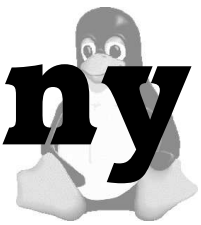
11/2000

Linuxové noviny



Úvodem

Pavel Janík ml., 12. listopadu 2000

Linuxové noviny opět vycházejí velmi, velmi opožděně. Tentokráte to nebylo chybou nedostatku článků, ale čistě lidským faktorem. Nebudu se tady však omlouvat nebo vychloubat získaným diplomem :-) Od toho nejsou Linuxové noviny.

Doufám, že Linuxové noviny budou opět vycházet častěji a že i Vy, naši čtenáři, k tomu přispějete. Pokud budete chtít cokoli zveřejnit, článek nebo recenzi apod., napište nám (noviny@linux.cz).

V úvodu bych chtěl také představit novou posilu v redakčním týmu Linuxových novin, kterou je David Häring. Jeho pozice je zástupce šéfredaktora. On to ještě neví, ale je to velmi nevděčná úloha. Pokud nevíte, co tato pozice obnáší, tak věřte, že je to velmi jednoduché: „Pokud šéfredaktor nemá čas nebo se mu moc nechce do dalšího čísla Linuxových novin, vezmi to za něj“.

■

Čeština v Linuxu

Vladimír Michl

Úvod

Na začátek je třeba říci něco o kódování češtiny. Existuje několik kódování, které lze pro češtinu a slovenštinu použít. Jsou to kódování Kamenických (cp895), cp852 (Latin 2), Windows-1250 (cp1250) a ISO-8859-2 (ISO Latin 2). Ve světě UNIXů a Internetu se používá kódování ISO-8859-2. Na převody mezi kódováními existuje několik utilit, vyzdvihnu ale dvě: **GNU recode** a **cstocs**. V rámci sjednocení nástrojů doporučuji (pokud je to možné) používat **GNU recode**. Například převod z Windows-1250 do ISO Latin 2:

```
recode windows-1250..l2 <soubor.txt >soubor-l2.txt
```

Konzole

Aby bylo jasné, co a jak nastavit pro to, aby fungovalo psaní českých písmenek na konzoli, je třeba udělat malý úvod, jak to funguje.

Ovladač konzole se skládá ze dvou částí, ovladače klávesnice a ovladače obrazovky. Je třeba vhodně nastavit oba, abychom byli spokojeni.

Klávesnice

Ovladač klávesnice umožňuje pracovat (mimo jiné) ve dvou režimech, a to osmibitový výstup a unicodový (UTF-8). Režim přepínáme pomocí příkazu `kbd_mode`. Ovladač je implicitně nastaven v osmibitovém režimu, který budeme používat i my. Ovladač klávesnice může být pro každou konzoli **nastaven zvlášť**.

Klávesové mapy zavádíme pomocí příkazu `loadkeys`. Pro český psací stroj s anglickými „z“ a „y“ to bude například:

```
loadkeys cz-lat2.map.gz
```

Dostupné klávesové mapy lze obvykle nalézt v adresáři `/usr/share/keymaps/i386/qwerty` nebo `/usr/lib/kbd/keymaps/i386/qwerty`.

K dispozici jsou ještě další klávesové mapy `qwerty/cz-lat2-prog` a `qwertz/cz-us-qwertz`. Klávesová mapa je **společná pro všechny konzole**. Mezi českou a anglickou klávesnicí se přepíná pomocí klávesy *Pause*.

Odkazy: **kbd_mode(1)**, **loadkeys(1)**, **keymaps(5)**

Obrazovka

Ovladač obrazovky lze nastavit do dvou módů: UTF-8 (Unicode) a do osmibitového. Zatím se obvykle používá osmibitový režim.

Vnitřně pracuje ovladač v Unicode, což nám teoreticky umožňuje zobrazit znaky všech abeced (samozřejmě v Unicode definovaných). Bohužel hardware nás omezuje, v případě VGA je to na 512 současně zobrazitelných různých znaků. Proto byl vymyšlen následující mechanismus.

Každý font může mít maximálně 512 znaků. S každým fontem se do jádra nahrává také tabulka, která říká, jakým znakem z fontu se mají zobrazovat znaky z Unicodu. Tato tabulka je v anglické literatuře označována jako Screen Font Map (SFM).

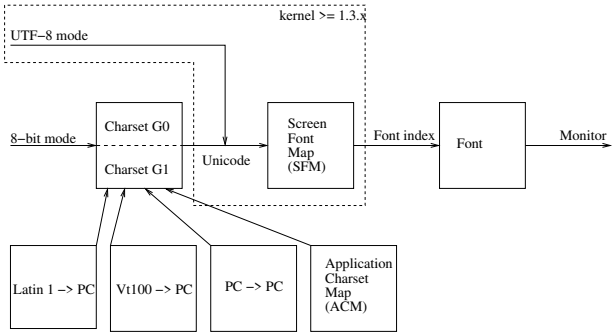
Pokud je konzole v UTF-8 režimu, převede se znak z UTF-8 na šestnáctibitový unicodový (UCS2) a pomocí SFM je stanoven znak z fontu, který má být zobrazen (viz obr. Schema konverzí znaků, při cestě z aplikace na obrazovku).

V případě osmibitového režimu je to trochu složitější. Osmibitový znak je podle aktivního charsetu (G0 nebo G1) překódován do Unicodu. Charset může být nastaven do čtyř režimů: Latin1 do PC, vt100 grafika do PC, PC do PC a uživatelsky definovaná tabulka ACM.

Tabulky SFM a ACM jsou **globální**, tj. společné pro všechny konzole, nastavení režimu (8bit, UTF-8), charsetu (která tabulka) a který charset (G0, G1), je pro **každou konzoli zvlášť**.

Pro nastavení klávesnice, fontu a tabulek (Maps) se používal balík **kbd**, který nastavoval klávesnici, font a pouze ACM, jelikož starší jádra měla pouze ACM tabulku. Později byl balík **kbd** nahrazen balíkem **console-tools**, který již umí zavést unicodové fonty a nastavit SFM.

Linuxové noviny připravuje a vydává České sdružení uživatelů operačního systému Linux



Obrázek č. 1: Schema konverzí znaků, při cestě z aplikace na obrazovku

Nastavení fontu:

```
consolechars --acm=iso02 --font=lat2u-16
```

případně `setfont`, používáte-li balík **kbd**. SFM tabulka je někdy (většinou) přibalena ve fontu, a proto ji není třeba zavádět zvlášť.

Pro každou konzoli zvlášť je třeba (například do `/etc/profile`):

```
echo -ne '\033%@'  # 8bitový režim
echo -ne '\017'    # Použití charsetu G0
echo -ne '\033(K'  # V G0 použít ACM
```

Odkazy: **consolechars(8)**, **console_codes(4)**, **unicode(7)**, **utf-8()**, **setfont(8)**, **mapscrn(8)**

Locales

Tato část systému se stará o definice chování prostředí v různých jazycích, a to definice znaků abecedy (co jsou malá písmena, co velká, převody z malých na velká a opačně), pravidla při abecedním řazení, definice zápisu numerických a peněžních údajů, umožnění komunikace programu ve více jazycích.

Locales jsou rozdělena do několika kategorií:

- LC_CTYPE – kategorie znaků, ovlivňuje rozpoznávání jednotlivých znaků programem (co je malé písmeno, co velké, jaké je malé písmeno k velkému apod.)
- LC_COLLATE – použita při porovnávání znaků, tj. abecedním řazení
- LC_TIME – použita při vstupu a výstupu časových údajů, definuje jejich formát.
- LC_NUMERIC – použita při vstupu a výstupu numerických údajů (čísel). Například pro zapnuté českého prostředí by se číselné údaje měli vypisovat následovně: 1 000 125,00
- LC_MONETARY – definuje výstup peněžních údajů, formát peněžních čísel, označení měny, kde má být označení umístěno a zda má být odděleno mezerou.
- LC_MESSAGES – v jakém jazyce bude program komunikovat (vypisovat zprávy).

Každou kategorii můžeme nastavit zvlášť a díky tomu můžeme program nastavit třeba tak, aby komunikoval v anglickém jazyce, ale texty, které budeme psát a upravovat, považoval za české a třeba i řadil slova podle českých pravidel.

Jednotlivé kategorie nastavujeme proměnnými prostředí, jejichž jména jsou shodná s názvy kategorií. Např:

```
export LC_CTYPE=cs_CZ.ISO-8859-2
```

Existují také proměnné, které ovlivňují globální nastavení. Jsou dvě:

- LC_ALL
- LANG

LC_ALL je proměnná s nejvyšší prioritou, pokud ji nastavíme, přebije všechna ostatní nastavení. Proměnná LANG má zase nejmenší prioritu. Pokud není nastavena proměnná LC_ALL nebo proměnná příslušné kategorie, použije se pro tuto kategorii hodnota z LANG.

Definiční soubory pro locales se obvykle nachází v `/usr/share/locale`, kde je pro každý jazyk vytvořen adresář, ve kterém pak jsou definiční soubory. Tyto soubory jsou v binárním tvaru. Zdrojové soubory locales pak jsou v adresáři `/usr/share/i18n`.

Jak proměnné nastavit

Hodnoty proměnných LC_* a LANG by se měly řídit podle pravidla:

> zkratkajazyka[_ZKRATKAZEMĚ[.kódování]]

přičemž pro český jazyk je zkratka „cs“ a zkratka České republiky je „CZ“. Pro slovenský jazyk je zkratka „sk“ a pro Slovenskou republiku „SK“. Proměnné tedy budeme nastavovat na „cs_CZ“ („sk_SK“) případně „cs_CZ.ISO-8859-2“ („sk_SK.ISO-8859-2“). Zkratky jazyka definuje norma ISO 639 a zkratky zemí definuje norma ISO 3166.

V souboru `/usr/share/locale/locale.alias` najdeme aliasy, kterých lze také pro nastavení proměnných použít. Pro češtinu tam najdeme „czech“, pro slovenštinu „slovak“.

Tyto proměnné nastavujeme obvykle buď pro všechny uživatele v `/etc/profile` řádkem:

```
export LC_ALL=cs_CZ.ISO-8859-2
```

nebo pro jediného uživatele stejným řádkem v `.profile` nebo `.bash_profile` v domovském adresáři uživatele.

Nastavení jednotlivých kategorií lze jednoduše zjistit pomocí příkazu `locale`.

Odkazy: **locale(7)**, **locale(1)**, **locale(5)**, Lokalizace (1)

X Window System

České prostředí v systému X Window je trochu více problematická záležitost.

Aby aplikace mohla přijímat české znaky, je třeba, aby knihovna jazyka C uměla tyto znaky rozpoznávat. K tomu musí být korektně nastavena kategorie LC_CTYPE na „cs_CZ“ případně „cs_CZ.ISO-8859-2“.

V případě, že spouštíme X server pomocí `startx` může být nastavení provedeno v `/etc/profile` nebo `~/.[bash_]profile`. V případě, že X server je spouštěn z xdm, je třeba nastavení provést buď v `.xsession` (Debian) nebo

`.Xclients` (Red Hat) před spuštěním správce oken (pro jednoho uživatele), nebo na začátku `/etc/X11/Xsession` (Debian) nebo `/etc/X11/xinit/Xclients` (Red Hat) pro všechny uživatele.

Dále program musí podporovat přijímání českých znaků, a to tzv. Input metody, a ještě musí korektně na začátku programu inicializovat internacionalizaci jak Xlib, tak libc, tak i přípravků, které používá. Mnoho programů tak nečiní, a tak je těžké je přinutit k českému (slovenskému) psaní.

Klávesnici můžeme nastavit dvěma způsoby: přes `Xmodmap` nebo `xkb`. V distribucích se obvykle používá xkb řešení, které je závislé na X serveru. Xmodmap řešení není na X serveru závislé. Nastavení mapy xkb klávesnice se provádí v `/etc/X11/XF86Config` v sekci Keyboard, klíčová slova Xkb*. To, jaká jména klávesnic pro XkbKeymap můžete použít, zjistíte v `/usr/lib/X11/xkb/keymap.dir`.

Xmodmap řešení najdete na webové stránce Zdeňka Kabeláče (2), kde také najdete knihovnu **libi18n**, kterou lze použít pro vnucení českých znaků do aplikací, které se nechovají internacionálně.

Mezi českou a anglickou klávesnicí se přepíná pomocí klávesy *Scroll Lock*.

Tisk

Pokud píšete nový dokument, je nejlépe psát ho v (La)TEXu, jelikož zde je podpora českého tisku zatím vyřešena nejlépe. Jako textové WYSIWYG editory pro systém X Window lze také použít Applixware (3), Star Office (4) nebo AbiWord (5). Pro LATEX existuje WYSIWYG nadstavba LyX (6).

Pro převod českých ASCII textů do PostScriptu pro tisk, lze použít balík **a2ps**. Tisk ze systému X Window zatím není uspokojivě řešen. Většinou je vygenerovaný PostScript kódován v ISO-8859-2, ale PostScript používá ISO-8859-1 fonty. Někdy je na PostScript možno použít balík **ogonkify** (7), který může problém vyřešit, například tisk z Netscape.

Pro tisk dokumentů z PostScriptu na tiskárnu se používá balík **Ghostscript** (8). Tento balík ve verzi 6.0 už konečně obsahuje Type1 fonty, které mají ISO-8859-2 znaky, což by mohlo znamenat konečně blýskání na lepší časy. Další Type1 fonty (postscriptové) jsou k dispozici na `ftp://ftp.penguin.cz/pub/cestina/fonty-cz`, `http://www.intersoft.cz/linux/fonts`.

Někdy je třeba mít nastaveno pro korektní zobrazování Type1 fontů ve WYSIWYG editorech kódování fontu v souboru `fonts.dir`, který je ve stejném adresáři jako Type1 fonty, na

```
*-adobe-fontspecific
```

Někdy také pomůže nastavení aliasu v souboru `fonts.alias` na

```
*-iso8859-2 *-adobe-fontspecific
```

Poznámka: za „*“ doplníte konkrétní název fontu z `fonts.dir` bez kódování.

Nastavení v distribucích

Red Hat

Pokud ve verzi 6.2cz (9) zatrhnete balík **czech-adaptation**, jsou všechna potřebná nastavení udělána za Vás. Následuje popis, kde je co nastaveno:

V `/etc/sysconfig/i18n` je definován font a nastavení proměnných `LC_*` a `LANG`. V `/etc/inputrc` je nastaveno povolení 8bit znaků na terminálu pro shell **bash**. Implicitní klávesová mapa je zadána v souboru `/etc/sysconfig/keyboard`. Konzolové fonty, klávesové mapy apod. najdete v adresáři `/usr/lib/kbd`.

V konfiguračním souboru `/etc/X11/xinit/Xkbmap` nebo `/etc/X11/XF86Config` definujeme klávesové mapy pro systém X Window. Cestu k fontům definujeme buď v `/etc/X11/fs/config` v případě X Font Serveru (implicitně) nebo v `/etc/X11/XF86Config`. Fonty pro X server najdeme v adresáři `/usr/lib/X11/fonts`.

Debian

Pro Debian 2.1 nainstalujte balík **skcz-slink** (10). Docílíte tím instalace klávesnicových map pro systém X Window.

Nejprve je třeba vyměnit balík **kbd** za **console-tools**. V `/etc/console-tools/config` nastavit `SCREEN_FONT` na lat2u-16 a `APP_CHARSET_MAP` na iso02. Pomocí `kbdconfig` nastavit klávesnici. V `/etc/inputrc` odkomentovat „convert-meta“. Do `/etc/profile` doplnit:

```
case ‘tty‘ in /dev/tty[0-9]|/dev/tty[0-9][0-9]
        echo -ne ’\033%@\017\033(K’
esac
export LANG=czech
export LC_ALL=czech
```

Po spuštění `/etc/rc.boot/console-tools` (Debian 2.1) nebo `/etc/init.d/console-screen.sh` (Debian 2.2) a `/etc/init.d/keymaps-lct.sh start` a novém přihlášení by vše mělo fungovat. Klávesové mapy pro konzoli najdeme v adresáři `/usr/share/keymaps` a konzolové fonty apod. v adresářích `/usr/share/console*`.

Klávesové mapy pro systém X Window nastavujeme v konfiguračním souboru `/etc/X11/XF86Config` v sekci Keyboard, např. na `XkbKeymap "xfree86(us_cz_qwerty)"`. Hodnoty můžete zjistit v souboru `/usr/lib/X11/xkb/keymap.dir`. ISO Latin 2 fonty najdete v balících **xfonts-biznet-iso-8859-2-*** a **xfonts-intl-european**. Pro Debian 2.1 balík **xfntil2**. Cesty k fontům se nastavují v `/etc/X11/XF86Config` v sekci Fonts. Pro Debian 2.2 není třeba cesty nastavovat, pro Debian 2.1 je třeba v `/etc/X11/XF86Config` v sekci Files doplnit na začátek cesty k il2 fontům:

```
FontPath   "/usr/X11R6/lib/X11/fonts/misc-il2/"
FontPath   "/usr/X11R6/lib/X11/fonts/75dpi-il2/"
FontPath   "/usr/X11R6/lib/X11/fonts/100dpi-il2/"
```

Je-li X server startován přes xdm (po nastartování počítače naběhne přihlašovací okno v grafickém režimu), musíme `LC_CTYPE` nastavit v `/etc/environment` nebo `/etc/X11/Xsession`.

Aplikace

Aby fungovali některé aplikace, je třeba jim trošku pomoci. U konzolových je většinou potřeba nastavit akceptování 8bit znaků, u některých X aplikací jim znaky vnutit pomocí nečistých metod (libi18n). Informace o „počeštění“ aplikací najdete v Czech-HOWTO (11).

less

Aby `less` zobrazoval české znaky, je třeba nastavit proměnnou prostředí LESSCHARSET na hodnotu latin1. Například v `/etc/profile`:

```
export LESSCHARSET=latin1
```

ls

Pro korektní zobrazování názvů souborů s českými znaky je třeba spouštět příkaz ls s přepínačem -N. Tedy do `/etc/profile` umístíme:

```
alias ls='/bin/ls -N'
```

Připojení disků z Windows

Pokud připojujeme disky z českých Windows, obvykle budeme chtít vidět názvy souborů i s českými písmenky. Předpokládám, že připojovaný oddíl máme napsán v `/etc/fstab`. Do položky *options* doplníme:

```
iocharset=iso8859-2,codepage=852
```

tedy kompletní položka může vypadat třeba následovně:

```
/dev/fd0 /mnt/a vfat user,noauto,\
        iocharset=iso8859-2,\
        codepage=852,umask=000,\
        quiet 0 0
```

nvi

Editor nvi zobrazuje místo znaků české abecedy jejich hodnotu v hexadecimálním tvaru. Tomuto zamezíme přidáním následujících řádků do `/etc/vi.exrc`:

```
set print=\
"áÁäÄčČďĎéÉěĚíÍĺĹĺĹňŇóÓôÔřŘřŘšŠťŤúÚůŮýÝžŽ"
set print=""
```

Gtk

Setkal jsem se s případem, že Gtk verze 1.2 mi nechtělo vypisovat zprávy, které obsahovaly nějaký český znak. Pomohlo následující nastavení proměnných prostředí:

```
export LC_ALL=cs_CZ
export LANG=cs_CZ
```

■

1 Jevíčko 98
http://www.linux.cz/seminare/jevicko1998/Lokalizace/lokalizace.ps
2 Xmodmap řešení
http://www.fi.muni.cz/~kabi/linux/linux.html
3 Applixware
http://www.nime.cz/suse_applix_cz.html
4 Office
http://www.intersoft.cz/linux/soffice
5 AbiWord
http://www.fi.muni.cz/~xmiksa/linux/abi
6 LyX
http://www.lyx.org
7 ogonkify
http://www.agh.edu.pl/ogonki
8 Ghostscript
http://www.ghostscript.com
9 Red Hat 6.2
ftp://ftp.linux.cz/pub/linux/redhat-cz/6.2/i386
10 Debian 2.1
ftp://ftp.debian.cz/debian-czsk/dists/slink/binary-i386
11 Czech-HOWTO
http://www.penguin.cz/czech-howto

Úvod do TEXu

Petr Olšák

Přednáška je zaměřena na začínající uživatele Linuxu, kteří právě objevili, že v Unixových systémech je obvyklé formátovat dokumenty prostřednictvím TEXu, ale ještě nevědí, co to přesně je. Budou nastíněny základní principy TEXu, vysvětlena terminologie a zmíněny odkazy na další dokumenty. Za příznivých okolností (konstelace hvězd a funkčnost techniky) může být předveden TEX „živě“ při úpravě jednoduchého dokumentu.

O programu

TEX je program na přípravu elektronické sazby. Název TEX je odvozen od počátečních písmen slova „technologie“, což v řečtině prý znamená technologii i umění. Proto se název čte řecky „tech“ ($\tau\varepsilon\chi$).

Program byl napsán profesorem Stanfordovy univerzity D. E. Knuthem koncem sedmdesátých let. Autor publikoval algoritmy programu k volnému použití ve speciálním jazyce WEB společně s nástroji, které konvertují zápis algoritmů do standardního Pascalu. To umožnilo program implementovat na libovolném operačním systému, na kterém běžel překladač Pascalu.

Nezávislost programu na výpočetním prostředí (architektura počítače, operační systém, výstupní zařízení) je jeho velkou předností. Mezi další přednosti patří:

- existence dokonalé dokumentace
- silný jazyk interního interpretu
- velmi dobrý návrh mnoha interních algoritmů pro zpracování sazby
- možnost bezplatného použití programu i jeho částí třeba ke komerčním účelům
- existence pevného bodu při přípravě dokumentů.

K poslednímu bodu: poslední verze TEXu z roku 1989 je autorem fixována a nebude do budoucna dále vyvíjena. Dokument, který jste od tohoto data připravili nebo připravíte, bude TEXem zpracovatelný se zcela stejným výsledkem třeba i za sto let.

Kde začít

Dobrým startovním bodem při pátrání po rozsáhlých informacích k programu může být webová stránka CSTUGu (1). CSTUG je (podobně jako CZLUG) sdružení uživatelů programu, kteří svými členskými příspěvky podporují rozvoj informovanosti o programu a případný vývoj souvisejí-

cího software. Dalším vhodným startovním bodem při pronikání do možností programu je dokument „První setkání s TEXem“, který lze najít například na (2).

Různé implementace

Na implementacích programu ze zdrojových textů v jazyce WEB pro jednotlivé operační systémy už nepracuje sám autor, ale mnoho dalších programátorů po celém světě. Mimo volně šířených implementací se můžeme setkat též s komerčními implementacemi, protože licence k programu je natolik volná, že to umožňuje. Mezi nejznámější implementace patří web2c, která má licenci GPL a jejím autorem je Karl Berry. Jedná se o implementaci použitelnou pro libovolný systém, pro který existuje rozumný překladač jazyka C. Mezi přímé následníky této implementace dnes patří teTeX (pro Unixy) a fpTeX (pro MS Windows). Kromě implementace programu TEX samotného a dalších spolupracujících programů pro danou počítačovou platformu je potřeba zabalit stovky maker, fontů a dalšího software a připravit vše ke snadné instalaci. Příkladem takové distribuce může být CDčko TEXlive, které připravuje každoročně TUG (sdružení uživatelů pracující v USA).

Dávkový způsob zpracování

TEX zpracovává dokumenty dávkovým způsobem. Je potřeba připravit do souboru vstupní text dokumentu společně se značkami, které vyznačují (nejlépe) logickou strukturu dokumentu nebo (hůře) přímo vyznačují nějaký požadavek na výsledný vzhled tisku. Kromě toho se TEXu předloží k načtení sada maker, které převádějí logické značky v dokumentu na příkazy určující konečný vzhled tiskoviny. Tato makra se dají dodatečně modifikovat a tím upravovat výstup podle konkrétního návrhu typografa. Pokud je soubor s textem dokumentu dobře strukturován logickými značkami, nemusí sazeč při závěrečné úpravě do tohoto souboru vůbec zasahovat. Stačí mu jen modifikovat TEXová makra, která jsou rovněž zapsána v textové podobě většinou v pomocných souborech.

Výstupem z TEXu je binární soubor s příponou `dvi`, který obsahuje přesný a definitivní popis rozmístění znaků na všech stranách dokumentu. Datový formát je samozřejmě nezávislý na platformě a na výstupním zařízení. Jsou tam uvedeny odkazy na fonty, přičemž o realizaci kreseb jednotlivých znaků se postarají až tzv. `dvi`-ovladače. Právě tyto programy musejí mít přístup k fontům a vytvářejí výstup závislý na použitém výstupním zařízení. Například program `dvips` převádí `dvi` do PostScriptu a při této činnosti zavádí do výstupu použité fonty, například ve formátu Type1. Prohlížeč TEXového výstupu, program `xdvi`, je vlastně také `dvi`-ovladač, který realizuje vzhled dokumentu v okénku systému X Window. Pokud `dvi` soubor odkazuje na postscriptové fonty, ovladač si na pozadí požádá Ghostscript o jejich vyrastrování.

Teoretická dělba práce

Jak bylo řečeno, TEX načítá při zpracování dokumentu makra, která definují chování TEXu v místě použitých logických značek v textu a určují tím konečný vzhled dokumentu. Makra tedy ovlivňují zpracování dokumentu ve dvou rovinách. Jednak je jimi určeno, jak mají vypadat logické značky v textu a jednak definují typografii dokumentu. To umožňuje určitou dělbu práce uživatelů TEXu. Odborník spolupracuje s typografem a navrhuje makra TEXu. Na druhé straně autor dokumentu si od odborníka vyzvedne seznam několika málo značek, kterými vyznačí logickou strukturu svého dokumentu. Text zapisuje v obyčejném editoru bez formátovacích vymožeností. Zajímá jej jen obsah dokumentu a starost o formu svěří odborníkovi.

Teoreticky by měl odborník nejprve navrhnout logické členění a z toho vyplývající značkování dokumentu a pak nechá autora dokument napsat (nebo písařku dokument přepsat z rukopisu). Dále před zpracováním TEXem musí odborník rozhodnout, jakou zvolí typografii dokumentu a podle těchto požadavků napíše pro TEX makra navazující na použité logické značky. Pak teprve dokument zpracuje TEXem, přičemž výsledný vzhled, který si promítne například prostřednictvím programu `xdvi` do vedlejšího okénka, může dodatečně upravovat v textovém editoru modifikací svých maker.

Takto „teoreticky“ se to ale dost často nedělá. Existují hotová makra pro TEX, která na jedné straně deklarují způsob značkování dokumentu a na druhé straně definují výchozí vzhled výstupu, který je možno více či méně komfortně dále modifikovat úpravou hodnot některých parametrů maker nebo vnitřních registrů TEXu. Snahou je, aby autor mohl svůj dokument sepsat pokud možno sám bez pomoci odborníka a bez hlubší znalosti poměrně složitého makrojazyka TEXu. Mezi nejjednodušší sadu maker patří plain, dílo samotného autora TEXu. Tato sada maker nezavazuje uživatele ani odborníka ke konkrétnímu značkování dokumentu, ale poskytuje výborné startovní pole pro vytváření vlastních maker, která se o základní makra z plainu mohou opírat.

Mezi hodně používanou výchozí sadu maker patří LATEX, dílo Leslie Lamporta, které dnes dále zvelebují lidé z tzv. LATEX teamu (3). Toto makro dává autorovi technické literatury do rukou hotový návrh logického značkování dokumentu a dále předpřipravené šablony pro výsledný vzhled – tzv. styly. Říkáme, že píšeme v LATEXu, pokud jsme přijali způsob značkování dokumentu popsaný v Lamportově manuálu „LATEX– A Document Preparation System“. Naopak, říkáme že píšeme v plainu, pokud jsme se rozhodli deklarovat vlastní značkování a připravit pro každý dokument vlastní makra na míru opírající se pouze o základní makra plainu.

TEX je schopen načtená makra uložit do binárních souborů, tzv. formátů, pro další použití. Při zpracování dokumentu pak TEX už nenačítá textovou verzi maker znova, ale velmi rychle si obnoví své znalosti z těchto binárních formátů. V TEXových instalacích najdeme většinou dva nejčastěji používané formáty: plain a LATEX. Fázi načítání textové verze těchto maker a ukládání do binární podoby nazýváme inicializací formátu. Pak už stačí jen formát opakovaně používat na různé dokumenty. I při takovém použití formátu lze TEXu dodatečně nabídnout další makra, která původní makra z formátu předefinují nebo definují nová jen pro daný konkrétní dokument.

Čeština v TEXu

Čeština a slovenština pro TEX je k dispozici v balíku CSTEX. Balík zahrnuje doplňující makra pro plain (tzv. csplain) a pro LATEX (tzv. cslatex). Dále jsou zde vzory dělení slov pro český a slovenský jazyk. Je zde také sada výchozích textových fontů, tzv. CSfonty. Ty obsahují úplnou českou abe-

cedu a vznikly modifikací defaultní rodiny fontů pro TEX Computer Modern. Kromě těchto fontů jsou zde nástroje umožňující přímo použít základní sadu 35 fontů obsažených v každém postscriptovém RIPu včetně české a slovenské abecedy. CSTEX najdete na (4).

Kromě balíčku CSTEX se snaží češtinu řešit multijazykový modul TEXových maker pro LATEX zvaný Babel. Dnes jej najdete skoro v každé distribuci TEXu. Bohužel, tento modul není kompatibilní s cslatexem, takže uživatelé LATEXu musejí být opatrní a musejí si vybrat, jakým způsobem chtějí ve svých dokumentech češtinu používat.

Uživatelské rozhraní

Začátečníci se často ptají na uživatelský komfort při použití TEXu. Mají rovněž tendenci srovnávat TEX s jinými programy na přípravu elektronické sazby jen z pohledu uživatelského rozhraní. Zapomínají ale na to, že uživatelské rozhraní není vlastností TEXu samotného, ale je dáno několika na TEXu nezávislými faktory:

- uživatelský komfort použitého textového editoru
- grafické prostředí (to závisí na použitém operačním systému)
- použitá sada značek v dokumentu (závisí na zvoleném makru nebo odborníkovi)
- možnost vyhledat, jaká se má použít logická značka (implementace nápovědy)

Ve snaze přiblížit uživatelům zvyklým na interaktivní zpracování textu (chci kus textu kurzívou a chci hned teď vidět, jak ta kurzíva bude vypadat, jsem ochotem kvůli tomu bloudit a klikat myší, nejsem ochoten si zapamatovat, že stačí napsat „`\it`“) vznikají programy, které se z uživatelského hlediska podobají Wordu, ale při povelu k uložení dokumentu se vytvoří textový soubor standardně značkovaný podle LATEXu. Příkladem takového projektu je program LyX (5).

O jednom mýtu zvaném WYSIWYG

Často lze slyšet nesprávný názor, že TEX není WYSIWYG. Doslovný překlad zkratky „co vidíš to dostaneš“ nás ale opravňuje tento názor vyvrátit. Za použití `dvi`-prohlížeče, který je samozřejmou součástí každé TEXové distribuce, můžeme na obrazovce skutečně vidět přesně to, co dostaneme při případném tisku na papír. Dokonce můžeme tento vzhled „za chodu“ měnit tak, že upravujeme TEXová makra ve vedlejším okénku textového editoru a na pozadí spouštíme opakovaně TEX.

Autoři myšlenky, že TEX není WYSIWYG, měli zřejmě na mysli, že TEX neumožňuje interaktivně zasahovat do vzhledu dokumentu v tom okénku, kde je výsledný vzhled zobrazen. Skutečně, dloubáním myší do okénka s programem `xdvi` nemůžeme zpětně ovlivnit obsah dokumentu a jeho maker, tedy TEXové vstupy. Proto se hodí spíše mluvit o dávkovém zpracování kontra interaktivním a zkratku WYSIWYG do toho raději netahat.

Následníci TEXu

Poté, co Knuth uzavřel projekt TEX a nedovolí nikomu program bez změny názvu měnit, vznikly některé další projekty, které navazují na algoritmy z TEXu a nějakým způsobem jej rozšiřují. Mezi nejznámější a zřejmě nejpoužívanější rozšíření patří pdfTEX, který umožní vygenerovat místo `dvi` přímo formát PDF. Podrobněji si o pdfTEXu můžeme přečíst na webové stránce pdfTEXu (6).

Můžeme se také setkat s projektem ΩTEXu (implementuje vnitřně 16bitové kódování). Na zcela novém následníku TEXu pracuje skupina pod zkratkou NTS (New Typesetting System). Výstupem její činnosti je zatím mírně rozšířený TEX nazvaný e-TEX (rozšíření na úrovni jazyka WEB). Také vytvářejí kompletní přepis TEXu do objektově orientovaného jazyka Java. ■

1 CSTUG
http://www.cstug.cz
2 První setkání
ftp://math.feld.cvut.cz/pub/cstex/doc/prvni.tar.gz
3 LaTeX
http://www.latex-project.org
4 CSTeX
http://math.feld.cvut.cz/olsak/cstex
5 LyX
http://www.lyx.org/
6 pdfTeX
http://www.cstug.cz/pdftex/index.html

Gimp – ilustrovaný průvodce

Dan Ohnesorg, 4. listopadu 2000

Jakub Steiner nám ve spolupráci s nakladatelstvím Neokortex připravil příjemné překvapení v podobě skoro manuálu k programu GIMP. Kniha je zajímavá tím, že popisuje verzi, které zatím nevyšla, ale je v poslední fázi vývoje. To je další důkaz výhod otevřeného vývoje, protože všichni zúčastnění již předem vědí přesně, jak bude produkt vypadat a jak se bude ovládat.

Kniha se hodně podobá manuálu, protože „jde“ při popisu po jednotlivých položkách menu. Na druhou stranu obsahuje řešené příklady a poskytuje informace o zpracování obrázků na počítači obecně, tedy bez vazby na GIMP jako takový.

V přílohách je popsána i konfigurace tabletu a použití řádkových konverzních utilit, které je možné pod Linuxem používat spolu s Gimpem.

Na celé knize je vidět, že si na ní dal autor velmi záležet a investoval do její přípravy obrovský kus práce. Nepočítal jsem ilustrace, ale bude jich okolo 200. A nejsou to žádné bezmyšlenkovitě udělané snímky obrazovky, spousta z nich je vyřezaná a kombinovaná z několika segmentů. Obvykle je obrázek minimálně půlen, přičemž v jedné části obsahuje původní stav a v druhé stav po úpravách popisovanou funkcí.

Co knize vytknout? Moc toho není. Sem tam se vloudil překlep, třeba barevný model red-green-blues mne pobavil. Někdy je problém se zlomem stránek, což je vzhledem k tomu, že kniha je sázena v TEXu překvapivé. Zásadní problém je v obrázcích, tato kniha volá po barevných ilustracích. Smutné je, že by si ji pak málokdo mohl dovolit, ale speciálně v kapitole o funkci jednotlivých filtrů jsou miniatury náhledů tak špatně vyrastrované, že někdy není vůbec zřejmé co vlastně na obrázku je. A vytisknout standardní

paletu barev pro webovské prohlížeče ve stupních šedi je více než zbytečné.

Podotýkám, že nejsem zrovna grafik, takže pokud autor někde ujel po odborné stránce tak to nemohu posoudit.

Přesto bych tuto knihu rozhodně doporučil každému, kdo uvažuje o přechodu na Gimp z nějakého jiného editoru. Kromě toho, že uživatelé Photoshopu zde naleznou popis rozdílů a mapování funkcí Photoshopu na ty v Gimpu, obsahuje Gimp spoustu možností, které nejsou pouhým procházením menu zřejmé a třeba na modifikace chování nástrojů jejich použitím s dalšími klávesami není možné intuitivně přijít.

GIMP – ilustrovaný průvodce, vydalo nakladatelství Neokortex s. r. o. v roce 2000, vydání první, cena: 299,-Kč ■

LinuxTag 2000

Dan Ohnesorg, 30. července 2000

Úvodem

29. 6. až 2. 7. 2000 se ve Stuttgartu konal LinuxTag, v pořadí již pátý. Svým způsobem byl ale první. Byl totiž výrazně největší, ale hlavně tentokrát se již jednalo o akci, o které psali ve všech novinách. V den zahájení byla zpráva o otevření výstavy ve večerních televizních zprávách uvedena hned jako druhá. LinuxTag otevíral ministr hospodářství spolkové vlády, který nad ním převzal patronát. V diskusních fórech vystupovali i státní úředníci a komisaři Evropské unie.

Možná bych měl ale říct, co to vlastně LinuxTag je. Jedná se o výstavu doprovázenou přednáškami. Asi by se to dalo přirovnat k veletrhu, ale má to svá specifika. Především přednášková část je garantována nějakou linuxovou organizací, takže neobsahuje (většinou) reklamní balast, ale jde do hloubky. Výstavní část je klasická, sestává ze spousty stánků, které jsou pronajaty komerčním firmám nebo zdarma zapůjčeny nekomerčním projektům.

Co tam bylo k vidění

LinuxTag trval čtyři dny. První den byl určen pro businessmany a byl zaměřen spíše populárně. Vstupné na tento den činilo 150 DM, nebo byla nutná pozvánka zúčastněného vystavovatele. (Vstupné v ostatní dny bylo zdarma.) Na začátku vystoupil spolkový ministr pro hospodářství a technologický rozvoj. Poté následovaly ve třech sálech paralelně přednášky a samozřejmě prohlídka výstavy. Z přednášek vybírám jen namátkou: Linux jako platforma pro nasazení v kritických obchodních operacích (přednesl pracovník firmy Siemens), Samba a KDE (L. Knetschke), Open Source aplikace jako platforma pro elektronické obchodování (Digital Consulting), GIMP v profesionálním prostředí, Linux jako platforma pro vysokorychlostní výpočetní systémy (W. Sawyer, Beowulf Project) a další.

Další dny již byly určeny široké veřejnosti a vstup byl volný. Na akci se přijelo podívat spousta školních výprav, ale i „usedle“ vyhlížejících padesátníků. Každý den byla spousta přednášek, přednášelo se paralelně ve 4 sálech (kapacity 2500, 1000 a 2 x 200 lidí). Kromě toho ve třech dalších místnostech probíhaly ukázkové instalace, diskuse a mini přednášky o specifických problémech. Přednášky byly organizovány tak, že minimálně v jednom sále byla přednáška v angličtině. Přednášet přijelo i několik „slavných“ osobností. Především Richard Stallman, který naplnil sál pro 2500 lidí tak, že kromě 2500 míst k sezení stál neurčitelný počet lidí v uličkách. Potom Alan Cox v červeném klobouku a další lidé z KDE, XFree a dokonce z projektu FreeBSD. FreeBSD mělo dokonce vlastní přednášku a stánek.

Popsat všechno co se na přednáškách a stáncích dělo je nemožné. Nicméně jedna věc byla vidět na každém kroku. Komerční a státní organizace v Německu berou Linux vážně. Například oznámení, že elektronický podpis bude realizován s použitím GPG je pro našince doslova šokující, ale i potěšitelné, protože máme naději, že naše státní správa okopíruje německé, a tím možné i evropskounijní řešení. Bylo oznámeno vypsání grantu ve výši 300 000 DM pro autory GPG, který jim má umožnit další vývoj a úpravy potřebné pro vydávání klíčů celostátní certifikační autoritou. Vystoupili členové komisí Evropské unie, kteří standardizují výměnu dokumentů v rámci Evropské unie a oznámili, že připravují zákaz předávání dokumentů ve formátech, které nejsou otevřené, a které mohou obsahovat spustitelný kód, který může poškodit systém příjemce. Úředník ministerstva hospodářství promluvil o škodách, které způsobilo nekontrolovatelné řádění viru ILoveYou a vyzval k větší rozmanitosti systémů používaných ve státní správě tak, aby nemohla být její činnost paralizována selháním jednoho textového editoru. Siemens, IBM, HP, Compaq a další se předháněli v okaté podpoře Linuxu a nabízeli sestavy s tímto OS.

Vedle těchto akcí probíhala burza pracovních míst. Německo trpí velikým nedostatkem kvalifikovaných linuxových odborníků. Linux do škol zatím prakticky neproniká, mimo jiné proto, že Microsoft vědom si výhodnosti takové investice vybavuje v německých školách zdarma učebny HW pod podmínkou, že na něm nebudou provozovat žádný jiný software, než kterým jim tam MS nainstaluje. I v tak bohaté zemi je pro školu těžké říct: „no sice tady máme 60 počítačů s PIII, ale potřebujeme od státu nové na Linux“. Na druhou strany mají výhodu v tom, že německý trh je veliký a koupěschopný, takže mají k dispozici spoustu literatury v němčině a při seznamování se s Linuxem odpadá jazyková bariéra. Na stáncích jsem viděl spoustu knih o všem možném, ale cena okolo 250 DM našince přesvědčí o nepotřebnosti takové literatury. Kromě toho spousta časopisů zvučných jmen vychází v německé variantě s obsahem přeloženým z anglického.

Co bude příště?

Příští rok má být výstava ještě velkolepější. Návštěvu bych každému doporučoval i když to přijde trochu draho. Já jsem bydlel u kamaráda, takže jsem neplatil nic než lístek na vlak, ale s noclehem by se návštěva posouvala do poměrně drastických částek. Návštěva na jeden den má smysl, pokud si chcete jen prohlédnout výstavu, pokud chcete navštívit i nějakou přednášku, tak jeden den určitě nestačí. Dostal jsem plakáty, abych dělal příštímu ročníku LinuxTagu reklamu i u nás, takže pokud jej někdo budete chtít vyvěsit na významném místě, můžu plakát poslat.

V následujícím článku si můžete přečíst, místy možná trochu kostrbatý, překlad úvodní řeči spolkového ministra hospodářství. ■

Deutschland goes Open Source

Dan Ohnesorg, 30. července 2000

Německo v minulém roce velmi pokročilo v budování informační a komunikační společnosti. Vybavení počítači a internetovými přípojkami značně pokročilo. Předpokládáme, že v roce 2000 přesáhne obrat v tomto odvětví 200 miliard marek. Spolková vláda i průmysl spustili několik iniciativ, které slibují ještě vyšší využití inovačního potenciálu a zvýšení zaměstnanosti. Německo má nyní všechny předpoklady stát se vedoucí zemí na trzích informačních technologií.

To je všechno v pořádku, ale nepřináší to jestě něco? Bezpečnost, hackeři, Denial of Service útoky, viry?

Právě události posledních týdnů ukázaly slabá místa naší infrastruktury. Dlouho zavírala veřejnost, jakožto i tak zvaní experti, oči před rizikem, které přináší dominance jednoho operačního systému a jediných softwarových aplikací. Globální propojení všech životních a pracovních oblastí vede k tomu, že informační technologie jsou neodmyslitelnou součástí života každého občana. O to větší jsou rizika a možné škody, které by vznikly při napadení této struktury.

V těchto souvislostech nabízí programy se zveřejněným zdrojovým kódem řešení. Vývoj Open Source programů, které mohou být, minimálně od odborníků, zevrubně prověřeny, nastoluje mnoho dalších bezpečnostních otázek, jak v oblasti software, tak v oblasti hardware. Z těchto důvodů podporuje spolkové ministerstvo hospodářství od roku 1999 vývoj Open Source aplikací v oblasti bezpečnosti. Na podzim bude vydána příručka pro střední podniky a státní správu, která zevrubně osvětlí principy Open Source programů. Kvůli tomu vypsalo spolkové ministerstvo hospodářství výzvu na vývoj a testování nových technologií, které mají přinést více bezpečnosti do internetových technologií. Já osobně doufám, že to přispěje k tomu, že přenos dat po veřejných sítích bude bezpečnější a spolehlivější.

Z hospodářského hlediska přináší Open Source mnoho výhod, inovace spočívající v lepších, uživatelsky příjemnějších a bezpečnějších programech mohou být rychleji vytvořeny a šířeny, protože vývoj na bázi zveřejněných zdrojových kódů může probíhat paralelně na mnoha místech. Speciálně v Německu vidíme potěšující známky velice živé a kreativní Open Source komunity.

To všechno potvrzuje i vývoj LinuxTagu 2000, ze kterého se stává celoevropské setkání členů této komunity. Je důležité, že byly umožněny průmyslové aplikace Open Source produktů. K tomu patří i bezproblémové zavedení, které je zajišťeno kompatibilitou s jinými produkty. Díky tomu bude umožněno i široké nasazení Open Source ve státní správě. Podpora ministerstva hospodářství je zajištěna na mnoha frontách, mimo jiné i na konferenci o aspektech softwarových patentů v Berlíně.

Pomozte nám udělat z Německa baštu E-Commerce na špičce světového vývoje.

Mnoho úspěchů Open Source software a celému LinuxTagu 2000.

Neutorizovaný překlad pozdravu Dr. Wernera Müllera, ministra pro hospodářství a technologický rozvoj návštevníkům LinuxTag 2000 ■

Správa uživatelských kont

Milan Čermák, 25. března 2000

Správa uživatelských kont je na unixových systémech (Linux nevyjímaje) velmi důležitá činnost, zvláště pokud chceme svůj počítač použít jako server (ať už internetový nebo v nějaké lokální síti). Jde zde hlavně o vyřešení problému zabezpečení dat jednotlivých uživatelů.

Ale i na běžném počítači, který ani není k síti připojen, je vhodné mít založeno kromě administrátorského i běžné konto. Práva superuživatele jsou totiž velmi široká (jako `root` si můžete dovolit vlastně všechno), a proto je vhodné pro běžnou práci (kreslení obrázků, psaní dokumentů) používat běžný uživatelský účet. Některé instalační programy sice nabízejí založení běžného konta automaticky, ale i tak je dobré vědět, jak s konty manipulovat.

Uživatelé

Může se stát, že po nainstalování systému máme k dispozici pouze superuživatelský účet, a proto je (z výše uvedených důvodů) vhodné založit běžné konto. K tomu slouží program **useradd**, kterému jako parametry zadáme uživatelské jméno a další věci. Pokud tedy chceme založit uživateli `pepa` konto, zadáme:

```
$ useradd -m pepa
```

Tím se založí uživatelské konto a další potřebné náležitosti. Parametr `-m` znamená, že se má založit domovský adresář (pokud ještě neexistuje) a do něj nakopírovat základní konfigurační soubory. Ty bývají uloženy v adresáři `/etc/skel`.

Mnoho věcí je možno dalšími parametry ovlivnit. Uživateli je možné nastavit jméno domovského adresáře přepínačem `-d`.

```
$ useradd -d /home/josef -m pepa
```

Standardně se domovský adresář jmenuje `/home/uživatel`.

Dále je možné specifikovat, do kterých skupin má uživatel patřit. Má-li tedy náš uživatel `pepa` patřit do skupin `pepa`, `users` a `gamblers`, zadáme:

```
$ useradd -g pepa -G users,gamblers -m pepa
```

Uvedené skupiny už musí existovat. Parametr `-g` udává, že se má skupina `pepa` použít jako výchozí. Taková skupina může být pouze jedna a bývá to buď skupina `users` nebo skupina se jménem uživatele. K manipulaci se skupinami se dostaneme později.

Z dalších údajů, které můžeme při zakládání konta nastavit, jmenujme nastavení příkazového interpretu — shellu (parametrem `-s`) a nastavení komentáře (parametrem `-c`). Komentář bývá využíván některými programy (např. **finger** nebo **mail**) pro zjišťování bližších údajů o uživateli.

Pokud programu **useradd** nezadáme některé položky, použije se implicitní nastavení. Jaké to nastavení je, můžeme zjistit zadáním:

```
$ useradd -D
```

A pokud nám nějaká položka nevyhovuje, můžeme ji změnit použitím stejného parametru jako pro zakládání konta, jen místo jména uživatele použijeme parametr `-D`.

Druhým možným programem pro založení uživatelského konta je **adduser**. I tomu je možné předávat parametry z příkazové řádky, ale většinou to není nutné. Automaticky vytvářené údaje (domovský adresář, skupina apod.) bývají přijatelné. Navíc lze způsob generování upravit v bohatě komentovaném souboru `/etc/adduser.conf`.

Pro založení konta uživateli `pepa` tedy zadáme:

```
$ adduser pepa
```

Program automaticky založí domovský adresář a skupinu uživatele a vyžádá si zadání některých údajů (informace o osobě a heslo).

Program **adduser** ale není obsažen ve všech distribucích (chybí například v distribuci Red Hat). Pak bývá nahrazen symbolickým odkazem na program **useradd**.

Třetí možností (a ta je naopak pouze v Red Hat Linuxu) je použití grafické nadstavby, která se spouští z prostředí systému X Window. Je součástí tzv. Ovládacího panelu (Control Panel).

Samozřejmě je vhodné zabezpečit konto každého uživatele heslem. Program **adduser** si zadání hesla vyžádá automaticky. Pokud ale zakládáme uživatele příkazem **useradd**, musíme mu heslo přiřadit ručně. Slouží k tomu program **passwd**. Zadáním:

```
$ passwd pepa
```

můžeme změnit heslo uživateli `pepa`. To můžeme samozřejme provést jen jako superuživatel, protože pouze ten má přístup k souboru `/etc/passwd`. Běžný uživatel také může měnit heslo, ale pouze svoje. Dosáhne toho spuštěním programu **passwd** bez parametrů.

Měl bych ještě dodat, že pro zvýšení bezpečnosti systému se hesla kódují 56-bitovým kódem DES (který zatím není dekódovatelný v přijatelném čase) a ukládají se buď do souboru `/etc/passwd` nebo pro ještě větší bezpečnost do speciálního souboru `/etc/shadow`.

Teď se dostáváme k obrácenému problému. Máme zlobivého uživatele a chceme ho vyloučit z našeho počítače. Použijeme k tomu příkaz **userdel**:

```
$ userdel pepa
```

Pokud před uživatelské jméno přidáme parametr `-r`, smažeme mu všechny soubory i s jeho domovským adresářem.

Pokud chceme našemu zlobivému uživateli jen zablokovat konto a ne ho rovnou rušit, půjdeme na to jinak. Všechny informace o uživatelských kontech jsou uloženy v souboru `/etc/passwd`. Jednotlivé položky jsou odděleny dvojtečkami. Napíšeme-li do druhé položky (kde je uloženo heslo) jako první znak hvězdičku „*“, nebude se moct daný uživatel přihlásit do systému.

Třetím (a posledním) problémem je změna údajů o uživateli. Dejme tomu, že si nějaký uživatel vzpomene, že by chtěl jiné jméno. Asi není vhodné uvedené konto zrušit a založit nové (i když možné to je). Vhodnější a efektivnější cestou je použití programu **usermod**. Uvedený problém bychom pro uživatele `pepa`, který se chce jmenovat `josef`, vyřešili například následovně:

```
$ usermod -l josef -g josef -d /home/josef -m pepa
```

Parametrem `-l` nastavíme nové uživatelské jméno, parametrem `-g` počáteční skupinu (která ale už musí existovat!) a parametrem `-d` nový domovský adresář. Přepínač `-m` navíc zajistí, aby se nový adresář založil a přesunuly se do něj všechny soubory z původního adresáře.

Můžeme samozřejmě změnit i obsah komentáře pomocí přepínače `-c`. Tato položka ale má vnitřní strukturu, kterou využívají další programy, a proto je k její modifikaci vhodnější použít utilitku **chfn**. Pomocí ní může informace o osobě změnit jak `root`, tak i uživatel po zadání hesla.

Skupiny uživatelů

Každý soubor má nastavena určitá práva pro vlastníka, skupinu a ostatní uživatele. Většinou si (jako správce nějakého počítače) vystačíme s právy vlastníka a ostatních. Do organizačních problémů se dostaneme, když bude chtít nějaká skupina uživatelů sdílet nějaké soubory (například kvůli vývoji nějakého programu). Nyní již s právy vlastníka a ostatních nevystačíme — vlastník musí mít přístup, ostatní uživatelé ne, ale co členové skupiny? Řešením je právě skupina uživatelů. Založíme novou skupinu (neboli group), vložíme do ní uživatele, kteří mají mít k daným souborům přístup a nastavíme souborům (a případně adresářům) vhodná práva.

(Příčinou k vytvoření skupiny uživatelů nemusí být jen sdílení souborů v rámci vývoje programu. Pomocí skupiny uživatelů můžeme například omezit přístup k disketové mechanice nebo jiným zařízením.)

To byl mírně filozofický pohled, teď se podíváme na samotné provedení. K založení skupiny můžeme použít program **groupadd**, který má jediný parametr, kterým je jméno skupiny. Skupinu `gamblers` založíme příkazem:

```
$ groupadd gamblers
```

Měl bych ještě dodat, že se záznamy o skupinách ukládají do souboru `/etc/group`, který je obdobou souboru `/etc/passwd`.

Skupinu jsme založili, teď do ní vložíme nějaké uživatele. K tomu můžeme použít program **usermod**, jak bylo ukázáno výše. Problém je, že pro každého uživatele, musíme zapsat všechny skupiny, do kterých patří. Druhou možností je opět program **adduser**, kterému můžeme zadat jako parametry jméno uživatele a jméno skupiny. Program přiřadí uživatele do skupiny, aniž by nějak ovlivnil předchozí přiřazení. Program **adduser** tedy zavoláme takto:

```
$ adduser pepa gamblers
```

A zbývá nám ještě poslední věc, jak skupinu zrušit. K tomu existuje program **groupdel**. Parametrem je opět název skupiny. Program odstraní ze souborů `/etc/passwd` a `/etc/group` všechny záznamy o rušené skupině. V případě záznamů o uživatelích odstraní, samozřejmě, pouze přiřazení, nikoli celý záznam.

Závěr

Cílem tohoto článku bylo ukázat začínajícím uživatelům způsob manipulace s konty. Na účet toho jsem se záměrně dopustil některých zjednodušení. Bližší podrobnosti je možné najít v manuálových stránkách. ■

Disky všech druhů spojte se

Milan Čermák, 23. ledna 2000

Tento článek je určen všem těm, kteří si v touze po levnějším a lepším operačním systému nainstalovali Linux a snaží se dopátrat (většinou marně), kde se skrývá jejich disketová mechanika, CD–ROMka, nebo kus disku, který obsahuje třeba M$–Windows (libovolné verze). Tento problém vychází z toho, že Linux (jak se ostatně dalo očekávat) používá pro správu mechanik a disků odlišný systém než operační systémy firmy Microsoft.

```
# <file system> <mount point> <type>  <options>             <dump> <pass>
/dev/hda1       /             ext2    defaults                0      1
/dev/hda2       none          swap    sw                      0      0
proc            /proc         proc    defaults                0      0
/dev/hdc        /mnt/cdrom    iso9660 defaults,noauto,ro,user 0      0
/dev/fd0        /mnt/floppy   msdos   defaults,noauto,user    0      0
```

Výpis č. 1: Struktura souboru `/etc/fstab`

Soubor	Označení v DOSu	Popis
/dev/fd0	A	První disketová mechanika
/dev/fd1	B	Druhá disketová mechanika
/dev/hda		První IDE/ATAPI disk
		(`master` na prvním (E)IDE řadiči)
/dev/hda1	C	První primární oddíl na prvním IDE disku
/dev/hda2	F	Druhý primární oddíl na prvním IDE disku
/dev/hda5	D	První rozšířený oddíl na prvním IDE disku
/dev/hda6	E	Druhý rozšířený oddíl na prvním IDE disku
/dev/hdb		Druhý IDE/ATAPI disk
		(`slave` na prvním (E)IDE řadiči)
/dev/hdb1	G	První primární oddíl na druhém IDE disku
/dev/hdc		Třetí IDE/ATAPI disk
		(`master` na druhém (E)IDE řadiči)
/dev/sda		První SCSI disk (s ID 0)
/dev/sda1	C	První primární oddíl na prvním SCSI disku
/dev/sdb		Druhý SCSI disk (s ID 1)

Tabulka č. 1: Přiřazení speciálních souborů k zařízením

Organizace zařízení

V systému Linux je každému zařízení přiřazen jeden (nebo i více) speciální soubor z adresáře `/dev`. Způsob přiřazení je dán jednak typem zařízení a jednak ovladačem, který se pro dané zařízení používá. Některé vybrané soubory jsou uvedeny v tabulce Přiřazení speciálních souborů k zařízením. DOSovské přiřazení písmen uvedené v tabulce téměř určitě nebude odpovídat vašemu. Šlo pouze o to ukázat, jakým způsobem přiděluje DOS označení.

V adresáři `/dev` nejsou samozřejmě pouze speciální soubory pro disky a mechaniky. Můžeme tam najít soubory pro paralelní a seriové porty, pro myš, pro zvukovou kartu a další zařízení.

Ruční připojování

Druhou odlišností proti DOSu nebo Windows je, že se všechny disky a mechaniky spojují do jediného adresářového stromu. Máme tedy nějaký disk, který je připojen do kořene a do adresářů se mohou „navěšovat“ další disky a mechaniky. Nejčastěji se k tomuto účelu používají adresáře `/mnt/zařízení` (například `/mnt/cdrom` nebo `/mnt/floppy`) nebo adresáře v kořeni (například `/cdrom` nebo `/floppy`).

K připojení diskového oddílu nebo mechaniky slouží příkaz (nebo spíše program) **mount**. Například disketovou mechaniku můžeme připojit příkazem:

```
$ mount -t msdos /dev/fd0 /mnt/floppy
```

Může to vypadat děsivě, ale není to tak složité (navíc si dále ukážeme, jak to ještě zjednodušit). Znamená to, že po programu **mount** chceme, aby připojil mechaniku zastoupenou souborem `/dev/fd0`, která obsahuje souborový systém „msdos“ (parametry `-t msdos`) do adresáře `/mnt/floppy`.

Obdobným příkazem můžeme připojit třeba mechaniku CD–ROM, která je zastoupena souborem `/dev/hdc`.

```
$ mount -t iso9660 /dev/hdc /mnt/cdrom
```

kde „iso9660“ je souborový systém používaný pro datová CD.

Ruční odpojování

Opačnou věcí, kterou je pod Linuxem v souvislosti s disky a mechanikami, nutné dělat, je nezapomenout je odpojit před vyjmutím z mechaniky. To je kritické zvláště u disketových mechanik, kde můžeme disketu bez problémů a bez odpojení vyjmout. U CD to nejde, protože Linux si mechaniku zamkne a dokud ji neodpojíte, tak se k CD dostanete pouze za pomoci šroubováku (nebo jiného nástroje). U disket jde hlavně o to, že Linux používá bufferovaný zápis a naší neopatrnou činností bychom mohli o dosud nezapsaná data přijít. Takže, dříve než disketu vyjmeme z mechaniky, jí odpojíme příkazem **umount**:

```
$ umount /dev/fd0
```

Tady už nejsou potřeba žádné další údaje, neboť Linux si vede přesnou evidenci připojených mechanik. (Konkrétně v souboru `/etc/mtab`.)

Bufferování je také důvod, proč je nutné Linux (a nejen ten) správně ukončit. Nějaká data, určená k zápisu, by totiž ještě mohla být v paměti, a my bychom tím, že počítač prostě vypneme, o ně mohli přijít.

```
#
# Config file for xvmount (v3.5)
#
# name     device       directory       type      options
Floppy:    /dev/fd0     /mnt/floppy     vfat      defaults
CD-ROM:    /dev/hdc     /mnt/cdrom      iso9660   ro
DOS-HD:    /dev/hda3    /mnt/dos        msdos     defaults
```

Výpis č. 2: Struktura konfiguračního souboru /etc/xvmounttab

Poloautomatické připojování

Bylo by samozřejmě velmi úmorné a nepraktické, kdybychom museli při každém připojení nějaké mechaniky zadávat znovu a znovu všechny parametry. Naštěstí existuje způsob, jak si tuto činnost zjednodušit. Jedná se o konfigurační soubor /etc/fstab, ve kterém jsou shromážděny popisy mechanik a diskových oddílů spolu s informacemi o souborových systémech, větvích, kam se mají připojovat, a dalších věcech.

Struktura souboru /etc/fstab je vidět na výpisu Struktura souboru /etc/fstab. Je to celkem přehledná tabulka s následujícím významem položek.

V prvním sloupci jsou uvedeny speciální soubory mechanik nebo diskových oddílů, v druhém pak místo, kam se mají připojit. Následuje souborový systém a doplňující nastavení. Položka *dump* určuje, zda se mají data na daném zařízení zálohovat. To je vhodné například pro server, kde je samostatný diskový oddíl s adresáři uživatelů. Položka může nabývat hodnot 0 nebo 1. Poslední položka *pass* může nabývat hodnot od nuly výše a udává, zda se má ten který souborový systém kontrolovat po špatném vypnutí počítače. Z toho je zřejmé, že například v druhém řádku (první řádek je komentář) souborový systém /dev/hda1 se připojí do adresáře /, což je kořen stromu, a obsahuje souborový systém „ext2“, což je standardní linuxový systém.

Zajímavý je třetí řádek, který představuje informace o odkládacím oddílu. Ten se samozřejmě nikam nepřipojuje, ale systém o něm musí vědět.

První tři záznamy by se měly v souboru /etc/fstab objevit hned při prvním startu, protože jsou zakládány při instalaci. Pokud je jich víc, je to v pořádku (možná je adresář /home nebo /usr na samostatné oblasti). Pokud je jich míň, může to znamenat problémy (třeba chybějící odkládací oddíl).

Teď se dostáváme k tomu, co v souboru /etc/fstab (možná) není a co bychom tam chtěli přidat. Pátý řádek obsahuje informace o jednotce CD–ROM, která je zastoupena souborem /dev/hdc (a je tedy připojena jako „master“ na druhém (E)IDE řadiči). Ta obsahuje souborový systém „iso9660“ a má se připojovat do adresáře /mnt/cdrom. Co znamenají přídavné volby ve sloupci *options* vysvětlím dále.

V dalším řádku jsou uvedeny informace o disketové mechanice. Ta se má připojovat do adresáře /mnt/floppy a používá souborový systém „msdos“.

Po přidání uvedených řádků stačí pro připojení například CD–ROMky napsat:

```
$ mount /dev/hdc
```

nebo

```
$ mount /mnt/cdrom
```

a program si už další parametry dohledá sám v souboru /etc/fstab.

Nyní se podívejme blíže na volby uvedené ve sloupci *options*. Zde jsou uvedeny různé přepínače, které ovlivňují práci se zařízením jak ze strany systému, tak ze strany uživatele.

Parametr „defaults“ uvedený u všech záznamů (vyjma odkládacího oddílu) nastavuje základní způsob přístupu (zařízení je pro čtení i zápis, lze z něj spouštět programy, připojí se při startu systému a další). Ostatní volby uvedené u CD–ROMky a disketové mechaniky upravují právě toto základní nastavení.

Volbou „noauto“ systému oznamujeme, že se nemá pokoušet dané zařízení připojovat při startu systému. Ono by se mu to stejně nepodařilo, pokud bychom náhodou nezapomněli v mechanice disketu a v CD–ROMce CD.

Dalším společným parametrem je volba „user“. Tou může správce poskytnout právo připojovat zařízení běžným uživatelům. Spolu s touto volbou se automaticky nastaví zákaz spouštění programů z tohoto zařízení. To je možné změnit přidáním volby „exec“.

Posledním použitým parametrem je volba „ro“ u jednotky CD–ROM. Tou systému sdělíme, že je tento souborový systém určen jen ke čtení. Ne snad, že by se systém pokoušel na CD zapisovat (to on si zjistí, že to nejde), ale vypsal by o tom otravnou hlášku.

Občas může nastat situace, že potřebujeme jedno zařízení s několika souborovými systémy. Většinou se jedná o diskety, které mohou obsahovat „vfat“ nebo (v unixových systémech běžnější) „minix“. Můžeme to vyřešit buď tak, že pro každou možnost vytvoříme záznam a adresář. Uživatel pak podle typu své diskety použije vhodný příkaz a obsah najde v příslušném adresáři.

Druhým možným řešením je vložit do sloupce *type* hodnotu „auto“. Tím se zbavíme starostí o formát diskety a přenecháme je systému k autodetekci. Tato metoda ale není všemocná. Rozezná sice „nápadně“ odlišné souborové systémy jako „ext2“ od „msdos“, ale nerozezná například souborový systém „vfat“. Ten identifikuje jako „msdos“, čímž uživateli upře používání dlouhých jmen souborů.

Připojování stiskem tlačítka

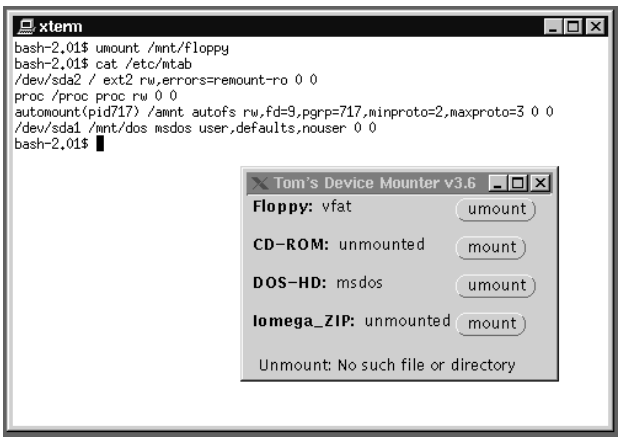
Dalším možným zjednodušením uživatelova života v souvislosti s připojováním mechanik a oddílů, je použití grafické nadstavby nad programy **mount** a **umount**. Program, který takovou nadstavbu poskytuje, se jmenuje **xvmount**. Po spuštění nabídne uživateli okno, kde jsou sepsány všechny mechaniky a oddíly a u každého záznamu je tlačítko s nápisem „mount“. To se po stisknutí a připojení daného zařízení změní na „unmount“.

Seznam připojitelných zařízení se řídí souborem /etc/xvmounttab, který můžete vidět na výpisu Struktura konfiguračního souboru /etc/xvmounttab. Je to (stejně jako soubor /etc/fstab) přehledná tabulka.

První sloupec udává identifikační text, který se vypíše do dialogového okna. Dále je uveden speciální soubor za-

řízení, adresář, kam se má souborový systém připojit, typ souborového systému a doplňující volby.

Naneštěstí může používání tohoto nástroje působit drobné potíže jak správci, tak uživateli. Program si totiž načítá stav mechanik pouze při startu a pak už s okolím nekomunikuje, což může vést k nepříjemným situacím. Jedna taková je na obrázku. K zobrazené situaci došlo, když jsem se pokusil odpojit zařízení, které jsem už předtím programem **umount** odpojil. Program **xvmount** pak neustále trvá na tom, že zařízení je připojeno, i když se v souboru `/etc/mtab` tvrdí opak. Obdobná situace může zaručeně nastat i při připojování již připojené mechaniky. Takovéto případy lze jednoduše řešit restartem programu, i když to není právě elegantní.



Správce systému může potkat podobný, ale obrácený problém. Zařízení připojené programem **xvmount** nejde programem **umount** odpojit. K tomu dojde, pokud je v souboru `/etc/xvmounttab` jako zařízení místo jména speciálního souboru uvedeno jméno linku (například `/dev/cdrom` místo `/dev/hdc`).

Další závadou je, že program neakceptuje volbu „nouser“ (opak „user“) a dovolí tak uživateli připojit libovolné zařízení, i když si to správce nepřeje. Naopak program **umount**, který tuto volbu zná, nedovolí dané zařízení odpojit. Kromě lehké nepříjemnosti to však má ještě jeden podstatnější rys. Pokud je program nevhodně nakonfigurován, může dojít ke zneužití a uživatel může získat přístup k datům, ke kterým přístup mít neměl.

Skoro úplně automatické připojování

Poslední a pro uživatele asi nejpříjemnější metodou připojování je použití tzv. **automounteru**. Jde o nástroj, který automaticky připojí určité zařízení, když ho uživatel požaduje. Takovým požadavkem už není volání nějakého programu, ale například přechod do adresáře (`cd`), kam má být ono zařízení připojeno. Bohužel přechod tak, jak ho provádí program **Midnight Commander** nestačí, protože se podadresáře pro připojení vytvářejí dynamicky (a adresář, ve kterém má obsah zařízení být, tam prostě není). Takže pro první vstup do adresáře (a s tím spojené připojení) je nutné použít příkaz `cd adresář` nebo `ls adresář`. A po určité době, kdy už se s adresářem nepracuje, jej **automounter** opět odpojí.

Ačkoliv to vypadá velmi lákavě při použití pro diskety, které by se pak mohly chovat podobně jako pod DOSem nebo Windows, musím všechny zklamat, protože právě pro diskety (které nejčastěji potřebuje běžný uživatel) se tento přístup naprosto nehodí. Vhodnější použití je například pro diskové oddíly na serveru (které potřebujeme jednou za čas), nebo pro FTP archivy.

Automounter totiž všechna zařízení připojuje s takovými právy, že zapisovat smí pouze administrátor (a to ještě pokud si to sám nezakáže).

Ke zprovoznění automounteru potřebujeme dvě věci. První se týká jádra, kde musí být přeložena podpora tohoto nástroje. (Článek o kompilaci jádra je možné najít v Linuxových novinách z prosince 1997.) Při konfiguraci je nutno odpovědět nebo zaškrtnout „yes“ u položky „Kernel automounter support“.

Druhou věcí je balík **autofs**, který je možné najít na adrese (1). Po jeho nainstalování a potřebné konfiguraci je **automounter** plně k dispozici.

```
# Sample auto.master file
# Format of this file:
# mountpoint map options
# For details of the format look at autofs(8).

/mnt/local      /etc/auto.local
/mnt/ftp        /etc/auto.ftp
```

Výpis č. 3: Struktura souboru `/etc/auto.master`

Konfigurace je obsažena v několika souborech `/etc/auto.*`. Základním konfiguračním souborem je soubor `/etc/auto.master`, který můžete vidět na výpise Struktura souboru `/etc/auto.master`. Ten určuje, se kterými adresáři bude automounter pracovat. Zařízení se pak budou připojovat do podadresářů tohoto adresáře. V druhém sloupci jsou názvy dalších konfiguračních souborů, které popisují způsob a místo připojení jednotlivých zařízení. Názvy těchto souborů nemusí mít nic společného s adresářem, ale pro přehlednost je vhodné je pojmenovat aspoň podobně.

V souboru na výpise jsou definovány dva adresáře s příslušnými konfiguračními soubory. První je určen pro lokální disky, druhý pro vzdálené FTP archivy.

```
# This is an automounter map and it has
# the following format
# key [ -mount-options-separated-by-comma ]
# location
# Details may be found in the autofs(5) manpage

dos             -fstype=msdos        :/dev/hda1
winnt           -fstype=ntfs,ro      :/dev/hda2
```

Výpis č. 4: Struktura podřízeného konfiguračního souboru `/etc/auto.*`

Formát podřízených konfiguračních souborů je vidět na výpise Struktura podřízeného konfiguračního souboru `/etc/auto.*` (je to přímo soubor `/etc/auto.local`). První sloupec označený jako klíč (vlastně „key“) obsahuje názvy adresářů, do kterých se připojují jednotlivá zařízení. Například diskový oddíl `/dev/hda1`, který obsahuje DOS, se pak bude připojovat do adresáře `/mnt/local/dos`.

V druhém sloupci jsou parametry obdobné těm ze souboru `/etc/fstab`. A konečně ve třetím sloupci jsou uvedeny speciální soubory připojovaných zařízení. Ta dvojtečka znamená, že jde o lokální zařízení. Pro vzdálený systém (třeba FTP archiv) by bylo na stejném místě napsáno například `ftp.fi.muni.cz:/pub/linux/`.

Když jsme si už nastavili všechno, co jsme chtěli, musíme ještě systému říct, aby si naši konfiguraci přečetl a zařídil se podle toho. Uděláme to příkazem:

```
$ /etc/init.d/autofs restart
```

Ta cesta nemusí nutně platit (závisí na používané distribuci), můžete případně vyzkoušet `/etc/rc.d/init.d/` nebo `/etc/rc.d/`.

Závěr

Cílem tohoto článku bylo poskytnout začátečníkům přehled a návod, jak pod Linuxem pracovat s disky a mechanikami, které vypadají na první pohled ztracené. Doufám, že se mi to podařilo, aniž bych způsobil újmu na zdraví těm, kteří tuto problematiku znají. ■

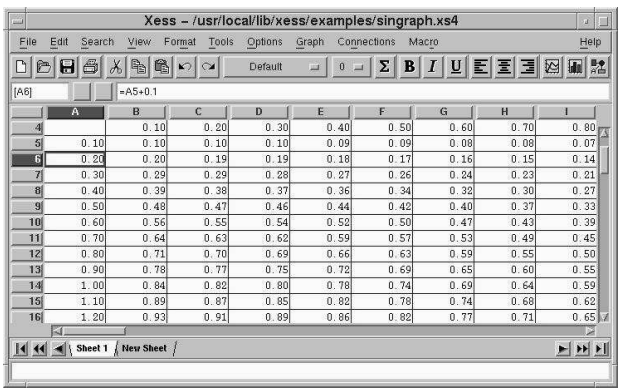
1 Balík autofs
ftp://ftp.kernel.org/pub/linux/deamons/autofs

Tabulkový procesor Xess

David Häring, 1. listopadu 2000

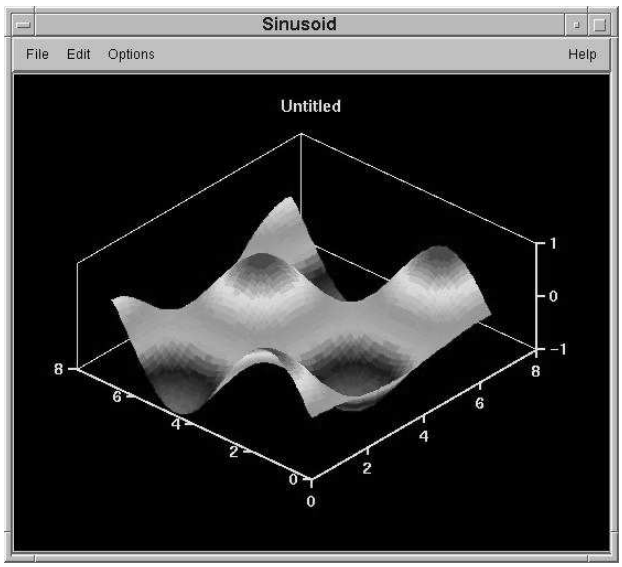
V poslední době už není problémem vybrat si z dostupného množství tabulkových procesorů, ať už jsou to ty, které jsou součástí kancelářských balíků jako jsou Staroffice (1) či Applixware (2), anebo samostatných aplikací (namátkou třeba Wingz (3), Nexs (4) či Xess). Některé z nich jsou čistě komerční aplikace (Applixware, Xess), jiné též existují v sharewarových verzích (Wingz, Xesslite) a další jsou dostupné jako freeware, případně jsou k dispozici i zdrojové kódy. Tento výčet samozřejmě není ani zdaleka úplný. Přes zdánlivě dostatečné množství aplikací zůstává však několik citelných problémů, se kterými se musíme vyrovnávat. Zpravidla to bývá neúplná podpora českého jazyka, ale hlavně to jsou problémy s přenositelností dat mezi jednotlivými aplikacemi a samozřejmě komptabilita s produkty Microsoftu. Podívejme se nyní na tabulkový procesor Xess, jehož verzi 5.0 nedávno uvedla firma AIS (5).

Xess je stabilní, rychlý, relativně jednodušší tabulkový procesor s menšími nároky na paměť, který je dostupný pro celou řadu platforem (namátkou Linux, SGI Irix, Sun Solaris a Sun OS, HP-UX, IBM AIX, Open-VMS, Microsoft Windows NT). Je tedy ideální pro nasazení v heterogenním prostředí.



Obrázek č. 2: Tabulkový procesor Xess

Kromě svého nativního formátu umí pracovat s formáty XLS (MS Excel; včetně Excel 97 / Excel 2000), WKS, WK1, WK3 (Lotus) a pochopitelně ASCII. Samozřejmostí je import/export dat do formátu HTML a zajímavostí je export do prostředí tabular v LaTeXu. Je možno vkládat obrázky formátů XPM, XBM, BMP, ICO, PBM, PNG, JPG a PCX. Disponuje obvyklou sadou matematických a statistických funkcí, obvyklými typy grafů, je možné vytvářet makra v xsBasicu. K dispozici je i SDK (obsahuje Xess API v jazyce C) pro integraci spreadsheetu do jiných aplikací.



Obrázek č. 3: Ukázka grafu v Xessu

Xess je pro Linux k dispozici v několika verzích. Ve verzi „Standard Edition“ existuje pouze pro Linux. Rozdíly mezi jednotlivými verzemi jsou shrnuty v tabulce:

	počet řádků	počet sloupců	listů	xsBasic
SE	99 999	1024	512	ne
Xess	10 000 000	4096	4095	ne
Xess Plus	10 000 000	4096	4095	ano

Pokud jde o podporu češtiny, na rozdíl od předchozích verzí už není zapotřebí používat různé „donucovací prostředky“ (předchozí verze nepříliš stabilně fungovaly s knihovnou libI18Nforce od Z. Kabeláče), pouze je třeba nastavit používání ISO-8859-2 fontů. Nicméně pro správný tisk českých znaků je zapotřebí upravovat výsledný PostScript. Potřebný filtr pro český tisk spolu s příkladem nastavení fontů je k dispozici (6).

Ve verzi 4 byla pro Linux k dispozici i „odlehčená“ verze Xessu pod názvem Xesslite, která byla uvolněna jako shareware(7). Xesslite ve verzi 5 již (alespoň prozatím) není, takže jsou k dispozici pouze komerční verze. Ceny jsou odstupňované podle počtu uživatelů (jedná se o plovoucí licence, licence tedy není vázána na počet instalací, ale na počet současně pracujících uživatelů), viz (8).

Co říci na závěr? Vzhledem k tomu, že se jedná o komerční produkt, většina uživatelů asi sáhne po dostupnějších alternativách, na druhou stranu ale Xess má své přednosti a to zejména ve stabilitě, rychlosti a dostupnosti na řadě platforem. Pokud tedy pracujete v heterogenní síti, kde potřebujete tabulkový procesor nejen do Linuxu či MS Windows, možná je Xess tím správným řešením. ■

1 StarOffice
http://www.sun.com/products/staroffice/
2 Applixware
http://www.vistasource.com/products/axware
3 Wingz
http://www.wingz.com/wingz/index.html
4 Nexs
http://www.nexs.com/
5 AIS
http://www.ais.com/
6 filtr pro český tisk z Xessu
http://salsero.ibp.cz/noauth/dave/xess/xess.html
7 Xesslite
ftp://ftp.linux.cz/pub/linux/metalab.unc.edu/apps/office/
8 Linux Corner
http://www.ais.com/linux_corner.html

Ochrana před scanováním portů: Portsentry

David Häring, 5. listopadu 2000

Scanování portů se používá ke zjištění otevřených portů systému, tedy portů, na kterých „poslouchá“ nějaká aplikace. Scanování probíhá tak, že se určitá aplikace (port scanner) postupně pokouší navázat spojení s vybranými porty na počítači, který je předmětem zájmu. Tímto způsobem lze získat přehled o službách, které na daném systému běží. Nástroje pro scanování portů ovšem neslouží pouze útočníkům pro získávání informací o vzdálených systémech. Scanování portů lze využít i pro kontrolu konfigurace systému – tedy pro ověření, že na stroji běží jen ty služby, které jsou skutečně potřebné, případně odkud jsou dostupné.

Připomeňme si, že v rámci síťových protokolů TCP či UDP je úplné síťové spojení dáno typem protokolu, IP adresou a číslem portu lokálního systému, IP adresou a číslem portu vzdáleného systému. K identifikaci portu slouží 16-bitové číslo (platná čísla portů jsou tedy v rozsahu 1 až 65535). Existují tzv. známé porty (v angličtině „well-known ports“, RFC 1700 (1)), které jsou vyhrazeny hojně používaným službám – tyto jsou definovány v souboru `/etc/services`.

Metody scanování portů

Existuje řada způsobů jak porty scanovat. Některé jsou více nápadné, jiné jsou postaveny takovým způsobem, aby si administrátor vzdáleného systému pokud možno ničeho nevšimnul. Protože portscan může být předzvěstí chystaného útoku, je pochopitelně žádoucí, aby administrátor včas scan rozpoznal a posoudil. Podívejme se proto nyní krátce na některé běžné metody užívané při scanování portů (dostupné např. v utilitě `nmap` (2)).

Nejjednodušší je tzv. „TCP connect scan“. V tomto případě se pokusíme navázat normální TCP spojení na určitý port vzdáleného počítače pomocí volání jádra `connect()`. Takové pokusy o navázání spojení jsou ovšem zpravidla nepřehédnutelné v systémovém logu vzdáleného počítače, protože řada aplikací na takové pokusy reaguje zapsaním chybového hlášení do systémového logu (např. časté hlášky „NOQUEUE: Null connection“ od sendmailu apod.). Méně nápadné jsou metody, které neotevírají plnohodnotná spojení, anebo využívají nestandardních typů paketů (paketů s nesmyslně nastavenými příznaky). Podle reakcí systému právě na tyto pakety pak odhadují, zda jsou porty otevřené či nikoliv.

Velmi častý „TCP SYN scan“ neotevírá úplné TCP spojení, ale pouze realizuje první část navazování TCP spojení; t.j. pošle SYN paket (žádost o navazání spojení) a vyhodnotí odpověď. Vzdálený počítač odpoví buď SYN+ACK paketem pokud je port otevřený (spojení je možno navázat) anebo RST paketem (port je zavřený). Takové pokusy nemusí skončit v logu vzdáleného počítače, pokud není instalován software, který s těmito pakety počítá (firewall, portsentry apod.).

Metoda „FIN scan“ používá paket, ve kterém je nastavený pouze příznak FIN (používaný při žádosti o ukončení spojení). Variantou je třeba „Xmas scan“, kdy jsou kromě příznaku FIN nastaveny příznaky URG a PUSH. Na takový paket vzdálený počítač odpoví paketem RST v případě, že je port zavřený, zatímco pokud je port otevřený, bude ho ignorovat. Některé operační systémy (např. produkty Microsoftu) se ale chovají nestandardně a odpovídají RST paketem v obou případech, což je mimochodem v rozporu s RFC 793 (3).

Metodou „ACK scan“ můžeme zjistit, zda je příslušný port vzdáleného počítače filtrován firewallem. Funguje tak, že vysíláme paket s nastaveným příznakem ACK a náhodně vygenerovaným číslem sekvence. Pokud port není filtrován, vzdáleným systém odpoví paketem RST (protože neodpovídá číslo sekvence). Pokud obdržíme jinou odpověď anebo neobdržíme žádnou, lze předpokládat že port je filtrován firewallem.

Existují i další typy neobvyklých TCP/IP paketů, které lze při monitorování otevřených portů použít.

Co umí portsentry?

Portsentry (4) bychom do češtiny přeložili jako „hlídač portů“ a dělá přesně to, co bychom od něj čekali. Detekuje pokusy o scanování vybraných portů našeho systému a umí na ně v závislosti na konfiguraci okamžitě reagovat. Jsou dostupné tři typy akcí:

- zabránění dalším pokusům o spojení ze stroje, ze kterého scan pochází (zablokování spojení pomocí firewallu anebo úpravou směrovací tabulky);
- zápis detekovaných pokusů do logu;
- vykonání libovolného předem specifikovaného příkazu.

Portsentry nabízí 3 režimy detekce scanů: v režimu „Classic mode“ neboli ”Basic port-bound TCP/UDP mode” se portsentry po startu naváže na zvolené TCP či UDP porty a čeká na příchozí spojení. V tomto režimu detekuje pouze ”connect” scany. V režimu „Enhanced Stealth mode“ se naváže na specifikované porty pomocí `bind()`, ale pro analýzu spojení používá tzv. „raw socket“ a detekuje klasické `connect()`, SYN, FIN, Xmas, NULL scany apod. V režimu „Advanced Stealth Mode“ portsentry nejprve zjistí, které porty ve zvoleném rozsahu jsou otevřené a tyto si zapamatuje. Dále pak „hlídá“ všechny ostatní porty zvoleného rozsahu. Pokud je později otevřen další port, portsentry to zaregistruje a po dobu, po kterou je port otevřený jej ignoruje. Tím se vyhne falešným poplachům v případech, kdy se například u služby FTP otevírá datové spojení na dalším portu.

```
Oct 18 13:00:04 server1 portsentry[28805]: adminalert: Psionic PortSentry 1.0 is starting.
Oct 18 13:00:04 server1 portsentry[28806]: adminalert: Advanced mode will monitor first 1023 ports
Oct 18 13:00:04 server1 portsentry[28806]: adminalert: Advanced mode will manually exclude port: 113
Oct 18 13:00:04 server1 portsentry[28806]: adminalert: Advanced mode will manually exclude port: 139
Oct 18 13:00:04 server1 portsentry[28806]: adminalert: Advanced Stealth scan detection mode activated.\
   Ignored TCP port: 21
Oct 18 13:00:04 server1 portsentry[28806]: adminalert: Advanced Stealth scan detection mode activated.\
   Ignored TCP port: 22
```

Výpis č. 5: ukázka logu portsentry - start portsentry

Pasivní nebo aktivní ochrana?

Portsentry nabízí několik způsobů, jak na detekovaný scan reagovat. Zapisování událostí je řešeno přes syslog(), záleží tedy na konfiguraci syslogu, zda pro logy portsentry vyhradíme zvláštní soubor anebo necháme události zapisovat standardním způsobem (což bývá zpravidla do souboru `/var/log/messages`). Protože portsentry nabízí možnost při detekování scanu vykonat určitý příkaz, můžeme to využít např. k poslání e-mailu, SMS zprávy apod. Důležitým aspektem je možnost zablokovat přístup vzdálenému stroji, z něhož scan přichází a to buď úpravou směrovací tabulky (na Linuxu příkazem `route`, toto ale není ideální řešení) anebo pomocí firewallu (nástroj `ipchains`, nebo starší `ipfwadm`). Je třeba si ovšem uvědomit, že automatické blokování IP adres potenciálně útočníkovi umožňuje navenek znepřístupnit služby poskytované takto chráněným systémem (tzv. „Denial of Service“ útok). Útočník poté co zjistí, že je jeho přístup blokován, může server zahltit pakety s podvrženými odchozími IP adresami a tím docílit odříznutí přístupu na server z použitých adres. Jestliže tedy na stroji provozujeme kritickou službu, je rozumnější porty pouze monitorovat a na jednotlivé případy reagovat individuálně.

Instalace a konfigurace

Instalace a konfigurace portsentry je snadná. Pokud kompilujeme ze zdrojových kódů, pouze v souboru `portsentry_config.h` nastavíme cestu ke konfiguračnímu souboru `portsentry.conf`, cestu k souboru `hosts.deny` (součást balíčku `tcp_wrappers`) a způsob logování. Pokud chceme události portsentry zapisovat do zvláštního souboru, předefinujeme SYSLOG_FACILITY z LOG_DAEMON na dejme tomu LOG_LOCAL0 a doplníme konfiguraci syslogu v souboru `/etc/syslog.conf` o řádek

```
local0.* /var/log/portsentry
```

(viz. dokumentace k balíčku `syslog`). Poté si prohlédneme a případně upravíme soubor `Makefile` (jsou zde definovány cesty, kam budeme portsentry instalovat), zkompilujeme („make linux“) a instalujeme („make install“). Toto je stručný popis instalace, samozřejmě není na škodu prostudovat si dokumentaci v přiložených souborech `README*`.

Konfigurace je také jednoduchá. Provádí se editací souboru `portsentry.conf`, ve kterém nastavíme vybrané porty, či rozsahy portů, které budou monitorovány v jednotlivých režimech a nadefinujeme příkazy, kterými se bude provádět blokace IP adres – v podstatě stačí odkomentovat jeden z předpřipravených řádků v sekci „Dropping Routes“. Podle verze jádra zvolíme buď starší `ipfwadm` nebo pro jádra verze 2.102 a vyšší `ipchains`. Pokud Vaše jádro neobsahuje podporu firewallu a nechcete jej kvůli tomu překompilovat, můžete vystačit i s příkazem

```
/sbin/route add -host TARGET reject
```

ovšem mějme na paměti, že takto zabráníme komunikaci pouze jedním směrem – pakety ze vzdáleného systému mohou stále přicházet. V sekci „External Command“ můžete nadefinovat libovolný příkaz, který se vždy po detekci scanu provede (poslání mailu, SMS apod.). Pokud chceme porty pouze monitorovat, bez blokace IP adres, nastavíme v sekci „Ignore Options“ BLOCK_TCP či BLOCK_UDP na hodnotu 0. Hodnota 2 znamená monitorování a provádění nadefinovaných akcí, ale bez blokování IP adres. Defaultní hodnota 1 znamená blokování IP adres i spouštění nadefinovaných akcí.

Porstsentry spustíme ve zvoleném režimu (volby „-tcp“, „-udp“, „-stcp“, „-sudp“, „-atcp“ a „-audp“, popsáno v souboru `README.install`). Po startu např. v režimu „atcp“ by se měla objevit zhruba hlášení podobná ukázce logu (ukázka logu portsentry - start portsentry).

V tomto případě bylo konfigurací zadáno, aby portsentry pouze monitoroval TCP porty v rozsahu 1-1023 (defaultní nastavení) a ignoroval pakety příchazející na TCP porty 113 a 139. Dále portsentry rozpoznal otevřené porty 21 a 22, na kterých běží služby ftp a ssh a pakety příchazející na tyto porty ignoruje. Portsentry jednoduše otestujeme tak, že se pomocí telnetu pokusíme přihlásit dejme tomu na port 1. V systémovém logu se pak objeví odpovídající záznam (ukázka logu portsentry - záznam scanu).

Co říci závěrem?

Monitorováním portů lze mnohdy předejít bezpečnostnímu incidentu. Na základě analýzy logů se můžeme pokusit odhadnout, zda se jedná o součást masového (a mnohdy značně nápadného) scanování služeb, anebo jde o cílené monitorování systému, které může předcházet připravovanému útoku. Pochopitelně zdaleka ne každý scan končí pokusem o napadení systému, nicméně dostatek včasných informací nikdy není na škodu a instalace portsentry je jednoduchou záležitostí, která se může vyplatit. ■

1 RFC 1700
ftp://ftp.fi.muni.cz/pub/rfc/rfc1700.txt.gz
2 Nmap
http://www.nmap.org/
3 RFC 793
ftp://ftp.fi.muni.cz/pub/rfc/rfc793.txt.gz
4 PortSentry
http://www.psionic.com/abacus/portsentry/

```
Oct 18 13:00:53 server1 portsentry[28806]: attackalert:\
   SYN/Normal scan from host: server2.domena.cz/10.12.34.56 to TCP port: 1
Oct 18 13:00:53 server1 portsentry[28806]: attackalert:\
   Ignoring TCP response per configuration file setting.
```

Výpis č. 6: ukázka logu portsentry - záznam scanu

K čemu slouží identd?

David Häring, 5. listopadu 2000

Identd je aplikace, která slouží k identifikaci uživatelů vlastnících dotazovaná síťová (TCP/IP) spojení. Představme si, že se chceme přihlásit, dejme tomu pomocí služby ftp, na vzdálený počítač. Během přihlašování se vzdálený počítač našeho počítače dotáže na uživatele, který vlastní přicházející spojení a tuto informaci pak může zapsat do systémového logu, nebo se rozhodnout spojení odmítnout apod. Tato služba používá standardně TCP port 113. Identd bývá obvykle spouštěn přes inetd (případně xinetd apod.). Podrobné informace o definici protokolu IDENT, který identd implementuje, lze nalézt v RFC 1413 (1).

Využití této služby je několik. Poskytuje přehled o uživatelích vlastnících příchozí spojení, může také posloužit k odhalení uživatelů provozujících nekalé praktiky. Dejme tomu, že uživatel „xy“ rád posílá poštu s falešnými „From:“ hlavičkami. Pokud administrátor systému používá identd, z logu poštovního agenta snadno zjistí, kdo poštu skutečně odeslal, protože např. v logu sendmailu bude následující záznam:

```
Oct 11 12:57:23 server sendmail[25433]:\
MAA25433: from=vymyslene_jmeno@server,\
size=5, class=0, pri=30005, nrcpts=1,\
msgid=<200010111057.MAA25433@server.domena.cz>,\
proto=SMTP, relay=IDENT:xy@localhost [127.0.0.1]
```

Ze záznamu „IDENT:xy@localhost“ je patrné, že poštu odeslal uživatel „xy“ a nikoliv uživatel „vymyslene_jmeno“.

Další příklad může být následující: uživatel „xy“ se přihlásí pomocí ftp na cizí účet „abc“ na vzdáleném počítači (pomineme způsob jakým se dostal k cizímu heslu). Majitel dotyčného účtu si všimne, že jeho účet je používán někým jiným a upozorní na to adminstrátora. Administrátor vzdáleného počítače ze systémového logu zjistí, že se na účet „abc“ přihlásil uživatel „xy“ ze stroje server1:

```
Oct 11 13:22:45 server2 in.ftpd[25487]:\
connect from xy@server1
Oct 11 13:22:47 server2 ftpd[25487]:\
FTP LOGIN FROM server1 [10.12.34.56], abc
Oct 11 13:23:10 server2 ftpd[25487]:\
FTP session closed
```

Administrátor stroje server2 pak může na aktivity uživatele „xy“ upozornit administrátora stroje server1. **Identd** může také zapisovat provedené akce do logu (volba „-l“). Pokud je logování zapnuto, v logu dotazovaného stroje se objeví záznam, ze kterého vyplývá, že bylo dotazováno spojení využívající port 21 (služba ftp):

```
Oct 11 13:22:37 server1 identd[25488]:\
from: 10.12.34.56 ( server1 ) for: 1323, 21
Oct 11 13:22:37 server1 identd[25488]:\
Successful lookup: 1323 , 21 : xy.users
```

Další možnost využití identd spolu s balíčkem tcp_wrappers je omezení přístupu k určitým službám (které musí obsahovat podporu tcp_wrappers) na základě informace o identitě uživatele, které identd poskytuje. Například omezení služby talk pro lokální uživatele a uživatele „abc“ vzdáleného počítače server2 zajistí následující řádek v souboru /etc/hosts.allow:

```
in.talkd:ALL@LOCAL abc@server2.domena.cz
```

Pokud navážeme na předchozí příklady, administrátor stroje server2 může zabránit uživateli „xy“ ze stroje server1 přihlášení na službu ftp editací souboru /etc/hosts.allow (za předpokladu, že ftp démon obsahuje podporu pro tcp_wrappers, viz dokumentace k tcp_wrappers) například přidáním následujícího řádku:

```
in.ftpd:ALL EXCEPT xy@server1.domena.cz
```

Rizika spojená s použitím identd

Pokud použijeme identd ve standardní konfiguraci, poskytuje zpravidla přímo jména účtů, která určitá spojení vlastní – tak jako jsme to viděli ve výše uvedených příkladech. Potenciálnímu útočníkovi to umožňuje získat informace o jménech účtů (na základě kterých se může např. pokusit hádat hesla) nebo zjistit pod jakými účty jsou provozovány síťové aplikace („reverse identd scanning“). To pochopitelně není žádoucí a proto se dále podíváme na možnosti, jak rizika spojená s provozováním služby identd omezit.

Jak rizika minimalizovat?

Identd zpravidla obsahuje podporu **tcp_wrappers**, takže přístup k službě můžeme omezit v konfiguračních souborech /etc/host.allow a /etc/host.deny. Dále kromě možnosti poskytovat přímo jména účtů vlastníků tcp spojení umožňuje identd tuto informaci skrýt v zašifrované podobě. Pak situace vypadá následovně: IP adresa lokálního a vzdáleného počítače, spolu s čísly portů, UID lokálního uživatele a datem jsou zašifrovány pomocí algoritmu DES. Výsledný řetězec je v base64 kódování předán vzdálenému stroji místo uživatelského jména. Administrátor vzdáleného počítače pak v logu vidí:

```
Oct 16 11:10:58 server1 ftpd[9677]:\
connect from [F2Q3+JSPEOdbsmGGTgmOPdDUrJtEiGVj]\
@server1.domena.cz
```

V případě, kdy administrátor vzdáleného stroje zjistí nějaké nekalé aktivity pocházející ze stroje server1, musí požádat administrátora serveru1 o dešifrování informace. K dešifrování slouží utilita idecrypt (jejíž výstup má formát datum, UID uživatele, lokální IP adresa a číslo portu, vzdálená IP adresa a číslo portu):

```
server1 / 20# idecrypt
[F2Q3+JSnEOdbamGGTwmOPqDUUJhEiMVm]
```

```
Mon Oct 16 11:10:58 2000 0 10.76.45.89\
4630 10.23.45.67 22
```

Abychom mohli tuto variantu použít, musí náš identd podporovat šifrování. V linuxových distribucích se většinou nachází identd bez této podpory, v tom případě je zapotřebí identd s podporou šifrování sestavit ze zdrojového kódu anebo třeba v případě Red Hatu použít balíček pidentd-crypt (2).

Varianta identd s podporou šifrování odstraní problém s nechtěným poskytováním informací potenciálním útočníkům (jména účtů apod.), na druhé straně ale znemožní používat identd pro omezení přístupu k službám na základě identifikace vzdáleného uživatele. V této souvislosti je třeba podotknout, že omezení přístupu k službám s pomocí protokolu IDENT je jen doplňkem a v žádném případě ne náhradou jiných mechanismů kontroly přístupu (mějme na paměti, že se zde spoléháme na korektnost údajů poskytnutých vzdáleným systémem).

Pokud identd potřebujeme a nemůžeme použít šifrování, můžeme soukromí alespoň částečně chránit spuštěním identd s volbou „-n“, která zajistí, že se místo uživatelských jmen budou vypisovat pouze číselné UID. Volba „-N“ umožní skrýt uživatele, kteří mají ve svém domovském adresáři soubor .noident – identd. V tomto případě vrátí místo identifikace uživatele chybu „HIDDEN-USER“. Užitečná je volba „-F“, která umožní definovat obsah a formát poskytované informace. Další užitečné volby jsou „-l“ (zápis událostí do systémového logu), „-o“ (vrací řetězec „OTHER“ místo identifikace operačního systému), „-e“ (vrací chybu „UNKNOWN-ERROR“ místo chyb „NO-USER“ a „INVALID-PORT“). **Identd** bychom neměli spouštět s oprávněními administrátora, pokud tedy pro identd založíme uživatele/skupinu, pak použijeme volby „-u“ a „-g“.

Závěr

Je na administrátorovi, aby podle místních potřeb zvážil výhody a nevýhody spojené s používáním identd; skutečně paranoidní uživatelé identd zatratí a nebudou jej používat vůbec :-), jiní v něm shledají užitečného pomocníka. ■

1 RFC 1413
ftp://ftp.fi.muni.cz/pub/rfc/rfc1413.txt.gz
2 Pident-crypt
ftp://ftp.fi.muni.cz/pub/linux/redhat-crypto/i386/pidentd-crypt-2.7-1.i386.rpm

Zasmáli jsme se!

Pavel Janík, 12. listopadu 2000

Opět po dlouhé, velmi dlouhé době, si teď můžete přečíst rubriku Zasmáli jsme se. Nebudu se omlouvat, že jsem neměl čas sbírat jednotlivé vtipné útržky z konferencí, a tak si tady můžete přečíst většinou jen příspěvky, které jsem obdržel mailem. Takže hned se na ně vrhněme.

Jeden z našich čtenářů asi nemá co dělat, a tak ve svých volných chvílích chatuje na Xchatu. Vyslechl tam přitom následující věty:

> Podle zatím nedoložených zpráv Microsoftu se z důvodu lepší srozumitelnosti pro běžné uživatele připravuje lokalizace názvů jednotlivých programů softwarového balíku Microsoft Kancelář 2000. Budou to programy: Slovo, Bezva, Vyhlídka, Přístup a Silná Tečka.

Už nevím, kdo mi poslal signaturu, ve které byl níže uvedený text, ale protože je o Perlu, tak to určitě byl Adelton :-)

> Any sufficiently perverted technology is indistinguishable from Perl.

Používáte newsového klienta, který umí thready (vlákna). Ne? A máte pro to rozumný důvod? Autor následující signatury ano:

> There are no threads in a.b.p.erotica, so there's no gain in using a — threaded news reader. (Unknown source)

Povzdech grafika nebo programátora webových aplikací:

> ... all I ask for is world peace, the end of hunger, and a single browser platform to develop for.

Antispammerských signatur existuje velká spousta, ale tahle mne dostala :-)

> Anyone sending unsolicited bulk email (UBE, SPAM) to this address will be charged a $25 handling fee plus a $5 network traffic fee per started kilobyte. By extracting my address from this message or its header, you agree to these terms.
> Nevertheless, spammers trying to auto-extract addresses from this message will definitely want to include uce@ftc.gov and report@fraud.org

Jednomu z našich pravidelných čtenářů se nelíbí, jak se „strefujeme“ do Windows, takže podle pravidla "Contribute nothing, expect nothing" nám poslal hlášení, které dostal od svého počítače (hmm, projekt gnokii je sice mým projektem, ale vtipná hlášení se jej netýkají :-)).

> $ rm -r gnokii*
>
> rm: do adresáře 'gnokii-0.3.1' je zakázán zápisů; vejít do něj jinak?
> Ve skutečnosti mi počítač asi chtěl říct:
>
> $ rm -r gnokii*
>
> rm: directory 'gnokii-0.3.1' is write protected; descend into it anyway?

Docela hezká ukázka úskalí počešťování aplikací, ne? Jak končí velcí spisovatelé s velkými ambicemi?

Jistý mladý muž se rozhodl, že se stane velkým spisovatelem. Když se ho ptali, co chápe pod pojmem „velký“, řekl: „Chci psát věci které bude číst celý svět, věci na které budou lidé citově reagovat, věci u kterých budou křičet, kvílet, sténat v bolesti a strachu!“ Nyní pracuje pro Microsoft. Píše tam chybové hlášky.

A jedna „Kdyby“:

Kdyby byly aerolinky jako operační systémy...
UNIX Airways: Každý si přinese jeden díl letadla s sebou na letiště. Pak jdou všichni na runway a skládají letadlo kousek po kousku za neustálého dohadování, jaký druh letadla to vlastně staví.
Air DOS: Všichni roztlačují letadlo až se vznese, pak do něj naskočí a nechají letoun letět, dokud zase nespadne na zem. Pak znovu roztlačují, naskakují znova a tak dále…
Mac Airlines: Všichni – stewardi, kapitáni, zřízenci i prodavači letenek vypadají stejně a dělají přesně to samé. Pokaždé, když se jich na něco zeptáte, řeknou vám slušně, ale rozhodně, že to nepotřebujete vědět, nechtějte vědět a že vše půjde dobře, aniž byste cokoliv znali a tak tedy už konečně mlčte.
Windows Air: Terminál je pěkný a barevný, stewardky jsou přátelské, odbavení je snadné, start je plynulý. Asi po 10 minutách ve vzduchu letoun bez varování exploduje.
Windows NT Air: Stejné jako Windows Air, ale dražší, používají o mnoho větší letadla a exploze zničí zároveň všechna letadla v okruhu 50 km.
Linux Air: Znechucení zaměstnanci všech ostatních OS aerolinek se rozhodli založit si svoji vlastní aerolinku. Sami si staví letadla, přepážky na prodej letenek a budují si runway. Chtějí jen malý poplatek, aby pokryli náklady na tisk letenek, ale můžete si také letenku stáhnout a doma si ji vytisknout. Když jste na palubě, dostanete sedadlo, čtyři šrouby, matky, klíč a kopii manuálu sedadlo-jaknato.html. Po usazení je toto plně nastavitelné sedadlo velmi pohodlné, letadlo přilétá a odlétá na čas bez jediného problému, jídlo na palubě je perfektní. Když ale zkusíte říci zákazníkům jiných aerolinek o takovém pěkném letu, všichni se ptají: „A co se musí udělat s tím sedadlem?“
OpenVMS Air: Prostý, jednoduchý terminál, drahá letenka, nehezké stewardky. Odlet je zpožděn o několik dní, protože si pilot vyjasňuje složité technické otázky s DEC/Compaq. Konečně se zjistí, že jeden z motorů není výrobkem DEC/Compaq a není proto podporován OS. Jakmile je však letoun konečně ve vzduchu, létá v jednom zátahu po měsíce kolem Země, aniž by musel přistát kvůli restartu.

Pracujete u Českého Telecomu nebo u Microsoftu? Stydí se za to vaše děti? Honzík ano:

Ve škole povídá paní učitelka: A čím je tvůj tatínek, Jirko? On je pekař. Hmm, to je hezký, na to paní učitelka, peče nám, děti, chléb. A tvoje maminka je čím, Maruško? Sím, ona je zdravotní sestra. Aha, vidíte děti, stará se o nemocné. To je krásné povolání! A co ty Honzíku? Můj tatínek tancuje nahatej v gayklubu. Hmmmm, povídá paní učitelka, i taková zaměstnání jsou… O přestávce přijde za Honzíkem a povídá: Opravdu tvůj otec tancuje v tom klubu? A Honzík odvětí: Ne, on pracuje v Microsoftu, ale my se za to všichni stydíme…

No a když už si naši čtenáři myslí, že tato rubrika je určena pouze pro kritiku Microsoft Windows případně jiných operačních systémů, tak jim to nebudeme vyvracet:

Stretli sa MS DOS a Windows. Windows hovori:
– Nazdar, kto si?
– Ja som operacny system MS DOS.
– Ano? A co dokazes?
– Ja viem ist pekne pom-pom-pom-pomaly.
A ty si kto?
– Ja som operacny system Windows.
– A co dokazes?
– Ja viem pekne zatu…

Problém začínajících uživatelů (a zejména čtenářů linuxové konference) je ten, že nechtějí číst manuály. V linuxové konferenci na tohle téma byla v květnu a dubnu velmi divoká diskuse. Já jsem si ji celou přečetl až později, neboť jsem v tu dobu měl úplně jiné problémy, ale jeden příspěvek jsem si označil tak, abych se k němu v budoucnu nezapomněl vrátit. Podělím se o něj i s vámi:

```
Subject: offtopic: k odpovedim
na zacatecnicke dotazy...
To: "'linux@linux.cz'" <linux@linux.cz>
Date: Wed, 31 May 2000 15:06:20 +0200
Reply-To: linux@linux.cz

... me napadl jeden vtip:

Prijde muz do zpovednice a rika:
"Otce, prave jsem soulozil s manzelkou
sveho pritele, zhresil jsem?"
Ze zpovednice se ozve: "RTFM!"
```

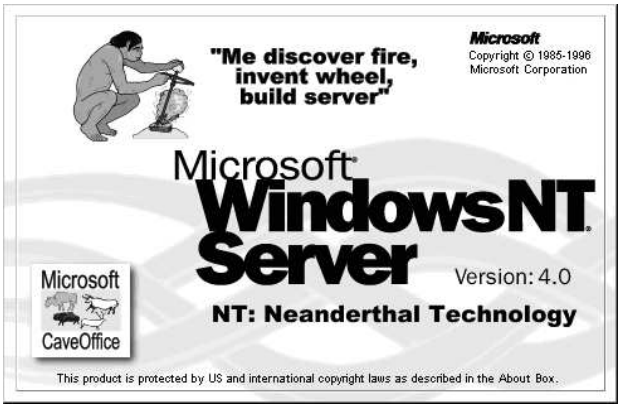
Řešení následujícího problému FAT 32 je velmi humorné. Inu Microsoft chce získat body i na poli vtipů a zábavy :-)

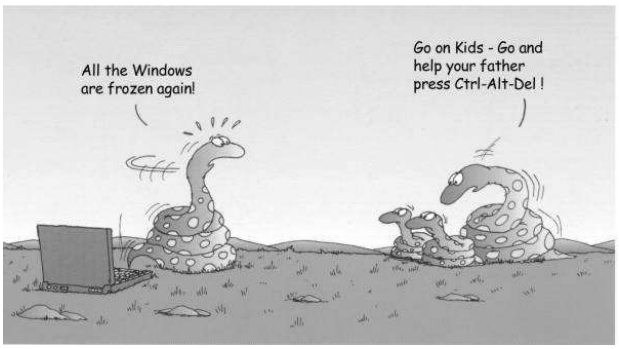
http://support.microsoft.com/support/serviceware/windows/win98/E9T9H7A69.ASP
Folder is slow to open in Windows Explorer due to folder size. Cause:
This problem can occur when opening a folder in Windows Explorer on a drive using the FAT32 file system. This can occur if the total space used by all directory entries in the folder exceeds 32KB.
Solution:
If this problem occurs when opening a folder in Windows Explorer on a drive using the FAT32 file system, move some of the files from their current location to another folder.
(c) Microsoft

A opět jeden „útok“ na Microsoft od našeho čtenáře.

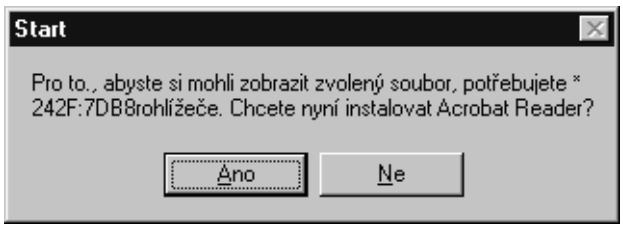
Microsoft random number generator: 0, 0, 0, 4.33e+67, 0, 0, 0…

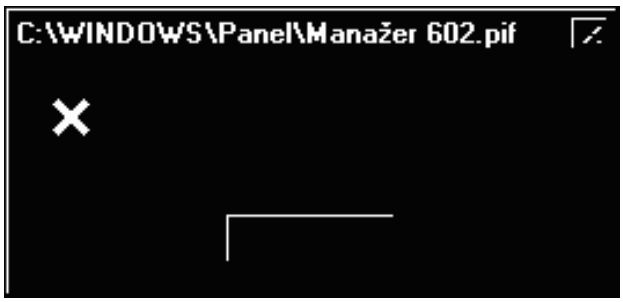
A na závěr ještě několik obrázků.

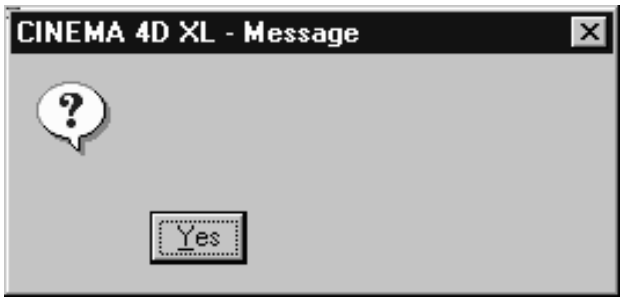




A na úplný závěr snad jenom několik opravdu zajímavých chyb, které nasbírali naši čtenáři na různých operačních systémech.







A už snad na úplný závěr jsem si ponechal jenom několik fotografií, které vznikly při odesílání plyšových tučňáků členům CZLUGu.

Tak takhle nějak asi vypadá SMP server s operačním systémem Linux – moře tučňáků.

Podívejte se, jaký má Dan uklizený pokojíček:

Ale v serverovně má neuvěřitelný nepořádek na pracovním stoje – všude samý tučňák.

Ale jinak oba (Pavel Janík i Dan Ohnesorg) doufáme, že vám tučňáci došli vcelku a pokud se vám tučňáci na těchto fotografiích líbí, objednejte si je buď v databázi členů CZLUGu nebo u společnosti SuSE CR, s.r.o a zažijete s nimi stejnou radost, jako my při jejich balení. No, možná jich budete mít méně :-)

Linuxové noviny a jejich šíření

Linuxové noviny vydává České sdružení uživatelů operačního systému Linux (1) pro své příznivce a sympatizanty. Vlastníkem autorských práv k tomuto textu jako celku je Pavel Janík ml. (Pavel.Janik@linux.cz). Autorská práva k jednotlivým článkům zůstávají jejich autorům.

Tento text může být šířen a tištěn bez omezení. Pokud použijete část některého článku zde uveřejněného v jiných dílech, musíte uvést jméno autora a číslo, ve kterém byl článek uveřejněn.

Linuxové noviny jsou otevřeny každému, kdo by chtěl našim čtenářům sdělit něco zajímavého. Příspěvky (ve formátu čistého textu v kódování ISO 8859-2) posílejte na adresu (2). 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.

Registrované známky použité v tomto textu jsou majetkem jejich vlastníků.

Chtěl bych poděkovat společnosti SuSE CR, s.r.o (3), že nám umožňuje i nadále pracovat na Linuxových novinách.

Linuxové noviny jsou k dispozici také ve formátu HTML na adrese (4). ■

1 České sdružení uživatelů operačního systému Linux
http://www.linux.cz/czlug
2 Adresa redakce
mailto:noviny@linux.cz
3 SuSE CR, s.r.o.
http://www.suse.cz/
4 Linuxové noviny ve formátu HTML
http://www.linux.cz/noviny

Šéfredaktor: Pavel Janík ml.
mailto:Pavel.Janik@linux.cz
zástupce šéfredaktora: David Häring
mailto:dave@ibp.cz
sazba: Ondřej Koala Vácha
mailto:koala@informatics.muni.cz
jazykové korekce: Bohumil Chalupa
mailto:bochal@met.mff.cuni.cz
překlady: Hanuš Adler
mailto:had@articon.cz
převod do HTML: Pavel Juran
mailto:juran@proca.cz