Administrace Unixu a sítí

```
inet6-adr: fe80::210:a4ff:fee1:9e5d/64 Rozsah:Linka
AKTIVOVÁNO VŠESMĚROVÉ_VYSÍLÁNÍ BĚŽÍ MULTICAST  MTU:1500  Metrika:1
RX packets:66690 errors:0 dropped:0 overruns:0 frame:0
TX packets:100149 errors:0 dropped:0 overruns:0 carrier:0
kolizí:0 délka odchozí fronty:0
RX bytes:21490419 (20.4 MiB)  TX bytes:10545763 (10.0 MiB)
```

5. Logování, instalace SW

Miroslav Spousta

2006

```
bug:/home/qiq# getent passwd|grep :1
uucp:x:10:10:uucp:/var/spool/uucp:/bin/sh
Debian-exim:x:102:102::/var/spool/exim4:/bin/false
qiq:x:1000:1000:Miroslav Spousta,,,:/home/qiq:/bin/bash
sshd:x:100:65534::/var/run/sshd:/bin/false
identd:x:101:65534::/var/run/identd:/bin/false
messagebus:x:103:104::/var/run/dbus:/bin/false
gdm:x:104:105:Gnome Display Manager:/var/lib/gdm:/bin/false
hal:x:106:106:Hardware abstraction layer,,,:/var/run/hal:/bin/false
saned:x:109:109::/home/saned:/bin/false
bind:x:105:110::/var/cache/bind:/bin/false
smmta:x:107:111:Mail Transfer Agent,,,:/var/lib/sendmail:/bin/false
smmsp:x:108:112:Mail Submission Program,,,:/var/lib/sendmail:/bin/false
test:x:1001:1001:Test User,,,:/home/test:/bin/bash
postfix:x:110:115::/var/spool/postfix:/bin/false
```

http://www.ucw.cz/~qiq/vsfs/

Xen

- virtuální servery (http://www.cl.cam.ac.uk/Research/SRG/netos/xen/)
- budete mít rootovská oprávnění ve virtuálním serveru
- přístup je po dobu výuky
- OS: Debian 3.1
- RAM: 32 MB, swap: 128 MB (/dev/sda2), root: 512 MB (/dev/sda1)
- několik síťových karet (eth0, eth1, ...)
- Síťové karty jsou propojeny virtuálními přepínači

Přístup k virtuálním serveru:

- **`ssh login@kozel.vsfs.cz`**
- **`xencons localhost 90xx`**
- xx je číslo serveru (login: **`root`**, heslo: žádné)

Log soubory

- významné události v systému je dobré ukládat do souboru, aby bylo možné později analyzovat, co se stalo
  - selhání HW, SW, (pokus o) průnik do systému
- zapisují se především údaje z daemonů, jádra, …
- ukládají se většinou do souborů v adresáři /var/log/
  - případně /var/adm/ (např. SunOS)
  - někdy bývají vytvořeny podadresáře podle jednotlivých služeb (např. /var/log/httpd/)
- log soubory jsou dvou druhů: textové a binární
- binární: wtmp, (btmp), utmp, lastlog
  - je potřeba je číst pomocí speciálního programu (last(1), lastb(1))
  - slouží k získání informací o přihlašování uživatelů do systému
- textové: ostatní (např. messages, syslog, mail.log, kern.log, access.log, …)
  - je možné je číst v libovolném editoru

Běžné logy

- messages/syslog – hlavní systémový log (= co nepatří jinam přijde sem)
- auth.log, faillog – o (ne)proběhlé autentizaci
- wtmp (utmp): historie přihlašování (binární)
  - lastlog: pro každého uživatel čas posledního přihlášení
  - btmp: jako wtmp, ale ukládají se tam neplatné pokusy o přihlášení
- boot.log: výstup z rc skriptů
- kern.log: záznamy z jádra systému
- apache/{access,error}.log: logy web serveru, zobrazené stránky/neplatné požadavky
- mail.log: záznamy mailového subsystému (mail server, POP3 a IMAP server, spam filter, ...)
- Xorg.Y.log: log soubor od X Window systém (Y je číslo terminálu)
- mysqld.log: MySQL server

Konfigurace logování

- často je možné v programu nastavit, jak se bude logovat, zda přímo do souboru (např. Apache), nebo přes syslog(3) interface a syslogd(8)
- syslogd umožňuje velkou flexibilitu v nastavení, kam se budou zprávy ukládat, konfigurace je uložena v /etc/syslog.conf
  - řádkově orientovaný, každý řádek popisuje jednu akci, skládá se ze selektoru a akce
- selektor: filtr na zprávy, které se budou zpracovávat v akci
  - služba.úroveň;služba.úroveň;...
- akce: co se bude dít se zprávami, které projdou filtrem
  - může tam být cesta k souboru: /path/to/file, roura: |, vzdálený server: @, uživatel: login, všichni uživatelé: *
  - např.: *.crit;kern.none /var/adm/critical
- klogd sbírá informace z kernelového bufferu (to, co vypíše dmesg) a přeposílá syslogu (případně kopíruje na systémovou konzoli)
  - úroveň logování na konzoli je možné nastavit přepínačem -c n (n = úroveň)
  - syslogu přeposílá jako od služby „kern“

Služby a úrovně logování

- informace, které se zapisují do logu mohou pocházet z různých zdrojů a mohou mít různou důležitost => služby a úrovně
- při logování se rozlišují různé služby (zdroje informace):
  - AUTHPRIV: autentizace, nemělo by být veřejně přístupné
  - KERN: jádro, LOCAL0-7: pro lokální použití
  - LPR, MAIL, UUCP, NEWS, FTP, CRON: pro daemona dané služby
  - DAEMON: ostatní daemoni, USER: default, když není kam jinam logovat
- několik různých úrovní logování (jak moc je informace důležitá):
  - používá se pro filtrování událostí (vždy se uvažuje daná úroveň a všechny vyšší)
  - EMERG: systém je nepoužitelný (kritická chyba)
  - ALERT: očekává se okamžitá reakce
  - CRIT, ERR: chybový stav
  - WARNING, NOTICE, INFO: informace
  - DEBUG: debugování (= chci vidět úplně všechno)

syslog.conf

```
# Logujeme všechno (kromě e-mailu) úrovně info nebo vyšší,
# nelogujeme autentizační informace
*.info;mail.none;authpriv.none;cron.none      /var/log/messages

# Zapisujeme zvlášť do souboru, který je čitelný jen pro roota
authpriv.*                                    /var/log/secure

# e-mail, vše od úrovně info výše
# '-' = neprovádět sync po každé zprávě (zrychluje)
mail.info                                     -/var/log/maillog

# e-mail, jen úroveň info
mail.=info                                    /var/log/mail.info

# e-mail, vše kromě úrovně info
mail.*;mail.!=info                            /var/log/mail.other

# Všichno uživatelé dostanou emergency zprávy
*.emerg                                       *
```

Rotace logů

- logovací soubory nemají omezenou maximální velikost
- neustále by rostly, ale velmi staré záznamy nás moc nezajímají
- řešení: čas od času vytvoříme nový logovací soubor a starý přejmenujeme (= rotace)
  - rotaci provádíme po uplynutí určitého času od poslední rotace daného souboru, po dosažení určité velikosti souboru, …
  - po rotaci je potřeba programu oznámit, že k rotaci došlo, aby znovu otevřel soubor
- uchováváme několik starších souborů
- starší verze se obvykle komprimují pro úsporu místa
- např.: maillog, maillog.1, maillog.2 s týdenní periodou rotace bude uchovávat záznamy dva týdny nazpět
- program pro rotování logů: logrotate(8)
- konfigurace: /etc/logrotate.conf
  - pro každý soubor je zde možné nastavit, jak často se rotuje, kolik se ukládá souborů, co se provede před rotací a po ní

Rotace: Apache

- příklad konfigurace rotace logu Apache web serveru
- jednou týdně, schovávat rok nazpět, po rotaci restartovat Apache

```
/var/log/apache3/*.log {
        weekly
        missingok
        rotate 52
        compress
        delaycompress
        notifempty
        create 640 root adm
        sharedscripts
        postrotate
                if [ -f /var/run/apache2.pid ]; then
                        /etc/init.d/apache2 restart > /dev/null
                fi
        endscript
}
```

Instalace SW

- pod Linux a UNIX spadají velmi rozmanité systémy, liší se v mnoha drobnostech i důležitých parametrech
  - adresářová struktura, jména konfiguračních souborů, nainstalované knihovny
  - nejroztodivnější architektury, HW
- velké množství SW je distribuování ve zdrojové podobě
  - balíky se zdrojovými soubory (tar.gz – tarball)
  - pro použití je potřeba je přeložit pro danou architekturu (a systém)
  - vyžaduje kompilátor pro použité jazyky (pro C/C++ často gcc) a program make
  - často je přenositelnost ošetřena pomocí autotools: autoconf/automake/libtool
- případně může být balík předkompilovaný a připravený k instalaci
  - používá se mnoho různých systémů (rpm, deb, pkg, ...)
- některý SW dodáván jen v binární podobě (většinou proprietární)
  - pak je většinou určen pro konkrétní kombinaci OS/architektura (např. Fedora Core 3/x86_64)

Instalace ze zdrojových kódů

- klasický „unixový“ způsob instalace:
  - **`tar -xvf install-tarball.tar.gz; make; make install`**
- překlad se řídí souborem Makefile, proměnné je možné předefinovat na příkazové řádce: **`make CFLAGS=-ggdb`**
- pro instalaci je většinou potřeba práv roota (pokud se instaluje do systémových adresářů)
- kam se instaluje bývá možné nastavit pomocí proměnné PREFIX
- bývá zvykem instalovat software do /usr/local, nebo do /opt
- může být vhodné instalovat do zvláštního podadresáře pro daný program
  - zjednodušuje odinstalování/upgrade
  - je ovšem potřeba upravit patřičně proměnnou PATH
  - a případně také cestu ke sdíleným knihovnám (/etc/ld.so.conf, SunOS: crle)
  - např. MySQL server nainstalujeme do /opt/mysql
  - pak je v daném podadresáři vytvořen strom /bin, /lib, /man, ...

Instalace s pomocí GNU Autotools

- pro snazší přenositelnost se používají nástroje z balíku GNU autotools
  - automake: slouží pro generování souborů Makefile (z Makefile.am)
  - autoconf: ze souboru configure.in (.ac) vytvoří spustitelný skript configure, který zkontroluje všechny závislosti, zjistí parametry daného OS a podle toho připraví kompilaci (např. parametry pro kompilátor, verze knihoven, cesty ke knihovnám, atd)
  - libtool: umožňuje vytvářet (sdílené) knihovny přenositelným způsobem
- práce je podobná jako při klasické instalaci ze zdrojových kódů, ale před kompilací je potřeba spustit skript configure (součástí balíku)
- **`./configure --prefix=/usr/local; make; make install`**
- configure
  - často se používá přepínač --prefix, který umožňuje instalovat do libovolného adresáře
  - implicitní instalační adresář většinou bývá /usr/local
  - pomocí přepínače --help je možné získat přehled možných parametrů
  - např. --enable-shared: povolí vytváření sdílených knihoven

Instalace binárních aplikací

- proprietární aplikace (např. Acrobat Reader)
  - ale např. i předkompilované balíky OSS projektů (MySQL)
- instalace většinou spočívá v „rozbalení“ archivu (unzip, tar -xz, tar -xjf)
  - často do adresáře /opt
  - platí stejná pravidla pro PATH a ld.so.conf, jako pro instalaci ze zdrojových kódů
- aby nebylo nutné každý SW zvlášť kompilovat, distribuce obsahují binární balíky, které tvoří zkompilované programy
  - jsou připravené pro instalaci (instalace je rychlá)
  - umožňují vytváření a kontrolu závislostí mezi balíky (např. perlová knihovna bude vyžadovat instalaci Perlu)
  - udržují databázi instalovaných balíků a souborů – lze snadno balík odinstalovat, případně upgradovat na novější verzi
  - různé systémy, ale podobné možnosti a ovládání: dpkg (Debian, Ubuntu), rpm (RedHat, Fedora, SUSE), ebuild (Gentoo), Solaris: pkg, BSD: ports

Použití balíčkovacích systémů

- instalace balíku:
  - dpkg -i balik.deb, rpm -ivh balik.rpm, pkgadd -d balik.pkg
- odinstalování balíku:
  - dpkg -P balik, rpm -e balik, pkgrmbalik
- výpis instalovaných balíků:
  - dpkg -l, rpm -qa, pkglist
- výpis obsahu (souborů a adresářů) nainstalovaného balíku
  - dpkg -L balik, rpm -ql balik
- nalezení, do kterého balíku patří daný soubor:
  - dpkg -S /path/to/file, rpm -qf /path/to/file
  - nalezení balíku, který chci instalovat: debian.org, rpmfind.net, …
- instalace balíků i se závislostmi, upgrade systému:
  - apt-get install balik, yum install balik
  - apt-get update; apt-get upgrade, yum upgrade