



Введение в `bash`. Работа с командной строкой

Подключение к серверу через `ssh`. Навигация и работа с каталогами. Работа с файлами и правами доступа.

18 сентября 2026 г.

Иван Ильницкий
ФББ МГУ

Операционные системы (ОС)

ОС - это системное программное обеспечение, которое управляет аппаратной частью компьютера и предоставляет основные функции для выполнения приложений



Операционные системы (ОС)

ОС - это системное программное обеспечение, которое управляет аппаратной частью компьютера и предоставляет основные функции для выполнения приложений

Преимущества Linux в работе с биоинформатическими данными:

- ❖ Открытый исходный код и бесплатное использование.
- ❖ Широкий выбор инструментов для разработки и анализа данных.
- ❖ Стабильность и безопасность.

Консоль. Среда Bash и её роль



Что такое bash? Интерпретатор команд, который позволяет пользователям взаимодействовать с системой с помощью текстовых команд.

```
kevin@localhost:~$ ls
#log.txt# boot dev init log.txt mnt root srv usr
acct cache etc lib lost+found opt run sys var
bin data home lib64 media proc sbin
kevin@localhost:~$ cd /mnt/
c/ d/ e/ f/
kevin@localhost:~$ cd /mnt/c/
kevin@localhost:~$ cd server/
kevin@localhost:~$ cd js
kevin@localhost:~$ emacs
```

Консоль. Среда Bash и её роль



Что такое bash?: Интерпретатор команд, который позволяет пользователям взаимодействовать с системой с помощью текстовых команд.

Особенности:

- ❖ Автоматизация задач: Написание скриптов для автоматизации повторяющихся операций.
- ❖ Управление файлами и системами: Легкий доступ и управление файлами, пакетами и ресурсами.
- ❖ Взаимодействие с удалёнными серверами: Использование SSH для удалённого доступа.
- ❖ Широкий набор встроенных команд и функций. Мощные программы для обработки текстовых данных (утилиты `grep`, `awk`, `sed`, ...).

Начало работы с Bash

- ❖ Откроем терминал/командную строку на системе
- ❖ Приглашение (например, `$` или `#`) указывает на то, что вы готовы вводить команды.
- ❖ Команды чувствительны к регистру и часто сопровождаются аргументами и опциями.
- ❖ Используйте команду `man`, чтобы получить информацию/мануал о любой команде.
- ❖ Введите `exit`, чтобы закрыть сеанс терминала.

Горячие клавиши в Bash

- **Tab** – автозаполнение
- **↑, ↓** – перемещение между введенными командами
- **Ctrl + C** – прервать выполнение текущей команды
- **Ctrl + D** – закрыть сессию терминала / прервать ввод
- **Ctrl + Shift + C; Ctrl + Shift + V** – скопировать; вставить
- **Ctrl + A** – переместиться в начало строки
- **Ctrl + E** – переместиться в конец строки
- **Ctrl + Z** – приостановить выполнение текущей команды
- **Ctrl + W** – удалить одно слово слева
- **Ctrl + R** – поиск по истории команд
- **Ctrl + S; Ctrl + Q** – приостанавливает вывод в терминале; возобновляет

Локальный компьютер vs Удалённый сервер

❖ Локальный компьютер:

- ✓ Находится в непосредственном доступе пользователя.
- ✓ Более ограниченные вычислительные ресурсы.
- ✓ Прямой доступ к аппаратному обеспечению.
- ✓ Используется для разработки и тестирования.

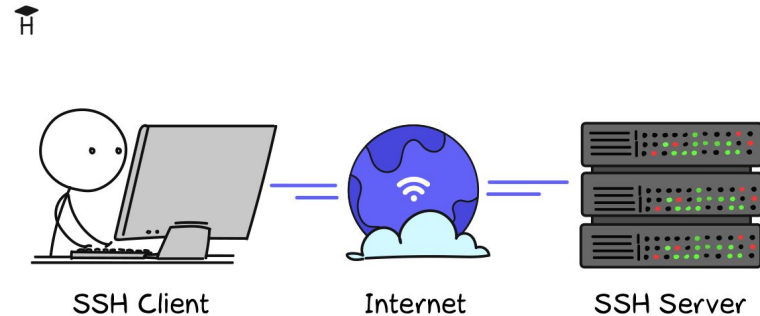
❖ Удалённый сервер:

- ✓ Может находиться в другом физическом месте.
- ✓ Большие вычислительные мощности, чаще всего используется для масштабных задач.
- ✓ Доступ осуществляется через сеть (SSH).
- ✓ Используется для финальных вычислений, масштабируемых решений.
- ✓ Могут отсутствовать права для установки программ и библиотек

SSH-подключение на удаленный сервер

- ❖ *SSH (Secure Shell)* – сетевой протокол для безопасного соединения и передачи данных между двумя вычислительными машинами

- ❖ Предустановлен на устройствах с Linux, MacOS и Windows, облегчает удалённый доступ к серверам и другим устройствам



Подключение к серверу через ssh

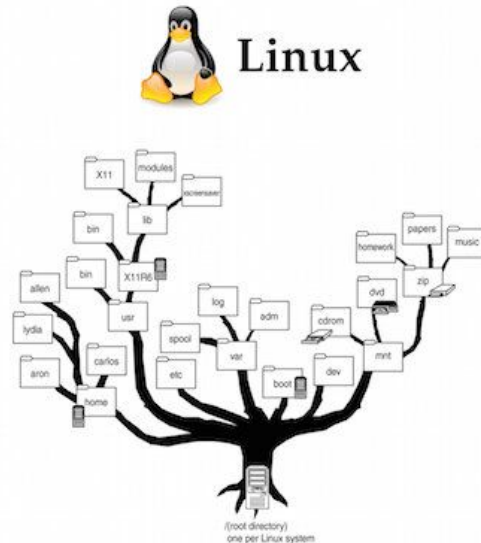
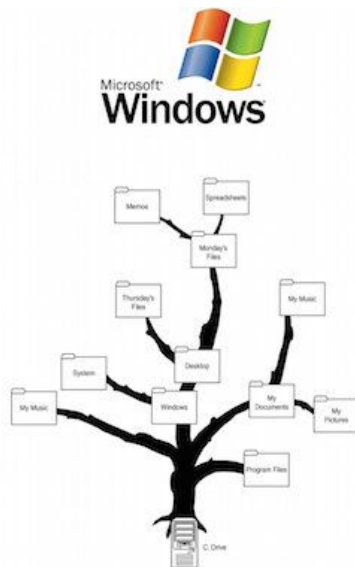
- ❖ Одноименная команда `ssh` широко используется для удаленного входа на сервер, безопасного управления файлами и выполнения команд на удаленной машине.
- ❖ Использование: `ssh [опции] имя_пользователя@имя_хоста`
Имя хоста – IP-адрес или доменное имя
- ❖ Опции: `-p` (указать порт)
- ❖ Примеры:
`ssh orca@bioinf.fbb.msu.ru`
`ssh fox@92.102.54.34`
`ssh -p 22221 user@kodomom.fbb.msu.ru`
- ❖ Вам потребуется имя пользователя и пароль ИЛИ аутентификация с помощью SSH-ключа.
- ❖ Команды, выполненные на удаленном сервере, отображаются локально.

Файловая система

Файловая система — метод, по которому файлы упорядочиваются и хранятся на устройствах хранения.

Иерархическая структура: Понимаем путь от корневого (root) каталога до конечного файла.

Права доступа: Чтение, запись, выполнение для файлов и каталогов.



Команда *pwd*

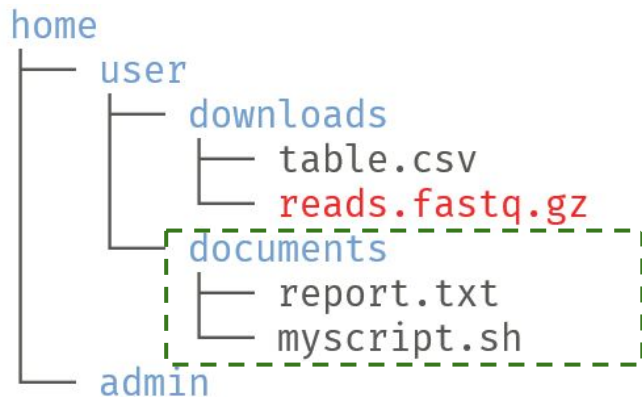
- ❖ Команда **pwd** ('Print Working Directory') отображает путь до рабочей директории в терминале, показывая местоположение пользователя в файловой системе.

Команда:

```
$ pwd
```

Результат

```
/home/user/documents
```



Абсолютный путь до файла или каталога

- ❖ Абсолютный путь начинается с корневого каталога (начинается с символа `/`) и указывает на точное местоположение файла или каталога в файловой системе.
- ❖ Пример: `/home/user/documents/report.txt` - абсолютный путь к файлу `report.txt` в каталоге `documents` пользователя `user`.
- ❖ Всегда однозначно определяет местоположение файла или каталога.
- ❖ Используется для доступа к ресурсам независимо от текущего каталога.



Относительный путь до файла или каталога

- ❖ Относительный путь определяет местоположение файла или директории относительно текущей директории .
- ❖ Пример: `downloads/reads.fastq.gz` - относительный путь к файлу `reads.fastq.gz` в каталоге `downloads`, который находится в текущей директории.
- ❖ Зависит от текущей рабочей директории.
- ❖ Более компактны, чем абсолютные пути, не начинаются с символа '/'.



Команда ls

- ❖ Команда `ls` выводит список всех файлов и директорий в рабочей/текущей директории, предоставляя информацию, такую как права доступа, владелец, размер и временные метки.

Команда ls

- ❖ Команда **ls** выводит список всех файлов и директорий в рабочей/текущей директории, предоставляя информацию, такую как права доступа, владелец, размер и временные метки.
- ❖ Основные опции:
 - l** – Отображает подробный список в формате «длинного» списка, включая права доступа, количество ссылок, владельца, группу, размер и время последнего изменения.
 - a** – Показывает все файлы и директории, включая скрытые (имена которых начинаются с точки **.** , например, директория **.ssh**).
 - t** – Сортирует файлы по времени последнего изменения, начиная с самого нового.
 - h** – Используется вместе с **-l** для вывода размеров файлов в удобочитаемом формате (например, КБ, МБ).
 - R** – Рекурсивно открывает содержимое каталогов, показывая все файлы и директории
 - S** – Сортирует файлы по размеру, начиная с самого большого.

Команда ls

```
home
├── user
│   ├── downloads
│   │   ├── table.csv
│   │   └── reads.fastq.gz
│   └── documents
│       ├── report.txt
│       └── myscript.sh
└── admin
```

```
> ls /home/user/
documents  downloads
```

```
> ls -l /home/user/
total 8
```

```
drwxr-xr-x 2 backup users 4096 Sep 10 02:37 documents
drwxr-xr-x 2 backup users 4096 Sep 10 02:05 downloads
```

CD

❖ Команда `cd` ("change directory") используется для перемещения между директориями (папками) в файловой системе.

❖ Использование: `cd [каталог]`

`cd downloads`

`cd /home/user/downloads`

Сокращения: '~', '!', '..'

'~' - Сокращение для домашней директории пользователя (обычно /home/имя_пользователя).

'.' - Текущий каталог.

'..' - Родительский каталог (каталог уровня выше текущего).

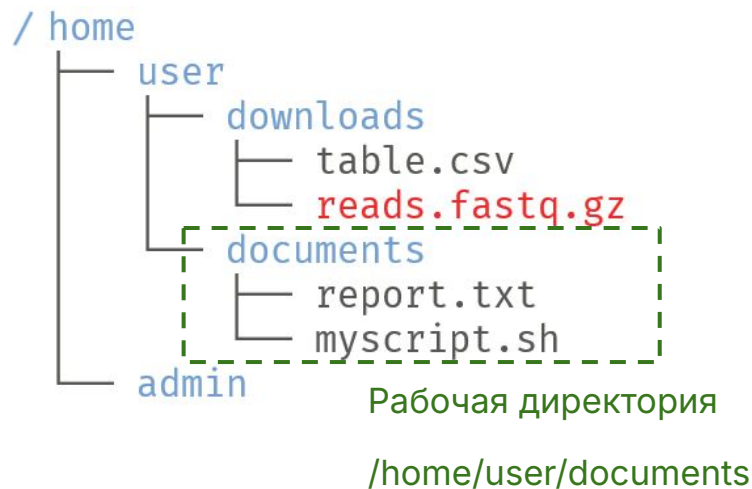
❖ Используются для более краткого и удобного обозначения путей.

❖ Примеры:

`~/documents` (домашняя директория /documents),

`./report.txt` (файл report.txt в текущей директории),

`../.. /admin` (директория admin на два уровня выше рабочей директории).



Примеры использования сокращений

```
cd /home/user/downloads = cd ~/downloads
```

```
cd ~
```

```
cd ..
```

```
cd ../ ../ ..
```

Команда tree

Быстрая визуальная проверка
структуры каталогов.

`tree`

```
home
├── user
│   ├── downloads
│   │   ├── table.csv
│   │   └── reads.fastq.gz
│   └── documents
│       ├── report.txt
│       └── myscript.sh
└── admin
```

Права доступа

❖ Типы прав доступа

- ✓ **r** (read): право чтения файла или списка содержимого директории.
- ✓ **w** (write): право изменения (записи) содержимого файла или создания/удаления файлов в директории.
- ✓ **x** (execute): право выполнения файла (для директории - право перехода в директорию).

Формат отображения прав:

Первый символ: тип файла (- для обычного файла, **d** для директории, **l** для символической ссылки)

-rw-r--r--

file.txt

drwxr-xr-x

directory

права всех остальных
права группы
права владельца

Для просмотра текущих прав можно использовать команду `ls -l`.

Права доступа

```
[drwxr-xr-x] home/
├── [drwxr-xr-x] user
│   ├── [drwxr-xr-x] downloads
│   │   ├── [-rw-r--r--] table.csv
│   │   ├── [-rw-r--r--] reads.fastq.gz
│   │   └── [drwxr-xr-x] documents
│   │       ├── [-rw-r--r--] report.txt
│   │       └── [-rw-r--r--] myscript.sh
└── [drwxr-xr-x] admin
```

Права доступа

- ❖ Для изменение прав на доступ существует команда

chmod — изменяет права доступа к файлу/директории

Примеры:

```
chmod u+x documents/myscript.sh
```

```
chmod g+w downloads/table.csv
```



Операции с файлами

- `less [файл]` – просмотр файла с возможностью листать файл
- `cat [файл]` – полностью выводит содержимое файла в стандартный вывод
- `head [файл]` – выводит первые несколько строк файла
- `tail [файл]` – выводит последние несколько строк файла

❖ Примеры:

```
less documents/myscript.sh  
cat file.txt  
tail -n 3 file.txt
```

“q” – выйти из less



Создание и удаление директорий `mkdir`, `rm`, `rmdir`

`mkdir [опции] [директория]` — создает новую директорию

Опции: `-p` (создать вложенные директории)

```
mkdir new_folder
```

```
mkdir folder1 folder2 folder3
```

```
mkdir -p parent/child/grandchild
```

`rmdir [опции] [директория]` — удаляет директорию

```
rmdir new_folder
```

`rm [опции] [файл/директория]` — удаляет файлы и директории (`-r`)

Опции: `-r` (рекурсивно удаляет директории с их содержимым)

```
rm ~/file
```

```
rm -r folder
```

```
rm alpha1 alpha2 alpha3 = rm alpha*
```

Удаление с помощью `rm` необратимо, особенно аккуратно стоит быть с флагом `-f`, который подавляет запросы на подтверждение операции

Копирование и перемещение файлов: mv, cp

Опции:

-r (рекурсивное копирование/перемещение директорий)

Примеры:

mv:

`mv file1.txt file2.txt` (переименование)

`mv file.txt /home/user/docs/` (перемещение)

`mv dir1 dir2` (перемещение/переименование директории)

cp:

`cp file.txt file_copy.txt` (копирование файла)

`cp -r dir1 /home/user/docs/` (рекурсивное копирование директории)

`cp file1.txt file2.txt /home/user/backup/` (копирование нескольких файлов)

Команда WC

- ❖ Используется для подсчета количества строк, слов, символов или байт в файле/файлах

`wc [опции] [файл]`

Опции: `-m` (количество символов), `-l` (количество строк), `-w` (количество слов), `-c` (количество байт)

Полезные советы для повышения эффективности:

- ❖ Используйте автозаполнение с помощью клавиши *Tab*, чтобы сэкономить время на наборе текста.
- ❖ Горячие клавиши для перемещения в командной строке.
- ❖ Не ленитесь читать мануалы (команда `man`) для получения детальной информации о командах.

WSL

- ❖ *WSL (Windows Subsystem for Linux)*: Позволяет запускать Linux-дистрибутивы на Windows. Это мощный инструмент для пользователей, которым необходимо использовать Linux-команды и софт на Windows.

Загрузка файлов из интернета: *wget*

wget:

Основная ориентированность на скачивание файлов.

Поддерживает рекурсивное скачивание (можно скачивать целые сайты с директориями).

- ❖ `wget [опции] [ссылка]`
- ❖ Опции: `-c` (продолжить загрузку), `-P` (указать директорию для скачивания), `-r` (рекурсивное скачивание), `-O` (задать имя скачанному файлу)
- ❖ Примеры:
`wget http://example.com/file.txt`
`wget -r http://example.com/`
`wget -P directory http://example.com/`
`wget -O arch.zip http://example.com/archive2024.zip`