

[Главная](#) / [Курсы](#) / [Linux: горячие клавиши и к...](#)

Что делает команда chmod и как ее использовать в Linux

В статье рассмотрим команду chmod, ее синтаксис и разберемся, как работает система прав в ОС семейства Linux.

Эта инструкция – часть курса «Linux: горячие клавиши и команды».

[Смотреть весь курс](#)

Мы написали эту инструкцию на базе инфраструктуры Selectel. Чтобы повторить ее по шагам, зарегистрируйтесь или авторизуйтесь [в панели управления](#).

Введение

В семействе UNIX-подобных систем безопасность и контроль доступа к файлам и каталогам – ключевые аспекты. Один из основных инструментов для управления правами доступа в таких системах – команда chmod. Она позволяет определить, кто и какие действия выполняет с вашими файлами.

Синтаксис команды chmod и дополнительные параметры

Основной синтаксис команды chmod выглядит следующим образом:

```
chmod [опции] <права> <файлы>
```

- Вместо **опций** – дополнительные параметры команды, которые могут управлять ее поведением.
- Вместо слова **права** – новые права доступа, которые нужно установить для указанных файлов или каталогов.
- Вместо слова **файлы** – список файлов или каталогов, для которых нужно изменить права доступа.

Кроме основного синтаксиса команда chmod также поддерживает различные опции, которые используются для изменения поведения команды. Некоторые из наиболее часто используемых опций включают:

- **-R** или **-recursive**: рекурсивно изменяет права доступа для всех файлов и подкаталогов внутри указанного каталога.
- **-f** или **-silent**, **-quiet**: подавляет вывод сообщений об ошибках или предупреждениях.
- **-v** или **-verbose**: выводит подробный отчет о каждом изменении прав доступа.
- **-c** или **-change**: ключ действует аналогично ключу **-verbose**. Он выводит информацию только если с файлом или директорией были произведены какие-либо действия.

Например, чтобы установить права чтения, записи и выполнения для владельца, только чтения для группы и никаких прав для остальных пользователей для файла example.txt, мы можем использовать следующую команду:

```
chmod 750 example.txt
```

Права в Linux

При работе с командой `chmod` важно понимать основные права доступа, которые назначают файлам или каталогам. В Linux используется три основных типа прав доступа:

- **Чтение (Read)** – обозначается буквой «r». Предоставляет возможность просматривать содержимое файла или каталога.
- **Запись (Write)** – обозначается буквой «w». Позволяет создавать, изменять и удалять файлы внутри каталога, а также изменять содержимое файла.
- **Выполнение (Execute)** – обозначается буквой «x». Дает разрешение на выполнение файла или на вход в каталог.

Каждый из указанных выше типов прав доступа может быть назначен трем группам пользователей:

- Владелец (Owner) – пользователь, который является владельцем файла или каталога.
- Группа (Group) – группа пользователей, к которой принадлежит файл или каталог.
- Остальные пользователи (Others) – все остальные пользователи системы.

Комбинация этих базовых прав доступа для каждой из групп пользователей определяет полный набор прав доступа для файла или каталога.

Например, права доступа `rw-r--r--` для файла означают, что владелец имеет все права доступа (чтение, запись и выполнение), группа имеет право только на чтение и выполнение, а остальные пользователи не имеют никаких прав доступа к этому файлу.

При использовании команды `chmod` мы можем изменять эти базовые права доступа для каждой из групп пользователей в соответствии с требованиями безопасности и политики доступа.

Просмотр прав на файлы и директории

Давайте рассмотрим, как можно просматривать текущие разрешения файлов и каталогов. Для этого мы будем использовать команду `ls` в сочетании с опцией `-l`, которая выводит длинный формат списка файлов и каталогов, включая информацию о разрешениях.

Вот пример использования команды `ls -l`:

```
-rw-r--r-- 1 user1 user1 12345 Oct 10 10:00 myfile.txt
drwxr-xr-x 2 user1 user1 4096 Oct 10 09:00 mydir
```

Вывод команды состоит из нескольких столбцов, нас интересует первый столбец, который содержит информацию о разрешениях. В данном примере:

- `myfile.txt` имеет права `-rw-r--r--`, что означает, что владелец файла (`user1`) имеет права на чтение и запись, а остальные пользователи имеют право только на чтение.
- `mydir` – это директория, ее разрешения `drwxr-xr-x` говорят нам, что владелец каталога может читать, записывать и выполнять его, а остальные пользователи могут только читать и выполнять.

Также просмотреть права на файл или директорию вы можете с помощью команды `getfacl` (get file access control lists). Данная команда выведет полный список прав для директории или файла. Пример вывода данной команды предоставлен ниже:

```
# file: file.txt
# owner: user1
# group: group1
user::rw-
group::r--
other::r--
```



Изменение прав доступа

Изменить права доступа к файлам и папкам можно с использованием команды `chmod`. Сделать это можно с помощью буквенной и численной нотаций, для директорий и файлов, а также рекурсивно, изменяя права для всех вложенных объектов в указанной директории.

Символьная нотация

Символьная нотация в команде `chmod` позволяет вам изменять права доступа к файлам и каталогам с использованием более интуитивного и читаемого подхода. Она позволяет указать, какие разрешения должны быть установлены или изменены для владельца файла, группы и остальных пользователей.

Синтаксис у символьной нотации следующий:

```
chmod [ugoa] [+-=] [rwx] файл(ы)
```

Где:

- [ugoa] указывает, к кому применяются изменения:
 - u: владелец файла.
 - g: группа файла.
 - o: остальные пользователи (не владелец и не входящие в группу).
 - a: все (при использовании заменяет собой ugo).
- [+-=] определяет, что нужно сделать с разрешениями:
 - +: добавить разрешение.
 - -: удалить разрешение.
 - =: установить разрешение точно (заменить текущие разрешения).
- [rwx] представляет собой комбинацию символов 'r', 'w' и 'x', обозначающих разрешения:
 - r: разрешение на чтение.
 - w: разрешение на запись.
 - x: разрешение на выполнение.

Примеры использования символьной нотации:

- `chmod u+x файл` – добавить право выполнения для владельца файла.
- `chmod go-gw файл` – удалить право чтения и записи для группы и остальных пользователей.
- `chmod a=rwx файл` – установить полные права (чтение, запись и выполнение) для всех.

Числовая нотация

Другим способом изменения прав доступа к файлам и каталогам в команде `chmod` является использование числовой нотации. В этом случае вы используете числа для представления разрешений и их комбинаций. Этот метод может показаться менее интуитивным по сравнению с символьной нотацией, но он также является мощным способом управления правами доступа.

Числовая нотация состоит из трех цифр, где каждая цифра представляет разрешения для владельца, группы и остальных пользователей соответственно. Каждая цифра состоит из суммы разных значений, присвоенных разным разрешениям:

- 4 – разрешение на чтение (read).
- 2 – разрешение на запись (write).
- 1 – разрешение на выполнение (execute).
- 0 – отсутствие разрешения.

Для установки конкретных разрешений, сложите соответствующие значения:

- Если вы хотите только разрешение на чтение, используйте 4.
- Если вы хотите разрешение на чтение и запись, сложите 4 и 2, что даст 6.
- Если вы хотите разрешение на чтение и выполнение, сложите 4 и 1, что даст 5.



```
chmod 644 myfile
```

Данная команда установит права `-rw-r--r--`, где владелец имеет права на чтение и запись, а группа и остальные пользователи могут только читать файл.

```
chmod 755 script.sh
```

Эта команда поможет установить права `-rwxr-xr-x`, где владелец имеет полные права, а остальные пользователи могут только читать и выполнять скрипт.

Изменения прав для каталога

Как и для обычных файлов, вы можете изменять права доступа для каталогов с помощью команды `chmod`. Права доступа к каталогам контролируют, какие операции можно выполнять внутри каталога: просмотр его содержимого, создание, удаление файлов и каталогов внутри него и так далее.

Для изменения прав доступа к каталогу используется та же числовая и символьная нотация, которую мы рассматривали ранее. Укажите имя каталога, к которому вы хотите применить изменения прав доступа вместо имени файла. Давайте рассмотрим примеры в двух нотациях.

Символьная нотация

- `chmod u+rwx директория` – добавить владельцу каталога права на чтение, запись и выполнение.
- `chmod go-g директория` – удалить права на чтение для группы и остальных пользователей для каталога.
- `chmod a+rx директория` – предоставить всем пользователям права на чтение и выполнение для каталога.

Числовая нотация

`chmod 700 директория` – установить права `-rwx---` для владельца каталога, позволяя ему читать, записывать и выполнять операции внутри него, но никаким другим пользователям.

`chmod 755 директория` – предоставить владельцу полные права `-rwxr-xr-x`, а остальным пользователям право только на чтение и выполнение. `chmod 644 директория` – предоставить права на чтение и запись для владельца каталога, а остальным пользователям и группам оставить только на чтение содержимого.

Рекурсивное применение команды `chmod`

Чтобы выполнить рекурсивное изменение прав доступа, используйте флаг `-R` или `-recursive` с командой `chmod`. Например:

- `chmod -R 755 каталог` – установит права `-rwxr-xr-x` для всех файлов и каталогов внутри указанного каталога и самого каталога.
- `chmod -R u+rwx,go-gwx каталог` – добавит владельцу права на чтение, запись и выполнение, а группе и остальным пользователям уберет все права для всех файлов и каталогов в указанной директории и её подкаталогах.

Рекурсивное изменение прав доступа – отличный инструмент для обновления разрешений для файлов и каталогов одновременно.

Ключ `--reference` и его использование

Ключ `--reference` в команде `chmod` предоставляет удобный способ копирования прав доступа с одного файла или каталога на другой. Этот ключ позволяет быстро и просто применить права доступа из одного

```
chmod --reference=RFILE file
```

- **--reference=RFILE**. Этот аргумент указывает путь к файлу или каталогу, из которого будут скопированы права доступа.
 - **file**: это целевой файл или каталог, для которого вы хотите установить права доступа, согласно указанному источнику.
- Важно отметить, что **--reference** не просто добавляет или удаляет определенные разрешения, а полностью перезаписывает права доступа целевого элемента.

Заключение

Мы рассмотрели синтаксис команды `chmod`, ее параметры и ключи, а также основные методы изменения прав доступа. Мы изучили символьную нотацию и числа, которые позволяют гибко настраивать необходимые вам права.

Linux

12 минут

Команда `ifconfig` в Linux: установка, синтаксис и примеры...

Предыдущий материал

- 1 Введение
- 2 Синтаксис команды `chmod` и дополнительные параметры
- 3 Права в Linux
- 4 Изменение прав доступа
- 5 **Заключение**

Сейчас вы на 9 статье **курса**

01 · 10 минут
Командная строка Linux: краткий курс для начинающих

Полный списоккурса (9)

Зарегистрируйтесь в панели управления

И уже через пару минут сможете арендовать сервер, развернуть базы данных или обеспечить быструю доставку контента.

Читайте также:

Инструкция

Как установить PostgreSQL 15 на Windows 10 и 11

12 августа 2024

Инструкция

Как создать веб-приложение на базе VK Mini Apps

8 августа 2024

Инструкция

Как развернуть приложение с помощью Helm в Kubernetes

29 июля 2024



Блог

Курсы

Мероприятия

В галерею

Блог

Курсы

Мероприятия

selectel.ru

- Инвесторам
- Прессе
- Юристам
- 0 компаний
- Документация
- Карьера

Email



Соглашаюсь на получение новостной рассылки
в соответствии с Политикой

