



Abíčko

Časopis serveru AbcLinuxu.cz

Červenec 2004

unixshop®

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 [AbcLinuxu.cz](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 kdokoli, tedy i vy.

Na [AbcLinuxu.cz](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.

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 [AbcLinuxu.cz](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>.

©2004 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
Kompilovanie jadra	4
Načo jadro vlastne je	4
Prečo?	4
Kompilovanie jadra	6
Kde zobrať zdrojový kód jadra?	6
Konfigurácia jadra	7
Moduly áno, či nie	11
Spustenie kompilácie	11
Čo sa ešte oplatí vedieť?	12
Ako funguje e-mail	14
Úvod	14
Posielanie pošty	14
SMTP server	14
Prijímanie pošty	16
Local	16
POP	16
IMAP	17
Čo nájdeme v mailovej správe	19
Hlavička	19
Telo	20
GIMP – Grafický editor pro Linux a Windows	21
Na co se často ptáme 4: X Window System	23
XFree 4.3.0	23
Konfigurace	23
Struktura konfiguračního souboru	23
Syntaxe konfiguračního souboru	24
Popis jednotlivých sekcí	25
ServerLayout	25
Files	25
ServerFlags	26
Module	27
InputDevice	28
Device	29
VideoAdaptor	31
Monitor	31
Modes	33
Screen	33
DRI	34
Uživatelské dotazy	34
Modelines	34
Chybové zprávy	35
Automatické spouštění XFree	35

Jak shodit XFree?	36
Instalace karty	36
Program lspci	36
Nálepka	37
Samotný X server	37
Změna rozlišení za běhu	37
Někdy musí nastat konec	38
Diskové kvóty	39
Úvod	39
Jak to pracuje	39
Hard limit	39
Soft limit	39
Inodes, Blocks	39
Grace period	39
Nutné ingredience	40
Základní konfigurace, spuštění	40
Nastavení pro jednotlivé uživatele	41
Spolupráce s Windows	42
Závěr	42
Extra multimediální systém XMMS	43
Pluginy	43
Vstupní pluginy	43
Výstupní pluginy	44
Efektové pluginy	44
Vizualizační pluginy	45
Všeobecné pluginy	45
Nastavení	46
Skiny	46
Rozhodně je co zlepšovat	46
Komix: RMS v Praze	48
Jaderné noviny 260–263	49
Zrychlování SATA, 115 e-mailů	49
Ext3 pre-alokace založená na rezervaci, 12 e-mailů	50
Správa SCSI ovladače atp870u, 2 e-maily	50
Správcovství BTTV, 3 e-maily	50
Porušení GPL firmou Tritton Technologies, 3 e-maily	51
LISP interpret v kernelu, 84 e-mailů	51
Obejití bufferové keše, 32 e-mailů	52
Ovladač adm8211 802.11b pro 2.6, 1 e-mail	53
Skript pro patchování jádra – ketchup – ve verzi 0.5, 2 e-maily	53
Přidání podpory SysFS do DVB subsystému, 3 e-maily	54
Crypto CRC32C modul, 11 e-mailů	54
Odstraňování IDE modulů; IDE a libata, 11 e-mailů	54
Opravy SiS ovladače, 2 e-maily	55
Ovladač AHCI; otevřené hardwarové specifikace, 2 e-maily	55
LinuxAnt lze o své licenci, 2 e-maily	56
Opravy a aktualizace ReiserFS verze 3, 22 e-mailů	56
Podpora tabulky atributů stránek v 2.6, 20 e-mailů	57
SATA bude ve 2.4.27, 27 e-mailů	58

Automatický hledač chyb odhalil v jádře několik děr, 19 e-mailů	58
Dokumentace ke kompilaci externích modulů, 4 e-mailů	58
Kompilace externích modulů	59
Kompilace modulu	59
Triky pro zjednodušení	60
Příprava jádra pro kompilaci externích modulů	61
Verze modulů	61
Lokální include soubory	61
Pouze binární .o soubory	61
Zprávičky	63

Kompilovanie jadra

Rastislav Stanik

Co je to vlastně jádro, kdy je potřeba jej kompilovat a jak na to jít. Způsoby konfigurace, kompilace a parametry, instalace. A několik šikovných tipů.

Načo jadro vlastne je

Keď sedíte za počítačom a čítate tento článok, robíte tak pomocou prehliadača a možno vám do toho počítač hrá nejakú muziku a tlačí na tlačiarňu dokument. Za tým všetkým je ale schované niečo iné. Nenápadné. Jadro. Jadro (angl. kernel) má tieto hlavné úlohy:

1. Štartovanie systému
2. Ovládačmi sprostredkováva prístup k hardwaru
3. Poskytuje programom prostriedky (angl. resources) ako sú pamäť alebo čas, kedy sa inštrukcie programu môžu vykonávať v procesore

Pri štarte počítača sa používa dohoda o tom, akým spôsobom BIOS odovzdá riadenie operačnému systému. Táto dohoda prakticky znamená, že po vykonaní štartovacích testov pamate a niektorých periférií sa riadenie odovzdá tzv. boot manageru. V Linuxe sa najčastejšie používa `lilo`, `grub` alebo `loadlin`, či boot manažér iného operačného systému – napr. Windows. Úlohou boot manažéra je nájsť na disku súbor jadra, nahráť ho do pamäti a odovzdať mu riadenie.

V podstate každý kus hardwaru v počítači potrebuje svoj ovládač. Klávesnica, video-karta, sieťová karta, sériové, paralelné či USB porty, disk, disketová mechanika a tak ďalej. Okrem toho existujú aj ovládače pre algoritmické záležitosti ako sú súborové systémy, sieťové protokoly či rôzne systémy ako napr. APM (advanced power management). Jadro poskytuje množinu funkcií, ktoré umožňujú programom robiť rôzne operácie ako napríklad otvorenie súboru.

Moderné operačné systémy sú mnohoúlohové (umožňujú beh viacerých programov naraz) a mnohoužívateľské (umožňujú prácu viacerých užívateľov naraz). Jadro sa stará o to, aby rozdelenie výkonu, pamati a sieťovej priepustnosti bolo „spravodlivo“ rozdelené medzi bežiace programy (potenciálne rôznych užívateľov).

Prečo?

Najčastejšie dôvody pre kompilovanie jadra:

1. Snaha niečo sa dozvedieť.
2. Pridanie podpory pre hardware, ktorý pôvodným jadrom podporovaný nie je.
3. Snaha o zmenu konfigurácie ovládačov z hľadiska toho, či sú priamo v jadre, alebo existujú ako moduly.
4. Opravy chýb v jadre.
5. Snaha niečo si dokazovať a dvíhať si sebavedomie :-)

Keď dostanete do ruky inštaláciu linuxového systému, tak spravidla vlastne máte nejakú distribúciu. Niektorí zbrali jadro a množinu programov a dali ich dohromady tak, aby spolu fungovali. Medzi známe distribúcie patrí Red Hat, Debian, Mandrake či Slackware alebo Gentoo. Ten, kto distribúciu zostavuje, sa spravidla postará o to, abyste dostali aj jadro (niekedy máte na výber z viacerých). Toto jadro bolo tvorcom distribúcie nejakým spôsobom nakonfigurované. Spravidla tak, aby podporovalo čo najrozmanitejšie konfigurácie hardwaru a požiadavky užívateľov.

V súčasnosti je veľmi zriedkavo nutné, abyste jadro kompilovali sami. Linuxové jadro je postavené ako skladačka. Skladá sa z mnohých častí nazývaných moduly, ktoré nie sú priamo v súbore jadra,

ale je možné jadro požiadať o to, aby ich funkcionality nahralo zo súboru dodatočne – už v čase, kedy je systém rozbehnutý. Preto je často možné, že ak vaše jadro neobsahuje podporu pre nejaký hardware, možno do neho túto podporu doplniť neskôr – nahraním modulu. Niekedy medzi modulmi existujú závislosti. Napríklad ak máte všeobecnú podporu pre SCSI aj podporu pre SCSI disky ako moduly, tak najprv musíte nahráť všeobecnú podporu pre SCSI a až potom môžete nahráť podporu SCSI diskov. Nahrávanie modulov (príp. ich odstraňovanie) sa spravidla robí pomocou programov z balíka `modutils`. Veľmi stručne:

- `lsmod` – vypíše moduly zavedené v pamäti v danom okamihu

```
# lsmod
Module                Size  Used by
ipt_state              1984  0
ipt_limit              2432  0
iptables_nat          22732  0
iptables_filter       2752  0
ip_tables              17920  4 ipt_state,ipt_limit,iptables_nat,
                        iptable_filter
ip_conntrack           34560  2 ipt_state,iptables_nat
uhci_hcd               30924  0
usb_storage            101328 1
```

- `insmod mojmodul` – sa pokúsi zaviesť do pamäti modul `mojmodul`

```
# insmod ipt_LOG.ko
# lsmod
Module                Size  Used by
ipt_LOG                6336  0
ipt_state              1984  0
ipt_limit              2432  0
iptables_nat          22732  0
iptables_filter       2752  0
ip_tables              17920  5 ipt_LOG,ipt_state,ipt_limit,
                        iptable_nat,iptables_filter
ip_conntrack           34560  2 ipt_state,iptables_nat
uhci_hcd               30924  0
usb_storage            101328 1
```

- `rmmod mojmodul` – sa pokúsi zrušiť modul `mojmodul`

```
# rmmod ipt_LOG
# lsmod
Module                Size  Used by
ipt_state              1984  0
ipt_limit              2432  0
iptables_nat          22732  0
```

```

iptables_filter      2752  0
ip_tables            17920  4 ipt_state, ipt_limit, iptable_nat,
                    iptable_filter
ip_conntrack        34560  2 ipt_state, iptable_nat
uhci_hcd            30924  0
usb_storage         101328  1
    
```

- `depmod` – zistí závislosti medzi modulmi
- `modprobe mojmodul` – nahrá modul `mojmodul` spolu so všetkými modulmi potrebnými pre jeho beh

```

# lsmod
Module              Size  Used by
uhci_hcd            30924  0
usb_storage         101328  1

# modprobe ipt_LOG
# lsmod
Module              Size  Used by
ipt_LOG             6336  0
ip_tables           17920  1 ipt_LOG
uhci_hcd            30924  0
usb_storage         101328  1
    
```

Pridať alebo ubrať modul môže len užívateľ `root`. Druhá varianta pridávania a odoberania modulov je použitie automatického nahrávania modulov, ktoré ešte spomeniem nižšie.

Kompilovanie jadra

Kde zobrať zdrojový kód jadra?

Ak dospejete k rozhodnutiu, že je nutné skompilovať vlastné jadro, musíme najprv získať jeho zdrojový kód. Existuje viacero variant jadier, ktoré môžete chcieť, a preto si najprv treba ujasniť zdrojový kód, ktorého jadra vlastne chceme:

Vanilla jadro

Vývojári jadra uverejňujú oficiálny zdrojový kód jadra na kernel.org [1]. Postupom času vznikajú novšie a novšie verzie jadra. V súčasnosti je hlavné číslo verzie 2. Verzie stabilné – dostatočne odskúšané – majú vedľajšie číslo párne. V súčasnosti sa najskôr stretnete s verziou 2.6, 2.4 alebo 2.2. Ten, kto má chuť riskovať, experimentovať alebo potrebuje niečo čo v aktuálnej verzii nie je, môže skúsiť čísla nepárne. V súčasnosti je poslednou verziou 2.6.6. Experimentálne verzie 2.3 a 2.5 sa už nevyvíjajú. Okrem oficiálneho zdrojového kódu tu nájdete aj odnože udržiavané poprednými vývojármi jadra. Napríklad Alan Cox tu má odnož s príponou `-ac`. Tieto odnože sa spravidla líšia použitím niektorých experimentálnych algoritmov.

Jadro distribúcie

Tvorcovia distribúcií tiež vychádzajú z vanilla jadra, ale často k nemu pridávajú vlastné úpravy alebo niektoré z vlastností nepárnej verzie. To neznamená, že také jadro je menej stabilné, ale len to, že podľa názoru distribútora je daná úprava dostatočne stabilná pre širšie použitie, resp. je to risk hodný zvýšeného výkonu a podobne. Pretože distributéri potrebujú nejaký čas na testovanie, tak svoju upravenú verziu jadra publikujú s istým oneskorením oproti vanilla jadru.

Zdrojový kód jadra má vo forme archívu cez 30MB. Preto v prípade, že máte k dispozícii zdrojový kód staršej verzie, môže vám stačiť zobrať len zmeny oproti tejto staršej verzii – tzv. patch (záplatu). Ak máte úplný zdrojový kód jadra, rozbaľte ho (tradične sa rozbaľuje do `/usr/src`):

```
# cd /usr/src
# tar xvfz kernel-2.6.3.tgz
```

alebo

```
# cd /usr/src
# tar xvfj kernel-2.6.3.tar.bz2
```

Podľa toho, či máte zdrojový kód jadra komprimovaný programom gzip, alebo bzip2. Rozbalením vznikne adresár, ktorý nesie meno verzie `/usr/src/linux-2.6.3/`. Je vhodné ho buď premenovať `# mv linux-2.6.3 linux`, alebo vytvoriť symbolickú linku `# ln -s linux-2.6.3 linux` tak, aby zdrojový kód bol dostupný pod adresárom `/usr/src/linux`.

Ak máte patch, musíte ho aplikovať na svoj zdrojový kód jadra

```
# cd /usr/src/linux
# gunzip -c patch-2.6.4.gz | patch -p1 -
```

alebo

```
# cd /usr/src/linux
# bunzip2 -c patch-2.6.4.bz2 | patch -p1 -
```

Číslo záplaty 2.6.4 znamená, že ho možno aplikovať na zdrojový kód jadra 2.6.3 a jeho aplikovaním získame kód jadra 2.6.4.

Na budúce si povieme o možnostiach konfigurácie jadra pred kompiláciou, o samotnej kompilácii a konečne o instalácii skompilovaného kernelu, aby bolo možno jej využiť pri boote systému.

Konfigurácia jadra

Prv než spustíme samotnú kompiláciu, musíme povedať, čo bude v jadre a čo nie. Táto informácia je uložená v súbore `/usr/src/linux/.config`. Tento súbor buď musíme vytvoriť, alebo upraviť už existujúci. Ak potrebujete len mierne upraviť vlastnosti jadra z distribúcie, skúste zistiť, či distributér tento súbor niekde neponúka. Ak ste na zdrojový kód jadra aplikovali patch na vyššiu verziu, môžete spustiť príkaz (predpokladám, že sa nachádzate v adresári `/usr/src/linux`):

```
# make oldconfig
```

Tento príkaz zistí, aké nastavenia pribudli v novej verzii a opýta sa vás len na tieto nastavenia. Ostatné zostanú nezmenené.

Ak potrebujeme urobiť kompletnú konfiguráciu alebo meniť pôvodné nastavenia, máme na výber niekoľkých možností v závislosti od toho, aké prostredie máme k dispozícii:

1.

```
# make config
make[1]: 'scripts/fixdep' is up to date.
scripts/kconfig/conf arch/i386/Kconfig
# using defaults found in .config
* Linux Kernel Configuration
* Code maturity level options
Prompt for development and/or incomplete \\  

                                code/drivers (EXPERIMENTAL) [Y/n/?]
Select only drivers expected to compile \\  

                                cleanly (CLEAN_COMPILE) [Y/n/?]
...

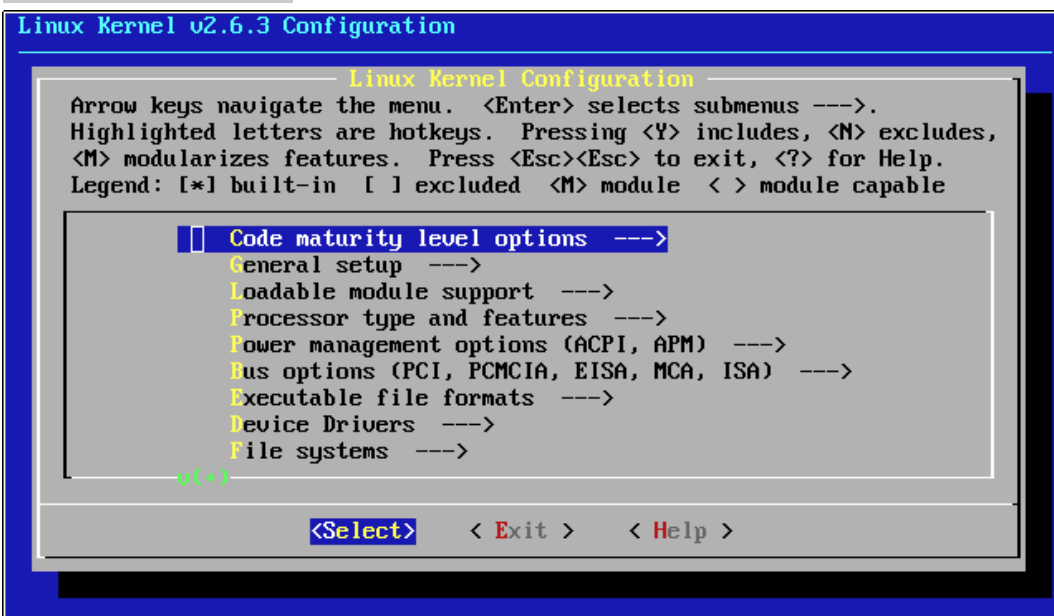
```

Tento spôsob patrí medzi historické. Stačí nám ľubovoľný textový terminál. Konfigurácia spočíva v tom, že dostávate rad otázok. Za každou otázkou sú v hranatých zátvorkách uvedené možnosti:

- o Yes – toto nastavenie bude použité
- o no – toto nastavenie nebude použité
- o module – daná voľba bude zkompilovaná ako modul
- o ? – vypíše nápovedu – stručný popis významu daného nastavenia

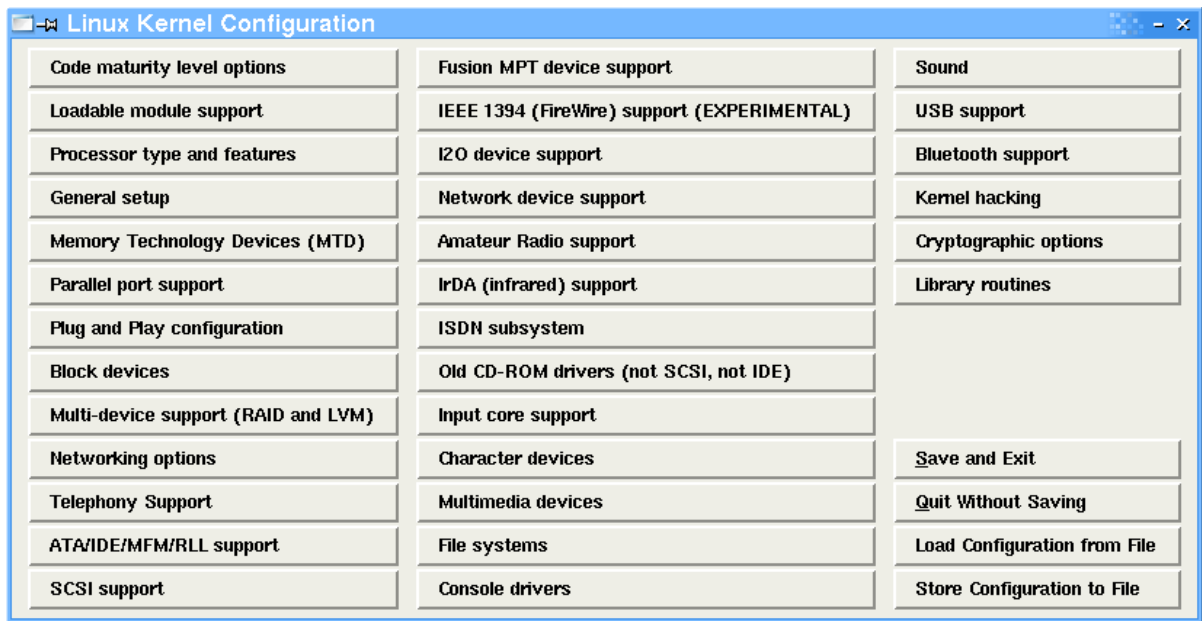
Voľba, ktorá bola nastavená v pôvodnom `.config` je uvedená veľkým písmenom a bude tiež použitá, keď otázku jednoducho odklepnete klávesou Enter.

2. # make menuconfig

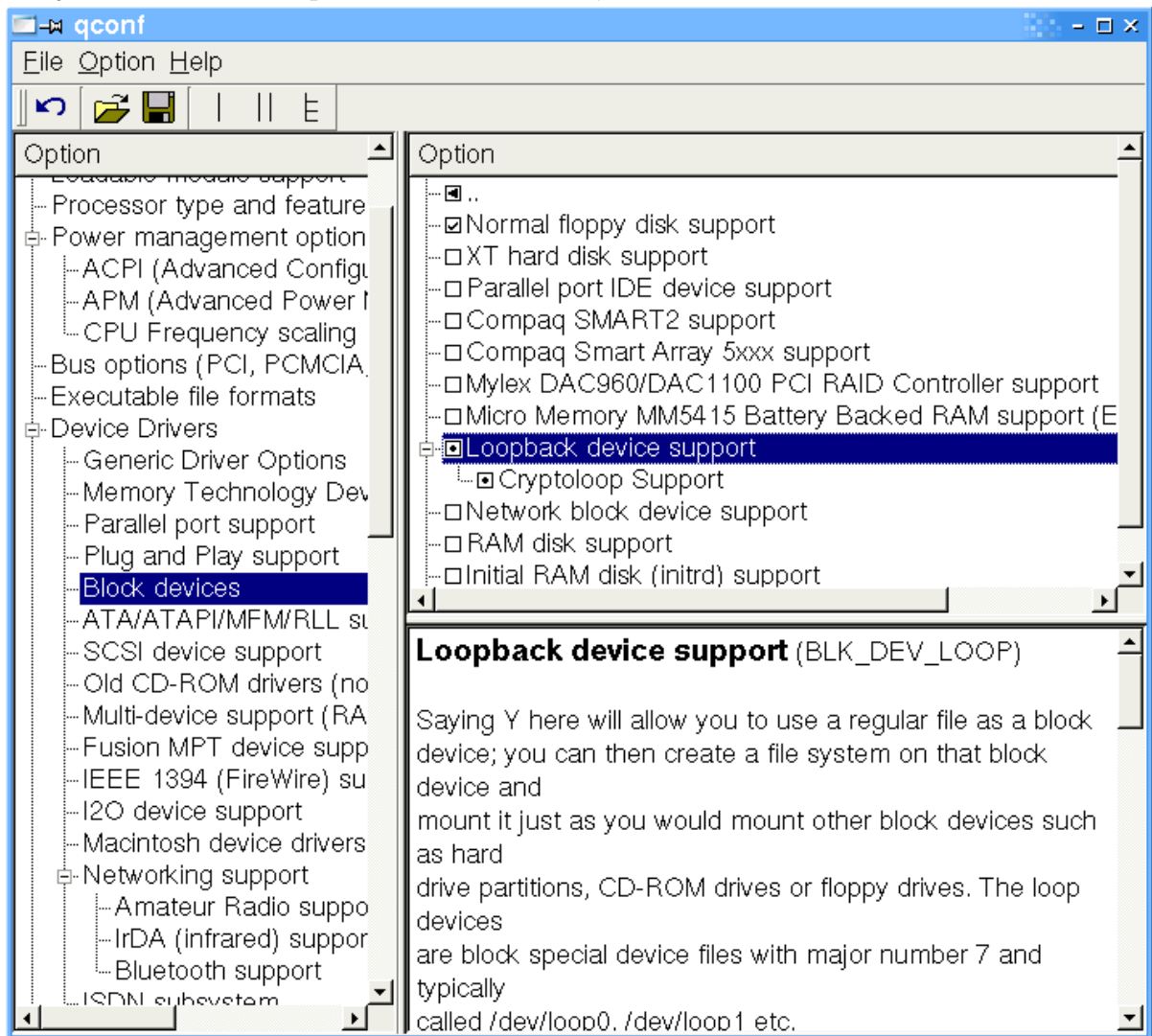


Tento spôsob môžeme použiť, ak má náš systém nainštalovanú knižnicu `ncurses`. Tak ako v predchádzajúcom prípade môžete zvoliť medzi Y/n/m/?

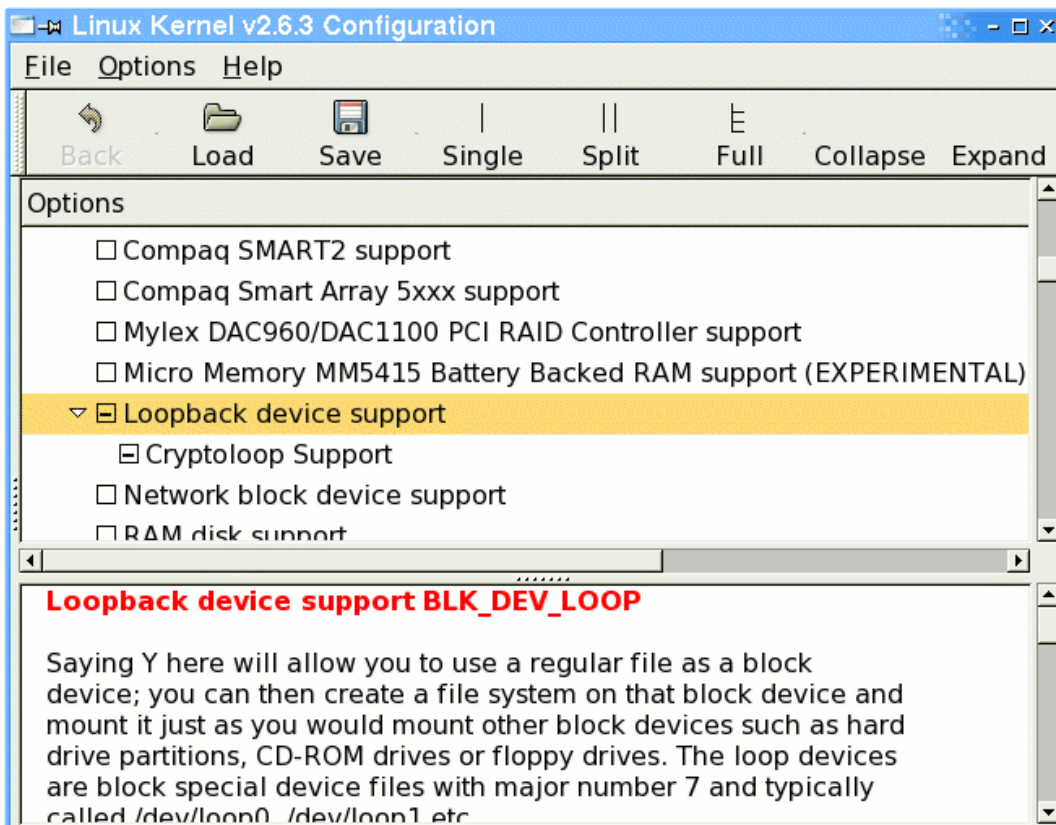
3. # make xconfig. Pri jadrách 2.2 a 2.4 budete potrebovať mať nainštalovaný jazyk Tcl/Tk:



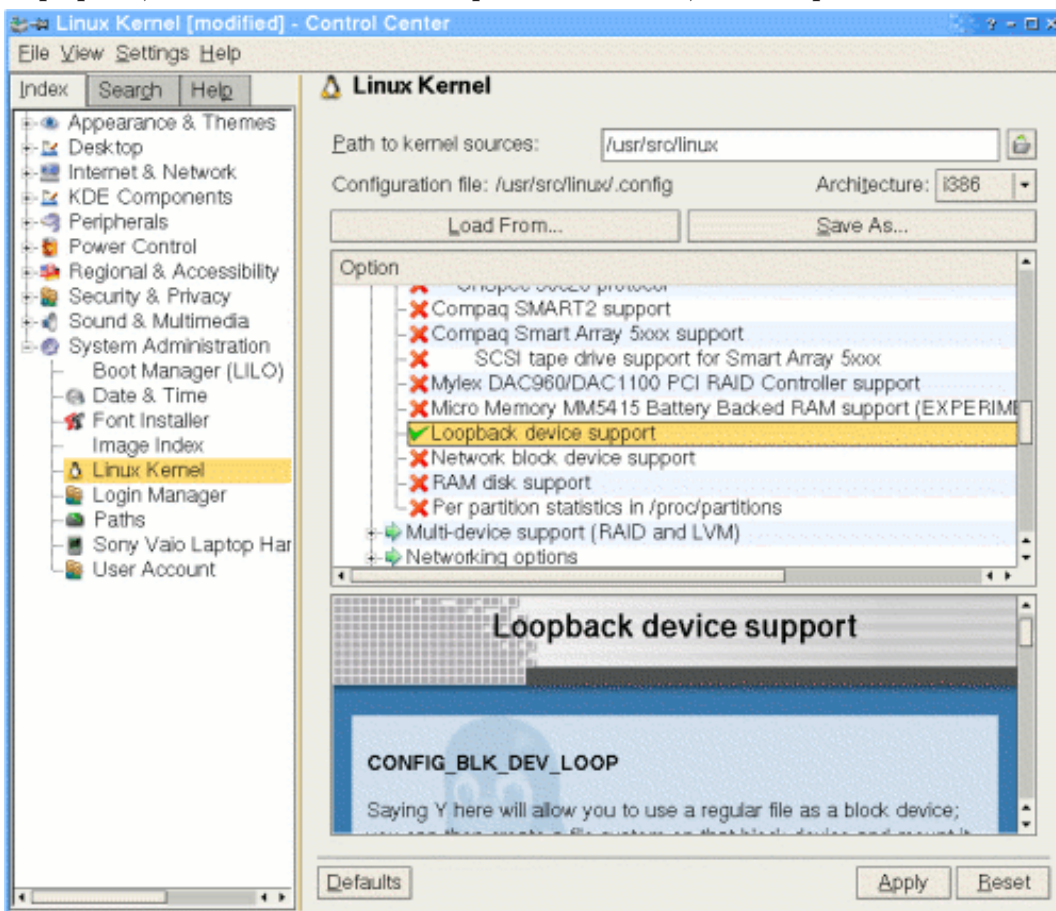
Pri jadrách 2.6 budete potrebovať knižnicu Qt:



4. # make gconfig. Dostupné len pri jadrách 2.6:



5. V prípade, že máte nainštalované prostredie KDE, môžete použiť Control Center:



KDE vo verzii 3.2.2 vie konfigurovať len jadrá do verzie 2.4, ale dá sa predpokladať, že podpora pre 2.6 čoskoro pribudne.

Nech už zvolíte ľubovoľný spôsob, je isté, že nato, abyste správne nastavili konfiguráciu budúceho jadra, je vhodné mať dosť rozsiahle znalosti ako o hardwéri, ktorý máte v počítači, tak aj o rôznych subsystémoch. V tom, či zvládnete tento krok spočíva 99 % úspechu. Ak ho nezvládnete, môžete skončiť s jadrom, ktoré nenabootuje alebo nebude podporovať hardware, ktorý máte, alebo nebude efektívne využívať možnosti tohoto hardwaru. Pri niektorých voľbách je tiež vhodné zvážiť, na aký účel budete potom celý systém používať.

Moduly áno, či nie

Ako som už spomínal, pri niektorých nastaveniach v konfigurácii možno zvoliť kompilovanie vo forme modulov. To znamená, že príslušná funkcionálna nebude začlenená do výsledného súboru jadra, ale do iného súboru. Meno tohoto súboru sa zvyčajne (no nie vždy) možno dočítať v nápovede pre danú voľbu alebo v dokumentácii.

Rozhodnutie či kompilovať ako modul, či nie, je na vás. Treba ale pamätať na to, že niektoré distribúcie a niektoré podsystémy očakávajú niektoré ovládače vo forme modulov. Príveľa *niektoré*, takže konkrétne: je zvykom, že ovládače pre sieťové karty sa kompilujú ako moduly, pretože to uľahčuje konfigurovanie systému pre použitie viacerých sieťových kariet naraz. Tiež to uľahčuje odovzdávanie parametrov do ovládača, akými môže byť IRQ, či DMA. Distribúcie ponúkajú štartovacie skripty, ktoré sa pokúšajú nahráť postupne všetky moduly až kým nenájdu ten správny pre vašu sieťovú kartu. Podobne napríklad podsystém pre podporu zvuku – ALSA – očakáva ovládače pre zvukovú kartu vo forme modulov.

Zodpovednosť za automatické nahrávanie modulov môžete prenechať podsystému jadra nazvanému KMOD – Automatic kernel module loading. Tento podsystém dokáže automaticky nahráť príslušný modul v čase, keď je podtrebný – napríklad ovládač pre súborový systém FAT sa nahrá v čase, keď sa vykoná `mount` na takýto súborový systém.

Spustenie kompilácie

Po nakonfigurovaní nám ostáva len spustiť samotné kompilovanie. Nástroje potrebné na skompilovanie sú popísané v súbore `/usr/src/linux/Documentation/Changes` spolu so spôsobom, ako zistiť verziu nástrojov, ktoré máte nainštalované:

```
o Gnu C          2.95.3    # gcc --version
o Gnu make       3.79.1    # make --version
o binutils       2.12      # ld -v
o module-init-tools 0.9.10   # depmod -V
...
```

a tiež informáciu, kde získať aktuálne verzie týchto nástrojov:

```
o <ftp://ftp.gnu.org/gnu/gcc/>
o <ftp://ftp.gnu.org/gnu/make/>
o <ftp://ftp.kernel.org/pub/linux/devel/binutils/>
...
```

Ak na vašom systéme potrebné nástroje nemáte, môže to byť kvôli tomu, že len nie sú súčasťou typickej inštalácie vašej distribúcie – takže sa najprv pozrite, či sa nedajú doinštalovať.

Ak váš systém má potrebné nástroje, môžeme sa pustiť do kompilácie. Opať je rozdiel medzi verziami 2.6 a nižšími. Ak máte konfiguráciu urobenú správne a chcete skompilované jadro hneď aj nainštalovať, môžete to urobiť takto:

Verzie 2.4 a nižšie: `# make dep clean bzlilo modules modules_install`.

Verzia 2.6: `# make bzlilo modules_install`.

To, čo nasleduje za príkazom `make`, je takzvaný cieľ. Existuje mnoho cieľov, ktoré môžete použiť. Tu je ich stručný popis:

`dep`

Kontrola závislostí medzi rôznymi nastaveniami. Tento cieľ je potrebný pre jadrá do verzie 2.4

`clean`

Vymazanie dočasných súborov v adresároch pod `/usr/src/linux`. Táto voľba je nutná pri jadrách do verzie 2.4.

`bzImage`

Skompilovanie obrazu jadra (Image), ktoré je skomprimované zipovacím algoritmom (`zip`) a je pravdepodobne väčšie ako je rozmer diskety (`big`). Výsledné jadro vznikne v adresári `/usr/src/linux/arch/i386/boot` (za predpokladu, že kompilujete na IBM kompatibilnom PC-čku). Pri tomto ciele sa nekompilujú moduly.

`modules`

Skompilovanie modulov.

`modules_install`

Nainštalovanie modulov do `/lib/modules/{verziajadra}/`.

`bzlilo`

To isté ako `bzImage` ale výsledné jadro je automaticky nainštalované pomocou `lilo`. Predpokladom je, že `lilo` je správne nainštalované a očakáva súbor s jadrom na tom istom mieste, ako ho špecifikuje súbor `/usr/src/linux/Makefile` na riadku `export INSTALL_PATH=/boot`. Inými slovami, ak `lilo` očakáva súbor `vmlinuz` inde ako v `/boot`, musíte tento riadok upraviť.

`mrproper`

Podobne ako `clean`, ale zmazaný bude aj súbor `.config`.

Čo sa ešte oplatí vedieť?

- V adresári so zdrojovým kódom jadra nájdete adresár `Documentation`. V tomto adresári nájdete súbory popisujúce všeobecné princípy fungovania jadra, spôsoby konfigurovania rôznych ovládačov, ale aj informácie o tom, ako začať, ak chcete pridávať do jadra vlastnú funkcionality. Do verzie 2.4 sa tu nachádzal aj súbor `Configure.help` obsahujúci popis jednotlivých volieb zobrazovaný pri konfigurovaní jadra. V jadre 2.6 bol obsah tohoto súboru porozdeľovaný do súborov `Kconfig` v adresároch obsahujúcich kód zodpovedajúcich ovládačov.
- Verziu jadra, ktoré práve beží, získate pomocou programu `uname -r`:

```
# uname -r
2.6.6
```

- Skompilované jadro v sebe nesie informáciu o tom, kde má hľadať hlavný súborový systém. Túto informáciu možno zistiť a aj zmeniť pomocou programu `rdev`:

```
# rdev /boot/vmlinuz
Root device /dev/hda2
# rdev /boot/vmlinuz /dev/hda3
# rdev /boot/vmlinuz
Root device /dev/hda3
```

V prípade, že informácia v jadre nezodpovedá tomu, kde v skutočnosti máte hlavný súborový systém, jadro pri štarte systému spravidla havaruje s hláškou podobnou tejto:

```
Kernel panic: VFS: Unable to mount root fs on XX:YY
```

Kde čísla **XX** a **YY** označujú minor a major číslo zariadenia, na ktorom jadro očakáva hlavný súborový systém.

Odkazy

[1] <http://kernel.org>

Ako funguje e-mail

Rastislav Stanik

E-mail (rovnako ako mnoho iných funkcií počítača) môžete používať aj bez toho abyste vedeli, ako to vnútri funguje. Ale vedieť viac je lepšie, nie?

Úvod

Medzi najpopulárnejšie spôsoby komunikácie, ktoré nám priniesol internet, je e-mail. Tento dokument popisuje princípy toho, ako e-mail funguje.

Posielanie pošty

Keď posielate e-mailovú správu môžete, vynechať telo správy. Môžete vynechať aj vyplnenie kolonky Predmet (Subject:), ale ak chcete, aby správa niekam odišla, musíte vyplniť položku Adresát (To:). Cesta e-mailu začína práve tu. Adresa môže vyzeráť rôzne. Môže pozostávať len z mena (napr. *rastos*), môže špecifikovať meno užívateľa a meno počítača (napr. *rastos@server*), meno počítača môže byť aj úplné (napr. *rastos@server.example.org*). Najbežnejšou formou však je meno a doména (napr. *rastos@example.org*). Túto adresu teda zadáme do programu na písanie e-mailov. Aby tento program vedel, kam správu poslať, má dve možnosti:

1. Odovzdať telo správy spolu so všetkými potrebnými údajmi inému programu. Napríklad mnohé unixové programy na písanie e-mailu samotné odoslanie správy prenechajú interaktívnemu módu programu `sendmail`. Ten potom použije postup popísaný v nasledujúcom bode.
2. Protokolom SMTP odovzdať takzvanému odchodnému serveru (angl. outgoing server). Týmto serverom je zvyčajne SMTP server poskytovateľa pripojenia na internet. Správca väčších lokálnych sietí (napr. vo firmách, alebo školských sieťach) môže zriadiť vlastný SMTP server.

SMTP server

SMTP server je server, ktorý v závislosti na druhu adresy môže urobiť jednu z nasledujúcich vecí:

- Ak je adresát správy špecifikovaný len menom, pokúsi sa správu doručiť lokálnemu užívateľovi s týmto menom.
- Ak adresa obsahuje aj meno počítača, pokúsi sa správu poslať SMTP serveru na špecifikovanom počítači.
- Ak adresa obsahuje doménu, musí SMTP server odovzdať ďalej. Čo to presne znamená, záleží na jeho konfigurácii a môže to byť jedna z nasledujúcich vecí:
 - Jednoducho odovzdá správu inému SMTP serveru.
 - Pokúsi sa správu doručiť sám. Ak sa mu to nepodarí, odovzdá ju inému SMTP serveru v nádeji, že ten bude úspešnejší. Tento druhý SMTP server sa v angličtine nazýva „smart host“.

Veta „Pokúsi sa správu doručiť sám“ je krátka, ale skrýva sa za ňou mnoho. SMTP server sa spojí s DNS serverom, aby si od neho vypýtal takzvaný MX záznam pre danú doménu. MX je z anglického slova `Mail eXchanger`, teda stroj, ktorý je zodpovedný za vymieňanie e-mailu pre danú doménu. DNS server vracia teda IP adresu tohto stroja. DNS server môže mať takýchto záznamov viacero – ukazujúcich na rôzne IP adresy. V takom prípade sa jednotlivé záznamy líšia takzvanou prioritou.

Je to celé kladné číslo, ktoré hovorí o tom, ktorý zo záznamov má vyššiu prednosť. Nižšie číslo znamená vyššiu prioritu. Ako vyzerajú tieto MX záznamy si môžeme pozrieť pomocou programu `nslookup`, na novších systémoch programom `host` či `dig`:

```
$ host -t mx rastos.org
rastos.org mail is handled by 20 mx.smtp.cz.
rastos.org mail is handled by 10 in.smtp.cz.

$ dig MX +short=yes rastos.org
20 mx.smtp.cz.
10 in.smtp.cz.

$ nslookup
> set type=mx
> rastos.org
Server:          192.168.1.102
Address:         192.168.1.102#53

Non-authoritative answer:
rastos.org      mail exchanger = 20 mx.smtp.cz.
rastos.org      mail exchanger = 10 in.smtp.cz.
```

Keď už máme adresu poštového servera, môžeme si vyskúšať, čo vlastne robí náš e-mailový klient, keď mu odovzdáva našu správu cez SMTP:

```
> telnet in.smtp.cz 25
Trying 81.95.97.116...
Connected to in.smtp.cz.
Escape character is '^]'.
220 in.smtp.cz ESMTP
ehlo yahoo.com
250-in.smtp.cz
250-AUTH LOGIN PLAIN
250-AUTH=LOGIN PLAIN
250-PIPELINING
250-STARTTLS
250 8BITMIME
mail from: smith@yahoo.com
250 ok
rcpt to: testaccount@rastos.org
250 ok
data
354 go ahead
Subject: citat
```

```
Four fifths of all our troubles in this life
would disappear if we would just sit down
and keep still. -C. Coolidge
```

```
.
250 ok 1086177079 qp 12124
quit
221 in.smtp.cz
Connection closed by foreign host.
```

Ako vidíte, server označil odosielateľa aj príjemcu správy za „ok“. Odosielateľ nie je „ok“, ak jeho doména má existujúci MX záznam. Príjemca je „ok“, ak daný server je skutočne mail exchanger pre doménu príjemcu. SMTP server môže tiež zo sieťového spojenia zistiť, z akej IP adresy mu správa prichádza, a porovnať to s tým, čo tvrdí odosielateľ.

Server, ktorý nerobí tieto kontroly, sa označuje termínom **open relay**. Takýto stroj je pokladom pre odosielateľov spamu, ktorí často falšujú adresu odosielateľa. Mnohé (no žiaľ nie všetky), takéto servery sú zaznamenané v zozname, ako je napríklad **ORDB [1]** (Open Relay Database). Keďže e-mail môže prechádzať cez viacero SMTP serverov, môžu v rámci protispamových opatrení jednotlivé servery tiež kontrolovať, či SMTP server, od ktorého preberajú správu, nie je zaznamenaný v zozname ORDB.

Prijímanie pošty

Local

V čase, keď si e-mail ešte len razil cestu na výslnie, sa najčastejšie pošta preberala z lokálnej e-mailovej schránky. Je to vlastne súbor, do ktorého e-mailový server ukladá prichádzajúce správy jednu za druhou. Tento súbor sa spravidla nachádza v adresári **/var/spool/mail** (na niektorých systémoch **/usr/spool/mail** alebo **/usr/lib/mail**). Program na čítanie e-mailu, označovaný ako mailer (alebo aj Mail User Agent – MUA), číta tento súbor a umožňuje jednotlivé správy zobraziť a spracovať. To samozrejme môže fungovať vtedy, keď mailer beží na tom istom počítači, na ktorom sa nachádza e-mailová schránka. Čo už dnes nie je tak časté.

POP

Najčastejší spôsob prístupu k e-mailovej schránke je asi sieťovým protokolom POP (Post Office Protocol). Je to pomerne jednoduchý protokol, ktorým si mailer preberá správy z poštového servera. Znamená to teda, že správa od odosielateľa dorazí na server a tam zostáva až do okamihu, kedy si ju odtiaľ nevyzdvihnete. Pri prevzatí správy z POP servera sa zvyčajne správa na serveri zmaže a zostáva teda len na klientovi. Komunikáciu POP protokolom možno jednoducho nasimulovať napríklad takto:

```
> telnet pop3.hosting.cz 110
Trying 81.95.97.117...
Connected to pop3.hosting.cz.
Escape character is '^'.
+OK <29567.1086177339@pop3.smtp.cz>
user testaccount@rastos.org
+OK
```

```
pass m0jehe5sl0
+OK
list
+OK
1 656
.
retr 1
+OK
Return-Path: <smith@yahoo.com>
Delivered-To: testaccount@rastos.org
Received: (qmail 12323 invoked from network);
          2 Jun 2004 11:54:37 -0000
Received: from unknown (HELO in.smtp.cz) (81.95.97.116)
          by master.smtp.cz with DES-CBC3-SHA encrypted SMTP;
          2 Jun 2004 11:54:37 -0000
Received: (qmail 15491 invoked from network);
          2 Jun 2004 11:51:19 -0000
Received: from unknown (HELO yahoo.com) (213.15.194.19)
          by in.smtp.cz with SMTP; 2 Jun 2004 11:51:03 -0000
Subject: citat
Message-ID: <20040602115437.12323.qmail@master.smtp.cz>

Four fifths of all our troubles in this life would disappear
          if we would just sit down and keep still. -C. Coolidge

.
dele 1
+OK
quit
+OK
Connection closed by foreign host.
```

Príkazy posielané klientom rozšifrujete asi ľahko: **user** a **pass** sa postarajú o našu autentifikáciu. **list** vypíše, koľko správ je v schránke a ako sú dlhé. **retr** požiada o výpis správy, **dele** správu zmaže a **quit** ukončí spojenie.

Protokol POP nie je šifrovaný, takže heslo je prenášané v čistom texte. Tento nedostatok môže pomôcť odstrániť zabalenie POP protokolu do šifrovacieho protokolu SSL – to ale musí prirodzene podporovať ako server, tak aj klient.

IMAP

IMAP protokol sa trochu podobá protokolu POP. Tiež je to protokol, ktorý umožňuje e-mailovému klientovi pristupovať k správam na serveri. Hlavný rozdiel je v tom, že správy zostávajú na serveri. Nepresúvajú sa na lokálny disk. Znamená to teda, že e-mailový klient len sprostredkováva pohľad na schránku sídliacu na serveri. Zmenou klienta či počítača, za ktorým sedíte, sa nič nedeje. Stála vidíte

tie isté správy, označenie prečítaných správ sa nemení, zostáva štruktúra priečinkov a tak ďalej. Uľahčuje to tiež systém zálohovania, pretože sa nemusia zálohovať schránky jednotlivých užívateľov roztrúsené po všetkých pracovných staniciach.

Komunikácia protokolom IMAP je trochu zložitejšia. Tu je stručná ukážka:

```
> telnet pop3.hosting.cz 143
Trying 81.95.97.117...
Connected to pop3.hosting.cz.
Escape character is '^]'.
* OK Courier-IMAP ready. Copyright 1998-2002 Double Precision, Inc.
See COPYING for distribution information.
a001 login testaccount@rastos.org m0jehe5l0
a001 OK LOGIN Ok.
a002 select inbox
* FLAGS (\Draft \Answered \Flagged \Deleted \Seen \Recent)
* OK [PERMANENTFLAGS (\Draft \Answered \Flagged \Deleted \Seen)] Limited
* 1 EXISTS
* 1 RECENT
* OK [UIDVALIDITY 1082124563] Ok
a002 OK [READ-WRITE] Ok
a003 fetch 1:1 full
* 1 FETCH (FLAGS (\Recent) INTERNALDATE "02-Jun-2004 14:00:32 +0200"
RFC822.SIZE 669 ENVELOPE (NIL "citac" NIL NIL NIL NIL NIL NIL NIL
"<20040602120031.21486.qmail@master.smtp.cz>") BODY ("text"
"plain" NIL NIL NIL "8bit" 134 2))
a003 OK FETCH completed.
a004 fetch 1:1 body[TEXT]
* 1 FETCH (BODY[TEXT] {134}
Four fifths of all our troubles in this life would disappear
        if we would just sit down and keep still. -C. Coolidge
)
* 1 FETCH (FLAGS (\Seen \Recent))
a004 OK FETCH completed.
a005 store 1:1 +FLAGS (\Deleted)
* 1 FETCH (FLAGS (\Seen \Deleted \Recent))
a006 OK STORE completed.
a006 expunge
* 1 EXPUNGE
* 0 EXISTS
* 0 RECENT
a006 OK EXPUNGE completed
a007 logout
* BYE Courier-IMAP server shutting down
```

```
a007 OK LOGOUT completed
Connection closed by foreign host.
```

Väčšina príkazov začína unikátnym identifikátorom (ja som pre jednoduchosť použil identifikátory začínajúce s `a00`). Príkaz `select` zvolí priečinok, ktorého sa budú týkať ďalšie príkazy. `fetch` môže vrátiť rôzne časti správy. V našom prípade to bola najprv hlavička a potom telo. Protokol IMAP ponúka tiež rozsiahlu sadu príkazov pre nastavovanie príznakov správ („videná“, „zmazaná“, „odpovedaná“, atď.) a príkazov na prácu s priečinkami („vytvoriť“, „zmazať“, „premenovať“ atď.). Podobne ako pri protokole POP, komunikácia klienta so serverom nemusí byť šifrovaná.

Čo nájdeme v mailovej správe

Každá e-mailová správa sa skladá z hlavičky a tela. Oddelené sú prázdny riadkom.

Hlavička

Hlavička (angl. header) obsahuje niekoľko druhov informácií. Ich význam určujú kľúčové slová na začiatku riadku nasledované dvojbodkov:

To:

e-mailová adresa príjemcu

Cc:

e-mailové adresy ľudí, ktorí dostanú kópiu správy (z anglického Carbon Copy)

Bcc:

e-mailové adresy ľudí, ktorí dostnú kópiu správy, ale normálni príjemcovia sa o tom nedozvedia (z anglického Blind Carbon Copy)

From:

e-mailová adresa

Subject:

Predmet, titulok správy.

Date:

Dátum a čas odoslania.

Received:

Jedno Received: pre každý server, cez ktorý správa prechádzala. Dá sa tu dozvedieť meno servera, doba, kedy server správu spracovával, akým protokolom správu dostal, unikátny identifikátor správy a pre koho je správa určená. Každý server zopakuje Received: riadky, ktoré už v správe sú a pridá svoj. Tento údaj teda umožňuje určiť trasu, po ktorej správa putovala – až k jej zdroju. Preto sa napríklad rozosielatelia SPAMu, či vírusy túto časť snažia falšovať. Niektoré anti-spamové riešenia sú založené na tom, že overujú platnosť týchto informácií.

Message-ID:

Unikátny identifikátor správy umožňujúci napríklad radenie správ do tzv. vlákien.

Reply-To:

Komu pôjde odpoveď, keď príjemca odpovie na túto správu.

Content-Type:

Správy obsahujúce prílohy v tejto položke nesú informáciu od druhu prílohy (video, zvuk, binárny súbor, archív, html, ...) a tiež reťazec, podľa ktorého sa dá určiť, kde príloha začína a končí (tzv. boundary).

X*

Existujú mnohé položky začínajúce písmenom X. Norma hovorí, že e-mailové programy môžu takéto položky ignorovať. Používajú sa preto napr. pre informácie špecifické pre istý druh e-mailového programu, pre mail-listy a podobne.

Telo

Telo e-mailovej správy tvorí užívateľ a preto by sa mohlo zdať, že v ňom môže byť čokoľvek. Ale tak ako aj v iných oblastiach nášho života, aj pre telo správy platia isté [pravidlá](#) [2]. Spomeniem aspoň niekoľko z nich:

- Pamätajte, že obsah prenášaných e-mailov nie je chránený pred nepovolanými očami. Ak potrebujete prenášať dôverné informácie, potrebujete asi e-mailový program podporujúci šifrovanie.
- Keď odpovedáte na e-mail, môžete citovať pôvodnú správu. Citujte to, čo príjemcoví pripomenie tému diskusie, ale necitujte zbytočne veľa.
- Než začnete písať odpoveď, overte si, že ste medzi príjemcami správy (kolonka To:) a nie len medzi tými, ktorí dostali len kópiu správy (kolonka Cc: alebo Bcc:).
- Neposielajte správu zbytočne ľuďom, ktorí ju nepotrebujú. Napríklad otázka poslaná skupine nemá byť zodpovedaná e-mailom celej skupine, ale len tomu, kto otázku položil.
- Odpoveď na e-mail môže prísť s oneskorením. Ak nemôžete čakať, použite telefón.
- Predmet (Subject:) správy by mal byť dostatočne výstižný. Ľudia si e-maily archivujú. Keď sa pozriem do archívu spreď dvoch rokov a nájdem správu s predmetom: pomoc – moc mi to nepovie.
- Pokiaľ si nie ste istí, že všetci, komu sa správa môže dostať do rúk, ju dokážu pohodlne prečítať, vyhnite sa všetkému, čo môže robiť problémy. To môže znamenať diakritiku, HTML formatované správy a podobne.
- Neposielajte nevyžiadanú poštu. Neširte neoverené a nepravdivé poplašné správy. Neširte reťazové a pyramídové e-maily.
- Pre prenos väčšieho množstva dát e-mail nie je vhodný. Dokáže to, ale príjemcovi to môže spôsobiť problémy. Niektoré servery obmedzujú veľkosť správy, ktorú sú ochotní spracovať.

Okrem týchto pravidiel, ktoré onedlho oslávajú 10. výročie vzniku, pridám ešte zopár rád vhodných v súčasnosti. V poslednej dobe sa na internete šíri nemálo vírusov práve pomocou mailu. Veľká časť z nich sa spolieha na neopatrnosť užívateľov. Preto buďte opatrní; neotvárajte pripojené súbory od neznámych odosielateľov. Vírusy tiež často využívajú chyby v programoch na čítanie e-mailov. Voľte preto dobre.

Odkazy

[1] <http://ordb.org>

[2] <http://www.faqs.org/rfcs/rfc1855.html>

GIMP – Grafický editor pro Linux a Windows

Vlastimil Ott

Recenze aktuální knížky GIMP – Grafický editor pro Linux a Windows.

Začátkem května vyšla [1] v nakladatelství Computer Press knížka GIMP – Uživatelská příručka, jejímž autorem je Josef Vybíral [2].

Redakci byla nabídnuta k recenzi, které jsem se rád ujal, protože jsem pouhým poučeným uživatelem GIMPu a knížka tohoto typu mi přišla velice vhod. Taky kvituji s povděkem, že se v edičním plánu různých nakladatelství začínají objevovat knihy o Linuxu, které se zvolené tematice věnují na úrovni začátečníků nebo věčně začínajících začátečníků. Netvrdím, že je to případ této knihy, neočekávejte ale žádný zevrubný návod či podrobný referenční manuál.

Díky nasazení a rozšířenosti Linuxu se dle mého názoru dostáváme do situace, kdy se konečně objevují knihy o – s prominutím – každé hlouposti. Myslím, že je to dobře a opět zdůrazňuji, že nemám na mysli tuhle knihu. Pouze mne to napadlo, když jsem prohlížel katalog nakladatelství a hledal další užitečné příručky.

Josef Vybíral aka Cornelius [3] je mladý a perspektivní autor působící na GIMP.cz [4]. Recenzovaným titulem přispěl dle mého názoru zejména k osvětě, protože titul bude v regálu ležet vedle knihy o Photoshopu, Wordu, Excelu a podobných aplikacích. Druhá hodnota už je opravdu ta, kterou všichni očekávají – obsažené informace. Nastíháme si obsah:

1. Úvod. Co je GIMP, co je vlastně obsahem knížky, typografické konvence.
2. Stažení a instalace. Nutno podotknout, že se v knize akcentuje zejména GNU-charakter aplikace, nikoliv to, že se jedná „o ten Linux“. Získání balíčků resp. instalačního archivu (přijde na to) je popsáno pro případ MS Windows a GNU/Linux. Součástí popisu pro Windows je i získání knihoven GTK+, které jsou standardní součástí instalace Linuxu. Proces je přiblížen prostřednictvím mnoha obrázků z obou systémů. Kapitola končí základním nastavením aplikace (rozlišení obrazovky, odkládací prostor apod.). Všechny kroky jsou vysvětleny spolu s odůvodněním, které může být pro uživatele Windows možná někdy nepochopitelné, neboť jejich pracovní prostředí má jiné zvyklosti.
3. Uživatelské rozhraní. Krátká třetí kapitola se zabývá hlavním problémem starší verze GIMPu – rozhraním, které je pro začátečníky relativně nepřehledné a složité. Je zde také nastíněno nastavení písme GTK+ v MS Windows, což musí být pro uživatele šok, protože se jedná o krátký popis souboru `gtkrc.cs` aka `.gtkrc`. Autor se také dostal do nepříjemné schizofrenie: popisuje zvláštnosti GIMPu starších verzí, i když GIMP 2.0, který zásadně mění ovládání, vyšel už dlouho před vydáním knihy. Pravděpodobně to nešlo zcela technicky zvládnout. Přesto je to hlavní výtky, neboť některé obrázky jsou z verze 2.0. Zásadní ale je, že v knize je popisováno rozhraní dnes již starých verzí.
4. Základní operace. Kapitola se zabývá zcela elementárními operacemi, jako je otevření a uložení souboru, skenování (jak ve Windows pomocí TWAIN ovladačů tak v Linuxu pomocí SANE), tisk (tisk ve Windows versus GIMP-print), lupa a měření vzdáleností a úhlů.
5. Práce s výběry. Sofistikovaný popis práce se všemi typy výběrů, překrývání, práce se schránkou. Výhody magické hůlky či Beziérových křivek a jejich použití.
6. Kreslíme. Vysvětlení práce s nástroji GIMPu – barvy, tužka, štětec, pero, guma a další. Vyplňování barvou nebo vzorkem, klonování a retušování. Součástí je i popis práce s informačními okny. Bohužel chybí – z výše uvedené příčiny – dokování panelů, které se objevují ve verzi 2.0. Na druhou stranu je zde zevrubné cvičení společně s řešením.
7. Transformace obrázku. Základní práce jako je posun, transformace, ořez či změna velikosti.

8. Vrstvy. Jedná se o vysvětlení termínu vrstva a principu práce s nimi. Dle mého soudu jedna z nejlepších kapitol. Je zde také popsána interakce mezi vrstvami (zesvětlení a podobné efekty).
9. Barvy a práce s digitální fotografií. Nástin barevných modelů (RGB, CMYK), kanálů, úprava charakteristik bitmapy (jas, kontrast, histogram, odstín, sytost, doladění). Na můj vkus autor příliš zabíhá do problematiky zpracování digitální fotografie a odbíhá od samotného programu, na druhou stranu se to tématu stále týká.
10. Filtry. Základní práce s filtry, lehký úvod do Script-Fu.
11. GIMP a web. Vytvoření klikacích map, nejvhodnější formáty pro web.
12. Animace. Princip animace, ruční animace, plugin pro složitější animace.
13. Formáty. Některé známé i méně známé formáty, jejich stručná charakteristika.
14. Závěr, odkazy, literatura, rejstřík.

Kvalita rejstříku je standardní a slouží svému účelu; poslední kapitoly pomímám. Musím ale zmínit další faktor, který mě bohužel zaujal. Jedná se o jazykovou stránku. Jsem z těch, kteří kladou na bezchybnost maximální důraz. Tohle bohužel pro toto dílko neplatí – absence čárek ve větách (interpunkce) přesahuje obvyklou míru tolerance. Korektor se prostě moc nenadřel, což není vina autora, nicméně to snižuje kvalitu knížky.

Přílohou je CD-ROM, který obsahuje všechny zmiňovaný software (GIMP pro oba systémy ve verzích 1.2.5 a 2.0), cvičení zadaná v knize spolu s řešeními, dokumentaci a zejména velice praktický přehled klávesových zkratk s popisem (vše na jednom listu).

Výsledek? Jen dál, jen houšť. Více takových knih. Každopádně očekávám aktualizované a rozšířené vydání.

Název:	GIMP – Grafický editor pro Linux a Windows
Autor:	Josef Vybíral
Vydavatelství:	Computer Press a.s.
ISBN:	80-251-0158-4
Počet stran:	154 černobílých + CD-ROM
Cena:	187 Kč / 287 Sk
Rok vydání:	2004

Odkazy

- [1] <http://knihy.cpress.cz/Book.asp?ID=985>
- [2] <http://www.abclinuxu.cz/Profile/2848>
- [3] <http://cornelius.wz.cz/>
- [4] <http://www.gimp.cz>

Na co se často ptáme 4: X Window System

Vlastimil Ott

Zcela nezajímavý článek s obyčejným tématem. Asi vás vůbec nezaujme – ovšem pokud vám „neklekl“ X Window System. Potom zde hledejte vyčerpávající návod jak jej nakonfigurovat.

XFree 4.3.0

X Window System (dále v článku také XFree nebo X) všichni známe, nebude se věnovat jeho historii, technologii nebo budoucnosti. Budu se věnovat té nejobyčejnější věci, která pro mnohé uživatele představuje noční můru. Jedná se o nastavení tohoto komplexního systému. Protože má mnoho vlastností (features), jedná se o práci složitou. Lze použít pomocné konfigurační nástroje, které bývají dodávány jednak se samotným XFree, jednak s konkrétní distribucí. Takové řešení bývá obvykle funkční a pro normální provoz dostačující. Někdy ale nastane situace, kdy tyto programy nepomohou. Jedná se o případy, kdy máte příliš starý nebo nový hardware (grafická karta, monitor, displej), nebo ve vaší distribuci daný konfigurační nástroj není nebo nefunguje tak, jak má. Nebo, do třetice, program vám grafický systém nastaví, ale jaksi „málo“. Často je použit standardní ovladač nebo parametr, který výkon vašeho hardwaru degraduje (typicky nízká obnovovací frekvence monitoru).

V textu se pokusím vyčerpávajícím způsobem popsat jednotlivé sekce a parametry konfiguračního souboru `/etc/X11/XF86Config`, kterým je nastaveno chování XFree verze 4.3.0 [1]. Rád bych upozornil, že se nejedná o poslední finální verzi, protože se ale u aktuální verze 4.4 změnila licence, mnozí producenti distribucí zůstali u verze 4.3, případně přecházejí ke konkurenčním grafickým systémům (x.org [2], Y Window System [3]).

Rád bych také upozornil, že syntaxe a dělení jednotlivých částí konfiguračního souboru se od starších verzí změnila a tento materiál se na ně nevztahuje. Veškeré informace, a to i ke starším verzím, lze nalézt na stránkách XFree [4].

Před započítím práce si ověřte, že soubor, který zpracováváte, je při spuštění XFree opravdu interpretován. Pracujete-li jako root, což je pro ladění běhu XFree nutné a máte-li v domovském adresáři soubor `XF86Config`, což se může stát, je interpretován právě on, nikoliv soubor ze standardní cesty `/etc/X11/`.

Také si okamžitě vytvořte nejméně jednu zálohu. To je ostatně vhodné opakovat pokaždé, když se vám bude zdát, že máte část souboru odladěnou. Soubor s veškerými výpisy (log) se nachází v adresáři `/var/log/` a nejprve se jmenuje `XFree.0.log`, starší verze mají ještě navíc koncovku `.old`, případně se vnitřní nula mění na čtyřku.

Konfigurace

Struktura konfiguračního souboru

Konfigurační soubor se skládá z několika sekcí, na jejichž pořadí obvykle (viz níže) nezáleží. Každá ze sekcí má tvar:

```
Section "Jmeno_sekce"  
    Polozka_sekce  
    Polozka_sekce
```

```
Polozka_sekce
EndSection
```

Místo pseudonázvu `Jmeno_sekce` se doplňuje některý z názvů konkrétních sekcí:

Název sekce	Význam a použití
<code>ServerLayout</code>	Integrovaná konfigurace serveru
<code>Files</code>	Cesty k souborům písem, palety apod.
<code>ServerFlags</code>	Příznaky X-serveru
<code>Module</code>	Načítání modulů XFree
<code>InputDevice</code>	Definice vstupního zařízení
<code>Device</code>	Definice grafického výstupního zařízení
<code>VideoAdaptor</code>	Definice Xv grafického adaptéru
<code>Monitor</code>	Definice monitoru
<code>Modes</code>	Definice grafických režimů
<code>Screen</code>	Konfigurace obrazovky
<code>DRI</code>	Konfigurace DRI (Direct Rendering)
<code>Vendor</code>	Konfigurace závislá na výrobci

Syntaxe konfiguračního souboru

Při interpretaci klíčových slov nehraje roli velikost písmen. Znak „-“ (podtržítka) je ignorován; u některých parametrů jsou ignorovány také znaky označující volný prostor (znaky pro mezeru a tabulátor). Každá položka v souboru zabírá obvykle jeden řádek, který se skládá z klíčového slova obvykle následovaného jedním nebo více argumenty. Tyto argumenty mohou nabývat hodnot celých čísel, reálných čísel nebo textových řetězců:

Název	Význam, hodnoty, rozmezí
<code>integer</code>	Celé číslo v desítkové, šestnáctkové (musí začínat „0x“) nebo osmičkové (musí začínat „0“) soustavě.
<code>real</code>	Reálné číslo s plovoucí čárkou.
<code>string</code>	Textový řetězec uzavřený do uvozovek.

Existuje speciální klíčové slovo `Option` (doslova „volba, možnost“), kterým lze serveru předat nestandardní parametr s volnou strukturou. `Option` má jeden, nebo dva argumenty. První je název volby a možný druhý je parametr volby. Ten může kromě výše uvedených typů (integer, real, string) nabývat také hodnot

Název	Význam, hodnoty, rozmezí
<code>boolean</code>	Stavový příznak, který nabývá pouze dvou hodnot: platí, nebo neplatí. Pokud není uvedena hodnota, předpokládá se hodnota „platí“. Tu lze také vyjádřit následujícími znaky, které se považují za rovnocenné: 1, on, true, yes. Hodnota „neplatí“ se vyjadřuje analogicky: 0, off, false, no.
<code>frequency</code>	Hodnota frekvence se vyjadřuje reálným číslem, které může být doplněno jednotkami Hz, k, kHz, M, MHz. Pokud označení jednotky chybí, je odhadnuto z rozmezí, které by mělo být schopno použít dané zařízení. Pochopitelně je vhodné jednotku vždy uvést. Nehrozí ani tak zničení zařízení (monitoru) jako spíše zavlečení chyby.

Všechny parametry volby `Option`, nejenom řetězce, musejí být uzavřeny v uvozovkách. Pokud parametr začíná předponou „No“, je jeho hodnota „neplatí“. Následující příklady vyjadřují naprosto totéž (zakázání akcelerace):

```
Option "Accel" "Off"  
Option "NoAccel"  
Option "NoAccel" "On"  
Option "Accel" "false"  
Option "Accel" "no"
```

Popis jednotlivých sekcí

ServerLayout

Tato sekce má v hierarchii nejvyšší váhu. Svazuje totiž konkrétní vstupní a výstupní zařízení. Vstupní zařízení jsou popsána v sekcích `InputDevice`. Parametr začínající `Core-` určuje standardní vstupní zařízení. V systému musí být přesně jedno pro jednotlivé kategorie (myš, klávesnice); není-li explicitně označeno, je použito první, které lze technicky využít. Toto lze určit také v příslušné sekci `InputDevice` nebo na příkazovém řádku.

Výstupní zařízení se obvykle skládají z vícero nezávislých komponent (grafické karty a monitoru). Tato zařízení jsou svázána v sekcích `Screen`; na tyto sekce se pak odvolává v sekci `ServerLayout`. Grafické karty jsou definovány v sekcích `Device` a monitory v sekcích `Monitor`. Záměrně vše uvádím v množném čísle, protože XFree podporuje více zařízení v jednom sezení (např. dvě karty a dva monitory, nebo jedna karta se dvěma výstupy na dva monitory apod.). Téma na samostatný článek.

```
Section "ServerLayout"  
    Identifier "XFree86 Configured"  
    Screen 0 "Screen0" 0 0  
    InputDevice "Mouse0" "CorePointer"  
    InputDevice "Keyboard0" "CoreKeyboard"  
EndSection
```

Files

```
Section "Files"  
    RgbPath "/usr/X11R6/lib/X11/rgb"  
    ModulePath "/usr/X11R6/lib/modules"  
    FontPath "/usr/X11R6/lib/X11/fonts/TTF/"  
    FontPath "/usr/X11R6/lib/X11/fonts/misc/"  
    FontPath "/usr/X11R6/lib/X11/fonts/Speedo/"  
    FontPath "/usr/X11R6/lib/X11/fonts/Type1/"  
    FontPath "/usr/X11R6/lib/X11/fonts/75dpi/"  
    FontPath "/usr/X11R6/lib/X11/fonts/100dpi/"  
    #FontPath "unix/:7100"  
    FontPath "tcp/192.168.1.10:7100"  
EndSection
```

Tato sekce je určena k předání informací o souborech, které jsou nutné pro běh serveru. Některé z těchto informací lze předávat serveru jako parametr při spuštění, nebo za běhu, např. programem `xset`. Takové parametry mají větší váhu než ty, které jsou uvedené v konfiguračním souboru. Položky této sekce jsou následující.

Parametr	Hodnota	Význam
FontPath	<i>adresarova_cesta</i> nebo <i>protokol/klient:</i> <i>cislo_portu</i>	Adresářová cesta k souborům s fonty. Je možné (a obvyklé) uvádět více těchto parametrů. Obvyklá cesta k souborům s fonty je <code>/usr/X11R6/lib/X11/fonts/</code> . Pokud daný adresář neexistuje nebo neobsahuje žádné soubory písem nebo není vytvořen jejich seznam (soubory <code>fonts.dir</code> , <code>fonts.alias</code> a <code>fonts.scale</code>), XFree oznámí, že odstraňuje tuto položku ze seznamu. Obvyklý případ hlavně při (nesprávném) použití TrueType fontů. Druhá hodnota je případ spolupráce s font-serverem běžícím v síti. Problematika přesahuje tento článek, viz např. článek o fontech v Mozille [5] .
FontPath	<i>adresarova_cesta</i>	Specifikace souboru s databází barev. K názvu je přidána přípona <code>.txt</code> . Soubor obsahuje RGB kódy a názvy barev. Obvykle <code>/usr/X11R6/lib/X11/rgb</code> .
ModulePath	<i>adresarova_cesta</i>	Specifikace adresáře, ve kterém bude systém XFree hledat moduly. Ty se obvykle používají pro akceleraci, podporu TrueType a Type1 písem, zařízení Video4Linux apod. Může být použito opakovaně. Obvykle <code>/usr/X11R6/lib/modules/</code> .

ServerFlags

```
# Nastavení "dvouhlavé" karty pro dva monitory
Section "ServerFlags"
    DefaultServerLayout "Multihead"
    Option "BlankTime" "0"
    Option "Xinerama" "true"
EndSection
```

Tato sekce je určena k specifikaci globálních vlastností X serveru. Všechny položky v této sekci jsou typu `Option` (viz výše). Jsou interpretovány také některé položky ze starších verzí XFree; těmi se zde ovšem nezabýváme.

Parametr	Typ	Význam hodnoty
DefaultServerLayout	string	Sekce <code>ServerLayout</code> , která bude použita jako standardní, pokud není její jméno definováno na příkazové řádce parametrem <code>-layout</code> . Je-li nalezeno více sekcí <code>ServerLayout</code> , platí <i>poslední uvedená!</i>
NoTrapSignals	boolean	Určeno pro ladění. Rozhodnutí, má-li X server překonat chyby, nebo se naopak ukončit a uložit ladicí informace.
DontVTSwitch	boolean	Je-li <code>true</code> , není možné přepínat se na další terminály pomocí klávesových zkratk <code>Ctrl+Alt+Fn</code> .
DontZap	boolean	Je-li <code>true</code> , není možné ukončit (nebo pouze restartovat) XFree pomocí <code>Ctrl+Alt+Backspace</code> .
DontZoom	boolean	Je-li <code>true</code> , není možné přepínat videorežimy pomocí <code>Ctrl+Alt+šedé Plus</code> a <code>Ctrl+Alt+šedé Minus</code> .
AllowMouseOpenFail	boolean	Je-li <code>true</code> , server se spustí, i když není připojena nebo správně nastavena myš.
XkbDisable	boolean	Povoluje, nebo zakazuje rozšíření XKEYBOARD, které slouží k přepínání klávesových map. Standardně povoleno.
BlankTime	integer	Čas (v minutách) definující dobu, než se spustí spořič obrazovky. Lze změnit za běhu pomocí <code>xset</code> . Různá prostředí (KDE) tuto hodnotu modifikují.
StandbyTime	integer	Čas (v minutách) definující dobu, než se monitor přepne do „stand-by“ režimu. Lze změnit za běhu pomocí <code>xset</code> . Nemusí fungovat u všech videodriverů a monitorů. Nastaveno pouze v případě, že je v sekci <code>Monitor</code> definován parametr <code>DPMS</code> (viz níže).
SuspendTime	integer	Čas (v minutách) definující dobu, než se monitor přepne do uspávacího režimu. Lze změnit za běhu pomocí <code>xset</code> . Nemusí fungovat u všech videodriverů a monitorů. Nastaveno pouze v případě, že je v sekci <code>Monitor</code> definován parametr <code>DPMS</code> (viz níže).
OffTime	integer	Čas (v minutách) definující dobu, než se monitor vypne. Lze změnit za běhu pomocí <code>xset</code> . Nemusí fungovat u všech videodriverů a monitorů. Nastaveno pouze v případě, že je v sekci <code>Monitor</code> definován parametr <code>DPMS</code> (viz níže). Všechny tyto čtyři sekce jsou závislé na hardwaru a jejich chování lze vyzkoušet snad jedině prakticky.
Xinerama	boolean	Zakáže, nebo povolí rozšíření XINERAMA, které umožňuje práci s více monitory a kartami, případně vícehlavými kartami (např. ATI).

Existují další parametry, které mají speciální význam a které se obvykle, tj. ve standardní uživatelské konfiguraci, nepoužívají.

Module

V této sekci se specifikují moduly, které se mají při sezení načíst. Jedná se obvykle o rozšíření X serveru nebo moduly pro práci s různými typy písem (TrueType, Type1, Speedo apod.). Většina ostatních modulů je načítána na žádost, takže se o ně nemusíme starat.

Existují dva způsoby jak specifikovat moduly. Nejčastější a nejpohodlnější je způsob

```
Load "nazev-modulu"
```

Argument `nazev-modulu` je název modulu, nikoliv souboru. Řetězec rozlišuje velikost znaků a nesmí obsahovat předponu `lib` stejně jako přípony `.a`, `.o` nebo `.so`.

Druhý, méně častý, způsob je položka `SubSection` s argumentem názvu modulu a parametry typu `Option`, které jsou předány modulu při jeho načtení. Tento způsob je určen pro opravdu speciální případy, kdy je nutné zcela přesně nastavit parametry hardwaru. Je tedy nutné znát parametry jednotlivých modulů, což jejich autor (v takovém případě spíše výrobce) určitě sděluje v dokumentaci k modulu.

Soubory s moduly jsou hledány v adresářích, které jsme určili v sekci `Files`. Každý z nich může obsahovat podadresáře (např. `drivers`, `input`, `extensions`, `fonts`), které jsou také prohledávány. Standardní adresář je `/usr/X11R6/lib/modules/`. Jako absolutní minimum se doporučuje nahrát alespoň modul `extmod`, `bitmap` pro správu systémových písem se nahrává automaticky.

```
Section "Module"
  Load "record"
  Load "xaa"
  Load "extmod"
  Load "drm"
  Load "dbe"
  Load "dri"
  Load "v4l"
  Load "GLcore"
  Load "glx"
  Load "xtrap"
  Load "type1"
  Load "freetype"
  Load "xtt"
  Load "speedo"
EndSection
```

InputDevice

Konfigurační soubor může obsahovat (a obvykle také obsahuje) více těchto sekcí. Minimálně dvě jsou vždy přítomny: jedna pro myš [6] a druhá pro klávesnici [7].

```
Section "InputDevice"
  Identifier "Keyboard0"
  Driver "Keyboard"
  Option "CoreKeyboard"
  Option "MapName" "Standard Keyboard [ pc105 + toggle ]"
  Option "Protocol" "Standard"
  Option "XkbLayout" "cz,us"
  Option "XkbModel" "pc105"
  Option "XkbOptions" "grp:shift_toggle"
  Option "XkbRules" "xfree86"
```

```
EndSection

Section "InputDevice"
    Identifier "Mouse0"
    Driver "mouse"
    Option "CorePointer"
    Option "Protocol" "auto"
    Option "Device" "/dev/mouse"
    Option "Buttons" "3"
    #Option "Resolution" "N" #ukázka
    Option "ZAxisMapping" "4 5"
EndSection
```

Celá sekce je platná, je-li na ni odkazováno v sekci `ServerLayout` nebo parametrem z příkazové řádky.

Parametr	Význam hodnoty
<code>Identifier</code>	Představuje jedinečné jméno vstupního zařízení.
<code>Driver</code>	Typ zařízení, nejčastěji <code>keyboard</code> a <code>mouse</code> .
<code>CoreKeyboard</code>	Definuje hlavní klávesnici, viz také výše.
<code>CorePointer</code>	Definuje hlavní polohovací zařízení, viz také výše.
<code>MapName</code>	Popis typu klávesnice.
<code>Protocol</code>	V případě klávesnice obvykle <code>standard</code> . V případě myši dnes již nejlépe <code>auto</code> , ale různé typy jsou stále rozlišovány [8].
<code>Device</code>	Platí pro myš, obvykle <code>/dev/mouse</code> , což bývá symbolický odkaz např. na <code>/dev/psaux</code> .
<code>Buttons</code>	X server neumí zjistit počet tlačítek myši (což mě udivuje, ale tak se to píše v dokumentaci [9]). Přednastavená hodnota je <code>3</code> tlačítka.
<code>Resolution</code>	Rozlišení pro pohyb kurzoru myši.
<code>ZAxisMapping</code>	Namapování ostatních tlačítek myši. V uvedeném případě je pomyslné tlačítko 4 (ve skutečnosti pohyb kolečkem od sebe) a 5 (pohyb kolečkem k sobě) namapováno na osu Z, tedy posunování obsahu okna (scrolling).
<code>XkbRules</code>	Soubory pravidel pro mapování klávesnice, soubory se nacházejí v adresáři <code>/usr/X11R6/lib/X11/xkb/rules/</code> .
<code>XkbModel</code>	Název modelu klávesnice, její typ.
<code>XkbLayout</code>	Rozložení klávesnice.
<code>XkbVariant</code>	Varianty rozložení.
<code>XkbOptions</code>	Další vlastnosti a konfigurace ovládání klávesnice.

Další parametry jsou pro speciální účely, případně nedokumentované. Konfigurace klávesnice prošla během vývoje XFree mnoha změnami a zejména komplexní prostředí jako KDE tento vývoj výrazně ovlivnila. Už fakt, že návody k nastavení klávesnice jsou nejednotné, stejně jako technologie, svědčí o tom, že řešení není ideální. Více informací lze nalézt v již zmiňovaném dokumentu [10].

Device

Konfigurační soubor může obsahovat těchto sekcí několik. Přítomná musí být alespoň jedna. V této sekci konfiguruje grafickou kartu. V případě vícehlavých karet je nutné vytvořit dva stejné záznamy; pokud se však každá hlava identifikuje jiným **BusID** číslem, pracujeme jakoby se dvěma oddělenými kartami.

Parametr	Význam hodnoty
Identifier	Představuje jedinečné jméno výstupního zařízení.
Driver	Ovladač grafické karty.
BusID	Specifikuje umístění grafické karty ve smyslu PCI:sběrnice:zařízení:funkce . Např. PCI:1:0:0 je ale AGP karta. Tuto hodnotu lze zjistit spuštěním X serveru s parametrem -scanpci , nebo příkazem lspci . 01:00.0 VGA compatible controller: ATI Technologies Inc \\ Radeon RV100 QY [Radeon 7000/VE]
Screen	Obvykle nemá význam v případě jedné karty. V případě více karet nebo vícehlavých karet určuje tento parametr, kterou obrazovku hlava/karta obsluhuje. Pojem „obrazovka“ chápeme ve smyslu „stínítka“, tedy plocha k zobrazování. Je úzce svázána se zařízením monitoru (sekce Monitor). Čísluje se od 0 do počet obrazovek-1.
Chipset	Doporučuje se nenastavovat, pokud to není nezbytně nutné. Moduly jsou schopné samy přítomný čipset rozpoznat. Osvědčí se ale u starých karet, které mají podobné, ale přesto se mírně lišící, čipsety (např. S3).
RAMDAC	Význam tohoto čipu objasní spíše elektrotechnik. Moduly opět samy poznají typ RAMDAC a chovají se podle toho.
Clocks	Frekvence časovače v MHz. Dnes pro moderní karty nevyužitelné; snad jen pro ty staré.
VideoRAM	Množství operační paměti karty v kilobajtech. Dnes již obvykle nevyužívané pro karty v pravém slova smyslu (zasunují se do sběrnice), naopak praktické pro integrované grafiky, např. SiS.

Existuje ještě několik parametrů, s jejichž pomocí je snad možné dosáhnout určitých výsledků. Neslyšel jsem ale o tom, že by se dal takto změnit (zvýšit) výkon karty.

Kromě toho má každý čipset své specifické volby, které jsou velmi důležité, protože výkon zásadně ovlivňují. Následující čipsety mají své manuálové stránky, kde lze konkrétní volby nalézt: chips, cirrus, cyrix, fbdev, glide, glint, i128, i740, i810, imstt, mga, neomagic, nv, radeon, r128, rendition, savage, s3virge, siliconmotion, sis, sunbw2, suncg14, suncg3, suncg6, sunffb, sunleo, suntcx, tdfx, tga, trident, tseng, v4l, vesa, vga.

Problémy s firemními ovladači mnozí z vás znají, proto znovu upozorňuji, že tyto manuálové stránky pojednávají o modulech XFree, nikoliv jednotlivých výrobců, kteří dodávají vlastní dokumentaci. Jako příklad uvádím konfiguraci obstarožního Radeonu 7000 (podrobnosti jsou jako obvykle v manuálové stránce).

```
Section "Device"
    Identifier "radeon0"
    Driver "ati"
    #Driver "radeon" # vyjde nastejno
    VendorName "ATI Radeon VE"
```



```
Option "AGPMode" "4"  
Option "ForcePCIMode" "on"  
Option "AGPFastWrite" "on"  
#Option "EnablePageFlip" "on"  
#Option "composite_sync" "off"  
BusID "PCI:1:0:0"  
Screen 0  
EndSection
```

VideoAdaptor

O této famózní sekci jsem se všude dočetl jedině: Nikdo neví, kdy a proč vznikla, jakou má funkci a proč vlastně existuje.

Monitor

V této sekci se definují monitory – fyzická zařízení, která budou později svázána s obrazovkou. Monitor je zobrazovací zařízení, které má za úkol zobrazit přijatý signál v co nejlepší kvalitě. Tím máme zejména na mysli:

1. přijatelné rozlišení;
2. bitovou hloubku;
3. frekvenci.

Platí přitom pravidlo, že všechny podmínky musejí být splněny zaráz; teprve poté je cíle dosaženo. Přijatelné pracovní rozlišení se pohybuje od hodnot 800*600 (dávná minulost), 1024*768 nebo 1152*864 (nedávná minulost), 1280*1024 (současnost) až po 1200*1600 a vyšší (budoucnost). Mám pochopitelně na mysli moderní stroje, protože všechna rozlišení v tomto rozsahu jsou dnes podle okolností používána.

Bitová hloubka čili schopnost zobrazit určitou paletu barev je zásadní pro dnešní mediální svět, a proto cokoliv pod 16bitů nemá smysl (vyšší už je pouze hloubka 32bitová, takže možností tolik není).

Frekvence je zase nejdůležitější pro zdraví. Člověk, který má špatně nastavený monitor, hazarduje se zdravím – a nejedná se jen o zrak. Byť není správcem svého počítače (v zaměstnání), měl by požadovat maximální hodnoty, které hardware umožňuje. Podle mého laického názoru lze takto aplikovat normy Bezpečnosti a ochrany zdraví při práci (BOZP), které jsou součástí/dodatkem Zákoníku práce.

Jak se pozná špatně nastavená frekvence monitoru? Necvičené oko to na první pohled nijak nepozná. Až se „ozve“ krční páteř, hlava, záda a kříž, poznáte. Cvičené oko se zadívá těsně vedle monitoru a periferním viděním se snaží poznat, jestli se určitá místa obrazu tetelí a chvějí. Pokud ano, je nastavena frekvence kolem 60Hz (někdy i trochu méně!). Na moderních (mladších pěti let) monitorech stačí stisknout tlačítko OSD (On Screen Display), což je informační nabídka monitoru. Slouží k ovládání a kalibraci monitoru a obvykle obsahuje základní informace o rozlišení a frekvenci. Správnou frekvenci však musíme nastavit softwarově.

Moderní grafické karty jsou schopné signál vysílat na vysokých frekvencích, pro monitory má význam rozmezí (75)85 až 120 Hz. Pro LCD displeje je pojem „frekvence“ irelevantní – zobrazovací mechanismus je odlišný od toho, na kterém pracuje monitor.

Parametr	Význam hodnoty
VendorName	Výrobce monitoru. Jakýkoliv řetězec.
ModelName	Model monitoru. Jakýkoliv řetězec.
HorizSync	Horizontální obnovovací frekvence monitoru. Pokud máte manuál, bývá obvykle v tabulce kolem strany osm až dvanáct. Zadává se seznam hodnot oddělená čárkami, nebo rozmezí. Např. 40-120.
VertRefresh	Vertikální obnovovací frekvence. Zadává se stejně jako HorizSync, pouze hodnoty jsou nominálně nižší, např. 40-80.
DisplaySize vyska sirka	Rozměr fyzické plochy monitoru v milimetrech. Použito pro vypočítání rozlišení pro obrazovku. Opravdu je jediná možnost vzít pravítko a měřit.
Gamma	Gamma korekce monitoru. Buď jediná hodnota, nebo tři složky RGB v rozmezí 0-1.
Mode	Volba pro vytvoření nového režimu; vhodnější řešení je Modeline
Modeline "jmeno" definice	V podstatě instrukce pro vytvoření nového režimu monitoru. Předem varuji, že hodnoty si nelze vymyslet a že jejich nesprávné užití může monitor poškodit. Taky je ale pravda, že moderní monitory se umí bránit a prostě zobrazí hlášení „Signal out of range“ (Signál mimo rozmezí) nebo podobné. O modelines více níže.
DPMS	Lze zapnout, nebo vypnout úsporný režim monitoru.
DDCmode	Zakáže, nebo povolí příjem informací o monitoru, tj. o frekvenčním rozmezí, předdefinovaných režimech apod.

Pokud tedy zjistíte, že rozlišení, které vidíte, je realizováno na frekvenci 60 Hz, a přitom máte nastavený správný modul pro grafickou kartu, zkuste změnit (roztáhnout) frekvenční rozsah monitoru. Monitor by si měl najít „tu svou“. Je také vhodné experimentovat (ne vždy je to přínos) s parametrem DDCmode. Téměř každý monitor má předdefinované režimy. Je na vás, abyste ho donutili je použít. Některý to udělá automaticky, jinému je nezbytné vnutit přesné frekvenční rozmezí (metodou pokus-omyl) za pomoci DDCmode.

*Příklad z praxe. Starší digitální monitor Fujitsu ErgoPro x152 z bazaru (tehdy cca 1500 Kč) reaguje na téměř jakýkoliv požadavek tím, že zvolí nejvyšší frekvenci, tj. od 75 do 85 Hz. Oproti tomu nový Hyundai ImageQuest QV790, údajně nejmenší 17" monitor na světě ;-), místo aby zvýšil frekvenci, zvýší rozlišení i za cenu vyloženě nízké frekvence 60 Hz. Stále jsem místo požadovaných 1280*1024*16bit*85+ dostával 1600*1200*16bit*60-. Sladění karty a monitoru trvalo několik hodin.*

Pojďme k modelines. O překlad se ani nesnažím, snad „definice režimu“. Jedná se o řádek, ve kterém zadáme fyzické hodnoty pro nový režim. Počet režimů je u každého monitoru jinak omezený (i když by měly platit standardy) a záleží na monitoru, zda je schopen hodnoty použít. Existují tři způsoby jak řádek s Modeline získat.

1. V souboru /var/log/XFree.0.log se zobrazují řádky Modeline, které jsou k dispozici, ale nebyly použity.

```
(II) RADEON(0): Not using mode "1152x864" (width too large \\  
for virtual size)  
(II) RADEON(0): Not using default mode "1152x768" (width too \\  

```

```

large for virtual size)
(--) RADEON(0): Virtual size is 1024x768 (pitch 1024)
(**) RADEON(0): *Mode "1024x768": 94.5 MHz, 68.7 kHz, 85.0 Hz
(II) RADEON(0): Modeline "1024x768" 94.50 1024 1072 1168 \
1376 768 769 772 808 +hsync +vsync
(**) RADEON(0): Default mode "832x624": 57.3 MHz, 49.7 kHz, \
74.6 Hz
    
```

Stačí vyseknout požadovanou část a umístit ji do sekce **Monitor** v konfiguračním souboru.

2. Online generátory. Ne vždy fungují na 100 %, ale obvykle dostačují. Např. [XTiming](#) [11], [Colas XFree Modeline Generator](#) [12] nebo [Modeline Tool](#) [13].
3. Ruční vytvoření za pomoci programu [xvidtune](#) nebo [videogen](#) [14]. Vhodné pouze pro adrenalinové sportovce; já jsem si vždycky pomocí [xvidtune](#) spolehlivě přizpobil systém.

Ukázka sekce **Monitor**.

```

Section "Monitor"
    Identifier "Monitor0"
    VendorName "Monitor Vendor"
    ModelName "Monitor Model"
    HorizSync 40-100
    VertRefresh 60-130
    Option "DDCmode" "off"
    Option "dpms" "off"
    Modeline "1600x1200" 202.50 1600 1664 1856 2160 1200 1201 1204 1250 +hsync +vsync
    Modeline "1400x1050" 155.80 1400 1464 1784 1912 1050 1052 1064 1090 +hsync +vsync
    Modeline "1280x1024" 157.50 1280 1344 1504 1728 1024 1025 1028 1072 +hsync +vsync
    Modeline "1280x1024" 135.00 1280 1296 1440 1688 1024 1025 1028 1066 +hsync +vsync
    Modeline "1152x864" 108.00 1152 1216 1344 1600 864 865 868 900 +hsync +vsync
    Modeline "1024x768" 94.50 1024 1072 1168 1376 768 769 772 808 +hsync +vsync
    Modeline "1280x960" 148.50 1280 1344 1504 1728 960 961 964 1011 +hsync +vsync
    Modeline "1280x960" 108.00 1280 1376 1488 1800 960 961 964 1000 +hsync +vsync
EndSection
    
```

Modes

Grafické režimy zde lze definovat i nezávisle na zobrazovacím zařízení. Nemám s tím zkušenosti.

Screen

Soubor může opět obsahovat více těchto sekcí. V každé z nich se spojuje zařízení grafické karty (sekce **Device**) se zobrazovacím zařízením (sekce **Monitor**), čímž vznikne obrazovka (screen). Každé originální spojení má své číslo. (V případě více karet je již uvedeno v sekci **Device**).

```

Section "Screen"
    Identifier "Screen0"
    Device "radeon0"
    
```

```
Monitor "Monitor0"
DefaultDepth 16
Subsection "Display"
    Depth 16
    #Modes "1280x1024" "1152x864"
    Modes "1024x768"
EndSubsection
EndSection
```

Každá sekce `Screen` musí obsahovat podsekcí `Display`. V ní určujeme buď standardní, nebo dříve – pomocí `Modeline` – vytvořené režimy. Myslím, že ukázka říká vše a je dostatečně srozumitelná. Existuje také množství velmi specializovaných parametrů, které povolují, či zakazují, různá rozšíření X serveru. Nikdy jsem je nepotřeboval a ani neznám jejich účel.

DRI

Nastavení Direct Rendering Interface, což je rozhraní pro hry, není téměř žádné. Je vlastně pouze nutné stanovit práva pro zařízení tak, aby ho mohli využívat všichni uživatelé. Podmínkou je samozřejmě načtený odpovídající modul a podpora v jádře, ale to je jiná kapitola.

```
Section "DRI"
    Mode 0666
EndSection
```

Uživatelské dotazy

Modelines

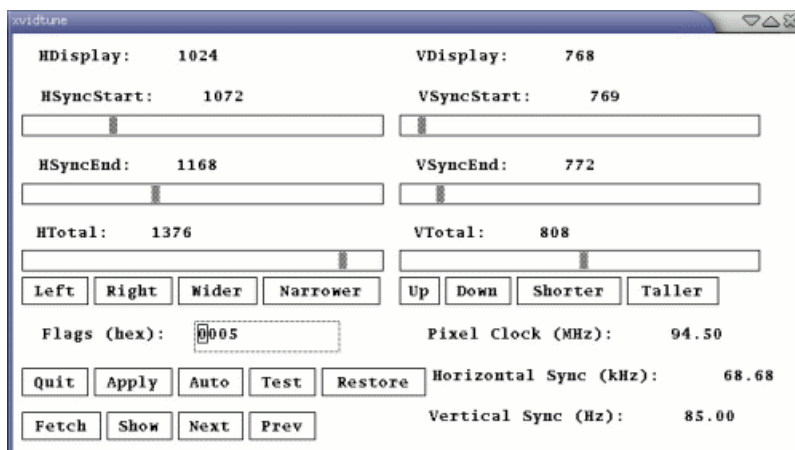
Dotaz. Prosím o radu, kterak vypočítat modeline pro rozlišení 1024x768 při 75 Hz. Při použití XFree 3.3.6 mi to jde bez problému samo, v XFree 4.3.0 mi však obraz zhasíná a problikává.

Odpověď. Nejlepší je použít některou z webových kalkulaček.

- <http://xtiming.sourceforge.net/cgi-bin/xtiming.pl> [15]
- <http://koala.ilog.fr/cgi-bin/nph-colas-modelines> [16]
- <http://www.hut.fi/Misc/Electronics/faq/vga2rgb/calc.html> [17]
- <http://www.dkfz-heidelberg.de/spec/linux/modeline/> [18]
- <http://www.dynaweb.hu/opensource/videogen/> [19]

Dalším řešením je prozkoumání souboru `/var/log/XFree.0.log`. Jsou v něm informace z běhu serveru. Mezi nimi je také možné nalézt modelines, které nebyly z nějakého důvodu použity. Občas lze najít i ten řádek, který můžeme využít.

Třetí příklad, který mě napadá, je utilita `xvidtune`, s jejíž pomocí je možné modelines nastavit. Je to činnost na starším hardwaru potenciálně nebezpečná a celkově nepřehledná. Spusťte program z emulátoru terminálu (`xterm`); do něj se budou požadované informace vypisovat.



Chybové zprávy

Dotaz. Po spuštění XFree mi systém vypíše nějaké chybové hlášky, ale nestačím je při startu přečíst. Zapisují se někam do logu?

Odpověď. Ano, najdete je v již zmiňovaném souboru `/var/log/Xfree.0.log`. Při spuštění X serveru je stávající soubor přejmenován, obvykle získá příponu `.old`. V některých systémech existuje ještě soubor `.xsession-errors`, který se nachází v domovském adresáři uživatele a obsahuje chybová hlášení procesů, které spouští *uživatel* při startu X serveru.

Automatické spuštění XFree

Dotaz. Chtěl bych, aby se mi XFree s KDE spustilo hned při startu. Abych nemusel pokaždé psát `startx`. Nikde jsem nenašel tuto volbu. Asi jsem blbě hledal, můžete mi helpnout?

Odpověď. Existují v podstatě dva způsoby jak spustit grafický server.

Příkazem `startx` z terminálu. Musí se to udělat ručně a spustíme tak pouze vybrané prostředí (KDE, GNOME, fluxbox, xfce nebo libovolné jiné).

Pomocí `runlevelu` neboli úrovně běhu systému. Systém se vždy nachází v některém runlevelu. Jeden je při načítání, další je pro přechod do víceuživatelského, nebo jednouživatelského režimu, další je pro podporu síťového prostředí, jiný je pro ukončení běhu systému. Existuje také runlevel pro grafické prostředí. Pro každou úroveň jsou v konfiguračních souborech stanoveny programy a služby, které se mají spustit, případně ukončit.

```
# Takto jsou runlevely definovány ve Slackware
# 0 = zastavení systému
# 1 = jednouživatelský (správcovský) režim
# 2 = nepoužito, stejné jako 3
# 3 = víceuživatelský režim, standard
# 4 = spuštění XFree se správcem přihlášení XDM, KDM, nebo GDM
# 5 = nepoužito, stejné jako 3
# 6 = reboot
#
#####
#
# Takto jsou runlevely definovány v SUSE 9.1
#
# runlevel 0 je zastavení systému - nepoužívat jako standard!
```

```
# runlevel 1 je jednouživatelský režim
# runlevel 2 je víceuživatelský režim bez podpory sítě
# runlevel 3 je víceuživatelský režim s podporou sítě
# runlevel 4 není použit
# runlevel 5 je víceuživatelský režim s podporou sítě a XFree
# runlevel 6 je restart systému - nepoužívat jako standard!
```

Pro XFree je to runlevel číslo 4, v některých distribucích 5. Příkaz pro manuální změnu runlevelu je `telinit číslo-runlevelu`. Takto může správce systému (root) manuálně měnit stav, ve kterém se systém nachází.

Pro nastavení průběhu startu systému existuje soubor `/etc/inittab`. V něm najdeme řádek

```
# Default runlevel. (Do not set to 0 or 6)
id:4:initdefault:
```

Číslo stanoví runlevel, do kterého se systém přepne po dokončení startovacích rutin. Uvedený příklad je pro Slackware. Dvojtečky jsou důležité!

Jak shodit XFree?

Dotaz. Ahoj, jak mám shodit XFree server? Potřebuju nainstalovat nvidia ovladače, a nevím jak shodit XFree.

Odpověď. Pokud máte X server spuštěný výše uvedeným způsobem, jediná možnost obvykle bývá přepnout se na některou konzoli (na třetí např. `Ctrl-Alt-F3`), přihlásit jako root a zadat příkaz pro přepnutí do víceuživatelského režimu bez grafického prostředí. Tedy ve Slackwaru např. příkazem `telinit 3`. Po provedení změn v konfiguraci XFree se vrátíte do čtvrtého runlevelu příkazem `telinit 4`.

Pokud nemáte v konfiguračním souboru zakázanou klávesovou zkratku (parametr `DontZap`), lze X server shodit kombinací `Ctrl-Alt-Backspace`. Více informací najdete v popisu sekce `ServerFlags` v prvním díle tohoto seriálu [20].

Tento postup je ale účinný pouze v případě, že jste X server spustili příkazem `startx`; pouze v tomto případě se okamžitě dostanete do terminálu. Pokud používáte správce přihlášení XDM, KDM nebo GDM, celý grafický systém se *pouze restartuje* a znovu se zobrazí nabídka k přihlášení. Z toho vyplývá, že můžete (za běhu X) provést změny, přepnout se do X a klávesovou kombinací `Ctrl-Alt-Backspace` je restartovat. **Jde opravdu o tvrdý restart, žádná data se neukládají, tak si je nezapomeňte uložit!**

Server se znovu spustí a načte již novou konfiguraci. (Neplatí pro font server, ten je nutné restartovat ručně. To je ale nutné pouze v případě, že provádíte změny v jeho nastavení.)

Instalace karty

Dotaz. Potřebuji nainstalovat ovladače grafické karty, ale nevím kde začít. Instalace přiřadila kartu do generické skupiny „VESA“. Přesné označení typu to asi nebude, asi spíš výrobce. Bohužel ani nevím, jak ty informace získat (tištěné materiály nemám k dispozici). Lze tu kartu detekovat automaticky? Půjde vůbec rozchodit?

Program lspci

Odpověď. Začít bychom měli utilitou `lspci`, která zobrazí zařízení sběrnice.

```
# lspci
```

```
00:00.0 Host bridge: VIA Technologies, Inc. VT8366/A/7 [Apollo KT266/A/333]
00:01.0 PCI bridge: VIA Technologies, Inc. VT8366/A/7 [Apollo KT266/A/333 AGP]
00:0a.0 Multimedia video controller: Brooktree Corporation Bt878 Video Capture (rev 11)
00:0a.1 Multimedia controller: Brooktree Corporation Bt878 Audio Capture (rev 11)
00:0b.0 Multimedia audio controller: Ensoniq ES1371 [AudioPCI-97] (rev 06)
00:10.0 USB Controller: VIA Technologies, Inc. USB (rev 80)
00:10.1 USB Controller: VIA Technologies, Inc. USB (rev 80)
00:10.2 USB Controller: VIA Technologies, Inc. USB (rev 80)
00:10.3 USB Controller: VIA Technologies, Inc. USB 2.0 (rev 82)
00:11.0 ISA bridge: VIA Technologies, Inc. VT8235 ISA Bridge
00:11.1 IDE interface: VIA Technologies, Inc. VT82C586/B/686A/B PIPC Bus Master IDE (rev 06)
00:12.0 Ethernet controller: VIA Technologies, Inc. VT6102 [Rhine-II] (rev 74)
01:00.0 VGA compatible controller: ATI Technologies Inc Radeon RV100 QY [Radeon 7000/VE]
```

Jak je vidět z výpisu, grafická karta má `BusID 1:00.0`. To je také první parametr, který zapíšeme ve formě `PCI:sběrnice:zařízení:funkce` do sekce `Device`. Hodnota `PCI:1:0:0` odpovídá v tomto případě AGP kartě.

```
Section "Device"
    Identifier "radeon0"
    Driver "radeon" # volitelne "ati"
    VendorName "ATI Radeon VE"
    Option "AGPMode" "4"
    Option "ForcePCIMode" "on"
    Option "AGPFastWrite" "on"
    BusID "PCI:1:0:0"
EndSection
```

Nálepka

Nepodceňujme také nálepky na samotném zařízení, mnohdy pro identifikaci postačují. Jen je nutné stroj rozdělat a trochu se potrápit při vytahování karty a dýchání x let staré prachové usazeniny ...

Samotný X server

Samotný X server má také metody pro detekci hardwaru – nejsou ale tak spolehlivé jaké první řešení.

```
X -scanpci nebo X -probeonly
```

Poté již následuje pouze hledání na Internetu a zkoušení různých kombinací. Režim VESA běží, pokud je mi známo, pouze v pevně stanovených frekvencích; pokud je váš monitor nepodporuje, bývá právě tohle jádrem problému. Je třeba experimentovat. Tohle se ale týká pouze starých karet, nové takové problémy nemívají.

Změna rozlišení za běhu

Dotaz. Nevíte někdo jak změnit rozlišení v Xkách (ne jenom velikost zobrazované plochy) pomocí nějakého programu? Mám na mysli něco jako je ve Win98 ikonka vpravo dole, na kterou když poklepu, rozbálí se nabídka možných rozlišení a když na jedno z nich kliknu, rozlišení se změní, tedy bez toho, abych musel editovat konfigurak Xek.

Odpověď. Pokud máte v konfiguračním souboru definováno více režimů a povoleny klávesové zkratky, lze rozlišení přepínat pomocí **Ctrl-Alt-šedé plus** a **Ctrl-Alt-šedé mínus**. Více v prvním dílu.

Existuje ale také rozšíření X serveru (načítá se obvykle automaticky), které umožňuje přepínání podle možností karty (není třeba režimy explicitně definovat). Nazývá se **XRandr**, jedná se o relativní novinku, které ještě nemá žádoucí podporu a není stoprocentně funkční ve všech kombinacích. Nové KDE nebo GNOME jej však samozřejmě podporují.

Někdy musí nastat konec

Problematicke bychom se mohli věnovat ještě dlouho, právě proto máme diskuze. Já jen doufám, že jsem tímto shrnutím pomohl někomu zkrátit čas při nastavování grafického systému Linuxu.

Odkazy

- [1] <http://www.xfree86.org/4.3.0/>
- [2] <http://www.x.org/>
- [3] <http://www.y-windows.org/>
- [4] <http://www.xfree86.org/>
- [5] <http://www.abclinuxu.cz/clanky/show/7724>
- [6] <http://www.xfree86.org/current/mouse.html>
- [7] <http://www.xfree86.org/4.3.0/XKB-Config.html>
- [8] <http://www.xfree86.org/current/mouse6.html#26>
- [9] <http://www.xfree86.org/current/mouse5.html#21>
- [10] <http://www.xfree86.org/4.3.0/XKB-Config.html>
- [11] <http://xtiming.sourceforge.net/>
- [12] <http://koala.ilog.fr/cgi-bin/nph-colas-modelines>
- [13] <http://www.dkfz-heidelberg.de/spec/linux/modeline/>
- [14] <http://www.dynaweb.hu/opensource/videogen/>
- [15] <http://xtiming.sourceforge.net/cgi-bin/xtiming.pl>
- [16] <http://koala.ilog.fr/cgi-bin/nph-colas-modelines>
- [17] <http://www.hut.fi/Misc/Electronics/faq/vga2rgb/calc.html>
- [18] <http://www.dkfz-heidelberg.de/spec/linux/modeline/>
- [19] <http://www.dynaweb.hu/opensource/videogen/>
- [20] <http://www.abclinuxu.cz/clanky/show/59104>

Diskové kvóty

Petr Bravenec

Co jsou diskové kvóty a jak jich využít k ukáznění nepořádných uživatelů. Popis instalace a nastavení. Spolupráce s Windows.

Úvod

Tímto článkem pravděpodobně neřeknu zkušenějším správcům Linuxu nic nového. Ale vždy se najde dostatečný počet začínajících, kteří potřebují nějakým způsobem omezit roztahovačného uživatele, a nevědí jak na to. Ale nakonec i zkušenější se někdy dají nachytat: „Strčím do serveru 160 GB disk, to ti čuníci hned tak nezapraskají...“ Jenže po pár týdnech nevycházejí z údivu, když je disk ze sedmadvadesáti procent plný životně důležitých dat – zálohy nějakého obskurního účetního serveru, který byl zrušen už v první polovině devadesátých let minulého století, zálohy záloh, patnáct celovečerních filmů, čtyři PST soubory, každý o velikosti neuvěřitelných dvou GB (jeden originál a tři nepostradatelné zálohy), kopie instalačního CD Windows 95 (a samozřejmě záloha) ...

Disky sice dneska nejsou příliš drahé, ale každého rozumného správce musí po uklidnění a odložení studeného obkladu z čela napadnout, že kupovat další disk možná nebude ideální řešení. No a protože služební brokovnice nebývá ve firmách zcela obvyklá, je nutné sáhnout po řešení někde mezi fyzickou likvidací a koupí dalšího disku. Takovým řešením mohou být diskové kvóty.

Jak to pracuje

Diskové kvóty nejsou jen jednoduchým omezením diskového prostoru. Po správné konfiguraci jsou uživatelé omezováni ve svém užívání hned několika způsoby (ponechávám anglické názvy, protože ty se používají při konfiguraci):

Hard limit

Pevná mez, přes kterou vlak nejede. Uživateli se nikdy nepodaří na disk uložit více, než je uvedeno v tomto parametru.

Soft limit

„Nezávazná mez“ – uživatel může uložit na disk i více, ale při překročení této meze dostane od systému varování.

Inodes, Blocks

Kvóty lze nastavit jak na celkový objem dat na disku, tak na počet souborů.

Grace period

Uživatel může dočasně uložit na disk více, než je uvedeno v parametru „soft limit“ na dobu zadanou parametrem „grace period“. Po uplynutí této doby se uživateli nepodaří na disk uložit více, i když ještě nepřekročil mez zadanou parametrem „hard limit“.

Limity lze pochopitelně nastavovat pro každého uživatele nebo skupinu uživatelů zvlášť. Omezit lze jak celkovou velikost souborů, tak i jejich počet.

Nutné ingredience

Základním předpokladem pro fungování diskových kvót je podpora v kernelu. V [howto \[1\]](#) je doporučováno implementovat diskové kvóty opatchováním kernelu. Nicméně osobně bych se tomuto postupu vyhnul. V inteligentní distribuci by měly být diskové kvóty součástí systému a mělo by postačovat pouze nainstalovat potřebné balíčky. Není-li ve vaší distribuci podpora kvót, sáhněte raději po nějaké jiné distribuci. Ta vaše bude orientovaná pravděpodobně více na desktop a na serverové použití bude méně vhodná.

Osobně používám distribuci Red Hat (už od verze 3) a můžu zodpovědně prohlásit, že kvóty byly určité ve verzi 4.2 (nejstarší CD, které se mi válí v šuplíku) – s největší pravděpodobností už dříve. V distribuci Red Hat stačí nainstalovat balíček quota a zkonfigurovat systém.

Vlastnímu kernelu bych se vyhnul i z toho důvodu, že existují dvě různé verze kvót. Zatímco Red Hat 8 například podporuje novější verzi, standardní kernel stejné verze (údajně) používá verzi starší. Potom samozřejmě můžete čekat nečekané, když si novější utility z distribuce nebudou rozumět s kernelem.

Dostanete-li se ovšem do situace, kdy distribuce kvóty neobsahuje a změna z jakéhokoliv důvodu není možná, najdete potřebné balíky zde:

Jádro

<ftp://ftp.kernel.org> [2]

Jaderná záplata

<ftp://atrey.karlin.mff.cuni.cz/pub/local/jack/quota/> [3]

Utility

<http://sourceforge.net/projects/linuxquota/> [4]

Quota mini-HOWTO

<http://www.tldp.org/HOWTO/Quota.html> [5]

Postup při záplatování, konfiguraci a překladau jádra byl popsán například zde na Abc Linuxu v článku [Kompilovanie jadra](#) [6].

Základní konfigurace, spuštění

Každý souborový systém, na kterém chcete mít aplikované kvóty, musí být přimontovaný s volbou `usrquota`. I když je v manuálové stránce příkazu `mount` uvedeno, že se tato volba ignoruje, rozhodně není ignorována různými utilitami – musí být proto uvedena. Volbu uvedeme do souboru `/etc/fstab`.

```
LABEL=/          /          ext3    defaults          1 1
LABEL=/uzivatele /uzivatele ext3    usrquota,grpquota 1 2
```

V horním řádku (systém souborů `/`) nejsou kvóty použité. Podobně vypadal před úpravou i druhý řádek. Na serveru poskytujícím diskový prostor by rozhodně měly být uživatelské diskové oblasti oddělené od systémových (`root`, `var`, `usr` a podobně).

Diskové kvóty nelze použít na všech typech souborových systémů – určitě se vám nepodaří aplikovat kvóty třeba na FAT nebo NFS. Bez problémů je to ale u obvyklého systému `ext2` a `ext3`.

V nejvyšším adresáři souborového systému vytvořte soubory pro práci kvót. I když podle manuálové stránky by měl příkaz `quotacheck` vytvořit soubory sám, nevěřte tomu:

```
touch aquota.user
touch aquota.group
chmod 600 aquota.*
```

U distribuce Red Hat tímto veškerá instalace téměř končí. Všechno ostatní by měly obstarat startovací skripty (konkrétně `/etc/rc.d/rc.sysinit`). Nejjednodušším pokračováním je restart. Nemůžete-li systém restartovat, proveďte instalaci ručně:

```
quotacheck -avug
quotaon -avug
```

Program `quotacheck` může protestovat, že systém souborů není přimontovaný pouze pro čtení a doporučí vám volbu `-m`. U distribucí, které nemají kvóty shůry dáno, musíte zajistit startování výše uvedených dvou příkazů při startu systému.

Nastavení pro jednotlivé uživatele

Příkazem `edquota` můžeme měnit kvóty jednotlivých uživatelů. Příkaz vyžaduje nejméně jeden parametr – uživatelské jméno. Máte-li v systému více diskových oblastí s kvótami, bude nutné použít ještě parametr `-F`:

```
edquota -F /home marena
```

V editoru se vám otevře tabulka, ve které můžete nastavit pro uživatele kvóty:

```
Disk quotas for user marena (uid 1030):
Filesystem  blocks  soft  hard   inodes  soft  hard
/dev/md0    24  1000  1500     3      0    0
```

Hodnoty upravíme, ukončíme editor – a od té chvíle by měl mít uživatel nastavené kvóty.

Grace period – doba, po kterou může mít uživatel překročený soft limit – se nastavuje pro celý disk společně, nelze ji nastavit pro každého uživatele zvlášť. Nastavuje se příkazem

```
edquota -t -F /home
```

Čas lze nastavit od sekund po měsíce – pro označení použijte anglické názvy (seconds, minutes, hours, days, weeks, months). Mezi číslicí a názvem se nesmí objevit mezera.

Příkaz `edquota` volá normálně editor `vi`. Nevyhovuje-li vám tento editor, můžete ovlivnit chování příkazu `edquota` pomocí proměnné prostředí `EDITOR`:

```
export EDITOR=ed
```

Místo editoru `ed` napište cestu na vlastní oblíbený editor. Ed jsem uvedl jen jako žert (dostanete se z něj písmenem `q`).

Nastavení editoru můžete využít při hromadném zavádění kvót pro větší množství uživatelů. Nastavte vzorové kvóty uživateli `standa` a pro celou vaši školku pak můžete nastavit kvóty podle tohoto vzoru jednoduchým skriptem:

```
set EDITOR=true
for i in majdalenka apolenka veronika verka zdenka majka lenka monika
```

```
do
edquota -p standa $i
done
unset EDITOR
```

Spolupráce s Windows

Kvóty se hlídají i při přístupu k disku z Windows protokolem smb. Při takovém použití kvót se můžete dostat do situace, že ačkoli je na disku nastavená kvóta 10 MB, podaří se vám uložit na disk třeba dva osmimégové soubory (dohromady tedy 16 MB). Příkaz `ls -l` říká 16, příkaz `du -s` říká 10, `repquota` říká 10. Soubory prostě mají menší velikost, než je udávaná.

Manuál (samba verze 2) pro tuto situaci doporučuje použití parametru samby

```
strict allocate = yes
```

Nevýhodou je údajné zpomalení na některých systémech.

Při experimentech jsem si nevšiml, že by se Windows nějakým způsobem dozvěděly o překročení soft limitu. Aby se dal efektivně využít mechanismus „grace period“, je potřeba dát uživatelům dát vědět jiným způsobem. K tomu se dá využít program `repquota`. Tím zjistíme využití kvót jednotlivými uživateli a při překročení pošleme uživateli poštu:

```
repquota /home |
awk '($2~"-" || $2~"+") && $4!=0 && $3>$4 {print $1, $3, $4, $5;}' |
while read user block soft hard;
do
export user, block, soft, hard
(echo "Mate na disku vice dat nez je povoleno:"
echo "Na disku: $block"
echo "Povoleno: $soft" ) |
mail -s "Prekrocene kvoty" $user
done
```

Dávka by měla být pravidelně spouštěná z cronu. Uvedený příklad je samozřejmě velmi jednoduchý, uživatelé by jistě uvítali například seznam jimi vlastněných souborů. Zvláště ve sdílených adresářích mívají uživatelé strašný binec a často vůbec netuší, které soubory jsou jejich.

Závěr

Diskové kvóty jsou mocným nástrojem na umravňování nepořádných uživatelů. Notoričtí zálohovači internetu vás za to nebudou mít rádi, takže tvrdá pravidla můžete uplatňovat jen tak dlouho, dokud se jednou ve firmě bez vašeho vědomí neobjeví služební brokovnice...

Odkazy

- [1] <http://www.tldp.org/HOWTO/Quota.html>
- [2] <ftp://ftp.kernel.org>
- [3] <ftp://atrey.karlin.mff.cuni.cz/pub/local/jack/quota/>
- [4] <http://sourceforge.net/projects/linuxquota/>
- [5] <http://www.tldp.org/HOWTO/Quota.html>
- [6] <http://www.abclinuxu.cz/clanky/show/59472>

Extra multimediální systém XMMS

Vlastimil Ott

Zaslouží si přehrávač XMMS pozornost? Co vlastně umí s pomocí pluginů?

XMMS je přehrávač zvukových souborů všech možných typů. Protože nedávno konečně vyšla další verze [1], pojďme si jej blíže představit. Aktuální verze 1.2.10 pouze opravuje dřívější chyby a nepřináší nic nového. To je asi fakt, který se mi na XMMS nejvíce nelíbí – jeho pomalý vývoj.

XMMS se stal standardem pro přehrávání zvuku v Linuxu, i když mu může slušně konkurovat třeba MPlayer [2], ačkoliv ten je určen spíše pro video. Téměř všechno je v XMMS realizováno pomocí pluginů. Modularita programu může být někdy na obtíž – když prostě nemáte plugin, nemůžete číst nebo ukládat to či ono. Ve standardním balíčku s programem jsou ale všechny běžně potřebné pluginy k dispozici.

Pluginy

Plugin – česky obvykle „zásuvný modul“ – je knihovna (má příponu `.so`), která je při startu programu načtena a její funkce jsou uživateli zpřístupněny nějakou volbou programu. Je možné sehnat knihovny již zkompileované, ale stejně tak je běžné, že si je uživatel musí zkompileovat sám.

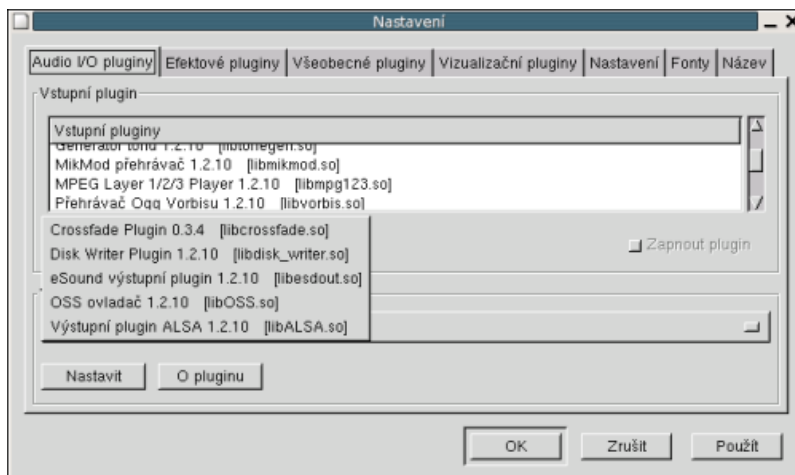
Místo, kde bychom měli pluginy hledat, je sekce na webu XMMS [3]. Podle funkce jsou rozděleny na vstupní, výstupní, efekty, vizualizace, všeobecné a různé. Je tedy možné instalovat plugin do libovolné kategorie. V programu je potom vyvoláme klávesovou zkratkou `Ctrl+P`.

Vstupní pluginy

V této sekci jsou zobrazeny pluginy pro čtení zvukových souborů. Standardně se jedná o přehrávání

- CD audio
- MPEG Layer 1/2/3
- OGG Vorbis
- FLAC, VQF (bohužel neznám)
- WAVE vzorků
- starých modulů typu `.mod` a dalších (zlatá Amiga)
- plus generování tónu o určité frekvenci

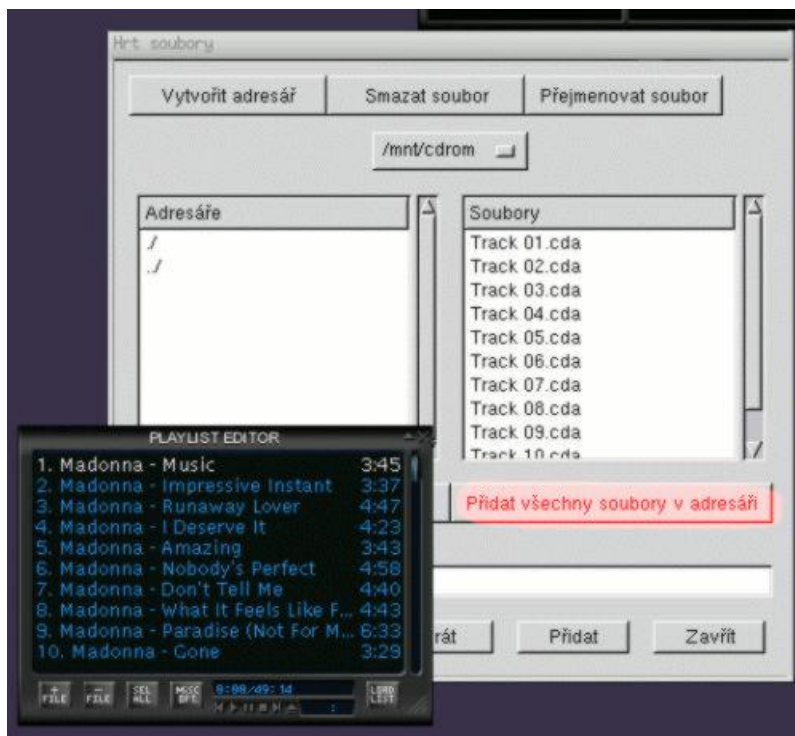
Každý plugin je standardně zapnutý, proto tento parametr zkontrolujte, pokud vám přehrávání nefunguje.



U mnohých pluginů lze tlačítkem **Nastavit** specifikovat přesnější parametry než ty, které byly standardně nastaveny. To se může týkat například pluginu pro přehrávání audio cd, kde je v případě více CD-ROM mechanik nutné vybrat tu „správnou“, nebo se hodí spuštění CDDb systému (vyhledá na Internetu název písničky).

Výstupní pluginy

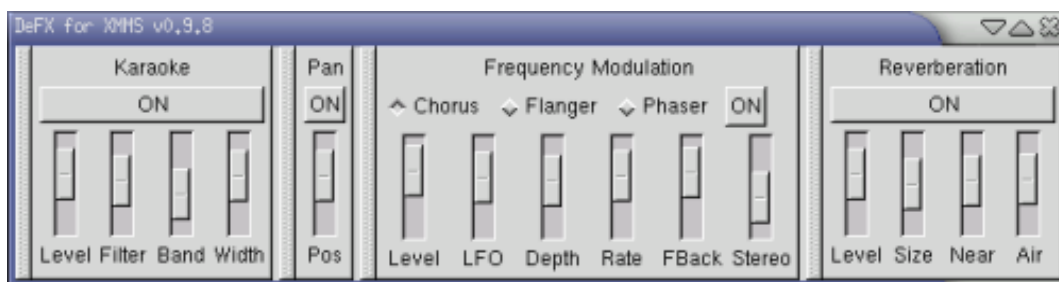
Jedná se o programový kód, který formuje zvuková data do konečné formy. Záměrně to píšu takto složitě, protože to není vždy zvuk, co je na výstupu. Standardním výstupem samozřejmě zvuk je; dnes je realizován zvukovým systémem **ALSA** [4], je ale stále podporován překonaný systém **OSS** [5]. Doporučení je jasné – volte ALSA. Dále mohou být na výstupu opět zvuková data, tentokrát jako WAV soubor uložený na disk. O to se stará **Disk Writer Plugin**; výsledkem je soubor typu **RIF** (little-endian) data, **WAVE** audio, **Microsoft PCM**, 16 bit, stereo 44100 Hz.

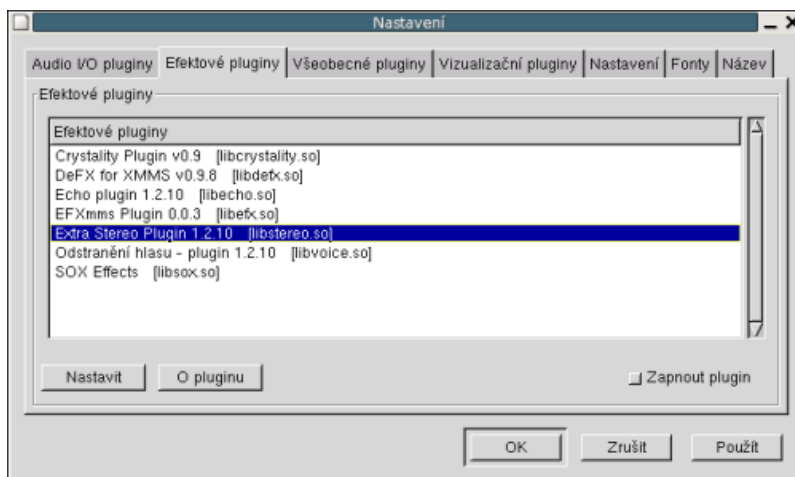


Posledním typickým zástupcem je podpora **Esound** [6], což je jeden ze zvukových démonů. Je ale možné doinstalovat jiné výstupní pluginy [7], například pro spolupráci s **aRtsd** [8] (zvukový server pro KDE), pro výstup do **OGG** [9] nebo **MP3** [10], pro **DJs** [11] či pro modulování výstupu – efekt **crossfade** [12]. Všechny zde uvedené (a některé navíc) najdete v odpovídající sekci [13].

Efektové pluginy

Efektové pluginy modulují proud zvukových dat a s jejich pomocí lze dosáhnout více či méně praktických a obskurních zvuků a pazvuků. Obvykle lze každý z nich nastavit, pokud to ovšem sám podporuje. Mně se osvědčily pluginy, které lze vidět na obrázku; bohužel jich zase nejsou stovky, některé nejdou zkompileovat, jiné tvrdě šazují XMMS. Na následujícím obrázku jsou „šavle“ efektu DeFX. Záměrně píšu „šavle“, protože tento dialog neslouží pouze k nastavení – je určen k reálnému modulování zvuku. To znamená, že zvuk se mění okamžitě. Nejlepší je volba Karaoke a Reverberation – zvuk zní velice amatérsky, a navíc ještě jakoby z velkého potrubí. No a když ten zvuk je třeba Britney...;-)





Zkrátka, je třeba trošku zkusit a hrát si. Zvukový efekt se špatně popisuje, je prostě nejlepší si ho poslechnout. Otázkou zůstává, jestli je dostatek efektů např. pro DJ nebo jestli lze XMMS využívat k něčemu jinému, než je domácí zábava.

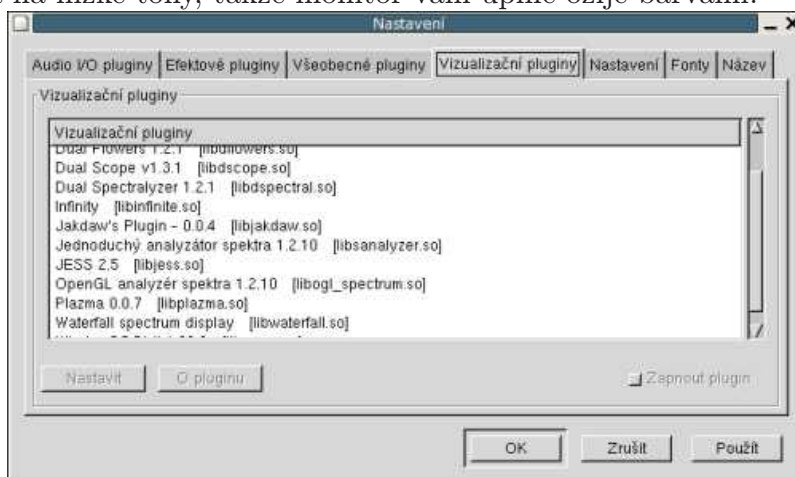
Vizualizační pluginy

Nejzábavnější použití pluginů jsou právě ty, které se starají o vizualizaci hudby. Více či méně se jim to daří. Mnoho z nich používá rozhraní OpenGL, což je jednak velice efektivní, a jednak také celkem náročné na výkon stroje. Ukázku některých efektů vidíte na následujících obrázcích. Všechny neoznačené obrázky mají redukovanou velikost kolem 20kB.

Poznámka editora. Obrázky jsou přístupné pouze na webu.

- JESS: jedna [14], dva [15], tři [16]
- Jakdaw's plugin: jedna [17], dva [18]
- Infinity: jedna [19] (64 kB), dva [20] (42 kB), tři [21] (64 kB)
- What a GOOM! [22]
- další: jedna [23], dva [24]

Některé z nich lze spustit v celoobrazovkovém režimu, takže efekt je dokonalý. Některé velice efektně reagují na rytmus a na nízké tóny, takže monitor vám úplně ožije barvami.



Všeobecné pluginy

Standardně jsou přístupné tři: pro ovládání dálkovým ovladačem, pro ovládání joystickem a pro spuštění programu v okamžiku změny skladby. Dálkový ovladač k televizní kartě se mi s XMMS zatím rozchodit nepodařilo, ale joystick jede od okamžiku, kdy je při startu inicializováno zařízení

`/dev/js0` nebo vyšší. Lze transparentně nastavit, co má vykonat pohyb pákou či stisk tlačítka. Hodně záleží na citlivosti, proto pozor na uši, máte-li sluchátka – pohyb kniplem k sobě vyžene hlasitost k maximu během sekundy.

Plugin Změna skladby umožní spouštět program při začátku či konci skladby s parametrem `nazev_skladby`. To znamená, že program může název zobrazit na obrazovce, poslat ho někomu mailem či zapsat do logu.

Nastavení

V této části lze do určité míry nastavit chování programu. Možností je ale méně, než by si člověk přál. Prostě se mi zdá, že jsou možnosti nastavení příliš „chudé“. Nevím, co mi chybí, ale něco asi ano. Ale možná je to můj subjektivní dojem. Velká část nastavení se věnuje názvům souborů a jejich správné interpretaci. Dále také několik voleb k přehrávání, organizaci playlistu a polohování okna, ale to je vlastně vše. K ovládání přehrávače slouží mnoho kláves, které najdete v hlavní nabídce. O té jsme ještě nemluvili – zobrazí se kliknutím na levý horní rožek aplikace a skrývá se v ní ještě mnoho jiných věcí. Je to například playlist neboli seznam přehrávaných skladeb. Ten lze různě modifikovat, načítat, ukládat nebo míchat. Praktickou pomůckou je ekvalizér (`Alt-e`): opravdu efektivně funguje a dokáže měnit jednotlivá zvuková pásma. Stav lze uložit a později načíst. Jenom jsem se na Internetu nedopátral přednastavených profilů.

Skiny

Skiny jsou klasická věc – změna vzhledu. Do jisté míry nás XMMS zklame, protože (pokud je mi známo) není možné změnit jeho „hranatý“ tvar. Žádná divoká párty se tedy nekoná. Tohle považuji za další nedostatek aplikace. Nejsem sice uživatel, který by si na takových drobnostech zakládal, ale domnívám se, že je to důležitá vlastnost. Skinů existují tony, hledejte opět [na stránce XMMS \[25\]](#).

Rozhodně je co zlepšovat

Program patří ke klasice, ale přiznejme si, že má co dohánět. Eufemisticky řečeno je onen pověstný vlak už o stanici dál. Je sice možné předat programu adresu zvukového souboru, který je uložen v internetu, ale QuickTime to rozhodně být nemůže. Také jiným funkcím chybí určitý komfort. Program je také relativně nestabilní, zohledníme-li fakt, že plugin je potenciální „shazovač“ – proto nikdy nespouštějte dva pluginy používající OpenGL. Pokud se aplikace „kousne“, příčinou je obvykle uzamčené zvukové zařízení, nebo nesprávně načtené moduly zvukového systému. Dle mého názoru je XMMS na vymření – a možná právem. Objeví se nějaká další generace přehrávače, třeba derivát MPlayeru? Nechme se překvapit...

Odkazy

- [1] <http://www.xmms.org/download.php>
- [2] <http://www.mplayerhq.hu/>
- [3] <http://www.xmms.org/plugins.php>
- [4] <http://www.alsa-project.org>
- [5] <http://www.opensound.com/>
- [6] <http://www.tux.org/~ricdude/Esound.html>
- [7] <http://www.xmms.org/plugins.php?category=output>
- [8] <http://www.arts-project.org/>
- [9] <http://www.vorbis.com/>
- [10] <http://www.iis.fraunhofer.de/amm/index.html>
- [11] <http://dbmix.sourceforge.net/>
- [12] <http://www.mynetcologne.de/~nc-eisenlpe2/xmms-crossfade/>
- [13] <http://www.xmms.org/plugins.php?category=output>
- [14] <http://www.abclinuxu.cz/images/clanky/ott/xmms-jess1.jpg>
- [15] <http://www.abclinuxu.cz/images/clanky/ott/xmms-jess2.jpg>
- [16] <http://www.abclinuxu.cz/images/clanky/ott/xmms-jess3.jpg>

- [17] <http://www.abclinuxu.cz/images/clanky/ott/xmms-jakdaw1.jpg>
- [18] <http://www.abclinuxu.cz/images/clanky/ott/xmms-jakdaw2.jpg>
- [19] <http://www.abclinuxu.cz/images/clanky/ott/xmms-infinity1.jpg>
- [20] <http://www.abclinuxu.cz/images/clanky/ott/xmms-infinity2.jpg>
- [21] <http://www.abclinuxu.cz/images/clanky/ott/xmms-infinity3.jpg>
- [22] <http://www.abclinuxu.cz/images/clanky/ott/xmms-goom.jpg>
- [23] <http://www.abclinuxu.cz/images/clanky/ott/xmms-dalsi1.jpg>
- [24] <http://www.abclinuxu.cz/images/clanky/ott/xmms-dalsi2.jpg>
- [25] <http://www.xmms.org/skins.php>

Komix: RMS v Praze

Leoš Literák

Pravda o tom, jak skutečně probíhala návštěva Richarda Stallmana v Praze.

Námět a scénář: Leoš Literák, kresba: Anselm



Dnešním dnem jsme zahájili nový nepravidelný seriál komiksů. Chceme v nich zábavnou formou glosovat aktuální linuxové dění. Doufáme, že vás první díl pobavil. Inspirací pro něj byli moji kamarádi Petr Krčmář a Martin Kysela, kteří se chlubili cenným úlovkem – fotografií s Richardem Stallmanem. Později se stala z toho mánie, viz [tato](#) [1] zprávička.

Další komiksy budeme publikovat, až najdeme další zajímavé téma. Rádi přitom uvítáme vaši pomoc. Ať už tipem nebo celým scénářem, který bychom odměnili honorářem. Pište mně nebo na adresu redakce.

Poznámka editora. Nepatrně zkráceno. Více na <http://www.abclinuxu.cz/clanky/show/60074> [2]

Odkazy

- [1] <http://www.abclinuxu.cz/news/show/59174>
- [2] <http://www.abclinuxu.cz/clanky/show/60074>

Jaderné noviny 260–263

Robert Krátký

Zrychlování SATA. Ext3 pre-alokace založená na rezervaci. Správa SCSI ovladače atp870u. Správcovství BTTV. Porušení GPL firmou Tritton Technologies. LISP interpreter v kernelu. Obejití bufferové keše. Ovladač adm8211 802.11b pro 2.6. Skript pro patchování jádra – ketchup – ve verzi 0.5. Přidání podpory SysFS do DVB subsystému. Crypto CRC32C modul. Odstraňování IDE modulů; IDE a libata. Opravy SiS ovladače. Ovladač AHCI; otevřené hardwarové specifikace. LinuxAnt lze o své licenci. Opravy a aktualizace ReiserFS verze 3. Podpora tabulky atributů stránek v 2.6. SATA bude ve 2.4.27. Automatický hledač chyb odhalil v jádře několik děr. Dokumentace ke kompilaci externích modulů.

Zrychlování SATA, 115 e-mailů

27. bře – 3. dub

Jeff Garzik napsal:

Funkce „lba48“ v ATA umožňuje adresovat sektory > 137GB, a také přenosy až 64K sektorů namísto 256 sektorů ve starším ATA. libata prostě omezilo všechny přenosy na 200 sektorů (pod tou hranicí 256 sektorů). To bylo hlavně kvůli opatrnosti a také jsem chtěl mít řešení, které funguje všude. Také jsem chtěl vědět, jak se ukáže iommu S/G. Vypadá to docela dobře, takže je načase zapnout přenosy s velikostí lba48. Většina SATA disků bude z tohoto a dalších důvodů stejně lba48 – i těch menších než 137GB. S tímto jednoduchým patchem se maximální velikost požadavku zvýší ze 128 K na 32 MB... takže vám může být jasné, že to určitě pomůže výkonu. Zvýší se propustnost. Sníží se počet přerušování. Zábava pro celou rodinu.

Příložený patch je pouze pro jádra 2.6.x. Měl by být aplikovatelný na 2.6.5-rc2 a pozdější, včetně posledního mého patche 2.6-libata z kernel.org. Měl by to být celkem neškodný patch, ale nedá se předvídat, co se může stát, když ten kohoutek otevřete naplno. Bylo by fajn to otestovat v -mm :). Hledá se dobrovolník, který vytvoří backport tohoto patche a pošle jej do linux-ide@vger.kernel.org – já jej pak začlením do mé fronty pro 2.4 libata.

Stefan Smietanowski se zeptal: Co se stane, když bude PATA disk ležet za Marvel bridgem – jak tomu je dnes u většiny SATA disků?

A Jeff odpověděl: Větší přenosy s PATA fungují také. U bridgů je obecně lepší nastavit omezení na UDMA/100 (udma), ale větší přenosy jdou také.

Stefan se ještě zeptal: A jsou v SATA LBA48 přenosy skutečně povinné?

A Jeff odpověděl:

Ano i ne, v tomto pořadí :). SATA nevyžaduje lba48, ale je velmi nepravděpodobné, že bys narazil na SATA disk bez lba48. Tak jako tak, libata podporuje to, co podporuje disk. Starší disky fungují bez problémů.

Na jiném místě poznamenal Nick Piggin, že maximální velikost požadavku 32M je příliš velká. Dodal: Vsadím se, že zisk se začne po asi 1MB rapidně zmenšovat.

Jeff nesouhlasil; řekl, že jeho implementace prostě exportuje maximální hodnoty hardwaru a je na administrátorovi systému, aby pro daný stroj zavedl vhodná pravidla rozvrhu práce disku [disk scheduling]. Následovala dlouhá diskuze, ve které mimo jiné Andrea Arcangeli vyjádřil souhlas s tím, že překročení hranice 1MB nepřinese žádné výrazné zlepšení. Jeff víceméně souhlasil, ale řekl:

Chci říci, že existují dvě maxima:

1. hardwarový limit
2. limit, který „dává smysl“, např. 512k nebo 1M ve většině případů

Ovladač by se měl starat pouze o #1 a měl by být „zpraven“ o #2.

Později dodal: *Myslím, že délka této diskuze jasně ukazuje, že nízkourovňový ovladač by o nastavení té hodnoty neměl rozhodovat.*

Jens Axboe připojil:

Tady je rychlý patch, který se pokouší upravit hodnotu podle posledních dokončených požadavků. Změní → max_sectors na hardwarový limit a přidá → optimal_sectors, který odpovídá největšímu volanému IO cíli. Je rozdělen na READ (čtení) a WRITE (zápis). Snahou je udržet dobu provádění požadavku pod BLK_IORATE_TARGET, který je v tomto patchi 50ms.

Jens pak vyjmenoval ještě několik věcí, které jeho narychlo spíchnutý patch opomíjel. Jeff a ostatní byli rádi, že někdo s něčím takovým přišel a pokračovali v diskuzi o dalších technických aspektech věci.

Ext3 pre-alokace založená na rezervaci, 12 e-mailů

30. bře – 5. dub

Mingming Cao řekl:

Ext3 prealokace teď není k dispozici. Tohle je první prototyp ext3 prealokace založené na rezervaci – podle nápadů Andrewa a Teda. Ta implementace není kompletní, ale rád bych znal váš názor na současný design. V této verzi prototypu už mám:

1. základní rezervační struktury a operace
2. ext3 blokovou alokaci založenou na rezervaci
3. a alokace rezervačních oken
4. blokovou alokaci, je-li fs rezervace vypnuta

Jednotlivé body pak Mingming rozebral podrobněji a Andrew Morton odpověděl:

Myslím, že to směřuje správnou cestou.

Kromě jiného pak Andrew ještě navrhl přidat parametr pro „mount“, který by tu funkčnost povolil (s výchozí hodnotou „off“ (vypnuto)). Mingming poslal upravený patch a ještě se mluvilo o různých technických záležitostech.

Správa SCSI ovladače atp870u, 2 e-mailů

4. dub – 7. dub

Stuart Longland se zeptal, kdo je správcem ovladače atp870u SCSI, protože nedávno získal PCI ACARD SCSI kartu a ta vypisuje při použití chyby. Marcelo Tosatti odpověděl:

Oficiálně to vlastně nikdo nespravuje. James Bottomley a Doug Ledford udělali několik oprav pro v2.6, takže možná kontaktuj přímo je.

Správcovství BTTV, 3 e-mailů

5. dub

Luis Miguel napsal: *Mám údaje o nové kartě založené na bttv 878. Myslím, že by měla být zařazena do databáze karet, aby mohla začít fungovat. Kdo teď ten ovladač spravuje?*

Gerd Knorr odpověděl, že právě on.

Porušení GPL firmou Tritton Technologies, 3 e-maily

7. dub

Roy Franz napsal:

Narazil jsem na prodejce, který prodává zařízení využívající upravený linuxový kernel, a který odmítá poskytnout zdrojový kód upraveného jádra. Tvrdí, že na jejich změny se GPL nevztahuje. Nezdá se, že by používali moduly. Nevím, co s tím dál, takže na to upozorňuji vývojáře jádra. Jedná se o Tritton Technologies NAS120. (Nabízejí i verzi s funkcí routeru, která je založena stejné desce.) Ta deska je založena na procesoru Toshiba TX39 (MIPS) a má na sobě ethernetový chip Realtek. Běží na tom kernel verze 2.4.16. Viz: <http://www.trittontechologies.com/products.html> [1]

Vyrábí to zjevně mct.com.tw: <http://www.mct.com.tw/prod/sa-100.html> [2], protože některé soubory to tak označují.

Několik dalších prodejců prodává podobné verze téhož.

http://www.iogear.com/main.php?loc=product&product_id=645 [3] a

http://www.claxan.ch/de/prod_det.asp?PRODID=CL-SA110&TOPNAVID=-1 [4] (claxan nabízí nějaký zdrojový kód ke stažení, ale ne jádro. Jejich zákazník je kontaktoval a i oni odmítají dát zdrojové kódy jádra k dispozici.) Následuje odpověď z Trittonu, ve které říkají, že upravené zdrojové kódy jádra nedají k dispozici. Následovalo to po několika vyměněných emailech, ve kterých jsem velmi srozumitelně říkal, že mám zájem o zdrojové kódy jádra.

Citovaný email od technické podpory:

Dříve jsem řekl, že jádro bude součástí balení. Ačkoliv to i nadále platí, buďte srozuměn s tím, že úpravy provedené v jádru tam obsaženy nebudou. Kvůli dvěma důvodům:

1. ty úpravy nespádají pod GPL a
2. vlastní je Toshiba.

Matthias Urlichs odpověděl:

Upozornil bych je na to, že se jim dostane velkého množství negativní reklamy.

A Erik Andersen dodal: Přidal jsem NAS120 do BusyBox Hall of Shame...

LISP interpret v kernelu, 84 e-mailů

3. dub – 9. dub

Tohle byla jedna z těch diskuzí typu „to by bylo zvláštní ale super“, „ne to by tedy nebylo“. Sergej Lozovsky ji začal docela nevinně, když se zeptal, jak řešit přetečení haldy v jádře. Pak se ukázalo, že jde o toto: [Dal jsem do jádra LISP interpret – http://vx.e.quercitron.com](http://vx.e.quercitron.com) [5] – [Funguje to, ale využívá hodně paměti haldy. Není to tak jednoduché přepsat, ale budu pátrat, proč to využívá tolik té paměti...](#)

Timothy Miller, který si v tomto vlákně nebral servítky, se dožadoval důvodu, který by ospravedlňoval přidávání interpretu Lispu do kernelu, a Sergej vysvětlil, že to bylo z bezpečnostních důvodů. Jednak to umožní systémovým administrátorům nakonfigurovat bezpečnostní pravidla systému bez překompilování jádra nebo psaní kódu v nízkoúrovňovém C; a zadruhé to ochrání systém před chybami v těchto bezpečnostních pravidlech, které se tam mohly dostat díky administrátorovi systému během konfigurace.

Sergej pokračoval: [LISP interpret je LISP Virtual Machine \(jako Java VM\). Takže všechny chyby jsou uvnitř a neovlivní jádro. I vážné chyby v tomto modulu interpretu LISPU mohou způsobit pouze pád uživatelských aplikací \(kromě toho přetečení haldy – o které mi jde\). Chybová hláška LISPU bude zaznamenána v logu jádra.](#)

Následovala poměrně dlouhá diskuze, ve které lidi především předkládali jednu stížnost za druhou a Sergej se je všechny snažil vysvětlit. Jeden z argumentů byl, že celá ta věc patří do uživatelského

prostoru a ne do kernelu. John Stoffel řekl, že by to celé muselo být přepracováno, aby byla alespoň malá naděje, že se to v jádře uchytlí. Poukázal na všechny ochrany, které Sergej implementoval, aby zabránil chybám LISPu v ohrožení jádra, jako na důvod, proč ta věc patří do uživatelského prostoru. V uživatelském prostoru by těchto složitých ochran nebylo třeba. Sergej na to odpověděl tím, že se pouze snaží rozšířit tradiční unixový bezpečnostní model přirozenou cestou. Kdyby jeho interpret LISPu patřil do uživatelského prostoru, pak by se dalo stejně argumentovat pro přesunutí kontroly oprávnění roota nebo přístupových práv k souborům do uživatelského prostoru.

Další častou připomínkou byla prostě velikost té obludy. Timothy to vyjádřil v jednom ze svých příspěvků: **LISP VM je velká, obrovská, nafouknutá. ... *CHRAPTÍM* *KAŠLU* *PRSKÁM* *DUSÍM SE* ... věc, která BY NIKDY NEMĚLA BÝT v jádře.** Sergej reagoval, že velikost je pouhých 100 K. Neimplementoval kompletní Common LISP interpret, ale pouze malou část. Poznamenal, že se mu nepodařilo najít žádný virtuální stroj, který by se vešel do menšího prostoru. Robin Rosenberg připomněl LeJOS [6] pro Lego Mindstorm, který se vejde do 32 K a je licencován Mozilla licenci.

Dalším námitky byly zaměřené na výběr jazyka LISP a že, jak se opět nechal slyšet Timothy, systémoví administrátoři by nikdy nepsali bezpečnostní pravidla v LISPu, ale spíše v něčem podobném unixovému shellu. **Další možností by bylo vyvinout sadu nástrojů, které by kompilovaly pravidla napsaná v C do modulů dynamicky natahovaných do jádra :).** Process kompilace by byl pro uživatele transparentní, protože nástroj „vložit toto pravidlo“ by kompilaci prováděl pomocí GCC (pokud by v keši již nebyl aktuální .ko, atd.).

Celkově poskytl Sergej zajímavé odpovědi na všechny námitky, ale nevypadalo to, že by někoho přesvědčil o tom důležitém, totiž jestli by taková věc měla v kernelu být.

Obejití bufferové keše, 32 e-mailů

6. dub – 10. dub

Andy Isaacson napsal: **Vypadá to, že v moderním Linuxu je správným způsobem, jak obejít bufferovou keš při zápisu na blokové zařízení, otevření blokového zařízení s O_DIRECT.** To například uživateli umožňuje jednodušeji zajistit realokaci jediného sektoru na vadném IDE disku (bez přepisování čehokoliv jiného než toho 1k „sektorového páru“ s chybou). dd(1) se pro tento účel hodí, ale postrádá způsob, který by zaručil použití O_DIRECT. Přiložený patch přidává parametr „conv=direct“, který to umožňuje.

Andrew Morton projevil velký zájem a požádal Andyho, aby provedl nějaké další pročišťující úpravy, když už v té oblasti pracuje. Andy však narazil na nějaké problémy a řekl: **V 2.4.25 se mi vůbec nedaří O_DIRECT rozchodit na ext3 – open(O_DIRECT) je úspěšné, ale zápis vrátí EINVAL.**

Andrew odpověděl: **ext3 O_DIRECT v 2.4 jádrech nepodporuje.** Kdysi jsem napsal patch a myslím, že je v 2.4-aa kernelech. ext3 podporuje O_DIRECT v jádrech 2.6. Dost dalších filesystémů rovněž.

Posléze začalo být zřejmé, že Andy má v plánu implementovat podporu O_DIRECT pouze pro zápisy na disk, protože předpokládá, že pro čtení z disku k tomu není důvod. Ale Andrew prohlásil, že pro čtení i zápis existují stejné důvody, totiž že to šetří užití CPU a zabraňuje přílišnému používání bufferové keše. Andy souhlasil, že využití CPU by bylo s podporou O_DIRECT pro čtení nižší, ale ten zisk by byl tak mizivý, že to nestojí za implementaci.

Pokračoval, že to považuje za práci jádra, aby přišlo na to, že „hele, vypadá to jako posloupné [streaming] čtení – ušetříme tedy bufferovou keš, protože se nebudeme pokoušet zachytit 20GB na systému s 512MB“. Jestli chceš říct, že lidi kolem jádra to vzdali a řídí se teď pravidlem „pokud nechceš vyhodit všechno ostatní kvůli posloupným datům, musíš použít O_DIRECT“, pak jsem tedy zklamaný.

A uzavřel:

Copak by otevření jak `if=`, tak `of=` s `O_DIRECT` neproměnilo `dd` v synchronní kopírku? To by bylo fakt na houby třeba v případě „`dd if=/dev/hda1 of=/dev/hdc1`“. Když dělá bufferová keš `readahead` [načítání dopředu], zvládne ten příkaz nějakých 40MB/s čtení a 40MB/s psaní; se synchronním čtením i psáním by to kleslo na 40MB/s čtení+psaní, pokud by byly velikosti bloků dostatečné na to, aby se vyplatilo zpomalení kvůli hledání [seek overhead]. Když je `O_DIRECT` pouze na `of=`, myslím, že se můžeme dostat zpět na 40MB/s+40MB/s. Tvrdím, že `O_DIRECT` na `of=` je důležité, protože vyškrábnout malé velikosti IDE bloků bez toho prostě nelze. Nevidím však žádnou podobnou výhodu použití `O_DIRECT` na straně `if=`.

Andrewa nic z toho nepřesvědčilo a řekl naopak:

Jestli chceš něco na vyškrábnutí bloků, napiš si to. Jestli chceš do `dd` přidat podporu `O_DIRECT`, pak by to mělo být implementováno správně, a to znamená implementováno jak pro zápis, tak pro čtení. Uživatel by měl vlastně mít možnost určit použití `read-O_DIRECT` a `write-O_DIRECT` nezávisle – třeba proto, že zdrojový a cílový filesystém nemusí oba `O_DIRECT` podporovat.

Andy neměl pocit, že by to byl moc dobrý nápad, ale i tak napsal a poslal patch podle Andrewových požadavků. Někdy v tu dobu se ozval také Paul Eggert se svým patchem, který řešil ještě několik dalších věcí. Tento patch byl pak akceptován do 'coreutils'.

Ovladač adm8211 802.11b pro 2.6, 1 e-mail

10. dub

Michael Wu napsal:

Tady máte ovladač pro chipset adm8211 802.11b.

<http://aluminum.sourmilk.net/adm8211/adm8211-20040411.tar.bz2> [7]

Je pouze pro jádra 2.6.x a ještě není tak docela hotov. Měl by fungovat v infrastrukturních a monitorovacích režimech, ale `adhoc` a `WEP` ještě nejdou. Běžná bezdrátová síť by měla fungovat dobře. Díky Jouni Malinenovi, že s tím ovladačem začal, Jerrittu Collordovi, že mi o tom řekl, a Davidu Youngovi za jeho NetBSD ovladač, který jsem použil jako referenci.

Skript pro patchování jádra – ketchup – ve verzi 0.5, 2 e-maily

11. dub

Matt Mackall napsal: `ketchup` je skript, který automaticky patchuje mezi verzemi jádra, přičemž stahuje a ukládá patche podle potřeby. Automaticky také určuje poslední verze několika stromů.

A pokračoval:

Novinky v této verzi na žádost uživatelů:

- změna jména `kpatchup` → `ketchup`
- automatické ověření podpisů
- funguje se staršími verzemi Pythonu

V současné době je výchozím chováním snaha použít `gpg`, což znamená, že musíte mít `gpg` nainstalované a příslušné klíče uložené ve vašem keyringu. Zbavit se toho můžete pomocí parametrů `-g` nebo `-no-gpg`. Najdete na: <http://selenic.com/ketchup/ketchup-0.5> [8]

Matt přiložil příklad použití `ketchup`:

```
$ ketchup 2.6-mm
2.6.3-rc1-mm1 -> 2.6.5-mm4
Applying 2.6.3-rc1-mm1.bz2 -R
Applying patch-2.6.3-rc1.bz2 -R
Applying patch-2.6.3.bz2
```

```
Applying patch-2.6.4.bz2
Applying patch-2.6.5.bz2
Downloading 2.6.5-mm4.bz2
Downloading 2.6.5-mm4.bz2.sign
Verifying signature...
gpg: Signature made Sat Apr 10 21:55:36 2004 CDT using DSA key ID 517D0F0E
gpg: Good signature from "Linux Kernel Archives Verification Key
<ftpadmin@kernel.org>"
gpg:          aka "Linux Kernel Archives Verification Key
<ftpadmin@kernel.org>"
owner.
gpg: WARNING: This key is not certified with a trusted signature!
gpg:          There is no indication that the signature belongs to the
Primary key fingerprint: C75D C40A 11D7 AF88 9981 ED5B C86B A06A 517D 0F0E
Applying 2.6.5-mm4.bz2
```

Přidání podpory SysFS do DVB subsystému, 3 e-maily

12. dub

V rámci pokračující snahy o zprovoznění všeho se SysFS pracoval Michael Hunold na přidání podpory SysFS do subsystému DVB. Nebyl si však jistý, jestli na to jde správně. Podařilo se mu vytvořit adresář `/sys/class/dvb/`, ale měl problémy s vytvářením podadresářů. Zeptal se proto na nějakou dokumentaci. Stephen Hemminger vysvětlil: [Layout adresáře sysfs je logickou reprezentací příslušných datových struktur v jádře. Každý adresář \(kromě skupin atributů\) je 1:1 ke kobjektu.](#)

Greg KH dodal:

[Adresář nelze vytvořit pouhým přidáním '/' ke jménu třídy zařízení. Pro adresář bude třeba vytvořit kobjekt a v tomto kobjektu vytvořit atributy, jako v případě síťování. Jo, je to trochu otrava, ale vytváření podadresářů jednoduchým způsobem je na seznamu pro 2.7. Jestli chceš všechno urychlit, což by ti ušetřilo práci s přípravou pravidel životnosti tvých dvb ovladačů, můžeš implementovat rozhraní `class_simple` – stejně to teď dělá mnoho dalších ovladačových subsystémů. Ve 2.7 by se to pak převedlo na správný model konverze ovladačů.](#)

Crypto CRC32C modul, 11 e-mailů

19. dub – 27. dub

Clay Haapala napsal: [Tento patch oproti 2.6.6-rc1-bk implementuje CRC32Calgoritmus \[9\] jako formu výtažku. Je to provedené jako wrapper pro `libcrc32c` a dostupné v samostatném patchi. Crypto CRC32C modul bude využíván ovladačem `iscsi-sfnet`.](#)

David S. Miller a James Morris patch schválili; Clay udělal ještě diff oproti oficiálnímu stromu 2.6.6-rc1 a po vyřešení několika malých připomínek jej David začlenil.

Odstraňování IDE modulů; IDE a libata, 11 e-mailů

21. dub – 26. dub

Bartłomiej Zolnierkiewicz poslal patch, který znemožnil odstraňovat `[unload]` moduly ovladačů starších IDE chipsetů. Spočívalo to v pouhém vyřazení volání `module_exit()` z ovladačů. Erik Andersen se zeptal: [Jen ze zvědavosti, co by bylo potřeba, aby bylo odstraňování všech IDE modulů bezpečné na systému s root filesystémem SCSI?](#)

Bartłomiej řekl, že na tom nezáleží, protože uživatel by i tak mohl nakonec odstranit moduly, které by byly v tu dobu ještě používány.

Erik Mouw napsal, že IDE moduly již úspěšně odstraňuje roky a zeptal se: V čem je IDE natolik odlišné od SCSI, že ty IDE host ovladače nemůžeme odstraňovat? Bartłomiej odpověděl, že to má na svědomí neexistence referenčního počítání a zároveň nedostatečné zamykání.

Erik M. pokračoval: A plánujete odstraňování modulů opravit v současném kódu nebo by bylo jednodušší napsat nový ovladač založený na libata (za předpokladu, že v libata už to je v pořádku)? Pokud jsem dobře porozuměl poslední aktualizaci libata od Jeffa, mělo by to jít už velmi brzy, je to tak? Bartłomiej odpověděl: Chystám se to opravit, ale aby to mohlo být uděláno správně, tak budou nutné zásadní změny v IDE kódu. A dodal:

Zdá se mi, že existuje všeobecný omyl co se libata týče: v dohledné budoucnosti se nechystá nahradit IDE ovladače. Chci přepsat+sloučit současný IDE kód s libata během 2.7 (a ano, stará pojmenování a uspořádání budou zachována!). Doufám, že se nikdo nepustí do přepisování existujících IDE ovladačů pro libata a nebude se je snažit prosadit do hlavního jádra → to by znamenalo daleko větší problémy se správou než OSS+ALSA. Nicméně psaní *nových* libata ovladačů pro „exotický“ PATA hardware je OK.

Opravy SiS ovladače, 2 e-maily

25. dub – 26. dub

Jeff Garzik napsal:

Připojen je patch plné verze SiS ovladače pro 2.6.x připravený k testování. Měl by řešit problém, při kterém nebyly SATA jednotky během prohlížení systému nalezeny. Jakékoliv reakce, včetně „nic jsi nezvorál“ bych moc ocenil, protože nemám v provozu žádný testovací SiS hardware.

Robin Parker to vyzkoušel a napsal: Co říct? Funguje to jedna báseň.

Ovladač AHCI; otevřené hardwarové specifikace, 2 e-maily

25. dub

Jeff Garzik napsal:

Tak ATA se konečně probudila do moderních časů. Lidi, kteří pracují s ATA, vědí, že rozhraní host řadiče byla vždy pomalá, často vyžadovala 8 a více IO zápisů na jeden příkaz. Lidi od SCSI a ethernetových řadičů už krotí očima několik let. Současná vlna SATA řadičů udělala velký pokrok... do současnosti. Buď DMA rings nebo fronty s hloubkou kolem 32 začínají být standardem. Všechno si to dobře rozumí s připojováním za chodu [hotplug] a cesty jsou vysoce optimalizované a rychlé. Samozřejmě, jak už je nešťastným zvykem v oboru ukládání dat, všechny tyto řadiče jsou pokryté smlouvami o neprozrazení (NDA). Výjimkou je ICH6 od Intelu, který odpovídá plně otevřené specifikaci AHCI. Abyste mohli podporovat takový řadič, potřebujete pouze PCI ID a následující specifikace:

AHCI: http://www.intel.com/technology/serialata/pdf/rev1_0.pdf [10]

SATA 1: <http://www.serialata.org/collateral/zipdownloads/serialata10a.ZIP> [11]

SATA 2: http://www.serialata.org/collateral/pdf/S2Ext_1_1_Gold.pdf [12]

ATA 1: http://www.t13.org/docs2004/d1532v1r4b_ATA-ATAPI-7.pdf [13]

ATA 2: http://www.t13.org/docs2004/d1532v1r4b_ATA-ATAPI-7.pdf [14]

ATA 3: http://www.t13.org/docs2004/d1532v1r4b_ATA-ATAPI-7.pdf [15]

Kromě toho, že je to otevřená architektura, tak to má všechno, co byste očekávali od moderního řadiče: všude podpora 64 bitového DMA (žádné přepínání banků), podpora front, triviální podpora PIO a ATAPI, přístup k nízkourovňovému SATA FIS a ještě více. Takže bravo lidem od AHCI (hlavně z Intelu) za to, že udělali slušný otevřený řadič. Vždy je mi milejší pracovat na řadičích pro slušný hardware, jehož hardwarové specifikace jsou otevřené a přístupné.

Sven Köhler odpověděl: Vypadá to, že SiS se chystá také používat AHCI:
<http://www.sis.com/products/chipsets/oa/athlon64/756Presentation.pdf> [16].

LinuxAnt lže o své licenci, 2 e-maily

26. dub

Jon DeVree oznámil:

Někdo by se měl podívat na řetězec `MODULE_LICENSE`, který vracejí ovladače HSF modemů od LinuxAntu. <http://www.linuxant.com/drivers/hsf/full/downloads.php> [17]

Vynalézavě do něj vložili znak `\0`. `MODULE_LICENSE(„GPL\0for files in the \”GPL\” directory; for others, only LICENSE file applies“)`; (Česky: „GPL\0pro soubory v adresáři \”GPL\”; pro ostatní platí soubor LICENSE“). Když spustíte `modinfo -F license` na zkompilevaném ovladači, ukáže vám to GPL díky jejich vynalézavému null znaku. Skutečná licence většiny souborů NENÍ GPL.

Chris Friesen poznamenal: Ummm... to je nepěkné. Možná bychom měli pro `modinfo` ukládat velikost licenčního řetězce, aby se nezastavil na první nule.

Opravy a aktualizace ReiserFS verze 3, 22 e-mailů

6. dub – 16. dub

Chris Mason napsal:

Sadu experimentálních patchů pro reiserfs v3 můžete stahovat odsud:

<ftp://ftp.suse.com/pub/people/mason/patches/reiserfs/2.6.5> [18].

Vzhledem k tomu, že některé z těchto již v `-mm` jsou a jiné ne, jsou připraveny dva soubory: `series.linus` obsahuje patche potřebné pro hlavní strom 2.6.5, `series.mm` pak patche pro 2.6.5-mm1. Většina je od Jeffa Mahoneyho a ode mě. Jsou mezi nimi:

```
opravy chyb
optimalizace logování
podpora data=ordered
xattr
acl
kvóty
chybové hlášky s názvy zařízení (založeno na Olegově patchi pro 2.4)
vylepšení alokátoru bloků
```

ACL a `xattr` od Jeffa Mahoneyho pro reiserfs v3 už jsou nějakou dobu používány v 2.4 a 2.6 kernelech od SUSE. Mnohokrát jsem je poslal do namesys ke kontrole, ale Hans ten kód vůbec nechce vzít na vědomí. Minulý měsíc jsem se znovu pokusil s ním o tom kódu mluvit, ale úplně mě ignoruje. Jeho tehdejší argumentem bylo, že nechce nové funkce ve verzi 3. Ta implementace ale nikterak nezmění formát disku (`xattr` jsou uloženy jako běžné soubory ve skrytém adresáři). Mám zato, že reiserfs tyto funkce potřebuje, aby byl v jádře i nadále aktuální, a proto je posílám k začlenění do `-mm`. Pro `-mm` sérii Andrewovi posílám následující patche ze `series.mm`:

```
reiserfs-end-trans-bkl
reiserfs-acl-mknod.diff
reiserfs-xattrs-04
reiserfs-acl-02
```

```
reiserfs-trusted-02
reiserfs-selinux-02
reiserfs-xattr-locking-02
reiserfs-quota
permission-reiserfs
reiserfs-warning
```

(což znamená vše kromě kódu nového alokátoru bloků)

Následovala kratší diskuze ohledně alokátoru bloků a problémech, se kterými se současná i nová implementace potýká. O něco dále nabídl Chris Wright patch, ale Chris Mason odpověděl: **Na počátek patche budeš muset přidat pár slov o tom, že Hansovi přenecháváš copyright.**

Ale Hans Reiser reagoval s tím, že taková formulace je **velmi nepřesná. Tento dokument:**

www.namesys.com/legalese.html [19] to vysvětluje lépe. Budeš-li tak laskav a povolíš mi licencovat příspěvek do reiserfs třetím stranám (kromě běžné GPL), budu za to rád, protože mi to občas umožňuje získat velmi potřebné prostředky pro vývoj reiser4.

Podpora tabulky atributů stránek v 2.6, 20 e-mailů

12. dub – 21. dub

Terence Ripperda napsal:

Před dost dlouhou dobou jsem poslal email o přidání podpory tabulky atributů stránek (PAT – Page Attribute Table) (http://www.kerneltraffic.org/kernel-traffic/kt20030616_219.html#3 [20]).

Tehdy šlo o to, abychom měli možnost označit remapované I/O stránky jako Write Combined v případě, že by pro ten účel došly MTRR. To bylo především kvůli apertuře AGP a framebufferu. Jak se objevují PCI Express systémy, potřeby se trochu mění. PCI Express nemá centralizovanou aperturu jako AGP, kterou by bylo možno označit WC. Namísto toho musí být jednotlivé systémové stránky paměti označeny WC přes PAT v tabulkách stránek. To výrazně zvyšuje potřebu podpory PAT, aby byla zachována vysoká úroveň výkonu na PCI Express systémech.

Měl jsem pocit, že nejlepší bude se trochu věnovat původním reakcím ještě než se obrátím zpět na LKML. Mluvil jsem o tom s Andi Kleenem, který navrhl, abych se nejprve zaměřil na jednodušší oblasti I/O a pak se vrátil k hlavní paměti po té, co to bude hotovo a budu mít od lidí reakce.

Pracoval jsem na mechanismu (cachemap), který sleduje, jaký druh kešování je pro určitou oblast paměti právě namapován. Při namapování nové oblasti paměti se zeptáme cachemap, jestli je druh kešování nové oblasti stejný jako starý druh (nebo je s ním kompatibilní). Uspěje-li dotaz na cachemap, může být nová I/O oblast bezpečně namapována, jinak ne.

Nejdříve jsem se soustředil na ioremap, aby byly dotazy na cachemap prováděny pouze z ioremap. Také jsem přidal kód, který nechává MTRR kód volat kód cachemap (v tomto případě jde spíše o zprávu než o dotaz – značí, že mapování už bylo provedeno). Provedl jsem několik testů na systémech, které tu mám k dispozici, a všechno se zdá fungovat docela dobře.

Ten současný patch obsahuje původní podporu PAT a ten nový mechanismus cachemap. Upozorňuji, že cachemap mechanismus ve skutečnosti žádné atributy kešování nemění – pouze atributy sleduje a testuje oblasti. Myslím, že finální podoba by mohla být taková, že ovladače by používaly běžné mechanismy ioremap/change_page_attr/remap_page_range jako doposud a tyto mechanismy by používaly cachemap, aby bylo jisté, že nemáme žádné konflikty. Vítám všechny připomínky k tomu, jak by jednotlivé detaily měly fungovat, a jaké je třeba provést změny.

Andi Kleen napsal, že Terencova práce vypadá pro začátek dobře a bylo by dobré to testovat. I tak je však nutné všechno pročistit. Pak dodal:

Co se rozhraní týče, stále si myslím, že by bylo čistší to prostě volat z `change_page_attr()`. Pak by další uživatelé museli volat jen jednu jedinou funkci. Ale to se dá snadno změnit.

Aby to bylo opravdu užitečné, myslím, že bychom potřebovali `ioremap_wrcomb()` a podporu ve funkci `bus/pci mmap` (PCI vrstva už pro tohle má `ioctl`, pouze to na i386 ignorovali). Pak by to mohl začít využívat X server.

Bez uživatelů asi nebude testování moc dobré, ale než to bude začleněno, musí se tomu nějakého skutečného testování dostat. Možná by to mohlo být prostě pověšeno na nějaký AGP a DRM ovladač. Pak by to lidi aspoň začali testovat.

Eric W. Biederman ještě diskutoval s Andim o přínosu, který by to mělo pro systémy s velkým objemem paměti, zatímco Terence posílal aktualizované patche.

SATA bude ve 2.4.27, 27 e-mailů

15. dub – 17. dub

Serial ATA, ačkoliv relativně nové, jde kupředu velmi rychle. Tento týden Marcelo Tosatti napsal: **Jeff Garzik mi poslal aktualizované SATA pro začlenění do 2.4.x. Hodně nových počítačů už se prodává pouze se SATA disky a bylo by dost špatné mít „stabilní“ sérii bez podpory takového rozšířeného standardu.**

Tohle bude poslední funkce přidaná do 2.4.x – a to pouze proto, že je nezbytná. Někdo proti?

Stefan Smietanowski řekl, že je to tak dobře, protože 2.4 bude lidmi i nadále používán, zatímco 2.6 se bude stabilizovat a stávat standardem.

Na jiném místě Andre Hedrick Marcelovi spílal za to, že měl SATA začlenit již dávno. Marcelo odpověděl: **Jeff poslal SATA 1.0 již dříve, ale radši jsem to ještě chvíli pozdržel (abych si byl jistý, že je to stabilní). Teď to vypadá, že Jeff už v to má důvěru.**

Následovalo další obviňování, dokud vlákno nevymizelo. Marcelova rozhodnutí ohledně 2.4 budou zjevně kontroverzní až do chvíle, než se plně přejde na 2.6.

Automatický hledač chyb odhalil v jádře několik dř, 19 e-mailů

16. dub – 20. dub

Ken Ashcraft poslal dlouhý seznam potenciálních chyb v jádře. Doprovodil to vysvětlením:

Pracuji ve společnosti, která se jmenuje Coverity. Děláme na derivátu MC Checkeru vytvořeném ve Stanfordu pro profesionální nasazení. Napsal jsem statický analytický checker, který v jádře vyhledává místa, kde kernel dostane od uživatele skalár a pak jej použije bez provedení kontroly hranic. Například když ovladač načte integer od uživatele a pak ten integer použije jako index pole nebo parametr délky pro `mempcy`, uživatel může způsobit přetečení bufferu.

Příložené chyby jsou zhruba seřazeny podle závažnosti. Posledních 12 chyb je celkem nicotných, protože jsou buď chráněny kontrolou `capable()` nebo jde pouze o 8-bitový skalár. Za malou chybu považuji i předání nezkontrolované hodnoty do `kmalloc()`. Uvědomuji si, že `kmalloc()` selže, bude-li požádán o více než 128k, ale i tak možná není vhodné nechávat uživatele alokovat tolik paměti. Všechny tyto chyby jsou ve zprávě označeny [MINOR] a/nebo [CAPABLE].

Chris Wright několik chyb potvrdil a poslal patche; některé šly přes Jeffa Garzika rovnou do 2.4 a 2.6 kernelů, jiné byly nejprve upraveny lidmi jako Andrea Arcangeli. Několik dalších chyb také Chris před ukončením vlákna vyvrátil.

Dokumentace ke kompilaci externích modulů, 4 e-mailů

20. dub – 21. dub

Sam Ravnborg poslal hrubé a nekompletní HOWTO ke kompilaci externích modulů:

Kompilace externích modulů

kbuidl poskytuje možnost kompilovat externí moduly za předpokladu, že existuje již zkompilevané jádro s plnými zdrojovými kódy. Některé z cílů dostupných pro kompilaci jádra jsou k dispozici při kompilaci externího modulu.

Kompilace modulu

Příkaz vypadá takto:

```
make -C <path to kernel src> M=$PWD
```

Aby byl zmíněný příkaz úspěšný, musel být kernel zkompileován s povolenými moduly. Právě kompilované moduly nainstalujete pomocí:

```
make -C <path to kernel src> M=$PWD modules_install
```

Další komplexnější příklady přijdou později. Uvedené příkazy by měly stačit ve většině případů.

Dostupné cíle:

```
make -C $KDIR M=$PWD
```

Stejně i při zadání 'modules'. Viz popis cíle modules níže.

```
make -C $KDIR M=$PWD modules
```

Zkompiluje moduly umístěné v aktuálním adresáři. Všechny výstupní soubory budou ve stejném adresáři jako zdrojový kód modulů. Neprovádí se žádný pokus o aktualizaci zdrojů jádra a předpokládá se, že make kernelu proběhl úspěšně.

```
make -C $KDIR M=$PWD modules_install
```

Instalace externích modulů

```
make -C $KDIR M=$PWD clean
```

Odstranění všech vygenerovaných souborů v aktuálním adresáři

Dostupné volby:

```
make -C $KDIR
```

Používá se k zadání místa, kde hledat zdrojáky jádra. **\$KDIR** představuje adresář se zdrojovými kódy kernelu. Make do zadaného adresáře při spuštění přepne.

```
make -C $KDIR M=$PWD
```

M= se používá k oznámení kbuidlu, že kompilujeme externí modul. Hodnota parametru M= je adresář, kde je externí modul umístěn. Při kompilaci externího modulu je k dispozici pouze omezená sada obvyklých cílů.

```
make -C $KDIR SUBDIRS=$PWD
```

Totéž co M=. SUBDIRS= je ponecháno kvůli zpětné kompatibilitě.

```
make -C $KDIR M=$PWD help
```

help vypíše dostupné cíle při kompilaci externího modulu.

Pokročilejší příklad

Tento příklad znázorňuje případ, kdy se distribuce moudře rozhodla oddělit zdrojové kódy jádra a výstupní soubory:

Zdrojové kódy jádra:

```
/usr/src/linux-<kernel-version>/
```

Výstup kompilace jádra, včetně .config:

```
/lib/modules/linux-<kernel-version>/build/
```

Externí moduly, které chceme kompilovat:

```
/home/user/module/src/
```

Moduly umístěné ve zmíněném adresáři zkompilujete následujícím příkazem:

```
cd /home/user/module/src
make -C /usr/src/linux-<kernel-version> \
O=/lib/modules/linux-<kernel-version>/build \
M=$PWD
```

Pak je nainstalujte pomocí příkazu:

```
make -C /usr/src/linux-<kernel-version> \
O=/lib/modules/linux-<kernel-version>/build \
M=$PWD modules_install
```

Výše jsou uvedeny poněkud dlouhé příkazy a následující část ukazuje, jak to všechno zjednodušit.

Triky pro zjednodušení

... Tohle je potřeba přepracovat...

Příkazová řádka s make a několika parametry je únavně dlouhá a náchylná k chybám. Následuje malý trik, který umožňuje kompilaci modulu s použitím jediného příkazu 'make'.

Vytvořte makefile pojmenovaný 'Makefile' s následujícím obsahem:

```
---> Makefile:
all:
    $(MAKE) -C /home/sam/src/kernel/v2.6 M=$(PWD) \
            $(filter-out all,$(MAKECMDGOALS))
obj-m := module.o
---> Konec Makefile
```

Při spuštění si make všimne pravidla all: a zavolá make znovu se správnými parametry. Je-li vyvíjen ovladač, který má za cíl začlenění do hlavního jádra, je lepší vložit pravidlo all: do Makefile pojmenovaného +makefile (menší velké m) - asi takto:

```
---> makefile
all:
```

```
$(MAKE) -f Makefile -C /home/sam/src/kernel/v2.6 \  
M=$(PWD) $(MAKECMDGOALS)
```

---> Konec makefile

Makefile pro kbuild bude obsahovat jedinou věc:

```
---> Makefile:
```

```
obj-m := module.o
```

```
---> Konec Makefile
```

Při spuštění bude make hledat makefile pojmenovaný makefile před souborem pojmenovaným Makefile. Takže make použije soubor s menším velkým 'm'.

Příprava jádra pro kompilaci externích modulů

Při kompilaci externích modulů se očekává, že je na to jádro připraveno. To zahrnuje přítomnost určitých binárek, konfiguraci jádra a symlink na include/asm. K tomuto účelu slouží pohodlný cíl:

```
make modules_prepare
```

U typické distribuce by to vypadalo asi takto:

```
make modules_prepare O=/lib/modules/linux-<kernel version>/build
```

ZBÝVÁ UDĚLAT: Naplnit následující kapitoly

Verze modulů

Lokální include soubory

```
CLFAGS := include ...
```

Pouze binární .o soubory

Použijte `_shipped` ...

Odkazy

- [1] <http://www.trittontechologies.com/products.html>
- [2] <http://www.mct.com.tw/prod/sa-100.html>
- [3] http://www.iogear.com/main.php?loc=product&product_id=645
- [4] http://www.claxan.ch/de/prod_det.asp?PRODID=CL-SA110&TOPNAVID=-1
- [5] <http://vxe.quercitron.com>
- [6] <http://www.lejos.org>
- [7] <http://aluminum.sourmilk.net/adm8211/adm8211-20040411.tar.bz2>
- [8] <http://selenic.com/ketchup/ketchup-0.5>
- [9] <http://www.haifa.il.ibm.com/satran/ips/Vince-Luben-crc32c-01.pdf>
- [10] <http://www.intel.com/technology/serialata/pdf/rev1.0.pdf>
- [11] <http://www.serialata.org/collateral/zipdownloads/serialata10a.ZIP>
- [12] http://www.serialata.org/collateral/pdf/S2Ext_1.1_Gold.pdf
- [13] <http://www.t13.org/docs2004/d1532v1r4b%20ATA-ATAPI-7.pdf>
- [14] <http://www.t13.org/docs2004/d1532v1r4b%20ATA-ATAPI-7.pdf>
- [15] <http://www.t13.org/docs2004/d1532v1r4b%20ATA-ATAPI-7.pdf>
- [16] <http://www.sis.com/products/chipsets/oa/athlon64/756Presentation.pdf>
- [17] <http://www.linuxant.com/drivers/hsf/full/downloads.php>
- [18] <ftp://ftp.suse.com/pub/people/mason/patches/reiserfs/2.6.5>
- [19] <http://www.namesys.com/legalese.html>
- [20] http://www.kerneltraffic.org/kernel-traffic/kt20030616_219.html#3

Stickfish s. r. o. nabízí

Proč Linux?

Protože jde o mimořádně stabilní, kvalitní a výkonný operační systém. Protože díky přístupnosti zdrojových kódů prochází neustálým a nezávislým bezpečnostním auditem veřejnosti a gigantů typu IBM či SAP. Protože má výhodné licenční podmínky, díky nimž můžete ušetřit desítky až stovky tisíc korun. Potřebujete další důvody? Rádi zodpovíme vaše dotazy na adrese info@abclinuxu.cz.

Proč právě my?

Protože uděláme maximum pro nalezení a uskutečnění toho nejlepšího řešení. Protože naše produkty upravíme přesně na vaši míru. Protože náš zájem o vás nekončí proplacením faktury. Protože máme špičkové odborníky. Protože nás důvěra desítek tisíc návštěvníků našeho portálu zavazuje odvádět tu nejlepší práci.

Jaké máme zákazníky?

Zaměřujeme se na menší a střední firmy, státní správu a rozpočtové organizace. Mezi naše zákazníky patří:

Správa sportovních zařízení města Hradce Králové	Ateliér P.H.A. Praha
Výzkumný ústav lesního hospodářství Jíloviště	Picka Data Servis Praha
Cestovní kancelář www.motylek.com Praha	Ministerstvo vnitra Praha

Jaké nabízíme služby a řešení?

Provedeme vám komplexní analýzu možností nasazení GNU/Linuxu a dalších Open Source produktů ve vaší organizaci. Prověříme existující řešení bezpečnosti a v případě potřeby navrhne a realizujeme zlepšení. Na základě analýzy vytvoříme kompletní řešení pro zálohování dat včetně metodiky a zaškolení vašich zaměstnanců. Tento výčet není ani zdaleka kompletní, máte-li zde neuvedenou potřebu, obraťte se na nás a naši specialisté se vám budou věnovat.

A jaké produkty?

Naše serverové produkty jsou postaveny na linuxové distribuci Gentoo, což nám umožňuje vyladit jejich rychlost na maximum, zároveň nám však ponechá veškerou flexibilitu a přehlednost GNU/Linuxu. Všechny ceny jsou bez DPH a zahrnují instalaci a základní konfiguraci u zákazníka.

ABC Secure Wall

Základní kámen pro připojení počítačové sítě do internetu. Bezpečná hráz, která vás spolehlivě ochrání před útočníky. Používá prověřenou technologii netfilter. Individuální přizpůsobení. Cena již od 8 000 Kč.

ABC Speed Internet

Trik, jak výrazně zrychlit připojení k internetu a zároveň určit, kteří zaměstnanci a v jakém rozsahu jej smí využívat. Stabilita, bezúdržbovost, žádné platby za klientské stanice. Cena již od 10 000 Kč.

ABC Secure Mail

Konečně komplexní řešení, které zajistí veškeré firemní potřeby ohledně emailu a zároveň vás zbaví otravných spamů a nebezpečných virů. Cena již od 10 000 Kč.

Chcete vědět více?

Stickfish s. r. o.	Email: info@stickfish.cz , info@abclinuxu.cz
Řehořova 1039/54	Telefon: +420 777 993222
130 00 Praha 3	Fax: +420 222 731 202

Společnost je zapsána v obchodním rejstříku u Městského soudu v Praze, vložka C, spis 90069. IČ: 267 29 687, DIČ: 003-267 29 687. Bankovní spojení: 958810001/2400.

Zprávičky

1.6.2004

Vojtěch Hála

Petr Olšák, předseda CSTUGu, poslal do českých konferencí o TeXu a o Linuxu příspěvek popisující zkušenost, která svobodné licence brání šíření svobodného software. A těší se na reakce.

1.6.2004

Petr Krčmář

MythTV je zajímavý projekt, který si klade za cíl vytvořit kompletní balík software pro potřebu domácích set-top boxů (multimediálních počítačů se snadnou obsluhou).

1.6.2004

Petr Krčmář

Firma Lindows vyhrála v Nizozemí soudní spor s Microsoftem a může tedy v této zemi používat své původní obchodní jméno. Více o historii celé kauzy píše The Register.

1.6.2004

Martin Dvořák

Vadí vám rozdílný vzhled aplikací napsaných pro různé grafické toolkity? Pokud ano, bude vás možná zajímat projekt MetaTheme.

2.6.2004

Petr Krčmář

Spammeři začali využívat PGP signatury vykradené z legitimní pošty, aby jednoduše oklamali antispam filtry.

2.6.2004

Petr Krčmář

MandrakeSoft uvolnil betaverzi druhého Mandrakemove (live CD distribuce), který je založen na MDK 10.0 a obsahuje mimo jiné například jádro 2.6.3.

2.6.2004

Petr Krčmář

U Intelu se chystá vývoj nového BIOSu, který nahradí standardní, dvacet let starý, a dnes již vysloužilý model. Důležité ovšem je, že se jej zároveň chystá uvolnit pod CPL.

3.6.2004

tomasini

Výborný otevřený dopis, popisující problematiku softwarových patentů, zaslal českým médiím vývojář Hynek Hanke z www.freebsoft.org (Svobodný software pro nevidomé a zrakově postižené). Velmi doporučuji jeho přečtení. Je dobré poukázat, jak může být znemožněno vývojářům

pomoci zrakově postiženým. Možná právě na to média budou slyšet.

3.6.2004

Petr Krčmář

Sun Microsystems pravděpodobně uvolní Solaris pod některou z Open Source licencí. Chce se tak vyrovnat svým rivalům a vylepsit své pozice na trhu.

3.6.2004

Petr Krčmář

Nový kancelářský balík EOffice, jehož recenzi si můžete přečíst na NewsForge, je zajímavý především svou multiplatformností. Je totiž kompletně napsán v Javě.

3.6.2004

Petr Krčmář

Microsoft získal patent na dvojklik. Slashdot to komentoval slovy: „Takže tohle je to, na co se můžeme těšit v EU?“

4.6.2004

Petr Krčmář

Zajímá vás, jak to vypadá v zákulisí provozu velkého linuxového portálu? Přečtěte si zajímavý článek na NewsForge.

4.6.2004

Petr Krčmář

Vznikla nová vývojová větev linuxu, pojmenovaná wireless-2.6, která se zaměří na začleňování kódu pro bezdrátovou komunikaci, který se nedostane do stable větve.

4.6.2004

Petr Krčmář

Osm z deseti nejspolehlivějších hosting providerů používá na svém železe Linux nebo FreeBSD.

4.6.2004

Antonín Slejška

SUSE ČR končí. SUSE Linux 9.1 v české verzi stále není (a není jisté, zda bude). OPRAVA: Původní zprávička interpretovala informaci nesprávně; SUSE ČR nekončí, jeho aktivity pouze přebírá firma Novell.

4.6.2004

Martin Hassman

Pokud máte zájem o novinky ze světa Mozilly, přihlašte se k odběru zpravodaje Mozilla Links. 19. číslo zpravodaje tu bude cobydup.

4.6.2004

dantm

SUSE Linux 9.1 je k dispozici ke stažení zdarma na ftp.

- 5.6.2004** *Jan Hykel*
Vyšlo GTK+ a Glib verze 2.4.2. Jedná se o bug-fixní vydání. Stahovat můžete zde.
- 7.6.2004** *Václav Hlobil*
Vyšla nová verze DTP programu Scribus 1.1.7, „Alyssa“. Jedná se převážně o „bug-fix“ vydání, ale ani o nové funkce není nouze. Více na domovské stránce projektu www.scribus.net.
- 7.6.2004** *Petr Krčmář*
Mladému studentovi se podařilo pomocí reverzního inženýrství zdokumentovat autentizační a komunikační protokoly iTunes. Získané informace mají umožnit práci s iTunes i na oficiálně nepodporovaných platformách.
- 7.6.2004** *Tomáš Hála*
Zeptejte se svých kandidátů do Evropského Parlamentu, jak budou hlasovat v kauze softwarových patentů (např. s odkazem na otevřený dopis médiím), a podělte se o reakce v diskusi.
- 7.6.2004** *Petr Krčmář*
10.–13. června se v Berlíně uskuteční konference o svobodném software, které se zúčastní více než stovka mluvčích z celého světa.
- 7.6.2004** *Petr Krčmář*
Mozilla foundation a Opera Software společně vytvořili novou pracovní skupinu, která se zaměří na vývoj nových specifikací pro web, CSS a další aplikace.
- 7.6.2004** *Petr Krčmář*
Víte jak se baví správní hackeři? Pokud ne, napoví vám tento animovaný gif. Zaslal Kaktus.
- 8.6.2004** *Petr Krčmář*
Sun Microsystems podepsal smlouvu s ministrem školství Ontaria o nasazení StarOffice do 72 školských zařízení.
- 8.6.2004** *Petr Krčmář*
Včera uběhlo 50 let od úmrtí Alana Turinga, známého matematika a počítačového průkopníka.
- 8.6.2004** *Petr Krčmář*
Pěkný ascii art poskládaný ze zdrojů jádra v rozlišení 257x303 znaků poslal Pajus.
- 8.6.2004** *Ondra 'Kepi' Kudlík*
Po několikaměsíční odmlce vzniklé kvůli kompromitování gnome.org webserveru se alespoň v prozatímním statickém kabátu vrátil art.gnome.org.
- 8.6.2004** *Martin Fiala*
MandrakeLinux bude pravděpodobně další distribucí, která vymění XFree86 za X.Org. Ve verzi Cooker k tomu došlo již dnes, upgrade proběhl bez problémů.
- 8.6.2004** *finn*
Vyšla další subverze legendárního textového editoru Vim ? 6.3. Bylo opraveno mnoho chyb, přidána podpora pro přeložené soubory nápovědy (ChangeLog).
- 9.6.2004** *srakysi*
Po mnoha žádostech byl spuštěn webový formulář, v němž můžete vyjádřit svou podporu otevřenému dopisu Hynka Hankeho o softwarových patentech.
- 9.6.2004** *Petr Krčmář*
Linus vydal nové jádro 2.6.7-rc3, které je jednodušší a hlavně obsahuje patche kolem DRM, AGP, cpufreq, sparc64 a dalších.
- 9.6.2004** *Petr Krčmář*
RedHat a Intel společně vyvinuli linuxovou podporu pro intelovskou technologii NX (no execute), která by měla zvýšit bezpečnost systému na hardwarové úrovni.
- 9.6.2004** *Petr Krčmář*
Před necelým týdnem jsme vás informovali o tom, že Sun microsystems uvažuje o uvolnění Solarisu pod GPL. Nechvalně známá firma SCO dnes vyrukovala proti Sunu s licenci, která mu prý tento krok zakazuje.
- 9.6.2004** *Tomáš Hála*
Oblíbený proxy server Squid obsahuje buffer overflow a umožňuje vzdálené vykonání kódu. Problém se týká pouze verzí zkompileovaných s autentizací NTLM. K dispozici je patch, více viz oznámení.
- 10.6.2004** *Martin Kysela*
V úterý 8. června proběhla v Liberci přednáška a beseda k příležitosti vydání nové knihy Petra

Krčmáře. Fotky jsou k dispozici na stránkách Ky-osukeho.

10.6.2004

Petr Krčmář

O tom, že Linux začíná nepříjemně přibírat na váze a je čím dál náročnější na hardware se v obsáhlém článku rozepsal Bob Marr.

10.6.2004

Petr Krčmář

Mandrakesoft dnes uvolnil Mandrakelinux 10.0 pro platformu AMD64. Slibuje tak spojení kvalitní distribuce a nejnovějšího hardware.

10.6.2004

Jozef Vondrák

Jestli chcete znát skutečný význam slov jako linux, windows, root atd., podívejte se na www.cyborgname.com.

11.6.2004

komodo

Vyšla nová mozilla 1.7 rc3, pravdepodobne posledná rc verzia, keďže je označená ako final draft. Mimo ďalších zmien prináša v browseri tzv. „whitelisting“, ktorý obsahuje zoznam lokalít, z ktorých je povolené inštalovať rôzne rozšírenia a prídavné funkcie. Mail by mal byť opat' o niečo rýchlejší pri sťahovaní, prezeraní a ukladaní pošty, ďalej ponúka podporu viacerých identít pre jedno konto. Podrobný popis zmien nájdete na stránke mozilla.org

11.6.2004

Michal Marek (twofish)

Nechvalně známá firma SCO má menší finanční problémy.

11.6.2004

Petr Krčmář

Německý McDonald kompletně přechází na SUSE Linux. V Německu se jedná o více než 1200 restaurací.

11.6.2004

Petr Krčmář

Mono, linuxová implementace microsoftího .NET prý mnoho vývojařů ani zákazníků neláká. Kdo používá linux ho prý raději spojí s Javou.

12.6.2004

Antonín Kolísek

Vyšla nová verze senzorů lm_sensors-2.8.7. Více informací o změnách viz CHANGES.

13.6.2004

Ondřej Čečák

Pozor, díky chybě v linuxové verzi PunkBusteru není možné hrát ve hře America's Army na honor

serverech. Výrobce o problému ví a doufejme, že ho i začne řešit. Více na fóru hry.

14.6.2004

Petr Krčmář

I Debian už oznámil svou 64bitovou verzi. Ta vychází z unstable vývojové větve a přeloženo je už téměř všechno (97 %).

14.6.2004

Petr Krčmář

Skvělá live distribuce Slax má konečně oficiální český web. Ke stažení je celá distribuce, doplňky a řada zajímavých informací.

14.6.2004

Rastislav Stanik

Veríte v stabilitu svojho Linuxového systému? Možno to nie je až také ružové.

14.6.2004

Petr Krčmář

Po dvou letech testování vyšla nová verze ray-traceru POV-Ray (3.6). Hlavní úpravy proběhly na renderovacím jádře, byla zvýšena stabilita a program je připraven na další implementace.

14.6.2004

finn

Před pár dny se na linuxreviews.org objevilo oznámení o chybě linuxových jader řad 2.4.2x a 2.6.x, útočníkovi stačí k napadení přístup k shellu.

14.6.2004

Vlastimil Ott

Ještě jsou volná místa na konferenci SLT 2004 (www.cstug.cz/slt/04/). Podrobnější informace (program, informace o registraci, atd.) najdete na adrese www.cstug.cz/slt/04/.

15.6.2004

Petr Krčmář

Počet linuxových řešení stále roste na úkor ostatních platforem. Nedávno jej například na svých strojích nainstaloval soudní dvůr ve Washingtonu.

15.6.2004

Petr Krčmář

Mozilla Firefox se v nejbližších měsících dočká první ostré verze (1.0). Měla by se objevit ke konci léta.

15.6.2004

finn

Vyšla očekávaná Mozilla Firefox 0.9. K dispozici jsou verze pro linux, mac i win32 v angličtině. Nová verze je štihlejší, byl změněn vzhled, byl

přidán správce témat, nový je systém nápovědy. Podrobnější seznam k nahlédnutí na mozilla.org.

15.6.2004

Josef Vybíral

Na FTPku GIMP.org se objevily zdrojové kódy dalšího bugfixu GIMPu, tentokrát ve verzi 2.0.2. Opravuje některé bugy z práce s tabletem, chování pluginů... Pro toho, komu se nechce stahovat celých 13 MB, jsou připraveny i patche na v2.0.1. Zase jsme o krůček blíž verzi 2.2.

16.6.2004

Petr Krčmář

O budoucnosti Mozilla Firefoxu, open source desktopu, GNOME a dalších tématech v zajímavém rozhovoru se Scottem Collinsem z Mozilla.org.

16.6.2004

Martin Krauskopf

Po bouřlivých změnách v Changelogu přichází na svět první „release candidate“ Slackwaru 10.0. Oficiální stránky prozatím mlčí. Na změny se můžete podívat přímo do Changelogu a ISO můžete stahovat třeba odsud nebo použít geniální Swaret. Keep on slacking ;)

16.6.2004

Daniel Čížinský

Do světa byla vypuštěna nová verze stabilní větve jádra Linuxu - 2.6.7. Z Čech pocházejí například poměrně výrazné změny v ovladačích zvukových karet ALSA, opravy v kódu kvót, či vstupních zařízení. Z často používaných zařízení došlo kupříkladu k rozdělení ovladače pro IDE řadiče Promise. Změn je opravdu mnoho, mezi nimi spousta oprav, takže stahujte, kompilujte, testujte... Nové jádro je zatím k dispozici na hlavním serveru. Český mirror na synchronizaci zatím čeká!

16.6.2004

Petr Krčmář

O tom, že Java je už dnes výkonostně srovnatelná s jinými jazyky, jsme už na ABC psali. Nově to potvrzuje i Sys-con.

16.6.2004

Petr Krčmář

Novým šéfredaktorem měsíčníku Linux+ se od sedmého čísla stává náš známý Martin Kysela. Podle zákulisních informací se chce založit o rapidní zvýšení kvality tohoto papírového magazínu.

16.6.2004

Vlastimil Ott

Rádio Akropolis, jedno z nejposlouchanějších internetových rádií v ČR, vyhlašuje konkurs na nového moderátora pořadu Terabajt, který je zaměřen na počítače a IT. Požadavkem je přehled v oblasti IT a chuť investovat trochu svého času. Předchozí praxi nepožadujeme. Případní zájemci se mohou hlásit na e-mailovou adresu: terabajt@radioakropolis.cz.

17.6.2004

Petr Krčmář

Sun oficiálně potvrdil svůj záměr uvolnit Solaris jako open source.

17.6.2004

Petr Krčmář

Poštovní klient Mozilla Thunderbird se dočkal verze 0.7. Je menší, rychlejší, obsahuje nového správce témat vzhledů a další vylepšení. Stahovat můžete třeba na ftp.mozilla.org.

18.6.2004

finn

Mozilla 1.7 (zatím jen v angličtině) je na světě. Celkový počet oprav a změn dosáhl úctyhodné hodnoty sedmnácti set. Ve srovnání s předchozí verzí je rychlejší a menší, zájemci najdou podrobný seznam nových vlastností (jsou jich desítky) na mozilla.org.

18.6.2004

Petr Krčmář

Na linuxdevices.com je k vidění velmi poučný seznam procesorů a jednočipů, které si rozumí s Linuxem, včetně odkazů na další informace. Jen pro zajímavost se jedná o více než 150 různých architektur!

18.6.2004

Petr Krčmář

Již tradiční přednášková akce OpenWeekend 2004 se letos koná opět v Praze, v budově ČVUT, 16.-17.10. 2004.

18.6.2004

tomasini

Chyběli jste na akci Openparty v Olomouci? Jaká byla účast a atmosféra, můžete posoudit z několika fotografií.

18.6.2004

Petr Krčmář

Nové, tentokrát již páté pokračování příběhu Yetiho a tučnaka je na světě. Tentokrát se s nimi podíváme mezi plameňáky do Afriky.

- 18.6.2004** *Vlastimil Ott*
Nic originálního (naštěstí): druhé největší norské město Bergen migruje na SUSE Linux. Důvod: úspora a otevřenost systému.
- 18.6.2004** *Vlastimil Ott*
Na světě je OpenOffice.org 1.1.2 Hakone. Více jazyků, lepší práce s XML, nová komponenta pro práci s fonty. Stáhnout.
- 18.6.2004** *Vlastimil Ott*
Podle heise.de se Nokia stala sponzorem Mozilla Foundations. Je pro ni vyvíjena platforma Minimo – prohlížeč pro chytré telefony.
- 19.6.2004** *Lukáš Zapletal*
Druhý kandidát na Eclipse 3.0 již bude týden venku, brzy se tedy dočkáme finální verze. Eclipse je vývojové prostředí napsané v Javě (a částečně v jazyce C), novinek ve verzi 3.0 je mnoho.
- 21.6.2004** *Petr Krčmář*
Pro ty z vás, kteří se ještě neodhodlali vyzkoušet GNOME 2.6, připravil Chris Dunphy rozsáhlý článek, který shrnuje jeho zkušenosti s tímto prostředím.
- 21.6.2004** *Bob Bobovich*
ISO SUSE 9.1 Personal je k dispozici ke stažení z FTP SUSE i mirrorů.
- 21.6.2004** *Petr Krčmář*
RedHatu se očividně daří. Prodej jeho produktů zaznamenal nárůst o 53 % proti stejnému (druhému) čtvrtletí minulého roku. Letos tedy činil 41.6 milionů dolarů.
- 21.6.2004** *Petr Krčmář*
Na webu update.mozilla.org vznikl zajímavý portálek, který nabízí ke stažení grafická témata a vylepšení (pluginy) pro Mozillu, Mozillu Firefox a Thunderbird.
- 21.6.2004** *Petr Krčmář*
Jak na Linux ve dvaceti krocích, co všechno musíte vědět a co si musíte ujasnit, vám poradí osnews.com. Zajímavé čtení pro úplné začátečníky.
- 22.6.2004** *Antonín Kolísek*
Program Skype pro internetovou telefonii byl konečně portován pro Linux.
- 22.6.2004** *Petr Krčmář*
Linux už běží na více než polovině počítačů, které jsou v současné době v žebříčku 500 nejrychlejších počítačů světa.
- 22.6.2004** *Petr Krčmář*
BBC oznámilo, že se v Iráku skupina Linuxových uživatelů rozhodla vytvořit vlastní svobodný operační systém, který by měl být levnou a dále se rozšiřující platformou. Chtějí tak pomoci své zemi k lepší budoucnosti.
- 23.6.2004** *Petr Krčmář*
Ohlašování podpory 64bitových procesorů se stává novým trendem. Dalším z ohlašovatelů je VMware. Podpora zahrnuje virtualizaci procesorů AMD64, Intel EM64T a možnost zároveň využít 64 i 32 bitové aplikace.
- 23.6.2004** *Petr Krčmář*
Nová studie antispamových filtrů prokázala, že nejúčinnějším z testovaných programů je Spamassassin, který dokáže redukovat počet spamů ze 150 na pouhé 2 denně.
- 24.6.2004** *Plesinger Jindrich*
Slackware 10.0 je na světě. Jádro 2.4.26 a 2.6.7, GNOME 2.6.1, KDE 3.2.3, X.org a tuny dalšího softwaru. ChangeLog, seznam všech balíčků.
- 24.6.2004** *Petr Krčmář*
Sun je často kritizován za licenční podmínky, které si u svých produktů klade. Velmi pozitivně v tomto světle působí zjištění, že je zároveň ochoten o těchto problémech diskutovat.
- 24.6.2004** *Petr Krčmář*
ION je moderní, komerční podnikové řešení, které slibuje pevné spojení dodaného hardware a software, kterým je Debian, upravený pro podnikový desktop. Jeho výrobce slibuje 100
- 24.6.2004** *Petr Krčmář*
RedHat a Oracle spolu založili Linux Enterprise Applications Porting Centre, které se má v budoucnu zaměřit na tvorbu (portování) enterprise aplikací na společných platformách obou firem.

25.6.2004

Petr Krčmář

New York City *BSD User Group přišla s velmi originální myšlenkou. Její členové založili fórum, kam uživatelé posílají výpisy dmesg. Cílem je vytvořit rozsáhlou databázi kombinací hardware, na kterém *BSD běží.

25.6.2004

Petr Krčmář

SuSE ČR, s.r.o. definitivně ukončila svou činnost. Produkty SUSE jsou nyní k dispozici prostřednictvím partnerské sítě společnosti Novell Praha. UPDATE: SUSE samozřejmě nekončí, pouze obchodní aktivity firmy přecházejí na Novell Praha [Ott].

28.6.2004

Petr Krčmář

Sun oznámil zveřejnění kódu Java 3D API na java.net. K dispozici jsou samotné kódy, základní utility a příklady.

28.6.2004

Petr Krčmář

Na myšlenky Linuxfromscratch se nyní pokouší navázat také nový projekt Minixfromscratch. Cílem je zpřístupnit tvorbu operačního systému i průměrným uživatelům, kteří si tak mohou vytvořit svou vlastní distribuci Minixu a proniknout tak hlouběji do podstaty systému.

28.6.2004

Petr Krčmář

Na mozilla.sk jsou ke stažení slovenské lokalizační balíčky pro Mozilla 1.7 a Mozilla Firefox 0.9.

28.6.2004

Petr Krčmář

Spolupráce RedHatu a Novellu přináší první viditelné ovoce. Helix player je plnohodnotnou náhradou komerčního RealPlayeru a zatím běží pod Linux, Solaris a Symbian. A to nejlepší: je pod GNU GPL.

29.6.2004

Bob Bobovich

Na stažení čekají Firefox 0.9.1 a Thunderbird 0.7.1. Řeší řadu chyb, které byly objeveny brzy po vydání předchozích verzí. Mimo jiné se jedná i opravu několika problémů, které se projevují v neanglických verzích programů.

29.6.2004

Petr Krčmář

Sun se očividně vydal správnou cestou. Dnes oznámil otevření dalšího ze svých projektů, ten-

tokrát se jedná o Projekt Looking Glass, který nabízí 3D prostředí na Linuxovém desktopu. Obrazky i video vypadají velmi slibně.

29.6.2004

Petr Krčmář

Používáte-li Mandrakelinux 10.0 nebo se na něj chystáte, podívejte se na poslední verzi errata, kde můžete najít seznam objevených problémů a jejich řešení.

29.6.2004

Petr Krčmář

Ty z vás, které zajímá ladění systému za účelem dosažení nejvyššího výkonu pravděpodobně zaujme zpráva, že Peter Williams vydal novou verzi jaderného patche CPU Scheduler Evaluation pro jádra 2.6.7, který umožňuje změnu scheduleru za jízdy(!).

29.6.2004

Petr Krčmář

Wi-Fi se dočkalo nového standardu 802.11i, který má pomocí AES a WPA2 šifrování konečně zajistit dostatečnou neprůstřelnost bezdrátových sítí.

29.6.2004

Vlastimil Ott

Redakce Databázového světa hledá redaktora s perfektním rozhledem v databázových technologiích a znalostí češtiny. Více na dbsvet.cz.

29.6.2004

Vlastimil Ott

Zajímavý článek a zajímavá diskuze pod ním se objevil/a na Lupě: Softwarové patenty aneb Poslední dějství počítačového programování?

30.6.2004

Petr Krčmář

Pokud se vám líbí sunovský projekt Looking Glass, rádi byste měli něco podobného, ale nechcete investovat tolik procesorového času, můžete se podívat třeba na Metisse. Používá to OpenGL a rozhýbe vaše okna ve 3D.

30.6.2004

Petr Krčmář

O tom, že vyšlo Skype (software pro IP telefonii) pro Linux jsme už informovali. Dnes se na NewsForge objevila krátká, ale zajímavá recenze.

30.6.2004

Petr Krčmář

Zajímavě vypadá společný projekt studentů mechatroniky z Linköping University (Švédsko) a Volvo Cars, který vám dovolí zaparkovat vaše auto zcela automaticky pomocí počítače s Linuxem. Můžete se podívat na video.