

Jemný úvod do operačního systému Linux

Dan Svozil, Jirka Černý, Petr Jurečka, Vojta Klusák, Honza Kučera
Centrum komplexních molekulových systémů a biomolekul
Ústav organické chemie a biochemie

Operační systém

- 1 Operační systém je základní softwarové vybavení počítače, které se stará o správu systémových zdrojů.
- 2 zdroje: výpočetní čas procesoru, paměť, diskový prostor, přístup k síťovým rozhraním apod.
- 3 OS komunikuje s hardwarem pomocí ovladačů (drivers).

Filosofie systému Linux

- 1 jádro - kernel (stabilní: 2.6.8.1)
- 2 interpret příkazů - shell (textový režim, např. bash, csh)
- mocný programovací nástroj (tzv. skriptovací jazyk) - jiné skriptovací jazyky: Python (www.python.org), Perl
- 3 služební programy - utilities (elementární operace se soubory)
- 4 Linux je víceuživatelský (terminály) a víceúlohový. **xterm**
- 5 základní softwarové vybavení - např. gcc, Emacs, bash
- 6 distribuce = úhledný kabát (instalační programy, detekce, správa balíčků)
- 7 distribuce pro nováčky: Mandrake (www.mandrake.com), Fedora (fedora.redhat.com)
- 8 live distribuce (Knoppix - www.knoppix.org)

Vstup uživatele do systému

- 1 koexistuje více uživatelů - účty (account)
- 2 totožnost: uživatelské jméno (username) a heslo (password)
- 3 uživatelé: běžní, root
- 4 lokální či vzdálené (ssh) přihlášení (local, remote login)
- 5 **přihlášení se pomocí ssh**
username: stud01 - stud15, password: stud

ssh stud01@steel

První kroky v systému

- 1 po přihlášení se spouští login shell, prompt (\$ či #)
- 2 spuštění programu = napsat jeho název

Kde sehnat informace o Linuxu?

- 1 man (<PgUp>/<PgDown>, hledání pomocí /, q), **man man, /games**
- 2 FAQ - Frequently Asked Questions
- 3 www.google.com
- 4 Linux Documentation Project, www.tldp.org, knihy.cpress.cz
- 5 Linux Příručka českého uživatele, Vilém Vychodil
<http://vychodil.inf.upol.cz/errata/download/>
- 6 diskusní skupiny (groups.google.com, comp.os.linux.xx), pozor na RTFM (Read The Fucking Manual)

Použití příkazového řádku - bash

- 1 na příkazovém řádku se zadávají příkazy kterým Linux rozumí a podle kterých také něco provede
- 2 šipka nahoru/dolů - vyhledej předchozí příkaz (**zkuste**)
- 3 <Tab> - doplňování možností **ma<Tab>**
- 4 <SHIFT><PgUp>/<PgDown> - prohlížení ujetých obrazovek (**zkuste**)
- 5 <Ctrl+C> - ukončí běžící program
- 6 <Ctrl+D> - odlogování se

Soubory

- 1 stromová struktura, **tree**, **man tree**, **vyhledat depth**, **vyhledat -d**
- 2 nejvyšší adresář . . . **cd /**, **poté tree -d -L 1**, **tree -d -L 2**
- 3 relativní vs. absolutní cesta
 - absolutní cesta pokrývá celou větev počínaje kořenem (/var/spool)
 - relativní cesta se vztahuje k adresáři, ve kterém se nacházíte (jste v adresáři /var - ./spool/cron, .. odkazuje na /, ~/wc)
- 4 adresář home, /home/uživatel, ~, samotné cd vrátí do houmu
- 5 adresáře . a ..
- 6 Nyní již víme, jak se pohybovat po adresářovém stromu, ale jakým způsobem zjistíme obsah adresáře? **ls**
- 7 zástupné znaky: * (všechno), *.txt (aa.txt, bb.txt), *.tx* (aa.txt, bb.txt, cc.txcp)

Cvičení

- 1 změna adresáře ... `cd wc`
- 2 zjištění polohy ... `pwd`
- 3 vypsání obsahu adresáře ... `ls, ls -la, ls -lt, ls *, ls *.txt`
- 4 vypsání obsahu textového souboru ... `cat sa.txt, more sb.txt`
- 5 `cd`
- 6 `cd w<TAB>`
- 7 vypsání obsahu souboru `sa.txt`
- 8 `<SHIFT><PgUp>/<PgDown>`
- 9 šipka nahoru, vypsání obsahu souboru `sb.txt`
- 10 spustit program `zobraz`, `./`, `<Ctrl-C>`
- 11 postup o adresář výše, `<Ctrl-D>`, zpětně se přilogovat

Cvičení

- 1 vypsání obsahu home adresáře
- 2 přepnutí se do wc
- 3 vytištění řádky s daným vzorem . . . grep jedna sa.txt
- 4 cat sa.txt a ověřit
- 5 zjištění MP2 energie z Gaussianovského logu
grep EUMP2 anion.log
- 6 zjištění HF energie z Gaussianovského logu
grep "SCF Done" anion.log
- 7 vytvoření prázdného souboru . . . touch ach
- 8 pomocí šipek listujte v historii příkazů a vytvořte ještě soubory "uch"
a "joj"
- 9 zobrazte jména souborů obsahujících písmeno "c"

Vytváříme zmatek

- 1 mc - Midnight Commander
- 2 pohyb v okně, vstup do adresáře wc/, o adresář výše
- 3 klávesou <INS> vybrat soubory "ach", "uch" a "joj", smazat F8
- 4 <F7> ... vytvoří adresář, <F5> ... kopírování
- 5 <F7> adr2, <TAB>, do adresáře wc/, <INS> sa.txt,sb.txt, <F5>
- 6 <F4> ... editace souboru
- 7 v adr2, <F4> na sa.txt, <ESC>
- 8 <F8> ... mazání
- 9 v adr2 <+> <ENTER>, <F8>, .., adr2 <F8>
- 10 <F10>

Textové soubory a mcedit

- 1 standardní textové editory: vi (vim), Emacs
- 2 přepnutí do wc/
- 3 mcedit novy.txt - založíme neexistující soubor
- 4 Napsat text "Toto je muj prvni text." <ENTER>
"Druha radka."
- 5 <F2> - uložení, <F7> - hledání
- 6 vyhledat řetězec "text"
- 7 práce s bloky textu: <F3> - vytvoření bloku
- 8 <F3> - šipkami označit slovo "prvni" - opět <F3> - přesunout kursor
jinam, <F5> - kopírování, <F6> - přesuntí, <F8> - smazání
- 9 <F10> či <ESC><ESC> - ukončení

Opakovací cvičení

- 1 vytvořte ve svém home adresáři podadresář s libolným jménem
- 2 zkopírujte všechny *.xpm z /usr/X11R6/include/X11/pixmaps/ do nového adresáře
- 3 vytvořte adresář pro všechny pixmapy jejichž jména začínají na “c” a jiný pro pixmapy začínající na “m”, přesuňte pixmapy dle jmen a zbylé pixmapy vymažte
- 4 do nového adresáře překopírujte soubor sa.txt z ~/wc
- 5 jaké je číslo řádky obsahující slovo “tady”? (náповěda: man grep, hledejte “number”)
- 6 smažte celý vámi vytvořený adresář

Procesy

- 1 program = data uložená na disku, proces = vykonávaný program
- 2 jádro zasahuje do života procesu při jeho vytvoření a ukončení a jádro se také stará o přepínání běhu mezi procesy - souběžné zpracování (multithreading)
- 3 procesy jsou identifikovány číslem PID a tvoří stromovou strukturu **pstree**
- 4 uživatelské (userspace) × systémové (daemons) procesy
- 5 informace o procesech **ps**
 - **ps** . . . seznam aktivních procesů spojených s aktuálním terminálem
 - **ps -u uživatel** . . . vypíše procesy patřící uživateli
- 6 informace o procesech **top**
 - **q, h**

Procesy - pokračování

- 1 běh procesu na popředí **wc/zapis**
- 2 pozastavení procesu **<Ctrl-Z>**, **tail zapout.txt**, **jobs**
- 3 přenesení procesu na pozadí **bg**, **tail zapout.txt**, **tail -f zapout.txt**
- 4 přenesení procesu na popředí **fg**
- 5 **<Ctrl-C>**
- 6 přímé spuštění procesu na pozadí **zapis &**
- 7 ukončování procesů = zasílání signálů **kill -signál PID**
 - kill PID
 - kill -9 PID

Přesměrování vstupu a výstupu

- 1 standardně je vstupem klávesnice a výstupem obrazovka
- 2 přesměrování výstupu
 - do souboru >
 - použít výstup jednoho programu jako vstup do druhého (pipe) |

Cvičení

- 1 `wc/zobraz` - spusťte, `Ctrl-C`
- 2 přesměrujte výstup ze `zobraz` do souboru a spusťte ho na pozadí
- 3 ověřte, že se do soubory opravdu vypisuje
- 4 zabijte proces `zobraz`

Přístupová práva

- 1 Každý soubor má svého vlastníka. Vytvořením souboru se stáváte vlastníkem vy. **vytvořte ve wc/ prázdný soubor mceditem a mcredit opustte, ls -la** Pouze root může změnit vlastníka.
- 2 Kromě uživatele je všem souborům ještě přiřazena skupina.
- 3 Přístupová práva k soubory se přidělují třem základním kategoriím uživatelů:
 - user (owner) - vlastník souboru
 - group - člen skupiny souboru
 - others - ostatní
- 4 Přidělují se tři základní práva:
 - read (r) - čtení
 - write (w) - zápis
 - execute (x) - spouštět (dáno právem, nikoliv příponou .exe)

Přístupová práva - nastavení

- 1 `chmod ugoa +-= rwx jmeno.souboru`
uoa . . . user group others all
+ -= . . . přidat ubrat absolutně
rwx . . . čtení zápis vykonávání

Cvičení

- 1 v adresáři `wc/` odeberte souboru `zobraz` právo vykonávání, spusťte
- 2 změňte na něm právo pro uživatele pouze na `x` (použijte `=`), spusťte
- 3 odeberte souboru `sa.txt` právo na zápis, `mcedit`
- 4 přidejte mu právo na zápis a uberte právo na čtení, `mcedit`
- 5 přidejte mu právo na čtení