



INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

PSK3-1

Název školy:	Vyšší odborná škola a Střední průmyslová škola, Božetěchova 3
Autor:	Ing. Marek Nožka
Anotace:	Základy ovládání unixového shellu
Vzdělávací oblast:	Informační a komunikační technologie
Předmět:	Počítačové sítě a komunikační technika (PSK)
Tematická oblast:	Operační systém Linux/Unix
Výsledky vzdělávání:	Žák zobrazí manuálovou stránku a dokáže v ní vyhledávat
Klíčová slova:	Linux, Unix, manuálová stránka, less
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

Úvod do Linuxu

Příkazový řádek

Základním komunikačním rozhraním v OS Unix je příkazový řádek. Ihned po přihlášení se uživateli spustí tzv. příkazový interpret neboli shell.

Ve většině Linuxových distribucí je jako výchozí shell nastaven program Bash. Ale nabídka je mnohem větší: Např. C shell, Korn shell nebo můj oblíbený Z shell.

Unixový shell

- poskytuje uživateli textové (příkazové) komunikační rozhraní -- CLI
- komunikuje s uživatelem
- interpretuje jeho příkazy
- spouští jiné programy

Shell je v podstatě skriptovací programovací jazyk. Shell vytváří prostředí příkazového řádku, do kterého uživatel zadává názvy příkazů, které chce spustit. Shell tyto příkazy interpretuje, spouští odpovídající programy a umožňuje sledovat jejich výstup. Umožňuje příkazům předávat parametry, seskupovat je, slučovat příkazy do skriptů a podobně. Shell představuje „skořápku“, která skrývá uživateli detaily ovládání kernelu (jádra) operačního systému. Mnoho uživatelů unixových systémů dosud považuje moderní příkazový řádek shellu za mnohem pohodlnější způsob ovládání počítače než grafické uživatelské rozhraní.

Příkaz obsahuje:

- jméno příkazu
- volby -- přepínače
- parametry
- metaznaky -- znaky se speciálním významem (mezera, > < * ? \ [] { })

Například:

```
ls --all -l data Documents
```

ls

je jméno příkazu

--all -l

jsou přepínače, které modifikují chování příkazu ls

data Documents

jsou parametry; v tomto případě názvy souborů nebo adresářů

mezera

je je tzv. metaznak (znak se speciálním významem) a slouží jako oddělovač parametrů a přepínačů

Volby (přepínače) mohou být dlouhé nebo krátké. Krátké přepínače jsou uvozené jednou pomlčkou - a je možné je spojovat: `ls -l i s a` je volání příkazu ls s volbami l, i, s, a a je ekvivalentní pro zápis `ls -l -i -s -a`.

Dlouhé (slovní) přepínače jsou uvozené dvěma pomlčkami --.

Například `ls --all` nebo `ls --format=si ngl e-col umn`.

Speciální parametr -- obvykle znamená konec voleb, tedy že další parametr není přepínač i když začíná pomlčkou, ale například název souboru, jehož jméno začíná pomlčkou.

Příkaz

```
ls -soubor
```

je totéž jako

```
ls -s -o -u -b -r
```

ale

```
ls -l -- -soubor
```

vypíše podrobné informace k souboru `-soubor`.

Ukázka přebírání parametrů programem

Každý programovací jazyk umožňuje převzít parametry, které mu na příkazové řádce shell předává. Většinou se tak děje přes proměnnou nebo pole `ARGV`. Dále uvedený příklad je v programovacím jazyku Python.

```
#!/usr/bin/python
# -*- coding: utf8 -*-
# Úloha: ukázka předávání parametrů příkazové řádky
#####
import sys, getopt

print "vsechno: ", sys.argv, '\n'

# funkci getopt řeknu, že povolené přepínače jsou -a -b a přepnaš -x má parametr
opts, args = getopt.getopt(sys.argv[1:], "abx:")

print "volby:", opts
print "parametry:", args, '\n'

' http://docs.python.org/2/library/getopt.html '
#try:
#    opts, args = getopt.getopt(sys.argv[1:], "ho:v", ["help", "output=1"])
#except getopt.GetoptError as err:
#    # print help information and exit:
#    print str(err) # will print something like "option -a not recognized"
#    usage()
#    sys.exit(2)
#output = None
#verbose = False
#for o, a in opts:
#    if o == "-v":
#        verbose = True
#    elif o in ("-h", "--help"):
#        usage()
#        sys.exit()
#    elif o in ("-o", "--output"):
#        output = a
#    else:
#        assert False, "unhandled option"
```

`--> stáhnout

```
./parametry.py -a -x param ahoj cau
vsechno: ['./parametry.py', '-a', '-x', 'param', 'ahoj', 'cau']

volby: [('-a', ''), ('-x', 'param')]
parametry: ['ahoj', 'cau']
```

Manuálové stránky

Každý příkaz (např. `ls` nebo `mkdi r`) má svou manuálovou stránku, ve které je popis daného příkazu, jeho syntaktická pravidla a popis

voleb a přepínačů. Manuálovou stránku zobrazíme příkazem `man`.

```
man jmenorogramu
```

Například

```
man ls
```

Příkaz `man` ve skutečnosti manuálovou stránku pouze najde a předá ji jinému programu, který ji zobrazí. Programu pro zobrazování textových souborů na terminál se říká stránkovač (pager). V našem případě (a doufám, že i na všech jiných rozumně nastavených Linuxových systémech) je stránkovačem program less.

Ovládání programu `less` (manuálové stránky)

- q ukončí manuálovou stránku
- h zobrazí nápovědu
- j, k pohyb v nahoru dolů stejně jako pomocí šípek
- /výraz najde první následující výskyt výrazu (řetězce)
- n najde další výskyt výrazu
- N najde předchozí výskyt výrazu
- ?výraz najde poslední předchozí výskyt výrazu (řetězce); význam n a N je přehozen

Výrazem může být obyčejná posloupnost znaků nebo regulární výraz.

Více se dozvíte při stisku klávesy h nebo v manuálové stránce.