

Отчёт по лабораторной работе №5

Операционные системы

Самсонова Мария Ильинична

Содержание

Цель работы	1
Задание	1
Выполнение лабораторной работы	2
Вывод.....	12
Ответы на контрольные вопросы:	12

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, по проверке использования диска и обслуживанию файловой системы.

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:
 - 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.
 - В домашнем каталоге создайте директорию `~/ski.places`.
 - Переместите файл `equipment` в каталог `~/ski.places`.
 - Переименуйте файл `~/ski.places/equipment` в `~/ski.places/equiplist`.
 - Создайте в домашнем каталоге файл `abc1` и скопируйте его в каталог `~/ski.places`, назовите его `equiplist2`.
 - Создайте каталог с именем `equipment` в каталоге `~/ski.places`.
 - Переместите файлы `~/ski.places/equiplist` и `equiplist2` в каталог `~/ski.places/equipment`.
 - 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:
 - `drwxr-r- ... australia`
 - `drwx-x-x ... play`
 - `-r-xr-r- ... my_os`
 - `-rw-rw-r- ... feathers` При необходимости создайте нужные файлы.
4. Прodelайте приведённые ниже упражнения, записывая в отчёт по лабораторной работе используемые при этом команды:
 - Просмотрите содержимое файла `/etc/password`.
 - Скопируйте файл `~/feathers` в файл `~/file.old`.
 - Переместите файл `~/file.old` в каталог `~/play`.
 - Скопируйте каталог `~/play` в каталог `~/fun`.
 - Переместите каталог `~/fun` в каталог `~/play` и назовите его `games`.
 - Лишите владельца файла `~/feathers` права на чтение.
 - Что произойдёт, если вы попытаетесь просмотреть файл `~/feathers` командой `cat`?
 - Что произойдёт, если вы попытаетесь скопировать файл `~/feathers`?
 - Дайте владельцу файла `~/feathers` право на чтение.
 - Лишите владельца каталога `~/play` права на выполнение.
 - Перейдите в каталог `~/play`. Что произошло?
 - Дайте владельцу каталога `~/play` право на выполнение.
5. Прочитайте ман по командам `mount`, `fsck`, `mkfs`, `kill` и кратко их охарактеризуйте, приведя примеры.

Выполнение лабораторной работы

1. Для начала выполним все примеры, приведённые в первой части описания лабораторной работы. То есть:
 - скопируем файл `~/abc1` в файл `april` и в файл `may`
 - скопируем файлы `april` и `may` в каталог `monthly`
 - скопируем файл `monthly/may` в файл с именем `june` (рис. [-@fig:001])

```
misamsonova@fedora:~  
[misamsonova@fedora ~]$ cd  
[misamsonova@fedora ~]$ touch abc1  
[misamsonova@fedora ~]$ cp abc1 april  
[misamsonova@fedora ~]$ cp abc1 may  
[misamsonova@fedora ~]$ mkdir monthly  
[misamsonova@fedora ~]$ cp april may monthly  
[misamsonova@fedora ~]$ cp monthly/may monthly/june  
[misamsonova@fedora ~]$ ls monthly  
april june may  
[misamsonova@fedora ~]$
```

Создание файла командой *touch* и каталога командой *mkdir*, копирование файлов командой *cp*

{ #fig:001 width=70% }

- Далее скопируем каталог *monthly* в каталог *monthly.00*:
- скопируем каталог *monthly.00* в каталог */tmp* (рис. [-@fig:002])

```
[misamsonova@fedora ~]$ mkdir monthly.00  
[misamsonova@fedora ~]$ cp -r monthly monthly.00  
[misamsonova@fedora ~]$ cp -r monthly.00 /tmp  
[misamsonova@fedora ~]$ ls /tmp  
monthly.00  
systemd-private-cdb84c7a88e24fdf81134bbf9a137c0e-chronyd.service-V85tJA  
systemd-private-cdb84c7a88e24fdf81134bbf9a137c0e-colord.service-EnsAXr  
systemd-private-cdb84c7a88e24fdf81134bbf9a137c0e-dbus-broker.service-YX019f  
systemd-private-cdb84c7a88e24fdf81134bbf9a137c0e-geoclue.service-HoPfY7  
systemd-private-cdb84c7a88e24fdf81134bbf9a137c0e-low-memory-monitor.service-f02k0f  
systemd-private-cdb84c7a88e24fdf81134bbf9a137c0e-ModemManager.service-i0rLui  
systemd-private-cdb84c7a88e24fdf81134bbf9a137c0e-power-profiles-daemon.service-8tvda6  
systemd-private-cdb84c7a88e24fdf81134bbf9a137c0e-rtkit-daemon.service-V0RxXl  
systemd-private-cdb84c7a88e24fdf81134bbf9a137c0e-switcheroo-control.service-n5qe5a  
systemd-private-cdb84c7a88e24fdf81134bbf9a137c0e-systemd-logind.service-UtdYNV  
systemd-private-cdb84c7a88e24fdf81134bbf9a137c0e-systemd-oomd.service-GpcaFI  
systemd-private-cdb84c7a88e24fdf81134bbf9a137c0e-systemd-resolved.service-X48KLU  
systemd-private-cdb84c7a88e24fdf81134bbf9a137c0e-upower.service-uqpDWe  
[misamsonova@fedora ~]$ ls mothly.00  
ls: невозможно получить доступ к 'mothly.00': Нет такого файла или каталога  
[misamsonova@fedora ~]$ ls monthly.00  
monthly  
[misamsonova@fedora ~]$
```

Создание каталога командой *mkdir*, копирование файлов командой *cp*, просмотр содержимого каталога командой *ls*

{ #fig:002 width=70% }

- Изменяем название файла *april* на *july* в домашнем каталоге с помощью команды *mv april july*
- Перемещаем файл *july* в каталог *monthly.00* командой *mv july monthly.00*
- Переименовываем каталог *monthly.00* в *monthly.01*

- Перемещаем каталог `monthly.01` в каталог `reports`:
- Переименовываем каталог `reports/monthly.01` в `reports/monthly` (рис. [-@fig:003])

```
[misamsonova@fedora ~]$ mv monthly.00 monthly.01
[misamsonova@fedora ~]$ mkdir reports
[misamsonova@fedora ~]$ mv monthly.01 reports
[misamsonova@fedora ~]$ mv reports/monthly.01 reports/monthly
[misamsonova@fedora ~]$
```

Переименование каталогов в текущем каталоге и перемещение каталога в другой каталог командой `mv`

{ #fig:003 width=70% }

- Создаём файл `~/may` с правом выполнения для владельца
- Лишаем владельца файла `~/may` права на выполнение
- Создаём каталог `monthly` с запретом на чтение для членов группы и всех остальных пользователей
- Создаём файл `~/abc1` с правом записи для членов группы (рис. [-@fig:004])

```
[misamsonova@fedora ~]$ cd
[misamsonova@fedora ~]$ touch may
[misamsonova@fedora ~]$ ls -l may
-rw-rw-r--. 1 misamsonova misamsonova 0 мая  5 15:42 may
[misamsonova@fedora ~]$ chmod u+x may
[misamsonova@fedora ~]$ ls -l may
-rwxrw-r--. 1 misamsonova misamsonova 0 мая  5 15:42 may
[misamsonova@fedora ~]$ cd
[misamsonova@fedora ~]$ mkdir monthly
mkdir: невозможно создать каталог «monthly»: Файл существует
[misamsonova@fedora ~]$ chmod g-r, o-r monthly
chmod: неверный режим: «g-r,»
По команде «chmod --help» можно получить дополнительную информацию.
[misamsonova@fedora ~]$ chmod go-r monthly
[misamsonova@fedora ~]$ cd
[misamsonova@fedora ~]$ touch abc1
[misamsonova@fedora ~]$ chmod g+w abc1
[misamsonova@fedora ~]$ ls -l abc1
-rw-rw-r--. 1 misamsonova misamsonova 0 мая  5 15:44 abc1
[misamsonova@fedora ~]$
```

Создание файлов командой `touch`, просмотр прав файлов командой `ls -l`, создание каталогов командой `mkdir` и изменение прав командой `chmod`

{ #fig:004 width=70% }

2. После выполнения примеров мы переходим к следующим пунктам:
 - Копируем файл `/usr/include/sys/io.h` в домашний каталог и называем его `equipment`
 - В домашнем каталоге создаём директорию `~/ski.plases`.

- Перемещаем файл equipment в каталог ~/ski.plases.
- Переименовываем файл ~/ski.plases/equipment в ~/ski.plases/equiplist.
- Создаём в домашнем каталоге файл abc1 и копируем его в каталог ~/ski.plases, назовите его equiplist2.
- Создаём каталог с именем equipment в каталоге ~/ski.plases.
- Перемещаем файлы ~/ski.plases/equiplist и equiplist2 в каталог ~/ski.plases/equipment.
- Создаём и перемещаем каталог ~/newdir в каталог ~/ski.plases, называем его plans (рис. [-@fig:005])

```
[misamsonova@fedora ~]$ cp /usr/include/sys/io.h ~
[misamsonova@fedora ~]$ ls
abc1  may          work          Изображения  Шаблоны
bin   misamsonova.github.io  Видео        Музыка
blog  monthly      Документы     Общедоступные
io.h  reports      Загрузки     'Рабочий стол'
[misamsonova@fedora ~]$ mv io.h equipment
[misamsonova@fedora ~]$ ls
abc1  may          work          Изображения  Шаблоны
bin   misamsonova.github.io  Видео        Музыка
blog  monthly      Документы     Общедоступные
equipment  reports      Загрузки     'Рабочий стол'
[misamsonova@fedora ~]$ mkdir ~/ski.plases
[misamsonova@fedora ~]$ mv equipment ~/ski.plases
[misamsonova@fedora ~]$ ls
abc1  misamsonova.github.io  work          Изображения  Шаблоны
bin   monthly              Видео        Музыка
blog  reports              Документы     Общедоступные
may   ski.plases            Загрузки     'Рабочий стол'
[misamsonova@fedora ~]$ ls ~/ski.plases
equipment
[misamsonova@fedora ~]$ mv ~/ski.plases/equipment ~/ski.plases/equiplist
[misamsonova@fedora ~]$ touch abc1
[misamsonova@fedora ~]$ mv abc1 ski.plases
[misamsonova@fedora ~]$ mv ski.plases/abc1 ski.plases/equiplist2
[misamsonova@fedora ~]$ mkdir ~/ski.plases/equipment
[misamsonova@fedora ~]$ mv ski.plases/equiplist ski.plases/equipment
[misamsonova@fedora ~]$ mv ski.plases/equiplist2 ski.plases/equipment
[misamsonova@fedora ~]$ mkdir ~/newdir
[misamsonova@fedora ~]$ mv ~/newdir ~/ski.plases
[misamsonova@fedora ~]$ mv newdir plans
mv: не удалось выполнить stat для 'newdir': Нет такого файла или каталога
[misamsonova@fedora ~]$ mv ~/newdir plans
mv: не удалось выполнить stat для '/home/misamsonova/newdir': Нет такого файла или каталога
[misamsonova@fedora ~]$ cd ~/ski.plases
[misamsonova@fedora ski.plases]$ mv newdir plans
[misamsonova@fedora ski.plases]$ ls
equipment  plans
[misamsonova@fedora ski.plases]$
```

Создание файлов командой touch, просмотр содержимого каталогов командой ls, создание каталогов командой mkdir, перемещение каталога в другой каталог и переименовывание каталогов командой mv

{ #fig:005 width=70% }

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

- drwxr-r- 774 australia (это каталог, владелец имеет право на чтение, запись и выполнение, группа владельца и остальные – только чтение)
- drwx-x-x 711 play (это каталог, владелец имеет право на чтение, запись и выполнение, группа владельца и остальные – только выполнение)
- -r-xr-r- 544 my_os (это файл, владелец имеет право на чтение и выполнение, группа владельца и остальные – только чтение)
- -rw-rw-r- 664 feathers (это файл, владелец и группа владельца имеют право на чтение и запись, остальные – только чтение)

Также с помощью команды `mkdir` создаём каталоги: `australia`, `play`. А командой `touch` создаём файлы: `my_os`, `feathers`: (рис. [-@fig:006])

```
[misamsonova@fedora ~]$ mkdir australia
[misamsonova@fedora ~]$ mkdir play
[misamsonova@fedora ~]$ touch my_os
[misamsonova@fedora ~]$ touch feathers
[misamsonova@fedora ~]$ chmod 744 australia
[misamsonova@fedora ~]$ chmod 711 play
[misamsonova@fedora ~]$ chmod 544 my_os
[misamsonova@fedora ~]$ chmod 664 feathers
[misamsonova@fedora ~]$ ls -l
итого 4
drwxr--r--. 1 misamsonova misamsonova  0 мая  5 15:59  australia
drwxr-xr-x. 1 misamsonova misamsonova  8 апр 30 16:29  bin
drwxrwxr-x. 1 misamsonova misamsonova 378 апр 30 17:58  blog
-rw-rw-r--. 1 misamsonova misamsonova  0 мая  5 15:59  feathers
-rwxrw-r--. 1 misamsonova misamsonova  0 мая  5 15:42  may
drwxrwxr-x. 1 misamsonova misamsonova 26 апр 30 17:50  misamsonova.github
ub.io
drwx-wx--x. 1 misamsonova misamsonova 24 мая  5 15:30  monthly
-r-xr--r--. 1 misamsonova misamsonova  0 мая  5 15:59  my_os
drwx--x--x. 1 misamsonova misamsonova  0 мая  5 15:59  play
drwxrwxr-x. 1 misamsonova misamsonova 14 мая  5 15:41  reports
drwxrwxr-x. 1 misamsonova misamsonova 28 мая  5 15:58  ski.plases
drwxrwxr-x. 1 misamsonova misamsonova 10 апр 30 15:37  work
drwxr-xr-x. 1 misamsonova misamsonova  0 апр 30 15:06  Видео
drwxr-xr-x. 1 misamsonova misamsonova  0 апр 30 15:06  Документы
drwxr-xr-x. 1 misamsonova misamsonova 664 апр 30 17:21  Загрузки
drwxr-xr-x. 1 misamsonova misamsonova  0 апр 30 15:06  Изображения
drwxr-xr-x. 1 misamsonova misamsonova  0 апр 30 15:06  Музыка
drwxr-xr-x. 1 misamsonova misamsonova  0 апр 30 15:06  Общедоступные
drwxr-xr-x. 1 misamsonova misamsonova  0 апр 30 15:06  'Рабочий стол'
drwxr-xr-x. 1 misamsonova misamsonova  0 апр 30 15:06  Шаблоны
[misamsonova@fedora ~]$
```

Создание файлов командой `touch`, просмотр прав каталогов и файлов командой `ls -l`, создание каталогов командой `mkdir`, изменение прав командой `chmod`

{ #fig:006 width=70% }

4. Далее посмотрим содержимое файла `/etc/passwd`: (рис. [-@fig:007])

```
[misamsonova@fedora ~]$ cat /etc/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
nobody:x:65534:65534:Kernel Overflow User:/:/sbin/nologin
apache:x:48:48:Apache:/usr/share/httpd:/sbin/nologin
dbus:x:81:81:System message bus:/:/sbin/nologin
systemd-network:x:192:192:systemd Network Management:/:/usr/sbin/nologin
systemd-oom:x:999:999:systemd Userspace OOM Killer:/:/usr/sbin/nologin
systemd-resolve:x:193:193:systemd Resolver:/:/usr/sbin/nologin
systemd-timesync:x:998:998:systemd Time Synchronization:/:/usr/sbin/nologin
systemd-coredump:x:997:997:systemd Core Dumper:/:/usr/sbin/nologin
tss:x:59:59:Account used for TPM access:/dev/null:/sbin/nologin
qemu:x:107:107:qemu user:/:/sbin/nologin
polkitd:x:996:996:User for polkitd:/:/sbin/nologin
avahi:x:70:70:Avahi mDNS/DNS-SD Stack:/var/run/avahi-daemon:/sbin/nologin
unbound:x:995:994:Unbound DNS resolver:/etc/unbound:/sbin/nologin
dnsmasq:x:994:993:Dnsmasq DHCP and DNS server:/var/lib/dnsmasq:/sbin/nologin
nm-openconnect:x:993:991:NetworkManager user for OpenConnect:/:/sbin/nologin
```

Просмотр содержимого файла с помощью команды cat

{ #fig:007 width=70% }

После чего выполним данные действия: - 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 назовите его games: (рис. [-@fig:008])


```

[misamsonova@fedora ~]$ cp ~/feathers ~/file.old
[misamsonova@fedora ~]$ mv ~/file.old ~/play
[misamsonova@fedora ~]$ cp -r ~/play ~/fun
[misamsonova@fedora ~]$ mv ~/fun ~/play
[misamsonova@fedora ~]$ mv play/fun games
[misamsonova@fedora ~]$ ls
australia  misamsonova.github.io  work      Общедоступные
bin        monthly                Видео     'Рабочий стол'
blog       my_os                  Документы  Шаблоны
feathers   play                  Загрузки
games      reports              Изображения
may        ski.places            Музыка
[misamsonova@fedora ~]$ ls play
file.old

```

Копирование файлов с помощью команды cp, перемещение и переименование файлов командой mv, а также просмотр содержимого каталога командой ls

{ #fig:008 width=70% }

Теперь выполним следующие команды: - Лишим владельца файла ~/feathers права на чтение. - Попытаемся просмотреть файл ~/feathers командой cat, однако отказано в доступе из-за изменения прав - Попытаемся скопировать файл ~/feathers, однако отказано в доступе из-за изменения прав - Дайте владельцу файла ~/feathers право на чтение. - Лишите владельца каталога ~/play права на выполнение. - Попытаемся перейдите в каталог ~/play, однако отказано в доступе из-за изменения прав - Дайте владельцу каталога ~/play право на выполнение (рис. [-@fig:009])

```

[misamsonova@fedora ~]$ cp ~/feathers ~/file.old
[misamsonova@fedora ~]$ mv ~/file.old ~/play
[misamsonova@fedora ~]$ cp -r ~/play ~/fun
[misamsonova@fedora ~]$ mv ~/fun ~/play
[misamsonova@fedora ~]$ mv play/fun play/games
[misamsonova@fedora ~]$ chmod u-r feathers
[misamsonova@fedora ~]$ cat feathers
cat: feathers: Отказано в доступе
[misamsonova@fedora ~]$ cp ~/feathers monthly
cp: невозможно открыть '/home/misamsonova/feathers' для чтения: Отказано в досту
пе
[misamsonova@fedora ~]$ chmod u+r feathers
[misamsonova@fedora ~]$ chmod u-x ~/play
[misamsonova@fedora ~]$ cd play
bash: cd: play: Отказано в доступе
[misamsonova@fedora ~]$ chmod u+x ~/play
[misamsonova@fedora ~]$

```

Перемещение и переименование файлов командой mv, просмотр содержимого каталогов командой ls, чтение содержимого файлов командой cat и изменение (лишение/предоставление) прав командой chmod

{ #fig:009 width=70% }

5. После этого прочитаем man по командам mount, fsck, mkfs, kill (рис. [-@fig:010])

```
[misamsonova@fedora ~]$ man mount
[misamsonova@fedora ~]$
[misamsonova@fedora ~]$ man fsck
[misamsonova@fedora ~]$ man mkfs
[misamsonova@fedora ~]$ man kill
[misamsonova@fedora ~]$
```

Получение описаний команд mount, fsck, mkfs, kill командой man

{ #fig:010 width=70% }

- Инструкция по команде mount: предназначена для монтирования файловой системы. Все файлы, доступные в Unix системах, составляют иерархическую файловую структуру, которая имеет ветки (каталоги) и листья (файлы в каталогах). Корень этого дерева обозначается как слеш. Физически файлы могут располагаться на различных устройствах. Команда mount служит для подключения файловых систем разных устройств к этому большому дереву. Наиболее часто встречающаяся форма команды mount выглядит следующим образом: «mount -t *vfstype* *device* *dir*». Такая команда предлагает ядру смонтировать (подключить) файловую систему указанного типа *vfstype*, расположенную на устройстве *device*, к заданному каталогу *dir*, который часто называют точкой монтирования: (рис. [-@fig:011])

```
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|runbindable]
mountpoint

Manual page mount(8) line 1 (press h for help or q to quit)
```

Получение описаний команды mount командой man

{ #fig:011 width=70% }

- Инструкция по команде `fsck`: 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. У команды `fsck` следующий синтаксис: `fsck [параметр] [-[параметры ФС] [...]]`. Например, если нужно восстановить («починить») файловую систему на некотором устройстве `/dev/sdb2`, следует воспользоваться командой: «`sudo fsck -y /dev/sdb2`». Опция `-y` необходима, т.к. при её отсутствии придётся слишком часто давать подтверждение: (рис. [-@fig:012])

```

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 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.

Manual page fsck(8) line 1 (press h for help or q to quit)

```

Получение описаний команды `fsck` командой `man`

{ #fig:012 width=70% }

- Инструкция по команде `mkfs`: создаёт новую файловую систему Linux. Имеет следующий синтаксис: `mkfs[-V] [-tfstype] [fs-options] filesys [blocks] mkfs` используется для создания файловой системы Linux на некотором устройстве, обычно в разделе жёсткого диска. В качестве аргумента `file sys` для файловой системы может выступать или название устройства (например, `/dev/hda1`, `/dev/sdb2`) или точка монтирования (например, `/`, `/usr`, `/home`). Аргументом `blocks` указывается количество блоков, которые выделяются для использования этой файловой системой. По окончании работы `mkfs` возвращает 0 - в случае успеха, а 1 при неудачной операции. Например, команда «`mkfs -t ext2 /dev/hdb1`» создаёт файловую систему типа `ext 2` в разделе `/dev/hdb1` (второй жёсткий диск): (рис. [-@fig:013])

```

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.

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

    In actuality, mkfs is simply a front-end for the various filesystem
    builders (mkfs.fstype) available under Linux. The filesystem-specific
    builder is searched for via your PATH environment setting only. Please

Manual page mkfs(8) line 1 (press h for help or q to quit)

```

Получение описаний команды mkfs командой man

{ #fig:013 width=70% }

- Инструкция по команда kill: посылает сигнал процессу или выводит список допустимых сигналов. Имеет следующий синтаксис: kill [опции] PID, где PID – это PID (числовой идентификатор) процесса или несколько PID процессов, если требуется послать сигнал сразу нескольким процессам. Например, команда «kill - KILL 3121» посылает сигнал KILL процессу с PID 3121, чтобы принудительно завершить процесс: (рис. [-@fig:014])

```
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 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

Manual page kill(1) line 1 (press h for help or q to quit)
```

Получение описаний команды kill командой man

{ #fig:014 width=70% }

Вывод

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

Ответы на контрольные вопросы:

1. Чтобы узнать, какие файловые системы существуют на жёстком диске моего компьютера, использую команду «df-Th». Из рисунка видно, что на моем компьютере есть следующие файловые системы: dev tmpfs,tmpfs,ext4,iso9660. dev tmpfs позволяет ядру создать экземпляр tmpfs с именем devtmpfs при инициализации ядра, прежде чем регистрируется какое-либо устройство с драйверами. Каждое устройство с майором / минором будет предоставлять узел устройства в devtmpfs.devtmpfs монтируется на /dev и содержит специальные файлы устройств для всех устройств.tmpfs–временное файловое хранилище во многих Unix-подобных ОС. Предназначена для монтирования файловой системы, но размещается в ОЗУ вместо ПЗУ. Подобная конструкция является RAM диском. Данная файловая система также предназначена для быстрого и ненадёжного хранения временных данных. Хорошо подходит для /tmp и массовой сборки пакетов/образов.Предполагает наличие достаточного объёма виртуальной

памяти. Файловая система tmpfs предназначена для того, чтобы использовать часть физической памяти сервера как обычный дисковый раздел, в котором можно сохранять данные (чтение и запись). Поскольку данные размещены в памяти, то чтение или запись происходят во много раз быстрее, чем с обычного HDD диска. ext4 – имеет обратную совместимость с предыдущими версиями ФС. Эта версия была выпущена в 2008 году. Является первой ФС из «семейства» Ext, использующая механизм «extent file system», который позволяет добиться меньшей фрагментации файлов и увеличить общую производительность файловой системы. Кроме того, в Ext4 реализован механизм отложенной записи (delayed allocation – delalloc), который так же уменьшает фрагментацию диска и снижает нагрузку на CPU. С другой стороны, хотя механизм отложенной записи и используется во многих ФС, но в силу сложности своей реализации он повышает вероятность утери данных. Характеристики: максимальный размер файла: 16 TB; максимальный размер раздела: 16 TB; максимальный размер имени файла: 255 символов. Рекомендации по использованию: наилучший выбор для SSD; наилучшая производительность по сравнению с предыдущими Ext-системами; она так же отлично подходит в качестве файловой системы для серверов баз данных, хотя сама система и моложе Ext3. ISO 9660 – стандарт, выпущенный Международной организацией по стандартизации, описывающий файловую систему для дисков CD-ROM. Также известен как CDFS (Compact Disc File System). Целью стандарта является обеспечить совместимость носителей под разными операционными системами, такими, как Unix, Mac OS, Windows.

2. Файловая система Linux/UNIX физически представляет собой пространство раздела диска разбитое на блоки фиксированного размера, кратные размеру сектора – 1024, 2048, 4096 или 8120 байт. Размер блока указывается при создании файловой системы. В файловой структуре Linux имеется один корневой раздел – / (он же root, корень). Все разделы жесткого диска (если их несколько) представляют собой структуру подкаталогов, «примонтированных» к определенным каталогам. - “/” – корень. Это главный каталог в системе Linux. По сути, это и есть файловая система Linux. Адреса всех файлов начинаются с корня, а дополнительные разделы, флешки или оптические диски подключаются в папки корневого каталога. Только пользователь root имеет право читать и изменять файлы в этом каталоге. - “/BIN” – бинарные файлы пользователя. Этот каталог содержит исполняемые файлы. Здесь расположены программы, которые можно использовать в однопользовательском режиме или режиме восстановления. - “/SBIN” – системные исполняемые файлы. Так же как и “/bin”, содержит двоичные исполняемые файлы, которые доступны на ранних этапах загрузки, когда не примонтирован каталог /usr. Но здесь находятся программы, которые можно выполнять только с правами суперпользователя. - “/ETC” – конфигурационные файлы. В этой папке содержатся конфигурационные файлы всех программ, установленных в системе. Кроме конфигурационных файлов, в системе инициализации Init Scripts, здесь находятся скрипты запуска и завершения системных демонов, монтирования файловых систем и автозагрузки программ. - “/DEV” – файлы устройств. В Linux все, в том числе внешние устройства являются файлами. Таким образом, все подключенные флешки, клавиатуры, микрофоны, камеры – это просто файлы в каталоге /dev/. Выполняется сканирование всех подключенных устройств и создание для них

специальных файлов. - `/PROC` – информация о процессах По сути, это псевдофайловая система, содержащая подробную информацию о каждом процессе, его `Pid`, имя исполняемого файла, параметры запуска, доступ к оперативной памяти и так далее. Также здесь можно найти информацию об использовании системных ресурсов. - `/VAR` – переменные файлы. Название каталога `“/var”` говорит само за себя, он должен содержать файлы, которые часто изменяются. Размер этих файлов постоянно увеличивается. Здесь содержатся файлы системных журналов, различные кешы, базы данных и так далее. - `/TMP` – временные файлы В этом каталоге содержатся временные файлы, созданные системой, любыми программами или пользователями. Все пользователи имеют право записи в эту директорию. - `/USR` – программы пользователя Это самый большой каталог с большим количеством функций. Здесь находятся исполняемые файлы, исходники программ, различные ресурсы приложений, картинки, музыку и документацию. - `/HOME` – домашняя папка. В этой папке хранятся домашние каталоги всех пользователей. В них они могут хранить свои личные файлы, настройки программ и т.д. - `/BOOT` – файлы загрузчика. Содержит все файлы, связанные с загрузчиком системы. Это ядро `vmlinuz`, образ `initrd`, а также файлы загрузчика, находящиеся в каталоге `/boot/grub`. - `/LIB` – системные библиотеки. Содержит файлы системных библиотек, которые используются исполняемыми файлами в каталогах `/bin` и `/sbin`. - `/OPT` – дополнительные программы В эту папку устанавливаются проприетарные программы, игры или драйвера. Это программы созданные в виде отдельных исполняемых файлов самими производителями. - `/MNT` – монтирование. В этот каталог системные администраторы могут монтировать внешние или дополнительные файловые системы. - `/MEDIA` – съемные носители. В этот каталог система монтирует все подключаемые внешние накопители – USB флешки, оптические диски и другие носители информации. - `/SRV` – сервер. В этом каталоге содержатся файлы серверов и сервисов. - `/RUN` - процессы Каталог, содержащий `PID` файлы процессов, похожий на `“/var/run”`, но в отличие от него, он размещен в `TMPFS`, а поэтому после перезагрузки все файлы теряются.

3. Чтобы содержимое некоторой файловой системы было доступно операционной системе необходимо воспользоваться командой `mount`.
4. Целостность файловой системы может быть нарушена из-за перебоев в питании, неполадок в оборудовании или из-за некорректного/внезапного выключения компьютера. Чтобы устранить повреждения файловой системы необходимо использовать команду `fsck`.
5. Файловую систему можно создать, используя команду `mkfs`. Ее краткое описание дано в пункте 5) в ходе выполнения заданий лабораторной работы.
6. Для просмотра текстовых файлов существуют следующие команды: - `cat`. Задача команды `cat` очень проста – она читает данные из файла или стандартного ввода и выводит их на экран. Синтаксис утилиты: `cat [опции] файл1 файл2 ...`. Основные опции: - `b` – нумеровать только непустые строки - `E` – показывать символ `$` в конце каждой строки - `n` – нумеровать все строки - `s` – удалять пустые повторяющиеся строки - `T` – отображать табуляции в виде `^I` - `h` – отобразить справку - `v` – версия утилиты - `nl`. Команда `nl` действует аналогично команде `cat`, но выводит еще номера строк в столбце слева. - `less`. Существенно более развитая команда для

пролистывания текста. При чтении данных со стандартного ввода она создает буфер, который позволяет листать текст как вперед, так и назад, а также искать как по направлению к концу, так и по направлению к началу текста. Синтаксис аналогичный синтаксису команды `cat`. Некоторые опции: `-g` – при поиске подсвечивать только текущее найденное слово (по умолчанию подсвечиваются все вхождения) `-N` – показывать номера строк `-head`. Команда `head` выводит начальные строки (по умолчанию – 10) из одного или нескольких документов. Также она может показывать данные, которые передает на вывод другая утилита. Синтаксис аналогичный синтаксису команды `cat`. Основные опции: `-c` (`-bytes`) – позволяет задавать количество текста не в строках, а в байтах `-n` (`-lines`) – показывает заданное количество строк вместо 10, которые выводятся по умолчанию `-q` (`-quiet`, `-silent`) – выводит только текст, недобавляя к нему название файла `-v` (`-verbose`) – перед текстом выводит название файла `-z` (`-zero-terminated`) – символы перехода на новую строку заменяет символами завершения строк `-tail` Эта команда позволяет выводить заданное количество строк с конца файла, а также выводить новые строки в интерактивном режиме. Синтаксис аналогичный синтаксису команды `cat`. Основные опции: `-c` – выводить указанное количество байт с конца файла `-f` – обновлять информацию по мере появления новых строк в файле `-n` – выводить указанное количество строк из конца файла `-pid` – используется с опцией `-f`, позволяет завершить работу утилиты, когда завершится указанный процесс `-q` – не выводить имена файлов `-retry` – повторять попытки открыть файл, если он недоступен `-v` – выводить подробную информацию о файле.

7. Утилита `cp` позволяет полностью копировать файлы и директории. Синтаксис: `cp [опции] файл-источник файл-приемник` После выполнения команды файл-источник будет полностью перенесен в файл-приемник. Если в конце указан слэш, файл будет записан в заданную директорию с оригинальным именем. Основные опции: `-attributes-only` – не копировать содержимое файла, а только флаги доступа и владельца `-f`, `-force` – перезаписывать существующие файлы `-i`, `-interactive` – спрашивать, нужно ли перезаписывать существующие файлы `-L` – копировать не символические ссылки, а то, на что они указывают `-n` – не перезаписывать существующие файлы `-P` – не следовать символическим ссылкам `-r` – копировать папку Linux рекурсивно `-s` – не выполнять копирование файлов в Linux, а создавать символические ссылки `-u` – скопировать файл, только если он был изменён `-x` – не выходить за пределы этой файловой системы `-p` – сохранять владельца, временные метки и флаги доступа при копировании `-t` – считать файл-приемник директорией и копировать файл-источник в эту директорию.
8. Команда `mv` используется для перемещения одного или нескольких файлов (или директорий) в другую директорию, а также для переименования файлов и директорий. Синтаксис: `mv [опции] старый_файл новый_файл` Основные опции: `-help` – выводит на экран официальную документацию об утилите `-version` – отображает версию `mv` `-b` – создает копию файлов, которые были перемещены или перезаписаны `-f` – при активации не будет спрашивать разрешение у владельца файла, если речь идет о перемещении или переименовании файла `-i` – наоборот, будет спрашивать разрешение у владельца `-n` – отключает перезапись уже существующих объектов `-strip-trailing-slashes` – удаляет завершающий

символ / у файла при его наличии -t [директория] —перемещает все файлы в указанную директорию -и —осуществляет перемещение только в том случае, если исходный файл новее объекта назначения -v —отображает сведения о каждом элементе во время обработки команды Команда `rename` также предназначена, чтобы переименовать файл. Синтаксис: `rename [опции] старое_имя новое_имя` файлы. Основные опции: -v —вывести список обработанных файлов -n —тестовый режим, на самом деле никакие действия выполнены не будут -f —принудительно перезаписывать существующие файлы.

9. Права доступа— совокупность правил, регламентирующих порядок и условия доступа субъекта к объектам информационной системы (информации, её носителям, процессам и другим ресурсам) установленных правовыми документами или собственником, владельцем информации. Права доступа к файлу или каталогу можно изменить, воспользовавшись командой `chmod`. Сделать это может владелец файла (или каталога) или пользователь с правами администратора. Синтаксис команды: `chmod режим имя_файла` Режим 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.