



Abíčko

Časopis serveru AbcLinuxu.cz

Květen 2006



Vychází také na CD-ROM jako příloha časopisu



Editoriál

Vítejte u čtení časopisu Abíčko.

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

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

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

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

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

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

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

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

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

Editor a sazba: Vlastimil Ott

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

Typografické konvence

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

Obsah

Editoriál	1
Obsah	2
Sdílení diskových oddílů mezi distribucemi	4
Proč dělit disk na oddíly?	4
Které oddíly sdílet?	4
swap	4
/tmp	4
/boot	4
Které adresáře NESdílet?	6
Uživatelská data	6
Závěr	6
LiveCD	7
Úvodem	7
Zavaděč	7
Syslinux	8
Používáme isolinux	8
Vlastnosti a omezení isolinuxu	10
Závěr	10
Komprimované souborové systémy – Úvodem	10
Komprimované souborové systémy	11
Cramfs	11
Cloop	11
zisoofs	12
e2compr	12
SquashFS	12
Podpora pro zápis – Úvodem	13
ovlfs	13
COW	13
UnionFS	13
Instalujeme	14
Používáme UnionFS	14
Povolení zápisu	15
Problémy UnionFS	16
Linux live, před spuštěním systému – Úvodem	16
initrd	16
Busybox	17
Po spuštění jádra	17
Linuxrc krok za krokem	17
chroot a spuštění <code>/sbin/init</code>	20
Závěrem	20
Úprava distribuce Slax – Úvodem	21
Adresář <code>rootcopy</code>	21
Skript <code>make_iso.sh</code>	21
Moduly	22
Co je to vlastně modul	22

Rozšiřujeme Slax pomocí modulů	23
Závěrem	24
Konfigurace multimediální klávesnice v X	25
Úvod	25
Řešení	25
Snadná konfigurovatelnost	28
Závěr	28
Rozhovor: Česko se stává linuxovou mocností	30
Jaderné noviny – 22. 3. 2006	31
Aktuální verze jádra: 2.6.16	31
Citát týdne	31
Technický poradní panel OSDL	31
Co bude nového v 2.6.17	31
Na poslední chvíli se diskutovalo o unshare()	32
Řešení problému s hladověním ve scheduleru	32
Jaderné noviny – 29. 3. 2006	35
Aktuální verze jádra: 2.6.16.1	35
Co bude nového v 2.6.17, část 2	35
Rámec pro pravidla nahrazování stránek	36
Správa stromu 2.6.16.y	37
Co z -mm půjde do 2.6.17	37
SCSI ID	38
Jaderné noviny – 5. 4. 2006	39
Aktuální verze jádra: 2.6.17-rc1	39
Citát týdne: Al Viro.	39
Dvě nová systémová volání: splice() a sync_file_range()	39
splice()	39
sync_file_range()	40
Odvozování priorit v jádře	40
O bezpečnosti sysfs rozhraní	42
Dokumentování paměťových bariér	43
Konec přijímání patchů do 2.6.17	43
Jaderné noviny – 12. 4. 2006	44
Aktuální verze jádra: 2.6.16.5	44
Kontejnery a odlehčená virtualizace	44
Systémová volání tee() a splice()	45
Wireless Networking Summit 2006	46
Zachování údajů o oops při resetu	46
Neostré hranice GPL	47
Jaderné noviny: Wireless Networking Summit 2006	48
Historie	48
Současný stav	49
Devicescape	50
Nástroje	51
Podpora hardwaru	51
Další otázky	52
Zprávičky	54

Sdílení diskových oddílů mezi distribucemi

Jirka Novák

Sdílení diskových oddílů připadá v úvahu v případě, že máte na počítači nainstalováno několik operačních systémů – např. BSD a Linux nebo třeba Windows 9x a Windows XP. My se ale podíváme na zoubek situaci, v níž máte několik různých distribucí Linuxu.

Proč dělit disk na oddíly?

Na takto položenou otázku lze sice jednoduše odpovědět, ale pokud se má administrátor správně rozhodnout, musí si uvědomit pro a proti, která to přináší:

Pro

- Obrana proti DoS útokům (např. zaplněním kořenového adresáře).
- Možnost optimalizace přístupu k jednotlivým diskovým oddílům.
- Snazší správa – zejména zálohování.
- Pohodlí při práci – máme k dispozici všechna svá data.
- Úspora diskového prostoru.

Proti

- Se zvyšujícím se počtem disk. oddílů se zvyšuje poměr nevyužitého volného místa k diskové kapacitě.

Z tabulky je vidět, že existuje několik důvodů pro. Jsem zastáncem dělení disku na oddíly, tak je možné, že mě nějaké zápory nenapadly. Laskaví čtenáři je mohou doplnit v diskusi pod článkem. Rozhodnutí je v každém případě na konkrétním administrátorovi.

Které oddíly sdílet?

swap

Odkládací oddíl je nepochybně přirozeným kandidátem. Současně není spuštěno více distribucí, takže v tomto směru není důvod swap nesdílet. Samozřejmě existují případy, kdy takové sdílení možné není. Jedním z nich je situace, která je naznačena v předchozím odstavci - totiž ta, kdy běží více distribucí současně, tj. v případě virtuálních strojů. Dalším případem je situace, kdy používáme swap partition na hibernaci. Pokud tedy můžeme sdílet odkládací oddíl, zapíšeme do souboru `/etc/fstab` následující řádek:

```
/dev/hda5 none swap
```

Použití zajistíme příkazem `swapon [1]` nebo restartem stroje.

/tmp

Dalším oddílem, který můžeme celkem bez problémů sdílet, je adresář `/tmp`. Zabránit nám v tom může jediná věc - pokud je adresář používán pro ukládání uživatelských dat, tj. dokumentů, zdrojových souborů, atd. V takovém případě je třeba zvážit, zda je to vhodné. Vlastní sdílení pak není nic složitého, stačí přidat stejný řádek do souboru `/etc/fstab` ve všech distribucích. Osobně ještě při spouštění systému adresář `/tmp` vymažu.

/boot

Sdílení předchozích dvou oddílů bylo víceméně přirozené. Sdílení adresáře `/boot` již vyžaduje trochu plánování. Přinese nám ale mnohem snazší správu bootování – budeme mít pouze jednu instalaci zavaděče a jednu konfiguraci. V Linuxu se používá většinou buď `LILLO` [2], nebo `GRUB` [3]. Pokud chceme mít pouze jeden konfigurační soubor, musíme jej uložit na sdílenou partition – přirozené umístění bude v adresáři `/boot` (sdílet adresář `/etc` nedoporučuji). V případě LILLO je tento proces poněkud násilný, proto doporučuji zaměřit se na GRUB – ten budu popisovat v následujícím textu. Myšlenka je prostá – potřebujeme partition, na které budou všechny soubory potřebné pro zavedení všech systémů a konfigurační soubor pro boot loader.

GRUB čte konfiguraci z `/boot/grub/menu.lst`. Takže už zbývá jen sdílet partition s adresářem `/boot`. Ale když si např. v ArchLinuxu aktualizujete kernel, zapíše se do `/boot/vmlinuz26`. V případě, že byste v tomto adresáři již měli takto pojmenovaný soubor určený navíc pro jinou distribuci, přidělali byste si zbytečné problémy. Proto je vhodné ukládat soubory specifické pro jednotlivé distribuce do samostatných adresářů. Vlastní mapování adresáře `/boot` pro konkrétní distribuci pak bude bohužel o něco složitější, ale není to nic hrozného. Do adresáře `/mnt/boot_partition` si připojíme diskový oddíl – do `/etc/fstab` přidáme:

```
/dev/hda1    /mnt/boot_partition    ext2    defaults    0    0
```

V tomto oddílu vytvoříme samostatné adresáře pro každou distribuci:

```
/mnt/boot
|-- arch
|-- gentoo
|-- grub
|-- initrd-tree
|   |-- bin
|   |-- dev
|   |-- etc
|   |-- lib
|   |   '-- modules
|   |       '-- 2.6.12.1
|   |-- mnt
|   '-- proc
|-- lost+found
'-- slack
```

Obsah původního adresáře `/boot` nakopírujeme do příslušného adresáře v `/mnt/boot_partition`:

```
# cp -R /boot/* /mnt/boot_partition/arch
```

Adresář `/boot` nalinkujeme na konkrétní adresář v `/mnt/boot_partition` příkazem (např.)

```
# rm -rf /boot
# ln -s /mnt/boot_partition /boot
# ls -l /boot
lrwxrwxrwx  1 root root 15 2006-04-24 23:27 /boot -> /mnt/boot_partition/arch/
```

Nakonec upravíme konfiguraci GRUBu v souboru `/mnt/boot_partition/grub/menu.lst`:

```
title ArchLinux 0.7.1 Noodle
root (hd0,8)
kernel (hd0,0)/arch/vmlinuz26 root=/dev/hda9 ro vga=normal
initrd (hd0,0)/arch/initrd26.img
```

Které adresáře NESdílet?

Na tomto místě se spokojíme pouze s výčtem adresářů a se stručným zdůvodněním.

/etc

V adresáři `/etc` je uložena konfigurace, která může být specifická pro konkrétní distribuci.

/home

V domovském adresáři uživatelů jsou uložena nastavení používaných aplikací.

/usr

Každá distribuce může používat jiné verze jiných programů.

/var

Hrozí poškození databázových souborů, nekonzistence logu, atd.

ostatní

Umísťovat ostatní adresáře na samostatný oddíl nemá smysl. Až na výjimky – jednu jsem si nechal na závěř.

Uživatelská data

Uvedený postup je možná zajímavou ukázkou možností úspory místa, ale zase tolik toho nepřináší. V tabulce v úvodu byl jako důvod „Pro“ uveden jeden zásadní: máme k dispozici všechna svá data. Ale my je zatím k dispozici nemáme! Adresář `/home` přece nesdílíme!

Ano, to je pravda. Existují speciální případy adresářů, které je příjemné sdílet, ačkoli se běžně neumísťují na samostatné oddíly. Pro takové adresáře doporučuji vytvoření jednoho oddílu, řekněme mu třeba datový, který bude připojen do `/mnt`, a jednotlivé adresáře budou nalinkovány do adresářové struktury. Obvykle se jedná o adresáře s webovými stránkami, adresáře s CVS [repository](#) [4], adresáře s multimediálními soubory a samozřejmě zejména o adresáře s uživatelskými daty. Popisovat poslední jmenovaný případ by bylo nošením dříví do lesa. Princip je stejný jako v případě adresáře `/boot` – do `/mnt/data_partition` připojíme zmíněný datový oddíl...

```
mount -t auto /dev/sda5 /mnt/data_partition
```

... a uživatelé si pak ve svých domovských adresářích mohou vytvořit link na svůj adresář v tomto oddíle:

```
link -s /mnt/data_partition/users/camlost ~/data
```

Je zřejmé, že by bylo nutné nastavit připojení oddílu v souboru `/etc/fstab`, aby byl připojen při každém spuštění systému. Umístění jmenovaných souborů na jeden společný datový oddíl má také zajímavé vedlejší efekty – při instalaci Linuxu nepřijdete o data a je snadné jej zálohovat.

Závěr

V tomto článku jsme si ukázali, co je nutné pro sdílení diskových oddílů mezi distribucemi Linuxu. Nyní vám již nic nebrání zkoušet nové verze vaší oblíbené distribuce nebo distribuce úplně nové.

Odkazy

- [1] <http://www.doc.ic.ac.uk/lab/labman/lookup-man.cgi?command=swapon>
- [2] <http://www.abclinuxu.cz/slovník/lilo>
- [3] <http://www.abclinuxu.cz/slovník/grub>
- [4] <http://www.abclinuxu.cz/slovník/repozitar>

LiveCD

Michal Vyskočil

Pojďte se podívat, jak fungují LiveCD. Po lehkém úvodu se dostaneme k velmi zajímavé problematice bootování počítačů a k zavaděčům. Dozvíte se o rodině zavaděčů syslinux. Nakonec si projdeme isolinux a tvorbu bootovacího CD.

Úvodem

Živý (Live) systém je operační systém, který je upraven tak, že dokáže běžet z CD-ROM, nebo DVD-ROM média. Některé umí [bootovat](#) [1] například i z USB Flash disku. Používají se jako:

- Prezentace – buď přímo celého operačního systému, nebo jeho konkrétní distribuce.
- Opravné nástroje – live systémy jsou velmi užitečné v případě, že je systém nainstalovaný na harddisku poškozen.
- Přenosné pracovní prostředí – stejný systém na libovolném počítači.

Asi nejčastěji bývají LiveCD založeny na upravené distribuci GNU/Linuxu. Příkladem je Knoppix nebo Danix založené na Debianu. Případně Slax, respektive [ABC Linux 2005](#) [2], založené na Slackware. Ale k dispozici je řada LiveCD i ostatních známých operačních systémů, jako OpenSolaris, xBSD, Microsoft Windows, Mac OS. A dokonce i těch méně známých, např. založených na jádru Hurd a podobně. Seznam některých nelineuxových liveCD:

- [Belenix](#) [3] – OpenSolaris
- [ReactOS LiveCD](#) [4] – ReactOS
- [Hurd LiveCD](#) [5] – Hurd
- [BootCD](#) [6] – nástroj pro tvorbu LiveCD založených na Mac OS X
- [Bart PE](#) [7] – nástroj pro tvorbu LiveCD založených na Windows

Předpokládám, že většina čtenářů Abička se už v nějaké formě s Live systémy setkala ([Exkurze do světa live distribucí](#) [8]), takže předchozí úvod byl tak trochu nošením dříví do lesa. V dalším textu se budu specializovat především na linuxová LiveCD, ale dnešní díl je aplikovatelný i pro jiné systémy.

Zavaděč

Vezměme to od Adama, tedy od bootování. První nezbytnou položkou je [zavaděč](#) [9] (bootloader), což je program, který zavolá počítač a zavádí (bootuje) vlastní systém. Zavaděče rozdělujeme na dvě kategorie:

Primární (first-stage)

jsou přímo součástí hardwaru počítače. Ve světě PC se používá poměrně hloupý BIOS od IBM. Společnost Sun vyvinula daleko pokročilejší zavaděč zvaný Open Firmware (nebo OpenBoot) napsaný v jazyce Forth (viz článek [OpenBoot prompt](#) [10]). Ten se používá v PowerPC, Sparc, Ibm Power a ostatních systémech.

Sekundární (second-stage)

kteří volá právě primární zavaděč. Určitě znáte programy jako [LILO](#) [11] (Linux Loader), [Grub](#) [12] (Grand Unified Bootloader), loadlin (aplikace, která je schopná z DOSu zavést Linux). Nebo NTLDR, zavaděč operačních systémů rodiny Windows NT. Dále sem patří některé univerzální bootovací manažery, například XOSL.

V této souvislosti si neodpustím poznámku. Jak Sun Microsystems, tak Apple Computers migrují na PC platformu, proto mě zajímalo, zda jejich intelovské produkty používají BIOS nebo OpenFirmware.

Operační systém firmy Sun Solaris (respektive OpenSolaris) může běžet na běžných PC s BIOSem. Společnost Apple sice od používání OpenFirmware odstoupila, ale na rozdíl od Sunu nepodporuje běh na běžných PC. Ve svých Intel-based počítačích používá Extensible Firmware Interface (EFI), který byl vyvinut společností Intel pro architektury IA-32 a Itanium.

Syslinux

Mimo výše zmíněné komplexní zavaděče, jako je třeba GRUB, existuje i velmi jednoduchý bootloader zvaný syslinux. Tedy ve skutečnosti se jedná i balík programů, protože pod označením syslinux se skrývá celá škála jednoúčelových zavaděčů:

- `syslinux`, pro bootování ze souborového systému FAT
- `isolinux`, pro bootování z ISO 9660 filesystému, který je používán na CD
- `pxelinux`, bootování ze síťového serveru, více najdete v článku [Jak nabootovat Linux po síti \[13\]](#)
- `extlinux`, bootování ze souborového systému ext2/ext3
- `memdisk`, bootování starších operačních systémů

Pro nás je pochopitelně nejzajímavější právě zavaděč isolinux. Většina z nás pravděpodobně poprvé viděla toto jméno při spuštění instalačního CD, které spolu s LiveCD bootují právě prostřednictvím tohoto zavaděče. Samotný proces bootování z CD popisuje El Torito Bootable CD Specification, což je rozšíření specifikace ISO 9660, které umožňuje bootovat z CD-ROM. Z historických důvodů existují dva režimy zavádění:

Floppy Emulation Mode

V tomto režimu je zavaděcí sekvence uložena v obraze (image file) diskety. Obraz je nahrán z CD a zavádění probíhá stejně jako z klasického pevného disku nebo diskety. Protože souborovým systémem obrazu diskety bývá FAT, používaným zavaděčem je syslinux. Tento režim vznikl z důvodu, že tehdejší BIOSy nenačítaly bootovací sekvence z ISO 9660 systémů.

No Emulation Mode

V tomto případě je zavaděcí sekvence uložena přímo na disku CD-ROM a bootování probíhá přímo z něj. Tento režim je vyžadován zavaděčem isolinux a vzhledem k tomu, že všechny BIOSy z posledních dejme tomu 7 až 10 let podporují zavádění z CD-ROM, tak není moc důvodů, proč používat emulační mód.

Používáme isolinux

Následující text je takové mini-howto k isolinuxu. Protože je ten zavaděč skutečně velmi jednoduchý, tak v podstatě není, o čem psát. K vytvoření obrazu CD potřebujete program `mkisofs` verze 1.13 a vyšší, který je součástí balíku `cdrtools`.

Nejprve si musíte do některého adresáře zkopírovat věci, které mají být na vašem LiveCD. Ten se stane kořenovým vašeho CD. Isolinux standardně hledá konfigurační soubor `isolinux.cfg` ve třech adresářích `/isolinux`, `/boot/isolinux` nebo v `/`. Můžete si vybrat libovolný z nich. V něm potom uvedete umístění dalších potřebných souborů. Následuje ukázka nastavení z distribuce ABC Linux 2005. Více podrobností se dočtete v dokumentaci v souboru `syslinux.doc`.

```
display boot/splash.cfg
default linux
prompt 1
timeout 0
F1 boot/splash.txt
```

```
F2 boot/license.txt

label abc
kernel boot/vmlinuz
append max_loop=255 initrd=boot/initrd.gz init=linuxrc load_ramdisk=1 \\
    prompt_ramdisk=0 ramdisk_size=4444 root=/dev/ram0 rw

label linux
kernel boot/vmlinuz
append max_loop=255 initrd=boot/initrd.gz init=linuxrc load_ramdisk=1 \\
    prompt_ramdisk=0 ramdisk_size=4444 root=/dev/ram0 rw

label memtest
kernel boot/memtest
```

Jak vidíte, formát souboru je poměrně primitivní a nepotřebuje příliš komentářů.

<code>display</code>	je uvítací zpráva, kterou má zavaděč zobrazit
<code>default</code>	je výchozí obraz jádra, který se má zavést
<code>prompt</code>	možnost zadávat vlastní bootovací parametry
<code>timeout</code>	počet sekund do automatického spuštění, 0 znamená neustálé čekání
<code>Fx</code>	soubor, jehož obsah se vypíše po stisku funkční klávesy
<code>label</code>	je krátký název jádra
<code>kernel</code>	definuje jeho umístění
<code>append</code>	další parametry předané jádru

Na příkladu rovněž vidíme, že `isolinux` nezavádí pouze linuxové jádro, ale kterýkoliv soubor typu `x86 boot sector`, v našem případě `memtest`. Soubory potřebné pro naboootování:

```
$ tree boot
boot
|-- initrd.gz
|-- isolinux.bin
|-- isolinux.boot
|-- splash.bmp.gz
|-- splash.cfg
|-- splash.lss
|-- splash.txt
'-- vmlinuz
```

<code>initrd.gz</code>	init ram jádra
<code>isolinux.bin</code>	vlastní binárka zavaděče
<code>isolinux.boot</code>	bootovací data
<code>splash.lss</code>	obrázek ve formátu LSS16 (podporuje pouze 4 barvy), lze ho získat programem <code>ppmtolss16</code> přímo z balíku <code>syslinux</code>
<code>vmlinuz</code>	Linux, jádro systému

V okamžiku, kdy máme vše připraveno, můžeme vytvořit image příkazem (za předpokladu, že pracovní adresář je kořenovým našeho CD)

```
mkisofs -o foo.iso -b boot/isolinux.bin -c boot/isolinux.boot \\  
-no-emul-boot -boot-info-table -boot-load-size 4 \\  
-v -J -R -D -A "LiveCD" -V "LiveCD" .
```

Parametry na předposledním řádku nejsou povinné, nicméně je používá Tomáš Matějček při tvorbě ISO souborů distribuce Slax. Pohled do dokumentace isolinuxu říká následující:

<code>-b</code>	bootovací obraz dle standardu El Torito
<code>-c</code>	katalog zavaděče (boot catalog)
<code>-no-emul-boot</code>	aktivuje neemulační režim bootování
<code>-boot-info-table</code>	zapiše El Torito boot info table
<code>-boot-load-size</code>	počet virtuálních sektorů které jsou načteny v neemulačním režimu
<code>-v</code>	„ukecaný“ verbose režim
<code>-J</code>	rozšíření Joliet
<code>-R</code>	rozšíření Rock Ridge
<code>-D</code>	zakáže deep directory relocation
<code>-A</code>	název aplikace (application id)
<code>-V</code>	název svazku (volume id)

Vlastnosti a omezení isolinuxu

Isolinux má některá omezení, s nimiž je nutné počítat. Především používá unixový styl psaní cest, přičemž délka je omezena na 255 znaků. Podporuje pouze názvy definované v ISO 9660, ale už ne soubory využívající rozšíření Rock Ridge nebo Joliet. Pod Unixem si můžeme názvy ověřit připojením obrazu s parametry `-o norockridge, nojoliet`. Nicméně isolinux podporuje dlouhé (level 2) názvy souborů. Takže pokud nepotřebujete podporovat systémy s omezenou délkou názvů (jako MS-DOS), můžete vygenerovat obraz s parametrem `-l`, nebo `-iso-level 2`. Tím se možná délka názvu zvýší až na 31 znaků.

Závěr

V úvodu tohoto dílu jsme se dozvěděli něco o Live systémech. Potom jsme přešli k bootování a zavaděčům. Nakonec jsme se seznámili se zavaděčem `isolinux`, který je používán pro tvorbu bootovatelných CD. Další díl bude věnován problematice souborových systémů pro LiveCD.

Komprimované souborové systémy – Úvodem

Souborový systém je způsob organizace dat na záznamovém médiu. Zahrnuje jednak způsob fyzického uložení dat a také poskytuje vysokoúrovňový přístup k nim. Existují stovky různých systémů. Počínaje klasickými diskovými, jako je primitivní FAT nebo transakční a žurnálovací ext3. Dále sem patří síťové souborové systémy, jako například NFS nebo Coda. Existují i databázové, jakým je BFS (BeOS FS). V Unixu a v Linuxu zvlášť je také silná tradice používání pseudo filesystémů, jako `procfs` [14], `relayfs` nebo `sysfs` [15]. No a také existuje řádka souborových systémů, které jsou velmi speciální.

Linux zdědil z Unixu skvělý mechanismus zvaný Virtual File System, zkráceně VFS. Toto je aplikací jednoho ze základních unixových principů, oddělit mechanismus od postupu. V praxi to znamená, že všechny kód je soustředěn v příslušném modulu (například `xfs.ko`, který implementuje abstraktní rozhraní VFS. Přidání nového podporovaného souborového systému je snadné, stačí napsat nový modul.

Ovšem důsledkem je, že linuxové jádro samo o sobě neumí pracovat s *žádným* souborovým systémem. Jenže při bootu se musí připojit kořenový souborový systém, ale naše jádro nekodáže načíst z disku potřebné moduly. Proto musíme do jádra podporu pro kořenový fs zakompilovat natvrdo. Anebo použít `initrd` (initial ram disk), což je v podstatě diskový oddíl, který se při startu připojí jako kořenový adresář a který obsahuje potřebné moduly pro jádro. Jádro tedy musí umět pouze filesystem ramdisku, což je ve většině případů `ext2`.

Poměrně horkou novinkou je `FUSE` [16], která je v hlavním jádře od verze 2.6.14 (v době psaní tohoto článku byla stabilní 2.6.1521). Tato zkratka znamená Filesystem in Userspace a jeho myšlenka pochází už z `Plan9` [17]. Ve zkratce umožňuje uživatelským programům používat další, speciální souborové systémy. Mezi ně patří třeba GmailFS, který interně používá poštovní schránku na serveru `gmail.google.com` [18], BitTorrent File System, nebo SshFS. Zatím je dostupný pro linuxová jádra 2.4 a 2.6 a jako port pro jádra FreeBSD.

Dost bylo obecné teorie a přejděme zpět k tématu, kterým jsou LiveCD. V minulém díle jsme se seznámili se souborovým systémem ISO 9660. Ten je ovšem pro unixový systém nepoužitelný. Především proto, že neobsahuje podporu pro POSIXová práva. Rozšíření RockRidge a ISO Layer 2 sice možnosti tohoto filesystemu zvětšily, ale zůstává další problém: omezená velikost místa na disku. Plná instalace Slackware zabere 3GB, a to je distribuce obsahující velmi málo software (ve srovnání s distribucemi jako Debian, Gentoo nebo SUSE).

Komprimované souborové systémy

Naštěstí ve světě Linuxu existuje mnoho komprimovaných souborových systémů. Díky tomu můžeme do omezeného místa na CD-ROM dostat daleko více software, ovšem za cenu větší zátěže procesoru při běžném procházení.

Cramfs

Celé jméno je *compressed ROM filesystem*. Je součástí jádra od verze 2.3 (respektive 2.4). Jak už napovídá název, jedná se o komprimovaný read-only souborový systém. Některé konvenční distribuce (například Debian) používají `initrd` právě v tomto formátu. Byl využíván staršími verzemi Slax (3 a starší) nebo v embedded zařízeních jako handheldy a podobně.

```
mkfs.cramfs source destination
```

Stejně jako mnoho dalších komprimovaných filesystemů používá knihovnu `zlib` [19]. Bohužel trpí spoustou omezení:

- Maximální velikost souboru je 16MB.
- Maximální velikost souborového systému je něco málo přes 256MB.
- Podporuje hardlinky, ale počet odkazů stále zůstává roven 1.
- Není přenositelný na stroje s různým endianness (tedy s různým pořadím vyšších a nižších bytů ve slově).

Cloop

Alias *compressed loopback device* není souborový systém, ale modul, který poskytuje komprimované read-only loopback zařízení. Vytvořil jej Paul Russell (autor `ipchains` a `iptables`) pro distribuci Linuxcare, pro jádra 2.2. Mohou být naformátovány libovolným souborovým systémem (podobně jako tradiční loopback zařízení). Asi nejnámější distribucí, která tyto obrazy používá, je Knoppix (a jeho deriváty). Samotný Klaus Knopper je správcem patchů pro jádra 2.4 a 2.6, protože ve vanilla jádře tento modul není.

```
# vytvoreni obrazu
```

```
mkisofs -r -l datadir | create compressed fs - 65536 > isoimg.z

# pripojeni (linuxrc distribuce Knoppix)
insmod cloop.o file=/cdrom/KNOPPIX/KNOPPIX
mount -r /dev/cloop /mnt/knoppix
```

Data jsou komprimována po jednotlivých blocích (opět pomocí `zlib`), typická velikost na LiveCD se pohybuje kolem 256kB. Ovšem i `cloop` má svoje problémy:

- `create_compressed_fs`, což je utilita, která vytváří obraz, nepodporuje proudové zpracování. Celý komprimovaný obraz se musí vejít do operační paměti.
- ovladač `cloop` je velmi pomalý při častých čtení, například v systémech s malou operační pamětí (neustálé načítání a rozbalování bloků) nebo při nahrávání programu s velkým množstvím sdílených knihoven.

zisofs

Je klasický iso 9660 s podporou transparentní komprese a dekomprese (jak jinak než pomocí knihovny `zlib`). V jádře se objevil ve verzi 2.4.14. O vytváření obrazů se stará balík nástrojů `zisofs-tools`, které spravuje H. Peter Anvin, správce zavaděčů `syslinux` a `isolinux`, s nimiž jsme se setkali už v minulém díle. Tento způsob komprese používala distribuce Slax ve verzi 4.x. Použití programu `mkzfree(1)`, kde `input` je vstupní adresářová struktura a `output` je výstupní komprimovaná struktura. Výsledek může být zpracován klasickými nástroji, jako `mkisofs`, nebo `growisofs`.

```
mkzftree source destination
```

<code>-z</code>	Stupeň komprese (1-9, výchozí 9)
<code>-u</code>	Rozbalit, užitečné v systémech, které nativně nepodporují <code>zisofs</code>
<code>-x</code>	Nepřístupovat na jiné oddíly, pouze vytvořit přípojný body
<code>-X</code>	Nepřístupovat na jiné oddíly a přípojný body nevytvářet.

e2compr

Toto je pouze patch ovladače souborového systému `ext2`, který přidává podporu transparentní komprese na úrovni souborů. Do konfiguračního souboru přidává volbu `EXT2_COMPR_FL`. Pro jeho použití není potřeba daný oddíl znovu přeformátovat, protože dokáže zcela transparentně používat jak komprimované, tak nekomprimované soubory. Ovšem jeho vlastnosti nejsou ty nejlepší pro LiveCD a nenašel jsem žádné, které by používalo tento souborový systém.

SquashFS

Phillip Lougher uvolnil v roce 2002 první verzi tohoto komprimovaného filesystému, který odstraňuje nevýhody `cramfs`. První verze byla pro jádro 2.4.19; bohužel až do teď nebyl tento souborový systém do `vanilla` [20] jádra začleněn. Ale GregKH už před časem zařadil patch do `genpatches`, což je patchset pro `gentoo-sources` (výchozí jádro distribuce Gentoo).

V čem se tedy liší od `cramfs`? Preferuje velikost bloku 32kB (od verze 2 64kB), která má mnohem lepší kompresní poměr než 4kB. Jádro nativně podporuje 4kB bloky a tak tento souborový systém dává zbytek dat explicitně do cache. Komprimuje i informace z adresářů a inodů, které, opět pro větší kompresi, shlukuje dohromady. Podporuje 32bitové uid/guid (`cramfs` má uid 16b a guid 8b), velikost souboru je až $2^{32}B$ (`cramfs` 2^{24}). A v neposlední řadě podporuje časové značky (`timestamps`). Pro práci s tímto souborovým systémem je nutné mít program `mksquashfs` z balíku `squashfs-tools`. Ten má poněkud nezvyklou syntaxi:


```
mksquashfs source1 source2 ... dest [options] [-e list of exclude dirs/files]
```

Kde **sourceX** označují zdrojové adresáře, **dest** je cílový soubor, **options** jsou volby a na konci je, uvozen parametrem **-e**, seznam souborů adresářů, jež se mají vynechat.

-all-root , nebo -root-owned	všechny soubory v novém filesystému bude vlastnit uživatel root
-be , nebo -le	explicitně nastav velký (b), nebo malý (l) endián
-ef <exclude_file>	seznam souborů, adresářů, které nebudou v cílovém systému; jeden na každý řádek
-noI , -noD , -noF	nekomprimuj tabulku inodů (I), datové bloky (D), fragmentované bloky (F)

Podpora pro zápis – Úvodem

První LiveCD trpěly tím, že na ně nebylo možné zapisovat. Kvůli technickým omezením to není možné dodnes, ale existují způsoby, jak systém přinutit, aby se tvářil, že to možné je. Princip je velmi jednoduchý; adresář, do něhož chceme zapisovat, je přípojným bodem pro dva jiné, které nazvěme **ro** a **rw**. Zatímco **ro** je, jak název napovídá, jen pro čtení, a tudíž se vyskytuje na disku CD-ROM, adresář **rw** je pouze v operační paměti a obsahuje změny oproti adresáři **ro**. Potom je nutný speciální souborový systém, který dokáže oba adresáře spojit do jednoho.

ovlfs

Prvním řešením byl takzvaný překryvný (overlay) souborový systém ovlfs. Vzhledem k tomu, že je dostupný pouze ve formě patchů pro jádra 2.4, uvádím ho zde především z historických důvodů. Ostatně i nápis *Last Updated: June 10, 2003* na [stránce projektu \[21\]](#) hovoří za vše.

Tento systém dokáže do jednoho přípojného místa připojit dva různé adresáře (souborové systémy). Primární, který je na LiveCD read-only, a sekundární, který bývá v operační paměti. Všechny změny jsou zapisovány na sekundární. Modul **squashfs** si udržuje v paměti jádra vlastní tabulku inodů, odkazů na konkrétní soubory a adresáře. Při požadavku na nový/změněný soubor vrátí odkaz na sekundární adresář. Tento systém se sice vyznačoval některými problémy (zamykání souborů a rychlost), ale vaz mu srazila spíše skutečnost, že nebyl portován na jádra 2.6.

COW

COW neznačí jenom samici tura domácího, ale i označení technologie *copy on write*. Starší implementací bylo [COW-Links \[22\]](#), které není příliš dobré a také už nežije. Poslední patch je pro jádro 2.6.11. Z něj vychází projekt [cowloop \[23\]](#): copy-on-write loop driver. Uvádím je spíše pro zajímavost a rozšíření obzorů, protože ani jeden z nich není bez hacků vhodný pro tvorbu LiveCD. Více vysvětluje [FAQ \[24\]](#) projektu [cowloop](#).

UnionFS

Unionfs znamená stackable unification file system (což by se dalo přeložit jako stohovatelný/vrstvitelný unifikující souborový systém). Mimo jeho funkčnosti je na něm velmi zajímavé to, že je součástí projektu [FiST \[25\]](#) (Stackable File System Language and Templates). Ten používá vysokoúrovňový jazyk specializovaný na tyto souborové systémy. Překladačem **fistgen** je potom proveden překlad na spustitelný modul jádra. Podporovány jsou jádra systémů Linux, FreeBSD a Solaris. Vychází to z unixové tradice tvorby specializovaných jazyků, o kterých se zmiňuje Eric S. Raymond v kapitole Minijazyky, ve své knize [The Art of Unix Programming \[26\]](#).

UnionFS vytváří jeden logický souborový systém, který je tvořen spojením obsahu mnoha různých adresářů (větví), přičemž jejich fyzické umístění je oddělené. Je možné například spojit oddělené

programové adresáře do jednoho a vytvořit systém podobný [GoboLinux](#) [27], ale bez starostí se symbolickými odkazy. Pro LiveCD je velmi podstatné, že dovoluje kombinovat větve jen pro čtení s těmi určenými pro zápis. Díky funkci copy-on-write je možné vytvořit iluzi zapisovatelného souborového systému s tím, že změny se transparentně přenáší do adresáře určeného pro zápis.

První LiveCD, které tento systém používalo na jádře 2.6, bylo [DeadCD](#) [28], alespoň se tak píše na stránkách projektu.

Instalujeme

Nejprve si zkontrolujte, zda nemáte balíček s unionfs přímo [ve své distribuci](#) [29]. Pokud ne, možná vám přijde vhod malý návod. Díky svému původu se modul přeloží a nainstaluje sám.

```
$ wget ftp://ftp.fsl.cs.sunysb.edu/pub/unionfs/unionfs-1.0.14.tar.gz
$ tar -xzf unionfs-1.0.14.tar.gz
```

Doporučuji si přečíst dokumentaci (soubor README) a instalační instrukce (soubor INSTALL). UnionFS má ve výchozím nastavení zapnutý ladící mód, což velmi nepříjemně zvětší velikost modulu (83kB versus přibližně 5MB). Pochopitelně musíte mít hlavičkové soubory jádra v `/lib/modules/‘uname -r’/build/include`. Když chcete instalovat modul pro jiné, než aktuálně běžící jádro, vytvořte soubor `fistdev.mk` a napište do něj `TOPINC=-I/cesta/k/jádro/linux-2.6.xx/include`. Dále je požadováno mít hlavičkové soubory `e2fsprogs` a program `ctags`.

```
$ # vypnutí debug flagu
$ cat fistdev.mk
EXTRACFLAGS=-DUNIONFS_NDEBUG
UNIONFS_DEBUG_CFLAG=
$ make
# make install
# depmod -a
```

Tím máme v systému jaderný modul `unionfs.ko` (pro jádra 2.4 `unionfs.o`), `unionctl` pro přidávání a odebírání větví a manuálové stránky.

Používáme UnionFS

K připojení se používá klasický příkaz `mount`, ale je nutné pomocí volby `-o` specifikovat další parametry.

```
# modprobe unionfs
# mount -w -t unionfs -o dirs=/tmp/foo/=ro:/tmp/bar/=ro unionfs /tmp/union/
```

Spojí adresáře `/tmp/foo` a `/tmp/bar` do jednoho adresáře `/tmp/union`. V případě konfliktů má první adresář největší prioritu. Připojené souborové systémy zjistíme příkazem

```
# mount | grep unionfs
unionfs on /tmp/union type unionfs (rw,dirs=/tmp/foo/=ro:/tmp/bar/=ro)
```

Nebo lépe, příkazem `unionctl`, kterým můžete dynamicky přidávat, nebo odebírat nové přípojné body.

```
# unionctl /tmp/union --list
/tmp/foo (r-)
/tmp/bar (r-)
```

```
# unionctl /tmp/union --add --after /tmp/bar --mode ro /tmp/ham
# unionctl /tmp/union --list
/tmp/foo (r-)
/tmp/bar (r-)
/tmp/ham (r-)
# unionctl /tmp/union --remove /tmp/bar
# unionctl /tmp/union --list
/tmp/foo (r-)
/tmp/ham (r-)
```

Pokud změníte obsah větví, můžete obnovit obsah přípojného bodu příkazem

```
# uniondbg -g /tmp/union/
New generation 3
```

Povolení zápisu

Spojování adresářů jen pro čtení nám v LiveCD nepřinese žádný užitek. Důvodem, proč `unionfs` používáme, je podpora „zápisu“ na read-only médium. Pojďme tedy něco zapsat na disk CD-ROM.

```
# mount /mnt/ro/
# ls -l /mnt/ro/
total 76
-rw-r--r-- 1 root root 18259 Sep 27 21:28 COPYING
-r--r--r-- 1 root root 339 Sep 27 21:28 LICENSE
-rw-r--r-- 1 root root 1406 Sep 27 21:27 abc.ico
...
-rw-r--r-- 1 root root 2546 Sep 27 21:28 requirements.txt
drwxr-xr-x 2 root root 2048 Sep 27 21:27 rootcopy
-rw-r--r-- 1 root root 3926 Sep 27 21:28 start.html
drwxr-xr-x 3 root root 2048 Sep 27 21:28 tools
# ls /mnt/ro/new-file.txt
ls: /mnt/ro/new-file.txt: No such file or directory
# touch /mnt/ro/new-file.txt
touch: cannot touch '/mnt/ro/new-file.txt': Read-only file system
```

V adresáři `/mnt/ro` (read-only) máme připojené CD s distribucí ABC Linux 2005. V adresáři nemáme žádný soubor `new-file.txt` a nejde ani vytvořit.

```
# mount -t unionfs -o dirs=/mnt/ro/=ro unionfs /mnt/cdrom/
# diff -r /mnt/ro/ /mnt/cdrom/
```

Tímto příkazem jsme vytvořili přípojný bod `/mnt/cdrom/`, který je zcela totožný s adresářem `/mnt/ro/`. Ale ani do jednoho z nich nemůžeme zapisovat. Je potřeba ještě určit adresář, kam se budou ukládat změny:

```
# unionctl /mnt/cdrom --add --before /mnt/ro/ --mode rw /mnt/rw/
# unionctl --list /mnt/cdrom/
/mnt/rw (rw)
```



```
/mnt/ro (r-)
```

A voilá, adresář `/mnt/cdrom/` s právem zápisu je k dispozici! Samozřejmě s tím, že se veškeré změny ukládají do adresáře `/mnt/rw`.

```
# echo "burned on CD-ROM :-)" $>$ /mnt/cdrom/new-file.txt
# cat /mnt/cdrom/new-file.txt
burned on CD-ROM :-)
# cat /mnt/rw/new-file.txt
burned on CD-ROM :-)
# cat /mnt/ro/new-file.txt
cat: /mnt/ro/new-file.txt: No such file or directory
# cmp /mnt/cdrom/new-file.txt /mnt/rw/new-file.txt
#
```

Problémy UnionFS

V části instalace bylo nezbytné odstranit ladící flagy z modulů `unionfs`, protože přes vysoké číslo verze trpí `unionfs` některými potížemi a není vhodný pro ostré, produkční nasazení. Občasné kernel oops (nebo kernel panic ve starších verzích) se při použití tohoto souborového systému stávají. Zvláště pokud použijete složitý systém všemožných připojení.

V této fázi byste měli mít dostatek povědomí o živých CD, známe proces bootování, speciální souborové systémy, přesto ještě něco chybí. Nevíme, co je nutné udělat, abychom proměnili obyčejný systém v takový, který je schopen bootovat z CD.

Linux live, před spuštěním systému – Úvodem

Příkladem distribucí, které jsou vytvořeny pomocí těchto skriptů [linux-live](#) [30], jsou [Slax](#) [31] (pochopitelně), [ABC Linux 2005](#) [32] (odvozena ze Slaxe) a [minimax](#) [33] (používá upravenou verzi, jinak vychází z distribuce Arch Linux). Jejich použití je velmi snadné a na stránce je také velmi stručný návod:

Just install your favourite distro, remove all unnecessary files (for example man pages and all other files which are not important for you) and then download and run these scripts.

Jednoduše nainstalujte vaši oblíbenou distribuci, odstraňte všechny nepotřebné soubory (například manuálové stránky a ostatní soubory, které vás nezajímají) a potom si stáhněte a spusťte tyto skripty. Ale za takhle krátký článek by mě v redakci asi nepochválili. Proto se jim v následujících dílech podíváme na zoubek a zároveň budeme pokračovat v objevování tajů LiveCD.

initrd

V prvním díle [LiveCD - 1 \(úvod, isolinux\)](#) [34] tohoto seriálu jsme se seznámili se zavaděčem `isolinux`, který umí spustit jádro z CD. Klasický postup bootování jádra je takový, že se

1. Zavede do paměti a spustí.
2. Provede se inicializace potřebného hardwaru.
3. Není-li určeno parametrem `init=` jinak, jádro spustí program `/sbin/init`.

Jenže u LiveCD to tak jednoduše nejde, protože to musí běžet bez načítání potřebných programů z disku. Stejný problém nastává v situaci, kdy máme distribuční jádro, které obsahuje podporu pro souborové systémy na disku...který nedokáže bez těch modulů přečíst. Tento problém se nazývá `bootstrapping` a je to problém, který se netýká jen počítačů, ale kosmologie a filozofie. Naštěstí vývojáři jádra přišli s jednoduchým řešením, které se jmenuje `initrd` (Initial Ram Disk), což je

speciální soubor, který se zavede do paměti a jádro z něho při bootu načte potřebné soubory. Jádro musí ovšem obsahovat patřičnou podporu:

```
Device Drivers > Block devices:  
<*> RAM disk support  
(4096) Default RAM disk size (kbytes) (NEW)  
[*] Initial RAM disk (initrd) support
```

V případě klasických distribucí bývá součástí `initrd` ovladač pro souborové systémy a případně také `bootsplash`. Live systémy tu mají uložen minimální systém, který nejprve připraví všechno potřebné pro spuštění systému a teprve potom spustí samotný `/sbin/init`. Ovšem ramdisky mají omezenou kapacitu, takže není snadné do nich dát byť jen minimální systém, který by obsahoval klasické GNU nástroje, na něž jsme zvyklí.

Busybox

Proto byl vytvořen nástroj `busybox` [35], což je *The Swiss Army Knife of Embedded Linux* (a v té chvíli si pokaždé vzpomenu na jiný švýcarský nůž, program `netcat` – *the network swiss army knife*). Jedná se o kolekci nejpoužívanějších unixových nástrojů, které jsou součástí jednoho spustitelného souboru. Plná instalace zahrnuje více než 190 programů, ale není pochopitelně potřeba překládat všechny nástroje. Lze jej přeložit s knihovnou `glibc`, ale i s menší `uClibc`, což je vhodné především pro zabudované systémy. Volat binární soubory `busyboxu` je možné dvěma způsoby.

Buď explicitně napsat `busybox příkaz [parametry]`, anebo vytvořit soustavu odkazů a `busybox` podle názvu souboru pozná, který program má spustit. Je nutné mít na paměti, že GNU nástroje, které denně používáme, mají oproti těm z `busyboxu` více možností.

Po spuštění jádra

Jak jsme si již řekli, v `initrd` je umístěn malý systém, který má za úkol připravit systém tak, aby mohl běžet z CD. Jenže konkrétní činnosti závisí na konkrétním řešení, a proto bude nejlepší tento proces ukázat na příkladu distribuce `Slax`, respektive skriptů `linux-live`, s jejichž pomocí je vytvářena. V prvním díle byl uveden příklad konfigurace zavaděče `isolinux`. Pokud si ji nepamatujete, tak zde je malé připomenutí:

```
label linux  
kernel boot/vmlinuz  
append max_loop=255 initrd=boot/initrd.gz init=linuxrc load_ramdisk=1  
prompt_ramdisk=0 ramdisk_size=4444 root=/dev/ram0 rw
```

Přímo ze zadaných parametrů můžeme vyčíst sekvenci kroků, které jádro po načtení zavaděčem provede. Rozbalí definovaný `initrd` do paměti a protože je explicitně určen jako kořenový adresář zařízení `/dev/ram0` (což je rozbalený `initrd`), spustí program definovaný parametrem `init`. Ten se podle konvence nazývá `linuxrc` a právě on je zodpovědný za správné nastavení systému před jeho naboootováním.

Pokud vás zajímá, jak probíhá prebootovací nastavení systému, tak máte dvě možnosti. Můžete si spustit `Slax` nebo `ABC Linux 2005` s parametrem `debug`. Ten způsobí, že se po každém kroku systém zastaví a dostanete se do shellu, což je `ash` z `busyboxu`. Druhou možností je zkopírovat si soubor `boot/initrd.gz`, rozbalit, připojit a obsah `linuxrc` si přečíst sám, protože se jedná o shellový skript.

Linuxrc krok za krokem

Skript `linuxrc`, které používá sada nástrojů `linux-live` potřebuje knihovnu funkcí zvanou `liblinuxlive`, která se používá i ve spoustě skriptů distribuce Slax. Proto prvním zajímavým příkazem je:

```
. liblinuxlive
```

Následuje připojení základních adresářů `/proc` a `/sys`, které jsou důležité pro jádro.

```
mount -t proc proc /proc
mount -t sysfs sysfs /sys
```

Dále jsou definovány důležité adresáře a proměnné:

```
UNION=/union
MEMORY=/memory
MOUNTDIR=mnt
CHANGES=$MEMORY/changes
COPY2RAM=$MEMORY/copy2ram
IMAGES=$MEMORY/images
INITRAMDISK=$MOUNTDIR/live
```

Skript natáhne základní moduly, které jsou nutné pro úspěšný start systému, což zařídí knihovná funkce `modprobe_essential_modules`. Ta natáhne moduly `loop`, `isofs`, `squashfs`, `unionfs`, `vfat`, `ntfs` a moduly `nls`, Native Language support, pro kódové stránky cp437, iso-8859-1 a iso-8859-2. Poslední akcí je vytvoření blokových zařízení funkcí `create_block_devices`. Jejich seznam je v adresářích `/sys/block/*/dev` a `/sys/block/*/*/dev`. Někdy je nutné dopředu nahrát i USB moduly (týká se to třeba USB klávesnice), proto se, v případě zadání bootovacího parametru `probeusb`, zavedou moduly `ehci-hcd`, `ohci-hcd`, `uhci-hcd` a `usb-storage` a znovu dojde k vytvoření případných nových blokových zařízení.

```
mount -t unionfs -o debug=0,dirs=$CHANGES=rw unionfs $UNION
```

Mimo tvorby důležitých adresářů se připojí adresář `/memory/changes` jako read-write větev pro hlavní adresář. Pokud jste ještě nezapomněli obsah minulého dílu, tak jistě tušíte, že tento příkaz zajistí, že se budou zapisovat změny vůči read-only adresáři. Pokud jste si prohlíželi obsah CD, možná jste objevili soubor `livecd.sgn` s textem

```
All available discs and CDRoms are mounted during the boot process.
When done, linuxrc script is looking for this livecd.sgn file.
It tells linuxrc where to mount the live filesystem from.
```

```
Don't delete this file, else your LiveCD won't work
```

Skript postupně prochází a připojuje všechny disky a oddíly, přičemž hledá, zda tento soubor existuje. Pokud ne, zkusí ještě zavést ovladače USB a toto hledání zopakovat. Pokud ani teď nic nenalezne, skončí boot chybovým hlášením. Tato funkce nastaví proměnnou `$DATA` na adresář `/base` na CD.

```
if [ "'cmdline_parameter toram'" != "" -o
    "'cmdline_parameter copy2ram'" != "" ]; then
```

```

echolog "copying modules to RAM, this may take some time"
copy_to_ram $DATA $COPY2RAM
cd_autoeject 1
umount $DATA 2>/dev/null
if [ $? -ne 0 ]; then umount `dirname $DATA` 2>/dev/null; fi
DATA=$COPY2RAM
cd_autoeject 0
fi

```

Ano, právě tento kód zajišťuje funkci *copy2ram*, díky které je obsah celého CD zkopírován do operační paměti a práce s ním je velmi rychlá (nemluvě o uvolněném CD). Jak je vidět, hlavní zásluhy na tom má funkce `copy_to_ram`, která zkopíruje obsah datového adresáře `$DATA` i všechny dostupné moduly do paměti. Bohužel nijak nekontroluje množství dostupné paměti, takže pokud jí nemáte dostatek, zkopírování skončí chybou. Dalším krokem je vložení modulů do souborového systému. To sestává ze dvou fází.

```

# pripojeni squashfs obrazu
mount -t squashfs -o loop,ro cesta_k_modulu pripojny_bod
echo "cesta_k_modulu pripojny_bod" >> /tmp/_mounts

# vlozeni pripojneho bodu do unionfs
unionctl pripojny_bod --add --after 0 --mode ro absolutni_cesta

```

Nejprve musíme někam připojit squashfs obraz a potom přidat to „někam“ do struktury unionfs (všimněte si, že to připojujeme jako `ro`). Potom následuje prosté zkopírování adresáře `/root-changes`. Velmi zajímavou funkcí je automatické vytváření souboru `/etc/fstab`, bohužel je funkce `activate_fstab` příliš rozsáhlá na to, abych jí sem napsal celou. Proto napíši pouze jeho jednotlivé kroky:

1. Tyto záznamy

```

tmpfs / tmpfs defaults 0 0
devpts /dev/pts devpts gid=5,mode=620 0 0
proc /proc proc defaults 0 0

```

se zapíše vždycky.

- 2.

Pro všechna zařízení z `/proc/sys/dev/cdrom/info` se vloží řádek

```

/dev/jednotka /mnt/jednotka_cdrom iso9660 noauto,users,exec 0 0

```

kde jednotka je název zařízení z `/dev` (například `hdc`).

3. Nastavení diskových oddílů je složitější. Jejich seznam je v `/proc/partitions`, přičemž se navíc kontroluje v přítomnost souboru `/sys/block/jednotka/removable`, který označuje výměnnou jednotku (připojují se do adresáře `/mnt/jednotka_removable`). Oddíl se připojí do dočasněho adresáře a ze souboru `/proc/mounts` se zjistí systém souborů a do `/etc/fstab` se zapíše:

```

/dev/jednotka /mnt/jednotka souborovy_system auto,users,suid,dev,exec 0 0

```

4. Za pomoci `fdisk -l` se zjistí swapovací oddíl a příslušný řádek je

```
/dev/jednotka swap swap defaults 0 0
```

5. Posledním krokem je explicitní vytvoření záznamu pro disketovou mechaniku

```
/dev/fd0 /mnt/floppy auto noauto,users,suid,dev,exec 0 0
```

Následuje vytvoření některých důležitých adresářů (`dev`, `proc`, `sys` a `tmp`) a vše se připravuje k velkému finále, `chroot` nutím do paměti spuštěním programu `/sbin/init`. Vzniklou adresářovou strukturu si můžete prohlédnout na unionfs.org [36] (úplně dole).

chroot a spuštění /sbin/init

Po provedení všech potřebných úkonů se schyluje k zahájení vlastního spuštění distribuce.

```
echolog "changing root directory..."
cd $UNION
mkdir -p $INITRAMDISK

umount /sys # we won't need it anymore
if [ ! -e $UNION/dev/console ]; then mknod $UNION/dev/console c 5 1; fi

if [ -x $UNION/usr/sbin/chroot ];
then CHROOT=/usr/sbin/chroot
else CHROOT=/usr/bin/chroot
fi

header "linux live end"
```

Systém změní pracovní adresář na `$UNION` (`/union`) a vytvoří adresář `/mnt/live`, pomocí něhož máme později přístup k modulům a podobně. Odpojí `/sys` a pokud neexistuje zařízení `console`, je vytvořena. Potom se již nastaví cesta k programu `chroot` a vypíše se hlášení o dokončení skriptu `linux live`.

```
pivot_root . $INITRAMDISK
exec $CHROOT . sbin/init dev/console 2>&1; fi

header "ERROR!"
echolog "You are not supposed to be here, something went wrong!"
echolog "Even Ctrl+Alt+Del won't help you in kernel panic."
```

Nejprve je příkazem `pivot_root` změněn kořenový adresář z `/union` na `/union/mnt/live` a zbytek systému je spuštěn v `chrootu` (s kořenovým adresářem umístěným v `/union`). Důležité je, že je vstup i výstup procesu `/sbin/init` přesměrovnán na konzoli!

Závěrem

Dnešní díl nás zavedl do tajů nastavení systému před vlastním startem. Podotýkám, že uvedená metoda je specifická pouze pro distribuce, které používají skripty `linux live`, ostatní se mohou chovat

úplně jinak. Příští díl by měl být přece jenom praktičtější a bude se týkat distribuce Slax a možnostmi jeho úprav.

Úprava distribuce Slax – Úvodem

Myslím, že po přečtení minulých dílů problematiku LiveCD znáte po teoretické stránce. Ovšem co je důležitější, jsou praktické vlastnosti. Proto jsem zařadil díl, v němž nebudete poznávat nové souborové systémy, nebo do detailu zkoumat ten, či onen aspekt nějakého spouštěcího skriptu. Tento díl by vás měl přesvědčit, že úprava Slaxe je velmi jednoduchá a můžete si jej velmi snadno a rychle uzpůsobit k obrazu svému. Způsobů, jak upravit Slax, je několik:

1. Změna některého souboru – díky existenci adresáře `rootcopy` můžete snadno změnit několik souborů.
2. Moduly – také si můžete stáhnout nebo vytvořit rozšiřující moduly, a doplnit tak softwarové vybavení.
3. Skripty Linux Live – s nimi jsme se seznámili už v minulém díle, a pokud chceme provádět opravdu zásadní změny, nebo dokonce vytvořit vlastní distribuci, nezbude nám, než využít jejich služeb.

Adresář rootcopy

Pokud si vypíšete seznam souborů v kořenovém adresáři CD, dostanete podobný výpis:

```
-rw-r--r-- 1 root root 18259 Sep 27 21:28 COPYING
-r--r--r-- 1 root root   339 Sep 27 21:28 LICENSE
-rw-r--r-- 1 root root  1406 Sep 27 21:27 abc.ico
...
-rw-r--r-- 1 root root  2546 Sep 27 21:28 requirements.txt
drwxr-xr-x 2 root root  2048 Sep 27 21:27 rootcopy
-rw-r--r-- 1 root root  3926 Sep 27 21:28 start.html
drwxr-xr-x 3 root root  2048 Sep 27 21:28 tools
```

Pokud chceme změnit jeden nebo několik málo souborů, je nejrychlejším způsobem je nahrát přímo do zmíněného adresáře `rootcopy`. Přijdeme tím sice o kompresi, kterou zajišťuje `squashfs`, ale pro malé soubory to nevadí. Vlastní překopírování obsahu zařídí startovací skript `linuxrc` (viz LiveCD – 4 (linux live, před spuštěním systému)). Například chceme-li změnit uvítací zprávu, není nic snazšího, než napsat:

```
mkdir -p /cesta/k/cd/rootcopy/etc
cp uvitaci_zprava /cesta/k/cd/rootcopy/etc/issue
```

Vytvořit ISO a vypálit. Je na místě podotknout, že tyto soubory jsou omezeny možnostmi souborového systému ISO 9660.

Skript makeiso.sh

Tomáš Matějčík napsal skript, který velmi usnadňuje tvorbu ISO. A aby vám to nebylo líto, tak jeho podstatné části okomentuji.

```
CDLABEL=""
```

Hned na začátek nejdůležitější věc, jméno CD. Pokud nechcete, aby šlo ve vypalovačce poznat, že jde o Slax, zde můžete definovat jméno nové. Doporučuji něco úderného, jako například `ABC_Linux_2005`.


```
# isolinux.bin is changed during the ISO creation,  
# so we need to restore it from backup.  
cp -f boot/isolinux.bi_ boot/isolinux.bin
```

Jak napovídá komentář, soubor `isolinux.bin` je během procesu změněn. Proto je na začátku mazané obnoven ze zálohy a vám doporučuji jej z adresáře `boot` nemazat. Pokud nevíte, co je to za soubor, pak si přečtete první díl s názvem LiveCD – 1 (úvod, isolinux).

```
mkisofs -o "$1" -v -J -R -D -A "$CDLABEL" -V "$CDLABEL" \\  
26 -no-emul-boot -boot-info-table -boot-load-size 4 \\  
27 -b boot/isolinux.bin -c boot/isolinux.boot .
```

Posledním a nejdůležitějším příkazem je `mkisofs`, který vytvoří bootovatelné CD. Pokud se vám zdá příkaz povědomý, pak máte dobrou paměť, protože byl uveden v prvním díle. Použití je pak velmi prosté:

```
./mkisofs kam/se/ma/ulozit/iso
```

Pro ty, kteří nemají zrovna po ruce Linux a pracují na Windows, je k dispozici ještě soubor `make_iso.bat` se stejnou funkcí. Jen pro zajímavost, ISO se vytvoří příkazem:

```
tools\DOS\mkisofs.exe @tools\DOS\config -o "%1" -A "%CDLABEL%" -V "%CDLABEL%" .
```

Moduly

Moduly jsou základem celého Slaxe, podobně jako je vše ve Smalltalku objekt, nebo vše v Lispu seznam, pak se dá s trochou nadsázky říct, že vše v této distribuci je modul. Pokud mi nevěříte, schválně si vypíšte obsah adresáře `base`:

```
ls -lh base/  
celkem 178M  
-rw----- 1 root root 12M led 27 21:24 01_kernel.mo  
-rw----- 1 root root 31M led 27 21:24 02_core.mo  
-rw----- 1 root root 21M led 27 21:24 03_xwindow.mo  
-rw----- 1 root root 8,0M led 27 21:24 04_xap_libs.mo  
-rw----- 1 root root 18M led 27 21:24 05_common.mo  
-rw----- 1 root root 37M led 27 21:24 06_kde_base.mo  
-rw----- 1 root root 20M led 27 21:24 07_kde_apps.mo  
-rw----- 1 root root 20M led 27 21:24 08_kde_office.mo  
-rw----- 1 root root 13M led 27 21:24 09_slax.mo
```

V něm jsou uloženy základní moduly tvořící systém.

Co je to vlastně modul

Což je poměrně logická otázka. Jindy všeznalý příkaz `file` mnoho nenapoví:

```
base/01_kernel.mo: writable, regular file, no read permission
```

Pochopitelně nemáme práva pro čtení, a proto je `file` němý. Přihlášením jako `root` zjišťujeme více.

```
01_kernel.mo: Squashfs filesystem, little endian, version 3.0, 0 bytes, \\
1835 inodes, blocksize: 65536 bytes, created: Fri Jan 27 20:55:47 2006
```

Moduly jsou tedy bloky dat komprimované souborovým systémem Squashfs (viz LiveCD – 2 (komprimované souborové systémy)). Proto hned připojíme:

```
# nezapomen, ze adresar . i .. jsou na pripojenem obrazu CD,
# tudiz jsou read-only
mount -o loop -t squashfs 01_kernel.mo ../../01_kernel
mount: wrong fs type, bad option, bad superblock on /dev/loop2,
missing codepage or other error
In some cases useful info is found in syslog - try
dmesg | tail or so
```

A možná dopadnete jako já, protože mám v jádře starší verzi 2.2, což potvrdil i `dmesg`:

```
Squashfs 2.2 (released 2005/07/03) (C) 2002-2005 Phillip Lougher
SQUASHFS error: Major/Minor mismatch, filesystem is (3:0),
I~support (1 : x) or (2 : <= 1
```

Potom nám nezbyde, než si stáhnout patch ze stránek projektu [37], aplikovat ho (více detailů je v souboru `INSTALL`) a přeložit jádro.

```
cd /usr/src/linux-2.{4,6}.x
patch -p1 < location-of-squashfs/linux-2.{4,6}.x/squashfs3.0-patch
```

Potom nám již nic nebrání se podívat do obsahu modulu:

```
ls ../../01_kernel/
boot etc lib sbin usr var
```

Jak jsem již tedy uvedl, modul je jednoduše základ celé distribuce a díky kompresi pomocí Squashfs můžete nosit na 200MB mini CD kompletní systém, včetně KDE a KOffice. Naprosto stejně je zařízena funkce ukládání uživatelského nastavení, kdy se změny ukládají do zapisovací větve (viz LiveCD – 3 (podpora pro zápis)) a nakonec jsou zabaleny do modulu, který je při následném startu přehrán a jeho obsah nahradí staší soubory na CD.

Rozšiřujeme Slax pomocí modulů

Na stránkách `Slaxe` [38] naleznete seznam `modulů` [39], které si můžeme stáhnout. Pro mě bývá nutností doinstalovat `modul pro Vim` [40], na který jsem si úspěšně vybudoval závislost. V tomto okamžiku máme na výběr dva adresáře, kam modul nahrát:

- `modules/` – obsah tohoto adresáře se nahraje při startu automaticky
- `optional/` – tyto moduly se nahrají, pouze pokud si je zvolíme při startu jako parametr (např. `slax load=vim`).

A potom již výše popsaným způsobem vytvoříme ISO a vypálíme.

Slax také poskytuje nástroje, jak nahrát modul za běhu. Příkazem `uselivmod cesta/k/modulu` dojde k dynamickému nahrání. Ale stává se, že systém potom úplně vytuhne, takže je lepší se této možnosti vyhnout. Alternativou je kliknutí na modul v Konqueroru, což má stejný efekt (včetně možného zatuhnutí).

Moduly se díky souborovému systému `unionfs` dynamicky připojí do systému, bohužel je smutnou skutečností, že právě on bývá nejslabším článkem celého systému a častokrát způsobuje záhadné chyby v systému, anebo přivodí i jeho pád.

Závěrem

Dnešní díl obsahoval velmi málo teorie, ale za to byl, doufám, nacpán až k prasknutí informacemi. I když popis malého skriptu `makeiso.sh` jsem si nemohl nechat ujít. Zároveň byl tento díl (alespoň pro mě jako pro autora) zajímavý v tom, že se odkazoval na znalosti získané ve všech čtyřech předchozích dílech. Teď už víte, jak modifikovat ISO distribuce Slax a také, že se nejedná o nic složitého. Naopak, myslím, že je to velice snadné. Jediné, co vám může malinko ztěžovat situaci, je rozhodnutí, do kterého adresáře to mám vlastně nakopírovat. V příštím díle se zaměříme na další praktickou a užitečnou dovednost pro váš spokojenější život se Slaxem, a tou je vytváření vlastních modulů.

Odkazy

- [1] <http://www.abclinuxu.cz/slovník/boot>
- [2] <http://www.abclinuxu.cz/projekty/abclinux/verze-2005>
- [3] http://belenix.sarovar.org/belenix_home.html
- [4] <http://www.reactos.org/xhtml/en/download.html>
- [5] <http://people.debian.org/~neal/hurd-live-cd/>
- [6] <http://www.charlessoft.com/>
- [7] <http://www.nu2.nu/pebuilder/>
- [8] <http://www.abclinuxu.cz/clanky/recenze/exkurze-do-sveta-live-distribuci>
- [9] <http://www.abclinuxu.cz/slovník/zavadek>
- [10] <http://www.abclinuxu.cz/clanky/system/openboot-prompt>
- [11] <http://www.abclinuxu.cz/slovník/lilo>
- [12] <http://www.abclinuxu.cz/slovník/grub>
- [13] <http://www.abclinuxu.cz/clanky/navody/jak-nabootovat-linux-po-siti>
- [14] <http://www.abclinuxu.cz/clanky/system/adresar-proc-brana-do-jadra>
- [15] <http://www.abclinuxu.cz/clanky/system/sysfs-brana-do-jadra>
- [16] <http://www.abclinuxu.cz/slovník/fuse>
- [17] <http://www.abclinuxu.cz/slovník/plan9>
- [18] <http://gmail.google.com>
- [19] <http://www.abclinuxu.cz/slovník/zlib>
- [20] <http://www.abclinuxu.cz/slovník/vanilla>
- [21] <http://ovlfs.sourceforge.net>
- [22] <http://wohnheim.fh-wedel.de/~joern/cowlink/>
- [23] <http://www.atconsultancy.nl/cowloop/>
- [24] <http://www.atconsultancy.nl/cowloop/faq.html>
- [25] <http://www.filesystems.org/>
- [26] <http://www.faqs.org/docs/artu/minilanguageschapter.html>
- [27] <http://www.abclinuxu.cz/clanky/ruzne/zajimavosti-ze-sveta-linuxu>
- [28] <http://www.deadcd.org>
- [29] <http://www.gentoo-portage.com/sys-fs/unionfs>
- [30] <http://www.linux-live.org>
- [31] <http://www.slax.org>
- [32] <http://www.abclinuxu.cz/projekty/abclinux/verze-2005>
- [33] <http://www.kotek.net/minimax/>
- [34] <http://www.abclinuxu.cz/clanky/system/livecd-1-uvod-isolinux>
- [35] <http://www.busybox.net>
- [36] <http://www.unionfs.org>
- [37] <http://squashfs.sourceforge.net/>
- [38] <http://www.slax.org>
- [39] <http://slax.linux-live.org/modules.php>
- [40] <http://slax.linux-live.org/modules.php?category=console&id=32&name=Vim>

Konfigurace multimediální klávesnice v X

Jiří Mlíka

V následujících několika řádcích přiblížím konfiguraci multimediálních (a jinak vylepšených) klávesnic způsobem, který považuji za systémový: mapování keycodů na symboly.

Úvod

Konfigurace „přídavných“ tlačítek na klávesnici je evergreen, který se táhne různými články a návody minimálně od doby, kdy na běžné PC klávesnice přibýly tři podivné klávesy (dvě s logem jakéhosi operačního systému, třetí s obrázkem menu). Pamatuji si své nadšení, když se mi po stisku jedné z takových záhadných kláves rozbalilo hlavní menu v KDE. Od těchto dávných dob přibýla na klávesnice spousta dalších tlačítek pro ovládání médií, webového prohlížeče, případně pro spouštění kancelářských programů. Nechat je bez využití by bylo ostudou pro každého správného linuxového geeka [1].

Svoje první experimenty s rozšiřujícími klávesami jsem prováděl podle výborného článku [Multimediální a jinak vylepšené klávesnice](#) [2] pana [Zdeňka Štěpánka](#) [3]. S dosaženými úspěchy jsem byl plně spokojen. Existuje však způsob, jak celou věc provést elegantněji a uživatelsky přívětivěji. Dokonce je zde možnost přispět do [X.Org](#) [4] i pro neprogramátory, a ulehčit tak život technicky méně zdatným uživatelům.

Přiznám se, že nejsem odborníkem na [X keyboard extension \(xkb\)](#) [5], nicméně metodou pokus-omyl jsem dosáhl úspěchu, o který bych se rád podělil se čtenáři [abclinuxu.cz](#). Předpokládám, že experti na tuto problematiku neopomenou v diskusi pod článkem opravit všechny moje diletantské chyby v terminologii a vůbec v chápání celého systému.

Řešení

Řešení spočívá v mapování tzv. keycodes na symboly (keysym). Když si odpustím nudnou teorii o scankódech a nebudu se zatěžovat detaily práce s klávesnicí v XFree86/X.Org a jeho subsystémech, mohu drze prohlásit, že keycode je hodnota vygenerovaná při stisku klávesy. Tato hodnota je předmětem dalšího zpracování, na jehož výstupu je symbol (keysym), který je poslán aplikaci. Sestavením tabulky mapující keycody na symboly dosáhneme podpory pro rozšiřující klávesy.

Ti, kdo četli zmíněný [článek](#) [6], nebo jsou jinak poučeni o celé problematice, vědí, že keycody jednotlivých rozšiřujících kláves zjistíme programem `xev`. Symboly pro „multimediální“ klávesnice nalezneme v souboru `/usr/X11R6/lib/X11/XKeysymDB`, jak radí v diskusi pod [článkem](#) [7] [Petr Ferschmann](#) [8]. Já jsem svoje experimenty prováděl s klávesnicí [Genius Twintouch Optical Office](#) [9]. Popiskám kláves na této klávesnici odpovídají následující symboly:

```
XF86Search,  
XF86AudioMedia,  
XF86WWW,  
XF86AudioPrev,  
XF86AudioNext,  
XF86AudioMute,  
XF86Calculator,  
XF86AudioPlay,
```

```
XF86AudioPause,  
XF86AudioStop,  
XF86AudioLowerVolume,  
XF86AudioRaiseVolume,  
XF86Sleep,  
XF86Favorites,  
XF86Refresh,  
XF86Stop,  
XF86Forward,  
XF86Back,  
XF86Mail,  
XF86MyComputer
```

Mapování keycodů na symboly najdeme v adresáři `/etc/X11/xkb/symbols`. Mapování pro „multi-mediální“ klávesnice pak v souboru `/etc/X11/xkb/symbols/inet`. Pro svoji klávesnici jsem doplnil následující sekci:

```
...  
// Genius Wireless Twintouch Optical Office  
partial alphanumeric_keys  
xkb_symbols "geniustwintouch" {  
  
    key <FK17> { [ XF86Search ] };  
    key <I01> { [ XF86AudioMedia ] };  
    key <I02> { [ XF86WWW ] };  
    key <I10> { [ XF86AudioPrev ] };  
    key <I19> { [ XF86AudioNext ] };  
    key <I20> { [ XF86AudioMute ] };  
    key <I21> { [ XF86Calculator ] };  
    key <I22> { [ XF86AudioPlay, XF86AudioPause ] };  
    key <I24> { [ XF86AudioStop ] };  
    key <I2E> { [ XF86AudioLowerVolume ] };  
    key <I30> { [ XF86AudioRaiseVolume ] };  
    key <I5F> { [ XF86Sleep ] };  
    key <I66> { [ XF86Favorites ] };  
    key <I67> { [ XF86Refresh ] };  
    key <I68> { [ XF86Stop ] };  
    key <I69> { [ XF86Forward ] };  
    key <I6A> { [ XF86Back ] };  
    key <I6C> { [ XF86Mail ] };  
    key <K67> { [ XF86MyComputer ] };  
  
};  
...
```

Keycodey se zapisují ve formě vyjadřující jejich relativní pozici na klávesnici (podivné kódy v ostrých závorkách za klíčovým slovem `key`). Alespoň tak tvrdí popis v souboru `/etc/X11/xkb/README.enhancing`. To je poměrně nepříjemné, protože program `xev` vrací v položce označené `keycode` jinou hodnotu. Po troše pátrání jsem přišel na to, že hodnoty vrácené programem `xev` jsou mapovány na kýžené keycodey (resp. formát vhodný pro mapování na symboly) v souboru `/etc/X11/xkb/keycodes/xfree86`. Všimněte si položky `xkb_symbols`.

Její hodnotu (v mém případě jsem si vymyslel `geniustwintouch`) použijeme jako identifikátor modelu klávesnice v souboru `/etc/X11/XF86Config` (položka `XkbModel`).

```
...
Section "InputDevice"
    Option      "XkbModel" "geniustwintouch"
    Identifier  "Keyboard1"
    Driver      "keyboard"
    Option      "XkbLayout" "cz"
    Option      "XkbOptions" "compose:rwin"
EndSection
...
```

Než tak učiníme, musíme ještě hodnotu `xkb_symbols` zapsat do `/etc/X11/xkb/rules/xfree86` (resp. `/etc/X11/xkb/rules/xorg`).

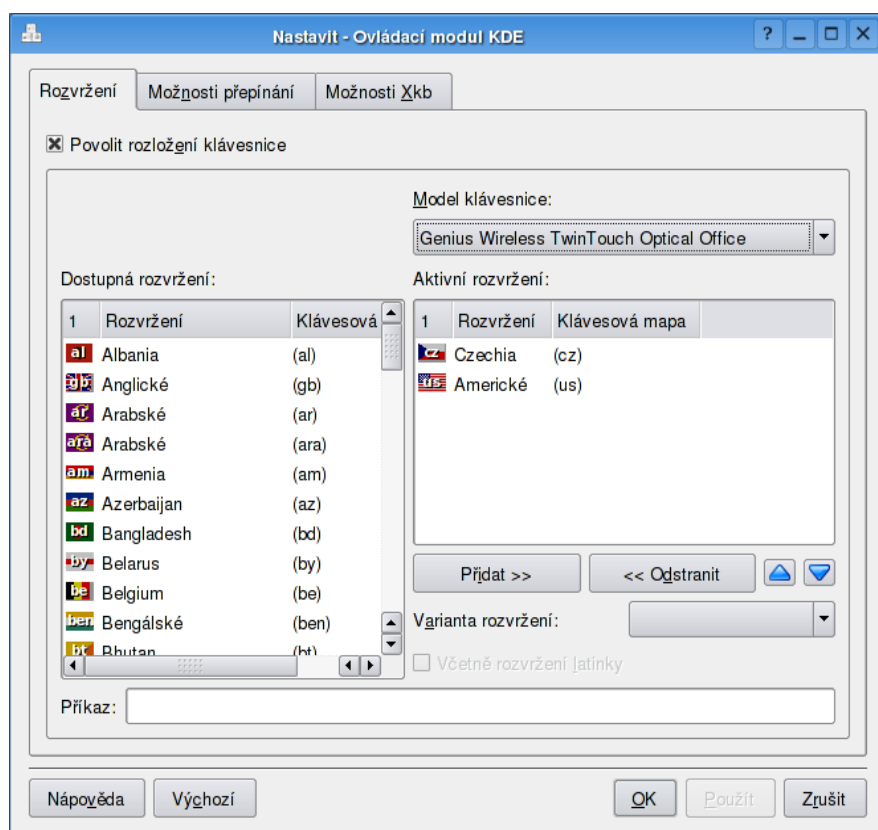
```
...
! $inetkbds = a4techKB21 a4techKBS8 acer_tm_800 acpi airkey azonaRF2300 \\
brother \\
btc5113rf btc5126t btc9000 btc9000a btc9001ah btc5090\\
cherryblue cherrybluea cherryblueb \\
chicony chicony9885 \\
compaqeak8 compaqik7 compaqik13 compaqik18 cymotionlinux \\
armada presario ipaq \\
dell inspiron dtk2000 \\
dexxa diamond genius geniuscomfy2 \\textit{geniustwintouch} \\
ennyah_dkb1008 gyration \\
hpi6 hp2501 hp2505 hp5181 hpxe3gc hpxe3gf hpxe4xxx hpzt11xx \\
hp500fa hp5xx hp5185 \\
honeywell_euroboard \\
rapidaccess rapidaccess2 rapidaccess2a \\
ltcd logiaccess logicdp logicdpa logicdit logicink logiciink \\
logiinkse logiinkseusb logiitc logiik itouch logiultrax \\
mx1998 mx2500 mx2750 \\
microsoftinet microsoftpro microsoftprousb microsoftproem microsoftprose \\
microsoftoffice microsoftmult \\
oretec \\
propeller scorpius \\
qtronix \\
samsung4500 samsung4510 \\
```

```
sk1300 sk2500 sk6200 sk7100 sp_inet \\
sven symplon toshiba_s3000 trust trustda yahoo
...
```

Nyní můžeme začít používat rozšiřující klávesy v aplikacích.

Snadná konfigurovatelnost

Chceme-li splnit předsevzetí maximální uživatelské přívětivosti, měli bychom umožnit uživateli, který nic netuší o nějakém souboru `/etc/X11/XF86Config`, aby si „naklikal“ model klávesnice například přímo v KDE (viz obrázek).



K tomu však musíme ještě upravit soubor `/etc/X11/xkb/rules/xfree86.lst` (resp. `/etc/X11/xkb/rules/xorg.lst`).

```
...
dtk2000          DTK2000
ennyah_dkb1008  Ennyah DKB-1008
genius           Genius Comfy KB-16M / Genius MM Keyboard KWD-910
geniuscomfy2    Genius Comfy KB-21e-Scroll
geniustwintouch Genius Wireless TwinTouch Optical Office
gyration        Gyration
hpi6            Hewlett-Packard Internet Keyboard
hp2501          Hewlett-Packard SK-2501 Multimedia Keyboard
hp2505          Hewlett-Packard SK-2505 Internet Keyboard
...
```

Závěr

Pokud někdo očekával, že se z článku dozví, jak celá věc funguje, pravděpodobně jsem jej zklamal. Musím se přiznat, že nevím. Nicméně jsem přišel na fungující postup, jak celou věc nakonfigurovat. Věřím, že není daleko doba, kdy se tímto postupem nebudu muset zabývat. Nově zakoupenou super-moderní klávesnici s řadou tlačítek na ovládání všeho možného si prostě vyberu v nějakém grafickém konfiguračním nástroji. Svoji úpravu souboru `/etc/X11/xkb/symbols/inet` zasílám autorům X.Org. Majitelé multimediálních klávesnic celého světa, spojte se!

Odkazy

- [1] <http://en.wikipedia.org/wiki/Geek>
- [2] <http://www.abclinuxu.cz/clanky/hardware/multimedialni-a-jinak-vylepsene-klavesnice>
- [3] <http://www.abclinuxu.cz/Profile/2431>
- [4] http://en.wikipedia.org/wiki/X.Org_Foundation
- [5] <http://en.wikipedia.org/wiki/XKB>
- [6] <http://www.abclinuxu.cz/clanky/hardware/multimedialni-a-jinak-vylepsene-klavesnice>
- [7] <http://www.abclinuxu.cz/clanky/hardware/multimedialni-a-jinak-vylepsene-klavesnice>
- [8] <http://www.abclinuxu.cz/Profile/188>
- [9] <http://www.geniusnet.com.tw/product/product-1.asp?pdtno=52>

Rozhovor: Česko se stává linuxovou mocností

Leoš Literák

Víte, že se Česko stává důležitým hráčem ve vývoji linuxových distribucí? Vyzpovídali jsme představitele SUSE i RedHatu, kteří zde mají svá vývojová centra.

Před několika lety založila německá distribuce SUSE v Praze svou pobočku. Nyní do České republiky přišel i americký Red Hat. Při této příležitosti jsme vyzpovídali jejich představitele.

★ **Otázka:** Kolik vývojářů Linuxu zaměstnáváte celosvětově?

▷ **Odpověď:** Michal Švec, SUSE: Novell zaměstnává na celém světě přibližně 600 vývojářů, z toho jich na vývoji Linuxu pracuje zhruba 200.

▷ **Odpověď:** Martin Šebesta, Red Hat: Red Hat má cca 1200 zaměstnanců. Kolik z toho je čistých vývojářů, to nevím. Vzhledem k tomu, že se jedná o technologickou firmu, myslím si, že jejich podíl bude výrazně vyšší než normálně.

★ **Otázka:** Kolik z nich pracuje v Česku?

▷ **Odpověď:** SUSE: SUSE má v České republice 45 zaměstnanců.

▷ **Odpověď:** Red Hat: Náš tým má v současnosti 20 členů.

★ **Otázka:** Jaký máte cílový stav pro Česko?

▷ **Odpověď:** SUSE: Naše pobočka roste podle potřeb Novellu a celého odvětví, uvažovaný počet je v dohledné době okolo 60 - 70, ale konkrétní plán zatím není hotový.

▷ **Odpověď:** Red Hat: Asi 200 lidí v Čechách v průběhu následujících tří let.

★ **Otázka:** Proč jste si vybrali právě Česko a ne třeba Polsko či země bývalého Sovětského svazu?

▷ **Odpověď:** SUSE: Česká republika byla vybrána z několika důvodů. Jednak proto, že tu tehdy bylo celkem hodně linuxových vývojářů, stabilní a rozvinuté prostředí, svoji roli hrála také výhodná vzdálenost mezi Prahou a sídlem mateřské společnosti - Norimberkem. A v neposlední řadě také finanční aspekt.

▷ **Odpověď:** Red Hat: Právě kvůli vyspělé linuxové komunitě, dostatku kvalifikovaných a motivovaných lidí za rozumnou cenu (proto bylo zvoleno Brno).

★ **Otázka:** Česko je malá země a počet schopných lidí není neomezený. Budete si vychovávat absolventy? Spolupracujete s univerzitami a vysokými školami?

▷ **Odpověď:** SUSE: Spíše než bychom spolupracovali se školami přímo, snažíme se vycházet vstříc individuálním požadavkům zaměstnanců, kteří ještě studují, například na uspořádání pracovní doby apod.

▷ **Odpověď:** Red Hat: Ano, program spolupráce s univerzitami se rozjíždí. Již teď máme v našem týmu několik studentů, kteří po ukončení školy budou dále pokračovat ve své kariéře v Red Hatu.

★ **Otázka:** Pracují v české pobočce jen Češi, nebo je to multikulturní prostředí?

▷ **Odpověď:** SUSE: V České pobočce SUSE pracují kromě Čechů také Slováci a v rámci výměnných pobytů i kolegové z USA.

▷ **Odpověď:** Red Hat: V současné době Češi a Slováci. V jednání jsou Poláci. Spolupracujeme s mezinárodní studentskou organizací AISEC.

Jaderné noviny – 22. 3. 2006

Robert Krátký

Aktuální verze jádra: 2.6.16. Citát týdne. Technický poradní panel OSDL. Co bude nového v 2.6.17. Na poslední chvíli se diskutovalo o unshare(). Řešení problému s hládováním ve scheduleru.

Aktuální verze jádra: 2.6.16

Aktuální verze jádra je 2.6.16. Vydána [1] 19. března.

Od 2.6.16-rc6 bylo začleněno poměrně dost oprav, ale nic zásadního. Pro ty z vás, kdo dění moc nesledují: mezi velké, pro uživatele patrné změny patří clusterový souborový systém OCFS2 [2], několik síťovacích změn, včetně kontroly zahlcení CUBIC, podpory TIPC [3] a IPv6 verze DCCP [4], migrace swapu [5] a patche pro přímou migraci [6], nové třídy scheduleru [plánovače] SCHED_BATCH, pár nových systémových volání pro souborové systémy [7] a kódu pro detekci a opravu chyb [8]. Velké interní změny zahrnují změnu na mutex [9] a kód časovače s vysokým rozlišením [10]. Podrobnosti najdete v dlouhém changelogu [11]. Git repozitář hlavního stromu obsahuje velkou hromadu patchů začleněných do 2.6.17-rc1; vizte shrnutí níže. Aktuální -mm strom je 2.6.16-rc6-mm2 [12]. V poslední době byl v -mm reorganizován kód migrace stránek (mezitím už byl začleněn), bylo provedeno pár změn v časovavačích s vysokým rozlišením, vylepšen scheduler a přidány patche předělávající MD RAID.

Citát týdne

Já jen vynucuji pěkné chování pod hrozbou pevného lana. Jeden konec připevním k balvanu, druhý na krk úlohy. Úloha se sama rozhodne, jestli bude oběšena, a uživatel rozhoduje o délce lana.

– Mike Galbraith [13]. Kdo říká, že je těžké porozumět patchům pro scheduler?

Technický poradní panel OSDL

Jak už bylo slibováno dříve, OSDL oznámila vytvoření „technického poradního panelu“, který má pomáhat se zlepšováním vztahů s vývojářskou komunitou kolem linuxového jádra. Prvními členy jsou James Bottomley, Wim Coekaerts, Randy Dunlap, Greg Kroah-Hartman, Christoph Lameter, Matt Mackall, Theodore Ts'o, Arjan van de Ven a Chris Wright.

Co bude nového v 2.6.17

Proces začleňování patchů do hlavního stromu pro 2.6.17 již probíhá několik dní. Bylo zařazeno kolem 1500 patchů, i když počet viditelných změn je relativně nízký. Zatím se do jádra dostalo toto:

- Velká aktualizace SPARC, která, mimo jiné, obsahuje podporu pro novou architekturu „Niagara“.
- Značný počet aktualizací bezdrátového síťování, včetně nového vývoje v oblasti 802.11. Ovladač ipw2200 prošel změnami, které – kromě jiného – vyžadují, aby měli uživatelé karty verzi firmwaru minimálně 3.0.
- Kód DCCP pokračuje ve vývoji; kromě dalších věcí byla přidána i CCID2 (využívající kontrolu zahlcení podobnou jako TCP).
- Netfilter má helper pro sledování spojení protokolu H.323.
- Velká aktualizace JFS.

- Obrovský balík video/DVB patchů, který přidává podporu pro množství nových zařízení a opravuje mnohé problémy.
- Velká aktualizace sériového ATA. Subsystémy SCSI a ALSA byly také rozsáhle aktualizovány.
- Několik USB audio ovladačů bylo odstraněno; USB audio hardware je lépe podporován subsystémem ALSA.
- Na mnoha místech stromu pokračuje změna semaforů na mutex.
- Začleněno `EXPORT_SYMBOL_GPL_FUTURE()` [14].
- Byla odstraněna volba `SLAB_NO_REAP`, která neustále způsobovala, že nebyl pročištěn slab, když měl systém nedostatek paměti. Typedef `kmem_cache_t` začíná být nahrazován pomocí `struct kmem_cache`.
- Rezervace „obrovských“ stránek byla zpřísněna, aby se při některých druzích použití zabránilo nedostatku paměti [out-of-memory]. Na velké stránky lze teď také použít `mprotect()`.

Na poslední chvíli se diskutovalo o unshare()

Jedním z mnoha nových systémových volání přidaných do jádra 2.6.16 je `unshare()`. Jeho účelem je dělat opak toho, co dělají různé sdílecí [sharing] parametry [flags] `clone()`: používá se k odpojení některých zdrojů procesu od zdrojů jeho předků a sourozenců. S `unshare()` může proces požádat o své vlastní souborové systémy, jmenné prostory nebo tabulku popisovače souboru. „Odsdílení“ [unsharing] dalších zdrojů, včetně undo informací semaforů, virtuální paměti, obsluhovačů signálů atd. je plánováno pro budoucí verze.

Těsně před finálním vydáním 2.6.16 se objevilo pár otázek ohledně `unshare()`; jen některé z těchto otázek byly ve výsledném jádře dořešeny. Jedna z nich se týkala implementace `unshare(CLONE_VM)`, která způsobuje, že volající proces přestane sdílet paměť s ostatními. Zdálo se, že tato funkčnost je hotová a kompletní, ale Oleg Nesterov si všiml, že kód nebere v potaz možnost, že by hlavní část adresného prostoru mohla být v procesu. Řešením je prozatím zakázání odsdílení paměti. Vypadá to, že není nikdo, kdo by tuto funkci rychle potřeboval, a už bylo příliš pozdě na pokusy o opravu funkce spravující hlavní paměť.

Eric Biederman vznesl několik otázek [15] týkajících se API `unshare()`, které by byl rád opravil dříve, než se stane součástí vydaného jádra. Jedním z problémů bylo použití stejné sady parametrů jako používá `clone()` pro specifikaci sdílení. Eric tvrdí:

sys_unshare neumí implementovat ani polovinu parametrů clone a ty, které implementuje, mají lehce jinou sémantiku než u clone. Použitím odlišné sady parametrů bychom naznačili, že jde o rozdílné věci.

Diskuze se příliš daleko nedostala, protože Linus dává přednost stejným parametrům [16] a nevypadá to, že by to někoho jiného nějak extra trápilo.

Erikova další připomínka byla o tom, že `unshare()` nezjišťuje, zda parametry existují; prostě je tiše ignoruje. Takže si uživatelský prostor může požádat o odsdílení zdrojů, které `unshare()` nezná nebo nepodporuje, a nebude vrácena žádná chyba. To by mohlo v budoucnu, kdy se očekává rozšiřování sady platných parametrů pro `unshare()`, představovat problém. Program napsaný tak, aby využíval nové parametry, by se nemusel chovat dle očekávání, pokud by byl spuštěn na jádře 2.6.16; funkčnost, kterou bude požadovat, nebude k dispozici, ale jádro o tom nepodá zprávu.

Eric poslal patch, který řešil obě záležitosti: názvy parametrů a zjišťování platných parametrů. Nebyl však do 2.6.16 začleněn. Samotný test na známé parametry začleněn být mohl (a skutečně byl začleněn do 2.6.17), ale kombinovaný patch přijat nebyl. Andrew Morton poznamenal [17]: „Tvůj patch dělal dvě rozdílné věci – z toho plyne poučení.“ Zvláště těsně před vydáním finálního jádra je důležité připravovat patche úzce zaměřené na jediný problém.

Řešení problému s hladověním ve scheduleru

Linuxový CPU scheduler [plánovač] urazil dlouhou cestu od dob čerstvého 2.6, kdy byl příčinou nemalých starostí. Plánovací domény napravily spoustu potíží na větších systémech, zatímco celá sada heuristiky pro interaktivitu pomohla lépe fungovat desktopům. Zvláště práce s interaktivitou je založena na pojmu „průměr spánku“ [sleep average]. Každý proces, který stráví hodně času spaním v poměru k času, kdy běží, je považován za „interaktivní“ a je mu přidělena vyšší priorita.

Tento mechanismus funguje tak dobře, že na reakční časy současných 2.6 jader si stěžuje málokdo. Občas se však stane, že někdo přijde se zátěží, které se podaří scheduler zmást, a celý desktop zatuhne. Mike Galbraith se některými z těchto případů zabývá a připravuje patche, které by měly pomoci se zmírňováním podobných problémů.

Linuxový scheduler udržuje dvě „pole“ front pro každý procesor. Každému procesu je při startu přidělen časový úsek a je zařazen do „aktivního“ pole, kde může bojovat o CPU. Jakmile časový úsek vyprší, proces se přesune do „prošlého“ pole, kde vyčkává, dokud všechny ostatní procesy nepoužijí své úseky. Když se všechny procesy ocitnou v prošlém poli, pole jsou prohozena a celý postup se opakuje.

V jádře 2.6 je však výjimka: proces považovaný za interaktivní (protože stráví dostatek času v přerušitelných spáncích) bude po vypršení svého časového úseku vrácen do aktivního pole. Díky tomu by neměl interaktivní proces být donucen čekat, až se nějaký dlouhotrvající dávkový proces prokouše svým časem. Aby tento mechanismus nemohl zcela zablokovat prošlé procesy, scheduler kontroluje, jestli procesy v prošlém poli nečekají příliš dlouho. Po překročení prahu „vyhladovění“ jdou všechny procesy po vypršení svých časových úseků do prošlého pole, což scheduleru umožní provést prohození polí v relativně blízké budoucnosti.

Mike zjistil, že na systému, kde běží silně zatížený Apache server, mohou některé úlohy hladovět velmi dlouhou dobu; vypadá to, že mechanismus, který má bránit vyhladovění, nefunguje správně. Problém vězel v probouzecím kódu. Kód vždy dával čerstvě probuzené procesy do aktivního pole bez ohledu na to, co se dělo ve zbytku systému. Protože bylo neustále probouzeno velké množství serverových procesů díky příchozím požadavkům, scheduler se vůbec nedostal k prohození polí. Řešením [18] bylo vložit kontrolu hladovění do `__activate_task()`; výsledkem je, že pokud prošlé procesy hladoví, budou procesy probouzeny do prošlého pole. Tato malá oprava se postarala o velkou část problému. Větší opravy však byla potřeba pro [patch přiškrcující úlohy](#) [19], na kterém Mike už nějaký čas pracuje. Součástí této práce je několik oprav, ale základní zjištění je následující: kód, který se stará o „průměr spánku“ může být příliš štědrý k procesům, které spí jen chvilku. Proces, kterému se podaří pravidelně na krátkou chvilku usnout, může výrazně zvýšit svoji prioritu. Natolik, že vytlačí ostatní procesy běžící na daném systému. A získá-li proces bonus interaktivity, může si ho nějakou dobu podržet. To vše je záměrné; některé interaktivní programy mohou velmi dlouho sedět a pak chvilku provádět náročné výpočty. Vezměte si třeba X server s tím pěkným kompozitním správcem oken; spoustu času je nečinný, jen aby procesor sešlápl, když začne uživatel tahat okna po obrazovce. Ale takové chování může také udělit prioritní bonus interaktivity procesům, které ve skutečnosti interaktivní nejsou.

Řešení zahrnuje několik změn. Jednou z nich je prostě menší štedrost při rozdělování bonusů. Ale jádrem patche je funkce nazývaná `refresh_timeslice()`. Tato funkce porovnává aktuální průměr spánku s časem, který proces doopravdy stráví v procesoru. Na základě tohoto srovnání je upraven škrtící čas pro každý proces. Je-li CPU používáno více než by odpovídalo průměru spánku, je čas přiškrcení posunut dozadu; jinak dopředu. Narazí-li proces na čas přiškrcení, začne jeho průměr spánku rychle mizet, což ho zbaví jeho bonusu interaktivity.

Čas přiškrcení poskytuje chvilky oddechu [grace periods], které procesům dovolí krátké využití CPU, aniž by byly penalizovány. Množství času na oddech může být upraveno pomocí dvou nastavení exportovaných kódem přiškrcení. „Oddech 1“ je množství času, které nové procesy dostanou, aby mohly být nastaveny jejich hodnoty průměrů předtím než budou předhozeny škrtícímu mecha-

nismu. „Oddech 2“ je doba, po kterou mohou procesy přesáhnout předpokládaný procesorový čas, než začne fungovat přiškrcování. Ohledně přidání těchto nastavovátek se objevilo několik stížností; vypadají jako další obskurní způsob ladění jádra, který stejně většina administrátorů nebude umět správně využít. Takže se tlačilo na to, aby byla nastavovátka nahrazena obyčejným přepínačem zapnuto/vypnuto. Systémy určené pro interaktivní využití ponechají přiškrcování zapnuté, serverové systémy ho prostě celé vypnou. Vyřešení této záležitosti možná zpozdí přijetí patche, i když proti jeho zbytku nikdo nic nenamítá.

Odkazy

- [1] <http://lwn.net/Articles/176193/>
- [2] <http://lwn.net/Articles/137278/>
- [3] <http://tipc.sourceforge.net/>
- [4] <http://lwn.net/Articles/149756/>
- [5] <http://lwn.net/Articles/157066/>
- [6] <http://lwn.net/Articles/160201/>
- [7] <http://lwn.net/Articles/164887/>
- [8] <http://lwn.net/Articles/168975/>
- [9] <http://lwn.net/Articles/167034/>
- [10] <http://lwn.net/Articles/167897/>
- [11] <http://kernel.org/pub/linux/kernel/v2.6/ChangeLog-2.6.16>
- [12] <http://lwn.net/Articles/176114/>
- [13] <http://lwn.net/Articles/176616/>
- [14] <http://lwn.net/Articles/171838/>
- [15] <http://lwn.net/Articles/176456/>
- [16] <http://lwn.net/Articles/176457/>
- [17] <http://lwn.net/Articles/176463/>
- [18] <http://lwn.net/Articles/176638/>
- [19] <http://lwn.net/Articles/173469/>

Jaderné noviny – 29. 3. 2006

Robert Krátký

Aktuální verze jádra: 2.6.16.1. Co bude nového v2.6.17, část2. Rámcem propravidla nahrazování stránek. Správa stromu 2.6.16.y. Co z-mm půjde do2.6.17. SCSIID.

Aktuální verze jádra: 2.6.16.1

Aktuální verze stabilního jádra je 2.6.16.1. Vydána [1] 27. března. Zároveň vyšla i verze 2.6.15.7 [2]. Oba patche obsahují slušnou řádku důležitých oprav, některé se týkají bezpečnosti.

Během minulého týdne nevyšla žádná vývojová předverze. Do hlavního git repozitáře však patche proudí vysokým tempem; viz shrnutí níže. Aktuální -mm strom je 2.6.16-mm2 [3]. Mezi nedávné změny patří možnost volat `poll()` na sysfs soubory (viz [článek na LWN](#) [4]), podpora 64bitových I/O a paměťových zdrojů, podpora futexů, které dědí prioritu, a nová sada patchů pro správu centrálního času.

Co bude nového v 2.6.17, část 2

Záplava patchů, které se hrnou do hlavního stromu, pokračuje bez ustání – i když čas na přidávání nových funkcí brzy vyprší. Následuje přehled nejzajímavějších částí kódu začleněného od minulého týdne:

- Patch pro podporu [lehkých robustních futexů](#) [5].
- Vrstva softwarového RAIDu (MD) zvládá za běhu měnit velikost RAID5 polí.
- Z SCSI subsystému byla odstraněna podpora devfs. V mnoha jiných částech jádra však stále zůstává.
- Patch s [uživatelským software suspend](#) [6].
- Velká aktualizace XFS.
- Softwarová implementace MAC pro 802.11. Byla také začleněna verze 20 wireless extensions API [rozšíření pro bezdrátové sítě].
- Byl přidán reverse-engineered [rozluštěn z binárky] [ovladač Broadcom 43xx](#) [7]. Tím pádem se seznam Linuxem podporovaných bezdrátových síťových karet znatelně rozrostl.
- Mechanismus „rozložení paměti“ [memory spreading] lze použít k rozproštění keše stránek a alokací v bufferech souborových systémů přes všechny uzly NUMA systému.
- [Dvě nové operace `fdadvise\(\)`](#) [8] pro ovládání asynchronního vypisování souborů.
- Podpora přeskupovacích funkcí v linkovaném obraze jádra. Účelem je dát dohromady hodně používané části kódu jádra, aby se tato vytížená oblast vešla do jediného TLB záznamu. V současné době má infrastrukturu pro přeskupování pouze x86-64.
- Souborový systém ext3 získal možnost alokovat a mapovat více bloků, což zvyšuje výkon při sekvenčních přístupech k souborům.
- Nově přidaná plánovací doména [scheduling domain] reprezentuje vícejádrové systémy.
- Byl začleněn nový RTC subsystém, který poskytuje podporu pro různé real-time hardwarové hodiny.

Bude-li dodržen obvyklý postup, přestanou být nové funkce přidávány někdy kolem konce měsíce a 2.6.17-rc1 vyjde krátce poté.

Rámec pro pravidla nahrazování stránek

„**Panejo.**“ Tak [reagoval](#) [9] Andrew Morton na 34dílný patch od Petera Zijlstry, který vytváří abstraktní API pro pravidla nahrazování stránek. Kód nahrazování stránek je v samém jádře systému virtuální paměti; jde vlastně o heuristické rozhodování o tom, které stránky by měly být vyhozeny z hlavní paměti a zpřístupněny pro jiné účely. Jde trochu o černou magii; ačkoliv je snadné poznat, kdy systém spravuje paměť špatně, není často vůbec jasné, jak situaci vylepšit. Správa paměti v Linuxu byla dlouhé roky choulostivou otázkou, ale nyní se zdá, že ve většině případů funguje dobře. Vzhledem k tomu, že se všechen ten záluždný kód konečně dostal do poměrně dobrého stavu, proč by se v něm chtěl ještě někdo vrtat?

Odpovědí je, že alternativní mechanismy pro nahrazování stránek jsou docela aktivně zkoumány, a Linux by z této práce mohl těžit. Konec konců málokdo by řekl, že linuxová virtuální paměť funguje tak dobře, že už by nebylo co vylepšovat.

Tato velká sada patchů vytváří API pro algoritmy nahrazování stránek a umožňuje jejich libovolné střídání. Tedy, přinejmenším při restartu; v současné době neexistuje možnost nahrávat nahrazovací algoritmy jako moduly nebo je měnit za běhu. Ale při konfiguraci jádra mohou administrátoři zvolit systém nahrazování stránek, který nejlépe vyhovuje jejich podmínkám. Programátoři, kteří pracují na virtuální paměti, si mohou hrát s různými algoritmy, aby zjistili, jak to s nimi funguje. Aby mohl nahrazovací algoritmus s tímto API fungovat, musí definovat sadu specifických funkcí. Takže například existuje dvojice inicializačních funkcí:

```
void page_replace_init(void);
void page_replace_init_zone(struct zone *zone);
```

Tyto funkce jsou volány při bootu a připravují nahrazovací kód k tomu, aby fungoval se systémem, na kterém poběží. Pokud jádro něco ví o konkrétních stránkách, může to nahrazovacímu algoritmu sdělit pomocí následujících volání:

```
void page_replace_hint_active(struct page *page);
void page_replace_hint_use_once(struct page *page);
```

První je voláno, když si jádro všimne, že stránka je aktivně používána. Druhé značí, že stránka pravděpodobně nebude v blízké budoucnosti používána. Existují různé další funkce pro udržování pořádku, ale jádrem API je tato funkce:

```
void page_replace_candidates(struct zone *zone, int count,
                           struct list_head *list);
```

Ta musí vybrat až `count` stránek z dané `zone` jako kandidáty na vyhození. V tom okamžiku nahlédne nahrazovací kód do své křišťálové koule, aby zjistil, které stránky nebudou v nejbližší době použity; ty budou vyčleněny a vráceny jádru.

Další funkce se věnují otázkám jako je migrace stránek, sledování neresidentních stránek, vypisování informací z nahrazovacího kódu a tak dále. Seznam a popis těchto dalších funkcí naleznete v [dokumentačním souboru](#) [10].

Sada patchů rovnou obsahuje čtyři různé nahrazovací mechanismy. První je kód z aktuálního jádra, který využívá metodu LRU (least-recently-used = nejdříve použité), upravený tak, aby používal nové API. Další je [CLOCK-PRO algoritmus](#) [11]. Pak implementace techniky CART vysvětlené v [této práci](#) [PDF] [12]. A nakonec jednoduchý systém pro náhodné nahrazování, zjevně jen tak pro legraci. I když, ten [patch pro náhodné nahrazování](#) [13] je díky své jednoduchosti vhodným startem pro ty, které zajímá, jak vypadá modularizovaný nahrazovací algoritmus.

Patch je trochu podobný [kódu pro vybírání CPU schedulerů](#) [14], který umožňuje měnit plánovací algoritmus. Patch je i nadále spravován, ale od doby, kdy byl v roce 2004 poprvé představen, se nikdy

vážněji neuvažovalo o začlenění do hlavního jádra. Daleko větší zájem je o nalezení chyb – jsou-li nějaké – v současném kódu, ne o vytváření mechanismu pro zkoušení úplně jiných implementací. Proto svou odpověď Andrew Morton doplnil takto:

Spíše než čtyřikrát celou věc nahrazovat bych uvítal přesný popis těchto problémů. Zjistit, jestli by nešlo situaci zlepšit po částech místo toho, abychom to rovnou celé vyhodili...

Linus má **podobný názor** [15] a kromě toho není přesvědčen, že by nahrazování stránek byl skutečně problém zasluhující zvláštní pozornost „**Případá mi to jako univerzitní výzkum.**“

Zastánci patche tvrdí, že doopravdy existují situace, při kterých jde současný kód do kolen. S ohledem na to by se zdálo, že dalším logickým krokem by bylo shromažďovat informace o případech, ve kterých linuxová správa paměti selhává. Pak mohou vývojáři začít uvažovat o tom, co je potřeba udělat pro to, aby byla tato selhání vyřešena. I kdyby patche nebyly nikdy začleněny, vypadá to, že by mohly pomoci nastartovat další fázi práce na algoritmech linuxové správy paměti. A to by bylo dobře bez ohledu na to, jestli do jádra půjdou nebo ne.

Poznámka redakce: Na podnět několika čtenářů jsem se dohodl s Jeremy Andrewsem [16] na využití materiálů z KernelTrap.org [17]. Jaderné noviny tedy budou kombinovat informace ze dvou zdrojů: LWN.net a KernelTrap. Následující obsah je ©KernelTrap.

Správa stromu 2.6.16.y

23. bře, **originál** [18]

Spolu s vydáním jádra 2.6.16 se Adrian Bunk vrátil ke svému **dříve probíranému záměru** [19] spravovat do budoucna jádro 2.6.16.y. První verze 2.6.x.y byla 2.6.8.1 od Linuse Torvaldse – rychlá jednořádková oprava v NFS. Myšlenka byla znovu vytažena o pár měsíců později v říjnu 2004 [článek [20]], ale chytla se až v březnu 2005 [článek [21]] [článek [22]]. Počínaje jádrem 2.6.11 byla procesu dána jasná pravidla a Greg KH spolu s Chrisem Wrightem začali oficiálně spravovat verze 2.6.x.y [článek [23]] – dokud nevyjde 2.6.(x+2). Například nyní budou Greg a Chris aplikovat stabilní patche na jádro 2.6.16.y, dokud někdy v budoucnu nevyjde 2.6.18.

Adrian plánuje v tu chvíli převzít vývoj jádra 2.6.16.y a spravovat ho podobně jako je spravována řada 2.4. Má v úmyslu strom udržovat tak dlouho, dokud jej lidi budou používat a budou posílat patche. Minule se v debatě o tomto záměru objevily smíšené reakce. Greg KH varoval: „**Chceš-li to dělat dlouhodobě, spolkne to spoustu času a energie. Být tebou, tak naslouchám těm, kteří tento typ jader spravovali a spravují; není to rozhodně nic snadného.**“

Co z -mm půjde do 2.6.17

27. bře, **originál** [24]

Andrew Morton [interview [25]] sestavil seznam patchů ze svého -mm stromu, ve kterém u každé položky shrnul své úmysly ohledně toho, jestli ji bude tlačit dál k Linusovi pro začlenění do 2.6.17 nebo ne. Komentáře u patchů se pohybují od prostého „**začlením**“ až po přenechání jiným k posouzení. Jeden ze zábavnějších komentářů se objevil u sady 33 patchů, kde Andrew poznamenal: „**Takhle se Oleg vyřádl v jádře jádra. Je tam hromada materiálu Pravděpodobně to celé pošlu Linusovi a požádám ho, aby to prošel (tj. myji si ruce).**“

Později vysvětlil: „**je to prostě spousta kódu v delikátních oblastech, kde pracuje jen pár lidí, a pro které je obtížné najít někoho, kdo by to zkontroloval.**“

Jedna sada patchů byla odmítnuta s komentářem: „**Pořád pro tohle neznám přesvědčivý argument.**“

Šlo o **předběžné načítání swapu** [26] od Cona Kolivase [interview [27]]. Funkce pak byla diskutována v několika následujících vláknech. V odpovědi na připomínky Jense Axboea vysvětlil Con implementaci trochu podrobněji: „**Je-li systém nečinný, nic nás načtení těchto stránek nestojí (v laptopovém režimu je předběžné načítání vypnuto, uvažujete-li o spotřebě energie u laptopů). A pokud systém potřebuje RAM, která byla těmi načtenými stránkami chybně zaplněna a není-li k nim přístupováno,**

budou to první, co se vyhodí - bez jakéhokoliv I/O, protože jsou na úplném konci neaktivního LRU seznamu a kopii mají uloženou.“

SCSI ID

28. bře, originál [28]

Zajímavé vlákno na LKML začalo patchem, který chtěl přesunout IOCTL volání používané k získání SCSI ID zařízení z blokové vrstvy do SCSI subsystému. Linus Torvalds poměrně vehementně argumentoval, že neexistuje nic, čemu by se říkalo SCSI ID, „Nechte toho. Měli bychom to ioctl zrušit, ne se pokoušet, aby vypadalo smysluplně. Takový způsob hledání SCSI zařízení nedává smysl. A skutečnost, že někteří lidi kolem SCSI si to myslí, na tom nic nemění.“

Podobně vzrušená diskuze se o tomto tématu rozběhla v konferenci několikrát, argumenty přicházely z obou stran. V zatím nejnovějším vlákně Linus vysvětloval:

„Byla to stupidní myšlenka i v 80. letech. Od té doby je čím dál stupidnější. V těch 80. letech byla alespoň *omluvitelná*. Bylo přijatelné si myslet, že řadiče lze prostě očíslovat. A bylo to snadné, a protože tehdy byly hotplug řadiče typu „obsluha zastrčí“, ne jako ty moderní ”magicky se objeví”, tak to fungovalo a čísla řadičů něco znamenala (přestože se měnila, když člověk věci zpřeházel, ale to se dalo očekávat).

„Dneska už není možné řadiče jakýmkoliv rozumným způsobem číslovat. Neřekl bych, že to vůbec kdy bylo možné, ale přinejmenším bylo u statického hardwaru jakékoliv náhodné číslování stejně užitečné jako každé jiné.“

Odkazy

- [1] <http://lwn.net/Articles/177396/>
- [2] <http://lwn.net/Articles/177398/>
- [3] <http://lwn.net/Articles/177394/>
- [4] <http://lwn.net/Articles/174660/>
- [5] <http://lwn.net/Articles/172149/>
- [6] <http://lwn.net/Articles/153203/>
- [7] <http://bcm43xx.berlios.de/>
- [8] <http://lwn.net/Articles/171941/>
- [9] <http://lwn.net/Articles/177119/>
- [10] <http://lwn.net/Articles/177123/>
- [11] <http://lwn.net/Articles/147879/>
- [12] <http://www.almaden.ibm.com/cs/people/dmodha/clockfast.pdf>
- [13] <http://lwn.net/Articles/177127/>
- [14] <http://lwn.net/Articles/109458/>
- [15] <http://lwn.net/Articles/177124/>
- [16] <http://kerneltrap.org/user/1>
- [17] <http://kerneltrap.org/>
- [18] <http://kerneltrap.org/node/6386>
- [19] <http://lkml.org/lkml/2005/12/3/55>
- [20] <http://kerneltrap.org/node/4100>
- [21] <http://kerneltrap.org/node/4793>
- [22] <http://kerneltrap.org/node/4800>
- [23] <http://kerneltrap.org/node/4803>
- [24] <http://kerneltrap.org/node/6402>
- [25] <http://kerneltrap.org/node/10>
- [26] <http://kerneltrap.org/node/5619>
- [27] <http://kerneltrap.org/node/465>
- [28] <http://kerneltrap.org/node/6408>

Jaderné noviny – 5. 4. 2006

Robert Krátký

Aktuální verze jádra: 2.6.17-rc1. Citát týdne: Al Viro. Odvozování priorit v jádře. O bezpečnosti sysfs rozhraní. Dokumentování paměťových bariér. Konec přijímání patchů do 2.6.17.

Aktuální verze jádra: 2.6.17-rc1

Aktuální předverze je 2.6.17-rc1. Linus ji vydal [1] 2. dubna. Od minulého týdne byly začleněny patche obsahující systémová volání `splice()` a `sync_file_range()` (viz níže), podporu hotplug paměti pro User-mode Linux, subsystem LED, konverzi `local_t` na typ se znaménkem, základní podporu braillových zařízení pro vstupní vrstvu a ovladač „ipath“ pro zařízení PathScale InfiniPath. Vizte shrnutí minulého a předminulého týdne – v nich najdete podrobné seznamy změn v 2.6.17-rc1. Ještě více podrobností je v krátkém [2] a dlouhém [3] changelogu. Od vydání 2.6.17-rc1 nebyly do hlavního stromu přijaty žádné patche. Aktuální -mm strom je 2.6.17-rc1-mm1 [4]. Mezi nedávné změny patří velké množství oprav a nová verze debuggeru kgdb. Jinak nic zásadního.

Citát týdne: Al Viro.

Které části z „patche pro sysfs by mohli psát idioti – a většinou je píší“ je tak těžké porozumět? Aha, moment. Už chápu... No dobře, tak to nic... – Al Viro [5] je zpět.

Dvě nová systémová volání: `splice()` a `syncfilerange()`

Jádro 2.6.17 bude obsahovat dvě nová systémová volání, která zajímavým způsobem rozšiřují možnosti uživatelských programů. Tento článek se dívá na současnou podobu těchto nových rozhraní.

`splice()`

Systémové volání `splice()` má dlouhou historii. Jako první jej navrhl Larry McVoy v roce 1998; bralo se to jako způsob zlepšení I/O výkonu na serverových systémech. Přestože se o tom v následujících letech často mluvilo, nebyla žádná implementace `splice()` pro hlavní jádro napsána. To se však změnilo těsně před ukončením přijímání patchů pro 2.6.17, když byl s mnohými úpravami začleněn `splice()` patch od Jense Axboea. V době psaní tohoto textu vypadalo rozhraní `splice()` takto:

```
long splice(int fdin, int fdout, size_t len, unsigned int flags);
```

Volání `splice()` způsobí, že jádro přesune až `len` bajtů z datového zdroje `fdin` do `fdout`. Data se budou pohybovat pouze v rámci jádra, kopírování bude minimum. V současné implementaci musí alespoň jeden z těchto dvou popisovačů souborů [file descriptors] ukazovat na rouru [pipe device]. Tato podmínka je omezením stávajícího kódu a mohla by být někdy v budoucnu odstraněna. Parametr `flags` upravuje, jak je kopírování prováděno. Implementovány jsou následující možnosti:

- `SPLICE_F_NONBLOCK`: `splice()` operace nebudou blokovat. Volání `splice()` by i tak mohlo blokovat v případě, že by jeden z popisovačů souborů nebyl nastaven na neblokovací I/O.
- `SPLICE_F_MORE`: řekne jádru, že v dalším volání `splice()` přijde více dat.
- `SPLICE_F_MOVE`: je-li výstup soubor, řekne tento parametr jádru, že se má pokusit přesouvat stránky přímo ze vstupního bufferu roury do výstupního adresního prostoru, čímž se předejde kopírování.

Interně funguje `splice()` pomocí mechanismu **rourového bufferu** [6], který přidal Linus na počátku roku 2005 – proto se zatím vyžaduje, aby byla na jedné straně operace roura. Dva přírůstky přibyly do stále větší struktury `file_operations` pro zařízení a souborové systémy, které chtějí `splice()` podporovat:

```
ssize_t (*splice_write)(struct inode *pipe, struct file *out,
                        size_t len, unsigned int flags);
ssize_t (*splice_read)(struct file *in, struct inode *pipe,
                       size_t len, unsigned int flags);
```

Tyto nové operace by měly přesunout `len` bajtů mezi `pipe` a buď `in` nebo `out`, podle zadaných `flags`. Pro souborové systémy existují obecné implementace těchto operací, které lze použít; také je k dispozici `generic_splice_sendpage()`, která se používá pro umožnění „splicování“ na socket. Zatím nejsou žádné implementace `splice()` pro ovladače zařízení, ale do budoucna tomu nic nebrání – alespoň pro znaková zařízení.

Diskuze v konferenci linux-kernel naznačují, že rozhraní `splice()` by mohlo ještě doznat změn předtím, než bude vytesáno do kamene v jádře 2.6.17. Andrew Tridgell **požadoval** [7] přidání offsetového parametru, aby šlo určit, kde má kopírování začít – buď to, nebo by se mělo přidat samostatné `psplice()`. Vyskytly se také pochybnosti o způsobu, kterým jsou zpracovávány chyby; pokud volání `splice()` vrátí chybu, jak aplikace zjistí, jestli je problém se vstupem nebo výstupem? Vyřešení těchto otázek si možná během příštího měsíce vyžádá nějaké změny rozhraní.

syncfilerange()

Na začátku příprav 2.6.17 byly začleněny **změny systémového volání** `posix_fadvise()` [8]. Ty nové – linuxové – volby mají za účel dát aplikacím lepší kontrolu nad tím, jak jsou data zapisovaná do souborů dávana na fyzická média. Zmíněné doplněné možnosti jsou potřeba, ale vyskytly se obavy z jakéhokoliv čistě linuxového rozšiřování POSIXové funkce. Výsledkem bylo, že Andrew Morton patch stáhl a nahradil ho novým systémovým voláním:

```
long sync_file_range(int fd, loff_t offset, loff_t nbytes, int flags);
```

Toto volání sesynchronizuje data souboru na disk počínaje zadaným `offset` a bude pokračovat `nbytes` bajtů (nebo do konce souboru, bude-li `nbytes` nula). Způsob synchronizace je určen pomocí `flags`:

- `SYNC_FILE_RANGE_WAIT_BEFORE` blokuje volající proces, dokud nebude dokončeno jakékoliv již probíhající vypisování stránek (v daném rozsahu).
- `SYNC_FILE_RANGE_WRITE` začne vypisovat všechny nečisté stránky v daném rozsahu, u kterých už neprobíhá nějaký I/O.
- `SYNC_FILE_RANGE_WAIT_AFTER` blokuje volající proces, dokud neskončí nově iniciované zápisy.

Aplikace, která chce iniciovat writeback [zápis na disk] všech nečistých stránek, by měla poskytnout první dva parametry. Když poskytne všechny tři, je zaručeno, že ty stránky skutečně budou na disku, až volání skončí.

Nová implementace nemění systémové volání `posix_fadvise()`. Zároveň umožňuje provádění synchronizačních operací jediným voláním místo několika, která vyžadoval předchozí pokus. V budoucnu bude pravděpodobně možné přidat do seznamu `flags` další operace; možnost vyžadovat synchronizaci metadat bude asi první. (Díky Michaelu Kerriskovi – který tuto změnu prosazoval – za poskytnutí bližších informací).

Odvozování priorit v jádře

Představte si systém se dvěma běžícími procesy, jeden s vysokou prioritou, druhý s o hodně nižší. Tyto procesy sdílejí zdroje chráněné zámky. V jednu chvíli se procesu s nižší prioritou podaří chvíli běžet a získat zámek jednoho ze zdrojů. Pokud se pak pokusí prioritní proces získat stejný zámek, bude muset počkat. Proces s nízkou prioritou tak vlastně přebil ten s vysokou – přinejmenším dokud bude držet ten požadovaný zámek.

A teď si představte třetí proces, který spotřebovává hodně procesorového času, a jehož priorita je někde mezi těmi dvěma. Začne-li tento proces makat, vytlačí ten s nízkou prioritou od CPU úplně. Výsledkem je, že třetí proces může ten proces s nejvyšší prioritou držet donekonečna v šachu.

Tato situace, která se nazývá „obrácení priority“ [priority inversion], mívá většinou na svědomí selhání systému, otrávené uživatele a nezaměstnané inženýry. Existuje několik přístupů k boji s obrácením priority: systém bez zámků, pečlivě promyšlené zamykačí postupy a technika známá jako odvozování priorit. Odvozování priorit má jednoduchý koncept: když proces drží zámek, měl by běžet alespoň s takovou prioritou, jako má proces s nejvyšší prioritou z těch, které na zámek čekají. Je-li zámek zachycen nízkoprioritním procesem, může být nutné jeho prioritu zvýšit až do chvíle, kdy bude zámek uvolněn.

Odvozování priorit může mít více podob. I jádro vlastně využívá velmi jednoduchou formu, když neumožňuje jadernému kódu preempci zatímco drží spinlock. Na některých systémech má každý zámek přiřazenou prioritu; vezme-li zámek proces, je jeho priorita zvýšena na prioritu zámku. Na jiných je procesu, který drží vyžadovaný zámek, priorita „odvozena“ od priority procesu s vysokou prioritou. Většina způsobů odvozování priorit projevuje tendenci komplikovat a zpomalovat zamykačí kód a dá se jich zneužít k zakamuflování mizerných designů aplikací. Takže jsou v mnoha kruzích neoblíbené. Linus se minulý prosinec o tématu [vyjádřil celkem jasně](#) [9]:

„Kamarádi nenechají kamarády používat odvozování priorit.“ Prostě to nedělejte. Pokud to skutečně potřebujete, máte tak jako tak vadný systém.

Pozn. „Kamarádi nenechají kamarády řídit v opilosti“ („Friends don't let friends drive drunk“) je hlavním sloganem známé kampaně amerických úřadů, která má za cíl poučit lidi o nebezpečnosti řízení pod vlivem alkoholu. Tváří v tvář takovému odporu by většina vývojářů potichu sbalila své plány na odvozování priorit zpátky do šuplíku a vrátila by se k něčemu jednoduššímu. Komunita vývojářů linuxového jádra má však jednoho člena, který proslul schopností prosadit svůj kód do jádra i přes podobné námitky: Ingo Molnar. Historie se možná bude opakovat, protože Ingo (ve spolupráci s Thomasem Gleixnerem) poslal [implementaci futexů s odvozováním priorit](#) [10] s žádostí, aby byla začleněna do hlavního jádra. Ingo tvrdí, že navrhovaný přístup poskytuje užitečné funkce uživatelskému prostoru (nemá poskytovat základní jaderné funkce vzájemného vyloučení s odvozováním priorit [priority-inheriting kernel mutual exclusion primitives]), ale vyhýbá se problémům, se kterými se potýkaly jiné implementace.

Patch s PI-futex (PI – Priority Inheritance = odvozování priorit) přidává pár nových operací do systémového volání `futex()`: `FUTEX_LOCK_PI` a `FUTEX_UNLOCK_PI`. Když není více zájemců, může být PI-futex zabrán, aniž by se zapojovalo jádro - stejně jako v případě obyčejného futexu. Existuje-li více zájemců, je od jádra vyžádána operace `FUTEX_LOCK_PI`. Žádající proces je zařazen do speciální fronty a, je-li to nutné, propůjčí svou prioritu procesu, který požadovaný futex drží. Odvozování priorit je zřetězené, takže je-li proces, který drží zámek, blokován na dalším futexu, bude zvýšená priorita přenesena i na držitele toho dalšího futexu. Jakmile je futex uvolněn, veškerá přidaná priorita je odebrána. Podobně jako u běžných futexů je to tak, že jádro musí vědět jen o těch PI-futexech, na které je více zájemců. Takže počet futexů v systému může docela narůst, aniž by to představovalo velkou režii pro jádro. V jádře je typ PI-futex implementován pomocí nové základní funkce nazývané `rt_mutex`. `rt_mutex` je na první pohled podobná běžným mutexům, ale je doplněna o možnost odvozování priorit. Jde však o zcela odlišný typ, s implementací mutexů nesdílí žádný kód. API však bude uživatelům mutexů povědomé; ve stručnosti:

```
#include <linux/rtmutex.h>

void rt_mutex_init(struct rt_mutex *lock);
void rt_mutex_destroy(struct rt_mutex *lock);

void rt_mutex_lock(struct rt_mutex *lock);
int rt_mutex_lock_interruptible(struct rt_mutex *lock,
                               int detect_deadlock);
int rt_mutex_timed_lock(struct rt_mutex *lock,
                       struct hrtimer_sleeper *timeout,
                       int detect_deadlock);
int rt_mutex_trylock(struct rt_mutex *lock);
void rt_mutex_unlock(struct rt_mutex *lock);
int rt_mutex_is_locked(struct rt_mutex *lock);
```

Pozornému čtenáři možná neuniklo, že to vypadá podobně jako realtime mutex z patche pro realtime preempci. Ingo jednou řekl, že realtime patche se do jádra postupně dostanou; a tímhle to možná začíná. S tímto patchem by byl PI-futex kód jediným uživatelem nového typu `rt_mutex`, ale to se může časem změnit.

PI-futex patch také obsahuje nový druh seznamu tříděného podle priorit, který by mohl najít uplatnění i na jiných místech jádra.

Zatím se o tomto patchi příliš nediskutovalo; byl zařazen v posledních -mm stromech. Pro 2.6.17 je už pozdě, ale pokud si nenajde vážného odpůrce, mohl by se PI-futex kód dostat do další verze.

O bezpečnosti sysfs rozhraní

Jedním z [patchů](#) [11] v 2.6.16.2 je oprava bezpečnostní chyby označené jako CVE-2006-1055. Jde o malou změnu v kódu, který implementuje možnost zapisovat do atributů sysfs. Změna omezuje maximální objem dat, který může být do atributu zapsán, na `PAGE_SIZE-1` bajtů, neboli 4095 na většině systémů. Od minulého června byl limit prostě `PAGE_SIZE`, což umožňovalo zapsání celé stránky. Protože před zaplněním je prostor vyplněn nulami, zajistí tato změna, že data přicházející z uživatelského prostoru budou při odeslání konkrétní sysfs `store()` funkci ukončena nulou. Bez této pojistky by funkce mohla vesele začít mimo konec toho jednostránkového bufferu a přistupovat k datům, která nepřišla z uživatelského prostoru, a případně i přepsat buffery jinde. Tato možnost stačila k tomu, aby to bylo považováno za bezpečnostní chybu, která si vyžaduje rychlou opravu. Zajímavé na tom je, že prototyp funkce `store()` vypadá takto:

```
ssize_t (*store)(struct kobject *kobj, struct attribute *attr,
                const char *buffer, size_t size);
```

Parametr `size` je objem posílaných uživatelských dat. Takže by bylo možné se ptát, proč se otravovat s ukončováním bufferu nulami, když byla jeho velikost už poskytnuta kódu, který data přijímá. Někteří vývojáři, jejichž kód dostával prostřednictvím sysfs atributů data o 4096 bajtech, se takto skutečně ptali.

Odpověď je v jistém smyslu obsažena v citátu týdne. Diplomatičtěji by šlo říci, že bez ohledu na to, jak je rozhraní navrženo, množství implementací sysfs atributů bylo programováno na základě předpokladu, že přichodící data budou ukončena nulami. Takže se neobtěžují s kontrolou objemu dat a v případě absence očekávaného ukončení budou provádět nehezke věci. V 2.6.16.2 je situace napravena a zmiňované implementace jsou opět bezpečné. Ale je těžké nebýt z toho alespoň trochu

nervózní. Je-li v jádře neadbale psaný kód, mohly by s ním být další potíže a ukončení nulami by nijak nepomohlo. Bylo by lepší, kdyby existoval způsob, jak ověřit, že jsou rozhraní využívána korektně. Prozatím by si lidé, kteří píšou sysfs rozhraní – z nichž každé je rozhraním k uživatelskému prostoru a možným cílem útoku – měli kód před odevzdáním lépe zkontrolovat. Následující obsah je ©KernelTrap.

Dokumentování paměťových bariér

2. dub, originál [12]

David Howells poslal do konference dokument o paměťových bariérách [memory barriers]. Paměťové bariéry představují způsob, jak uspořádat I/O přístupy jádra k zařízením. Dokument prošel mnoha revizemi na základě reakcí vývojářů. Výtažek:

...nezávislé operace s pamětí jsou prováděny v náhodném pořadí, což může být problém pro součinnost dvou a více CPU a I/O. Je potřeba nějakým způsobem zasáhnout a říci kompilátoru a procesoru, aby dodržovaly pořadí. Paměťové bariéry jsou takovými zásahy. Zavádějí pomyslné dílčí řazení paměťových operací na obou stranách bariéry. Vyžadují, aby posloupnost generovaných událostí vypadala z pohledu zbytku systému jakoby byla bariéra na daném procesoru doopravdy.

Konec přijímání patchů do 2.6.17

3. dub, originál [13]

Linus Torvalds oznámil jádro 2.6.17-rc1: Od 2.6.16 uplynuly dva týdny, takže končí doba vymezená pro začleňování patchů.

Zmiňovaná doba je součástí vývojového modelu od Summitu vývojářů linuxového jádra 2005, na kterém se rozhodlo, že všechny zásadní změny musí být začleněny během dvou týdnů od vydání hlavní verze, přičemž zbytek vývojového cyklu bude zaměřen na opravování chyb [viz [Jaderné noviny 327: nová pravidla pro rychlejší vydávání nových verzí jádra \[14\]](#)]. K aktuální verzi Linus poznamenal:

Jako obvykle je -rc1 patch zatraceně velký a dokonce i krátká verze changelogu se nevejde do limitu na velikost zpráv v konferenci. Ale mám takový dojem, že většina jsou věci jako nové ovladače a hodně je toho méně děsivého než první změny v 2.6.16. Nejnápadnější změnou bude pravděpodobně začlenění podpory architektury Niagara od Sunu, ale je tam toho ještě víc: reorganizace DVB, aktualizace nfs/knfsd, aktualizace x86-64/parisc/mips/powerpc, ALSA, SCSI, Infiniband... A zároveň je tu slušná várka běžných čistek (konverze BUG_ON, konverze semafor→mutex, pročištění bitops atd.).

Odkazy

- [1] <http://lwn.net/Articles/178176/>
- [2] <http://lwn.net/Articles/178178/>
- [3] <http://kernel.org/pub/linux/kernel/v2.6/testing/ChangeLog-2.6.17-rc1>
- [4] <http://lwn.net/Articles/178368/>
- [5] <http://lwn.net/Articles/178632/>
- [6] <http://lwn.net/Articles/119682/>
- [7] <http://lwn.net/Articles/178208/>
- [8] <http://lwn.net/Articles/171941/>
- [9] <http://lwn.net/Articles/178258/>
- [10] <http://lwn.net/Articles/177111/>
- [11] <http://lwn.net/Articles/178635/>
- [12] <http://kerneltrap.org/node/6431>
- [13] <http://kerneltrap.org/node/6435>
- [14] <http://www.abclinuxu.cz/clanky/jaderné-noviny/jaderné-noviny-327>

Jaderné noviny – 12. 4. 2006

Robert Krátký

Aktuální verze jádra: 2.6.16.5. Kontejnery a odlehčená virtualizace. Systémová volání `tee()` a `splice()`. Zachování údajů o oops při resetu. Neostré hranice GPL.

Aktuální verze jádra: 2.6.16.5

Aktuální verze stabilního jádra je [2.6.16.5](#) [1]. Byla vydána 7. dubna s početnou řádkou oprav; [2.6.16.3](#) [2] a [2.6.16.4](#) [3] – z nichž každá obsahuje jen jedinou bezpečnostní opravu – vyšly obě 11. dubna a [2.6.16.5](#) [4], se dvěma opravami pro x86-64, byla vydána 12. dubna.

Aktuální předverze je i nadále [2.6.17-rc1](#); během minulého týdne žádné další předverze nevyšly. [2.6.17-rc2](#) je však zjevně na spadnutí a v době, kdy tohle čtete, už je možná venku. Patche začleněné od [2.6.17-rc1](#) jsou většinou opravy, ale najde se i několik významnějších změn, včetně zjednodušené formy patche pro předcházení hladovění scheduleru, vyladění algoritmu používaného při přetížení paměti [memory overcommit], odstranění nespravovaných ovladačů `blkmttd` a `Sangoma WAN` a změn v systémovém volání `splice()`.

Aktuální -mm strom je [2.6.17-rc1-mm2](#). Mezi nedávné změny patří nová sada patchů pro sledování zdrojů na 64bit a úpravy kódu pro výpis jádra [`core dump`].

Kontejnery a odlehčená virtualizace

„Virtualizace“ spočívá v přesvědčení skupiny procesů o tom, že má pro sebe vyhrazený systém. K problému virtualizace je přistupováno několika způsoby, přičemž Xen, VMWare a User-mode Linux patří mezi ty známější možnosti. Jde však o relativně těžkotonážní řešení, protože pro každý virtuální stroj je spouštěno samostatné jádro. To je často vhodným řešením; provoz nezávislých jader zaručuje pevné oddělení jednotlivých prostředí a umožňuje spuštění více operačních systémů na jednom hardwaru.

Plná virtualizace a paravirtualizace ale nejsou jedinými způsoby. Alternativou je odlehčená virtualizace, která bývá založena na nějakém druhu kontejnerového konceptu. V případě kontejnerů to vypadá, jako by skupina procesů i tak měla svůj vyhrazený systém, ale ve skutečnosti běží ve speciálně izolovaném prostředí. Všechny kontejnery jsou spuštěny nad jediným jádrem. Při použití kontejnerů je ztracena možnost provozovat různé operační systémy a oddělení virtuálních systémů také není tak pevné. Na druhou stranu mají kontejnery nezanedbatelné výhody z hlediska výkonu, takže jich může být na jednom fyzickém hostiteli spuštěn velký počet.

Kontejnerových projektů rozhodně není nedostatek. Řadí se mezi ně relativně prosté způsoby, například [BSD jail module](#) [5], ale také propracovanější – například [Linux-VServer](#) [6], [OpenVZ](#) [7] a proprietární [Virtuozzo](#) [8] (založený na OpenVZ). Mnohé z těchto projektů by rády protlačily alespoň nějaký kód do jádra, aby se zbavily břemene udržování patchů mimo hlavní strom. Není však velký zájem o začlenění kódu, který podporuje jen některé z těchto projektů. Lidé zabývající se kontejnery se budou muset dohodnout na společných řešeních, která by mohli používat všichni.

A vypadá to, že přesně o to se snaží. Začala platit [volná dohoda](#) [9], v jejímž rámci budou vývojáři z několika projektů diskutovat o navrhovaných změnách a snažit se je dostat do stavu, který by vyhovoval potřebám všech. Jakmile se konkrétní patch dostane tak daleko, že jej budou všichni vývojáři ochotni odsouhlasit, bude přeposlán k začlenění do hlavního jádra.

Zdá se, že komplexnější a náročnější změny, jako třeba [virtualizace PID](#) [10], jsou prozatím pozastaveny. Místo toho to vypadá, že prvním společně odsouhlaseným patchem bude virtualizace [jmeného](#)

prostoru UTS [11]. Cíl patche je poměrně přímočarý: umožňuje každému kontejneru (reprezentovanému stromem procesů) mít vlastní verzi struktury `utsname`, která bude držet jméno uzlu, jméno domény, verzi operačního systému a pár dalších věcí. V podstatě to nahrazuje jedinou globální strukturu několika strukturami připojenými na různá místa stromu procesů. Je k tomu potřeba pětidílný patch, který nahradí každou zmínku o globální struktuře `system_utsname` voláním nové funkce `utsname()`.

Dlouhodobé plány mluví o virtualizaci všech globálních jmenných prostorů v jádře – včetně SYSV IPC, ID procesů a dokonce pravidel netfilteru. O virtualizaci bezpečnostních modulů proběhla zajímavá diskuze; někteří si myslí, že každý kontejner by měl mít možnost nahrát svou vlastní bezpečnostní politiku, zatímco další argumentují ve prospěch jediného systému bezpečnostní politiky, který si je kontejnerů vědom (a je schopen je využívat). Nepřekvapí, že SELinux už je vybaven mechanismem pro hierarchii typů, který může být s kontejnery používán při řešení s jednotnou bezpečnostní politikou.

Některé vývojáře ještě možná bude těžké o kontejnerech přesvědčit, protože v nich budou spatřovat komplikaci pro přístup k mnoha interním datovým strukturám jádra, aniž by přinášely nějak moc výhod. Je však zřejmé, že po tomto druhu odlehčené virtualizace existuje poptávka – samotné OpenVZ tvrdí [12], že provozuje přes 300 tisíc virtuálních prostředí. Takže tlak na standardizaci kódu a přesunutí do hlavního jádra bude časem ještě sílit. Až bude kód dostatečně čistý, aby s ním byla vývojářská komunita spokojena, je pravděpodobné, že budou části kontejnerového konceptu začleněny.

Systemová volání `tee()` a `splice()`

O novém systémovém volání `splice()` byla řeč minulý týden. Jak jsme očekávali, nové API prošlo dalším vývojem; mnohé z patchů, které neobsahovaly opravy chyb, se týkaly změn v `splice()`. Především se změnil prototyp `splice()`:

```
long splice(int fd_in, loff_t *off_in, int fd_out, loff_t *off_out,
            size_t len, unsigned int flags);
```

Dvě nové offsetové hodnoty (`off_in` a `off_out`) označují, kde má být před začátkem přenosu dat umístěn popisovač souboru. Všimněte si, že tyto offsety jsou předávány ukazateli, takže uživatelský prostor může využít ukazatel `NULL`, a tím dát najevo, že má být použit aktuální offset. Také si všimněte, že tyto offsety nefungují stejně jako offsety v `pread()` nebo `pwrite()`: skutečně změní offset přiřazený k popisovači souborů. Zadání offsetu pro popisovač souboru přiřazený k rouře je chyba.

Samotnému kódu `splice()` se dostalo několika zajímavých změn. Jednou z nich (vlastně v kódu roury) je vytvoření nové struktury `pipe_inode_info`, která reprezentuje hlavní funkčnost roury. Tato struktura může být oddělena od běžné struktury `inode`. Mnohá interní rozhraní byla změněna tak, aby novou strukturu využívala. To se týká i nových metod ve struktuře `file_operations`:

```
ssize_t (*splice_write)(struct pipe_inode_info *pipe,
                        struct file *out, size_t len,
                        unsigned int flags);
ssize_t (*splice_read)(struct file *in, struct pipe_inode_info *pipe,
                       size_t len, unsigned int flags);
```

Protože je implementací těchto metod v jádře pořád málo, nejde o příliš rušivé změny. Další je na seznamu podpora pro přímé „splicování“ dvou popisovačů souborů, z nichž ani jeden není roura.

Tato funkčnost (zatím) není prostřednictvím `splice()` dostupná uživatelskému prostoru, ale interně je využívána k implementaci `sendfile()` se `splice()` mechanismem. Přímé „splicování“ je implementováno pomocí skryté struktury `pipe_inode_info` (tj. roury); je uloženo v novém poli `splice_pipe` struktury úloh, takže každý proces může mít v určitý okamžik puštěno jen jedno takové spojení. Jeden patch začleněn nebyl – a pravděpodobně si bude muset počkat až na 2.6.18 – systémové volání `tee()` [13]:

```
long tee(int fdin, int fdout, size_t len, unsigned int flags);
```

Toto volání vyžaduje, aby oba popisovače souborů byly roury. Jednoduše propojí `fdin` na `fdout` a přeneše mezi nimi až `len` bajtů. Na rozdíl od `splice()` však `tee()` nespolyká vstup, takže mohou být vstupní data volajícím procesem později normálně přečtena. Jens Axboe nabídl vzorovou implementaci uživatelské utility `tee`, která není nic než pár volání:

```
len = tee(STDIN_FILENO, STDOUT_FILENO, INT_MAX, SPLICE_F_NONBLOCK);
splice(STDIN_FILENO, out_file, len, 0);
```

Vstupní data budou zapsána jak na standardní výstup, tak do souboru určeného pomocí `out_file`, aniž by byla kdy kopírována z nebo do uživatelského prostoru. Aby se zajistilo zkopírování kompletního proudu vstupních dat, musí samozřejmě aplikace provádět výše zmíněná volání ve smyčce. Celý příklad najdete na konci patche s voláním `tee()` [14]. Toto volání je docela nové a možná bude změněno, než se objeví v hlavním stromě. Kromě jiného by mohlo být přejmenováno, protože něco tak jednoduchého jako `tee()` už je možná použito v mnoha jiných aplikacích.

Wireless Networking Summit 2006

Reportáž o setkání vývojářů podpory pro bezdrátové síťování v Linuxu byla kvůli své délce vydána jako samostatný článek: [Wireless Networking Summit 2006](#) [15]. Následující obsah je ©KernelTrap.

Zachování údajů o oops při resetu

10. dub, originál [16]

James Courtier se v konferenci LKML ptal, jestli je možné obnovit kruhový buffer jádra po resetu. Navrhl prostě zapsat data v mnoha kopiích do paměti a doufat, že při bootu nebude vymazána veškerá RAM, což umožní obnovu bufferu. Kruhový buffer jádra je obvykle prohlížen pomocí příkazu `dmesg`. James přiblížil (přičemž se odkazoval na metodu, při které jsou údaje o oops sbírány přes sériovou linku): **Něco takového by pomohlo hlavně u novějších základních desek, které nemají sériový port.**

Existující řešení je použít `kexec` k naboťování speciálního odlehčeného jádra, se kterým pak lze posbírat jaderný výpis o pádu.

Všeobecný názor byl, že data zapsaná do RAM před resetem už v ní po restartu nebudou, i když se diskutovalo o tom, kolik přesně paměti bude přepsáno. Eric Biedermann, autor `kexec`, vysvětlil:

Vyčištění paměti lze provést nejvyšší podporovanou rychlostí, takže jsou to vteřiny. Na systémech s ECC je potřeba inicializovat všechny kontrolní bity, a proto nějaký zápis do paměti musí proběhnout.

K tomu doplnil: **V praxi reset paměť nečistí a jen pár bitů bude převrácených.**

Andi Kleen navrhl jiné řešení: **Definuj obecné rozhraní, které ovladačům umožní registrovat obsluhovače pro ukládání do paměti [memory storage handlers]. Do funkcí, které při oops oznamují `die` a `panic`, přidej záznam, který jaderný log uloží do těchto záloh.**

Jako příklad Andi uvedl ovladače grafiky, které by mohly kousek RAM na kartě vyšetřit pro uchování údajů o pádu během restartu.

Neostré hranice GPL

12. dub, originál [17]

Linus Torvalds vydal v září 1991 linuxové jádro nejprve s velmi omezující licencí, která vyžadovala, aby zdrojový kód zůstal vždy dostupný a nikdo na něm nevydělával peníze. O pár měsíců později přešel na GPL, neboli Všeobecnou veřejnou licenci – a od té doby už byla pro jádro použita vždy. Nedávná diskuze ve vývojářské konferenci se zabývala neostrými hranicemi legálnosti v oblastech, kde není zcela jasné, co GPL povoluje. Alan Cox byl jedním z mnoha vývojářů, kteří vysvětlovali:

Hranicí GPL je takzvané „odvozené dílo“. To je základní právní koncept používaný v copyrightu, který klade otázku: „Je toto dílo vytvořeno takovým způsobem, aby bylo možné říci, že je založeno na původním díle?“ Jde o komplikovanou oblast práva a pouze právník může dát definitivní odpověď. „Aplikace pro Linux“ je jasný případ. „Úprava jádra“ je také jasná. Mezi tím je mlhavý prostor pro právníky.

Odkazy

- [1] <http://lwn.net/Articles/179043/>
- [2] <http://lwn.net/Articles/179477/>
- [3] <http://lwn.net/Articles/179483/>
- [4] <http://lwn.net/Articles/179668/>
- [5] <http://lwn.net/Articles/120752/>
- [6] <http://linux-vserver.org/>
- [7] <http://openvz.org/>
- [8] <http://www.virtuozzo.com/>
- [9] <http://lwn.net/Articles/179368/>
- [10] <http://lwn.net/Articles/171017/>
- [11] <http://lwn.net/Articles/179345/>
- [12] <http://lwn.net/Articles/179395/>
- [13] <http://lwn.net/Articles/179434/>
- [14] <http://lwn.net/Articles/179434/>
- [15] <http://www.abclinuxu.cz/clanky/jaderne-noviny/jn-reportaz-wireless-networking-summit-2006>
- [16] <http://kerneltrap.org/node/6452>
- [17] <http://kerneltrap.org/node/6452>

Jaderné noviny: Wireless Networking Summit 2006

Robert Krátký

Bývaly časy, kdy bylo nastavení linuxového systému zdlouhavým a problematickým procesem, přičemž nebyla žádná jistota, že daný systém bude fungovat bez velkých potíží. Většina těchto problémů je již léta překonána a do jisté míry lze očekávat, že Linux bude „prostě fungovat“. Zbývá však několik problémových oblastí – a bezdrátové síťování je jednou z nich. I když je hardware podporován (což často nebývá), plné zprovoznění bezdrátového připojení může být dost obtížné.

Článek vyšel původně jako součást zpravodajství o jádře na serveru LWN a jako takový by patřil do článku *Jaderné noviny* – 12. 4. 2006. Kvůli jeho délce jsem jej však zařadil samostatně.

V oblasti bezdrátového síťování se však děje hodně. Aby se věci pohnuly dále kupředu, zorganizovaly Open Source Development Labs (OSDL) v Portlandu 6. a 7. dubna setkání vývojářů pracujících na bezdrátovém síťování. Podařilo se tak dát dohromady různorodou skupinu vývojářů z celého světa, z nichž mnozí se setkali poprvé. Účel setkání – načrtnout cestu k vytvoření rozumně fungující implementace bezdrátového síťování v Linuxu – byl, jak se zdá, z větší části naplněn.

Váš redaktor [Jonathan Corbet] měl to štěstí, že se mohl setkání zúčastnit. Následující reportáž je pokusem o shrnutí závěrů setkání – nejedná se o podrobné zápisky minutu po minutě a v průběhu psaní budu muset provést trochu chronologických úprav. Doufám, že z výsledku bude patrné, jak si věci v současné době stojí, a kam budou směřovat v budoucnu.

Historie

Jak podotkl John Linville (nedávno jmenovaný správcem bezdrátového síťování) během našeho rozhovoru, rané bezdrátové adaptéry byly prodávány jako „bezdrátový Ethernet“, a vývojáři linuxového jádra s nimi nakládali jako s pomalými, nespolehlivými a nešikovnými ethernetovými adaptéry. Ale bezdrát není Ethernet ani v nejmenším – jde o naprosto odlišné síťovací standardy s vlastními manýry, speciálními funkcemi a jinými potřebami. Nahlížení na bezdrát, jako by šlo o formu Ethernetu, zpomalovalo implementaci podpory zmiňovaných speciálních funkcí a především ztěžovalo vývoj jakékoli interní podpory v jádře. Každý vývojář, který se pustil do vývoje ovladače pro bezdrátový adaptér, nakonec musel všechno implementovat od nuly. Neexistovalo tedy žádné obecné API pro bezdrátové síťování, žádná ucelená podpora bezdrátových funkcí a kód ovladačů byl nejednotný navzdory množství duplicit.



V roce 1997 se Jean Tourrilhes rozhodl něco se situací provést. Výsledkem byly WE-1 – první verze Linux wireless extensions (rozšíření pro bezdrátové síťování). V té době ještě nebyl žádný 802.11

standard, ale WE API umožňovalo konfiguraci a provoz bezdrátových adaptérů s pomocí jediné sady nástrojů. Jeanovy wireless extensions jsou i nadále hlavními utilitami pro správu bezdrátových adaptérů, i když většina uživatelů je nahrazuje grafickými rozhraními.

Vývoj wireless extensions pokračoval – WE-9 podporovaly v roce 1999 nový standard 802.11. Minulý rok začleněné WE-18 přidávají podporu WPA (WiFi Protected Access – chráněný přístup). Aktuální verze WE-20 obsahuje nové rozhraní založené na netlink, které má v budoucnu nahradit současné ioctl() API.

Ačkoliv vývoj pokračuje, všeobecně převládající názor naznačuje, že doba, po kterou byly wireless extensions považovány za užitečné, se chýlí ke konci. Náhradní API – které zatím neexistuje – by mělo pracovat s celým stackem pro bezdrátové síťování, místo aby bylo rozhraním pro přímou komunikaci z nízkoúrovňovými ovladači. Avšak bez ohledu na to, jak to dopadne, je pravděpodobné, že wireless extensions tu budou ještě dlouho.

Současný stav

Současná snaha o vytvoření řádného bezdrátového stacku pro Linux začala v roce 2004, když Jeff Garzik [oznámil](#) [1] založení speciálního stromu pro bezdrátové síťování, který byl zpočátku naplněn kódem HostAP. Začlenění HostAP umožnilo podporu relativně nových síťových karet a využívání Linuxu jako bezdrátového přístupového bodu. Vytvoření stromu pomohlo dát věci do pohybu, ale HostAP nakonec nenaplnil tak docela očekávání. Konkrétně není v HostAP podpora pro karty, které potřebují implementaci softwarové MAC („softmac“). Nemálo dnešních karet však s mnoha nízkoúrovňovými operacemi spoléhá na hostitelský software; tyto karty nemohou být stackem pro bezdrátové síťování současného (2.6.16) jádra podporovány.



Výsledkem je, jak řekl John Linville, implementace, která podporuje „vše, co je zastaralé“. Některé karty se těší větší podpoře než jiné. Ze současného hardwaru stojí za zmínku hlavně sada Intel IPW, která je, díky Intelu, jádrem velmi dobře podporována – ale tyto karty nevyžadují podporu softmac.

V tuto chvíli chybí menší množství okrajově potřebných funkcí, včetně podpory velmi široce používaného hardwaru. Nedostatečná je také uživatelská přívětivost – přes pokroky grafických nástrojů může být konfigurace bezdrátového připojení nepříjemná zkušenost. Nejlépe tyto dva problémy pravděpodobně vystihovala situace na samotném setkání, kde musela přibližně čtvrtina účastníků používat ethernetový kabel, aby se připojili přímo k síti OSDL.

Mezi další problémy patří jednotnost (či spíše její absence) hardwaru – v jádře a v ovladačích mimo strom je neustále několik API se specifickými funkcemi pro různé adaptéry. Dokumentace API jaksi neexistuje; zazněla stížnost, že kniha Linux Device Drivers nepopisuje, jak programovat bezdrátové ovladače. Neexistuje žádný koordinovaný postup pro rozšiřování API. Není podpora pro kvalitu služby – k této otázce se za chvíli vrátíme. V oběhu nejsou žádné testovací balíky pro ovladače.

Nad bezdrátovým síťováním se vznáší otázka regulace, která je největší příčinou buď neexistujících nebo nezačleněných ovladačů. Hodně výrobců má prostě pocit, že nemohou zveřejnit informace pro programátory nebo svobodné ovladače a zároveň neporušit regulační režimy na celém světě.

Jádro 2.6.17 však přináší několik vylepšení své podpory bezdrátového síťování. John začlenil jednu z mnoha dostupných implementací softmac – na základě předpokladu, že jako jeden z neaktivnějších projektů pomůže s vývojem ovladačů. Byl také začleněn ovladač bcm43xx (Broadcom), který softmac používá, a ve vývoji je pár dalších ovladačů založených na softmac. Přesto však převládá názor, že softmac není krokem vpřed; skutečná budoucnost linuxové podpory bezdrátového síťování leží ve stacku od Devicescape.

Devicescape

Devicescape [2] je společnost nabízející množství produktů a služeb týkajících se bezdrátového síťování. V rámci vývoje své nabídky vytvořila Devicescape svůj vlastní linuxový 802.11 stack s několika pěknými funkcemi – včetně dobré podpory softmac a WPA. Nedávno byl stack vydán s GPL a Jiří Benc [3] ho upravil pro jádro. Mnoho lidí jej považuje za nejlepší svobodný stack.

Když spravoval bezdrátový strom Jeff Garzik, postavil se rezolutně proti přechodu na stack od Devicescape s tím, že místo toho by měl být kód v jádře přizpůsoben potřebám. Vypadá to však, že je teď v menšině a John Linville pravděpodobně stack začlení do některé budoucí verze jádra. John spravuje samostatný vývojový strom obsahující Devicescape a několik ovladačů (například bcm43xx) bylo na stack portováno. Na setkání se neobjevil nikdo, kdo by měl proti začlenění Devicescape námitky.

Simon Barber, hacker z Devicescape, o tomto kódu mluvil, oddělilo se i samostatné sezení, které se této otázce věnovalo. Tento stack je velký kus kódu. Uvolněná část obsahuje 802.11 stack, kód „openap“ pro přístupové body a modul mostu [brindging] s linkovací vrstvou. Brzy bude uvolněna i další část, která obsahuje zlepšení démona hostapd (mimo jiné i podporu 802.11g; tento kód je teď začleňován), integraci propojování a VLAN a různá vylepšení Ethernet pro vývojáře bezdrátového síťování. Pracuje se také na „kompletní distribuci pro domácí gateway“. Všechno tento kód bude zpřístupněn na připravovaném portálu.

Předpokládá se, že na Devicescape stacku bude ještě hodně práce. Interně se skládá z dlouhého seznamu obslužných funkcí, které se starají o rámce (jak datové pakety, tak rámce pro správu 802.11) na jejich cestě od a k adaptéru. V plánech do budoucna figuruje umožnění nahrávání modulů s vlastními obslužnými funkcemi. Více kódu pro správu bude také přesunuto do uživatelského prostoru. Kvůli tomu bude hostapd [4] obohacen o možnosti, které se budou starat autentizaci a správu. V plánu je také spojení hostapd s wpa_supplicant [5], který má na starosti proces autentizace ze strany klienta; hodně věcí bude zjevně snazších, budou-li tyto dvě funkce sjednoceny do jednoho procesu.

Velké zesložnění bude také zapříčiněno dlouhou řadou budoucích 802.11 standardů. Ty budou postupně schvalovány a bude nutné je podporovat.

Jedna ze zajímavých oblastí vývoje se týká podpory pro kvalitu služby. 802.11 definuje 4 úrovně služby: „hlas“, „video“, „největší úsilí“ a „pozadí“. Každá z úrovní má svůj rozsah priorit, rozsahy se překrývají. Všechny hlasové pakety půjdou dříve než pakety pozadí, ale zbytek úrovní bude sdílet dostupnou šířku pásma. Se správnou podporou QoS (Quality of Service – kvalita služby) může uživatel bezdrátu konverzovat pomocí VoIP, streamovat video s „nejčerstvějšími zprávami“ o zahlednutých celebritách ze CNN, stahovat nové jádro z FTP a distribuovat materiály (radši se neptejte jaké) přes Bittorrent. Každá činnost může běžet s vlastní úrovní kvality služby a všechny by měly mít k dispozici nejlepší dostupný výkon.

Některé bezdrátové síťové adaptéry mají podporu kvality služeb ve formě čtyř samostatných přenosových front. Umístí-li hostitel každý paket do příslušné fronty, rozdělí mezi ně adaptér dostupné pásmo tak, aby to odpovídalo kvalitě služby každé z úrovní. Potíží je, že linuxový síťovací stack pod-

poruje pouze jednu přenosovou frontu u každého zařízení. To představuje problém ve chvíli, kdy se zaplní jedna z těch čtyř úrovněových front na zařízení. Neexistuje způsob, jak například jádru říci, že už není možné zařadit do fronty další pakety, ale místo na hlasové pakety by ještě bylo; ovladači nezbývá než uzavřít frontu všem paketům.

Hackeri v Devicescape tento problém obešli s pomocí mechanismu pro ovládání provozu zabudovaného do síťovacího stacku, který běžně operuje na úrovni, kterou kód ovladače nevidí. Díky vytvoření samostatné interní fronty pro každou úroveň služeb dokáže Devicescape stack pro každou úroveň implementovat i samostatnou přenosovou frontu. A ještě lépe, tím pádem je možné nastavit pravidla – které druhy provozu dostanou jakou úroveň – z uživatelského prostoru s použitím běžných nástrojů pro ovládání provozu. Bylo by však fajn, kdyby se toto využití kódu pro ukázněné řazení do fronty zobecnilo a zpřístupnilo i jiným druhům hardwaru.

Další oblast, na které je potřeba pracovat, je definice API pro uživatelský prostor. Není žádné dobře srozumitelné API, které by mohla použít například grafická utilita pro správu bezdrátových spojení ke komunikaci se síťovacím stackem a procesy jako hostapd. Dokonce se ani nemluví o tom, jak by takové API mělo vypadat.

Mezi další otevřené otázky patří obvyklé handrkování o regulacích, neexistence uživatelského prostředí pro správu MAC vrstvy, potřeba lepšího skenování, podpora adaptérů, které spravují MAC hardwarově, podpora ovládání napájení [power management] a přepracování konfiguračního rozhraní. O konfiguraci se starají volání `ioctl()` a rozhraní `/proc`. Bylo dáno zřetelně najevo, že kód od Devicescape se do jádra nedostane, dokud bude obsahovat `/proc` soubory. Také to vypadá, že stack bude muset být upraven, než začne správně fungovat na SMP systémech.

A konečně, přidání řádné podpory bezdrátového síťování do jádra bude obnášet vytvoření specifického `net_device` typu pro 802.11 zařízení. Také by měla být definována `sk_buff` struktura specifická pro 802.11. Současný kód stále využívá ethernetové typy a ty další potřebné informace tahá bokem.

Největším problémem však může být toto: co se stane právě začleněnému `softmac` kódu po začlenění Devicescape? Spousta funkcí je tam duplicitně a nikoho neláká představa spravování dvou různých 802.11 stacků donekonečna. Zřetelně to připomíná situaci s ovladači zvukových karet ALSA a OSS; ALSA měla OSS nahradit, ale odstranění OSS ovladačů se ukázalo jako dost složité. Není jasné, jak zařídit, aby bylo odstranění `softmac` snazší.

Nástroje

Setkání se účastnili především vývojáři zaměřeni na jádro, ale mluvilo se i o uživatelských nástrojích; byl tam Daniel Williams, hacker programu [NetworkManager](#) [6]. Všichni se shodují v tom, že ačkoliv se během posledních pár let dostupné nástroje výrazně zlepšily, stále je potřeba ujít kus cesty. Konkrétně jde o konfiguraci rozhraní, která funguje bezbolestně až do chvíle, kdy se něco pokazí. Pak je po pohodlí.

Zlepšení situace si bude vyžadovat podporu ze strany jádra. Když se vyskytne problém, musí mít uživatelský prostor informace o tom, co se vlastně stalo. Neexistuje jednotná sada chybových kódů vracených jádrem, které by například mohly sdělit, že není k dispozici potřebný firmware, nebo že poskytnutý WEP klíč není platný. Některé ovladače mají podporu pro větší část současného API než jiné a dokumentace k API v podstatě neexistuje. Lepší skenování by se také hodilo.

Podpora hardwaru

Přestože je nezbytné zdokonalit uživatelské nástroje a síťovací stack, lepší podpora hardwaru je také nutným krokem k implementaci bezdrátového síťování, která „prostě funguje“. Některé druhy hardwaru jsou už teď podporovány velmi dobře (Intel a další), jiné budou mít podporu brzy (Broadcom). Na některé (Atheros čipset) si možná ještě počkáme. Stávající ovladač pro Atheros (součástí

OpenBSD) má pravděpodobně kód z natolik pochybného zdroje, že šance na začlenění do Linuxu je mizivá. Je tu snaha o dokumentaci Atheros hardware na základě dostupného kódu, aby byla v budoucnu možná čistá implementace, ale stále ještě zbývá udělat hodně práce.

V této souvislosti přišla na přetřes otázka dodržování regulačních pravidel. Některé adaptéry (například Atheros) jsou vlastně obyčejná rádia, která lze naprogramovat k provozu úplně mimo specifikaci 802.11. Je-li pro takový hardware vyvinut svobodný ovladač, bylo by Správné zajistit, aby byl s jeho pomocí hardware provozován v souladu s příslušnými předpisy, i když by jej jako takový nešlo certifikovat. Takové testování vyžaduje specializované vybavení a jedná se o několikadenní záležitost. Potřebné vybavení mají k dispozici firmy jako Nokia a některé univerzity, ale v současné době není vyvíjena žádná snaha získat k němu kvůli testování přístup.

Většina vývoje ovladačů se odehrává mimo hlavní strom a vývojáři jádra by to rádi změnili. Jakmile se kód dostane do hlavního stromu, je snazší ho kontrolovat a vylepšovat. Greg Kroah-Hartman nabádal vývojáře ovladačů, aby svůj kód začlenili co nejdříve – i když ještě nefunguje.

Bylo dohodnuto, že komunikace týkající se ovladačů bezdrátových zařízení zůstane prozatím v konferenci netdev. Kdyby se někdy zdálo, že by tato témata mohla provoz netdev zahltit, bude vytvořena nová konference. V blízké budoucnosti bude pravděpodobně také sestavena informační stránka o ovladačích bezdrátových zařízení.

Další otázky



Jedním z cílů setkání bylo vtáhnout více zúčastněných vývojářů do širšího vývojového okruhu jádra. Kvůli tomu se mluvilo také o systémech pro správu verzí, především i git a guilt. Často se připomínal přístup „začlenit co nejdříve“.

Stephen Hemminger promluvil o stavu kódu mostů. Mosty vývojáře bezdrátového síťování zajímají – lze je použít pro sdílení spojení a aplikace pro mesh [síťové, pletivové] síťování. Kvůli tomu bude asi propojovací kód přepracován a přesunut do uživatelského prostoru. Stejně jako je teď routování většinou obstaráváno uživatelskými démony, správa mostů – včetně správy spanning [napnutých] stromů – se v budoucnu přesune do uživatelského prostoru.

Někteří zástupci [Personal Telco Project](http://www.personaltelco.net) [7] měli dostatek kuráže k tomu, aby se během oběda utkali s donáškou pizzy o pozornost vývojářů. Sestavili síť 100 linuxových bezdrátových přístupových bodů po celém Porlandu. Na vývojáře jádra měli několik požadavků, včetně svobodných ovladačů pro Atheros, které neshodí systém, a dobrého bezkonfiguračního mesh síťování. Je to zajímavý projekt, který ukazuje, co zvládne pár „nezaměstnaných geeků“.

Celkově bylo setkání optimistickou akcí. Zatímco nedostatky podpory bezdrátového síťování v Linuxu byly identifikovány a pochopeny, bylo zřejmé, že mohou být nejen vyřešeny, ale že některá řešení jsou už velmi vyspělá. Půjde-li vše podle plánu, tak den, kdy bude bezdrátové síťování v Linuxu „prostě fungovat“, není daleko.

Odkazy

- [1] <http://lwn.net/Articles/87952/>
- [2] <http://devicescape.com/>
- [3] <http://www.abclinuxu.cz/Profile/3083>
- [4] <http://hostap.epitest.fi/hostapd/>
- [5] http://hostap.epitest.fi/wpa_supplicant/
- [6] <http://www.gnome.org/projects/NetworkManager/>
- [7] <http://www.personaltelco.net>

Zprávičky

1.4.2006

rtx511

Vyšla verze 2.0 oblíbeného hudebního přehrávače amaroK. Mezi hlavní novinky patří podpora tagů last.fm, panelové playlisty a podpora MS Windows.

2.4.2006

Petr Tomeš

Ubuntu Flight 6, nejnovější alfavydání Dapper Drake, je připraveno. Tato vydání jsou testována, aby neobsahovala nepřiměřené množství zásadních chyb, ale kvalitativně jsou stále na úrovni alfaverze, takže je nepoužívejte na pracovních počítačích. Seznam důležitých změn tohoto vydání v rámci celé distribuce je (díky Mattovi Galvinovi) dostupný na: http://wiki.ubuntu.cz/Dapper_Flight_6

2.4.2006

David Watzke

Vyšla nová verze Wine (0.9.11), která přidává pár systémových DLL knihoven pro zlepšení podpory instalátorů, opravuje pár problémů s OpenGL a jako vždy přináší spousty dalších menších vylepšení.

3.4.2006

Robert Krátký

Nachytil vás někdo letos aprílem? Wikipedia má (stále rostoucí) seznam internetových žertíků, které na své čtenáře tento rok weby nastražily.

3.4.2006

Robert Krátký

Jak zapnout a vypnout spouštění jednotlivých služeb při startu systému? Článek Enabling and disabling services during start up in GNU/Linux na blogu All about Linux vám poradí, jak to provést v Red Hatu, Debianu, Gentoo a „po staru“.

3.4.2006

Robert Krátký

Každý zná wget. Nebo ne? Každopádně vám může přijít vhod několik tipů na pokročilejší použití tohoto nástroje: Mastering Wget (stahování celých stránek, automatické stahování hudby apod.).

3.4.2006

Robert Krátký

Jak nastavit FreeBSD tak, aby bylo možné spouštět programy zkompilevané pro Linux, vám poradí článek na NewsForge: Setting up Linux compatibility on FreeBSD 6.

3.4.2006

Robert Krátký

Guardian přináší portrét Ebena Moglena, právníka Free Software Foundation a spoluautora nové

verze GPL: 'A lawyer who is also idealist – how refreshing'.

3.4.2006

Daniel Kvasnička ml.

Členové argentinské CaFeLUG, kteří mají na kontě mnoho různých konferencí a installfestů, popisují (NewsForge) jak takové akce organizovat, čeho se vyvarovat a co naopak dobře funguje.

3.4.2006

Daniel Kvasnička ml.

TheServerSide.NET přináší rozhovor s Miguelem De Icazou, vývojářem majícím na svědomí Mono, open-source implementaci platformy .NET. Mluví se o současnosti a budoucnosti Mona a dalších zajímavých tématech jako implementace Ruby pro .NET či používání Mona na Windows.

3.4.2006

Daniel Kvasnička ml.

Matt Asay z InfoWorldu reaguje na nenápadné narážky Steva Ballmera na Linux a jeho možné porušování intelektuálního vlastnictví Microsoftu. Asay konstatuje, že tyto výroky jen dokazují, že krize MS není malá.

3.4.2006

komodo

Po dlhé době vyšla nová verzia prehrávača Kaffeine. Hlavne bola vylepšena podpora DVB a s tým súvisiacích funkcií ako EPG a podobne.

3.4.2006

Lukáš Zapletal

Dubnové číslo časopisu LinuxEXPRES doznalo menších změn v designu. Je věnováno virtualizaci a emulaci: XEN, VMware, WINE, Dosbox, Atari nebo souborovým systémům pomocí cloop zařízení. Ukázka čísla ve formátu PDF.

3.4.2006

Vladislav Korecký

Microsoft údajně umožní běh GNU/Linuxu na strojích s MS Windows. K tomuto účelu dá k dispozici Virtual Server 2005 R2, jenž běh Linuxu ve Windows umožňuje, a to zcela zdarma. Více na serveru novinky.cz.

3.4.2006

Bob Bobovich

Máte-li potomka bažícího po vychrtlých panenkách od Matella, můžete jej překvapit stylovou distribucí Barbie OS. Nyní v betaverzi 0.99, na Vánoce prý finální 1.0. Pokud nemáte potomka, nezoufejte – zkuste štěstí v práci u sekretářky ;)

4.4.2006

Filip Molčan

Na nově spuštěném portálu LINUXBIZWORLD vyšly dva zajímavé rozhovory na téma OSS/FS se zástupci společností Microsoft a BSA.

4.4.2006*elviin*

Sun Microsystems představuje preview verzi NetBeans C/C++ Development Pack. Balík umožňuje vývojářům v C a C++ psát, kompilovat a sestavovat aplikace v C a C++ na vícero platformách, zejména Solaris, Linux a Windows. Plug-in je podporován v NetBeans 5.0 a NetBeans 5.5.

4.4.2006*Katarína Machálková*

Příznivci OpenSolarisu a všichni zájemci o nové technologie jsou zváni na sedmé setkání CZOSUG spojené s přednáškou, které se bude konat v úterý 11. dubna od 18:00 v posluchárně S4 na MFF UK. Program je tentokrát nabitý: Luboš Koščo představí Sun Grid Engine a po něm vystoupí Frank Hoffman se svojí přednáškou na téma „How to write a filesystem driver for Solaris“.

4.4.2006*bpsa*

Na <http://stellarium.sourceforge.net> si můžete stáhnout planetárium, které vám umožní vykreslovat hvězdnou oblohu, a to opravdu dobře. Převzato z LinuxEXPRES 12/05.

4.4.2006*Lubomír Pastorek*

Na www.kde-look.org sa objavila informácia o rozbehání KDE na XGL, k dispozícii je aj video, download cez torrent ktorý mne však nefungoval.

4.4.2006*Daniel Kvasnička ml.*

Nat Friedman ve svém blogu oznámil, že software pro jednoduché ukládání a zálohování dat iFolder je teď open source a to pod GNU GPL.

4.4.2006*Daniel Kvasnička ml.*

Jeden z vývojářů přehrávače Songbird oznámil, že aplikaci se již podařilo spustit na Linuxu. Nevypadá nejlíp a také občas padá, ale jede.

4.4.2006*Daniel Kvasnička ml.*

Vyšel Python & Jython plug-in pro Eclipse PyDev 1.0.4. Přibylo několik nových vlastností týkajících se práce v Jythonu a také zbrusu nový odsazovací engine. Download.

5.4.2006*Robert Krátký*

Nicholas Negroponte, předseda organizace připravující 100dolarové notebooky pro děti z rozvojových zemí, prohlásil na konferenci LinuxWorld, že Linux je jako tlustý člověk, který většinu své energie spotřebuje na to, aby se mohl hýbat.

5.4.2006*Robert Krátký*

OSDL ve spolupráci s freedesktop.org připravuje projekt Portland, který má vývojářům aplikací nabídnout společné rozhraní pro KDE a GNOME. Program pak bude lehce integrovatelný do zvoleného desktopového prostředí. Informuje The Inquirer.

5.4.2006*Robert Krátký*

Článek na IslamOnline.net popisuje, jak v Sýrii, na kterou Spojené státy uvalily technologické embargo, využívají svobodný software.

5.4.2006*Robert Krátký*

Red Hat se rozhodl reorganizovat Fedora Project, protože Fedora Foundation se za rok své existence příliš neosvědčila. Diskutovat o novém vývoji se bude také na nadcházejícím setkání FUDcon Boston 2006.

5.4.2006*Daniel Kvasnička ml.*

Peter Enseleit na Linux.com vypráví o své výpravě za dokonalým hudebním přehrávačem pro Linux. Nechybí ani přehled schopností různých přehrávačů ve formě tabulky či slušná porce odkazů: My quest for a Linux audio player.

5.4.2006*Daniel Kvasnička ml.*

Oracle na LinuxWorldu potvrdil, že linuxový kernel bude mít ve verzi 2.6.16 integrovanou podporu jejich souborového systému Cluster File System 2. FS je už dnes součástí RHEL 4 a SuSE Linux Enterprise a to pod GNU GPL. Informuje ZDNet.

5.4.2006*Daniel Kvasnička ml.*

NewsForge přináší (jako jediný pozvaný novinářský subjekt) reportáž z Government Day na výstavě LinuxWorld. Mluví se hlavně o ODF a s ním spojené kauze „Massachusetts“.

5.4.2006*Robert Krátký*

Firma PalmSource na probíhající konferenci LinuxWorld představuje svůj nový projekt pro embedded zařízení – Access Linux Platform. ALP nabízí J2ME, GTK+ a emulaci PalmOS.

5.4.2006*Max Devaine*

Společnost VMware uvolnila specifikace svého vmdisk (Virtual Machine Disk Format) souboru. Tyto specifikace obratem zašle po odeslání emailu s vaší žádostí.

6.4.2006 *Filip Molčan*
Pokud vás zajímá, co zajímavého bylo ohlášeno na konferenci LinuxWorld 2006 v Bostonu, sledujte LINUXBIZWORLD, který z konference přináší každý den čerstvé zprávy.

6.4.2006 *Robert Krátký*
Používáte Gnome a líbí se vám prostředí MacOS X? Třeba vám přijde vhod návod, jak Gnome vzhledově připodobnit prostředí Aqua: Making GNOME Look Like OS X.

6.4.2006 *Robert Krátký*
Chyby, které byly nalezeny v open source projektech v rámci automatické prohlídky pomocí softwaru od Coverity, jsou opravovány velmi rychle, říká Ben Chelf, CTO. Open source coders' speed astounds Coverity.

6.4.2006 *Robert Krátký*
Na IBM developerWorks vyšel článek o používání nástroje GNU profiler k nalezení nejpomalejších částí kódu a využití těchto informací k zlepšení výkonu programu: Speed your code with the GNU profiler.

6.4.2006 *Daniel Kvasnička ml.*
Eben Moglen (Software Freedom Law Center), s jehož pomocí R. Stallman pracuje na GPLv3, se nechal na LinuxExpu slyšet, že nová licence bude na rozdíl od Windows Vista a Office 12 k dispozici včas. Více na NewsForge.

6.4.2006 *Daniel Kvasnička ml.*
NewsForge představuje PHP nástroj pro generování flashových grafů: PHP/SWF Charts. Podporuje mnoho různých typů grafů, tisknutelné i animované grafy a funguje s PHP ver. 3, 4 i 5.

6.4.2006 *Michal Čihař*
Pokud vám vadí XSS chyby a používáte phpMyAdmin, tak by bylo záhodno upgradovat na právě vydanou verzi 2.8.0.3. Stahovat můžete ze stránek projektu.

6.4.2006 *Daniel Kvasnička ml.*
Vyšel Subversion 1.3.1 (změny, download), bugfix verze řady 1.3, ve které byl zvýšen výkon serveru i klienta, přidáno několik argumentů pro příkazovou řádku, výrazných změn doznalo i API pro Python a Ruby a bylo opraveno více než 30 bugů.

6.4.2006 *Martin Beránek*
Dnes v 17 hodin se koná chat uživatelů NetBeans. Jako klient slouží velmi zajímavý plugin – Developer Collaboration module. Jak se připojit zjistíte tady.

6.4.2006 *Daniel Kvasnička ml.*
NewsForge oznamuje výsledky soutěže SourceForge.net Community Choice Award.

6.4.2006 *Jose*
Dnes vyšla nová verzia distribúcie Kororaa XGL Live CD. Okrem opráv chýb a aktualizácii software pridáva tiež podporu pre grafické karty od Intelu skrz DRI.

6.4.2006 *Dunric*
Na ZDNetu vyšel článek o kampani zaměřené na britské prodejce PC proti prodeji počítačů bez předinstalovaného operačního systému. Těm se ale takový nátlak zřejmě moc nezamlouvá.

7.4.2006 *Daniel Kvasnička ml.*
Na stránkách Zenwalk Linuxu vyšel článek o tom, jak „používat“ server Distrowatch.com. Autor rozebírá hlavně jednotku HPD (Hits Per Day) a její roli jako ukazatele popularity distribucí.

7.4.2006 *Daniel Kvasnička ml.*
Bruce Byfield z NewsForge se zaměřil na firmu ActiveState, která se nedávno z divize Sophosu změnila na samostatný subjekt a která poskytuje užitečné nástroje pro práci s jazyky Python, Perl, PHP, Ruby a Tcl. V článku popisuje její výhledy do budoucnosti, business model a letmo se dotýká také vztahu k FOSS.

7.4.2006 *Daniel Kvasnička ml.*
Linux je možné provozovat téměř na čemkoliv, i na iPodu. Článek na Linux.com představuje iPodLinux Project a popisuje instalaci iPodLinuxu na iPod Nano.

7.4.2006 *Luboš Doležel*
NVIDIA vydala ovladač grafických karet ve verzi 1.0-8756. Podporuje nejnovější hardware, opravuje chyby s Composite a RenderAccel a obsahuje další vylepšení.

8.4.2006 *Michal Vyskočil*
Společnost UGS portovala svůj PLM (řešení pro správu životního cyklu výrobku) systém NX4 i pro Linux. Více oznámení, nebo computer design.

9.4.2006*Leoš Literák*

Pokud používáte PDA verzi Abíčka, přečtěte si zápisek v mém blogu. Hledám jak podněty na zlepšení, tak i správce této zanedbané verze.

10.4.2006*Robert Krátký*

The Register pokračuje ve sledování případu, na jehož počátku chtěl jeden oklahomský starosta volat FBI na CentOS, když našel místo městského webu konfigurační stránku Apache... Only 'freaks' waste their time with Linux in Oklahoma.

10.4.2006*Robert Krátký*

Vývojáři něm. verze Kubuntu (Ubuntu s KDE) mají pocit, že by si ze strany Canonical (sponzor Ubuntu) zasloužili více pozornosti. Chybí jim prostředky na provoz webu i další vývoj.

10.4.2006*Daniel Kvasnička ml.*

Laboratoře Kaspersky Lab informovaly o výskytu multiplatformního viru (Virus.Linux.Bi.a/Virus.Win32.Bi.a) schopného infikovat jak binární soubory ve Windows tak ELF v Linuxu. Virus sice nijak neškodí (je to pouhý „proof of concept“), avšak lidé z Kasperského laboratoří varují, že zneužití jeho kódu je jen otázkou času. Informuje SC Magazine.

10.4.2006*Daniel Kvasnička ml.*

Norsko je další zemí, které se nelíbí být otrokem monopolu softwarových gigantů. Tamní vláda by ráda více používala open-source a také plánuje sestavit skupinu expertů, která by se měla starat o hladký přístup k veřejným informacím. Informuje BetaNews.

10.4.2006*Daniel Kvasnička ml.*

Máme otevřené formáty pro text, grafiku, zvuk i video, ale každý blogovací systém je osamělým ostrovem. Joe 'Zonker' Brockmeier z NewsForge s tím není spokojen a rád by, aby vznikl otevřený formát i pro blogy.

10.4.2006*Jiří Němec*

Pokud rádi zkusíte nový software, možná vás zaujme další DC klient pro Linux: GtkDC. Prozatím neoplývá mnoha funkcemi a pokud se připojíte na hub s větším počtem lidí, tak se chová dost nestabilně. Pro svůj běh potřebuje knihovny DCLibC (také ke stáhnutí na stránkách).

11.4.2006*Robert Krátký*

dot.kde.org nabízí přehled aplikací pro KDE: All About the Apps. Každý program je krátce před-

staven. První díl přináší informace o amaroKu, Tellico, K3b a Kaffeine.

11.4.2006*boblin*

Nová verze LTSP 4.2 je venku. Nově obsahuje např. podporu pro lokální bloková zařízení (USB memory, cdrom, floppy), podporu SANE, rychlejší bootování, multi-head X atd. (New features)

11.4.2006*Pavel 'lingeek' Szalbot*

Rozhovor s Joshem Berkusem, jedním z hlavních vývojářů PostgreSQL, se objevil na ITtoolboxu. Dozvíte se v něm, kdy vyjde verze 8.2, čeho si na Postgresu nejvíce cení velké firmy jako Cisco či jak úspěšným se stal relativně nový port pro Windows.

11.4.2006*Robert Krátký*

Matej Urbančič se na linuxforums.org zamýšlí, jestli může být někdy software „dokončen“. Takový program by uměl vše, co pro svou práci potřebuje, a už by jej nebylo potřeba nijak vylepšovat nebo přidávat další funkce.

11.4.2006*Robert Krátký*

Význam a funkci linuxových systémových volání (system calls) vám přiblíží Puru Govind v článku na OSWeekly.com: How System Calls Work in Linux?

11.4.2006*Robert Krátký*

Vyšla verze 1.0 programu RPM Package Maker, grafické nadstavby pro vytváření RPM balíčků. Nově podporuje Installwatch. Více informací na kde-apps.org.

11.4.2006*David Majda*

Tristan Nitot, prezident Mozilla Europe, který byl na návštěvě v Praze, poskytl včera v rámci svého programu rozhovor internetovému Rádiu Akropolis. Mluvil v něm především o cílech Mozilla Foundation a Mozilla Europe, významu open-source technologií pro další rozvoj webu a také o své návštěvě. Rozhovor si můžete poslechnout dnes od 18 hodin v rámci pořadu Terabajt.

11.4.2006*Daniel Kvasnička ml.*

Na PCMECH.com vyšel celkem rozsáhlý článek pro ty, kteří by rádi místo Windows nasadili Ubuntu Linux. Řeč je jak o úplně základních aplikacích, tak třeba o PHP či MySQL: Windows to Ubuntu Transition Guide.

11.4.2006*Daniel Kvasnička ml.*

OSWeekly.com za pomoci dvojice přehledných diagramů stručně představuje formát binárních spustitelných souborů používaný v Linuxu: ELF.

11.4.2006*Daniel Kvasnička ml.*

Red Hat koupil (LinuxInsider) firmu JBoss a to za 350 milionů dolarů. The Register ale zjistil, že hlavní představitel JBossu Marc Fleury nebyl vždy s Red Hatem takový kamarád: Red Hat wraps Linux in sh*t.

11.4.2006*Daniel Kvasnička ml.*

Linux.com představuje pspi (Photoshop Plugin Interface) – software, který umožní GIMPu používat plug-in moduly pro Adobe Photoshop. Zkompilovat pspi není tak úplně jednoduché, avšak většina uživatelů to snad nebude potřebovat. Binárky pro Windows, SuSE, Ubuntu a Fedoru jsou totiž k dispozici ke stažení.

11.4.2006*Ondřej Prokop*

Článek o základním nastavení budoucí distribuce Ubuntu 6.06 nejen na notebooku najdete na sweb.cz/samba/ubuntu.

12.4.2006*Robert Krátký*

NewsForge vysvětluje, co znamenají zprávy o tom, že je na FTP mozilla.org ke stažení alfa verze prohlížeče Firefox 3.0. Ve skutečnosti jde o založení nové vývojové větve, která se od 1.6 liší jen tím číslem verze.

12.4.2006*Robert Krátký*

dot.kde.org informuje o vydání kancelářského balíku KOffice 1.5, který nabízí plnou podporu ODF a obsahuje i programy pro kreslení (Krita – rečenze), vektorovou grafiku (Karbon), vytváření grafů (Kivio) a databázových aplikací (Kexi). Oficiální oznámení.

12.4.2006*Daniel Kvasnička ml.*

Tvůrci distribuce SimplyMEPIS 10. dubna oznámili vydání první alfa verze řady 6.0 (download). Hlavní změnou je to, že distribuce teď vychází z Kubuntu Linuxu. Obsahuje kernel 2.6.15 a co se týká KDE (3.5.2), přejímá i nové System-Settings nahrazující KControl. Informuje DesktopLinux.com.

12.4.2006*Robert Krátký*

OSnews odkazuje na projekt KDE Look and Feel – Swing téma, díky kterému mohou Java programy vypadat jako KDE.

12.4.2006*Daniel Kvasnička ml.*

Brandon Egbert, student zabývající se PHP a pixel artem, na NewsForge popisuje, jak na jeho starém počítači funguje Damn Small Linux jako desktopový OS a poskytuje mu vše, co potřebuje.

12.4.2006*Daniel Kvasnička ml.*

Na LinuxExpo byla uvedena nová verze Linux Terminal Server Project 4.2. Obsahuje kernel řady 2.6, podporu scannerů a více monitorů a snižuje paměťové nároky. Informuje NewsForge.

13.4.2006*František Bublík*

V Teplicích se uskuteční již třetí setkání příznivců Linuxu. Datum Setkání je 21.4.06 – začátek v 18:00 v Hospůdce u černé kočky. Více se dozvíte zde.

13.4.2006*Robert Krátký*

Rychlý návod ke zprovoznění Xgl na SUSE 10.1 najdete v blogu SUSE Linux Rants: Quick Xgl Tutorial for SUSE Linux 10.1.

13.4.2006*Robert Krátký*

Německý tým Kubuntu(.de) v čele s Andrea-sem Muellerem vyhrocuje protesty proti nezájmu o tento KDE klon Ubuntu ze strany Canonical (firmy zaštiťující vývoj distribuce) [zprávička].

13.4.2006*Robert Krátký*

The Inquirer komentuje spekulace o tom, že by Google, eBay a Amazon mohly vybudovat ve Spojených státech svůj vlastní „internet“ – obejít telekomunikační společnosti a poskytnout lidem vysokorychlostní bezdrátové připojení ve svých vlastních pásmech.

13.4.2006*Daniel Kvasnička ml.*

Steve Mallet z OSDDir.com reaguje na reportáž ZDNetu o Novellu, jeho „vášnivě linuxové evangelizaci“ a stanicích s Windows: Novell still running Windows – So What?

13.4.2006*Daniel Kvasnička ml.*

Pokud vlastníte některý z telefonů Nokia 6630, 6680 nebo N70 a rádi byste ho v Linuxu používali jako GPRS/EDGE/3G modem, pak právě pro vás je návod na BitUbique.com. Psán je pro Ubuntu „Dapper Drake“, avšak neměl by být problém ho upravit pro jinou distribuci.

13.4.2006*Daniel Kvasnička ml.*

Tectonic přináší rozhovor s otcem jazyka Java, Jamesem Goslingem. Řeč je o jeho pozici v Sunu, vztahu firmy k open source či o C# a .NETu.

13.4.2006*Ladislav Michnovič*

Firma ATi vydala verzi 8.24.8 proprietárního linuxového ovládače pro svoje grafické karty. Podľa Releas Notes, podporuje X.org 6.9 a j 7.0.

13.4.2006*Jiří Poláček*

S denním zpožděním je na mirrorech dostupný první kandidát na ostrou verzi distribuce SUSE 10.1. RC1 stahujte například ze suse.inode.at.

13.4.2006*Filip Molčan*

Od dnešního dne je k dispozici počestěná sekce portálu věnovaného volně dostupné linuxové distribuci openSUSE – cz.opensuse.org.

14.4.2006*mog2*

Vyšla nová verze Mozilla Firefox 1.5.0.2. Jedná se o bezpečnostní aktualizaci a také zlepšení stability prohlížeče. 10 z 9 programátorů doporučuje aktualizaci! Česká verze bude brzy následovat.

14.4.2006*mog2*

Oprav chyb se dočkal také GIMP a povýšil na verzi 2.2.11.

14.4.2006*mog2*

The Open Phishing Database si klade za cíl vytvořit otevřenou databázi stránek, které jsou „phishingem“. Čili se tváří jako originální webové stránky, avšak mají jinou IP adresu a mají za cíl vylákat z vás citlivé údaje (čísla účtů, PIN, hesla atd.) Jako součást projektu vznikl také plugin pro Konqueror – Konqueror Anti-Phishing Toolbar.

14.4.2006*Daniel Kvasnička ml.*

Linux.com vysvětluje, jak mít na disku Windows a dvě linuxové distribuce – jednu bootující přes LILO (Xandros) a druhou přes GRUB (Fedora Core).

14.4.2006*Daniel Kvasnička ml.*

Matt Asay na InfoWorldu stručně shrnuje obsah diskuse na téma skutečných nákladů na implementaci open-source řešení, která se udála na letošním LinuxWorldu: The cost of open source: Implementations

14.4.2006*CIJOML*

Prohlížeč Seamonkey, nástupce Mozilla Suite, se dočkal nové opravné revize 1.0.1. Stahujte z mozilla.org.

14.4.2006*Daniel Kvasnička ml.*

Vyšlo GNOME 2.14.1 plné bugfixů, doplněné dokumentace a překladů. Odkazy na seznamy změn, zdrojové kódy a další užitečné informace najdete na LinuxToday.

14.4.2006*Daniel Kvasnička ml.*

Mozilla Foundation vydala v rámci projektu Firefox Flicks první várku amatérských videoklipů natočených fanoušky Firefoxu. Informuje CNET News.com

14.4.2006*Věroš Kaplan*

Google vyhlašuje další ročník soutěže Summer Of Code. Pokud jste student a přijdete se zajímavým OpenSource projektem, máte možnost získat zajímavé stipendium. Můžete si také vybrat některý z navržených projektů. Cílem soutěže je získat nové vývojáře pro OSS.

15.4.2006*David Watzke*

Vyšla nová verze Wine (0.9.12), která přináší pár vylepšení podpory fontů, IDL kompilátoru, rychlejší vykreslování karet v Solitaire :-)) a jako vždy mnoho dalších oprav.

17.4.2006*Tomáš*

Pokud je vaším snem stát se tím nejlepším hackerem z nejlepších, rozhodně vám nemůže ujít tento článek na owebu.cz – Jak odhalit útok.

18.4.2006*Daniel Kvasnička ml.*

HowtoForge přináší podrobný návod, jak vystavět na 64bit verzi Fedory Core 5 kompletní server (web, mail, DNS, FTP, MySQL, firewall,...).

18.4.2006*Daniel Kvasnička ml.*

Stan Beer se na iTWire zamýšlí nad zvyšující se popularitou Googlu a Linuxu, která by z Microsoftu (přirovnávaného k „neporazitelnému“ IBM z 80. let) mohla udělat firmu minulosti: Will Google and Linux make Microsoft yesterday's company.

18.4.2006*Daniel Kvasnička ml.*

Databázový gigant Oracle by rád rozšířil sortiment produktů o svou vlastní linuxovou distribuci, přičemž v rámci své studie pokukoval i po Novellu. Rád by totiž, podobně jako Microsoft či

Red Hat, nabízel se svými aplikacemi i operační systém. Informují Financial Times.

18.4.2006

Daniel Kvasnička ml.

Sun Microsystems pokračuje v otevírání svých produktů. Tentokrát firma oznámila, že na webu NetBeans.org nabídne značnou část zdrojových kódů produktu Java Studio Enterprise. Informuje internetnews.com.

18.4.2006

Radek Vokál

Nové číslo Red Hat magazínu popisuje distribuci Fedora Core 5, její novinky a směr, kterým se bude dále Fedora ubírat. Další článek popisuje vývojové prostředí Eclipse a pluginy, které je možné do Eclipse doinstalovat. V magazínu najdete i oficiální výjádření o získání společnosti JBoss.

19.4.2006

David Jaša

Nezadržitelně se blíží třetí pátek v měsíci a s ním sedmé velikonoční setkání LvB (přátel Linuxu a svobodného softwaru v Brně). Přijďte také pokecat do Onyxu :-).

19.4.2006

Robert Krátký

Byl oznámen vítěz letošních voleb do funkce Debian Project Leader. Novým vedoucím projektu se stal Anthony Towns, přičemž Branden Robinson, dosavadní project leader, znovu nekandidoval.

19.4.2006

Robert Krátký

NewsForge testovalo účinnost tzv. obojživelného viru [zprávička], který má být schopen nakazit jak linuxové, tak windows programy. Jak virus „obstál“, si můžete přečíst v článku Hands-on testing of the new Linux virus.

19.4.2006

Robert Krátký

Článek na howudune.com přibližuje, jak nainstalovat Knoppix na USB klíčenku a pak z ní také úspěšně nabootovat: How to boot Knoppix 4.0 from a USB Flash Drive.

19.4.2006

Daniel Kvasnička ml.

Nad Mac OS X se stahují mračna a Applu zbývá jediné: udělat z OS X open-source projekt, píše John C. Dvorak na PCMag.com.

19.4.2006

Daniel Kvasnička ml.

Vyšla první beta verze desktopového prostředí Xfce 4.4 (4.3.90.1). Obsahuje dlouho očekávaný souborový manažer Thunar a další velká vylepšení. Download.

19.4.2006

Daniel Kvasnička ml.

Tým vývojářů Kanotixu vydal speciální velikonoční verzi svého live CD: Easter Special Edition live CD RC2. Obsahuje opatchovaný kernel 2.6.16.5, Xorg 7.0.12, KDE 3.5.2, KOffice 1.5 a další. ISO (633 MB), BitTorrent. Informuje DesktopLinux.com.

19.4.2006

Michal Vyskočil

Začíná vznikat LiveCD pro vývojáře v Ruby/Ror. Je založené na Ubuntu a jeho velmi kreativní název je Rubuntu. Jedna ze základních otázek ale zní „dát tam KDE, či Gnome“?

19.4.2006

Michal Vyskočil

Linux jako screensaver ve Windows? Ač se to může zdát nemožné, tak vývojáři IBM vyzkoušeli a popsali způsob, jak to provést. Řešením je qemu, takže systém je sice pomalejší, ale plně funkční. Stahovat můžete přes bittorrent.

19.4.2006

Tomáš Hála

Je tomu již pár týdnů, co vyšla nová verze MidpSSH – SSH a telnet klienta pro Váš mobilní telefon podporující Javovské (MIDP 1.0/2.0) aplikace. Podporuje mj. SSH1 i SSH2, makra pro zadávání častých příkazů, zadávání speciálních kláves, VT320 terminál emulaci a poskytuje informaci o přenesených datech po skončení session.

20.4.2006

Robert Krátký

Lee Schlesinger je přesvědčen, že Bruce Perens se stal pro linuxovou komunitu spíše přítěží. Zatímco dříve Linux podporoval, teď je jeho chování spíše na škodu. Více v článku na NewsForge: Bruce Perens has become a distraction.

20.4.2006

Robert Krátký

Otevřený dopis novému Debian Project Leader [zprávička] mluví o všem a o ničem. Co Debianu chybí, čeho má dost, a co se může naučit jinde?

20.4.2006

Robert Krátký

Linus Torvalds zjistil díky obojživelnému „viru“ [zprávička], že některé verze linuxového jádra obsahovaly malou chyбку, která „viru“ neumožňovala dělat to, co zamýšlel. Chyba byla opravena, aby mohl „virus“ správně fungovat. Torvalds patches Linux kernel, fixes broken virus.

20.4.2006

Robert Krátký

AMD64 byla oficiálně přidána na seznam architektur podporovaných distribucí Debian Etch (příští stabilní Debian). Tento krok následuje ně-

kolik týdnů po uploadu prvního balíčku do unstable.

20.4.2006

Lukáš Zapletal

MySQL bude mít nový storage engine ze solidDB – zajímavá úvaha nad budoucností databáze MySQL v podání Miroslava Suchého na Lupě.

20.4.2006

Daniel Kvasnička ml.

Portland Project sponzorovaný OSDL se nezabývá jen sjednocováním GUI. Před pár dny uspořádal také Desktop Linux Printing Summit, na němž se započala práce na konsolidaci tiskových služeb v Linuxu. Krátké shrnutí přináší DesktopLinux.com.

20.4.2006

Daniel Kvasnička ml.

Eric Doyle se na ComputerWeekly.com zamýšlí nad schopnostmi Linuxu konkurovat Windows Vista. Podle článku masivní rozšíření Linuxu na desktopu rozhodně nebude tak růžové, jak ho barví marketingové oddělení Novellu a dalších.

20.4.2006

Daniel Kvasnička ml.

The Salt Lake Tribune přináší článek analyzující situaci kolem Oracle, který by rád měl svojí vlastní distribuci [zprávička] a mlsně pokukuje po Novellu. Nechybí ani grafy znázorňující pohyby cen akcií obou subjektů.

20.4.2006

Daniel Kvasnička ml.

Vyšla první beta verze prohlížeče Opera 9. Mezi novinkami jsou widgets, BitTorrent klient či blokování obsahu. Download.

20.4.2006

Daniel Kvasnička ml.

Reklama na Firefox se dala do pohybu. Myslíte, že autor dekorace věděl, odkud motiv pochází?

20.4.2006

Petr Tomeš

Tým Ubuntu s potěšením oznamuje betavydání Ubuntu 6.06 LTS – s kódovým označením „Dapper Drake“. Betavydání přichází s novým Desktop CD, které může být použito jak pro vyzkoušení Ubuntu spuštěného z CD, tak pro instalaci systému. Ubuntu 6.06 LTS (long-term support – dlouhodobě podporované) bude prvním vydáním Ubuntu, které bude podporované 3 roky na stolních počítačích a 5 let na serverech. Seznam novinek v Ubuntu 6.06 LTS Beta oproti Ubuntu 5.10 naleznete na www.ubuntu.com.

21.4.2006

Daniel Kvasnička ml.

Pokud máte pocit, že váš linuxový stroj bootuje příliš pomalu, třeba vám přijde vhod návod na

Linux.com, který vysvětluje, jak boot zrychlit paralelním startováním služeb.

21.4.2006

Daniel Kvasnička ml.

Na Linuxu je nejlepší, že naplňuje vaše potřeby a dává vám svobodu pohybu aniž byste ho museli vlastnit. IBM to moc dobře chápe a vzkazuje Oraclu: You can't buy open source.

21.4.2006

Daniel Kvasnička ml.

Python Software Foundation hledá (a bude sponzorovat) studenty a mentory, kteří by se rádi účastnili Google Summer of Code 2006 a pracovali na pythonských projektech. Python Summer of Code Wiki. Google Summer of Code.

21.4.2006

Franta

Nedávno vyšla nová verze programu keyTouch – 2.1.3. Je to program pro nastavování funkcí multimediálních klávesnic s velmi user-friendly GUI rozhraním. Dostupné jsou balíčky pro SuSE, Ubuntu, Gentoo, Arch a Mandriva Linux.

22.4.2006

Pavel 'lingeek' Szalbot

Existuje spousta benchmarků linuxových filesystemů, ale většinou se jedná o kousky založené na vyumělkovaných testech, nebo starých kernelech. Na debian-administration.org si tak můžete přečíst výsledky 11 skutečných testů filesystemů.

22.4.2006

Pavel 'lingeek' Szalbot

Čínská společnost YellowSheepRiver nabízí linuxový notebook postavený na čtyřcestném superskalárním MIPS procesoru Godson-2 s rozhraním USB 2.0, IrDA, VGA, S-video a 40GB externím USB diskem. Vybavením, cenou ale i dostupností by tak mohl konkurovat \$100 notebook MIT. Více na LinuxDevices.com.

23.4.2006

Ondřej Prokop

Jak dopadla instalace OpenSUSE 10.1 RC1 na notebook ACER? Moc dobře ne. Oproti minule testované distribuce UBUNTU Dapper Drake Flight 6 mám k OpenSUSE velké výhrady.

23.4.2006

Radim Krivošík

V konferenci mandrake@mandrake.cz dnes proběhl odkaz na novou, česky psanou uživatelskou příručku pro Mozilla Firefox 1.5. Stáhnout lze na adrese www.chovancik.cz.

24.4.2006

Daniel Kvasnička ml.

Ubuntu Dapper Drake ještě není venku, ale Mark Shuttleworth už poodhaluje detaily příští verze naplánované na říjen tohoto roku: Edgy Eft. Pů-

jde tak trošku o experimentální verzi obsahující podporu pro smíšené architektury (32/64bit) či např. SMART package manager. Více na ZDNetu.

24.4.2006*Daniel Kvasnička ml.*

Další verzi plánují i tvůrci Fedora Core. Jeden z hlavních vývojářů poslal do mailing listu návrh časového plánu vývoje FC6, podle kterého to vypadá, že např. GNOME 2.16 v nové verzi nebudeme. Více na Heise Online.

24.4.2006*Daniel Kvasnička ml.*

Internetnews.com přináší rozhovor s Billem Hillem, Platform Strategy Managerem firmy Microsoft, který je zodpovědný za to, jak se firma staví k open source. Řeč je o GPL, MS Office v Linuxu, reverse engineeringu, intelektuálním vlastnictví a dalších tématech.

25.4.2006*Robert Krátký*

Článek na linuxforums.org představuje uživatelským regulární výrazy a mluví o tom, jak je využít při práci s běžnými programy: Where to Use Regular Expressions.

25.4.2006*Robert Krátký*

Firma Linspire se rozhodla svoji nabídku linuxových distribucí rozšířit o zdarma dostupnou Freespire – podle osvědčeného vzoru Red Hat + Fedora nebo SUSE + openSUSE.

25.4.2006*Daniel Kvasnička ml.*

Scott McNealy, který v r. 1982 spoluzakládal a od r. 1984 vedl firmu Sun Microsystems, odchází z postu CEO. Pozici po něm převezme prezident společnosti Jonathan Schwartz, přičemž McNealy zůstane ve správní radě společnosti. Informuje LinuxInsider.

25.4.2006*Daniel Kvasnička ml.*

Rob Enderle na TechNewsWorld předpovídá Linuxu na desktopu těžké časy. Kritizuje vysoký počet distribucí, fanatismus a další neduhy: Why Linux May Never Be a True Desktop OS.

25.4.2006*Daniel Kvasnička ml.*

Jistý blogger podrobně popisuje, proč si myslí, že Ruby on Rails nebude nikdy mainstream. Uváděné důvody jsou např. lenost PHP a VisualBasic programátorů učit se Ruby, „přílišná vyspělost“ Ruby on Rails či absence opravdu kvalitního IDE.

26.4.2006*Lukáš Jelínek*

Britské listy přinesly informaci, že dánský obchodní soud potvrdil legalitu odkazů vedoucích

přímo do hloubi webu (namísto titulní stránky). Toto rozhodnutím může mít velké dopady i mimo Dánsko a je výraznou podporou pro vyhledávací služby.

26.4.2006*Robert Krátký*

Co má společného Charles Dickens a Thomas Hardy s open source? Ve svém blogu se nad tím zamýšlí Dave Rosenberg: Classic literature and open source.

26.4.2006*Robert Krátký*

Carla Schroder začátečníkům vysvětluje, jak v Linuxu zabíjet. Článek Killing With Linux: A Primer představuje procesy, služby, demony a ovládání pomocí signálů.

26.4.2006*Robert Krátký*

Druhá část minisérie KDE 3: All About the Apps na dot.kde.org seznamuje (mimo jiné) s novým konfiguračním nástrojem Guidance, programem pro free VoIP Twinkle a šikovnými utilitkami Kerry a yaBi.

26.4.2006*Katarína Machálková*

Příznivci OpenSolarisu a všichni zájemci o nové technologie jsou zváni na osmé setkání Czech OpenSolaris Users Group spojené s přednáškou, které se bude konat ve středu 3. května od 18:00 v posluchárně K9 v prostorách FEL ČVUT na Karlově náměstí. Na programu jsou i tentokrát dvě prezentace: Tomáš Heran představí Solaris Service Management Facility a Vladimír Kotal seznámí posluchače se zdroji binárních balíčků pro OpenSolaris mimo hlavní distribuci a jejich výhodami a nevýhodami.

26.4.2006*Robert Krátký*

V reakci na založení distribuce Freespire [zprávička] kritizovala firmu (a distribuci) Linspire Pamela Jones z Groklaw.net. Odpověď napsala Carla Schroder: Linspire: The Revolutionary Linux.

26.4.2006*Pavel Beníšek*

HP se rozhodla pomáhat svým zákazníkům s nasazením Open Source programů spolu s proprietárními řešeními prostřednictvím nového programu Open Source Integrated Portfolio (OSIP). Informuje echannelline.com.

26.4.2006*Daniel Kvasnička ml.*

GPL není jediná licence, která prochází tento rok změnami. Perl Foundation vydala návrhy Artistic

License 2.0 (na které se, podobně jako na Perlu 6, pracuje už od r. 2000) a Contributor License Agreement a očekává komentáře od veřejnosti. Informuje NewsForge.

26.4.2006*Daniel Kvasnička ml.*

V dalším článku ze série „My desktop OS“ na NewsForge si můžete přečíst zkušenosti člověka, který před 6 lety vyměnil Linux za FreeBSD a dnes používá verzi 6.0-STABLE na desktopu.

26.4.2006*Martin Tesař*

Na adrese fora.gentoo.cz byl zahájen pokus resuscitovat diskusi českých a slovenských uživatelů Gentoo. Zájemci o spolupráci i diskutující jsou srdečně zváni.

26.4.2006*Daniel Kvasnička ml.*

Linux.com přináší stručný úvod do práce s platformou .NET v Linuxu, tedy s frameworkem Mono a jazykem C#. Nechybí ani návod na zprovoznění XSP serveru umožňujícího používat Mono na vývoj webových aplikací.

26.4.2006*Robert Krátký*

Podle blackhole.sk bylo velmi snadné proniknout na servery (FreeBSD) Národního bezpečnostního úřadu SR. Na vině je laxní počínání administrátorů. Narodny Bezpecnostny Urad pwn3d.

26.4.2006*Filip Korbel*

Hezký článek od „lovce“ pro Red Hat, či spíše desatero pro absolventy a váš průvodce personálním výběrovým řízením, vyšel na pracovním portálu PráceAbc.

27.4.2006*Robert Krátký*

Linux.com radí, jak zvýšit bezpečnost při bootu počítače nastavením hesla v GRUBu: Enhance boot-time security with GRUB passwords.

27.4.2006*Robert Krátký*

Joe Barr píše o svých dojmech ze závěru konference Desktop Linux Summit, kdy promluvil Rob Enderle o tom, proč OEM distributoři nepoužívají jako operační systém i GNU/Linux.

27.4.2006*Robert Krátký*

Americký kongres připravuje nový zákon o copyrightu v digitální oblasti. Podle analýzy CNet.com bude však ještě „přísnější“ než kritizovaný Digital Millennium Copyright Act (DMCA).

27.4.2006*Robert Krátký*

Podle The Inquirer se firma Dell chystá představit notebooky Latitude DX20 s předinstalovaným Mandriva Linuxem: Dell to intro Linux based notebook.

27.4.2006*Robert Krátký*

GNOME 2.15.1 je první vývojovou verzí na cestě ke GNOME 2.16 (září 2006). Opravuje množství chyb, přidává nové funkce a obsahuje hodně změn. Odkazy na oznámení a místa, odkud lze stahovat, najdete na gnomedesktop.org.

27.4.2006*elviin*

Návrh změn (Working draft) pro nový standard jazyka C++ si můžete nyní prohlédnout v přehledném seznamu na open-std.org. Obsahuje komponenty z Technical Reportu (std::tr1::), které odsouhlasila komise při setkání v Berlíně tento měsíc.

27.4.2006*Daniel Kvasnička ml.*

Pokud byste rádi zkusili psát ovladače pro Linux, pak pro vás má Free Software Magazine zajímavý článek: Writing device drivers in Linux: A brief tutorial.

27.4.2006*Daniel Kvasnička ml.*

S neotřelými radami superhrdiny známého pod jménem Henry the Adequate, kterému i samotný Linus volá, aby se s ním poradil co a jak, bude financování vašich open-source projektů hračkou: Henry s Guide to Financing Your Free or Open Source Project and Really Long Article Titles.

27.4.2006*Daniel Kvasnička ml.*

Vytvořit a číst PDF v Linuxu není vůbec problém, avšak manipulace s již hotovými dokumenty je složitější. Linux.com proto představuje software, který je svými schopnostmi asi nejbližší Adobe Acrobatu: PDF Toolkit.

27.4.2006*Ladislav Michnovič*

V dnech 13. a 14. 5. 2006 sa uskutoční v Košiciach v poradí 14. víkend s Linuxom – Lugcon. Tentokrát sa akcia koná v priestoroch Univerzity Pavla Jozefa Šafárika.

27.4.2006*Frantisek Hucek*

Týdeník EURO zveřejnil velmi tendenční článek o rizicích nasazení Open Source softwaru. Obzvláště „povedená“ tvrzení jsou k nahlédnutí na linuxsoft, stejně jako dopis šéfredaktoru EURA, jeho první reakce a vyjádření autora článku.

27.4.2006*Michal Vyskočil*

Podle jistého Američana je Linux evropská hrozba pro jejich počítače. Údajně porušuje patentová práva americké SCO, přináší s sebou marxistickou ideologii, kopíruje americké technologie a dokonce dojde i na Al Quaidu a slavné čínské hackery.

28.4.2006*Robert Krátký*

Reg Developer píše o Apache Portable Runtime a výhodách, které přináší programovacímu jazyku C. Mezi ty hlavní patří správa zdrojů a přenositelnost kódu na jiné platformy.

28.4.2006*Robert Krátký*

dot.kde.org představuje Phonon – nové API pro multimédia v KDE 4. Phonon bude moci používat libovolný backend (Xine, GStreamer, atd.). Goodbye aRts...

28.4.2006*Robert Krátký*

arstechnica.com informuje o plánech Intelu, které oznámil CEO Paul Otellini – kvůli ostřejší konkurenci a značnému poklesu zisků v posledním čtvrtletí se společnost pustí do velké „restrukturalizace“. Ve firmě prý nezůstane „kámen na kameni“.

28.4.2006*Daniel Kvasnička ml.*

Marc Wagner ze ZDNetu došel k závěru, že vybrat si OS je jako vybrat si náboženství. Pro mnohé je prý výběr OS otázka víry. Ve svém blogpostu se zamýšlí nad tím, co činí z některých lidí fanatiky a proč jsou někteří lidé schopni verbálně napadat druhé za volbu odlišného OS.

28.4.2006*Daniel Kvasnička ml.*

Windows Vista je jeden velký malér. Vědí to odborníci na Windows a vědí to i zaměstnanci MS. I Steven J. Vaughan-Nichols z Linux-Watch si toho je vědom a nebojí se mluvit nahlas o řešení, které by Vistu mohlo opět postavit na nohy: open source.

28.4.2006*Daniel Kvasnička ml.*

Howtoforge.com stručně popisuje, jak stahovat webové stránky do Palmu pro off-line prohlížení. Stránky jsou uloženy ve formátu Plucker, který dostaneme s pomocí programu Sunrise.

28.4.2006*Petr Tomeš*

Týmy Ubuntu, Kubuntu a Edubuntu s potěšením uvádějí druhé betavydání Ubuntu/Kubuntu/Edebuntu 6.06 LTS – s kódovým označením „Dapper Drake“. Toto vydání opravuje některé vážné

chyby instalátoru na Desktop CD v prvním betavydání [zprávička]. Přestože je textové instalační CD také součástí tohoto vydání, nebylo od první bety změněno. V přípravě je také aktualizované vydání Xubuntu a bude brzy oznámeno.

29.4.2006*Michal Eibl*

Tutoriál Java3D najdete na adrese me.euweb.cz/skola/apg. Je to semestrální práce z předmětu Aplikace počítačové grafiky.

30.4.2006*Mirek*

29.4. byla uvolněna finální verze PC-BSD 1.0. Jedná se o systém založený na FreeBSD 6.0, zaměřený na obyčejného uživatele. Programy se instalují pouhým kliknutím na ikonu instalovaného programu(.PBI), podobně jako ve Windows. Uživatelé odpadají problémy s repozitáři, kompilacemi či peklem závislostí. Možnost klasického postupu instalace zůstává.

30.4.2006*René Baran*

V dubnu 2006 vznikla česká divize distribuce Aurox. Oficiální český WEB (<http://www.aurox-cz.tk>) by měl sdružovat všechny české uživatele této distribuce. V současnosti web není ještě zcela zaplněn. Přivítáme proto každý nápad nebo informaci. Chceme zde uveřejňovat manuály, zajímavé odkazy, SW od uživatelů a developerů, ...