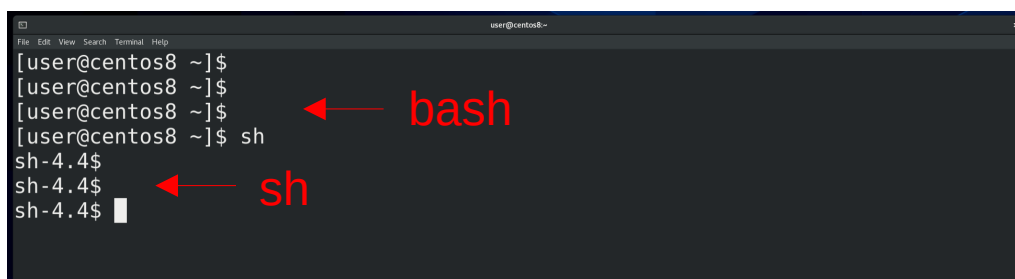
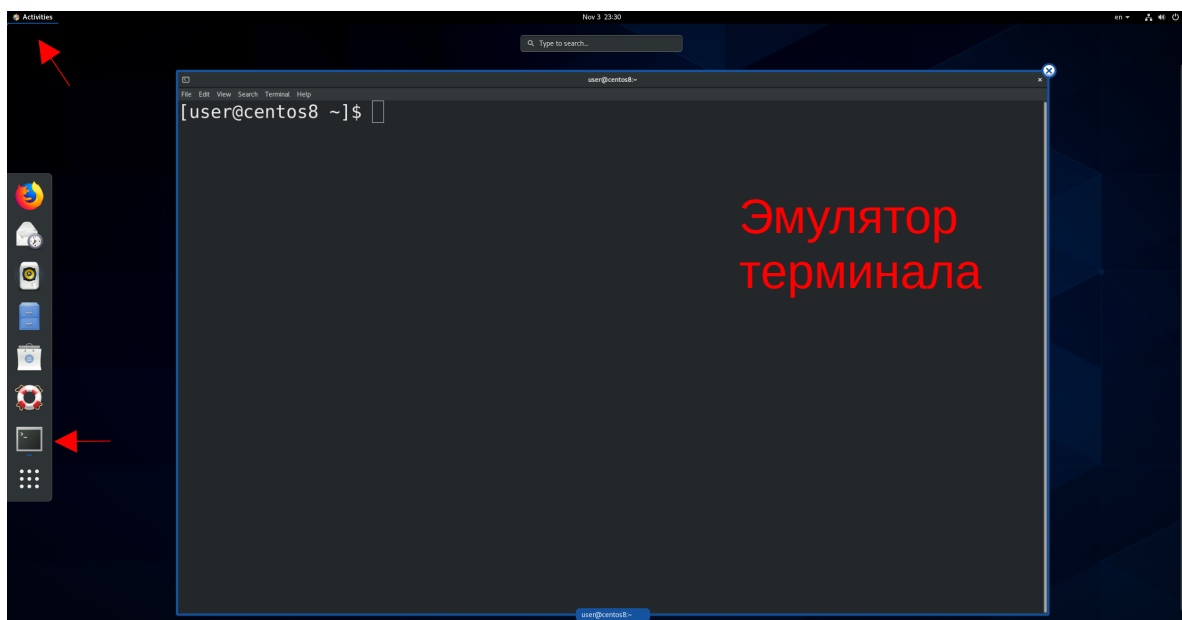


Чтобы пользователь мог взаимодействовать с компьютером, операционная система предоставляет два интерфейса – графический и текстовой. Есть заблуждение, что текстовой интерфейс нужен исключительно для задач администрирования, но, на самом деле, в текстовом интерфейсе можно слушать музыку, читать новости, общаться и даже сёрфить интернет. Графический или текстовой интерфейс – это понятие, а различные реализации называются оболочками.



```
user@centos8 ~]$  
user@centos8 ~]$  
user@centos8 ~]$ ← bash  
user@centos8 ~]$ sh  
sh-4.4$  
sh-4.4$ ← sh  
sh-4.4$
```

Благодаря тому, что дистрибутивы GNU/Linux разрабатывает не одна компания, а любой желающий, существует множество различных графических и текстовых оболочек, на любой вкус и цвет. Текстовые оболочки называют интерпретаторами командной строки или shell. По сути, это программы, которые могут отличаться функционалом, но стандартной оболочкой для UNIX-подобных операционных систем является sh, также называемая шелл. А одна из самых популярных называется bash. С помощью интерпретаторов пользователи могут делать практически всё, что может быть связано с запуском программ – ставить условия запуска, использовать переменные, повторять с разными значениями, автоматизировать и т.д. Одно из преимуществ bash над sh - автодополнение. Когда вы что-то печатаете на bash, часто вы можете нажать tab, который допишет команду до конца, а если есть несколько вариантов, то двойное нажатие по tab покажет их.



Есть 3 основных способа получить доступ к текстовому интерфейсу. При наличии графического интерфейса, чаще всего используется эмулятор терминала. Есть множество различных эмуляторов терминала, в основном они отличаются внешним видом и горячими клавишами.

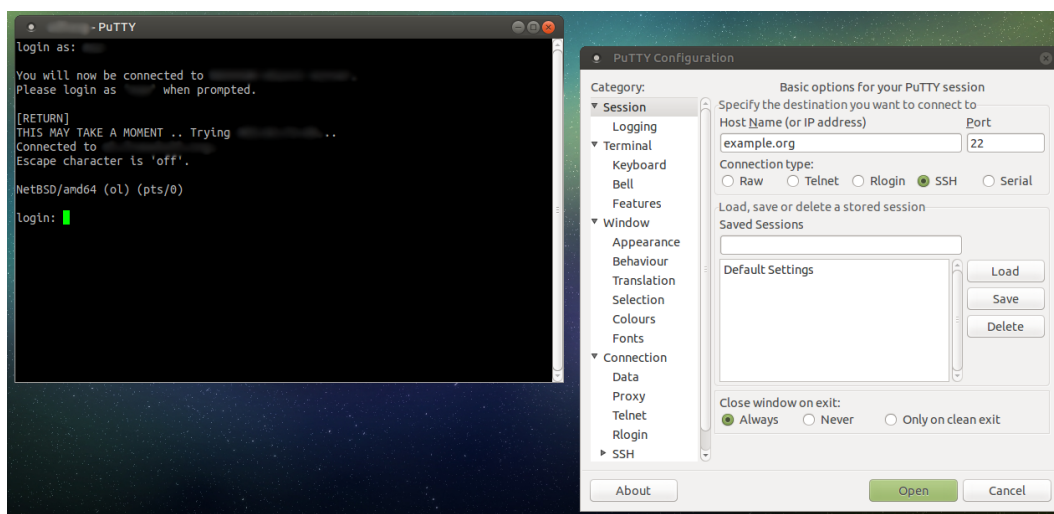
```
CentOS Linux 8 (Core)
Kernel 4.18.0-147.8.1.el8_1.x86_64 on an x86_64

Activate the web console with: systemctl enable --now cockpit.socket

centos8 login: user
Password:
Last login: Tue Nov  3 23:45:49 on tty3
[user@centos8 ~]$ _
```

Виртуальный  
терминал

При отсутствии графического интерфейса, либо при каких-либо проблемах с ним, используется виртуальная консоль, также называемая виртуальным терминалом. Их обычно много и к ним можно подключиться даже имея графический интерфейс, обычно для этого используется комбинация клавиш Alt + Ctrl + F1, F2 и т.д, обычно до F6. Так как VirtualBox перехватывает клавиши, на нём это можно сделать с помощью “правый Ctrl”+F1 и т.п. Графический интерфейс запускается на одной из виртуальных консолей.



Но, чаще всего, администраторы работают с компьютерами не напрямую, а удалённо. Для этого они используют программы для удалённого доступа, работающие с протоколом SSH. Одна из самых популярных на Windows называется Putty, но в Windows 10 это можно делать и без дополнительных программ, просто написав в терминале Windows команду ssh пользователь@айпи.адрес и введя пароль пользователя.

Программы, работающие в текстовом интерфейсе, часто называют командами. Этих команд огромное количество и знать всё наизусть не нужно. Большинство команд, с которыми вы будете работать, вы без труда запомните, так как их создавали для людей - они просты и логичны. Сложные команды всегда можно где-нибудь записать или найти. Чтобы знать, как работать с командой, вы должны знать её синтаксис.

```
user@centos8 ~$ ls
Desktop Documents Downloads Music Pictures Public temp Templates Videos
user@centos8 ~$ touch file FILE fILE fIlE
user@centos8 ~$ ls
Desktop Downloads file FILE Pictures temp Videos
Documents file fILE Music Public Templates
user@centos8 ~$ Ls
bash: Ls: command not found...
Similar command is: 'ls'
user@centos8 ~$ lS
bash: lS: command not found...
Similar command is: 'ls'
user@centos8 ~$
```

Давайте, для примера, рассмотрим команду `ls`. `ls` – от слова `list` – команда, с помощью которой можно просмотреть содержимое директории. Запомните – GNU/Linux – регистрозависимая операционная система. Если вы напишете `L` большое или `S` большое – получится уже другое значение, а так как таких команд нет, интерпретатор выдаст ошибку. То же самое касается файлов – файлы `file` маленькими, `FILE` большими, `fIlE`, `fIlE` и т.д. это разные файлы, которые никак друг с другом не конфликтуют. Но это больше относится файловым системам.

Когда у вас в терминале много текста это может напрягать, поэтому терминал можно легко очищать с помощью комбинации `Ctrl+L` или команды `clear`. А с помощью стрелок вы можете листать ранее выполненные команды.

```
user@centos8 ~$ ls -li
52439679 Desktop 52457985 fIlE 16798168 Pictures 34872740 Videos
51633878 Documents 52455083 fIlE 34872739 Public
2488280 Downloads 52455081 FILE 17103851 temp
52439663 file 2488281 Music 16798167 Templates
user@centos8 ~$ ls -li file
52439663 file
user@centos8 ~$
```

Возвращаясь к команде `ls`. У большинства команд есть различные опции и многим командам можно передавать значения. Для примера, команда `ls -li` покажет содержимое директории с соответствующими инодами. “`-li`” в данном случае – опция. Опции часто называют ключами команды. Если написать `ls -li file`, то мы передали команде ключ `-li` и значение в виде имени файла. Как результат, баш нам показал только информацию по одному файлу.

Что делать, если вы забыли или не знаете какой-то ключ или в целом синтаксис? Для этого в системе есть документация, которая также будет доступна во время экзамена по Red Hat. Кстати, об экзамене. Во время экзамена вы вольны пользоваться как графическим интерфейсом, так и текстовым, но, поверьте, на работу с графическим интерфейсом у вас просто не хватит времени. Работать в текстовом интерфейсе гораздо эффективнее. Возвращаясь к документации. Есть 3 основных способа получить доступ к документации – с помощью утилиты `man`, с помощью утилиты `info` и при помощи встроенной в большинство команд опции `help`.

```
user@centos8 ~]$ ls --help
Usage: ls [OPTION]... [FILE]...
List information about the FILES (the current directory by default).
Sort entries alphabetically if none of -cftuvSUX nor --sort is specified.

Mandatory arguments to long options are mandatory for short options too.
  -a, --all                do not ignore entries starting with .
  -A, --almost-all        do not list implied . and ..
      --author              with -l, print the author of each file
  -b, --escape              print C-style escapes for nongraphic characters
      --block-size=SIZE    with -l, scale sizes by SIZE when printing them;
                           e.g., '--block-size=M'; see SIZE format below
  -B, --ignore-backups     do not list implied entries ending with ~
  -c                        with -lt: sort by, and show, ctime (time of last
                           modification of file status information);
                           with -l: show ctime and sort by name;
```

Самый быстрый и простой способ – опция help. Как правило, это не документация, а небольшая подсказка по ключам и синтаксису, которая доступна, если писать команду, а затем два дефиса и help.

```
LS(1)                                User Commands                                LS(1)

NAME
  ls - list directory contents

SYNOPSIS
  ls [OPTION]... [FILE]...

DESCRIPTION
  List information about the FILES (the current directory by default).
  Sort entries alphabetically if none of -cftuvSUX nor --sort is specified.

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

man – утилита для чтения документации, доступная на большинстве UNIX-подобных систем. Допустим, меня интересует документация по команде ls. Я пишу man ls и открывается страница документации. Чтобы её закрыть, нажимаем q. q часто используется различными терминальными программами для выхода. Документацию можно листать с помощью стрелок и клавиш PgUp/PgDn.

```

-i, --inode
    print the index number of each file

-I, --ignore=PATTERN
    do not list implied entries matching shell PATTERN

/ -i

```

Если написать / , а потом какой-то текст, допустим ключ -i, и нажать enter программа попытается найти этот текст в документации. Как правило, в начале документации показывается имя программы, синтаксис, описание и ключи, а в конце у некоторых программ можно найти примеры команд в секции EXAMPLES, а также связанные программы.

```

user@centos8:~$ man -k password
Next: dir invocation, Up: Directory listing

10.1 'ls': List directory contents
=====

The 'ls' program lists information about files (of any type, including
directories). Options and file arguments can be intermixed arbitrarily,
as usual.

```

Что касается утилиты info (info ls), она пришла на замену man, хотя большинство до сих пор пользуются man. info лучше работает с большими документами и поддерживает гиперссылки.

```

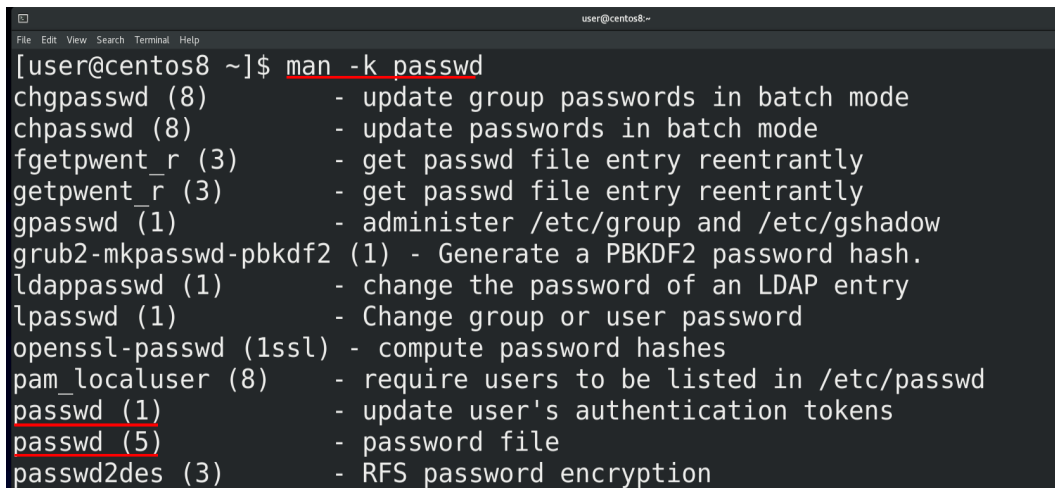
user@centos8:~$ man -k password
chage (1) - change user password expiry information
chpasswd (8) - update group passwords in batch mode
chpasswd (8) - update passwords in batch mode
cracklib-check (8) - Check passwords using libcrack2
create-cracklib-dict (8) - Check passwords using libcrack2
endpwent (3) - get password file entry
endspent (3) - get shadow password file entry
fgetpwent (3) - get password file entry
fgetspent (3) - get shadow password file entry
fgetspent_r (3) - get shadow password file entry

```

А что делать, если вы не знаете команду? В таком случае можно попытаться её найти. Кроме гугла, это можно сделать с помощью команды man с опцией -k, либо команды arporos. Допустим, я не знаю, как поменять пароль пользователю. Я могу написать man -k password и man выдаст мне все страницы документации, в секции NAME которых есть слово password. Результатов может быть много, но, если поискать, можно найти подходящий вариант.

```
1 Executable programs or shell commands
2 System calls (functions provided by the kernel)
3 Library calls (functions within program libraries)
4 Special files (usually found in /dev)
5 File formats and conventions eg /etc/passwd
6 Games
7 Miscellaneous (including macro packages and conventions), e.g.
  man(7), groff(7)
8 System administration commands (usually only for root)
9 Kernel routines [Non standard]
```

Кстати, вы заметили циферки рядом с названиями? Давайте посмотрим, что это такое. Для этого зайдём в документацию самого man, написав `man man`. Здесь написано, что man состоит из нескольких секций и эти цифры относятся к определённым секциям. Допустим, 1 – это про команды и функции оболочки, 5 – это про формат файлов и т.д.



```
user@centos8 ~
File Edit View Search Terminal Help
[user@centos8 ~]$ man -k passwd
chpasswd (8) - update group passwords in batch mode
chpasswd (8) - update passwords in batch mode
fgetpwent_r (3) - get passwd file entry reentrantly
getpwent_r (3) - get passwd file entry reentrantly
gpasswd (1) - administer /etc/group and /etc/gshadow
grub2-mkpasswd-pbkdf2 (1) - Generate a PBKDF2 password hash.
ldappasswd (1) - change the password of an LDAP entry
lpasswd (1) - Change group or user password
openssl-passwd (1ssl) - compute password hashes
pam_localuser (8) - require users to be listed in /etc/passwd
passwd (1) - update user's authentication tokens
passwd (5) - password file
passwd2des (3) - RFS password encryption
```

В некоторых случаях, для одного слова есть несколько страниц в документации в разных секциях, как например для `passwd`. `passwd` у нас это не только команда для смены пароля, но и специальный файл, в котором хранится информация о пользователях. Чтобы посмотреть документацию по команде, пишем `man 1 passwd`, а чтобы посмотреть документацию по файлу, пишем `man 5 passwd`.

```
user@centos8 ~$ history
47 history
48 ls
49 man man
50 man ls
51 man -k passwd
52 ls file
53 history
[user@centos8 ~]$ ls
Desktop  Downloads  file  FILE  Pictures  temp  Videos
Documents  file      FILE  Music  Public   Templates
[user@centos8 ~]$ !!
ls
Desktop  Downloads  file  FILE  Pictures  temp  Videos
Documents  file      FILE  Music  Public   Templates
[user@centos8 ~]$ !48
ls
Desktop  Downloads  file  FILE  Pictures  temp  Videos
Documents  file      FILE  Music  Public   Templates
[user@centos8 ~]$
```

Ну и напоследок. Всё что вы вводите в терминале, сохраняется в истории. Для просмотра введённых команд используйте команду `history`. Для повторения предыдущей команды используйте два восклицательных знака, а для повторения какой-то определённой команды из истории, напишите восклицательный знак и номер команды.

Я постарался рассмотреть самые базовые принципы работы с текстовым интерфейсом, но это очень огромная тема с большим количеством всяких плюшек, которые облегчают работу администратору. По мере того, как мы будем проходить другие темы, я буду рассказывать и о других нюансах, но на пока этого достаточно. Без практики вы быстро забудете все команды, поэтому обязательно уделите время практике.