

Шаблон отчёта по лабораторной работе номер 6

Дисциплина: Операционные системы

Крестененко Полины Александровны НПИМ бд 01-20

Содержание

1	Цель работы	5
2	Задание	6
3	Выполнение лабораторной работы	7
4	Выводы	25

List of Tables

List of Figures

3.1	Первые примеры 1/4	8
3.2	Первые примеры 2/4	8
3.3	Первые примеры 3/4	9
3.4	Первые примеры 4/4	9
3.5	действия 1-6	10
3.6	действия 7 и 8	10
3.7	создаем каталоги	10
3.8	создаем файлы	11
3.9	команда chmod	12
3.10	команда cat	13
3.11	выполнение команд 2-11	14
3.12	mount	15
3.13	выполнение fsck	16
3.14	выполнение mkfs	17
3.15	выполнение kill	18
3.16	df -Th	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 Задание

Заданием являются пункты “Последовательность выполнения лабораторной работы” текстового файла, размещенного на ТУИС

3 Выполнение лабораторной работы

1) Для начала выполним примеры, описанные в первой части описания лабораторной работы. (рисунок 1)

1. Скопируем файл `~/abc1` в файл `april` и в файл `may`. Для этого создадим файл `abc1`, используя команду `touch abc1`, далее осуществим копирование с помощью команд `ср abc1 april` и `ср abc1 may`.
2. Скопируем файлы `april` и `may` в каталог `monthly`, используя команды `mkdir monthly` – для создания каталога `monthly` и `ср april may monthly` – для копирования.
3. Скопируем файл `monthly/may` в файл с именем `june`. Выполним команды `ср monthly/may monthly/june` и `ls monthly` (для просмотра содержимого каталога).
4. Скопируем каталог `monthly` в каталог `monthly.00`. Для этого создадим каталог `monthly.00` командой `mkdir monthly.00` и осуществим копирование, используя команду `ср -r monthly monthly.00` (команда `ср` с опцией `r` (recursive) позволяет копировать каталоги вместе с входящими в них файлами и каталогами).
5. Скопируем каталог `monthly.00` в каталог `/tmp`, используя команду `ср -r monthly.00 /tmp`. (рис. 3.1)

```

pakrestenenko@dk4n62 ~ $ cd
pakrestenenko@dk4n62 ~ $ touch abc1
pakrestenenko@dk4n62 ~ $ cp abc1 april
pakrestenenko@dk4n62 ~ $ cp abc1 may
pakrestenenko@dk4n62 ~ $ mkdir monthly
pakrestenenko@dk4n62 ~ $ cp april may monthly
pakrestenenko@dk4n62 ~ $ cp monthly/may monthly/june
pakrestenenko@dk4n62 ~ $ ls monthly
april  june  may
pakrestenenko@dk4n62 ~ $ mkdir monthly.00
pakrestenenko@dk4n62 ~ $ cp -r monthly monthly.00
pakrestenenko@dk4n62 ~ $ cp -r monthly.00 /tmp

```

Figure 3.1: Первые примеры 1/4

рисунок 2: 1. Изменим название файла april на july в домашнем каталоге, используя команду «mv april july». 2. Переместим файл july в каталог monthly.00 с помощью команды «mv july monthly.00». Проверим результат командой «ls monthly.00». 3. Переименуем каталог monthly.00 в monthly.01, используя команду «mv monthly.00 monthly.01». 4. Переместим каталог monthly.01 в каталог reports. Для этого создадим каталог reports с помощью команды «mkdir reports» и выполним перемещение командой «mv monthly.01 reports». 5. Переименуем каталог reports/monthly.01 в reports/monthly командой «mv reports/monthly.01 reports/monthly».(рис. 3.2)

```

pakrestenenko@dk4n62 ~ $ cd
pakrestenenko@dk4n62 ~ $ mv april july
pakrestenenko@dk4n62 ~ $ mv july monthly.00
pakrestenenko@dk4n62 ~ $ ls monthly.00
july  monthly
pakrestenenko@dk4n62 ~ $ mv monthly.00 monthly.01
pakrestenenko@dk4n62 ~ $ mkdir reports
pakrestenenko@dk4n62 ~ $ mv monthly.01 reports
pakrestenenko@dk4n62 ~ $ mv reports/monthly.01 reports/monthly

```

Figure 3.2: Первые примеры 2/4

рисунки 3 и 4: 1. Создадим файл ~/may с правом выполнения для владельца. Для этого выполним следующие команды: «touch may» (создание файла), «ls -l may» (просмотр сведений о файле), «chmod u+x may» (изменение прав), «ls -l may». 2. Лишаем владельца файла ~/may права на выполнение, используя команды: «chmod

u-x may» (изменение прав), «ls -l may» (просмотр сведений о файле). 3. Создаем каталог monthly с запретом на чтение для членов группы и всех остальных пользователей. Выполняем команды: «mkdir monthly»(создание каталога), «chmod go-monthly»(изменение прав). 4. Создаем файл ~/abc1 с правом записи для членов группы, используя команды: «touch abc1» (создание файла), «chmod g+w abc1» (изменение прав).(рис. 3.3)

```

pakrestenenko@dk4n62 ~ $ cd
pakrestenenko@dk4n62 ~ $ touch may
pakrestenenko@dk4n62 ~ $ ls -l may
-rw-r--r-- 1 pakrestenenko studsci 0 мая 14 12:38 may
pakrestenenko@dk4n62 ~ $ chmod u+x may
pakrestenenko@dk4n62 ~ $ ls -l may
-rwxr--r-- 1 pakrestenenko studsci 0 мая 14 12:38 may
pakrestenenko@dk4n62 ~ $ chmod u-x may
pakrestenenko@dk4n62 ~ $ ls -l may
-rw-r--r-- 1 pakrestenenko studsci 0 мая 14 12:38 may
pakrestenenko@dk4n62 ~ $ cd
pakrestenenko@dk4n62 ~ $ mkdir monthly
mkdir: невозможно создать каталог «monthly»: Файл существует

```

Figure 3.3: Первые примеры 3/4

(рис. 3.4)

```

pakrestenenko@dk4n62 ~ $ chmod go-r monthly
pakrestenenko@dk4n62 ~ $ cd
pakrestenenko@dk4n62 ~ $ touch abc1
pakrestenenko@dk4n62 ~ $ chmod g+w abc1

```

Figure 3.4: Первые примеры 4/4

2)Выполняем следующие действия, отображенные на Рисунках 5 и 6: 1.Копируем файл/usr/include/aio.h(т.к. у меня нет каталога /usr/include/sys/, то беру произвольный файл из каталога /usr/include/)вдомашнийкаталог(команда«cp/usr/include/aio.h~») и назоваем его equipment(команда «mvaio.hequipment»). 2.В домашнем каталоге создаем директорию ~/ski.plases(команда «mkdirski.plases»). 3.Перемещаем файл equipment в каталог ~/ski.plases(команда «mvequipmentski.plases»). 4.Переименовываем файл/ski.plases/equipmentв/ski.plases/equiplist(команда«mvski.plases/equipment

ski.places/equiplist»). 5.Создаем в домашнем каталоге файл abc1(команда«touchabc1»)и копируем его в каталог~/ski.places(команда«сrabc1 ski.places»), называем его equiplist2(команда«mv ski.places/abc1 ski.places/equiplist2»). 6.Создаем каталог с именем equipment в каталоге~/ski.places(команда«mkdir ski.places/equipment»). 7.Перемещаем файлы~/ski.places/equiplist иequiplist2 в каталог~/ski.places/equipment(команда«mv ski.places/equiolist ski.places/equiplist2 ski.places/equipment»). 8.Создаем(команда«mkdirnewdir») иперемещаем каталог~/newdir в каталог~/ski.places(команда«mv newdir ski.places»)и называем его plans(команда«mv ski.places/newdir ski.places/plans»).(рис. 3.5)

```
pakrestenenko@dk4n62 ~ $ cp /usr/include/aio.h ~
pakrestenenko@dk4n62 ~ $ mv aio.h equipment
pakrestenenko@dk4n62 ~ $ mkdir ski.places
pakrestenenko@dk4n62 ~ $ mv equipment ski.places
pakrestenenko@dk4n62 ~ $ mv ski.places/equipment ski.places/equiplist
pakrestenenko@dk4n62 ~ $ touch abc1
pakrestenenko@dk4n62 ~ $ cp abc1 ski.places
pakrestenenko@dk4n62 ~ $ mv ski.places/abc1 ski.places/equiplist2
pakrestenenko@dk4n62 ~ $ mkdir ski.places/equipment
mkdir: невозможно создать каталог «ski.places/equipment»: Нет такого файла или к
аталога
pakrestenenko@dk4n62 ~ $ mkdir ski.places/equipment
```

Figure 3.5: действия 1-6

(рис. 3.6)

```
pakrestenenko@dk4n62 ~ $ mv ski.places/equiplist ski.places/equiplist2 ski.place
s/equipment
pakrestenenko@dk4n62 ~ $ mkdir newdir
mkdir: невозможно создать каталог «newdir»: Файл существует
pakrestenenko@dk4n62 ~ $ mv newdir ski.places
pakrestenenko@dk4n62 ~ $ mv ski.places/newdir ski.places/plans
```

Figure 3.6: действия 7 и 8

3)Определяемопции команды chmod, необходимые для того, чтобы присвоить соответствующим файлам выделенные права доступа, считая, что в начале таких прав нет. Предварительно создаем необходимые файлы, используя команды: «mkdiraustralia», «mkdirplay», «touchmy_os», «touchfeathers»(Рисунок 7).(рис. 3.7)

```
pakrestenenko@dk4n62 ~ $ mkdir australia
pakrestenenko@dk4n62 ~ $ mkdir play
```

Figure 3.7: создаем каталоги

(рис. 3.8)

```
|pakrestenenko@dk4n62 ~ $ touch my_os  
|pakrestenenko@dk4n62 ~ $ touch feathers
```

Figure 3.8: создаем файлы

drwxr-r-... australia: команда«chmod744 australia» (это каталог, владелец имеет право на чтение, запись и выполнение, группа владельца и остальные –только чтение)(Рисунок9). drwx-x-x ... play: команда«chmod711play»(это каталог, владелец имеет право на чтение, запись и выполнение, группа владельца и остальные –только выполнение) (Рисунок 9). -r-xr-r-... my_os: команды«chmod544my_os»(это файл, владелецимеет право на чтение и выполнение, группа владельца и остальные –только чтение) (Рисунок 9) -rw-rw-r-... feathers: команды «chmod664feathers»(это файл, владелец и группа владельца имеют право на чтение и запись, остальные –только чтение)(Рисунок 9). Командой «ls-l» проверяем правильность выполненных действий.(рис. 3.9)

```

pakrestenenko@dk4n62 ~ $ chmod 744 australia
pakrestenenko@dk4n62 ~ $ chmod 711 play
pakrestenenko@dk4n62 ~ $ chmod 544 my_os
pakrestenenko@dk4n62 ~ $ chmod 664 feathers
pakrestenenko@dk4n62 ~ $ ls -l
итого 5160
-rw-rw-r-- 1 pakrestenenko studsci      0 мая 14 12:46 abc1
-rwxr-xr-x 1 pakrestenenko studsci    1128 сен 18 2020 asdfg
-rw-r--r-- 1 pakrestenenko studsci     535 сен 18 2020 asdfg.asm
-rw-r--r-- 1 pakrestenenko studsci    1104 сен 18 2020 asdfg.o
drwxr--r-- 2 pakrestenenko studsci    2048 мая 14 12:53 australia
drw-rw-r-- 2 pakrestenenko studsci    2048 мая 14 14:02 feathers
drwxr-xr-x 4 pakrestenenko studsci    2048 мая 14 13:24 krestenenko
drwxr-xr-x 2 pakrestenenko studsci    2048 сен 11 2020 lab02
drwxr-xr-x 2 pakrestenenko studsci    2048 сен 18 2020 lab03a
drwxr-xr-x 2 pakrestenenko studsci    2048 сен 18 2020 lab03b
-rwxr-xr-x 1 pakrestenenko studsci    1328 окт 2 2020 lab05
-rw-r--r-- 1 pakrestenenko studsci     966 окт 2 2020 lab05.asm
-rw-r--r-- 1 pakrestenenko studsci    1376 окт 2 2020 lab05.o
-rwxr-xr-x 1 pakrestenenko studsci    1288 окт 9 2020 lab06
-rw-r--r-- 1 pakrestenenko studsci     864 окт 9 2020 lab06.asm
-rw-r--r-- 1 pakrestenenko studsci    1296 окт 9 2020 lab06.o
-rwxr-xr-x 1 pakrestenenko studsci    1480 окт 16 2020 lab07
-rwxr-xr-x 1 pakrestenenko studsci    1104 окт 16 2020 lab07.asm
-rw-r--r-- 1 pakrestenenko studsci    1552 окт 16 2020 lab07.o
-rwxr-xr-x 1 pakrestenenko studsci     936 сен 11 2020 lab2
-rw-r--r-- 1 pakrestenenko studsci    1131 сен 11 2020 lab2.asm
-rw-r--r-- 1 pakrestenenko studsci     880 сен 11 2020 lab2.o
-rw-r--r-- 1 pakrestenenko studsci     966 сен 11 2020 lab.asm
-rw-r--r-- 1 pakrestenenko studsci      0 мая 14 12:38 may
drwx--x-x 2 pakrestenenko studsci    2048 мая 14 12:28 monthly
dr-xr--r-- 2 pakrestenenko studsci    2048 мая 14 14:02 my_os
drwx--x-x 5 pakrestenenko studsci    2048 мая 14 13:14 play
-rw-r--r-- 1 pakrestenenko studsci    1059 сен 25 2020 program.asm
-rw-r--r-- 1 pakrestenenko studsci    4082 сен 25 2020 program.lst
drwxr-xr-x 3 pakrestenenko studsci    2048 мая 14 12:01 public
lrwxr-xr-x 1 pakrestenenko studsci      18 сен 10 2020 public_html -> public/public_html
drwxr-xr-x 3 pakrestenenko studsci    2048 мая 14 12:35 reports
drwxr-xr-x 4 pakrestenenko studsci    2048 мая 14 12:52 ski_places
drwxr-xr-x 2 pakrestenenko studsci    2048 мая 13 18:27 tmp
drwxr-xr-x 2 pakrestenenko studsci    2048 сен 4 2020 Видео
drwxr-xr-x 2 pakrestenenko studsci    2048 сен 4 2020 Документы
drwxr-xr-x 2 pakrestenenko studsci    2048 мая 14 11:54 Загрузки
drwxr-xr-x 2 pakrestenenko studsci    2048 сен 4 2020 Изображения
drwxr-xr-x 2 pakrestenenko studsci    2048 сен 4 2020 Музыка
drwxr-xr-x 2 pakrestenenko studsci    2048 сен 4 2020 Общедоступные
-rw-r--r-- 1 pakrestenenko studsci    242176 сен 18 2020 отчет_лаб_шаблон3.doc
-rw-r--r-- 1 pakrestenenko studsci    461312 сен 25 2020 отчет_лаб_шаблон4.doc

```

Figure 3.9: команда chmod

4)Выполняем следующие действия, отображенные на Рисунках 10 и 11:

- 1.Просмотрим содержимое файла /etc/passwd (команда «cat/etc/passwd»).
- 2.Копируем файл ~/feathers в файл ~/file.old(команда «cpfeathersfile.old»). 3.Переместим файл ~/file.old в каталог ~/play(команда «mvfile.ordplay»).
- 4.Скопируем каталог ~/play в каталог ~/fun(команда «cp-rplayfun»).
- 5.Переместим каталог ~/fun в каталог ~/play(команда «mvfunplay»)и назовем его games(команда «mvplay/funplay/games»).
- 6.Лишим владельца файла ~/feathers права на чтение(команда «chmod u-rfeathers»).
- 7.Если мы попытаемся просмотреть файл ~/feathers командой cat, то получим отказ в доступе, т.к. в предыдущем пункте лишили владельца права на чтение данного файла.
- 8.Если мы попытаемся скопировать файл ~/feathers, например, в каталог monthly, то получим отказ

вдоступе, по причине, описанной в предыдущем пункте. 9. Дадим владельцу файла ~/feathers право на чтение (команда «chmod+rfeathers»). 10. Лишим владельца каталога ~/play права на выполнение (команда «chmod-xplay»). 11. Перейдем в каталог ~/play (команда «cdplay»). Получим отказ в доступе, т.к. в предыдущем пункте лишили владельца права на выполнение данного каталога. 12. Дадим владельцу каталога ~/play право на выполнение (команда «chmod+xplay»). (рис. 3.10)

```
pakrestenenko@dk4n62 ~ $ cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/bin/false
daemon:x:2:2:daemon:/sbin:/bin/false
adm:x:3:4:adm:/var/adm:/bin/false
lp:x:4:7:lp:/var/spool/lpd:/bin/false
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 program user:/var/spool/mail:/sbin/nologin
news:x:9:13:news:/usr/lib/news:/bin/false
uucp:x:10:14:uucp:/var/spool/uucppublic:/bin/false
operator:x:11:0:operator:/root:/bin/bash
man:x:13:15:System user; man:/dev/null:/sbin/nologin
postmaster:x:14:12:Postmaster user:/var/spool/mail:/sbin/nologin
cron:x:16:16:cron:/var/spool/cron:/bin/false
ftp:x:21:21::/home/ftp:/bin/false
sshd:x:22:22:User for ssh:/var/empty:/sbin/nologin
at:x:25:25:at:/var/spool/cron/atjobs:/bin/false
squid:x:31:31:Squid:/var/cache/squid:/bin/false
gdm:x:32:32:GDM:/var/lib/gdm:/bin/false
xfs:x:33:33:X Font Server:/etc/X11/fs:/bin/false
games:x:35:35:games:/usr/games:/bin/bash
named:x:40:40:bind:/var/bind:/bin/false
mysql:x:60:60:MySQL program user:/dev/null:/sbin/nologin
postgres:x:70:70:PostgreSQL program user:/var/lib/postgresql:/bin/sh
nut:x:84:84:nut:/var/state/nut:/bin/false
cyrus:x:85:12::/usr/cyrus:/bin/false
vpopmail:x:89:89::/var/vpopmail:/bin/false
alias:x:200:200::/var/qmail/alias:/bin/false
qmaild:x:201:200::/var/qmail:/bin/false
qmail1:x:202:200::/var/qmail:/bin/false
qmailp:x:203:200::/var/qmail:/bin/false
qmailq:x:204:201::/var/qmail:/bin/false
qmailr:x:205:201::/var/qmail:/bin/false
qmails:x:206:201::/var/qmail:/bin/false
postfix:x:207:207:postfix:/var/spool/postfix:/bin/false
smmsp:x:209:209:smmsp:/var/spool/mqueue:/bin/false
portage:x:250:250:System user; portage:/var/tmp/portage:/sbin/nologin
nobody:x:65534:65534:nobody:/bin/false
sockd:x:101:2:added by portage for dante:/etc/socks:/bin/false
rpc:x:111:111:added by portage for portmap:/dev/null:/bin/false
ntp:x:123:123:user for ntp daemon:/dev/null:/sbin/nologin
ldap:x:439:439:added by portage for openldap:/usr/lib/openldap:/bin/false
messagebus:x:102:440:System user; messagebus:/dev/null:/sbin/nologin
haldaemon:x:103:441:added by portage for hal:/dev/null:/bin/false
dhcpcd:x:104:10148:user for dhcpcd daemon:/dev/null:/sbin/nologin
```

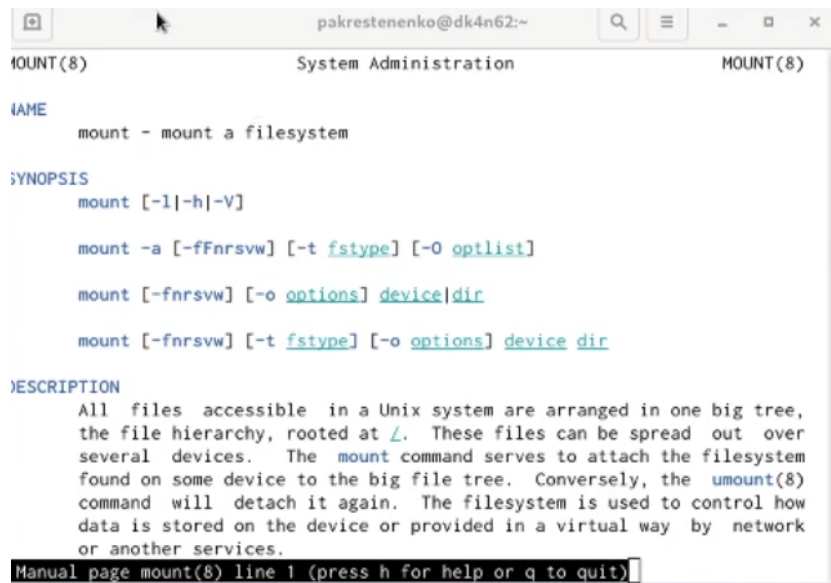
Figure 3.10: команда cat

(рис. 3.11)

```
pakrestenenko@dk4n62 ~ $ cp -r feathers file.old
pakrestenenko@dk4n62 ~ $ mv file.old play
pakrestenenko@dk4n62 ~ $ cp -r play fun
pakrestenenko@dk4n62 ~ $ mv fun play
pakrestenenko@dk4n62 ~ $ mv play/fun play/games
pakrestenenko@dk4n62 ~ $ chmod u-r feathers
pakrestenenko@dk4n62 ~ $ cat feathers
```

Figure 3.11: выполнение команд 2-11

5)Используя команды «manmount», «manfsck»,«manmkfs»,«mankill», получим информацию о соответствующих командах (Рисунки 12, 13, 14, 15). Команда 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 /. Физически файлы могут располагаться на различных устройствах. Команда mount служит для подключения файловых систем разных устройств к этому большому дереву.Наиболее часто встречающаяся форма командыmount-выглядит следующим образом:«mount -tvfstype device dir»Такая команда предлагает ядру смонтировать (подключить) файловую систему указанного типаavfstype, расположенную на устройстvedevice, к заданному каталогуdir, который часто называют точкой монтирования.(рис. 3.12)



```

pakrestenenko@dk4n62:~
mount(8)                                System Administration                                mount(8)

NAME
    mount - mount a filesystem

SYNOPSIS
    mount [-l|-h|-V]

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

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

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

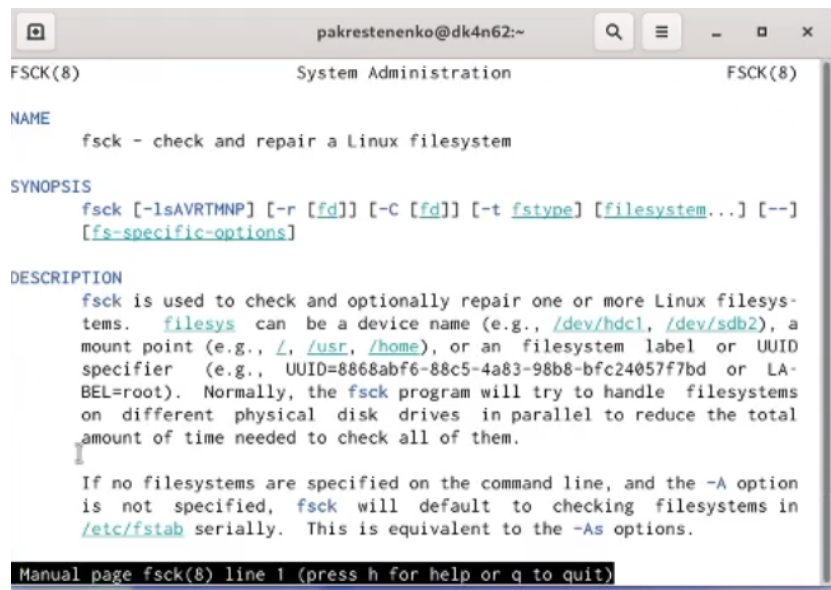
DESCRIPTION
    All files accessible in a Unix system are arranged in one big tree,
    the file hierarchy, rooted at /. These files can be spread out over
    several devices. The mount command serves to attach the filesystem
    found on some device to the big file tree. Conversely, the umount(8)
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

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

```

Figure 3.12: mount

Команда `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. 3.13)



```

pakrestenenko@dk4n62:~
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/hdcl, /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.

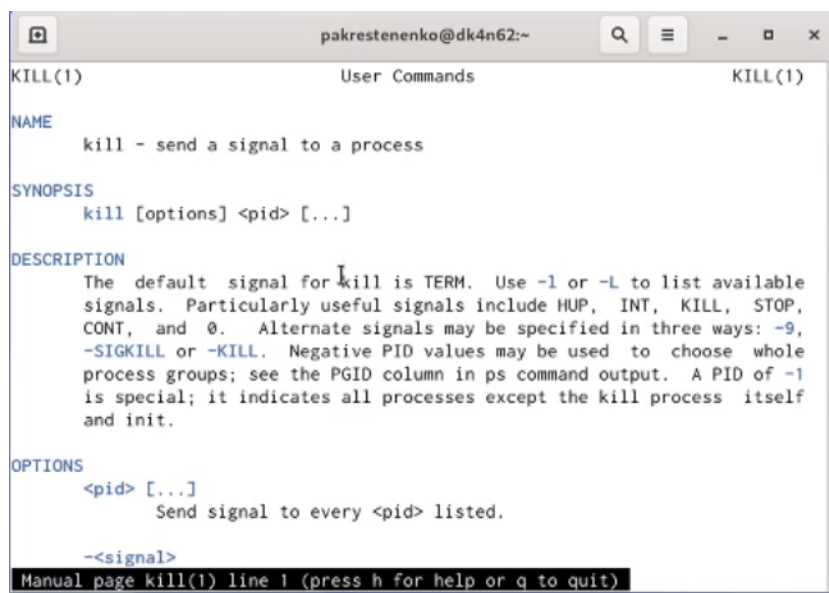
Manual page fsck(8) line 1 (press h for help or q to quit)
```

Figure 3.13: выполнение fsck

Команда mkfs: создаёт новую файловую систему Linux. Имеет следующий синтаксис: `mkfs[-V] [-tfstype] [fs-options] filesystem[blocks]` mkfs используется для создания файловой системы Linux на некотором устройстве, обычно в разделе жёсткого диска. В качестве аргумента filesystem 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 ext2 в разделе /dev/hdb1 (второй жёсткий диск). (рис. 3.14)

Figure 3.14: выполнение mkfs

Команда kill:посылает сигнал процессу или выводит список допустимых сигналов.Имеет следующий синтаксис: kill[опции]PID, гдеPID–это PID (числовой идентификатор) процесса или несколько PIDпроцессов, если требуется послать сигнал сразу нескольким процессам.Например, команда «kill -KILL 3121» посылает сигнал KILL процессу сPID 3121, чтобы принудительно завершить процесс.(рис. 3.15)



```
KILL(1)                                User Commands                                KILL(1)

NAME
    kill - send a signal to a process

SYNOPSIS
    kill [options] <pid> [...]

DESCRIPTION
    The default signal for kill is TERM. Use -l or -L to list available
    signals. Particularly useful signals include HUP, INT, KILL, STOP,
    CONT, and 0. Alternate signals may be specified in three ways: -9,
    -SIGKILL or -KILL. Negative PID values may be used to choose whole
    process groups; see the PGID column in ps command output. A PID of -1
    is special; it indicates all processes except the kill process itself
    and init.

OPTIONS
    <pid> [...]
        Send signal to every <pid> listed.

    -<signal>

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

Figure 3.15: выполнение kill

3.Контрольные вопросы: 1)Чтобы узнать, какие файловые системы существуют на жёстком диске моего компьютера, использую команду«df-Th» (Рисунок 16). Из рисунка видно, что на моем компьютере есть следующие файловые системы: devtmpfs,tmpfs,ext4,iso9660.devtmpfspозволяет 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 «extentfile 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: 16TB; 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 (рис. 3.16)

```

pakrestenenko@dk4n62 ~ $ df -Th

```

Файловая система	Тип	Размер	Использовано	Дост	Использовано%	Смонтировано в
/	tmpfs	3,9G	20M	3,8G	1%	/run
udev	devtmpfs	3,8G	0	3,8G	0%	/dev
tmpfs	tmpfs	3,9G	116M	3,8G	3%	/dev/shm
/dev/sda8	ext4	470G	57G	389G	13%	/
tmpfs	tmpfs	4,0M	0	4,0M	0%	/sys/fs/cgroup
tmpfs	tmpfs	3,9G	94M	3,8G	3%	/tmp
/dev/sda6	ext4	88G	777M	83G	1%	/var/cache/openafs
mark.sci.pfu.edu.ru:/com/lib/portage	nfs4	700G	111G	590G	16%	/com/lib/portage
mark.sci.pfu.edu.ru:/usr/portage	nfs4	8,0G	4,2G	1,8G	71%	/usr/portage
mark.sci.pfu.edu.ru:/usr/local/share/portage	nfs4	8,0G	4,2G	1,8G	71%	/usr/local/share/portage
AFS	afs	2,0T	0	2,0T	0%	/afs
tmpfs	tmpfs	782M	308K	781M	1%	/run/user/3714
mark.sci.pfu.edu.ru:/usr/local/share/portage	nfs4	8,0G	4,2G	1,8G	71%	/usr/local/share/portage

Figure 3.16: df -Th

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) пред-

ставляют 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 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, не добавляя 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 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 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` – перемещает все файлы в указанную директорию `-u` – 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 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. `g` чтение, `w` запись, `x` выполнение. `u` (user) владелец файла, `g` (group) группа, к которой принадлежит владелец файла, `o` (others) все остальные.

4 Выводы

В ходе выполнения данной лабораторной работы я ознакомилась с файловой системой Linux, её структурой, именами и содержанием каталогов, получила навыки по применению команд для работы с файлами и каталогами, по управлению процессами (и работами), по проверке использования диска и обслуживанию файловой системы.