



Abíčko

Časopis serveru AbcLinuxu.cz

Únor 2006



Vychází také na CD-ROM jako příloha časopisu

PC WORLD

Editoriál

Vítejte u čtení časopisu Abíčko.

Abíčko vychází jako měsíční příloha serveru <http://www.abclinuxu.cz> a obsahuje výběr toho nejzajímavějšího obsahu, který zde byl v minulém měsíci publikován. Touto formou chceme předat čtenářům informace v snadno čitelné podobě vhodné i pro tisk.

Cílem serveru <http://www.abclinuxu.cz> je pomáhat všem uživatelům Linuxu, nezávisle na jejich zkušenostech, platformě či použité distribuci. Motorem, který nás pohání vpřed, je idea vzájemné pomoci a spolupráce. Proto i velkou část obsahu tvoří samotní uživatelé. Zapojit se může kdokoliv, tedy i vy.

Na <http://www.abclinuxu.cz> najdete rozsáhlou databázi návodů na zprovoznění hardwaru pod Linuxem, velice aktivní diskusní fórum, podrobné návody a tutoriály, recenze, archiv ovladačů, informace o linuxovém jádře (včetně populárních Jaderných novin) i rozcestník po ostatních linuxových serverech. Novinkou posledních měsíců, která našla brzy odezvu, jsou blogy neboli internetové deníčky. Každý registrovaný uživatel si jej může založit a psát si do něj poznámky nejen o Linuxu.

V neposlední řadě chceme upozornit také na výkladový [slovník pojmů](#) a vznikající [elektronickou učebnici Linuxu](#), na níž se můžete podílet i vy!

Náměty na články zasílejte do konference našich autorů: autori@abclinuxu.cz.

Sponzoring Abíčka a jiné formy reklamy si objednávejte na adrese: reklama@abclinuxu.cz.

Ostatní dotazy směřujte na adresu: literakl@abclinuxu.cz.

Server <http://www.abclinuxu.cz> provozuje firma Stickfish s.r.o., která poskytuje profesionální služby v oblasti Linuxu firmám i jednotlivcům. Zabývá se hlavně bezpečností, instalacemi Linuxu a konfigurací síťových služeb. Více na <http://www.stickfish.cz>.

©2006 Stickfish s. r. o. a autoři článků

Editor a sazba: Vlastimil Ott

Pro nekomerční účely smíte tento dokument jakkoliv šířit v tištěné i digitální podobě. V ostatních případech nás požádejte o svolení na adrese info@abclinuxu.cz.

Typografické konvence

Ve výpisech zdrojových textů mohou být použity znaky `\\`. Značí přechod na nový řádek, který ovšem *není* součástí samotného zdrojového textu, byl přidán editorem z důvodu lepšího vzhledu případně nemožnosti text formátovat bez jejich použití.

Obsah

Editoriál	1
Obsah	2
Kanotix	6
Kanotix aneb vylepšený Knoppix	6
Odkud a co stahovat	6
Kanotix Live	6
Instalace Kanotixu na pevný disk	7
Poinstalační správa	8
Shrnuto	9
Fluxbox	11
1. Úvod	11
2. Instalace	11
2.1 Instalace v distribuci Debian	11
2.2 Instalace v distribuci Fedora Core	11
2.3 Instalace v distribuci Slackware	11
2.4 Instalace ze zdrojových kódů	12
3. Spuštění	12
3.1 Spuštění z textového režimu	12
3.2 Spuštění z login manageru	13
4. Nastavení	13
4.1 Téma vzhledu	13
4.2 Pozadí pracovní plochy	13
4.3 Hlavní nabídka	14
4.4 Klávesové zkratky	15
5. Užitečné programy	15
5.1 GKrellM	15
5.2 Torsmo	16
5.3 iDesk	16
5.4 Nastavení spouštění programů po startu Fluxboxu	16
6. Závěrem	17
Myšlenkové mapy a aplikace FreeMind	18
Co jsou to myšlenkové mapy, jejich praktický význam	18
Aplikace FreeMind se představuje	19
Instalace aplikace FreeMind	19
Java Runtime Environment 1.4	20
Instalace na všech unixových systémech	20
Instalační balíčky pro vybrané distribuce Linuxu	20
Instalace na systémech Microsoft Windows	21
Popis grafického uživatelského rozhraní	21
Hlavní menu	21
Kontextová menu	23
Horní toolbar	23
Levý toolbar	24
Scrollbary	25
Posun a změna měřítka mapy	25

Operace s uzly myšlenkové mapy	25
Vytváření a rušení uzlů	25
Editace uzlů	26
Hypertextové odkazy a obrázky	27
Styl písma a formátování celého uzlu	27
Ikony vkládané do uzlů	28
Operace s hranami myšlenkové mapy	28
Styl hran	29
Barva hran	29
Tloušťka hrany	29
Vyhlazení hran	29
Přídavné hrany	29
Přenos dat mezi FreeMindem a dalšími aplikacemi	29
Možnosti exportu myšlenkových map nebo jejich částí	29
Export myšlenkové mapy do HTML a XHTML	30
Export myšlenkové mapy do formátu PDF	31
Export myšlenkové mapy do formátu SVG	32
Export myšlenkové mapy do rastrového obrázku	32
Export myšlenkové mapy do formátu OpenOffice.org	33
Možnosti importu myšlenkových map	33
Import další myšlenkové mapy	34
Import adresářové struktury	34
Import záložek z aplikace Internet Explorer	34
Import dat z aplikace MindManager	34
Formát souborů aplikace FreeMind	34
Závěr	35
Kalendář dostupný odkudkoliv?	36
Zapnutí podpory WebDAV v Apache	36
Zkouška WebDAV	38
Mozilla Sunbird	38
Vytvoření kalendáře	39
Co s tím, když něco nefunguje?	40
Webový přístup	40
Něco na závěr	41
Distribuce OpenWrt prakticky	42
Úvod	42
Odkazy	42
Systém OpenWrt	43
Na čem lze OpenWrt provozovat	43
Co vlastně obsahuje Asus wl500g deluxe	43
Zkušenosti z provozu	43
Instalace, základní nastavení – download	44
Instalace	44
První přihlášení	46
Paměť nvram	46
Nastavení Wan	46
Nastavení Lan a DHCP	47
Nastavení firewallu	48

Na závěr	49
Pocketlinux 2.0	50
Vznik a určení distribuce	50
Průvodce instalací	50
Poinstalační konfigurace	51
Programové vybavení	52
Správa systému: instalace programů	53
Několik slov závěrem	54
FreeBSD v malej firme	56
Small Business systémy	56
Čo je to FreeBSD?	56
Prečo do firmy zvolit práve FreeBSD?	57
Použitie – prípadová štúdia	58
Získanie inštaláčiek	58
Inštalácia začína	59
Poinštaláčny zásahy	60
Finále	61
The FreeBSD Documentation License	61
RAID – čo to je, ako to funguje a prečo sa tým zaoberať	62
Ako ho realizovať vo FreeBSD	63
Čo robiť v prípade problémov?	63
Xserver – základná konfigurácia	64
Inštalácia a základná konfigurácia KDE	65
Lokalizácia systému a KDE	65
Ešte trochu psychológie	66
Inštalácia CUPS a ostatných potrebných modulov	66
Prvý štart CUPS – overenie činnosti	68
Pridávame do systému tlačiareň	68
Podpora lpd v CUPSe pre aplikácie nepodporujúce CUPS	68
Terminály sa vracajú späť – alebo vývoj ide v špirále	69
Prečo teda terminálové riešenie v malej firme?	69
Licencie komerčných riešení sa môžu predražiť	70
Dajme užívateľom to najlepšie!	71
Topológia siete	71
Konfigurácia na strane servera (X klienta)	71
Konfigurácia na strane terminálu (X servera)	72
Inštalácia aplikácií z balíčkov	72
Inštalácia softwaru z portov	74
Aktualizácia (upgrade) aplikačného softwaru	75
Integrace linuxového serveru do domény Windows 2003	77
Úvod, distribuce, prostředí	77
Základní instalace serveru	77
Ověřování přes Kerberos	78
PAM	79
Konfigurace společné uživatelské databáze v AD	82
Konfigurace LDAP klienta	83
SAMBA a její konfigurace	87
Nastavení ACL	92

Zbývá udělat	92
Stromy v SQL	94
Definice	94
Jaký strom zasadit?	94
Sebereferenční tabulky	94
Genealogické stromy	95
Nested set aneb DFS strom	96
Závěr	98
Testování služeb mailservru telnetem	99
Použité nástroje	99
Telnet	99
SMTP	99
POP3	100
IMAP	102
SSL	103
A to je vše, přátelé	105
Jaderné noviny 335	106
Popis a debata vývojového procesu, 78 e-mailů	106
Hledají se testeři Sharp Zaurus-5500, 7 e-mailů	107
Nový souborový systém eCryptFS, 42 e-mailů	108
Podpora řadiče Secure Digital Host, 1 e-mail	108
Zprávičky	109

Kanotix

Martin Tesař

Live distribuce Linuxu, původně založena na legendárním Knoppixu, ale více zaměřena na laické uživatele. Článek se věnuje poslední verzi (2005-04) s kernelem 2.6.14.5 a KDE 3.4.3, včetně instalace na pevný disk.

Nedávno jsem si po delší době vzpomněl na distribuci, která mi kdysi posloužila coby rychlé a spolehlivé řešení, když jsem něco chtěl nainstalovat na poněkud starší notebook s podivným hardwarem, na který se špatně sháněly ovladače i pro Windows, a kde původně první volba Mandrake (tuším s ještě příliš novým jádrem 2.6) způsobovala podivné zkratky větráku. Jelikož to všechno mělo být ještě ke všemu němčině, zvědavě a odevzdaně jsem spustil náhodně nalezené „německé“ live CD Kanotix a hned ho také nainstaloval na harddisk. Bingo! Pak už si vzpomínám jen na to, že si na žádné problémy nevzpomínám.

Kanotix aneb vylepšený Knoppix

Kanotix ani jeho tvůrce (Jörg Schirottke alias „Kano“) se nijak netají skutečností, že jde vlastně o úpravu legendárního průkopníka live distribucí Knoppixu. Vznikl z touhy Knoppix ještě vylepšit, tj. poopravit některá problematická místa Knoppixu, na která si stěžují uživatelé ve fórech, a také proměnit primárně *záchrannou* a vývojářskou live distribuci v pohodlný uživatelský systém, již od raných verzí s dobrou podporou instalace na pevný disk.

První „preview release“ vyšlo na Štědrý den roku 2003, na který následovala v roce 2004 řada vydání pod názvem „Bug Hunter“. Nejnovější verze označená pragmaticky pouze rokem a čtvrtletím 2005-04 byla vydána právě v poslední den minulého roku a skrývá kromě známých nástrojů Knoppixu vnitřnosti aktuálního Debianu SID, KDE 3.4.3 (alternativně je k dispozici okenní manažer IceWM), vlastní patche a řadu vybraných aplikací. Pokud vím, ještě se neobjevila iniciativa „Kanotix“, která by se po vzoru Gnoppixu pokusila o totéž s důrazem na desktopové prostředí Gnome.

O distribuci je pečováno (vycházejí updaty a upgrady) a na webové stránce projektu se objevují aktuální zprávy, návody, řešení či upozornění na problémy, např. ovladačů ATI s případným upgradem X.Org, živá jsou také [diskusní fóra](#) [1] (v němčině a angličtině) a vývojáři včetně „otce“ distribuce jsou (prý) často k dosažení na [irc.freenode.net](#) v kanále [#kanotix](#); na ten se, stejně jako na [homepage Kanotixu](#) [2], dostanete rychle a jednoduše kliknutím na připravenou ikonku, které jsou umístěny na ploše (v KDE).

Odkud a co stahovat

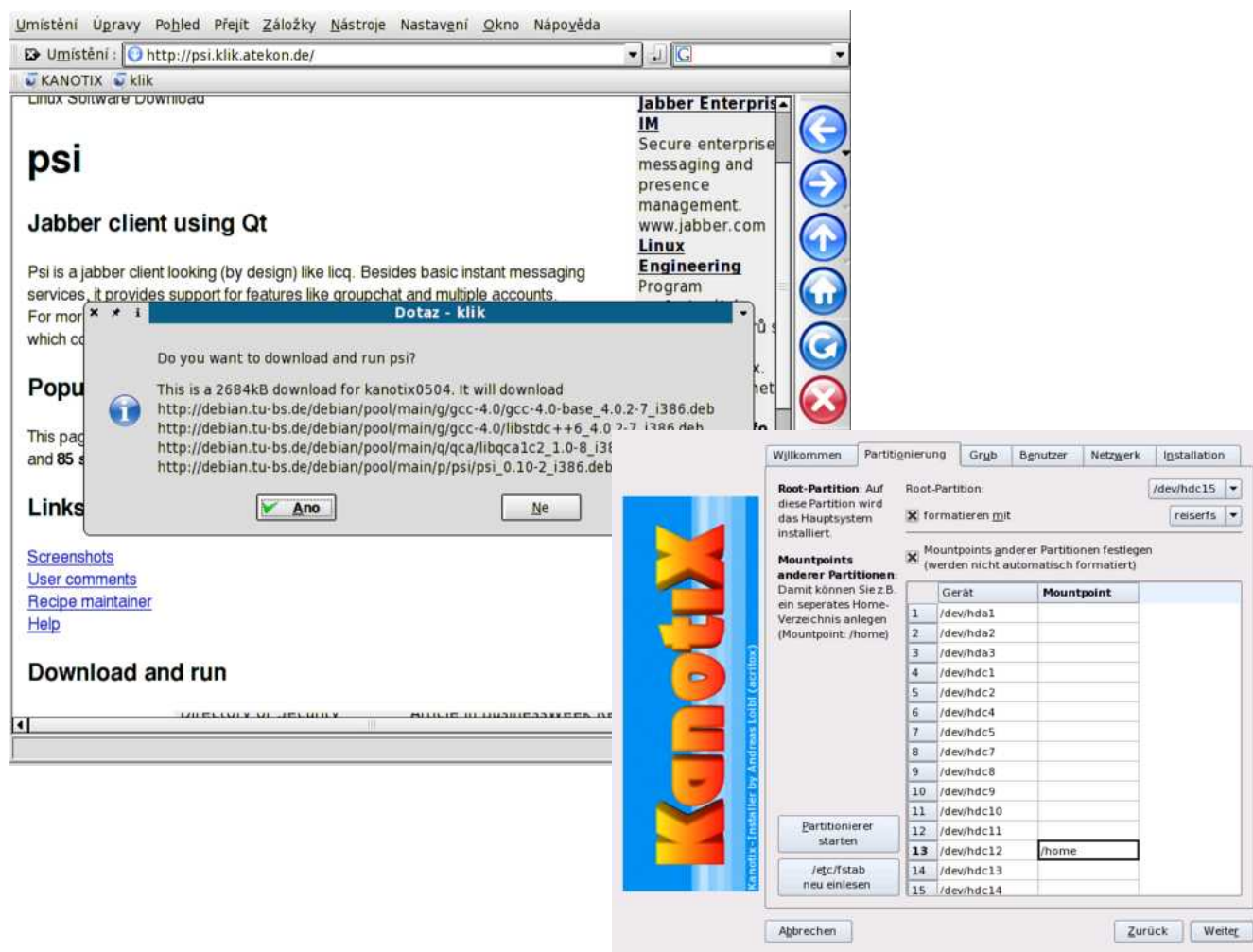
Kanotix je možné stáhnout jako ISO obraz ve třech variantách: 32bitové buď plné verzi nebo jako LITE, odlehčené (o některé velké balíky jako OpenOffice.org, hru SuperTux aj.) nebo 64bitové LITE verzi, např. z [debian.tu-bs.de/kanotix](#) [3] nebo z nějakého [mirroru](#) [4].

Kanotix Live

Kanotix bootuje rychle se skvělou detekcí pravděpodobně velmi širokého spektra hardware (pomocí [hwsetup-ng](#)). Od začátku vývoje sází Kanotix na zavaděč GRUB [5], což tehdy ještě zdaleka nebylo rozšířeným zvykem (já jsem se tuším potkal v Kanotixu s GRUBem vůbec poprvé), který pomocí přehledného menu umožňuje pohodlný start různých variant opatchovaného [kernelu 2.6.14.5](#) [6] jakož i možnost ručního zadání široké [palety parametrů](#) [7] (cheatcodes). V mém případě (CD-RW 4x-12x, PII 400 MHz, 320 MB RAM) trvá celá startovací procedura od spuštění zavaděče až do funkčního prostředí KDE (včetně otevření úvodní stránky v Konqueroru) přesně 3 minuty.

Jak již bylo řečeno, startuje Kanotix do mírně upraveného KDE 3.4.3, doplněného o menu Debianu a nástrojů Knoppixu a Kanotixu. Samozřejmostí je dnes již běžná distribuční výbava jako vypalovací program K3b, kancelářský balík OpenOffice.org 2.0.0-5 (ve variantě LITE ho nahrazuje KOffice 1.4.2), programy ke sdílení souborů Bittorrent/Bittornado, pokečové programky XChat, Kopete, poštovní klient Mutt, nadto užitečné nástroje přístupu k sítím (Samba, Wlan-karty, NdisWrapper, Fritz!-Card DSL aj.), záchrany dat (QtParted, Partimage, Testdisk aj.), webové prohlížeče (Konqueror, Firefox 1.5), FTP-klient, server Apache, Firewall GuardDog, program pro internetové telefonování Skype (1.2), editor GVIM (6.4.6), vývojový nástroj pro webové stránky Quanta Plus, GIMP (2.2.10), Python, CUPS – to jsou jen vybrané z řady [předinstalovaných programů](#) [8]. Kanotix používá samozřejmě [udev](#) [9] a podle dostupných recenzí detekuje a konfiguruje systém bez problému digitální kamery, PDA a úložná zařízení „on the fly“.

Kanotix nabízí nadto díky účasti v projektu [Klik](#) [10] a díky podpoře [UnionFs](#) [11] možnost jakési „live instalace“ nových programů z repozitáře „Kliku“ jednoduchým kliknutím přes webové rozhraní (včetně závislostí), a to dokonce pro uživatele bez přístupu ke kontu `root` nebo právě při běhu live-CD. Takový balíček software se totiž navenek chová jako komprimovaný soubor s příponou `.cmg`, který můžete zálohovat kamkoliv si jen přejete, např. na plochu KDE a program pak spustíte prostým kliknutím.



Instalace Kanotixu na pevný disk

Ačkoliv primárně Kanotix vznikl a slouží jako live CD, je instalace na pevný disk maximálně usnadněna: probíhá přímo z běžící live distribuce, aniž ji musíte opustit. Prostě nabootujete live CD a přes položku KDE menu `Kanotix-installer` spustíte přehledný instalační dialog, který vás provede několika málo nutnými kroky základní konfigurace. Předtím se ujistěte, že máte alespoň 2 GB místa

na pevném disku, neboť asi tolik kompletní instalace zabere a z pochopitelných důvodů nelze volit pouze instalaci částečnou, operační paměti byste měli mít pro rozumnou práci aspoň 256 MB.

Tím prvním krokem konfigurace je výběr **zavaděče** [12], a ačkoliv Kanotix dává nepokrytě přednost GRUBu, nabízí zájemcům i klasické LILO [13]. Nelze zvolit možnost zavaděč neinstalovat, patrně z bezpečnostních důvodů, ale chceme-li si ponechat původní zavaděč na našem počítači v MBR, stačí zvolit možnost instalovat na kořenový oddíl (root partition).

V přehledné tabulce zvolíte podle přání kořenový oddíl (který bude formátován zvoleným souborovým systémem (předvolen je ReiserFS) a zvolit případně umístění přípojných bodů (mount points) jako např. `/home` na jiné oddíly pevného disku. Některá speciálnější konfigurační nastavení instalace lze v případě potřeby změnit pouze v konfiguračním souboru `/home/knoppix/.knofig` (sic!), na což budete upozorněni, ale nemyslím, že by tuto možnost mnoho uživatelů chtělo využít. Kanotix také automaticky rozezná, pokud máte už někde na disku linuxový swapový oddíl, a použije ho jak při běhu live distribuce, tak při instalaci na pevný disk. Další kroky jako zadání hesla administrátora a uživatele apod. není třeba podrobněji probírat.

Samotná instalace je pak velmi rychlá: u mě díky nevykonnému systému (PII 400 MHz 320 MB RAM) trvalo formátování kořenového oddílu (4 GB) a hlavně dekomprimace a kopírování z CD-ROM celkem 22 minut 17 sekund, ale v manuálu Kanotixu se hovoří o obvyklém trvání v délce 5–10 minut. Čas vím tak přesně, protože ho instalátor po skončení hrdě ohlašuje :-). Bude-li vám trvat instalace déle jako mně, nevyděste se, když po chvíli obrazovka náhle zhasne: to se jen aktivoval screensaver, neboť i během instalace se nacházíte v normálním pracovním prostředí spuštěné live distribuce.

Poinstalační správa

Po instalaci na pevný disk by některé mohlo překvapit, že Kanotix zapomene původní (většinou automatickou) konfiguraci přístupu do sítí, a je třeba konfigurace nová. Stejně jako instalace je i tato procedura usnadněna skriptem, který je možné spustit z extra KDE-menu *Network Card Configuration* a prostě odpovídat na otázky. Užíváte-li si například výhody DHCP, stačí odpovědět pouze na dvě: zda ho máte a zda ho chcete spouštět automaticky při startu. Hotovo, jste (a zůstanete) připojeni. Také instalace proprietárních ovladačů grafických karet je usnadněna skripty. Stačí v konsoli (`Ctrl+Alt+F2`) jako `root` provést následující povely (ze všeho nejdříve pro jistotu `init 3`, přepnutí do runlevelu 3, tedy ukončení grafického prostředí):

```
init 3
update-scripts-kanotix.sh
install-radeon-debian.sh
```

Pro majitele karet nVidia:

```
init 3
update-scripts-kanotix.sh
install-nvidia-debian.sh
```

Automaticky se najde poslední verze debianního balíčku příslušného ovladače, ten se stáhne a nainstaluje. Skript nakonec sám spustí grafické prostředí, takže nenastane-li nějaká nepředvídatelná nehoda, zbývá uživateli už jen přihlásit se a užívat si nového ovladače (nejlépe ihned vyzkoušet 3D funkci vyvoláním `glxgears` z terminálu).

Nabízena ke stažení jsou průběžně také opatchovaná a předkonfigurovaná **aktuální jádra** [14] – při výměně je však třeba mít na paměti, že může být občas nutné vyměnit také některé nainstalované ovladače či moduly za novější! K pohodlnému instalování a upgradování nabízí Kanotix vlastní

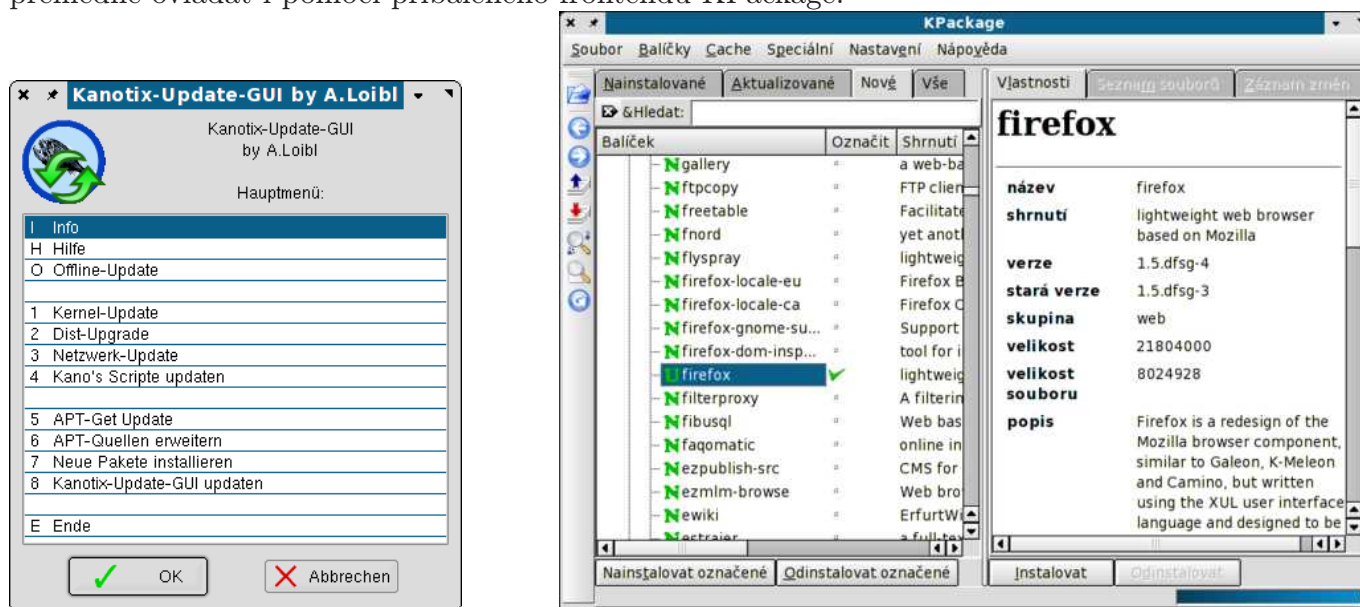
Kanotix-Update-GUI, soubor skriptů, které lze spouštět pomocí integrovaného KDE-menu a grafických („klikacích“) dialogů:

Instalace nových programů lze v Kanotixu samozřejmě také provádět jako v běžném Debianu rovnou pomocí správce balíčků APT (v souboru `/etc/apt/sources.list` jsou předvybrané zdroje, které můžete v případě potřeby doplnit o vlastní; podrobné APT HOWTO, dokonce v češtině, najdete v [dokumentaci](#) [15] Debianu).

Pro dobrodružné povahy by mohlo být zajímavé otestovat integritu Kanotixu povely

```
apt-get update && apt-get dist-upgrade
```

a upgradovat tak kompletní systém na horké novinky z repozitářů Debianu SID, jako třeba X.Org momentálně na verzi 6.9 nebo KDE na 3.5. I když je i takový upgrade „oficiálně“ podporovaný a doporučovaný, předem si rozmyslete možné následky. Tak např. kvůli podpoře proprietárních ovladačů grafických karet ATI používá Kanotix vědomě X.Org stále ve verzi 6.8.2 a spontánní výměna za 6.9 by nemusela dopadnout nejlépe. Tyto problémy se však vyřeší postupně vydáváním nových ovladačů, stačí sledovat homepage a diskusní fora Kanotixu. Někteří možná také uvítají, že APT lze přehledně ovládat i pomocí přibaleného frontendu KPackage:



I po instalaci na pevný disk samozřejmě funguje výše zmíněný „Klik“- pak zkuste poslouchat nářky, že v Linuxu je instalace programů obtížná! A ještě poznámka pro případné přeběhlíky ze systému MS Windows, kteří jsou zvyklí po nové instalaci systému zahájit únavné instalační maratóny potřebných programů: vzhledem k obsahu Kanotixu byste se měli nejdřív pořádně rozhlédnout, jestli požadovanou aplikaci už dávno nemáte nainstalovanou, navzdory těm neuvěřitelným deseti minutám, co instalace trvala. A aby nevznikl prostor pro nějaká nevyplněná přání, lze pro ty línější nebo pro offline systémy dokonce [upgradovat kompletní distribuci](#) [16] z nového liveCD Kanotixu, až zase nějaké vyjde.

Shrnutí

Kanotix vstupuje do třetího roku své existence jako vyžralá live distribuce, která se dá nadto velice rychle a pohodlně instalací na pevný disk přeměnit na funkční Debian SID. Místo obvyklé ruční práce (např. instalace proprietárních ovladačů grafiky nebo zvýšení rozlišení X Window systému) nabízí vlastní skripty, které pouhým příkazem udělají práci za uživatele. Také instalace nových programů je díky řadě poskytovaných nástrojů a jejich předkonfiguraci hračkou.

Snad jediná, ovšem pro české a slovenské uživatele dost podstatná, nevýhoda Kanotixu je zatím nepříliš pokročilá lokalizace do národních jazyků, i když systém lze samozřejmě dodatečně počestit nebo poslovenštit jako kteroukoliv jinou instalaci Debianu. Myslím, že kdyby se chtěl někdo (se znalostmi angličtiny a nejlépe i němčiny) zapojit do [překladatelského týmu](#) [17] webových stránek a manuálů, bude jistě srdečně vítán, stejně jako kdyby se někdo ujal počestění či poslovenštění této kvalitní distribuce.

Krátce a stručně: máte-li potřebné jazykové znalosti a hledáte-li dobré live CD nebo toužíte-li po přívětivé, vyzkoušené, kvalitní a „nabouchané“ instalaci aktuálního Debianu s předvoleným KDE, a to všechno do 20 minut, pořídte sobě nebo svým přátelům Kanotix.

Odkazy

- [1] <http://kanotix.com/PNphpBB2.html>
- [2] <http://kanotix.com>
- [3] <http://debian.tu-bs.de/kanotix>
- [4] <http://kanotix.com/WebLinks-req-viewlink-cid-1.html>
- [5] <http://www.abclinuxu.cz/slovník/grub>
- [6] <http://kernel.org/pub/linux/kernel/v2.6/ChangeLog-2.6.14.5>
- [7] http://kanotix.com/FAQ-id_cat-62.html#q243
- [8] <http://kanotix.com/files/kanotix/kanotix-packages.txt>
- [9] <http://www.abclinuxu.cz/slovník/udev>
- [10] <http://klik.atekon.de/>
- [11] <http://www.fsl.cs.sunysb.edu/project-unionfs.html>
- [12] <http://www.abclinuxu.cz/slovník/zavadec>
- [13] <http://www.abclinuxu.cz/slovník/lilo>
- [14] <http://debian.tu-bs.de/project/kanotix/kernel/>
- [15] <http://www.debian.org/doc/user-manuals.en-gb.html#apt-howto>
- [16] http://linux.kopporama.de/de/Kanotix_HB.html#x1-320006.3
- [17] <http://kanotix.com/Article26.html>

Fluxbox

Jaromír Hradílek

Fluxbox je minimalistický, hardwarově nenáročný a příjemný správce oken pro X window system vycházející z Blackboxu. Přečtěte si o jeho výhodách, instalaci, spouštění, konfiguraci a šikovných doplňkových programech.

1. Úvod

Než jsem na Fluxbox přešel, nějakou dobu jsem byl spokojeným uživatelem [Window Makeru](#) [1]. Fluxbox mě však zaujal natolik, že jsem u něj už zůstal. Proč? Některé jeho vlastnosti se pokusím shrnout v následujícím výčtu:

- *Vysoká konfigurovatelnost.* Rád mám možnost si jednoduše nastavit věci tak, jak mi vyhovují, a Fluxbox mi v tomto dává široké spektrum možností.
- *Možnost vkládat okna do sebe.* Tato věc je naprosto úžasná, neboť mohu libovolná okna sloučit do sebe a přepínat mezi nimi pomocí záložek.
- *Panel.* Snadno lze nastavit jeho šířka, umístění a poloha jednotlivých položek, jako je indikátor aktuální plochy, hodiny, nebo panel spuštěných aplikací.
- *Přepínání pracovních ploch pomocí kolečka myši.*
- *Plynulé přesouvání oken mezi jednotlivými plochami přetažením k okraji obrazovky.*
- *Podpora dockapps* známých z Window Makeru.

Co mi naopak oproti Window Makeru trochu chybí je přichytávání oken, ale je možné, že si to jen neumím nastavit :-).

V následujícím textu se vás pokusím provést instalací a základním nastavením tohoto prostředí, v závěru vás pak seznámím se třemi užitečnými aplikacemi. Zkušený uživatelé tohoto prostředí se s největší pravděpodobností nedozví nic nového, začátečníkům by se však tyto informace mohly hodit. Zároveň by tento článek mohl přesvědčit ty, kteří se vyzkoušet Fluxbox zatím chystají.

2. Instalace

Pokud to vaše distribuce umožňuje nebo nemáte-li ke kompilaci jiný důvod, doporučuji použít jejího balíčkovacího systému a příslušných utilit. V následujícím textu popíši instalaci v systémech Debian, Fedora Core a Slackware (tj. v těch, které mám zrovna po ruce), pro ostatní je však postup podobný (vizte dokumentaci a manuálové stránky).

V následujícím textu také předpokládám jisté zkušenosti s prací s operačním systémem GNU/Linux a přístup k superuživatelskému účtu (root). Pokud některý z těchto předpokladů nesplňujete, požádejte o pomoc správce svého systému nebo někoho zkušenějšího.

2.1 Instalace v distribuci Debian

V distribuci Debian a od něj odvozených (Ubuntu, Kubuntu, Danix, Knoppix apod.) využijeme služeb utility apt-get:

```
root@debian:~# apt-get install fluxbox
```

2.2 Instalace v distribuci Fedora Core

V distribuci Fedora Core využijeme nástroje yum:

```
root@fedora:~# yum install fluxbox
```

2.3 Instalace v distribuci Slackware

Používáte-li distribuci Slackware, jedním ze způsobů instalace je balíček stáhnout, a to buď ze stránek slackware.it/en/pb/ [2] nebo www.linuxpackages.net [3]. Poté se jako root přesuňte do adresáře se staženým *.tgz balíkem a spusťte instalaci:

```
root@slackware:~/packages# installpkg fluxbox-0.9.14-i486-1arf.tgz
```

Verze balíčku se může (a pravděpodobně bude) lišit. Pokud jsou splněny všechny závislosti a vše proběhne v pořádku, je Fluxbox nainstalován a připraven k použití.

2.4 Instalace ze zdrojových kódů

Pokud pro vaši distribuci balík neexistuje nebo k tomu máte jiné důvody (balík je zastaralý, ...), je možné si Fluxbox zkompilevat ze zdrojových kódů. V tom případě si obstarajte ze stránek www.fluxbox.org [4] poslední verzi, přesuňte se do adresáře se staženým *.tar.bz2 (*.tar.gz) archivem a spusťte:

```
blackened@debian:~/download$ tar xvjf fluxbox-0.1.14.tar.bz2
```

popř.:

```
blackened@debian:~/download$ tar xvzf fluxbox-0.1.14.tar.gz
```

Tím došlo k rozbalení archívu; podotýkám, že verze balíku se může lišit. Nyní se přesuňte do vzniknuvšího adresáře, spusťte konfigurační skript a počkejte na jeho dokončení:

```
blackened@debian:~/download$ cd fluxbox-0.1.14
blackened@debian:~/download/fluxbox-0.1.14$ ./configure
```

Pokud vše proběhne v pořádku, spusťte samotný překlad aplikace:

```
blackened@debian:~/download/fluxbox-0.1.14$ make
```

Po úspěšném přeložení je pak potřeba zalogovat se jako root, přesunout se zpět do adresáře se zdrojovými kódy a spustit příkaz `make install`:

```
blackened@debian:~/download/fluxbox-0.1.14$ su -- root
Password:
root@debian:~# cd ~blackened/download/fluxbox-0.1.14/
root@debian:/home/blackened/download/fluxbox-0.1.14# make install
```

Proběhly-li výše uvedené kroky bez obtíží, měl by být Fluxbox ve vašem systému úspěšně nainstalován.

3. Spuštění

3.1 Spuštění z textového režimu

V případě, že váš systém startuje do textového režimu, je třeba definovat, který správce oken se má spustit po zadání příkazu `startx`. Toho docílíte editací souboru `/.xinitrc`, konkrétně přidáním řádku na jeho konec: `exec /usr/bin/fluxbox`

Pokud soubor `.xinitrc` ve vašem domovském adresáři ještě neexistuje, nebojte se jej vytvořit. Cesta ke spustitelné aplikaci se může lišit, aktuální umístění zjistíte pomocí příkazu `which`:

```
blackened@debian:~$ which fluxbox
/usr/bin/fluxbox
```

Nyní by nic nemělo bránit startu do grafického režimu:

```
blackened@debian:~$ startx
```

3.2 Spuštění z login manageru

V případě, že váš systém startuje přímo do grafického režimu a používáte tedy nějaký správce přihlášení (obvykle `gdm`, `kdm` nebo `xdm`), je třeba zajistit, aby byl Fluxbox obsažen v nabídce. Ve většině případů se o to postará instalátor. Pokud se tak však nestane, je třeba do adresáře `/usr/share/xsessions/` přidat soubor `fluxbox.desktop` ručně, a to například s následujícím obsahem:

```
[Desktop Entry]
Encoding=UTF-8
Name=Fluxbox
Comment=Lehky a vykonny spravce oken
Exec=fluxbox
Terminal=False
TryExec=fluxbox
Type=Application

[Window Manager]
SessionManaged=true
```

Po restartu X serveru byste již měli mít položku Fluxbox v nabídce.

4. Nastavení

4.1 Téma vzhledu

Pokud jste se nezalekli ve většině distribucí přišerného výchozího vzhledu a Fluxbox v panické hrůze neinstalovali, můžeme se podívat, jak jej přizpůsobit k obrazu svému.

První možností jsou předdefinované systémové styly, jež byste měli najít někde v hlavním menu (které vyvoláte stiskem pravého tlačítka myši kdekoli na volné ploše). Pokud vám žádný z nabídnutých stylů nevyhovuje (nebo jste jen zvědaví na další možnosti), můžete se vypravit na průzkum stránek themes.freshmeat.net [5].

Instalace probíhá jednoduše rozbalením staženého archívu do adresáře `~/.fluxbox/styles/`. Pokud se vám nový styl neobjeví v nabídce stylů v menu, je třeba dopsat do souboru `~/.fluxbox/menu` následující řádky:

```
[submenu] (Uzivatelске styly) {}
  [stylesdir] (~/.fluxbox/styles)
[end]
```

4.2 Pozadí pracovní plochy

Ne vždy nám musí vyhovovat výchozí pozadí k danému stylu, nebo třeba máme nějaké oblíbené, které bychom chtěli mít i ve Fluxboxu. Ke změně pozadí slouží příkaz `fbsetbg` a jeho nejjednodušší použití je prosté: `fbsetbg dokumenty/wallpapers/Asteroidtest-by-DKF.jpg`

Jak se bude obrázek chovat (zda je vycentrován, roztažen, zmenšen, ...) můžeme ovlivnit zadanými parametry, vizte `man fbsetbg`. Toto má jednu nevýhodu: po restartu Fluxboxu o naše pozadí přijdeme a budeme jej muset nastavovat znovu. Pokud chceme, aby byla nějaká tapeta pevně spjata s určitým stylem, můžeme na konec definičního souboru stylu přidat následující řádek:

```
rootCommand: /usr/bin/fbsetbg -c ~/.fluxbox/backgrounds/Planetarium_by_sican.jpg
```

Adresa obrázku i umístění programu `fbsetbg` se pochopitelně mohou lišit. Připomínám, že uživatelské styly se nachází v adresáři `~/.fluxbox/styles/`.

4.3 Hlavní nabídka

V předchozí části jsem už zlehka nakouzl editaci hlavního menu, teď se na jeho strukturu podíváme blíže. Jak už jsem řekl, konfigurační soubor je umístěn v `~/.fluxbox/menu`. K dispozici máme následující příkazy:

```
[begin] (Nadpis menu)
[submenu] (Nadpis submenu)
[exec] (Název aplikace) {/cesta/k/programu}
[include] (/cesta/k_souboru/s_menu)
[end]
[nop] (-----)
[separator]
[workspaces] (Název submenu s~pracovními plochami)
[stylesdir] (/cesta/k_adresari/se_styly)
[config] (Název submenu s~konfigurací fluxboxu)
[restart] (Restart Fluxboxu)
[exit] (Konec sezení ve Fluxboxu)
```

Věřím, že většina příkazů už ze své struktury prozrazuje, k čemu slouží. K ostatním jen krátké vysvětlení:

- `[nop]` – Slouží jako oddělovač, do závorek je možné uvést libovolný text. Tento bude uveden v menu, nespouští však žádný příkaz.
- `[separator]` – Oddělovač v podobě horizontální linky.

Pro lepší představu krátký výřez z mého menu:

```
[begin] (Fluxbox 0.9.14)
[exec] (xterm) {xterm -fg gray -bg black -fn xterm -fn terminus-iso8859-2-12}
[submenu] (Audio)
  [exec] (beep media player) {beep-media-player}
  [exec] (aumix) {xterm -fg white -bg black -e aumix}
[end]
[submenu] (Fluxbox)
  [config] (Konfigurace)
  [submenu] (Systemove styly) {}
    [stylesdir] (/usr/share/fluxbox/styles)
  [end]
```

```
[submenu] (Uzivatel'ske styly) {}
  [stylesdir] (~/.fluxbox/styles)
[end]
[workspaces] (Pracovni plochy)
[restart] (Restart)
[end]
[separator]
[exit] (Konec)
[end]
```

4.4 Klávesové zkratky

Nastavení klávesových zkratk se nachází v souboru `~/.fluxbox/keys` a má jednoduchou strukturu: `kombinace kláves :akce`. Jelikož příklad je mnohdy lepší než spousta slov, přikládám výpis mého nastavení:

```
Control Mod1 Right :nextworkspace
Control Mod1 Left :prevworkspace
Control F1 :Workspace 1
Control F2 :Workspace 2
Control F3 :Workspace 3
Control F4 :Workspace 4
Control F5 :Workspace 5
Control F6 :Workspace 6
Control F7 :Workspace 7
Control F8 :Workspace 8
Control F9 :Workspace 9
Control F10 :Workspace 10
Control F11 :Workspace 11
Control F12 :Workspace 12
Mod1 Tab :nextwindow
Control Tab :prevwindow
Mod1 F1 :rootmenu
Mod1 F4 :close
```

Popis jednotlivých akcí hledejte v dokumentaci.

5. Užitečné programy

Samotný Fluxbox je sice moc hezká věc, ale po nějaké chvíli možná začnete mít dojem, že mu něco chybí. V následujícím textu předvádím 3 programy, které by vás mohly tohoto dojmu zbavit.

5.1 GKrellM

GKrellM je velice šikovný monitor systému a nepochybuji, že ho většina z vás dávno používá. Na instalovat jej můžete buď pomocí balíčkovacího systému, nebo jej stáhnout ze [stránek projektu](#) [6]. A co dovede? Zobrazovat název systému, datum a čas, senzory teploty, větráku a napětí, stav baterie u notebooku, monitor procesoru, procesů, disků, síťových zařízení, poštu, uptime, stav využití paměti

(RAM i SWAP). Kromě toho umožňuje i pohodlné připojování/odpojování zařízení (více na výše uvedených stránkách). Rozsáhlou sbírku témat vzhledu pak najdete na stránkách www.muhri.net [7].

5.2 Torsmo

```
asus - Debian GNU/Linux, 2.6.8-2-386
Processes: 64 Running: 1
CPU Usage: 0%
RAM Usage: 157M/242M - 64%
Swap Usage: 0/510M - 0%

eth0
Download speed: 0 k/s Upload: 0 k/s
Total download: 2.5M Upload: 676k

eth2
Download speed: 0 k/s Upload: 0 k/s
Total download: 0 Upload: 0

Current uptime: 4h 0m
```

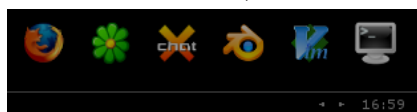
Torsmo je další z řady monitorů systému. Oproti výše uvedenému GKrellM je sice ve svých schopnostech podstatně skromnější, přesto však užitečný (a taky velice dobře vypadá). Získat jej můžete opět prostřednictvím svého balíčkovacího systému nebo na [domovské stránce](#) [8]. Veškerá konfigurace probíhá editací textového souboru `~/torsmorc`. Ten je velice dobře okomentovaný a nemáte-li potíže porozumět anglickému textu, je jeho přizpůsobení otázkou pár minut.

5.3 iDesk

Pokud máte rádi na své ploše ikony, existuje hned několik programů, které vám je zprostředkují. Z těch, které jsem zkoušel, mi zatím nejvíce vyhovuje [iDesk](#) [9]. Jeho chování můžete ovlivnit editací souboru `~/ideskrc`. Nastavení jednotlivých ikon na ploše pak zajišťují soubory v adresáři `~/idesktop/`. Ty mají příponu `*.lnk` a každý zastupuje právě jednu položku. Jeho struktura je například takováto (vypis obsahu souboru `~/idesktop/gvim.lnk`):

```
table Icon
  Command: gvim
  Icon: /usr/share/icons/crystalsvg/32x32/apps/gvim.png
  Width: 32
  Height: 32
  X: 925
  Y: 699
end
```

Pro více informací doporučuji prohlédnout manuálovou stránku (`man idesk`), konkrétně příklady v ní odkazované (v mém případě umístěné v `/usr/share/doc/idesk/examples/`).



5.4 Nastavení spouštění programů po startu Fluxboxu

Pokud vás některá z výše uvedených aplikací zaujala, budete pravděpodobně chtít, aby se spouštěla spolu s vaším správcem oken. V nejnovějších verzích fluxboxu je možné tyto aplikace jednoduše připsat do startovacího skriptu `~/fluxbox/startup`. Ten je dobře okomentovaný a neměl by být problém najít správné místo, kam řádek s programem umístit. Samotný program pak zapíšeme ve tvaru:

```
/usr/bin/gkrellm &
/usr/bin/torsmo &
```

Na znak `&` byste neměli zapomenout, slouží ke spuštění programu na pozadí. Pokud jste takto učinili a aplikace se přesto nespouští, máte nejspíše verzi, která tento skript nepodporuje (kupříkladu aktuální verze v repozitářích Debianu Sarge). Po chvilce hledání jsem v diskusním fóru objevil řešení v zapsání následujících řádků do souboru `~/fluxbox/apps`:

```
[startup] {gkrellm}  
[startup] {torsmo}  
gkrellm &  
torsmo &
```

Toto opatření bylo v mém případě funkční. Pokud se vám požadované aplikace přesto nespouštějí, zkuste použít vyhledávání na těchto stránkách.

6. Závěrem

Fluxbox je sice lehký, přesto však příjemný správce oken. Pro své nízké hardwarové nároky je velice vhodný pro instalaci na starších strojích (odzkoušeno na staříčkém Pentiu 100 MHz, 32 MB RAM). Po přečtení tohoto článku byste měli být schopni nastavit si základní chování prostředí Fluxboxu tak, aby pokud možno vyhovovalo vašim představám. Bohužel není možné v tomto rozsahu popsat všechny možnosti nastavení; pokud vás to zajímá, zkuste si projít konfigurační soubory v `~/.fluxbox/`, popř. níže zmíněnou dokumentaci.

Vám, kdo jste si jej nainstalovali, přeji, aby vám sloužil k vaší plné spokojenosti tak, jako už nějakou dobu mně. Pokud se k tomu teprve rozhodujete, upřímně doufám, že vás tento článek neodradil.

Odkazy

- [1] <http://www.windowmaker.org/>
- [2] <http://slackware.it/en/pb/>
- [3] <http://www.linuxpackages.net/>
- [4] <http://www.fluxbox.org/>
- [5] <http://themes.freshmeat.net>
- [6] <http://gkrellm.net/>
- [7] <http://www.muhri.net/>
- [8] <http://torsmo.sourceforge.net/>
- [9] <http://idesk.sourceforge.net/>

Myšlenkové mapy a aplikace FreeMind

Pavel Tišnovský

Seriál, ve kterém se budeme zabývat principem tvorby myšlenkových map a také způsoby jejich použití při praktické práci. Také si podrobně popíšeme aplikaci FreeMind určenou pro práci s myšlenkovými mapami.

Z technických důvodů (velikost výsledného PDF souboru) zde nejsou uvedeny zdaleka všechny obrázky, které autor použil. Pozn. editora.

Co jsou to myšlenkové mapy, jejich praktický význam

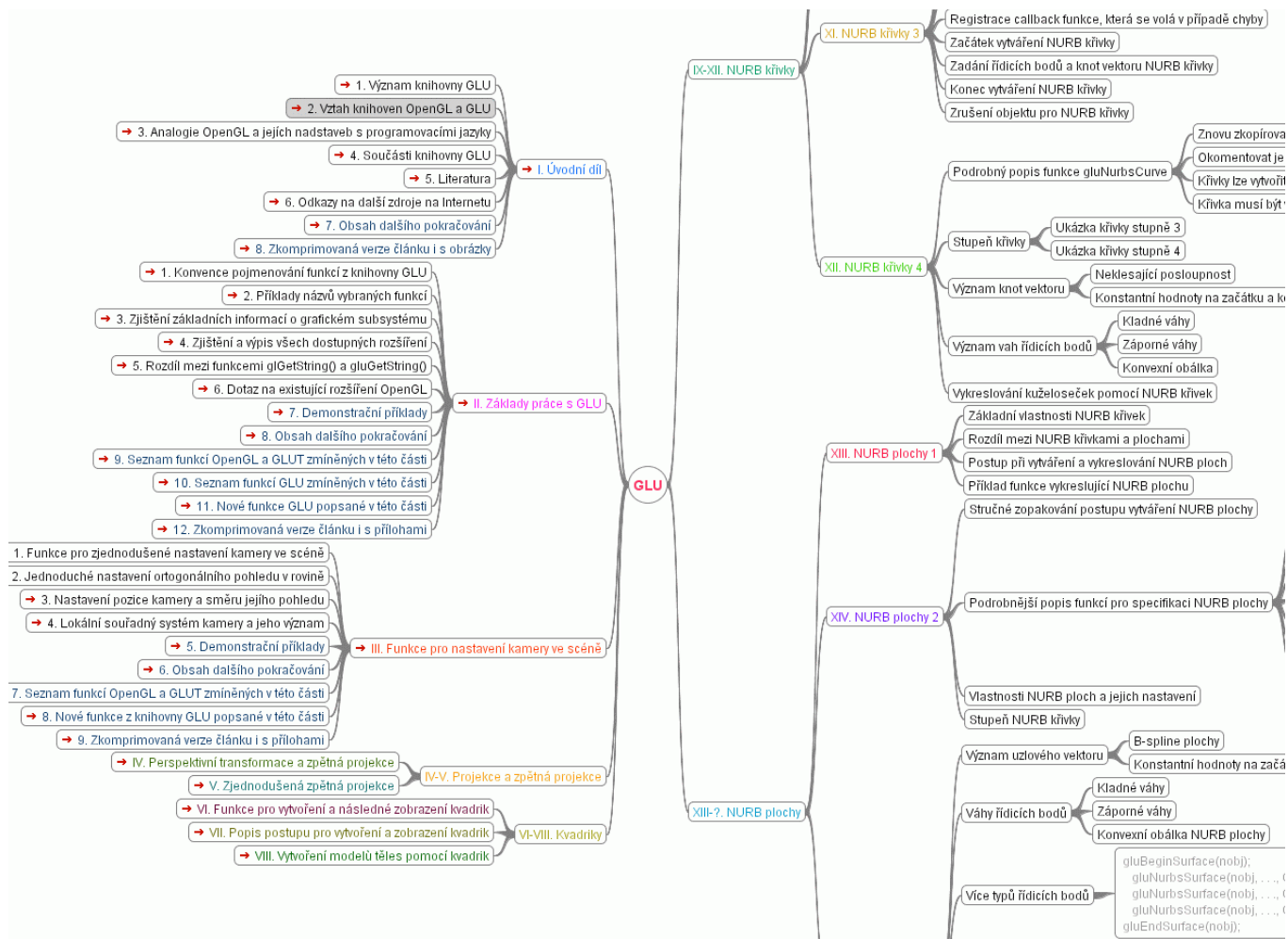
V současné době jsme svědky změny přístupu větších i menších technologických firem k vývoji nových výrobních postupů a technologií. Dříve zcela běžná (a mnohdy velmi rozsáhlá) vývojová oddělení, z nichž se každé zabývalo specifickým oborem, pomalu ztrácí svůj význam. Místo nich nastupují expertní týmy, které jsou dočasně vytvořeny pro vyřešení nějaké problematiky. Po ukončení vývoje či výzkumu jsou většinou tyto týmy rozpuštěny a jejich členové se začnou věnovat (například v dalším týmu) jiné problematice. Týmy jsou mnohdy složeny tak, aby se v nich nacházeli odborníci mnoha profesí. Má to svoje nesporné výhody, například v tom, že se mohou opustit některé „vyjeté koleje“, ke kterým typicky tíhnou klasicky pojatá vývojová oddělení s pevně danou hierarchií a postupy. Na druhou stranu může nastat problém s předáváním informací mezi lidmi mnoha (značně) odlišných profesí a názorů – z tohoto důvodu byly a do dnešní doby jsou hledány postupy a techniky pro vylepšení a urychlení komunikace.

Také práce expertních týmů probíhá poněkud jinak, než bývalo v minulosti běžné. Velký důraz je zde kladen na interakci mezi členy – většinou se členové týmu schází na společných jednáních, na kterých se vede buď neřízená či řízená (moderovaná) diskuse, mnohdy označovaná dnes velmi populárním termínem *brainstorming* (tento termín s oblibou používají manažeři, i když mnohdy neznají jeho přesný význam). Při *brainstormingu* jednotliví členové týmu přichází se svými nápady a idejemi, které jsou buď přímo rozvíjeny nebo odloženy na pozdější projednání. Při tomto principu komunikace, kde se nepostupuje podle předem známé osnovy, je velmi důležité, aby byly vyslovené myšlenky vhodnou formou zaznamenány, a to co nejrychleji, bez zbytečného zdržování řešitelského kolektivu.

Jednou z forem nelineárního zápisu (či znázornění) informací jsou i *myšlenkové mapy*, jejichž princip navrhl Tony Buzan. Principem myšlenkových map je volný zápis klíčových slov, celých textů, obrazů a dalších informací. Důležité je, že tyto základní „informační bloky“ jsou navzájem propojeny pomocí vizuálních spojů a tvoří tak graf – většinou se jedná o strom, v některých případech jsou však dovoleny i další vazby mezi uzly grafu, které čistou stromovou strukturu narušují. Myšlenkové mapy značným způsobem rozšiřují možnosti textových editorů, ve kterých je umožněn pouze lineární zápis. Stejně jako vlastní zápis nových informací je důležitou vlastností myšlenkových map i způsob jejich prohlížení. Jednotlivé uzly je totiž možné skrývat a znovu rozbalovat. Pokud je nějaký uzel skryt, jsou automaticky skryty i všechny jeho poduzly. I s velmi složitou myšlenkovou mapou je tak možné manipulovat i na relativně malé obrazovce.

Použití myšlenkových map však není omezeno pouze na *brainstorming*, díky skrývání uzlů se tyto mapy hodí i pro záznam a utřídění myšlenek jednoho samostatně pracujícího či studujícího člověka. Vhodné je například použití myšlenkových map při studiu; klasické poznámky se totiž u složitější látky stávají neefektivní. Některé firmy také pomocí myšlenkových map zaznamenávají postupy pro své technické pracovníky – uvádí se, že takto použité myšlenkové mapy mají po vytištění mnohdy plochu několika metrů čtverečních. Někteří uživatelé (včetně mne) používají aplikace určené pro myšlenkové mapy také pro tvorbu hierarchických databází.

Na tomto místě je vhodné si uvědomit poměrně značný rozdíl mezi vytvářením myšlenkových map (které jsou do značné míry hierarchicky organizované) a vesměs chaotickým psaním poznámek ve stylu aplikace *One Note*, i když i tento typ aplikací má zcela jistě své uplatnění. Předností myšlenkových map je to, že již při jejich vytváření je uživatel nucen do přesnějšího členění informací – tato počáteční námaha se však několikanásobně vrátí při vyhledávání či dalším zpracování informací.



Aplikace FreeMind se představuje

V tomto seriálu budu paralelně popisovat dvě verze aplikace *FreeMind*. První popisovanou verzí je *FreeMind 0.7.1*, druhou verzí aktuální *FreeMind 0.8.0* (tato verze čirou náhodou vyšla 6. 9., tj. v den mých narozenin :-). Každá z popisovaných verzí má své použití. Verze 0.7.1 zabírá po instalaci na disku cca 600 kB, takže ji například lze stále nosit s sebou na disketě. Také paměťové nároky verze 0.7.1 jsou nižší a rychlost spouštění naopak vyšší než u verze aktuální. Verze 0.8.0 naproti tomu přináší některé další možnosti při vytváření a editaci myšlenkových map, zejména operaci Undo, která v předchozí verzi citelně chyběla. Také byly do značné míry rozšířeny možnosti importu a exportu dat, dodána vícejazyčná verze (o některých problémech s českým překladem se zmíním později) a opraveny některé nedostatky. Vzhledem k tomu, že se při importech a exportech dat používají další přibalené knihovny, je verze 0.8.0 již objemnější – podle požadované konfigurace se objem nainstalovaných dat pohybuje mezi 2,7 MB po cca 8 MB, takže přenos na disketách zde ztrácí svoji efektivitu.

Instalace aplikace FreeMind

Jednou z předností programu *FreeMind* je, zejména ve starší verzi 0.7.1, malá velikost instalačního balíku a současně velmi jednoduchá instalace – ovšem pouze za předpokladu, že na daném počítači

je již korektně nainstalované běhové prostředí jazyka Java pro programy přeložené do javovského bytekódu (jedná se o takzvané JRE – Java Runtime Environment). Na mnoha instalačních CD s Linuxem, nebo na stránkách nabízejících balíky aplikací, je k dispozici hned několik běhových prostředí různé kvality, většinou však díky (stále) nevhodně formulovaným licenčním podmínkám chybí originální JRE od firmy Sun. Toto prostředí, pod kterým byl *FreeMind* testován, je však k dispozici k volnému stažení pro všechny významné platformy přímo na internetových stránkách výrobce <http://java.sun.com/j2se> [1]. V následujících podkapitolách si popíšeme, jakým způsobem je možné *FreeMind* instalovat a poté spouštět na různých operačních systémech a platformách. Kromě JRE od firmy Sun Microsystems lze samozřejmě použít i jiné JRE, ovšem s tou podmínkou, že jsou pro něj dostupné knihovny se stejnou funkcionalitou, jako pro Sunovské JRE 1.4.

Java Runtime Environment 1.4

Jak již bylo řečeno v předchozím odstavci, je nutnou podmínkou pro úspěšné provozování *FreeMindu* instalace běhového prostředí jazyka Java – tzv. *JRE* čili Java Runtime Environment. V tomto prostředí jsou interpretovány programy přeložené do javovského bytekódu. *FreeMind* vyžaduje, aby byla nainstalována *JRE* verze alespoň 1.4, což může na některých platformách způsobovat problémy. Například na starších počítačích s nainstalovanou první verzí Microsoft Windows 95 není možné *JRE 1.4* úspěšně provozovat, proto zde nefunguje ani *FreeMind* (částečně pomáhá instalace staršího JRE a nahrazení balíků s knihovnami získanými z vyšší verze). V současné době je k dispozici i JRE verze 1.5 – i tuto verzi je samozřejmě možné použít, *FreeMind* s ní spolupracuje bez problému. Na starších verzích JRE *FreeMind* buď nepracuje vůbec nebo s chybami. Je to trochu škoda, protože například JRE verze 1.1.8 je/byla poměrně oblíbená (zejména vzhledem k velikosti a stabilitě).

Instalace na všech unixových systémech

Pro obecné unixové systémy, které nemusí obsahovat žádný systém balíčků, jsou určeny předkompilované binární soubory, které jsou na domovské stránce programu k dispozici zabalené pomocí utility *zip* do jednoho archivního souboru. Pro verzi 0.7.1 existuje pouze jeden archivní soubor, verze 0.8.0 je dostupná ve dvou archivech, které se liší tím, které knihovny (a tím pádem i funkce) obsahují. Základní verze má velikost cca 3 MB, plná verze potom 8 MB. Verze se liší podle toho, zda podporují vektorový souborový formát SVG. Po rozbalení jednoho z archivů vznikne adresářová struktura, v níž se (mimo jiné) nachází i soubor `freemind.jar`, který je možné spustit pomocí následujícího příkazu (samozřejmě za předpokladu, že je nainstalované JRE):

```
java -jar freemind.jar
```

Poznámka: soubory typu `.jar` jsou ve skutečnosti také archivy ve formátu ZIP (nebo v nezkomprimovaném formátu). Po rozbalení archivu `freemind.jar` zjistíme, že mimo přeložených javovských tříd obsahuje i několik souborů s koncovkou `.properties`. V těchto souborech jsou v několika jazycích zapsány řetězce používané při zobrazování všech textů v grafickém uživatelském rozhraní. Editací těchto souborů a následným zpětným vytvořením archivu `freemind.jar` (například pomocí utility `jar` či `zip`) je možné například dokončit překlad české verze (v současnosti je v novější verzi proveden překlad cca dvou třetin všech textů).

Instalační balíčky pro vybrané distribuce Linuxu

V současné době jsou oficiálně k dispozici instalační balíčky pro dva linuxové balíčkovací systémy – Red Hatu a Debianu. Přímo na oficiálních stránkách *FreeMindu* (freemind.sourceforge.net [2]) jsou odkazy na stránky obsahující tyto instalační balíčky. Po stažení souboru, který má podle verze a typu balíčkovacího systému jméno `freemind_0.8.x-1_all.deb` nebo `freemind-0.8.x-1.noarch.rpm`,

je možné *FreeMind* nainstalovat obvyklým způsobem. Je však zapotřebí dát pozor na to, že verze 0.8.0 obsahuje v balíku pouze základní program, přičemž knihovny pro rozšíření možnosti aplikace (například pro export do vektorového formátu SVG) jsou uloženy ve zvláštním balíku.

Instalace na systémech Microsoft Windows

Instalace na operačních systémech Microsoft Windows probíhá velmi jednoduše. Stačí spustit stažený instalační program, který se pouze zeptá na umístění spustitelných i datových souborů aplikace, asociaci se soubory s koncovkou `.mm` a umístění ikony s programem na pracovní plochu. Instalační program je na výše uvedené oficiální stránce dostupný jak pro verzi 0.7.1 (cca 750 kB), tak i pro verzi 0.8.0 (3 MB bez podpory SVG a 8 MB s podporou SVG). Po instalaci je možné ušetřit místo na disku tak, že se smaže obsah podadresáře `backup` – tento podadresář totiž obsahuje přesnou kopii právě nainstalovaného celého balíku.

Na třicetidvoubitových systémech Microsoft Windows je možné pro spouštění aplikace *FreeMind* použít spustitelný soubor nazvaný jednoduše `FreeMind.exe`. Tento soubor interně nedělá nic jiného, než je spuštění javovské aplikace umístěné v balíku `FreeMind.jar`. Jedná se o aplikaci s grafickým uživatelským rozhraním, pro jejíž běh se nemusí vytvářet nové konzolové okno. Alternativně je možné *FreeMind* spustit jedním z následujících příkazů, u nichž se přímo zavolá interpreter Javy `java` nebo GUI verze interpreteru `javaw` se zadaným balíkem obsahujícím všechny `.class` soubory:

```
java -jar freemind.jar
javaw -jar freemind.jar
```

Interpreter si v balíku automaticky vyhledá metodu `public static void main`, kterou spustí. Pokud při spouštění či běhu *FreeMindu* dochází k potížím, je nejvhodnější aplikaci spouštět pomocí interpreteru `java` (nikoli `javaw` nebo pomocí souboru `FreeMind.exe`) a přečíst si případné chybové hlášky přímo z konzole, případně provést přesměrování do logovacího souboru.

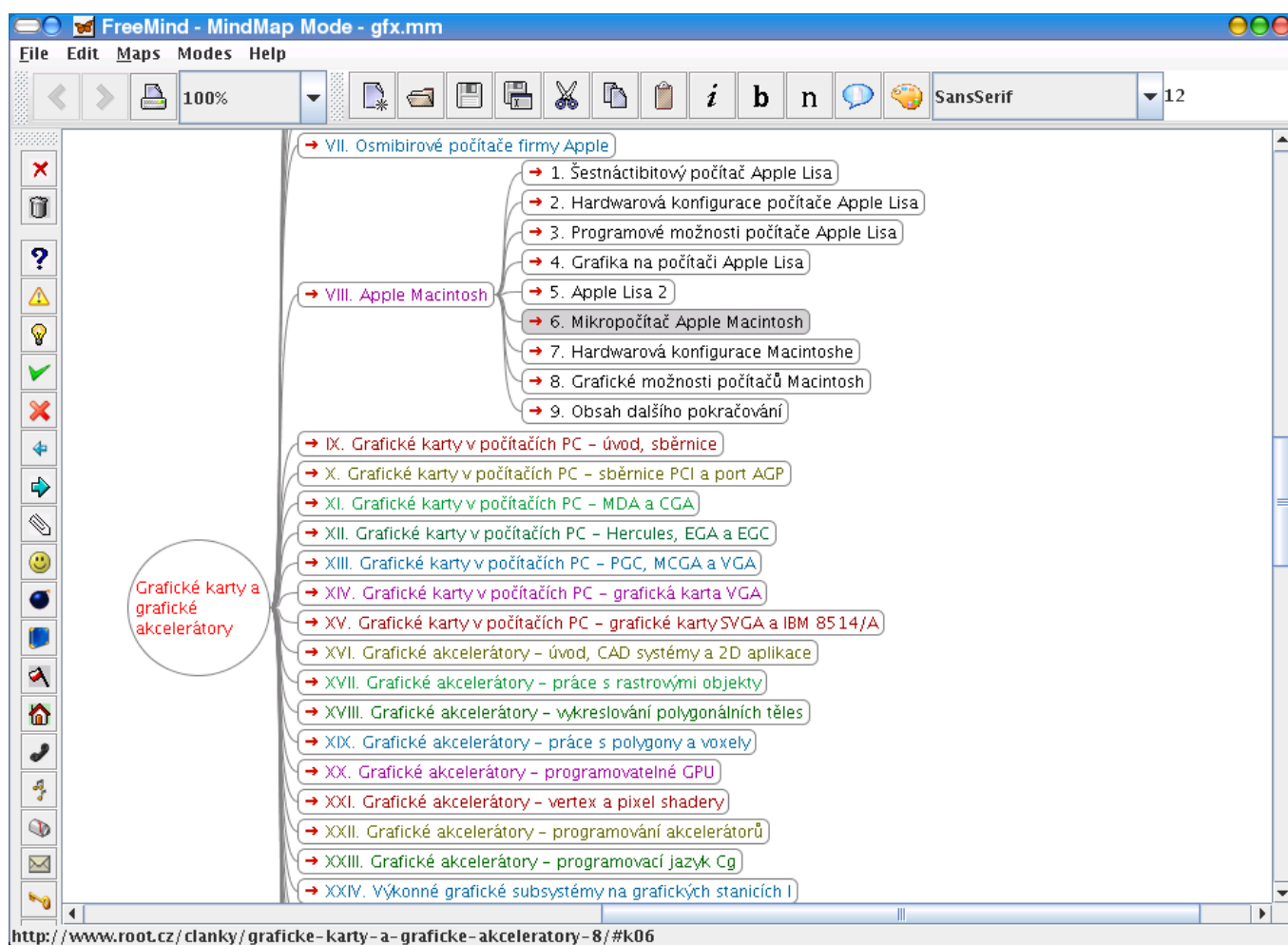
Popis grafického uživatelského rozhraní

Po spuštění aplikace *FreeMind* se zobrazí hlavní okno programu, které je rozděleno do několika oblastí. V horní části se nachází, podobně jako v dalších aplikacích s grafickým uživatelským rozhraním, položky z hlavního menu. Pod hlavním menu je zobrazen horní toolbar s příkazy určenými pro úpravu formátování uzlů a textu zapsaného v těchto uzlech. Na levé straně se nachází toolbar s ikonami, které je možné vkládat do uzlu spolu s textem. Největší část hlavního okna však zabírá pracovní plocha k níž jsou přidruženy horizontální a vertikální scrollbar. Ukázka aplikace *FreeMind* běžící na Linuxu se systémem *X Window* a *Fluxboxem* je zobrazena na prvním obrázku.

Pokud je kurzor myši umístěn na pracovní ploše a není aktivováno hlavní menu ani žádné dialogové okno, je po stisku pravého tlačítka myši zobrazeno kontextové menu. V dalších odstavcích si jednotlivé ovládací prvky této aplikace popíšeme podrobněji.

Hlavní menu

Hlavní menu programu *FreeMind* je zobrazeno, jak již bývá dobrým zvykem, na horním okraji okna. Z celé struktury menu je zobrazeno pouze pět (verze 0.7.1) resp. deset (verze 0.8.0) nejvyšších položek, po jejichž výběru se rozbálí celá struktura menu pod vybranou položkou:



Obrázek 1: Ukázka grafického uživatelského rozhraní programu

- Pod položkou *File* se skrývají (jak se dá očekávat) příkazy určené pro práci se soubory (včetně importu a exportu do několika souborových formátů), příkazy pro tisk a také pro uzavření myšlenkové mapy či celé aplikace.
- Položka *Edit* skrývá příkazy pro editaci uzlů a hran. Příkazy z tohoto menu jsou dostupné i z kontextových menu, které budou popsány v dalších odstavcích. Toto menu bylo v nové verzi rozšířeno o další užitečné příkazy, například pro kopírování formátu jednotlivých uzlů.
- Pod položkou *View* se nachází příkazy pro zobrazení či skrytí obou toolbarů a také příkazy pro změnu měřítka pohledu na myšlenkovou mapu. Toto menu je dostupné pouze v novější verzi (0.8.0).
- Další novou položkou je položka *Insert*, pod níž jsou ukryty příkazy pro vytváření nových uzlů a poduzlů, jakož i příkazy pro vkládání dalších objektů do myšlenkové mapy (poznámek, hyperlinků apod.). V předchozí verzi byly tyto příkazy ukryty pod jinými položkami, zejména v menu *Edit*.
- Pod (taktéž novou) položkou *Format* najdeme příkazy pro formátování uzlů. Je zde nutné poznamenat, že příkazy pro kopírování formátu se nenacházejí zde, ale v menu *Edit*.
- Příkazy ukryté pod položkou *Navigate* umožňují procházení myšlenkovou mapou včetně rozbalování a sbalování podstromů mapy.
- Položka *Tools*, jež se poprvé objevila ve verzi 0.8.0, obsahuje další moduly, které rozšiřují možnosti celé aplikace; například se jedná o kalendář, zobrazení modifikací myšlenkové mapy nebo jednoduchou správu verzí. V budoucnu je možné očekávat další rozšiřování těchto modulů.
- Položka *Maps* obsahuje dynamicky se měnící seznam otevřených myšlenkových map, mezi kterými se dá velmi jednoduše přepínat. I tento seznam je dostupný pomocí kontextových menu.

- Pomocí příkazů skrytých pod položkou *Modes* je možné aplikaci přepínat mezi třemi módy činnosti – mód editace myšlenkových map (*MindMap*), mód procházení adresářovou strukturou na disku (*File*) a mód procházení myšlenkovými mapami bez možnosti jejich úprav (*Browse*). Další módy činnosti je možné programově doplnit pomocí pluginů.
- Poslední položka hlavního menu – *Help* – obsahuje odkazy na dokumentaci a také FAQ. Zajímavé je, že samotná dokumentace je taktéž vytvořena jako myšlenková mapa, takže jsou na ní prezentovány prakticky všechny možnosti programu (kromě práce s obrázky). Po výběru položky *Help – About* se dále dozvíme, že *FreeMind* je vytvořen pod licencí GPL, což je zcela jistě potěšitelné.

Kontextová menu

Kontextových menu existuje ve *FreeMindu* několik. Liší se samozřejmě tím, pro jaký objekt je menu vyvoláno. Podobně jako u jiných aplikací, i ve *FreeMindu* se kontextové menu vyvolává stiskem pravého tlačítka myši nad vybraným objektem či přímo nad pracovní plochou aplikace.

Pokud je pravé tlačítko myši stlačeno nad volnou pracovní plochou, objeví se kontextové menu, ve kterém je možné zvolit myšlenkovou mapu, která se má v dalším kroku prohlížet či editovat. Tato část kontextového menu tedy supluje položku *Maps* z hlavního menu. Kromě toho je možné z kontextového menu zapínat a vypínat horní a pravý toolbar.

Druhé významné kontextové menu se zobrazí v případě, že je pravé tlačítko myši stlačeno nad určitým uzlem v myšlenkové mapě. Ze zobrazeného menu je možné uzly přenášet přes schránku, měnit vlastnosti uzlů a v neposlední řadě také měnit vlastnosti hran, které uzly navzájem spojují. Toto kontextové menu nahrazuje velkou část příkazů, jež jsou dostupné z položek *Edit*, *Insert* a *Format* hlavního menu.

Horní toolbar

Pod hlavním menu se nachází horní toolbar (nástrojová lišta), který obsahuje ikony se základními editačními a formátovacími příkazy. Tento toolbar je možné zviditelnit či zneviditelnit pomocí kontextového menu, které se zobrazí po stlačení pravého tlačítka myši na pracovní ploše aplikace. Horní toolbar je také možné pomocí myši přetáhnout na jiné místo v pracovním okně, nebo ho lze dokonce rozdělit na více relativně samostatných částí.

Na tomto toolbaru se nachází ikony pro přesun na předchozí a následující myšlenkovou mapu, ikona pro tisk dokumentu, výběrový seznam umožňující přiblížení a oddálení pohledu od plochy (plátna) s myšlenkovou mapou a dále ikony, pomocí nichž se provádějí základní editační a formátovací operace. Význam jednotlivých ikon je následující:

- Vytvoření nové myšlenkové mapy. Mezi jednotlivými mapami se lze přepínat buď pomocí kontextového menu nebo pomocí dvou (většinou modrých) šipek zobrazených taktéž na horním toolbaru. Novou myšlenkovou mapu lze rovněž vytvořit přes klávesovou zkratku **Ctrl+N**.
- Otevření souboru s myšlenkovou mapou. Stejnou operaci lze provést z hlavního menu nebo pomocí klávesové zkratky **Ctrl+O**.
- Uložení myšlenkové mapy do externího souboru. Tuto operaci lze také provést pomocí klávesové zkratky **Ctrl+S**. Myšlenková mapa je v externím souboru uložena ve formátu XML.
- Uložení myšlenkové mapy do externího souboru pod jiným jménem. Po zadání tohoto příkazu se aplikace pomocí standardního dialogového boxu pro výběr souboru zeptá na jméno a umístění nově vytvářeného souboru. Stejnou operaci lze ve verzi 0.7.1 provést pomocí klávesové zkratky **Ctrl+A** – pozor, jedná se o nekonzistenci s běžnými GUI, kde se tato klávesová zkratka používá ve významu „vyber vše“. Nová verze již používá konzistentnější klávesovou zkratku **Ctrl+Shift+S**.
- Vyjmutí aktivního uzlu či vybraných uzlů a vložení do schránky (clipboardu). Pokud se provede vložení uzlu či uzlů do jiné aplikace, přenesou se do této aplikace text uvnitř uzlu. Podobnou

operaci lze provést pomocí klávesové zkratky **Ctrl+X** s tím rozdílem, že tuto zkratku lze použít i při editaci na část vybraného textu. Ikona však pracuje pouze s celými uzly.

- Zkopírování aktivního uzlu či vybraných uzlů do schránky (clipboardu). Tato operace je podobná jako operace předchozí s tím rozdílem, že se uzly z myšlenkové mapy nevyjmou. Podobnou operaci zajistí i klávesová zkratka **Ctrl+C**.
- Vložení uzlu ze schránky (clipboardu) do myšlenkové mapy. Pokud je ve schránce umístěn text (například z textového editoru či webového prohlížeče), je vytvořen nový uzel a text je vložen do tohoto uzlu. Víceřádkový text ve schránce je možné použít pro vytvoření více uzlů. Obdobnou operaci lze provést pomocí klávesové zkratky **Ctrl+V**.
- Pomocí této ikony nebo klávesové zkratky **Ctrl+I** se změní text v uzlu či ve vybraných uzlech na italiku (kurzivu). Další stlačení této ikony způsobí navrácení operace.
- Tato ikona slouží k zapnutí nebo vypnutí tučného písma. Stejnou operaci lze provést pomocí klávesové zkratky **Ctrl+B**.
- Stlačením této ikony se text v aktivním uzlu nebo ve vybraných uzlech vrátí do své původní podoby, tj. zruší se všechny příkazy pro změnu písma.
- Po výběru této ikony se provede vyznačení aktivního uzlu či vybraných uzlů pomocí „obláčků“. Tímto způsobem lze zvýraznit určité části myšlenkové mapy. Oblaka lze zapínat a vypínat i pomocí poněkud krkolomné klávesové zkratky **Ctrl+Shift+B**.
- Tímto příkazem lze zvolit barvu oblak, která přísluší právě aktivnímu uzlu či vybraným uzlům. Pro tuto operaci prozatím neexistuje klávesová zkratka, i když se nabízí například volba **Ctrl+Shift+C** (tím by se zachovala konzistence s předchozím příkazem i s anglickým slovem pro „barvu“).
- Tento ovládací prvek slouží k výběru písma, které je použito v aktivním (právě vybraném) uzlu. Pro tuto činnost není vyhrazena žádná klávesová zkratka. Po rozbalení tohoto ovládacího prvku se samozřejmě zobrazí pouze ta písma (fonty), která jsou na daném systému dostupná. Při zkopírování myšlenkové mapy do jiného systému mohou nastat problémy s převodem neexistujících fontů.
- Tímto ovládacím prvkem je možné měnit velikost písma v aktivním uzlu či uzlech. Velikost písma lze také měnit pomocí klávesových zkratk **Ctrl+L** (zvětšit písmo) a **Ctrl+M** (zmenšit písmo).

Verze 0.8.0 přichází s novými (a podle mého méně přehlednými ikonami), které jsou zobrazeny na osmnáctém obrázku.



Obrázek 18: Nové ikony na toolbaru ve verzi 0.8.0

Všechny ikony jsou umístěny v samostatných souborech typu GIF nebo PNG. Tyto soubory jsou uloženy v balíku *FreeMind.jar*, takže po dekomprimaci tohoto archivu je možná jejich změna a opětovné přidání do archivu.

Levý toolbar

Na levém toolbaru, který je samozřejmě taktéž přemístitelný, se nacházejí ikony, které lze vkládat do aktivního (právě vybraného) uzlu. Použití tohoto toolbaru je jednoduché – po vybrání uzlu se levým tlačítkem myši klepne na některou z ikon, která se posléze do uzlu vloží. V uzlu může být ikon hned několik, ale po jejich vložení je již není možné přemísťovat, pouze je lze odebrat a znovu přidat v jiném pořadí. První dvě ikony na levém toolbaru však mají speciální význam.

Po stlačení první ikony se z aktivního uzlu vymaže naposledy vložená ikona – jedná se tak o jednoduchý editační příkaz.

Tento příkaz slouží k odebrání všech ikon z aktivního uzlu. Poznámka: v ikonách je dostupný i pěkný tučňáček, okna s obláčky však budete hledat marně :-).

Scrollbary

Scrollbary jsou většinou umístěny po pravé a dolní straně pracovní plochy aplikace. Jejich použití je triviální – pomocí nich je umožněn posun po ploše, na které je vytvořena myšlenková mapa. O scrollbarích bych se zde díky jejich známosti ani nezmiňoval, nebýt jednoho nedostatku, který se projevuje zejména ve verzi 0.7.1: při otevírání a zavírání jednotlivých poduzlů (tím dochází ke změně velikosti zobrazené myšlenkové mapy) se neprovede proporcionální změna velikosti posuvníku na scrollbaru, takže se v některých případech může posuvník stát téměř nepoužitelný (zabírá díky své délce značnou část své dráhy). V případě, že tento problém nastane, je vhodné změnit měřítko zobrazení a po posunu myšlenkové mapy vrátit hodnotu měřítka na původní rozsah.

Činnost scrollbarů je možno nahradit i kolečkem myši. Při prostém rolování kolečkem se provádí vertikální posun myšlenkové mapy, při rolování se stlačenou klávesou **Shift** či některém z tlačítek myši se provádí rolování horizontální.

Posun a změna měřítka mapy

Posun mapy byl již popsán v předchozí podkapitole, zde si však popíšeme i další možnosti změny pohledu na myšlenkovou mapu. Mapu je možné v okně aplikace posouvat třemi způsoby:

1. stlačením levého tlačítka myši na pracovní ploše (mimo uzly) a posunem myši
2. pomocí rolovacího kolečka na myši s případným použitím klávesy **Shift** či některého tlačítka myši
3. pomocí scrollbarů, které jsou umístěny po pravé a dolní straně pracovní plochy aplikace

Změnu měřítka zobrazené myšlenkové mapy je taktéž možné provést více způsoby. Nejjednodušší je použití rozbalovacího seznamu umístěného na horním toolbaru, po jehož rozbalení je možné zvolit z několika zvětšení myšlenkové mapy. Druhou možností je použití klávesových zkratk, kterými jsou **Alt+šipka nahoru** pro zmenšení mapy a **Alt+šipka dolů** pro její zvětšení. Jedinou nevýhodou změny měřítka je fakt, že při rozsáhlých myšlenkových mapách je nutné chvíli počkat (a to i několik minut).

Operace s uzly myšlenkové mapy

Při tvorbě a následné úpravě myšlenkové mapy jsou jednou z nejdůležitějších činností operace prováděné s jednotlivými uzly mapy. *FreeMind* používá techniku automatického umístování uzlů v pracovní ploše, což znamená, že uživatel pouze specifikuje vztah nově vytvářeného uzlu k uzlům ostatním. Nové uzly lze vytvářet na dvou místech: buď se jedná o uzly na stejné úrovni, jako je uzel aktivní, nebo se jedná o poduzel, tj. o uzel na nižší úrovni. Po vytvoření uzlů je možné začít s úpravami myšlenkové mapy: uzly lze přemísťovat, měnit jejich hierarchii (vzájemné vazby) a styl, přepisovat části textu v uzlech, pracovat s hypertextovými odkazy a obrázky apod. Všechny tyto možnosti budou zmíněny v následujících podkapitolách.

Vytváření a rušení uzlů

Pro vytvoření nového poduzlu stačí použít klávesu **Insert**. Po stlačení této klávesy se vytvoří nový poduzel, do kterého je ihned možno zapisovat nový text nebo vkládat předpřipravené ikony. To, že je možné do uzlu zapisovat text, je symbolizováno textovým kurzorem, který se v uzlu objeví. Uzel se v této chvíli chová jako systémové textové pole, takže je například možné pracovat ze schránkou (clipboardem) apod. Nový uzel lze také vytvořit pomocí kontextového menu, ze kterého se vybere příkaz *New Child Node*.

Uzel na stejné úrovni v mapě se vytvoří po stlačení klávesy **Enter**. Také po vytvoření tohoto uzlu je možné do něj ihned zapisovat informace. Po stlačení klávesové kombinace **Shift+Enter** se taktéž vytvoří nový uzel na stejné úrovni v hierarchii, ale tento uzel bude umístěn nad aktivním uzlem, nikoli pod ním.

I tyto dva příkazy jsou dostupné z hlavního i kontextového menu pod položkami *Insert – New Sibling Node* a *Insert – New Previous Sibling Node*.

Pokud se po zápisu nového textu do uzlu stiskne pouze klávesa **Enter**, bude nový uzel obsahovat jen jeden řádek textu. Je však možné vytvářet i uzly s delšími (odstavcovými) texty – takové uzly se ve *FreeMindu* nazývají „dlouhé uzly“ (*long nodes*) a jsou vytvořeny tak, že se po zadání prvního řádku textu do uzlu stiskne místo pouhého **Enteru** kombinace kláves **Alt+Enter**. Po tomto příkazu se otevře malé okno s textovým polem, do kterého je možné zapsat několikařádkový text. O odstavcový text se jedná z toho důvodu, že se konec řádku musí specifikovat pouze tam, kde je zapotřebí ukončit odstavec. Vestavěný textový editor (tvořený javovským widgetem) totiž automaticky ukončuje řádky podle šířky okna a přitom na další (obrazové) řádky přesouvá celá slova, tj. jedná se o známý režim *word wrap*. Při zápisu textu do „dlouhého uzlu“ se pro ukončení jednotlivých řádků používá buď pouze **Enter** nebo klávesová kombinace **Ctrl+Enter**. Konkrétní chování aplikace závisí na nastavení zatrhávacího boxu *Enter confirms*, který podle uživatelského zvyku přehazuje význam klávesových zkratk **Enter** a **Ctrl+Enter**. Text v „dlouhém uzlu“ může obsahovat i tabulační zarážky, tímto způsobem je možné do myšlenkové mapy vkládat i jednoduché tabulky bez nutnosti vázat uzel myšlenkové mapy na externí soubor. Texty vkládané do uzlů je možné přenášet i přes schránku (například z textového editoru nebo internetového prohlížeče. V tomto případě, se každý řádek textu vloží do nového uzlu.

O rušení uzlů se postará klávesa **Delete**. Stiskem této klávesy je možné vymazat i celý podstrom. Naproti tomu klávesa **BackSpace** slouží k vymazání textu z uzlu a přepnutí aplikace do režimu, ve kterém je možné do právě aktivního uzlu zapsat nový text. Vzhledem ke zmatkům v řízení kláves **Delete** a **BackSpace** v javovských aplikacích běžících pod (resp. nad :-)) X Window mohou být funkce těchto dvou kláves prohozeny.

Editace uzlů

Základním editačním příkazem je samozřejmě změna textu, který je v uzlu uložen. Po výběru některého uzlu lze stlačit klávesu **F2**, po jejímž stisku se celá aplikace přepne do režimu změny textu v uzlu, což je symbolizováno textovým kurzorem, který se v uzlu objeví. Nyní se také změní funkce některých kláves, zejména kláves pro posun kurzoru (**[šipka doleva]**, **[šipka doprava]**, **Home** a **End**) a pro práci se schránkou. Tyto klávesy nyní fungují tak, jak je to běžné v textových editorech. Při editaci „dlouhého uzlu“ (viz předchozí podkapitola) se otevře samostatné okno, ve kterém je možné úpravy provádět. Nastavení klávesových zkratk **Enter** a **Ctrl+Enter** zůstává nastavené stejně, jako při vytváření nového uzlu. V okně, které se otevře pro editaci dlouhého uzlu lze použít i další užitečnou operaci nazvanou *Split*, pomocí které lze uzel rozdělit na dva uzly. Pouze stačí najet textovým kurzorem na místo rozdělení a stlačit tlačítko *Split* – část textu před kurzorem se stane jedním uzlem, zbylý text se stane uzlem druhým. Postupným použitím operace *Split* a přesunem uzlů je do myšlenkové mapy možné dostat původně lineární text, což jej mnohdy zpřehlední a urychlí jeho pochopení.

Dalším často používaným editačním příkazem je přesun uzlu na jiné místo v myšlenkové mapě. Tento příkaz lze provádět jak pomocí klávesnice, tak i pomocí myši. Pro přesun uzlu nahoru či dolů na stejné úrovni hierarchie je možné použít klávesové zkratky **Ctrl+šipka nahoru** a **Ctrl+šipka dolů**. Stejnou operaci je možné provést tažením uzlu pomocí myši. Nové umístění uzlu je indikováno gradientním přechodem – viz ilustrační obrázky zobrazené níže.

Pokud je nutné přesunout uzel na libovolné (i dosti vzdálené) místo v myšlenkové mapě, je použití klávesnice omezené na práci se schránkou, tj. na klávesové zkratky **[Ctrl+X]**, **[Ctrl+C]**, **[Ctrl+V]**, což je poněkud těžkopádné. V tomto případě je mnohem výhodnější použít myš a uzel na požadované místo přetáhnout (*drag*) se stlačeným levým tlačítkem. Pokud je zapotřebí přetáhnout uzel na jinou úroveň hierarchie, musí se kurzor myši umístit na místo, kde se objeví horizontální gradientní přechod. Uvedená operace je znázorněna na dalších dvou ilustračních obrázcích.

Verze FreeMindu 0.8.0 zahrnuje i (před uživateli poněkud ukrytou) možnost manuálního přesunu uzlu – ostatně vedle operace *Undo* je operace přesunu uzlu jednou z největších vymožeností nové verze oproti starší verzi 0.7.1. Manuální přesun uzlu se provádí následovně: kurzor myši se umístí k levému okraji uzlu tak, aby se u kurzoru zobrazila malá elipsa. Tuto elipsu je možné pomocí levého tlačítka „chytit“ a uzel následně přesunout. Spolu s přesouvaným uzlem se dynamicky mění i část myšlenkové mapy v okolí uzlu tak, aby se dodržela stanovená hierarchie, tj. aby přesouvaný uzel nepřeskočil svého předchůdce nebo následníka. Pomocí příkazu *Format – Reset Position* je uzel navrácen do své počáteční pozice. Současná verze FreeMindu 0.8.0 prozatím korektně neřeší protínání uzlů a hran, takže hrana vedoucí k manuálně přesunutému uzlu může protínat další části myšlenkové mapy (vzhledem k tomu, že hrana je tvořena buď jednou úsečkou nebo Bézierovou kubikou, není možné problematiku protínání vyřešit stoprocentně, k tomu by bylo nutné použít složitější tvary, například lomené čáry nebo křivky s více řídicími body). Ukázka manuálního přesunu uzlu je ukázána na pátém a šestém obrázku.

Hypertextové odkazy a obrázky

Do uzlů lze kromě jednořádkového či odstavcového textu vkládat i hypertextové odkazy a obrázky. Vkládání hypertextových odkazů je jednoduché. Stačí vybrat některý uzel a stlačit klávesovou zkratku **[Ctrl]+[K]**. Objeví se dialogové okno, do kterého lze napsat (či přenést přes schránku například z internetového prohlížeče) URL, tj. celou cestu hypertextového odkazu. Pomocí klávesové kombinace **[Ctrl]+[Shift]+[K]** lze vyvolat dialogové okno, ve kterém je možné URL vybrat, takže se odkaz nemusí vypisovat ručně. Uzel, který obsahuje hypertextový odkaz, má na své levé straně zobrazenou šipku. Po kliknutí levým tlačítkem na šipku se provede skok na hypertextový odkaz, kliknutí do jiných částí uzlu způsobí zobrazení či skrytí poduzlů. Skok na hypertextový odkaz je možné provést i z klávesnice klávesovou zkratkou **[Ctrl]+[Enter]**.

Pro vkládání obrázků lze použít klávesovou zkratku **[Alt]+[K]**. Po stlačení této kombinace kláves se objeví dialogové okno, pomocí kterého lze obrázek nalistovat. Při tomto způsobu výběru obrázku se do uzlu zapíše jednoduchý kód v jazyce HTML, který obsahuje odkaz na zobrazovaný obrázek. Kód lze samozřejmě opravit (klávesou **[F2]**) a ručně změnit například rozměry obrázku. Obrázky je vhodné vkládat pouze do uzlů, které neobsahují další poduzly, protože *FreeMind* (zejména ve verzi 0.7) nedokáže korektně zobrazit obrázek v uzlu, jež se nenachází na nejnižší úrovni v hierarchii. Tento nedostatek je již částečně odstraněn v současné verzi 0.8.0. U obrázků musíme mít dále na mysli, že do souboru s myšlenkovou mapou se ukládají pouze ODKAZY na obrázky a ne samotné bitmapy. Při posílání či přesunu souboru s myšlenkovou mapou je nutné tuto skutečnost vzít v úvahu, jinak se odkazy na obrázky „rozsypanou“. Ukázka myšlenkové mapy s vloženými obrázky je ukázána na sedmém screenshotu.

Styl písma a formátování celého uzlu

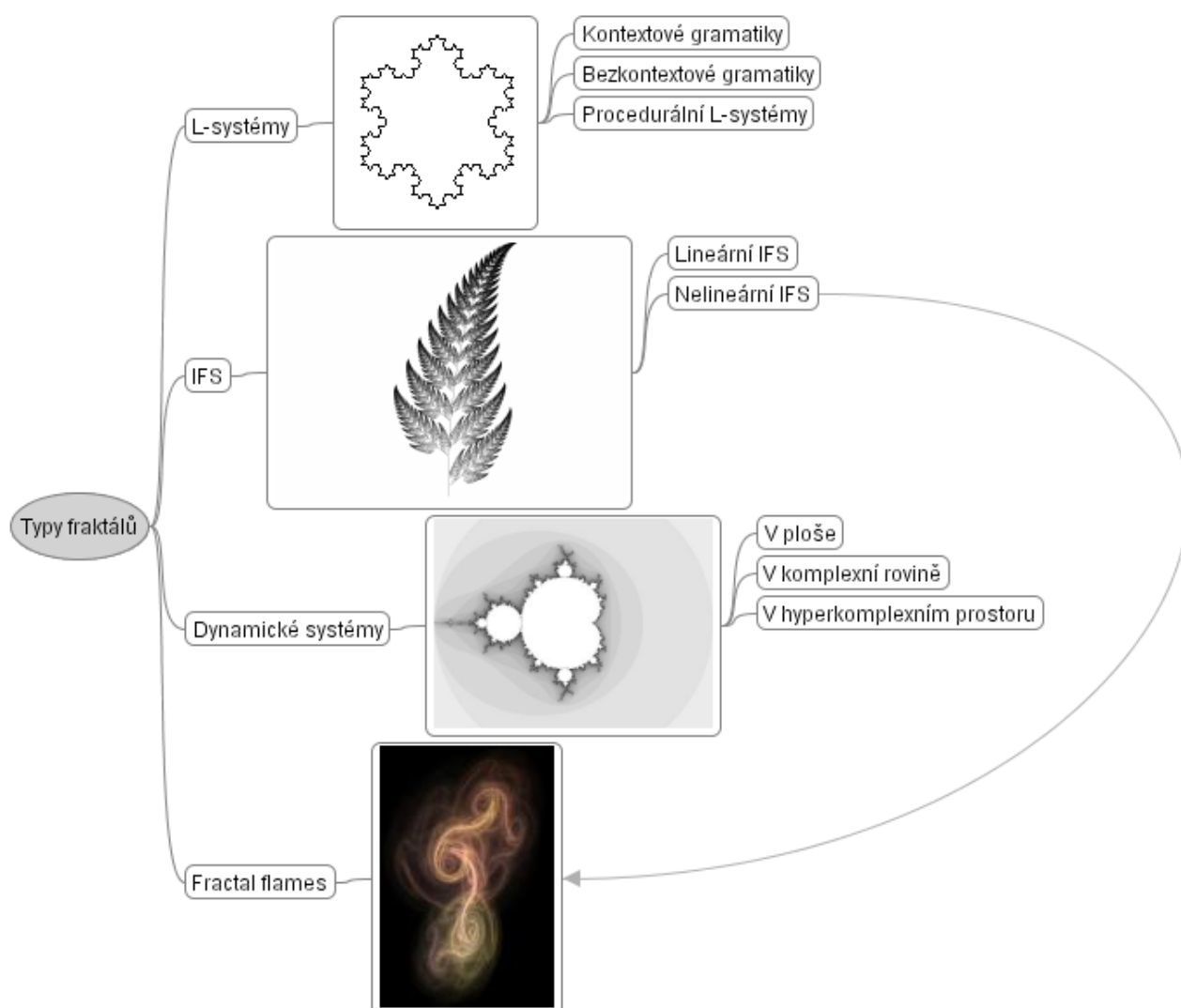
Text v uzlu může být různým způsobem naformátován a mohou mu být přiděleny různé fonty. Nejjednodušší je použití horního toolbaru, který obsahuje prakticky všechny příkazy pro formátování textu v uzlu. Pomocí příslušných ikon nebo klávesových zkratk **[Ctrl]+[B]** a **[Ctrl]+[I]** lze zapínat a vypínat tučný text, popř. kurzívu. Na horním toolbaru se rovněž nachází rozbalovací boxy, ve kterých je možné vybrat velikost textu a použitý font. Výběr dostupných fontů je samozřejmě závislý na fontech, které jsou nainstalované v systému, samotný *FreeMind* žádné vlastní fonty neobsahuje.

Dalším atributem uzlů, který lze změnit, je jejich barva. Pomocí klávesové zkratky **[Alt+C]** (verze 0.7) či **[Alt+B]** (verze 0.8.0) nebo výběrem příslušného příkazu z kontextového menu uzlu (*Format – Node Color*) se zobrazí dialogové okno, ve kterém je možné zadat barvu textu v uzlu. Barvu pozadí je možné měnit pouze ve verzi 0.8.0 pomocí příkazu *Format – Node Background Color*. Speciálním příkazem je také možné vypočítat barvu nějakého uzlu jako průměr barev všech poduzlů. Tento příkaz lze vyvolat buď klávesovou zkratkou **[Alt+B]** nebo výběrem příkazu *Node – Blend Color* z kontextového menu. Uzly mohou mít kolem sebe zobrazen okraj – to se provádí příkazem *Format – Bubble*.

Ikony vkládané do uzlů

Ikony mohou v některých případech ulehčit orientaci v rozsáhlých myšlenkových mapách, stejně tak jako barvy a styly písma jednotlivých uzlů. Vložení ikony do uzlu je velice jednoduché. Stačí vybrat některou ikonu nabízenou na levém toolbaru a tato ikona se vloží do právě aktivního uzlu. Pokud je levý toolbar vypnutý, lze použít i kontextové menu, které se zobrazí kliknutím pravým tlačítkem myši na některém z uzlů. Z tohoto menu stačí zvolit příkaz *Icons* a rozbalí se submenu s názvy a náhledy jednotlivých dostupných ikon. V současné verzi FreeMindu není možné používat vlastní ikony, místo nich se do uzlů dají vkládat odkazy na obrázky (bitmapy i pixmapy) uložené v externích souborech – viz přecházející text.

Pomocí dvou příkazů je možné odstranit buď poslední zadanou ikonu nebo všechny ikony. Příkaz *Icons – Remove Last Icon* odstraní poslední vloženou ikonu, která je umístěna napravo. Příkaz *Icons – Remove All Icons* odstraní všechny ikony v uzlu. Žádné další operace s ikonami však nejsou (prozatím) podporovány, takže například prohození dvou ikon se musí provádět jejich odstraněním a opětovným vložením.



Obrázek 7: Myšlenková mapa s vloženými obrázky

Operace s hranami myšlenkové mapy

Každá myšlenková mapa se kromě uzlů skládá i z hran (na tomto místě se nenechte zmást pojmenováním *mapa*, z matematického pohledu se vlastně jedná o orientovaný graf). Operace, které je možné s hranami provádět, budou uvedeny v následujících podkapitolách.

Styl hran

Hrany mohou být v současné verzi zobrazeny čtyřmi různými způsoby. Nejpoužívanějším stylem je klasická úsečka s konstantní šířkou, která je použita i u dalších aplikací zpracovávajících myšlenkové mapy. Dalším stylem je úsečka s proměnlivou šířkou, která tak částečně nahrazuje šipky. Následuje hrana ve tvaru Bézierovy kubiky (tj. křivky určené pomocí čtyř řídicích bodů nebo dvou bodů a dvou tečen) a konečně hrana ve tvaru Bézierovy kubiky s proměnlivou šířkou, podobně jako u úseček. Styl hran v myšlenkové mapě se vybírá z hlavního či kontextového menu *Format – Edge Styles*. Hrany se změní pouze pro podstrom specifikovaný právě aktivním uzlem (nebo pro více podstromů, pokud je uzlů vybráno několik).

Barva hran

Barvy hran (ale i okrajů uzlů) je samozřejmě také možné měnit. Slouží k tomu příkaz *Format – Edge Color* vyvolaný z hlavního či kontextového menu. Kromě toho je také možné (alespoň v nové verzi) použít klávesovou zkratku `[Alt]+[E]`. Podobně jako u změny stylů hran, i jejich barvy se změní pouze v podstromu příslušejícím k aktivnímu uzlu či aktivním uzlům.

Tloušťka hrany

Tloušťku hran je možné měnit příkazem *Format – Edge Widths*. Tloušťka se zadává v pixelech, přičemž se samozřejmě nejedná o přesně zadanou hodnotu, protože hrany jsou zobrazovány pod různým sklonem, popř. i se zapnutým vyhlazováním. Pokud jsou použity takové styly hran, že u nich dochází ke zužování, platí zadaná tloušťka hran pro jejich tlustší začátek, tenčí začátek je vždy nastaven na jeden pixel.

Vyhlazení hran

Vyhlazení hran, které je vhodné zapnout zejména při práci na LC displejích, se zapíná pomocí volby dostupné na formuláři *Preferences*, který lze vyvolat buď z hlavního menu, nebo pomocí klávesové zkratky `[Ctrl]+,`. Volba pro vyhlazování se jmenuje *Appearance/Antialias*.

Přídavné hrany

Myšlenkové mapy tvoří z matematického hlediska stromovou strukturu. V některých případech je však vhodné do mapy zavést i další vazby, které čistý strom porušují a „povyšují“ ho na obecný graf. Ve *FreeMindu* se další vazby vytváří buď pomocí klávesnice (klávesová zkratka `[Ctrl]+[L]`), nebo myši, a to tak, že se vazba vytvoří tažením kurzoru myši se stlačeným levým tlačítkem myši a současně klávesami `[Ctrl]+[Shift]`. Výsledná myšlenková mapa s dalšími vazbami je zobrazena na dalším obrázku.

Přenos dat mezi FreeMindem a dalšími aplikacemi

Aplikace *FreeMind* obsahuje při komunikaci s dalšími aplikacemi překvapivě mnoho možností. Jedná se jak o přenos dat pomocí schránky (clipboardu), tak i o exporty a importy dat do a z různých souborových formátů. Ve druhé kapitole budou popsány možnosti exportu a v kapitole třetí možnosti importu částí myšlenkových map. Kapitola čtvrtá bude věnována popisu interního formátu *FreeMindu*.

Možnosti exportu myšlenkových map nebo jejich částí

Z *FreeMindu* je možné provést export vytvořené myšlenkové mapy do několika formátů. Nejčastěji používaný bude pravděpodobně export do HTML souboru zobrazitelného takřka v jakémkoli WWW browseru, je však možné generovat i statické dokumenty ve formátech PDF či rastrové i vektorové obrázky. V dalších podkapitolách si jednotlivé typy exportů popíšeme podrobněji. V podkapitole bude popsán způsob exportu do formátů HTML a XHTML, v další podkapitole export do *Portable Document Formatu* (PDF), následující podkapitola je věnována exportu do formátu *Scalable Vector Graphics* (SVG), další výstupu do rastrových obrázků typu PNG a JFIF a konečně podkapitola poslední se věnuje výstupu myšlenkových map do formátu aplikací z aplikačního balíku OpenOffice.org.



Obrázek 1: Myšlenková mapa, na které budou ukazovány možnosti exportu

Export myšlenkové mapy do HTML a XHTML

Export myšlenkové mapy do formátu HTML (HyperText Markup Language) se v nové verzi *FreeMindu* značně vylepšil. Nyní jsou k dispozici tři různé způsoby exportu. Při použití prvního způsobu, který se volí z hlavního menu položkou *File – Export – As HTML*, se vytvoří HTML stránka, ve které jsou jednotlivé uzly podle svých vzájemných souvislostí převedeny do navzájem vnořených seznamů. To však není vše – do výsledného HTML jsou vloženy i funkce napsané v *JavaScriptu*, pomocí nichž je možné jednotlivé uzly myšlenkové mapy kliknutím myši na příslušné značky *[+]* a *[-]* zabalovat a rozbalovat. HTML stránka při otevření v prohlížeči s podporou JavaScriptu a CSS ožije a chová se do značné míry podobně jako při prohlížení přímo ve *FreeMindu* – samozřejmě zde však nejsou dostupné příkazy pro interaktivní změnu takto zobrazené myšlenkové mapy. HTML soubor je vytvo-

řený tím způsobem, že je ho možné zobrazit i v textových prohlížečích či prohlížečích nepodporujících *JavaScript* a/nebo CSS. V tomto případě nebudou značky pro rozbalení a zabalení jednotlivých uzlů funkční. Při exportu do HTML je zapotřebí si dát pozor na to, že původně rozbalené uzly nebude možné v HTML stránce zabalit – seznamy sice budou vytvořeny korektně, ale bez volání příslušné funkce v *JavaScriptu*. Těžko říci, jestli je to chyba FreeMindu nebo jeho vlastnost (můžu si představit situace, kdy tato funkcionalita může být výhodná).

All + All -

~Forth

- ± I. Forth - vznik a historie
- ± II. Forth na osmibitových počítačích a mikroprocesorech
 - ~ 1. Historie vzniku jména Forth
 - ~ 2. Forth na mikropočítačích
 - ~ 3. Forth na osmibitových mikroprocesorech
 - ~ 4. Forth na osmibitové domácí počítače
 - ~ 5. Příčiny neoblíbenosti Forthu
 - ~ 6. Zápis matematických operací v postfixové notaci
 - ~ 7. Syndrom jednoho programovacího jazyka
 - ~ 8. Obsah dalšího pokračování
- ± III. Chuck Moore, dostupná vývojová prostředí Forthu
- ± IV. Zásobník, dvouzásobníkový procesor, postfixová notace
- ± V. Procedury, operátory, podmínky, počítané smyčky
- ± VI. Práce s proměnnými, logické a relační operátory
- ± VII. Matematické operace, zásobník návratových adres, pole
- ± VIII. Operace s řetězci
- ± IX. Vstupně-výstupní operace
- ± X. Blokované čtení a zápis dat, formátování čísel na výstupu
- ± XI. Přesun bloků dat, práce s řetězci, rozšíření překladače
- ± XII. Jazyky založené na Forthu a zásobníkových procesorech
- ± XIII. Zásobníkové procesory - úvod
- ± XIV. Kategorie zásobníkových procesorů
- ± XV. Zásobníkový procesor F21
- ± XVI. Zásobníkové procesory WISC CPU/16 a MISC M17
- ± XVII. Zásobníkové procesory NC4016 a RTX 2000
- ± XVIII. Zásobníkové procesory FRISC 3 a RTX 32P
- ± XIX. Javovské zásobníkové procesory
- ± XX. Závěr



Obrázek 2: Myšlenková mapa převedená do HTML a zobrazená v grafickém WWW prohlížeči

Obrázek 4: Mapa převedená do XHTML a zobrazená v grafickém WWW prohlížeči s podporou JavaScriptu

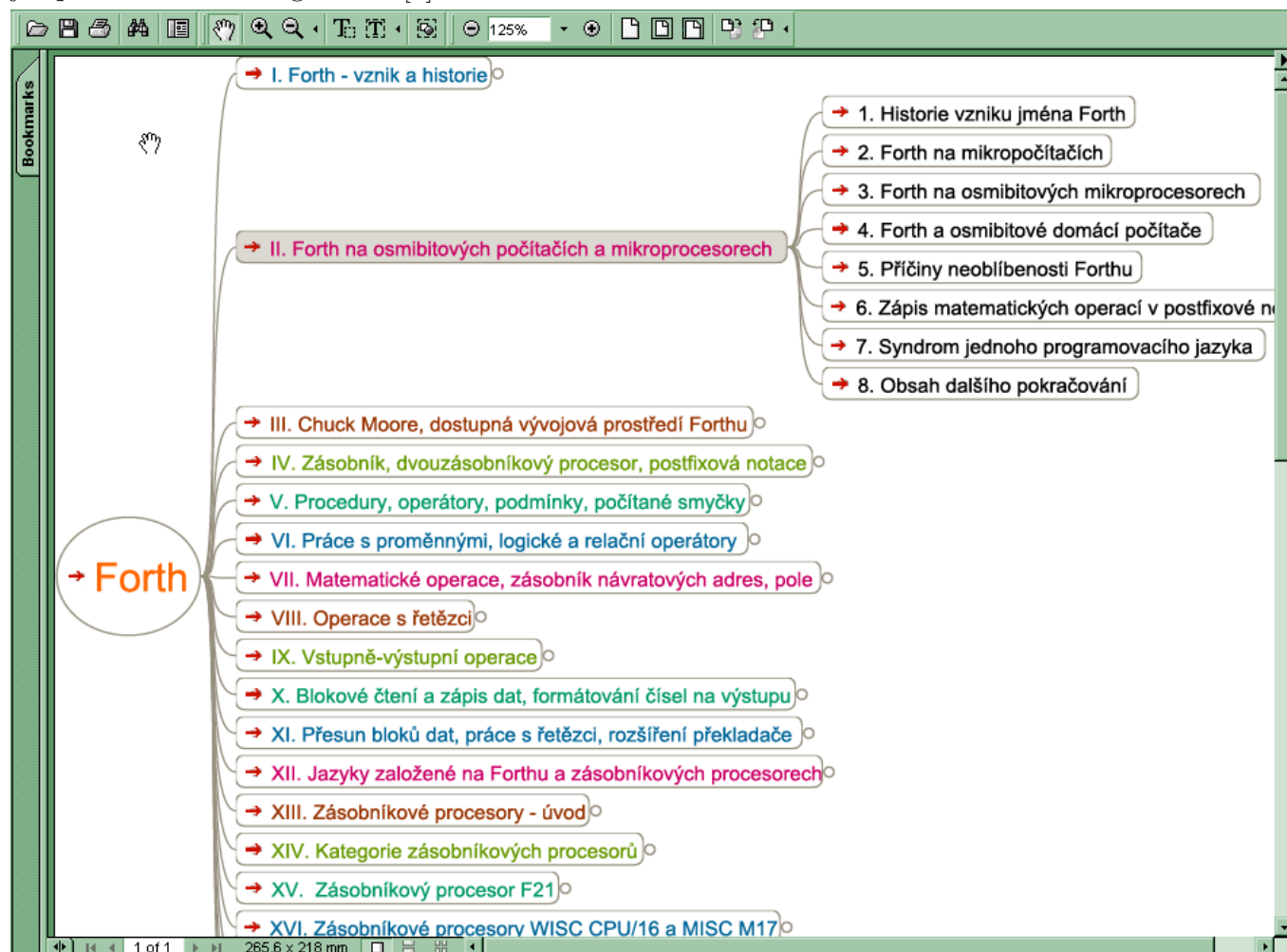
Druhý příkaz pro export (*File – Export – As XHTML JavaScript Version*) je v mnohém podobný předchozímu příkazu. Namísto HTML souboru je však vytvořen korektní soubor typu XHTML. Spolu s tímto souborem je vytvořen i podadresář s názvem stejným jako soubor XHTML a navíc s přidáním řetězcem „FILES“. V tomto podadresáři je umístěn jak soubor se styly, tak i soubor obsahující potřebné funkce v *JavaScriptu* a dále několik souborů s ikonami použitými v uzlech. Při zobrazení vytvořené WWW stránky v prohlížeči je nalevo od uzlů umístěna ikona se značkou pro rozbalení či zabalení uzlu. Díky vhodně navrženému stylu je zobrazení XHTML souborů vizuálně přitažlivější, než tomu bylo u výše zmíněných souborů HTML. Zajímavé je, že při tomto typu exportu se neprojevuje výše popsaná chyba/vlastnost při zobrazování původně rozbalených uzlů.

Třetí příkaz pro export do HTML (*File – Export – As XHTML Clickable Map Version*) vytvoří soubor, který se při svém zobrazení v prohlížeči chová ještě „přítulněji“ než předchozí dva typy. Kromě vnořených seznamů s jednotlivými uzly (ty lze opět rozbalovat a zabalovat) je v horní části zobrazena i klikací mapa, tj. rastrový obrázek reagující na stlačení tlačítka myši se zobrazenou myšlenkovou mapou. Kliknutím na uzel vyobrazený na obrázku se provede skok přímo na vybraný uzel. Při tomto typu exportu je zapotřebí dát pozor na to, že obrázek s myšlenkovou mapou přesně odpovídá aktuálnímu zobrazení myšlenkové mapy ve *FreeMindu*, tj. skryté uzly nebudou v rastrovém obrázku přítomny! To je však spíše výhoda, zejména při práci s rozsáhlými myšlenkovými mapami, kde by byl rastrový obrázek celé rozbalené mapy obrovský.

Export myšlenkové mapy do formátu PDF

V některých případech, například při potřebě tisku myšlenkové mapy na jiném počítači, se může hodit export myšlenkové mapy do formátu PDF (*Portable Document Format*). Myšlenková mapa vyexportovaná do tohoto formátu je však statická, tj. způsob jejího zobrazení již není možné změnit. Na druhou stranu je myšlenková mapa zobrazena přesně v té podobě jako v samotném *FreeMindu*.

Formát PDF je zde využit velmi rozumným způsobem, obrázek myšlenkové mapy je striktně vektorový, takže je výsledný soubor relativně malý a přitom je zajištěn kvalitní tisk na prakticky všech tiskárnách. Zajímavé je, že PDF se ve FreeMindu interně vytváří přes vektorový formát SVG, který je zpracován technologií FOP [3].



Obrázek 6: Myšlenková mapa převedená do formátu PDF zobrazená v Acrobat Readeru

Export myšlenkové mapy do formátu SVG

SVG, neboli *Scalable Vector Graphics*, je souborový formát určený pro uložení vektorové grafiky, jehož už několik let ohlašovaný raketový nástup do světa Internetu se – nutno říci bohužel – prozatím z mnoha důvodů nekoná. FreeMind tento formát (mimo jiné založený na jazyku XML) využívá pro export statických dokumentů, podobně, jako tomu bylo u výše popsaného formátu PDF. Vytvořený soubor typu SVG je možné použít více způsoby: buď ho zobrazit na HTML stránce, zobrazit ho ve specializovaném prohlížeči (viz screenshot na obrázku 7) nebo ho naimportovat do některého z dostupných vektorových grafických editorů (viz screenshot na obrázku 8). První způsob je však problematický, neboť doposud ne všechny WWW prohlížeče umí SVG korektně zpracovat.

Export myšlenkové mapy do rastrového obrázku

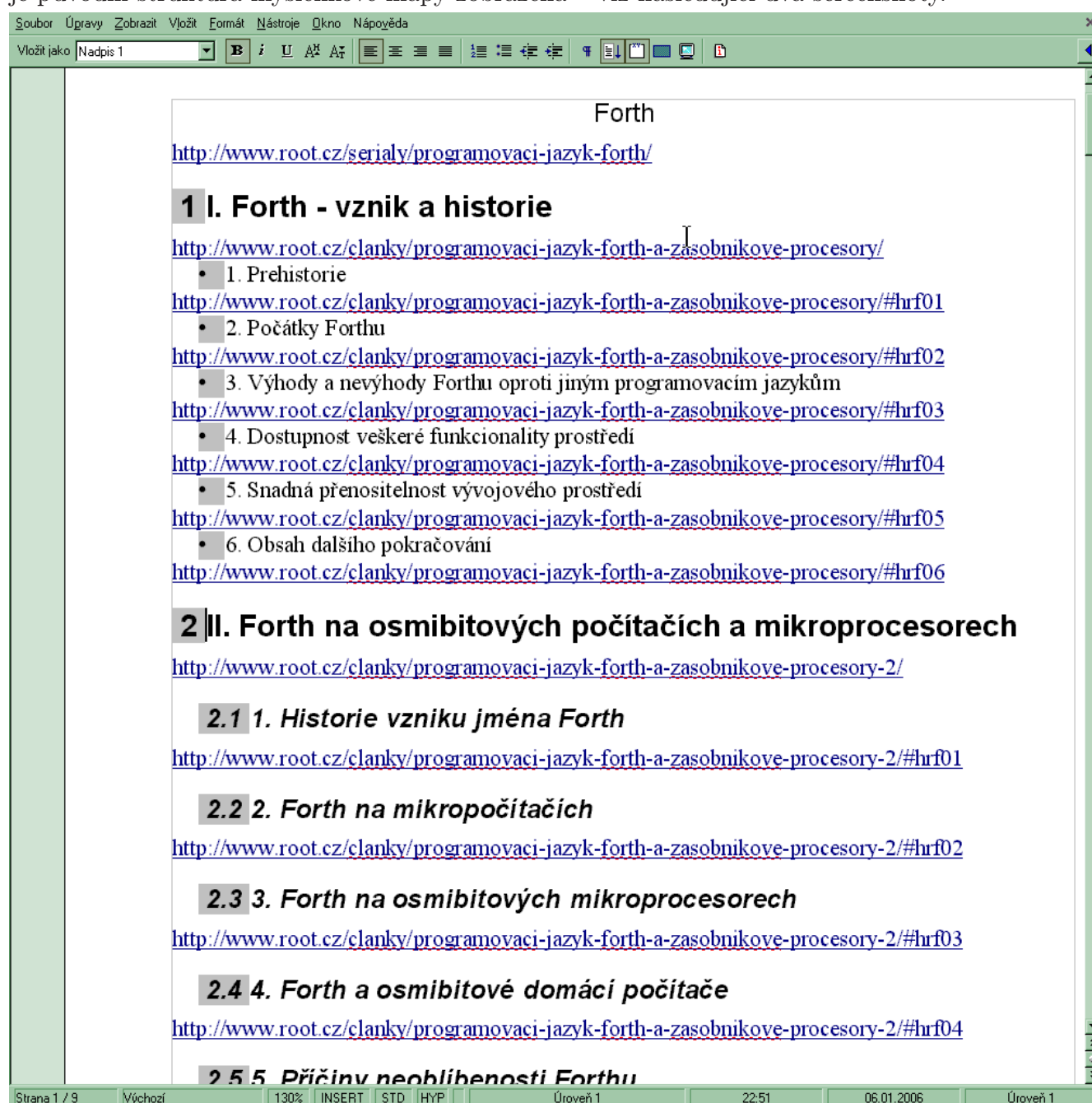
Myšlenkovou mapu je možné vyexportovat i do rastrového obrázku. V tomto případě se však musíme smířit s tím, že výsledné soubory budou (navzdory použitým formátům s dobrou kompresí obrazových dat) objemné a tisk mnohdy nekvalitní či problematický. Na druhou stranu si takto vyexportovanou myšlenkovou mapu mohou prohlédnout prakticky všichni uživatelé, díky knihovně AALIB dokonce i na textovém terminálu :-).

V současné verzi *FreeMindu* jsou pro export podporovány rastrové obrázky uložené ve formátu JFIF (nazývaného také JPEG) a PNG (*Portable Network Graphics*). Pokud to není výslovně zakázáno,

jsou výsledné obrázky vykresleny s využitím antialiasingu, což zajistí kvalitní zobrazení na monitoru (a roztřepané okraje vektorů v případě tisku na některých tiskárnách). Další rastrové formáty nejsou podporovány, což však není příliš na škodu, protože v dnešní době je k dispozici prakticky nepřeberné množství různých konvertorů rastrových obrázků.

Export myšlenkové mapy do formátu OpenOffice.org

Po výběru příkazu *File – Export – As OpenOffice Writer Dokument* se myšlenková mapa uloží jako formátovaný text do souboru s koncovkou *.sxw*. Tento soubor, jehož vnitřní formát je mimochodem opět založen na dnes snad všudypřítomném jazyku XML :-), je možné otevřít a dále zpracovat v kancelářském balíku OpenOffice.org (je možné použít jak verzi 1.x, tak i verzi 2.x). Vzhledem k tomu, že je hierarchie uzlů a poduzlů myšlenkové mapy převedena na kapitoly, podkapitoly a seznamy v textovém dokumentu, je základní formát myšlenkové mapy zachován. Při práci s myšlenkovými mapami převedenými na formát OpenOffice.org je výhodné používat takzvaný Navigátor, ve kterém je původní struktura myšlenkové mapy zobrazena – viz následující dva screenshoty.



Obrázek 9: Myšlenková mapa převedená do formátu SXW a zobrazená v OpenOffice.org Writeru

Možnosti importu myšlenkových map

Import myšlenkových map je ve *FreeMindu* propracovaný stejně dobře jako export dat. Myšlenkové mapy je možné navzájem spojovat, načíst adresářovou strukturu nebo naimportovat záložky ve formátu aplikace Internet Explorer. Podrobnosti o importu jsou uvedeny v dalších podkapitolách.

Import další myšlenkové mapy

Aplikace *FreeMind* umožňuje, aby se velmi jednoduchým způsobem daly vzájemně sloučit dvě či více myšlenkových map. Po výběru příkazu *File – Import – Branch* se zobrazí dialog nabízející výběr libovolného souboru s uloženou myšlenkovou mapou (jedná se o soubory s koncovkou *.mm*). Vybraná myšlenková mapa se vloží do stávající (tj. právě editované) myšlenkové mapy tak, že se stane součástí aktivního uzlu.

Import adresářové struktury

Po zadání příkazu *File – Import – Folder Structure* se zobrazí dialog umožňující výběr libovolného adresáře. Po potvrzení výběru nějakého adresáře se do právě editované myšlenkové mapy vloží uzly, které daný adresář (a samozřejmě i případné podadresáře) graficky reprezentují. Tímto způsobem je možné provést rychlý náhled na adresářovou strukturu – výsledek je podobný výstupu z příkazu *tree*.

Import záložek z aplikace Internet Explorer

Následující příkaz, který je vlastně obdobou příkazu předchozího, umožňuje načíst a zpracovat data z adresáře, ve kterém jsou obsaženy záložky WWW prohlížeče Internet Explorer. Tento WWW prohlížeč (ale nejenom on, i další prohlížeče tento formát záložek mohou zpracovat, například formou importu a exportu) ukládá každou záložku do samostatného souboru s koncovkou *.url*. V tomto souboru, jehož struktura je podobná souborům typu *.ini*, je uložen odkaz na nějakou HTML stránku pomocí unifikovaného lokátoru, dále je zde uložen čas poslední modifikace souboru a popřípadě i další informace nutné například pro správný odkaz na stránku zobrazenou v rámečcích. Po zadání příkazu *File – Import – Explorer Favorites* se zobrazí dialog pro výběr adresáře, ve kterém jsou uloženy záložky – *FreeMind* tedy nedokáže ze systémové konfigurace automaticky zjistit, kde jsou záložky uloženy (to však uživatelům Linuxu naprosto nemusí vadit). Vybraný soubor se naimportuje tak, že nově vytvořené uzly obsahují kromě svého popisného textu i odkaz (URL) na danou stránku.

Import dat z aplikace MindManager

Příkazem *File – Import – MindManager Map* je možné provést import dat z aplikace *MindManager*. Importuje se vždy celá myšlenková mapa, která je posléze připojena k právě editované mapě, podobně jako u výše popsáných importů adresářové struktury a záložek.

Formát souborů aplikace FreeMind

FreeMind ukládá myšlenkové mapy do souborů s koncovkou *.mm*. Tento souborový formát, který *FreeMind* používá interně i při pravidelném ukládání zálohy myšlenkové mapy, je založený na značkovacím jazyku XML (*eXtensible Markup Language*). Tím pádem se jedná o formát, který je dobře čitelný jak člověku tak i aplikacím, samozřejmě za předpokladu použití jednoho z mnoha nabízených parserů XML, protože „ruční“ načítání je složitější. Jedinými vážnějšími vadami použitého formátu je neexistence korektní hlavičky (prologu) XML, což může způsobit některým parserům problémy, a dále pak způsob kódování českých znaků (a samozřejmě i dalších znaků a paznaků mimo standardní sedmibitové ASCII) – české znaky jsou totiž zapsány pomocí svého číselného kódu. V době masivního prosazování kódování UTF-8 je to trochu škoda, ruční editace vytvořených souborů je totiž složitější. Stromová struktura myšlenkové mapy je takřka přesně promítnuta i v souboru. Údaje o jednotlivých uzlech jsou uloženy mezi značkami `<node>` a `</node>`, mezi těmito značkami jsou umístěny údaje

o vložených uzlech – značky pro uzly se tedy mohou rekurzivně zanořovat. Každá značka `<node>` může obsahovat několik atributů. Nejdůležitějším atributem je `TEXT`, do kterého je zapsán řetězec zobrazený v daném uzlu. Pokud je v uzlu zapsán odstavcový text, je značka pro konec řádku kódována jako `
`, celý odstavec se tedy při uložení do souboru nachází na jediném řádku. Dalším použitým atributem je `ID` (jednoznačná identifikace uzlu pro potřeby odkazů), `CREATED` (čas vytvoření uzlu), `MODIFIED` (čas poslední modifikace uzlu), `FOLDED` (příznak, zda jsou poduzly srolovány či zobrazeny) a `POSITION` (relativní pozice uzlu vůči jeho sousedům). Kromě značek pro uzly může soubor obsahovat i další značky, například nepárovou značku ``, kterou se specifikuje změna použitého písma od daného místa myšlenkové mapy. Následuje ukázka souboru s uloženou myšlenkovou mapou:

```
<map version="0.8.0">
<!-- To view this file, download free mind mapping software FreeMind \\
  from http://freemind.sourceforge.net -->
<node COLOR="#ff6600" CREATED="1136753299450" ID="Freemind_Link_1989779260" \\
  LINK="http://www.root.cz/serialy/programovaci-jazyk-forth/" \\
  MODIFIED="1136753299450" STYLE="bubble" TEXT="Forth">
<font NAME="SansSerif" SIZE="24"/>
<node COLOR="#006699" CREATED="1136753299450" ID="Freemind_Link_1740185041" \\
  MODIFIED="1136753346800" POSITION="right" TEXT="I. Forth - vznik a historie">
<node CREATED="1136753299450" ID="Freemind_Link_236132652" \\
  MODIFIED="1136753352790" TEXT="1. Prehistorie"/>
<node CREATED="1136753299450" ID="Freemind_Link_483166048" \\
  MODIFIED="1136753367070" TEXT="2. Počátky Forthu"/>
<node CREATED="1136753299450" ID="Freemind_Link_1132275731" \\
  MODIFIED="1136753371460" TEXT="3. Výhody a nevýhody Forthu oproti \\
  jiným programovacím jazykům"/>
<node CREATED="1136753299450" ID="Freemind_Link_1293488331" \\
  MODIFIED="1136753374430" TEXT="4. Dostupnost veškeré funkcionality prostředí"/>
<node CREATED="1136753299450" ID="Freemind_Link_703263967" \\
  MODIFIED="1136753377670" TEXT="5. Snadná přenositelnost vývojového prostředí"/>
<node CREATED="1136753299450" ID="Freemind_Link_666580593" \\
  MODIFIED="1136753381510" TEXT="6. Obsah dalšího pokračování"/>
</node>
```

Závěr

V tomto krátkém seriálu jsme si popsali možnosti aplikace *FreeMind*. Na závěr je nutné poznamenat, že se tato aplikace poměrně intenzivně vyvíjí – od verze 0.7 k verzi 0.8 prodělal celý program velké změny, můžeme se tedy jen těšit na celočíslnou „jedničkovou verzi“. I za současného stavu se však jedná o plnohodnotnou a stabilní aplikaci, kterou osobně ke své spokojenosti používám prakticky každý pracovní den.

Odkazy

- [1] <http://java.sun.com/j2se>
- [2] http://freemind.sourceforge.net/wiki/index.php/Main_Page
- [3] <http://www.root.cz/formatovane-dokumenty-a-tiskove-sestavy-v-jave-2>

Kalendář dostupný odkudkoliv?

Zdeněk Burda

Grafický program, přenositelný formát dat, bezpečný přístup z webu, sdílení s dalšími lidmi. Jak takový kalendář zprovoznit?

Během posledních několika let jsem zkusil různé webové kalendáře, ale většinou jsem u nich moc dlouho nezůstal. Velké množství webových kalendářů používá vlastní formát pro uchovávání dat a případná změna aplikace znamená ztrátu všech údajů. Při poslední změně programu jsem se našel a rozhodl se pro definitivní řešení – přechod na program s přenositelným formátem dat. Zadal jsem si následující podmínky pro výběr vhodného kalendáře.

- Dostupnost odkudkoliv.
- Možnost využít klasickou desktopovou grafickou aplikaci.
- Webový přístup ke kalendáři, stačí pouze pro čtení.
- Standardní formát pro uchovávání dat, abych v budoucnosti neměl problémy s převodem do jiného programu.
- Bezpečný přístup ke kalendáři.
- Možnost zpřístupnit kalendář někomu dalšímu.

Nakonec jsem si vybral následující řešení:

- Formát dat: iCalendar
- Způsob uložení: WebDAV přes HTTPS
- Aplikace na desktop: Mozilla Calendar
- Webový přístup: PHP iCalendar

Formát dat iCalendar je standard ([RFC 2445 \[1\]](#)) pro výměnu kalendářových dat. iCalendar je podporován většinou běžně používaných programů jako jsou Mozilla Calendar ([Mozilla Sunbird \[2\]](#)), Microsoft Outlook, KOrganizer, Apple iCal, Ximian Evolution nebo třeba Lotus Notes.

WebDAV protokol nám umožní ukládat kalendář na web přímo z aplikace Sunbird. Nelíbí se mi představa, že by se heslo ke kalendáři a vlastně celý můj kalendář potuloval po internetu v čistém textu, proto používám WebDAV přes HTTPS. WebDAV mám spuštěný na vlastním serveru. Pokud vlastní server nemáte a váš webhosting WebDAV nepodporuje, můžete využít například služby [icalx \[3\]](#), která slouží ke sdílení kalendářů.

Pro práci s kalendářem na desktopu si můžete vybrat z více aplikací, já jsem si zvolil program Mozilla Sunbird případně Mozilla Calendar. Samozřejmě můžete použít KOrganizer, Evolution nebo třeba Outlook. Pro webový přístup ke kalendáři použijeme program PHP iCalendar; je relativně jednoduchý na nastavení a rychle se v něm zorientujete.

Zapnutí podpory WebDAV v Apache

Nebudu zde popisovat kompletní nastavení Apache webservru, od toho je oficiální dokumentace na httpd.apache.org [4]. Ukázka konfigurace (jde o umístění a názvy souborů), kterou uvedu, je z distribuce Debian a platí pro webserver Apache 2.

Podpora WebDAV je standardní součástí Apache. Aktivujeme ji přidáním modulu do konfigurace webservru, to samé provedeme pro modul `mod_userdir`. Modul `userdir` nám umožní jednoduše nastavit pro každého uživatele adresář pro web, v našem případě pro kalendář.

```
cd /etc/apache2
ln -s mods-available/dav.load mods-enabled/
```

```
ln -s mods-available/dav_fs.load mods-enabled/
ln -s mods-available/dav_fs.conf mods-enabled/
ln -s mods-available/userdir.load mods-enabled/
ln -s mods-available/userdir.conf mods-enabled/
```

Povolíme pro uživatelské adresáře WebDAV. Soubor `/etc/apache2/mods-enabled/userdir.conf` upravíme takto:

```
<IfModule mod_userdir.c>
  UserDir public_html
  UserDir disabled root
  <Directory /home/*/public_html>
    <IfModule mod_dav.c>
      Dav On
    </IfModule>
    AllowOverride FileInfo AuthConfig Limit
    Options MultiViews Indexes SymLinksIfOwnerMatch IncludesNoExec
  </Directory>
</IfModule>
```

A na závěr restartujeme webserver:

```
/etc/init.d/apache2 restart
```

Pokud vyrobíte v domácím adresáři uživatele (v mém případě uživatel *tsunami*) adresář `public_html/` a nastavíte mu práva tak, aby do něj mohl webserver přistupovat a číst, budete mít jeho obsah dostupný na `http://server/~tsunami/`. Pro umístění kalendářů a povolení zápisu přes WebDAV si jako uživatel vyrobíme adresář `~tsunami/public_html/kalendar` a nastavíme mu práva na 777. Pokud se vám nechce práva takhle hloupě nastavovat, tak je nastavte na 770 a změňte skupinu toho adresáře na `www-data` (nebo pod čím vám běží Apache). Zápis přes `webdav` totiž probíhá pod uživatelem, pod kterým běží webserver. Aby nám nemohl každý kolemjdoucí lézt do kalendáře, je vhodné ho schovat za heslo.

```
order allow,deny
allow from all
AuthName "Kalendar"
AuthType Basic
AuthUserFile ~tsunami/public_html/kalendar/.htpasswd
AuthGroupFile /dev/null
require valid-user
```

Soubor s heslem pro uživatele *tsunami* vytvoříme utilitou `htpasswd`:

```
htpasswd -c ~tsunami/public_html/kalendar/.htpasswd \textit{tsunami}
New password:
Re-type new password:
Adding password for user tsunami
```

Nastavení webserveru, které jsem zde ukázal jako příklad, je to nejjednodušší, které může být – uživatelé mají plná práva. Pokud chcete uživatele omezit, tak se zaměřte na volby `<Limit>` případně

<LimitExcept>. Samozřejmě nemusíte mít nastavení v souboru `.htaccess`, ale můžete vše napsat přímo do konfiguračních souborů Apache.

Když už vám bude vše fungovat přes HTTP protokol, tak zkuste i HTTPS. Přece nechcete, aby vaše kalendáře poletovaly po síti v plaintextu :-).

Zkouška WebDAV

Pro vyzkoušení přístupu na webdav použijeme program `litmus`.

```
litmus https://server/~tsunami/kalendar/ login heslo
-> running 'basic':
0. init..... pass
1. begin..... pass
2. options..... pass
.
.
.
38. finish..... pass
<- summary for 'locks': of 39 tests run: 36 passed, 3 failed. 92.3%
-> 1 warning was issued.
See debug.log for network/debug traces.
```

Test proběhl, ale nahlásil 3 chyby. Vzhledem k tomu, že to jsou chyby, které nevadí, prohlásím webdav za správně nastavený :-). Pro přístup na webdav je možné použít řádkového klienta `cadaver` nebo namountovat vzdálený webdav jako filesystem. Pro přimountování je třeba nainstalovat `davfs` [5].

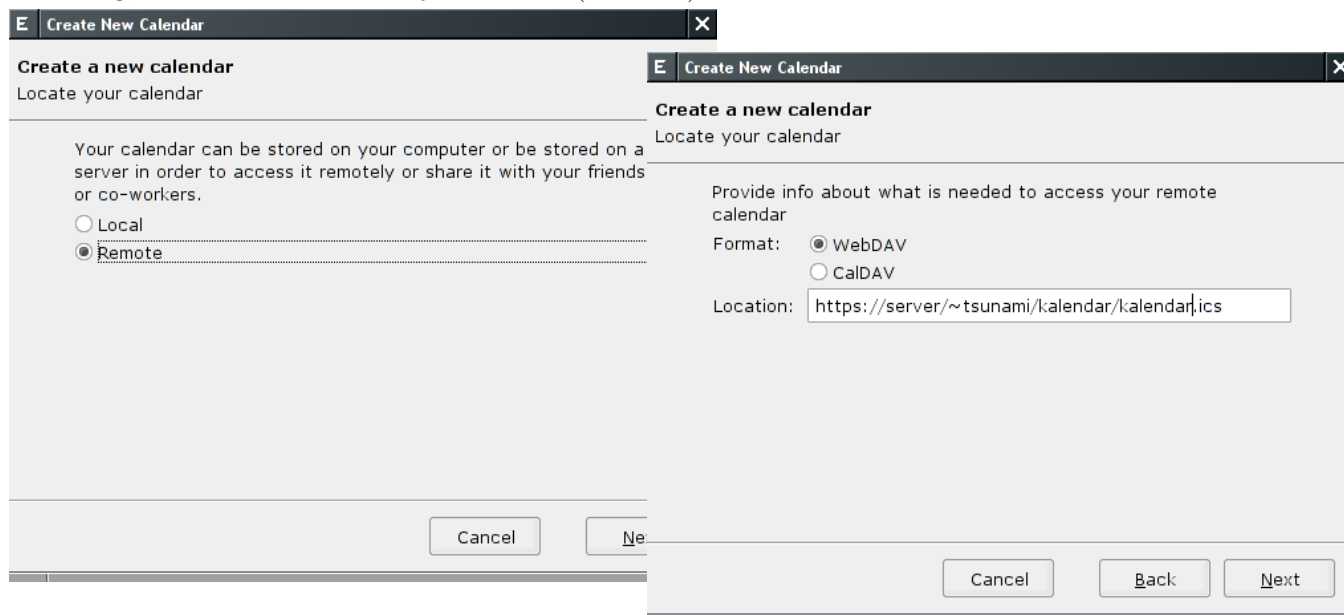
```
vodik:~# mount -t davfs https://server/~tsunami/kalendar/ /mnt
Please enter the username for authentication with server
https://server/~tsunami/kalendar/ or hit enter for none.
Username: tsunami
Please enter the password to authenticate tsunami with server
https://server/~tsunami/kalendar/ or hit enter for none.
Password:
Server certificate could not be verified.
presented for 'server':
Issuer: server, Zdenek Burda, Prag, Czech Republic, CZ
Subject: server, Zdenek Burda, Prag, Czech Republic, CZ
Fingerprint: 8c:98:55:32:84:f1:5b:74:b2:cf:7f:d0:26:b8:23:6f:e7:d5:9e:f4
If you can't verify the fingerprint the server may be faked
or there may be a man-in-the-middle-attack!
I'm not a coward and accept the certificate anyway [y,N]?
y
```

Mozilla Sunbird

Program `Mozilla Calendar` [6] je dostupný jako rozšíření pro Mozilla Suite, případně Firefox a Thunderbird. Pokud raději chcete úplně samostatnou aplikaci, můžete zkusit `Mozilla Sunbird` [7]. Další nastavení bude odpovídat Sunbirdu, ale na `Mozilla Calendar` půjde určitě taky aplikovat.

Vytvoření kalendáře

Vzdálený kalendář se vytvoří pomocí „průvodce“. Začněte v menu *File – New Calendar File*, otevře se dialog, kde zvolíme vzdálený kalendář (*Remote*).



Po potvrzení můžeme vybrat způsob, jakým budeme ke kalendáři přistupovat, a kde bude uložen. Zvolíme protokol WebDAV a jako místo, kde bude kalendář, napíšeme kompletní adresu <https://server/~tsunami/kalendar/kalendar.ics> [8]. Kalendářů si samozřejmě můžete vyrobit víc, takže je vhodné zvolit dobré jméno.

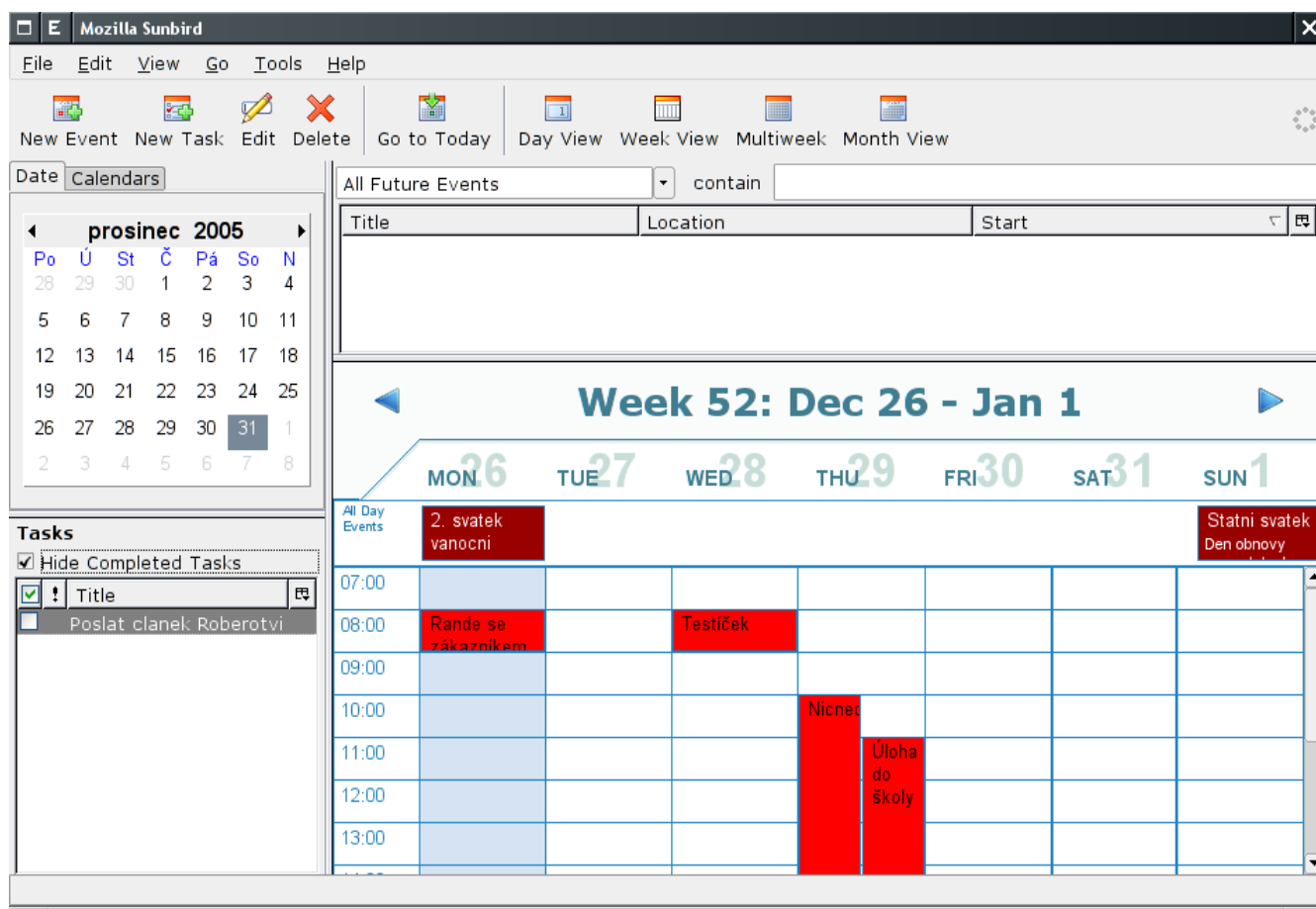
Poslední dialog umožní pojmenovat kalendář a zvolit barvu, jakou se bude zobrazovat.



Po potvrzení se vás kalendář pravděpodobně zeptá na to, zda se vám líbí SSL certifikát serveru, na kterém je webdav, a poté na login a heslo. Po odklepnutí se mi ukázalo varování, že je kalendář pouze pro čtení (*Read-only*).

Náprava je jednoduchá, prostě zrušíme Read-only ve vlastnostech daného kalendáře. Pravým tlačítkem myši klikněte v seznamu kalendářů na náš kalendář, zvolte *Edit Calendar* a v dialogu, který se otevře, zrušte zatržení u *Read-only*.

Nyní můžete začít ládovat do kalendáře data. Pokud se rozhlédnete po internetu, najdete zajímavé kalendáře. Jeden z nich (seznam státních svátků) najdete na [stránce](#) [9] Filipa Molčana.



Co s tím, když něco nefunguje?

Pokud vám něco nefunguje, zeptejte se ve [fóru \[10\]](#) věnovaném kalendáři.

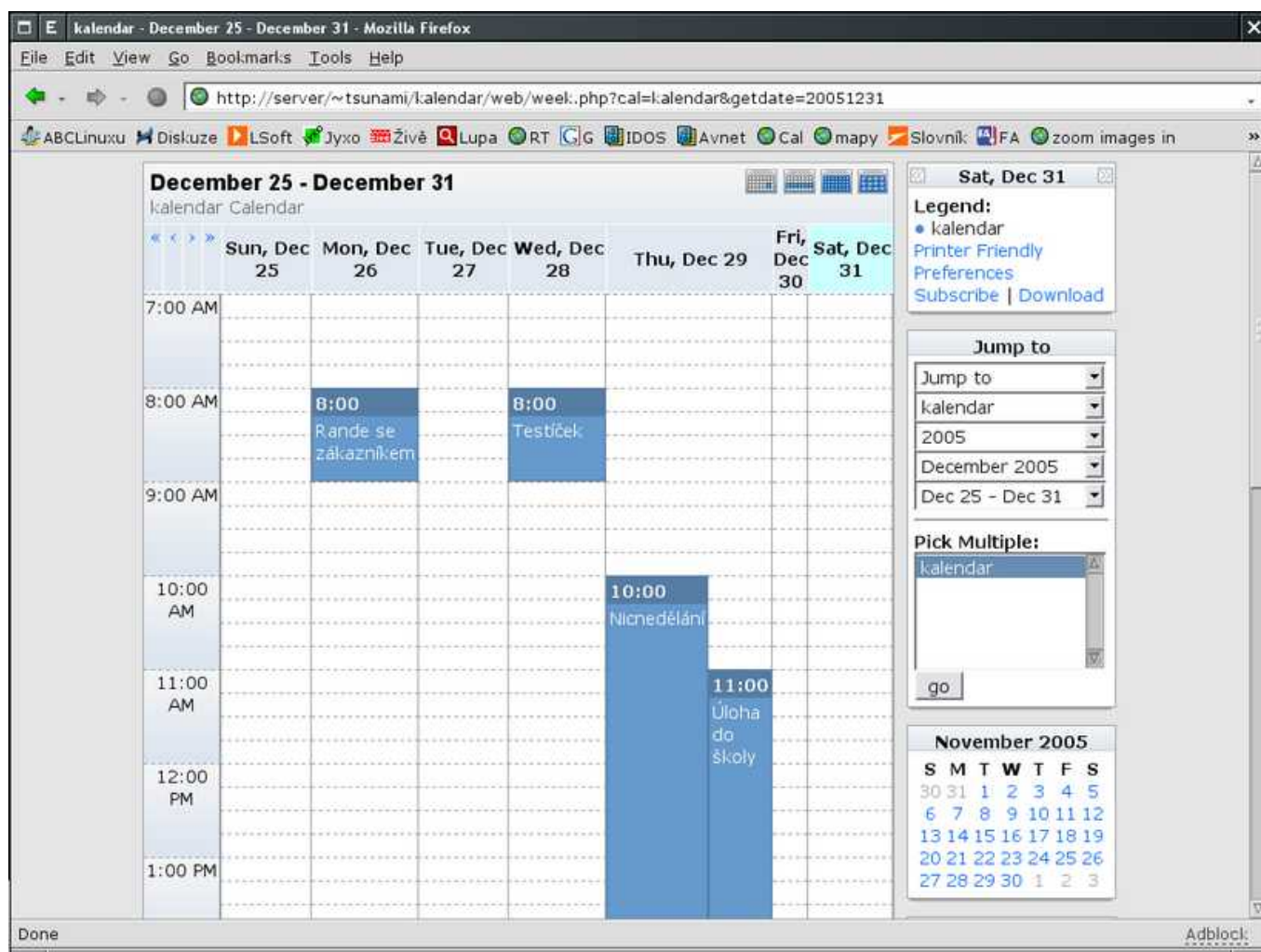
Webový přístup

Přístup přes web se hodí, pokud nejste zrovna u svého počítače a nechcete/nemůžete nastavit Sunbird tak, aby zobrazoval vaše kalendáře. Pro zobrazení kalendáře na webu použijeme zmiňovaný [PHP iCalendar \[11\]](#) (aktuálně ve [verzi 2.1 \[12\]](#)).

PHP iCalendar rozbalte například do adresáře `/public_html/kalendar/web/`. Přesuňte se do adresáře s kalendářem, smažte adresář `calendars` a místo něj si vytvořte symlink na adresář `/public_html/kalendar/`.

```
$ cd ~/tsunami/public_html/kalendar/web
rm -rf calendars
ln -s ../ calendars
```

Tím dosáhneme zveřejnění všech kalendářů, které si vyrobíme, na webu. Pokud potřebujete zveřejňovat pouze nějaké kalendáře, nemusíte mazat adresář `calendars`, ale můžete si do něj požadované kalendáře nalinkovat. Konfiguraci webového kalendáře můžete ovlivnit v souboru `config.inc.php`.



Něco na závěr

Popsané řešení zřejmě nebude pro spoustu lidí dostačující, ale mně vyhovuje. Zlepšení vidím ve vhodnějším nastavení práv pro WebDAV přístup ke kalendářům tak, aby se daly sdílet jednotlivé kalendáře s různým oprávněním pro různé uživatele. Možná by se hodila i editace kalendáře přes web, třeba existuje vhodná webová aplikace. Pokud o ní někdo víte, napište do diskuze.

Odkazy

- [1] <http://www.ietf.org/rfc/rfc2445.txt>
- [2] <http://www.abclinuxu.cz/clanky/recenze/sunbird>
- [3] <http://www.icalx.com/>
- [4] <http://httpd.apache.org/docs/2.0/>
- [5] <http://dav.sourceforge.net/>
- [6] <http://www.mozilla.org/projects/calendar/>
- [7] <http://www.abclinuxu.cz/clanky/recenze/sunbird>
- [8] <https://server/~tsunami/kalendar/kalendar.ics>
- [9] <http://www.mylinux.cz/?q=node/19>
- [10] <http://forum.czilla.cz/viewforum.php?f=24>
- [11] <http://phpicalendar.net/>
- [12] <http://prdownloads.sourceforge.net/phpicalendar/phpicalendar-2.1.tgz?download>

Distribuce OpenWrt prakticky

Petr Bravenec

WiFi přístupový bod a router Asus wl500g Deluxe: instalace nového firmware, používání minimalistické distribuce OpenWrt. Článek popisuje nahrání nového firmwaru, přihlášení, konfiguraci WAN, zmiňuje specifika paměti nvrám a spuštění firewallu.

Úvod

Zhruba před rokem jsem si domů na hraní opatřil malou linuxovou mašinku - Wifi access point a router wl500gx od Asusu. Router běží na operačním systému Linux a navíc může celé zařízení komunikovat se světem i přes dva USB 2.0 porty. Původní firmware od Asusu je přístupný pouze přes internetový prohlížeč, ale dostupné zdrojové kódy dovolují firmware upravit a vytvořit si v routeru vlastní prostředí.

Zřejmě nejbližší má k firmwaru „opravdovému od Asusa“ Olegův firmware. Nastavení zůstává téměř stejné, přibylo jen několik vylepšení. Olegův systém bude vyhovovat především lidem, kteří chtějí používat router v takové podobě, která se co nejvíce blíží původnímu záměru výrobce. Z mého hlediska však přinesl Oleg do routeru jednu z nejpodstatnějších věcí: příkazovou řádku.

Jakmile jsem se dovedl k routeru připojit pomocí telnetu či ssh, začal jsem experimentovat s Linuxem v routeru trochu více. Nicméně mě Olegův software nijak neuchvátil, protože jeho orientace na zpětnou kompatibilitu s původním Asus firmware a zaměření na webové nastavení je velmi omezující. Dalším omezením Olegova firmware je nepřepisovatelný souborový systém squashfs. Připojil jsem tedy k routeru disk přes USB a zkusil nainstalovat nějakou normálnější distribuci – Gentoo. Po pár dnech hráčekování jsem Gentoo vzdal a odložil router k ledu. Až poměrně nedávno jsem narazil na systém OpenWrt a zajásal jsem: tohle je přesně ono – systém, který se snaží tvářit jako systém a ne jako soubor utilit na nastavování jednoho konkrétního kusu hardware.

Moje nadšení z distribuce OpenWrt pramení možná z toho, že jsem občas postaven před požadavek zajistit někde mimo civilizaci jednoduchý systém pro sběr dat – wl500 & OpenWrt mi nabízí lacinou a spolehlivou alternativu k obvyklému PC či k čistě hardwarovému řešení (8051).

Odkazy

<http://oleg.wl500g.info/> [1]

Olegův firmware je vhodný pro všechny, kteří chtějí mnohem více, než co routery od Asusu standardně nabízejí, ale nechtějí si příliš lámat hlavu s Linuxem samotným. Olegův firmware je vytvořen zcela v původním duchu Asusu.

<http://www.openwrt.org/> [2]

Systém OpenWrt je minimalistický Linux pro bezdrátové routery. Narozdíl od Olegova firmware přichází OpenWrt s minimem přídatných funkcí, ale balíčkovací systém dovoluje doinstalovat další programy, případně odinstalovat programy, které jsou v základní distribuci, ale nechcete je používat. OpenWrt se tak více podobá běžným distribucím Linuxu na PC.

<http://wl500g.info/> [3]

Diskusní forum o routeru wl500 a jeho příbuzných. I když se většina otázek týká softwarového vybavení, lze zde najít odpovědi i na dotazy typu „Jak rozšroubovat router?“ nebo „Jak rozšířit router o RS232?“

<http://www.macsat.com/> [4]

Stránky orientované na wl500, OpenWrt a Olegův firmware. Na stránkách je poměrně přehledná základní dokumentace. I když je na tomto webu informací méně, než na stránkách OpenWrt, jsou zde informace podané z trochu jiného úhlu pohledu a rozhodně stojí za to prostudovat i tento web.

System OpenWrt

Firmware v každém podobném zařízení předkládá uživateli hotovou sestavu funkcí, které lze jen těžko rozšiřovat nebo nahradit nepoužívané funkce jinými požadovanými vlastnostmi. OpenWrt se svými vlastnostmi podobá spíše běžným distribucím na PC. Základní firmware obsahuje jen minimální funkčnost, kterou lze dále rozšiřovat instalací dalších softwarových balíčků. V zařízení je firmware uloženo v paměti flash a OpenWrt umí zpřístupnit tuto paměť i pro zápis běžnými systémovými nástroji (textovým editorem). Jiné typy firmware jsou spíše jen sbírkou utilit, které můžeme či nemusíme používat, ale nelze se jich zbavit jinak, než přepsáním celého firmware. V kontrastu s tím připomíná práce s OpenWrt spíše práci na vzdáleném serveru – systém je možné přizpůsobovat momentálním podmínkám přímo za provozu.

Protože je systém OpenWrt určený především pro bezdrátové routery, je skladba balíčků orientovaná hlavně tímto směrem. Nechybí balíčky pro nastavení firewallu (iptables), pro vzdálený přístup k souborům (ftp, nfs, samba), pro dálkovou správu (telnet, ssh). Lze zprovoznit dokonce i http server a php.

Systém může běžet na dvou různých filesystémech: squashfs a jffs2. Squashfs je odolnější vůči haváriím (systém je pouze pro čtení) a jsou v něm uloženy všechny balíčky firmware; zbytek systému (adresář /etc a podobně) je uložený na filesystému jffs2. Filesystém jffs2 je navržený firmou Axis speciálně pro embedded zařízení s pamětí flash. I tento filesystém by měl být odolný vůči výpadkům a haváriím a dovoluje pracovat s pamětí flash podobně jako s pevným diskem v běžném počítači.

Na čem lze OpenWrt provozovat

OpenWrt lze provozovat na procesorech Broadcom. Zařízení s tímto procesorem je celá řada. Na stránce <http://wiki.openwrt.org/TableOfHardware> [5] můžete vyhledat svoje zařízení a přesvědčit se, jestli na něm lze OpenWrt zprovoznit.

Já jsem dostal do rukou pouze dvě zařízení schopná provozovat OpenWrt: Asus wl500g Deluxe a Asus wl300g. Výhodou obou strojů je velmi snadná dostupnost – obě zařízení se prodávají v cenách kolem dvou a půl tisíc, model wl500g je o pár stokorun dražší.

Co vlastně obsahuje Asus wl500g deluxe

Router obsahuje 200 MHz procesor Broadcom, 32 MB RAM, 4 MB paměti flash, dva porty USB 2.0, šest (snad) nezávislých 10/100 Mbit ethernetových rozhraní a wifi jednotku. Na jednom ethernetovém rozhraní je připojena wifi, další rozhraní je použito pro WLAN (připojení na internet) a zbývající čtyři rozhraní jsou propojena ve switchi pro připojení lokální sítě. V případě potřeby by mělo jít upravit rozdělení ethernetových portů – použít jeden port například pro připojení na internet, jeden port pro připojení demilitarizované zóny a zbývající tři porty spolu s wifi použít pro připojení lokální sítě.

Velikou výhodou jsou dva USB porty. V závislosti na nainstalovaném software lze připojit k USB například tiskárnu, disk či kameru (a vytvořit si jednoduchý systém pro hlídání objektu). Existují i experimentální drivery bluetooth zařízení, našinci mohou ocenit i možnost připojení k internetu přes CDMA modem od Eurotelu. Zprovoznění kamery nebo CDMA modemu bude pravděpodobně jednodušší s Olegovým systémem, OpenWrt je ve své základní verzi koncipovaný příliš minimalisticky.

Zkušenosti z provozu

Na disku je strašně málo místa a instalace dodatečných balíčků trvá velmi, velmi dlouho, zvláště v případě, že místo na disku dojde. Nevydržíte-li s nervy a pokillujete rozjeté instalační procesy, můžete se dostat do stavu, že ipkg přestane fungovat úplně, a pak jsem našel asi jediné řešení – kompletní reinstalace firmware. Až při psaní tohoto článku jsem objevil soubor `/usr/lib/ipkg/status` (textový soubor s informacemi o nainstalovaných balíčcích). Cítím šanci, že až příště rozbourám

balíčkovací systém, nebudu muset sahat po všeléku, který sluší spíše neunixovým typům operačních systémů.

Filesystem jffs2 se chová ve srovnání s normálními filesystemy značně nezvykle. Na disk, na kterém je 400 KB volného místa, není problém překopírovat 500 KB veliký soubor, ale na dokončení celé operace si můžete počkat třeba i několik desítek minut. Přerušeni může mít za následek ztrátu místa na disku – naštěstí mi zatím vždy pomohl restart.

Pokud už sáhnu po reinstalaci, oceňuji, že velká část nejzákladnějšího nastavení je v paměti nvram, takže alespoň síťovou komunikaci nemusím nastavovat po úspěšné reinstalaci celého systému znovu.

Instalace samotná dokáže být velmi problematičká. Zkusím jednou, podruhé, potřetí... a takhle mne dokáže router zaměstnat několik hodin. Pochopitelně, co už mám taky dělat s wl500 ve stavu oživilé mrtvolky, kdy na mě pouze smutně poblikává ledka, ze všech funkcí routeru funguje jen ping a tftp pro upload systému a nervozita narůstá. To ovšem nebude zřejmě problém systému, jako spíše routeru samotného.

Dlouhé roky jsem pracoval na počítači s procesorem 386 – nevádí mi, že si na některé věci musím chvíli počkat. 386 určitě není rychlejší, než procesor v routeru, ani 32 MB RAM není příliš omezující. Paměť flash pro uložení firmware se dá snadno rozšířit připojením externího disku, ale v momentě, kdy zkusím nastartovat větší množství aplikací, zjistím, že nelze zapnout swap. Shell busybox určený pro odlehčená řešení umí suplovat funkci programu swapon, ale příliš horlivý odlehčovatel při vytváření distribuce tuto možnost z překladu vyloučil. Olegům systém je v tomto směru okleštěný mnohem méně. Naštěstí není problém překopírovat z Olegova systému binárku busyboxu do OpenWrt. Bohužel v busyboxu Olegova systému zase chybí některé funkce, které se používají v OpenWrt.

Běžící systém na routeru wl500g je (dočasně) k nahlédnutí na adrese <http://wl500.bravenec.org> [6]. Všetečky, kteří by snad chtěli zkusit na zařízení portscanning, musím předem upozornit, že porty 135-139,445 filtruje můj ISP, nainstalovaná verze iptables filtrování jednotlivých portů neumí.

Instalace, základní nastavení – download

Pro jednoduché připojení nepotřebujeme žádné další balíčky, stačí z internetu stáhnout pouze základní firmware. V době psaní tohoto článku byla aktuální verze systému OpenWrt Whiterussia RC4: [openwrt-brcm-2.4 – jffs2-4MB.trx](#) [7].

Systém OpenWrt není omezen pouze na router Asus wl500g, běhá na velkém množství dalších podobných zařízení. V adresáři proto najdeme několik dalších verzí pro různé procesory a pro různé veliké flash paměti. Pro wl500g deluxe vybírejte vždy firmware pro procesor Broadcom (ve jméně souboru s firmwarem je napsáno brcm) a flash o velikosti 4 MB.

V adresáři o patro výše je několik podadresářů. Adresáře **bin** a **default** by měly být totožné. V adresáři **micro** je skutečně jen minimální instalace, v adresáři **pptp** je instalace určená pro připojení k internetu protokolem pptp. Systém lze ale jednoduše doplnit o další balíčky, takže máte-li v routeru verzi default, lze ji snadno doplnit o balíčky potřebné pro připojení přes pptp.

Instalace

Ze zadu na routeru je tlačítko označené „Reset“ – pro účely instalace je lze využít hned dvěma způsoby. Když stisknete tlačítko reset na zapnutém routeru přibližně na dvacet vteřin, nastaví se do paměti nvram výchozí hodnoty. Po úspěšném vynulování by měla začít zběsile blikat ledka PWR. Pokud máte router ještě s originálním firmware od výrobce, měl by být po restartu nastavený na adresu 192.168.1.1 na některém z lokálních ethernetových rozhraní (označené LAN) a heslo nastavené na standardní admin/admin. Přišli jste o konfiguraci, ale router by měl ještě stále fungovat. Počítač, ze kterého budete instalovat OpenWrt, nastavte na adresu ze stejného segmentu IP adres, například 192.168.1.10. Na router na adrese 192.168.1.1 musí být možné se dopingat. Pro propojení používejte přímý kabel a k routeru během instalace nic jiného nepřipojujte!

Druhé použití tlačítka reset je nastavení do režimu „upload“. Router na několik vteřin vypněte, stiskněte tlačítko reset, podívejte se na účet za router, abyste věděli, o kolik přicházíte, a router

zapněte. Po pár vteřinách začne pomalu ledka PWR blikat. Pro kontrolu můžete zkusit ping, router musí odpovídat. Pro další postup budete potřebovat klient tftp. Jsem přesvědčený, že tftp by měl být snad v každé linuxové distribuci. Můžete zkusit nahrát systém do routeru:

```
tftp 192.168.1.1
binary
put openwrt-brcm-2.4-jffs2-4MB.trx
```

Po úspěšném nahrání systému router vypněte a zapněte. Ledka PWR by nyní měla svítit – nikoli blikat – a po několika okamžicích, routerovo zmrtvýchvstání trvá poprvé dlouho, by měl začít router odpovídat na ping. Tolik teoretický postup. Prakticky spíše začne opět pomalu blikat ledka PWR a systém nenajede. Zdála se vám částka na účtu za router vysoká? Můžete zkusit pro nahrání systému použít originální recovery utility od Asus (pouze pro Windows), či Olegovu utilitu `recover.sh`:

```
#!/bin/bash

IP=192.168.1.1

if [ -z "$1" ] || [ ! -f $1 ]; then
    echo Usage: $0 firmware.trx
cat << EOF
IMPORTANT:
    1) be sure POWER led is flashing (If this is not a case
        poweroff the device, push the reset button & power on
        it again, then release button)
    2) connect your pc to the LAN port
    3) be sure your link is up and has an address in the
        $IP/24 address range (and not the $IP)
EOF
    exit 0
fi
echo Confirming IP address setting\dots
echo -en "get ASUSSPACELINK\x01\x01\xa8\xc0 /dev/null\nquit\n" | tftp $IP
echo Recovering $IP using $1\dots
echo -en "binary\nput $1 ASUSSPACELINK\nquit\n" | tftp $IP
echo Please wait until leds stops flashing.
```

Moje zkušenosti jsou takové, že ani Olegova utilita `recover.sh` někdy nepomůže. Nebojte se experimentovat – nyní už nemáte co ztratit. Oleg ve své utilitě před nahráním firmware testuje, jestli je router funkční (protokolem tftp downloaduje z routeru soubor), pak teprve zkouší nahrát nový firmware. Nebude-li se vám dařit ani s Olegovou utilitou, zkuste pouze nahrání firmware:

```
tftp 192.168.1.1
binary
put openwrt-brcm-2.4-jffs2-4MB.trx ASUSSPACELINK
```

Nahrával jsem do routeru OpenWrt třikrát – pokaždé jsem se s tím potrápil několik hodin, asi dělám něco špatně. Naposledy jsem nahrával firmware bez vyresetování paměti nvram – nemusel jsem potom

nastavovat základní funkce routeru znovu (například IP adresu jsem měl z rozsahu 10.10.1.0/24 a pro nahrání jsem nepoužíval 192.168.1.1). Po úspěšném nahrání firmware byste se měli být schopni připojit k routeru protokolem telnet. Ve Windows je telnet dodnes standardní součástí systému (klient), do některých distribucí Linuxu bude třeba telnet doinstalovat.

První přihlášení

Po prvním nabootování se připojte k routeru telnetem. Připojit se lze přímo do účtu roota, aniž by se vás systém zeptal na uživatelské jméno či heslo. Jako první musíte změnit heslo uživatele root:

```
passwd
```

načež systém začne (někdy) protestovat:

```
passwd: An error occured updating the password file
```

Chybová hláška je poprvé normální (mně se objevila vždy). Nejsnazším řešením je restart systému příkazem `reboot`. Při dalším přihlášení telnetem už systém pracuje normálně a heslo lze bez problémů změnit. Po dalším restartu už není možné připojení telnetem, je nutné použít ssh – uživatel root a heslo byste si snad mohli ještě pamatovat, zadávali jste je před chvílí.

Paměť nvram

Než začneme systém konfigurovat, bude vhodné si něco říci o paměti nvram. V normální distribuci jsou všechna potřebná nastavení shromážděna někde v adresáři `/etc`. V routeru je to trochu složitější. Pro uložení konfigurace slouží vyhrazená část paměti flash, k níž se dostanete příkazem nvram:

```
nvram show | sort # vypíše všechny proměnné
nvram set promenna1=aaa # nastaví proměnnou
nvram set promenna2="aaa bbb" # řetězce s mezerou uzavírat do uvozovek
nvram commit # nahrát změny do paměti flash
nvram get promenna1 # vypíše pouze jednu proměnnou
```

Po nastavení konfiguračních proměnných nezapomeňte konfiguraci zapsat příkazem `nvram commit`, jinak konfigurace nepřežije nejbližší restart. Takto důsledně je uložená konfigurace například v Olegově systému nebo v původním systému Asus, neboť oba systémy pracují nad filesystémem squashfs, který je pouze pro čtení. OpenWrt obsahuje mix proměnných nvram a klasických konfiguračních souborů v `/etc`.

Další utilita, kterou se lze dostat k paměti flash v routeru (a to nejen k nvram pro konfigurační proměnné), je `mtd`. S její pomocí lze například nahrát nový firmware (bez použití tftp), před další prací ji využijeme k vyčištění nvram od konfiguračních proměnných předchozího systému v routeru:

```
mtd erase nvram
reboot
```

Nastavení Wan

Nyní připojte router k internetu – síťový kabel zasuňte do dírky označené WAN. Na tomto síťovém rozhraní by měl běžet DHCP klient - přidělí-li vám váš ISP adresu dynamicky, jste nyní připojeni k internetu a můžete vyzkoušet ping:

```
ping www.abclinuxu.cz
```

(Nezkoušejte ping na seznam.cz. Doba je divoká a někdo by se vás mohl ptát, kdo na tom seznamu je a jestli byste jej mohli předložit.) Nefunguje? Možná vám váš ISP přiděluje adresu z rozsahu 192.168.1.1/24, pak je potřeba změnit adresu LAN rozhraní:

```
nvram set lan_ipaddr=192.168.0.1
nvram commit
reboot
```

Stále nefunguje? Třeba musíte mít nastavenou statickou adresu. Než začnete nastavovat statickou adresu na internetovém rozhraní, sáhněte si do svědomí a koukněte se, jestli nejste úplná, ale úplná lama – no, ale neočekávám, že byste se v takovém duševním rozpoložení dostali až tak daleko (prostě nyní už musíte vědět, co děláte, a proč to děláte, jako například já, když jsem [minule](#) [8] psal zasvěceně o tom, jak v OpenWrt nefungují iptables):

```
nvram set wan_hostname=<vaše hostname>
nvram set wan_ipaddr=<vaše IP adresa>
nvram set wan_proto=static
nvram commit
reboot
```

Nezapomeňte také na fakt, že statickým nastavením IP adresy nezískáte informaci o DNS serverech. Tu je třeba doplnit ručně do souboru `/etc/resolv.conf` (editorem vi, je součástí distribuce, balíčku busybox). Nyní už by snad měl ping konečně fungovat.

Nastavení Lan a DHCP

Výborný vynález je DHCP server – dokáže nastavit síťové parametry u počítačů, které do vaší lokální sítě připojíte. Návod na [www.macsat.com](#) [9] mixuje nastavení v nvram a v `/etc`. Další návod na [webu OpenWrt](#) [10] tvrdí opět něco trochu jiného. Abych se nelišil, zvolím třetí, vlastní variantu – nastavení pouze přes `/etc/dnsmasq.conf`. Zásahům do konfiguračního souboru se stejně nevyhnu, v nvram nelze nastavit vše (ono by to asi šlo, ale systém OpenWrt s tím nepočítá), a svou vlastní, třetí cestou se vyhnu konfiguraci jedné věci na dvou různých místech. Přesto je třeba nastavit něco málo i v nvram:

```
nvram set lan_ipaddr=192.168.1.1
nvram set lan_netmask=255.255.255.0
nvram set lan_proto=static
nvram set lan_ifname=br0
nvram set lan_ifnames="vlan0"
nvram commit
```

Síťové rozhraní lokální sítě je pojmenováno `br0` – jsou zde mostem (bridge) spojená rozhraní `vlan0` (čtyři ethernetové porty LAN) a později si ukážeme, jak sem připojit ještě bezdrátové rozhraní routeru (wifi). Obsah `/etc/dnsmasq.conf`:

```
domain-needed
bogus-priv
filterwin2k
localise-queries
local=/lan/
```



```

domain=lan

# Rozsah přidělovaných adres:
# interface, odkud, pokud, čas pronájmu
dhcp-range=br0,192.168.1.2,192.168.1.10,12h

dhcp-authoritative
dhcp-leasefile=/tmp/dhcp.leases

# v /etc/ethers můžou mít uložené statické adresy -- jedné síťové
# kartě se přiřadí vždy stejná adresa. Záznamy jsou ve formátu:
# <hwaddr> [<hostname>] <ipaddr>
read-ethers

# Další užitečnosti
# default route(s): dhcp-option=3,192.168.1.1,192.168.1.2
# dns server(s): dhcp-option=6,192.168.1.1,192.168.1.2
# mask: dhcp-option=1,255.255.255.0
# broadcast: dhcp-option=28,10.10.1.255
dhcp-option=1,255.255.255.0
dhcp-option=3,192.168.1.1
dhcp-option=6,192.168.1.1
dhcp-option=28,192.168.1.255

```

Ještě poslední rekvizita, soubor `/etc/init.d/S49dnsmasq` pro nastartování dnsmasq při spuštění systému:

```

#!/bin/sh
dnsmasq

```

Soubor může být takto jednoduchý, nezapomeňte jen nastavit jeho práva tak, aby byl soubor spustitelný:

```

chmod +rx /etc/init.d/S49dnsmasq

```

Po restartu je vaše lokální síť připravena na připojení počítačů.

Nastavení firewallu

Posledním krokem nutným k tomu, abyste se z lokální sítě dostali na internet, je nastavení firewallu. V tomto okamžiku si nejsem jistý, jestli je vše potřebné už nastaveno, nebo je třeba doplnit některé věci ručně. Datum poslední modifikace souboru `/etc/S45firewall` mám nastaveno na 1.1.2000 a já už netuším, jestli to má na svědomí špatně nastavený čas v mém routeru v době, kdy jsem se snad v souboru hrabal editorem, nebo to znamená, že jsou iptables nastavené správně od výroby. Z myšlenky, že bych měl jen kvůli tomuto článku znovu reinstalovat systém, mi naskakuje kopřivka. Zkuste proto ze své lokální sítě (ne z routeru) ping někam do internetu. Funguje-li, zvítězili jste. Nefunguje-li (a z routeru ano), zkontrolujte obsah souboru `/etc/init.d/S45firewall`, téměř na konci by měl být řádek

```
iptables -t nat -A POSTROUTING -o $WAN -j MASQUERADE
```

S největší pravděpodobností tam řádek je, není-li, doplňte do souboru `/etc/firewall.user` pravidlo:

```
iptables -t nat -A postrouting_rule -o $WAN -j MASQUERADE
```

Startovací skript vytváří pro uživatele pravidla `input_rule`, `output_rule`, `forwarding_rule` a v tabulce NAT pravidla `prerouting_rule` a `postrouting_rule`. Ta lze měnit a doplňovat v souboru `/etc/firewall.user`. Do startovacího skriptu `/etc/init.d/S45firewall` není třeba vůbec sahat.

Iptables jsou kapitolou sama pro sebe, zcela mimo rozsah tohoto článku. Naštěstí existuje i na českém webu množství užitečných návodů, zkuste zapátrat v archivu root.cz [11] nebo underground.cz [12]. Někde zvenku z internetu doporučuji překontrolovat nastavení vašeho firewallu programem `nmap`. Router by měl být z venku neviditelný.

Na závěr

Podle dnešního článku o OpenWrt byste měli být schopní nainstalovat systém a zprovoznit základní síťovou komunikaci. Nastavení wifi jsem zatím vůbec nezmiňoval. O tom zkusím napsat něco příště. A úplně na konec jednu otřepanou, ale užitečnou radu:

Nepropadejte panice!

Odkazy

- [1] <http://oleg.wl500g.info/>
- [2] <http://www.openwrt.org/>
- [3] <http://wl500g.info/>
- [4] <http://www.macsat.com/>
- [5] <http://wiki.openwrt.org/TableOfHardware>
- [6] <http://wl500.bravenec.org>
- [7] <http://downloads.openwrt.org/whiterussian/rc4/default/openwrt-brcm-2.4-jffs2-4MB.trx>
- [8] <http://www.abclinuxu.cz/clanky/site/distribuce-openwrt-prakticky-1>
- [9] <http://www.macsat.com/OpenWRT/InstallOpenWRT.php>
- [10] <http://wiki.openwrt.org/Faq#head-54dcdf54ca16ad00d163cf53c0bc5efdb56fee09>
- [11] <http://www.root.cz>
- [12] <http://www.underground.cz>

Pocketlinux 2.0

Martin Tesař

Pocketlinux je distribuce Linuxu, která vychází ze Slackware a snaží se pomocí odlehčeného KDE (KDE Light) nabídnout jednoduché uživatelské prostředí.

Vznik a určení distribuce

Hned na začátku varujeme, že „kapesních linuxů“ existuje vícero a tento článek není ani o jednodiskové distribuci [Pocket Linux](#) [1], ani vývojové platformě pro mobilní zařízení [PocketLinux](#) [2], ale o distribuci „pocketlinux“ [3] na bázi Slackware, vyznačující se desktopovým prostředím nazývaným KDE Light. Navzdory názvu nejde o žádnou miniaturní distribuci v pravém slova smyslu (jako je např. [Damn Small Linux](#) [4]), ani o komfortní live minidistro „do kapsičky“ (jako je např. [Slax](#) [5]).

Autor „pocketlinuxu“ Marcus Moeller si prý během své letité praxe administrátora povšiml, že většina uživatelů bývá množstvím nabízených alternativ aplikací spíše zmatena než potěšena (soudě podle občasných diskusí zde na [ABCLinuxu.cz](#), není se svým míněním osamocen), a proto pojal úmysl vyjít takovým „zmateným“ uživatelům vstříc a sestavit pro ně „ideální“ distribuci. Dle vlastních slov na jinak stručných webových [stránkách projektu](#) [6] se snaží o Linux „small, fast and easy to use (malý, rychlý a snadný k používání)“.

Těsně před Vánocemi vyšla verze s hrdým označením 2.0, což jako by slibovalo, že dětské nemoci projektu už jsou minulostí, ale ještě uvidíme, že tomu tak není. Spíše je „dvojka“ připomínkou skutečnosti, že tato sotva čtyři měsíce stará distribuce je upravený Slackware 10.2. Projektů vycházejících ze Slackware jako lety osvědčené a díky pořádku v konfiguračních souborech mimořádně tvárné distribuce je ovšem celá řada (připomeňme si aspoň už zmíněný [Slax](#), [Zenwalk Linux](#) [7] nebo [UltimaLinux](#) [8]), a tak co „pocketlinux“ odlišuje, je hlavně zvláštní sestava desktopového prostředí („KDE Light“). Potenciálním zájemcem o tuto distribuci bude tedy uživatel, který touží vyzkoušet na vlastní kůži dobrodiní tolik opěvovaného Slackware, a přitom by rád začal s nějakou hotovou předvybranou konfigurací, nejlépe „malou, rychlou a snadnou k používání“. Jak tyto naděje „pocketlinux“ splňuje, to se pokusím objasnit následujícím popisem instalace, konfigurace a vybavení této nové distribuce.

Průvodce instalací

Že jde v případě „pocketlinuxu“ o velmi malý projekt v začátcích, se ukáže hned v okamžiku, kdy se pokusíte z přetížených [zdrojů](#) [9] stáhnout ten jediný nabízený ISO obraz instalačního CD: zpravidla musíte víckrát pokoušet štěstí, případně volit cestu BitTorrent (tímto zároveň budiž sděleno, že autor distribuce hledá volné mirrorové kapacity). Obraz je o něco menší než obvyklá distribuční CD-ROM, ale i tak obsahuje zhruba 490 MB dat, jejichž stahování vám nikdo neušetří (jiný způsob instalace, např. via FTP, zatím neexistuje).

Hned při instalaci se ukáže, že slogan „jednoduché a malé distro“ neznamená nutně, že je určeno naprostým začátečníkům. „Pocketlinux“ totiž používá mírně upravený Slackware instalátor, který sice v textovém režimu provází uživatele všemi potřebnými kroky instalace, ale do „blbuvzdorného“ pohodlí klikacích distribucí má daleko.

Např. budete omezeni v pohybu po instalátoru libovolným směrem (tj. chybí tak často postrádané „tlačítko Zpět“ – jednotlivé bloky instalace však končí vždy ve výchozím velmi jednoduchém menu a lze je tedy v případě potřeby opakovat). Po projektu v tak raném stádiu také nelze vyžadovat obsáhlou lokalizaci do národních jazyků, nicméně v případě, že nejste v angličtině příliš pevní v kramflecích, můžete si alternativně zvolit německé (mateřština tvůrce distribuce) nebo nově i francouzské jazykové prostředí.

Volba prvního bootování obsahuje stejně jako výchozí Slackware stále jádro 2.4.31 (pro IDE/ATAPI bez podpory ACPI; pomocí nabídky můžete zvolit např. podporu SCSI, ATA RAID atd.). Úpravu do budoucna bude patrně vyžadovat poněkud zbytečná a matoucí výzva zalogovat se v příkazovém řádku jako `root` (bez hesla). Již v této fázi jsou nabízeny nástroje `fdisk` nebo `cfdisk` k úpravě disku (k čemuž bude později také příležitost) nebo nastavení sítě (kdybyste ji náhodou během instalace potřebovali). Ze zvědavosti jsem odklepl párkrát `Enter` (tj. volby výchozích nabídek) a skutečně: síťová karta byla automaticky detekována a příslušný modul (`ne2k-pci`) s úspěchem zaveden.

Zbývá už jen na pokyn zadat do příkazové řádky `setup` a zahájit instalační proceduru v jednotlivých blocích nabízených ve výchozím menu se stručnými, nepříliš popisnými položkami. Ten první je vytvoření aspoň jednoho (resp. se swapem dvou) linuxových oddílů na pevném disku. Můžete samozřejmě použít již oddíly předem vytvořené, ale každopádně si předem rozmyslete, kam budete chtít instalovat a které oddíly určitě mazat nechcete, neboť instalátor vám nabídne pouze obvyklé textové linuxové nástroje `fdisk` (`cfdisk`), s nimiž byste se měli předem seznámit – nikoliv průvodce za ručičku v grafické podobě a se spoustou varování vždycky, když ztropíte nějakou nepředloženost.

Dalším blokem výchozího menu je položka TARGET (cíl), pod níž se de facto skrývá konfigurace instalace, která začíná výběrem buď právě vytvořeného nebo předem připraveného cílového oddílu disku pro instalaci. Bude vám nabídnuta možnost formátování oddílu (eventuálně včetně jeho kontroly) a jako souborový systém si můžete vybrat tradiční `ext2`, žurnálovací `ext3` nebo `reiserfs` (doporučuje autor distribuce, já jsem po několika nepříjemných zkušenostech zvolil raději `ext3`). Můžete si pak zvolit i vlastní hustotu uzlů (*Inode Density*), když se nechcete spokojit s výchozím nastavením (*1 per 4096 bytes*) – což by mohlo asi zaskočit nováčky toužící původně po „jednoduché“ distribuci. Totéž by se dalo říci i o samotném procesu formátování, který probíhá „undercover“ (bez *progress baru* i jiných zpětných hlášení), takže neztrácejte nervy, když se chvíli (zejména máte-li větší disk) jakoby nebude nic dít.

Následuje výběr dodatečných bodů připojení se stručnou nápovědou, čehož jsem využil a připojil si (bez formátování!) další už v ostatních nainstalovaných distribucích používaný oddíl jako `/home`. Ihned si můžete zkontrolovat, jak budou vypadat vaše volby v konfiguračním souboru `/etc/fstab` a také možná oceníte, že můžete rovněž ihned a bezbolestně vytvořit a do `/etc/fstab` nechat zapsat přípojný bod pro jiné operační systémy (se souborovým systémem FAT/NTFS): stačí zadat jejich názvy, např. `/winxp`.

návratu z konfigurace instalace do výchozího menu zvolíme další blok: `INSTALL (transfer data)`, který zajistí extrakci souborů z instalačního média a nainstaluje je na patřičné místo. Opět budete asi postrádat *progress bar* nebo aspoň nějaký feedback o tom, co se právě děje, ale už zhruba po pěti minutách (PII 400 Mhz) vám bude nabídnuta ještě volba instalace jiného jádra, než s pomocí kterého jste nabootovali (což je pro začátečníka dost riziková záležitost).

Jako výchozí zavaděč je zamýšlen LILO, který normálně v tomto instalačním bloku můžete nechat nainstalovat. Já jsem na jeho instalaci rezignoval, neboť na svém PC používám jeden jediný GRUB pro všechny operační systémy, který už mám nainstalovaný na zvláštním oddílu a neměl jsem ani z testovacích důvodů chuť s instalací jiného zavaděče experimentovat.

Ještě zbývá zadat na výzvu heslo pro administrátora systému (`root`) a je hotovo: odstraňte CD z mechaniky a restartujte (buď trojhmatem `Ctrl+Alt+Del`, nebo příkazem `reboot`).

Poinstalační konfigurace

Pokud zvolíte, co se týče zavaděče, podobnou cestu jako já, nezapomeňte právě teď upravit z jiné distribuce nebo LiveCD zavaděč (`/boot/grub/menu.lst`) a umístění přípojného bodu `/boot` v souboru `/etc/fstab`, abyste mohli nové distro vůbec spustit. Ve výchozí instalaci přidejte ve startovacím menu GRUB parametr `ro` (*read only*) k povelu `kernel`: budete tak ušetřeni chybové hlášky a přerušení bootování, např.:

```
kernel /vmlinuz root=/dev/hdc15 ro
```

Důležitá věc je rovněž zajistit, abyste mohli používat po rebootu klávesnici a myš. „Pocketlinux“ je totiž stejně jako Slackware nastaven na PS/2 myš i klávesnici, proto musíte pro USB zařízení aktivovat příslušné moduly v `/etc/rc.d/rc.modules` (`/sbin/modprobe usb-uhci`, `usbmouse`, `usbkbd` apod.). Zde se musím přiznat, že se mi USB myš rozběhat nepodařilo a že dokud jsem nezastrčil PS/2 myš, byla blokována i PS/2 klávesnice. Ručně musíte také upravit soubor `xorg.conf`, chcete-li používat kolečko myši.

Distribuce je po instalaci nastavena k automatickému startu grafického prostředí, pokud přesto něco selže (a můžete ovládat aspoň klávesnici), přepněte se do konzole (`Ctrl+Alt+F2`), ukončete běžící sezení KDE a vyvolejte konfigurační skript `xorgsetup`:

```
killall kdm
xorgsetup
```

Pokud ani ten neuspěje, samozřejmě můžete editovat soubor `/etc/X11/xorg.conf` ručně (editor `vi` je součástí výchozí instalace). Zvukovou kartu rozběháte pomocí nástroje `alsaconf` (spustíte v příkazovém řádku jako `root`), hlasitost výstupu upravíte pomocí `alsamixer` a zapíšete příkazem `alsactl store`.

Připojení přes DHCP zafungovalo naštěstí „out-of-the-box“, takže se nastavování sítě věnovat nebudu (v případě problémů nahlédněte do `/etc/networks` nebo zkuste nástroj `netconfig`). Kdo bude chtít tisknout, musí vědět, že ve výchozím nastavení je zablokován i CUPS a že je třeba jeho spuštění umožnit příkazem (jako `root`):

```
chmod +x /etc/rc.d/rc.cups
```

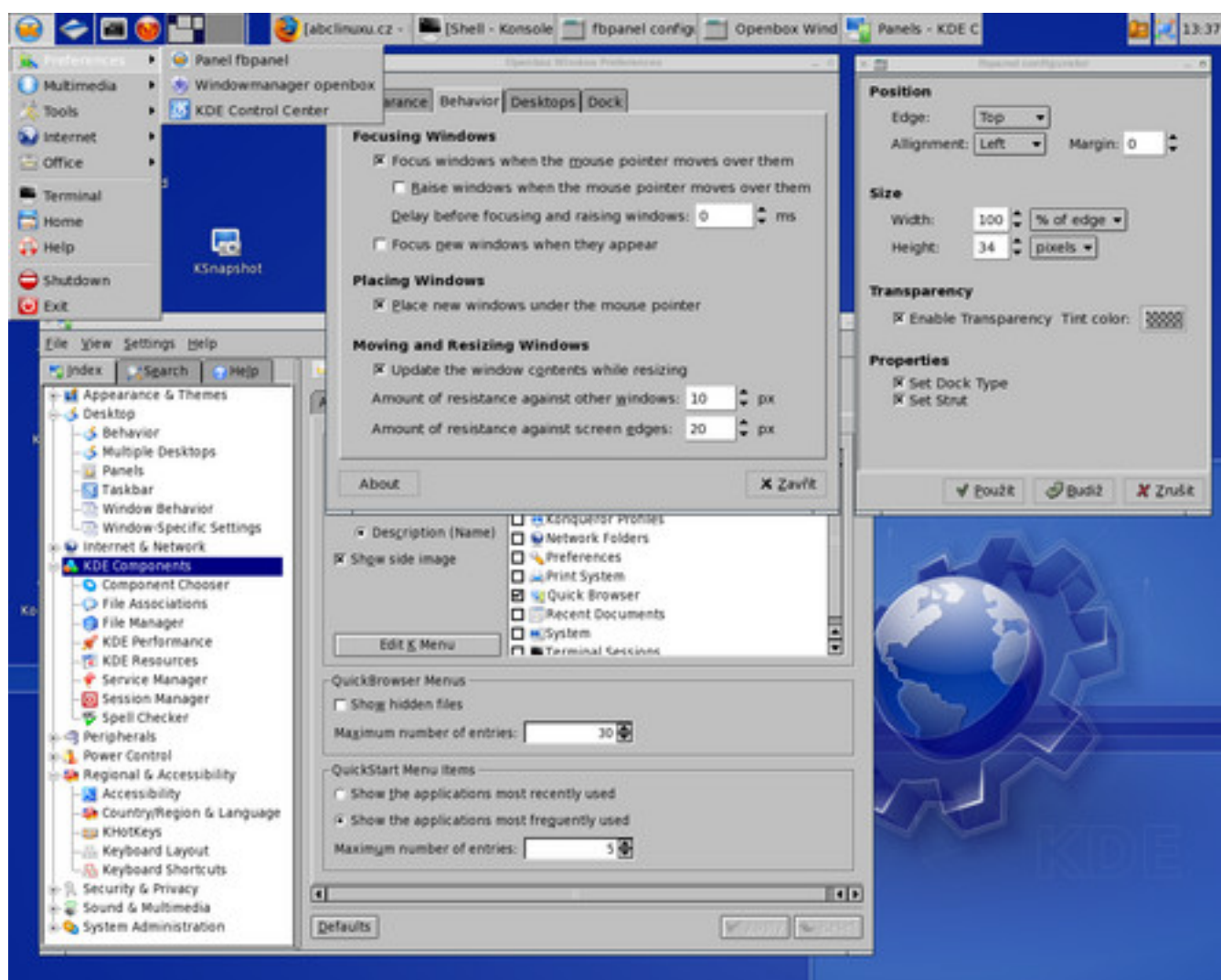
Předdefinován je běžný uživatelský účet s uživatelským jménem `gnu` a heslem `linux` (což ale se dozvíte pouze předem z dokumentace), tím se můžete přihlásit do objevivšího se přihlašovacího dialogu KDM. Další uživatelé se přidávají v Linuxu běžným způsobem (`adduser` v příkazové řádce – příslušná KDE klikací utilitka KUser nainstalována není).

Programové vybavení

Tím se také dostáváme k vybavení této „jednoduché“ distribuce, neboť právě tím se „pocketlinux“ má od jiných distribucí odlišovat. Po poněkud „hardcore“ instalaci (vzhledem k ostatním distribucím s nárokem na „snadné užívání“) budeme jistě zvědaví, jestli se naše námaha vyplatila a co naše „jednoduchá“ a „malá“ distribuce umí „out of the box“.

Jak už bylo stručně zmíněno, jako jediné výchozí desktopové prostředí slouží tzv. „KDE Light“. Pod tímto lákavým pojmenováním se v praxi skrývá skládanka z Kdesktopu (3.4.2, vykresluje pozadí a ikony), okenního manažeru `OpenBox` [10] (namísto obvyklého KWin) a nástrojového panelu `Fbpanel` [11] založeného na knihovně `gtk+` (náhrada aplikace Kicker).

Zatímco obě posledně jmenované minimalistické aplikace se nastavují v jednoduchých dialogích, přitomno je také obvyklé ovládací centrum KDE, a to včetně dialogů k nastavování aplikací, které neexistují (jako např. panelu Kicker). To není zrovna srozumitelný přístup pro nic zlého netušícího uživatele, který ať nastavuje, jak nastavuje, žádnou změnu neuhlídá. Dle mého skromného názoru může tato zrada daleko více frustrovat než „příliš mnoho“ možností nastavení Kickeru, která viditelně fungují.



„Pocketlinux“ se podle vlastní propagace snaží v zájmu jednoduchosti nabídnout jednu jedinou vybranou aplikaci pro určitý úkol, např. tedy v rámci desktopu jeden textový editor (KEdit), jeden textový procesor (KWord 1.4.2) a jeden internetový prohlížeč (Mozilla Firefox 1.5). Překvapivě, zdánlivější uživatelé brzy objeví, že přítomny jsou také jak editory KWrite a Kate, tak prohlížeč/souborový manažer Konqueror – ovšem z menu i z plochy dovedně ukryty tak, že se dají spustit pouze z příkazového řádku, a to ještě pouze z Konsole (neboť v rámci boje proti „rozptylování“ či z přehlédnutí byla odstraněna tradiční zkratka pro spouštění programů `Alt+F2`) i položka menu *Run...*.

Totéž platí i o ostatních KDE aplikacích: nemáte možnost na první pohled zjistit, které přítomny jsou a které ne. Zkušenější uživatelé tedy vyzkoušejí z Konsole povely, na které jsou zvyklí (a připraví se tak částečně o dobrodiní „lehkého“ KDE), nováčci si naopak bezděčně nainstalovali něco, co pravděpodobně nikdy neobjeví a tedy používat nebudou.

Z multimediálního vybavení nabízí menu PocketLinuxu videopřehrávač Xine (`xine-ui-99-3-cvs`, bohužel bez potřebných kodeků, takže první spontánní pokus přehrát DVD skončí chybovou hláškou), přehrávač hudby Xmms (1.2.10, bez podpory MP3), prohlížeč obrázků Gwenview (1.3.0) a vypalovací frontend K3B (chybí ovšem `dvd+rw-tools`).

Kdo nevypátrá ukryté Kopete (0.10.3), bude muset ve výchozí instalaci rezignovat na protokoly ICQ i Jabber, neboť autor PocketLinuxu rozhodl za vás, že IRC (Xchat) postačí. Kmail a FTP klient (gFTP) naopak nabídnuty jsou, stejně jako klient pro vzdálenou správu desktopu Krdc a formátovač disket Kfloppy. KOffice suite (Kword, Kspread, Kpresenter, Kivio) spolu s prohlížečem PDF formátu Kpdf už ale také seznam „připustných“ (tj. viditelně přístupných) aplikací končí.

Správa systému: instalace programů

Zdráhám se uvěřit, že výběr nabízených i (skrytě) nainstalovaných aplikací byl učiněn na základě hlubší analýzy jejich obecné potřebnosti a předpokládám, že uživatel této distribuce okamžitě zatouží po jejím doplnění. Protože se však v podstatě jedná o linuxovou distribuci Slackware, skýtá instalace dodatečného software oproti uživatelsky přívětivějším distribucím pro linuxové nováčky poněkud drsný háček.

K instalaci nového software je možné užívat nástroj `slackpkg`, který je předkonfigurován k instalaci přes internet. Použít lze i všechny kompatibilní balíčky z linuxpackages.net (ručně stáhnout a pak pomocí `slackpkg` nainstalovat). K tomuto účelu je dokonce implementováno kontextové menu do KDE pro ty, kteří neholdují příkazové řádce: v souborovém manažeru (Konqueror, spustitelný viditelně pouze přes ikonku „Home“ či „System“) můžete pro soubory s koncovkou `.tgz` vybrat nabídku *Actions – Slackware – Install* (vyžaduje heslo administrátora). Druhým nástrojem starajícím se o balíčky je textová aplikace `pkgtool`.

Tato zdánlivě snadná procedura instalace však nemile kontrastuje se známou skutečností (mezi zkušenými „linuxáky“), že tyto „slackwarovské“ nástroje tradičně neřeší za administrátora případné závislosti balíčků. Dosavadní uživatelé Windows mohou být nemile překvapeni, že stažené a tak snadno instalované binární balíčky nemají všechny potřebné knihovny přibaleny. Dosavadní uživatelé jiných distribucí Linuxu než Slackware zase tím, že se o o závislosti správce balíčků nepostará automaticky. Také Kdialog podávající o instalaci zprávu je poněkud strohý a nedomyšlený, např. nekončí zprávou o proběhlé instalaci, ale jakoby uprostřed instalace: posledním příkazem, který je vykonán, což zanechá uživatele na pochybách, zda už je hotovo, není hotovo...

Zdárně jsem si našel (na linuxpackages.net [12]), stáhl a úspěšně nainstaloval programy bez extra závislostí, jako je Midnight Commander nebo OpenOffice.org 2.0, a také bez problémů přidal na desktop ikonky utajených a pro mě potřebných aplikací, ale závislosti např. jiného mého oblíbeného programu `mocp` (podle chybové hlášky při spouštění konkrétně knihovny `libFLAC.so.7` a `libid3tag.so.0`) jsem v rámci testu distribuce coby „malé, lehké a snadné k používání“ zatím nedohledával.

Instalace nových programů v „pocketlinuxu“ tedy vyžaduje zkušeného administrátora (nejlépe „slackwaristu“); stejně jako další konfigurace: např. případné počestění distribuce ručně podle návodu pro Slackware a jednotlivé součásti grafického prostředí.

Několik slov závěrem

Mé první dojmy z této distribuce jsou smíšené. Zdá se, že pozorované klady (konfigurace v textových souborech, relativní svižnost grafického prostředí) i zápory (řešení závislosti balíčků) jsou podmíněny volbou základní distribuce Slackware. Z vlastního přínosu „pocketlinuxu“ pak zůstává ořezaný instalátor neumožňující individuální výběr instalovaných programů, ořezané KDE Light s ještě ořezanější nabídkou pouze některých fakticky nainstalovaných aplikací.

Sympatická může být snaha o „něco jiného“ v poněkud uniformním světě KDE a snad také o optimální předvýběr nejdůležitějších aplikací pro „běžného uživatele“, na čemž je ale podle mého názoru třeba ještě intenzivně zapracovat. Distribuci také možná ocení majitelé slabších strojů, i když musím uvést, že na dnešní poměry už dost slabá hardwarová konfigurace (PII 400 MHz 320 MB RAM) nepřekvapila oproti „plné“ verzi KDE např. v Gentoo nebo ArchLinuxu žádným citelným zrychlením (vyjma doby startu grafického prostředí samozřejmě). Naopak zmíněná omezení jsem coby „power user“ pociťoval citelně, až nelibě.

Shrnutí a podtrženo: „pocketlinux“ je v nynějším stádiu kvůli způsobu instalace a nutnosti ručního nastavování určen hlavně pro zkušené uživatele, jimž může posloužit třeba jako pozvánka do světa Slackware (od něhož se asi prakticky neliší). Naopak zkušení slackwaristi by mohli tuto distribuci možná použít jako základ pro další vlastní úpravy nějakých jednoúčelových uživatelských stanic

(informační pointy, internetová kavárna?), k čemuž sám autor nabízí stručný [popis postupu](#) [13] tvorby distribuce. Zatím jsem však nenašel dost přesvědčivých důvodů, proč v tomto případě nedat přednost „plnohodnotnému“ Slackware a vlastní konfiguraci desktopu na míru nebo třeba poměrně vyzrálému [Slaxu](#) [14].

Odkazy

- [1] <http://www.pocket-lnx.org/>
- [2] <http://www.linuxdevices.com/products/PD9917117472.html>
- [3] <http://gnulinux.de/pocketlinux/>
- [4] <http://damnsmalllinux.org/>
- [5] <http://slax.linux-live.org>
- [6] <http://www.gnulinux.de/pocketlinux/index.php/Main/WhyPocketLinux>
- [7] <http://www.zenwalk.org/>
- [8] <http://ultimapcs.dyndns.org/ultimalinux/about.html>
- [9] <http://www.gnulinux.de/pocketlinux/index.php/Main/Download>
- [10] <http://icculus.org/openbox/>
- [11] <http://fbpanel.sourceforge.net/>
- [12] <http://linuxpackages.net>
- [13] <http://www.gnulinux.de/pocketlinux/index.php/Main/MakingOfP1SlackwareDE>
- [14] <http://slax.linux-live.org>

FreeBSD v malej firme

Michal Kyseľ

Tento článok je úvodom do seriálu o tom, ako si postaviť funkčný firemný informačný systém na báze unixového systému FreeBSD.

Z technických dôvodů (velikost výsledného PDF souboru) zde nejsou uvedeny zdaleka všechny obrázky, které autor použil. Pozn. editora.

Tento článok je úvodom do seriálu o tom, ako si postaviť funkčný firemný informačný systém na báze unixového systému FreeBSD. Popisované riešenie bolo realizované vo firme *PROKOM, s.r.o. Sečovce* (naša firma sa zaoberá kovovýrobou).

Small Business systémy

V dnešnej dobe sa čoraz viac stretávame s požiadavkou na vytvorenie firemného informačného systému pre malú firmu. Vo svete sa stále viac udomácňujú takéto riešenia, predtým bežné len u veľkých korporácií a nadštandardne solventných firiem, aj v sektore drobného a stredného podnikania. Mnoho firiem má niekoľko separátnych počítačov, mnohé sú už technicky zastaralé, a používa ich iba ako „písacie stroje“. Pritom nároky na spracovanie informácií stále narastajú a vedieť rýchlo získať požadované informácie, efektívne komunikovať s obchodnými partnermi a spracovať informácie, či už z vnútrofirmerného prostredia alebo z vonkajšieho sveta, sa dnes často rovná pojmu byť konkurencieschopný. V zásade sa vždy jedná o systém, ktorý splňuje tieto základné (a niekedy aj iné) požiadavky:

- Prístup k internetu a mailu.
- Prevádzka kancelárskych aplikácií.
- Intranet.
- Databáza.
- Centrálné úložisko dát, možnosť zdieľania dát a kooperácie medzi užívateľmi.
- Centrálna záloha.
- Centrálna tlačové služby.
- Aplikačný server.
- Odolnosť proti zlyhaniu, havarijným stavom a zámerným poškodeniam dát.
- Jednoduchá obsluha systému.
- Ekonomická nenáročnosť, možnosť plnohodnotného využitia aj staršej techniky.

Existujú pochopiteľne komerčné riešenia, ktoré dokážu bez problémov zabezpečiť požadovanú funkcionálnosť. Sú však drahé a niektoré dokonca aj viazané na špecifický hardware. Navyše väčšinou obchádzajú problém spolupráce viacerých platforiem tak, že jednoducho preferujú tú vlastnú a ostatné nepodporujú, prípadne podporujú za príplatok, alebo podporujú iba čiastočne. Mnohé vyžadujú aj enormne veľkú investíciu do komunikačnej infraštruktúry, niektoré majú zase podivnú licenčnú politiku... skrátka, treba platiť, platiť a ešte raz platiť, a až potom je k dispozícii použiteľné riešenie. Práve táto finančná náročnosť je problémom pri nasadzovaní týchto systémov do firiem. Pritom riešenie je veľmi jednoduché a dostupné pre každú firmu. Základom pre každý firemný informačný systém je výber vhodnej serverovej platformy. Najviac sú na to vhodné unixové systémy, asi najpoužívanejšie sú rôzne distribúcie Linuxu. Nie je to však jediná cesta vedúca k úspechu riešenia, sú aj iné rovnako použiteľné platformy. V tomto seriáli budem opisovať systém postavený na FreeBSD.

Čo je to FreeBSD?

Je to slobodný systém unixového typu. Nie je to však linuxová distribúcia, ale priamy nasledovník pôvodného Unixu od AT&T. Vznikol na Kalifornskej univerzite v Berkeley odštiepením od spomínaného komerčného systému. Jeho základné vlastnosti v skratke:

- Nemá runlevely, na rozdiel od Linuxu. Preto je jeho štruktúra konfiguračných súborov jednoduchšia, nie je treba konfigurovať služby pre každý runlevel zvlášť.
- Podporuje iba jeden druh filesystemu – UFS (Unix File System).
- Partície a zariadenia sa označujú ináč ako u Linuxu, a to podľa starých unixových konvencií z pred – PC éry – napríklad disky sú ad4, ad6, cd je acd0 atď. . .
- Jadro a userland tvoria neoddeliteľnú časť celku, keď sa mení jadro za novšie, musí sa vymeniť aj userland za nový zodpovedajúci verziou jadra.
- Je iba jediná distribúcia – tá existuje vo vetvách STABLE a CURRENT. Výhodou je jednoduchšia a konzistentnejšia dokumentácia.
- Upgrade systému je riešený rekompiláciou jadra a userlandu zo zdrojového kódu – toto po ňom prevzali dve linuxové distribúcie - Gentoo a LFS. U FreeBSD je to primárny, rokmi overený fungujúci systém updatu. Výhodou je, že nový systém možno skompilovať presne na mieru železu – presne to je hlavný tromf Gentoo Linuxu, ktorý bol prevzatý práve z BSD.
- Inštalácia aplikácií sa rieši dvoma spôsobmi – buď cez zbalené binárky (packages) alebo cez porty. To sú laicky povedané vopred pripravené makefile súbory, po ktorých spustení systém stiahne všetky potrebné zdrojáky a automaticky zostaví aplikáciu. Opäť je to črta, ktorú prevzalo Gentoo v podobe Portage systému.
- Aktuálny zdrojový kód systému a strom portov sa udržujú pomocou utility CVSUP. Updaty sú inkrementálne, t.j. menia sa iba tie vetvy zdrojáku a stromu portov, u ktorých nastali zmeny.
- Výborné sieťové a bezpečnostné parametre.
- Vynikajúci multitasking.

Nechcem to pitvať do hĺbky, podrobný popis je mimo rozsah tohto článku, viac je na stránke <http://www.freebsd.org> [1].

Prečo do firmy zvoliť práve FreeBSD?

Na prvý pohľad to vyzerá ako trochu príliš odvážna voľba. O systéme FreeBSD chodia po webe strašidelné legendy o tom, aký je to ťažko spravovateľný, komplikovaný, predpotopný, zle ovládateľný a užívateľsky neprítulný systém, vhodný pre elitárov, snobov, hackerov a maniakov. Úlohou tohto seriálu je vyvrátiť tieto zlomyseľné klebety a ukázať, že BSD je rovnako dobrá alternatíva pre firemný systém, ako ktorékoľvek distro Linuxu. Tieto dezinformácie často vznikajú z mylnej snahy chápať FreeBSD ako ďalšiu linuxovú distribúciu a následného sklamanie z toho, že sa to „nespráva štandardne“.

Ostatne tento problém je známy aj u linuxových začiatočníkov, ktorí zmažú Windows z disku, nahodia Linux (o ktorom im rovnako „znalí“ kamaráti hovoria, že je ako Windows, ale zadarmo a stokrát lepší) a potom sa divia, prečo to nefunguje rovnako ako predtým. Proste nepochopia, že Linux nie je a nikdy nebude Windows. Takisto ani BSD systémy nie sú a nikdy nebudú Linuxom. Sú iné, svojské, ale pritom elegantné.

Ak by som mal FreeBSD prirovnať k nejakému systému z Linux rodiny, tak je to taký kríženec medzi Slackwarom, Debianom a Gentoo. Inštaluje sa podobne ako Slackware, inštalácia aplikačného softwaru a update systému sa robí podobným spôsobom ako u Gentoo a ponúka podobnú stabilitu, robustnosť a škálovateľnosť ako Debian. Nespornou výhodou je aj to, že konfiguračných súborov je podstatne menej a preto sa, ak si človek zvykne na jeho štruktúru, ktorá je trochu iná ako u Linuxu, podstatne ľahšie orientuje v systémových nastaveniach.

Teda FreeBSD nie je pre usera ani admina ťažší ako Linux, nie je komplikovanejší ani problematicky spravovateľný. Je proste iný a má trochu inú filozofiu ako Linux. Napriek podpore moderných features je to oproti Linuxu konzervatívnejšie poňatý „klasický“ Unix držiaci sa tradícií. To môže znamenať výhodu aj nevýhodu, závisí od požiadaviek a vkusu užívateľa.

Vo firemnom prostredí sa však naplno prejavia jeho výborné vlastnosti. Typickým príkladom je inštalácia nového softwaru cez systém portov. Keď procesor kompiluje, aktuálna utilizácia CPU je *99.9 percent* a napriek tomu užívatelia robia cez terminály naplno *bez rušivých spomalení* a mrznutia okien a otvorených programov. Ak im nepoviem, ani nevedia, že stroj je totálne zahľtený a robí na doraz. To svedčí o perfektne napísanom multitaskingu, ktorý optimálne využíva prednosti použitého HW.

Toto je hlavný tromf FreeBSD – *výborné parametre v záťažovom nasadení*. Nie nadarmo ho používa Google a Walnut Creek, a svojho času na ňom bežali aj servery Microsoftu. Skúste si, ako sa tvári taký Win, keď sa na ňom spustí nejaká náročnejšia aplikácia a záťaž CPU presiahne polovicu. . .

Podstatná je aj jeho ďalšia vlastnosť, ktorú som už spomenul, a to jednotnosť prostredia a ovládania s tradičnými Unixami, daná konzervatívnejším dizajnom systému a tým, že tento systém je priamy potomok pôvodného Unixu, nie jeho nová implementácia. V systéme je poznať veľmi silný dôraz na unixovú tradíciu. Trebárs *vi* je skutočne starý editor *vi*, nie symlink na *vim*, užívateľ sa maže klasickým unixovým príkazom *rmuser*, nie príkazom *userdel* či *deluser* (v každom linuxovom distre je to ináč) a podobné detaily. Aj použitý súborový systém je klasika – *ufs*, teda Unix File System, teraz už vo verzii 2.

Proste stará škola Unixu, pamätníci čias sálových počítačov zaslzia dojatím. . . ale pritom je tento „dinosaur“ dostatočne moderne vybavený na to, aby obstál v dnešných požiadavkách. Práve toto – spojenie „retro dizajnu“ a moderných technológií – ma nadchlo pri výbere systému. Pre domáceho užívateľa sú tieto malé odchýlky samozrejme drobné nepodstatnosti, ktoré nijak neznižujú kvality a použiteľnosť ich obľúbeného OS, či už je to Linux alebo BSD (a niekomu tieto odlišnosti BSD oproti Linuxu môžu dokonca aj prekážať). Ale vo firemnej prevádzke profík ocení, že sa každá mašina spravuje aj updatuje úplne rovnako, a že tam nájde takmer presne to, čo pri práci s inými komerčnými Unixami. Napríklad taký bývalý SMEP-kár sa veľmi ľahko adaptuje na BSD.

Výber aplikácií je do značnej miery zhodný s tými pre Linux, samozrejme vďaka Linux kompatibilite je možné väčšinu linuxových programov spúšťať na FreeBSD. Pre domáceho bežného usera by som však asi viac doporučil Linux, kvôli lepšej podpore multimédií a väčšiemu počtu užívateľov tohto systému, ľahšie sa dá nájsť riešenie prípadného problému. Nie je však pravdou, ako sa traduje, že FreeBSD je systém vhodný iba pre serverové nasadenie. Ako si ukážeme, dá sa urobiť veľmi pekné riešenie plnohodnotného kancelárskeho desktopu na báze FreeBSD, ktorý komfortom a použiteľnosťou nijak nezaostáva za linuxovým.

Použitie – prípadová štúdia

V našej firme sa tento systém použil na server aj klientske terminálové stanice. Použil sa systém tenkých klientov a výkonného servera, čo umožnilo použiť pre plnohodnotnú prácu v grafike v reálnom čase aj staré PC, ktorých výkon by stačil možno tak na písanie vo Worde 6 pod Win95. . . výkonnejšie Win stanice zase využívajú diskovú kapacitu servera ako úložisko dát. Na systéme beží aj SQL server MySQL a Apache web server s rozšírením pre PHP4. Fungujú tlačové služby pre Win stanice aj pre tenkých terminálových klientov. Server má diskové pole RAID1, denné zálohy sú realizované zápisom na prepisovateľné DVD.

V nasledujúcich dieloch seriálu si popíšeme krok za krokom, ako si takýto systém čo najjednoduchšie zostaviť a nakonfigurovať. Každý diel bude venovaný inej časti problematiky. Budem sa zaoberať napríklad inštaláciou, základnou konfiguráciou, lokalizáciou, nastavením siete a systému X-window, výberom a inštaláciou vhodných kancelárskych aplikácií, rozobraná bude aj SAMBA, CUPS, inštalácia APACHE a PHP. Zatiaľ sa majte a nech vám vaše systémy idú.

Získanie inštaláčiek

Prvou vecou je získanie inštaláčnych médií. Najjednoduchšie je si ich stiahnuť z [webovej stránky FreeBSD](#) [2], kde sa nachádza link na podstránku s ftp serverami, odkiaľ je možné ťahať *ISO* obrazy inštaláčnych médií. Vyberte si najnovšiu Release verziu pre platformu použitého počítača. Sú tam vystavené inštaláčky pre rôzne druhy platforiem (od alpha po sparc64), pre bežné PC je potrebné si stiahnuť *i386* verziu, pre 64-bitový Athlon *amd64*. Inštaláčne CD sú iba dve, čo je pre ľudí zvyknutých na distribúcie na niekoľkých CD či dokonca na DVD nezvyklé. Vzhľadom na trochu iný defaultný spôsob inštalácie aplikácií oproti väčšine linuxových distribúcií (Gentoo a LFS nerátam) je to však pochopiteľné, dokonca keď inštalujem iba základný systém, nie je mi treba ani druhé CD, stačí mi iba prvé.

Po stiahnutí si vypálime z ISO inštaláčne CD. Tu by som doporučoval použiť ako médium CD-R, nie CD-RW, mnoho starších CD mechaník má problém z CD-RW bootovať. Nastavíme BIOS tak, aby bootoval z CD. Nabootujeme si počítač a počkáme si na spustenie základného systému. Necháme, nech spúšťacie menu spustí defaultný systém pod voľbou 1.

Inštalácia začína

Privíta nás menu programu *sysinstall*, čo je základná inštaláčna a konfiguračná utilita FreeBSD. Tu je možné si nastaviť klávesovú mapu a niektoré systémové voľby, ale samozrejme je to možné riešiť aj po inštalácii, preto sa tým nebudeme zdržiavať. Vyberieme si *Standard instalation*. Spustí sa séria otázok inštalátora, nie nepodobná inštalátoru Slackware Linuxu. Na začiatku sa inštalátor spýta, na ktorý disk má umiestniť primárnu partíciu, keďže FreeBSD musí byť nainštalovaný na primárnej partícii. Pohyb v menu *sysinstallu* je jednoduchý, cez kurzorové klávesy a Tab, voľba sa zaškrta s medzerníkom.

Disky sú, ako som [spomínal](#) [3], označované staršou terminológiou z pred-PC éry, takže netreba byť prekvapený, ak tu človek nenájde žiadne *hda* ani nič podobné. Názvy diskov začínajú na *a*, napríklad *ad0* – u mňa na serveri sú konkrétne *ad4* a *ad6*. Tu FreeBSD rozhodne nezaprie, že je priamym potomkom Berkeley Unixu od AT&T... Pripájam obrázok z oficiálnej dokumentácie FreeBSD.

Pristúpime k vytvoreniu partície, v BSD systémoch nazývanej *slice* (opäť reminiscencia na staré časy sálových počítačov). Tú má v FreeBSD inštalátore na starosti klasická utilita *fdisk*.

Všetky jej funkcie sú dostupné cez klávesové skratky, v spodnej časti obrazovky dostatočne kvalitne komentované, takže nie je to nič komplikované. S *fdiskom* sa už stretol snáď každý, takže tu nie je veľa čo riešiť. Keďže dualboot tu rozoberať nebudem, rozoberám tu čistú inštaláciu od začiatku, najrýchlejšia cesta je zmazať všetko a použiť celý disk cez voľbu *A*. Prácu s *fdiskom* ukončíme cez skratku *Q*. Viac povie tento obrázok:

```

Disk name:      ad0                               FDISK Partition Editor
DISK Geometry: 16383 cyls/16 heads/63 sectors = 16514064 sectors (8063MB)

Offset      Size(ST)      End      Name  PType      Desc  Subtype  Flags
-----
0           63             62      -     6         unused   0
63      16514001    16514063  ad0s1  3         freebsd  165     CA

The following commands are supported (in upper or lower case):
A = Use Entire Disk      G = set Drive Geometry  C = Create Slice      F = `DD' mode
D = Delete Slice        Z = Toggle Size Units   S = Set Bootable     I = Wizard m.
T = Change Type         U = Undo All Changes    Q = Finish

Use F1 or ? to get more help, arrow keys to select.

```

Po úspešnom vytvorení *slice* treba zapísať *boot manager*, čo je niečo podobné ako z linuxových distribúcií známe *LILO* alebo *GRUB*. Úlohou tohto programu je zaviesť systém pri štarte. Ak nemáme

v stroji iný systém ako FreeBSD, je treba zadať voľbu *Standard*. Ak máme systémov viac a chceme používať dual-boot, vyberieme *BootMgr*. Ak máme nejaký iný boot manager od „tretej strany“, dáme *None*.

A samozrejme si môžeme vybrať aj to, na ktorý disk príde boot partícia.

Ďalšou časťou inštalácie je vytvorenie logických oddielov na fyzickej partícii (tu sa nazývajú *disklabel*). Klasicky sa jedná o oddiely pre */*, */var*, */tmp* a *swap*. O ich vytvorenie sa stará *FreeBSD Disklabel Editor*. Pre začiatočníka je ideálne nechať systém rozdeliť si disk automaticky, cez voľbu *A*.

To by malo na aktuálnom Slice vytvoriť takúto štruktúru:

```

FreeBSD Disklabel Editor
Disk: ad0 Partition name: ad0s1 Free: 0 blocks (0MB)
Part      Mount      Size Newfs  Part      Mount      Size Newfs
-----
ad0s1a    /           128MB UFS     Y
ad0s1b    swap        503MB SWAP
ad0s1e    /var        256MB UFS+S Y
ad0s1f    /tmp        256MB UFS+S Y
ad0s1g    /usr        6919MB UFS+S Y

The following commands are valid here (upper or lower case):
C = Create      D = Delete     M = Mount pt.  W = Write
N = Newfs Opts  Q = Finish    S = Toggle SoftUpdates
T = Toggle Newfs U = Undo      A = Auto Defaults  R = Delete+Merge

Use F1 or ? to get more help, arrow keys to select.

```

Opäť ukončíme zápisom cez skratku *Q*. To by sme už mali mať rozdelenie disku úspešne za sebou. Okrem tohto v praxi väčšinou postačujúceho spôsobu je aj možnosť vytvoriť a nastaviť disklabley ručne. To je však už pre špecifické prípady, záujemca postup nájde vo FreeBSD príručke, z ktorej čerpám aj tieto screenshoty.

Po rozdelení disku sa opäť dostaneme do sysinstallu a bude nám ponúknutý výber typických druhov inštalácie. Pre firemný server sa najviac hodí *X-Kernel developer*, ktorý ku klasickému *X-User* setu pribalí všetky vývojové nástroje a kompletne zdrojáky systému. To sa nám, ako ukážem v ďalšom dieli, bude neskôr náramne hodiť. Pre pracovnú stanicu alebo *X-terminál* úplne stačí *X-user*. Voľbu v sysinstalle vyberieme kurzorom hore-dole, zaškrtneme cez medzedník a cez *Exit* sa vrátíme do inštalátora.

Ešte by sa mal systém pýtať, čo všetko treba do X-systému nainštalovať. Doporučujem do X-iek zvoliť *všetko, čo sa dá*, nie je toho až tak extrémne mnoho a človek sa potom nemusí starať, že mu v systéme trebárs chýba nejaký font a nezobrazuje mu kvôli tomu korektné diakritiku. Po tomto úkone sa systém spýta, či chceme aj *kolekciu portov*. Doporučujem zvoliť *Yes*, bez portov sa v budúcnosti príliš dobre nezaobídeme. Praktický zmysel zadania *Novidím* iba v prípade vytvorenia nejakej minimálnej inštalácie systému pre špecializované použitie, kde treba šetriť každý megabajt.

Teraz sa dostávame k bodu, kde inštalátor chce, aby sme mu určili zdroj inštalačných dát. Je to možné, ako je vidno, z rôznych zdrojov, v našom prípade vyberieme *CD*.

Po tomto sa ešte inštalátor spýta, či definitívne súhlasíme s tým, čo sme mu vybrali. Ak potvrdíme *Yes*, rozbehne sa inštalácia. Zapiše sa súborový systém, sformátuje sa a spustí sa kopírovanie súborov. Pokiaľ všetko dopadne dobre, FreeBSD sa nainštaluje a upozorní nás na to hláškou. Po inštalácii je však nutné urobiť ešte zopár zásahov, ktoré nám systém ponúkne.

Poinštalčné zásahy

Sem spadajú rôzne nastavenia HW a služieb. Ako prvá sa ponúkne konfigurácia sieťovej karty. Tá je vedená pod názvom zodpovedajúcim jej typu, napríklad Novell Ethernet 2000 kompatibilná karta sa hlási ako *ed0*. Karty s čipom Realtek sú zase vedené ako *rl0*.

Po odklepnutí příslušnej sieťovky sa objaví hláška, či chcem zapnúť podporu *ipv6*. *Pozor! Silne doporučujem zapnúť!* Pri jej vypnutí totiž nefunguje *xmcp* protokol, teda nedá sa pripojiť na X klienta zo vzdialeného stroja. Po povolení *ipv6* sa objaví jednoduchá forma, kde sa dajú napísať TCP/IP parametre pre sieťovku.

Podrobnosti sa dajú nájsť v handboku, ale predpokladám, že s týmto nebude mať nikto problém. Po tomto úkone sa FreeBSD spýta, či chcem nastavenia, ktoré som urobil, potvrdiť a spustiť sieťovku s týmito parametrami. Pochopiteľne zvolíme *Yes*. Potom nasleduje konfigurácie sieťových služieb. Tam možno pokojne povoliť iba *inetd* a dať všade inde *No*. To, čo budem neskôr potrebovať, si spustím ručným editovaním súboru `inetd.conf`.

Nasleduje nastavenie klávesnice, časovej zóny, myši a podobných vecí. Dôležitá vec – systém by sa mal spýtať, či chceme dať inštalovať balíčky. Doporučujem dať, pokiaľ je to možné, nainštalovať KDE, ak je málo miesta, tak IceWM. Túto voľbu možno aj preskočiť – balíčky sa dajú doinštalovať kedykoľvek neskôr, prípadne sa dá software aj priamo skompilovať z portu. Ale kompiláciu takých obrov ako KDE či OpenOffice.org, najmä pre majiteľov slabších strojov, striktne nedoporučujem – môže sa natiahnuť na dlhé hodiny.

Po vybratí sa má ukázať zoznam balíčkov, ktoré sme vybrali, na schválenie, bude to vyzerať asi takto: Počas inštalácie balíčkov si systém môže vypýtať aj CD2, keďže napr. KDE sa na prvé CD nevošlo. Po inštalovaní vybraných balíčkov silne doporučujem *potvrdiť* aj voľbu inštalovania podpory pre beh linuxových programov – *linux base compatibility layer*. Umožňuje na systéme spúšťať linuxové binárky, navyše niektoré porty (prevzaté z linuxových aplikácií) bez nej nejdú – napríklad Adobe Reader.

Finále

Úplne na koniec sa nastavujú používatelia a heslo roota. Zakladanie userov možno zatiaľ preskočiť, urobia sa hocikedy inokedy, *heslo roota však zadať rozhodne treba!* Po tomto poslednom kroku sa *sysinstall* vráti do základného okna a je možné ukončiť inštaláciu.

Systém vás ešte vyzve, aby ste vybrali inštaláciu z mechaniky, a po vybratí a potvrdení by mal re-bootovať. Ak všetko prebehlo správne, mala by vás po výpise rozpoznaného HW a hláškach ohľadne spustenia služieb privítať konzola s prihlasovacím dialógom. To by malo byť ohľadne základnej inštalácie všetko. Záverom chcem ešte raz podotknúť, že *všetky obrázky* použité v tomto návode sú prebraté z *FreeBSD Handbooku* ([licence](#) [4]).

The FreeBSD Documentation License

Copyright 1994-2005 The FreeBSD Project. All rights reserved. Redistribution and use in source (SGML DocBook) and 'compiled' forms (SGML, HTML, PDF, PostScript, RTF and so forth) with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code (SGML DocBook) must retain the above copyright notice, this list of conditions and the following disclaimer as the first lines of this file unmodified.
2. Redistributions in compiled form (transformed to other DTDs, converted to PDF, PostScript, RTF and other formats) must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS DOCUMENTATION IS PROVIDED BY THE FREEBSD DOCUMENTATION PROJECT "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE FREEBSD DOCUMENTATION PROJECT BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR

PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS DOCUMENTATION, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

RAID – čo to je, ako to funguje a prečo sa tým zaoberať

Pre firemné účely je nesmierne dôležité, aby boli dáta uložené bezpečne a spoľahlivo. Nesmie nikdy dôjsť ku strate alebo k poškodeniu dát z dôvodu fyzickej poruchy disku. Takáto porucha by mala pre firmu samozrejme veľmi tvrdé nepríjemné dôsledky, preto je dobré na túto eventualitu myslieť už pri samotnom návrhu a zostavení systému a za každú cenu sa jej snažiť vyhnúť.

Pre zamedzenie havarijných situácií spojených s poruchou diskov a následnou stratou dát sa používajú dve základné stratégie. Jednou z nich je klasická záloha dát na iný nosič. Bez zálohy sa nedá hovoriť o spoľahlivom uložení dát. Záloha však nerieši jednu dôležitú vec, a to je funkčnosť systému aj po výskyte havarijnej udalosti. Predstavme si napríklad server, na ktorom sú údaje na jedinom disku. Tento disk náhodne zhavaruje (napríklad prehriatím, skrytou vadou, prirodzeným opotrebením, ...) a stane sa nečitateľný. Obsah disku bol pravidelne odzálohovaný na nejaké záložné médium (páska, DVD-RW, vymeniteľný disk ...). Firemné dáta sa teda nestretia. Systém je však po poruche nefunkčný, je nutné v ňom vymeniť disk, odznova nainštalovať systém, skonfigurovať ho a následne obnoviť dáta. To samozrejme nejaký čas trvá. A počas tohto času firma nemôže pracovať so svojimi dátami, čo pre ňu znamená rôzne problémy a komplikácie, dokonca aj finančnú stratu.

O veľa lepšie je mať systém, ktorý aj po výpadku disku je schopný ďalšej práce a minimalizuje potrebu prestojov v prevádzke. Táto funkcionálna sa dosahuje pridaním viacerých diskov a redundantným (laicky povedané, „nadbytočným“) uložením dát. To značí, že dáta nie sú fyzicky uložené iba raz, ale viackrát. Od toho vznikol pojem RAID (Redundant Array of Independent Disks) – redundantné pole nezávislých diskov.

RAID má niekoľko foriem (RAID 0, RAID 1 atď ...), tie tu nebudem bližšie rozoberať. V našom prípade sa zameriame na RAID 1, teda mirroring. Znamená to, že dáta sú uložené na dvoch (alebo teoreticky viacerých) diskoch, pričom na každom z nich je samostatná kópia súborového systému. Diskové pole sa pritom systému javí ako jeden fyzický disk. V prípade výpadku ktoréhokoľvek disku je k dispozícii minimálne jedna alebo viac kópií filesystému a k zlyhaniu servra nedôjde, je možné na diskové pole zapisovať aj čítať z neho. Zápis trvá približne rovnako dlho ako na jeden fyzický disk, ale čítanie je v prípade RAID 1 s 2 diskami cca 2-krát rýchlejšie! To je pri výrazne zvýšenej odolnosti voči zlyhaniu ďalší výrazný argument v prospech nasadenia diskového poľa v malom firemnom riešení IS.

Vo väčšine prípadov sa používajú pre RAID 1, teda mirroring, iba dva disky, pri diskovom poli s viacerými diskami je tento spôsob ukladania už neefektívny a používajú sa iné spôsoby uloženia (napr. RAID 5), ktoré umožňujú pri zväčšení počtu diskov zväčšovať kapacitu diskového poľa pri rovnakej odolnosti proti výpadku dát. Pre server small business systému je preto RAID 1 ideálnym kompromisom, ako získať za rozumnú cenu rozumne vysokú mieru spoľahlivosti.

Diskové polia môžu byť hardwarové a softwarové. Hardwarové diskové polia nepotrebujú osobitný software na riadenie, majú vlastný procesor a firmware. Ich nevýhodou sú rozmery a samozrejme aj značne vysoká cena. Softwarové RAID riešenia sú postavené na emulácii RAID jednotky prostriedkami systému, bez použitia špeciálneho HW. Výhodou je možnosť implementácie RAID bez špeciálneho hardwaru, nevýhodou je, že je to vlastne užívateľská aplikácia a tým pádom jednak zafažuje CPU a po druhé za istých podmienok môžu nastať problémy so stabilitou (ako u každej aplikácie). Taktiež niektoré riešenia majú problém s bootovaním z SW RAIDu (najskôr sa musí zaviesť systém a príslušný program zabezpečujúci RAID funkciu, až potom je možné RAID používať) a musia potom základ systému alebo aspoň zavádzaciu oblasť držať mimo poľa na samostatnom disku. Zvláštnu pozornosť si zasluhujú „hybridné“ radiče SATA RAID, ktorých I/O operácie sú v skutočnosti riadené procesorom počítača, ale ostatné časti sú fyzicky nezávislé a samostatné. Tvoria tak akýsi

medziprechod medzi „pravými hardwarovými“ radičmi a čisto softwarovými riešeniami. Najčastejšie sa jedná o rôzne integrované radiče na matičnej doske (typickým príkladom je Intel ICH6). Obvykle sa takéto periférne zariadenie správa ako RAID na úrovni BIOSu, ale v skutočnosti je jeho chipset po zavedení OS ovládaný CPU. Spájajú výhody HW RAIDu (bezproblémový boot z RAIDu, vysoká stabilita, nízka záťaž CPU ...) s nízkou cenou. To robí tieto riešenia veľmi lákavými pre small business aplikácie. Práve toto riešenie som použil na vytvorenie diskového mirroru.

Ako ho realizovať vo FreeBSD

Pre server som použil matičnú dosku s integrovaným SATA RAID radičom Intel ICH6 a dva 80 GB disky Seagate. Postup spojzdenia je jednoduchý a logický. Prvým krokom je inštalovať holý minimálny systém bez balíčkov, grafiky, služieb, nie je nutné rozbehnúť ani sieť (spomínate si, hovoril som predtým niečo o tom, že minimálny install sa v istých prípadoch môže hodiť, a toto je jeden z nich). Inštalátor by mal vidieť oba disky, u mňa konkrétne sa hlásili ako *ad4* a *ad6*. Systém môžeme inštalovať na hociktorý z nich, ja som ho dal na *ad4*. Po boote do čerstvo inštalovaného systému si necháme vypísať `dmesg | grep ata` a vo výpise by sme mali vidieť rozpoznaný radič a disky, k tomu aj číslo zbernice ATA (to je tá *ata2* a *ata3*).

```
ad4: 76319MB <ST380817AS/3.42> [155061/16/63] at
ata2-master SATA150
ad6: 76319MB <ST380817AS/3.42> [155061/16/63] at ata3-master SATA150
```

Nasledujúcim krokom je zapnutie RAIDu. To sa urobí pod rootom príkazom

```
atacontrol create RAID1 ad4 ad6
```

Po tomto kroku rebootujeme systém a znova spustíme inštaláciu. Tentokrát nám inštalátor ponúkne tri diskové zariadenia – ku *ad4* a *ad6* pribudne aj zariadenie *ar0* (to je novovytvorené pole RAID). Urobíme inštaláciu systému, tentokrát „ostrú“, na *ar0*, podľa návodu v minulom dieli. Na *ar0* z pochopiteľných dôvodov treba dať aj bootovaciu oblasť. Po inštalácii a základnej poinštaláčnej konfigurácii rebootujeme. A to je všetko. Máme bežiaci systém s RAID 1. Jednoduché, že?

Systém by už mal štartovať z RAIDu a používať ho na zápis a čítanie. Presvedčíme sa o tom príkazom `df -H`, ktorý by mal vrátiť tabuľku zaplnenia súborových systémov. Mali by v nej byť uvedené oddiely z *ar0*, nie z *ad4* či *ad6*. Stav RAIDU si overíme príkazom `atacontrol status ar0`, pochopiteľne tiež pod rootom. Mal by vracieť výpis diskov zaradených do RAIDu a taktiež stav poľa, v štandardnom prípade je to *READY*. Príklad: `ar0: ATA RAID1 subdisks: ad4 ad6 status: READY`

Čo robiť v prípade problémov?

V prvom rade si zachovať chladnú hlavu. Pri zlyhaní jedného disku z dvoch sa nič extra nestane, systém pobeží ďalej na jednom s degradovaným poľom. Treba ho však dostať opäť do plne funkčného stavu, pretože v prípade ďalšej poruchy nás už nezachráni od pádu a straty dát nič. Postup je nasledovný:

1. Prečítame si man stránky k príkazu `atacontrol`.
2. Vypíšeme si stav RAIDu cez `atacontrol status ar0`
3. Odpojíme si chybný disk príkazom `atacontrol detach 2`, kde 2 je číslo ATA zbernice. Tento príkaz odpojí *ad4*. Samozrejme ak chceme odpojiť *ad6*, miesto 2 dáme 3.
4. Vymeníme chybný disk za nový so zodpovedajúcou alebo väčšou kapacitou.
5. Pripojíme nový disk do ATA zbernice príkazom `atacontrol attach 2`
6. Pripojíme disk logicky do poľa `atacontrol addspare ar0 ad4`

7. Spustíme synchronizáciu RAIDu, počas ktorej sa na novom disku vytvorí kópia dát udržiavaných poľom `atacontrol rebuild ar0`

Po tomto by sa malo diskové pole dostať do pôvodného stavu. Oba postupy (inštalácia RAIDu aj postup rekonštrukcie) som získal cez Google. V prípade nejasností vrele doporučujem si príslušné veci dohľadať na webe, určite tam niekde budú.

Xserver – základná konfigurácia

Každý small business systém by mal byť užívateľsky prívetivý, intuitívne ovládateľný a prehľadný aj pre bežného užívateľa s priemernými či dokonca podpriemernými znalosťami z obsluhy počítačov. Isteže, je veľa ľudí, ktorí uprednostňujú prácu v príkazovom riadku a úplne zavrhnú grafické rozhranie. Ale títo ľudia sú aj medzi skúsenými užívateľmi veľmi zriedkaví (dnes aj unixový správca, ak kráča s dobou, má rád komfort a intuitívny spôsob práce, čo si tu budeme navrávať, napríklad ja to mám hodené tak pol na pol, niektoré veci sa ľahšie a efektívnejšie robia cez X-ká a niektoré cez príkazový riadok), navyše typický užívateľ small business systému medzi týchto ľudí *rozhodne nepatrí*, pre návrh takéhoto systému je grafika nie spríjemnením práce či nejakou funkciou navyše, ale *holá nutnosť a základná podmienka použiteľnosti*. Ťažko možno chcieť od dielenského majstra alebo sekretárky, aby písala do komandlajny príkazy, to snáď pochopí aj najväčší odporca grafického prostredia. X systém je skrátka pre small business systémy zásadná vec, na ktorej takýto systém stojí a padá, preto ju netreba podceňovať.

Vo FreeBSD sa, rovnako ako v novších distribúciách Linuxu, používa ako defaultný grafický subsystém *X.org* (voliteľne sa však dá miesto neho nainštalovať aj *XFree86*). Jeho konfigurácia je úplne zhodná ako u Linuxu, nie je nijak systémovo špecifická, lebo sa jedná vlastne o aplikáciu bežiacu na systéme, nie o systémovú komponentu ako takú, tvoriacu súčasť jadra či userlandu. Zájemcovia si môžu nájsť potrebné informácie v dokumentácii na serveri projektu X.org, prípadne môžu využiť svoje skúsenosti z konfigurácie X v Linuxe. Pre začiatočníkov však uvediem základný orientačný návod, ktorý by mal fungovať u väčšiny štandardných prípadov. Dajme sa teda do konfigurácie.

Pre úplných začiatočníkov doporučujem použiť na konfiguráciu X-servera jednoduchú grafickú utilitu *xorgcfg*, ktorá sa spustí z príkazového riadku pod rootom. Utilita sa sama snaží nájsť optimálne nastavenie grafického subsystému. Po nadetekovaní je možné niektoré nastavenia ešte ručne zmeniť (doporučujem iba vtedy, keď viete presne, čo robíte) a potom treba utilitu ukončiť, pričom pred koncom sa spýta, či chceme nastavenia uložiť. Tie sú následne uložené do súboru `/usr/X11R6/lib/X11/xorg.conf`. Utilita vie vo väčšine prípadov urobiť celkom dobrú konfiguráciu X-servera, ale málokedy nastaví skutočne najlepšie hodnoty rozlíšenia a obnovovacej frekvencie, niekedy sa dokonca netrafí do použiteľného nastavenia vôbec. Samozrejme treba rátať aj s tým, že táto utilita nenastaví sama od seba akceleráciu a podobné veci.

Pre tých, ktorí si chcú systém nastaviť detailnejšie, doporučujem utilitu *xorgconfig*. Tu už treba vedieť konkrétne parametre použitej grafickej karty a monitoru. U karty je treba vedieť model a veľkosť Video RAM, u monitora zase horizontálny a vertikálny pracovný frekvenčný rozsah. Utilita beží na rozdiel od predchádzajúcej v textovom režime a prevedie užívateľa radom otázok o použitej klávesnici, myši, jazyku klávesovej mapy, frekvenčných rozsahoch monitora, parametroch grafickej karty (je možné si vybrať prednastavenia grafického adaptéra zo zoznamu) a na záver sa po určení defaultného módu zobrazenia a defaultnej bitovej hĺbky farebnej škály spýta, či chceme súbor s konfiguráciou uložiť. Tentokrát nastavenia ukladá do `/etc/X11/xorg.conf`. Opäť platí, že je treba ju spustiť pod užívateľom root. Dve dôležité upozornenia:

1. Systém ako prioritný využíva konfiguračný súbor z `/etc/X11`, ten z `/usr/X11R6/lib/X11` používa iba vtedy, keď v `/etc/X11` nijaký `xorg.conf` nenájde. Treba na to myslieť, ináč sa môže stať, že budem nejak meniť súbor v `/usr/X11R6/lib/X11` a budem sa čudovať, prečo tie zmeny nie sú vidieť...

2. Pri nastavovaní cez `xorgconfig` je potrebné vedieť potrebné parametre pracovných frekvenčných rozsahov monitora skutočne presne, lebo nevhodným nastavením môže dôjsť až k poškodeniu monitora!

Tip: ak má niekto myš s kolieskom, môže si ho v `xorg.conf` spojzduť pridaním riadku do sekcie parametrov myši:

```
Option "ZAxisMapping" "4 5"
```

Samostatnú kapitolu tvorí inštalácia špecifických ovládačov od výrobcov grafických kariet. To je niekedy nevyhnutnosťou, lebo niektoré karty majú problémy s ovládačmi dodávanými priamo s X servrom. Napríklad moja NVidia GeForce mi v nepredvídateľných náhodných intervaloch pripravovala s generickým ovládačom nepríjemné prekvapenie v podobe rozsypania sa obrazu na farebné pruhy. Riešením bola inštalácia originálneho ovládača od NVidie, po ktorom sa daný problém už nikdy nevyskytol. Tu je nemožné dať nejaký konkrétny návod, každý výrobca má iný odporúčaný inštalčný postup. Najčastejšie sa jedná o spustenie shellového skriptu, ktorý urobí automatickú kompiláciu ovládača priamo na daný systém. Po kompilácii je treba ešte podľa pokynov výrobcu upraviť `xorg.conf` tak, aby sa zaviedol patričný ovládač a niekedy aj ďalšie moduly potrebné na jeho bezchybnú činnosť (napríklad `glx` alebo `dri`).

Tu by som rozhodne pred samotnou inštaláciou doporučoval dôkladne preštudovať konkrétny README súbor, ktorý je štandardne ku ovládaču pribalený, a pozrel si, čo všetko má byť ako nastavené.

Inštalácia a základná konfigurácia KDE

Vo FreeBSD je veľmi jednoduchá. Stačí napísať `pkg_add -r kde` a systém stiahne balíky desktopu vrátane závislých balíčkov z ftp a nainštaluje ich. Popisované sťahovanie môže byť dosť dlhou záležitosťou, preto pre užívateľov s pomalým pripojením doporučujem KDE nainštalovať z CD cez `sysinstall`. Výhodou predkompilovaných balíčkov na ftp je však to, že sú väčšinou aktuálnejšie ako tie z CD.

Teraz treba zabezpečiť, aby sa KDE spúšťalo samo po štarte. Po používatelovi Windows predsa nemôžem chcieť, aby sa hlásil do konzoly a spúšťal si X cez príkaz `startx`. Najelegantnejšou cestou je použiť K Display Manager, alebo tiež v skratke KDM. Upravíme si súbor `/etc/ttys` tak, aby sa na deviatej konzole (teda na `tty8`) po štarte spúšťal KDM:

```
ttyv8 "/usr/local/bin/kdm"    xterm    on    secure
```

A tým pádom máme zabezpečené, že pri štarte systému sa rozbehne KDM a bude ponúkať prihlásenie sa užívateľom. Grafika bude bežať na deviatej konzole a pri štarte systému sa rovno do nej aj prepne. Kompiláciu KDE doporučujem iba pre userov s veľmi silnými strojmi a veľmi veľkou trpezlivosťou, trvá celé hodiny... *Pozor! Nezabudnite pred spustením KDM pridať do systému minimálne jedného obyčajného užívateľa.* KDM v štandardnom nastavení totiž z bezpečnostných dôvodov nepovolí prihlásiť roota a tým pádom nezobrazí nijakého užívateľa – do grafiky sa tým pádom nebude dať prihlásiť. Ak ešte nijakého usera nemáte, rýchlo si jedného urobte cez jednoduchú utilitu `adduser` v príkazovom riadku. KDM sa samozrejme dá nastaviť tak, aby sa root mohol prihlásiť. Ako, to si ukážeme neskôr.

Po prvom prihlásení užívateľa do systému cez KDE sa spustí prepracovaný wizard (sprievodca), ktorý užívateľa jednoducho prevedie základnou konfiguráciou prostredia – vzhľad, správanie sa okien, myši, nastavenie animácií objektov atď. Neľakajme sa, keď bude na výber iba anglický jazyk. V popise lokalizácie si ukážeme, ako veľmi jednoducho KDE poslovenčiť.

Lokalizácia systému a KDE

Národné nastavenia sa v systéme FreeBSD dajú urobiť veľmi jednoducho. Základný systém lokalizujeme editáciou súboru `/etc/profile`, kde stačí pridať na koniec tento riadok:

```
LANG=sk_SK.ISO8859-2; export LANG
```

Prípadne pre český jazyk: `LANG=cs_CZ.ISO_8859-2; export LANG`

Slovenský keymap pre X sme si už v minulej časti (FreeBSD v malej firme – 3 (RAID a Xserver) [5]) zadefinovali rovno v konfigurácii X.org. Tým sa ale lokalizácia neskončila – je treba poslovenčiť (respektíve počeštiť) aj samotné KDE. To, na rozdiel od iných desktopov či window managerov, má v sebe svoj vlastný systém národných nastavení. Teraz musíme jazyk do KDE doinštalovať. Urobíme to príkazom: `pkg_add -r sk-kde`, prípadne pre českú lokalizáciu: `pkg_add -r cs-kde` po vykonaní ktorého nám do systému pribudne lokalizačný balík pre KDE. Potom pôjdeme do Ovládacieho centra KDE (čo je niečo podobné Ovládacím panelom vo Windows) a nájdeme si Nastavenie regiónu a jazyka. Tam stačí pridať slovenský jazyk (eventuálne český) a po potvrdení, odhlásení a prihlásení do KDE by malo všetko korektne fungovať, menu systému by malo bežať v slovenčine (češtine). Kto niekedy aspoň chvíľu používal Windows, ľahko sa v tomto menu zorientuje, je logické a prehľadné.

Obdobne si musíme nastaviť aj klávesnicu. Postup je analogický ako pri zmene jazyka, jednoducho si pridáme klávesnicu cez klikacie menu rozloženia klávesnice v Ovládacom centre a zapíšeme zmenu. Malo by to bežať korektne. Možno si samozrejme nahodiť národných klávesníc viac a meniť ich. Nezabudnite si nastaviť národnú klávesnicu ako výchozí.

Ako vidíte, lokalizácia FreeBSD nie je nič náročné. Tento postup je možné samozrejme použiť pre každý podporovaný jazyk. To by bolo ohľadne lokalizácie FreeBSD snád' všetko, čo človek potrebuje pre bežnú prax. Pokiaľ sa všetko podarilo, mali by ste mať na stole mašinu, ktorej prívetivé a veľmi pekne vyzerajúce grafické rozhranie beží v národnom prostredí.

Ešte trochu psychológie

Dovolím si tu jednu poznámku na okraj problematiky pre ľudí, ktorí to myslia s implementáciou akéhokoľvek Unixu do firmy vážne – *pri návrhu riešenia netreba podceňovať ani vzhľad a prítulnosť systému*. Je to často dosť podstatný argument pri zavádzaní nového systému do firmy a mnohí admini, ktorí sú ináč vynikajúci odborníci v svojom systéme, na tento aspekt často nemyslia a pri návrhu firemného riešenia sa nevedia pozrieť na vec očami obyčajného užívateľa. Nový systém musí vyzeráť už na prvý pohľad lákavo, „dospelo“, nie ako nejaká napochytre zbúchaná seminárna práca. Je veľmi dôležité, aby sa systém okamžite na prvý pohľad zapáčil obyčajným ľuďom, ktorí ho budú používať denne a ktorým by mal nahradiť ich doteraz preferovaný Windows. Uľahčí to akceptáciu nového systému firemnými užívateľmi.

Asi ťažko presvedčím človeka zhýčkaného XP-čkom, aby ho vymenil za Unix, keď sa pri štarte na neho škerí nejaké hnusné hranaté šedé okno s pre neho nič nehovoriacimi anglickými odbornými slovami, či dokonca nedajbože čierna obrazovka s blikajúcim kurzorom. Taký user mi rovno povie – „Tak na denné používanie tejto hrôzy ma asi nikto na svete neprehovorí“. A nezabudne svoj názor šíriť ďalej. Ale ak uvidí niečo, čo je už na prvý pohľad krajšie ako Windows, povie si – „No, vyskúšame, vyzerá to šikovne“. A keď zistí, že tá vec nielenže dobre vyzerá, ale aj veľmi rýchlo chodí a šikovne sa s ňou pracuje, už je ruka v rukáve. Miesto odporcu migrácie z Win na Unix tu máme jej obhajcu, ktorý sa taktiež svojím názorom netají. Hovorím tu o praktickej skúsenosti z firmy.

Holt, obal predáva. . . asi tu sa skrýva tajomstvo obrovského komerčného úspechu Windows a MacOS. Treba stroje priblížiť ľuďom, a nie prevychovávať ľudí, aby si na namáhavé, nechutne technokratické, neintuitívne a neprehľadné ovládanie zvykli. Viem, z úst unixáka to znie divne až paradoxne, ale je to holý fakt. Netreba si zakrývať oči pred pravdou. Sme ľudia a systém sa má prispôbiť nám. Ľudia z Apple to dokázali u MacOS (tiež je to vlastne Unix, dokonca je to príbuzný BSD), tak prečo sa nepoučiť z úspechov iných a nepoužiť rovnakú filozofiu aj pre iné Unixy?

Inštalácia CUPS a ostatných potrebných modulov

V tomto dieli sa naučíme nastavovať tlačiareň. Pre FreeBSD je ako výchozí tlačový systém použitý klasický LPD. Má však niekoľko zásadných nevýhod, ktoré spôsobujú jeho problematické nasadenie v prevádzke. Je trochu ťažkopádne konfigurovateľný. Nie je príliš vhodný tam, kde treba tlačiť rôzny

užívateľom určený typ tlačových úloh (tlač textu vs tlač postscriptovej grafiky), na každý typ úlohy je totiž treba tlačiarňi priradiť v konfiguračnom súbore zodpovedajúci filter a ťažko toto môžem žiadať od bežného užívateľa. Je trochu komplikované ho integrovať do sieťových tlačových služieb. Preto sa v súčasnosti ako jeho nástupca začína presadzovať CUPS (Common Unix Printing System). Je elegantne spravovateľný cez webové rozhranie a konfiguračné súbory aj z KDE a má vynikajúcu sieťovú podporu. Ukážeme si, ako spojazdniť tlačiareň od najrozšírenejšieho výrobcu Hewlett-Packard pod CUPSom.

CUPS nainštalujeme príkazom `pkg_add -r cups`, prípadne ho môžeme nainštalovať z portu príkazmi (samozrejme pod rootom)

```
cd /usr/ports/print/cups
make install clean
```

Pozn. autora – Pozor! CUPS sa v systéme nastaví ako výchozí tlačový subsystém a prepíše existujúci lokálny tlačový systém! Na danú vec upozorňuje aj administrátorská príručka CUPSu. To by malo pri fungujúcej sieti zabezpečiť stiahnutie zdrojčkov na CUPS, ich skompilovanie, inštaláciu a vyčistenie systému od pomocných súborov po kompilácii. Proces kompilácie je spravidla bez akýchkoľvek problémov, tu by som sa však ešte zmienil o tom, že je dobré do `/etc/make.conf` pridať riadky, ktoré nastavia v kompilátore optimalizáciu pre použitý procesor, čím je možné na rozdiel od hotovej binárky z balíka skompilovať optimálnu verziu pre daný procesor. Príklad nastavenia optimalizácie pre Pentium 4:

```
CPUTYPE=p4
CFLAGS= -O -pipe
COPTFLAGS= -O -pipe
```

Prvý riadok hovorí o použitom procesore, druhý a tretí o zapnutí 1. stupňa optimalizácie a ukladaní medzivýsledkov kompilácie do RAM cezrúru, nie priamo na disk, čo zvýši rýchlosť samotnej kompilácie. Uvedené voľby je dobré mať nastavené aj preto, že sa použijú nielen pre skompilovanie optimalizovaných aplikácií, ale aj pri aktualizácii systému zo zdrojového kódu. Ďalej budeme potrebovať sadu tlačových filtrov foomatic. Sú to skripty, ktoré pri požiadavke na tlač konvertujú PostScript dáta na binárny formát zodpovedajúci danej tlačiarňi, nezávisle na použitom tlačovom systéme. Sadu filtrov foomatic nainštalujeme cez `pkg_add` do systému príkazom `pkg_add -r foomatic`, ktorý do systému pritiahne túto trojicu balíkov:

```
foomatic-db
foomatic-engine
foomatic-filters
```

Pozn. autora – V súčasnosti je k dispozícii aj štvrtý balík *foomatic-hpijs*, ktorý v čase, keď som inštaloval našu tlačiareň, ešte nebol a ani môj setup ho nevyužíva.

Nakoniec je potrebné nainštalovať balík *hpijs* (Hewlett-Packard InkJets). Jedná sa o balík ovládačov pre atramentové tlačiarne HP (a aj pre niektoré laserové). Firma HP ako jedna z mála uvoľnila špecifikácie dátových formátov svojich tlačiarní, a preto sú tlačiarne HP výborne podporované v Unixe. Pre laserové tlačiarne je, ak nenájdeme zodpovedajúci ovládač v balíku *hpijs*, vhodné použiť balík *hpoj* (Hewlett-Packard OfficeJet) zameraný viac na laserové tlačiarne. Tu by som ešte uviedol, že celkom solídne sú podporované aj tlačiarne Canon BJC série a tlačiarne Epson Stylus. Podobné sady ovládačov sú totiž k dispozícii aj pre ne. V prípade, že nenájdeme ovládač pre tú svoju tlačiareň, ešte nie je nič stratené, môžeme použiť niektorý obecný ovládač v CUPS-e alebo vyskúšať ovládač podobného modelu. Napríklad ovládač pre HP DeskJet 1120C bude určite chodiť bez väčších problémov aj s pripojenou HP DeskJet 1125C... Niektoré ovládače sú zase dostupné ako komerčné a dajú sa zakúpiť.

Na záver bychom mali nainštalovať službu `cups-lpd`: `pkg_add -r cups-lpd`. Táto služba nám zabezpečí, že staršie unixové programy, ktoré „nepoznajú“ CUPS a posielajú svoj tlačový výstup na lpd, budú vedieť bez problémov tlačiť, a to aj cez sieť. CUPS bude emulovať starý systém lpd, samozrejme pri plnom zachovaní svojich výhod jednoduchej administrácie a bezproblémovej sieťovej konektivity.

Prvý štart CUPS – overenie činnosti

Skontrolujeme si bežiacie procesy v systéme `ps -aux | grep cupsd`. Mala by sa nám objaviť hláška o cupsd, napríklad takáto:

```
root  476  0,0  0,3  8256  6716  ??  Is  7:02  0:01,87  /usr/local/sbin/cupsd
```

ktorá hovorí o tom, že démon `cupsd` je spustený s právami užívateľa root. Do ľubovoľného webového browsera (stačí aj Konqueror z KDE) zadáme `http://localhost:631` a mala by sa nám objaviť obrazovka webového rozhrania CUPS. Je tam možné pridávať, mazať a spravovať tlačiarne. Rozhranie obsahuje aj kompletnú dokumentáciu ku CUPS. Navyše je rozhranie dosť intuitívne, takže predpokladám, že so správou tlačiarne nebude nijaký problém. Ak máme v systéme bežiaci proces a rozbehne sa aj webové rozhranie, máme v podstate vyhraté, CUPS nám beží!

Pridávame do systému tlačiarne

Pristúpme teraz k inštalácii ovládača tlačiarne. Najjednoduchšie sa to robí cez wizard v KDE, ktorý nájdeme v *Ovládacom centre* v sekcii *Periférie – Tlačiarne*. Zvolíme režim administrátora a po zadaní mena a hesla roota prepne momentálne používaný tlačový systém na CUPS. Potom cez položku *Pridať tlačiareň/triedu* spustíme wizard so sériou otázok. Bude sa pýtať na typ rozhrania (sériová, paralelná, USB, sieťová...), na meno, umiestnenie a ďalšie potrebné údaje. Bez námahy si takto možno dosť podrobne naklikať celú konfiguráciu tlačiarne.

Akonáhle sa wizard bude pýtať na ppd súbor definície tlačiarne, nepoužijeme defaultne ponúkané ovládače CUPSu, ale musíme systému vnútiť ovládač z hpijs, teda zadať zvolenie cesty k súboru ručne a určiť cestu k danému ovládaču. U mňa je to `/usr/ports/print/hpijs/work/hpijs-1.7.1/ppd/HP-DeskJet_1125C-hpijs.ppd`.

Pozn. autora – súčasťou wizardu je aj nastavenie spôsobu riešenia oprávnení užívateľov k používaniu tlačiarne. Sú možné dva spôsoby – alebo akceptovať všetkých a tlač zakázať užívateľom uvedeným v zozname, alebo zakázať tlač všetkým a povoliť ju vymenovaným užívateľom. Toto nechám na zváženie každému správcovi. Ak sa jedná o tlačiareň, ktorá je spoločná a využíva ju v rozumnej miere každý, asi nebude mať zmysel nejak striktné limitovať prístup na ňu... v istých prípadoch (napr. ak mám vo firme ľudí vyžívajúcich sa v tlačení kvanta najrôznejších fotiek z webu a podobných zbytočných srandičiek), to však asi bude nutné spraviť.

Ak sme wizard statočne vyplnili do konca, tlačiareň by mala byť pripravená k tlači. V menu tlačiarne nájdeme možnosť vytlačiť testovaciu stránku. Ešte jedna vec – dobré je si po úspešnom nainštalovaní tlačiarne aj prezrieť jej základné nastavenia (formát tlačového média, kvalitu tlače, použitý defaultný podávač atď...) a v prípade nesúladu s požadovaným stavom ich prestaviť, aby potom nedošlo k nepríjemným prekvapeniam pri tlači. Uvedený postup pridávania tlačiarne KDE wizardom je možné opakovať pre nastavenie ďalších tlačiarne.

Podpora lpd v CUPSe pre aplikácie nepodporujúce CUPS

Aby sme vedeli použiť aj staršie programy a programy z nejakej príčiny nekompatibilné s CUPSom, ktoré očakávajú obsluhu tlačie cez lpd, musíme rozbehnúť aj službu cups-lpd. Táto služba nie je samostatným démonom, ale (logicky, keďže obsluhuje aj sieťovú tlač) používa sieťový démon inetd. Postup je opäť nekomplikovaný. V `/etc/rc.conf` musí byť

```
inetd_enable="YES"
lpd_enable="NO"
```

Do `/etc/inetd.conf` pridáme tento riadok:

```
printer stream tcp nowait root \\  
  /usr/local/libexec/cups/daemon/cups-lpd cups-lpd
```

a v `/usr/bin/` vykonáme tieto kroky:

- zálhujeme `lpr` ako `lpr.orig` príkazom `mv /usr/bin/lpr /usr/bin/lpr.orig`
- urobíme symlink príkazom `ln -s /usr/local/bin/lpr /usr/bin/lpr`

Po tomto kroku bude chodiť aj tlač zo starých aplikácií, prípadne z aplikácií priamo nepodporujúcich CUPS (typickým príkladom sú staršie verzie OpenOffice.org či QCAD), a to aj cez sieť.

Terminály sa vracajú späť – alebo vývoj ide v špirále

Myšlienka terminálu je stará takmer ako počítače samy. V princípe sa jedná o to, ako pristupovať k využívaniu služieb centrálného počítača pomocou klientskeho zariadenia, pričom klientské zariadenie využíva užívateľ iba na vstup a výstup a samotný program beží na hlavnom počítači. Tieto riešenia (ešte na báze znakových – alfanumerických terminálov) boli s príchodom prvých sálových počítačov bežne využívané a samozrejme, kým ich nevytlačila éra PC s filozofiou samostatného plnohodnotného počítača na každom stole. Na začiatku sa zdalo, že toto riešenie má spústu výhod. Áno, má, ale iba pre niektoré prípady nasadenia. Žiaľ, pre firemný informačný systém toto usporiadanie prináša veľa nevýhod, či už z hľadiska prevádzky, funkcionality alebo nákladov na prevádzku a udržiavanie takéhoto systému v konzistentnom a stále aktuálnom stave. Proste pri nárokoch na firemný IT systém je jednoznačne výhodnejšie, úspornejšie a elegantnejšie riešenie na princípe centralizovaného systému.

Dnes sa teda oprášená myšlienka terminálu po rokoch vracia späť, avšak obohatená o možnosti použitia grafického režimu, čo je pre dnešné firemné riešenia už nutnosť. Výhody tohoto usporiadania sú zjavné:

1. spoločný dátový priestor s hromadným zálohovaním
2. jednoduchá administrácia
3. bezproblémová integrácia celého firemného prostredia do jedného kompaktného celku
4. všetky IT prostriedky je možné bez problémov pružne zdieľať v prípade potreby ktorýmkoľvek z užívateľov a sú stále k dispozícii
5. každý užívateľ sa môže prihlásiť z ľubovoľného stroja a okamžite má k dispozícii svoje aplikácie, svoje dáta aj svoje nastavenie prostredia, samozrejme aj rovnaký výpočtový výkon
6. optimalizácia využitia systémových prostriedkov
7. jednoduchšie riešenie havarijných situácií, prestojov a výpadkov
8. ekonomická výhoda takéhoto riešenia – možnosť využitia morálne zastaralej výpočtovej techniky na plnohodnotnú prácu
9. bezpečnosť
10. škálovateľnosť

Aj firmy, ktoré predtým odmietali terminálové riešenia, sa k nim teraz potichu vracajú. Napríklad taký Microsoft a jeho Terminal Server...

Prečo teda terminálové riešenie v malej firme?

Ak sú dáta uložené na jednom mieste, je záloha veľmi jednoduchá, stačí zálohovať dáta na serveri. Pracovná stanica nemá nič, čo by stálo za námahu pri zálohovaní a čo by bolo potrebné nejak extra

chrániť (body 1, 9). V prípade potreby sa dá nahradiť ktoroukoľvek inou (viď bod 7). Výhodné je aj to, že súbory sú na jednom mieste a teda sú v rámci celej firmy informačne konzistentné. Iste každý pozná situáciu, keď je jeden dokument v rôznom štádiu rozpracovanosti na desiatich či viacerých strojoch, užívateľ začne písať do jednej kópie a zabudne na to, že včera písal do inej, čo sa síce volala rovnako, ale bola na inom stroji / v inom adresári (pre usera je to iba drobný nepodstatný detail...). Nakoniec takýto prípad skončí chaosom („včera niečo napíšem, dnes to tam nemám, zase to vyzerá inakšie“) a nastupuje informatik, ktorý sa zúfalo snaží dať dáta dohromady a urobiť z n verzií jednu aktuálnu a úplnú... pri jedinom dátovom úložisku a rozumne nastavených prístupových právach sa to pochopiteľne stať nemôže.

Správa takéhoto centralizovaného systému sa potom redukuje na správu centrálného počítača. Všetko je pohromade a pod kontrolou správcu, useri, aplikácie, dáta, služby. Kdekoľvek sa user prihlási, všade mu systém ponúkne rovnaké, jeho do detailu nastavené prostredie. Každý má presne definované práva. Všetky zdieľané prostriedky sú na serveri, teda ak sa užívateľ úspešne prihlási a má na ne práva, môže ich využívať bez problémov a riešenia zdieľaní, komunikácie a autentifikácie medzi n stanicami. Takisto správca môže väčšinou odstrániť problém bez zbytočného behania od stanice k stanici a obojstranne nepríjemného odstavovania userov od práce. Nikdy sa nijaké nastavenie nerozhodí „samo od seba“, ani nikto z userov nemá možnosť zasiahnuť do niečoho, čo môže ohroziť systém alebo dáta v ňom. Dáta nikdy neopustia server a neskopírujú sa na klientskú stanicu, klientska stanica prijíma iba výstup z ich spracovania a posieľa serveru vstupné údaje, čo je veľké plus pre bezpečnosť citlivých údajov. Kto sa niekedy staral len o niekoľko Win staníc, chápe rozdiel.

Ako klientské stroje možno použiť staré PC s pamäťou minimálne 64 MB (viac RAMky je samozrejme lepšie, terminály potom neswapujú a systémy im bežia priamo z RAM, naše majú 128 a jeden dokonca 256), procesorom min. Pentium 200 MHz a malým diskom, na ktorom je inštalácia minimálneho základného systému a X. Klientské stroje sú prakticky bezúdržbové a nie je okrem občasnej kontroly konzistencie disku ani nutné ich akokoľvek spravovať. V prípade nasadenia bezdiskových staníc odpadá aj tento posledný krok (tam je zasa nevýhodou o hodne pomalší štart, daný tým, že si musí stanica počas bootu celý systém stiahnuť zo siete). Kto chce a má trochu viac peňazí, môže bootovať aj z Compact Flash ROM a vyhnúť sa tak hrkotajúcemu a potenciálne poruchovému disku, predávajú sa CF adaptéry na bežný IDE kábel.

Za pozornosť stoja body 6, 8 a 10. Veľmi spolu súvisia. Ak systém prestane zvládať požadovanú záťaž, stačí vymeniť server za výkonnejší a ide sa ďalej naplno, na rýchlosť systému nemá klientska stanica takmer nijaký vplyv. Výkon skôr ovplyvňuje rýchlosť prenosu dát po sieti, latencia prenosu (teda časové oneskorenie – doba od vyslania dát po ich doručenie) a samozrejme výkon servera samotného. Preto sa po určitej dobe nemusia vyradovať staršie PC, hoci by už nespĺňali požadované HW parametre na samostatnú prevádzku, na funkciu zobrazovacej vstupno-výstupnej jednotky majú stále výkonu dostatok a možno ich použiť prakticky dovtedy, dokedy slúžia. To je veľmi výhodné aj z hľadiska nízkych nákladov na zostavenie a upgrade takéhoto riešenia. Navyše server plynule rozdeľuje výpočtový výkon tým procesom, ktoré ho potrebujú, a preto aj keď je vyťažený takmer naplno, užívatelia pracujú plynule a rýchlo. K tomu však netreba kupovať n výkonných strojov (ktoré by po 80 percent času iba stáli a počítali šetrič a iba 20 percent ich výkonu by sa skutočne využilo na prácu), ale iba jeden. Teda výhoda investičná aj prevádzková zároveň...

Licencie komerčných riešení sa môžu predražiť

Silným dôvodom pre nasadenie unixového terminálového systému je aj cenová a licenčná politika MS – je rozdiel kúpiť trebárs 5 samostatných plne vybavených PC pracovných staníc, ku každej z nich OS Windows + kancelársky a iný software, a kúpiť 5 lacných starších počítačov iba za cenu HW, bez OS, bez platenia softwarových licencií na každú inštaláciu. Takto možno aj s malým rozpočtom postaviť veľmi solídny firemný systém. Celá vec stojí (a padá) len na fantázii a schopnostiach admina – tvorcu firemného informačného systému.

V případě Microsoft Terminal server to nie je žiadna výhra – majiteľ systému musí zaplatiť cenu licencie OS Windows v klientskom stroji (z niečoho ten TS klient predsa spustiť musí), licenciu MS Server 2000, licenciu MS Terminal Server a potom ešte licenciu z každého sedenia (per seat). A má holý systém bez aplikácií, ktorého klientske stanice vyžadujú úplne rovnakú starostlivosť ako samostatné stroje (ten Windows treba pravidelne čistiť od temporaries, defragmentovať, udržiavať aktuálny antivírus atď). Jediným zachovaným plus z celej terminálovej koncepcie tu ostáva to, že stanice používajú výpočtový výkon servera.

Ekonomický aj prevádzkový efekt nasadenia terminálov v prostredí MS sa teda viacmenej stráca a pre malú firmu je to dokonca silne nevýhodné riešenie... záujemcom doporučujem ako povinné čítanie cenníky Microsoftu. Obdobná situácia je, keď sa miesto MS Terminal Servera kúpi Citrix - všetky hore uvedené veci (až na iný typ terminálovej nadstavby MS servera) platia opäť.

Dajme užívateľom to najlepšie!

Ešte jedna poznámka, skôr psychologická ako odborná – aby užívateľ nemal pocit, že sedí len pred nejakým „orezátkom“ a že mu admin hodil na stôl ten posledný šrot z celej firmy, je vhodné dať terminálom pokiaľ možno veľké monitory a optické myši (a samozrejme aj skrinky PC vyčistiť od prachu, nálepiek a inej špiny, aby vyzerali k svetu). Z hľadiska prevádzky je to síce bez významu, ale z hľadiska akceptácie nového systému nedôverčivým Win užívateľom to má význam obrovský. Ťažko niekoho nadchne stará zaprášená kraksňa s blikajúcim 15” monitorom. Zato keď niekomu postavím na stôl monitor devätnástku s perfektným ostrým obrazom, na ktorom je radosť čítať, to je niečo iné... A keď ešte spozná tú rýchlosť práce, nazad ku Windowsu sa mu nebude chcieť. Monitor 19” sa dá v bazári zohnať za pár tisíc. Celá zostava sa v prípade bazárového predaja dá kúpiť tak za 4 až 5 tisícok Sk. Porovnajme si to s cenou novej PC zostavy...

Topológia siete

Pre terminálovú sieť je optimálne a najčastejšie využívané hviezdicové usporiadanie, kde základ siete tvorí switch, na ktorý sa pripájajú stanice aj server. Nedoporučujem používať hub, pre rýchlu prácu s xdmcp reláciami je potrebné mať malú latenciu a pokiaľ možno vysokú priepustnosť siete. Tu má switch oproti hubu výhodu, že posielajú pakety iba príjemcovi a ostatní účastníci siete nie sú bombardovaní hromadou cudzích paketov, ktoré by ich spomaľovali a zbytočne zvyšovali traffic. Navyše je to aj výhoda z hľadiska bezpečnosti. V praxi sa osvedčil malý osemportový 100 MBit switch, chodí to rýchlo a odozvy terminálov mám v reálnom čase. Plánujem rozšírenie na 16 portový rackový switch.

Udržanie nízkej latencie komunikácie je pre X dokonca dôležitejšie ako prenosová rýchlosť samotná. Skúšal som pripojiť terminál cez 11 MBit WiFi a napriek tomu, že šírka pásma stačila a nebola využitá, „na doraz“, oneskorenie prenosu bolo tak strašné, že sa dalo robiť iba s veľkým sebazaprením... niekedy človek videl reakciu na úkon, ktorý urobil, až po pár sekundách. Použiteľné to bolo snáď iba na núdzové účely, rozhodne nie na serióznu prácu.

Konfigurácia na strane servera (X klienta)

Pre rozbehnutie terminálových služieb X je potrebné nastaviť na serveri (tu sa myslí počítač, na ktorom aplikácie fyzicky pobežia, z hľadiska terminológie systému X window je to X klient) konfiguračný súbor správcu prihlásenia, aby umožňoval pripojovanie vzdialených sedení cez protokol xdmcp (X Display Manager Communication Protocol). V našom prípade sa použil správca prihlásení kdm. Jeho konfiguračný súbor sa nazýva `kdmrc` a nájdeme ho v adresári `/usr/local/share/config/kdm`. Súbor má veľmi ľahko pochopiteľné názvy premenných a jeho jednotlivé voľby teda nie je problém nájsť a nastaviť. Pre povolenie vzdialených sedení je treba nastaviť sekciu `[Xdmcp]` takto:


```
[Xdmcp]
Enable=true
Port=177
Willing=/usr/local/share/config/kdm/Xwilling
Xaccess=/usr/local/share/config/kdm/Xaccess
```

Prvý riadok zapína xdmcp protokol, v druhom sa určuje, na akom tcp porte má X komunikovať. Pre xdmcp v firemnej LAN sieti sa doporučuje nechať tam výchozí port 177. Meniť to má význam iba v odôvodnených prípadoch, napríklad keď chceme riešiť bezpečné pripojenie vo verejnej sieti tunelovaním protokolu xdmcp cez ssh. Tretí riadok hovorí o ceste na skript **Xwilling**, ktorý umožňuje samotné vzdialené prihlásenie. Pre bezpečnosť je podstatný štvrtý riadok, ktorý ukazuje cestu na súbor **Xaccess**. Jedná sa o konfiguračný súbor v tom istom adresári ako **kdmrc**, ktorý povoľuje prístup na X jednotlivým IP adresám. Po inštalácii je nastavený tak, aby povolil prihlásenie akéhokoľvek stroja, ktorý o vytvorenie vzdialenej X session požiada. Jeho editáciou môžeme zakazovať a povoľovať prístupy jednotlivým klientským staniciam.

Ešte tri poznámky. Z bezpečnostných dôvodov je zakázaný login roota do KDE. Ak ho z nejakých príčin potrebujeme, dá sa to urobiť v **kdmrc** v sekcii **[X-*-Core]** cez voľbu **AllowRootLogin=true** – pôvodne je tam nastavené **false**.

Druhá dôležitá vec – je dobré na serveri zakázať možnosť vypnúť počítač zo vzdialenej X session. Predídeme tým nepríjemnostiam, keď nejaký „snaživiec“ namiesto ukončenia sedenia vypne server a s ním aj ostatných užívateľov s rozrobenou prácou. Toto je možné najjednoduchšie urobiť v ovládacom centre KDE, kde v sekcii Správca prihlásenia nastavíme, že shutdown je povolený iba z lokálneho prihlásenia. Samozrejme je možné nastaviť to aj editáciou **kdmrc**, kde povolíme shutdown z lokálu a zakážeme ho z vzdialeného prihlásenia.

Tretia poznámka, ktorú som už uviedol v časti **Inštalácia** [6], pre každý prípad ju však opakujem. Z mne neznámeho dôvodu protokol xdmcp vo FreeBSD nechodí bez povolenia IPv6 v konfigurácii siete na strane klienta aj servera. Sám som strávil dva týždne rozmýšľaním nad tým, prečo mi terminál nepripojí vzdialený server. Potom som narazil na popis chyby na jednej konferencii o FreeBSD a KDE. Riešenie popisované v konferencii bolo jednoduché, povoliť IPv6. Po nastavení **ipv6_enable="YES"** v **/etc/rc.conf** na serveri aj termináli sa terminál okamžite pripojil. Zrejme blokovanie IPV6 nechceme zablokuje aj xdmcp... ale môže to byť len môj dohad. Každopádne to po príslušnom nastavení funguje a chodí bez chyby.

Konfigurácia na strane terminálu (X servera)

Na strane X terminálu (v terminológii X window sa nazýva X server) je iba potrebné spustiť X s parametrom **query** na daný server. Keďže od obyčajných userov nemôžem chcieť, aby sa prihlásili lokálne do konzoly a odtiaľ si ručne spustili vzdialenú reláciu X, musíme tento proces automatizovať. Mne sa osvedčilo jednoduché riešenie. Do adresára **/etc/rc.d** treba vytvoriť súbor s názvom **X.sh**. Je to jednoduchý shellový skript, ktorý obsahuje iba dva príkazy:

```
/usr/X11R6/bin/ X -query 10.10.10.10
shutdown -p now
```

kde 10.10.10.10 je adresa nášho terminálového servera. Po štarte systému na terminálovej stanici automaticky nabehne vzdialené prihlásenie na server a objaví sa nám rovnaké prihlasovacie okno ako na serveri samotnom. Ak chce užívateľ skončiť prácu s terminálom, ukončí beh X na termináli cez Ctrl+Alt+Backspace a ihneď sa spustí ukončovacia sekvencia, ktorá zastaví systém a (v prípade, že to matičná doska umožňuje a táto vlastnosť je u nej podporovaná) automaticky vypne napájanie terminálu. Ak nie, iba ukončí systém a treba terminál ručne vypnúť.

Inštalácia aplikácií z balíčkov

Systém FreeBSD umožňuje správu aplikačného softwaru v zásade dvomi spôsobmi. Prvý spôsob je inštalácia predkompilovaných balíčkov z FTP zdrojov, druhým je správa softwaru pomocou systému portov (v zásade je to do určitej miery automatizovaný preklad aplikácie zo zdrojového kódu). Každý zo spôsobov má svoje výhody aj nevýhody, v reálnej prevádzke je teda pre FreeBSD administrátora vynikajúcim riešením pri správe a udržiavaní softwaru kombinovať oba spôsoby. Oba spôsoby sú spolu previazané, na čo tiež v článku poukážem.

Tu je tento systém značne podobný balíčkovacím systémom na iných unixových systémoch. Spočíva v nainštalovaní predkompilovaného softwaru do systému, kde sú po stiahnutí komprimovaného archívu na disk stroja z neho vybalené binárky, dátové súbory a konfiguračné súbory. Tieto sú nakopirované do patričných adresárov a niekedy (v prípade niektorých systémových programov) aj automaticky inicializované a spustené. Systém si udržiava závislosti (podobne ako u iných balíčkových systémov, napríklad u RPM), a tak pri požiadavke na inštaláciu softwaru sa stiahnu aj tie balíčky, od ktorých je požadovaný balík závislý, a bez ktorých nepobeží.

Výhody balíčkov sú zrejme už zo samotného princípu tohto systému – aplikáciu možno stiahnuť a nainštalovať veľmi rýchlo, neprevádza sa nijaká kompilácia, stiahnu sa hotové binárky. To je veľké plus v prípade takých obrov medzi programami, ako OpenOffice.org, KDE, GIMP a podobné veľké aplikácie, u ktorých sa čas prekladu zo zdrojového kódu môže značne natiahnúť. Ďalšou výhodou môže byť aj to, že tento systém umožňuje v prípade staticky prilinkovaných knižníc aj spúšťať tie aplikácie, ku ktorým nemáme v systéme nainštalované patričné knižnice zodpovedajúcej verzie. Viete dobre, aké je problematické, keď sa na systéme nedarí skompilovať program napríklad preto, že na kompiláciu (nie na samotnú prevádzku) potrebujem knižnicu libXYZ vo verzii napr. 1.1.2, kdežto v systéme mám už 1.1.8 a tú 1.1.2 si nemôžem nasilu hodiť do systému, lebo si znefunkčím 10 ďalších aplikácií, ktoré sú od nej závislé a bez nej nebežia, alebo sú nestabilné. Stiahnutie hotového predkompilovaného balíčka s funkčnou aplikáciou je potom pre správcu hotová spása...

Nevýhody sú tiež zjavné. Môžeme zabudnúť na nejaké optimalizovanie kompilácie pre použitý HW a zapnutie špeciálnych funkcií danej aplikácie. Balíčky sú skompilované tak, aby podľa možnosti bežali na všetkom, a tak sú skompilované iba obecné pre danú platformu (i386, amd64...). Potom nemôžem využiť to, že mám trebárs Pentium 4, balíček bude ten istý, čo pre Celeron II a výhody môjho procesora nedokáže daná aplikácia využiť naplno. Takisto nie je vždy zaručené, že balíčky budú tie najnovšie. Veľakrát sú balíčky už trochu staršie a v aktuálnej verzii sa medzitým hodne vecí vylepšilo. A potom sú isté veci, na ktoré sa jednoducho už zo samotného princípu nedá balíčkovanie použiť. Typickým prípadom je binárka, kde sa proste nedá pripraviť nejaká obecná všade chodiaca verzia, lebo bezprostredne súvisí s aktuálnym jadrom a aktuálnymi verziami závislých aplikácií. Ako príklad uvediem ovladač pre grafické karty nVidia. Ten sa kompiluje presne pre danú verziu jadra a X.org.

Balíčky a ich závislosti sú udržiavané pomocou databázy *pkgdb*. Táto databáza obsahuje prehľad o všetkých nainštalovaných aplikáciách a zdieľaných balíčkoch. Nachádza sa v adresári `/var/db/pkg`. Databázu tvorí súbor `pkg.db` a adresárová štruktúra s podadresármi nazvanými podľa jednotlivých balíkov, kde sú zapísané všetky údaje o inštalovaných balíkoch a závislostiach.

Obsah databázy si môžeme vypísať pomocou príkazu `pkg_info`.

Aktuálnu databázu balíkov si systém stiahne po zadaní `pkgdb -f`, do režimu manuálnych opráv sa dostaneme pomocou príkazu `pkgdb -F`. Balíček si pridáme pomocou príkazu `pkg_add` s nasledovnou syntaxou: `pkg_add -r názov_balíka`, čo spôsobí, že sa systém napojí na FTP zdroj balíčkov a stiahne si balíček aj závislé balíčky, bez ktorých software nepôjde. Parameter `-r` tu značí, že má hľadať akýkoľvek balík s príbuzným názvom. V opačnom prípade je nutné zadať presný názov balíka aj s číslom verzie. V prípade, že verzia závislých balíčkov nebude zodpovedať požadovanej, ale budú spätne kompatibilné, pri inštalácii sa zobrazí varovanie, ale aplikácia pobeží. V prípade, že dostupné závislé balíčky nie sú inštalované v systéme, alebo nie sú kompatibilné a zároveň nie sú aktuálne verzie dostupné v danom zdroji balíčkov, k inštalácii nedôjde. V prípade, že máme v systéme starú závislú

komponentu a balík, ktorý inštalujeme, striktne požaduje novú, systém nám odporučí najskôr túto komponentu upgradovať a až potom nám povolí inštaláciu. Takisto inštalácia neprebehne, ak má náš operačný systém inú verziu ako tá, pre ktorú bol balík skompilovaný (napríklad balík je skompilovaný pre 5.4 RELEASE a my máme 5.4 STABLE), čo si pred samotnou inštaláciou balíčkovací systém sám overí.

Tu by som sa ešte zmienil o dôležitej veci – ceste k FTP mirroru s balíčkami. Táto je v systéme nadefinovaná ako premenná prostredia PACKAGESITE a možno ju meniť pomocou `setenv`. Toto využijete v situácii, kde upgradujete systém zo zdrojového kódu, a tým pádom nám nebude inštalácia balíčkov chodiť. Pri prvom probléme ma to trochu vydesilo, systém mi písal hlášku, že nevie nájsť nijaký požadovaný balík, aj keď to pred upgradom normálne chodilo, ale riešenie je jednoduché:

```
setenv PACKAGESITE \\  
ftp://ftp.freebsd.org/pub/FreeBSD/ports/i386/packages-5-stable/Latest/
```

čo nám zmení predvolenú cestu na FTP zdroj softwarových balíkov. Obdobný postup sa dá uplatniť pri prechode na verziu 6-STABLE z 6-RELEASE. Pozor, na konci cesty *musí byť* lomítko, ak ho nedáte, `pkg_add` vám spojí predvolenú cestu s názvom balíka a pochopiteľne nijaký balík nenájde!

Odiinštalovanie je tiež veľmi jednoduché. Robí sa pomocou príkazu `pkg_delete -x názov_balíka` čo spôsobí okamžité odiinštalovanie balíka a jeho vyradenie z databázy. Prepínač `-x` značí, že sa majú odiinštalovať všetky balíky s názvom obsahujúcim uvedený reťazec. Poznámka pre začiatočníkov – nemusím snáď zdôrazňovať unixovú samozrejmosť, že akákoľvek inštalácia aj odiinštalácia sa dá robiť iba s právami roota.

Inštalácia softwaru z portov

Tu sa jedná, ako už bolo uvedené, o inštaláciu softwaru s prekladom zo zdrojového kódu. Princíp je jednoduchý. V systéme sa nachádza adresár `/usr/ports`, kde sú uvedené jednotlivé kategórie portov. V koreni adresára portov je súbor `Index.db` – je to zoznam portov. Viac napovie obrázok.

Každý adresár obsahuje podadresáre, kde sa nachádzajú adresáre samotných aplikácií. V adresároch sa nachádzajú iba popisy, pomocné súbory a samotný Makefile, teda súbor, ktorý riadi kompiláciu a inštaláciu portu.

Strom portov je vhodné udržiavať v aktuálnom stave. Najjednoduchšie to spravíme cez utilitu `cvsup`. Túto utilitu pre príkazový riadok, ak ju v systéme nemáme, si nainštalujeme týmto príkazom: `pkg_add -r cvsup-without-gui`. Potom prevedieme tieto kroky:

```
cd /usr/share/examples/cvsup  
cvsup ports-supfile
```

Spustený proces nám edituje strom portov, čiže vymazáva podadresáre zrušených portov, edituje obsah podadresárov aktualizovaných portov a pridáva podadresáre nových portov. Je to ďaleko elegantnejšie, ako si celý strom portov sťahovať z FTP. Takto sa prenesú iba rozdiely voči aktuálne udržiavanému stromu portov (inkrementálna aktualizácia). Keď máme strom portov aktuálny, môžeme prísť k inštalácii. Uvediem typický príklad – inštaláciu Firefoxu z portu.

```
cd /usr/ports/www/firefox  
make  
make install  
make clean
```

V prvom riadku sa prepnem do adresára príslušného portu, v druhom zadám príkaz na kompiláciu `make`. Začne sa spracovávať Makefile, čo znamená, že sa prevedie táto postupnosť krokov – systém

si do `/usr/ports/distfiles` stiahne aktuálne zdrojáky k danému softwaru z FTP servera tvorca aplikácie (práve preto je potrebné udržiavať si strom portov aktuálny), aplikuje na ne patche a následne ich automaticky skompiluje. V prípade, že aplikácia vyžaduje aj nejaké závislé komponenty a v systéme sa nenachádzajú, rovnaký postup sa vykoná aj pre ne.

Patche sú vlastne nutné úpravy obecných zdrojových kódov potrebné na bezproblémové skompilovanie a beh na FreeBSD platforme, nakoľko stiahnuté zdrojáky sú v drvivej väčšine určené pre iný systém (obecný POSIX systém, Linux, Solaris, ...) a nerešpektujú určité drobné odlišnosti BSD voči iným Unixom, z ktorých sa však môže stať vážny problém pri stabilite a funkcionalite danej aplikácie.

Proces kompilácie sa riadi nastavením súboru `/etc/make.conf`. Teda ak mám tam zapnutú optimalizáciu na dané CPU, bude pri kompilácii zohľadnená. Samozrejme sa dá vnútiť aj cez parameter za príkazom `make`. Kompilácia istý čas potrvá, závisí od veľkosti aplikácie a výkonu nášho systému. V adresári portu sa vytvorí pomocný adresár `work`. Po ukončení kompilácie sa nám vráti kurzor príkazového riadku a zadáme `make install`. Tento príkaz nám skompilovaný port nainštaluje do systému. Posledný príkaz `make clean` nám zmaže medzivýsledky po kompilácii, ktoré nám iba zbytočne zaberajú diskový priestor. Uvedené príkazy môžeme vykonať aj v jednom, zadaním `make install clean`.

Tu by som ešte dodal malú poznámku – v niektorých zriedkavých prípadoch sa môže stať, že nejaký software má ve výchozom nastavení zapnutú aj kompiláciu nejakých vývojových možností, ktoré spomaľujú (debug) alebo môžu ukončiť kompiláciu s chybami (napríklad podpora Samby v Mozille). Tie sa dajú vypnúť dvoma spôsobmi. Buď zadám za príkazom `make` parameter vypínajúci danú voľbu, alebo zadám `make config` a v textovom okne si pohodlne vypnem danú problematickú funkciu. Nasledovný `make` už prebehne bez kompilovania danej časti.

Software, ktorý si skompilujem z portu, mi zároveň automaticky pribudne aj do zoznamu balíkov v `pkgdb`. Teda ak si skompilujem napríklad Firefox z portu, po zadaní príkazu `pkg_info` sa mi medzi vypísanými balíkmi objaví aj on. To znamená, že systém s ním ďalej narába ako s bežným balíkom, môžem ho vymazať ako ktorýkoľvek iný balík, pozná svoje závislé balíky a iné balíky závislé na ňom ho tiež poznajú.

Všetok spravovaný software sa teda javí ako jeden celok, čo sa týka jeho evidencie v systéme aj udržiavania jeho závislostí, a to bez ohľadu na to, či bol inštalovaný kompiláciou zo zdrojákov, alebo stiahnutý ako už hotová predkompilovaná binárka. Táto centralizovaná správa prinášajúca prehľad a poriadok v inštalovaných aplikáciách je veľkou výhodou FreeBSD. Pre správu softwaru by bolo iste frustrujúce a pri veľkom počte aplikácií aj nepoužiteľné, keby sa časť aplikačných programov spravovala v jednom a časť v druhom systéme správy softwaru. Našťastie vďaka premyslenému systému správy aplikácií nám toto vo FreeBSD nehrozí.

Túto prepojenosť oboch systémov možno s výhodou použiť v praxi, keď veci, ktoré z nejakého dôvodu (enormný čas kompilácie, vo veľmi zriedkavých prípadoch problém pri kompilácii, ...) nechcem kompilovať, si iba stiahnem ako balíček a zvyšok si skompilujem na mieru svojmu systému cez porty. S výhodou to neskôr možno použiť pri aktualizácii aplikácií, kedy sa môžu upgradovať všetky jednotne, bez ohľadu na to, akým spôsobom som ich do systému nainštaloval.

Aktualizácia (upgrade) aplikačného softwaru

Všetky aplikácie sa neustále priebežne menia. Niekedy z dôvodov pridania novej funkcionality, často z dôvodu vylepšenia stability, niekedy dokonca z dôvodu odstránenia vážnych bezpečnostných problémov. Preto by mal správca firemného systému, ktorý to myslí vážne, aspoň raz za čas aktualizovať aplikačný software. Našťastie to nie je nič zložité ani obtiažne, FreeBSD má pokročilý systém na pohodlné riešenie aktualizácie nainštalovaných aplikácií.

Tento systém je postavený na kombinácii databázy balíčkov a systému portov. V zásade sa jedná o do značnej miery automatizovaný proces, kde sa z databázy zistia nainštalované balíčky, tie sa potom overia, či sú aktuálne, a ak nie, stiahne sa do systému zdrojový kód aktuálnej verzie softwaru a

z neho sa nový software skompiluje a nainštaluje sa do systému. V prípade, že kompilácia nedopadne úspešne, sa do systému vráti stará verzia aplikácie.

Ako prvý krok je treba si zaktualizovať strom portov cez `cvsup` a databázu portov cez `portsdb -Uu`, keďže aktuálne čísla verzií sa berú z neho. Je to logické, keďže porty obsahujú vždy najaktuálnejšie zdrojáky, zatiaľčo u pripravených bináriek to už tak nemusí byť. Samotná aktualizácia je jednoduchá, používa sa na ňu príkaz `portupgrade`. Ak chcem aktualizovať iba jednu aplikáciu, napríklad wine, zadám príkaz `portupgrade wine`, ktorý vykoná nasledovné činnosti – overí si, či je verzia v databáze balíkov staršia ako tá v portoch. Ak áno, stiahne sa zdroják, opatchuje sa, skompiluje sa nová verzia vrátane závislých komponent, stará verzia sa odloží a nová sa otestuje. Pokiaľ je v poriadku, nainštaluje sa, pokiaľ nie, vráti sa pôvodná verzia zo zálohy. Toto všetko je po príkaze prevedené automaticky. Ak chcem aktualizovať všetky aplikácie, použijem tento postup:

```
cd /usr/ports
portupgrade -a
```

Tento príkaz môže niekedy robiť problém, keď sa nejaká aplikácia odvoláva na závislosť na inom (ešte neupgradovanom) balíku, preto je lepšie použiť príkaz `portupgrade -arR`, kde si `portupgrade` pri kompilácii najskôr zistí závislosti kompilácie (build dependencies, teda zdieľané komponenty, bez ktorých sa aplikácia neskompiluje, zapína sa to cez prepínač `-R`) a závislosti pre beh (to sú zas komponenty, bez ktorých sa program korektné nerozbehne, zapína sa to cez prepínač `-r`). V prípade, že sa objavia jedny alebo druhé závislé programové komponenty, upgradujú sa prednostne najprv ony, až potom program samotný.

V prípade, že nechcem naraz kompilovať všetko, ale chcem stiahnuť všetky potrebné zdrojáky včetně zdrojákov závislých komponent (čo môže trvať pri slabšej linke hodne dlho), urobím to príkazom `portupgrade -aFrR`, ktorý stiahne zdrojáky do `/usr/ports/distfiles` (`-F` značí Fetch). Dá sa to použiť napríklad tak, že večer a cez noc si nechám takto pripraviť zdrojáky a ráno začnem upgradovať. Pri zadaní príkazu `portupgrade -arR` sa potom systém nezdržuje sťahovaním a rovno berie zdrojáky z distfiles, čo proces upgradovania aplikácií dosť urýchli.

Pre paranoidné, nervózne a nedôverčivé týpky tu ešte máme prepínač `-n`, ktorý umožní urobiť „simuláciu“ upgradu. To znamená, že iba vypíše, čo sa počas upgradu urobí, ale na aplikáciu sa nesiahne a všetko ostane, ako bolo. Viac o správe softwaru nájdete vo FreeBSD Handboku a v manuálových stránkach `portupgrade`, ale myslím si, že toto by malo v praxi vo väčšine prípadov bohato stačiť na celkom solídnu správu aplikácií. Zatiaľ sa majte a nech vám vaše systémy idú!

Odkazy

- [1] <http://www.freebsd.org/>
- [2] <http://www.freebsd.org/>
- [3] <http://www.abxlinuxu.cz/clanky/ruzne/freebsd-v-malej-firme-1-lahko-a-bez-namahy>
- [4] <http://www.abclinuxu.cz/clanky/show/106723?page=1>
- [5] <http://www.abclinuxu.cz/clanky/navody/freebsd-v-malej-firme-3-raid-a-xserver>
- [6] <http://www.abclinuxu.cz/clanky/navody/freebsd-v-malej-firme-2-instalacia>

Integrace linuxového serveru do domény Windows 2003

Jaromír Kučera

Miniseriál pojednává o integraci linuxového serveru do domény Windows2003 tak, aby sloužil jako NFS souborový server pro stanice s UNIXem a zároveň jako souborový server se Sambou a ověřováním přístupu z Active Directory (AD) s PDC na Windows 2003.

Úvod, distribuce, prostředí

Přístupová práva k souborům požadujeme podle normy POSIX [1]. Vedlejším cílem bude přiblížení se k Single Sign-On mezi Linuxem a MS Windows. Ač předpokládáme, že server bude sloužit i pro NFS, nebudeme se jeho konfigurací zabývat.

Popisovaný server je postaven na SUSE LINUX Professional 9.2. Pro server byla vybrána distribuce SUSE Professional 9.2 především z toho důvodu, že stanice jsou na SUSE provozovány také, mám s ní největší zkušenosti, je docela slušně připravená na použití autentizace přes Kerberos, má dostatečně aktuální Sambu a je možné ji dobře updatovat z Internetu. Pokud si budete chtít zvolit jinou distribuci, doporučuji použít nějakou nepřiliš starou. (Článek jsem napsal před pár měsíci a před publikací na abclinuxu.cz jsem jej nechal uležet v naivní víře, že uzraje. Neuzrálo však ani písmenko, ale protože si redakce přála rozdělení článku na části, tak jsem jej přece jen mírně doplnil. Není o posledních dostupných distribucích. Naštěstí SUSE 10.0 se konfiguračně přiblížila níže popsanému Debianu.)

Články zaměřené pouze na jednu distribuci (zvláště pokud to není ta moje), mívají občas tu chybu, že se tváří, že všechno funguje právě tak, jak je v nich napsáno. Jenže vy potřebujete spravovat zrovna jinou distribuci a obvykle vám trvá i několik hodin, než najdete ekvivalentní konfigurační soubory mezi popisovanou a vaší distribucí. Dobré je, že si tím rozšíříte znalosti o obecných principech. Rozhodně vás odlišnosti nepotěší, když chcete někomu rychle ukázat, že řešení na Linuxu je jednoduché a výhodnější než řešení na systémech s méně distribucemi. Zabrousím proto i do konfigurace v Debianu 3.1 Sarge a jiných distribucí, které moc neznám, ale nainstaloval jsem si je pro srovnání. I když jsem server postavil na SUSE, Debian se mi ve výsledku jevil jako lepší volba. Uvidíte, že především konfigurace PAM má do jednotnosti dost daleko. Nebudeme se zabývat žádným překladem použitého software, použijeme pouze balíčky z distribucí nebo jejich aktualizace. To bohužel vyloučí například Mandriva Limited Edition 2005 z plnohodnotného použití Samby včetně rozšířených přístupových práv k souborům (Access Control List – ACL).

V konstrukční kanceláři, která má více než sto PC s operačními systémy (OS) od Microsoftu, několik desítek stanic s OS Irix a Solaris a okolo deseti PC s OS SUSE Linux, jsme se rozhodli postavit fileserver s Linuxem sloužící pro všechny operační systémy. Možností, jak zkonfigurovat služby je značné množství. Tady se budeme věnovat popisu konfigurace, nasazené do reálného provozu.

Na konec úvodu ještě poznámka o bezpečnosti: pokud jste paranoidní administrátor se slabší povahou, pak tento článek není pro vás. Popisovanou konfiguraci doporučuji nasadit pouze na serveru ve vnitřní firemní síti, která je od Internetu oddělena firewallem (stejně by asi málokoho napadlo Sambu vystavit do Internetu). Pokusím se upozornit na bezpečnostní problémy tam, kde na ně téměř jistě narazíte, a budu vám vděčný za jakoukoliv radu, jak je řešit (s výjimkou vytažení síťového kabelu z počítače). Řešení klade větší důraz na sjednocení databáze uživatelů než na bezpečnost.

Základní instalace serveru

Server budeme instalovat minimalisticky tak, aby uměl právě to, co od něj vyžadujeme. Jakýkoliv další balíček přidáváme teprve v okamžiku, kdy zjistíme, že nám na serveru chybí. [Filesystém](#) [2]

volíme tak, aby umožňoval rozšíření přístupových práv podle POSIX (XFS [3], ReiserFS [4], Ext3 [5]). Jak bylo řečeno v úvodu, server se nachází ve vnitřní firemní síti za firewallem a popis možného zabezpečení proti „vnitřnímu“ nepříteli je mimo záměr tohoto článku (kdybych se k tomu někdy odhodlal, určitě by to vydalo na další článek). K základním instalačním balíčků musíme přidat další tak, abychom měli k dispozici:

- Kerberos klient a utility
- Open LDAP
- PAM
- NSS
- Sambu
- ACL utility (pro rozšířená přístupová práva)
- NTP klient (nebo jinak zajistit synchronizaci hodin)

Při vyhledávání balíčků v naší distribuci se nám budou hodit následující klíčová slova: *kerberos*, *krb5*, *ldap*, *nss*, *pam*, *nfs*, *acl*, *samba*, *ntp*. Konkrétně v SUSE 9.2 se jedná o balíčky *heimdal-lib*, *heimdal-tools*, *ldapcpplib*, *openldap2-client*, *nss_ldap*, *pam-modules*, *pam*, *pam_krb5*, *samba-client*, *samba*, (*samba-winbind*), *acl*, *xntp*, *pam_ldap* (poslední není v SUSE nezbytný). Pokud si budete překládat některou část softwaru sami, právě tato kouzelná slovíčka budete muset předkládat příkazu `configure` obvykle s prefixem `with`.

Server bude nakonfigurován jako `mujlinux.windom.diiradu.cz`, ověřování bude zajišťovat Windows 2003 PDC server `mujw2k3pdc.windom.diiradu.cz` s IP adresou [6] `10.8.8.1`, jméno domény je zřejmě `windom.diiradu.cz` a jméno windows domény je `windom`. (Pro testování s oblibou používám VMware. Proto jména počítačů, ze kterých kopíruji příklady, mohou začínat `vmw...`) Větami v první osobě jednotného čísla se snažím oddělit subjektivní zkušenosti a názory od obecných pravidel. Další popis bude veden maximální snahou o dodržení správného pořadí konfigurace potřebných služeb. Synchronizaci času mezi počítači budeme předpokládat, ale popisovat ji nebudeme. Proto začínáme konfigurací Kerberos klienta.

Ověřování přes Kerberos

Kerberos je síťový autentizační protokol, který umožňuje ověření totožnosti přes nedůvěryhodnou síť na vzdáleném serveru. Protokol vznikl na MIT a jeho mírně přepracovanou verzi (jak jinak :-)) používá Microsoft v AD. Současné unixové implementace jsou naštěstí s tou od MS kompatibilní.

Nainstalujeme si Kerberos klienta a Kerberos utility. Konfigurace Kerberos klienta je uložena v souboru `/etc/krb5.conf`, do kterého je nutné zadat především kerberos realm, kerberos server a povolený časový rozdíl mezi systémovými hodinami serveru a klienta (`clockskew`). Ostatní položky, které bude náš konfigurační soubor obsahovat, se budou lišit podle distribuce. Odlišnost je způsobena především tím, že k dispozici jsou dvě různé implementace – Heimdal nebo MIT Kerberos. Náš Heimdal Kerberos v SUSE 9.2 používá následující konfiguraci:

```
[libdefaults]
    clockskew = 600
    default_realm = WINDOM.DIIRADU.CZ

[realms]
WINDOM.DIIRADU.CZ = {
    kdc = mujw2k3pdc.windom.diiradu.cz
    default_domain = WINDOM.DIIRADU.CZ
    kpasswd_server = mujw2k3pdc.windom.diiradu.cz
```

```

}

[domain_realm]
    .WINDOM.DIIRADU.CZ = WINDOM.DIIRADU.CZ

[logging]
    default = SYSLOG:NOTICE:DAEMON
    kdc = FILE:/var/log/kdc.log
    kadmind = FILE:/var/log/kadmind.log

[appdefaults]
pam = {
    ticket_lifetime = 1d
    renew_lifetime = 1d
    forwardable = true
    proxiable = false
    retain_after_close = false
    minimum_uid = 0
    debug = false
}

```

Všechny názvy zapsané v konfiguračním souboru velkými písmeny musí být napsány opravdu velkými písmeny! SUSE, Debian, Mandrake i Aurox bez problémů vytvoří tento soubor konfiguračními utilitami (a někdy nám tam nechají i konfigurace realmů, které nepotřebujeme). V tomto okamžiku provedeme test správné funkce Kerberos klienta pomocí příkazu `kinit` případně `kinit mujlogin`. U Debianu, který používá primárně MIT Kerberos, doporučuji použít i parametr `-V`, který učiní `kinit` mírně upovídaným. Budeme dotázáni na heslo a po jeho zadání nám počítač s Heimdal klientem potvrdí přijetí ticketu:

```

jkucera@mujlinux:~> kinit
jkucera@WINDOM.DIIRADU.CZ's Password:
kinit: NOTICE: ticket renewable lifetime is 1 week

```

U Debianu bez parametru `-V` mlčení znamená souhlas. Pokud vše proběhlo tak, jak je výše popsáno, můžeme pokračovat v další konfiguraci.

PAM

Kerberos nám funguje, ale do systému se obecně ještě nepřihlásíme. K tomu potřebujeme patřičnému PAM modulu říct, že má Kerberos pro přihlášení použít. PAM je obecným rozhraním mezi různými autentizačními mechanismy a službami systému, které autentizaci vyžadují. PAM se obvykle instaluje z několika balíčků. Kromě základu budeme potřebovat modul pro spolupráci s Kerberem `pam_krb5` a možná `pam_ldap` modul. Konfiguraci PAM provádíme v adresáři `/etc/pam.d` a `/etc/security`. Ve způsobu konfigurování se různé distribuce liší.

Obecně platí, že přihlašování a s ním spojené zjišťování potřebných uživatelských údajů se konfiguruje v adresáři `/etc/pam.d` zvlášť pro každou službu, která přihlašování vyžaduje. Například soubor `login` je používán při přihlašování se z lokální konzole a soubor `sshd` je používán při vzdáleném

přihlašování se přes ssh. Protože část mechanismu přihlašování pro většinu služeb obsahuje stejné kroky, používají distribuce Linuxu způsob, jak společnou část konfigurovat pouze jednou v jednom souboru. Bohužel právě tady se distribuce značně liší. PAM modul je důležitou bezpečnostní součástí instalace a měli bychom mu věnovat dost velkou pozornost. V testovací fázi stačí s pomocí příkladů dosáhnout nějaké funkční konfigurace. V případě ostrého nasazení si určitě pročtete manuály k jednotlivým modulům a konfiguraci doladíte. (Bližší vysvětlení k PAM najdete např. v [Linux-PAM-html](#) [7] a [pam_unix2](#) [8].) V SUSE je používán pam modul `pam_unix2.so`, který umí zprostředkovat práci s modulem Kerberosu. V Debianu (a většině dalších) má podobnou funkci `pam_unix.so`, který s Kerberosem nepracuje, a je proto potřeba doplnit další modul `pam_krb5.so`. V SUSE editujeme pouze soubor `/etc/security/pam_unix2.conf` do následující podoby:

```
auth: use_krb5 nullok
account:          use_krb5
password:         use_krb5 nullok
session:         none
```

V Debianu (a SUSE 9.3) měníme čtyři začleňované soubory `/etc/pam.d/common-...` (viz poznámka na prvním řádku každého souboru):

```
# /etc/pam.d/common-account
# autorizacni nastaveni spolecne vsem sluzbam

account sufficient      pam_ldap.so
account required       pam_unix.so
```

Bez `pam_ldap.so` mě nechtěl (s použitím údajů z AD) Debian přihlásit.

```
# /etc/pam.d/common-auth
# autentifikacni nastaveni spolecne vsem sluzbam
# pam_unix je tradicni Unix mechanismus.
#
auth    sufficient      pam_unix.so nullok_secure
auth    required       pam_krb5.so use_first_pass
```

```
# /etc/pam.d/common-password
# zachazeni s hesly -- nastaveni spolecne vsem sluzbam
#
password required      pam_unix.so nullok obscure min=4 max=8 md5
password required      pam_krb5.so try_first_pass
```

```
# /etc/pam.d/common-session -- session
# -- nastaveni spolecne vsem sluzbam
#
session required      pam_unix.so
session required      pam_mkhomedir.so skel=/etc/skel/umask=0022auth
```

V Mandrake, Aurox a Fedora hraje podobnou roli jako začleňované soubory v Debianu soubor `/etc/pam.d/system-auth`. I když jsem zmíněné tři distribuce testoval, s laděním jsem se moc nena-máhal.

```

#%PAM-1.0

auth      required      pam_env.so
auth      sufficient    pam_unix.so likeauth nullok
auth      required      pam_krb5.so use_first_pass
auth      required      pam_deny.so dump

account   sufficient    pam_unix.so
account   required      pam_deny.so dump

password  required      pam_cracklib.so retry=3 minlen=4 dcredit=0 ucredit=0
password  sufficient    pam_unix.so nullok use_authtok md5 shadow
password  required      pam_deny.so # i tady by mel byt pam_krb5.so

session   required      pam_limits.so
session   required      pam_unix.so
# pokud chcete, doplnte sem pam_mkhome.so

```

Rozdílné proti standardní konfiguraci jsou pouze řádky s `krb5`, `ldap` u Debianu a `pam_mkhome` (poslední dva nás budou zajímat až v příští části). Proti konfiguracím, které nám distribuce vytvoří sama hned po instalaci, ještě přepíšeme v místech, kam přidáme Kerberos modul, původní záznam `required pam_unix.so` na `sufficient pam_unix.so` a ostatní asi budete moct ponechat nezměněné. Pouze pro úplnost si ukažme ještě příklad souboru `/etc/pam.d/login` ze SUSE, který se od základní konfigurace liší jen přidáním `pam_mkhome`.

```

#%PAM-1.0
auth      requisite    pam_unix2.so      nullok #set_secrcp
auth      required      pam_securetty.so
auth      required      pam_nologin.so
# auth    required      pam_homecheck.so
auth      required      pam_env.so
auth      required      pam_mail.so
account   required      pam_unix2.so
password  required      pam_pwcheck.so   nullok
password  required      pam_unix2.so     nullok use_first_pass use_authtok
session   required      pam_unix2.so     none # debug or trace
session   required      pam_mkhome.so   skel=/etc/skel/ umask=0022
session   required      pam_limits.so
session   required      pam_resmgr.so

```

Pokud by naše nároky na Active Directory končily u společného hesla v MS a Unix síti, mohli bychom tady skončit. Za předpokladu, že dobře funguje standardní lokální přihlašování, si to můžeme vyzkoušet tak, že vytvoříme např. uživatele `jkucera` v Active Directory (zatím bez podpory unixových

položek) a uživatele se stejným loginem v Linuxu. V `/etc/shadow` vyblokujeme jeho účet vložení `*` místo zašifrovaného hesla do druhého pole a zkusíme se přihlásit. Místo `/etc/shadow` by měl být použit Kerberos z PDC a přihlášení by se mělo zdařit (v opačném případě musíme nahlédnout do `/var/log/messages`, `/var/log/?dledistribuce?` a problém řešit). Příště se podrobněji zmíníme o položce `pam_mkhome.so`. Pokud ji v konfiguraci PAM máme, nemusíme pro uvedený test v Linuxu plnohodnotně vytvářet uživatele, stačí nám dopsat jeho záznam do `/etc/passwd`.

```
jkucera:x:1152:100:Kucera Jaromir:/home/jkucera:/bin/bash
```

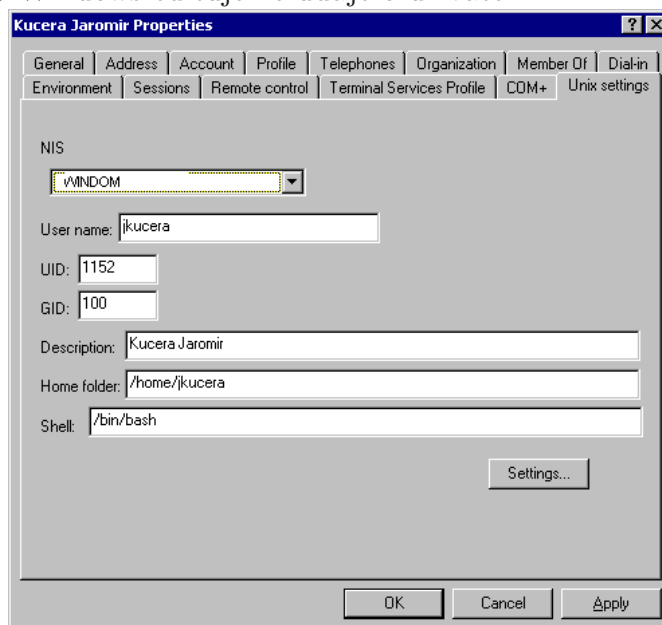
Při uvedeném testu se pokusný uživatel nejen přihlásí, ale automaticky se mu vytvoří i domovský adresář. K ladění nefungujícího PAM použijeme také možnosti dopsat parametr `debug` za odpovídající pam modul do konfiguračního souboru např.

```
auth      required      pam_krb5.so use_first_pass debug
```

Nakonec ještě poznámka ke změně hesla. Heslo bychom měli měnit příkazem `kpasswd`. Jde sice použít i klasické `passwd`, ale chce si to vyzkoušet. V SUSE se mi `passwd` chovalo normálně tak, že změnilo heslo na Kerberos serveru, ale při nevhodném nastavení systému ořezalo jeho délku. V Mandrake šlo `passwd` do chyby, kterou jsem nezkoušel řešit. Domnívám se, že ne vše se vždy podaří vyřešit dobrým nastavením PAM sekce `password`, která změnu hesla řídí.

Konfigurace společné uživatelské databáze v AD

Při větším počtu linuxových stanic není výhodné, aby údaje o uživateli obsahovala každá zvlášť. Proto k centralizaci hesel přidáme i centralizaci údajů, které jsou obvykle uloženy v `/etc/passwd`, NIS nebo na LDAP serveru. Jedná se o login, uživatelské číslo, číslo jeho primární grupy, domovský adresář, shell. MS Windows server ukládá (nejen) uživatelské údaje do LDAP databáze, která je základem AD. Pro popis databáze se používá *schema* (terminus technicus). Schema použité Microsoftem není kompatibilní s normou POSIX [9] používanou ve světě UNIXu [10]. Naštěstí lze toto schema rozšířit do dobře použitelné podoby. Rozšíření schematu je zdarma dodáváno přímo Microsoftem jako součást *Services for Unix* [11] (SFU). Při instalaci SFU se nám toho nainstaluje ovšem víc, než bychom si přáli, a proto zde existuje alternativa v podobě balíčku *AD4UNIX* [12] a jeho binární podoby *MKSADPlugins.msi* [13], kterou je potřeba na MS Windows PDC nainstalovat. Plugin obsahuje kromě rozšíření schematu i doinstalování záložky *Unix Settings* do menu *Active Directory Users and Computers*, kterým v MS Windows editujeme údaje o uživateli.



Ve firmě jsme si rozšíření nechali v rámci migrace z Windows NT na 2003 udělat našim dodavatelem MS Windows 2003 řešení a při tom jsem se dozvěděl, že toto rozšíření schematu je krok **nevratný**. Doporučuji při této operaci postupovat s rozmyslem a dělat ji teprve v okamžiku, kdy víme, co děláme. Pro milovníky novinek od Microsoftu uvedme, že rozšíření schematu je součástí nového vydání Windows Server, 2003 R2.

Moje vlastní zkušenost s pluginem je taková, že spolehlivě nainstaluje rozšíření schematu, což jsem si vyzkoušel na dvou testovacích čerstvě nainstalovaných Windows 2003 PDC na VMware. Zkušební instalace serverů neobsahovala žádný update. Plugin obvykle první nainstaluje administrátorskou záložku pro editaci *Unix Settings* a pak se každém případě zeptá před nainstalováním rozšíření schematu, zda to opravdu chceme. Bohužel v případě našeho PDC, BDC a Exchange serveru záložku *Unix Settings* do menu nenainstaloval. Nainstalovat funkční záložku se mi povedlo pouze na samostatném serveru v doméně a na Win XP s administrátorskými nástroji. Na druhou stranu na zkušebním PDC serveru se záložka bez nejmenších problémů nainstalovala, ale dokud jsem nezadal alespoň nějaké relevantní UNIXové uživatelské údaje do AD jinou cestou, způsobovala chybu aplikace *Active Directory Users and Computers*. Naštěstí odinstalace pluginu nemá žádný vliv na již rozšířené schéma a díky volitelnosti rozšíření schematu se lze později pokusit o přeinstalaci. V případě, že se nám nepovede záložku *Unix Settings* nainstalovat, nezbyvá nám nic jiného, než údaje doplnit z Linuxu pomocí utility `ldapmodify`. Jak na to, si ukážeme později.

Konfigurace LDAP klienta

Linuxový server musí mít někde nastaveno, že se má do AD (LDAP databáze) dívat, a musí vědět, kde LDAP server najde, a jak se jej má zeptat. První údaje se dozví ze souboru `/etc/nsswitch.conf`, druhé ze souboru `/etc/ldap.conf`. Samozřejmě předpokládáme, že LDAP klienta máme nainstalovaného. Během konfigurace budeme mít na paměti, že lokální uživatel *root* v Linuxu a uživatel *administrator* ve Windows jsou dva zcela nezávislí uživatelé a pro jistotu nebudeme konfigurovat žádnou korespondenci mezi nimi. NSS je mechanismus, který umožňuje různým funkcím systému hledat jména a jim příslušející údaje v různých zdrojích, které jsou vyjmenovány, včetně pořadí právě v souboru `/etc/nsswitch.conf`. Většinu údajů necháme nezměněnu, pouze změníme položky `passwd` a `group` tak, že k nim přidáme `ldap`. Tady je příklad mého `nsswitch.conf`:

```
# passwd: files nis
# shadow: files nis
# group:  files nis

passwd: compat ldap
group:  compat ldap
# zbytek je sice nutny, ale nas uz nezajima

hosts:          files lwres dns
networks:       files dns

services:       files
protocols:      files
rpc:            files
ethers:         files
netmasks:      files
netgroup:       files
publickey:      files
```

```
bootparams:    files
automount:    files nis
aliases:      files
```

V příkladu jsme záměrně ponechali i zapoznámkové položky. Pokud ve vašem původním souboru `nsswitch.conf` bude `compat` místo `files`, ponechte tam původní hodnotu a pouze přidejte `ldap`. Kvůli lokálním uživatelům nemusíme zapoznámkovávat `shadow` (jak bylo zmíněno v minulém dílu), příkaz `passwd` se nemusí chovat podle očekávání).

Než se pustíme do konfigurace LDAP klienta, vyzkoušíme přístup k LDAP databázi na Windows serveru. Použijeme k tomu příkaz `ldapsearch` s parametry tak, že nebudeme potřebovat konfigurační soubor. Zadejte (vše na jeden řádek)

```
ldapsearch -x -W -h 10.8.8.1 -b "cn=users,dc=windom,dc=diiradu,dc=cz" \\  
-D "cn=Kucera Jaromir,cn=users,dc=windom,dc=diiradu,dc=cz" \\  
"sAMAccountName=*" sAMAccountName
```

Parametry najdete v `man ldapsearch`. Za prepínačem `-D` si všimněte

```
cn=Kucera Jaromir,cn=users
```

ke kterému se váže heslo a nevyskytuje se tam login (ten si necháváme vypsat). První výskyt `cn=users` nemusíte měnit, druhý výskyt je závislý na vaší implementaci a může vypadat třeba takto: `ou=admini`. Paranoiky upozorňuji na fakt, že komunikace není šifrovaná. Proto jsme raději nepoužili v téměř každé instalaci přítomný záznam `"cn=Administrator"`. Výstup vede na zhruba něco takového (v prvním řádku zadáváme heslo):

```
Enter LDAP Password:  
# extended LDIF  
#  
# LDAPv3  
# base with scope sub  
# filter: sAMAccountName=*  
# requesting: sAMAccountName  
#  
  
# AD Reader, Users, windom.diiradu.cz  
dn: CN=AD Reader,CN=Users,DC=windom,DC=diiradu,DC=cz  
sAMAccountName: adreader  
  
# Administrator, Users, windom.diiradu.cz  
dn: CN=Administrator,CN=Users,DC=windom,DC=diiradu,DC=cz  
sAMAccountName: Administrator  
.  
.  
.  
# Zak Ivan, Users, windom.diiradu.cz
```

```
dn: CN=Zak Ivan,CN=Users,DC=windom,DC=diiradu,DC=cz
sAMAccountName: izak

# search result
search: 2
result: 0 Success

# numResponses: 395
# numEntries: 394
```

Přirozeně můžete dostat nejrůznější chybové hlášky. `Invalid credentials` znamená, že je něco špatně v řetězci za parametrem `-D` nebo heslo. Taky může být problém ve firewallu, který neotevře ldap port na Windows serveru. V každém případě, dokud chybu nenajdete, nemá smysl pokračovat dál.

Teď se vrátíme ke zmíněnému doplnění údajů uživatele pomocí `ldapmodify`. Předpokládejme, že již máme v AD zavedeného uživatele `"Kucera Jaromir"`. Údaje ekvivalentní řádku v `/etc/passwd` vyžadované UNIXem doplníme tak, že vytvoříme soubor `jkucera.ldif` (na jméně nezáleží) s následujícím obsahem:

```
dn: cn=Kucera Jaromir,cn=users,dc=windom,dc=diiradu,dc=cz
changetype: modify
add: uidNumber
uidNumber: 1152
-
add: gidNumber
gidNumber: 100
-
add: msSFUName
msSFUName: jkucera
-
add: gecos
gecos: Kucera Jaromir
-
add: msSFUHomeDirectory
msSFUHomeDirectory: /home/jkucera
-
add: loginShell
loginShell: /bin/bash
-
```

a jeho obsah přidáme do databáze AD příkazem (vše na jednom řádku)

```
ldapmodify -x -W -h 10.8.8.1 -D \\
"cn=Administrator,cn=users,dc=windom,dc=diiradu,dc=cz" -f jkucera.ldif
```

Uživatel zapisující do AD musí mít práva administrátora, proto použijeme "cn=Administrator" za přepínačem `-D`. Pokud si znovu vypíšeme pomocí `ldapsearch` uživatelské údaje, měli bychom mezi nimi najít i právě zadané.

Všechno funguje skvěle :-)) a můžeme editovat `/etc/ldap.conf`. Soubor `ldap.conf` najdete na svém počítači pravděpodobně i v adresáři `/etc/openldap/`, tento soubor není pro účely NSS a PAM používán. Konfigurační údaje v něm uvedené používají LDAP utility jako již dříve zmíněná `ldapsearch`. Debian používá dva různé soubory – `/etc/libnss-ldap.conf` pro NSS a `/etc/pam_ldap.conf` pro PAM. V našem případě budou oba stejné a dokonce budou stejné jako `ldap.conf` v SUSE. Konečně se dostáváme ke komentovanému souboru:

```
# /etc/ldap.conf
# ldap servery PDC a BDC (staci jeden)
host 10.8.8.1 10.8.8.3
base dc=windom,dc=diiradu,dc=cz
ldap_version 3
# k~bezpecnosti prihlasovacich informaci (2 radky) se vratime
binddn cn=AD Reader,cn=users,dc=windom,dc=diiradu,dc=cz
bindpw "heslo_v_citelne_podobe"
scope sub
# ssl proti windows AD se mi nepovedlo rozbehnout
ssl no

pam_filter objectclass=user
pam_login_attribute sAMAccountName
pam_password ad

# Nasledujici tri radky hodne zavisi na cleneni uzivatelu
# do ruznych organizacnich jednotek (OU) v~ramci AD.
# Mozna budete muset vsude dat DC=windom,DC=diiradu,DC=cz
# coz povede k~hledani v~ramci cele databaze.
nss_base_passwd DC=windom,DC=diiradu,DC=cz
# když pouzivame Kerberos nepotrebujeme shadow
# nss_base_shadow CN=Users,DC=windom,DC=diiradu,DC=cz
nss_base_group CN=Users,DC=windom,DC=diiradu,DC=cz
# Na zbyvajicich radcich prirazujete POSIX parametrum pouzivany
# v~UNIXu parametry pouzivane v~AD ve Windows
nss_map_objectclass posixAccount User
# MS SFU definuji pro uid msSFUName, nicmene nasledujici volbu
# povazuji za stastnejsi.
nss_map_attribute uid sAMAccountName
nss_map_attribute uniqueMember Member
nss_map_attribute userPassword msSFUPassword
nss_map_attribute homeDirectory msSFUHomeDirectory
nss_map_objectclass posixGroup Group
nss_map_attribute cn sAMAccountName
```

Přihlašovací informace na řádcích s `binddn` a `bindpw` jsou v `ldap.conf` veřejné a ani zprovoznění SSL nás od jejich zveřejnění nechrání, protože `ldap.conf` musí být podobně jako `/etc/passwd` čitelné pro všechny. Do LDAP databáze AD máme možnost dát anonymní přístup nebo podobně jako v našem příkladě vytvořit uživatele, který má ve Windows minimální práva. V našem případě je to `AD Reader` ve skupině `Domain Guests`. Pro otevření anonymního přístupu nebo pro zajištění přístupu do AD pouze pro čtení použijeme na Windows 2003 serveru administrátorskou aplikaci *Active Directory Users and Computers* a pravým tlačítkem myši na doméně vyberte *Delegate Control*. V otevřeném wizardu si vybereme to, co nám nejlépe vyhovuje.

Pokud se nám povedlo nainstalovat výše zmíněný plugin AD4UNIX, můžeme na Windows serveru doeditovat UNIXové údaje uživatelů nebo, pokud si dobře rozumíme s LDAP utilitami, nebude nám určitě dělat problém napsat si skript pro zápis unixových uživatelských údajů do AD. Uživatelé se pak budou moci přihlásit k našemu linuxovému serveru. Chybí nám ovšem ještě způsob, jak vytvořit domovský adresář uživatele na serveru. Bohužel to asi budeme muset udělat nezávisle na vytvoření uživatele v AD. Pokud vše, co bylo popsáno výše, aplikujeme na pracovní stanici, máme k dispozici možnost, jak domovský adresář vytvořit při prvním přihlášení uživatele (nejen) na lokální textové konzole. V souboru `/etc/pam.d/login`, to zajistí řádek:

```
session    required    pam_mkhomedir.so skel=/etc/skel/ umask=0022
```

Pokud chceme neexistující domovský adresář vytvořit i při jiném způsobu přihlášení, musíme uvedený řádek přidat do všech odpovídajících souborů nebo jej dát do společného souboru, jak je to uvedeno v příkladu `/etc/pam.d/common-session` pro Debian. V adresáři `/etc/skel/` musí být umístěna šablona domovského adresáře (a téměř jistě ji tam najdeme). Pokud vytvoření domovského adresáře nezajistíme, uživatel bez domovského adresáře se může přihlásit s omezenými právy třeba do adresáře hlavního `/` (např. při použití `ssh`), což asi není žádoucí. Je to problém, kterému budeme muset přizpůsobit konfiguraci, abychom dostali uspokojivý výsledek pro každou možnost vzdáleného přístupu na server, která s domovským adresářem počítá. Než přistoupíme k testování, doporučuji pro ladění systému vypnout `nscd` démon (`/etc/init.d/nscd stop`), protože jinak mohou být získávány z jeho cache neaktuální údaje. Nemusíme pak démon po každé změně restartovat. Dokumentace od Samby obecně doporučuje při použití `winbind` nepoužívat `nscd` vůbec.

Konečně vše, co potřebujeme pro přihlášení uživatele definovaného pouze v AD na Windows serveru, máme připraveno a můžeme jej zkusit přihlásit z lokální textové konzole. Správnou funkci NSS a `ldap` lze ověřit i tak, že vytvoříme pod `rootem` třeba soubor `pokus` a přiřadíme jej vlastníkovvi `testuser` definovanému pouze v AD s `id (uidNumber)` např. `54321` (`touch pokus; chown 54321 pokus`). `ls -l` by pak mělo správně vypsat

```
-rw-r--r--  1 testuser root          0 2005-06-18 21:59 pokus
```

K ladění nefungujícího PAM použijeme obsah souborů ve `/var/log/` a možnosti dopsat parametr `debug` za odpovídající pam modul do konfiguračního souboru. Např.

```
auth    required    pam_krb5.so use_first_pass debug
```

případně použijeme `Ethereal` pro sledování paketů na síťové kartě a grafickou utilitu `GQ` [14] pro přístup k LDAP. I když vývoj `GQ` ještě neopustil testovací fázi (určitě si toho všimnete), utilita umožní přehledné prohlížení LDAP databáze na PDC včetně schématu. V okamžiku, kdy se uživatel definovaný pouze v AD může přihlásit k linuxovému serveru, nám zbývá pouze konfigurace `Samby`.

SAMBA a její konfigurace

Pokud chceme zajistit sdílení souborů s operačními systémy od Microsoftu, musíme na našem serveru nainstalovat a nakonfigurovat `Sambu`. `Samba` vyžaduje minimálně demony `nmbd` a `smbd`. Pak

ovšem nefunguje jednoznačně korespondence mezi UNIXovým uid a windowsovým SID a zacházení s přístupovými právy k souborům. Toto nevádí, pokud nám stačí velmi omezená přístupová práva k souborům. Ve všech jiných případech v Sambě použijeme ještě démon `winbind`, který danou korespondenci řeší. Dokumentace od Samby [15] je velmi kvalitní a řeší několik různých případů konfigurace integrace UNIXu do MS Windows sítí. I integrace do AD je pokryta několika možnostmi. My si vybíráme pouze jednu z nich, při které budeme `winbind` používat v minimalistické konfiguraci potřebné pro sdílení souborů a používání ACL. Tato konfigurace potřebuje vše, co jsme si popsali v předchozích dílech, ale nenutí nás vytvářet žádnou konstrukci (pomocnou databázi) navíc. Osobně toto řešení považuji za logicky nejelegantnější a to i přesto, že oficiální dokumentace jej považuje za „nedotažené, ale často dobře fungující“ (The use of this method is messy. The information provided in the following is for guidance only and is very definitely not complete. This method does work; it is used in a number of large sites and has an acceptable level of performance).

Pokud máte se Sambou nějaké zkušenosti, je konfigurace poměrně jednoduchá. Pokud zkušenosti nemáte, může vás zaskočit extrémně vysoké množství parametrů, jež konfigurační soubor Samby může obsahovat. Konfigurace Samby se obvykle nachází v adresáři `/etc/samba/` a její hlavní část v souboru `/etc/samba/smb.conf`. Podívejme se teď na jeho komentovaný příklad:

```
# smb.conf je hlavni konfiguracni soubor Samby
; i strednik uvozuje radek s~komentarem
# nasleduje hlavni spolecna sekce
[global]
    realm = WINDOM.DIIRADU.CZ
    password server = mujw2k3pdc
    security = ADS
    encrypt passwords = yes
    workgroup = WINDOM
    interfaces = 127.0.0.1 eth0
    bind interfaces only = true
# tiskarny konfigurovat nebudeme, ale ponechame obvykle hodnoty
# pro pripad jejich pridani
    printing = cups
    printcap name = cups
    printer admin = @ntadmin, root, administrator
# Nektere zdroje nabidneme i uzivatelum bez linuxoveho uctu,
# ktery se bude pripojovat jako nobody
    map to guest = Bad User
    username map = /etc/samba/smbusers
    passdb backend = smbpasswd
    server string = Samba Server
    domain master = false
    domain logons = no
    local master = no
    acl compatibility = Auto
    preferred master = no
    wins server = 10.8.8.1
# jak uz jsme poznamenali, tiskarny nekonfigurujeme
```

```

load printers = no
# nasledujici parametr je zapoznamkovany, aby bylo explicitne
# videt, ze se nepouziva
# ldap suffix = dc=windom,dc=diiradu,dc=cz
# nasledujici dva parametry se maji dle samba dokumentace pouzit,
# ale jsou nekdy ignorovany -- vysvetleni je v~textu
# idmap uid = 1000-55000
# idmap gid = 100-55000
template shell = /bin/bash
# Normalni jsou jednotky sekund, puvodni samba v~SUSE 9.2
# vyzaduje stovky sekund
winbind cache time = 5
winbind use default domain = Yes
winbind trusted domains only = Yes
winbind nested groups = Yes
[homes]
comment = Home Directories
valid users = %S
browseable = no
read only = No
guest ok = no
printable = no
[prenos]
path = /home/prenos
read only = No
public = Yes
create mode = 0666
delete readonly = Yes

```

V souboru jsou uvedeny pouze hodnoty odlišné od výchozího nastavení. Pokud chcete znát hodnotu každého parametru, který je popsán v `man smb.conf`, můžete ji získat příkazem `testparm -v`. V sekci `global` definujeme obecné vlastnosti Samba serveru. V následujících sekcích jsou definována jednotlivá sdílení. U nás je to sdílení domovského adresáře, který z MS Windows uvidí pouze uživatel, jemuž adresář patří. Podmínky k vytvoření domovského adresáře najdete v předchozím dílu. Dále sdílíme adresář `/home/prenos`, který je přístupný i pro uživatele v Linuxu neznámé. Jeho vlastníkem je uživatel `nobody` a má vhodně nastavená přístupová práva. Nás zajímá integrace s AD, a proto se budeme věnovat pouze sekci `global`.

Samba je neskutečně variabilní projekt pokrývající velkou škálu možností propojení mezi UNIXovým a microsoftím světem. Vzhledem k vývoji MS platformy je také projektem, který velmi často přidává nové vlastnosti, a je proto nutné číst pozorně dokumentaci tak, abychom věděli, zda námi používaná verze podporuje to, co v `smb.conf` konfiguruje. Slovo *verze* budeme dále chápat ve smyslu úplného značení (např. 3.0.20 je jiná verze než 3.0.21). Výše uvedená konfigurace v podstatě opisuje doporučené nastavení z oficiální dokumentace [Samba- HOWTO-Collection](#) [16] a [Samba-Guide](#) [17], kde nás zajímají části hovořící o *Samba Domain Member* a *Active Directory Membership*. Hypertextové odkazy na tyto kapitoly záměrně neuvádím, protože se mohou měnit s každou verzí Samby. Vyplývá

z toho doporučení číst dokumentaci k právě používané verzi. Naše konfigurace je odvozena z příkladu k *IDMAP and NSS Using LDAP from ADS with RFC2307bis Schema Extension*

```
[global]
workgroup = BOBBY
realm = BOBBY.COM
security = ADS
idmap uid = 150000-550000
idmap gid = 150000-550000
template shell = /bin/bash
winbind cache time = 5
winbind use default domain = Yes
winbind trusted domains only = Yes
winbind nested groups = Yes
```

Příklad ilustruje parametry, které jsou pro nás klíčové. První dva jsou samozřejmé a nevyžadují komentář, o těch ostatních si něco povíme. Naše výše uvedená konfigurace obsahuje navíc některé parametry, jejichž hodnoty nejsou povinné, význam lze najít v manuálových stránkách a autor je použil prostě proto, že mu právě takové na serveru vyhovovaly. Parametr `security` nastavuje ověřování pomocí AD.

`Winbind` démon zajišťuje korespondenci windowsových SID a unixových `uid` a `gid`. (Na rozdíl od jiných typů konfigurací nic dynamicky nevytváří.) Démon čte nastavení parametrů `idmap`, které určují rozsah použitých `uid` a `gid`. Toto nemá kupodivu na Sambu 3.0.14 v Debianu a 3.0.9 v SUSE 9.2 žádný vliv, a proto je to v naší konfiguraci zapoznámkové. Ve stávajících verzích Samby (např. 3.0.20 v SUSE 10.0) jsou však důležité a musí být nastaveny na rozsah, který pro uživatele a grupy používáme v databázi AD. V opačném případě očekávejte potíže s ACL. Předpokládáme samozřejmě, že uživatelské parametry jako `uid`, `shel` atd. máme plně pod kontrolou a přidělujeme je individuálně a nepředpokládáme, že každý, kdo se přihlašuje k MS Windows, se může přihlásit i k linuxovému serveru. `template shell` není proto podstatný.

`winbind cache time` určuje počet sekund, po který se korespondence SID s `uid` a `gid` udržuje v cache Samby a není na ni dotazován PDC. 5 sekund je s výjimkou Samby 3.0.9 v SUSE 9.2 dobré číslo. Dokumentace od Samby doporučuje vypnout `nscd` démon tak, jak to bylo popsáno v předchozím dílu seriálu. Vyhneme se tím rozporům mezi cache obou démonů (`nscd` a `winbind`). Ostatní `winbind ...` parametry jsou doporučenými parametry pro naši konfiguraci. Konfigurační soubor tak, jak je uveden výše, pracuje skvěle v Debianu.

Bohužel v SUSE 9.2 má buď Samba 3.0.9 obsažená v distribuci nějaké mouchy, které se mi nepodařilo odstranit ani s pomocí Google, nebo jsem někde zapomněl něco dokonfigurovat. Samba si neumí na základě dodaného SID zjistit v LDAP (AD) odpovídající `uid`. Umí to pouze tehdy, když si `winbind` těmito informacemi naplní cache. Toho se dá dosáhnout například příkazem `wbinfo -u > /dev/null`. Dostatečně velkým parametrem `winbind cache time` v souboru `smb.conf` pak zajistíme, že si cache údaje pamatuje dlouho. Opakovaným prováděním příkazu je musíme ve vhodném intervalu obnovovat. Toto řešení asi není nic, co by se normálnímu administrátorovi líbilo. Naštěstí existuje poměrně jednoduché východisko.

Na začátku jsme slíbili, že nebudeme nic překládat, což dodržíme pokud si stáhneme ze [SUSE projektu](#) [18] rpm binárky novější verze Samby a nainstalujeme je. Jak už bylo dříve řečeno, pro správnou funkci musíme odpoznámkovat a nakonfigurovat `idmap` parametry. Vyzkoušel jsem to s verzí Samby 3.0.20, což je stejná verze, s jakou je dodávána distribuce SUSE 10.0.

Až dosud jsme předpokládali, že čtenář ví téměř vše o principech spouštění a fungování Samby a potřebuje být pouze popostrčen v konfiguraci `smb.conf` tak, aby odpovídala našim požadavkům

zařazení do Windows 200x domény (AD). Některé distribuce vám občas vnutí automatickou konfiguraci Samby včetně spuštění. Je téměř jisté, že tato konfigurace se námi požadované nebude podobat. Vzhledem k tomu, že už možná máte Sambu nakonfigurovanou a spuštěnou tak, že funguje jiným způsobem, řekneme si, jak ji zbavit všech relikvů staré konfigurace (i když to obvykle není nutné). Pokud demony samby běží, zastavíme je. V SUSE příkazy:

```
/etc/init.d/winbind stop
/etc/init.d/smb stop
/etc/init.d/nmb stop
```

a v Debianu

```
/etc/init.d/winbind stop
/etc/init.d/samba stop
```

Pak vymažeme databázové soubory `*.tdb` v adresářích `/var/lib/samba` a `/etc/samba`. V adresáři `/etc/samba` zkonfigurujeme soubory `smb.conf` a `smbusers` (může zůstat prázdný).

Ještě než spustíme Sambu, musíme provést úpravu na MS Windows PDC (bohužel se jedná o snížení zabezpečení). Samba se zatím nevyrovnala s postupným zabezpečováním SMB protokolu ze strany Microsoftu, a je proto potřeba vypnout „signing“, tj. změnit jeden nebo dva klíče v *Policy* na PDC. Můžete použít následující postup:

- Spusťte *Domain Controller Security Policy* a ve stromu *Security Settings – Local Policies – Security Options* nastavte *Microsoft Network Server: Digitally sign communications (always): DISABLED*.
- A pokud chcete přistupovat na linuxové zdroje i z PDC (a naopak), nastavte ještě *Microsoft Network Server: Digitally sign communications (if client agrees): DISABLED*.

Nakonec vytvoříme (případně přepíšeme) účet počítače v *Active Directory*. Na Samba serveru to obstará příkaz:

```
net ads join -U administrator
```

alternativně

```
kinit administrator
net ads join
```

kde `administrator` je login uživatele oprávněného přidávat počítače do AD (tento AD login obecně nemá žádný vztah k uživateli root na Linuxu). Pak opět spustíme demony Samby v SUSE příkazy (v daném pořadí):

```
/etc/init.d/nmb start
/etc/init.d/winbind start
/etc/init.d/smb start
```

a v Debianu

```
/etc/init.d/samba start
/etc/init.d/winbind start
```

Pokud vše proběhlo podle předpokladů, budou počítače s OS Windows schopny používat zdroje z našeho linuxového serveru a naopak. V naší konfiguraci tak uživatelé MS Windows uvidí na Samba

serveru sdílené položky `prenos` a svůj domovský adresář. Uveďme ještě příklad zavedení počítače se jménem `vmw1x1` do domény:

```
vmw1x1:~ # net ads join -U administrator
administrator's password:
[2005/09/12 14:09:30, 0] libads/ldap.c:ads_add_machine_acct(1400)
  ads_add_machine_acct: Host account for vmw1x1 already exists -- modifying
  old account
Using short domain name -- WINDOM
Joined 'VMWLX1' to realm 'WINDOM.DIIRADU.CZ'
vmw1x1:~ #
```

nebo

```
vmw1x1:~ # kinit administrator
administrator@WINDOM.DIIRADU.CZ's Password:
kinit: NOTICE: ticket renewable lifetime is 1 week
vmw1x1:~ # net ads join
[2005/09/12 14:14:57, 0] libads/ldap.c:ads_add_machine_acct(1400)
  ads_add_machine_acct: Host account for vmw1x1 already exists -- modifying
  old account
Using short domain name -- WINDOM
Joined 'VMWLX1' to realm 'WINDOM.DIIRADU.CZ'
vmw1x1:~ #
```

Přestože Sambu máme zkonfigurovanou, stále ještě neumožňujeme běžným uživatelům MS Windows nastavovat přístupová práva způsobem, na který jsou zvyklí.

Nastavení ACL

Již dříve jsme se zmínili o rozšíření přístupových práv k souborům podle normy POSIX, které obsahuje většina filesystémů používaných v Linuxu. Není to totéž, co mají k dispozici MS Windows, ale stačí to ke splnění požadavků uživatelů tohoto OS. Pokud používáte XFS, tak nemusíte pro použití ACL dělat vůbec nic, protože XFS s ACL pracuje automaticky. ReiserFS a Ext3 vyžadují při mountování zadat parametr `acl`. Například v Debianu zadáte na příslušný řádek v `/etc/fstab`:

```
/dev/sda2 /home ext3 defaults,acl 0 1
```

Pro zadávání a čtení rozšířených práv slouží v Linuxu utility `setfacl` a `getfacl`. Použití je dobře popsáno v manuálových stránkách, neodpustím si však doporučení, abyste se podívali na změněné chování výpisu `ls -l` přístupových práv pro grupu. V OS MS Windows můžete nyní měnit práva ve stejném grafickém rozhraní jako to děláte při změně práv na svazku NTFS. Nemůžete ovšem překročit omezení na `read`, `write`, `execute` dané POSIX normou. Zafajfkovat můžete cokoliv, ale pouze tři zmíněná nastavení se zapíše. Nyní máme k dispozici vše, co jsme si na začátku předsevzali. V popisu vám určitě něco bude chybět, ale to podstatné zde doufám najdete.

Zbývá udělat

Většina projektů obsahuje sekci, která nám dává nahlédnout do plánů autorů, případně do návrhů, co by se mělo udělat, kdyby se našel někdo, kdo to udělá. Když už se namáháme ověřovat uživatele

Samby přes Kerberos, měli bychom to samé udělat i pro NFS (použít AFS). Vzhledem k tomu, že ne všechny distribuce obsahují potřebné utility přeložené s parametry podle našich potřeb, stálo by asi za to vypsát potřebné parametry při jejich překladu. Konečně by bylo vhodné nějak lépe vyřešit zabezpečení dotazů do AD a nepoužívat pro ně uživatele s malými právy a otevřeným heslem případně anonymního uživatele. Jako řešení bych viděl nějakou bezpečnou replikaci významných veřejných údajů z AD do OpenLDAP na Linuxu a následné standardní využití této sekundární databáze UNIXovými klienty.

Odkazy

- [1] <http://www.abclinuxu.cz/slovník/posix>
- [2] <http://www.abclinuxu.cz/slovník/filesystem>
- [3] <http://www.abclinuxu.cz/clanky/system/moderni-souborove-systemy-xfs>
- [4] <http://www.abclinuxu.cz/clanky/system/moderni-souborove-systemy-reiserfs>
- [5] <http://www.abclinuxu.cz/clanky/system/moderni-souborove-systemy-ext3>
- [6] <http://www.abclinuxu.cz/slovník/ip-adresa>
- [7] <http://www.kernel.org/pub/linux/libs/pam/Linux-PAM-html/pam.html>
- [8] http://www.thkukuk.de/pam/pam_unix2/
- [9] <http://www.abclinuxu.cz/slovník/posix>
- [10] <http://www.abclinuxu.cz/slovník/unix>
- [11] <http://www.microsoft.com/windowsserversystem/sfu/default.msp>
- [12] <http://sourceforge.net/projects/ad4unix/>
- [13] <http://www.padl.com/download/MKSADPlugins.msi>
- [14] <http://sourceforge.net/projects/gqclient>
- [15] <http://www.samba.org/samba/docs/>
- [16] <http://www.samba.org/samba/docs/man/Samba-HOWTO-Collection/>
- [17] <http://www.samba.org/samba/docs/man/Samba-Guide/>
- [18] <ftp://ftp.suse.com/pub/projects/samba/3.0/>

Stromy v SQL

Pavel 'lingeek' Szalbot

Stromy lze s pomocí jazyka SQL sázet, kácet i česat vícero způsoby. Ukážeme si jeden klasický a dva pokročilejší.

Definice

Pojmem strom se zpravidla označuje neorientovaný graf, jehož každé dva vrcholy jsou spojeny právě jednou cestou (pokud nerozumíte, zkuste se podívat třeba do wiki – [graf](#) [1], [strom](#) [2]).

Strom může reprezentovat vybranou hierarchickou (stromovou) strukturu a ta je předmětem toho článku. Kde že jste mohli strom mimo park zahlédnout? Zajisté jste se s nimi setkali minimálně v diskusních fórech ABCLinuxu.cz, nebo při nakupování v internetovém obchodě, na jehož kategoriích výrobků si budeme práci se stromovou strukturou prezentovat. Důvodem je několik „zajímavých“ akcí, jež se v internetovém obchodě provádějí. Uložištěm našeho stromu bude RDBMS komunikující jazykem SQL (konkrétně MySQL a v jednom případě i PostgreSQL).

Jaký strom zasadit?

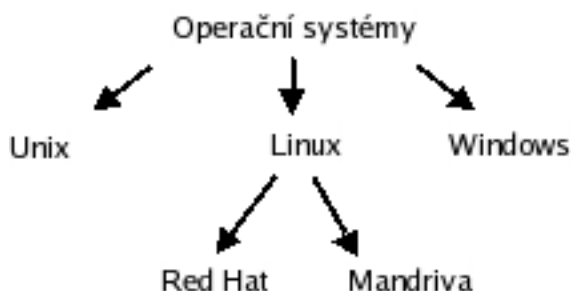
V perexu jsem naznačil, že si ukážeme celkem tři příklady reprezentace stromu v SQL. Než si z nich vůbec začneme vybírat, měli bychom si ujasnit to, co od stromu budeme očekávat. Aplikace typu internetový obchod může provádět následující akce:

- základní práce se stromem – CRUD operace s uzly (kategoriemi) (Create, Retrieve, Update, Delete)
- rozbalení stromu ve vybraném uzlu
- nalezení všech „podkategorií“ dané kategorie (pro různé filtry, ale také zobrazení všech výrobků z kategorie a jejich podkategorií)
- zobrazení cesty k vybranému uzlu od kořene

Operací s hierarchickou strukturou bude pravděpodobně více a mnohé mohou být efektivněji provedeny na aplikační úrovni (např. načtením celé struktury a jejím zpracováním namísto několika SQL dotazů). Ponechme je ale stranou a zkusme se podívat, co nám nabízí samotné SQL. Vzhůru do lesů!

Sebereferenční tabulky

Prvním způsobem uložení hierarchické struktury v SQL tabulce budou tzv. sebereferenční tabulky, definující strom seznamem následníků. Sebereferenční tabulky využívají vazby rodič-syn/dcera v hierarchické struktuře přítomné. Pro jednoduchost budeme předpokládat, že každý uzel má nanejvýš jednoho rodiče. Uvažujme následující hierarchii kategorií:



SQL tabulka by mohla být vytvořena příkazem:

```
CREATE TABLE categories(
  id INT NOT NULL PRIMARY KEY,
  name VARCHAR(32),
  parent INT NOT NULL);
```

a její obsah by vypadal takto:

```
+----+-----+-----+
| id | name           | parent |
+----+-----+-----+
|  1 | Operační systémy |      0 |
|  2 | Unix            |      1 |
|  3 | Linux           |      1 |
|  4 | Windows         |      1 |
|  5 | Red Hat         |      3 |
|  6 | Mandriva        |      3 |
+----+-----+-----+
```

Jak vidíte, každý uzel má unikátní identifikátor (číslo ID) a také jsme mu přiřadili rodiče. Všimněte si, že kategorie „Operační systémy“ má ve sloupci parent nulu, i když žádná taková kategorie v tabulce není. Její syny budeme označovat jako kořenové kategorie a pro ni samotnou nebudeme požadovat rodiče.

Podívejme se teď na operace, které nás při práci se stromovou strukturou budou otravovat.

CRUD operace jsou vcelku triviální. Vkládání zajistí prostý INSERT. Při odebírání bychom měli dbát na to, abychom dle potřeby rekurzivně smazali i uzly-syny, což je v košatém stromě docela náročné. Přesun uzlu (a celého jeho podstromu!) v rámci stromu, tj. změnu jeho rodiče, realizujeme jednoduchým UPDATE.

Zobrazení podstromu provedeme podobně jako smazání – rekurzivně vybereme všechny potomky právě zpracovávaného uzlu. Podotkněme, že výhoda načtení celého stromu a jeho zpracováním aplikací sice ušetří práci databázi, ale nelze ji dost dobře použít při získávání všech výrobků patřících do kategorií podstromu, kterých může být velmi mnoho.

Rozbalení stromu dle vybrané kategorie je variací na téma zobrazení podstromu s tím, že do hloubky jdeme jen po cestě od kořene k rozbalovanému uzlu. Nalezení této cesty je při této reprezentaci znovu náročné.

Předností seberefrenčních tabulky je jejich jednoduchost – při práci s ní si bohatě vystačíte se znalostí rekurze. Cenou ovšem bude neúměrné zatížení databázového serveru zvláště v případě, že se stromem budete pracovat často, což se u internetového obchodu děje obvykle s každou zobrazenou stránkou, či když strom bude hezky košatý (uzly mají mnoho potomků) a vysoký (mnoho generací potomků). Odlehčit si můžete jistou úroveň cachování stránek, nicméně časem se nejspíš poohlédnete po výkonnějším řešení.

Genealogické stromy

A narazíte možná na genealogické stromy. Genealogický strom také využívá vazbu rodič-syn mezi uzly, ale navíc pro každý uzel definuje i tzv. genealogický identifikátor. Tento identifikátor je unikátní pro každý uzel a dají se z něj vyčíst informace o jeho předcích (rodičích, prarodičích, prapra...) i potomcích. Identifikátor potomka totiž získáme tak, že za identifikátor předka připojíme identifikátor

potomka. Zvolíme-li za identifikátor písmeno abecedy, pak bude naše rozšířená tabulka obsahovat tyto záznamy:

id	name	parent	path
1	Operační systémy	0	A~
2	Unix	1	AA
3	Linux	1	AB
4	Windows	1	AC
5	Red Hat	3	ABA
6	Mandriva	3	ABB

CRUD operace tentokrát váže několik nepříjemných podmínek. Jednou z nich je omezení počtu potomků dle volby uložení identifikátoru (v případě písmen abecedy smít uzel mít „jen“ 26 přímých potomků). Vložení nového uzlu do stromu provedeme tak, že zjistíme genealogický identifikátor rodiče a za identifikátor uzlu zvolíme nejmenší možné písmeno abecedy, které je na dané úrovni volné (úroveň rozumíme množinu přímých potomků rodiče). Zavedme proto požadavek, aby na sebe identifikátory sourozenců lexikálně navazovaly.

Odstranění uzlu je velmi snadné. Stačí smazat všechny uzly, jejichž genealogický identifikátor začíná identifikátorem odstraňovaného uzlu. Pokud bychom v naší hierarchii chtěli z nabídky odstranit podstrom s kořenovou kategorií „Linux“, provedli bychom SQL příkaz:

```
DELETE FROM categories WHERE genealogical LIKE 'AB%';
```

Tím nám ovšem může vzniknout mezera na úrovni mazaného uzlu (Linux), což si nepřejeme a musíme proto po smazání uzlu aktualizovat identifikátory postižených uzlů. Přesun uzlu provedeme příslušnou změnou umístění (změna rodiče) a následnou aktualizací identifikátorů.

Poznamenejme, že návaznost identifikátorů se nakonec zdá být spíše na škodu, jelikož nám práci docela komplikuje, nicméně se bez procedury na odstranění hluchých míst ve stromu neobejdeme. Zobrazení kompletního podstromu je poněkud svízelné. Sice nám postačí SELECT s klauzulí `ORDER BY genealogical`, aplikace ovšem často požaduje, aby byl výstup abecedně seříděn. Tuto komplikaci lze vyřešit už během CRUD operací, totiž vkládáním nových uzlů na správné místo. Cenou je bohužel režie spojená s tříděním a následnou aktualizací identifikátorů.

Chcete-li si ušetřit nepříjemnosti s nedostatkem písmen a přepočítáváním po mazání, můžete použít jiný identifikátor. V praxi se často používá např. speciální oddělovač následovaný číselnou sekvencí. Identifikátor uzlu Red Hat by byl „/1/3/5“. Konečně nalezení cesty k uzlu zařídí:

```
SELECT * FROM categories WHERE 'ABB' LIKE genealogical||'%'
nebo
SELECT * FROM categories WHERE 'ABB' LIKE concat(genealogical, '%')
```

Nested set aneb DFS strom

Posledním a dle mého názoru nejvýkonnějším řešením je tzv. *nested set* reprezentace stromu. (Pozn.: Tento název používá Joe Celko a z různých článků se zdá, že není sám. Ačkoli podstatu uložení informace o uzlech charakterizuje čitelně i pro základních grafových algoritmů neznalé, lepší název by mohl být DFS strom.) Podívejme se nejprve na tabulku:

id	name	parent	left	right
1	Operační systémy	0	1	12
2	Unix	1	2	3
3	Linux	1	4	9
4	Windows	1	10	11
5	Red Hat	3	5	6
6	Mandriva	3	7	8

Vychází ze sebereferenční tabulky, kterou rozšiřuje atributy *left* a *right*. Jejich hodnoty jsou získány průchodem stromu DFS (depth first search) algoritmem. Pseudokód algoritmu:

```

DFS(graf)
  foreach uzly_grafu as uzel do
    uzel->barva = bila
  done
  cas = 0
  foreach uzly_grafu as uzel do
    if uzel->barva = bila
      DFS-PROJDI(uzel)
    done
  end

DFS-PROJDI(uzel)
  uzel->barva = seda
  cas = cas + 1
  uzel->nalezen = cas
  foreach soused[uzel] as soused do
    if soused->barva = bila
      DFS-PROJDI(soused)
    done
  uzel->barva = cerna
  uzel->opusten = cas

```

Algoritmus začíná voláním funkce DFS, které je předán zkoumaný graf. Ta nastaví barvu všech uzlů na bílou (uzel dosud nebyl navštíven), seřídí čas a následně prochází uzly grafu s tím, že pokud je uzel bílý, zavolá funkci DFS-PROJDI. DFS-PROJDI přebarví uzel na šedou (byl navštíven, ale dosud se zpracovává), zvedne čas o jedničku a použije jej jako čas navštívení uzlu a poté rekurzivně prochází dosud nenavštívené sousedy uzlu předaného jako parametr. Jakmile jsou všichni sousedé zpracováni, přebarví uzel na černou (zpracování dokončeno), nastaví čas opuštění uzlu a vrací se. Čas navštívení a opuštění uzlu se použijí jako hodnoty atributů *left* resp. *right* v SQL tabulce. Lepší představu o výsledku můžete získat z obrázku.



Všimněte si, že interval $left;right$ libovolného uzlu je podintervalem intervalu vlastního rodiče (odtud nested set = vnořené množiny). Tato vlastnost plyne z toho, jak DFS prochází strom a ukládá časy a právě ona nám ulehčí práci s hierarchickou strukturou v SQL. Podívejme se na operace, které chceme nad strukturou provádět. Všechny podkategorie zvolené kategorie získáme jednoduchým SELECTem:

```
SELECT * FROM categories WHERE left >=x AND right <=y
```

Oproti rekurzi seberefrenčních tabulek podmíněnou mnoha dotazy či spíše přenosem většího množství dat jsme ve výhodě, ovšem genealogický identifikátor umožňuje totéž, ač s nutností použití pomalejšího operátoru LIKE. Získat výstup seřazený podle názvu uzlů je tentokrát o něco jednodušší. Po každé CRUD operaci totiž musíme aplikovat DFS algoritmus na celý strom znovu. Aby DFS generoval časy s ohledem na abecední pořadí uzlů, stačí našťestí jen vhodně připravit pořadí uzlů, ve kterém jsou algoritmem zpracovávány (nezapomeňte abecedně seřadit i pole sousede[uzel]). Cestu k uzlu nalezneme také velmi jednoduše. Stačí si uvědomit, že každý předek uzlu byl navštíven dříve a opuštěn později než uvažovaný uzel.

```
SELECT * FROM categories WHERE left <= x AND right >=y
```

DFS strom se zdá být velmi vhodný pro statické, či málo upravované struktury. Uplatnění si však zajisté najde i v případě potřeby vyhledávání ve velmi rozsáhlých hierarchických strukturách.

Závěr

Předvedli jsme si tři možné reprezentace hierarchické struktury v SQL databázích. Jejich přednosti a nevýhody je třeba zvážit vždy s ohledem na konkrétní využití, přičemž společnou výhodou se jeví zvláště přenositelnost mezi různými databázovými servery. Dle předložených indicií a zdrojů na internetu si zajisté zvolíte správnou reprezentaci pro požadované použití.

Odkazy

- [1] <http://cs.wikipedia.org/wiki/Graf>
- [2] <http://cs.wikipedia.org/wiki/Strom>

Testování služeb mailserveru telnetem

Zdeněk Burda

*Srozumitelný a ukázkový návod, jak si na nižší úrovni popovídat s vaším serverem.
Telnet může posloužit jako univerzální řešení problémů.*

Při instalaci služeb zajišťujících běh (nejen) poštovního serveru se často setkáte s potřebou snadné kontroly funkčnosti služby pomocí jednoduchého programu. Možnost ručně si zadat příkazy a sledovat při tom odezvu systému je většinou neocenitelná. V článku vám ukážu, jak snadno testovat služby POP3, IMAP, SMTP a jejich varianty zabalené do SSL.

Použité nástroje

Pro otestování použijeme běžné nástroje dostupné snad v každé „normální“ distribuci linuxu, a to programy telnet a openssl. Telnet nám poslouží pro připojení na POP3, IMAP a SMTP server. Openssl využijeme pro ověření funkčnosti POP3s, IMAPs a SMTPs. Jako SMTP server používám postfix, o POP3 a IMAP se stará dovecot. Výpisy zde uvedené se můžou mírně lišit podle toho, jaké programy použijete.

Telnet

Program telnet je dostupný v Linuxu (a UNIXech) i v MS Windows, umožňuje interaktivní TCP spojení na server. Spouští se z příkazového řádku:

```
$ telnet SERVER PORT
```

SERVER můžete zapsat pomocí doménového jména nebo IP adresy. PORT je číslo TCP portu a lze ho zapsat buď jako číslo nebo jako textový název, který je danému číslu portu přiřazen v souboru `/etc/services`. Program telnet (a navázané spojení) ukončíte pomocí stisku kombinace kláves `^]` a `^d`.

```
[tsunami@vodik]$ telnet tsunami.locnet pop3
Trying 192.168.0.10\dots
Connected to tsunami.locnet.
Escape character is '^]'.
+OK Dovecot ready.
^]
telnet> Connection closed.
```

Pokud má někdo odpor k aplikacím spouštěným z příkazového řádku, může použít například grafický program PuTTY (Linux, Windows), ale dle mého názoru je to zbytečné.

SMTP

SMTP protokol je relativně jednoduchý protokol založený na textové komunikaci, používá TCP port 25. Původní popis protokolu najdete v [RFC821 \[1\]](#) (z r. 1982), aktualizovaný popis najdete v [RFC2821 \[2\]](#) (z r. 2001). Pro kompletní popis protokolu tu není místo, proto zde uvedu praktickou ukázkou toho, jak se přes telnet pošle email. Všechny tučně zapsané řádky jsou zaslané klientem (mnou)

nařukané do klávesnice). Připojíme se telnetem na server na port 25 a počkáme si na odpověď od serveru.

```
[tsunami@vodik]$ telnet tsunami.locnet 25
Trying 192.168.0.10...
Connected to tsunami.locnet.
Escape character is '^]'.
220 tsunami ESMTP Postfix (Debian/GNU)
```

Nejprve se identifikujeme:

```
EHLO vodik.locnet
250-tsunami
250-PIPELINING
250-SIZE 10240000
250-VRFY
250-ETRN
250 8BITMIME
```

Uvedeme, od koho email pochází:

```
MAIL FROM: od.koho.ten@email.je
250 Ok
```

a komu ho zasíláme

```
RCPT TO: tsunami@tsunami.locnet
250 Ok
```

Potom můžeme začít psát vlastní zprávu, psaní ukončíme pomocí tečky na samostatném řádku a „odentrování“.

```
DATA
354 End data with <CR><LF>.<CR><LF>
Subject: pozdrav od anonyma

Ahoj, jak se mas?
.
250 Ok: queued as 0AA3929FAA
```

Email byl zařazen do fronty a my můžeme ukončit svou práci:

```
QUIT
221 Bye
```

Jak vidíte, docela jednoduše můžete zasílat emaily, i když nemáte vhodného emailového klienta, jako je Mozilla nebo Outlook :-)

POP3

Protokol pop3 slouží ke stažení pošty ze serveru ke klientovi. POP3 přístup k poště je vhodný, pokud potřebujete rozumný přístup k poště přes pomalou linku případně nemůžete/nechcete uchovávat všechnu poštu na serveru. Pop3 je většinou dostupný na portu 110/tcp, je popsán v [RFC1939](#) [3], [RFC2449](#) [4] a [RFC1734](#) [5]. Připojíme se k serveru

```
[tsunami@vodik]$ telnet tsunami.locnet 110
Trying 192.168.0.10...
Connected to tsunami.locnet.
Escape character is '^]'.
+OK Dovecot ready.
```

Přihlásíme se jako uživatel k serveru:

```
USER tsunami
+OK
PASS heslo
+OK Logged in.
```

Příkazem STAT můžeme zjistit počet zpráv a jejich velikost, případně si je příkazem LIST vypíšeme.

```
STAT
+OK 3 1619
LIST
+OK 3 messages:
1 433
2 486
3 700
.
```

Rozhodl jsem se stáhnout zprávu číslo 2.

```
RETR 2
+OK 486 octets
Return-Path: <od.koho.ten@email.je>
X-Original-To: tsunami@tsunami.locnet
Delivered-To: tsunami@tsunami.locnet
Received: from vodik.locnet (vodik [192.168.0.13])
by tsunami (Postfix) with ESMTP id 0AA3929FAA
for <tsunami@tsunami.locnet>; Sat, 7 Jan 2006 23:36:40 +0100 (CET)
Subject: pozdrav od anonyma
Message-Id: <20060107223653.0AA3929FAA@tsunami>
Date: Sat, 7 Jan 2006 23:36:40 +0100 (CET)
From: od.koho.ten@email.je
To: undisclosed-recipients;
```

```
Ahoj, jak se mas?
```

```
.
```

Stažený email označím ke smazání.

```
DELE 2
```

```
+OK Marked to be deleted.
```

Ukončení spojení zároveň znamená i smazání označených zpráv.

```
QUIT
```

```
+OK Logging out, messages deleted.
```

IMAP

Další protokol, kterým se dostaneme ke své poště na serveru, se jmenuje IMAP (Internet Message Access Protocol, aktuálně ve verzi 4v1). Protokol je složitější než POP3 a nabízí o mnoho víc funkcí. První rozdíl, kterého si všimnete, je, že pošta se nechává na serveru. Protokol je také navržen pro současné připojení více klientů k jednomu mailboxu. IMAP funguje přes TCP port 143. Protokol IMAP je popsán v RFC3501 [6], zajímavé informace k IMAPu najdete také na webu www.imap.org [7] Připojíme se na server a přihlásíme se jako uživatel:

```
[tsunami@vodik]$ telnet tsunami.locnet 143
Trying 192.168.0.10...
Connected to tsunami.locnet.
Escape character is '^]'.
* OK Dovecot ready.
. LOGIN tsunami heslo
. OK Logged in.
```

Každý příkaz musí být uveden unikátním identifikátorem v rámci daného sezení. Vypíšeme si, co nám server nabízí za funkce.

```
IDENT01 CAPABILITY
* CAPABILITY IMAP4rev1 SORT THREAD=REFERENCES MULTIAPPEND
UNSELECT LITERAL+ IDLE CHILDREN NAMESPACE LOGIN-REFERRALS
IDENT01 OK Capability completed.
```

Pro další ukázky raději použiju kratší identifikátory. Zkusíme si vypsat dostupné složky.

```
id01 LIST "" "*"
* LIST (\NoInferiors \UnMarked) "/" "Trash"
* LIST (\NoInferiors) "/" "Pracovni"
* LIST (\NoInferiors \UnMarked) "/" "Soukrome"
* LIST (\NoInferiors) "/" "Linux"
* LIST (\NoInferiors) "/" "Solaris"
* LIST (\NoInferiors) "/" "INBOX"
id01 OK List completed.
```

Z předchozího výpisu jsme zjistili, že na serveru existuje několik složek. „Přepneme“ se do složky Linux.

```
id02 SELECT "Linux"
* FLAGS (\Answered \Flagged \Deleted \Seen \Draft)
* OK [PERMANENTFLAGS (\Answered \Flagged \Deleted \Seen \Draft \*)] Flags permitted.
* 2 EXISTS
* 0 RECENT
* OK [UIDVALIDITY 1136680099] UIDs valid
* OK [UIDNEXT 4] Predicted next UID
id02 OK [READ-WRITE] Select completed.
```

Vypíšeme si seznam všech emailů ve složce:

```
id03 FETCH 1:* FLAGS
* 1 FETCH (FLAGS (\Seen))
* 2 FETCH (FLAGS (\Seen))
id03 OK Fetch completed.
```

Stáhneme si email č. 2:

```
id04 FETCH 2 body[text]
* 2 FETCH (BODY[TEXT] {75}
mkdir: cannot create directory '/var/run/sendmail/msp': Permission denied
)
id04 OK Fetch completed.
```

Myslím, že to na ukázání stačilo, takže se odhlásíme.

```
id05 logout
* BYE Logging out
id05 OK Logout completed.
```

SSL

Dnes je téměř samozřejmostí, že admin přehodí většinu veřejně dostupných služeb na jejich zabezpečenou variantu (zabalenou do SSL). Klasickým telnetem v tomto případě nic nevyzkoušíte, protože telnet neumí SSL. Pro připojení na server proto použijeme program openssl. O programu openssl si spousta lidí myslí, že to je ta divná věc na generování certifikátů, ale on umí být i klientskou aplikací pro SSL spojení. Stačí ho spustit s parametrem s_client:

```
$ openssl s_client -host SERVER -port PORT
```

Openssl s_client umožňuje zadat spoustu dalších parametrů, například umožní využít certifikát certifikační autority případně zaslat serveru klientský certifikát. Seznam všech parametrů dostanete, pokud zadáte jakýkoliv neznámý parametr:

```
[tsunami@vodik]$ openssl s_client -h
unknown option -h
```



```
usage: s_client args

-host host      -- use -connect instead
...
```

Když se připojíte k serveru pomocí openssl, na obrazovce se zobrazí informace o navazovaném SSL spojení:

```
[tsunami@vodik]$ openssl s_client -host tsunami.locnet -port 995
CONNECTED(00000003)
depth=1 /C=CZ/ST=-/L=Prag/O=Zdenek Burda/CN=Zdenek Burda/emailAddress=zdenda@
zdenda.com
verify error:num=19:self signed certificate in certificate chain
verify return:0
---
Certificate chain
 0 s:/C=CZ/ST=-/L=Prag/O=Zdenek Burda/CN=tsunami.locnet/emailAddress=zdenda@zdenda.com
 i:/C=CZ/ST=-/L=Prag/O=Zdenek Burda/CN=Zdenek Burda/emailAddress=zdenda@zdenda.com
 1 s:/C=CZ/ST=-/L=Prag/O=Zdenek Burda/CN=Zdenek Burda/emailAddress=zdenda@zdenda.com
 i:/C=CZ/ST=-/L=Prag/O=Zdenek Burda/CN=Zdenek Burda/emailAddress=zdenda@zdenda.com
---
Server certificate
-----BEGIN CERTIFICATE-----
MIIC4jCCAcOCAQIwDQYJKoZIhvcNAQEFBQAweDELMAkGA1UEBhMCQ1oxCjAIBgNV
...
gBRJXBktFFaRCI+4pUiUi02o+pfkkgeGoVBRtycau0XZ7TVT97aihjk8Qxj98Bys
lDt47imjTMwwh8LS0z0M60smIupVTw==
-----END CERTIFICATE-----
subject=/C=CZ/ST=-/L=Prag/O=Zdenek Burda/CN=tsunami.locnet/emailAddress=zdenda@zdenda.com
issuer=/C=CZ/ST=-/L=Prag/O=Zdenek Burda/CN=Zdenek Burda/emailAddress=zdenda@zdenda.com
---
No client certificate CA names sent
---
SSL handshake has read 2016 bytes and written 340 bytes
---
New, TLSv1/SSLv3, Cipher is AES256-SHA
Server public key is 1024 bit
Compression: NONE
Expansion: NONE
SSL-Session:
Protocol   : TLSv1
Cipher    : AES256-SHA
Session-ID: EACB77B795913F22DC65CD867301E84913CDBEAA1FE89E5A90A32E2891297E58
Session-ID-ctx:
```

```
Master-Key: F1480F1397A2A1DB995B4FB8AC720FEF6CFE77F02FBB95407576B3C023620
6736ABDE4FE2108AA936EC115B365CF70BA
Key-Arg    : None
Start Time: 1136712490
Timeout    : 300 (sec)
Verify return code: 19 (self signed certificate in certificate chain)
---
+OK Dovecot ready.
```

Po připojení pomocí openssl můžete zadávat příkazy stejně jako při připojení telnetem. Openssl ukončíte pomocí `^d` nebo `^c`. V případě mého spojení pomocí openssl se ukázalo hlášení o self signed certifikátu. Bylo to způsobené tím, že openssl nezná certifikát certifikační autority, která podepsala certifikát serveru. Pokud openssl vnutíme certifikát CA (CA.crt), bude vše v pořádku.

```
[tsunami@vodik]$ openssl s_client -host tsunami.locnet \\  
-port 995 -CAfile CA.crt
```

Docela zajímavou funkcí je, pokud server vyžaduje, aby mu klient zaslal svůj certifikát, jinak mu nedovolí připojení/přihlášení. OpenSSL tento certifikát předáme volbami `-cert` a `-key`.

```
[tsunami@vodik]$ openssl s_client -host tsunami.locnet \\  
-port 995 -CAfile CA.crt -cert burda.crt -key burda.key
```

Nejčastěji používané porty pro SSL varianty běžných služeb:

- imaps: 993
- pop3s: 995
- smtps: 465
- https: 443

A to je vše, přátelé

Jednotlivé protokoly jsem zde nepopisoval, protože je vše pěkně vysvětlené v odkazovaných RFC. Uvedený způsob ovládání emailu se nehodí jen pro testování, ale někdy se dá využít i v normálním životě. Pokud máte ve schránce velký email, který nechcete/nemůžete stáhnout, tak jeho smazání přes pop3 telnetem je záležitost na chvílku. Telnet na SMTP server se hodí, pokud zkoušíte, jestli můžete přijímat poštu zvenku, a přitom ještě není správně nastavený DNS server (pokud byste si zkusili poslat email klasickým způsobem z nějakého freemailu, nepřišel by, protože neví, kam má jít).

Odkazy

- [1] <http://www.ietf.org/rfc/rfc821.txt>
- [2] <http://www.ietf.org/rfc/rfc2821.txt>
- [3] <http://www.ietf.org/rfc/rfc1939.txt>
- [4] <http://www.ietf.org/rfc/rfc2449.txt>
- [5] <http://www.ietf.org/rfc/rfc1734.txt>
- [6] <http://www.ietf.org/rfc/rfc3501.txt>
- [7] <http://www.imap.org>

Jaderné noviny 335

Leoš Literák

Popis a debata vývojového procesu. Hledají se testeři Sharp Zaurus-5500. Nový souborový systém eCryptFS. Podpora řadiče Secure Digital Host.

Popis a debata vývojového procesu, 78 e-mailů

29. říj – 7. lis

Paolo Ciarrocchi napsal:

Rád bych napsal krátký článek o novém vývojovém procesu diskutovaném na posledním Linux Kernel Developers summitu. Neboť nemůžu najít přesný zápis, o čem se během setkání diskutovalo, snažím se shrnout, jak aktuálnímu procesu rozumím: Existují dva druhy vydání, jádra 2.6.x a 2.6.x.y 2.6.x.y jsou udržovány *stabilním* týmem (stable at kernel dot org) a vydávány skoro každý týden. Zde je poslední (jak doufám) oznámení, které jsem našel v archivu, obsahující vysvětlení, jak stabilní vývoj funguje: Pravidla, jaké patche jsou přijaty a jaké ne, do „-stable“ větve:

- Musí být samozřejmě správné a otestované.
- Nesmí to být delší než sto řádek včetně kontextu.
- Musí to opravovat jedinou věc.
- Musí to opravovat skutečnou chybu, která obtěžuje lidi (ne „To by mohl být problém“).
- Musí to opravovat problém, který způsobuje problém při kompilaci, oops, zatuhnutí, poškození dat, skutečný bezpečnostní problém, prostě něco kritického.
- Nesmí to obsahovat žádné triviální opravy (překlepy, úpravy počtu mezer atd.).
- Musí to být přijato správcem daného podsystému.
- Musí to dodržovat zásady z Documentation/SubmittingPatches.

Postup pro zasílání záplat do -stable větve:

- Zašlete patch, po zkontrolování výše uvedených pravidel, do stable@kernel.org.
- Odesílatel obdrží potvrzení, že patch byl přijat do fronty, nebo zamítnut. Odpověď může trvat několik dní, podle rozvrhu vývojáře.
- Pokud bude přijat, patch bude zařazen do -stable fronty pro prohlédnutí ostatními vývojáři.
- Bezpečnostní záplaty by neměly být zasílány na tuto schránku, ale do dokumentované schránky security@kernel.org.

Kontrolní cyklus:

- Když se -stable správci rozhodnou pro přezkumný cyklus, záplaty jsou zaslány kontrolní komisi [review committee] a správci dotčené oblasti (pokud není zároveň odesílatelem patche) a do kopie i konference linux-kernel.
- Kontrolní komise má 48 hodin na přijetí či zamítnutí patche.
- Pokud bude záplata zamítnuta členem komise nebo členové konference linux-kernel naleznou problémy, na které správce či členové komise nepřišli, patch bude vyřazen z fronty.
- Na konci kontrolního cyklu budou akceptované patche zařazeny do nejnovější -stable verze a ta bude vydána.
- Bezpečnostní patche budou přijaty do -stable větve přímo bezpečnostním jaderným týmem a nikoli běžným kontrolním cyklem. Kontaktuje tento tým pro detaily této procedury.

Kontrolní komise:

- Bude vytvořena z většího počtu vývojářů jádra, kteří se dobrovolně přihlásili pro tento úkol, a několika, kteří ne.

Jádro 2.6.x je spravováno Linusem Torvaldsem a Andrew Mortonem (oba z OSDL) a vývoj je následující:

- Jakmile nový kernel je vydán, otevře se dvoutýdenní okno, během kterého mohou správcové zaslat Linusovi velké diffy (obvykle prověřené několika týdny v -mm jádrech). Preferovaný způsob je skrze git (více na <http://git.or.cz/> [1] a v <http://www.kernel.org/pub/software/scm/git/docs/tutorial.html> [2]).
- Po dvou týdnech je vydáno -rc1 jádro a od té doby je možné zasílat jen patche, které nepřidávají novou funkčnost.
- Po dvou týdnech je vydáno -rc2 jádro.
- Proces pokračuje, dokud není jádro považováno za stabilní, proces by měl trvat tři měsíce (měla by být tedy vydána čtyři jádra za rok).

Je mi jasné, že jsem zapomněl na spoustu detailů, takže bych byl vděčen za podporu při psaní tohoto článku, který snad bude užitečný mému lokálnímu LUGu :-)

Jesper Juhl reagoval s tím, že nedávno napsal dokument o různých jaderných stromech a jak do nich zasílat patche. Dokument je možné najít v Documentation/applying-patches.txt nebo [online](#) [3].

Tony Luck napsal Paolovi ohledně -rc1: Původně Linus napsal, že bude přijímat pouze emailové patche po rc1 ..., ale zdravý rozum zvítězil a později ujasnil, že git merge mohou být také použity, ale jen pro opravy chyb.

Pak dodal, že zcela nový ovladač či souborový systém může být přijat i po -rc1, protože zde není riziko zhoršení. Tony dále napsal, že oficiální pevný termín má pouze první kandidát -rc1, další přicházejí, jak přicházejí. Pokud jde o tříměsíční termín, cílem je vytvořit novou verzi v osmi týdnech, což je blíže šesti verzím za rok. Linus Torvalds napsal:

Ano, tři měsíce by byly příliš mnoho. Osm týdnů je lepší, než bývalo zvykem, ale myslím, že šest týdnů by mělo být naším cílem. Ale abychom toho dosáhli, měli bychom zkrátit začleňovací okno z dvou týdnů na jeden, jinak nebude dost prostoru pro vychladnutí. Protože se nejvíce vývoje děje během předchozí chladicí fáze, tak by to nemělo být nemožné: měli bychom jen jeden týden na začlenění, ale to není ten týden, kdy se vývoj děje, takže komu by to vadilo?

Takže plánuji pokračovat prozatím beze změny. Dvoutýdenní začleňovací okno až do -rc1, pak další -rc vydání každý týden až do releasu. Cíl je šest týdnů s tím, že osm týdnů je v pořádku. Nemyslím, že by někdo byl z tohoto skutečně nešťastný, že?

Vývojáři diskutovali nad smyslem -rc vydání a zda by měl být poslední -rc kernel přejmenován na finální verzi. Linus odpověděl:

To by vůbec nepomohlo. Proč? V první řadě protože jen malé procento lidí testuje -rc jádra. Takže pokud vydáte poslední -rc verzi jako standardní kernel, tak za první stejně nezískáte žádné výhodnější pokrytí a za druhé jste promarnili týden snahou získat nějaké pokrytí, které nedostanete. Aktuální model vývoje kernelu je začlenit novinky na začátku, což doufejme motivuje lidi zkoušet -rc jádra pro testování -rc1, neboť vědí, že to je to jádro, které má všechny dobrůtky.

Většina z testerů -rc1 nikdy nenarazí na žádné problémy. A ti, kteří ano, budou nejspíše testovat další -rc verze, dokud jejich problém není vyřešen. A ti, kteří netestují -rc verze (protože neupgradují moc často) nebudou nikdy testovat -rc verzi, ať už první nebo poslední. Takže s čím musíme bojovat, je stabilní řada. Vypouštíme finální 2.6.x kernel s co nejlepším testováním, ale je nepopiratelným faktem, že spousta lidí bude zkoušet toto jádro a často až po několika týdnech od uvedení. Vytváření -rc verzí nic neprovede pro tyto případy. Takže opakujte po mně: „Většina lidí nikdy netestuje -rc verze“.

Hledají se testeři Sharp Zaurus-5500, 7 e-mailů

2. lis – 4. lis

Pavel Machek napsal, že hledá testery pro sharp zaurus sl-5500 ochotných otestovat kernel. Na kernel.org je k dispozici linux-z strom, který se snaží držet synchronizován s hlavní větví a který se začíná stávat použitelným. Hlavní nevýhodou je, že nabíjení baterie ještě není hotovo, doteková obrazovka již má podporu, ale Pavel neměl možnost ji otestovat.

Nový souborový systém eCryptFS, 42 e-mailů

2. lis – 7. lis

Phillip Hellewell napsal: Tato sada záplat vytváří eCryptfs verze 0.1. Posíláme ji pro prohlédnutí a případné začlenění do jádra. eCryptfs je stohovatelný [stackable] souborový systém založený na Cryptfs, který je generován pomocí frameworku FiST na stohovatelné souborové systémy napsaného Erezem Zadokem: <http://filesystems.org/> [4].

eCryptfs ukládá kryptografická metadata do hlavičky každého souboru, hlavičky obsahují pakety podobné OpenPGP (viz RFC 2440). To umožňuje kopírování kryptovaných souborů mezi počítači, přičemž informace nutné pro rozšifrování souborů zůstávají se soubory. eCryptfs se snaží, aby šifrování a dešifrování každého souboru bylo úplně transparentní uživatelským aplikacím, pokud má příjemce požadovaný klíč či heslo pro daný soubor.

Michael Halcrow předvedl eCryptfs v roce 2004 a 2005 na Ottawském linuxovém symposiu, základní přehled začíná na stránce 209 tohoto PDF:

http://www.linuxsymposium.org/2005/linuxsymposium_procv1.pdf [5].

Tato sada patchů obsahuje výrazně ořezanou verzi eCryptfs oproti tomu, co bylo posláno do LKML minule. Verze 0.1 obsahuje podporu zadávání hesla jen při připojování svazku, což činí eCryptfs snadnějším pro analýzu a debugování, než budou složitější vlastnosti jako veřejný klíč či rozšířená pravidla začleněna. eCryptfs funguje dobře při spoustě testů včetně FSX a Connectathon. eCryptfs používá keyring Davida Howella, verze 0.1 očekává existující autentizační token v uživatelské keyring relaci. Zdrojový kód si můžete stáhnout na SourceForge (ecryptfs-v0.1.tar.bz2): <http://sourceforge.net/projects/ecryptfs/> [6].

Podpora řadiče Secure Digital Host, 1 e-mail

8. lis

Pierre Ossman napsal: Začal jsem pracovat na ovladači podle specifikace Secure Digital Host Controller. Zdá se, že je používán většinou řadičů dnešních osobních počítačů, takže ovladač přidá podporu řadě čipů. Informace jsou vzácné, takže nevím, zda jej dotáhnu do konce, ale pro ty, kteří chtějí trochu nebezpečí ve svém životě: <http://mmc.drzeus.cx/wiki/Linux/Drivers/sdhci> [7]

Odkazy

- [1] <http://git.or.cz/>
- [2] <http://www.kernel.org/pub/software/scm/git/docs/tutorial.html>
- [3] <http://sosdg.org/~coywolf/lxr/source/Documentation/applying-patches.txt>
- [4] <http://filesystems.org/>
- [5] http://www.linuxsymposium.org/2005/linuxsymposium_procv1.pdf
- [6] <http://sourceforge.net/projects/ecryptfs/>
- [7] <http://mmc.drzeus.cx/wiki/Linux/Drivers/sdhci>

Zprávičky

2.1.2006

Sládeček Lukáš

Vžijte se do role hackera budoucnosti ve hře Uplink:Hacker Elite. Jedná se bohužel o komerční projekt (demo pro Linux najdete zde).

2.1.2006

Daniel Kvasnička ml.

Vyšly nové verze programu The GIMP a to vývojová 2.3.6 (seznam změn, poznámky k vydání, download) a stabilní 2.2.10 (seznam změn, download).

2.1.2006

Daniel Kvasnička ml.

C, seznamte se s Pythonem. Python, tohle je C. Tak těmito slovy začíná článek na LinuxJournal.com, ve kterém se dozvíte, jak integrovat Python do programů v C a ušetřit tak spoustu času.

2.1.2006

Daniel Kvasnička ml.

Potřebujete udělat prezentaci bez instalování jakéhokoliv kancelářského balíku? Pylize (CLI nástroj na tvorbu prezentací v HTML napsaný v Pythonu), o kterém se píše na NewsForge, by vám mohl pomoci.

2.1.2006

Daniel Kvasnička ml.

Včerejší zápis v blogu Zacka Rusina na KDE Developer Journals nese nadpis: It's going to be a good year. Mimo jiné v něm slibuje, že v KDE 4.0 bude možno (díky projektu Plasma) používat widgety určené pro Mac OS X Dashboard.

2.1.2006

Katarína Machálková

Nejen příznivci OpenSolarisu jsou zváni na další setkání Czech OpenSolaris Users Group spojené s přednáškou, které se koná 10.1.2006 v prostorách české pobočky Sunu. Tématem bude „OS Virtualization“, přednášet bude Frank Hofmann, který představí technologie v oblasti virtualizace (VMWare, Xen/Hypervisors, BrandZ).

3.1.2006

Daniel Kvasnička ml.

Zápis v blogu na XYZ Computing kritizuje vývojáře Linuxových aplikací za nedostatečný důraz na tzv. naming conventions – vybírání srozumitelných jmen. Uvádí také, že je to jeden z hlavních faktorů ovlivňujících prosazování Linuxu na desktopu.

3.1.2006

Daniel Kvasnička ml.

Terry Hancock se ve Free Software Magazine s časovým odstupem zamýšlí nad \$100 notebookem

pro rozvojové země. Myslí si, že i přes problémy s ním spojené to byl dobrý nápad.

3.1.2006

Milan Lajtoš

Znovu sme bližšie k X serveru na OpenGL. David Reveman zverejnil zdrojové kódy po tom, čo vývoj prebiehal za zatvorenými dverami. V krátkosti informuje OSnews. Ak by niekto nevedel, tak reč je o Xgl...;)

3.1.2006

Daniel Kvasnička ml.

The Register dnes informuje o záměru Googlu vytvořit a v síti obchodů Wal-Mart prodávat levné PC bez Microsoft Windows. Spekulace o Google OS se tedy opět vynořují na povrch a centrem pozornosti je samozřejmě Linux.

4.1.2006

Daniel Kvasnička ml.

Podle NewsForge to není s OCR lehké na žádné platformě. Nathan Willis vysvětluje proč a popisuje svoji cestu za kvalitním open-source OCR řešením.

4.1.2006

Daniel Kvasnička ml.

Ze zápisu v blogu Kaourantin.net je patrné, že Flash pro 64bit Linux bude. Autor popisuje problémy s portováním a navíc se dozvídáme, že místo Flashe 8 budeme mít na Linuxu rovnou verzi 8.5.

4.1.2006

Daniel Kvasnička ml.

Na LinuxDevices.com si můžete přečíst o projektu Xanadux, který se snaží nahradit Windows CE Linuxem a to v mobilních zařízeních firmy HTC.

4.1.2006

Martin Tesař

V poslední den minulého roku vyšla nová verze velmi kvalitní live distribuce Kanotix (2005-04).

4.1.2006

Lukáš Zapletal

Google dnes svým logem oslavuje narozeniny L. Braillea. Logo je to velmi pěkné, možná by si ho měl Google nechat napořád.

4.1.2006

Ruda

Co uveřejnila americká CIA o České republice? Článek na iDNES zmiňuje i údaj o umístění ČR v počtu linuxových serverů.

4.1.2006

Jiří Culka

Pokud hledáte volně stažitelnou českou literaturu k Unixu, tak se vám bude hodit tato stránka. Najdete tam manuál k AWK, učebnici Unixu či popis Unixových shellů.

5.1.2006*Daniel Kvasnička ml.*

Google promluvil. Žádné levné PC (zprávička) vyrábět ani distribuovat nebude, protože partnerů, kteří kvalitně pokrývají danou část trhu, má dost. Informují TIMES ONLINE.

5.1.2006*Daniel Kvasnička ml.*

Vypadá to, že ledy se pohnuly. Na dot.kde.org se objevila zpráva o nově připravovaném systému správy zařízení pro KDE 4/Plasma s názvem Solid.

5.1.2006*Daniel Kvasnička ml.*

Bitva je vyhrána. V blogu na Consortiuminfo.org se objevila informace, že administrativa státu Massachusetts definitivně schválila používání otevřených formátů a to přímo ODF.

5.1.2006*Daniel Kvasnička ml.*

Na Slashdotu se dočtete, že na PortableApps.com je ke stažení přenositelná verze OpenOffice.org 2.01, kterou můžete používat např. ze svého USB flashdisku.

5.1.2006*Petr Vaněk*

Ano, tohle není vánoční dárek. Nová verze stabilní řady DTP programu Scribus 1.2.4 (všechny změny) nemá s vánoci, ni s chanukou nic společného.

6.1.2006*Daniel Kvasnička ml.*

V jednom italském blogu o OSS je k přečtení interview s člověkem jménem Fabio Massimo Di Nito, který je vedoucím týmu majícího na starosti Ubuntu Server Project.

6.1.2006*Michal Karas*

Téměř po roce od vydání poslední verze vyšel včera Arch Linux 0.7.1 (Noodle). Stahovat je možné z HTTP/FTP i přes BitTorrent.

6.1.2006*Daniel Kvasnička ml.*

Vyšla nová verze WYSIWYG XML editoru Etna 0.2.5 (Windows, Linux). Program je založen na Firefoxu 1.0.7 a licencován je MPL/GPL/LGPL. Informuje CZilla.

6.1.2006*Daniel Kvasnička ml.*

Na NewsForge se můžete blíže seznámit s novou verzí populárního nástroje pro tvorbu blogů: WordPress 2.0. Řeč je snad o všech důležitých vlastnostech a změnách, se kterými se setkáte.

6.1.2006*David Watzke*

GnomeDesktop.org informuje o nové vývojové verzi Gnome (2.13.4). Jde o poslední vydání řady 2.13 před „zmrazením“ API.

6.1.2006*black*

Mimo nové verze Archlinuxu vyšel včera též Vector SOHO-5.1-rc2 a dnes i Lunar-1.6.0-i686-rc1. Lunar linux je pro ty, kteří si kdy představovali, jaké by to bylo získat Gentoo a Archlinux v jednom. Samozřejmě dělá to trochu jinak ;)

8.1.2006*Jan Kunderát*

Po necelých deseti měsících práce :-)) se konečně český překlad Gentoo Handbooku stal oficiálním. Díky všem, kdo pomáhali.

8.1.2006*Mirek*

Byla uvolněna PBI instalace knihoven pro Free Pascal lazarus 0.9.11 pro PC-BSD.

8.1.2006*Mirek*

Byla vydána nová verze Wine 0.9.5, přímý odkaz, binární balíčky vyjdou během pár dní. Opravena spousta chyb, přidáno mnoho vylepšení IDL kompilery a řada MSI oprav.

8.1.2006*Leoš Literák*

Projekt online linuxové učebnice má své vlastní diskusní fórum. Můžete v něm diskutovat o jejím zaměření, jednotlivých kapitolách atd.

8.1.2006*Michal Vyskočil*

Už jste zanesení ve Frappr! mapě uživatelů editoru vim?

9.1.2006*Daniel Kvasnička ml.*

LinuxInsider píše o výzkumu odborníků z US-CERT (United States Computer Emergency Readiness Team), kteří zjistili, že UNIXové systémy mají za r. 2005 na kontě více bezpečnostních děr než MS Windows. ZDNet píše o rychlé reakci Red Hatu.

9.1.2006*Daniel Kvasnička ml.*

Na arstechnica.com se dozvíte více o rozhodnutí firmy HP zrušit projekt multimediálního PC s Linuxem (počítaného spíše už do oblasti spotřební elektroniky), jelikož pro něj nevidí na trhu místo. HP se přesto míní dále více soustředit na trh se spotřební elektronikou (HDTV, digitální kamery).

9.1.2006*Lukáš Zapletal*

LinuxEXPRES informoval, že firma Grisoft dnes uvedla antivirus AVG zdarma také pro Linux.

9.1.2006*Jan Grmela*

Včera vyšly nové verze driverů pro bezdrátové karty Intel PRO/Wireless 2100 a 2200. Seznam změn první z nich zahrnuje mimo jiné například přidání podpory LEAP autentizace nebo správné zobrazování výstupního výkonu, druhý uvádí pouze opravy chyb.

10.1.2006*Daniel Kvasnička ml.*

Na CNETu vyšel rozhovor s Jamesem Goslingem, otcem programovacího jazyka Java. Nese se v duchu otázky: Lepší se Java s věkem?

10.1.2006*Daniel Kvasnička ml.*

OSDir upozorňuje na zajímavý článek, ve kterém jeho autor vysvětluje, že cíl FOSS komunity není „Linux na každém desktopu“ a poukazuje na rozdíl mezi alternativou k Windows a náhradou za Windows.

10.1.2006*Daniel Kvasnička ml.*

První návrh GNU GPL v3 by měl být veřejnosti představen už tento týden a to na Massachusetts Institute of Technology (MIT) v USA. Informuje ComputerWeekly.com

10.1.2006*Daniel Kvasnička ml.*

Jon Trowbridge, který byl v Novellu zodpovědný za projekt indexovacího/vyhledávacího engine Beagle, odchází pracovat pro Google. Více na CNET News.com

10.1.2006*Pavel 'lingeek' Szalbot*

Red Hat patřil k největším odpůrcům Mono platformy ve svých produktech či jejího začlenění do jádra Gnome. V pekle však zřejmě začíná mrznout, jelikož Fedora Core 5 bude obsahovat Mono. Informují OSNews.

10.1.2006*Radek Vokál*

Ve čtvrtek 19. ledna 2006, od 9:00 do 17:00, se pod záštitou brněnského vývojového centra Red Hatu v budově fakulty informatiky Masarykovy univerzity koná Fedora Install Fest 2006. Celou akci zahájí přednáškou zástupci Red Hat, Inc. Markus Mitter a Karen Clark. Poté si budete moci vyzkoušet linuxovou distribuci Fedora Core 4, donést si vlastní počítač a distribuci si nainstalovat a nakonfigurovat s pomocí vývojářů z Red Hatu.

10.1.2006*black*

A do třetice všeho dobrého, byla vydána finální verze STX Linuxu. Ten je určen na starší počítače, má kontrolní centrum a schopnost využívat balíčky pro Slackware. Přehled. (obsahuje např. Gimp 2.3.6)

10.1.2006*Daniel Kvasnička ml.*

Na LinuxElectrons vyšel článek zaměřený (nejen) na ovladače grafických karet pro Linux a na problematiku jejich případného uvolnění jako open-source.

11.1.2006*Daniel Kvasnička ml.*

LinuxToday tlumočí oznámení FSF o projektu Gnash – GNU GPL Flash přehrávači/Mozilla pluginu. Gnash byl FSF zařazen mezi projekty s vysokou prioritou.

11.1.2006*Daniel Kvasnička ml.*

Na apcmag.com vyšel krátký rozhovor s Martinem Gregorem, jehož role ve společnosti Microsoft je prostá: lobovat proti open-source.

11.1.2006*Daniel Kvasnička ml.*

Firma Apple včera s velkou pompou představila své produkty s procesory Intel, PowerPC procesory tím ale nekončí. The Register informuje o portu kernelu OpenSolarisu pro tuto platformu nazvaném Polaris.

11.1.2006*Filip Korbel*

Pokud vás oslovuje nabídka náročné práce šéfredaktora LinuxBizWorld, při které lze však sledovat výsledky téměř přímo pod rukama, ozvěte se na praceabc.cz.

11.1.2006*Petr*

Dnes ve 21:14 zaslal Kev do diskuzního fóra informaci o uvolnění nové verze populárního IM klienta Psi. Odkazy ke stažení zatím pouze v tomto fóru.

12.1.2006*Daniel Kvasnička ml.*

CZilla dnes informuje o vydání finální verze poštovního klienta Thunderbird 1.5 (poznámky k vydání). Mezi nejdůležitější změny patří podpora automatických aktualizací, kontrola pravopisu při psaní a další. Česká verze pro Linux je zde, slovník pro kontrolu pravopisu v češtině pak zde.

12.1.2006*Daniel Kvasnička ml.*

Britský ZDNet přináší informaci o tom, že vláda USA vyčlenila více než milion dolarů na vyhledá-

vání a odstraňování chyb v open-source. Peníze půjdou (v podobě grantu) do pokladny Stanfordské univerzity a firem Coverity a Symantec.

12.1.2006*Daniel Kvasnička ml.*

V jednom blogu na Lobby4Linux se objevil krásný dopis psaný uživatelem Linuxu a adresovaný jeho operačnímu systému. Uživatel vylévá své srdce způsobem, který je mnoha lidem znám i z běžných partnerských vztahů...

12.1.2006*Pavel 'lingeek' Szalbot*

Vyšlo PHP 5.1.2. Mezi novinkami najdete vyřešené bezpečnostní problémy a spoustu opravených chyb spolu s maličkostmi jako např. PNG komprese pro GD;-).

13.1.2006*Daniel Kvasnička ml.*

NewsForge informuje, že Jabber Software Foundation (JSF) uvolnila specifikace dvou rozšíření protokolu Jabber: Jingle Signaling a Jingle Audio. V ten samý den vypustil Google (který se na tvorbě rozšíření podílel) knihovnu libjingle licencovanou v BSD stylu. Knihovna je implementací zmíněných rozšíření a je srdcem aplikace Google Talk.

13.1.2006*Daniel Kvasnička ml.*

Nedávnou překvapivou zprávu o zařazení platformy Mono do distribuce Fedora Core 5 doplňuje článek na CNETu informující o tom, že v Red Hat Enterprise Linuxu se Mono v dohledné době neobjeví.

13.1.2006*Daniel Kvasnička ml.*

The Register píše o firmách Sun Microsystems a Apple, které měly málem společnou historii. Sun se pokusil Apple jednou koupit a dvakrát málem došlo k fúzi. Existovaly také plány na využívání společného UI a spolupráci na architektuře SPARC.

13.1.2006*Ladislav Michnovič*

Zdá sa, že USA sú vskutku krajinou neobmedzených možností. Americký patentový úrad opäť priznal patent na súborový systém FAT istej firme, napriek tomu, že ten istý patent jej v roku 2004 odobral na popud Public Patent Foundation.

14.1.2006*Lukáš Zapletal*

V anketě na serveru Živě.cz (pravá dole) si Linux v anketě „Jaký používáte OS“ vede slušně. Kdo ještě nehlasoval, může to napravit.

15.1.2006*Vlastimil Ott*

Redakční systém Drupal dnes slaví páté narozeniny.

16.1.2006*Daniel Kvasnička ml.*

Na O'Reillyho LinuxDevCenter.com vyšel rozhovor s Aaronem J. Seigo, jedním z předních vývojářů KDE. Kromě problematiky portování KDE na jiné OS byla řeč samozřejmě o KDE 4, jehož preview verzi měl autor článku možnost vidět v akci na Aaronově notebooku.

16.1.2006*Daniel Kvasnička ml.*

Linux.com krátce recenzuje live distribuci GnuPup, která má za cíl poskytnout přenositelnou sadu nástrojů pro grafiky a designéry.

16.1.2006*Daniel Kvasnička ml.*

NewsForge píše o dvou sondách na Marsu (Spirit a Opportunity), které byly vypuštěny v r. 2003 a ačkoliv jejich mise měla už dávno skončit, stále pracují. Článek se samozřejmě zaměřuje na důležitou roli open-source software v tomto projektu.

16.1.2006*Daniel Kvasnička ml.*

Multiplatformní Stani's Python Editor se dostal do verze 0.8.1.d. Mezi hlavní změny patří absence okna terminálu při spouštění skriptu či možnost editovat několik souborů vedle sebe. Asi nejdůležitější změnou je ale lepší optimalizace pro Ubuntu Linux (a Linux celkově) a možnost spouštět SPE z místa (např. USB flashdisku), bez instalace.

16.1.2006*Aleš Kapica*

Řada uživatelů Linuxu si z linuxových aplikací pro děti vybaví maximálně „bramborového chlápka“, ovšem již nějaký čas se utěšeně rozrůstá projekt Gcompris, který zahrnuje celou řadu dalších programků pro děti a s vaší pomocí jich může být českým dětem zpřístupněno ještě víc.

16.1.2006*Peter Kotrcka*

Na niektorých serveroch (sunsite.mff.cuni.cz) je už druhé testovacie vydanie novej distribúcie Fedora Core 5. Testujme všetci.

16.1.2006*Tomáš Hála*

Vyšel nový mod_security 1.9.2, který filtruje dotazy na www server Apache od potenciálně nebezpečných. Jedná se o bugfix release, který opravuje všechny známé chyby z verze 1.9.1.

16.1.2006*Vlastimil Ott*

Free Software Foundation vydala draft GNU GPL verze 3.

17.1.2006*Daniel Kvasnička ml.*

Pokud vás zaujal článek o sdílení záložek přes internet (social bookmarking), pak by vám mohlo přijít vhod i Python API pro jednu z nejpoužívanějších služeb tohoto druhu – del.icio.us.

17.1.2006*Daniel Kvasnička ml.*

Novell nedávno ohlásil otevření kódu (GPL) a vytvoření projektu na podporu svého bezpečnostního software AppArmor. Ten plní podobnou funkci jako např. dobře známý SELinux. Informace přináší NewsForge.

17.1.2006*Daniel Kvasnička ml.*

Linuxgangster.org píše o frontendu k antivirovému systému ClamAV s názvem KlamAV. Jak už název napovídá, jde o KDE/Qt aplikaci.

17.1.2006*Ondřej Surý*

Ubuntu Dapper Drake Flight CD 3 je připraveno. Jde o třetí z řady milníkůvých obrazů CD, které budou vydány během vývojového cyklu pro Dappera. Více informací naleznete na webu Ubuntu CZ.

18.1.2006*Daniel Kvasnička ml.*

Pokud sledujete vývoj nadějného a nedávno otevřeného programu pro práci s vektorovou grafikou Xara LX, pak vězte, že včera vyšla další verze 0.1d. Zdaleka není hotová, ale pokrok vidět je. Download pro Linux.

18.1.2006*Daniel Kvasnička ml.*

V jistém blogu hostovaném na doméně Lobby4-Linux.com se objevil zápis (17.1.06, 6:37) ve kterém se tvrdí, že během 72 hodin bude oznámen projekt, který by mohl mít mohutný dopad na Linux na desktopu a komunitu jako celek. Autor používá formulace jako „den, kterého se Microsoft nechtěl nikdy dožít“ apod. Údajně vše závisí na tom, jak se k věci postaví komunita sama.

18.1.2006*Daniel Kvasnička ml.*

Vyhledávací a indexovací engine Beagle má nově navržené uživatelské rozhraní. Na Linux Edge se můžete podívat na okomentované screenshoty.

18.1.2006*Robert Krátký*

Přijďte si vyzkoušet a nainstalovat poslední verzi distribuci Fedora Core či setkat se a podiskutovat

se členy českého Red Hat týmu. Více na LinuxBizWorld.

18.1.2006*Jiří Hlínka*

ATI vydala novou verzi proprietárních ovladačů grafických karet (pro Radeon 8500 až X850). Novou verzi 8.21.7 je možné stáhnout ze stránek ATI. Mezi nejdůležitější změny patří úplná podpora OpenGL 2.0.

19.1.2006*Robert Krátký*

LinuxToday informuje, že byl aktualizován neoficiální průvodce pro uživatele začínající s Ubuntu Linuxem. Viz EasyLinux.info: Ubuntu.

19.1.2006*Pavel 'lingeeek' Szalbot*

Google podle informačního serveru DesktopLinux.com otevírá svoji IM službu GoogleTalk všem ostatním IM sítím využívajícím protokol XMPP. Časem dojde i k podpoře SIPu, který umožní hlasovou komunikaci.

19.1.2006*Petr Vaněk*

Po několika letech zatuhlosti jsem oživil Oraschemadoc – nástroj, který generuje dokumentaci Oracle schémat (databází). Kompletní oznámení.

19.1.2006*Robert Krátký*

Evropa poprvé předstihla Ameriku v počtu prodaných PC. Hodně z toho profitoval například Acer, který byl v posledních 2 letech nejrychleji rostoucím výrobcem PC hardwaru.

19.1.2006*Daniel Kvasnička ml.*

Britský ZDNet informuje o tom, že HP přemýšlí o uvolnění svých interních nástrojů pro ulehčení participace na open-source projektech. Předmětem rozhodování teď je, zda půjde o klasický produkt či službu.

19.1.2006*Daniel Kvasnička ml.*

Nedávno bylo ve vývojářském mailing listu Ubuntu Linuxu otevřeno téma podpory MP3, a to na základě vydání volně dostupného MP3 dekodéru (ve formě plug-inu pro GStreamer) firmou Fluendo. Plug-in však není vydán pod GPL a to omezuje možnosti jeho zařazení do linuxových distribucí. Informuje NewsForge.

19.1.2006*Daniel Kvasnička ml.*

Vývojářům Ubuntu se zalíbila služba .Mac a chystají se vytvořit něco podobného. K dispozici bude e-mailové konto, ukládání kalendáře a adresáře, blogování, ukládání souborů a další. Jméno Ubuntu

tu.Mac ale právníci firmy Apple asi těžko povolí. Informuje PCPro.

19.1.2006

Daniel Kvasnička ml.

Přibližně před měsícem byl spuštěn český web věnovaný vektorovému grafickému editoru Inkscape. Najdete ho na <http://inkscape.prosim.org/>.

19.1.2006

Lukáš Zapletal

V pátek vychází lednový LinuxEXPRES věnovaný školství, výuce a vzdělávacímu softwaru. Začíná nový seriál o vývoji jaderných modulů, přečtete si dlouhou recenzi Quake 4 a podrobný článek o přechodu firmy na VoIP. Stručný obsah s PDF náhledem na 10 stran.

20.1.2006

Jiří Větvička

Jak si zaktualizovat Mono, které získalo ocenění produkt roku 2006, v SUSE Linux 10 za pomoci RedCarpetu se dočtete na suse.portal.cz.

20.1.2006

Filip Krška

Dlouho očekávaná druhá verze Gnomemeetingu se blíží, vydána 2.00 BETA 1, už pod novým jménem Ekiga. K dříve podporovanému protokolu H323 přidává SIP.

20.1.2006

Daniel Kvasnička ml.

Linux.com představuje program Krita jako něco více než jen „GIMP pro KDE“. Program míří dál, než je manipulace s pixely a je zaměřen více na klasické způsoby tvorby obrázků.

20.1.2006

Daniel Kvasnička ml.

Trolltech přejmenoval svůj produkt Qt/Embedded na Qtopia Core, které je teď založeno na Qt 4.0 a navíc plánuje rozšířit produktovou řadu Qtopia o tři nové produkty: Qtopia Platform, Qtopia Profiles a Qtopia Editions. Informuje LinuxDevices.com.

20.1.2006

Daniel Kvasnička ml.

Dokumentární tým KDE ve čtvrtek vydal dokumentaci k balíku KOffice ve formátu PDF. Přečíst si další informace a stahovat můžete na DesktopLinux.com.

20.1.2006

Jesus Jimenez

Další z řady vynikajících článků populárního JXD je k nalezení na reflex.cz: Je linuxářství psychiatrická diagnóza?

20.1.2006

David Watzke

Vyšlo Wine 0.9.6. Vylepšuje DirectSound, přidává podporu duplexu a opravuje spousty chyb. Více informací na winehq.org.

20.1.2006

Ladislav Michnovič

Komu chýba hra typu warcraft pod linuxom, tomu doporučujem vyskúšať hru Dark Oberon. Zaujímava je svojou grafikou. Modely sú totiž z plastelíny. Možno vás bude zaujímať aj rozhovor s autormi hry.

20.1.2006

Alois Nešpor

Stáhnout si můžete novou opravnou verzi přehrávače Totem 1.3.1 (pro Gnome). Bylo opraveno dost závažných chyb (plugin pro Mozillu, xine-lib, Gstreamer 0.10). Přidána podpora přehrávání Flash videa a OGM. S přidáním přehrávání DVD v ISO,CUE a BIN a přepínání pohledů (DVD) v předchozí verzi se z Totemu stává jeden z předních přehrávačů videa ve světě Linuxu.

21.1.2006

Pavel Šefrámek

72 hodin uplynulo a máme tady „den, kterého se Microsoft nechtěl nikdy dožít“. Jak mnozí tipovali, jde o reklamní kampaň na Linux.

21.1.2006

Luboš Doležel

Schyluje se k dalšímu boji o SW patenty v EU? Günther Schmalz ze společnosti SAP tvrdí, že ano.

22.1.2006

Petr J. (eagle)

RedHat chce vybudovat v Brně vývojové centrum a hledá lidi, letos chce přijmout zhruba 50 až 70 zaměstnanců a do roku 2008 200 lidí, psalo o tom sobotní Právo.

23.1.2006

Robert Krátký

Herschel Cohen na LXer.com uvažuje o tom, v čem je současný přístup k vývoji open source kancelářských balíčků pochybený, a navrhuje jiné řešení: Wasted Efforts in F/OSS – Office Suites.

23.1.2006

Robert Krátký

K dispozici pro testování je SUSE 10.1 Beta 1. Seznam změn (oproti Alhpa 4). Download.

23.1.2006

Daniel Kvasnička ml.

Pokud vás zajímá, jaký je rozdíl mezi GNU GPL v2 a v3 a nechce se vám je obě podrobně pročitat, je tu Groklaw.net a jeho přehledná tabulka.

23.1.2006*Daniel Kvasnička ml.*

Na Wine Wiki jsou ke shlédnutí výsledky testu, ve kterém dosáhl software spouštěný přes Wine lepších výkonů než ten samý software spouštěný nativně ve Windows XP. Informoval osViews.com.

23.1.2006*Daniel Kvasnička ml.*

Open-source, armáda, zbraně a obrana státu. Máte pocit, že některý z těchto pojmů mezi ty ostatní nepatří? NewsForge v tom problém nevidí a vysvětluje, proč ideály open-source komunit nejsou v rozporu s používáním jejich software ve vojenství.

23.1.2006*Petr Vaněk*

Scribus 1.3.2 – Egalité: sice se zpožděním oproti plánu, ale s o to lepšími vlastnostmi je k dispozici další vývojová verze tohoto populárního DTP programu: kompletní oznámení. Přestože je to tuze horké zboží, je používáno v mnoha produkčních prostředích.

23.1.2006*Robert Krátký*

Návod na debian-administration.org nabízí rychlý úvod do vytváření vlastních balíčků pro Debian. Článek vhodně doplňuje již dříve vydaný postup pro překompilování balíčků.

23.1.2006*Marek Stopka (M4r3k)*

Setkání uživatelů svobodného operačního systému GNU/Linux se koná dne 3. 2. 2006 v Ostravě v restauraci Zlatý orel, podrobnosti naleznete na adrese www.m4r3k.org.

24.1.2006*Daniel Kvasnička ml.*

CNET News upozorňuje na kritickou bezpečnostní chybu v KDE (konkrétně v JavaScript interpretu používaném mj. Konquerorem). K napadení dojde ve chvíli, kdy v browseru otevřete útočnickou stránku a může se vám to stát jak v KDE 3.5.0, tak už v 3.2.0.

24.1.2006*Daniel Kvasnička ml.*

Tom Adelstein na LXeru poukazuje na to, že lodi Microsoftu se v poslední době nepodařilo vyhnout všem ledovcům a že Linux rozhodně jen nestojí opodál a nekouká.

24.1.2006*Daniel Kvasnička ml.*

Na Linux.ie najdete návod na zprovoznění sluchátek s mikrofonom připojených ke stroji s Linuxem přes bluetooth. Návod je sice pro Fedoru, ale v jiných distribucích to určitě bude podobné. Informoval LinuxToday.com

24.1.2006*Michal Kubeček*

Vývojový tým relačního databázového serveru Firebird vydal verzi 1.5.3, obsahující opravy chyb a drobná vylepšení, některá backportovaná z vývojové větve 2.0. K dispozici jsou buildy pro Linux a Windows a Release Notes.

24.1.2006*Andrej Krivulčík*

Po vyše mesiaci práce bola dnes uvoľnená druhá betaverzia verzie 2.0.0 instant messaging klienta Gaim. Vývojári si vzali k srdcu spätnú väzbu od používateľov prvej betaverzie a zmenili hlavne funkcie, na ktoré sa opakovali námietky. Je to však stále betaverzia, a tak obsahuje niekoľko neprijemných chýb.

24.1.2006*Robert Krátký*

Ne všichni vývojáři linuxového jádra se shodují v tom, jestli by měla být používána připravovaná verze GPL 3. Obzvláště obtížné by bylo zajistit souhlas všech, kteří do jádra přispěli kódem. . . Diskuze na LKML.

25.1.2006*Mirek*

Na www.pcbsd.com byla uvolněna nová verze PC-BSD 1.0RC2. Ke stažení je CD#1 (systém) + CD#2 (lokalizace).

25.1.2006*Robert Krátký*

Slashdot odkazuje na zprávu AP, ve které se píše, že Google souhlasil se zavedením cenzury výsledků vyhledávání v Číně. Management firmy tímto rozhodnutím prý trpí, ale přesto myslí, že to stojí zato. . .

25.1.2006*Daniel Kvasnička ml.*

Na Groklaw vyšel rozhovor s Peterem Quinnem (Massachusetts CIO), který velkou měrou přispěl k používání ODF ve státní správě Massachusetts a o jehož následné rezignaci jsme psali. Řeč je např. o důvodech jeho rezignace či o budoucnosti OpenDocument formátu.

25.1.2006*Daniel Kvasnička ml.*

Nové počítače firmy Apple osazené procesory Intel zajímají i linuxový svět. Gillian Farquhar (tiskový mluvčí firmy Red Hat) potvrdil, že firma bude oficiálně podporovat portaci Linuxu na Mac-tel. Více v blogu na CNET News.com.

25.1.2006*Daniel Kvasnička ml.*

K dispozici je první technické preview Samby 4.0.0 vyvíjené paralelně s řadou 3.x a kladoucí

důraz na porozumění protokolu Active Directory, který je používán ve Windows 2000 a vyšších.

25.1.2006

Robert Krátký

Článek na DevX.com si všímá, že GNU Gnash, svobodná implementace přehrávače formátu (Macromedia) Flash (SWF), je na seznamu šesti nejdůležitějších projektů FSF.

26.1.2006

fipa

Zapojte se do jednoho z největších open source projektů vůbec – český lokalizační tým OpenOffice.org hledá nové kolegy.

26.1.2006

Hunter

Vyšla první verze CDnavigatoru! CDnavigator je aplikace evidující vaše CD, DVD v databázi. Zároveň ukládá informace o fotkách (JPG), hudbě (MP3) a filmech uložených na médiích. K těmto datům můžete ukládat další metadata jako poznámky, hodnocení, atd. Přes všechny položky můžete vyhledávat a výběry ukládat. CDnavigator podporuje automatickou cache, která ukládá nejpoužívanější data na disk. www.hdf.cz/cdnavigator/

26.1.2006

Daniel Kvasnička ml.

Linus Torvalds se v LKML vyjádřil, že kernel byl vždy vydáván pod GPLv2 a ani příchod GPLv3 na tom nic nezmění. Informuje NewsForge.

26.1.2006

Daniel Kvasnička ml.

Firma Technalign, oficiální distributor SimplyMEPIS Linuxu se rozhodla implementovat princip sériových čísel. Absence takového čísla uživateli nepovolí stažení aktualizací z oficiálních repozitářů. Informuje NewsForge.

26.1.2006

Daniel Kvasnička ml.

Novell ve spolupráci s firmou Concurrent vytvořil real-time verzi SUSE Linuxu nazvanou Concurrent Real-Time Extensions Powered By SUSE Linux. Výrobce u systému garantuje obsluhu přerušení do 30 mikrosekund. Informuje Internetnews.com

26.1.2006

Libor Tvrdlík

Evropská komise se snaží získat stanovisko zástupců průmyslu a výzkumu k otázce patentové ochrany v EU. Pokud si myslíte, že máte k tomu tématu co říci, máte k dispozici dotazník. Další informace: Stránky IDABC.

27.1.2006

Daniel Kvasnička ml.

Novell se prostřednictvím své komunitní stránky CoolSolutions ptá, jaké aplikace z Windows nám nejvíce chybí. Slibuje, že následně kontaktuje výrobce vítězné aplikace, aby mu nabídl spolupráci na portování. Informuje DesktopLinux.com.

27.1.2006

Daniel Kvasnička ml.

LinuxInsider informuje o dobrovolném rozhodnutí firmy Microsoft uvolnit ty zdrojové kódy z Windows Serveru, které byly předmětem právního rozhodnutí Evropské komise v r. 2004. Hlavním důvodem tohoto kroku je fakt, že zdrojový kód je sám o sobě nejlepší dokumentací technologií použitých v uvedeném produktu.

27.1.2006

Daniel Kvasnička ml.

Na IBM developerWorks najdete pěkný návod na vytvoření mozaiky s pomocí programovacího jazyka Perl a programu ImageMagick.

27.1.2006

Ivan Bibr

Nezisková společnost Liberix o.p.s, která podporuje také free a open source software, zveřejnila na svých stránkách krátkou zprávu o činnosti za minulý rok.

27.1.2006

mrzout

Vyšel Sun Java Studio Creator 2 – profesionální vývojové prostředí pro webové aplikace. Na webu sun.com lze po registraci bezplatně stáhnout verzi pro Linux.

28.1.2006

Tomáš

Pravděpodobně všem uživatelům Netlab Jabber Serveru přijde zpráva oznamující ztrátu osobních profilů za posledních 30 dnů. Pro uživatele PSI je i v ní stručný návod na obnovu rosteru. Ostatní, pokud můžou, ať si raději zjistí, jak o data nepřijít, – radši ještě před přihlášením.

29.1.2006

fipa

Projekt OpenOffice.org oznámil dostupnost prvních testovacích buildů kancelářského balíku OpenOffice.org 2 pro MacOS X na procesorech Intel. OpenOffice.org je tak na Intelu dříve, než komerční MS Office.

29.1.2006

Leoš Literák

Pravděpodobně příští týden v nočních hodinách se server, na kterém běží www.abclinuxu.cz, fyzicky přestěhuje k ostatním serverům služby AbcHosting. Přesné datum upřesníme, během pře-

vozu budou naše služby nedostupné. Za případné potíže se omlouváme.

30.1.2006*Elatio Trancelis*

Acovea – Analysis of Compiler Options via Evolutionary Algorithm je program, který pomocí genetických algoritmů najde optimální sadu flagů pro GCC, což se vyplatí hlavně uživatelům source distribucí jako je např. Gentoo a Source Mage. Získat jej můžete na stránkách projektu nebo v Portage.

30.1.2006*fipa*

Organizace OSS Alliance zveřejnila monitoring svých aktivit v roce 2005.

30.1.2006*Daniel Kvasnička ml.*

Vyšlo Eclipse IDE ve verzi 3.1.2. Pokud již používáte řadu 3.1.x, můžete updatovat přímo přes update manager. Stahovat kompletní balík můžete zde.

30.1.2006*Daniel Kvasnička ml.*

Novell uveřejnil průběžné výsledky své ankety (zprávička), jejímž cílem je zjistit, kterou aplikaci by uživatelé Linuxu nejraději viděli oficiálně portovanou pro svůj systém. Co se týká počtu hlasů, Česká republika je v první desítce ve společnosti o mnoho větších zemí.

30.1.2006*Daniel Kvasnička ml.*

Intel v roce 2007 plánuje oficiálně uvést první procesory vyráběné 45nm technologií. Společnost už své schopnosti dokázala, když demonstrovala plně funkční 45nm SRAM (Static Random Access Memory). Informuje AppleInsider.

30.1.2006*Jan Pinkas*

Na stránkách <http://jabber.cz/wiki> začíná vznikat český zdroj informací o Jabberu. Měl by to být základ pro nezávislý portál o těchto technologiích. Pokud umíte psát a myslíte, že byste mohli přidat ruku k dílu, budete srdečně vítáni. Více informací na uvedených stránkách a na MUC (groupchat) konferenci talk@conf.netlab.cz.

30.1.2006*Robert Krátký*

Jason Norwood-Young se v článku na webu tectonic.co.za pozastavuje nad tím, jak se někteří bloggeři obouvají do open source a FSF/GNU/GPL: Open source must die die die.

30.1.2006*Libor Tvrdlík*

Vyšla nová verze populární kapesní distribuce SLAX, konkrétně verze 5.0.7.

30.1.2006*Robert Krátký*

Jaké měl dětství Richard Stallman, jak žije, s kým se přátelí. Odpovědi na tyto trochu nezvyklé otázky nabízí článek na InformationWeek.

31.1.2006*Vlastimil Ott*

Michala Čihaře, který je známý nejen na Abičku, vyzpovídal časopis LinuxEXPRES.

31.1.2006*Daniel Kvasnička ml.*

Článek na PC WORLDu přirovnává nekompletnost Ubuntu Linuxu po instalaci k el. spotřebiči bez baterií, ale zároveň nabízí aplikaci, která uživateli pomáhá dostat se přes překážky – Automatic.

31.1.2006*Daniel Kvasnička ml.*

Jen pár dní po první betaverzi tu máme OpenSUSE 10.1 Beta 2. Jde o bugfix verzi, je dostupná pro x86, x86-64 i PPC platformu a stahovat můžete zde. Informoval DesktopLinux.com.

31.1.2006*Martin Malec*

Server security-portal.cz, věnovaný bezpečnosti, Linuxu a hackingu, včera náhle skončil, stejně jako phorum.cz. Provoz již pravděpodobně nebude nikdy obnoven. (Vzpomínka...)

31.1.2006*Daniel Kvasnička ml.*

Ne každý si pochvaluje připravovanou GNU GPL v3 a jejího hlavního autora. John Carroll ve svém blogu na ZDNetu vysvětluje, proč má pocit, že Stallman žene GPL z útesu do vln.

31.1.2006*Tomáš Hála*

Vyšlo nové sponly – alternativní „shell“, který umožňuje zabezpečený filetransfer, ale nedovolí nic spouštět.

31.1.2006*Robert Krátký*

Podle Regdeveloper připravuje Google svou vlastní linuxovou distribuci pro desktop. Má být založena na Ubuntu a její pracovní jméno je (vynálezavé) Goobuntu.