

# GNU/Linux

## Správa a údržba systému



**slackware**  
l i n u x

**Zostavil a prednášal:**

Bc. Jaroslav Imrich  
jariq@jariq.sk

**Editoval a exportoval:**

Bc. Michal Kopček

KAIA MfF STU 2005/2006

# GNU/Linux – Správa a údržba systému

Lekcia 1 – Úvod	4
Lekcia 2 – Práca so súbormi a adresármi	12
Lekcia 3 – Práca s pamäťovými médiami	33
Lekcia 4 – Získavanie informácií o systéme	43
Lekcia 5 – Inštalácia Slackware linuxu	51
Lekcia 6 – Inštalácia dodatkového softvéru	69
Lekcia 7 – Skriptovanie v bashi	87
Lekcia 8 – Sieť	100
Lekcia 9 – Sieťové služby	111
Lekcia 10 – Kompilácia jadra	122
Lekcia 11 – Úvod do bezpečnosti	132



# **GNU/LINUX**

## **Správa a údržba systému**

### **Lekcia 1**

#### **Úvod**

**Zostavil: Bc. Jaroslav IMRICH**

# Richard Stallman - GNU

- 1984 – GNU project (Gnu's Not Unix)
- Pôvodný zámer bol vyvinúť voľne dostupný unixový operačný systém GNU
- Popri tom vyvinuli GCC (GNU Compilers Collection), Emacs..



# Linus Torvalds - Linux

- 1991 – kernel s názvom “Linux”
- Skratka z "Linus's Minix"
- Linux je jadro, nie OS !
- Podporuje architektúry Intel, Alpha, SPARC, PowerPC a ďalšie
- V súčasnej dobe dve vetvy 2.4 a 2.6
- [www.kernel.org](http://www.kernel.org)





# GNU/Linux



Operačný systém !

Jadro (Linux)  
a obslužný softvér (prevažne GNU)

Bežný používateľ používa rôzne distribúcie  
systému GNU/Linux

Mandriva, Fedora Core, OpenSuse  
Gentoo, Debian, Slackware

# Výhody systému GNU/Linux

- Viacpoužívateľský systém
- Možnosť optimalizovať systém pre použitý hardvér
- Možnosť automatizovať úlohy skriptami
- Rozsiahla komunita a podpora na internete
- Dostupnosť zdrojových kódov
- Dostupnosť množstva softvéru pre server i pracovnú stanicu
- Cena



# GNU/Linux vs. Windows

- Konfigurovateľnosť systému
- Otvorenosť zdrojových kódov
- Oddelené grafické prostredie
- Uvedomelosť používateľov
- Princíp KISS (Keep It Simple Stupid)
- Nedostupnosť zdrojových kódov
- Integrované grafické prostredie
- Uvedomelosť používateľov je menšia

# Zdroje informací

- Manuálové stránky – příkaz *man* a *apropos*
- Vyhledávač Google - [www.google.com](http://www.google.com)
- Diskusné fóra – napr. [www.linuxquestions.org](http://www.linuxquestions.org)
- The Linux Documentation Project - [www.tldp.org](http://www.tldp.org)
- Kysela M. Přecházíme na Linux. Brno: Computer Press, 2003. ISBN 80-7226-844-9
- Průručka systémového administrátora  
<http://deja-vix.sk/sysadmin/>
- Webové portály: napr. [www.linuxos.sk](http://www.linuxos.sk), [www.root.cz](http://www.root.cz)

[www.fuckinggoogleit.com](http://www.fuckinggoogleit.com)



# **GNU/LINUX**

## **Správa a údržba systému**

### **Lekcia 2**

Práca so súbormi a adresármi

**Zostavil: Bc. Jaroslav IMRICH**

# Prihlásenie a odhlásenie

## Prihlásenie

Pri výzve “**login:**” zadáme platné používateľské meno a heslo

Po úspešnom prihlásení sa spustí príkazový interpret tzv. shell - tcsh, bash a ďalšie

## Odhlásenie

Príkazmi *logout*, *exit* alebo kombináciou kláves **CTRL+D**

# Midnight Commander



The screenshot displays the Midnight Commander (mc) interface, a dual-pane file manager. The top menu bar includes 'Left', 'File', 'Command', 'Options', and 'Right'. The main area is split into two vertical panels, both showing a directory listing of the root directory. The listing includes columns for 'Name', 'Size', and 'MTime'. The left panel shows the current directory as '/bin'. The right panel shows the current directory as '/bin'. Below the panels, a hint message reads: 'Hint: You can do anonymous FTP with mc by typing 'cd /#ftp:machine.edu''. The command prompt shows 'jariq@joybook:/\$'. At the bottom, a status bar lists keyboard shortcuts: 1Help, 2Menu, 3View, 4Edit, 5Copy, 6RenMov, 7kdir, 8Delete, 9PullDn, 10Quit.

Name	Size	MTime	Name	Size	MTime
/bin	4096	Sep 18 22:32	/bin	4096	Sep 18 22:32
/boot	4096	Sep 18 20:54	/boot	4096	Sep 18 20:54
/dev	40960	Oct 22 18:24	/dev	40960	Oct 22 18:24
/etc	4096	Oct 22 18:24	/etc	4096	Oct 22 18:24
/home	4096	Sep 25 11:05	/home	4096	Sep 25 11:05
/lib	4096	Oct 12 12:28	/lib	4096	Oct 12 12:28
/lost+found	16384	Sep 18 21:58	/lost+found	16384	Sep 18 21:58
/mnt	4096	Sep 18 20:49	/mnt	4096	Sep 18 20:49
/opt	4096	Oct 21 09:33	/opt	4096	Oct 21 09:33
/proc	0	Oct 22 20:24	/proc	0	Oct 22 20:24
/root	4096	Oct 23 00:23	/root	4096	Oct 23 00:23
/sbin	4096	Oct 12 12:28	/sbin	4096	Oct 12 12:28
/sys	4096	May 12 2004	/sys	4096	May 12 2004
/tmp	4096	Oct 23 00:03	/tmp	4096	Oct 23 00:03
/usr	4096	Sep 3 23:06	/usr	4096	Sep 3 23:06
/var	4096	Sep 3 23:06	/var	4096	Sep 3 23:06

Hint: You can do anonymous FTP with mc by typing 'cd /#ftp:machine.edu'  
jariq@joybook:/\$  
1Help 2Menu 3View 4Edit 5Copy 6RenMov 7kdir 8Delete 9PullDn 10Quit

Populárny súborový manažér, pracujúci v textovom režime, použiteľný takmer na všetkých unixových systémoch. Spúšťa sa príkazom *mc*.

# Adresárový strom

- / (root)
  - Koreňový adresár
- **/bin** (binaries)
  - Základné spustiteľné programy
- **/boot**
  - Obrazy jadier a ďalšie súbory potrebné pre zavedenie (bootovanie) systému
- **/dev** (devices)
  - Špeciálne súbory reprezentujúce hardvér

# Adresárový strom

- **/etc**
  - Konfiguračné súbory, zavádzacie skripty...
- **/home**
  - Domovské adresáre používateľov systému
- **/lib** (libraries)
  - Zdieľané knižnice programov
- **/mnt** (mount)
  - Adresár pre pripájanie pamäťových médií



# Adresárový strom

- **/opt** (optional)
  - Adresár pre inštaláciu dodatočných programov, prístupných všetkým používateľom
- **/proc**
  - Pseudo súborový systém zobrazujúci procesy a informácie o systéme a jeho stave
- **/sbin** (system binaries)
  - Základné spustiteľné programy pre správcu systému, ktorým je používateľ **root**

# Adresárový strom

- **/tmp** (temporary files)
  - Adresár pre dočasné (pomocné) súbory...
- **/usr** (user)
  - Všetky ostatné programy a knižnice, ktoré nie sú potrebné pri zavádzaní systému
- **/var** (variable data)
  - Súbory, ktoré počas behu menia svoju veľkosť  
napr. logy, elektronická pošta, databázy

# Adresárový strom

- Zástupné znaky pre adresáre
  - Koreňový adresár sa označuje /
  - Adresár o úroveň vyššie sa označuje ..
  - Aktuálny adresár sa označuje .
  - Domovský adresár sa označuje ~
- Príklad:
  - Adresár /home/jariq/projekt môžem zapísať ako ~/projekt, pokiaľ je /home/jariq môj domovský adresár.

# Používatelia

- Správca a absolútny pán systému je **root**
- Používatelia sú zoskupovaní do skupín
- Každý používateľ má, resp. môže mať:
  - Meno (login name)
  - Heslo
  - UID (user ID) – jedinečný identifikátor používateľa
  - GID (group ID) – identifikátor primárnej skupiny
  - Popis (user description) – napr. celé meno
  - Domovský adresár – napr. /home/jariq
  - Predvolený shell – napr. /bin/bash

# Používatelia

- Zoznam používateľov s informáciami o nich je v súbore **/etc/passwd**
- Zoznam skupín s priradenými členmi je v súbore **/etc/group**
- Ak je v systéme použitý program shadow, nie sú šifrované heslá používateľov uložené v **/etc/passwd** ale v **/etc/shadow**, ktorý musí byť čitateľný iba pre používateľa root.
- Ak niekto získa súbor **/etc/shadow** po relatívne jednoduchom, ale časovo náročnom dešifrovaní, získa heslo roota, a teda vlastní celý systém.

# Práca so súbormi a adresármi

- ***pwd***
  - Zobrazí názov aktuálneho adresára
- ***ls [parametre] [adresár]***
  - Zobrazí obsah adresára
  - Príklad: *ls -la /home/jariq*
- ***cd adresár***
  - Zmení aktuálny adresár
  - Príklad: *cd ../praca*

# Práca so súbormi a adresármi

- ***touch* súbor**
  - Dá sa pomocou neho vytvoriť prázdny súbor
  - Príklad: *touch test.txt*
- ***rm* [parametre] [súbor/adresár]**
  - Vymaže súbory alebo adresáre
  - Príklad: *rm test.txt*
- ***mkdir* adresár**
  - Vytvorí prázdny adresár
  - Príklad: *mkdir praca*

# Práca so súbormi a adresármi

- ***rmdir*** adresár
  - Vymaže prázdny adresár
  - Príklad: *rmdir praca*
  - Neprázdny adresár vymaže: *rm -r praca*
- ***cp* [parametre] zdroj cieľ**
  - Skopíruje súbory/adresáre zo zdroja do cieľa
  - Príklad: *cp test.txt test-zaloha.txt*
  - Príklad: *cp test.txt /home/jariq/praca/final.txt*
  - Príklad: *cp -r ~/praca ~/praca-zaloha*



# Práca so súbormi a adresármi

- ***mv zdroj cieľ***
  - Presunie (premenuje) súbor
  - Príklad: *mv ~/test.txt ~/praca*
- ***file súbor***
  - Zobrazí typ súboru
  - Príklad: *file test.txt*
- ***du [parametre] [súbor/adresár]***
  - Zobrazí veľkosť súboru/adresára
  - Príklad: *du -sh praca*

# Práca so súbormi a adresármí

- ***cat* súbor**

- Vypíše obsah súboru
- Príklad: *cat ~/test.txt*

- ***tac* súbor**

- Vypíše obsah súboru odzadu
- Príklad: *tac ~/test.txt*

- ***more* súbor**

- Vypíše obsah súboru po obrazovkách
- Posúva sa stisnutím medzerníka
- Príklad: *more /etc/services*

# Práca so súbormi a adresármí

- ***less* súbor**
  - Interaktívne vypíše obsah súboru
  - Posúva sa šípkami hore a dole
  - Príklad: *less /etc/services*
- ***tar* [parametre] archív [súbory]**
  - Vytvorí/rozbalí archív
  - Príklad: *tar -cf zaloha.tar ~/praca*
  - Príklad: *tar -xvf zaloha.tar*

# Práca so súbormi a adresármí

- ***gzip* súbor a *gunzip* súbor**
  - Komprimuje/dekomprimuje súbor
  - Príklad: *gzip zaloha.tar*
  - Príklad: *gunzip zaloha.tar.gz*
- ***bzip2* súbor a *bunzip2* súbor**
  - Komprimuje/dekomprimuje súbor
  - Používa výkonnejšie algoritmy než *gzip*
  - Príklad: *bzip2 zaloha.tar*
  - Príklad: *bunzip2 zaloha.tar.bz2*

# Práca so súbormi a adresármi

- ***ln*** [parametre] zdroj cieľ
  - Vytvára odkaz na súbor/adresár
  - Pevné a symbolické odkazy
  - Príklad: *ln /etc/passwd ~/pouzivatelia.txt*
  - Príklad: *ln -s muzika /mnt/disk/hudba/mp3*
- ***chown*** používateľ súbor
  - Zmení vlastníka súboru/adresára
  - Príklad: *chown jariq test.txt*
  - Podrobný výpis sa robí pomocou ***ls -la***

# Práca so súbormi a adresármi

- ***chmod*** práva súbor

- Mení prístupové práva pre súbor/adresár
- Právo čítať (4), zapisovať (2) a spúšťať (1)
- Výsledné práva vznikajú súčtom hodnôt
- Každý súbor má definované práva pre tri objekty: vlastníka, vlastníacu skupinu a všetkých ostatných
- Príklad: *chmod 644 test.txt* (pridelí súboru právo čítať a zapisovať pre vlastníka a právo čítať pre vlastníacu skupinu a ostatných)

# Práca so súbormi a adresármí

- ***chgrp*** skupina súbor
  - Mení vlastníacu skupinu súboru/adresára
  - Príklad: *chgrp users test.txt*
  - Vlastnícka skupina sa dá zmeniť aj pomocou príkazu *chown*
  - Príklad: *chown jariq:users test.txt*





# **GNU/LINUX**

## **Správa a údržba systému**

### **Lekcia 3**

Práca s pamäťovými médiami

**Zostavil: Bc. Jaroslav IMRICH**

# Rekapitulácia lekcie 2

- Vieme, čo znamenajú symboly: / . .. ~
- Vieme, čo sa nachádza v súboroch:  
**/etc/passwd, /etc/group, /etc/shadow**
- Vieme používať príkazy a programy: **logout, exit, mc, pwd, ls, cd, touch, rm, mkdir, rmdir, cp, mv, file, du, cat, tac, more, less, tar, gzip, gunzip, bzip2, bunzip2, ln, chown, chmod, chgrp**

# Práca s pamäťovými médiami

- **Označovanie diskových jednotiek v Linuxe**
  - Na primárnom ATA radiči je master disk reprezentovaný špec. súborom zariadenia /dev/hda, slave disk zas /dev/hdb
  - Na sekundárnom ATA radiči je master disk reprezentovaný špec. súborom zariadenia /dev/hdc, slave disk zas /dev/hdd
  - Partície sa označujú pridaním poradového čísla, napr. /dev/hda1, /dev/hdc3 atď.
  - SATA disky sú reprezentované špec. súbormi zariadení /dev/sda, /dev/sdb atď.

# Práca s pamäťovými médiami

- ***fdisk* [parametre] zariadenie**
  - Umožní vytvárať a mazať partície na diskoch
  - Príklad: *fdisk -l /dev/hda*
  - Príklad: *fdisk /dev/hda*
- ***cdisk* zariadenie**
  - Program podobný *fdisku*, ale má prívetivejšie používateľské prostredie
  - Príklad: *cdisk /dev/hda*

# Práca s pamäťovými médiami

- ***mkfs -t fs partícia***

- Vytvorí na partícii súborový systém ext2/ext3

- Príklad: *mkfs -t ext2 /dev/hda1*

- Príklad: *mkfs -t ext3 /dev/hda1*

- ***mkswap partícia***

- Vytvorí swap partíciu

- Príklad: *mkswap /dev/hda3*

- Ak chceme, aby systém začal swap partíciu používať okamžite po vytvorení, je nutné ju pripojiť príkazom: *swapon /dev/hda3*

# Práca s pamäťovými médiami

- ***mount -t typfs -o voľby zariadenie adresár***
  - Pripojí súborový systém (filesystem)
  - Typy fs: ext2, ext3, reiserfs, vfat, ntfs, smbfs, iso9660 a ďalšie
  - Príklad: *mount /dev/cdrom /mnt/cdrom*
  - Príklad: *mount -t ext3 /dev/hdc1 /var/storage*
  - Príklad: *mount -t smbfs -o username=administrator //10.1.1.1/C\$ /mnt/samba*

# Práca s pamäťovými médiami

- ***umount* zariadenie | adresár**
  - Odpojí súborový systém (filesystem)
  - Príklad: *umount /dev/cdrom*
  - Príklad: *umount /mnt/cdrom*
- Statické záznamy o pripojiteľných súborových systémoch sú zapísané v súbore **/etc/fstab** (man fstab). Každý záznam v tomto súbore obsahuje informácie o pripájanom zariadení, pripájacom bode, type súborového systému, voľbách pripojenia fs ! (man mount), o tom, či má byť fs zálohovaný a o poradí kontrolovania pri bootovaní.

# Práca s pamäťovými médiami

- ***df* [parametre] zariadenie**
  - Zobrazí info o využití miesta
  - Príklad: *df -h /dev/hda1*
- ***find* adresár -name súbor -print**
  - Vyhľadá súbor v zadanom adresári a jeho podadresároch
  - Pri vyhľadávaní program prechádza všetky adresáre spĺňajúce podmienku. Preto môže vyhľadávanie dlho trvať, avšak je aktuálne.
  - Príklad: *find / -name fstab -print*



# Práca s pamäťovými médiami

- ***locate* súbor**

- Vyhľadá súbor v databáze súborov
- Výhoda oproti *find* je rýchlosť, nevýhoda je, že databázu treba aktualizovať
- Databáza sa aktualizuje príkazom *updatedb* alebo *locate -U*
- Príklad: *locate fstab*



# **GNU/LINUX**

## **Správa a údržba systému**

### **Lekcia 4**

Získavanie informácií o systéme

**Zostavil: Bc. Jaroslav IMRICH**

# Získavanie informácií o systéme

- **w**
  - Zobrazí, kto je v systéme prihlásený a čo robí
  - Príklad: *w*
- **who**
  - Zobrazí, kto je v systéme prihlásený
  - Poskytuje menej informácií než *w*
  - Príklad: *who*
- **uptime**
  - Zobrazí, ako dlho je systém spustený
  - Príklad: *uptime*

# Získavanie informácií o systéme

- ***uname* [parametre]**
  - Zobrazí informácie o systéme, jadre...
  - Príklad: *uname -r*
  - Príklad: *uname -a*
- ***cat /proc/cpuinfo***
  - Zobrazí informácie o procesore
  - Využitie pseudo súborového systému /proc
  - Príklad: *cat /proc/cpuinfo*

# Získavanie informácií o systéme

- ***free***

- Zobrazí množstvo využitej a voľnej pamäte
- Príklad: *free*

- ***lspci***

- Zobrazí informácie o PCI zariadeniach
- Príklad: *lspci*

# Získavanie informácií o systéme

- ***ps* [parametre]**

- Zobrazí informácie o bežiacich procesoch
- Príklad: *ps aux*
- Vo výpise príkladu je dôležitý najmä druhý stĺpec – PID. PID je číslo jednoznačne identifikujúce procesy systému. Ak chceme ovplyvniť bežiaci proces (napr. ukončiť ho), musíme poznať jeho PID.
- Bežný používateľ môže ovplyvňovať len svoje procesy, používateľ root všetky.

# Získavanie informácií o systéme

- ***ps***

- Zobrazí strom procesov
- Náзорnejšia interpretácia než pri *ps*
- Príklad: *ps*

- ***top***

- Zobrazí informácie o systéme a rebríček procesov podľa vyťaženia procesora
- Príklad: *top*



# Získavanie informácií o systéme

- ***kill* -signal PID**

- Preruší proces
- Príklad: `kill -9 34263`
- Ak zmrzne proces v textovom režime na prvom termináli (`tty1`), môžeme sa kombináciou kláves `ALT+F1`, `F2`, `F3`.. prepnúť na ďalší terminál a z toho po prihlásení ukončiť proces v `tty1` práve pomocou príkazu *kill*. PID procesu zistíme už známym príkazom *ps aux*.



# **GNU/LINUX**

## **Správa a údržba systému**

### **Lekcia 5**

#### **Inštalácia Slackware linuxu**

**Zostavil: Bc. Jaroslav IMRICH**

# Rekapitulácia lekcie 3 a 4

- Vieme, čo znamená:  
**hda1, sdb3, ext2, ext3, swap**
- Vieme, čo sa nachádza v súbore: **/etc/fstab**
- Vieme používať príkazy a programy:  
**fdisk, cfdisk, mkfs, mkswap, mount, umount, df, find, locate, w, who, uptime, uname, cat /proc/cpuinfo, free, lspci, ps, pstree, top, kill**

# Inštalácia Slackware Linuxu

- Slackware Linux je distribúcia s viac ako desaťročnou tradíciou.
- Vytvoril a udržiava ju prevažne jeden človek – Patrick Volkerding.
- Slackware Linux je známy hlavne svojou jednoduchosťou, "čistotou" a podobou BSD systémom.
- Neobsahuje zbytočne veľa grafických nástrojov, preto panuje názor, že nie je vhodný pre začínajúcich používateľov.

# Inštalácia Slackware Linuxu

- Súčasná verzia s označením 10.2 je na mirroroch dostupná v 4 iso obrazoch CD.
- Prvé CD obsahuje celý systém, na druhom CD je grafické prostredie KDE s lokalizáciami a tretie a štvrté CD obsahuje zdrojové kódy jednotlivých programov.
- Existuje viacero možností ako inštalovať Slackware. Najjednoduchšia je použiť inštalačné CD. Inštalácia sa spúšťa naboťovaním prvého CD.

# Inštalácia Slackware Linuxu

- Po naboťovaní z CD1 sa zobrazí výzva *boot:*
- Tu si môže použivateľ vybrať, aké jadro pouivije počas inštalácie a poprípade mu odovzdať ďalšie parametre.
- Na systéme bez podpory ACPI stačí, ak použivateľ stlačí enter. Pre podporu ACPI treba zadať *bareacpi.i* a pre podporu sata diskov je k dispozícii kernel *sata.i*

# Inštalácia Slackware Linuxu

- Po zavedení jadra má používateľ možnosť vybrať si rozloženie klávesnice.
- Do inštalačného systému sa prihlásime ako root, bez hesla.
- Následne by sme mali rozdeliť disk a pripraviť min. jednu partíciu typu linux a jednu typu swap. Na to môžeme použiť program *cfdisk*.
- Sprievodcu inštaláciou spustíme zadaním príkazu *setup*.



# Inštalácia Slackware Linuxu

- V inštalačnom programe sú jednotlivé časti inštalácie systému zobrazené zvisle - zhora nadol
- Vo väčšine prípadov stačí začať treťou možnosťou – pridaním swap partície.
- V ďalších krokoch len nasledujeme rady sprievodcu inštaláciou, až sa dostaneme k výberu skupín balíkov.

# Inštalácia Slackware Linuxu

- Na výber máme z predvolených skupín balíkov:  
a, ap, d, e, f, k, kde, kdei, l, n, t, tcl, x, xap, y.
- Balíky v skupine **a** tvoria základný systém. Pre minimalistický systém by postačila táto skupina a jadro.
- Balíky zo skupín **d,k,l** sú potrebné, ak chceme kompilovať softvér zo zdrojových kódov.
- Ak chceme, aby systém mohol pracovať so sieťou, je nutné nainštalovať balíky zo skupiny **n**.
- Pre systém, na ktorom sa bude používať grafické rozhranie, sú potrebné balíky zo skupín **x, xap**, poprípade **kde** a **kdei**.

# Inštalácia Slackware Linuxu

- Po výbere skupín balíkov nasleduje užší výber jednotlivých balíkov, ktorý môže byť vykonávaný napríklad spôsobmi *full*, *expert* a *newbie*.
- Pri type výberu *full* sú nainštalované všetky balíky z vybratých skupín.
- Pri type výberu *expert* sú zobrazené názvy balíkov bez popisu a používateľ si z nich môže vybrať.
- Pri type výberu *newbie* je pre každý balík zobrazovaný kompletný popis a používateľ sa môže rozhodnúť, či balík bude inštalovaný.

# Inštalácia Slackware Linuxu

- Po nainštalovaní balíkov prichádza na rad posledná, tzv. konfiguračná fáza.
- Vyberáme si v nej jedno z pripravených distribučných jadier. Pre väčšinu inštalácií by mali stačiť jadrá *bareacpi.i* alebo *sata.i*
- Pre zavedenie systému je nutné mať nainštalovaný zavádzač tzv. boot loader.
- Slackware linux používa boot loader **lilo**.
- Boot loader je dobré vo väčšine prípadov nainštalovať do **MBR** disku (master boot record).

# Inštalácia Slackware Linuxu

- Konfiguráciu adres siet'ových rozhraní môžete vykonať manuálne, alebo ak sa na vašej sieti používa, prenecháte ju DHCP serveru.
- V konfiguračnej fáze sa robí aj výber démonov (služieb) spustených po štarte systému. Ideálne riešenie je nechať po štarte spúšťať len *syslog* démona a ostatné vypnúť. **Spustené siet'ové služby bez dôkladnej konfigurácie znamenajú bezpečnostné riziko!**
- Tiež je dobré nechať spúšťať *hotplug* skripty, ktoré zabezpečia zavedenie modulov (ovládačov) pre hardvér vášho počítača.

# Inštalácia Slackware Linuxu

- Ďalšou z častí konfiguračnej fázy je výber predvoleného správcu okien. Dostupní sú iba správcovia, ktorých balíky sme nainštalovali. Napríklad fluxbox, xfce, kde atď.
- Dôležitou časťou je aj vytvorenie hesla pre používateľa root. Správne heslo by malo mať minimálne 8 znakov a malo by obsahovať malé i veľké písmena abecedy, čísla a tiež nejaké iné znaky napr. bodku alebo čiarku.
- Na záver treba už len opustiť inštalátor voľbou “*Exit*”, vybrať inštalačné CD a reštartovať počítač klávesovou skratkou CTRL+ALT+DEL.

# Inštalácia Slackware Linuxu

- Prvý krát sa do systému prihlásime ako používateľ root. Ako root by sme však v unixových systémoch mali pracovať iba v prípade, ak potrebujeme meniť konfiguráciu systému. Preto si hneď po prvom prihlásení vytvoríme používateľské konto pre našu bežnú prácu.
- ***useradd* [parametre] login**
  - Pridá do systému nového používateľa.
  - Príklad: *useradd -m -s /bin/bash jariq*

# Inštalácia Slackware Linuxu

- ***passwd*** [login]
  - Zmení heslo pre používateľský účet
  - Príklad: *passwd jariq*
- Taktiež je po inštalácii dobré skontrolovať konfiguráciu boot loadera lilo. Nachádza sa v súbore **/etc/lilo.conf** (man lilo.conf). Po vykonaní zmien v tomto súbore je nutné spustiť príkaz *lilo*, čím budú zmeny akceptované.
- Grafický server X-window (x.org) spustíme príkazom *startx*. Konfigurácia tohto serveru sa nachádza v súbore **/etc/X11/xorg.conf**



# Inštalácia Slackware Linuxu

- ***shutdown* [parametre] čas**
  - Bezpečne vypína/reštartuje systém
  - Príklad: *shutdown -h now*
  - Príklad: *shutdown -r now*
  - Príklad: *shutdown -h 14:32 &*
  - Príklad: *shutdown -c*
  - S podobnou funkciou existujú aj príkazy *halt*, *reboot* a *poweroff*. Podrobnosti o nich získate z manuálových stránok.

# Inštalácia Slackware Linuxu

- **Windows prepísal MBR a nemôžem sa dostať do Linuxu. Čo s tým ?**
  - Inštalačný program systému MS Windows bez upozornenia prepisuje MBR, kde je nainštalovaný lilo (linux loader).
  - Systém však fyzicky na disku ostáva, až kým neodstránime partície, kde bol nainštalovaný.
  - Obnova boot loaderu je jednoduchá. Stačí naboťovať inštalačné CD1, zaviesť z neho niektoré jadro a odovzdať mu ako parameter názov našej root partície.
  - Príklad: `bareacpi.i root=/dev/hda1 noinitrd ro`
  - Po prihlásení do systému ako root spustíme príkaz *lilo* a boot loader sa znovu zavedie do MBR.

# Inštalácia Slackware Linuxu

- **Už ma ten Linux nebaví. Ako sa dá odstrániť boot loader ?**
  - Prihlásite sa do systému ako root a príkazom *lilo -U* odinštalujete zavádzač z MBR.
  - Ak ste už odstránili linuxové partície a teda sa do systému už nemôžete prihlásiť, naboťujte inštalačné CD systému MS Windows. Vyberte možnosť opravy systému a v „recovery console“ zadajte príkaz *fixmbr*.



# **GNU/LINUX**

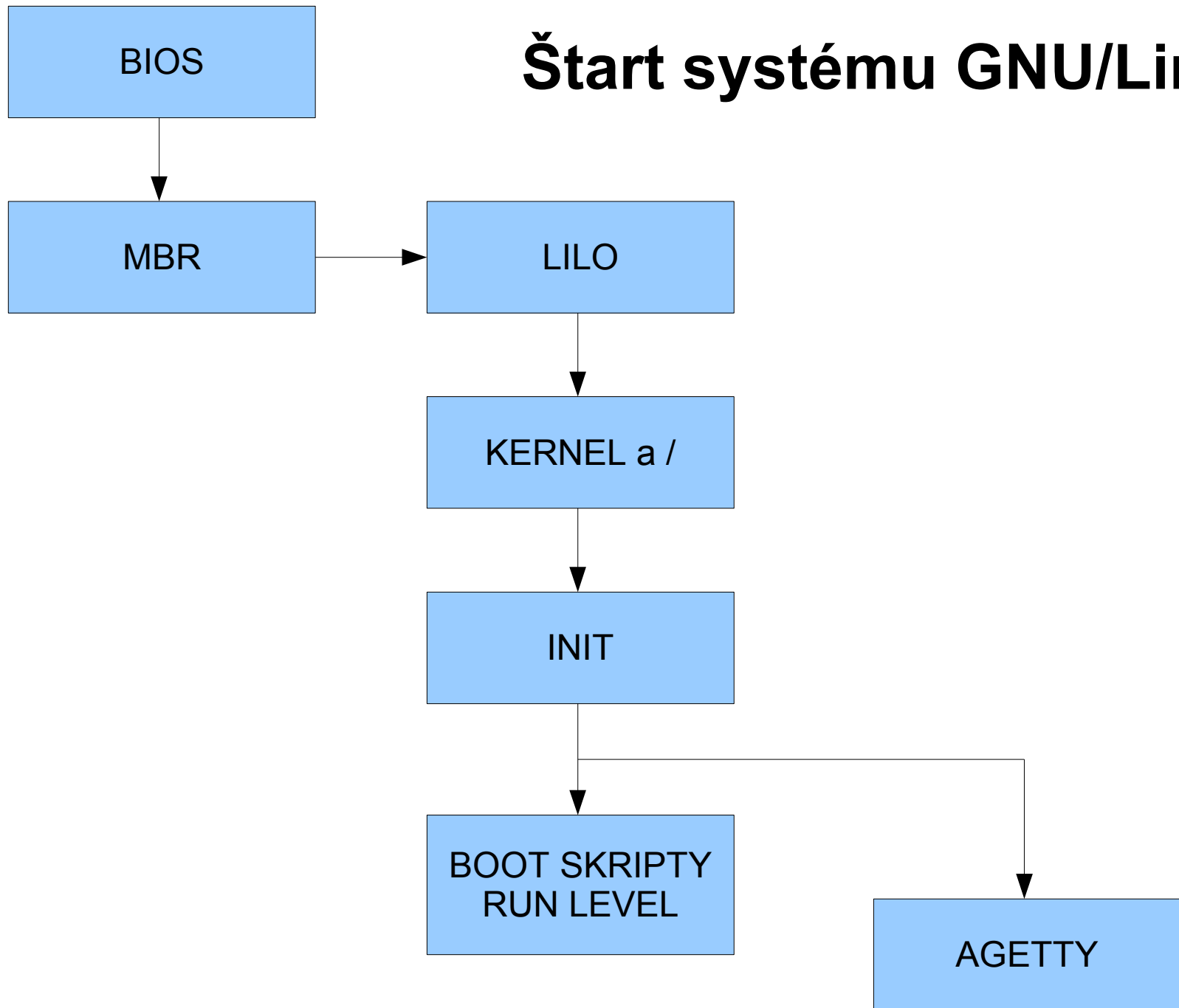
## **Správa a údržba systému**

### **Lekcia 6**

Inštalácia dodatkového softvéru

**Zostavil: Bc. Jaroslav IMRICH**

# Štart systému GNU/Linux



# Štart systému GNU/Linux

- Proces **init** má špeciálny význam. Jeho PID je vždy 1. Zabezpečuje spustenie boot skriptov pre konkrétne run levely a tiež spúšťa program **agetty**.
- Konfiguračný súbor init démona je **/etc/inittab**
- Run level je profil, v ktorom systém štartuje resp. vykonáva určité príkazy. Linuxový systém môže pri štarte v rôznych run leveloch spúšťať rôzne služby.

# Štart systému GNU/Linux

- Slackware Linux má 7 run levelov:
  - 0.Halt
  - 1.Single user mód
  - 2.Nevyužitý (nakonfigurovaný ako 3)
  - 3.Multiuser mód (predvolený)
  - 4.Grafický mód
  - 5.Nevyužitý (nakonfigurovaný ako 3)
  - 6.Reštart systému



# Štart systému GNU/Linux

- Boot skripty pre jednotlivé run levely a služby sa nachádzajú v adresári **/etc/rc.d**
- Tieto skripty môže používateľ upraviť podľa vlastnej vôle, ak je to ale možné, mal by sa snažiť svoje časti skriptov umiestňovať do súboru **/etc/rc.d/rc.local**
- Boot skripty Slackware Linuxu sú napísané s predpokladom:  
Ak má byť služba spustená, má jej štartovací skript spustiteľný atribút.

# Štart systému GNU/Linux

- Ak teda chceme zamedziť spusteniu samba démona pri štarte systému, odoberieme mu spustiteľný atribút.  
Príklad: *chmod 644 rc.samba*
- Procesom init je spúšťaný aj programagetty, ktorý na jednotlivých termináloch zobrazuje výzvu “login”. Ak sa používateľ úspešne prihlási, programagetty spustí jeho predvolený shell. Po odhlásení používateľa proces init opätovne spustí na danom termináli programagetty.

# Inštalácia dodatkového softvéru

- Rôzne distribúcie Linuxu používajú rôzne balíčkovacie systémy.
- Distribúcia Slackware používa **tgz** balíky, Fedora Core a Suse **rpm** balíky a Debian **deb** balíky..
- Balík obsahuje skompilované programy, ktoré sú priamo spustiteľné.
- Balíčkovací systém Slackware Linuxu nepozná závislosti jednotlivých balíkov.

# Inštalácia dodatkového softvéru

- Pre označovanie balíkov distribúcie slackware sa používa nasledovná norma:  
*názov-verzia-architektúra-revizia.tgz*  
Príklad: *micq-0.5.0.4-i686-1.tgz*
- Distribučné balíky Slackware Linuxu je možné stiahnuť z [www.slackware.org/pb](http://www.slackware.org/pb)
- Na správu balíkov je možné v distribúcii Slackware použiť napr. program **pkgtool**, pomocou ktorého je možné balíky inštalovať, odoberať..

# Inštalácia dodatkového softvéru

- Základné akcie s balíkmi je možné vykonávať v shelli pomocou programov **installpkg**, **removepkg**, **upgradepkg**, **explodepkg** a **makepkg**

Príklad: *installpkg micq-0.5.0.4-i686-1.tgz*

Príklad: *removepkg micq-0.5.0.4-i686-1.tgz*

# Inštalácia dodatkového softvéru

- **SLACKPKG**

- Pre pohodlnú správu balíkov, podobnú tej s nástrojmi apt Debian-u, je možné použiť program **slackpkg**. Tento program sa nachádza priamo v distribučných balíkoch vo vetve “extra”.
- Hlavná výhoda tohoto programu spočíva v tom, že je schopný požadovaný distribučný balík samostatne stiahnuť z predvoleného mirroru a vykonať s ním zadanú akciu.

# Inštalácia dodatkového softvéru

- **SLACKPKG**

- Preferovaný mirror sa definuje v súbore */etc/slackpkg/mirrors*
- Po vybratí resp. zmene mirroru je nutné aktualizovať zoznam dostupných balíkov príkazom *slackpkg update*
- Zoznam balíkov, s ktorými nemá nástroj slackpgg pracovať, sa uvádza v súbore */etc/slackpkg/blacklist* a je dobré uviesť tu napríklad *aaa\_elflibsv* a balíky jadra systému

# Inštalácia dodatkového softvéru

- **SLACKPKG**

- Po aktualizácii zoznamu balíkov dostupných na mirrore je možné s programom slackpkg vykonávať nasledovné akcie:
  - Vyhľadávať balíky: *slackpkg search dhcp*
  - Inštalovať balíky: *slackpkg install dhcpcd*
  - Odoberať balíky: *slackpkg remove dhcpcd*
  - Upgradovať balíky: *slackpkg upgrade dhcp*
  - Aplikovať všetky dostupné patche a upgrady: *slackpkg upgrade-all*



# Inštalácia dodatkového softvéru

- **Kompilácia zdrojových kódov**

- Mnohé programy pre systém GNU/Linux sú distribuované vo forme zdrojových kódov
- Aby sme mohli program používať (spúšťať), je nutné preložiť ho do formy zrozumiteľnej počítaču – “skompilovať ho”.
- Ak nie je v sprievodných README súboroch uvedené inak, vykonáva sa kompilácia pomocou príkazov:

**configure, make, make install**

# Inštalácia dodatkového softvéru

- **Kompilácia zdrojových kódov**

- **configure** je skript dodávaný priamo so zdrojovými kódmi daného programu, ktorý overuje schopnosť systému kompilovať a tiež kontroluje prítomnosť potrebných knižníc.
- Príkaz **make** automaticky zistí, ktoré časti zdrojových kódov treba prekompilovať a vykoná príkazy, ktoré to zabezpečia.
- Príkaz **make install** zabezpečí skopírovanie výsledných spustiteľných súborov na správne miesto v adresárovej štruktúre.

# Inštalácia dodatkového softvéru

- Nevýhoda štandardného postupu (configure, make, make install) je, že súbory tvoriace skompilovaný program sú nekontrolovateľne rozkopírované do rôznych adresárov a takto nainštalovaný program nie je vo väčšine prípadov možné odinštalovať.
- Túto, ale aj iné nevýhody, odstraňuje program **checkinstall**, ktorý dokáže počas kompilácie vytvoriť z daného programu balík tgz, rpm alebo deb.
- Checkinstall je možné nainštalovať pomocou:  
*slackpkg install checkinstall*

# Inštalácia dodatkového softvéru

- Pomocou programu checkinstall sa kompilácia vykonáva podobne ako bez neho. Tretí príkaz *make install* sa nahradí príkazom *checkinstall*.
- Kompiláciu teda vykonáme pomocou príkazov **configure, make, checkinstall**.
- Checkinstall vyžiada zadanie podrobností o vytváranom balíku a po vytvorení ho nainštaluje. Takto vytvorený balík potom môžeme odinštalovať štandardným postupom.
- Vytvorený balík samozrejme môžeme použiť na jednoduchú inštaláciu skompilovaného programu aj na inom systéme.

# Inštalácia dodatkového softvéru

- **Inštalácia softvéru a bezpečnosť**
  - Vždy sťahujte softvér iba z dôveryhodných zdrojov.
  - Preferujte distribučné balíky pred balíkmi z iných zdrojov.
  - Ak nedôverujete tvorcovi balíku, nemôžete dôverovať balíku!
  - Ak je to možné, overujte signatúry balíkov a zdrojových kódov.
  - Nikdy neexperimentujte na produkčnom systéme!



# **GNU/LINUX**

## **Správa a údržba systému**

### **Lekcia 7**

#### **Skriptovanie v bashi**

**Zostavil: Bc. Jaroslav IMRICH**

# Skriptovanie v bashi

- Odporúčaná literatúra:  
**Advanced Bash-Scripting Guide**  
<http://www.tldp.org/LDP/abs/html/>
- Shell skript je obdobou “dávkového” súboru. Na jeho začiatku musí byť uvedený interpret skriptu:  
Príklad: *#!/bin/bash*
- Na konci súboru by sa mal nachádzať exit code, ktorý je v prípade úspešného zbehnutia skriptu rovný nule.  
Príklad: *exit 0*



# Skriptovanie v bashi

- Znaky `>` `>>` `<` `|` slúžia na presmerovanie štandardného výstupu:
  - Príklad: `date > ~/info.txt`
  - Príklad: `date >> ~/info.txt`
  - Príklad: `mail jariq@jariq.sk < mail.txt`
  - Príklad: `cat /etc/passwd | grep jariq`
  - Znak `|` sa nazýva rúra (z angl. pipe) a je skratkou pre sériu príkazov:  
`cmd1 > subor; cmd2 < subor; rm subor`

# Skriptovanie v bashi

- ***grep* [parametre] fráza súbor**
  - Zobrazí riadky súboru obsahujúce frázu
  - Príklad: *grep ssh /etc/services*
  - Príklad: *grep -v tcp /etc/services*
- ***sort* [parametre] súbor**
  - Zoradí riadky súboru podľa abecedy
  - Príklad: *sort ~/tel-zoznam.txt*
  - Príklad: *sort -u ~/tel-zoznam.txt*

# Skriptovanie v bashi

- ***sleep* sekundy**

- Počká s vykonávaním skriptu uvedený počet sekúnd
- Príklad: *sleep 10*

- ***test* podmienka**

- Testuje zadanú podmienku a výsledok vráti ako exit code.
- Dostupné podmienky: `man test`
- Príklad: *test -f /bin; echo \$?*

# Skriptovanie v bashi

- ***sed príkaz [súbor]***
  - Sed je neinteraktívny editor schopný vypisovať, nahrádzať a vypúšťať časti súboru.
  - Spolupracuje s regulárnymi výrazmi.
  - Príklad: `sed -e '3,8d' ~/subor.txt`
- ***gawk príkaz [súbor]***
  - GNU implementácia programu awk
  - Podobne ako sed, nie je to jednoduchý príkaz, ale samostatný programový interpret s obrovským množstvom možností.
  - Príklad: `echo "ahoj svet" | gawk '{ print $2 }'`

# Skriptovanie v bashi

- ***expr* výraz**
  - Vyrieši zadaný výraz
  - Príklad: *expr 1 + 2*
- ***break***
  - Ukončí aktuálny cyklus
- ***continue***
  - Skočí na ďalšie pokračovanie cyklu

# Skriptovanie v bashi

- **Vetvenie**

- **if [ test ]; then**  
    prikaz  
**else**  
    prikazy  
**fi**

- **case "\$1" in**  
    'a')  
        prikaz  
        ;;  
    'b')  
        prikaz  
        ;;  
    'c')  
        prikaz  
        ;;  
    \*)  
        default\_prikaz  
**esac**

# Skriptovanie v bashi

- **Cykly**

- OBJEKTY=`ls /var/log`  
**for** i **in** \$OBJEKTY  
**do**  
    echo \$i  
**done**

- var=0  
LIMIT=10  
**while** [ "\$var" -lt "\$LIMIT" ]  
**do**  
    echo \$var  
    var=`expr \$var + 1`  
**done**

- KONIEC=end  
**until** [ "\$var" = "\$KONIEC" ]  
**do**  
    echo "Zadaj retazec: "  
    echo "(\$KONIEC pre koniec)"  
    read var  
    echo "Zadany retazec je \$var"  
    echo  
**done**

# Skriptovanie v bashi

- Na jednotlivé časti spúšťacieho príkazu sa v skripte môžeme odvolávať pomocou premenných \$0 \$1 \$2 atď
- Výstup z ľubovoľného príkazu uložíme do premennej tak, že jej priradíme príkaz uzavretý medzi znaky ` ` (angl. grave accent, sl. opačný dĺžeň).
  - Príklad: *DATUM*=`date`
- Premenná \$RANDOM pri každom použití obsahuje iné náhodné číslo.



# Skriptovanie v bashi

- Správne napísaný program v linuxe vracia tzv. exit code. To isté platí aj pre skripty. Ak nenastane vo vykonávaní programu chyba, je hodnota exit code 0. V prípade chyby, je to iné celé číslo. Hodnota exit code je po vykonaní programu uložená v premennej \$? . Pomocou hodnoty exit code vieme ľahko určiť, či nastala pri vykonávaní programu chyba.
  - Príklad: `ls /var; echo $?`
  - Príklad: `ls /neexistujem; echo $?`

# Skriptovanie v bashi

- Ak je výstup z nejakého príkazu nepodstatný, resp. nežiadúci, môžeme ho presmerovať do špeciálneho zariadenia **/dev/null**. Toto zariadenie môžeme nazvať “čiernou dierou”, pretože čokoľvek, čo do neho presmerujeme, nebude nikam uložené.
  - Príklad: */zaloha.sh > /dev/null*



# **GNU/LINUX**

## **Správa a údržba systému**

### **Lekcia 8**

#### **Sieť**

**Zostavil: Bc. Jaroslav IMRICH**

# Siet'

- ***ifconfig* [rozhranie [parametre]]**
  - Konfiguruje sieťové rozhrania
  - Príklad: *ifconfig*
  - Príklad: *ifconfig eth0*
  - Príklad: *ifconfig eth0 10.1.1.2 netmask 255.255.255.0 up*
  - Príklad: *ifconfig eth0:0 10.1.1.2 netmask 255.255.255.0 up*
  - Príklad: *ifconfig eth1 down*

# Siet'

- ***route ...***

- Nástroj na zobrazenie a editovanie smerovacej tabuľky počítača.
- Príklad: *route*
- Príklad: *route add default gw 10.1.1.1*
- Príklad: *route del default gw 10.1.1.1*
- Príklad: *route add -net 10.1.2.0 netmask 255.255.255.0 dev eth0*
- Príklad: *route del -net 10.1.2.0 netmask 255.255.255.0 dev eth0*

# Sieť

- Klient služby DNS, tzv. resolver, sa konfiguruje v súbore **/etc/resolv.conf**. Pre základné používanie DNS servera stačí v tomto súbore záznam o jeho IP adrese v tvare:

*nameserver 10.1.1.1*

- Počítač je v sieti reprezentovaný IP adresou. Sieť je však reprezentovaná aj maskou podsiete. Online nástroj na vypočítavanie masky je dostupný napr. na adrese *<http://jodies.de/ipcalc>*

# Siet'

- ***mii-tool* [rozhranie [parametre]]**
  - Slúži na kontrolu stavu média (kábla) a prepínanie prenosovej rýchlosti
  - Príklad: *mii-tool*
  - Príklad: *mii-tool eth0 -R*
  - Príklad: *mii-tool eth0 -F 10baseT-FD*
- ***dhcpcd* [parametre] rozhranie**
  - DHCP klient slúžiaci na získanie IP adresy a ďalších údajov z DHCP servera
  - Príklad: *dhcpcd eth0*



# Siet'

- ***ping* [parametre] host**

- Zašle hostu ICMP paket typu ECHO\_REQUEST.
- Vhodný na testovanie dostupnosti hostu.
- Príklad: ping 10.1.1.1

- ***arp* ...**

- Utilita pre správu systémovej arp tabuľky.
- Príklad: arp -a
- Príklad: arp -s 10.1.1.1 11:22:33:AA:BB:CC

# Sieť

- ***netstat* [parametre]**

- Zobrazí aktívne sieťové spojenia, smerovacie tabuľky, štatistiky sieťových rozhraní ...
- Príklad: *netstat --inet -an*
- Príklad: *netstat -r*
- Príklad: *netstat -s*

- ***nslookup* IP/host**

- Preloží IP na hostname a naopak
- Príklad: *nslookup www.iana.org*
- Príklad: *nslookup 192.0.34.162*

# Siet'

- ***whois* IP**

- Zobrazí informácie o organizácii, ktorá spravuje zadanú IP adresu.
- Príklad: *whois 192.0.34.162*

- ***wget* súbor**

- Stiahne súbor prostredníctvom protokolu HTTP, HTTPS alebo FTP.
- Príklad: *wget http://kurz.jariq.sk/linux01.odp*

# Siet'

- ***traceroute* IP/host**
  - Zobrazí cestu (odozvy z brán), ktorou paket prejde k cieľovému hostu.
  - Príklad: *traceroute www.iana.org*
  - Príklad: *traceroute -I www.iana.org*
- ***nmap* [parametre] IP/host/siet'**
  - Zobrazí stav portov na cieľovom hoste
  - Príklad: *nmap 10.1.1.2*
  - Príklad: *nmap -p0 10.1.1.2*

# Siet'

- ***tcpdump* [parametre]**
  - Analyzátor sieťovej prevádzky.
  - Príklad: `tcpdump -i eth0`
- ***ethereal***
  - Analyzátor sieťovej prevádzky pre grafický server X-window.



# **GNU/LINUX**

## **Správa a údržba systému**

### **Lekcia 9**

#### **Sieťové služby**

**Zostavil: Bc. Jaroslav IMRICH**

# Sieťové služby

- Sieťové služby sú základnou výhodou, ktorú poskytuje sieťové prostredie. Obyčajne sú nainštalované na jednom alebo viacerých serveroch. Bežné sieťové služby sú napríklad: ftp, www (http), samba, ssh, dhcp, e-mail (smtp, pop3, imap) a iné.
- S poskytovaním sieťových služieb úzko súvisí aj obmedzovanie prístupu k nim. Niektoré služby na to poskytujú možnosti, iné nie. Odporúča sa zabezpečiť systém globálne pomocou firewallu.
- <http://www.jariq.sk/item-3.html>



# Siet'ové služby

- **FTP (File Transfer Protokol)**

- Protokol FTP slúži na prenos súborov.
- FTP server zvyčajne počúva na porte 21/TCP.
- Prenášané dáta nie sú kryptované.
- Ako démon je možné použiť napríklad *vsftpd* alebo *proftpd*.
- Klient pre textový režim je napríklad *ftp* alebo *mc*, pre grafický režim *gftp* alebo *krusader*.
- <http://www.jariq.sk/item-8.html>

# Siet'ové služby

- **WWW (World Wide Web)**
  - Prenos dát protokolom HTTP (HyperText Transfer Protocol)
  - Web server zvyčajne počúva na porte 80/TCP.
  - Prenášané dáta nie sú kryptované.
  - Najpoužívanejší démon je APACHE (httpd).
  - Klient pre textový režim je napríklad *lynx*, pre grafický režim *firefox*.
  - Konfiguračné súbory démona Apache sa nachádzajú v adresári */etc/apache*.

# Siet'ové služby

- Predvolený adresár, ktorého obsah je publikovaný na webe je */var/www/htdocs*.
- Pre server Apache existuje skriptovací jazyk *PHP* (Hypertext PreProcessor) - [www.php.net](http://www.php.net)
- Server Apache rozšírený o jazyk PHP je schopný spolupracovať s databázovým serverom *MySQL* – [www.mysql.com](http://www.mysql.com)
- Znalosť značkovacieho jazyka HTML a skriptovacieho jazyka PHP sa v dnešnej dobe považuje za základnú zdatnosť človeka vzdelaného v oblasti IT.

# Siet'ové služby

- **Samba**

- Prenos dát protokolom SMB (Server Message Block) využíva napríklad zdieľanie tlačiaraní a súborov v systéme MS Windows.
- Ak chceme z Linuxu pristupovať na systéme Windows zdieľané prostriedky, musíme nainštalovať *Sambu* – [www.samba.org](http://www.samba.org)
- Základný konfiguračný súbor je v */etc/samba/smb.conf*
- Samba využíva porty 139/TCP a 445/TCP.

# Siet'ové služby

- **SSH (Secure Shell)**

- Protokol SSH slúži na vzdialenú správu prevažne unixových systémov.
- SSH server zvyčajne počúva na porte 22/TCP.
- Prenášané dáta sú kryptované asymetrickými šifrovacími algoritmami.
- Klient aj démon sú dostupní napríklad v balíku *OpenSSH* - [www.openssh.org](http://www.openssh.org)
- Známy klient pre grafický režim, ale aj pre systém MS Windows je *putty*.

# Siet'ové služby

- **DHCP (Dynamic Host Configuration Protocol)**
  - Protokol DHCP sa využíva na vzdialenú konfiguráciu sieťových parametrov klienta.
  - DHCP server počúva na porte 67/TCP.
  - Je ním možné priradiť napr. doménové meno, IP adresu, masku siete, bránu a DNS servery.
  - Démon pre systém Linux je *dhcpd*. Ako klienta možno použiť napríklad *dhcpcd*.
  - Konfiguračný súbor démona *dhcpd* je `/etc/dhcpd.conf`

# Siet'ové služby

- **E-mail (elektronická pošta)**

- Pri používaní e-mailu sa využíva viacero protokolov.
- Protokol **smtp** (Simple Mail Transfer Protocol) sa používa na odosielanie pošty. Známi SMTP démoni sú napr. *sendmail*, *qmail* a *postfix*.
- Na prijímanie pošty využíva klient protokol **pop3** (Post Office Protocol) alebo **imap** (Internet Message Access Protocol). Rozdiel je v tom, že cez pop3 sa ihneď všetky správy stiahnu ku klientovi a imap umožňuje vybrať, ktoré správy sa zo servera stiahnu.

# Siet'ové služby

- **DNS (Domain Name System)**

- Služba DNS slúži na preklad doménových názvov na IP adresy a naopak.
- DNS server používa porty 53/TCP a 53/UDP.
- Najpoužívanejší démon je *bind*.
- Bind môže bežať napríklad ako forwarding server, ale môže byť aj master server nejakej domény.
- Doména .sk je spravovaná firmou Euroweb Slovakia, a.s a záznamy o .sk doménach je možné prehliadať na [www.sk-nic.sk](http://www.sk-nic.sk)





# **GNU/LINUX**

## **Správa a údržba systému**

### **Lekcia 10**

#### **Kompilácia jadra**

**Zostavil: Bc. Jaroslav IMRICH**

# Kompilácia jadra

- Kernel je jadro operačného systému. Jednou z jeho hlavných úloh, je poskytovať cez jednotné rozhranie bezpečný prístup k hardvéru počítača iným procesom (aplikačnému softvéru).
- Jadro Linuxu je v súčasnej dobe vyvíjané v dvoch hlavných vetvách 2.4 a 2.6
- Zdrojové kódy jadra Linuxu sú dostupné na [www.kernel.org](http://www.kernel.org)

# Kompilácia jadra

- Najčastejšia príčina rekompilácie jadra je pridávanie ovládačov pre nový hardvér.
- Ovládače jednotlivých zariadení môžu byť v jadre “zakompilované” na pevno, alebo môžu byť od neho oddelené – moduly.
- Výhoda modulov je, že ak nepoužívame dané zariadenie, môžeme ich z jadra odobrať a tak uvoľniť operačnú pamäť.
- Obrazy jadier sa nachádzajú v adresári */boot* a moduly v */lib/modules*

# Kompilácia jadra

- ***lsmod***
  - Zobrazí stav modulov zavedených v jadre
  - Príklad: *lsmod*
- ***modprobe* [parametre] modul**
  - Pokúsi sa zaviesť modul do jadra
  - Príklad: *modprobe 3c59x*
- ***rmmmod* [parametre] modul**
  - Odoberie modul z jadra
  - Príklad: *rmmmod 3c59x*

# Kompilácia jadra

- Po stiahnutí a rozbalení zdrojových kódov jadra je nutné previesť nasledovnú sériu príkazov (kompiláciu):
  - make mrproper
  - make menuconfig
  - make bzImage
  - make modules
  - make modules\_install

# Kompilácia jadra

- ***make mrproper***
  - Zabezpečí odstránenie nepotrebných súborov po predchádzajúcej kompilácii.
- ***make menuconfig***
  - Spustí konfiguračný program založený na ncurses knižniciach (podobný mc), v ktorom sa prevádza výber súčastí jadra. Alternatívou pre grafické prostredie je *make xconfig*.
- ***make bzImage***
  - Skompiluje a vytvorí obraz jadra.

# Kompilácia jadra

- ***make modules***
  - Skompiluje súčasti jadra, ktoré boli vybraté ako moduly.
- ***make modules\_install***
  - Prekopíruje skompilované moduly do */lib/modules*



# Kompilácia jadra

- Po skompilovaní je obraz jadra v jedinom súbore *arch/i386/boot/bzImage* a je nutné prekopírovať ho do adresára */boot*.
- Výber jednotlivých komponentov jadra je zaznamenaný v súbore *.config*, ktorý je vhodné držať v adresári */boot* spolu s obrazom jadra.
- Posledný krok je pridanie záznamu o novom jadre do boot loaderu a reštart systému.

# Kompilácia jadra

- Ďalšie zdroje informácií:
  - <http://www.digitalhermit.com/linux/Kernel-Build-HOWTO.html>
  - [http://media.sh.cvut.cz/avc/AVC\\_projekty/2005\\_04\\_21\\_AVC\\_ruik-Kompilace\\_jadra\\_2.6.12.avi](http://media.sh.cvut.cz/avc/AVC_projekty/2005_04_21_AVC_ruik-Kompilace_jadra_2.6.12.avi)



# **GNU/LINUX**

## **Správa a údržba systému**

### **Lekcia 11**

Úvod do bezpečnosti

**Zostavil: Bc. Jaroslav IMRICH**

# Úvod do bezpečnosti

- **Bezpečnosť**

- **FYZICKÁ**

- (zamknuté dvere, alarm, kamerový systém, zamknutá skriňa, šifrovaný súborový systém ...)

- **SYSTÉMOVÁ**

- (dôveryhodný softvér, silné heslá, správne nastavené prístupové práva ...)

- **SIEŤOVÁ**

- (firewall, DMZ, bezpečnostné perimetre, tunely, proxy, nekontrolované prístupové body, WIFI, podvrhnutie IP adresy, ARP spoofing, syn flooding, VPN, šifrované prenosy dát ... )

# Úvod do bezpečnosti

- **SEDEM SMRTEL'NÝCH HRIECHOV**

1. Slabé heslá

2. Otvorené sieťové porty

3. Staré verzie softvéru

4. Nebezpečné a chybné nakonfigurované programy

5. Nedostatočné prostriedky a chybné stanovené priority

6. Staré a nepotrebné používateľské účty

7. Čo môžeš odložiť..

# Úvod do bezpečnosti

- Teória bezpečnostných bariér
- Neverit' nikomu !
- MONITOROVAŤ !
  - Logy (logcheck ...)
  - Traffic (snmp, mrtg, bandwidthd ...)
  - IDS (snort ...)
  - otvorené porty (nmap, nessus ...)
  - integrita súborov (tripwire ...)

# Úvod do bezpečnosti

- Definovanie a striktné dodržiavanie bezpečnostnej politiky.
- Bezpečnostný audit





# GNU/Linux – Správa a údržba systému

Táto práca podlieha licencií

**Creative Commons Attribution-ShareAlike 2.5 License**

Pre zobrazenie textu tejto licencie navštívte

**<http://creativecommons.org/licenses/by-sa/2.5/>**

alebo si ju vyžiadajte zaslaním listu na adresu

**Creative Commons  
543 Howard Street, 5th Floor  
San Francisco  
California, 94105  
USA**



<http://www.this-page-intentionally-left-blank.org/>