

Bash: История команд – history.

Category: bash & sh,GNU/Linux,history
2025-02-12

Работа с историей команд в **bash** является очень удобным инструментом, который позволяет быстро повторять ранее введенные команды.

Вот основные способы работы с историей, включая примеры:

Просмотр истории команд.

`history` – Если просто ввести команду `history`, вы получите список всех команд, которые были введены в текущей сессии или ранее, в зависимости от настроек.

```
$ history
```

Эта команда выведет список всех команд, которые вы вводили в текущей сессии и/или ранее, в зависимости от настроек.

Хранится список в файле `.bash_history`, который расположен в корневом каталоге пользователя `~/.bash_history`.

```
$ cat .bash_history
```

Выполнение команд из истории.

`!!` – Выполнить последнюю введенную команду.

`!n` – Выполнить команду с номером `n` из списка истории, который вы видите при помощи команды `history`. Например, `!42` выполнит 42-ю команду.

`!-n` – Выполнить команду, которая была введена `n` команд назад. Например, `!-2` выполнит команду, введенную два раза ввода в консоль назад.

`!string` – Выполнить последнюю команду, начинающуюся с `string`. Например, если последняя команда была `echo hello`, команда `!echo` выполнит её снова.

Выполнение последней команды.

```
$ !!
```

Выполните последнюю введенную команду.

Например, если последняя команда была `ls -l`, то `!!` выполнит её снова.

Выполнение команды по номеру.

```
$ !42
```

Если вы видите, что 42-я команда в списке – это `git status`, то `!42` выполнит `git status`.

Выполнение команды, введенной n команд назад.

```
$ !-2
```

Эта команда выполнит команду, которую вы вводили два раза ввода в консоль назад.

Выполнение крайней команды, начинающейся с заданной строки.

```
$ !git
```

Если крайняя команда, начинающаяся с `git`, была `git commit`, то это выполнит `git commit`.

Поиск по истории.

`Ctrl + r` – Запустить поиск в обратном направлении. Нажимая комбинацию `Ctrl + r`, вы можете начинать вводить текст, и **bash** будет искать команды, которые содержат этот текст. Нажмите `Ctrl + r` повторно, чтобы найти предыдущие совпадения.

`Ctrl + s` – В некоторых системах это может позволить вам искать вперёд, однако в большинстве значений по умолчанию эта комбинация обычно замораживает терминал. Для её использования может потребоваться отключить функцию управления терминалом.

Поиск в обратном направлении.

Нажмите `Ctrl + r`. Начнется поиск команд в обратном порядке. Введите часть команды, и вы увидите совпадения. Если вы хотите найти предыдущие совпадения, нажмите `Ctrl + r` снова.

Пример поиска по строке `git`:

После нажатия `Ctrl + r`, введите `git`, вы увидите, например, `git status`, и если нажмете `Ctrl + r` снова, увидите `git commit`.

Настройки истории.

Файл хранения истории: История команд по умолчанию сохраняется в файле `~/.bash_history`.

Вы можете изменить этот файл с помощью переменной `HISTFILE`:

```
export HISTFILE=~/.my_history_file
```

Установка количества сохраняемых команд.

Количество команд в истории: `HISTSIZE` и `HISTFILESIZE` определяют, сколько команд будет храниться в памяти и в файле соответственно.

Например, чтобы хранить 1000 команд, в вашем файле `~/.bashrc` добавьте:

```
export HISTSIZE=1000
export HISTFILESIZE=2000
```

Это задаст количество команд, которые будут храниться в памяти и в файле истории.

Игнорирование дубликатов.

Опции для управления поведением истории: Пользователь может настроить такие параметры, как игнорирование дубликатов или игнорирование некоторых команд.

Чтобы избежать хранения дубликатов и игнорирования команд, начинающихся с пробела, добавьте в `~/.bashrc`:

```
export HISTCONTROL=ignoredups:ignorespace
```

Автоматическое сохранение истории.

Сохранять историю сразу после выполнения команды: По умолчанию история сохраняется при выходе из оболочки.

Установите следующую переменную в `~/.bashrc`, чтобы команды добавлялись в историю сразу после их выполнения:

```
export PROMPT_COMMAND="history -a"
```

Очистка и удаление истории.

`history -d n` – Удалить строку с номером `n` из истории.

`history -c` – Очистить всю историю текущей сессии.

`rm ~/.bash_history` – Удаление файла с историей в будущем. Чтобы изменения вступили в силу, выйдите из сессии или перезапустите оболочку.

Удаление конкретной команды.

```
$ history -d 42
```

Эта команда удалит 42-ю команду из вашей истории.

Очистка всей истории.

```
$ history -c
```

Это очищает историю текущей сессии.

Удаление файла истории.

```
$ rm ~/.bash_history
```

Удаляет файл истории. В следующей сессии история будет новой.

Использование истории в скриптах.

Историю можно также интегрировать в скрипты для автоматизации, читая файл `~/.bash_history` и анализируя его содержимое для выполнения определённых действий на основе ранее введённых команд.

Например, для анализа или повторного выполнения команд:

```
#!/bin/bash
while read command; do
echo "Executing: $command"
eval "$command"
done < ~/.bash_history
```

Это простой скрипт, который прочитает каждую команду из истории и выполнит её.

Заключение.

Работа с историей команд в **bash** — это полезный функционал, который помогает значительно ускорить работу и упростить ввод повторяющихся команд. Используя эти команды и техники, вы сможете эффективно управлять своей историей в **bash**.

Оригиналы источников информации.

1. younglinux.info «История команд и автодополнение».