



## Abíčko

Časopis serveru [AbcLinuxu.cz](http://AbcLinuxu.cz)

prosinec 2002

Sponzorem tohoto čísla je společnost [gebbeth](http://gebbeth.com)



---

Vychází také na CD-ROM jako příloha časopisu:

**PC WORLD**

# Obsah

<b>Editoriál</b>	<b>4</b>
<b>GNU Lilypond – The music Typesetter – 1. část</b>	<b>5</b>
Takže popořádku – co, proč, kdy a kde? . . . . .	5
Instalace . . . . .	6
Použitelné odkazy . . . . .	6
Něco k nalákání . . . . .	7
<b>Samba – Linux jako server v sítích s Windows</b>	<b>8</b>
Úvod . . . . .	8
Obsah . . . . .	8
Hodnocení . . . . .	8
<b>Moderní souborové systémy – 1. část</b>	<b>9</b>
Úvod . . . . .	9
Motivace nových souborových systémů . . . . .	9
Žurnálovací souborové systémy . . . . .	9
Omezení souborových systémů . . . . .	11
Optimalizace výkonu nových souborových systémů . . . . .	11
Vyhledávání volných bloků . . . . .	11
Problémy s velkými filesystémy . . . . .	11
Řídké soubory . . . . .	12
Odkazy a zdroje . . . . .	12
<b>Jak bylo v Seči</b>	<b>13</b>
Úvod . . . . .	13
Pátek . . . . .	13
Jiří Kosina: Základy bezpečného programování nejen pro OS Linux . . . . .	13
Cijoml: Bluetooth na Linuxu . . . . .	14
Petr Adámek: Java a Linux . . . . .	14
Sobota . . . . .	14
Ondřej Jakl, Karel Krečmer: Numerické výpočty na linuxovém clusteru . . . . .	14
Martin Grombřík: Linux do škol a otevřené systémy na středních školách . . . . .	14
Alois Vitásek: Aplikační server s XML-RPC rozhraním . . . . .	14
Jiří Kosek: DocBook . . . . .	14
Neděle . . . . .	15
Petr Olšák: C <sub>S</sub> T <sub>E</sub> X – historie, současný stav a budoucnost . . . . .	15
Marcel Kolaja: Komplexní řešení pošty za použití OSS . . . . .	15
Martin Mareš: Podivuhodný svět Linuxu, kapitola 2.5 . . . . .	15
Závěr . . . . .	15
Sborník . . . . .	15
<b>Cracking bez tajemství</b>	<b>16</b>
Úvod . . . . .	16
Obsah . . . . .	16
Hodnocení . . . . .	16

<b>Jaderné noviny 188–192</b>	<b>17</b>
Úvod . . . . .	17
Feature Freeze v říjnu . . . . .	17
Plug and Play Layer 0.6 . . . . .	17
Driverfs mění jméno . . . . .	18
Další stabilní řada: 2.6 nebo 3.0? . . . . .	18
Paketový zápis . . . . .	18
Souborový systém CIFS . . . . .	19
Sdílené správcovství IDE . . . . .	19
Linux Kernel conf 1.0 . . . . .	19
Vypalování na IDE . . . . .	19
Nový bezpečnostní projekt . . . . .	19
Účtování zdrojů . . . . .	20
Detekce zařízení na AGP 3.0 . . . . .	20
Nový vysoce komprimovaný souborový systém . . . . .	20
Swap-Space Mini-Howto . . . . .	21
Placení za opravy . . . . .	21
Pokus protlačit Reiser4 po zmražení jádra . . . . .	21
EVMS mění směr . . . . .	22
Testy IDE-CD . . . . .	22

## Editoriál

Vítejte u čtení časopisu Abíčko.

Abíčko vychází jako měsíční příloha serveru *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 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 AbcLinuxu.cz najdete rozsáhlou databázi hardwaru (návodů na instalaci 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.

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 směrujte na adresu: *literakl@abclinuxu.cz*.

Server AbcLinuxu.cz provozuje firma AbcLinuxu s.r.o., která poskytuje profesionální služby v oblasti Linuxu. Nabízíme například instalaci a konfiguraci serverů a desktopů, školení a poradenství.

©2002 AbcLinuxu s.r.o. a autoři článků

Sazba: Ondřej Krejčík

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

# GNU Lilypond – The music Typesetter – 1. část

Tomáš Kopeček

## Takže popořádku – co, proč, kdy a kde?



Skalní  $\text{T}_{\text{E}}\text{X}$ isté toužící sázet kromě téměř dokonalých textů také téměř dokonalou hudbu kdysi dávno napsali  $\text{MusiX}_{\text{T}}\text{E}_{\text{X}}$ – sadu maker pro  $\text{T}_{\text{E}}\text{X}$ , které byly schopny sázet velmi kvalitní hudbu, ovšem za použití velmi hrůzných postupů. Možná jste se s ním někdo setkal a pokud ano, tak bych se věru nedivil, že vás odradil a už jste zase sázeli notičky v nějakém klikacím programu.

Něco podobného se stalo také jistému Janu Nieuwenhuizenovi (to jméno se mi snad nikdy nepodařilo vyslovit). Ten se však nenechal odradit a začal psát preprocessor pro  $\text{MusiX}_{\text{T}}\text{E}_{\text{X}}$  (tedy něco, v čem se bude psát podstatně jednodušeji a výstupem bude vstup pro  $\text{MusiX}_{\text{T}}\text{E}_{\text{X}}$ ). Tento projekt se jmenoval MPP ( $\text{MusiX}_{\text{T}}\text{E}_{\text{X}}$  preprocessor). Ale pak to řekl kamarádovi a ten mu to rozmluvil, že takhle teda rozhodně ne. Pak se spolu s ním (Han-Wen Nienhuys) pustil do pořádného projektu pro sazbu hudby. To všechno se stalo někdy mezi lety 1995–96. O šest let později vychází GNU Lilypond již ve verzích 1.6.x (stabilní řada) a zároveň jede na plné obrátky další vývoj. Ale dost už historie – takže o čem to celé je.


GNU Lilypond je překladač z popisu hudby ve vlastním formátu do několika výstupních formátů. Původním výstupním formátem byl  $\text{T}_{\text{E}}\text{X}$ ový zdrojový kód. Postupem let se přidaly možnosti výstupu přímo do PostScriptu, Scheme a poněkud pofidérně i do ASCIIartu. Zatím stále nejjistějším a nejpoužívanějším je  $\text{T}_{\text{E}}\text{X}$ ový výstup. Tímto způsobem lze získat jak kvalitní PostScript, tak PDF. Lilypond je schopen kompletního zpracování tiskové strany pomocí skriptu `ly2dvi` (`.ly` je standardní přípona Lilypondu, `.dvi` je pak device independent formát pocházející z  $\text{T}_{\text{E}}\text{X}$ u). Druhou variantou je začlenění zdrojového kódu pro Lilypond přímo do dokumentu v jiném jazyce. Tak můžeme získat kousky notového partu téměř kdekoli. Tímto způsobem jsou podporovány formáty  $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$ , HTML a  $\text{T}_{\text{E}}\text{X}$ info.

V současnosti se Lilypond snaží zcela oprostít od závislosti na  $\text{T}_{\text{E}}\text{X}$ u jako finálním sázecím prostředku a hledá se možnost nějakého univerzálnějšího prostředku pro popis sazby. Uvažuje se například i o XML. Ale to vše je hudbou budoucích verzí. Zatím je stále nejjistějším způsobem průchod přes  $\text{TeX}$ , který už je doveden téměř k dokonalosti a i tudy se dostaneme ke všem nabízeným výstupním formátům.

A proč vlastně neklikat? Aspoň pro mne je výrazně rychlejší něco psát, než klikat myší někam do osnovy (ovšem i pro Lilypond existuje jakýsi grafický frontend Denemo. Bohužel mám pocit, že se na něm již delší dobu nepracuje). Abych uklidnil alespoň nějakou část myší veřejnosti, tak existuje fičurka point-and-click, která alespoň umožní kliknutím v prohlížeči (`xdvi`) výsledného souboru odskočit na patřičné místo ve zdrojovém textu. Druhý důvod je celkem nasnadě – výstup Lilypondu je opravdu kvalitní a plně ovladatelný. Lilypond je propracovaný překladač. Ostatně binárka má svých 16 MB. No dobře, po ostřipování se dostaneme na 2,5 MB. Nicméně ani základní dokumentace není z nejmenších – má 164 stran.

Abych vás dlouho nenapínal, tak krátký kousek kódu pro Lilypond může vypadat třeba takto:

```
\key c \minor c'8
```



Nyní se možná rozpomenou někteří ”trackisté”. Podobnou syntaxi pro zápis not totiž používaly svého času snad všechny hlavní programy pro psaní modulů (`FastTracker`, `ScreamTracker`, atd.) Než se ovšem pustíme do složitějšího příkladu, bylo by dobré vědět, jak Lilypond nainstalujeme, takže:

## Instalace

Lilypond pracuje jak na unix-like systémech, tak na MS Windows (pomocí Cygwinu). Nutnou prerekvizitou je především interpret jazyka scheme a to je v tomto případě guile. Většina nových distribucí ignoruje vývoj tohoto interpretu, ač je použit na mnoha místech (např. GIMP). Bude tedy pravděpodobné, že si jej budete muset stáhnout sami (nebo sáhnout po starší verzi Lilypondu). Další potřebné programy pro běh jsou  $\TeX$ , **ghostscript** a Python. Pro prohlížení výstupu pak ideálně **xdvi** (součást většiny distribucí  $\TeX$ u). Na serveru se čas od času dá narazit i na aktuální RPM či deb. Pokud to není váš případ, tak následuje stažení aktuální verze a klasický postup:

```
configure; make; make install
```

Ovšem ke kompilaci je potřeba o něco více programů než ke běhu. Jsou to: **gcc** (c++), **make**, **flex**, **bison** a  $\TeX$ info (kvůli dokumentaci).

Dokumentaci lze vygenerovat pomocí **make web-doc** (pro korektní zpracování tohoto cíle budou zapotřebí ještě programy **autotrace**, **pktrace** a **tlasm** – všechny jsou dosažitelné ze serveru Lilypondu). Nainstalují se tyto spustitelné soubory:

- **lilypond** – vlastní překladač, ten budete sami spouštět asi málokdy, většinu práce za vás odedřou následující dva skripty
- **ly2dvi** – skript pro převod souboru .ly na .dvi, PostScript nebo PDF
- **lilypond-book** – převede LaTeXový soubor s vnořenými příkazy Lilypondu na jakýsi druhotný vstup pro LaTeX, který už je jím přímo zpracovatelný.
- **abc2ly**, **etf2ly** ... – skripty pro převod z jiných formátů do .ly
- **midi2ly** – skript pro převod souboru midi na .ly. Kvalita převodu je přinejmenším různá.
- **convert-ly** – skript umožňující dopřednou konverzi souborů pro aktuální verzi syntaxe. Dojde-li totiž k nějakým změnám syntaxe (a to občas dochází z důvodů větší systematičnosti a přehlednosti), tak jsou tyto změny podchyceny v tomto skriptu. Nemusíte tedy mít obavy, že soubory napsané pro starší verzi už nikdy nezpracujete.

V adresáři `/usr/local/share/lilypond/cislo.verze/scm` (tedy pokud jste pomocí **configure** nezměnili `--prefix`) se nacházejí všechny inicializační soubory pro scheme. Je to vhodné místo pro náhled, jak vypadají například definice jmen akordů. Poblíž v adresáři `ly` se nacházejí všechny soubory, které byste mohli chtít vložit do své skladby (definice formátů papíru, jiné názvy not – pro nás je aktuální soubor `deutsch.ly`).

Teď vám nechám trochu času na vydýchání a kompilaci a příště se pustíme do nějaké větší skladby.

## Použitelné odkazy

- Hlavní stránka projektu – <http://www.lilypond.org>
- Projekt GUILE – <http://www.gnu.org/software/guile/guile.html>

## Něco k nalákání

### Trudova žena

moravská lidová

Um-ře - la dnes-ka Tru-do - vi že - na

a Trud jí ne - vě - ří „Vstaň žen - ko mo - je

sr-dén-ko mo - je šak fa nic ne - bo - lí“

# Samba – Linux jako server v sítích s Windows

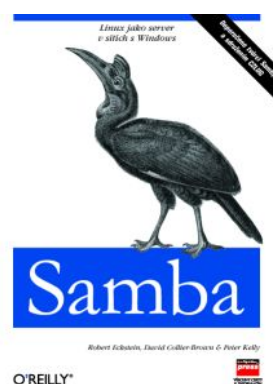
Petr Kačenka

## Úvod

Tato kniha, jak již sám název napovídá, se věnuje nástroji na sdílení prostředků Samba. Kniha je určena pro odborníky na systém Unix a předpokládá pouze minimální znalost systému Microsoft Windows. Je velká škoda, že autoři nerozšířili knihu o kapitolu, která by popisovala systém Unix alespoň v podobném rozsahu, jako je popisován systém Windows. Kniha by potom byla dostupnější také správcům, kteří používají pouze Windows a Linuxu se z libovolného důvodu obávají. Určitě by ubylo správců, kteří jsou podle autorů knihy zralí pro lékařskou péči, protože o Linuxu doposud ani neslyšeli.

## Obsah

Kniha je formálně členěna do devíti kapitol a šesti dodatků. V kapitole první jsou vysvětleny základy, které jsou později využívány v dalším textu. V kapitole druhé je podrobně popsána instalace Samba serveru na počítači s Linuxem. Třetí kapitola se podrobně věnuje nastavení klientů Windows pro práci se Sambou. V kapitolách čtyři a pět jsou diskutována disková sdílení. V kapitole šest jsou diskutovány uživatelské účty a jejich zabezpečení. Informace o tisku a službě WINS jsou popsány v kapitole sedmé. V posledních dvou kapitolách se autoři věnují lahůdkám a problémům, včetně jejich řešení. Pro správce samby jako takové jsou velice důležité dodatky, obzvláště dodatek C – Referenční příručka konfiguračních nastavení Samby.



## Hodnocení

Autoři se v knize nebojí upozornit na nedostatky Samby v porovnání s nativním chováním Microsoft Windows. Správce sítě si tak může ověřit, jestli je Samba řešením pro jeho síť ještě dříve, než započne samotnou instalaci. V knize jsou také často vysvětlovány způsoby řešení rozdílných filosofí přístupu autorů Unixu a autorů Windows.

Ačkoliv knihy s podobnou tematikou stárnou velice rychle, autoři i překladatelé se s tímto jevem vyrovnali velice dobře. Kniha se snaží vysvětlovat obecné vlastnosti Samby a upozorňuje na vybraná specifika některých verzí. Časté jsou také upozornění na záludnosti různých verzí Microsoft Windows a na nutnost instalace odpovídající verze Samby.

Pěkný dojem z knihy kazí relativně časté formální chyby a nedostatky. Zpravidla se jedná o nesoulad mezi psaným textem a obrázky. Jako příklad uveďme situaci, kdy v textu hovoříme o sdíleném disku G, ale na obrázku je vyznačen disk H. Časté chyby jsou v IP adresách. Dobrý pocit z knihy kazí také nedůslednost překladatelů, kteří některé obrázky nepřeložili do češtiny, nebo naopak přeložili obrázek, ale nepřeložili výpis v textu. Nežřídká se proto z obrázků, které mají usnadnit orientaci v textu, stávají spíše hlavolamy.

Název:	Samba – Linux jako server v sítích s Windows
Autor:	Robert Eckstein, David Collier-Brown & Peter Kelly
Vydal:	Computer Press
Počet stran:	384
Doporučená cena:	397 Kč
Datum vydání:	2001



# Moderní souborové systémy – 1. část

Petr Novický

## Úvod

Mnozí z Vás se jistě chtěli seznámit s pojmem souborový systém hlouběji. Ačkoliv si myslím, že se drtivá většina s pojmem souborový systém setkala, přesto ho nejprve objasním. Je to software, který slouží k organizování a používání dat uložených na záznamových médiích (pevný disk, CD, apod). Souborový systém zajišťuje integritu dat (přeloženo do českého jazyka ucelenost). Tudíž informace, které uložíme, budeme moci později vyvolat v nezměněné podobě.

Pro svou činnost filesystém ukládá informace o souborech a informace o sobě samotném (vlastnictví, datum, kontrola přístupu, délka souboru a jeho lokace na disku, apod). Bez těchto informací, tzv. metadat, by souborový systém nemohl pracovat. Nebudu se zde zabývat souborovými systémy jako ext2fs (běžně používaný souborový systém v OS Linux), o něm byla jistě již spousta materiálu sepsána a tak nebudu nosit dříví do lesa. Navíc nové filesystémy mají lepší vlastnosti a nemají tak přísná omezení jako třeba právě ext2fs.

## Motivace nových souborových systémů

Modernější filesystémy použijeme zvláště tehdy, když chceme zajistit integritu dat i při náhlém přerušení práce počítače. Existuje několik možností, co se při něm může stát:

- Systém stihl uložit soubor. Nic se neděje, můžeme pokračovat v práci.
- Systém nestihl uložit soubor (ani nezačal). Přišli jsme tedy o všechny změny, ale alespoň máme zachovanou starou verzi.
- Systém "spadl" během procesu ukládání. Toto je nejhorší případ! Dostáváme soubor, který se skládá částečně ze starého a částečně z nového. Když navíc zrovna zapisujeme metadata (jako třeba informace o adresářích), můžeme ztratit třeba celý adresář nebo dokonce data na diskovém oddílu (když se poruší metadata týkající se kořenového adresáře)!

Standardní linuxový souborový systém (ext2fs) svému poškození částečně předchází tím, že udržuje redundantní kopii metadat, takže se většinou nestává, že bychom o ně přišli. Pomocí kontroly integrity souborového systému (fsck), klasicky během bootování, je systém schopen zjistit, kde jsou metadata poškozená a nahradí je prostým zkopírováním redundantní verze. Nebo dojde ke smazání souboru, který byl přerušením poškozen. Samozřejmě že kontrola trvá tím déle, čím větší máme diskový oddíl a kontrola opravdu velkého disku může trvat velmi dlouho. Žádná z uvedených vlastností se samozřejmě nikomu z nás nelíbí, naštěstí existuje alternativa k těmto klasickým druhům filesystémů. Jsou jimi souborové systémy, které pracují s tzv. žurnálem.

## Žurnálovací souborové systémy

O co se jedná? Stručně řečeno žurnálovací filesystém si uchovává informace o operacích, které provedl a je pak v případě výpadku schopen rychle se dostat zpět do konzistentního stavu. Změny jsou evidovány jako tzv. transakce. Jedná se o nezávislé atomické operace. Po každé transakci následuje potvrzení, když dojde k uskutečnění daného úkonu (např. zápis na disk). Proto pokud systém „spadne“, můžeme najít v záznamech informace o provedených změnách a vrátit vše do původního stavu. Mezi tyto souborové systémy patří např. ext3, ReiserFS, XFS a JFS. V našem seriálu se postupně zmíníme o každém z nich podrobněji.

**AbcLinuxu s.r.o. nabízí tyto služby:**

- Instalace
  - firewall
  - DNS
  - proxy
  - e-mail
  - samba
  - web
  - databáze
  - ftp
  - Account server
- Virtuální privátní síť (VPN)
- Zabezpečení spojení IPSEC
- Návrh a nasazení systému zálohování
- Analýza a rekonstrukce poškozených systémů
- Celopodniková řešení na bázi Linuxu a Open Source
- Odborné konzultace v oblasti Open Source a Linuxu
- AbcPalírna – pálené distribuce v centru Prahy, ceny od 20 Kč / CD
- Prodej reklamy na serveru AbcLinuxu.cz (35 tisíc návštěvníků měsíčně)



**Kontakt:**

e-mail: *info@abclinuxu.cz*  
telefon: 721 300 742  
adresa: Křemencova 10, Praha 1

## Omezení souborových systémů

Problémy při výpadcích však nejsou zdaleka jedinými nevýhodami tradičních souborových systémů jako ext2fs. Všechny byly navrženy v době, kdy záznamová média neměla takovou kapacitu jako v současné době. Dnes máme větší soubory, adresáře a také diskové oddíly a starší souborové systémy už nestačí, ať už z hlediska různých omezení velikostí nebo výkonu. Tyto problémy jsou důsledkem interních struktur, na kterých jsou založeny. Hlavním problémem je, že mají pevně danou délku, což limituje jejich možnosti. Také metody, které v dřívější době vyhovovaly, jsou už při dnešních možnostech z hlediska výkonu nedostačující. Souborové systémy nové generace jsou navrhovány tak, aby problémům omezení předcházely.

V následující tabulce si můžete porovnat omezení jednotlivých souborových systémů:

Filesystem	Max. velikost filesystemu	Velikost bloků	Max. velikost souboru
Ext2	4 TB	1 KB – 4 KB	2 GB
Ext3	4 TB	1 KB – 4 KB	2 GB
ReiserFS	16 TB	až 64 KB	$2^{10}$ PB <sup>*1</sup>
XFS	18000 PB <sup>*1</sup>	512 B – 64 KB	9000 PB <sup>*1</sup>
JFS	512 B / 4 PB <sup>*2</sup>	512 B – 4 KB	512B / 512TB <sup>*2</sup>

\*1) 1PB =  $10^{15}$  B

\*2) Maximální velikost souborového systému závisí na velikosti bloku dat, velikosti pro ostatní hodnoty dostaneme jednoduše pomocí trojčlenky.

## Optimalizace výkonu nových souborových systémů

### Vyhledávání volných bloků

Dalším problémem jsou struktury, pomocí kterých souborový systém hledá volné bloky při ukládání dat. Často se jedná o seznam, kde jsou udržována čísla jednotlivých volných bloků. UFS a ext2fs používá bitmapu, což je pole bitů, kde každý z nich odpovídá jednomu logickému bloku na diskovém oddílu. S narůstající kapacitou délka pole narůstá a výkon klesá.

Problémům, týkajících se hledání volných bloků dat, se vyvarujeme použitím tzv. "extents" a balancovaného stromu. Extents je skupina sousedících logických bloků, které jsou používány některými filesystemy. Deskriptor extents obsahuje 3 údaje:

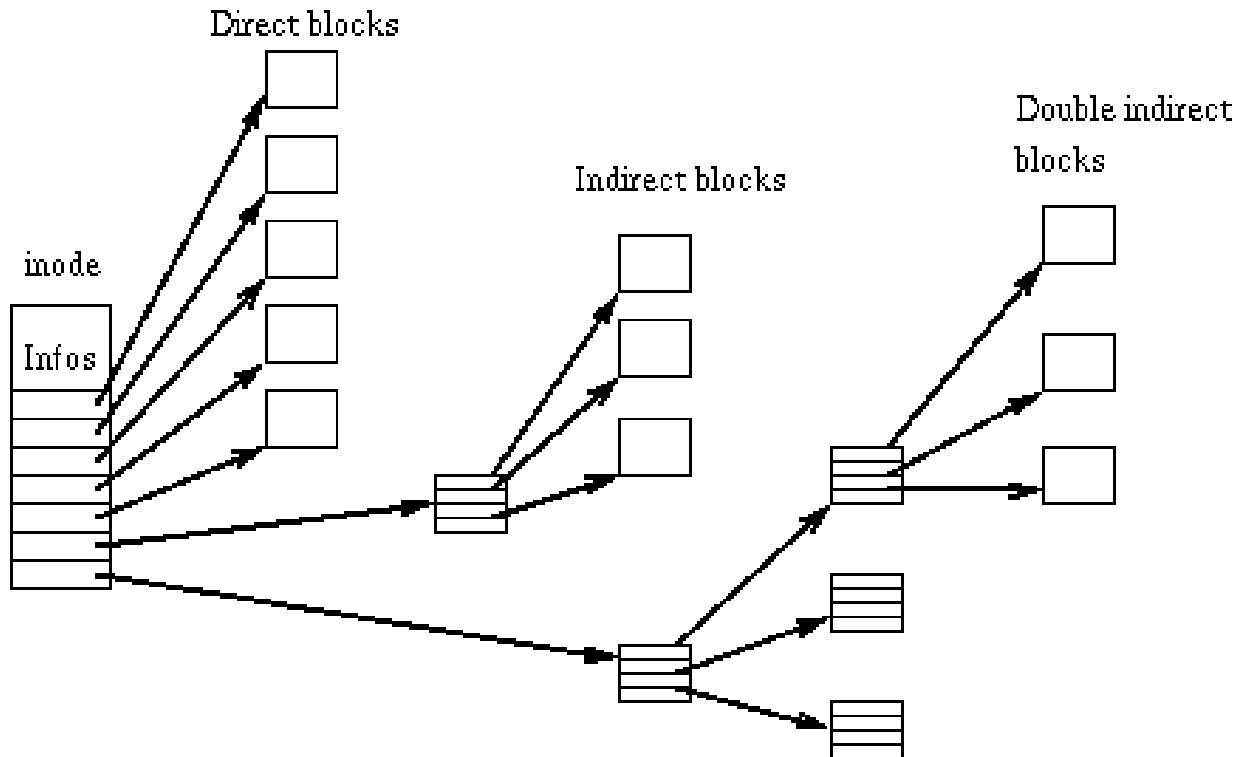
- adresa prvního bloku, kde extent začíná
- velikost v blocích
- údaj, který nám říká offset v rámci souboru, kde začínají data v daném záznamu uložená

V případě použití extents nezávisí velikost struktury, kde si uchováváme informace o volném místě, na velikosti filesystemu. Také použitím balancovaného stromu místo prostého seznamu dochází k dalšímu zvýšení výkonu.

### Problémy s velkými filesystemy

V případě velkého počtu položek adresáře je efektivita u starších souborových systémů opět slabší. Často jsou položky adresáře ukládány do seznamu, a tudíž jejich následné vyhledávání je zbytečně zdlouhavé. Jedním z řešení je opět použití balancovaného stromu, kde jsou tyto položky uspořádány podle jmen.

Výkonnost však nedostačuje ani u velkých souborů. Pro vysvětlení této oblasti si musíme nejdříve objasnit pojem i-node. Jedná se o strukturu, kde souborový systém udržuje informace o souboru, jako jsou práva, typ souboru a hlavně ukazatele na bloky souborového systému, kde je soubor uložený. Obsahuje jednak přímé ukazatele, a také tzv. nepřímé ukazatelé, odkazující na bloky s ukazateli přímými. Viz obrázek:



Problém spočívá v návržení struktury i-nody. Starší souborové systémy byly vytvářeny převážně pro práci s menšími soubory. Struktura i-nodů je proto ne zrovna efektivní. Čím větší soubor používáme, tím vícekrát přistupujeme k disku díky nepřímým pointerům. Důvodem, proč nepřímé pointerů vůbec ext2fs používá, je to, že i-node má pevnou velikost.

Problém velkých souborů může být odstraněn použitím dynamického alokování i-nodů. Bohužel musíme vyřešit otázku jak zařídit mapování logických bloků i-nodu a jaké použít struktury na vyhledávání v rámci i-nodu. Řešením je většinou použití balancovaných stromů.

## Řídké soubory

Omezení externí fragmentace a podpora řídkých souborů (sparse files) je také jednou z výhod nových file-systémů (ovšem jejich podpora je už v ext2fs). Pojem fragmentace většina čtenářů jistě zná, jen připomenu, že se jedná o rozptýlení jednotlivých bloků souboru po disku a hlavička disku pak při čtení musí přejíždět z místa na místo. Samozřejmě výhodnější je mít bloky za sebou, operace pak budou rychlejší.

Řídké soubory jsou soubory, které vzniknou následujícím způsobem:

Zapišeme například několik počátečních bytů dat a poté se chceme zapsat data uvnitř souboru, která odpovídají offsetu třeba 50 000. Pokud nemá náš filesystem podporu pro řídké soubory, alokujeme všechny byty mezi počátkem a těmi bloky uvnitř souboru. Bloky mezi počátkem a offsetem 50 000 nás však nezajímají a vůbec by alokovány být nemusely. Pokud použijeme souborové systémy s podporou řídkých souborů, pak se alokuje jen tolik místa, kolik zapisovaná data skutečně zabírají.

## Odkazy a zdroje

- <http://www.linuxgazette.com/issue68/dellomodarme.html>
- <http://www.linuxgazette.com/issue55/florido.html>
- <http://web.mit.edu/tytso/www/linux/ext2intro.html>

# Jak bylo v Seči

František Hucek

## Úvod

Celá akce začínala již ve čtvrtek večer, my jsme přijeli v pátek ráno. Cesta z Prahy autem trvala asi 2 hodiny. Ubytování bylo zajištěno v Juniorhotelu, což je rozsáhlý areál bývalé politické školy, v jejímž objektu se nacházejí přednáškové sály, školní jídelna, aj. Od pátku stál pobyt se stravou pro nečleny 1725 Kč.

Pokoje byly buňkového typu 3+3 osoby, naše počáteční obavy o velký počet lidí na jedno WC a sprchu se nakonec ukázaly jako neopodstatněné vzhledem k tomu, že jsme se na pokoji vyskytovali minimálně. Našeho spolubydlícího jsme potkali poprvé až v sobotu večer :). Školní jídelně, kde jsme se stravovali, se také co do kvality jídel a velikosti porcí nedalo nic vytknout.

Velmi jsme uvítali 270 stránek dlouhý sborník (sázený v T<sub>E</sub>Xu) se zpracovanými přednáškami, odkazy a kontakty na přednášející. Velmi mne potěšily samolepky Linux zdarma a legálně s Tuxem, které už máme nalepené na našich počítačích.

Přednášek bylo hodně, navíc někdy probíhaly zároveň přednášky o Linuxu i o T<sub>E</sub>Xu. Vybíráme tedy přednášky, na kterých jsme byli a které nás zaujaly.



## Pátek

### Jiří Kosina: Základy bezpečného programování nejen pro OS Linux

Zajímavá přednáška, kde byly na příkladech ukázány nejběžnější typy bezpečnostních chyb, kterých se programátor může při programování (nejen) pod Linuxem dopustit, se zaměřením na buffer overflow, chyby při programování řetězců, atd.

### **Cijoml: Bluetooth na Linuxu**

Přednáška očekávaná zvláště hardwaráři a těmi, co chtějí technologii Bluetooth použít pro svůj mobilní telefon. Cijoml, který o tomto tématu psal na AbcLinuxu a Mobilu, obecně představil technologii Bluetooth, včetně krátkého úvodu do historie vzniku tohoto projektu. Vysvětlil standardy a na názorných příkladech ukázal snahy některých výrobců mobilních telefonů o proprietární verze Bluetooth. Také ukázal script, který zamezuje tuhnutí Nokií při komunikaci s Bluetoothem. Živě bylo předvedeno připojení přes Bluetooth se zapůjčenou SIM kartou od Českého Mobilu.

### **Petr Adámek: Java a Linux**

Večerní přednáška, kde byla obecně popsána Java a konkrétně její použití pod linuxem. Juraj Michálek pak pozval na sezení SKLUG na jaře do Košic a na podzim do Bratislavy.

### **Sobota**

V sobotu jsme samozřejmě zaspali, takže jsme nestihli přednášku Ladislava Heringa Softwarový Raid pod Linuxem.

### **Ondřej Jakl, Karel Krečmer: Numerické výpočty na linuxovém clusteru**

Velmi zajímavá přednáška, pojednávající o konkrétním projektu výstavby linuxového clusteru v Ústavu geoniky AV ČR, určeného pro náročné numerické výpočty. Konkrétně se jednalo o klastr Thea postavený na běžném železe – 8 počítačích s procesorem Athlon 1,4 Ghz, 768 MB RAM, 20 GB HDD. Projekt byl popsán velmi detailně, od samotné myšlenky přes softwarové řešení po konečné nasazení v praxi. Na konec přednášky byly ukázány výsledky testů porovnávající tento klustrový počítač s dalšími počítači postavenými na jiné technologii. Cluster – kolekce samostatných počítačů (uzlů), schopných svébytné existence, spojených do jednoho funkčního celku. V současnosti tato řešení nahrazují počítače používající drahý proprietární hardware a software.

### **Martin Grombřík: Linux do škol a otevřené systémy na středních školách**

Nejdříve na konkrétním příkladu bylo ukázáno, jak taková počítačová výuka na střední škole s použitím Open Source aplikací funguje. Následovně pan Grombřík, který v současnosti vede internetové stránky anti-indoš, vysvětloval výhody použití Open Source aplikací ve školství. Zmiňoval se mimo jiné o tom, že v původním plánu Internetu do škol bylo vytvořit multiplatformní prostředí, jenž je v rozporu se současnou realizací projektu. Další významným argumentem byly finanční úspory.

Na dotaz, kdo z přítomných je nějak zainteresován ve školství se zvedlo několik málo rukou, jeden z účastníků vtipně dodal: "všichni, protože tam chodí naše děti", čímž myslím velmi přesně vystihl závažnost tohoto tématu. V celkem bouřlivé diskusi se např. řešilo to, že Autocont (výhradní dodavatel projektu InDoš) kromě toho, že používá software hlavně jedné známé softwarové firmy, nedovoluje instalovat bez povolení žádný jiný software. Na to je tu jakási "certifikační autorita", která za poplatek povolení vydá či nevydá, s tím že některé firmy jsou si opět rovnější než ty druhé. Byl představen velmi mladý projekt Jednoty Školských Informatiků, jenž se daným problémem zabývá.

### **Alois Vitásek: Aplikační server s XML-RPC rozhraním**

Přednáška o protokolu sloužícímu k jednoduchému a přenositelnému vzdálenému volání funkcí.

### **Jiří Kosek: DocBook**

XML/SGML formát pro tvorbu dokumentace, který používá mnoho open-source i komerčních projektů.

## Neděle

### **Petr Olšák: C<sub>S</sub>T<sub>E</sub>X – historie, současný stav a budoucnost**

Asi vedoucí osobnost co se české verze T<sub>E</sub>Xu týče zrekapituloval vznik, okolnosti vzniku a české verze T<sub>E</sub>Xu, jeho současný stav a naději, že už na něm nebude muset nic měnit :).

### **Marcel Kolaja: Kompexní řešení pošty za použití OSS**

Pan Kolaja z firmy Solnet, dle vizáže nejmladší z přednášejících, na konkrétním příkladě, který demonstroval na svém notebooku, předvedl řešení elektronické pošty pro organizace na bázi OSS. Použity byly programy Exim, OpenLDAP, Apache, Horde, IMP, Turba...

### **Martin Mareš: Podivuhodný svět Linuxu, kapitola 2.5**

Po tom, co byl organizátory přednášející nalezen :), měl přednášku, ve které se věnoval nově vyvíjenému jádru 2.5, popisoval problémy, které má současné jádro a které by verze 2.5 měla vyřešit.

## Závěr

Tři dny velmi kvalitních přednášek za velmi příznivou cenu, možnost setkání s lidmi z linuxové komunity z celé ČR a dvou ze Slovenska. Nechybělo ani spadnutí aplikace pod Windows :), pod kterými běžela asi polovina notebooků určených na přednášky.

Návštěvníci přednášek měli možnost zakoupit literaturu týkající se převážně Linuxu od knihkupectví Mareček s 10% slevou. Těšíme se na další setkání...

## Sborník

Zájemci o sborník ze semináře SLT 2002 si jej mohou objednat v knihkupectví Mareček (<http://marecek.kup.to>) nebo v nakladatelství Konvoj (<http://www.konvoj.cz>). Zájemci z Prahy si mohou zakoupení sborníku domluvit s Petrem Olšákem ([olsak@math.feld.cvut.cz](mailto:olsak@math.feld.cvut.cz)). Cena sborníku je 190 Kč.

# Cracking bez tajemství

Michal Vymazal

## Úvod

Musím říci, že při psaní této recenze jsem dlouho váhal. Není to dáno ani tak faktem, že bych snad nevěděl, co napsat. Je to dáno spíše zamyšlením nad rozdílností několika světů, mohu-li tak nazvat ono jádro pudla. V podstatě jde o to, že kniha samotná by se spíše měla jmenovat Reverzní inženýrství ve světě MS-Windows. Cítíte ten rozdíl? Teď už zkrátka nebudete očekávat něco o síťových službách a protokolech, nebudete očekávat nezávislost na operačním systému. Zkrátka víte, že dostanete jen střípek mozaiky a víte, který to bude. Ale pojďme pěkně popořádku.

Knihka je poměrně pěkně napsaná, nicméně nepopisuje to, co slibuje. Slibuje totiž cracking bez tajemství, ale cracking je poměrně široký pojem. Pokud by obsahovala již výše uvedené, pak si troufám tvrdit, že je výborným doplňkem domácí knihovny každého profesionála nebo fandy z oblasti informačních technologií. Ovšem zde musím autorovi vytknout, že zaměňuje pojmy hacker, cracker a reverzní inženýrství. Jde skutečně o tři samostatné oblasti a dobrý popis nalezneme (v důvěrně známém) jargon file: <http://www.tuxedo.org/~esr/jargon/jargon.html#hacker>. Vysvětlení pojmu Reverse-engineering na Jargon File nenajdete a zde nabídnu jednu malou soutěžní otázku: Co myslíte, proč? Ale nebuďte smutní. Základní vysvětlení můžete nalézt na adrese <http://adam.wins.uva.nl/~x/reeng/REanno.html>.

## Obsah

Jak jsem již uvedl dříve, naleznete zde poměrně hezky vysvětlené příklady z reverzního inženýrství. Řada věcí je bohužel silně závislá na prostředí MS-Windows, ale to je již chyba prostředí, ne autora. Nechybějí ani názvy a popis jednotlivých nástrojů a odkazy na jejich stránky.

Postrádám však odkazy na normy, standardy apod. Hodně je to znát v kapitole o protikopírovacích ochranách datových CD, kde by nebylo špatné uvést nějaký ten odkaz na ISO 9660, joliet, rockridge, (co že to používá MAC?) a další. Podobný stav je i v kapitole o šifrování. Jsou zde sice zmíněny šifrovací algoritmy (např. BlowFish, DES, 3DES, chybí zde RSA, které je samostatně vysvětleno později), ale opět chybí odkazy na normy, standardy, stránky, RFC, apod. Škoda. U DES si dovolím podotknout, že prolomitelný sice je, ale nějak v knize postrádám právě ty podmínky, za nichž by se vám prolomení mohlo i podařit. Určitě to nestihnete na zastávce autobusu :-)

Hezkým zpestřením je i kapitola o známých crackerech a crackerských skupinách. Opět se zde projevuje rozdílnost světů. Já zde například postrádám Binary Division...

## Hodnocení

Co dodat závěrem? To hlavní, kniha nepojednává o crackingu, ale o reverzním inženýrství. Druhým hlavním faktem je, že kniha je bohužel zaměřena jen na jeden operační systém a tím je MS-Windows (dobře, tak řada MS-Windows). Kniha je dobře napsaná a pokud se přenesete přes fakt, že jaksi není univerzální (ruku na srdce, reverzní inženýrství není univerzální záležitost) a příklady jsou zaměřeny jen na jeden operační systém, pak bude vhodným doplňkem vaší knihovny.

Název:	Cracking bez tajemství
Autor:	Jakub Zemánek
Vydal:	Computer Press
Počet stran:	313
Doporučená cena:	369 Kč / 553 Sk
Datum vydání:	2002



# Jaderné noviny 188–192

Leoš Literák

## Úvod

Následující článek shrnuje nejzajímavější témata z konference linux-kernel. Na tomto místě vývojáři v čele s Linusem Torvaldsem diskutují nad otázkami kolem vývoje a budoucnosti jádra Linuxu (kernelu). Článek je zkráceným překladem seriálu Kernel Traffic a je zveřejněn pod licencí GPL verze 2. Originál v angličtině včetně archivu najdete na adrese <http://kt.zork.net>.

## Feature Freeze v říjnu

Linus Torvalds ohlásil jádro 2.5.40 a přitom zavedl řeč na feature freeze (zákaz přidávání nových vlastností):

Zároveň připomínám, že jsme oficiálně v posledním měsíci nových vlastností a protože budu poslední týden v říjnu mimo, osobně považuji 20. říjen jako poslední den, pokud na sobě nebudete mít skutečně dobrý a strašidelný kostým . . . Tak to nenechávejte na poslední den.

Pokud to není jasné lidem nepoužívajícím jádra řady 2.5, tak ano, MĚLI byste testovat kód ještě před zmražením. IDE vrstva už není tak hrozná a přestože tu jsou problémy jako sestavy, kde se jádro nedá zkompileovat, tak je obecně dobrým nápadem vyzkoušet tyto věci, než bude příliš pozdě na stěžování si.

Většinou si lidé nestěžovali na zkrácení termínu, jen Hans Reiser napsal, že by bylo lepší, aby se posunul termín, když si Linus bere volno. Mají problémy s dokončením reiser4 před původním termínem o Halloweenu a pracují na něm do noci. Reiser4 je dramaticky lepší a rychlejší než reiserfs. Nicméně nikdo ani Linus na tento email nereagoval.

Někdo napsal, že by rád vyzkoušel nová jádra, ale bojí se korupce dat na disku. Linusův soukromý názor byl, že šance na ztrátu dat je malá. Je možné, že některé ovladače nepůjdou zkompileovat nebo někteří lidé budou mít problémy při bootování, ale dlouhou dobu už nikdo nehlásil poškozený disk. Samozřejmě Linus nemůže říci, že se tak nestane, ale pravděpodobnost není velká.

## Plug and Play Layer 0.6

Adam Belay ohlásil verzi 0.6 podpory Linux Plug and Play, což je přepis aktuálního ovladače. Obsahuje spoustu vylepšení, jakými jsou:

### 1. Globální vrstva pro Plug and Play

- Ovladače se nyní už nemusí obávat o to, který Plug and Play protokol používají. Volání jdou do vrstvy Linux Plug and Play a pak jsou poslány správnému protokolu. To zjednoduší integraci podpory ACPI PnP, až bude připravena.

### 2. Úplný ovladač Plug and Play BIOS

- Plug and Play BIOS nyní podporuje čtení a zápis konfigurací zdrojů [resource].
- Nyní je možné povolit zakázaná [disabled] PnP BIOS zařízení. Takže uživatelé mohou bezpečně povolit podporu PnP OS support v BIOSu.

### 3. Integrace modelu ovladače [driver model]

- Celá vrstva PnP je integrována do driver modelu.
- Uživatelská rozhraní se nacházejí zde.
- PnP protokoly jsou vypsané pod sběrnici [bus] pnp

#### 4. Mocné rozhraní pro konfiguraci globálních zdrojů

- Pro aktivaci zařízení má uživatel spustit následující příkaz ve výpisu driverfs:

```
# echo "auto" > resources
```

- Pro manuální nastavení spusťte následující příkaz:

```
# echo "manual <depnum> <static/dynamic>" > resources
```

<depnum> = configuration number

<static> = pro příští bootování, <dynamic> = nyní

příklad:

```
# echo "manual 1 dynamic" > resources
```

- Zařízení vypnete takto:

```
# echo "disable" > resources
```

#### 5. Automatická alokace zdrojů pro potřebné moduly

#### 6. Databáze jmen PnP zařízení

### Driverfs mění jméno

Patrick Mochel napsal, že driverfs je neskutečně proměnlivý souborový systém. Takže jak bylo diskutováno na Jaderném summitu v Ottawě a jak bylo vyhrožováno poslední tři měsíce, tento patch přejmenovává driverfs na kfs. Upravte si svůj `/etc/fstab`. H. Peter Anvin napsal, že by preferoval kernelfs nebo kernfs a Patrick se zeptal na Linusův názor.

### Další stabilní řada: 2.6 nebo 3.0?

Během diskuse se Jeff Garzik zeptal Linuse Torvaldse, zda je už rozhodnuto, že nová řada se bude jmenovat 2.6. Jeff cítil, že spíše by se hodilo pojmenování 3.0. Linus odpověděl, že neexistuje opravdický důvod, proč jej tak nazvat. Vylepšení vláken o několik řádů je blízko Další velké věci, jenže jádro neobsahuje novou architekturu ani jiné fundamentální věci. Přejít mezi 2.2 a 2.4 byl větší než bude mezi 2.4 a 2.6.

Ingo Molnar argumentoval, že vylepšení virtuální paměti a I/O patří mezi nejdůležitější změny za posledních pět let. Konečně máme špičkový VM a IO subsystém, takže bychom to měli uživatelům dát najevo. Pokud toto si nezaslouží změnu hlavního čísla, tak v budoucnu už nic jiného. A stejně bychom za 2–4 roky museli přejít na 3.0, protože po řadě 2.8 jiné číslo nebude dostupné.

J. W. Schultz poznamenal, že přechod z 1.0 na 2.0 byl zapříčiněn změnami v Application Binary Interface (ABI) [LL: podle mých vzpomínek spíše zavedením modulů]. Podle něj řada 2.5 obsahuje příšerné zrychlení, ale nepřidává žádné důležité nové vlastnosti ani neodstraňuje stará rozhraní. Takže 2.6 bude upgrade, zatímco 3.0 bude nahrazení.

Vzápětí si Linus začal dělat legraci, že kvůli virtuální paměti klidně nazve novou řadu 5.0. Pak však vyzval veřejnost, aby se zapojila do testování řady 2.5, jinak není možné ověřit správnost změn. Někteří lidé se však ozvali, že se obávají nestability IDE. Linus a další se je snažili uklidnit, že řada 2.5 jim jede bez potíží.

### Paketový zápis

Jure Repinc se zeptal na to, zda bude nové stabilní jádro obsahovat podporu paketového zápisu. Rád by viděl podporu Mount Rainier, který usnadňuje práci s CD-RW. Nějaké patche napsal Jens Axboe a Jure chtěl vědět, zda budou použity. Jens odpověděl, že patche potřebují menší úpravy, ale blížící se zmražení (nových funkcí) kernelu není problém, neboť se jedná jen o drobné vylepšení ovladače cdrom. To potěšilo Jureho, který urgoval o tuto práci. Andre Hedrick také reagoval:

Zatlačil jsem na právní oddělení v Mt-Rainer a pokud se nesnažíme provádět ne-nativní operace, je vše jasné. Přesněji nepotřebujeme licenci pro čtení a zápis MRW medií na MRW zařízeních. Nezákonné je ale číst MRW média na CD-RW.

## Souborový systém CIFS

Steven French ohlásil, že CIFS byl upraven tak, aby reflektoval nedávné změny do struktury superbloku. Dále byl vytvořen repozitář BitKeeperu pro snadnou aplikaci na aktuální jádro (<bk://cifs.bkbits.net/linux-2.5-with-cifs>). Jedná se o poměrně bezpečnou změnu, neboť se téměř nedotýká hlavních jaderných souborů. Jeremy Allison odpověděl, že by hrozně rád viděl CIFS začleněný do řady 2.5, neboť se jedná o velmi hezky napsaného moderního CIFS klienta, díky němuž bychom mohli začít přidávat do Samby Unixu specifická rozšíření a nakonec skončit u souborového systému s unixovou sémantikou při komunikaci se Sambou a s windowsovou sémantikou při komunikaci s obyčejným Windows serverem. Takto by bylo možné dosáhnout stavu, kdy Linux bude univerzálním klientským lepidlem.

## Sdílené správcovství IDE

Andre Hedrick napsal, že se rozhodl jmenovat Bartłomiej Zolnierkiewicz svým zástupcem a že společně budou spravovat IDE. Bartłomiej je člověk, kterému Andre bezvýhradně věří. Úkolů je totiž tolik, že je skutečně třeba vytvořit tým. Bartłomiej pozvání přijal.

## Linux Kernel conf 1.0

Roman Zippel ohlásil verzi 1.0 konfiguračního nástroje pro kompilaci jádra. Finální verze je k dispozici ke stažení na adrese <http://www.xs4all.nl/~zippel/lc/>. Roman pak předal tento projekt do rukou Linuse, aby jej použil ve svém stromu a překonvertoval existující data. Projekt se ještě jmenuje kconfig, ale Linus jej může snadno změnit na cokoliv jiného.

## Vypalování na IDE

Linus Torvalds zaslal oznámení pro jádro 2.5.43. V něm se zmínil, že strávil pár dní nad odstraněním nutnosti používat modul ide-scsi a SCSI vrstvu pro vypalování CD na IDE zařízení. Je třeba ještě provést změny v programu cdrecord, ale ty už Linus poslal jeho správci.

Bill Davidsen doufal, že stále bude fungovat vypalování pomocí ide-scsi, neboť spousta lidí v reálném životě používá řadu 2.4 a testují 2.5. Překonfigurovat systém, aby používal ide-scsi nebo ne, je další překážkou, kterou bychom museli překonat. Jenže Linus potvrdil, že ide-scsi funguje tak, jako vždycky, tedy nijak zvlášť dobře. Nativní IDE implementace je prostě čistší a jednodušší a hlavně zatraceně snadnější k používání.

## Nový bezpečnostní projekt

Bosko Radivojevic napsal:

Tým LinSecu hrdě oznamuje první stabilní verzi LinSecu, což je akronym pro Linux Security Protection System. Jeho hlavním smyslem je včlenit do Linuxu mechanismus Mandatory Access Control (MAC) (na rozdíl od existujícího mechanismu Discretionary Access Control). Model LinSecu je založen na:

- Capabilities
- Domény pro přístup k souborovému systému [Filesystem Access Domain]
- IP Labeling Lists
- Řízení přístupu k socketům

Díky Capabilities Linux silně rozšiřuje nativní model capabilit, čímž umožňuje jemně řízenou delegaci jednotlivých capabilities uživatelům a programům. Nikdy více všemocný root!

Filesystem Access Domain dovolují omezit dostupnost částí souborového systému uživatelům či programům. Nyní můžete omezit uživatelské aktivity na jeho domovský adresář, poštovní schránku, apod. Domény pracují na úrovni zařízení, adresáře či souboru.

IP Labeling lists omezují síťové konekce na úrovni programů. Od nynějška je možné nastavit politiku, že nikdo kromě vašeho oblíbeného MTA se nesmí připojit na vzdálený port 25.

Socket Access Control model poskytuje jemnou kontrolu nad sockety tím, že asociuje ke každému socketu množinu capabilities nutných pro připojení lokálního programu k socketu.

LinSec se skládá z patche do jádra (aktuálně 2.4.18) a uživatelských nástrojů. Více na naší domovské stránce <http://www.linsec.org>.

## Účtování zdrojů

John Hesterberg zaslal oznámení na verzi 2.5.43 projektů CSA, PAGG a Job. PAGG (Process Aggregates) je obecný framework pro implementaci kontejneru procesů, Job je kontejner procesů, ze kterého se nedá uniknout a používá se většinou ze vstupního bodu, jakým je login. CSA (Comprehensive System Accounting) shromažďuje informace o používání zdrojů jednotlivými procesy. Tako můžete účtovat poplatky jednotlivým uživatelům podle využití systémových zdrojů.

Patche najdete na adresách <http://oss.sgi.com/projects/csa/> a <http://oss.sgi.com/projects/pagg/>.

## Detekce zařízení na AGP 3.0

Matthew E Tolentino zaslal patch pro detekci generických AGP 3.0 zařízení a patch obsahující podporu pro čipset Intel 7505. Patch dokáže detekovat vícenásobná [multiple] zařízení a správně alokovat pásmo pro každá z nich a dále umí provádět typickou inicializaci host bridge.

## Nový vysoce komprimovaný souborový systém

Phillip Lougher ohlásil:

První verze squashfs. Squashfs je vysoce komprimovaný souborový systém určený jen ke čtení. Používá metodu zlib pro kompresi souborů, inodů a adresářů. Inody jsou velmi malé a všechny bloky jsou shlukovány pro minimalizaci režie dat. Jsou podporovány velikosti bloků od 4 KB po 32 KB.

Squashfs je určen pro použití jako read-only filesystem, pro archivaci a pro embedded zařízení, kde je vyžadována malá režie. Najdete jej na adrese <http://squashfs.sourceforge.net>. Patch je dostupný pro jádro 2.4.19.

Samuel Flory se zeptal, jak se squashfs liší od existujícího projektu cramfs a Phillip vysvětlil:

Cramfs byl inspirací pro squashfs. Squashfs obecně dává lepší kompresi, podporu větších souborů a souborových systémů a obsahuje více informací v inodu.

1. Bloky až po 32 KB – data komprimovaná po jednotkách o velikosti 32 KB získají lepší kompresní poměr než o velikosti 4 KB. Obecně je použití více než 4 KB bloků špatný nápad, neboť volání VFS používají 4 KB stránky, ale Squashfs explicitně tlačí extra bloky do cache stránek.
2. Squashfs komprimuje kromě dat ze souborů navíc informace z inodů a adresářů. Tyto údaje obvykle mají kompresní poměr 50%, jinými slovy režie 8 a méně bytů na inode.
3. Všechna data jsou zarovnána na byte [byte alignment], což opět ušetří pár bytů na inodu či adresáři.
4. Podpora 32 bitových uid/guid, velikost souboru až  $2^{32}$ , časové značky [timestamp]. Cramfs zkracuje uid na 16 bitů, guid na 8 bitů, velikost souboru do  $2^{24}$ , žádné časové informace.
5. a další...

## Swap-Space Mini-Howto

Randy Dunlap byl překvapen, když nenašel žádné mini-howto, které by pokrývalo swap a tak jedno vytvořil a zaslal na něj dočasný odkaz: <http://www.xenotime.net/linux/swap-mini-howto.txt>. Pár lidí bylo překvapeno, že nic takového skutečně neexistuje, jiní si návod přečetli a nabídli doplnění. Pár lidí doporučilo Linux Documentation Project jako správné místo, kam jej uložit.

## Placení za opravy

Alan Cox zaslal patch a vysvětlil:

Čím dál více firem se na mne obrací s tím, že hledají lidi, kteří by jim opravili za peníze chyby v Linuxu. Je jasné, že zaslání \$1000 za opravu ovladače nebude fungovat, když se jedná o velké společnosti. A tak jsem se zeptal v konferenci linux-kernel a dostal jsem pár nabídek. Toto je seznam lidí, kterým můžete zaplatit, aby něco opravili. Seznam je řazen podle abecedy.

Linus Torvalds odpověděl:

Rád bych viděl explicitní požadavky po jednotlivcích nebo firmách (méně než X lidí) nebo jiný způsob, jak zaručit, že ta věc bude vyvážená a že nastavíme správná očekávání (od uživatelů i účastníků seznamu). Dále, je tato konference opravdu správným místem pro takový seznam, když vezmeme v úvahu, že neexistuje způsob, jak odstranit problematické záznamy z vypuštěných jader? Nebylo by lepší umístit jej na web a do jader jen přidat odkaz?

Alan odpověděl:

Správná poznámka. Rád to umístím na web. Rozdělení podle společností je také snadné, například na sekce Zajímáme se o kontrakty pod \$1000, \$10000, \$100000, \$1M.

## Pokus protlačit Reiser4 po zmražení jádra

Hans Reiser se pokusil přesvědčit Linuse Torvaldse, aby začlenil Reiser4 do řady 2.5 i po zmražení jádra vůči novým vlastnostem:

Proč začlenit:

- dvojnásobná výkonnost (viz [http://www.namesys.com/v4/fast\\_reiser4.html](http://www.namesys.com/v4/fast_reiser4.html)), bylo měřeno na zápisu a čtení adresáře s jádrem
- umožní testovat a měřit změny ve VM a VFS na nejrychlejším souborovém systému
- provádí všechny operace jako atomické transakce, takže například `write()` či `truncate()` se buď provedou kompletně nebo vůbec.
- poskytuje infrastrukturu pro atomické fs transakční API
- škálovatelné na velký počet CPU
- eliminuje pevnou velikost žurnálovací oblasti
- poskytuje pluginovou infrastrukturu. Tak bude možné vytvořit vlastnosti jako omezení či dědičnost v podobě pluginu. Nebo nové bezpečnostní atributy budou mít podobu souboru s pluginem
- první funkční soupeř microsoftího projektu OFS. Žádný jiný linuxový souborový systém se ani nesnaží být alternativou OFS.

Jsme vzrušeni dramatickým nárůstem výkonnosti u atomických transakcí i infrastrukturou pluginů. Tato funkčnost zabila výkon u jiných souborových systémech. Jednou jsi [Linusi] souhlasil, že souborové systémy by měly mít 6 týdnů poté, co se VM/VFS stabilizuje. Je mi líto, že ti to musím připomenout. Reiser4 ale nemohl být hotov dříve. Doufám, že můj kolega domorodec se po mém návratu z lovu podívá na mamuta na mých zádech a odpustí mi, že jsem přišel pozdě na večeri, myslíte na ubohý a hladový MS kmen :-).

Linus napsal pár poznámek svědčících o respektu vůči Reiser4, nicméně se nevyjádřil, zda jej začlení.

## EVMS mění směr

Kevin Corry ohlásil:

Chceme oznámit důležitou změnu ve směřování projektu Enterprise Volume Management System. Jak jistě víte, řada 2.5 byla zmrazena pro nové vlastnosti a zdá se jasné, že jaderný ovladač EVMS nebude začleněn. Majíce toto na paměti rozhodli jsme se přepracovat administrační nástroje v uživatelském prostoru (Engine), aby pracoval s existujícími ovladači včetně device mapperu a MD.

Proč tato změna? Protože se EVMS nedostala do řady 2.5, jeho budoucnost se stala velmi nejistou. Mohli jsme pokračovat ve vývoji a dodávat patche vůči aktuálním jádrům. Dostali jsme spoustu užitečných komentářů a změn za poslední měsíc. Mohli jsme trávit čas nad změnami a opravami. Nicméně jedna věc nebyla zmíněna: do jádra integrovaný mechanismus vyhledávání EVMS svazků [volume]. Věříme, že i kdyby se vše ostatní vyřešilo, později bychom narazili na tento problém. Jasně, mohli bychom vyhledávání přesunout do uživatelského prostoru, pak by ale bylo obtížné odlišit ovladač EVMS od device mapperu, neboť oba provádějí podobné úkoly.

Navíc pak by nebyl důvod duplikovat kód MD, abychom poskytovali kompatibilitu s existujícími softwarovými RAIDy. Když se objevování svazků přesune do uživatelského prostoru, nástroje EVMS mohou být snadno přepsány tak, aby komunikovaly s existujícím MD ovladačem. Chceme zdůraznit, že tato změna nebyla rozhodnuta unáhleně, uvažovali jsme o ní již od Linuxového sympósia v Ottawě v červenci. V posledních týdnech se však ukázalo, že aktuální přístup není v souladu s tím, co jaderná komunita chce a tak jsme se začali zabývat náročností a důsledky změny.

Jak se to projeví u EVMS uživatelů? Ideálně nijak. Neplánujeme změny v uživatelském rozhraní. Změny budou transparentní. Stejně funkce, pluginy i schopnosti. Samozřejmě budou menší změny. Instalace bude mírně odlišná, neboť bude vyžadovat jiné volby v jádře. Řada 2.5 obsahuje vše potřebné, takže nebude třeba ji patchovat. Neboť device mapper nebyl ještě začleněn do řady 2.4, tak její uživatelé budou muset patchovat.

Důležitou odlišností ale bude přesun objevování svazků do uživatelského prostoru. Proč? I když například softwarový RAID stále provádí objevování v kernelu, trendem je přesun do uživatelského prostoru. Takže i RAID se jednou bude muset přizpůsobit. Nevýhodou je ztráta automatického objevení svazků při bootu. Aktivace svazku nyní vyžaduje zavolání utility, které musí být přidáno do init skriptů.

Toto přepnutí však komplikuje možnost mít root na EVMS svazku. To se dá vyřešit pomocí initramfs z řady 2.5, což poskytne snadnou metodu, jak na to. Prostě byste přidali nástroje z EVMS do initramfs. Uživatelé 2.4 musí vytvořit ram disk přes initrd. Postup je bohužel často specifický jednotlivým distribucím. Budeme se snažit poskytnout dostatečné instrukce.

Alan Cox napsal, že vyhození velkého kusu kódu je špatné, nicméně vývojáři EVMS jdou správným směrem. Výsledkem bude čistší framework. Alan ocenil, že udělali to, co je pro Linux nejlepší, přestože pro ně osobně to musí být hořké zklamání.

Alexander Viro také reagoval:

Toto rozhodnutí (přesunout kód do uživatelského prostoru a izolovat změny v jádře) určitě znamená z dlouhodobého hlediska méně práce – zvláště pro vývojáře EVMS. Je mnohem snadnější psát v uživatelském prostoru. Máte k dispozici runtime prostředí a to samo o sobě znamená moc. Není zde žádné omezení 8 KB na zásobníku. Je tu ochrana paměti. Nemusíte se obávat změn ve vnitřních strukturách jádra. A dá se zde mnohem snadněji ladit kód v debuggeru.

Cílem je poskytovat funkčnost, ne být v jádře – to druhé znamená mnohem těžší život. Toto rozhodnutí vyžadovalo velkou odvahu – tleskám jim. Je úžasné, co udělali, mají můj respekt, že si zvolili správný směr a dokázali jej obhájit. Bravo. Hodně štěstí – opravdu. Vážím si jen několika lidí. Těchto lidí si vážím. Velmi.

Mike Diehl byl zklamán tímto vývojem, neboť EVMS se mu jevila jako dobrá alternativa k LVM. Matthias Andree se zeptal na stav LVM2. Joe Thornber odpověděl, že Sistina má nové stránky: <http://www.sistina.com>.

## Testy IDE-CD

Adam Kropelin vzrušeně oznámil:

Včera jsem se nudil a tak jsem se rozhodl, že je ten správný čas otestovat ide-cd s cdrecordem. Většinou to fungovalo skvěle! Po upgradu cdtools šlo všechno hladce. Běžel jsem pod SMP + preempt a byl jsem

schopen pálit rychlostí 12x při méně než 2% zatížení CPU po celou dobu. Kapacita vyrovnávací paměti nikdy neklesla pod 98%.

Skoro jsem byl zklámán, jak všechno jede a tak jsem se rozhodl to trochu vystresovat. Kompiloval jsem tedy jádro 2.5.46 příkazem:

```
make -j10 bzImage
```

zatímco jsem páčil a ono to opět fungovalo! Buffer nikdy neklesl pod 96%. (Stroj je 2xXeon 450, 256 MB RAM).

A tak jsem paralelně při dalším pálení spustil

```
dd if=/dev/zero of=foo bs=1M
```

Konečně se to povedlo ;-) Tento zápis přebil pálení a po vyčerpání vyrovnávací paměti se překvapivě objevilo oops.

Jens Axboe byl rád za tuto informaci a požádal o informaci o hardwaru. Adam poslal, že harddisk je sdc na integrovaném řadiči AIC7xxx, vypalovačka je hdc na sekundárním IDE kanálu. Ostatní disky nebyly během testu používány. Jens napsal, že u SCSI je velice snadné vyhledovět požadavky [starve requests]. Poprosil jej, ať nastaví maximální počet tagu na 2-4 a pak nebude schopen zabít pálení. Mimochodem, více než 4 tagy nezlepší výkonnost, zato riskujete příšerné výluky.