
KAPITOLA 3

Práva a co s nimi

Práva přístupu k souborům, adresářům

Každý soubor a adresář někomu patří, má svého vlastníka. Vlastník patří do určité skupiny uživatelů. Kdo není vlastníkem souboru, ani nepatří do skupiny uživatelů, patří mezi ostatní.

Takto by se velice stručně dala popsat pravidla oprávnění přístupu k souborům a adresářům. Vypište si obsah jakéhokoliv adresáře pomocí:

```
$ ls -al
drwxr-xr-x 3 pavel users 4096 1. kvě 21.55 kniha
-rwxr--r-- 1 pavel users 25 25. dub 21.19 kralik
drwxr-xr-x 3 pavel users 4096 27. dub 10.11 .local
lrwxrwxrwx 1 pavel users 9 27. dub 12.11 mtab -> /etc/mtab
```

Zajímavé jsou v současné chvíli jen začátky řádků. Jako první znak zde může být buď pomlčka, potom se jedná o soubor. Nebo jako první znak bude d, to se jedná o adresář. Ve výpisu vidíte také jako první znak písmeno l, to se jedná o symbolický link – odkaz na soubor, něco jako ve Windows zástupce.

Další znaky se počítají vždy po třech. První trojice, tedy znak na druhém až čtvrtém místě, vám zobrazuje oprávnění vlastníka. Další tři znaky jsou pro oprávnění skupiny a poslední tři znaky, tedy znak osm až deset, je pro oprávnění ostatních:

- ◆ Písmeno `r` – znamená oprávnění pro čtení,
- ◆ písmeno `w` – znamená oprávnění zapisovat,
- ◆ písmeno `x` – znamená oprávnění soubor spouštět, adresář procházet.

Za těmito znaky následuje číslo, to značí, kolik je odkazů na tento adresář nebo soubor. Dále vidíte jméno vlastníka a do které patří skupiny.

Jak můžu změnit vlastníka?

Ve většině případů může změnu vlastnictví provést pouze root. Provádí se pomocí příkazu:

```
#chown kmp kralik
#ls -al kralik
-rwxr--r-- 1 kmp users 25 25. dub 21.19 kralik
```

Příkaz zadáváte ve tvaru:

```
chown novy_vlastnik jmeno_souboru.
```

Můžu měnit oprávnění přístupu k souboru

Toto oprávnění může měnit i běžný uživatel. Pokud budete psát skripty, dokonce budete oprávnění měnit často. Změna se ale týká pouze souborů a adresářů, u kterých jste vlastníkem anebo máte oprávnění pro změnu. Root může měnit cokoliv.

Příkaz `chmod` poprvé

U starších systémů se oprávnění měnilo pomocí čísel. Dnes jsou způsoby provedení dva, ale někde druhý systém nemusí fungovat, a proto je lépe se seznámit s oběma.

Princip spočívá v tom, že máte tři skupiny, pro které nastavujete oprávnění. Nastavovat budete pomocí číslic, tedy tři skupiny, tři číslice. Podle výčtu uvedeného dále vyberete oprávnění:

- ◆ 0 – žádné oprávnění
- ◆ 1 – jen právo spouštět
- ◆ 2 – jen právo zápisu
- ◆ 3 – právo zápisu a může spouštět
- ◆ 4 – jen právo čtení
- ◆ 5 – právo čtení a možnost spouštět
- ◆ 6 – právo čtení a zápisu
- ◆ 7 – plné oprávnění, čtení zápis a spouštění

Změnu provedete například takto:

```
chmod soubor 777
```

V tomto případě dáváte všem uživatelům právo na čtení, zápis i možnost spouštět tento soubor. V případě, že zadáte oprávnění 700, dáváte vlastníkově úplné oprávnění a skupina a zbytek světa nemůže dělat se souborem nic. A tak podobně jste schopni nakombinovat variantu, jakou potřebujete.

Příkaz chmod podruhé

Druhou, novější variantou je provádění změny pomocí písmen `u`, `g`, `o`, `a`. Zde písmeno `u` značí vlastníka, písmeno `g` skupinu a písmeno `o` je pro ostatní. Písmeno `a` je pro všechny naráz. Dále použijete znaménko plus a nebo minus; pomocí plus práva přidáváte a pomocí minus práva odebíráte. Přidávání a odebírání je nezávislé na tom, jaká práva jsou nyní nastavena. Podívejte se na příklad:

```
$ ls -al vstup
-rwx----- 1 pavel users 59 30. dub 10.02 vstup
```

```
$ chmod a+rwx vstup
```

```
$ls -al vstup
-rwxrwxrwx 1 pavel users 59 30. dub 10.02 vstup
```

Zde byla provedena změna práv z původního maximálního oprávnění pro uživatele pavel na maximální oprávnění pro všechny.

Další ukázkou bude odebírání oprávnění pro skupinu:

```
$ chmod g-xw vstup
```

```
$ls -al vstup
-rwxr--rwx 1 pavel users 59 30. dub 10.02 vstup
```

Podobným způsobem můžete opět provádět všechny možné varianty, důležité je si zapamatovat, že nejprve píšete, pro koho změnu provádíte, následně to, jestli práva přidáváte anebo odebíráte, a na závěr napíšete, jaké oprávnění budete měnit.

Jak se dají nastavit práva

V případě potřeby lze nastavit, jak se budou nastavovat práva při vytvoření nového souboru. Příkazem `umask` nadefinujete, jakým způsobem se budou nastavovat práva u nových souborů. Příkaz funguje opět pomocí čísel, ale tentokrát se odpočítává. Maximální oprávnění je tedy 777, příkazem `umask 222` snížíte výsledné oprávnění na 555.

```
$umask 777
$cat > soubor
$ls -al soubor
----- 1 pavel users 0 2. kvě 20.33 soubor
```

Trochu nesmyslný příkaz ukazuje, že to opravdu tak funguje.

Pokud zadáte příkaz `umask` bez parametrů, dostanete výpis, jak jsou práva nastavena.

Můžu změnit skupinu?

Stejně jako vlastníka souboru nebo adresáře můžete změnit skupinu u souboru, adresáře. K tomuto účelu použijte příkaz:

```
$ chgrp student /home/pavel/zaloha
```

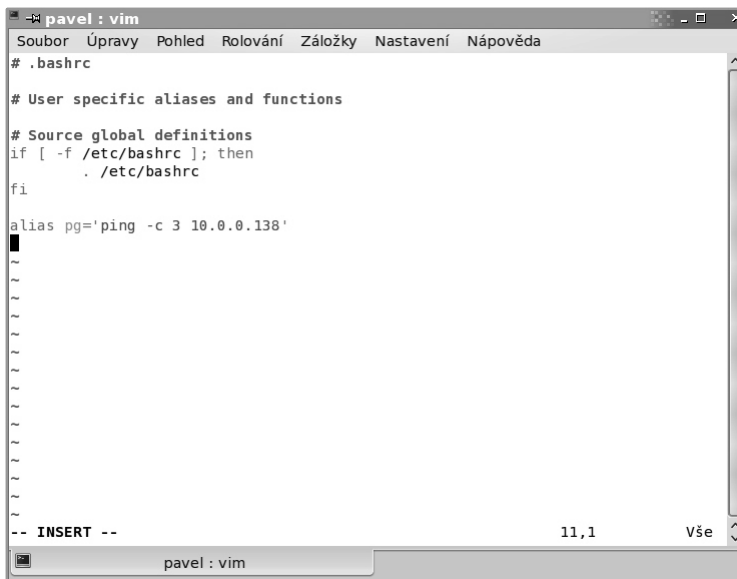
Pro adresář `zaloha` jste změnili skupinu z `users` na `student`.

K čemu je dobrý alias

Alias má velice jednoduchý úkol. Snaží se vám zjednodušit život. Pomocí nějaké zkratky vám umožní příkaz na celou řádku zadávat pomocí zkratky. Jak na to?

V domovském adresáři každého uživatele je soubor nazvaný `.bashrc`. Do tohoto souboru si můžete nadefinovat, kolik chcete aliasů. Důležité je si pamatovat zkratky příkazů. V terminálu zadáváte zkratku a ta vykoná příkaz pod touto zkratkou uložený. Nejprve se přesvědčte, zda jste ve svém domovském adresáři – `pwd`. Otevřete si soubor `.bashrc` v některém z editorů.

```
vi .bashrc
```



Obrázek 3.1: Editujete `.bashrc`

Podle obrázku provedete editaci svého souboru `.bashrc`. To znamená, že doplníte na konec souboru:

```
alias zkratka='celý příkaz'
alias lsa1='ls -al |more'
```

a ukončíte klávesou `Enter`. Následně se musíte odhlásit z terminálu a znovu přihlásit anebo znovu spustit `. .bashrc`. Nyní se po použití vaší nové zkratky `lsa1` provede příkaz

```
ls -al |more
```



Upozornění

Nesmíte zapomenout na apostrofy (můžete použít i uvozovky), jinak vám alias nebude fungovat. Důležité je se odhlásit a znovu přihlásit, protože po přihlášení se kontroluje soubor `.bashrc`. Nový alias se vám objeví i při použití tabulátoru, pokud napíšete první znaky ze začátku aliasu.

Přidám ještě jedno další využití. Hodně často zkontrolujete dostupnost některého serveru a nechce se vám neustále vypisovat `ping...` Přidáte si alias:

```
alias seznam="ping -c 3 www.seznam.cz"
```

a potom stačí napsat jen seznam a příkaz `ping www.seznam.cz` se provede.

Jak vytvořit alias pro všechny nové uživatele

Při vytvoření nového uživatele se několik souborů nahrává i z adresáře `/etc/skel`. Právě v tomto adresáři najdete i výchozí `.bashrc`. Zde můžete doplnit aliasy. Vyzkoušejte tedy například

```
ls -al /etc/skel
```

Všem novým uživatelům se načítá základní nastavení. Vytvoří se domovský adresář a do něj se nakopírují důležité adresáře a soubory. V tomto adresáři vyberete `.bashrc` a provedete stejné operace jako v předchozí části.

Jak se můžu přihlásit jako jiný uživatel

Přihlášení se jako jiný uživatel se provádí pomocí příkazu:

```
su uživatel
```

Pokud zadáte příkaz `su` bez parametru, jste vyzváni na zadání hesla superuživatele – roota. Bere se to tak, že se hodláte přihlásit jako administrátor.

Například:

```
su kmp
```

Tento příkaz použijete, hodláte-li se přihlásit jako uživatel `kmp`. Jako parametr můžete ještě doplnit `-l` a v tomto případě se projde celá přihlašovací sekvence. Budete mít tedy stejné prostředí jako uživatel `kmp`.

Jak se přihlásím jako superuživatel – root

Jak jste se mohli přesvědčit o něco dříve v textu, můžete se přihlásit jako superuživatel pomocí příkazu

```
$ su
Heslo:
su: chybné heslo
```

Jestliže heslo neznáte anebo zadáte chybné heslo, přihlášení neproběhne.

Stejně se můžete přihlásit, pokud zadáte příkaz ve tvaru:

```
$ su root
Heslo:
#
```

Po zadání správného hesla jste přihlášení.

Chci zadat jen příkaz s oprávněním uživatele

Tuto operaci můžete provést, pokud k příkazu su přidáte parametr -c:

```
$ su -c 'ls -al /etc/ |grep tab > vypis.txt'
Heslo:
```

Tímto způsobem provedete příkaz jako root. Pokud se příkaz skládá z více slov, musíte jej uzavřít do apostrofů. Příkaz ale také můžete napsat ve tvaru

```
su kmp -c 'prikaz1 prikaz2 ...'
```

V tomto případě provedete příkaz jako uživatel kmp.

Ověříte si, že se akce opravdu povedla.

```
$ more vypis.txt
-rw-r--r--. 1 root root 541 19. úno 07.51 anacrontab
-rw-r--r--. 1 root root 448 25. čec 2009 crontab
-rw-r--r--. 1 root root 0 10. kvě 10.23 crypttab
-rw-r--r--. 1 root root 1042 12. kvě 13.18 fstab
-rw-r--r--. 1 root root 992 12. kvě 08.42 fstab~
-rw-r--r--. 1 root root 853 10. kvě 12.17 inittab
-rw-r--r--. 1 root root 572 27. kvě 2010 mtab

-rw-----. 1 root pavel 0 10. kvě 11.16 mtab.fuselock
-rw-r--r--. 1 root root 257 13. říj 2008 quotatab
-rw-r--r--. 1 root root 930 28. dub 20.26 rwtab
drwxr-xr-x. 2 root root 4096 28. dub 20.26 rwtab.d
-rw-r--r--. 1 root root 212 28. dub 20.26 statetab
drwxr-xr-x. 2 root root 4096 28. dub 20.26 statetab.d
```

Spouštění příkazu jako superuživatel – příkaz sudo

Tento velice zajímavý nástroj používejte, ale vždy s rozumem, protože jinak můžete zplakat nad výdělkem.

Příkaz sudo vznikl jako zkratka dvou slov: superuser do. To hovoří jasně o tom, proč vznikl. Nadefinování možnosti spojené s tímto příkazem provádíte editací souboru /etc/sudoers. Tento soubor neupravujte pomocí editorů gedit, kedit a podobně. K úpravě využijte nástroj

```
visudo
```

Pomocí tohoto příkazu se vám spustí vi editor s otevřeným souborem /etc/sudoers. Při editaci tohoto souboru je pomocí tohoto příkazu kontrolována i syntaxe, tedy to, jestli zadáváte parametry správně.

Podívejte se na ukázkou toho, jakým způsobem můžete upravovat záznamy. Před každou sekcí vidíte, k čemu oddíl slouží, a jednoduše jsou naznačeny možnosti, jak by měl zápis vypadat.

```
## Allows members of the users group to mount and unmount the
### cdrom as root
# %users ALL=/sbin/mount /mnt/cdrom, /sbin/umount /mnt/cdrom
```

Zde například vidíte povolení připojení skupině users a odpojení Cdrom jako root.

```
## Allows members of the users group to shutdown this system
# %users localhost=/sbin/shutdown -h now
```

A zde je povolení skupině `users` k vypnutí systému. V těchto případech stačí odkomentovat, smazat mřížku `#` na začátku řádku, ale jen u

```
%users localhost=/sbin/shutdown -h now
```

Řádek nebo řádky nad touto definicí oprávnění slouží jako komentář – ten (ty) necháte, jak je (jsou).

Následující ukázku berte jako ilustrativní a opravdu si dobře rozmyslete, jestli tento způsob potřebujete.

Otevřete si pomocí příkazu (`s` právy `roota`):

```
# visudo /etc/sudoers
```

Najděte sekci uvedenou níže a dopište řádek s uživatelem, pro kterého hodláte nastavit práva stejná jako `root`.

```
## Allow root to run any commands anywhere
root    ALL=(ALL)    ALL
pavel   ALL=(ALL)    ALL
```

Uložte soubor a můžete vyzkoušet, jestli vše funguje, například takto

```
less /etc/sudoers
/etc/sudoers: Operace zamítnuta
```

To je správně, protože v normálním případě obyčejný uživatel nesmí ani číst soubor `/etc/sudoers`.

Stejná varianta s upravenými právy, s použitím příkazu `sudo`:

```
$ sudo more /etc/sudoers
### Sudoers allows particular users to run various commands as
### the root user, without needing the root password.
###
### Examples are provided at the bottom of the file for collections
.
```

Uvědomte si, že čím více máte uživatelů s nejvyšším oprávněním, tím více může být ohrožen váš systém. Spíš používejte varianty na povolení jen určité operace pro určitou skupinu nebo jednotlivé uživatele.