Duben 1998

Linuxové noviny

Úvodem

Pavel Janík ml., 11. dubna 1998

Tématem dubnového čísla Linuxových novin jsou databázové systémy. Dan Ohnesorg, Nathan L. Cutler a Michal Polák nás ve svém článku SQL server pod Linuxem?! A kde jinde!? seznámí se základními možnostmi pro výběr SQL serveru pro operační systém Linux.

Nakladatelství Neokortex vydalo dvě nové knihy, které jsou tematicky blízké Linuxu. První je věnována serveru Apache a PHP/FI. Recenzi této knihy si můžete přečíst v článku Apache a PHP/FI – recenze knihy, jehož autorem je Milan Šorm. Druhá kniha je druhým vydáním úspěšné knihy *Linux–Internet server* renomovaných autorů Pavla Satrapy a Jiřího A. Randuse. Recenzi této knihy uvedeme v příštím čísle, ale již v tomto čísle budete moci danou knihu vyhrát v soutěži, kterou společně s vydavatelstvím Neokortex pořádá redakce Linuxových novin. Více se můžete dočíst v článku Nejmenší vyhrává.

Milan Zamazal zakončí třetím dílem seriál o dokumentačních možnostech editoru Emacs. Možná nejzajímavějším článkem pro ty čtenáře, kteří neměli tu možnost se zúčastnit Linuxového semináře na Cikháji, bude nestranný pohled Petra Olšáka na celou akci podaný v článku Linuxový seminář na Cikháji z mého pohledu. Linuxový seminář pořádalo České sdružení uživatelů operačního systému Linux.

Dubnové číslo Linuxových novin přináší také něco pro programátory – Karel Žák nás v článku Programujeme v C s libproc seznámí s knihovnou libproc, která dokáže mnohé usnadnit a zpřehlednit.

Jako již tradičně nechybí pravidelné rubriky Měsíc v comp.os.linux.announce, Co nového na sunsite.unc.edu?, Linux Journal a Zasmáli jsme se!.

Přeji příjemné chvíle strávené nad četbou Linuxových novin a těším se na vaše připomínky. ■

Nejmenší vyhrává

Pavel Janík ml., 10. dubna 1998

A už je to tady. Linuxové noviny vypisují soutěž. Jakou? Samozřejmě linuxovou. Jaká jsou pravidla? Jednoduchá. Kdo vyhrává? Ten nejlepší.

Redakce Linuxových novin a vydavatelství Neokortex, spol. s r. o., vyhlašují soutěž o nejkratší spustitelný program pro operační systém Linux/i386/Glibc2. Program musí být dodán tak, aby byl přeložitelný na počítači s nejnovější distribucí Linuxu, posledním vývojovým jádrem a s `gcc 2.7.2.3` (zkrátka, abych ho dokázal přeložit). Program by měl obsahovat stručný komentář své činnosti (ano, program by měl něco rozumného dělat), měl by být přiložen `Makefile`. Přijímáme pouze programy pod licencí GPL. Další podmínkou pro přijetí programu do soutěže je to, že program musí být identifikován standardní utilitkou `file` jako *executable*, tedy např.:

```
SnowWhite:/tmp$ file muj_program
muj_program: ELF 32-bit LSB executable, \
   Intel 80386, version 1, ...
SnowWhite:/tmp$
```

Je zcela zřejmé, že vítězem soutěže se stane někdo, jehož program bude naprosto geniální, ale svůj účel dobře plnící, „hack“.

Své soutěžní příspěvky posílejte na adresu (1) se subjectem „LN: Soutez“ (bez uvozovek). Uzávěrka soutěže je 8. května tohoto roku a výsledky budou oznámeny v květnovém čísle Linuxových novin.

Jak a kdo bude hodnotit výsledky? Výsledky se budou hodnotit objektivně – každý program, který bude vyhovovat podmínkám soutěže bude vyzkoušen a pokud opravdu bude dělat to, co má a neskončí podobně jako program na následujícím výpise, bude zařazen do soutěže.

```
SnowWhite:/home/pavel$ netscape
Segmentation fault (core dumped)
SnowWhite:/home/pavel$
```

Vítězem se stane majitel programu, jehož délka (`wc --bytes muj_program`) bude nejmenší. V případě shodné délky programů rozhoduje čas doručení zdrojové podoby. (Pokud budete chtít, můžete i tipovat délku vítězného programu – třeba se najde ještě nějaká vhodná cena pro vítěze tipovací soutěže…).

A nyní to nejpodstatnější – cenou pro vítěze bude druhé vydání knihy LINUX–Internet server autorů Pavla Satrapy a Jiřího A. Randuse. Knihu vydal Neokortex, spol. s.o. (2). ■

1 Redakce Linuxových novin
`mailto:noviny@linux.cz`
2 Neokortex, spol. s r. o.
`http://www.neo.cz`

Měsíc v comp.os.linux.announce

Pavel Janík ml., 1. dubna 1998

XMRM tým, který se skládá ze dvou studentů počítačové grafiky a jejich vyučujícího, oznámil alpha verzi programu XMRM (1). XMRM je program, který umí *morphing*. Ptáte se, co to morphing je? Zkuste se podívat na manuál (2) k tomuto programu a pochopíte.

Kent Robotti (robotti@erols.com) vytvořil další mini distribuci Linuxu založenou na UMSDOS filesystému. Jmenuje se DosLinux (3) a k její instalaci potřebujete nejméně 36MB na vašem disku s operačním systémem DOS nebo Microsoft Windows. Distribuce ovšem neobsahuje X server ani překladač.

Thomas J. Nelson (tjnelson@las1.ninds.nih.gov) ozná-

Linuxové noviny připravuje a vydává České sdružení uživatelů operačního systému Linux

mil novou verzi svého programu `tnimage`, který umožňuje spoustu operací s obrázky. Podporuje nejméně 20 grafických formátů, umožňuje přidávat další formou pluginů a navíc umožňuje interaktivní definici nového formátu. Program je doplněn manuálem o 137 stranách. Nevýhodou tohoto programu je, že ke své činnosti potřebuje knihovnu Motif, ale autor poskytl i staticky linkované binárky. Vše naleznete na adrese (4).

Knut Bartels (bartels@rz.uni-potsdam.de) oznámil vytvoření programu pro statistické výpočty XploRe (5). Program může být užitečný pro ty z nás, kteří mají problémy s odchylkami, rozptyly, kovariancemi, marginálními a simultánními četnostmi apod.

Máte problémy s klávesami `BackSpace` či `Delete` a nevíte, jak je správně nakonfigurovat? Tak právě pro vás je určena stránka „BackSpace and Delete Configuration" (6).

Společnost Hewlett-Packard GmbH, Germany přenesla svůj systém HP Eloquence (7) na Linux. K dispozici je i Personal edition, která je omezena pouze na dva uživatele. Společnost Hewlett-Packard je první z „velkých a mocných" počítačových firem, která začíná portovat své produkty pro Linux. Doufejme, že přibudou i další. Ale na druhou stranu, doufejme, že ne všechny...

Werner Almesberger (almesber@lrc.epfl.ch) vytvořil novou verzi balíku psmisc, který obsahuje utilitky `fuser`, `killall`, `pstree` a `pidof`. Balík naleznete na adrese (8).

3. března byla oznámena nová verze editoru XEmacs – 20.4. Najdete ji na FTP serveru (9).

Peter Toft (pto@sslug.imm.dtu.dk) vytvořil program `cvs2html`, který využijí všichni uživatelé systému CVS (*Concurrent Versions System*). Program je distribuován pod licencí GPL na adrese (10).

Derek B. Noonburg (derekn@aimnet.com) vytvořil další verzi svého prohlížeče formátu PDF – `xpdf 0.7a`, kterou naleznete na adrese (11).

Kevin Fenzi (kevin@scrye.com) vytvořil dokument nazvaný Linux Security HOWTO. Dokument by se měl zabývat bezpečnostními aspekty operačního systému Linux. Nejnovější verze najdete na adrese (12) a na českých mirrorech (13), (14).

Florian La Roche (florian@suse.de) oznámil update knihovny `ncurses` na verzi 4.2. Knihovna obsahuje celkem 257 dokumentovaných volání totožných s voláními ze SVr4 `curses`. Novou verzi naleznete na adrese (15).

Vývojové prostředí pro X-Window s názvem `xwpe` má svůj mailing list. Více informací naleznete na adrese (16).

Alain Knaff (knaff@tux.org) vytvořil novou verzi pravděpodobně nejznámějších utilitek pro práci s disketami – `mtools`. Balík naleznete na adrese (17).

Společnost IBM uvolnila svůj jazyk Object REXX i pro Linux. Podrobnosti naleznete na stránce (18).

Pamatujete staré dobré časy, kdy se váš počítač hlásil nikoli výzvou `login:` , ale `READY.`? Tak pro vás je určen *The Versatile Commodore Emulator* (19), emulátor počítačů Commodore 64 a podobných příšer založených na procesorech 6502 a 6510.

Linux-on-SimICS – takhle se jmenuje projekt ohlášený v konferenci *comp.os.linux.announce* Peterem S. Magnussonem (psm@sics.se). Tento projekt umožňuje simulovat instrukční sadu podle Sparc V8 Book. Jeho autoři dokázali rozběhnout nemodifikovanou verzi Red Hat Linux 4.2 na virtuální stanici (sun4m). Další informace o tomto zajímavém projektu naleznete na stránce (20). ■

1 XMRM homepage
`http://www.cg.tuwien.ac.at/research/ca/mrm/`
2 manuál k programu XMRM
`http://www.cg.tuwien.ac.at/~xmrm/`
3 Distribuce DosLinux
`http://www.tux.org/pub/people/kent-robotti/`
4 Tnimage
`ftp://las1.ninds.nih.gov/pub/unix/`
5 XploRe
`http://www.xplore-stat.de`
6 BackSpace and Delete Configuration
`http://www.ibbnet.nl/~anne/keyboard.html`
7 HP Eloquence
`http://www.hp-eloquence.com`
8 Psmisc
`ftp://lrcftp.epfl.ch/pub/linux/local/psmisc/`
9 XEmacs 20.4
`ftp://ftp.xemacs.org/pub/xemacs/xemacs-20.4/`
10 cvs2html
`http://sslug.imm.dtu.dk/cvs2html/`
11 Xpdf 0.7a
`http://www.aimnet.com/~derekn/xpdf/`
12 Linux Security HOWTO
`http://sunsite.unc.edu/mdw/HOWTO/Security-HOWTO.html`
13 Linux Security HOWTO na síti TEN34-CZ
`http://www.linux.cz/linuxdoc/HOWTO/Security-HOWTO.html`
14 Linux Security HOWTO na síti IGN
`http://www.inet.cz/linuxdoc/HOWTO/Security-HOWTO.html`
15 Ncurses 4.2
`ftp://prep.ai.mit.edu/pub/gnu`
16 Xwpe
`http://www.rpi.edu/~payned/xwpe/alpha.html`
17 Mtools 3.9
`http://linux.wauug.org/pub/knaff/mtools`
18 Object REXX
`http://service2.boulder.ibm.com/dl/rexx/orexxlinux-d/`
19 VICE 0.14.0
`ftp://ftp.funet.fi/pub/cbm/crossplatform/emulators/VICE`
20 Linux-on-SimICS
`http://www.sics.se/simics/`

Co nového na sunsite.unc.edu?

Pavel Janík ml., 6. dubna 1998

X11

X11/fonts/Xfstt-0.9.8.tgz – X Font server pro fonty ve formátu `*.ttf`

X11/screensavers/xlockmore-4.09.tgz – screen saver pro X-Window

X11/xutils/xruskb-1.5.2.tar.gz – umožňuje přepínat dvě klávesnice

apps

apps/database/postgres/sqlforms-0.6.1.tar.gz – vizuální editor formulářů pro PostgreSQL

apps/database/postgresSQL/mpsql-1.5.3.tgz – interaktivní prostředek pro dotazy na databázový systém Postgres95 *apps/doctools/doxygen-0.3.src.tar.gz* – dokumentační systém pro C++

apps/editors/zed-1.0.3.tar.gz – výkonný a konfigurovatelný textový editor

apps/financial/accounting/moneydance-0.9.tgz – manažer peněženky napsaný v Javě

apps/graphics/capture/cutil-0.3.tar.gz – podpora digitální kamery Olympus C-400 a vyšší

apps/graphics/draw/XPhotoEditor-0.1.20.tar.gz – editor obrázků ve formátu JPEG

apps/graphics/viewers/X/xpdf-0.7a.tar.gz – prohlížeč souborů ve formátu PDF

apps/math/calc/hp67-1.4.tar.gz – programovatelný vědecký kalkulátor podobný HP-67

apps/math/fractals/XaoS-3.0.tar.gz – nová verze starého známého prohlížeče fraktálů

apps/misc/dawn-linux-4.0b1.tar.gz – konvertor e-mailových adresářů pro různé aplikace

apps/serialcomm/machines/newtonlink-1.27.tar.gz – přenos dat mezi Apple Newton PDA a Linuxem

apps/sound/editors/mp3tag0.2b.src.tgz – MP3 tag editor

apps/wp/maxwell-0.5.2.tar.gz – WYSIWYG word processor s podporou RTF

apps/www/converters/wmf-1.0.2.tar.gz – konvertor mailů do HTML

apps/www/servers/WWWserver-0.1.tar.gz – HTTP server v Tcl

apps/www/servers/wwwoffle-2.1.tgz – jednoduchý proxy server

devel

devel/lang/fortran/g77-0.5.22.bin.tgz – překladač GNU Fortran

devel/sscript-1.8.tar – skriptový jazyk pro vytváření síťových aplikací

games

games/arcade/koules/koules1.4-src.tar.gz – koules

games/strategy/phalanx-13.tar.gz – nová verze šachového programu

hardware

hardware/cmostool-1.0.tar.gz – editor údajů v paměti CMOS

kernel

kernel/iso9660-compress-2.0.tar.gz – transparentní dekomprese dat na filesystému ISO-9660

libs

libs/cgilib-0.4.tar.gz – jednoduché API pro Common Gateway Interface

logos

logos/penguins/linux_logo-2.00.tar.gz – ANSI Penguin Logo

system

system/boot/loaders/chos-0.82.tar.gz – zavaděč založený na systému menu

system/emulators/commodore/vice-0.14.1.tar.gz – emulátor Commodore C64/C128/PET/VIC20

system/emulators/cpm-0.2.1.tar.gz – emulátor Z80 a CP/M

system/filesystems/ovlfs1.0.src.tgz – podpora pro overlay filesystém

system/network/admin/firewallct-1.0.5b.tar.gz – konfigurace firewallu poněkud příjemněji

utils

utils/disk-management/mtools-3.9.tar.gz – práce s disketami s DOSovým filesystémem

utils/endian.taz – detekce big- nebo little-endian

■

Linux Journal

Petr Bárta, 10. dubna 1998

Linux Journal se ve svém březnovém čísle zaměřil na grafická uživatelská rozhraní (GUI, *Graphical User Interfaces*). V prvním z článků věnovaných tomuto tématu nás Otto Hammersmith seznamuje s GUI knihovnou `gtk+` – *GIMP Tool Kit*.

John Blair seznamuje čtenáře se svým pohledem na *TkDesk*, jeden z desktop managerů pro Linux.

Chcete pro programování GUI aplikací používat XView toolkit? Michael Hall ve svém článku probírá příklad takového použití.

S *CDE* (*Common Desktop Enviroment*) se můžete seznámit prostřednictvím článku Gerge Krafta.

Pokud se zajímáte o AfterStep, bude vás asi zajímat článek Guylhema Aznara, ve kterém se věnuje historii, filosofii, ale hlavně budoucímu vývoji tohoto window manageru.

Zápasíte-li (jako já) s programem GhostScript, možná vám pomůže Robert Kiesling ve svém článku ze série „Take Command".

Bob Hepple měl potíže se spravováním rozsáhlého souboru HTML dokumentů. Ke zjednodušení práce použil preprocesor `m4` a napsal o tom zprávu pro ty, kteří by se chtěli vydat v jeho stopách.

Druhý díl článku pánů Griffina a Nelsona o síťovém programování je zaměřen na tvorbu démonů.

Pokud jste někdy zkoušeli použít některý scanner pod Linuxem, možná oceníte článek Davida Mosbergera o *SANE* (*Scanner Access Now Easy*) – univerzálním rozhraní pro zařízení produkující rastrované obrázky.

Timotej Ecimovic seznamuje s GPIB (*General Purpose Interface Bus*) a s možností jejího využití v Linuxu.

Nevím, kolik z vás má k dispozici digitální fotoaparát Casio QV-10. Pokud ale přece jen někdo ano, pak vězte, že pro něj existuje software pro Linux, který napsal Bob Hepple.

Nevíte-li, jak nastavit `procmail` tak, aby odfiltroval spamy došlé do mailboxu, přečtěte si článek Brandona M. Browninga.

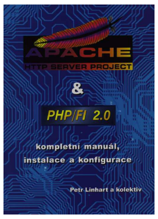
X-Designer, komerční software pro tvorbu GUI, představuje Timotej Ecimovic.

Pokud máte stále problémy se zaplňováním adresáře `/var/spool/mail/`, možná vám pomůže Michael S. Keller se svými skripty na mazání starých zpráv z mailboxů, které jejich vlastníci neprohlížejí.

Síťování přes paralelní port, PLIP protokol apod. popisuje Alessandro Rubini v dalším článku ze seriálu „Kernel Korner“. ■

Apache a PHP/FI – recenze knihy

Milan Šorm, 13. dubna 1998



Asi 320 stránková modrá kniha formátu A5 má jednu z těch lepších lepených vazeb, jak už u Neokortexu (1) bývá zvykem. Kniha se neudrží otevřená, avšak listy nevypadávají.

Prvních 15 kapitol (asi 220 stran) je věnováno instalaci, konfiguraci a popisu principu práce nejrozšířenějšího webového serveru Apache HTTPd (2), druhá (tenčí) část, je věnována vysvětlení slibného nástroje pro vývojáře webových sídel, jazyku PHP/FI (3) a jeho propojení s Apachem.

Trochu netradičně působí, že vlastně první kapitola má číslo 3 (číslované kapitoly jsou i úvod a obsah). Najdeme v ní detailní popis spuštění Apache a seznam souborů, které Apache potřebuje pro svoji práci (konfigurační soubory a vznikající protokoly).

Následující kapitola je věnována překladu a instalaci Apache. Podstatnou část kapitoly zabírá překlad originálního manuálu, v druhé části je však velmi pěkně popsána část týkající se konfigurací virtuálních hostitelů. Toto je významná vlastnost Apache, kdy server dokáže vytvářet několik virtuálních serverů (např. *www.aaa.cz* a *www.bbb.cz*) pod jednou spuštěnou kopií Apache a dokonce i na jediné IP adrese. Autor zde čtenáře provádí úskalími jednotlivých variant (více spuštěných démonů, jeden démon apod.) včetně popisu odstranění chyb, které vás při konfigurování mohou potkat.

Kapitola 5 rozšiřuje tuto problematiku i výkladem, jak server provádí rozpoznání virtuálních hostitelů a další podobné teoretické záležitosti, které se vám budou hodit tehdy, až se vám nepodaří server nakonfigurovat ani podle příručky.

Šestá kapitola je věnována bezpečnosti a provozování uživatelských CGI skriptů. Je zde vysvětlena práce programu `suEXEC`, který by vám měl zaručit podstatně vyšší bezpečnost, pokud při používání serveru Apache dovolíte uživatelům tvořit vlastní CGI skripty, a navíc chcete zajistit pohodlí při práci jednotlivých uživatelů.

Kapitola 7 se zabírá kompilací a provozováním Apache pod systémem UnixWare, kapitola 8 pak přináší překlad manuálového článku týkajícího se změn mezi verzemi 1.1 a 1.2 (pro informaci, poslední verzí řady 1.2 je 1.2.5, nyní už existuje řada 1.3 nově podporující i platformu WindowsNT).

Za hodně zajímavou část této kapitoly považuji část, která stručně popisuje rozhraní API, které umožňuje dotvořit si další vlastnosti serveru přímo konkrétním uživatelem (zde myšleno samozřejmě správcem serveru). Možnosti API jsou bohaté a mimo standardní distribuce existuje celá řada doplňků, jako je např. modul pro podporu různých českých kódování, modul pro výše zmíněné PHP/FI, modul pro SSL (bezpečnostní protokol) či např. modul, který uživatelům usnadní zadávání URL na vašem serveru (např. chybně zadané písmenko se opraví aj.).

Devátá kapitola rozebírá činnost serveru při problémovém stavu TCP protokolu, tzv. stavu `FIN_WAIT_2`. V tomto stavu je již komunikace mezi serverem a klientem ukončena, avšak stále ještě server eviduje spojení (nedošlo ke korektnímu ukončení). Kapitola osvětluje, proč k tomuto dochází a jak to na různých platformách odstranit.

V druhé části kapitoly se nachází překlad dokumentu HOWTO (Jak na to), který dává odpovědi na základní problémy, se kterými se správce může setkat (např. jak omezit chování internetových robotů, jak přesměrovat server či resetovat protokolové soubory). Rovněž jsou zde zmíněny problémy, které mohou vzniknout na nelinuxových platformách (BSD, Digital Unix, ...).

Část 9.6 popisuje, co může vést ke zdánlivému (či skutečnému) zpomalení serveru. Je zde objasněn problém, který se na Linuxu občas vyskytne, tj. zamrznutí serveru na několik sekund až minut a jeho opětovné probuzení v důsledku činnosti Apache.

Kapitola 10 přináší zkušenosti, které mohou zvýšit bezpečnost vašeho serveru a usnadnit vám konfigurování. Druhá část kapitoly je referenční příručkou pro konfigurování serveru. Jde de facto o překlad originální dokumentace na *www.apache.org*, s podrobnějším popisem konfigurace modulu pro authentizaci vůči databázi mSQL.

V druhé části kapitoly je ukázána nová vlastnost Apache 1.3, o které se málo ví. Server dokáže obsluhovat dynamicky zaváděné moduly (`mod_dld`), tj. dotvářet si konfiguraci serveru až za chodu, bez nutnosti překladu. Apache 1.3 navíc přináší možnost používat i moduly ve formátu Win32 DLL. Toto samozřejmě využijí ti, kteří budou Apache používat ve Windows NT. Doufám, že toto bude jedna z klíčových vlastností (spolu s `mod_isapi` pro vytváření rozhraní mezi Apache a programy společnosti Microsoft), která nakonec vytlačí MIIS z oblasti Windows NT a nahradí jej Apachem.

V samém závěru je ještě rozebrán interní proxy modul Apache (`mod_proxy`), který dokáže pracovat i s protokolem SSL) a přepisovací modul (`mod_rewrite`), který umožňuje provádět přesměrování URL na základě regulárních výrazů. Tím lze např. dosáhnout, že uživatelská URL budou *http://www.nekde.cz/˜Milan.Sorm/*, i když účty na serveru jsou např. `xsorm`. Navíc lze při přepisování volat externí skripty např. v Perlu a tam už se fantazii meze nekladou.

Jedenáctá kapitola rozebírá konfiguraci více IP adres na jednom síťovém rozhraní pro SunOS 4.1.x, 12. kapitola obsahuje překlad dokumentu FAQ (často kladené otázky) a popis známých chyb jednotlivých klientů (abyste věděli, až budete rozjíždět svůj vlastní server, komu dávat vinu). Nakonec kapitoly následuje seznam známých chyb Apache.

Třináctá kapitola přináší pokračování referenční příručky z desáté kapitoly (bývalo by bylo asi inteligentnější dát je k sobě, ale autor použil tuto skladbu). Tato část se týká konfigurování samotného jádra Apache (kdežto v desáté kapitole byl popis konfigurace jednotlivých modulů).

Čtrnáctá kapitola ukazuje možnosti nového Apache 1.3 při spojení s Microsoft Windows s rozhraním ISAPI (Internet Server Applications API). Patnáctá kapitola pak podává výčet vlastností jednotlivých verzí Apache.

V druhé části knihy je popsán serverový skriptovací jazyk PHP/FI, který vám uvnitř HTML dovolí psát vlastní aktivní skripty (např. hodiny ukazující přesný čas apod.). Že už jste toto někde viděli? Ano, bylo to v JavaScriptu, ale ten byl závislý na implementaci v prohlížeči. PHP/FI je čistě záležitostí serveru.

PHP/FI je rychlejší než CGI (protože jej provádí již běžící Apache), umožňuje připojení na programy pomocí mechanismu trubky (kolony), jak to známe např. z Perlu – např. posílání pošty, komunikace s aplikacemi apod. Dále obsahuje značné možnosti pro protokolování své činnosti, takže uživatelé si sami mohou tvořit statistiky návštěvnosti jejich stránek apod., obsahuje vlastní ochranné metody přístupu k dokumentům, umí se napojit na řadu existujících SQL serverů (mSQL, Postgres, MySQL, Sybase, Oracle, Illustra, Adabas, Filepro), zvládá komunikaci skrz specifikaci ODBC či standardní DBM databáze. Zvládá uploadování souboru na server, ovládá matematické výpočty, matice, Perlovská asociativní pole (lze přenášet mezi jednotlivými webovými stránkami), možnosti skriptového jazyka C, regulární výrazy, tvorbu GIF obrázků atd.

Jak je vidět, určitě má PHP/FI budoucnost, navíc je zadarmo. Kapitoly 16 a 17 poskytují základní úvod do problematiky PHP/FI a popis instalace. PHP/FI může pracovat jednak jako serverový modul nebo jako CGI skript zprostředkující PHP/FI vašim skriptům (pak lze PHP/FI užít i v jiných serverech než Apache, ovšem ne už v takové rychlosti). V kapitole 17 je rovněž rozebrána konfigurace PHP/FI do bezpečného režimu (aby mohly být využity všechny bezpečnostní techniky PHP/FI).

Druhá část kapitoly 17 rozebírá konfiguraci jednotlivých částí PHP/FI a napojení na konkrétní SQL servery, využití jeho specifických vlastností apod. 18. kapitola podrobně popisuje vlastní jazyk PHP/FI. Kapitola 19 je potom referenční příručkou všech funkcí PHP/FI.

V závěru 19. kapitoly je potom rozebrána možnost dodefinovat si vlastní příkazy do PHP/FI. To přináší jedinečnou možnost upravit si jazyk podle svého. Navíc autor PHP/FI nabízí, že pokud mu nové funkce zašlete, rozšíří o ně novou verzi PHP/FI. No není to skvělé?

Dvacátá kapitola je pak dobře zpracovaný rejstřík. Mám však podezření, že není zcela kompletní. Např. `mod_perl`, o kterém autor hovoří v 16. kapitole, není v rejstříku k nalezení.

Kniha budí dojem překladu originálních manuálů, o čemž se autor na začátku zmiňuje. Na mnoha místech je však tento překlad rozšířen o cenné zkušenosti, což zvyšuje jeho kvalitu. Navíc nesrozumitelné pasáže originálního manuálu autor převyprávěl, takže jsou nyní pochopitelnější.

Kniha bude přínosem nejen pro toho, kdo se hodlá s Apachem seznámit (i když pro něj bude možná příliš stručná), ale především pro toho, kdo hodlá Apache vyladit na maximální možnou míru. Kniha je výbornou referenční příručkou a popisem toho, jak server pracuje (technický manuál).

Ten, kdo se chce seznámit s PHP/FI a vážně v něm pracovat, by měl po této knize sáhnout zcela určitě. Negativně působí jen trochu zmatené uspořádání některých kapitol, přičemž by některé dlouhé kapitoly (10, 17) stálo za to rozdělit do několika menších.

Sympatické by bylo, kdyby bylo alespoň za kapitolami vynecháno trochu volného místa na zapsání poznámek, protože do referenčních příruček si řada lidí vpisuje nově se objevivší direktivy či parametry. Na druhé straně toto by knihu prodražilo, a tak se musí člověk spokojit se čtyřmi volnými stranami na konci knihy.

V knize mi rovněž chybí seznam literatury či webových stránek, kde by člověk získal další informace a rozšířil si tak vědomosti (např. podrobnější specifikaci API Apache i API PHP/FI modulu apod.).

Knihu můžete zakoupit za 294 Kč v každém počítačovém knihkupectví. ■

1 Neokortex, spol. s r. o.
`http://www.neo.cz`
2 Apache HTTPd
`http://www.apache.org`
3 PHP/FI
`http://www.php.net`

SQL server pod Linuxem?! A kde jinde!?

Dan Ohnesorg, 10. dubna 1998

Před časem jste jistě zaznamenali, že jsem v linuxové konferenci hledal spoluautory pro článek o databázích pod Linuxem. Nakonec jsme se sešli tři a předáváme naše zkušenosti s databázemi, které používáme. Nebudeme se zde zmiňovat o databázi ADABAS, která byla podrobně popsána v prosincovém čísle Linuxových novin a databázi ORACLE, kterou lze sice na Linuxu úspěšně provozovat, ale nepodařilo se mi sehnat někoho, kdo by o ní něco napsal a navíc se jedná o databázi, která přece jen trochu vybočuje ze záběru většiny linuxových uživatelů. Nicméně toto zcela jistě není poslední číslo Linuxových novin a tak se k tomuto jistě zajímavému tématu ještě vrátíme.

PostgreSQL

Nathan L. Cutler

Pro ty z nás, kteří hledají databázový software, který je opravdu free, tj. nejen pro samotné užívání, ale i pro případnou distribuci za úplatu, se nabízí PostgreSQL (1). Tato robustní implementace standardu SQL-92 umí zpracovat dotazy obsahující relace mezi jednotlivými tabulkami. Je pod aktivním vývojem a každá nová verze má blíž k úplné implementaci standardu. Již nyní je implementace standardu téměř kompletní, a navíc PostgreSQL obsahuje některá rozšíření, se kterými standard nepočítá, např. uživatelem definované datové typy, funkce a inheritance.

PostgreSQL je postaven na principu klient-server, nebo front-end/back-end. Back-end v případě PostgreSQL je program, který beží jako daemon a zpracovává SQL-dotazy, které mohou přicházet i po síti TCP/IP. Back-end PostgreSQL je schopen obsloužit jak více databází, tak více uživatelů najednou. Distribuce PostgreSQL obsahuje i jednoduchý front-end `psql`, což je takový příkazový řádek, přes který lze zadávat holé SQL dotazy a prohlížet si jejich výstup.

Velmi důležitou součástí distribuce PostgreSQL je `libpq`, což je knihovna funkcí umožňující psaní front-endů k databázím. Pomocí této knihovny lze vyvíjet programy v jazyce C, které komunikují přímo s PostgreSQL back-endem. Pro ty, kteří umí programovat v jazyce C představuje tato knihovna spolu s PostgreSQL back-endem mocný nástroj pro vývoj všech druhů databázových aplikací.

Pro ty, kteří v céčku neprogramují, existuje řada jiných rozhraní, např. knihovna jazyka Perl, nástroje pro integraci s WWW a jiné.

Pro ty, kteří neumí nebo nechtějí získat PostgreSQL kompilací zdrojového kódu, existují PostgreSQL balíky s binárkami pro populární distribuce Linuxu.

miniSQL 1.0.11

Michal Polák

MiniSQL (dále jen mSQL) vytvořil David J. Hughes (bambi@Hughes.edu.au) z Austrálie. Tato verze je již poněkud staršího data – leden 1996. Volně používat jej mohou pouze studenti, vzdělávací nekomerční instituce, nevýdělečné a charitativní organizace a nekomerční výzkumná střediska. Ostatní musí zaplatit registrační poplatek (pro zajímavost: pro privátní použití 65 austr. dolarů, pro komerční 225 austr. dolarů). Další šíření produktu (např. jako součást nově vytvořeného programu) je povoleno pouze s výslovným písemným povolením Hughes Technologies.

Distribuční balík se „rozzipuje“ a „roztaruje“ do adresáře `/usr/local/Minerva` (implicitně nastaveno). Nachází se zde adresář `doc` s dokumentací v PostScriptu, a adresáře `include` a `lib` pro zakomponování balíku do naší aplikace. Časem zde ještě přibude adresář `msqldb`, kde budou uloženy všechny databáze, které tímto balíkem budeme spravovat. Podrobněji se budeme zabývat adresářem `bin` ve kterém se nachází několik spustitelných programů: `msql`, `msqld`, `msqladmin`, `msqldump`, `relshow`. Chování balíku lze samozřejmě konfigurovat. Pokud by někomu nevyhovoval adresář `/usr/local/Minerva`, může si balík nainstalovat do přijatelnějšího adresáře, pak je ale zapotřebí nastavit proměnnou `MSQL_HOME`. Kdyby byly problémy se sítí, lze ještě nastavit buď `/etc/services`, nebo proměnné prostředí `MSQL_TCP_PORT`, `MSQL_UNIX_PORT` a `MSQL_HOST`.

Monitorovat činnost „mSQL Engine“ lze nastavením proměnné `MINERVA_DEBUG`:

- *api* – vypisuje interní informace (jako jsou např. detaily připojení)
- *cache* – vypisuje využití cache tabulek
- *query* – vypisuje všechny dotazy zasílané databázovému serveru
- *malloc* – vypisuje pozice a velikosti alokovaných částí paměti
- *error* – vypisuje chyby (stejné jako zasílané klientovi)
- *key* – vypisuje detaily při práci s klíči
- *trace* – vypisuje volání funkcí, které se spouštějí
- *mmap* – vypisuje detaily o obsazeném regionu paměti
- *general* – vypisuje ostatní informace

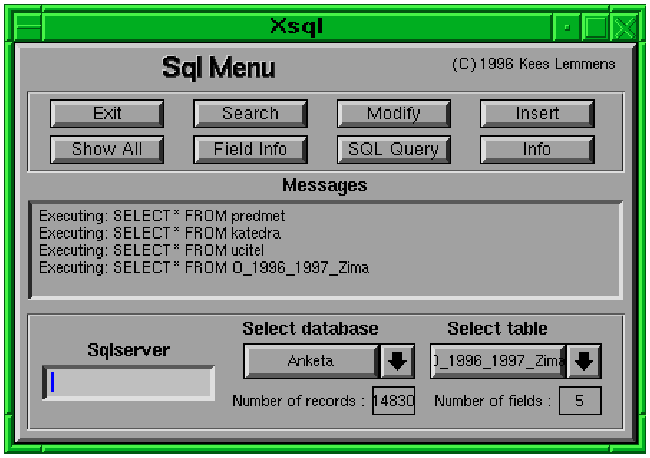
Tyto režimy lze kombinovat, pak se ale musí mezi jednotlivými slovy napsat dvojtečka. Např:

```
setenv MINERVA_DEBUG api:query
```

Pojďme si nyní popsat slíbené programy:

- *msqld* – démon
- *msqladmin* – tvorba/rušení databáze, shutdown/reload databázového serveru
- *msql* – klient (zadáváním SQL dotazů pracujeme s databází)
- *msqldump* – převede všechna data z určené databáze do SQL dotazů na výstup (vhodné k archivaci)
- *relshow* – zobrazuje názvy existujících tabulek nebo struktury tabulek

Všechny tyto programy (kromě démona) lze spouštět s parametrem `-h Host`, kterým se připojíte k databázovému serveru na jiném počítači. U programů `msql` a `msqldump` se vždy musí zadat jméno databáze, se kterou chceme pracovat. Jelikož je práce s `msql` celkem neohrabaná, je lepší (zvláště pokud potřebujete zadat více dotazů najednou) si tyto napsat do textového souboru a tento potom přesměrovat programu `msql` na vstup. Zde je nutno podotknout, že jednotlivé SQL dotazy je nutno oddělit znaky `\g` (příkaz v programu pro provedení – go – dotazu). Existuje nadstavba programu `msql` pro prostředí X-Window, která se překvapivě jmenuje `xmsql`.



Jazyk SQL (v dokumentaci deklarován jako ANSI SQL) je podle mého názoru na dnešní dobu celkem chudý.

Datové typy použitelné v klauzuli `CREATE TABLE`:

- `char(len)` – text maximálně délky `len`
- `int` – celé číslo se znaménkem (4 byty)
- `real` – reálné číslo

Za jednotlivými typy lze ještě uvést `not null` nebo `primary key`.

Lze pracovat s konstrukcemi `CREATE TABLE`, `DROP TABLE`, `INSERT INTO`, `DELETE FROM`, `UPDATE`. Konstrukci SELECT si popišme podrobněji, nejdříve uvedeme syntaxi:

```
SELECT [DISTINCT] [table.]column \
     [, [table.]column]*
  FROM table [= alias] [, table [= alias]]*
  [ WHERE [table.]column OPERATOR VALUE \
          [ AND | OR [table].column \
          OPERATOR VALUE]* ]
  [ ORDER BY [table.]column [DESC] \
  [, [table.]column [DESC]]* ]
```

Výraz v hranatých závorkách je nepovinný.
[...]* – výraz v hranatých závorkách se může libovolněkrát opakovat.
OPERATOR – =, <, >, <=, >=, <>, like
VALUE – hodnota nebo název sloupce.

Nelze použít *nested functions* (vložené funkce – sum(), count()), join (spojení tabulek), views (pohledy) a *nested queries* (vložené dotazy). Kontrola přístupu se nastavuje v textovém souboru msql.acl, který se nachází v /usr/local/Minerva (nebo v MSQL_HOME). Zde se pro jednotlivé databáze nastavují práva pro čtení, zápis apod. Před drahnou dobou jsem četl v linuxové konferenci, že tuto kontrolu lze velice snadno obejít. Protože ale nemám patřičné crackerské (nebo hackerské ?) znalosti, tak nemohu posoudit.

V dokumentaci je celkem podrobně probráno napojení na server z jazyka C. Informace o napojení z ESL, Perl 5, Python, Tcl jsou strohé, jsou zde pouze uvedeny adresy výrobců a názvy jednotlivých podpůrných modulů.

Tabulka, kde je kolem 14830 záznamů a 5 sloupců typu int, zabírá na disku něco přes 385 kB ($14830 \cdot 20 = 296600$). Řekl bych, že náklady na režii jsou celkem vysoké, zvlášť když v tabulce neexistuje primární klíč. V konferenci někdo porovnával rychlosti mSQL a MySQL. Balík mSQL byl pomalejší.

Z výše uvedených skutečností a absence triggerů, procedur, zamykání záznamů apod. vyplývá, že balík mSQL se hodí pouze do soukromého použití, kde budou relativně malé tabulky v databázi a tabulky na sobě nebudou hodně závislé.

Jako klady bych vyzvedl snadnou instalaci a relativně málo zabraného místa na disku.

Já osobně používám mSQL jako back-end v programu na vyhodnocení odpovědí studentů v anketě o studiu. Jako front-end mi slouží aplikace napsaná v C. Za celou dobu provozu (zhruba rok a půl) nebyl s mSQL žádný problém (Pozn. redakce: s tímto tvrzením bych si dovolil nesouhlasit. Můžu doložit výstupem příkazu grep mpolak ~/Mail/Archive/Received* :–). V tabulkách jsou řádově stovky záznamů, vyjímkou jsou tabulky odpovědí, kde je v každé kolem 15000 záznamů. Jediné, co bych chtěl reklamovat u výrobce, je skutečnost, že se mi nepovedla vytvořit tabulka s názvem delším než asi 20 znaků. Možná je to součást nějakého standardu, nevím.

MySQL

Dan Ohnesorg

Příliš restriktivní licence mSQL a špatná srozumitelnost jeho kódu byla trnem v oku skupině programátorů pod vedením Michaela „Montyho“ Wideniuse, Davida Axmarka a Kima Aldala. Proto se rozhodli vytvořit nové mSQL se stejným rozhraním, ale mnohem dokonalejším front-endem a back-endem. MySQL má tedy mnoho společného s mSQL a existují nástroje pro upgrade z mSQL na MySQL, jak na úrovni přenosu dat tak na úrovni ekvivalentních knihoven, jejichž použití předpokládá v běžných případech pouze použití jiného hlavičkového souboru a jiného linkovacího parametru.

MySQL přesto není pod GNU licencí, jak by možná někdo po přečtení předchozího odstavce očekával. Licence je poněkud složitější. Program se pro její účely dělí na klientskou část, která je pod GPL licencí a server samotný, který je volně šiřitelný, ale nesmí být prodáván a nesmí být použit jako integrální součást komerčně prodávaných produktů. Pokud má někdo takovéto ambice (a mnoho takových se najde) musí si zakoupit komerční verzi serveru. Ta má výhodu v tom, že obsahuje některé funkce, které ve volně šiřitelné verzi nejsou. V současné době se ale jedná pouze o možnost pracovat s komprimovanými databázemi. V případě zájmu si uživatelé mohou zakoupit placenou technickou podporu. Ta obsahuje mnoho stupňů, od zodpovídání e-mailových dotazů, až po vzdálenou administraci systému a návštěvy vývojářů u zákazníků. Nechci nikoho zrazovat od finanční podpory autorů, ale musím konstatovat, že když jsem upgradoval na RedHat Linux 4.9 a MySQL přestalo fungovat, Monty problém vyřešil asi po pěti dopisech za 4 hodiny a to žádnou podporu zaplacenou nemám. (Jednalo se jak možná mnozí tuší o první verzi RedHatu s glibc2, tehdy se MySQL rozběhlo po zakomentování několika funkcí v hlavičkových souborech. Dnes si s glibc2 rozumí, ale požaduje nejnovější verzi z RedHat/upgrade.) Pro srovnání náklady jsou takovéto: server pro komerční využití a jednoho uživatele 200$, server pro komerční využití a deset uživatelů 1500$, placená podpora – základní verze 200$ na rok, placená podpora nejkomfortnější 5000$ na rok (obsahuje vyhotovení speciálních dotazů typu SELECT můj_vzoreček(první sloupec, druhý sloupec) FROM databáze; v jednotkách hodin, právo požadovat speciální úpravy serveru, instalace u zákazníka a mnohé další). Závěrem snad jen tolik, že autoři slibují, že po dokončení dostatečně dokonalé verze (podle jejich subjektivního názoru) budou všechny předchozí uvolněny pod názvem GNU MySQL pod GNU GPL licencí. Pro tu dokonalou verzi zůstane v platnosti omezení prodeje a integrace do vlastních produktů, do té doby než bude ještě dokonalejší verze.

Kolik databáze může stát už tedy víme a teď se podíváme co umí. Je toho opravdu hodně. Nebudu zde zmiňovat ani zdaleka všechny možnosti, jistě si je prohlédnete na adrese (2). Za základní považuji podporu následujících operačních systémů

- Solaris 2.5, 2.6 with native threads.
- SunOS 4.x with the included MIT threads package.
- BSDI 2.x with the included MIT threads package.
- BSDI 3.0 and 3.1 native threads.
- SGI IRIX 6.x with native threads.
- AIX 4.x whith native threads.
- DEC UNIX 4.x with native threads.
- Linux 2.0+ with LinuxThreads 0.5, 0.6 and 0.7.
- FreeBSD 2.x with the included MIT threads package.
- FreeBSD 3.x with native threads.
- SCO OpenServer with a recent port of the FSU-threads package.
- NetBSD
- OpenBSD 2.1
- HPUX 10.20 with the included MIT threads package.
- Win95 and NT. Tato verze je k dispozici jen pro platící zákazníky.

MySQL je multithreadovou aplikací, proto dokáže vynikajícím způsobem rozkládat zátěž na více procesorů.

Podpora kompletní normy ANSI SQL 92 s vyjímkou vložených procedur, triggerů, cizích klíčů, pohledů (`views`) a transakcí. (Vysvětlení toho proč MySQL neumí transakce zabírá několik odstavců v dokumentaci, zhruba takto: databáze bez transakcí je 3x rychlejší a většinu věcí lze řešit alternativní cestou přes zamykání tabulek.) Nicméně mi transakce scházejí a jsou zřejmě nejpádnějším důvodem, proč se MySQL prosazuje především v oblasti prohlížení dat statičtějšího charakteru. Mnozí jistě namítnou, že transakce jsou potřeba především tam, kde operační systém, na kterém databázový server běží, neposkytuje dostatečnou stabilitu, ale přesto je databázové programování s transakcemi snažší, srovnal bych to s vyjímkami v C.

MySQL má určité omezení v konstrukcích používajících `OUTER JOIN`s, např. nemá vůbec implemetován `FULL OUTER JOIN`. Bežné dotazy ale takové speciality stejně nepoužívají.

Naopak navíc oproti normě SQL 92 má mnoho matematických funkcí, nastavování parametrů, aliasy pro mnoho funkcí, tak aby syntaxe byla kompatibilní s co největším počtem databází, automatické generování ID a porovnávání pomocí regulárních výrazů. Dá se předpokládat, že pokud umíte psát SQL dotazy pro ORACLE, INFORMIX či INTERBASE, MySQL jim porozumí.

Protože máte k dispozici bohatě komentované zdrojové texty není problém si dopsat další speciální funkce. Pro funkce matematického typu existuje unifikované rozhraní. Jak praví manuál:

> MySQL sice nemá vložené funkce, tak jak jste zvyklí z jiných databází, kde se vytvářejí v SQL jazyce. MySQL má vnořené funkce psané v jazyce C, každý si může napsat co potřebuje.

Při běhu volitelně vytváří `isamlog`, ze kterého je možné rekonstruovat poškozenou databázi, případně provést příkazy od poslední zálohy znovu.

Kromě `BLOB`u neomezené velikosti (tedy omezené velikostí dostupné paměti) má typ `TEXT` se stejnými vlastnostmi, ale při porovnávání v poli typu `TEXT` se ignoruje velikost písmen.

A v neposlední řadě, MySQL má dotaženou podporu češtiny. Nejen, že vypisuje (samozřejmě jen pokud si přejete) hlášení o provedených akcích česky, ale umí porovnávat české výrazy u výběrových kritérií bez ohledu na velikost písmen a třídí maximálně algoritmizovatelně věrně podle platné ČSN s respektováním specifik písmenka ch. Jediné co musíte udělat je při překladu přidat parametr

```
./configure -with-charset=czech
```

Ale to stále není všechno, MySQL interně pracuje v ISO-8859-2, ale klient může používat jiné kódování. Žádný problém – stačí zadat `set character set cp1250_latin2` (v jiné verzi může být i `cp1250_il2`) a klient se serverem komunikuje v `cp1250`. Tato možnost zatím není podchycena jako parametr na příkazové řádce při konfiguraci a tak je nutné v `.../src/sql/convert.cc` zrušit komentář, který blokuje `#define DEFINE_ALL_CHARACTER_SETS`.

Ve verzi 3.21.26-gamma je ještě chybička, zřejmě vypadlo několik řádek z patche, a tak přepínaní charsetů úplně nefunguje. Předpokládám, že v době kdy budete článek číst, bude vše opravené.

V blízké době bude uvolněna verze 3.21.30, která bude mít ve standardní distribuci jak plně funkční překódování znakových sad na straně serveru, tak drobné opravy v konstrukci `SELECT LIKE` při použití češtiny.

Monty říká:

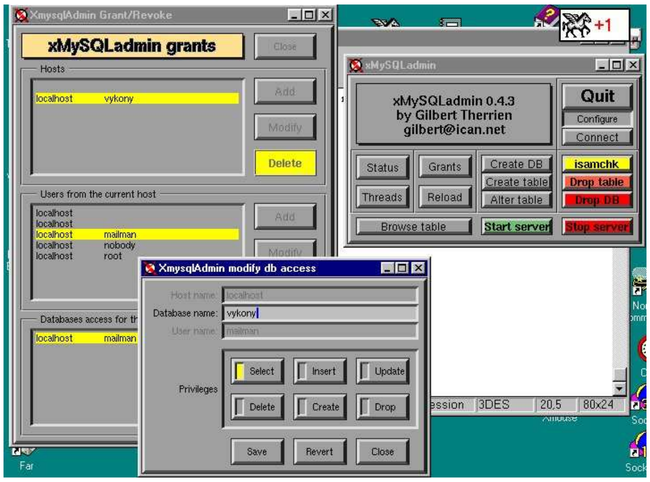
> Do MySQL je možné implementovat všechno, pokud chcete vidět opravdu složitý algoritmus podívejte se do `ctype-czech` na české třídění.

(Třídění Monty nedělal, je to dílo zlatých českých ručiček. Nebudu jmenovat, nerad bych na někoho zapomněl).

Pro MySQL existuje podpora ODBC, tedy ODBC ovladače pro Windows (pouze 95 a NT), JDBC a samozřejmě sdílená knihovna `libmysqlclient`, kterou může použít libovolný program napsaný v jazyce C. Dále jsou dostupné knihovny pro Perl a Python.

Server komunikuje s klientskou částí dvěma způsoby, buď před socket, nebo po síti. Druhou variantu lze potlačit parametrem při startu démona. Databáze jsou před neoprávněnou manipulací zajištěny soustavou přístupových práv. Práva se přidělují ve třech stupních: na stroj, ze kterého se uživatel přihlašuje, databázi a tabulku v databázi. Jedná se o běžná práva jako čtení, zápis, přidání, vytvoření tabulky, restart serveru, výpis běžících úloh a podobně. Stejně jako u mSQL není tato ochrana neprůstřelná, neprovádí se žádné šifrování dat, takže při získání fyzického přístupu, mohou být data odnesena. Není také problém podvrhnout serveru jinou databázi uživatelských práv. Opět to ale vyžaduje fyzický přístup k disku a právo na restart serveru, ten totiž práva čte jen jednou při startu nebo restartu. Hesla samotná jsou v databázi uložena v šifrované podobě a stejně tak každý klient obsahuje šifrovací engine, takže hesla se po sítí nepřenášejí, nebo alespoň ne v otevřené podobě. Démon se má startovat dávkou `safe_mysqld`, která se snaží odhalit běžnejší podvody se symlinky a provádí několik dalších kontrol.

Pro import a export souborů existuje několik nástrojů. Základní je možnost exportovat data ve formátu SQL příkazů do souboru. Ale existuje i konverzní program z a do DBF souborů, nebo možnost číst a zapisovat data do souborů typu *comma delimited*.



Pro administraci a prohlížení dat existují Xové nástroje. Sice umožňují komfortní nastavování práv klikáním myší, ale nevynikají právě rychlostí (to ale bude zajímat jen majitele 486). Stejný komfort na příkazové řádce poskytu-

```
Datum odeslání:   Fri, 17 Apr 1998 10:17:52 -0700
Od:               "Wyliam Holder" <wholder@interbase.com>
Poštovní prog.:   Mozilla 4.03 [en] (WinNT; I)
Komu:             "Dan Ohnesorg, admin on power" <Dan@feld.cvut.cz>
Věc:              Re: Interbase for linux - yes
Hi Dan:
I can indeed.  A free, downloadable copy will be available from our
internet site www.interbase.com.  It is scheduled to be there on April
29, 1998.
As you stated, this is the 4.0 version of Interbase.  An Interbase 5
version is planned to be released in July of 1998.
Regards, wholder@interbase.com
```

Výpis č. 1: Dopis od koordinátora Interbase pro Linux

je mysqladmin, který kromě zakládání a rušení databází zobrazuje stav serveru, obsazení paměti a počet vyřízených požadavků. Struktura databází se dá zobrazit přes mysqlshow. Když se stane, že server nedokáže databázi po výpadku otevřít, lze ji opravit programem isamchk, naštěstí jsem jej ale nikdy nepotřeboval.

Velikost databází je omezena jen velikostí disku. Autoři uvádí, že největší implementace běží nad 50 000 000 záznamy bez nejmenšího zaváhání.

Závěrem zmíním několik produktů, které s MySQL úzce souvisí. Na prvním místě musí jistě být scriptovací jazyk pro tvorbu dynamických WWW stránek PHP/FI, který naleznete na adrese (3). Tento program existuje ve dvou verzích, 2.x a 3.x, které se liší syntaxí a především použitým jazykem. Řada 3.x byla přepsána do C++. Funkčně jsou zatím stejné, ale 2.x nemá asi budoucnost, proto se pro nové implementace doporučuje 3.x. Trojková řada však zavádí komfort v generování dynamických stran nevídaný. Existují v ní dokonce objekty a dědičnost. Obsahuje moduly pro zobrazení zdrojového kódu stránek s obarvováním a mnoho, mnoho dalšího zajímavého. Zcela jistě se k PHP/FI vrátíme v příštích číslech.

O něco méně dokonalý (podle mne) je WWW-SQL na adrese (4), pro někoho však může být příjemné, že se syntaxí přibližuje ASP z Windows NT.

O totální zpřístupnění databáze na Internetu se snaží WDB z adresy (5), produkt se mi moc líbí, bohužel v balíku omylem chybí jedna část a autor nestihl dodat opravu. Jedná se v zásadě o formuláře přístupné přes WWW rozhraní s běžnou funkčností, tedy zobrazení, úpravy, vkládání nových, definice filtrů a podobně. Možná se k WDB vrátíme v příštích číslech.

Velmi zajímavý je internetový obchodní dům používající MySQL a PHP/FI na adrese (6).

Pro úplnost je nutné dodat, že výše zmíněné produkty lze připojit i na PostgreSQL. MySQL má ale o něco menší paměťové nároky.

Dále existují podpory pro BIND, kdy BIND v rozsáhlých sítích používá extrémně rychlé vyhledávací stromy implementované v MySQL, podpora pro autentizaci uživatelů a logování přístupů na server APACHE, fulltextové vyhledávací nástroje (bohužel dostupné jen v angličtině a dánštině, vzhledem k tomu, že obsahují databáze synonym a podobné vymyšlenosti je jejich portování na češtinu extrémně obtížné). A mnohé další.

Možná jsem vás přesvědčil, že MySQL si trochu pozornosti zaslouží. Při rychlosti jakou nabírá nové možnosti a funkce aspiruje na nejrozšířenější databázi vůbec. Problém free software je samozřejmě v tom, že není znám počet jeho uživatelů, takže až tento okamžik nastane, tak to nikdo nepozná. Pokud plánujete zveřejňování dat na internetu, rozhodně si na MySQL ve spojení s PHP najděte chvilku času. Pro ty, kteří si nedovedou život představit bez FrontPage: PHP 3.0 s ní dokáže pracovat, resp. používá takové značky, které FrontPage nezničí. Sice zatím nejsou české binárky, ale překlad netrvá na pentiích déle než hodinu.

Závěrem

Která je tedy vlastně nejlepší? Těžko říci, stejně jako ve velkých databázích typu ORACLE, INFORMIX či DB2, každý má svého favorita, na kterého nedá dopustit. Vývojáři spolu soutěží jak ve spolehlivosti, tak rozsahu poskytovaných funkcí a tak jakékoliv dnes provedené srovnání nemusí za měsíc s příchodem další verze souhlasit. Mým osobním favoritem je, jak jste si jistě všimli, MySQL. Přestože není až tak úplně free vítězí především podporou češtiny. ■

1 PostgreSQL
http://www.postgresql.org
2 MySQL homepage
http://www.tcx.se
3 PHP/FI
http://www.php.net
4 WWW-SQL
http://www.daa.com.au/~james/www-sql/
5 WDB
http://www.lava.net/beowulf/programming/wdb/
6 Obchodní dům nad MySQL a PHP/FI
http://www.minivend.com/minivend/

Neuvěřitelné se stalo skutkem!

Dan Ohnesorg, 17.dubna 1998

Pravděpodobně jste zaznamenali těžko uvěřitelnou zprávu v linuxové konferenci o uvolnění Interbase pro Linux. Protože se mi to nezdálo možné, okamžitě jsem kontaktoval zastoupení firmy Borland, které uvolnění verze pro Linux potvrdilo. Následuje dopis od koordinátora celé akce, Wyliama Holdra (Dopis od koordinátora Interbase pro Linux).

Co dodat, snad jen několik základních postřehů někoho, kdo již nějakou dobu používá Interbase na Novellském serveru. Interbase workgroup server, jak se produkt jmenuje celým jménem, je relační, transakční databáze, existující ve verzích pro Windows NT, Novell Netware a mnoho různých verzí Unixu. Je navžena jako databáze bezúdržbová, oproti třeba MySQL se na ní nedá moc nastavit, přesto funguje velice spolehlivě a plně využívá možnosti, které ji poskytuje operační systém. Základní vlastnosti:

- kompatibilita s SQL 92, včetně transakcí, triggerů a vložených procedur
- zrcadlení databází, kdy server data ukládá na libovolný počet míst současně, není tedy potřeba zrcadlit celé disky, stačí jen jednotlivé kopie směřovat na různé disky
- zasílání zpráv aplikacím, např. databáze zašle klientovi upozornění o změně nějaké hodnoty, pokud se klient zaregistruje, že chce o změně vědět
- pojmenovávání transakcí
- automatický dvoufázový „commit“ změn v databázi
- databázový server komunikuje s klienty současně přes protokoly TCP/IP, IPX/SPX a NetBEUI (to asi nebude platit na Linuxu)
- k databázi existuje detailně popsané API včetně několika příkladů komunikace v jazyce C

Velmi propracovaná je podpora cizích jazyků. Každý sloupec v tabulce může být deklarován v jiné kódové stránce, seznam podporovaných kódových stránek je velmi dlouhý a obsahuje jak všechny, které jsem kdy viděl u nás, tak unicode. Navíc je možné v příkazu SELECT specifikovat jak chceme výsledek setřídit (např. `pdox_csy`, `pdox_slo`, `pdox_ansi` a podobně). Ovšem kvalit čtyřprůchodového třídění v MySQL Interbase nedosahuje.

Interbase je velice robustní databáze a vyniká především ve spojení s Delphi či IntraBuilderem. Osobně se domnívám, že po uvolnění verze pro Linux zadarmo ztrácí firma Borland možnost prodat třeba jen jedinou licenci pro WindowsNT a Novell. Je sice možné, že lidé budou chtít novou verzi 5.0, ale verze 4.0 poskytuje všechen potřebný komfort a pro toho, kdo ještě zůstává u lokálních tabulek v Paradoxu je to ideální řešení. ■

Linuxový seminář na Cikháji z mého pohledu

Petr Olšák, 3. dubna 1998

Ve dnech 28. a 29. března 1998 pořádalo České sdružení uživatelů operačního systému Linux první „Linuxový seminář“. Akce se odehrála v rekreačním a výukovém středisku Masarykovy university situovaném do malinké vesničky Cikháj, kterou můžeme najít na podrobnější mapě zhruba 10 kilometrů od Žďáru nad Sázavou směrem na sever. Byl jsem pověřen napsat o této akci několik řádků.

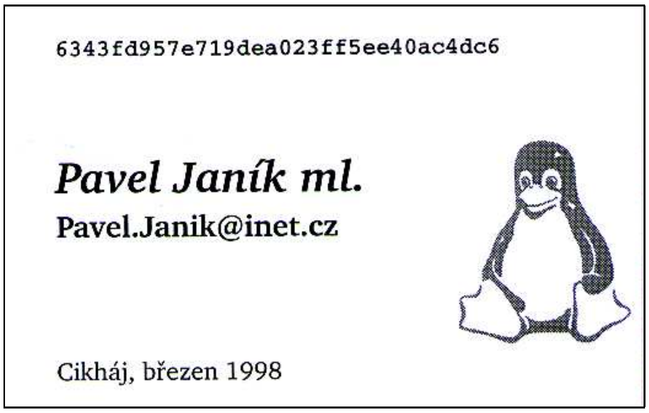
Jako reportér této události jsem v poněkud obtížné situaci. Čtenáře, který se nemohl zúčastnit, by zřejmě nejvíce zajímaly podrobnosti z jednotlivých přednášek. Dopředu upozorňuji, že takového čtenáře zklamu. Většina přednášejících totiž přislíbila, že dodá na síť (nebo redakci Linuxových novin) materiály svých přednášek. Nemá tedy cenu věci opakovat. Můj často podstatně méně fundovaný náhled na jednotlivá témata přednášek by mohl vést jen ke zbytečnému zkreslení. Rozhodl jsem se proto popsat Cikhájskou událost svým pohledem na věc zcela bez nároku na přesnost a hlavně bez nároku na objektivitu hodnocení.

Pohodlný způsob dopravy z Prahy až na Cikháj mi poslalo samo nebe, přesněji Pavel Janík, který mi poslal tip na Michala Chouru, jenž pojede z Prahy na Cikháj autem. Doprava autobusem a dalšími dopravními prostředky mi připadala příliš komplikovaná, tak jsem nabídku svezení autem s radostí přijal. Nakonec jsme se do škodovky vešli čtyři, ještě Dan Ohnesorg a Petr Škoda, který prohlásil, že neví o žádných svých předcích, kteří by se stejnojmenným typem vozu měli cokoli společného.

Na místo jsme dojeli hluboce v předstihu, protože jsme se domnívali (mylně), že bude hezky a projdeme se po okolí. Mírně poprchávalo a parkoviště před budovou zelo prázdnotou. Z hlubin budovy se vynořil příjemný pán v bílem plášti, který s námi prohodil pár slov. Zřejmě vedoucí podniku. Informoval nás, že nejbližší hospoda je až v další vesnici tři kilometry daleko. V závěsu za námi přijela ještě jedna posádka vozu. Společně jsme se rozhodli ono nejbližší restaurační zařízení prozkoumat. Hospoda to byla příjemná, jídlo dobré, ale shodli jsme se na tom, že by potřebovali hardwarový urychlovač, nebo rovnou upgrade centrální kuchařské jednotky. Hodinové čekání na jídlo (když jsme byli v celém lokále v podstatě sami) neodpovídá parametrům současných zařízení tohoto druhu.

Po návratu na ubytovací středisko Cikháj přetékala z parkoviště auta na všechny strany a u vchodu se vyskytovali lidé se jmenovkami na hrudích – neklamné to znamení, že se zde skutečně začíná něco dít. Bylo půl druhé odpoledne. U dveří jsme byli zkontrolováni, že jsme zaplatili ubytovací poplatek a ošpendleni cedulkami se svými jmény, e-mailovými adresami a kontrolními součty těchto dvou údajů.

V budově byla jedna větší místnost, vpředu prostor pro tabuli nebo plátno pro zpětný projektor, dále řada židlí a vzadu bar a výdejna jídla pro ubytované. Dále se v přízemí i na patře nalézaly pokoje po třech až čtyřech lůžkách. Celková kapacita ubytovny je asi padesát míst. Stačil jsem akorát z tašky vybalit a na čestném místě umístit diplom a obludnou keramiku od Softwarových novin za Produkt roku 1997 a akce začala.



Velmi krátce zahájil seminář Yenya Kasprzak. Upozornil na to, že předchozí noc spal nejvýše hodinu, protože se snažil připravit obsah CD-ROM s aktuální verzí Red Hat Linuxu 5.0 včetně všech aplikovaných záplat a úprav, které byly do dnešního dne známy a doporučeny k použití a včetně záležitostí, týkajících se českého jazyka. Vyjádřil naději, že CDčko by mohlo dokonce bootovat a dotázal se, kdo z přítomných by měl o takto zabalený software zájem. Pro řádné členy sdružení byla stanovena cena za jedno CD na 75 Kč a pro ostatní 120 Kč. Zájem byl poměrně velký. Yenya přivezl svůj počítač opatřený SCSI se dvěma vypalovačkami a umístil jej do koutu místnosti a jal se vypalovati, aby byl do konce semináře stanovený počet 18 kusů CDček hotov. Proto záhy předal slovo prvnímu přednášejícímu, Danu Ohnesorgovi.

První přednáška se týkala Samby, což je server i klient běžící na UNIXových platformách pro komunikaci na sítích používaných firmou Microsoft. Dozvěděli jsme se podrobně o historii tohoto projektu a o současných možnostech. Byly též zmíněny možnosti připojení faxu, který běží na serveru a je v klientovi viděn jako tiskárna. Autor Samby Andrew Tridgell nabízí svůj software jako free, pouze s dodatkem, že pokud se někomu Samba líbí, může mu zaslat pizzu, protože pizzu má autor rád. Dan promítl barevné slídy s ukázkami WWW stránek, které sumarizují počet registrovaných uživatelů, počet zaslaných pizz a uvedl některé větší organizace, které Samba server používají (řádově tisíce současně připojených klientů na jeden server). Mezi takové organizace patří třeba Bank of America. Dan prozradil, že na katedře elektroenergetiky FEL ČVUT používají pro diskový a tiskový server raději Linux se Sambou než Windows NT, protože jednoduše chtějí, aby server držel a nepadal. Mohu i za sebe konstatovat, že mám se Sambou velmi dobré zkušenosti. Snad jenom záhadné počítání volného místa na síťovém disku klientem je poněkud vada na kráse. Myslím si, že to bude tím, že Microsoftí klient nepředpokládá příliš velkou celkovou kapacitu síťového disku ještě navíc pro každého uživatele quotovanou.

Po krátké diskusi na téma Samby jsem měl nastoupit já se svými informacemi o TEXu. Požádal jsem ale pořadatele o vyhlášení krátké přestávky, abych se mohl z nárazu informací o Sambě vzpamatovat a abych mohl instalovat tabuli. Přeci jenom inklinuji při přednášení ke klasickým vyjadřovacím prostředkům – ke křídě a tabuli. Slídy, rámečky, meotary a počítačové prezentace nepovažuji za zcela nezbytné. Také jsem tím chtěl zakrýt skutečnost, že jsem byl nucen mluvit s patra bez předem připravených materiálů, které by jinak bylo možné touto technikou prezentovat.

Při své přednášce jsem byl postaven do jedné z nejkomplikovanějších situací za celou dobu svého pedagogického působení. Uvědomil jsem si totiž, že mezi posluchači jsou jedinci, kteří by mě svými znalostmi o TEXu mohli konkurovat a na druhé straně tam určitě byli posluchači, kteří o tomto programu nevěděli skoro nic. Bylo jasné, že svou přednáškou zřejmě neuspokojím ani jeden ani druhý extrém. Vsadil jsem tedy na nezávazné povídání o historii projektu TEX a o základech implementace TEXu z výchozích zdrojových WEBových textů a o instalaci formátů. Při povídání jsem mezi přítomnými nechal kolovati všech pět Knuthových základních knih o TEXu, několik časopisů amerického TUGu (TUGboat) a CSTUGu (Zpravodaj). Upozornil jsem na existenci relativně nové verze `web2c-7.2` a zmínil jsem stručně nové projekty TEXu (e-TEX, pdfTEX, Omega), které jsou v nové verzi zahrnuty. Přislíbil jsem aktualizaci CSTEXového `README.web2c` pro tuto verzi během týdne. Sepíšu tam své zkušenosti a popis implementace TEXu pro Linux ze zdrojových textů včetně použití svého rozšíření encTEX. Ideální pak bude, když jednotliví správcové jednotlivých balíčků zabalí výsledek kompilace provedené podle `README.web2c` do úhledných balíčků včetně binárních programů k okamžitému instalování a použití uživatelem. Balíčky se budou lišit podle typu distribuce Linuxu a podle verze `libc`, proti které byly binární programy kompilovány.

Mé vystoupení bylo ukončeno budíkem, jehož rachot vycházel odněkud zezadu z místa pro výdej jídla. To byl zcela zřejmý povel k nástupu na večeři. Během chvilky se vytvořila před výdejním místem fronta typu FIFO na smažený řízek s bramborovou kaší.

Po večeři vystoupil Petr Staněk s povídáním a s ukázkami počítačových her na Linuxu. Přinesl si svůj notebook, ke kterému připojil rámeček na meotar a promítal obsah své obrazovky na plátno. Občas se na plátnu vyskytovaly záběry ne nepodobné těm, které se odehrávají na obrazovkách herních automatů. Stručně řečeno – střílačky. Přednášející používal tvrzení typu: „každý z vás ví, co to je Doom“. Takové tvrzení mě jako matematikovi připadalo nesprávné. Protože je zde použit tzv. obecný kvantifikátor, negace tohoto tvrzení zní: „Existuje aspoň jeden z vás, který neví, co to je Doom“. Právě tato negace je podle mého názoru pravdivá. Tím aspoň jedním z přítomných jsem byl určitě já osobně. Při pohledu na plátno se střílačkami jsem přitom dospěl k samostatnému závěru, že jsem o nic světoborného nepřišel. Prostě, každého zajímá něco jiného. Z přednášky jsem si přeci jenom odnesl jeden zajímavý poznatek, který jsem do té doby neznal: existují střílačkové hry, které se dají hrát ve dvojicích i ve větším počtu hráčů, přičemž každý hráč má svůj počítač a „zabíjí“ ze svého pohledu toho druhého. Jednotlivé scény jsou promítány z pohledu jednotlivých hráčů a jsou časově synchronizovány přes síť tak, že po síti neběží grafika, ale jen synchronizační údaje.

Závěrečná přednáška prvního dne semináře patřila Yenyovi Kasprzakovi, který zmínil technické pozadí implementace firewallu na Linuxu. Nejprve uvedl obecné bezpečnostní problémy a pak podrobně na příkladech rozebral dva přístupy k nastavování parametrů firewallu – `ipfwadm` a `ipchains`. Barevné „slídy“ s ukázkami a kostrou přednášky promítal na plátno prostřednictvím notebooku. Ten byl z jedné strany připojen k rámečku na meotar a z druhé strany síťovým kabelem k „serveru“, na kterém se v koutku vypalovala výše zmíněná CDčka. Prezentace byla realizována programem `xv`, přičemž texty pro jednotlivé bitové mapy byly vyrobeny v TEXu a opepřeny duhovými přechody pomocí programu GIMP. Zůstalo utajeno, zda použitá minisíť (skládající se ze dvou počítačů) měla implementován firewall či nikoli. Osobně se domnívám, že nikoli, protože by u této konfigurace bylo těžko možné rozlišit, které napadení je napadením zevnitř a které zvenčí. Modemy pro připojení na vnější síť nebyly na Cikháji použity a mobilní telefony fungovaly jen některé.

V diskusi na téma firewall někdo uvedl zajímavou informaci, že prý banka Expandia připravuje připojení na Internet, a přitom prý v konfiguraci sítě budou pracovat i servery s důležitými daty banky. Technické zabezpečení proti zneužití prý dělali fundovaní odborníci a nyní v rámci pokusného provozu s neostrými daty vypisují soutěž: komu se podaří systém nabourat, bude odměněn a bude mu okamžitě nabídnuto v bance dobře placené místo.

Po poslední přednášce účastníci přešli na tzv. „práci v sekcích“. Jednalo se o nezávazné debaty na téma Linux i jiné u stolů s různými nápoji. Já nejsem noční pták, takže jsem tuto „práci v sekcích“ ukončil sotva se zvedla první skupina k odchodu. To bylo (na mě osobně velmi pozdě) ve dvě hodiny v noci. Slyšel jsem později z doslechu, že v ještě hlubších nočních hodinách nastaly technické problémy s SCSI pro vypalovačky. O noční vypalování se staral Petr Macháček, který se snažil rozjet druhou vypalovačku. Ta se ale tvářila, jako by byla na druhém SCSI řadiči (který tam ovšem vůbec nebyl). Přesto nakonec Petr stihl vypálit potřebný počet CDček, i když se zřejmě příliš mnoho nevyspal. Noc byla pro nás pro všechny o hodinu kratší, protože se přecházelo na letní čas.

Hned po snídani v 9 hodin ráno letního času už byli všichni na svých místech. Pan Milan Zamazal předvedl editor Emacs v akci. Použil k tomu Yenyův počítač a notebook jako grafický terminál. Na plátně hrál Emacs všemi barvami. Bohužel tyto barvičky byly promítacím rámečkem dost výrazně zkresleny. Pan Zamazal ukázal, jak se v Emacsu vytváří program v jazyce C. Napsal jednoduchou knihovnu, hlavičkový soubor a hlavní program. Pak za pomocí gdb dokázal v Emacsu celý program pohodlně krokovat. Jednotlivé části kódu se Milanovi pod prsty automaticky doplňovaly. Analogicky byla předvedena výroba jednoduchého LaTeXového vstupního textu včetně databáze bibliografií a odkazů na tuto databázi. V činnosti byl lispový kód Emacsu pro pořizování LaTeXu s názvem AUCTeX. Emacs opět automaticky doplňoval jednotlivé LaTeXové řídicí sekvence a to i podle kontextu. Jednotlivé části LaTeXového kódu byly vyznačeny různými barvami. Vrcholem ukázky byl Emacs+Gnus, což je prostředí pro zpracování e-mailů s obrovskými možnostmi. Uživatel si může definovat plno parametrů pro jednotlivé přijaté zprávy, pomocí kterých se usnadňuje třídění a orientace. Takový komfort je důležitý zvláště tehdy, dostáváme-li stovky dopisů denně. Závidím panu Zamazalovi jeho bezkonfliktní soužití s Emacsem, což je věc, kterou bych taky potřeboval. Já jsem v tomto editoru sice napsal už dvě knihy, ale pořád nad ním trpím, protože nevím, jak rychle vyřešit hodně jednoduché věci, které jsem v DOSových editorech měl raz dva. Rovněž mě odrazuje polymorfní chování tohoto editoru. Například když jednou edituji soubor s příponou `txt` a podruhé soubor zcela bez přípony, pokaždé se chová jinak tabelátor.

Velmi žádaná byla přednáška pana Petra Koláře o `sendmailu`. V úvodu byly zmíněny obecné principy transportu elektronické pošty a historické souvislosti. Pak se přednášející zaměřil na `sendmail` a jeho konfiguraci. Dnes se bohužel vyskytuje stále více „spamů“ a administrátorům nezbývá nic jiného, než proti tomu bojovat. Proto považuji za důležitou tu část přednášky, kde jsme se dozvěděli principy napadení počítače spamem prostřednictvím SMTP protokolu a možnosti ochrany e-mailového serveru proti takovým lumpům.

Po obědě vystoupil se svou prezentací podruhé pan Petr Staněk. Dnes se nejednalo o hry, ale o možnosti konfigurace window manageru. Hovořilo se především o `fvwm2`, což je zřejmě jakási modifikace mě dobře známého `fvwm`. Přiznám se, že jsem nepostřehl příliš podstatný rozdíl. Věc se lišila snad jen v tom, že horní lišty okének nebyly jednobarevné, ale duhové. Také nabídka, která se vynoří po šlápnutí myší na pozadí, neobsahovala pouze texty, ale byla ošperkována ikonkami. Myslím si, že jsem nenašel důvod přecházet na `fvwm2`. Je ovšem docela možné, že jsem něco ne zcela dobře pochopil.

Mezi druhou a čtvrtou hodinou odpolední byl vymezen prostor pro přednášku Michala Mühlpachra o Pythonu. Pochopil jsem, že to je interpret podobný Perlu, ale mezi Pythonem a Perlem vládne určitá rivalita. Jazyk je postaven na objektovém programování a vede programátora ke tvorbě přehledných kódů. Měl by též být snadný k naučení. Kostra přednášky byla promítnuta na plátno Netscapem jako WWW stránka, přičemž přednášející postupně posunoval výřez stránky směrem dolů. Mě osobně přepadl během této přednášky značný duševní útlum. Nechci se přednášejícího nijak dotknout, ale na takový monotónní přednes přesně krok po kroku podle textu na WWW stránce jsem nebyl psychicky připraven. Docela mi to připomínalo komerční prezentace a nikoli přednášky běžné na akademické půdě. Díval jsem se na rolovníček po pravé straně Netscapu, který se dostal do první třetiny stránky, když mě začal přepadat spánek. Musel jsem situaci řešit. Prošel jsem se tedy po vesnici Cikháj a obešel všech pět chalup této vísky dokola. Bylo krásné slunné odpoledne. Když jsem se vrátil, rolovníček byl ve dvou třetinách…

O přestávce proběhlo losování cen, které věnovali Linuxovému semináři štědří sponzoři. INET, a.s. (1) věnoval poukázky na nákup na serveru Virtuální Obchodní Dům!® (2) v cenách 100, 200 a 300 Kč. Nakladatelství Computer Press Brno (3) věnovalo hodnotné knihy (Používáme Linux, PGP, Apache – správa webového serveru, Konfigurace a správa TCP/IP sítě, Programování v Perlu) a předplatné časopisu CONNECT!, společnost IPEX (4) věnovala 6CD Infomagic Developers Resource. Ondřej „Koala“ Vácha věnoval anglický originál jedné z výše zmíněných knížek – Programming in Perl. Ceny byly vystaveny během semináře a nyní došlo k losování podle kopií vizitek všech přítomných. U losování sice nebyl přítomen notář, ale všichni účastníci uznali výherce za právoplatné. Výherci si vybrali ceny podle pořadí. Mezi vylosovanými byl i pan Můhlpachr, propagátor Pythonu. Přestože mezi knihami už nebylo skoro nic jiného, než knížky o Perlu, nevybral si je.

Dokonale mě probudila svěží přednáška posledního vystupujícího pana Martina Mareše. Vrátil se ke klasickým vyjadřovacím prostředkům, které mám nejraději – ke křídě a tabuli. S obrovským nadhledem (ale přesto velmi srozumitelně) nás seznámil se základními koncepčními principy jádra Linuxu včetně historického vývoje. K dobru přišly i nejrůznější historky programátorů jádra (mezi které Martin Mareš patří) a pozadí vývoje linuxového kódu. Dozvěděli jsme se například, že kód jádra je velmi málo komentován.

> Opravdový programátor ze zásady nepíše komentáře – když to bylo těžké napsat, musí to být těžké i přečíst.
> Martin Mareš ve své přednášce

Na závěr se otevřela dlouhá diskuse, ve které se posluchači ptali přítomného odborníka na nejrůznější fajnovosti jádra operačního systému. Bohužel, tato zajímavá diskuse taky musela skončit, protože v šest hodin večer odjížděl z Cikháje poslední autobus a záhy nato i ostatní mobilnější účastníci semináře museli ukončit sezení a uvolnit ubytovací zařízení. Vzájemně jsme se rozloučili slovy „ahoj, uvidíme se zítra na elektronické konferenci“. Mám dojem, že většina účastníků odjížděla ze setkání spokojena a obohacena o mnohé nové poznatky.

Závěrem bych chtěl připomenout, že od okamžiku, kdy byl seminář ohlášen na síti, do doby, kdy byla zcela zaplněna kapacita ubytovacího zařízení závazně přihlášenými zájemci, uplynula historicky velmi krátká doba. Znám lidi, kteří se chvíli rozmýšleli a pak už bylo pozdě. Takový velký zájem o akci samozřejmě příjemně překvapil pořadatele. Přirozeně to znamená, že by určitě nebylo špatné podobnou akci zopakovat. Při rozhovoru na toto téma se znavenými pořadateli se zatím nevyvrbil žádný konkrétní termín. Shodli jsme se na tom, že zopakování akce není špatný nápad, ale zatím chybějí okamžité síly se ujmout organizace. Je potřeba si nejprve odpočinout, nadechnout a až uplyne nějaký čas, pak se k otázce o dalším semináři opět vrátit… ■

1 INET, a.s.
`http://www.inet.cz/`
2 Virtuální Obchodní Dům!®
`http://www.SHOP.cz`
3 Computer Press Brno
`http://www.cpress.cz`
4 IPEX, s. r. o.
`http://www.ipex.cz`

Jak jsem vyráběl prezentaci

Jan „Yenya“ Kasprzak, 15. února

Účastníci Linuxového semináře na Cikháji měli možnost vyslechnout moji přednášku na téma „Firewall pod Linuxem“. Slajdy, které jsem tam promítal, jsou k dispozici i na Internetu (1). Již na semináři jsem byl několikrát dotazován, jaké nástroje byly použity k výrobě těchto obrázků. A právě této otázce se budu věnovat v článku, který právě čtete.

Promítání

Začnu od konce – od výsledného efektu. Výsledkem mého snažení byly běžné bitmapové obrázky ve formátu PNG (*Portable Network Graphics* (2)). Tyto obrázky jsem promítal na monitor a tedy i na plátno prohlížečem `xv`. Tento program je pro promítání obrázků velmi vhodný. Umí zobrazovat postupně několik obrázků, jejichž jména může uživatel specifikovat například z příkazové řádky. Takže moje prezentace se spouštěla následujícím příkazem:

```
$ xv stranka*.png
```

Prohlížeč `xv` lze ovládat z klávesnice – pro prezentace postačí vědět, že mezerníkem se zobrazí další obrázek a pomocí BackSpace předchozí obrázek. Dalším kladem tohoto programu je, že nemusíte složitě laborovat s umístěním okna – jsou-li obrázky stejně velké jako obrazovka, `xv` je automaticky zobrazí přes celou velikost obrazovky s tím, že horní lišta okna ani rámečky nejsou viditelné.

Sazba textu

Zpět na začátek – k vlastnímu vytváření textu. Text jsem sázel v TEXu s formátem `eplain` Karla Berryho (doporučuji vaší pozornosti – tento formát mírně rozšiřuje `plain` TEX s tím, že na rozdíl od LATEXu nevnucuje uživateli žádný konkrétní styl sazby). Jako písmo jsem použil font Charter od firmy Bitstream, které je dostupné volně v každé distribuci XFree86. Počeštění této rodiny písem (stejně jako fontu IBM Courier, použitého pro strojopisné texty) jsem provedl balíkem `fontinst` a spoustou ruční práce. Obrázky jsem kreslil programem `XFig` a importoval do dvi souboru jako encapsulated PostScript.

Výsledné slajdy jsou tvaru *landscape* tedy „naležato“. Toho jsem dosáhl tak, že jsem nastavil `\vsize=5in`. Významný text slajdu tedy leží v horní polovině stránky A4.

Od TEXu k bitmapě

Významným meziproduktem na cestě k výsledné prezentaci jsou soubory, obsahující ve formě polotónového (černo-šedo-bílého) obrázku jednotlivé stránky. Napsal jsem si krátký shellový skript, který realizuje převod z DVI do těchto obrázků:

```
#!/bin/sh
dvips -t a4 -f cikhaj98 | \
  gs -dSAFER -sDEVICE=pgmraw -r400x400 -q \
  -dNOPAUSE -sOutputFile=stranka%03d.pgm -
for i in stranka*.pgm; do
  pnmcut 80 0 3200 2400 $i >$i.cropped
  pnmscale -xscale 0.25 -yscale 0.25 \
  $i.cropped >$i
  rm $i.cropped
done
```

Tedy pomocí `dvips` převedu prezentaci do PostScriptu, pomocí GhostScriptu převedu jednotlivé stránky do `pgm` (v rozlišení čtyřikrát větším, než budu dále potřebovat), výsledný `pgm` soubor ořížnu na požadované rozměry programem `pnmcut` a pomocí `pnmscale` zmenším na čtvrtinu (tedy 800 krát 600 bodů) a získám tím i antialiasing (novější GhostScript umí antialiasing sám, ale tehdy jsem ještě tuto verzi neměl nainstalovanou).

Barevná magie – GIMP na scéně

Nyní už zbývá takto vytvořené polotónové obrázky vzít, nabarvit text na žluto, doplnit k textu černý, trochu neostrý stín, pod nadpis vložit vodorovnou čáru se žluto-červeno-žlutým barevným přechodem a stínem ve stejném stylu a pod to všechno vložit pozadí, přecházející shora dolů od světle modré do tmavě modré.

Udělat zmíněné není žádný problém – stačí k tomu libovolný alespoň trochu slušný editor bitmapových obrázků (z komerčních vzpomenu například Adobe Photoshop). Ovšem udělat tohle pro sedmadvacet slajdů pokaždé stejně, to už jen tak jednoduché není. Pominu-li možnost dělat to sedmadvacetkrát ručně, což může napadnout asi jen notorického uživatele předpředloňských Windows, zbývá najít způsob, jak výše popsané operace realizovat dávkově.

Jednou z možností je balík `NetPBM`, jehož součásti – programy `pnmcut` a `pnmscale` jsme viděli v akci výše. Nakonec jsem se ale rozhodl, že tento balík není pro moje účely dostačující. Další alternativou je program Gimp (3). Pokud jste snad o Gimpu ještě neslyšeli: jde o velmi kvalitní rozšiřitelný program pro manipulaci s obrázky. Podívejte se na WWW stránku Gimpu, tam je uvedeno více. Na tomto místě zmíním jen pár zajímavostí: v Gimpu byl mimo jiné nakreslen známý linuxový tučňák (4). Vedlejším produktem vývoje Gimpu je i toolkit gtk+, jeden z nejlepších toolkitů pro X11 (na gtk+ je založeno i uživatelské objektové rozhrani GNOME (5)).

Program Gimp má tu skvělou vlastnost, že jej lze programovat. Jednak lze dopisovat různé filtry a grafické efekty pomocí mechanismu *plug-inů*, a jednak každou operaci, kterou v Gimpu může uživatel udělat vyvoláním funkce menu nebo stiskem tlačítka myši, lze též zavolat jako proceduru interního skriptovacího jazyka. Implicitně má Gimp vestavěný skriptovací jazyk s názvem *script-fu*. Tento jazyk je variantou funkcionálního programovacího jazyka Scheme (pro ty, kdo vůbec neví o co jde: je to vzdáleně podobné LISPu, který znáte třeba z Emacsu, a který se zejména vyznačuje na můj vkus nadměrným množstvím kulatých závorek). Docela by mě zajímalo, proč autoři Gimpu zvolili tento jazyk, když na Internetu roste velké množství již hotových přilinkovatelných (*embeddable*) interpreterů (například Perl nebo Tcl) s rozumnějšími jazyky, než je Scheme.

Naštěstí podobný vztah ke script-fu má více lidí. Marc Lehmann (6) napsal modul pro Perl, který zpřístupní interní funkce Gimpu perlovému skriptu. Tento modul lze nalézt na libovolném zrcadle CPANu, například na adrese (7). Tento modul se standardním způsobem zkompiluje a nainstaluje. Dále je nutno vzít program Net-Server z tohoto modulu a nakopírovat jej do adresáře, kde Gimp hledá plug-iny (například do `~/.gimp/plug-ins/`).

Net Server – Gimp po síti

Plug-in Net-Server je po startu Gimpu dostupný z menu Extensions. Tento plug-in po spuštění vytvoří UNIXový socket v adresáři `/tmp` a přes tento socket mu perlovské skripty mohou zadávat příkazy. Tyto příkazy jsou pak předány Gimpu, který je vykoná. Takto můžeme mít někde spuštěný Gimp a z jiného okna spouštět jednotlivé skripty. Nebudu uvádět celý skript, který jsem používal pro vytváření prezentace. Skript najdete v adresáři s Linuxovými novinami.

Je vidět, že každý takový skript musí obsahovat podprogram `net`, který je vykonáván Net-Serverem. Dále může obsahovat podprogram `query`, který slouží k psaní plug-inů v Perlu. Tento podprogram volá Gimp při startu, pokud je script v adresáři `plug-ins`. Dále skript musí obsahovat zavolání jádra Gimpu – rutiny `gimp_main`.

Závěr

Tento článek má sloužit jednak jako odpověď na dotazy typu „Jak jsi to udělal?“ ohledně mé přednášky na Cikháji, a jednak jako upozornění na Gimp – poměrně silný program pro editaci a tvorbu grafiky s vlastním skriptovacím prostředím. Myslím, že Gimp si zaslouží samostatný článek v tomto časopise, ne-li přímo seriál článků. Ještě uvidím, co se s tím dá dělat… ■

1 Slidy z Linuxového semináře
http://www.linux.cz/cikhaj/yenya
2 Portable Network Graphics
http://www.cdrom.com/pub/png/
3 Gimp
http://www.gimp.org
4 Linuxový tučňák
http://www.isc.tamu.edu/~lewing/linux/
5 GNOME
http://www.gnome.org
6 Marc Lehman
mailto:pcg@goof.com
7 Perlový modul pro Gimp
ftp://ftp.fi.muni.cz/pub/perl/modules/by-module/GIMP

Programujeme v C s libproc

Karel Žák, 10. dubna 1998

V následujícím článku se pokusím vás seznámit s jednou velmi jednoduchou knihovnou pro vytváření programů pracujících s daty o procesech a systému. Často je slyšet hlasy, které tvrdí, že céčko je složité a v jiných programovacích jazycích lze programovat daleko rychleji atd. Osobně jsem toho názoru, že obtížnost, s jakou lze naprogramovat to či ono, je více než vlastním jazykem dána existencí knihoven, které řeší obtížné nebo (na naprogramování) zdlouhavé úkoly za programátora. Jednou takovou (řešící) knihovnou je i libproc.

O libproc se stará v současné době Helmut Geye (1). Knihovna je ve většině distribucí ve verzi 1.2.6. Dále je dostupná i experimentální verze této knihovny a to verze 1.12.2. Osobně se mi líbí více 1.12.2. Ta obsahuje daleko více maker a i po stránce množství dat o jednotlivých procesech je daleko obsažnější. Doporučuji dát si velký pozor na rozdílnost těchto knihoven při psaní programů. Stejnými verzemi jako knihovny jsou značeny i programy `ps`, `top`, `uptime`, `pstree`, `free` a to proto, že vše je distribuováno v balíku procps. Zajímavostí může být, že v Debianu v unstable distribution je verze 1.2.6 a ve stable distribution je verze 1.12.2. Vývojářské knihovny jsou v balíku libproc-dev odpovídající verze.

Ale zpět k vlastnímu programování. Začneme tím jednodušším. Pokusíme se udělat prográmek, který nám vypíše na standardní výstup aktuální uptime a obsazení paměti a swapu našeho počítače. Pokud bychom tento úkol chtěli splnit bez libproc znamenalo by to přečíst soubory `/proc/uptime` a `/proc/meminfo` a obojí analyzovat (a to v případě uptimu není zas taková banalita). Řešení pomocí libproc je uvedeno v příkladu Použití knihovny libproc. Knihovna umožňuje u uptimu přímo tisk uptime stringu na standardní výstup a to pomocí `print_uptime(void)`, takto je ostatně udělán i program `uptime(1)`. S údaji o paměti je to trochu složitější. Zde je pomocí funkce `meminfo` vráceno pole typu `unsigned (**mem)`. To obsahuje dvě řádky: `meminfo_main` a `meminfo_swap`. A každá řádka několik sloupců: `meminfo_free`, `meminfo_total` atd. Kombinací sloupců a řádek se lze dobrat jednotlivých údajů. Ty jsou v našem případě ještě před vypsáním převedeny z b na kB (bitová operace >> 10). A to je vše na cca 30 řádcích. Snad ještě poznámka pro ty co by nevěděli jak kompilovat:

```
gcc -Wall -O2 prog.c -o prog -lproc
```

Teď o něco těžší prográmek, který zjistí jaký proces používá nejvíce rezistentní paměti (tedy té, která není swapována) a jaký proces nejvíce zatěžuje procesor. Řešení je uvedeno na výpise Použití knihovny libproc podruhé. Na počátku je nutné nastavit verzi Linuxu (`set_linux_version()`), protože pokud to neuděláte, můžete se dočkat i poněkud zkreslených údajů (mě se to stalo např. u `ttyc`). Pak je nutné nastavit časové údaje,

```
/* ----- příklad 1. ----- */
#include <stdio.h>
#include <proc/sysinfo.h>       /* meminfo()      */
#include <proc/whattime.h>      /* print_uptime() */
#define RE_ERROR        1
#define RE_OK           0
   int main()
   {
      unsigned **mem;
      puts("\nUPTIME:");
      print_uptime();
      if (!(mem = meminfo()))
         exit(RE_ERROR);
      puts("\nSYSTEM MEMORY (kB):");
      printf("\tTotal  : %u\n",  (mem[meminfo_main][meminfo_total]   >> 10));
      printf("\tFree   : %u\n",  (mem[meminfo_main][meminfo_free]    >> 10));
      printf("\tUsed   : %u\n",  (mem[meminfo_main][meminfo_used]    >> 10));
      printf("\tShared : %u\n",  (mem[meminfo_main][meminfo_shared]  >> 10));
      printf("\tBuffers: %u\n",  (mem[meminfo_main][meminfo_buffers] >> 10));
      printf("\tCached : %u\n",  (mem[meminfo_main][meminfo_cached]  >> 10));
      puts("\nSYSTEM SWAP (kB):");
      printf("\tTotal  : %u\n",  (mem[meminfo_swap][meminfo_total]   >> 10));
      printf("\tFree   : %u\n",  (mem[meminfo_swap][meminfo_free]    >> 10));
      printf("\tUsed   : %u\n\n",(mem[meminfo_swap][meminfo_used]    >> 10));
      exit(RE_OK);
   }
```

Výpis č. 2: Použití knihovny libproc

které budeme potřebovat pro výpočet zatížení procesoru. Zde se pozastavím – funkce `get_pcpu()` je původní funkce, kterou používá program `ps(1)` po výpočet %CPU (proto ty podivné proměnné…). Tato funkce umožňuje buď počítat zatížení CPU pro vlastní proces nebo i včetně potomků (*cumulative*). Proto i možný parametr `-c` u našeho programu. A pak už následuje vlastní načtení dat o procesech. Libproc umožňuje načíst informace buď o jednom procesu (`readproc()`) nebo pro více procesů (`readproctab()`). Informace o procesech jsou těmito funkcemi vráceny v pointeru na strukturu typu `proc_t`, ta je definována a okomentována v hlavičkové souboru `readproc.h`. Struktura obsahuje asi 60 různých údajů o daném procesu a to od UID (verze 1.12.2 rozlišuje několik typů UID, GID, USER, GROUP) až po takové věci jako detailní rozlišení kolik paměti je použito knihovnami procesu, kolik je sdíleno, jaký je paměťový limit procesu a mnoho a mnoho dalšího. Nahlédnou do `readproc.h` nebo alespoň do manuálu k `/proc` doporučuji i (ne)programátorům, budete příjemně překvapeni kolik mnoho se lze o procesech dozvědět (zde má jistý komerčně „úspěšný" operační systém ještě co dohánět). Pokud získáváme data pomocí `readproctab()` je nám vráceno pole pointerů na struktury, kde jedna řádka pole je jedna struktura typu `proc_t` a reprezentuje právě jeden proces – poslední řádka pole je pointer `NULL`. Funkce `readproc()` a `readproctab()` umožňují velice efektivně definovat co o procesu chceme načíst a je možné i definovat filtr, podle kterého jsou vybrány jen naší podmínce vyhovující procesy. (Toto je hlavně u verze 1.12.2.) Více už makra `PROC_NĚCO` v souboru `readproc.h`. Funkce `readproctab()` je vlastně celé kouzlo našeho programu a asi i nejdůležitější částí `libproc`. Nyní už je jen naší starostí co s daty, které nám tato funkce vrátila. (A vše se v podstatě odbylo na jednom řádku – a pak že je céčko složité).

Ještě malé upozornění pro ty co by rádi něco podnikali s libproc. U polí jako např. `**cmdline` je dobré před započetím práce testovat jestli vůbec něco obsahuje. A testovat také délku položek v něm uložených. Ona příkazová řádka některých procesů by mohla váš program odeslat kamsi do `core`…

Pochopitelně výše uvedené je jen částí toho co dovede libproc. Dostupné jsou i funkce převádějící čísla zařízení do stringů (`devname.h`), speciální outputy (`output.h`), zacházení s „psdatabází" (`psdata.h`), jména signálů (`signames.h`), vytváření stromu procesů dle vztahu rodič – dítě (`tree.h`) atd.

Aby nikdo nemusel opisovat nebo jinak složitě získávat uvedené příklady (pokud by si je chtěl vyzkoušet), může je nalézt na FTP serverech Linuxových novin nebo na adrese (2).

Všem zájemcům lze jen doporučit nainstalovat balík libproc-dev a začít samostudium s knihovnou, která dokáže usnadnit programátorský život všem, kteří ji dokáží efektivně využít. ■

1 Helmut Geye
mailto:Helmut.Geyer@iwr.uni-heidelberg.de
2 Zdrojové texty příkladů
http://home.zf.jcu.cz/~zakkr/LN/

Emacs? Help! (3.část)

Milan Zamazal, 7. dubna 1998

Hledáme program

Může se stát, že potřebujete provést nějakou akci, ale nemůžete nikde v info dokumentaci najít žádnou funkci, která by ji implementovala. To ještě neznamená, že patřičná

```
/* ----- příklad 2. ----- */
#include <stdio.h>
#include <string.h>             /* strcmp()            */
#include <proc/readproc.h>      /* readproctab         */
#include <proc/version.h>       /* set_linux_version() */
#include <proc/sysinfo.h>       /* uptime()            */
#include <sys/param.h>  /* HZ                  */
#include <time.h>               /* time()              */
#define RE_ERROR        1
#define RE_OK           0
   int  GL_current_time, CL_Sum;
   long GL_time_now;
/* This function is from ps(1) - ps.c
 * Copyright (c) 1992 Branko Lankester
 * Changes Copyright (C) 1993, 1994 Michael K. Johnson,
 *   and   Copyright (C) 1995, 1996 Charles Blake,
 *   and   Copyright (C) 1996       Helmut Geyer
 */
   void get_pcpu(proc_t *p) {
      int total_time, seconds;
      time_t start;
      unsigned int pcpu=0;
      seconds = (((GL_current_time * HZ) - p->start_time) / HZ);
      start = GL_time_now - seconds;
      total_time = (p->utime + p->stime  +
                   (CL_Sum ?   p->cutime + p->cstime : 0));
      pcpu = seconds ? (total_time * 10 * 100/HZ) / seconds : 0;
      if (pcpu > 999) pcpu = 999;
      p->pcpu = pcpu;
   }
   int main(int argc, char **argv)
   {
      proc_t    **pt;
      int       proc_num=0, mem_max=0, cpu_max=0;
      if (argc > 1)
         if (!(strcmp(argv[1], "-c")))
            CL_Sum = 1;
      set_linux_version();
      if (!(GL_current_time = uptime(0,0)))
         exit(RE_ERROR);
      GL_time_now = time(0L);
      pt = readproctab(PROC_FILLTTY | PROC_FILLUSR | PROC_FILLMEM);
      while(pt[proc_num] != NULL) {
      /* --- mem --- */
         if (pt[proc_num]->resident > pt[mem_max]->resident)
            mem_max = proc_num;
      /* --- cpu --- */
         get_pcpu(pt[proc_num]);
         if (pt[proc_num]->pcpu > pt[cpu_max]->pcpu)
            cpu_max = proc_num;
         ++proc_num;
      }
```

Výpis č. 3: Použití knihovny libproc podruhé

funkce neexistuje, pouze se musíme poohlédnout někde jinde, pravděpodobně ji bude implementovat nějaký elispovský program.

První, co byste měli vyzkoušet, je *Finder*. Vyvoláte jej příkazem `C-h p`, jenž vám zobrazí seznam tematických oblastí podle klíčových slov. Každá z těchto oblastí obsahuje seznam programů z distribuce Emacsu, které mají s daným tématem co společného. Pro každý takový program si můžete nechat zobrazit stručnou informaci o něm. Avšak pozor, ne všechny programy z distribuce Emacsu naleznete i ve Finderu, jejich přítomnost zde závisí na zvolených klíčových slovech.

Pokud jste svůj program ve Finderu nenašli, zkuste se podívat do FAQ (`C-h F`). Zobrazená verze FAQ odpovídá době vypuštění vaší verze Emacsu, novější verze se občas objevují v diskusní skupině *gnu.emacs.help*. Po FAQ lze pohodlněji procházet po provedení `C-u 2 C-x $` (do normálního režimu se vrátíte příkazem `C-x $`). FAQ by mě-

```
    printf("\nPROCESS  : %d\n", proc_num);
    puts("PROCES WITH MAX. RESIDENT MEMORY:");
    printf("\tName     : %s\n",        pt[mem_max]->cmd);
    printf("\tPID      : %d\n",        pt[mem_max]->pid);
    printf("\tSTATE    : %c\n",        pt[mem_max]->state);
    printf("\tUSER     : %s\n",        pt[mem_max]->user);
    printf("\tTTY      : %s\n",        pt[mem_max]->ttyc);
    printf("\tRESIDENT : %ld (pages)\n", pt[mem_max]->resident);
    puts("PROCES WITH MAX. %CPU:");
    printf("\tName     : %s\n",        pt[cpu_max]->cmd);
    printf("\tPID      : %d\n",        pt[cpu_max]->pid);
    printf("\tSTATE    : %c\n",        pt[cpu_max]->state);
    printf("\tUSER     : %s\n",        pt[cpu_max]->user);
    printf("\tTTY      : %s\n",        pt[cpu_max]->ttyc);
    printf("\t%%CPU     : %.1f", (float) pt[cpu_max]->pcpu/10);
    if (CL_Sum)
       puts(" (cumulative)\n");
    else
       puts("\n");
    exit(RE_OK);
 }
```

Výpis č. 4: Použití knihovny libproc podruhé

ly být nejenom zdrojem odkazů na externí programy, ale měli byste do nich nahlédnout vždy, než se budete na něco ptát svého kolegy nebo v některé diskusní skupině.

Máte-li i po prozkoumání FAQ pocit, že nevíte, kde svůj program hledat, znamená to, že jste je zkoumali málo. Jednou z otázek, na které zde naleznete odpověď, je totiž „How do I find a GNU Emacs Lisp package that does XXX?“. V odpovědi se dozvíte o existenci LCD (*Lisp Code Directory*) archívu různých elispovských prográmků. Dozvíte se také, jak tento archív ze svého počítače prohledávat pomocí LCD apropos.

Nepříjemným problémem LCD archívu je to, že jenom poměrně málo autorů elispovských programů svá díla do tohoto archívu zasílá. Navíc za dobu jeho existence se v něm nakupila řada programů, které již nejsou dále udržovány a tudíž nevyužívají možností nových verzí Emacsu. Tímto nedostatkem naopak netrpí newsová skupina *gnu.emacs.sources*. Ta je autory různých balíků a pomůcek využívána daleko častěji než LCD archív a vyplatí se ji sledovat. Pozor, tato skupina je určena *výhradně* pro zdrojové texty a posílat do ní příspěvky neobsahující žádný užitečný elispovský kód nebo patch je obecně velmi nevhodné. Pro dotazy typu „Kde najdu XXX?“ slouží skupina *gnu.emacs.help*.

Poměrně pozoruhodný archív elispovských prográmků jsou *tiny tools*. Jedná se o několik megabytů kódu, který vlastnoručně napsal Jari Aalto. Vždy, když si autor potřebuje v Emacsu něco vylepšit nebo do něj něco přidat, vytvoří nebo rozšíří nějakou tiny pomůcku a umístí ji do tohoto archívu. Je tedy poměrně pravděpodobné, že zde naleznete řadu prográmků, které se vám mohou hodit. Tiny tools jsou k mání na adrese (1).

Jestliže jste program implementující hledanou funkci stále nenašli, zbývá (kromě použití webovských vyhledávacích strojů) už jen jediné – zeptat se v newsech. Emacsu jsou věnovány zejména tyto newsové skupiny:

- *comp.emacs*: všechno možné o Emacsu
- *gnu.emacs.help*: sem byste měli primárně směrovat své emacsovské dotazy
- *gnu.emacs.gnus*: skupina věnovaná Gnus, na ty se ptejte přednostně zde a ne v *gnu.emacs.help*
- *gnu.emacs.sources*: již zmíněná skupina vyhrazená pro zdrojové texty
- dále několik dalších diskusních skupin věnovaných například XEmacsu; nalézt je lze snadno hledáním řetězce „emacs“ v seznamu všech newsových skupin
- *cz.comp.editors*: česká diskusní skupina o editorech a Emacsu

Většinu těchto odkazů a několik dalších informací naleznete na webové GNU stránce o Emacsu, která se nachází na adrese (2).

Přechod z vi

Emacs samozřejmě myslí i na ty uživatele, kteří se v mládí omylem naučili (nebo byli naučeni) pracovat s editorem `vi`. Vzhledem k nekompatibilnímu ovládání `vi` si teď netroufají přejít na Emacs, protože mají obavy, že by v něm nemohli bez důkladné přípravy pracovat. Tyto obavy jsou však naprosto neopodstatněné. Emacs disponuje pětistupňovou emulací `vi` (zvanou *Viper*) od skoro `vi` po skoro Emacs. Ostřílení víajisté tak mají na počátku k dispozici prostředí s klávesovým ovládáním, na které jsou zvyklí, a nemusí být zbytečně mateni množstvím různých funkcí. Součástí Viperu je i info návod vysvětlující mimo jiné některé pro víajisty nezvyklé vlastnosti Emacsu.

Do Viperu se dostanete spuštěním Emacsu a provedením příkazu `M-x viper-mode`. Zodpovíte-li několik otázek, které vám budou při prvním použití tohoto příkazu položeny, může se vám Viper příště startovat již automaticky.

Přestože Viper velmi dobře umožňuje ovládat Emacs ve stylu `vi`, je lépe co nejdříve přejít na skutečné emacsovské ovládání. Jak jsme již nahlédli, Emacs má své standardy pro klávesové kombinace, které jsou respektovány většinou rozšiřujících programů. Používáte-li základní ovládání ve stylu `vi`, ztrácíte do značné míry výhody konzistentního systému klávesových kombinací a některé funkce dokonce vůbec nemusí být klávesovými zkratkami dostupné, protože mohou být předefinovány vi příkazy. To samé samozřejmě platí i pro emulace jiných editorů.

O čem ještě nebyla řeč

Vše, co zde bylo řečeno, není nic, co by se muselo dlouho hledat a studovat. Vše naleznete v menu Help nebo prostřednictvím příkazu `C-h C-h`.

Nápověda Emacsu myslí i na problematiku nových verzí. Uživatelský manuál je sice s každou novou verzí Emacsu patřičně aktualizován, ale to příliš nepomůže uživatelům, kteří nechtějí znovu číst celý manuál a místo toho by rádi viděli seznam změn od poslední verze. Všechny viditelné změny ve funkcích (v Emacsu i v Elispu) jsou již dlouho odpovídajícím způsobem dokumentovány. Seznam změn v posledních verzích si zobrazíte příkazem `C-h n`. Seznamy změn ve starých verzích jsou pak k dispozici v adresáři .../etc/ v souborech O*NEWS. Seznam změn by měli sledovat zejména programátoři v Elispu, protože referenční příručka Elispu není updatována s každou verzí Emacsu (zatím poslední verze odpovídá Emacsu 19.34).

Již zmíněný adresář .../etc/ obsahuje i jiné zajímavé informace než seznam novinek. Upozorněme zejména na referenční kartu s nejpoužívanějšími klávesovými příkazy (refcard*, pro Viper vipcard*) a soubor PROBLEMS dokumentující známé potíže, kdy něco zdánlivě nebo doopravdy nefunguje, a možnosti jejich řešení. Dále zde najdete dokumenty o free softwaru a GNU, různé dokumenty věnované GNU politice a taktéž několik spisů poněkud lehčího charakteru.

Na závěr přiznejme, že se v Emacsu občas vyskytují chyby. Co udělat v případě, že narazíte na chybu? Rozhodně byste ji měli ohlásit. Nespoléhejte příliš na to, že tak učiní někdo jiný! Ohlášení chyby totiž mnohdy vyžaduje nezanedbatelné úsilí a řada uživatelů není ochotna ani tímto malým dílem přispět k vývoji svého programu. Odměnou za řádně ohlášenou chybu by vám měl být patch, který tuto chybu opraví. Před ohlášením chyby si však každopádně dobře přečtěte sekci Bugs v manuálu Emacsu. To je nutná podmínka k tomu, abyste vývojářům Emacsu byli schopni pomoci co nejlépe. Pokud víte, že se jedná o chybu elispovského souboru a který to je, podívejte se ve zdrojovém souboru na položku Maintainer v úvodním komentáři. Je-li zde uvedeno FSF, ohlašte chybu běžným způsobem. Pokud je tam naopak uvedena konkrétní osoba, pošlete oznámení o chybě se všemi náležitostmi přímo na její adresu.

Závěrem

Snad se podařilo ukázat, na jakých principech je dokumentační systém Emacsu vystavěn, co vše je v něm obsaženo a jakými různými způsoby jej lze využívat. Ač někteří lidé odkojení prostředími typu Visual Developer Studio pohlíží na Emacs s netajenou arogancí, nelze než konstatovat, že Emacs má i po stránce dokumentační před podobnými produkty několikaletý náskok. Nevítá vás sice po svém spuštění animacemi v barvách duhy, zmatenými kouzelníky a příběhy o zatýkání svých uživatelů, zato se vám však snaží po všech stránkách pomáhat. Proto není správné k rozsáhlému a sofistikovanému systému dokumentace Emacsu přistupovat se strachem, nýbrž je lépe si jej ochočit.

> „Víš, který editor je nejlepší?"
> „Vim."
> „Ano, správně."
> Jan Šerák

Pokud se vám i po přečtení tohoto miniseriálu bude zdát, že systém nápovědy Emacsu je přeci jen příliš komplikovaný, nezoufejte. Stejně jako většina věcí v životě, i zvládnutí obsáhlé dokumentace Emacsu vyžaduje určitou praxi. Čím více budete trénovat, tím lépe si budete s dokumentačním systémem a Emacsem samotným rozumět. Nevěříte? Zkoušejte to! ■

1 Tiny Tools
ftp://cs.uta.fi/pub/ssjaaa/
2 GNU stránka o Emacsu
http://www.gnu.org/software/emacs/emacs.html

Kontrola integrity RPM balíků

Jan „Yenya" Kasprzak, 15. února

V předchozí části tohoto seriálu jsme viděli, jaké informace systém RPM udržuje o jednotlivých balících a jejich souborech. V minulém čísle jsme si dokonce ukazovali, jak tyto údaje ze systému získat. V tomto čísle bude řeč o kontrole integrity balíků – co dělat, chceme-li zjistit, jestli soubory, které se v systému nacházejí, mají vůbec ještě nějaký vztah k RPM balíku, ze kterého byly kdysi nainstalovány.

Soubory a verifikační skripty

K verifikaci RPM balíků slouží přepínač -V, resp. -y nebo též --verify příkazu rpm. Systém RPM umí kontrolovat jednak integritu souborů příslušných RPM balíkům, a jednak správnost konfigurace. První z kontrol se provádí oproti RPM databázi nebo .rpm-balíku samotnému, druhá je prováděna pomocí *verifikačního skriptu*, který může být součástí RPM balíku. Tam si autor balíku může sám napsat cokoli, co zkontroluje například syntaxi konfiguračních souborů (uživatelé systému UUCP si jistě vzpomenou na program uucheck). Poslední věcí, kterou RPM kontroluje, jsou závislosti mezi balíky.

Kontrola souborů

Následujících devět atributů se může u každého souboru v daném balíku kontrolovat:

- vlastník souboru
- skupina souboru
- přístupová práva
- kontrolní součet (metoda MD5)
- velikost souboru
- hlavní číslo zařízení
- vedlejší číslo zařízení
- cíl symbolického linku
- čas poslední modifikace

Ne všechny atributy jsou pochopitelně smysluplné u všech souborů – například cíl symbolického linku lze zřejmě kontrolovat pouze u souborů typu symbolický link. Taktéž hlavní a vedlejší číslo je platné pouze u speciálních souborů. Naopak MD5 součet je u speciálních souborů k ničemu. Navíc sám autor balíku má možnost zakázat verifikaci určitého atributu pro konkrétní soubor. Například terminálová zařízení mění svého vlastníka (nikoli však skupinu) podle toho, kdo je momentálně na daném terminálu přihlášen. Proto by bylo nemoudré verifikovat vlastníka takového souboru.

Nesouhlasí-li hodnota některého atributu ve skutečnosti a v databázi, RPM o tom vypíše zprávu. Pro každý změněný soubor je ve výstupu jeden řádek. Na řádku je seznam atributů, které nesouhlasí, případně písmeno c, jde-li o konfigurační soubor, a cesta k souboru. Seznam je uveden ve formě osmiznakového řetězce, kde na místě změněných atributů je písmeno, a na místě nezměněných nebo neverifikovaných atributů je tečka. Výstup může vypadat asi takto:

```
$ rpm -V bash
.M5....T c /etc/bashrc
```

Na místě teček mohou být tyto atributy (podle pořadí v seznamu):

- S – velikost
- M – přístupová práva
- 5 – MD5-součet
- D – hlavní a vedlejší číslo zařízení
- U – vlastník
- G – skupina
- T – čas poslední modifikace

Je vidět, že v našem příkladu byl soubor `bashrc` modifikován. Je změněn jeho obsah (MD5 součet), čas modifikace a přístupová práva. Pokud soubor, který se má verifikovat, vůbec neexistuje, je místo seznamu atributů vypsáno slovo `missing`. Je-li verifikace balíku úspěšná, RPM nevydává žádný výstup.

Co se má kontrolovat?

Jak jsme již viděli výše, stačí příkazu `rpm -V` dát jako parametr seznam balíků, které má zkontrolovat. Kromě toho existují tyto další možnosti:

- `-a` – kontrola všech balíků, které jsou právě v systému nainstalovány.
- `-f soubor` – kontrola balíku, kterému patří daný soubor.
- `-p soubor.rpm` – provádí kontrolu balíku nikoli proti RPM databázi, ale proti balíku samotnému. Toto se použije zejména v případě, kdy si nejsme úplně jisti platností RPM databáze nebo když databáze vůbec neexistuje.
- `-g skupina` – kontrola všech nainstalovaných balíků, patřících do dané skupiny (například `rpm -Vg Shells` zkontroluje všechny shelly).

Další možnosti

Kromě výše uvedených přepínačů lze také specifikovat, které části verifikačního procesu se mají provést. Jednotlivé části lze vypnout těmito volbami:

- `--nodeps` – neprovádí kontrolu závislostí mezi balíky.
- `--noscripts` – nespouští verifikační skript.
- `--nofiles` – neprovádí kontrolu

Za zmínku stojí ještě přepínače `-v`, resp. `-vv`, které zvyšují podrobnost vypisovaných informací (například zapínají výpis standardního vstupu verifikačního skriptu, resp. i jednotlivé příkazy, tímto skriptem prováděné).

Tento díl našeho seriálu o systému RPM uzavírá první část zaměřenou na běžné uživatele. Od příště se budeme věnovat vlastní tvorbě RPM balíků.

Zasmáli jsme se!

Pavel Janík ml., 11. dubna 1998

V minulém čísle jsme odhalili, proč je hračka Tamagotchi tak úspěšná. Dnes se o Tamagotchi dozvíme něco více…

> Víte jak se liší Windows a Tamagochi?
> Windows chcípnou, i když se o ně staráte…
> Petr Kubišta v konferenci jokes@satoya.cz

Petr Kubišta mi bohužel nechce prozradit, co má proti oné společnosti.

A protože máme duben (po slovensky apríl), nemůžeme opomenout prvoaprílové žertíky. Mezi ty nejzajímavější patří např. zpráva o tom, že se Donald. E. Knuth (autor TEXu) rozhodl po dvaceti letech svůj produkt zaprodat komerční společnosti. Této zprávě ještě nikdo neuvěřil, ale když se v *linux-kernel@vger.rutgers.edu* objevilo

> My friend, who is running 2.1.90 on a K6-200, 32Meg system got these errors. Anyone know what they mean? I suspect it is because he is overclocking his CPU.

```
Apr 1 00:34:35 kernel: Warning! /dev/fd0 \
    is a toaster.
Apr 1 00:34:37 kernel: Panic! /dev/null \
    is full!
Apr 1 00:34:57 kernel: Mounting \
    Audio CD as /
```

> Můj přítel, který běží 2.1.90 na K6-200/32 MB, objevil tyto chyby. Ví někdo, co to znamená? Předpokládám, že je to kvůli přetaktování jeho procesoru.

a také naprosto seriózní odpověď na položenou otázku, zcela jistě se čtenáři konference dobře zasmáli.

8. dubna, den před uveřejněním verze 2.1.94, si Jauder Ho – Linusův kolega – řekl, že se podívá, jaký má Linus počítač a kernel a výsledek poslal do konference *linux-kernel*:

```
cpu: (2) -266BM
mem: 512 MB
swapfree: 422 MB
```

```
vartmpfree: 2243 MB
host: Intel 82441FX Natoma (rev 2)
scsi: BusLogic BT-958 MultiMaster (rev 8)
disk: QUANTUM XP39100W
version: 2.1.94 (torvalds@penguin) \
    (gcc version 2.7.2.1) #3 SMP
```

Ale to neměl dělat, protože se ihned diskuze zvrhla na téma „Linus soukromničí!?“, případně „Dejte mu Pentium 100, ať vidí, jak jsou na tom ostatní“ apod. Linus zachoval chladnou hlavu, vysvětlil, že se nyní snaží neudělat stejnou chybu, kterou jsme popsali v minulém čísle Linuxových novin (Linus zapomněl zvednou číslo verze kernelu) a během několika minut byla nová verze na světě. Linus přitom napsal:

> I'll make a 94 available real soon now (ie within minutes), once I've rebooted it to test that it still works. If it compiles, it is good, if it boots up it is perfect.
>
> ---
>
> Verze 94 bude v brzké době (během několika minut), jenom co ji nabootuji a zkusím, jestli stále funguje. Pokud se přeloží, je to dobré, pokud nabootuje, je to perfektní.
> Linus Torvalds

A protože se ještě stále spousta lidí ptá, co to vlastně je to SMP, přikládáme vysvětlení:

> Once upon a time in the dim and distant past, people used to design computers which only had a single processor in them. Imagine the cruelty of locking a single penguin in a small box without even a playmate!
> "UP" refers to a "UniProcessor" computer, one of these cruel torture chambers. Do your penguin a favour – upgrade it to SMP, and let them play :)
>
> ---
>
> Bylo nebylo, kdysi v temné dávné minulosti, lidé vyráběli počítače, které měly jen jeden procesor. Představte si, jak kruté bylo zamknout samotinkého tučňáka v malé krabici. Vždyť si ani neměl s kým hrát!
> „UP“ znamená „UniProcessor“ (jednoprocesorový) počítač, tedy právě jednu z těchto krutých mučíren. Udělejte něco pro svého tučňáka – upgradujte na SMP [tedy více tučňáků] a nechte je, aby si hráli. :)
> David Woodhouse v linux-kernel

■

New dad (again). .

Pavel Janík, 19.dubna 1998

(Ne, já opravdu ještě nemám dítě (další)...) Možná někteří z vás zaznamenali ve svém mailboxu thread se stejným subjectem jako je název tohoto článku. Já tedy ano :–) Mail s tímto subjectem poslal Linus Torvalds do *linux-kernel@vger.rutgers.edu.*

```
Date:   Thu, 16 Apr 1998 19:37:34 -0700 (PDT)
From: Linus Torvalds <torvalds@transmeta.com>
To: Kernel Mailing List
Subject: New dad (again)..
My second daughter (aka "Linus v3.0") was
born today at 9:04AM, and both
mother and child are doing well.
A national holiday has been announced,
and I'll be even less
responsible^H^H^Hve than usual --
so please excuse
any lost emails and patches.
Weight: 3270g (7lb 3oz),
height: 52cm (20.5")
                Linus
```

Linusovi samozřejmě blahopřejeme a jeho dceři (zatím se mi nepodařilo zjistit, jak se bude jmenovat) také.

Ale nejen kvůli blahopřání jsem chtěl napsat tento článek. Linus ve svém dopise začal vtipkovat, ostatní se přidali...Já jsem se dobře bavil, doufám, že i vy se pobavíte :–)

Pierfrancesco Caci napsal:

```
Congratulations!
hmmm, I wonder... What will say my wife
if I tell her "let's have a
feature freeze tonight ?"
```

V konferenci také padl dotaz, kde že Linus na to vzal čas. Okamžitá odpověď zněla:

```
Just for Your information, it can be done
between compilations  :)
```

(No, když se podívám na konfiguraci Linusova stroje...) Někteří lidé dokonce pátrali, který kernel byl vydán přibližně před devíti měsíci. Byla to verze 2.1.44, která je nechvalně známá možností zničení ext2fs filesystému (viz soubor WARNING-2.1.44) ...Doporučuji všem, aby si tento thread přečetli celý. ■

Oprava

Leo Hadacz, 14. dubna 1998a

V minulém čísle Linuxových novin jste si mohli přečíst můj článek o tcp wrapperech. Byl jsem upozorněn na jednu chybičku, a tak bych rád uvedl vše na pravou míru. V článku jsem napsal, že slovo „wrapper“ se do češtiny překládá jako „přepínač“. Není to pravda. Skutečný překlad tohoto slova je (např.) „obal“. I tento význam je smysluplný. Wrapper je prostě bezpečnostní obal pro jiné programy.

(Pozn. redakce: v mém anglicko–českém slovníku je uveden překlad slova wrapper také jako „župan“ :–). ■

Linuxové noviny a jejich šíření

Linuxové noviny jsou otevřeny každému, kdo by chtěl našim čtenářům sdělit něco zajímavého. Příspěvky (ve formátu čistého textu v kódování ISO 8859-2, grafiku nejlépe ve formátu PNG) posílejte na adresu (1). Autor nemá nárok na finanční odměnu a souhlasí s podmínkami uvedenými v tomto odstavci. Vydavatelé si vyhrazují právo rozhodnout, zda Váš příspěvek uveřejní, či nikoli.

Chtěl bych poděkovat Fakultě informatiky Masarykovy university v Brně, INET, a.s., Juraji Bednárovi a společnosti OptiCom za poskytnutí diskového prostoru pro Linuxové noviny.

Linuxové noviny můžete najít na akademické síti CESNET (2), na síti IBM Global Network na adrese (3), na serveru časopisu Netáčik (4), který je připojen do slovenského SIXu, případně na serveru společnosti OptiCom (5). ■

1 Adresa redakce
`mailto:noviny@linux.cz`

2 Linuxové noviny na síti CESNET
`ftp://ftp.fi.muni.cz/pub/linux/local/noviny`

3 Linuxové noviny na síti IBM Global Network
`ftp://ftp.inet.cz/pub/People/Pavel.Janik/noviny`

4 Slovenské zrcadlo Linuxových novin
`ftp://netacik.sk/pub/linux/cz-noviny`

5 Linuxové noviny – OptiCom
`http://www.mathew.sk/noviny`