



# Sergios Tseprounof

Me, My hobby, My projects

## Команды Linux. Краткое описание.

by **Stseprounof** on 24/02/2014 in **Linux, Управление сервером**



Описываются основные команды Linux, необходимые для управления сервером. [The same in English](#)

В статье «[НЕСКОЛЬКО СЛОВ О LINUX](#)» я описал идею прав доступа к объектам файловой системы Linux и управление сервисами. Теперь описание основных команд Linux будет немного понятнее.

Для управления нашим сервером, создание и управление сайтами через [безопасное соединение SSH](#) необходимо знание нескольких основных команд Linux:

### 1. *ls* *имя*

выводит на экран содержимое папки *имя* или описание файла *имя*. Если *имя* опущено, то выводится содержимое текущей папки. Без ключей просто выводится список имен файлов, причем скрытые файлы (которые начинаются с точки, например, `.my.cnf`) не выводятся совсем. С ключем `-la` (`ls -la имя`) выводится таблица, как на рис. 1.

```
[root@www admin]# ls -la
total 28
drwx----- 3 admin admin 4096 Feb  2 18:50 .
drwxr-xr-x. 4 root  root  4096 Jan 30 21:26 ..
-rw----- 1 admin admin  170 Feb 14 18:24 .bash_history
-rw-r--r-- 1 admin admin   18 Jul 18  2013 .bash_logout
-rw-r--r-- 1 admin admin  176 Jul 18  2013 .bash_profile
-rw-r--r-- 1 admin admin  124 Jul 18  2013 .bashrc
drwx----- 2 admin admin 4096 Feb 10 23:34 .ssh
[root@www admin]# ls
[root@www admin]#
```

Рис. 1. Команда ls

Для каждого файла, кроме имени выводится его дата создания, размер, владелец, группа и права доступа (справа налево). Скрытые файлы и папки выводятся в списке наравне с обычными.

### 2. *cd* *имяпапки*

перейти в папку *имяпапки*.

- `cd ~` переходит в домашнюю папку текущего пользователя.
- `cd` – возвращается в папку, с которой был осуществлен переход в текущую.

### 3. `mkdir имяпапки`

создает папку с именем *имяпапки*, принадлежащую текущему пользователю с группой, в которую входит текущий пользователь с правами доступа по умолчанию (обычно 755).

Могут быть полезны два ключа:

- `mkdir -m доступ имяпапки` – создает папку *имяпапки* с кодом доступа *доступ* в виде XXX, где X – число из трех бит (описано в статье «[НЕСКОЛЬКО СЛОВ О LINUX](#)»). Например, `-m 700` – папка доступна на чтение, запись и выполнение только для владельца (и, конечно, для суперпользователя).
- `mkdir -p имяпапки` – создает родительские папки, если они отсутствуют.

### 4. `chown владелец.группа имя`

изменяет владельца и группу файла/папки *имя* на *владелец.группа*. Ключ `-R` применяет команду рекурсивно ко всем входящим в указанную папку файлам и дочерним папкам.

### 5. `chmod доступ имя`

изменяет права доступа файла/папки *имя* в соответствии с кодом доступ. Значения *доступ* такие же, как и в команде `mkdir` с ключем `-m` (см. 3. `mkdir`). Ключ `-R` применяет команду рекурсивно ко всем входящим в указанную папку файлам и дочерним папкам.

### 6. `cp имя имя1`

копирует файл/папку *имя* в *имя1*. Если не стоит цель переименовать копируемый объект, то *имя1* должно быть именем папки, куда помещается копируемый объект. Если объект копируется в текущую папку, то вместо *имя1* ставится точка «.» – символ текущей папки. Могут быть полезны ключи:

- Ключ `-R` применяет команду рекурсивно ко всем входящим в копируемую папку *имя* файлам и дочерним папкам.
- Ключ `-p` (строчное) сохраняет владельца и группу копируемых объектов, в противном случае владельцем назначается текущий пользователь и группа, в которую он входит.
- В случае задания ключа `-i` программа копирования спрашивает разрешения пользователя заменить существующий файл копируемым в случае совпадения имен.

### 7. `mv имя имя1`

перемещает файл/папку *имя* в *имя1*. После завершения операции объект *имя* больше не существует. Самый простой способ переименовать файл/папку в Linux.

В случае задания ключа `-i` программа перемещения спрашивает разрешения пользователя заменить существующий файл перемещаемым.

### 8. `rm имя`

удаляет файл *имя*. Необходимо использовать с осторожностью. **ОПЕРАЦИЯ АБСОЛЮТНО НЕОБРАТИМА!**

Могут быть полезны ключи:

- применение ключа *-r* позволяет удалять папки более сложные иерархические объекты файловой системы рекурсивно.
- ключ *-f* подавляет запрос на подтверждение пользователю удаления объекта. Полезен при удалении сложных иерархических объектов файловой системы.
- В случае задания ключа *-i* программа удаления спрашивает разрешения пользователя удалить каждый объект, указанный в *имя*. Рекомендуется практически всегда использовать команду удаления в виде

```
rm -i имя
```

## 9. *useradd* имяпользователя

добавляет нового пользователя Linux с именем *имяпользователя*, создает (RedHat/CentOS Linux) группу *имяпользователя* и добавляет в нее только что созданного пользователя, создает домашнюю папку */home/имяпользователя* и переносит в нее пользовательские конфигурационные файлы по умолчанию. Команда доступна только суперпользователю.

**Внимание!!** Необходимо сразу после добавления нового пользователя использовать команду *passwd* для задания пароля новому пользователю.

## 10. *passwd* имяпользователя

Задаёт/изменяет пароль пользователя *имяпользователя*. Программа дважды спросит пароль, и, если оба ответа совпадут, запомнит введенное значение как пароль пользователя «*имяпользователя*».

Суперпользователь может менять пароль для любого пользователя. Все остальные – только свой.

Понятно, что с простыми паролями жизнь с одной стороны легче – хлопот меньше, но с другой стороны – можно получить другие хлопоты, если Ваш сервер начнет работать на каких-то других людей. Поэтому, я очень рекомендую придумывать хорошие пароли. Что такое хороший пароль я описал в статье [«ЧТО ТАКОЕ SSH/SFTP И ДЛЯ ЧЕГО ОН НУЖЕН?»](#).

## 11. *su* имяпользователя

текущий пользователь заменяется пользователем *имяпользователя*. Если *имяпользователя* опустить, то становится суперпользователем. Надо ввести пароль пользователя *имяпользователя*. Работая под сеперпользователем, пожалуйста будьте **ВНИМАТЕЛЬНЫ!** Можно нанести непоправимый вред системе.

## 12. *exit*

завершает сеанс пользователя. Если была использована команда `su`, то завершается сеанс суперпользователя и возврат к первоначальному пользователю.

В публикации кратко описаны основные команды Linux, необходимые для управления сервером, построенном на основе нашего шаблона виртуальной машины.



### About stseprounof

[View all posts by stseprounof](#) →

## Subscribe

Subscribe to our e-mail newsletter to receive updates.



### Related Posts:

- [Как добавить виртуальный сетевой адаптер к виртуальной машине vSphere](#)
- [Bitvise SSH Client \(Tunnelier\) for Windows. How to install it and setup](#)
- [PSFTP is SFTP client PuTTY for Windows](#)
- [SSH client PuTTY for Windows. Installation and setup manual](#)
- [How to create certificate service. Step by step manual](#)

## Linux

[← Несколько слов о Linux](#)

[Как восстановить сайт из резервной копии. Пошаговая инструкция.](#) >

**No comments yet.**

Leave a Reply