

РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

Факультет Физико-Математических Наук

Кафедра прикладной информатики и теории вероятностей

ОТЧЁТ

ПО ЛАБОРАТОРНОЙ РАБОТЕ № 5

дисциплина: Операционные системы

Студент: Манаева Варвара Евгеньевна

Группа: НФИбд-01-20

Преподаватель: Кулябов Дмитрий Сергеевич

МОСКВА

2021 г.

Техническое оснащение:

- Персональный компьютер с операционной системой Windows 7;
- Планшет для записи видеосопровождения и голосовых комментариев;
- Виртуальная коробка VirtualBox, виртуальная машина с установленной на ней операционной системой CentOS;
- Microsoft Teams, использующийся для записи скринкаста лабораторной работы;
- Приложение MarkPad 2 для редактирования файлов формата *md*;
- *pandoc* для конвертации файлов отчётов и презентаций.

Объект и предмет исследования: Командная строка операционной системы *Linux* и особенности взаимодействия с системой посредством командной строки.

Цель: Приобрести практические навыки взаимодействия пользователя с системой посредством командной строки.

Задачи:

- 1) Изучить базовые команды для работы с системой;
- 2) Научиться находить необходимую информацию о командах и их использовании при помощи командной строки.

Теоретические вводные данные:

Формат команды. Командой в операционной системе называется записанный по специальным правилам текст (возможно с аргументами), представляющий собой указание на выполнение какой-либо функций (или действий) в операционной системе. Обычно первым словом идёт имя команды, остальной текст — аргументы или опции, конкретизирующие действие.

Общий формат команд можно представить следующим образом:

`<имя_команды><разделитель><аргументы>`

Команда *man*. Команда *man* используется для просмотра (оперативная помощь) в диалоговом режиме руководства (*manual*) по основным командам операционной системы типа *Linux*.

Формат команды:

`man <команда>`

Пример (вывод информации о команде *man*):

`man man`

Для управления просмотром результата выполнения команды *man* можно использовать следующие клавиши:

- ***space*** — перемещение по документу на одну страницу вперёд;
- ***enter*** — перемещение по документу на одну строку вперёд;
- ***q*** — выход из режима просмотра описания.

Команда *cd*. Команда *cd* используется для перемещения по файловой системе операционной системы типа *Linux*.

Замечание 3. Файловая система ОС типа *Linux* — иерархическая система каталогов, подкаталогов и файлов, которые обычно организованы и сгруппированы по функциональному признаку. Самый верхний каталог в иерархии называется корневым и обозначается символом */*. Корневой каталог содержит системные файлы и другие каталоги.

Формат команды: `cd [путьккаталогу]` Для перехода в домашний каталог пользователя следует использовать команду `cd` без параметров или `cd ~`.

Например, команда

`cd /afs/dk.sci.pfu.edu.ru/home`

позволяет перейти в каталог `/afs/dk.sci.pfu.edu.ru/home` (если такой существует), а для того, чтобы подняться выше на одну директорию, следует использовать:

`cd ..`

Подробнее об опциях команды *cd* смотри в справке с помощью команды *man*:

`man cd`

Команда *pwd*. Для определения абсолютного пути к текущему каталогу используется команда *pwd* (*print working directory*). Пример (абсолютное имя текущего каталога пользователя *dharm*):

`pwd`

результат:

Сокращения имён файлов. В работе с командами, в качестве аргументов которых выступает путь к какому-либо каталогу или файлу, можно использовать сокращённую запись пути.

Чтобы вывести на экран подробную информацию о файлах и каталогах, необходимо использовать опцию `l`. При этом о каждом файле и каталоге будет выведена следующая информация:

- тип файла,
- право доступа,
- число ссылок,
- владелец,
- размер,
- дата последней ревизии,
- имя файла или каталога.

Пример:

```
cd /
```

```
ls
```

Результат:

```
bin boot dev etc home lib media mnt
opt proc root sbin sys tmp usr var
```

В этом же каталоге команда

```
ls -alF
```

даст примерно следующий результат:

```
drwxr-xr-x 21 root root 4096 Jan. 17 09:00 ./
drwxr-xr-x 21 root root 4096 Jan. 17 09:00 ../
drwxr-xr-x 2 root root 4096 Jan. 18 15:57 bin/
drwxr-xr-x 2 root root 4096 Apr. 14 2008 boot/
drwxr-xr-x 20 root root 14120 Feb. 17 10:48 dev/
drwxr-xr-x 170 root root 12288 Feb. 17 09:19 etc/
drwxr-xr-x 6 root root 4096 Aug. 5 2009 home/
lrwxrwxrwx 1 root root 5 Jan. 12 22:01 lib -> lib64/
drwxr-xr-x 8 root root 4096 Jan. 30 21:41 media/
drwxr-xr-x 5 root root 4096 Jan. 17 2010 mnt/
drwxr-xr-x 25 root root 4096 Jan. 16 09:55 opt/
dr-xr-xr-x 163 root root 0 Feb. 17 13:17 proc/
drwxr-xr-x 31 root root 4096 Feb. 15 23:57 root/
drwxr-xr-x 2 root root 12288 Jan. 18 15:57 sbin/
drwxr-xr-x 12 root root 0 Feb. 17 13:17 sys/
drwxrwxrwt 12 root root 500 Feb. 17 16:35 tmp/
drwxr-xr-x 22 root root 4096 Jan. 18 09:26 usr/
drwxr-xr-x 17 root root 4096 Jan. 14 17:38 var/
```

Команда `mkdir`. Команда `mkdir` используется для создания каталогов.

Формат команды:

`mkdir имякаталога1 [имякаталога2...]` Пример создания каталога в текущем каталоге:

```
cd
```

```
pwd
```

```
/afs/dk.sci.pfu.edu.ru/home/d/h/dharma
```

```
ls
```

```
Desktop public tmp
```

```
GNUstep publichtml work
```

```
mkdir abc
```

```
ls
```

```
abc GNUstep publichtml work
```

```
Desktop public tmp
```

Замечание 4. Для того чтобы создать каталог в определённом месте файловой системы, должны быть правильно установлены права доступа.

Можно создать также подкаталог в существующем подкаталоге:

```
mkdir parentdir  
mkdir parentdir/dir
```

При задании нескольких аргументов создаётся несколько каталогов:

```
cd parentdir  
mkdir dir1 dir2 dir3
```

Можно использовать группировку:

```
mkdir parentdir/{dir1,dir2,dir3}
```

Если же требуется создать подкаталог в каталоге, отличном от текущего, то путь к нему требуется указать в явном виде:

```
mkdir ../dir1/dir2  
или  
mkdir ~/dir1/dir2
```

Интересны следующие опции:

--mode (или -m) — установка атрибутов доступа;

--parents (или -p) — создание каталога вместе с родительскими по отношению к нему каталогами.

Атрибуты задаются в численной или символьной нотации:

```
mkdir --mode=777 dir  
или  
mkdir -m a+rwX dir
```

Опция --parents (краткая форма -p) позволяет создавать иерархическую цепочку подкаталогов, создавая все промежуточные каталоги:

```
mkdir -p ~/dir1/dir2/dir3
```

Команда rm. Команда *rm* используется для удаления файлов и/или каталогов.

Формат команды:

```
rm [-опции] [файл]
```

Если требуется, чтобы выдавался запрос подтверждения на удаление файла, то необходимо использовать опцию *i*. Чтобы удалить каталог, содержащий файлы, нужно использовать опцию *r*. Без указания этой опции команда не будет выполняться.

Пример:

```
cd  
mkdir abs  
rm abc  
rm: abc is a directory  
rm -r abc
```

Если каталог пуст, то можно воспользоваться командой *rmdir*. Если удаляемый каталог содержит файлы, то команда не будет выполнена — нужно использовать *rm -r имя_каталога*.

Команда history. Для вывода на экран списка ранее выполненных команд используется команда *history*. Выводимые на экран команды в списке нумеруются. К любой команде из выведенного на экран списка можно обратиться по её номеру в списке, воспользовавшись конструкцией *!<номер_команды>*.

Пример:

```
history  
1 pwd  
2 ls  
3 ls -a  
4 ls -l  
5 cd /  
6 history  
!5  
cd /
```

Можно модифицировать команду из выведенного на экран списка при помощи следующей конструкции:

```
!<номер_команды>:s/<что_меняем>/<на_что_меняем>
```

Пример:

```
!3:s/a/F  
ls -F
```

Замечание 5. Если в заданном контексте встречаются специальные символы (типа «.», «/», «*» и т.д.), надо перед ними поставить символ экранирования \ (обратный слэш).

Использование символа «;». Если требуется выполнить последовательно несколько команд, записанный в одной строке, то для этого используется символ точка с запятой.

Пример:

```
cd; ls
```

Этапы работы:

1) Определила полное имя домашнего каталога.

```
[vemanaeva@vemanaeva ~]$ pwd  
/home/vemanaeva
```

2) Выполнила следующие действия:

2.1) Перешла в каталог /tmp.

```
[vemanaeva@vemanaeva ~]$ cd /tmp
```

2.2) Вывела на экран содержимое каталога /tmp. Для этого использовала команду ls с различными опциями.

```
[vemanaeva@vemanaeva tmp]$ ls  
anaconda.log  
hsperfdata_root  
ifcfg.log  
ks-script-Cx163E  
packaging.log  
program.log  
sensitive-info.log  
ssh-2kAaF67fWmim  
ssh-6LA0frmLMLHv  
ssh-j16eTWXKsucB  
ssh-pVA046q5fG2B  
storage.log  
systemd-private-21c3bc13375c4db989532e798a5cd040-bolt.service-Vbcw15  
systemd-private-21c3bc13375c4db989532e798a5cd040-chronyd.service-aA0ztV  
systemd-private-21c3bc13375c4db989532e798a5cd040-colord.service-bkzk0e  
systemd-private-21c3bc13375c4db989532e798a5cd040-cups.service-MC016z  
systemd-private-21c3bc13375c4db989532e798a5cd040-fwupd.service-q0tVK1  
systemd-private-21c3bc13375c4db989532e798a5cd040-rtkit-daemon.service-opnTGq  
systemd-private-27a87d0daa824cab9f8d6453cd4b7536-bolt.service-N01fBJ  
systemd-private-27a87d0daa824cab9f8d6453cd4b7536-chronyd.service-dvIVb3  
systemd-private-27a87d0daa824cab9f8d6453cd4b7536-colord.service-y0uaJe  
systemd-private-27a87d0daa824cab9f8d6453cd4b7536-cups.service-LYs9MR  
systemd-private-27a87d0daa824cab9f8d6453cd4b7536-fwupd.service-NLii8S  
systemd-private-27a87d0daa824cab9f8d6453cd4b7536-rtkit-daemon.service-IT4gPZ  
systemd-private-2925b59188dc49208d09f71fd9cb18e1-bolt.service-X6FETH  
systemd-private-2925b59188dc49208d09f71fd9cb18e1-chronyd.service-lU5dbU  
systemd-private-2925b59188dc49208d09f71fd9cb18e1-colord.service-chwSKR  
systemd-private-2925b59188dc49208d09f71fd9cb18e1-cups.service-cS5AHo  
systemd-private-2925b59188dc49208d09f71fd9cb18e1-fwupd.service-M5nSgt  
systemd-private-2925b59188dc49208d09f71fd9cb18e1-rtkit-daemon.service-DxMa5l  
systemd-private-f0fa3a02de15416e99deabf39bd45a94-bolt.service-jgb7o3  
systemd-private-f0fa3a02de15416e99deabf39bd45a94-chronyd.service-Xhbgls  
systemd-private-f0fa3a02de15416e99deabf39bd45a94-colord.service-Y32iXl  
systemd-private-f0fa3a02de15416e99deabf39bd45a94-cups.service-0ZssAA  
systemd-private-f0fa3a02de15416e99deabf39bd45a94-fwupd.service-noVFYL  
systemd-private-f0fa3a02de15416e99deabf39bd45a94-rtkit-daemon.service-FhLMh0  
tracker-extract-files.1000  
vboxguest-Module.symvers  
yum.log  
yum_save_tx.2021-05-01.13-16.110Zah.yumtx  
yum_save_tx.2021-05-04.11-41.2tBDcv.yumtx
```

```
[vemanaeva@vemanaeva tmp]$ ls -l  
итого 336  
-rw-r--r--. 1 root root 1148 май 1 13:14 anaconda.log  
drwxr-xr-x. 2 root root 18 май 1 13:03 hsperfdata_root  
-rw-r--r--. 1 root root 420 май 1 13:14 ifcfg.log  
-rwx-----. 1 root root 836 май 1 13:13 ks-script-Cx163E  
-rw-r--r--. 1 root root 0 май 1 13:14 packaging.log  
-rw-r--r--. 1 root root 0 май 1 13:14 program.log  
-rw-r--r--. 1 root root 0 май 1 13:14 sensitive-info.log  
drwx-----. 2 vemanaeva vemanaeva 24 май 1 15:46 ssh-2kAaF67fWmim  
drwx-----. 2 vemanaeva vemanaeva 24 май 1 17:47 ssh-6LA0frmLMLHv  
drwx-----. 2 vemanaeva vemanaeva 24 май 1 14:47 ssh-j16eTWXKsucB  
drwx-----. 2 vemanaeva vemanaeva 24 май 4 11:40 ssh-pVA046q5fG2B  
-rw-r--r--. 1 root root 0 май 1 13:14 storage.log  
drwx-----. 3 root root 17 май 1 14:46 systemd-private-21c3bc13375c4db989532e798a5cd040-bolt.service-Vbcw15  
drwx-----. 3 root root 17 май 1 14:46 systemd-private-21c3bc13375c4db989532e798a5cd040-chronyd.service-aA0ztV  
drwx-----. 3 root root 17 май 1 14:46 systemd-private-21c3bc13375c4db989532e798a5cd040-colord.service-bkzk0e  
drwx-----. 3 root root 17 май 1 14:46 systemd-private-21c3bc13375c4db989532e798a5cd040-cups.service-MC016z  
drwx-----. 3 root root 17 май 1 14:47 systemd-private-21c3bc13375c4db989532e798a5cd040-fwupd.service-q0tVK1  
drwx-----. 3 root root 17 май 1 14:46 systemd-private-21c3bc13375c4db989532e798a5cd040-rtkit-daemon.service-opnTGq  
drwx-----. 3 root root 17 май 1 17:47 systemd-private-27a87d0daa824cab9f8d6453cd4b7536-bolt.service-N01fBJ  
drwx-----. 3 root root 17 май 1 17:46 systemd-private-27a87d0daa824cab9f8d6453cd4b7536-chronyd.service-dvIVb3  
drwx-----. 3 root root 17 май 1 17:47 systemd-private-27a87d0daa824cab9f8d6453cd4b7536-colord.service-y0uaJe  
drwx-----. 3 root root 17 май 1 17:46 systemd-private-27a87d0daa824cab9f8d6453cd4b7536-cups.service-LYs9MR  
drwx-----. 3 root root 17 май 1 17:47 systemd-private-27a87d0daa824cab9f8d6453cd4b7536-fwupd.service-NLii8S  
drwx-----. 3 root root 17 май 1 17:46 systemd-private-27a87d0daa824cab9f8d6453cd4b7536-rtkit-daemon.service-IT4gPZ  
drwx-----. 3 root root 17 май 4 11:40 systemd-private-2925b59188dc49208d09f71fd9cb18e1-bolt.service-X6FETH  
drwx-----. 3 root root 17 май 4 11:39 systemd-private-2925b59188dc49208d09f71fd9cb18e1-chronyd.service-lU5dbU  
drwx-----. 3 root root 17 май 4 11:40 systemd-private-2925b59188dc49208d09f71fd9cb18e1-colord.service-chwSKR  
drwx-----. 3 root root 17 май 4 11:39 systemd-private-2925b59188dc49208d09f71fd9cb18e1-cups.service-cS5AHo  
drwx-----. 3 root root 17 май 4 11:40 systemd-private-2925b59188dc49208d09f71fd9cb18e1-fwupd.service-M5nSgt  
drwx-----. 3 root root 17 май 4 11:39 systemd-private-2925b59188dc49208d09f71fd9cb18e1-rtkit-daemon.service-DxMa5l  
drwx-----. 3 root root 17 май 1 15:45 systemd-private-f0fa3a02de15416e99deabf39bd45a94-bolt.service-jgb7o3  
drwx-----. 3 root root 17 май 1 15:45 systemd-private-f0fa3a02de15416e99deabf39bd45a94-chronyd.service-Xhbgls  
drwx-----. 3 root root 17 май 1 15:45 systemd-private-f0fa3a02de15416e99deabf39bd45a94-colord.service-Y32iXl  
drwx-----. 3 root root 17 май 1 15:45 systemd-private-f0fa3a02de15416e99deabf39bd45a94-cups.service-0ZssAA  
drwx-----. 3 root root 17 май 1 15:46 systemd-private-f0fa3a02de15416e99deabf39bd45a94-fwupd.service-noVFYL  
drwx-----. 3 root root 17 май 1 15:45 systemd-private-f0fa3a02de15416e99deabf39bd45a94-rtkit-daemon.service-FhLMh0  
drwx-----. 2 vemanaeva vemanaeva 6 май 1 13:14 tracker-extract-files.1000  
-rw-r--r--. 1 root root 25965 май 1 13:16 vboxguest-Module.symvers  
-rw-----. 1 root root 0 май 1 12:59 yum.log  
-rw-----. 1 root root 150012 май 1 13:16 yum_save_tx.2021-05-01.13-16.110Zah.yumtx  
-rw-----. 1 root root 150012 май 4 11:41 yum_save_tx.2021-05-04.11-41.2tBDcv.yumtx
```

```
[vmanaeva@vmanaeva tmp]$ ls -F
anaconda.log
hsperrdata_root/
lfcfg.log
ls -script Cx163E*
packaging.log
program.log
sensitive-info.log
ssh-2KaA67Fwmim/
ssh-6LA0frMLMLHv/
ssh-j16eTWXKsUC8/
ssh-pVA046q5fG2B/
storage.log
systemd-private-21c3bc13375c4db989532e798a5cd040-bolt.service-Vbcw15/
systemd-private-21c3bc13375c4db989532e798a5cd040-chronyd.service-aA0ZtV/
systemd-private-21c3bc13375c4db989532e798a5cd040-colord.service-bkzk0e/
systemd-private-21c3bc13375c4db989532e798a5cd040-cups.service-MC016z/
systemd-private-21c3bc13375c4db989532e798a5cd040-fwupd.service-g0tVK1/
systemd-private-21c3bc13375c4db989532e798a5cd040-rtkit-daemon.service-opnTgq/
systemd-private-27a87d0daa824cab9f8d6453cd4b7536-bolt.service-N01f8J/
systemd-private-27a87d0daa824cab9f8d6453cd4b7536-chronyd.service-dvIVB3/
systemd-private-27a87d0daa824cab9f8d6453cd4b7536-colord.service-y0uale/
systemd-private-27a87d0daa824cab9f8d6453cd4b7536-cups.service-LYs9MR/
systemd-private-27a87d0daa824cab9f8d6453cd4b7536-fwupd.service-NLI18S/
systemd-private-27a87d0daa824cab9f8d6453cd4b7536-rtkit-daemon.service-IT4gPZ/
systemd-private-2925b59188dc49208d09f71fd9cb18e1-bolt.service-X6FEtH/
systemd-private-2925b59188dc49208d09f71fd9cb18e1-chronyd.service-LU5dbU/
systemd-private-2925b59188dc49208d09f71fd9cb18e1-colord.service-chwSKR/
systemd-private-2925b59188dc49208d09f71fd9cb18e1-cups.service-c5SAHo/
systemd-private-2925b59188dc49208d09f71fd9cb18e1-fwupd.service-M5n5gt/
systemd-private-2925b59188dc49208d09f71fd9cb18e1-rtkit-daemon.service-bxMa5l/
systemd-private-f0fa3a02de15416e99deabf39bd45a94-bolt.service-jgb7o3/
systemd-private-f0fa3a02de15416e99deabf39bd45a94-chronyd.service-Xhbgls/
systemd-private-f0fa3a02de15416e99deabf39bd45a94-colord.service-Y32iXl/
systemd-private-f0fa3a02de15416e99deabf39bd45a94-cups.service-0ZssAA/
systemd-private-f0fa3a02de15416e99deabf39bd45a94-fwupd.service-noVfYL/
systemd-private-f0fa3a02de15416e99deabf39bd45a94-rtkit-daemon.service-PhLMh0/
tracker-extract-files.1000/
vboxguest-Module.symvers
yum.log
yum_save_tx.2021-05-01.13-16.11QZah.yumtx
yum_save_tx.2021-05-04.11-41.ZtBDcv.yumtx
```

Разница в выводимой на экран информации:

Команда `ls` выводит содержимое каталога.

Команда `ls -l` выводит подробный список, в котором будет отображаться владелец, группа, дата создания, размер и другие параметры.

Команда `ls -F` показывает тип объекта.

2.3) Определила, есть ли в каталоге `/var/spool` подкаталог с именем `cron`.

```
[vmanaeva@vmanaeva tmp]$ cd /var/spool
[vmanaeva@vmanaeva spool]$ ls
abrt abrt-upload anacron at cron cups lpd mail plymouth postfix
```

2.4) Перешла в свой домашний каталог

и вывела на экран его содержимое. Определила, кто является владельцем файлов и подкаталогов (третий столбик после команды `ls -l`).

```
[vmanaeva@vmanaeva spool]$ cd ~
[vmanaeva@vmanaeva ~]$ ls -la
.      .bash_logout      .cache      .esd_auth    laboratory    .pkgi      .vboxclient-display-svga-x11.pid  Видео      Изображения      Рабочий стол
..     .bash_profile     .config     .gitconfig   .local       .ssh       .vboxclient-draganddrop.pid      Документы     Музыка
.bash_history .bashrc         .dbus       .ICEauthority .mozilla     .vboxclient-clipboard.pid      .vboxclient-seamless.pid      Загрузки      Общедоступные
[vmanaeva@vmanaeva ~]$ ls -l
итого 0
drwxr-xr-x. 3 vmanaeva vmanaeva 83 май 1 18:08 laboratory
drwxr-xr-x. 2 vmanaeva vmanaeva 6 май 1 13:14 Видео
drwxr-xr-x. 2 vmanaeva vmanaeva 6 май 1 13:14 Документы
drwxr-xr-x. 2 vmanaeva vmanaeva 6 май 1 13:14 Загрузки
drwxr-xr-x. 2 vmanaeva vmanaeva 6 май 1 13:14 Изображения
drwxr-xr-x. 2 vmanaeva vmanaeva 6 май 1 13:14 Музыка
drwxr-xr-x. 2 vmanaeva vmanaeva 6 май 1 13:14 Общедоступные
drwxr-xr-x. 2 vmanaeva vmanaeva 6 май 1 13:14 Рабочий стол
drwxr-xr-x. 2 vmanaeva vmanaeva 6 май 1 13:14 Шаблоны
```

3) Выполнила следующие действия:

3.1) В домашнем каталоге создала новый каталог с именем `newdir`.

```
[vmanaeva@vmanaeva ~]$ mkdir newdir
```

3.2) В каталоге `~/newdir` создала новый каталог с именем `morefun`.

```
[vmanaeva@vmanaeva ~]$ cd ~/newdir
[vmanaeva@vmanaeva newdir]$ mkdir morefun
[vmanaeva@vmanaeva newdir]$ ls
morefun
```

3.3) В домашнем каталоге создала одной командой три новых каталога с именами `letters`, `memos`, `misk`. Затем удалила эти каталоги одной командой.

```
[vmanaeva@vmanaeva newdir]$ cd ~
[vmanaeva@vmanaeva ~]$ mkdir letters memos misk
[vmanaeva@vmanaeva ~]$ ls
laboratory letters memos misk newdir Видео Документы Загрузки Изображения Музыка Общедоступные Рабочий стол Шаблоны
[vmanaeva@vmanaeva ~]$ rmdir letters memos misk
[vmanaeva@vmanaeva ~]$ ls
laboratory newdir Видео Документы Загрузки Изображения Музыка Общедоступные Рабочий стол Шаблоны
```

3.4) Попробовала удалить ранее созданный каталог `~/newdir` командой `rm`. Проверила, был ли каталог удалён.

```
[vmanaeva@vmanaeva ~]$ rm newdir
rm: невозможно удалить «newdir»: Это каталог
[vmanaeva@vmanaeva ~]$ ls
laboratory newdir Видео Документы Загрузки Изображения Музыка Общедоступные Рабочий стол Шаблоны
```

3.5) Удалила каталог `~/newdir/morefun` из домашнего каталога. Проверила, был ли каталог удалён.

```
[vmanaeva@vmanaeva ~]$ rm -r newdir
[vmanaeva@vmanaeva ~]$ ls
laboratory Видео Документы Загрузки Изображения Музыка Общедоступные Рабочий стол Шаблоны
```

4) С помощью команды `man` определила, какую опцию команды `ls` нужно использовать для просмотра содержимого не только указанного каталога, но и подкаталогов, входящих в него.

```
-R, --recursive
list subdirectories recursively
```

5) С помощью команды `man` определила набор опций команды `ls`, позволяющий отсортировать по времени последнего изменения

выводимый список содержимого каталога с развёрнутым описанием файлов.

```
--time-style=STYLE
with -l, show times using style STYLE: full-iso, long-iso, iso, locale, or +FORMAT; FORMAT is interpreted like in 'date'; if FORMAT is FORMAT1<newline>FORMAT2, then FORMAT1 applies to
non-recent files and FORMAT2 to recent files; if STYLE is prefixed with 'posix-', STYLE takes effect only outside the POSIX locale

-t      sort by modification time, newest first

-T, --tabsize=COLS
      assume tab stops at each COLS instead of 8

-u      with -lt: sort by, and show, access time; with -l: show access time and sort by name; otherwise: sort by access time
```

6) Использовала команду *man* для просмотра описания следующих команд: *cd*, *pwd*, *mkdir*, *rmdir*, *rm*.

```
cd [-L|-P [-r]] [dir]
Change the current directory to dir. The variable HOME is the default dir. The variable CDPATH defines the search path for the directory containing dir. Alternative directory names in
CDPATH are separated by a colon (:). A null directory name in CDPATH is the same as the current directory, i.e., '.'. If dir begins with a slash (/), then CDPATH is not used. The -P
option says to use the physical directory structure instead of following symbolic links (see also the -P option to the set builtin command); the -L option forces symbolic links to be fol-
lowed. If the -e option is supplied with -P, and the current working directory cannot be successfully determined after a successful directory change, cd will return an unsuccessful status.
An argument of - is equivalent to $OLDPWD. If a non-empty directory name from CDPATH is used, or if - is the first argument, and the directory change is successful, the absolute pathname of
the new working directory is written to the standard output. The return value is true if the directory was successfully changed; false otherwise.
```

```
pwd [-LR]
Print the absolute pathname of the current working directory. The pathname printed contains no symbolic links if the -P option is supplied or the -o physical option to the set builtin com-
mand is enabled. If the -L option is used, the pathname printed may contain symbolic links. The return status is 0 unless an error occurs while reading the name of the current directory or
an invalid option is supplied.
```

MKDIR(1)

User C

NAME

mkdir - make directories

SYNOPSIS

mkdir [OPTION]... DIRECTORY...

DESCRIPTION

Create the DIRECTORY(ies), if they do not already exist.

Mandatory arguments to long options are mandatory for short options too.

-m, --mode=MODE

set file mode (as in chmod), not a=rwx - umask

-p, --parents

no error if existing, make parent directories as needed

-v, --verbose

print a message for each created directory

-Z

set SELinux security context of each created directory to the default type

--context[=CTX]

like -Z, or if CTX is specified then set the SELinux or SMACK security context to CTX

--help display this help and exit

--version

output version information and exit

```

RMDIR(1) User Commands

NAME
    rmdir - remove empty directories

SYNOPSIS
    rmdir [OPTION]... DIRECTORY...

DESCRIPTION
    Remove the DIRECTORY(ies), if they are empty.

    --ignore-fail-on-non-empty
        ignore each failure that is solely because a directory
        is non-empty

    -p, --parents
        remove DIRECTORY and its ancestors; e.g., 'rmdir -p a/b/c' is similar to 'rmdir a/b/c a/b a'

    -v, --verbose
        output a diagnostic for every directory processed

    --help
        display this help and exit

    --version
        output version information and exit

```

```

RM(1) User Commands RM(1)

NAME
    rm - remove files or directories

SYNOPSIS
    rm [OPTION]... FILE...

DESCRIPTION
    This manual page documents the GNU version of rm. rm removes each specified file. By default, it does not remove directories.

    If the -I or --interactive=once option is given, and there are more than three files or the -r, -R, or --recursive are given, then rm prompts the user for whether to proceed with the entire operation. If the response is not affirmative, the entire command is aborted.

    Otherwise, if a file is unwritable, standard input is a terminal, and the -f or --force option is not given, or the -i or --interactive=always option is given, rm prompts the user for whether to remove the file. If the response is not affirmative, the file is skipped.

OPTIONS
    Remove (unlink) the FILE(s).

    -f, --force
        ignore nonexistent files and arguments, never prompt

    -i
        prompt before every removal

    -I
        prompt once before removing more than three files, or when removing recursively; less intrusive than -i, while still giving protection against most mistakes

    --interactive[=WHEN]
        prompt according to WHEN: never, once (-I), or always (-i); without WHEN, prompt always

    --one-file-system
        when removing a hierarchy recursively, skip any directory that is on a file system different from that of the corresponding command line argument

    --no-preserve-root
        do not treat '/' specially

    --preserve-root
        do not remove '/' (default)

    -r, -R, --recursive
        remove directories and their contents recursively

    -d, --dir
        remove empty directories

```

```

--preserve-root
    do not remove '/' (default)

-r, -R, --recursive
    remove directories and their contents recursively

-d, --dir
    remove empty directories

-v, --verbose
    explain what is being done

--help
    display this help and exit

--version
    output version information and exit

By default, rm does not remove directories. Use the --recursive (-r or -R) option to remove each listed directory, too, along with all of its contents.

To remove a file whose name starts with a '.', for example '-foo', use one of these commands:

    rm -f -foo
    rm ./-foo

Note that if you use rm to remove a file, it might be possible to recover some of its contents, given sufficient expertise and/or time. For greater assurance that the contents are truly unrecoverable, consider using shred.

```

Основные опции этих команд:

cd – команда перемещения по файловой системе.

pwd – команда показывающая директорию, в которой находится пользователь.

mkdir – команда, создающая новую директорию.

rmdir – команда, удаляющая файлы, которые должны быть пустыми.

rm - команда, удаляющая файлы или целые деревья каталогов.

7) Используя информацию, полученную при помощи команды *history*, выполнила модификацию и исполнение нескольких команд из буфера команд.

```
[vemanaeva@vemanaeva ~]$ history
 1  hostnamectl
 2  dmesg | less
 3  dmesg | grep -i version
 4  dmesg | grep -i Mhz
 5  dmesg | grep -i CPU0
 6  dmesg | grep -i memory
 7  dmesg | grep -i hypervisor
 8  dmesg | grep -i filesystem
 9  dmesg | grep -i mount
10  pwd
11  cd /tmp
12  ls
13  ls -l
14  ls -F
15  cd /var/spool
16  ls
17  cd ~
18  ls -a
19  ls -l
20  mkdir newdir
21  cd ~/newdir
22  mkdir morefun
23  ls
24  cd ~
25  mkdir letters memos misk
26  ls
27  rmdir letters memos misk
28  ls
29  rm newdir
30  ls
31  rm -r newdir
32  ls
33  man ls
34  man cd
35  man mkdir
36  man rmdir
37  man rm
38  history
[vemanaeva@vemanaeva ~]$ !11
cd /tmp
[vemanaeva@vemanaeva tmp]$ !20:s/newdir/ararardir
mkdir ararardir
[vemanaeva@vemanaeva tmp]$ ls
anaconda.log
ararardir
```

Выводы: я приобрела практические навыки взаимодействия с системой посредством командной строки.

Контрольные вопросы:

1) Что такое командная строка?

Ответ: Интерфейс командной строки - управление программами с помощью команд. Команды состоят из букв, цифр, символов, набираются построчно, выполняются после нажатия клавиши Enter. Основной инструмент здесь клавиатура. Данный интерфейс встроен в ядро системы, он будет доступен, даже если графический интерфейс не запустится. Добраться до командной строки можно двумя способами: через консоль или терминал.

2) При помощи какой команды можно определить абсолютный путь текущего каталога? Приведите пример.

Ответ: При помощи команды *pwd* можно определить абсолютный путь текущего каталога. Например, если вбить *pwd var* на экран выведется */home//var*.

3) При помощи какой команды и каких опций можно определить только тип файлов и их имена в текущем каталоге? Приведите примеры.

Ответ: При помощи команды `ls -F` можно определить только тип файлов и их имена в текущем каталоге.

4) Какие файлы считаются скрытыми? Как получить информацию о скрытых файлах? Приведите примеры.

Ответ: Файл (или директория) считается скрытым, если его название начинается с символа точки «.». Например, «`.myfile`». Обычно такие файлы используются приложениями для хранения настроек, конфигураций и другой информации, которую нужно скрыть от пользователя. Зачастую пользователю требуется отредактировать соответствующий конфигурационный скрытый файл, чтобы настроить какую-нибудь программу, и пользователи сталкиваются с тем, что не знают, как их вообще просмотреть. По умолчанию файловые менеджеры обычно не отображают такие файлы. Для просмотра списка файлов в командной строке используется команда `ls`. Чтобы по команде `ls` также выводились скрытые файлы, существует опция `-a`.

5) При помощи каких команд можно удалить файл и каталог? Можно ли это сделать одной и той же командой? Приведите примеры.

Ответ: При помощи команд `rm` и `rmdir` можно удалить файл и каталог. Это нельзя сделать одной и той же командой. `rmdir` используется, чтобы удалить файлы, которые должны быть пустые. Команда `rm` используется, чтобы удалить непустые файлы или целые деревья каталогов.

6) Как определить, какие команды выполнил пользователь в сеансе работы?

Ответ: Определить какие команды выполнил пользователь в сеансе работы можно с помощью команды `history`.

7) Каким образом можно исправить и запустить на выполнение команду, которую пользователь уже использовал в сеансе работы? Приведите примеры.

Ответ: Исправить и запустить на выполнение команду, которую пользователь уже использовал в сеансе работы, можно с помощью команды: `!<номер_команды>:s/<что_меняем>/<начтоменяем>` Например,

```
history
.
.
3 ls -a
.
.
!3:s/a/F
ls -F
```

8) Можно ли в одной строке записать несколько команд? Если да, то как? Приведите примеры.

Ответ: В одной строке можно записать несколько команд. Если требуется выполнить последовательно несколько команд, записанный в одной строке, то для этого используется символ точка с запятой. Пример: `cd; ls`.

9) Что такое символ экранирования? Приведите примеры использования этого символа.

Ответ: Экранирование — это способ заключения в кавычки одиночного символа. Экранирующий символ (`\`) сообщает интерпретатору, что следующий за ним символ должен восприниматься как обычный символ. Пример:

```
echo "Привет"
echo "Он сказал: \"Привет\"."
```

Вывод:

Привет

Он сказал: "Привет".

10) Какая информация выводится на экран о файлах и каталогах, если используется опция `l` в команде `ls`?

Ответ: Если используется опция `l` в команде `ls`, то на экран выводится подробный список, в котором будет отображаться владелец, группа, дата создания, размер и другая информация о файлах и каталогах.

11) Что такое относительный путь к файлу? Приведите примеры использования относительного и абсолютного пути при выполнении какой-либо команды.

Ответ: Относительный путь — это путь к файлу относительно текущей папки. При использовании команды `pwd` на экран выведется относительный путь текущей директории, а при использовании команды `realpath` на экран выведется абсолютный путь текущей директории.

12) Как получить информацию об интересующей вас команде?

Ответ: Получить информацию об интересующей вас команде можно с помощью команды *man*. Например, команда *man ls* выведет все опции команды *ls*.

13) Какая клавиша или комбинация клавиш служит для автоматического дополнения вводимых команд?

Ответ: Клавиша *Tab* служит для автоматического дополнения вводимых команд.