2 в каталог ~/ski.plases/equipment:

```
ijalloh@ijalloh:~$ mkdir ski.please/equipment
ijalloh@ijalloh:~$ mv ski.please/equiplist ski.please/equiplist2 ski.please/equipment/
ijalloh@ijalloh:~$ ls ski.please/equipment/
equiplist equiplist2
ijalloh@ijalloh:~$
```

Рис. 3.13: Создание каталога equipment, перемещение файлов

Создаю и перемещаю каталог ~/newdir в каталог ~/ski.plases и назову его plans:

```
ijalloh@ijalloh:~$ mkdir newdir
ijalloh@ijalloh:~$ mv newdir ski.please/plans
ijalloh@ijalloh:~$ ls ski.please
equipment plans
ijalloh@ijalloh:~$
```

Рис. 3.14: Создание и перемещение каталога ~/newdir

Создаю каталог australia. Удаляю права на исполнение для группы (g-x) и владельца(u-x):


```

ijalloh@ijalloh:~$ mkdir australia
ijalloh@ijalloh:~$ ls -l
total 24
-rw-rw-r--. 1 ijalloh ijalloh    0 Mar 26 02:02 abc1
drwxr-xr-x. 1 ijalloh ijalloh    0 Mar 26 02:09 australia

```

Рис. 3.15: Создание australia

```

ijalloh@ijalloh:~$ chmod o-x australia
ijalloh@ijalloh:~$ chmod g-x australia
ijalloh@ijalloh:~$ ls -l
total 24
-rw-rw-r--. 1 ijalloh ijalloh    0 Mar 26 02:02 abc1
drwxr--r--. 1 ijalloh ijalloh    0 Mar 26 02:18 australia

```

Рис. 3.16: Удаление права

Изменяю права доступа к каталогу play и проверяю:

```

ijalloh@ijalloh:~$ chmod u+x play
ijalloh@ijalloh:~$ chmod g-r+x play
ijalloh@ijalloh:~$ chmod o-r+x play
ijalloh@ijalloh:~$ ls -l
total 24

```

Рис. 3.17: Изменение прав

```

drwx--x--x. 1 ijalloh ijalloh    0 Mar 26 02:12 play

```

Рис. 3.18: Проверка изменений

Изменяю права доступа к файлу feathers и проверяю:

```

ijalloh@ijalloh:~$ chmod g+w feathers
ijalloh@ijalloh:~$ ls -l
total 24
-rw-rw-r--. 1 ijalloh ijalloh    0 Mar 26 02:02 abc1
drwxr--r--. 1 ijalloh ijalloh    0 Mar 26 02:18 australia
drwxr-xr-x. 1 ijalloh ijalloh   22 Mar 17 10:26 bin
drwxr-xr-x. 1 ijalloh ijalloh    0 Feb 28 04:34 Desktop
drwxr-xr-x. 1 ijalloh ijalloh    0 Feb 28 04:34 Documents
drwxr-xr-x. 1 ijalloh ijalloh  614 Mar 22 21:30 Downloads
drwxrwxr-x. 1 ijalloh ijalloh    0 Mar 26 02:12 feathers

```

Рис. 3.19: Изменение прав к файлу feathers

Смотрю содержимое файла /etc/passwd:

```

ijalloh@ijalloh:~$ 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

```

Рис. 3.20: команда cat

Копирую файл ~/feathers в файл ~/file.old, перемещаю файл ~/file.old в каталог ~/play и копирую каталог ~/play в каталог ~/fun:

```

ijalloh@ijalloh:~$ cp feathers file.old
ijalloh@ijalloh:~$ mv file.old play/
ijalloh@ijalloh:~$ cp play fun
cp: -r not specified; omitting directory 'play'
ijalloh@ijalloh:~$ cp -r play fun

```

Рис. 3.21: Перемещение и копирование файлов и каталогов

Перемещаю каталог ~/fun в каталог ~/play и назову его games:

```

ijalloh@ijalloh:~$ cp -r fun play/games
ijalloh@ijalloh:~$ ls -l play
total 0
-rw-r--r--. 1 ijalloh ijalloh  0 Mar 26 02:33 file.old
drwx--x--x. 1 ijalloh ijalloh 16 Mar 26 02:34 games
ijalloh@ijalloh:~$

```

Рис. 3.22: Перемещение каталога fun

Лишаю пользователя файла ~/feathers права на чтение:

```

ijalloh@ijalloh:~$ chmod u-r feathers
ijalloh@ijalloh:~$ ls -l feathers
--w-rw-r--. 1 ijalloh ijalloh 0 Mar 26 02:31 feathers
ijalloh@ijalloh:~$

```

Рис. 3.23: Лишение права на чтение

Когда я попытаюсь просмотреть файл ~/feathers командой cat, система запрещает мне:

```

ijalloh@ijalloh:~$ cat feathers
cat: feathers: Permission denied
ijalloh@ijalloh:~$

```

Рис. 3.24: Посмотра файла feathers

Лишаю владельца каталога ~/play права на выполнение. Когда я попробую перейти в этот же каталог, система запрещает мне:

```
ijalloh@ijalloh:~$ chmod u-x play
ijalloh@ijalloh:~$ cd play
bash: cd: play: Permission denied
ijalloh@ijalloh:~$
```

Рис. 3.25: Лишение права на выполнение

Даю владельцу каталога ~/play право на выполнение:

```
ijalloh@ijalloh:~$ chmod u+x play
ijalloh@ijalloh:~$
```

Рис. 3.26: Название рисунка

С помощью man прочитаю по следующим командам: mount — утилита командной строки в UNIX-подобных операционных системах. Применяется для монтирования файловых систем.

```
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|rprivate|runbinda
ble]
    mountpoint

DESCRIPTION
    All files accessible in a Unix system are arranged in one big tree, the
    file hierarchy, rooted at /. These files can be spread out over several
    devices. The mount command serves to attach the filesystem found on
    some device to the big file tree. Conversely, the umount(8) command
```

Рис. 3.27: mount

fsck (проверка файловой системы) - это утилита командной строки, которая позволяет выполнять проверки согласованности и интерактивное исправление

в одной или нескольких файловых системах Linux. Он использует программы, специфичные для типа файловой системы, которую он проверяет.

```
FSCK(8)                                System Administration                                FSCK(8)

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. 3.28: fsck

mkfs используется для создания файловой системы Linux на некотором устройстве, обычно в разделе жёсткого диска. В качестве аргумента filesystem для файловой системы может выступить или название устройства

```
MKFS(8)                                System Administration                                MKFS(8)

NAME
    mkfs - build a Linux filesystem

SYNOPSIS
    mkfs [options] [-t type] [fs-options] device [size]

DESCRIPTION
    This mkfs frontend is deprecated in favour of filesystem specific
    mkfs.<type> utils.

    mkfs is used to build a Linux filesystem on a device, usually a hard
    disk partition. The device argument is either the device name (e.g.,
    /dev/hda1, /dev/sdb2), or a regular file that shall contain the
    filesystem. The size argument is the number of blocks to be used for
    the filesystem.
```

Рис. 3.29: mkfs

Команда Kill посылает указанный сигнал указанному процессу. Если не указано ни одного сигнала, посылается сигнал SIGTERM. Сигнал SIGTERM завершает лишь те процессы, которые не обрабатывают его приход. Для других процессов может быть необходимым послать сигнал SIGKILL, поскольку этот сигнал перехватить невозможно.

KILL(1)	User Commands	KILL(1)
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 <u>signal</u> 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 it is terminated.		

Рис. 3.30: Kill

4 Выводы

При выполнении данной лабораторной работы я ознакомилась 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.

4.1 Ответы на контрольные вопросы

1. Ext2, Ext3, Ext4 или Extended Filesystem - это стандартная файловая система для Linux. Она была разработана еще для Minix. Она самая стабильная из всех существующих, кодовая база изменяется очень редко и эта файловая система содержит больше всего функций. Версия ext2 была разработана уже именно для Linux и получила много улучшений. В 2001 году вышла ext3, которая добавила еще больше стабильности благодаря использованию журналирования. В 2006 была выпущена версия ext4, которая используется во всех дистрибутивах Linux до сегодняшнего дня. В ней было внесено много улучшений, в том числе увеличен максимальный размер раздела до одного экзабайта.

Btrfs или B-Tree File System - это совершенно новая файловая система, которая сосредоточена на отказоустойчивости, легкости администрирования и восстановления данных. Файловая система объединяет в себе очень много новых

интересных возможностей, таких как размещение на нескольких разделах, поддержка подтомов, изменение размера не лету, создание мгновенных снимков, а также высокая производительность. Но многими пользователями файловая система Btrfs считается нестабильной. Тем не менее, она уже используется как файловая система по умолчанию в OpenSUSE и SUSE Linux.

2. / — root каталог. Содержит в себе всю иерархию системы;

/bin — здесь находятся двоичные исполняемые файлы. Основные общие команды, хранящиеся отдельно от других программ в системе (прим.: pwd, ls, cat, ps);

/boot — тут расположены файлы, используемые для загрузки системы (образ initrd, ядро vmlinuz);

/dev — в данной директории располагаются файлы устройств (драйверов). С помощью этих файлов можно взаимодействовать с устройствами. К примеру, если это жесткий диск, можно подключить его к файловой системе. В файл принтера же можно написать напрямую и отправить задание на печать;

/etc — в этой директории находятся файлы конфигураций программ. Эти файлы позволяют настраивать системы, сервисы, скрипты системных демонов;

/home — каталог, аналогичный каталогу Users в Windows. Содержит домашние каталоги учетных записей пользователей (кроме root). При создании нового пользователя здесь создается одноименный каталог с аналогичным именем и хранит личные файлы этого пользователя;

/lib — содержит системные библиотеки, с которыми работают программы и модули ядра;

/lost+found — содержит файлы, восстановленные после сбоя работы системы. Система проведет проверку после сбоя и найденные файлы можно будет посмотреть в данном каталоге;

/media — точка монтирования внешних носителей. Например, когда вы вставляете диск в дисковод, он будет автоматически смонтирован в директорию /media/cdrom;

/mnt — точка временного монтирования. Файловые системы подключаемых устройств обычно монтируются в этот каталог для временного использования;

/opt — тут расположены дополнительные (необязательные) приложения. Такие программы обычно не подчиняются принятой иерархии и хранят свои файлы в одном подкаталоге (бинарные, библиотеки, конфигурации);

/proc — содержит файлы, хранящие информацию о запущенных процессах и о состоянии ядра ОС;

/root — директория, которая содержит файлы и личные настройки суперпользователя;

/run — содержит файлы состояния приложений. Например, PID-файлы или UNIX-сокеты;

/sbin — аналогично /bin содержит бинарные файлы. Утилиты нужны для настройки и администрирования системы суперпользователем;

/srv — содержит файлы сервисов, предоставляемых сервером (прим. FTP или Apache HTTP);

/sys — содержит данные непосредственно о системе. Тут можно узнать информацию о ядре, драйверах и устройствах;

/tmp — содержит временные файлы. Данные файлы доступны всем пользователям на чтение и запись. Стоит отметить, что данный каталог очищается при перезагрузке;

/usr — содержит пользовательские приложения и утилиты второго уровня, используемые пользователями, а

не системой. Содержимое доступно только для чтения (кроме root). Каталог имеет вторичную иерархию и похож на корневой;

/var — содержит переменные файлы. Имеет подкаталоги, отвечающие за отдельные переменные. Например, логи будут храниться в /var/log, кэш в /var/cache, очереди заданий в /var/spool/ и так далее.

3. Монтирование тома.

4. Один блок адресуется несколькими mode (принадлежит нескольким файлам). Блок помечен как свободный, но в то же время занят (на него ссылается onode). Блок помечен как занятый, но в то же время свободен (ни один inode на него не ссылается). Неправильное число ссылок в inode (недостаток или избыток ссылающихся записей в каталогах). Несовпадение между размером файла и суммарным размером адресуемых inode блоков. Недопустимые адресуемые блоки (например, расположенные за пределами файловой системы). “Потерянные” файлы (правильные inode, на которые не ссылаются записи каталогов). Недопустимые или неразмещенные номера inode в записях каталогов.
5. mkfs - позволяет создать файловую систему Linux.
6. Cat - выводит содержимое файла на стандартное устройство вывода. Выполнение команды head выведет первые 10 строк текстового файла. Выполнение команды tail выведет последние 10 строк текстового файла. Команда tac - это тоже самое, что и cat, только отображает строки в обратном порядке. Для того, чтобы просмотреть огромный текстовый файл применяются команды для постраничного просмотра. Такие как more и less.
7. Cp – копирует или перемещает директорию, файлы.
8. Mv - переименовать или переместить файл или директорию
9. Права доступа к файлу или каталогу можно изменить, воспользовавшись командой chmod. Сделать это может владелец файла (или каталога) или пользователь с правами администратора.