

Лабораторная работа № 5

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

Шулуужук Айраана Вячеславовна НПИбд-02-22

Содержание

1	Цель работы	5
2	Задание	6
3	Теоретическое введение	8
3.1	Команды для работы с файлами и каталогами	8
3.2	Копирование файлов и каталогов	9
3.3	Перемещение и переименование файлов и каталогов	9
3.4	Права доступа	9
3.5	Изменение прав доступа	10
3.6	Анализ файловой системы	10
4	Выполнение лабораторной работы	12
5	Контрольные вопросы	18
6	Выводы	23

Список иллюстраций

4.1	файл equipment	12
4.2	перемещение файла equipment в каталог ski.places	12
4.3	файл equiplist	12
4.4	каталог ski.places	13
4.5	создание каталога equipment в каталоге ski.places	13
4.6	перенос файлов equiplist и equiplist2 в каталог equipment	13
4.7	каталог plans в каталоге ski.places	13
4.8	создание необходимых файлов и каталогов	14
4.9	изменение прав для файлов и каталогов	14
4.10	файл passwd	14
4.11	копирование файла feathers в файл file.old	15
4.12	перемещение файла file.old в каталог play	15
4.13	копирование каталога play в каталог fun.	15
4.14	каталог play	15
4.15	изменение прав доступа на чтение к файлу feathers	16
4.16	изменение прав доступа на выполнение к каталогу play	16
4.17	man mount	16
4.18	man fsck	17
4.19	man mkfs	17
4.20	man kill	17

Список таблиц

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. Выполните все примеры, приведённые в первой части описания лабораторной работы

2. Выполните следующие действия, зафиксировав в отчёте по лабораторной работе используемые при этом команды и результаты их выполнения:

2.1. Скопируйте файл `/usr/include/sys/io.h` в домашний каталог и назовите его `equipment`. Если файла `io.h` нет, то используйте любой другой файл в каталоге `/usr/include/sys/` вместо него.

2.2. В домашнем каталоге создайте директорию `~/ski.places`.

2.3. Переместите файл `equipment` в каталог `~/ski.places`.

2.4. Переименуйте файл `~/ski.places/equipment` в `~/ski.places/equiplist`.

2.5. Создайте в домашнем каталоге файл `abc1` и скопируйте его в каталог `~/ski.places`, назовите его `equiplist2`.

2.6. Создайте каталог с именем `equipment` в каталоге `~/ski.places`.

2.7. Переместите файлы `~/ski.places/equiplist` и `equiplist2` в каталог `~/ski.places/equipment`.

2.8. Создайте и переместите каталог `~/newdir` в каталог `~/ski.places` и назовите его `plans`.

3. Определите опции команды `chmod`, необходимые для того, чтобы присвоить перечисленным ниже файлам выделенные права доступа, считая, что в начале таких прав нет:

3.1. drwxr-r- ... australia

3.2. drwx-x-x ... play

3.3. -r-xr-r- ... my_os

3.4. -rw-rw-r- ... feathers

4. Прodelайте приведённые ниже упражнения, записывая в отчёт по лабораторной работе используемые при этом команды:

4.1. Просмотрите содержимое файла /etc/password.

4.2. Скопируйте файл ~/feathers в файл ~/file.old.

4.3. Переместите файл ~/file.old в каталог ~/play.

4.4. Скопируйте каталог ~/play в каталог ~/fun.

4.5. Переместите каталог ~/fun в каталог ~/play и назовите его games.

4.6. Лишите владельца файла ~/feathers права на чтение.

4.7. Что произойдёт, если вы попытаетесь просмотреть файл ~/feathers командой cat?

4.8. Что произойдёт, если вы попытаетесь скопировать файл ~/feathers?

4.9. Дайте владельцу файла ~/feathers право на чтение.

4.10. Лишите владельца каталога ~/play права на выполнение.

4.11. Перейдите в каталог ~/play. Что произошло?

4.12. Дайте владельцу каталога ~/play право на выполнение.

5. Прочитайте man по командам mount, fsck, mkfs, kill и кратко их охарактеризуйте, приведя примеры.

3 Теоретическое введение

3.1 Команды для работы с файлами и каталогами

Для создания текстового файла можно использовать команду `touch`. Формат команды:

```
1 touch имя-файла
```

Для просмотра файлов постранично удобнее использовать команду `less`. Формат команды:

```
less имя-файла
```

Следующие клавиши используются для управления процессом просмотра:

- Space — переход к следующей странице,
- ENTER — сдвиг вперёд на одну строку,
- b — возврат на предыдущую страницу,
- h — обращение за подсказкой,
- q — выход из режима просмотра файла.

Команда `head` выводит по умолчанию первые 10 строк файла. Формат команды:

```
head [-n] имя-файла ,
```

где `n` — количество выводимых строк.

Команда `tail` выводит по умолчанию 10 последних строк файла. Формат команды:

```
tail [-n] имя-файла ,
```

где `n` — количество выводимых строк.

3.2 Копирование файлов и каталогов

Команда `cp` используется для копирования файлов и каталогов. Формат команды:

```
1 cp [-опции] исходный_файл целевой_файл
```

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`.

3.3 Перемещение и переименование файлов и каталогов

Команды `mv` и `mkdir` предназначены для перемещения и переименования файлов и каталогов. Формат команды `mv`

```
mv [-опции] старый_файл новый_файл
```

3.4 Права доступа

Каждый файл или каталог имеет права доступа

В сведениях о файле или каталоге указываются:

- тип файла (символ `-` обозначает файл, а символ `d` — каталог);
- права для владельца файла (`r` — разрешено чтение, `w` — разрешена запись, `x` — разрешено выполнение, `-` — право доступа отсутствует);
- права для членов группы (`r` — разрешено чтение, `w` — разрешена запись, `x` — разрешено выполнение, `-` — право доступа отсутствует);
- права для всех остальных (`r` — разрешено чтение, `w` — разрешена запись, `x` — разрешено выполнение, `-` — право доступа отсутствует).

3.5 Изменение прав доступа

Права доступа к файлу или каталогу можно изменить, воспользовавшись командой `chmod`. Сделать это может владелец файла (или каталога) или пользователь с правами администратора. Формат команды:

`chmod режим имя_файла`

Режим (в формате команды) имеет следующие компоненты структуры и способ записи:

= установить право

- лишить права
- дать право

r чтение

w запись

x выполнение

u (user) владелец файла

g (group) группа, к которой принадлежит владелец файла

o (others) все остальные

В работе с правами доступа можно использовать их цифровую запись (восьмеричное значение) вместо символьной

3.6 Анализ файловой системы

Файловая система в Linux состоит из файлов и каталогов. Каждому физическому носителю соответствует своя файловая система. Существует несколько типов файловых систем. Перечислим наиболее часто встречающиеся типы:

- ext2fs (second extended filesystem);
- ext3fs (third extended file system);

- ext4 (fourth extended file system);
- ReiserFS;
- xfs;
- fat (file allocation table);
- ntfs (new technology file system).

4 Выполнение лабораторной работы

Копируем файл `/usr/include/sys/io.h` в домашний каталог и назовем его `equipment`, используя команду `cp` (рис. 4.1)

```
[avshuluuzhuk@fedora ~]$ cp /usr/include/sys/io.h ~/equipment
[avshuluuzhuk@fedora ~]$ ls
bin          work  Документы  Изображения  Общедоступные  Шаблоны
equipment    Видео  Загрузки   Музыка        'Рабочий стол'
```

Рис. 4.1: файл `equipment`

Создаем каталог `ski.places` и переместим в созданную директорию файл `equipment` (команда `mv`) (рис. 4.2)

```
[avshuluuzhuk@fedora ~]$ mkdir ski.places
[avshuluuzhuk@fedora ~]$ ls
bin          ski.places  Видео  Загрузки   Музыка        'Рабочий стол'
equipment    work  Документы  Изображения  Общедоступные  Шаблоны
[avshuluuzhuk@fedora ~]$ mv equipment ski.places
[avshuluuzhuk@fedora ~]$ ls ski.places
equipment
```

Рис. 4.2: перемещение файла `equipment` в каталог `ski.places`

Переименуем файл `equipment` в `equiplist` (команда `mv`) (рис. 4.3)

```
[avshuluuzhuk@fedora ~]$ mv ski.places/equipment ski.places/equiplist
[avshuluuzhuk@fedora ~]$ ls ski.places
equiplist
[avshuluuzhuk@fedora ~]$
```

Рис. 4.3: файл `equiplist`

Создаем файл `abc1` (команда `touch`) и копируем его в каталог `ski.places`, назовем `equiplist2` (рис. 4.4)

```
[avshuluuzhuk@fedora ~]$ touch abc1
[avshuluuzhuk@fedora ~]$ cp abc1 ski.places/equiplist2
[avshuluuzhuk@fedora ~]$ ls ski.places
equiplist  equiplist2
[avshuluuzhuk@fedora ~]$
```

Рис. 4.4: каталог ski.places

Создаем каталог с именем equipment в каталоге ski.places и переместим ski.places/equiplist и equiplist2 в каталог ski.places/equipment (рис. 4.5) (рис. 4.6).

```
[avshuluuzhuk@fedora ~]$ mkdir ski.places/equipment
[avshuluuzhuk@fedora ~]$ ls ski.places
equiplist  equiplist2  equipment
[avshuluuzhuk@fedora ~]$
```

Рис. 4.5: создание каталога equipment в каталоге ski.places

```
[avshuluuzhuk@fedora ~]$ mv ski.places/equiplist ski.places/equiplist2 ski.places/equipment
[avshuluuzhuk@fedora ~]$ ls ski/places
ls: невозможно получить доступ к 'ski/places': Нет такого файла или каталога
[avshuluuzhuk@fedora ~]$ ls ski.places
equipment
[avshuluuzhuk@fedora ~]$ ls equipment
ls: невозможно получить доступ к 'equipment': Нет такого файла или каталога
[avshuluuzhuk@fedora ~]$ ls ski.places/equipment
equiplist  equiplist2
[avshuluuzhuk@fedora ~]$
```

Рис. 4.6: перенос файлов equiplist и equiplist2 в каталог equipment

Создаем новый каталог newdir в каталоге ski.places и переименуем в plans (рис. 4.7)

```
[avshuluuzhuk@fedora ~]$ mkdir newdir
[avshuluuzhuk@fedora ~]$ ls
abc1  newdir  work  Документы  Изображения  Общедоступные  Шаблоны
bin  ski.places  Видео  Загрузки  Музыка  'Рабочий стол'
[avshuluuzhuk@fedora ~]$ mv newdir ski.places/plans
[avshuluuzhuk@fedora ~]$ ls ski.places
bash: ski.places: команда не найдена...
[avshuluuzhuk@fedora ~]$ ls ski.places
equipment  plans
[avshuluuzhuk@fedora ~]$
```

Рис. 4.7: каталог plans в каталоге ski.places

Создадим файлы my_os, feathers и каталоги australia, play. Присвоим им следующие права, используя команду chmod (рис. 4.8) (рис. 4.9).

```
[avshuluuzhuk@fedora ~]$ mkdir australia play
[avshuluuzhuk@fedora ~]$ touch my_os
[avshuluuzhuk@fedora ~]$ touch feathers
[avshuluuzhuk@fedora ~]$ ls
abc1      feathers  ski.places  Документы  Музыка      Шаблоны
australia my_os     work       Загрузки   Общедоступные
bin       play      Видео      Изображения 'Рабочий стол'
```

Рис. 4.8: создание необходимых файлов и каталогов

```
[avshuluuzhuk@fedora ~]$ man chmod
[avshuluuzhuk@fedora ~]$ chmod 744 australia
[avshuluuzhuk@fedora ~]$ chmod 711 play
[avshuluuzhuk@fedora ~]$ chmod 544 my_os
[avshuluuzhuk@fedora ~]$ chmod 644 feathers
[avshuluuzhuk@fedora ~]$ ls -lt
итого 0
drwx--x--x. 1 avshuluuzhuk avshuluuzhuk 26 мар  4 15:17 play
-rw-r--r--. 1 avshuluuzhuk avshuluuzhuk  0 мар  4 14:35 feathers
-r-rx-r--r--. 1 avshuluuzhuk avshuluuzhuk  0 мар  4 14:34 my_os
drwxr--r--. 1 avshuluuzhuk avshuluuzhuk  0 мар  4 14:34 australia
```

Рис. 4.9: изменение прав для файлов и каталогов

Посмотрим содержимое файла /etc/passwd (рис. 4.10)

```
[avshuluuzhuk@fedora etc]$ cd
[avshuluuzhuk@fedora ~]$ cd /etc
[avshuluuzhuk@fedora etc]$ cat passwd
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/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:/sbin/nologin
operator:x:11:0:operator:/root:/sbin/nologin
games:x:12:100:games:/usr/games:/sbin/nologin
ftp:x:14:50:FTP User:/var/ftp:/sbin/nologin
```

Рис. 4.10: файл passwd

Скопируем файл feathers в файл file.old (рис. 4.11)

```
[avshuluuzhuk@fedora ~]$ ls
abcl      feathers  ski.plases  Документы  Музыка      Шаблоны
australia my_os     work        Загрузки  Общедоступные
bin       play      Видео       Изображения 'Рабочий стол'
[avshuluuzhuk@fedora ~]$ cp feathers file.old
[avshuluuzhuk@fedora ~]$ ls
abcl      feathers  play        Видео       Изображения 'Рабочий стол'
australia file.old  ski.plases  Документы  Музыка      Шаблоны
bin       my_os     work        Загрузки  Общедоступные
```

Рис. 4.11: копирование файла feathers в файл file.old

Переместим файл file.old в каталог play (рис. 4.12)

```
[avshuluuzhuk@fedora ~]$ mv file.old play
[avshuluuzhuk@fedora ~]$ ls
abcl      feathers  ski.plases  Документы  Музыка      Шаблоны
australia my_os     work        Загрузки  Общедоступные
bin       play      Видео       Изображения 'Рабочий стол'
[avshuluuzhuk@fedora ~]$ ls play
file.old
[avshuluuzhuk@fedora ~]$
```

Рис. 4.12: перемещение файла file.old в каталог play

Скопируем каталог play в каталог fun. (рис. 4.13)

```
[avshuluuzhuk@fedora ~]$ mkdir fun
[avshuluuzhuk@fedora ~]$ cp -r play fun
[avshuluuzhuk@fedora ~]$ ls fun
play
[avshuluuzhuk@fedora ~]$
```

Рис. 4.13: копирование каталога play в каталог fun.

Переместим каталог fun в каталог play и назовем его games.(рис. 4.14)

```
play
[avshuluuzhuk@fedora ~]$ mv fun play/games
[avshuluuzhuk@fedora ~]$ ls play
file.old  games
[avshuluuzhuk@fedora ~]$ ls
```

Рис. 4.14: каталог play

Лишим владельца файла feathers права на чтение, попробуем просмотреть и скопировать этот файл. В результате без прав на чтение владелец файла не сможет просмотреть и скопировать его (рис. 4.15)

```
[avshuluuzhuk@fedora ~]$ chmod u-r feathers
[avshuluuzhuk@fedora ~]$ cat feathers
cat: feathers: Отказано в доступе
[avshuluuzhuk@fedora ~]$ cp feathers feathers2
cp: невозможно открыть 'feathers' для чтения: Отказано в доступе
[avshuluuzhuk@fedora ~]$
```

Рис. 4.15: изменение прав доступа на чтение к файлу feathers

Лишим владельцу каталога play права на выполнение и попробуем перейти в него. Без прав на выполнение владелец не сможет перейти в него. Далее восстановим владельцу права на выполнение (рис. 4.16)

```
[avshuluuzhuk@fedora ~]$ chmod u-x play
[avshuluuzhuk@fedora ~]$ cd play
bash: cd: play: Отказано в доступе
[avshuluuzhuk@fedora ~]$ chmod u+x play
[avshuluuzhuk@fedora ~]$ cd play
[avshuluuzhuk@fedora play]$
```

Рис. 4.16: изменение прав доступа на выполнение к каталогу play

Используя команду man, посмотрим описание команд: mount, fsck, mkfs, kill
Команда mount используется для просмотра используемых в операционной системе файловых систем (рис. 4.17)

```
avshuluuzhuk@fedora:~ — man mount

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 will detach it again. The filesystem is used to control how data is
stored on the device or provided in a virtual way by network or other services.

The standard form of the mount command is:

mount -t type device dir
```

Рис. 4.17: man mount

Команда fsck нужна для проверки и восстановления целостности файловой системы (рис. 4.18)


```
avshuluuzhuk@fedora:~ — man fsck

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 an 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.

If no filesystems are specified on the command line, and the -A option is not
specified, fsck will default to checking filesystems in /etc/fstab serially. This is
equivalent to the -As options.
```

Рис. 4.18: man fsck

Команда mkfs используется для создания файловой системы Linux на устройстве, обычно в разделе жесткого диска. (рис. 4.19)

```
avshuluuzhuk@fedora:~ — man mkfs

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.

The exit status returned by mkfs is 0 on success and 1 on failure.
```

Рис. 4.19: man mkfs

Команда kill отправляет сигнал процессу, указанному с помощью каждого из операндов идентификатора процесса (рис. 4.20)

```
avshuluuzhuk@fedora:~ — man kill

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
```

Рис. 4.20: man kill

5 Контрольные вопросы

1. Дайте характеристику каждой файловой системе, существующей на жёстком диске компьютера, на котором вы выполняли лабораторную работу

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 экзбайта. JFS или Journaled File System была разработана в IBM для AIX UNIX и использовалась в качестве альтернативы для файловых систем ext. Сейчас она используется там, где необходима высокая стабильность и минимальное потребление ресурсов. При разработке файловой системы ставилась цель создать максимально эффективную файловую систему для многопроцессорных компьютеров. Также как и ext, это журналируемая файловая система, но в журнале хранятся только метаданные, что может привести к использованию старых версий файлов после сбоев. ReiserFS - была разработана намного позже, в качестве альтернативы ext3 с улучшенной производительностью и расширенными возможностями. Она была разработана под руководством Ганса Райзера и поддерживает только Linux. Из особенностей можно отметить динамический размер блока, что позволяет упаковывать несколько

небольших файлов в один блок, что предотвращает фрагментацию и улучшает работу с небольшими файлами. Еще одно преимущество - в возможности изменять размеры разделов на лету. Но минус в некоторой нестабильности и риске потери данных при отключении энергии. Раньше ReiserFS применялась по умолчанию в SUSE Linux, но сейчас разработчики перешли на Btrfs. XFS - это высокопроизводительная файловая система, разработанная в Silicon Graphics для собственной операционной системы еще в 2001 году. Она изначально была рассчитана на файлы большого размера, и поддерживала диски до 2 Терабайт. Из преимуществ файловой системы можно отметить высокую скорость работы с большими файлами, отложенное выделение места, увеличение разделов на лету и незначительный размер служебной информации. XFS - журналируемая файловая система, однако в отличие от ext, в журнал записываются только изменения метаданных. Она используется по умолчанию в дистрибутивах на основе Red Hat. Из недостатков - это невозможность уменьшения размера, сложность восстановления данных и риск потери файлов при записи, если будет неожиданное отключение питания, поскольку большинство данных находится в памяти. 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 на 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. Приведите общую структуру файловой системы и дайте характеристику каждой директории первого уровня этой структуры.

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 суперполь-

зователя;

/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. Какая операция должна быть выполнена, чтобы содержимое некоторой файловой системы было доступно операционной системе?

Монтирование тома

4. Назовите основные причины нарушения целостности файловой системы. Как устранить повреждения файловой системы?

Отсутствие синхронизации между образом файловой системы в памяти и ее данными на диске в случае аварийного останова может привести к появлению следующих ошибок:

- 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. Как создаётся файловая система?

mkfs - позволяет создать файловую систему Linux

6. Дайте характеристику командам для просмотра текстовых файлов

cat - выводит содержимое файла на стандартное устройство вывода

7. Приведите основные возможности команды cp в Linux

cp – копирует или перемещает директорию, файлы

8. Приведите основные возможности команды mv в Linux

mv - переименовать или переместить файл или директорию

9. Что такое права доступа? Как они могут быть изменены?

Права доступа к файлу или каталогу можно изменить, воспользовавшись командой chmod. Сделать это может владелец файла (или каталога) или пользователь с правами администратора

6 Выводы

В ходе выполнения лабораторной работы мы ознакомились с файловой системой Linux, ее структурой, именами и содержанием каталогов. Приобрели практические навыки по применению команд для работы с файлами и каталогами, по управлению процессами и работами, по проверке использования диска и обслуживанию файловой системы