

20. Управление на права за достъп до файлове и директории за различни потребители

Собственик и права за достъп

В Linux всички файлове принадлежат на някого - съществува понятието *собственик* на файла. **Собственикът**, това е потребителят, на когото принадлежи файлът. В повечето случаи това е потребителят, създал файла. С помощта на команда **chown** обаче, която ще бъде разгледана по-късно, **собственикът може да бъде променен**.

Правата за достъп определят кой и какво може да прави с файла. Правата за достъп до файла могат да бъдат променяни от *собственика на файла* или от потребителя **root**. Собственикът може да определи, например, кой има право да чете и променя файла. Собственикът също така може да „подари“ файла на друг потребител. След това негов собственик ще стане другият потребител.

Правата за достъп на потребителя **root** (суперпотребителя) са максимални, а това означава, че той може да променя собственика на всеки файл (вие може да създадете файл, а **root** може да направи негов собственик всеки друг потребител), както и да променя правата за достъп на всеки файл.

Суперпотребителят може да изтрие и промени всеки файл, също така може да създаде файл във всяка папка и т.н. От една страна, това е хубаво, но ако акаунтът на **root** попадне в чужди ръце, последствията могат да бъдат непоправими.

Именно поради това акаунтът на **root** в Ubuntu е **деактивиран по подразбиране**, което не може да се каже за останалите дистрибуции.

Правата за достъп в Linux по подразбиране са настроени така, че един потребител да притежава само своята домашна папка **/home/<потребителско_име>**. Затова да създава файлове и да изпълнява други операции с файлове (изтриване, редактиране, копиране и т.н.) потребителят може само в тази папка и то при условие че файловете му принадлежат.

Ако в домашната папка на потребителя **root** е създал файл, обикновеният потребител не може да го изтрие или промени, тъй като не се явява негов собственик. А дали ще може да прочете този файл, зависи от правата за достъп до файла.

Останалите файлове, които се намират извън пределите на домашната папка, потребителят може само да чете и то ако това е разрешено от правата за достъп. Например файлът **/etc/passwd** може да бъде четен от потребителите, но файлът **/etc/shadow** - не.

Обикновеният потребител не може да създава файлове в кореновата файлова система или във всяка друга папка, която не му принадлежи, ако не са му предоставени права за достъп до тази папка.

Команда chown

Команда chown се използва за промяна на собственика на файл/папка. Форматът ѝ е следният:

chown <потребител> <файл/папка>

Тук потребител е новият собственик на файла. За да може да подарите файл на друг потребител, вие трябва да бъдете негов собственик или потребител root.

Команда chmod

Команда chmod се използва за промяна на правата за достъп до файл или папка. За промяна на правата за достъп, както и в случая с командата chown, вие трябва да бъдете собственик на файла/папката или root потребител.

Форматът на командата е следният:

chmod <права> <файл/папка>

Правата за достъп се състоят от три групи:

- За собственика;
- За групата на собственика;
- За останалите потребители.

Първата група задава възможностите за собственика на файла, втората - за групата, в която влиза собственикът, а третата - за останалите потребители.

Във всяка група може да има три вида права - четене (r), записване (w) и изпълнение (x). За файл правото на изпълнение означава възможността за стартиране на файла (обикновено се използва за програми и сценарии). Правото на изпълнение за папка означава възможност за отваряне и преглеждане на папката.

Правата за достъп във всяка група се определят ясно в строго определен порядък и могат да бъдат представени както в символен, така и в числов вид (в двоична или осмична система). Да разгледаме няколко комбинации от права за достъп:

100 - само за четене

110 - четене и писане

101 - четене и изпълнение

111 - четене, писане, изпълнение.

Отчитайки, че права за достъп се задават за собственика, групата и останалите потребители, пълният набор от права за достъп може да изглежда така:

110 100 000

В този случай ние предоставяме пълни права за достъп (освен за изпълнение) на собственика. На групата на собственика е разрешено само четене, а за останалите потребители достъпът към файла/папката е напълно забранен.

Право за изпълнение има смисъл да бъде задавано само за програми и сценарии за командната обвивка. За останалите файлове няма смисъл от това.

Малцина са онези, които описват правата за достъп в двоична система. Основно се преобразуват в осмична. Онези, които не са запознати с нея, могат да погледнат таблицата.

Преобразуване от двоична в осмична система

Двоична система	Осмична система
000	0
001	1
010	2
011	3
100	4
101	5
110	6
111	7

И така, 666 означава пълен достъп до обикновен файл и за трите групи потребители, 660 - пълен достъп само за собственика и групата, а на останалите потребители е забранено дори четенето на този файл. Обикновено за файлове се задават права за достъп 664, т.е. на собственика и групата е разрешен пълен достъп, а всички останали могат само да четат файла.

За папка пълните права за достъп изглеждат така - 777 - четене, промяна и разглеждане на папката за собственика, групата и останалите потребители.

Може да прегледате текущите права за достъп с командата **ls -l <файл/папка>**, например:

```
ls -l file.txt
```

```
-rw-r--r-- 1 root root 2179 Mar 16 08:28 file.txt
```

Както виждаме, зададени са три групи **rw-**, **r--**, **r--**. Излиза, че на собственика е разрешено четене и писане във файла, а за останалите потребители (групата и всички останали) — само четене. В осмичната система тази комбинация права за достъп ще изглежда така - 644.

Първият символ (в нашия случай това е „-“) е признак за файл. Ако извеждаме правата за достъп на папка, тогава вместо символа „-“ би имало символ **d**. За файл се извежда просто „-“

Символният начин за задаване на права за достъп е малко по-прост. Да разгледаме пример с неговото използване:

```
chmod +x program
```

Да разгледаме отново правата за достъп:

```
ls -l program
```

```
-rwxr-xr-x 1 root root 1176 Mar 19 18:28 program
```

Както виждате, право за изпълнение беше добавено във всичките три групи права за достъп.

Правата за достъп могат да бъдат променени и с помощта на графичен интерфейс. Щракнете с десния бутон на мишката върху файла или папката и изберете командата **Properties**. В отворилия се прозорец могат да бъдат определени права за достъп за собственика (**Owner**), групата (**Group**), а също така и за останалите потребители (**Others**). Отметка на **Allow executing file as program** ви позволява да направите файла изпълним. С графичен интерфейс обаче няма да може да промените собственика на файла.