



INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

PSK3-3

Název školy:	Vyšší odborná škola a Střední průmyslová škola, Božetěchova 3
Autor:	Ing. Marek Nožka
Anotace:	Přehled základních příkazů systému Unix
Vzdělávací oblast:	Informační a komunikační technologie
Předmět:	Počítačové sítě a komunikační technika (PSK)
Tematická oblast:	Operační systém Linux/Unix
Výsledky vzdělávání:	Žák používá základní příkazy v unixovém shellu
Klíčová slova:	Linux, Unix, shell, mv, cp, rm, ln, ls, mkdir, rmdir
Druh učebního materiálu:	Online vzdělávací materiál
Typ vzdělávání:	Střední vzdělávání, 4. ročník, technické lyceum
Ověřeno:	VOŠ a SPŠE Olomouc; Třída: 4L
Zdroj:	Vlastní poznámky, Vilém Vychodil: Linux Příručka českého uživatele

Základní příkazy

Na této stránce naleznete přehled základních příkazů. Pokud se chcete dozvědět víc podívejte se na seriál Unixové nástroje.

Zápis cesty

V příkazech zapisujeme velice často cesty k souborům. Jako oddělovač adresářů se používá znak lomeno /. Například

```
/home/students/pepa  
MojeData/soubor.txt
```

Cestu lze zapsat relativně, nebo absolutně.

Absolutní cesta

začíná vždy lomítkem a je vždy stejná. Například
`/usr/games/bombardier`

Relativní cesta

nezačíná lomítkem a při jejím zápisu je vynechána část cesty, která odpovídá pracovnímu adresáři.

Pracovní adresář

je místo v adresářovém stromu, na kterém se zrovna shell nachází. Pracovní adresář je klíčový pro zápis relativní cesty. Změna pracovního adresáře se provádí příkazem `cd`. Výpis pracovního adresáře zajistí program `pwd`.

Jako příklad uvedeme výpis adresáře s dokumentací k programu `bombardier`:

```
ls /usr/share/doc/bombardier/
```

Nyní změním pracovní adresář a dokumentaci vypíšeme znovu:

```
cd /usr/share
pwd
  /usr/share
ls doc/bombardier
  copyright  DEDICATION  changelog.gz  README
```

a dále

```
cd doc
pwd
  /usr/share/doc
ls bombardier
  copyright  DEDICATION  changelog.gz  README
```

Všimněte si, že jak u příkazu `ls`, tak u příkazu `cd` je možné cestu zapsat jak relativně tak absolutně.

Pokud příkaz `cd` zavoláme bez parametrů, přepne nás vždy do našeho domovského adresáře.

```
cd
pwd
  /home/students/pepa
```

Příkaz `cd` - přepne do předchozího pracovního adresáře:

```
cd -
pwd
  /usr/share/doc
cd -
pwd
  /home/students/pepa
```

adresář. a . .

V každém (i prázdném) adresáři se nachází dva speciální adresáře . a .. Adresář . odkazuje vždy do nadřazeného adresáře:

```
pwd
  /home/students/pepa
cd ../..
pwd
  /home
```

Adresář . odkazuje vždy aktuální adresář (sám na sebe). Hodí se například při zápisu cesty kopírovat nebo přesunout souboru do aktuálního adresáře:

```
pwd
  /home/students/pepa
cp /etc/fstab .
```

je stejné jako

```
cp /etc/fstab /home/students/pepa
```

Výpis adresáře

Výpis obsahu adresáře provádí program `ls`. Některé často používané volby:

Přepínače programu `ls`

Přepínač

popis

- l dlouhý formát (anglicky long format).
- F přidá za jméno znak podle typu.
Hvězdičku * pro spustitelný soubor, lomítko / pro adresář, l pro symbolický odkaz. Normálním souborům není přidání nic.
- a zobrazí i soubory které začínají tečkou.
Soubory začínající tečkou jsou jinak považovány za skryté a nejsou vypisovány.
- R rekurzivně vypisuje i obsah podadresářů.
Například `ls -R ~` vypíše všechny soubory v domovském adresáři.
- d vypíše informace o symbolickém odkazu nebo adresáři namísto výpisu jeho obsahu nebo cíli odkazu.
- t seřadí výpis podle času poslední úpravy souboru.
- X seřadí výpis podle přípony souboru.
- h velikost se vypíše v lidsky čitelném formátu (např. 1K, 234M, 5G)

Výstup příkazu `ls -l /tmp`

```
drwx----- 2 pepa doma 4096 říj 7 18:54 mc-marek
-rw-r--r-- 1 pepa doma 577813 říj 11 15:03 mozilla.pdf
prw-r--r-- 1 pepa doma 0 říj 20 16:13 mypipe
lrwxrwxrwx 1 pepa roma 26 říj 15 20:50 printn ->
/etc/cups/ppd/tiskarna.ppd
```

Ve výpisu první znak udává druh souboru

- d adresář
- běžný soubor
- l symbolický odkaz
- p pojmenovaná roura
- c znakové zařízení
- b blokové zařízení
- s socket

Dále je 9 znaků zobrazujících přístupová práva. Dále číslo udává počet jmen souboru. Dále jméno vlastníka pepa a skupiny vlastníků doma. Dále velikost souboru, datum poslední změny a nakonec jméno souboru.

Vytvoření a odstranění adresáře

Adresář lze vytvořit pomocí příkazu `mkdir` (make directory).

```
mkdir Documents
mkdir -p a/b/c
```

Příkaz `rmdir` (remove directory) adresář odstraní. Odstraní ale pouze *prázdný adresář*.

```
rmdir Documents
rmdir -p a/b/c
```

Pro odstranění neprázdného adresáře je třeba použít příkaz `rm -R adresarPlnyDat`.

Kopírování, mazání a přejmenování souborů

Pro kopírování slouží příkaz `cp` (copy), pro mazání `rm` (remove) a pro přejmenování nebo přesun `mv` (move). Všechny tyto programy mají několik společných prepínačů a chovají se podobně.

Přepínače programů `cp`, `mv`, `rm`

- | Přepínač | popis |
|----------|---|
| -v | příkaz bude "ukecaný" a bude vypisovat co právě dělá |
| -i | interaktivní – při přepisu/odstranění se bude ptát |
| -f | (force) nebude se na nic ptát a bude přepisovat/odstraňovat |

-R -r od

Příklad použití příkazů cp, mv, rm

Příkaz	popis
cp souborA cesta/nekde/souborB	kopíruje souborA do souborB
cp souborA souborB cesta/k/cilovemu/adresar/	kopíruje souborA a souborB do cílového adresáře
mv soubor cilovy/adresar/	přesune soubor z pracovního adresáře do cílového adresáře
rm souborA souborB cesta/souborC	vymaže soubory A,B,C
rm -R adresar	smaže celý adresář včetně všeho, co je uvnitř
cp -R cesta/adresar cilove/umi steni	umí steni existuje, takže adresar bude nakopírován do umí steni
cp -R adresar umi steni /noveJmeno	noveJmeno neexistuje, takže adresar bude nakopírován do umí steni pod jménem noveJmeno.

Odkazy

V operačním systému Unix rozlišujeme pevné odkazy -- hardlink a symbolické odkazy -- symlink. Pro jejich vytvoření se používá příkaz `ln`.

Pevný odkaz přidává další jméno, pomocí kterého můžeme k souboru přistoupit. Metadata o souboru jsou uložena v tzv. i-uzelu. Mimo jiné i-uzel obsahuje počet odkazů na daný soubor. Jestli-že je vytvořen nový pevný odkaz zvětší se počet odkazů o jeden.

Pevný pevný odkaz na adresář může vytvořit jen superuživatel `root`.

Vytvořením **symbolického odkazu** dojde ke vzniku nového souboru s vlastním i-uzlem, který ukazuje na jiný soubor nebo adresář. Cesta k souboru, na který symlink ukazuje může být absolutní nebo relativní. Pro vytvoření symbolického odkazu je třeba k příkazu `ln` přidat parametr `-s`.

Informace o odkazech si můžeme prohlédnout pomocí příkazu `ls -l`

```
$ ls -l muj_adresar
celkem 28K
-rwxr-xr-x 1 marek marek 406 lis  3 18:20 index.pl*
lrwxrwxrwx 1 marek marek  15 úno 21  2013 kontext.menu -> ./kontext.menu
-rw-r--r-- 1 marek marek 7,4K lis 14 11:11 obsah.mdwn
-rw-r--r-- 1 marek marek 13K lis  7 21:01 tisk.html
```

V druhém sloupci výpisu vidíme číslo 1. To říká, že na daný soubor existuje jen jeden odkaz -- má jen jedno jméno. Soubor `kontext.menu` je symbolický odkaz a ve výpisu je patrné, na jaký soubor ukazuje.

Nyní vytvoříme jeden pevný a jeden symbolický odkaz pro soubor `tisk.html`

```
$ ln tisk.html print.html
$ ln -s print.html sajtna
```

... a můžeme se podívat co se stalo. Přepínač `-i` příkazu `ls` vypisuje číslo i-uzlu.

```
$ ls -li
celkem 48K
96479118 -rwxr-xr-x 1 marek marek 406 lis 3 18:20 index.pl*
96478792 lrwxrwxrwx 1 marek marek 15 úno 21 2013 kontext|menu -> ../kontext.menu
96473567 -rw-r--r-- 1 marek marek 8997 lis 14 11:22 obsah.mdwn
96479327 -rw-r--r-- 2 marek marek 12428 lis 7 21:01 print.html
96479361 lrwxrwxrwx 1 marek marek 10 lis 14 11:20 sajtna -> print.html
96479327 -rw-r--r-- 2 marek marek 12428 lis 7 21:01 tisk.html
-rw-r--r-- 2 marek marek 13K lis 7 21:01 tisk.html
```

Vidíme, že soubor `tisk.html` a souboru `print.html` mají počet jmen 2 a číslo jejich i-uzlu je stejné.

Další příkazy

Pokud vás zajímá více, podívejte se na [Seznam unixových utilit](#) a na seriál [Unixové nástroje](#).

Zde je seznam těch nejvíce používaných:

- `test` -- porovnávání řetězců, čísel, souborů
- `df` -- obsazenost disků
- `du` -- velikosti souborů a adresářů
- `find` -- hledání souborů
- `basename` -- jméno souboru
- `dirname` -- adresář ve kterém je soubor
- `cmp` -- porovnání souborů
- `touch` -- změna časového razítka souboru
- `find` -- hledání souborů
- `grep` -- prohledávání souborů
- `diff` -- rozdíly mezi soubory
- `patch patch` -- aplikace rozdílů
- `comm` -- porovnávání souborů
- `cut` -- výpis sloupců
- `cat` -- spojování souborů
- `alias` -- vytváření zkratk nových příkazů
- `echo` -- tisk
- `printf` -- formátovaný tisk
- `head` -- začátek souboru
- `tail` -- konec souboru
- `less` -- zobrazení/stránkování souborů
- `more` -- zobrazení/stránkování souborů
- `awk` -- jazyk pro zpracování textu
- `sed` -- proudový editor
- `sort` -- seřazení řádků
- `paste` -- pojení souborů podle sloupců
- `nslookup` -- [DNS](#) dotazy
- `host` -- [DNS](#) dotazy
- `dig` -- [DNS](#) dotazy
- `ping` -- test síťového spojení

- netstat -- otevřená síťová spojení
- false, true -- Ano/Ne
- yes ano, ano, samozřejmě, že ano
- wc -- počítáč slov, řádků, znaků
- uniq -- vynechá opakující se řádky
- tee -- zapíše vstupní proud do souboru a na výstup
- whoami -- jak se jmenuji?
- who -- kdo je přihlášený?
- w ... a co dělá?
- uptime -- jak dlouho už to běží?
- uname -- jak se jmenuješ?
- wait -- čekám až skončíš
- xargs -- spustí zadaný příkaz pro každé "slovo" na vstupu
- file -- identifikuje soubor
- ps -- výpis procesů
- fuser -- který proces otevřel tento soubor?
- kill -- zasílání zpráv
- nice -- změna priority procesu
- wget -- změna priority procesu