

# 42 КОМАНДЫ LINUX КОТОРЫЕ ВЫ ДОЛЖНЫ ЗНАТЬ

Списки    Январь 1, 2016    10    admin    bash, команды

---

Во всех операционных системах, в том числе и в Linux, термин 'команда' означает либо утилиту командной строки, либо определённую возможность, встроенную в командную оболочку системы. Тем не менее, для самих пользователей это различие не имеет особого значения. В конце концов, и те и другие команды терминала Linux вызываются одинаково. Вы вводите слово в вашем эмуляторе терминала и получаете результат выполнения команды.

Я уже писал о командах терминала Linux, но тогда затронул лишь несколько самых интересных, самых полезных команд, рассчитывая на то, что пользователь уже достаточно знаком с возможностями терминала. Но нужно сделать ещё одну статью, рассчитанную на новичков, тех, кто только делает свои первые шаги в освоении Linux.

И вот она. Её цель — собрать основные простые и сложные команды Linux, которые должен знать каждый пользователь, чтобы наиболее эффективно управлять своей системой. Для удобства запоминания опций команд я в скобках добавил слова, от которых они произошли — так намного проще, на себе проверено.

Это не значит, что я перечислю все команды — я постараюсь охватить всё самое полезное, то, что может пригодиться в повседневной жизни. Чтобы было удобнее читать, разделим этот список на категории команд по назначению. Большинство рассмотренных здесь утилит не требуют дополнительной установки, они будут предустановлены в

любом дистрибутиве Linux, а если не будут, то их несложно найти в официальных репозиториях.

Содержание статьи:

- Команды Linux для управления файлами
- Linux команды консоли для работы с текстом
- Команды Linux для управления процессами
- Команды Linux окружения пользователя
- Команды Linux для управления пользователями
- Linux команды для просмотра документации
- Команды Linux для управления сетью
- Выводы

## КОМАНДЫ LINUX ДЛЯ УПРАВЛЕНИЯ ФАЙЛАМИ

### 1. LS

Утилита для просмотра содержимого каталогов. По умолчанию показывает текущий каталог. Если в параметрах указать путь, то она перечислит содержимое конечного каталога. Полезные опции -l (List) и -a (All). Первая форматирует вывод в виде списка с более подробной информацией, а вторая включает показ скрытых файлов.

### 2. CAT

Печатает содержимое файла, переданного в параметре, в стандартный вывод. Если передать несколько файлов, команда склеит их. Также можно перенаправить вывод в ещё один файл с помощью символа '>'. Если нужно вывести только определенное количество строк, используйте опцию -n (Number).

### 3. CD

Позволяет перейти из текущего каталога в указанный. Если запустить без параметров — возвращает в домашний каталог. Вызов с двумя точками возвращает на уровень вверх относительно текущего каталога. Вызов с тире (cd -) возвращает к предыдущему каталогу.

### 4. PWD

Печатает на экран текущий каталог. Это может быть полезно, если ваша командная строка Linux не выводит такую информацию. Эта команда будет востребована в Bash программировании, где для получения ссылки на каталог выполняется скрипт.

## 5. MKDIR

Создание новых каталогов. Наиболее удобная опция -p (**P**arents), позволяет создать всю структуру подкаталогов одной командой, даже если они ещё не существуют.

## 6. FILE

Показывает тип файла. В Linux файлы не обязаны всегда иметь расширения для того, чтобы с ними работать. Поэтому пользователю иногда трудно определить, что за файл перед ним. Эта маленькая утилита решает проблему.

## 7. CP

Копирование файлов и каталогов. Она не копирует каталоги по умолчанию рекурсивно (то есть все поддиректории и все файлы в поддиректориях), поэтому не забудьте добавить опцию -r (**R**ecursive) или -a (**A**rchive). Последняя включает режим сохранения атрибутов, владельца и временного штампа в дополнение к рекурсивному копированию.

## 8. MV

Перемещение или переименование файлов и каталогов. Примечательно, что в Linux это одна и та же операция. Переименование — это перемещение файла в ту же папку с другим именем.

## 9. RM

Удаляет файлы и папки. Очень полезная команда Linux: с её помощью вы можете убрать весь беспорядок. Если нужно рекурсивное удаление, используйте опцию -r. Однако будьте осторожны: конечно, для того чтобы повредить систему вам нужно будет серьёзно постараться, однако можно удалить собственные важные файлы. Rm удаляет файлы не в корзину, из которой потом всё можно будет восстановить, а полностью стирает. Действия оператора **rm** необратимы. Поверьте, ваши оправдания в духе «rm съела мою курсовую» никому не будут интересны.

## 10. LN

Создает жёсткие или символические ссылки на файлы. Символические или программные ссылки — это что-то похожее на ярлыки в Windows. Они предоставляют удобный способ доступа к определённому файлу. Символические ссылки указывают на файл, но не имеют никаких метаданных. Жёсткие ссылки, в отличие от символических, указывают на физический адрес области диска, где хранятся данные файла.

## 11. CHMOD

Изменяет права доступа к файлу. Это чтение, запись и выполнение. Каждый пользователь может изменять права для своих файлов.

## 12. CHOWN

Изменяет владельца файла. Только суперпользователь может изменять владельцев. Для рекурсивного изменения используйте опцию -R.

## 13. FIND

Поиск в файловой системе, файлах и папках. Это очень гибкая и мощная команда Linux не только из-за своих способностей ищeyки, но и благодаря возможности выполнять произвольные команды для найденных файлов.

## 14. LOCATE

В отличие от find, команда locate ведёт поиск в базе данных updatedb для шаблонов имён файлов. Эта база данных содержит снимок файловой системы, что позволяет искать очень быстро. Но этот поиск ненадёжен, потому что вы не можете быть уверены, что ничего не изменилось с момента последнего снимка.

## 15. DU

Показывает размер файла или каталога. Самые полезные опций: -h (**H**uman), которая преобразует размеры файлов в легко читаемый формат, -s (**S**ummarize), которая выводит минимум данных, и -d (**D**epth), устанавливающая глубину рекурсии по каталогам.

## 16. DF

Анализатор дискового пространства. По умолчанию вывод достаточно подробный: перечислены все файловые системы, их размер, количество использованного и свободного пространства. Для удобства есть опция -h, делающая размеры легко читаемыми.

## 17.DD

Как сказано в официальном руководстве, это команда терминала для копирования и преобразования файлов. Не очень понятное описание, но это всё, что делает dd. Вы передаёте ей файл-источник, пункт назначения и пару дополнительных опций. Затем она делает копию одного файла в другой. Вы можете задать точный размер данных,

которые нужно записать или скопировать. Работает утилита со всеми устройствами. Например, если вы хотите перезаписать жёсткий диск нулями из `/dev/zero`, можете сделать это. Также она часто используется для создания LiveUSB или гибридных ISO образов.

## 18 MOUNT / Umount

Это команды консоли Linux для подключения и отключения файловых систем Linux. Можно подключать всё: от USB накопителей, до ISO образов. И только у суперпользователя есть права для этого.

# LINUX КОМАНДЫ КОНСОЛИ ДЛЯ РАБОТЫ С ТЕКСТОМ

## 19. MORE / LESS

Это две простенькие команды терминала для просмотра длинных текстов, которые не вмещаются на одном экране. Представьте себе очень длинный вывод команды. Или вы вызвали `cat` для просмотра файла, и вашему эмулятору терминала потребовалось несколько секунд, чтобы прокрутить весь текст. Если ваш терминал не поддерживает прокрутки, вы можете сделать это с помощью `less`. `Less` новее, чем `more` и поддерживает больше опций, поэтому использовать `more` нет причин.

## 20. HEAD / TAIL

Ещё одна пара, но здесь у каждой команды своя область применения. `Head` выводит несколько первых строк из файла (голова), а `tail` выдает несколько последних строк (хвост). По умолчанию каждая утилита выводит десять строк. Но это можно изменить с помощью опции `-n`. Ещё один полезный параметр `-f`, это сокращение от **f**ollow (следовать). Утилита постоянно выводит изменения в файле на экран. Например, если вы хотите следить за лог файлом, вместо того, чтобы постоянно открывать и закрывать его, используйте команду `tail -nf`.

## 21. GREP

`Grep`, как и другие инструменты Linux, делает одно действие, но делает его хорошо: она ищет текст по шаблону. По умолчанию она принимает стандартный ввод, но вы можете искать в файлах. Шаблон может быть строкой или регулярным выражением. Она может вывести как совпадающие, так и не совпадающие строки и их контекст. Каждый раз, когда вы выполняете команду, которая выдает

очень много информации, не нужно анализировать всё вручную — пусть `grep` делает свою магию.

## 22. SORT

Сортировка строк текста по различным критериям. Наиболее полезные опции: `-n` (**N**umeric), по числовому значению, и `-r` (**R**everse), которая переворачивает вывод. Это может быть полезно для сортировки вывода `du`. Например, если хотите отсортировать файлы по размеру, просто соедините эти команды.

## 23. WC

Утилита командной строки Linux для подсчёта количества слов, строк, байт и символов.

## 24. DIFF

Показывает различия между двумя файлами в построчном сравнении. Причём выводятся только строки, в которых обнаружены отличия. Измененные строки отмечаются символом «с», удаленные — «d», а новые — «a».

Кстати, я подготовил ещё одну подробную статью, в которой описан именно [просмотр содержимого текстового файла в Linux](#) с помощью терминала.

# КОМАНДЫ LINUX ДЛЯ УПРАВЛЕНИЯ ПРОЦЕССАМИ

## 25. KILL / XKILL / PKILL / KILLALL

Служат для завершения процессов. Но они принимают различные параметры для идентификации процессов. `Kill` нужен PID процесса, `xkill` — достаточно кликнуть по окну, чтобы закрыть его, `killall` и `rkill` принимают имя процесса. Используйте ту, которая удобна в определенной ситуации.

## 26. PS / PGREP

Как уже говорилось, чтобы уничтожить процесс, нужен его идентификатор. Один из способов получить его, это утилита `ps`, которая печатает информацию о запущенных процессах. По умолчанию вывод очень длинный, поэтому используйте опцию `-e`, чтобы увидеть информацию об определённом процессе. Это только снимок состояния на момент вызова, и информация не будет обновляться. Команда `ps` с ключом `aux` выводит полную информацию о процессах. `Pgrep` работает

следующим образом: вы задаете имя процесса, а утилита показывает его идентификатор.

## 27. TOP / HTOP

Обе команды похожи, обе отображают процессы и могут быть использованы как консольные системные мониторы. Я рекомендую установить htop, если в вашем дистрибутиве он не поставляется по умолчанию, так как это улучшенная версия top. Вы сможете не только просматривать, но и контролировать процессы через его интерактивный интерфейс.

## 28. TIME

Время выполнения процесса. Это секундомер для выполнения программы. Полезно, если вам интересно, насколько сильно ваша реализация алгоритма отстает от стандартной. Но, несмотря на такое название, она не сообщит вам текущее время, используйте для этого команду date.

# КОМАНДЫ LINUX ОКРУЖЕНИЯ ПОЛЬЗОВАТЕЛЯ

## 29. SU / SUDO

Su и sudo — это два способа выполнить одну и ту же задачу: запустить программу от имени другого пользователя. В зависимости от вашего дистрибутива вы, наверное, используете одну или другую. Но работают обе. Разница в том, что su переключает вас на другого пользователя, а sudo только выполняет команду от его имени. Поэтому использование sudo будет наиболее безопасным вариантом работы.

## 30. DATE

В отличие от time, делает именно то, чего вы от неё и ожидаете: выводит дату и время в стандартный вывод. Его можно форматировать в зависимости от ваших потребностей: вывести год, месяц, день, установить 12-ти или 24-ти часовой формат, получить наносекунды или номер недели. Например, date +%j %V», выведет день в году и номер недели в формате ISO.

## 31. ALIAS

Команда создаёт синонимы для других команд Linux. То есть вы можете делать новые команды или группы команд, а

также переименовывать существующие. Это очень удобно для сокращения длинных команд, которые вы часто используете, или создания более понятных имен для команд, которые вы используете нечасто и не можете запомнить.

### 32. UNAME

Выводит некую основную информацию о системе. Без параметров она не покажет ничего полезного, кроме строки Linux, но, если задать параметр -a (**All**), можно получить информацию о ядре, имени хоста и узнать архитектуру процессора.

### 33. UPTIME

Сообщает вам время работы системы. Не очень существенная информация, но может быть полезна для случайных вычислений или просто ради интереса, чтобы узнать, как давно был перезагружен сервер.

### 34. SLEEP

Вам, наверное, интересно как же её можно использовать. Даже не учитывая Bash-скриптинг, у неё есть свои преимущества. Например, если вы хотите выключить компьютер через определенный промежуток времени или использовать в качестве импровизированной тревоги.

## КОМАНДЫ LINUX ДЛЯ УПРАВЛЕНИЯ ПОЛЬЗОВАТЕЛЯМИ

### 35. USERADD / USERDEL / USERMOD

Эти команды консоли Linux позволяют вам добавлять, удалять и изменять учетные записи пользователей. Скорее всего, вы не будете использовать их очень часто. Особенно если это домашний компьютер, и вы являетесь единственным пользователем. Управлять пользователями можно и с помощью графического интерфейса, но лучше знать об этих командах на всякий случай.

### 36. PASSWD

Эта команда позволяет изменить пароль учетной записи пользователя. Как суперпользователь вы можете сбросить пароли всех пользователей, даже несмотря на то, что не можете их увидеть. Хорошая практика безопасности — менять пароль почаще.



# LINUX КОМАНДЫ ДЛЯ ПРОСМОТРА ДОКУМЕНТАЦИИ

## 37. MAN / WHATIS

Команда man открывает руководство по определённой команде. Для всех основных команд Linux есть man страницы. Whatis показывает, какие разделы руководств есть для данной команды.

## 38. WHEREIS

Показывает полный путь к исполняемому файлу программы. Также может показать путь к исходникам, если они есть в системе.

# КОМАНДЫ LINUX ДЛЯ УПРАВЛЕНИЯ СЕТЬЮ

## 39. IP

Если список команд Linux для управления сетью вам кажется слишком коротким, скорее всего вы не знакомы с утилитой ip. В пакете net-tools содержится множество других утилит: ipconfig, netstat и прочие устаревшие, вроде iproute2. Всё это заменяет одна утилита — ip. Вы можете рассматривать её как швейцарский армейский нож для работы с сетью или как непонятную массу, но в любом случае за ней будущее. Просто смириться с этим.

## 40. PING

Ping — это ICMP ECHO\_REQUEST дейтаграммы, но на самом деле это неважно. Важно то, что утилита ping может быть очень полезным диагностическим инструментом. Она поможет быстро проверить, подключены ли вы к маршрутизатору или к интернету, и даёт кое-какое представление о качестве этой связи.

## 41. NETHOGS

Если у вас медленный интернет, то вам, наверное, было бы интересно знать, сколько трафика использует какая-либо программа в Linux или какая программа потребляет всю скорость. Теперь это можно сделать с помощью утилиты nethogs. Для того чтобы задать сетевой интерфейс используйте опцию -i.

## 42. TRACEROUTE

Это усовершенствованная версия ping. Мы можем увидеть не только полный маршрут сетевых пакетов, но и доступность узла, а также время доставки этих пакетов на каждый из узлов.

## ВЫВОДЫ

Мы рассмотрели основные команды Linux, которые могут пригодиться вам при повседневном использовании системы. Если вы считаете, что есть другие команды, которые нужно добавить в этот список, напишите в комментариях!