

Основные команды Bash

Перемещение по файловой системе и работа с ней.

1. cd

```
cd </path/to/directory>
```

Сокращенное *change directory*. Позволяет перемещаться по файловой системе.

Путь может быть как абсолютным (*начинаться с /*), так и относительным (*подразумевается расположение пути относительно текущего*).

```
dize@jupiter:~/Documents
dize@jupiter ~ $
dize@jupiter ~ $ cd /home/dize/Documents/
dize@jupiter ~/Documents $
```

- **cd** без параметров перемещает в домашнюю директорию пользователя.
- **cd** - перемещает по прошлому пути.
- **.** (точка) текущая директория
- **..** (две точки) родительская директория
- **~** (тильда) домашняя директория пользователя

2. pwd

```
pwd
```

Сокращение от *print work directory*. Выводит текущий абсолютный путь.

```
dize@jupiter:~/Documents
dize@jupiter ~/Documents $ pwd
/home/dize/Documents
dize@jupiter ~/Documents $
```

3. ls

```
ls </path/to/directory>
```

Сокращение от *list*. Отображает все **файлы** и **директории** в директории </path/to/directory>.

```
dize@jupiter:~/Documents
dize@jupiter ~/Documents $ ls
123.odt  bu   ctf_meetup  learn  script  test  xpe.odt
2.5.odt  CTF  ictf        poib   sfedu   tum
dize@jupiter ~/Documents $
```

- **ls** без параметров отображает все файлы и директории по **текущему** пути
- **ls -a** отображает скрытые файлы и папки
- **ls -l** отображает расширенную информацию о файлах и папках

4. cat

```
cat <file_name>
```

Отображает содержимое файла file_name.

```
dize@jupiter:~/Documents/sfedu
dize@jupiter ~/Documents/sfedu $ cat simple.txt
hello world!
dize@jupiter ~/Documents/sfedu $ █
```

5. less

```
less <file_name>
```

Отображает содержимое файла file_name, выводит только содержимое помещающееся в окно терминала. Навигация по файлу возможна **клавишами-стрелками**, кнопка "q" закрывает просмотр.

```
dize@jupiter:~/Documents/sfedu
dize@jupiter ~/Documents/sfedu $ less tum.md █
```

6. file

```
file <file_name>
```

Выводит на экран тип файла. Настоящий тип, а не указанный в расширении.

```
dize@jupiter:~/Download
dize@jupiter ~/Download $ file install.zip
install.zip: Zip archive data, at least v2.0 to extract
dize@jupiter ~/Download $ █
```

7. cp

```
cp <file1> <file2>
```

Сокращение от **copy**. Копирует файл **file1** по пути **file2**.

8. mv

```
mv file1 file2
```

Сокращение от **move**.

Перемещает файл **file1** по пути **file2**.

9. rm

```
rm <file1>
```

Сокращение от **remove**.

Удаляет файл **file1**.

- **rm -r** удаляет директорию

10. mkdir

```
mkdir <directory>
```

Сокращение от **make directory**.

Создает директории с именем **directory**.

11. locate

```
locate <file>
```

Возвращает все пути с вхождениями **file** в них.

12. man

```
man <cmd>
```

Отображает справку по команде **cmd**.

13. echo

```
echo <string>
```

Выводит заданную строку **string** на экран.

14. strings

```
strings <file1>
```

Выводит все строки с печатными символами из файла **file1** на экран.

Перенаправление вывода

Bash позволяет перенаправлять стандартный вывод в файлы с помощью оператора **>**. Если файл не существовал, он будет создан, иначе он будет перезаписан.

```
$ echo hello! > 1.txt
$ ls
1.txt
$ cat 1.txt
hello!
```

Если требуется *дописать* строку в файл используется оператор **>>**.

Конвейер

Если нужно перенаправить вывод одной команды на вход другой используется оператор **|**.

```
$ locate bin | less
```

Поиск по содержимому файлов

```
$ grep [opts] "pattern" file
$ cat file | grep [opts] "pattern"
```

- v - вывести строки, где не встречается образец
- i - игнорировать регистр символов
- n - печатать номера строк
- a - искать по бинарным файлам
- E - расширенные регулярные выражения
- o - вывести только совпадающую с образцом часть строки

Регулярные выражения

```
$ egrep [opts] "pattern" file
$ cat file | egrep [opts] "pattern"
```

- ^ "^license" - начало строки
- \$ "license\$" - конец строки
- [abc] "[0-9]" - любой символ из перечисленных
- [^abc] "[^]1" - любой кроме перечисленных
- [ab]{8} "w{3}" - повторяется несколько раз
- (abc) "(license)" - группа символов
- ."Li.en.e" - любой символ
- * "\([a-z]*\)" - повторить ноль или больше раз
- + "\(a.+\\)" - повторить один или больше раз
- ? "(copy)?right" - повторить ноль или один раз

Обработка текста

Вывести символы со 2 по 5 каждой строки

```
$ cat file | cut -c 2-5
```

Отсортировать строки по алфавиту

```
$ cat file | sort
```

Удалить одинаковые строки, идущие подряд

```
$ cat file | uniq
```

Вывести уникальные строки

```
$ cat file | sort | uniq
$ cat file | awk 'program'
```

'{print \$0}' - вывести каждую строку '{print \$1}' - вывести первое слово каждой строки '{print "1: " \$1 ", 2: " \$2}' - вывести для каждой строки 1:<слово 1>, 2:<слово 2>