

Priručnik za administratore Linux sistema

Verzija 0.7

Lars Wirzenius <liw@iki.fi>

Joanna Oja <viu@iki.fi>

Stephen Stafford <stephen@clothcat.deamon.co.uk>

Prijevod na bosanski jezik:

Verzija CURRENT

Adis Nezirović <adis.n@bih.net.ba>

27. ožujka 2003.

Naslov originala:

The Linux System Administrators' Guide¹: Version 0.7

by Lars Wirzenius

by Joanna Oja

by Stephen Stafford

Priručnik za administratore Linux sistema

Prijevod na bosanski jezik: Verzija² 0.6-27.03.2003

by Adis Nezirović <adis.n@bih.net.ba>

Home page: <http://howto.linux.org.ba/sag/>



*An introduction to system administration of a **Linux** system for novices.*

*(Uvod u administraciju **Linux** sistema za početnike.)*

Copyright 1993-1998 Lars Wirzenius

Copyright 1998-2001 Joanna Oja

Copyright 2002 Stephen Stafford

Copyright 2003 Adis Nezirović

Trademarks are owned by their owners.

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.1; with no invariant Sections, with no Front-Cover Texts, and with no Back-Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation License".

¹Note to english readers: The most recent version of this book in English is available from:
<http://www.tldp.org>

²Ovo je još uvijek nedovršen rad. Kod broja verzije prijevoda (0.n) n označava tekući stadij prijevoda, tj. broj prevedenih poglavlja. Kada budu prevedena sva poglavlja, broj verzije prijevoda postaje 1.0.

Sadržaj

1	Uvod	6
2	O knjizi	8
2.1	Zahvale	8
2.1.1	Joanna	8
2.1.2	Stephen	8
2.1.3	Adis	9
2.2	Tipografske konvencije	9
3	Kratak pregled Linux sistema	11
3.1	Različiti dijelovi operativnog sistema	11
3.2	Važni dijelovi kernela	12
3.3	Glavni servisi UNIX sistema	12
3.3.1	INIT	12
3.3.2	Prijavljivanje sa terminala	13
3.3.3	Syslog	13
3.3.4	Periodičko izvršavanje komandi: cron i at	13
3.3.5	Grafički korisnički interfejs	14
3.3.6	Umrežavanje	14
3.3.7	Prijavljivanje preko mreže	14
3.3.8	Mrežni datotečni sistemi	15
3.3.9	Elektronska Pošta (e-mail)	15
3.3.10	Ispis (printanje)	15
3.3.11	Izgled datotečnog sistema	15
4	Osvrt na stablo direktorija	16
4.1	Izgled datotečnog sistema	16
4.2	Korijenski datotečni sistem	17
4.3	<i>/etc</i> direktorij	19
4.4	<i>/dev</i> direktorij	21
4.5	<i>/usr</i> datotečni sistem	21
4.6	<i>/var</i> datotečni sistem	22
4.7	<i>/proc</i> datotečni sistem	23
5	Datoteke uređaja	25
5.1	<i>MAKEDEV</i> skripta	25
5.2	<i>mknod</i> naredba	25
5.3	Lista (datoteka) uređaja	26

6	Korištenje diskova i ostalih medija za smještaj podataka	30
6.1	Dvije vrste uređaja	31
6.2	Hard diskovi	31
6.3	<i>Floppy</i> diskovi	33
6.4	CD-ROM	33
6.5	Trake	34
6.6	Formatiranje	34
6.7	Particije	36
6.7.1	MBR, <i>boot</i> sektori i tabela particija	36
6.7.2	Proširene (<i>extended</i>) i logičke particije	37
6.7.3	Tipovi particija	37
6.7.4	Dijeljenje hard diska na particije	38
6.7.5	Datoteke uređaja i particije	39
6.8	Datotečni sistemi	39
6.8.1	Šta su to datotečni sistemi?	39
6.8.2	Mnoštvo datotečnih sistema	39
6.8.3	Koje datotečne sisteme treba koristiti?	39
6.8.4	Kreiranje datotečnog sistema	39
6.8.5	Montiranje i demontiranje	39
6.8.6	Provjera integriteta datotečnog sistema sa <i>fsck</i>	40
6.8.7	Provjera ispravnosti diska sa <i>badblocks</i>	40
6.8.8	Fragmentacija	40
6.8.9	Drugi alati za sve datotečne sisteme	40
6.8.10	Drugi alati za <i>ext2</i> datotečni sistem	40
6.9	Diskovi bez datotečnog sistema	40
6.10	Dodjela prostora na disku	40
6.10.1	Šeme za kreiranje particija	40
6.10.2	Zahtjevi za prostorom na disku	40
6.10.3	Primjeri dodjele prostora na hard diskovima	40
6.10.4	Dodavanje prostora na disku za Linux	40
6.10.5	Savjeti za štednju prostora na disku	40
7	Upravljanje memorijom	41
7.1	Šta je virtualna memorija?	41
7.2	Kreiranje <i>swap</i> prostora	41
7.3	Korištenje <i>swap</i> prostora	41
7.4	Dijeljenje <i>swap</i> prostora sa drugim operativnim sistemima	41
7.5	Dodjela <i>swap</i> prostora	41
7.6	Međuspremnik (<i>buffer cache</i>)	41
8	Pokretanje i gašenje	42
8.1	Pregled	42
8.2	<i>Boot</i> proces izbliza	42
8.3	Više o gašenju	42
8.4	Restart	42
8.5	Jednokorisnički mod	42
8.6	<i>Boot</i> diskete za podizanje sistema u poslije katastrofa	42

9	<i>init</i>	43
9.1	<i>init</i> je prvi	43
9.2	Podešavanje <i>init</i> -a za pokretanje <i>getty</i> -a: datoteka <i>/etc/inittab</i>	43
9.3	Izvršni nivoi (<i>run levels</i>)	43
9.4	Specijalne postavke u <i>/etc/inittab</i>	43
9.5	Pokretanje sistema u jednokorisničkom modu	43
10	Prijavljivanje i odjavljivanje	44
10.1	Prijave preko terminala	44
10.2	Prijave preko mreže	44
10.3	Šta to radi <i>login</i>	44
10.4	<i>X</i> i <i>xdm</i>	44
10.5	Kontrola pristupa	44
10.6	Pokretanje <i>shell</i> -a	44
11	Upravljanje korisničkim računima (<i>account</i>-ima)	45
11.1	Šta je to <i>account</i>	45
11.2	Dodavanje novog korisnika	45
11.2.1	<i>/etc/passwd</i> i druge datoteke sa informacijama	45
11.2.2	Odabir numeričkih identifikacija za korisnike i grupe korisnika	45
11.2.3	Inicijalne postavke: <i>/etc/skel/</i>	45
11.2.4	”Ručno” kreiranje korisnika	45
11.3	Izmjene atributa korisnika	45
11.4	Uklanjanje korisnika	45
11.5	Privremeno onemogućavanje korisnika	46
12	<i>Backup</i>	47
12.1	Značaj <i>backupa</i>	47
12.2	Odabir medija za <i>backup</i>	47
12.3	Odabir alata	47
12.4	Jednostavni <i>backup</i>	47
12.4.1	Korištenje komande <i>tar</i> za <i>backup</i>	47
12.4.2	Restauracija datoteka sa <i>tar</i>	47
12.5	<i>Backup</i> i u više nivoa	47
12.6	Šta zaslužuje <i>backup</i>	47
12.7	Kompresovani <i>backup</i> -i	48
13	Ukorak s vremenom	49
13.1	Vremenske zone	49
13.2	Hardverski i softverski sat	49
13.3	Prikazivanje i podešavanje vremena	49
13.4	Kada je sat netačan	49
14	Traženje pomoći	50
14.1	<i>News</i> grupe i <i>mailing</i> liste	50
14.1.1	Nalaženje pravog foruma	50
14.1.2	Prije oglašavanja	50
14.1.3	Kako napisati oglas (<i>post</i>)	50
14.1.4	Format oglasa	50

14.1.5	??? Follow Up ???	50
14.1.6	Dodatne informacije	50
14.2	IRC	50
14.2.1	Boje	51
14.2.2	Budite ljubazni	51
14.2.3	Tipkajte ispravno, na engleskom jeziku	51
14.2.4	Skeniranje portova	51
14.2.5	Ograničite se na kanal	51
14.2.6	Ne skrećite s teme	51
14.2.7	CTCPs	51
14.2.8	Hacking, cracking, phreaking, warezing	51
14.2.9	Suma sumarum	51
14.2.10	Daljnje čitanje	51
15	Dodatak A: GNU Free Documentation License	52
15.1	PREAMBLE	52
15.2	APPLICABILITY AND DEFINITIONS	52
15.3	VERBATIM COPYING	53
15.4	COPYING IN QUANTITY	53
15.5	MODIFICATIONS	54
15.6	COMBINING DOCUMENTS	56
15.7	COLLECTIONS OF DOCUMENTS	56
15.8	AGGREGATIONS WITH INDEPENDENT WORKS	56
15.9	TRANSLATION	57
15.10	TERMINATION	57
15.11	FUTURE REVISIONS OF THIS LICENSE	57
15.12	How to use this license for your documents	57
16	GLOSSARY	59

Poglavlje 1

Uvod

"In the beginning, the file was without form, and void; and emptiness was upon the face of the bits. And the Fingers of the Author moved upon the face of the keyboard. And the Author said, Let there be words, and there were words."

Ovaj priručnik, *Linux System Administrator's Guide* opisuje aspekte korištenja **Linux**-a sa stajališta sistemskog administratora¹. Namijenjen je ljudima koji ne znaju skoro ništa o administriranju (kao npr. "Šta je administriranje?"), ali koji su već usvojili barem osnove korištenja **Linux**-a. Također, ništa se ne govori o tome kako instalirati **Linux**; to je opisano u dokumentu "*Installation and Getting Started*". Za više informacija o **Linux** priručnicima pogledajte dolje.

Administracija predstavlja sve one neophodne postupke koje neko mora raditi da bi računarski sistem bio u upotrebljivom stanju. To uključuje stvari kao što su backup datoteka, instaliranje novih programa, kreiranje i brisanje korisničkih računa (*accounta*), provjeravanje integriteta datotečnog sistema... Kada bi kompjuter bio kuća, administracija sistema zvala bi se održavanje i uključivala bi čišćenje, popravljavanje pokvarenih prozora i slične stvari.

Struktura ovog priručnika je takva da bi poglavlja trebala biti upotrebljiva nezavisno jedna od drugih, tako da ako npr. trebate informaciju o backup-u možete pročitati samo to poglavlje. Ovo također olakšava upotrebu ove knjige kao podsjetnika tako da je moguće pročitati samo mali dio, po potrebi, umjesto da morate pročitati sve. Međutim, ovaj priručnik je prvenstveno tutorial i može se čitati dio po dio ili kao cjelina.

Nije dovoljno da koristite ovaj priručnik sam za sebe. Mnogo druge **Linux** dokumentacije također je važno za administratore. Ipak, administrator je samo korisnik sa specijalnim pravima i dužnostima. Veoma važan resurs su **man** stranice, koje bi uvijek trebalo konsultirati ako naredba nije poznata. Ako ne znate koju naredbu trebate, tada možete koristiti naredbu **apropos** Konsultirajte odgovarajuću **man** stranicu.

Premda je tema ovoga priručnika **Linux**, može se reći da bi općenito trebao biti od pomoći i sa drugim **UNIX** baziranim operativnim sistemima. Nažalost, budući da u opštem slučaju postoje velike razlike između verzija **UNIX**-a, a posebno razlika u administriranju, teško se nadati da će se pokriti sve varijante. Čak je i pokrivanje svih mogućnosti **Linux**-a teško, a sve zbog prirode njegovog razvoja.

Ne postoji samo jedna zvanična **Linux** distribucija, tako da različiti ljudi imaju različite postavke (distribucije) ili čak distribuciju koju su sami kreirali². Ova knjiga nije posvećena samo jednoj

¹op.p. U nastavku teksta samo administratora, ili ponekad *root-a*, u ovisnosti od konteksta

²op.p. U tom kontekstu zanimljiv je projekat *Linux From Scratch* koji je u posljednje vrijeme dosta popularan način kreiranja vaše "sopstvene distribucije" - <http://www.lfs.org>

distribuciji. Distribucije se mogu razlikovati, i to poprilično. Kada god je to bilo moguće, pokušao sam ukazati na razlike i prezentirati alternative.

U nastojanju da stvarno opišem kako stvari rade, a ne samo da nabrojim "pet brzih koraka" za svaki zadatak, prezentirano je dosta informacija koje nisu potrebne svakome, ali ti dijelovi su tako i označeni i mogu se izbjeći ako već koristite unaprijed konfigurisan **Linux** sistem. Naravno, čitajući sve, povećati ćete vaše razumijevanje sistema, što bi zauzvrat trebalo učiniti njegovo korištenje i administriranje efikasnijim.

Kao i svi drugi radovi koji se tiču **Linux**-a, ovaj rad je urađen na dobrovoljnoj bazi. Zato što sam mislio da bi moglo biti zabavno i zato što sam osjećao da bi trebao biti urađen. Međutim, kao i kod svakog dobrovoljnog rada postoji ograničenje u tome koliko truda sam uspio uložiti i koliko znanja i iskustva posjedujem. To znači da ovaj priručnik nije toliko dobar kao što bi to mogao biti da je neki "čarobnjak - haker" bio dobro plaćen da ga napiše i da je potrošio par godina usavršavajući ga. Nemojte reći da vas nisam upozorio.

Jedina stvar u kojoj sam birao lakši put je ta da nisam opsežno razrađivao mnoge stvari koje su već dobro dokumentovane u drugim besplatnim priručnicima. Ovo se posebno odnosi na dokumentaciju specifičnu za određeni program, kao što su npr. detalji korištenja programa **mkfs**. Umjesto toga samo opisujem svrhu programa i objašnjavam njegovu upotrebu toliko koliko je dovoljno za potrebe ovog priručnika. Za daljnje informacije upućujem obzirnog čitaoca na ove druge priručnike. Obično su svi spomenuti dokumenti dio standardne **Linux** dokumentacije.

Poglavlje 2

O knjizi

2.1 Zahvale

2.1.1 Joanna

Lars je pokušao da učini ovaj priručnik što je moguće boljim i voljela bih kao osoba koja je trenutno zadužena za održavanje da nastavim taj dobar rad. Zaista bih voljela čuti vaše ideje o tome kako ga unaprijediti. Jezičke pogreške, pogrešne činjenice, ideje o novim temama koje treba obraditi, ponovo napisane odjeljke, informacije kako funkcionišu različite verzije **UNIX**-a, su samo neke od tema za koje sam zainteresovana. Moje kontakt informacije su dostupne na <http://www.iki.fi/viu>

Mnogo ljudi mi je pomoglo pri pisanju ove knjige, direktno ili indirektno. Posebno bih se željela zahvaliti Matt Welshu za inspiraciju i za vođenje **LDP**-a, Andy Oramu što me je natjerao da ponovo radim s njegovim *feedbackom*, Olafu Kirchu zato što mi je pokazao da se ovaj posao može završiti i Adamu Richteru u Yggdrasil-u i drugim ljudima zato što su mi pokazali de će ovaj rad i drugi smatrati interesantnim.

Stephen Tweedie, H. Peter Anvin, Remy Card, Theodore Ts'o su dopustili da posudim njihov rad (i da tako učinim da knjiga izgleda deblja i mnogo impresivnija): usporedba između **xia** i **ext2** datotečnih sistema, lista uređaja i opis **ext2** datotečnog sistema. Mada ovo više nisu dijelovi knjige, veoma sam im zahvalna i izvinjavam se za ranije verzije kojima su nedostajale zahvale.

Mnogo korisnih komentara je poslao veliki broj ljudi. Moja minijturna crna rupa od arhive mi ne dopušta da nađem sva njihova imena, ali neka od njih su, abecedno: Paul Caprioli, Alex Cepek, Marie-France Declerfayt, Dave Dobson, Olaf Flebbe, Helmut Geyer, Larry Greenfield i njegov otac, Stephen Harris, Jyrki Havia, Jim Haynes, York Lam, Timothy Andrew Lister, Jim Lynch, Michael J. Micek, Jacob Navia, Dan Porier, Daniel Quinlan, Jounni K Seppanen, Phillipe Steindl, G. B. Stotte. Izvinjenja svima koji su zaboravljeni.

2.1.2 Stephen

Kao najnoviji *maintainer* volio bih zahvaliti Larsu i Joanni za njihov rad na priručniku.

U knjizi kao što je ova vrlo je vjerovatno da postoje makar manje netačnosti. I sigurno će biti sekcija koje će postati zastarjele. Ako bilo šta od ovoga primjetite molim vas da pošaljete e-mail na: [<bagpuss@debian.org>](mailto:bagpuss@debian.org). Prihvatam bilo kakav oblik ispravki (diffs, obični tekst, html, bilo šta). Ja nikako nisam iznad svih drugih ljudi da ne dopustim održavanje ovako velikog teksta kao što je ovaj :o)

Velika hvala Helen Topping Shaw zato što je uzela crvenu olovku i učinila ovaj tekst mnogo boljim nego što bi inače bio.

Trenutna *home* stranica priručnika je: <http://people.debian.org/~bagpuss>

2.1.3 Adis

Hvala ljudima koji su napisali ovu knjigu (Larsu, Joanni i Stephenu).

Hvala svima onima koji su mi pomogli pri prijevodu:

Unaprijed hvala svim budućim čitaocima i kritičarima ovoga rada.

2.2 Tipografske konvencije

Pokušao sam koristiti iste tipografske konvencije kroz cijelu knjigu. Nadam se da će doprinijeti čitljivosti. Ako mislite da možete sugerirati poboljšanja slobodno me kontaktirajte.

(op. p. Tipografske konvencije se odnose na prijevod knjige! I to prvenstveno na PostScript dokumente (*.ps, *.pdf), za ostale ne mogu garantovati.)

Imena datoteka, direktorija i dokumenata :

/usr/share/doc/foo

Imena naredbi (u običnom tekstu):

fsck

Naredbe (listinzi; prompt: \$ običan korisnik, # - administrator):

\$ date

Sat Jan 25 17:20:36 CET 2003

\$

useradd

```
usage: useradd [-u uid [-o]] [-g group] [-G group,...] [-d home]
           [-s shell] [-c comment] [-m [-k template]]
           [-f inactive] [-e expire ] [-p passwd] [-r] name
           useradd -D [-g group] [-b base] [-s shell] [-f inactive]
           [-e expire ]
```

#

e-mail adrese (unutar < i >) :

<lug@linux.org.ba>

URL adrese :

<http://www.linux.org.ba>

Engleski termini¹ : (obično u zagradi)

boot

Akronimi:

SAG

I naravno, čisto iz rešpekta ;o)

Linux

UNIX

¹Neki široko prihvaćeni termini, termini za koje ne postoji adekvatan prijevod ili je prijevod "ružan" i neadekvatan. Naravno ne pada mi na pamet prevoditi riječi softver i hardver.

Poglavlje 3

Kratak pregled Linux sistema

"God looked over everything he had made, and saw that it was very good."(Genesis 1:31)

Ovo poglavlje vam daje kratak pregled **Linux** sistema. Kao prvo, opisani su glavni servisi koje obezbjeđuje sam operativni sistem. Zatim su opisani programi koji implementiraju ove servise bez suvišnih detalja. Svrha ovog poglavlja je obezbjediti razumijevanje operativnog sistema kao cjeline, a dijelovi koji čine cjelinu su opisani drugdje.

3.1 Različiti dijelovi operativnog sistema

UNIX operativni sistem sastoji se od kernela i sistemskih programa. Također postoje i aplikacijski (korisnički) programi koji rade stvarni posao za krajnjeg korisnika. Kernel je srce operativnog sistema¹. On prati stanje datoteka na disku, pokreće programe i konkurentno ih izvršava, dodjeljuje memoriju i druge resurse procesima, prima i šalje mrežne pakete, itd. Sam kernel obavlja veoma malo posla, ali on obezbjeđuje alate s kojima se svi servisi mogu izgraditi. Kernel također onemogućava direktan pristup hardveru, tako prisiljavajući svakoga da se služi alatima koje obezbjeđuje. Na ovaj način obezbjeđuje se zaštita korisnika, jednih od drugih. Ovi alati koriste se putem sistemskih poziva. Pogledajte **man** stranice sekcija 2 za više informacija o sistemskim pozivima.

Sistemski programi koriste alate koje je obezbjedio kernel da implementiraju razne servise koji se očekuju od jednog operativnog sistema. Sistemski i svi drugi programi izvršavaju se "povrh kernela" u tzv. korisničkom modu (*user mode*). Razlika između sistemskih i korisničkih programa je u namjeri: korisnički se koriste da bi se uradile korisne stvari (ili za igranje ako je u pitanju neka igra) dok su sistemski programi potrebni da bi sistem uopšte pravilno radio. Program za obradu teksta (*Word processor*) je korisnički program (aplikacija); **telnet** je sistemski program. Ta razlika je često mutna i važna samo onima koji nasilu pokušavaju sve kategorizirati.

Operativni sistem također može sadržati prevodioce (kompajlere) i njihove prateće biblioteke (naročito **GCC** i **C** programsku biblioteku posebno pod **Linux**-om), premda programski jezici ne moraju biti dio operativnog sistema. Dokumentacija, pa čak ponekad i igre mogu također biti dio operativnog sistema. Tradicionalno se operativni sistem definisao po sadržaju instalacijskih traka ili diskova, sa **Linux**-om je to nejasno jer raspršen po **FTP** serverima širom svijeta.

¹Ustvari, sam kernel često se pogrešno smatra operativnim sistemom, ali on to nije. Operativni sistem pruža mnogo više servisa od sirovog kernela.

3.2 Važni dijelovi kernela

Linux kernel se sastoji od nekoliko važnih dijelova koji kontrolišu: upravljanje procesima, upravljanje memorijom, upravljanje hardverskim uređajima (*driver-i* - upravljački programi), upravljanje datotečnim sistemom, pristup mreži i razne druge stvari.

* SLIKA 3-1 *

Vjerovatno najvažniji zadaci kernela su upravljanje procesima i memorijom (ništa drugo ne radi bez njih). Upravljanje memorijom sastoji se u dodjeljivanju memorije i *swap* prostora procesima, dijelovima kernela i *bufferu* (međuspremniku). Upravljanje procesima sastoji se u kreiranju procesa i implementiranju višezadačnosti (*multitasking-a*), mijenjanjem aktivnog procesa na procesoru.

Na najnižem nivou, kernel sadrži upravljačke programe (*driver-e*) za svaku vrstu hardvera koju podržava. Budući da postoji mnogo različitog hardvera, njihov broj je veliki. Postoje mnogi u principu isti komadi hardvera koji se razlikuju u načinu na koji su softverski kontrolisani. Sličnosti čine mogućim postojanje opštih klasa upravljačkih programa koji podržavaju slične operacije; svaki član klase ima isti interfejs prema ostatku kernela dok se razlikuje u samoj implementaciji. Npr. svi upravljački programi za diskove izgledaju isto ostatku kernela, tj. svi imaju operacije kao što su "inicijaliziraj", "pročitaj sektor N", "upiši sektor N" itd.

Neki softverski servisi koje obezbjeđuje kernel također imaju slična svojstva tako da mogu biti izdvojeni u apstraktne klase. Npr. razni mrežni protokoli izdvojeni su u jedinstveni programski interfejs, **BSD socket library**. Drugi primjer je virtuelni datotečni sistem (**VFS**) koji izdvaja apstraktne operacije datotečnog sistema od njihove implementacije. Za svaki tip datotečnog sistema postoji implementacija svake operacije. Kada neki program pokuša koristiti datotečni sistem zahtjev ide preko **VFS-a**, koji usmjerava zahtjev upravljačkom programu za stvarni datotečni sistem.

3.3 Glavni servisi UNIX sistema

Ovaj odjeljak opisuje neke važnije **UNIX** servise, ali bez detalja koje možete naći u kasnijim poglavljima.

3.3.1 INIT

Najvažniji servis **UNIX** sistema obezbjeđuje **init**. **init** se pokreće kao prvi proces u svakom **UNIX** sistemu, kao zadnja stvar koju kernel uradi prilikom podizanja sistema (*boot-a*). Kada se **init** pokrene on nastavlja *boot* proces radeći razne "dosadne" poslove (provjeravanje i montiranje datotečnih sistema, startanje pozadinskih procesa itd.)

Tačna lista stvari koje *init* radi zavisi od njegove vrste (arome), postoji više njih koje možete birati². **init** obično obezbjeđuje koncept jednokorisničkog moda (*single user mode*), gdje se niko ne može prijaviti osim *root* korisnika koji koristi *shell* u konzolnom režimu rada; uobičajeni mod se naziva višekorisnički mod (*multiuser mode*). Neke vrste **UNIX-a** generaliziraju sve ovo u tzv. "izvršne nivoe" (*run levels*). Jednokorisnički i višekorisnički modovi se smatraju za dva različita nivoa. Mogu postojati i drugi, npr. za pokretanje **X-a** sa konzole.

²op.p. Ustvari, trenutno se najviše koriste dvije vrste; **SysV** i **BSD**; obje su dobile imena po glavnim razvojnim granama **UNIX** operativnog sistema. Moderne **Linux** distribucije preferiraju **SysV** aromu.

Linux dozvoljava do 10 izvršnih nivoa, ali su obično samo neki od ovih deset već definisani. Izvršni nivo 0 je definisan kao zaustavljen sistem (*system halt*), izvršni nivo 1 kao jednokorisnički mod, izvršni nivo 6 kao restart sistema (*system reboot*). Drugi izvršni nivoi zavise od toga kako ih je vaša distribucija definisala i oni se značajno razlikuju od distribucije do distribucije. Sadržaj datoteke */etc/inittab* bi vam trebao ilustrirati koji su izvršni nivoi predefinisani i kako su definisani.

Ako je sve u redu **init** se brine o tome da je **getty** pokrenut (da bi se korisnici mogli prijaviti na sistem) i da usvoji procese bez roditelja (Svi procesi na **UNIX**-u moraju biti dio jednog i samo jednog stabla, ukoliko roditelj procesa umre **init** ga mora usvojiti - **init** je otac/majka svih procesa.)

init je zadužen da prilikom gašenja sistema "ubija" (zaustavlja) druge procese, demontira datotečne sisteme i zaustavi procesor, zajedno sa drugim zadacima za koje ste ga konfigurisali.

3.3.2 Prijavljivanje sa terminala

Prijavljivanje sa terminala (preko serijske veze) i konzole (kada **X**-i nisu pokrenuti) obezbjeđuje **getty** program. **init** pokreće posebnu instancu (kopiju) **getty**-a za svaki terminal za koji je dozvoljeno prijavljivanje. **getty** učitava ime korisnika i pokreće **login** program koji učitava šifru korisnika. Ako su korisničko ime i šifra ispravni **login** pokreće *shell*³. Kada *shell* program završi svoj rad npr. korisnik se odjavi, ili ako je **login** prekinuo izvršavanje zbog neispravnog korisničkog imena ili lozinke, **init** ovo registruje i pokreće novu instancu **getty**-a. Kernel ne zna ništa o prijavama jer sve potrebne poslove obave sistemski programi.

3.3.3 Syslog

Kernel i mnogi sistemski programi stvaraju upozorenja, poruke o greškama i druge poruke. Ponekad je veoma važno da se ove poruke mogu naknadno pregledati, čak i nakon mnogo vremena, tako da bi trebale biti zapisane u datoteke. Program koji se brine o ovome zove se **syslog**. Može se konfigurisati da sortira poruke u različite datoteke prema pošiljaocu ili stepenu važnosti. Npr. poruke kernela se često zapisuju u odvojene datoteke budući da sa često mnogo važnije i da se trebaju redovno pregledati da bi se uočili problemi.

3.3.4 Periodičko izvršavanje komandi: cron i at

I obični korisnici i sistemski administrator često mora periodično pokretati razne komande. Npr. sistemski administrator može htjeti izbrisati stare datoteke iz direktorija koji sadrže privremene datoteke (*/tmp* i */var/tmp*), radi štednje prostora na disku, budući da svi programi "ne pospremaju iza sebe".

cron servis je određen za ovaj posao. Svaki korisnik ima *crontab* datoteku, gdje upisuje komande koje želi izvršiti i vrijeme kada se trebaju izvršiti. **cron daemon**⁴ se brine o startanju komandi u specifikovanom vremenu.

³op. p. *shell* je u stvari korisnikova radna okolina; može biti tekstualna kao npr. DOS prompt, Bourne komandni *shell* ili čak grafički *shell* - **X**-i...

⁴op.p. *daemon* je proces koji se "vrti" u pozadini, obično bez mnogo interakcije sa korisnikom (osim rudimantarne mogućnosti pokretanja i zaustavljanja).

at servis je sličan **cron**-u s tom razlikom da se izvršava u zadano vrijeme, ali samo jednom, bez ponavljanja.

Pogledajte **man** stranice **cron(1)**, **crontab(1)**, **crontab(5)**, **atd(8)** za iscrpnije informacije.

3.3.5 Grafički korisnički interfejs

UNIX i **Linux** ne inkorporiraju korisnički interfejs unutar kernela; umjesto toga dozvoljavaju da on bude implementiran kao korisnički program. Ovo važi kako za tekstualni mod tako i za grafička okruženja.

Ovo čini operativni sistem fleksibilnijim, ali ima manu da je jednostavno implementirati različit korisnički interfejs za svaki program posebno, čineći tako učenje korištenja sistema težima.

Najčešće grafičko okruženje koje se koristi na Linux-u naziva se **X Window System** (skraćeno **X-i**). **X-i** također ne implementiraju korisnički interfejs nego samo alate pomoću kojih se taj interfejs može implementirati. Neki popularni window manager-i su: **windowmaker**, **icewm**, **fvwm** i **blackbox**. Postoje također i dva popularna desktop okruženja, **KDE** i **GNOME**.

3.3.6 Umrežavanje

Umrežavanje je čin povezivanja dva ili više računara tako da oni mogu međusobno komunicirati. Sami načini povezivanja i komunikacije su prilično komplikovani, ali krajnji rezultat je jako koristan.

UNIX sistemi imaju mnogo poželjnih mrežnih osobina (*features*). Većina osnovnih servisa (datotečni sistemi, ispis, *backup*, itd.) radi i preko mreže. Ovo može učiniti administraciju sistema lakšom budući da dozvoljava centralizaciju, a još uvijek iskorištava prednosti mikroracunarstva i distribuiranog računarstva, kao što su niži troškovi i veća otpornost na pogreške.

Ova knjiga daje samo kratak uvid u umrežavanje; za više informacija uključujući i osnovne informacije o funkcionisanju računarskih mreža pogledajte knjigu "*Linux Network Administrators' Guide*" (**NAG**).

3.3.7 Prijavljivanje preko mreže

Prijavljivanje preko mreže se odvija nešto drugačije od normalnog prijavljivanja. Postoji fizička serijska veza za svaki terminal preko kojeg se moguće prijaviti. Za svaku osobu koja se prijavljuje preko mreže stvara se odvojena virtuelna mrežna veza, kojih može postojati proizvoljan broj⁵. Zbog toga nije moguće pokretati odvojenu instancu **getty**-a za svaku moguću virtuelnu vezu. Također postoji mnogo različitih načina za prijavljivanje preko mreže, najvažniji predstavnici na **TCP/IP** mrežama su **telnet** i **rlogin**.

Prijave preko mreže umjesto mnoštva **getty**-a imaju po jedan daemon za svaki mogući način prijavljivanja (**telnet** i **rlogin** imaju odvojene *deamone*) koji osluškuje sve dolazeće pokušaje prijavljivanja. Kada *daemon* ustanovi pokušaj prijavljivanja pokreće novu instancu samog sebe da obradi taj pokušaj; originalna instanca nastavlja osluškivati druge pokušaje. Nova instanca radi slično **telnet** i **getty**-u.

⁵Tj. može ih biti mnogo. Mrežna propusnost je još uvijek oskudan resurs, tako da u praksi još uvijek postoji gornja granica broja istovremenih prijava preko jedne fizičke mrežne veze.

3.3.8 Mrežni datotečni sistemi

Jedna od korisnijih stvari koje se mogu uraditi s mrežnim servisima je dijeljenje datoteka preko mrežnog datotečnog sistema. Onaj koji se obično koristi razvio je **Sun** i naziva se **Network File System** ili **NFS**.

Sa **NFS**-om bilo koja operacija na datotekama koju je uradio program na jednoj mašini šalje se preko mreže drugom računaru. Ovo zavarava program tako da izgleda da su datoteke koje se nalaze na drugom računaru na istom računaru na kojem se izvršava program. Ovo izuzetno pojednostavljuje dijeljenje informacija budući da nije potrebna modifikacija programa.

3.3.9 Elektronska Pošta (e-mail)

Elektronska pošta je obično najvažniji način komunikacije pomoću računara. Elektronsko pismo je smješteno u datoteku koristeći specijalni format i specijalni *mail* programi koriste se za slanje i čitanje pošte.

Svaki korisnik posjeduje poštanski sandučić (*mailbox*) za novopristiglu poštu (datoteka specijalnog formata) gdje se smješta nova pošta. Kada neko šalje poštu, *mail* program locira *mailbox* primatelja i dodaje pismo u *mailbox* datoteku. Ako je primatelj *mailbox* na drugom računaru, pismo se šalje drugom računaru koji je onda odgovoran dostaviti ga u *mailbox*.

Mail sistem se sastoji od mnogo programa. Isporku pošte u lokalne ili udaljene *mailbox*-ove obavlja jedan program (tzv **MTA** - **Mail transfer agent** npr. **sendmail** ili **smail**) dok korisnici koriste mnoge različite programe (tzv. **MUA** - **Mail User Agent** npr. **pine** ili **elm**). *Mailbox*-ovi su obično smješteni u */var/spool/mail*.

3.3.10 Ispis (printanje)

U jednom trenutku pisač može koristiti samo jedna osoba, ali je dijeljenje pisača između više korisnika ekonomično. Pisačem dakla upravlja softver koji implementira red ispisivanja, svi poslovi na pisaču su stavljeni u jedan red i kada god je pisač završio sa jednim poslom sljedeći mu se šalje automatski. Ovo oslobađa korisnike od organiziranja reda ispisivanja i borbe oko kontrole pisača⁶.

3.3.11 Izgled datotečnog sistema

Datotečni sistem je podijeljen na mnogo dijelova; obično duž linija korijenskog datotečnog sistema sa */bin*, */lib*, */etc*, */dev* i još par drugih; */usr* datotečni sistem sa programima i podacima koji se ne mijenjaju; */var* datotečni sistem sa podacima koji se mijenjaju (kao npr. log datoteke); i */home* datotečni sistem za lične datoteke. u zavisnosti od hardverske konfiguracije i odluka sistemskog administratora podjela može biti drugačija, čak može sve biti unutar jednog datotečnog sistema.

Treće poglavlje opisuje izgled datotečnog sistema, ali za više detalja pogledajte "*Filesystem Hierarchy Standard*" (**FHS**).

⁶Umjesto toga oni formiraju red oko samog pisača, čekajući ispise, kako niko ne može natjerati softver tačno zna kada je nečiji ispis zaista gotov. Ovo je veliki podsticaj društvenim relacijama unutar ureda.

Poglavlje 4

Osvrt na stablo direktorija

Ovo poglavlje opisuje važne dijelove standardnog Linux stabla direktorija, bazirano na "*Filesystem Hierarchy Standard*" (FHS). Skiciran je standardni način razdvajanja stabla direktorija na različite datotečne sisteme sa različitim namjenama i objašnjen je motiv koji stoji iza ovog razdvajanja. Sve Linux distribucije ne prate "slijepo" ovaj standard, ali je dovoljno općenit tako da zaslužuje pažnju.

4.1 Izgled datotečnog sistema

Ovo poglavlje je donekle zasnovano na FHS, verzija 2.1, koji teži da postavi standardan način organizacije stabla direktorija Linux-a. Jedan takav standard ima prednost zbog toga što će biti lakše pisati ili prenijeti ("portati") softver na Linux i administrirati Linux mašine, jer bi sve trebalo biti na standardnim mjestima. Ne postoji neki zvanični organ-autoriteta iza ovog standarda koji prisiljava bilo koga da mu se povinuje, ali standard je dobio podršku mnogih distribucija. Nije dobra ideja kršiti standard bez nekog jakog razloga. FHS pokušava pratiti UNIX tradiciju i današnje trendove, približavajući tako Linux sisteme onima koji imaju iskustvo sa drugim UNIX sistemima. Naravno vrijedi i obrnuto.

Ovo poglavlje nije tako detaljno kao FHS. Administrator bi također trebao pročitati cijeli FHS radi potpunijeg razumijevanja.

Također se ne objašnjava detaljno svaka datoteka. Namjera je bila da se da osvrt na sistem sa aspekta datotečnog sistema. Dalje informacije o svakoj datoteci su dostupne drugdje u ovom priručniku ili u man stranicama.

Puno direktorijsko stablo je i dizajnirano tako da ga se može razbiti u manje dijelove, tako da svaki dio može biti na posebnom disku ili particiji, zbog prilagođavanja ograničenjima veličine diskova i da bi olakšao *backup* i druge administrativne zadatke. Glavni dijelovi su korijenski direktorij / , */usr* , */var* i */home* datotečni sistem (pogledajte sliku 4.1). Svaki dio ima različitu namjenu. Direktorijsko stablo je dizajnirano tako da radi dobro u mreži Linux računara koji mogu dijeliti dijelove datotečnog sistema preko ROM uređaja (npr. CD-ROM), ili preko mreže sa NFS.

* SLIKA 4-1 *

Uloge različitih dijelova stabla direktorija su opisane ispod:

- / korijenski datotečni sistem je specifičan za svaki računar posebno (obično je smješten na lokalnom disku, mada također može biti i na ramdisku ili mrežnom *drive-u*) i sadrži datoteke

koje su neophodne za podizanje sistema, i to u takvo stanje da je moguće montiranje drugih datotečnih sistema. Tako da možemo zaključiti da je sadržaj korijenskog datotečnog sistema dovoljan za jednokorisnički mod rada (*single user mode*). On će također sadržavati alate za popravke oštećenog sistema ili za snimanje datoteka sa *backup* traka.

- */usr* datotečni sistem sadrži sve komande, biblioteke, **man** stranice i druge statične datoteke koje su potrebne pri svakodnevnoj upotrebi. Ne bi trebao sadržavati datoteke specifične za bilo koju datu mašinu, niti bi te datoteke trebale biti modificirane. Ovim se omogućava dijeljenje datoteke preko mreže, što može smanjiti troškove budući da se štedi diskovni prostor (*/usr* može biti velik nekoliko stotina megabajta ili čak gigabajta). Dijeljenje */usr* preko mreže (mrežno montiranje) može olakšati administraciju (samo glavni */usr* datotečni sistem se treba mijenjati kada dodajemo i/ili nadograđujemo neku aplikaciju, a ne za svaku mašinu posebno). Čak i ako je */usr* na lokalnom disku, moguće ga je montirati *read-only* (omogućeno samo čitanje) umanjujući mogućnost oštećenja datotečnog sistema prilikom pada sistema.
- */var* datotečni sistem sadrži datoteke koje se mijenjaju, kao što su *spool* direktoriji (za *mail*, *news*, pisače itd.), log datoteke, formatirane **man** stranice i privremene datoteke. Tradicionalno, sve u */var* je bilo negdje ispod */usr*, ali je tada bilo nemoguće montirati */usr* u modu za čitanje, bez pisanja (*read-only*).
- */home* datotečni sistem sadrži korisničke *home* direktorije, tj. sve stvarne podatke u sistemu. Odvajanje korisničkih direktorija u posebno podstablo direktorija ili poseban datotečni sistem olakšava *backup*; drugi dijelovi ne trebaju *backup*, ili bar ne trebaju često jer se rijetko mijenjaju. Veliki */home* može zahtijevati razbijanje na više datotečnih sistema što zahtijeva još jedan nivo indirekcije/imenovanja ispod */home*. npr. */home/studenti* i */home/osoblje*.

Premda su različiti dijelovi stabla direktorija iznad bili nazivani datotečni sistem, nema zahtjeva da oni zaista i moraju biti na različitim datotečnim sistemima. Oni lahko mogu biti u jednom datotečnom sistemu ako je to mali, jednokorisnički sistem za korisnika koji hoće zadržati jednostavnost. Stablo direktorija također može biti podijeljeno na datotečne sisteme drugačije, u zavisnosti od veličine diskova i načina alokacije prostora u različite svrhe. Međutim, važna stvar je da sva standardna imena rade; čak i ako su */usr* i */var* na istoj particiji, imena */usr/lib/libc.a* i */var/log/messages* moraju raditi, na primjere pomjerajući datoteke ispod */var* u */usr/var* i postavljajući */var* kao simbolički link (*symlink*) na */usr/var*.

UNIX datotečni sistem struktuirira grupe datoteka prema njihovoj namjeni, tj. sve komande su na jednom mjestu, sve datoteke sa podacima na drugom, dokumentacija na trećem, itd. Druga alternativa bi bila grupisati datoteke prema programu kojem pripadaju, tj. sve datoteke **emacs**-a bi bile na jednom mjestu, sve **TeX** na drugom, itd. Problem sa ovim drugim pristupom je što otežava dijeljenje datoteka (direktorij programa često sadrži oboje, i statične, dijeljive datoteke i promjenjive, nedjeljive datoteke), pa čak i pronalaženje datoteka (npr. **man** stranice su na više različitih mjesta, a podešavanje **man** programa za pronalaženje svih je noćna mora).

4.2 Korijenski datotečni sistem

Korijenski datotečni sistem općenito bi trebao biti što je moguće manji, budući da sadrži veoma važne datoteke i da mali, rijetko modificirani datotečni sistem ima više šansi da ne bude oštećen.

Oštećen korijenski datotečni sistem obično znači da više ne možete pokrenuti sistem, osim uz specijalne mjere (npr. sa *floppy* diskete), tako da vjerovatno ne želite preuzeti takav nepotreban rizik.

Korijenski datotečni sistem općenito ne sadrži datoteke, osim možda standardnog *boot image*-a¹ obično nazvanog */vmlinuz*. Sve druge datoteke su u poddirektorijima:

<i>/bin</i>	Komande potrebne prilikom pokretanja sistema koje mogu koristiti i obični korisnici. (vjerovatno nakon <i>boot</i> -a).
<i>/sbin</i>	Slično <i>/bin</i> , ali komande nisu namijenjene običnim korisnicima, premda ih mogu koristiti ako je to potrebno i ako imaju dozvolu. <i>/sbin</i> obično nije u podrazumjevanoj stazi običnih korisnika, ali jeste u podrazumjevanoj stazi <i>root</i> -a.
<i>/etc</i>	Konfiguracijske datoteke specifične za dati računar.
<i>/root</i>	Home direktorij za <i>root</i> -a. Obično nedostupan za druge korisnike. (op. p. Neke vrste UNIX -a koriste korijenski direktorij / kao home direktorij za <i>root</i> -a)
<i>/lib</i>	Dijeljene biblioteke potrebne programima na korijenskom dat. sistemu.
<i>/lib/modules</i>	Kernel moduli, posebno oni potrebni za podizanje sistema prilikom oporavka od katastrofa (tj. mrežni i <i>driver</i> -i za datotečne sisteme)
<i>/dev</i>	datoteka uređaja. Neki od češće upotrebljivanih datotečnih uređaja su obrađeni u Poglavlju 5 on page 25.
<i>/tmp</i>	Privremene datoteke. Programi koji su pokrenuti poslije <i>boot</i> -a bi trebali koristiti <i>/var/tmp</i> , a ne <i>/tmp</i> budući da je prvi vjerovatno na disku sa više prostora. Često je <i>/tmp</i> simbolička veza (<i>symlink</i>) na <i>/var/tmp</i> .
<i>/boot</i>	Datoteke koje koristi <i>bootstrap</i> ² program, npr LILO . Slike kernela često su smještene ovdje umjesto u korijenskom direktoriju. Ukoliko postoji mnogo slika kernela, ovaj direktorij može poprilično narasti i možda bi ga bilo bolje držati u odvojenom datotečnom sistemu. Drugi razlog bi mogao biti da se osigura da su slike kernela unutar granice od prva 1024 cilindra IDE hard diska [10].
<i>/mnt</i>	Lokacija za privremena montiranja koju može koristiti sistemski administrator. Nije predviđeno da programi automatski montiraju bilo šta u <i>/mnt</i> . Može biti podijeljen u poddirektorije (npr. <i>/mnt/dosa</i> bi mogao biti <i>floppy</i> drive koji koristi MS-DOS datotečni sistem).

¹op.p. Datoteka koja sadrži kompajliran i kompresovan kernel, tj. binarna slika kernela unutar datoteke.

²op. p. bootstrap?

<i>/proc,</i>	
<i>/usr,</i>	
<i>/var,</i>	
<i>/home</i>	Lokacije za montiranje drugih datotečnih sistema [11].

4.3 */etc* direktorij

/etc direktorij sadrži mnogo datoteka. Neke od njih su opisane ispod. Za druge, utvrdite kojem programu pripadaju i pročitajte **man** stranicu tog programa. Datoteke za konfiguraciju mreže se također nalaze u */etc* i opisane su u "*Networking Administrator's Guide*" (NAG).

<i>/etc/rc0r,</i>	
<i>/etc/rc.d0r,</i>	
<i>/etc/rc?.d</i>	Skripte ili direktoriji skripti koje se izvršavaju prilikom pokretanja sistema ili mijenjanja izvršnog nivoa. Za dalje informacije pogledajte Poglavlje 9.
<i>/etc/passwd</i>	Lista korisnika, sa poljima koja označavaju korisničko ime, puno ime, <i>home</i> direktorij, enkriptiranu šifru i druge informacije o svakom korisniku. Format podataka je dokumentiran man stranici passwd(1) . U novije doba, enkriptirane šifre se češće nalaze u <i>/etc/shadow</i> . To znači da su skoro svi podaci o korisniku (osim šifre) smješteni u <i>passwd</i> datoteci. Historija i konvencije učinile su promjenu imena nepoželjnom.
<i>/etc/fdprm</i>	Tabela parametara za <i>floppy</i> disk. Opisuje izgled različitih formata <i>floppy</i> diskova. Koristi je program setfdprm . Za više informacija pogledajte man stranicu setfdprm(8) .
<i>/etc/fstab</i>	Lista automatski montiranih datotečnih sistema, prilikom pokretanja sistema sa komandom mount -a (u <i>/etc/rc</i> ili sličnoj datoteci). Pod Linux -om također sadrži podatke o <i>swap</i> područjima koje koristi swapon -a . Pogledajte sekciju 6.8.5 i man stranice mount(8) i fstab(5) za više informacija.
<i>/etc/group</i>	Slično datoteci <i>/etc/passwd</i> , samo što umjesto korisnika opisuje korisničke grupe. Pogledajte man stranicu group(5) .
<i>/etc/inittab</i>	Konfiguracijska datoteka za init .
<i>/etc/issue</i>	Poruka koju ispisuje getty prije prijave na računar. Obično sadrži kratak opis sistema ili pozdravnu poruku. Sadržaj definiše sistemski administrator.

- /etc/magic* Konfiguracijska datoteka za **file**. Sadrži opise različitih formata datoteka na osnovu kojih **file** određuje tip datoteke. Pogledajte **man** stranice **magic(5)** i **file(1)**.
- /etc/motd* Dnevna poruka, automatski ispisana nakon uspješne prijave na sistem. Sadržaj definiše sistemski administrator. Često se upotrebljava da bi se informisali korisnici, npr. upozorenja za predviđeno gašenje sistema i sl.
- /etc/mtab* Lista trenutno montiranih datotečnih sistema. Inicijalno postavljena od strane *boot* skripti i automatski modifikovana **mount** komandom. npr. koristi je komanda **df**.
- /etc/shadow* *Shadow* (eng. sjenka) datoteka sa korisničkim šiframa (lozinkama) ako je odgovarajući softver instaliran. Taj softver premješta enkriptirane šifre iz datoteke */etc/passwd* u datoteku */etc/shadow* za koju ovlasti čitanja ima samo *root*. Ovo otežava probijanje korisničkih šifri. Ako vaša distribucija pruža mogućnost korištenja *shadow* paketa preporučujemo da to i učinite.
- /etc/login.defs* Konfiguracijska datoteka za login komandu. **man** stranica je obično **login.defs(5)**.
- /etc/printcap* Slično */etc/termcap*, ali odnosi se na pisače. Koristi drugačiju sintaksu. **man** stranica **printcap(5)**.
- /etc/profile,*
- /etc/csh.login,*
- /etc/csh.cshrc* Datoteke (skripte) koje prilikom prijave ili pokretanja sistema izvršavaju Bourne ili C *shell*. Ovo omogućava administratoru da postavi globalne uobičajene postavke za sve korisnike. Pogledajte **man** stranice za željeni *shell*.
- /etc/securetty* Identificira sigurne terminale, tj. terminale sa kojih je dozvoljena prijava *root*-a. Obično su izlistani samo virtuelni terminali, tako da postaje nemoguće (ili mnogo teže) dobiti *root* privilegije provaljujući u sistem preko modema ili mreže. Nemojte dozvoliti *root* prijave preko mreže. Preferirajte prijavljivanje kao neprivilogovani korisnik i koristite komande **su** ili **sudo** da dobijete *root* privilegije.
- /etc/shells* Lista *shell*-ova kojima se vjeruje. Komanda **chsh** dopušta korisnicima izmjene njihovog **login shell**-a samo ukoliko je traženi *shell* izlistan u ovoj datoteci. **ftpd** (FTP *daemon*) proces koji obezbjeđuje **FTP** service za sistem provjerava da li je korisnikov *shell* izlistan u */etc/shells* i neće dopustiti prijavu ukoliko to nije slučaj.

/etc/termcap Baza podataka koja opisuje mogućnosti terminala. Opisuje kojim *escape* sekvencama se mogu kontrolisati razni terminali. Programi se pišu tako da umjesto direktnog upotrebljavanja sekvence koja radi na samo jednom tipu terminala zatraže ispravnu sekvencu iz */etc/termcap*. Posljedica toga je da većina programa radi na većini terminala. Pogledajte **man** stranice **termcap()** i **terminfo(5)**.³

4.4 /dev direktorij

/dev direktorij sadrži specijalne datoteke (datotečne uređaje) za sve podržane uređaje. Datoteke uređaja se imenuju prema posebnim konvencijama koje su opisane u Poglavlju 5. Kreiraju se tokom instalacije ili sa */dev/MAKEDEV* skriptom. */dev/MAKEDEV.local* je skripta koju kreira lokalni administrator za kreiranje specifično lokalnih datotečnih uređaja ili linkova (tj. onih uređaja koji nisu dio standardnog *MAKEDEV*, kao što su datoteke uređaja za neke nestandardne driver-e).

4.5 /usr datotečni sistem

/usr datotečni sistem je često veoma velik, budući da su ovdje instalirani svi programi. Obično sve datoteke u */usr* potiču od same **Linux** distribucije; lokalno instalirani programi i ostalo ide u direktorijsku strukturu ispod */usr/local*. Ovo omogućava *update* sistema novijom verzijom distribucije, ili čak novom distribucijom, bez potrebe za ponovnom instalacijom svih programa. Neki od poddirektorija */usr* su izlistani ispod (manje važni su izostavljeni; pogledajte **FHS** za više informacija).

/usr/X11R6 *X Window* sistem, sve datoteke. Da bi se pojednostavila instalacija i razvoj softvera za *X-e*, datoteke *X-a* nisu integrisane u ostatak sistema. Ispod */usr/X11R6* nalazi se direktorijsko stablo slično onome ispod */usr*.

/usr/bin Skoro sve komande. Neke komande se nalaze u */bin* ili u */sbin*.

/usr/sbin Slično */etc/termcap*, ali odnosi se na pisače. Koristi drugačiju sintaksu. **man** stranica **printcap(5)**.

/usr/share/man,

/usr/share/info,

/usr/share/doc **man** stranice, GNU **info** dokumentacija i razni druge dokumentacijske datoteke, respektivno.

/usr/include Zaglavne datoteke (*header files*) programskog jezika **C**. Ustvari, zbog konzistentnosti trebale bi biti ispod */usr/lib*, ali tradicija je preovlađujuća u korist ove lokacije.

³op.p. Primjetite da je na većini novijih UNIX sistema **termcap** baza zamijenjena sa novom bazom **terminfo**, ali je zadržan skoro identičan interfejs. Stoga pogledajte **man** stranicu **terminfo(5)**.

/usr/lib Datoteke sa podacima koji se ne mijenjaju, za programe i podsisteme, uključujući neke široko upotrebljavane lokalne konfiguracijske datoteke. Ime **lib** dolazi od engleske riječi *library* (biblioteka). Originalno su biblioteke podprograma bile smještene u */usr/lib*.

/usr/local Mjesto za lokalno instalirani softver i druge datoteke. Distribucije ne smiju instalirati softver ovdje. Ono je rezervirano samo za upotrebu lokalnog administratora. Na ovaj način on može biti apsolutno siguran da nadograđivanja distribucije neće izbrisati dodatni lokalni softver koji je on instalirao.

4.6 */var* datotečni sistem

/var sadrži podatke koji mijenjaju kada je sistem u ispravnom stanju. Poseban je za svaki sistem, tj. ne dijeli se preko mreže s drugim računarima.

/var/cache/man Keš direktorij za **man** stranice koje se formatiraju "u letu". Izvorni kod **man** stranica je obično smješten u */usr/share/man/man?* (gdje je ? broj odjeljka. Pogledajte **man** stranicu **man(7)**). Neke **man** stranice koje mogu doći unaprijed formatirane ponekad su smještene u */usr/share/man/cat**. Druge **man** stranice trebaju biti formatirane prilikom prvog pregleda. Tada se formatirana verzija smješta u */var/cache/man* tako da sljedeća osoba koja bude gledala istu stranicu ne mora čekati na formatiranje.

/var/games Svi promjenljivi podaci koji pripadaju igrama u */usr* trebali bi biti smješteni u */var/games*. To se odnosi na slučaj kada je */usr* montiran samo za čitanje (*readonly*).

/var/lib Datoteke koje se mijenjaju prilikom normalnog rada sistema.

/var/local Promjenljivi podaci za programe instalirane u */usr/local* (tj. za programe koje je instalirao sistemski administrator). Primjetite da bi čak i lokalno instalirani programi trebali koristiti druge */var* direktorije, ukoliko je to prikladno, npr. */var/lock*.

/var/lock Zaključane (*lock*) datoteke. Mnogi programi prate konvenciju kreirajući *lock* datoteke u */var/lock* označavajući time da koriste određeni uređaj ili datoteku. Drugi programi će uočiti *lock* datoteku i neće pokušati koristiti zauzeti uređaj ili datoteku.

/var/log Log datoteke za razne programe, posebno **login** (*/var/log/wtmp*, koja bilježi sve prijave i odjave sa sistema), **syslog** (*/var/log/messages*, gdje su smještene sve poruke kernela i sistemskih programa). Datoteke u */var/log* mogu proizvoljno rasti i mogu zahtijevati brisanje u redovnim intervalima.

<i>/var/mail</i>	Ovo je FHS odobrena lokacija za korisničke <i>mailbox</i> datoteke. U zavisnosti od toga koliko je vaša distribucija usaglašena sa FHS , ove datoteke su možda još uvijek u <i>/var/spool/mail</i> .
<i>/var/run</i>	Datoteke koje sadrže podatke o sistemu koji su validni do sljedećeg pokretanja. Npr. <i>/var/run/utmp</i> sadrži informacije o korisnicima koji su trenutno prijavljeni na sistem.
<i>/var/spool</i>	Direktoriji za <i>news</i> , poslove koji čekaju u redu na pisaču i druge poslove koji se mogu urediti na osnovu nekog poretka. Svaki različit <i>spool</i> ima svoj poddirektorij ispod <i>/var/spool</i> , npr. <i>news spool</i> je <i>/var/spool/news</i> . Primjetite da neke distribucije koje ne zadovoljavaju u potpunosti FHS imaju korisničke <i>malibox</i> -ove ispod <i>/var/spool/mail</i> .
<i>/var/tmp</i>	Privremene datoteke koje zauzimaju mnogo prostora ili trebaju biti dostupne duže vrijeme nego što je dozvoljeno za <i>/tmp</i> (Premda sistemski administrator ne mora dozvoliti postojanje veoma starih datoteka u <i>/var/tmp</i>).

4.7 */proc* datotečni sistem

/proc datotečni sistem je iluzorni datotečni sistem. Ne postoji na disku, umjesto toga kernel ga kreira u memoriji. Koristi se da bi obezbijedio informacije o sistemu (na početku samo o procesima, odatle potiče i naziv). Neke od važnijih datoteka i direktorija su opisane ispod. Za detaljnije informacije pogledajte **man** stranicu **proc(1)**.

<i>/proc/1</i>	Direktorij koji sadrži informacije o procesu sa identifikacijskim brojem 1. Svaki proces ima direktorij ispod <i>/proc</i> sa imenom njegovog identifikacijskog broja. (PID - <i>Process Identification Number</i>)
<i>/proc/cpuinfo</i>	Informacije o procesoru, kao što su tip, proizvođač, model, performanse...
<i>/proc/devices</i>	Lista svih <i>device driver</i> -a za koje je konfigurisan trenutno pokrenuti kernel.
<i>/proc/dma</i>	Pokazuje koji se DMA kanali koriste u datom momentu.
<i>/proc/filesystems</i>	Datotečni sistemi za koje je konfigurisan kernel.
<i>/proc/interrupts</i>	Pokazuju koji su <i>interrupt</i> -i u upotrebi (koliko ih ima) i koliko je puta korišten svaki od njih.
<i>/proc/ioports</i>	Pokazuje koji su I/O portovi trenutno u upotrebi.

<i>/proc/kcore</i>	Slika fizičke memorije sistema (RAM -a). Mada se čini da je iste veličine kao RAM , ne zauzima toliko prostora, generiše se "u letu" kada vaš program pristupa memoriji. (Zapamtite da ništa unutar <i>/proc</i> ne zauzima prostor na disku, osim ako ga ne iskopirate negdje drugo. <i>/proc</i> je samo datotečni interfejs prema kernelu.)
<i>/proc/kmsg</i>	Poruke koje izbacuje kernel. Prosljeđuju se <i>syslog</i> -u.
<i>/proc/ksyms</i>	Tabela simbola za kernel.
<i>/proc/loadavg</i>	"Prosječno opterećenje" sistema, tri besmislena indikatora o tome koliko posla treba uraditi sistem u datom momentu.
<i>/proc/meminfo</i>	Informacije o upotrebi memorije, i fizičke i <i>swap</i> -a.
<i>/proc/modules</i>	Pokazuje učitane kernel module.
<i>/proc/net</i>	Statusne informacije za mrežne protokole.
<i>/proc/self</i>	Simbolički link (<i>symlink</i>) na direktorij procesa koji pregleda <i>/proc</i>
<i>/proc/stat</i>	Razne statistike sistema, kao što je npr. broj grešaka (memorijskih) stranica (<i>page faults</i>) od pokretanja sistema.
<i>/proc/uptime</i>	Vrijeme koje je sistem pokrenut.
<i>/proc/version</i>	Verzija kernela.

Primjetite da iako su gore navedene datoteke često lahko čitljive tekstualne datoteke, ponekad mogu biti u formatu koji nije razumljiv ljudima. Postoje mnoge komande koje čine samo nešto malo više od čitanja ovih datoteka i prikazuju ih u razumljivom formatu. npr. program **free** čita */proc/meminfo* i pretvara veličine izražene u bajtima u kilobajte (i dodaje nešto malo drugih informacija također).

Poglavlje 5

Datoteke uređaja

Ovo poglavlje daje pregled datoteka uređaja i objašnjava kako se one kreiraju. Također se spominju neke uobičajene datoteke uređaja. Ukoliko imate instaliran izvorni kod kernela kanonska lista datoteka uređaja se može naći u `/usr/local/linux/Documentation/devices.txt`. Izlistani uređaji se odnose na verziju kernela 2.2.17.

5.1 *MAKEDEV* skripta

Većina datoteka uređaja će već biti kreirane i spremne za upotrebu nakon instalacije **Linux** sistema. Ukoliko, nekim slučajem, trebate kreirati neku koja ne postoji prvo bi ste trebali pokušati sa *MAKEDEV* skriptom. Ova skripta je obično locirana u `/dev/MAKEDEV`. Ukoliko se ispostavi da ona nije u vašoj podrazumijevanoj stazi (op. p. *shell* varijabla `$PATH`) onda ćete morati eksplicitno navesti njenu lokaciju.

Općenito, komanda se koristi kao npr. :

```
# /dev/MAKEDEV -v ttyS0
create ttyS0    c 4 64 root:dialout 0660
#
```

Što kreira datoteku uređaja `/dev/ttyS0` sa glavnim brojem čvora (*major node number*) 4 i sporednim (*minor*) 64, kao znakovni uređaj, s dozvolom pristupa 0660, s *root*-om kao vlasnikom i grupom *dialout*.

tyS0 je serijski port. Glavni i sporedni broj čvora su brojevi razumljivi kernelu. Kernel pristupa hardverskim uređajima prema njihovim brojevima. Budući da ljudi teško pamte brojeve koristimo imena (datoteka). Dozvola pristupa 0660 znači dozvolu čitanja i pisanja za vlasnika datoteke (*root*-a) i članove grupe *dialout*, bez dozvole pristupa za ostale korisnike.

5.2 *mknod* naredba

MAKEDEV je preferirani način kreiranja novih datoteka uređaja. Međutim, ponekad *MAKEDEV* skripta neće poznavati tip datoteke uređaja koju želite kreirati. Tu uskače naredba **mknod**. Da biste je koristili morate znati glavni i sporedni broj čvora uređaja koji dodajete. Datoteka *devices.txt* u dokumentaciji izvornog koda kernela je tu kanonski izvor informacija.

npr. pretpostavimo da naša verzija *MAKEDEV* skripte ne zna kreirati datoteku uređaja `/dev/ttyS0`. Moramo koristiti **mknod** naredbu da bi je kreirali. Ako pogledamo u *devices.txt* datoteku vidjećemo

da bi `/dev/ttyS0` trebao biti znakovni uređaj sa glavnim brojem čvora 4 i sporednim 64. Tako da imamo sve potrebne informacije da bismo kreirali tu datoteku.

```
# mknod /dev/ttyS0 c 4 64
# chown root.dialout /dev/ttyS0
# chmod 644 /dev/ttyS0
# ls -l /dev/ttyS0
crw-rw----  1 root:dialout  4,   64 Oct 23 18:23 /dev/ttyS0
```

Kao što vidite potrebni su i mnogi drugi dodatni koraci da bi se kreirala datoteka uređaja. U ovom primjeru ste vidjeli proces. Mogućnost da **MAKEDEV** skripta ne zna kreirati `/dev/ttyS0` je ekstremno mala.

5.3 Lista (datoteka) uređaja

Slijedeća lista uređaja nije ni iscrpna ni detaljna koliko bi mogla biti. Mnoge od ovih datoteka uređaja će kernel koji je kompajliran sa podrškom za dati hardver. Morat ćete pročitati dokumentaciju koja dolazi za kernel za svaki specifični uređaj.

Ukoliko mislite da postoje i drugi uređaji koji ovdje trebaju biti uključeni javite mi. Pokušat ću ih uključiti u slijedećoj verziji.

<code>/dev/dsp</code>	<i>Digital Signal Processor</i> . U osnovi, ovaj uređaj je interfejs između softvera koji proizvodi zvuk i vaše zvučne kartice. To je znakovni uređaj sa glavnim brojem čvora 14 i sporednim 3.
<code>/dev/fd0</code>	Prvi <i>floppy</i> disk uređaj. Ukoliko imate više ovih uređaja onda će oni biti numerisani sekvencijalno. Znakovni uređaj sa glavnim brojem čvora 2 i sporednim 0.
<code>/dev/fb0</code>	Prvi <i>framebuffer</i> uređaj. <i>framebuffer</i> je apstraktni sloj između softvera i grafike (hardvera). To znači da softver ne mora znati kakav hardver imate nego samo kako da komunicira sa <i>framebuffer</i> API -jem koji je dobro definisan i standardiziran. Također je znakovni uređaj sa glavnim brojem čvora 29 i sporednim 0.
<code>/dev/hda</code>	<code>/dev/hda</code> je glavni (<i>master</i>) IDE drive na primarnom IDE kontroleru. <code>/dev/hdb</code> je sporedni (<i>slave</i>) drive na primarnom kontroleru. <code>/dev/hdc</code> i <code>/dev/hdd</code> su respektivno <i>master</i> i <i>slave</i> uređaji na sekundarnom kontroleru. Svaki disk je podijeljen na particije. Particije 1-4 su primarne particije; particije 5 i više su logičke particije unutar proširenih (<i>extended</i>) particija ¹ . Datoteke uređaja koje se odnose na neku particiju sastoje se iz nekoliko dijelova. npr. <code>/dev/hdc9</code> odnosi se na particiju 9 (logičku particiju unutar neke proširene particije) na glavnom IDE drive-u na drugom IDE kontroleru. Glavni i sporedni broj čvora su malo kompleksniji nego inače. Za prvi IDE kontroler sve particije su blok uređaji

¹op. p. Primjetite da ukoliko želite koristiti logičke particije jednu od primarnih particija (obično 2, 3 ili 4) morate proglasiti *extended* particijom, unutar koje će se smjestiti ostale logičke particije. naravno to se samo odnosi na PC arhitekturu.

sa glavnim brojem čvora 3. Glavni drive (*master*) ima sporedni broj 0, a sporedni (*slave*) drive ima sporedni broj čvora 64. Za svaku particiju unutar drive-a dodajte broj particije na sporedni broj čvora drive-a. Npr. */dev/hdb5* ima glavni čvorni broj 3 a sporedni 69 (64+5). Ako je drive na drugom kontroleru princip ostaje isti, samo što je glavni broj čvora 22.

- /dev/ht0* Prvi **IDE** drive za čitanje i pisanje traka (tj. *tape drive*). Prvi slijedeći je nazvan *ht1* itd. To su znakovni uređaji sa glavnim brojem čvora 37 i sporednim brojem čvora 0 za *ht0*, 1 za *ht1* itd.
- /dev/js0* Prvi analogni džojstik. Slijedeći su *js1*, *js2* itd. Digitalni se nazivaju *djs0*, *djs1* itd. Znakovni su uređaji na glavnom čvoru 15. Analogni počinju na sporednom borju čvora 0 pa sve do 127 (više nego dovoljno i za najzagriženije igrače). Digitalni počinju na sporednom broju čvora 128.
- /dev/lp0* Prvi štampač (na paralelnom portu). Slijedeći se nazivaju *lp1*, *lp2* itd. Znakovni su uređaji na glavnom čvoru 6 i sporednom čvoru koji počinje od 0 pa naviše.
- /dev/loop0* Prvi *loopback* uređaj. *Loopback* uređaji se koriste za montiranje datotečnih sistema koji nisu locirani na blok uređajima (kao što su npr. diskovi). Npr. ako želite montirati iso9660 **CDROM image** bez "zapržavanja" na **CD** onda vam treba *loopback* uređaj. Ovo je obično transparentno korisniku i za to se brine **mount** komanda. Pogledajte **man** stranice **mount(8)** i **losetup(8)**. To su blok uređaji na glavnom čvoru 7 i sa sporednim brojem čvora koji počinje od 0.
- /dev/mixer* Ovo je dio **OSS** (*Open Sound System*) driver-a. Pogledajte **OSS** dokumentaciju na <http://www.opensound.com>
- /dev/loop0* Prvi *loopback* uređaj. *Loopback* uređaji se koriste za montiranje datotečnih sistema koji nisu smješteni na blok uređajima kao što su diskovi. Npr. ako želite montirati iso9660 **CD ROM image** izbjegavajući prvo "zapržavanje" na **CD**, pa tek onda montiranje, trebate *loopback* uređaj. Ovo je obično transparentno krajnjem korisniku i za to se brine **mount** komanda. Pogledajte **man** stranice za **mount(8)** i **losetup(8)**. *Loopback* uređaji su blok uređaji sa glavnim brojem čvora 7 sa sporednim brojem koji počinje od 0 pa sekvencijalno naviše.
- /dev/md0* Prva grupa *metadiskova*. *Metadiskovi* imaju veze sa **RAID** (Redundant Array Of Independent Disks² - Redundantno polje nezavisnih diskova). Pogledajte razne **RAID HOWTO** dokumente za više detalja. *Metadisk* uređaji su blok uređaji sa glavnim brojem čvora 9 i sporednim 0 pa naviše.
- /dev/mixer* Ovo je dio **OSS** (*Open Sound System*) driver-a. Pogledajte **OSS** dokumentaciju na <http://www.opensound.com> za više detalja. To je znakovni uređaj sa glavnim brojem čvora 14 i sporednim brojem 0.

²Možda treba Inexpensive umjesto Independent?

- /dev/null*** Kofa za bite. Crna rupa u koju možete slati podatke koje više nikada nećete vidjeti. Sve što pošaljete u ***/dev/null*** će nestati. Ovo može biti korisno, npr. ako želite pokrenuti neku komandu, ali ne želite bilo kakav odgovor na ekranu. To je znakovni uređaj sa glavnim čvorom 1 i sporednim 3.
- /dev/psaux*** **PS/2** port za miša. Znakovni uređaj sa glavnim čvorom 10 i sporednim 1.
- /dev/pda*** **IDE** diskovi na paralelnim *port*-ovima. Imenovani su slično diskovima na unutrašnjim **IDE** kontrolerima (***/dev/hd****). To su blok uređaji na glavnom čvoru 45. Sporedni brojevi čvorova zahtijevaju pojašnjenje. Prvi uređaj je ***/dev/pda*** i on je na sporedno čvoru 0. Particije na ovom uređaju se nalaze dodavanjem broja particije na sporedni broj čvora uređaja. Svaki uređaj je limitiran na 15 particija (za razliku od 63 kod **IDE** diskova). ***/dev/pdb*** starta na sporednom čvoru 16, ***/dev/pdc*** na 32, ***/dev/pdd*** na 48. Ova vas šema ograničava na 4 paralelna diska, svaki po 15 particija.
- /dev/pcd0*** CD ROM uređaji na paralelnim *port*-ovima . Numerisani od 0 pa naviše. To su blok uređaji na glavnom čvoru 46. ***/dev/pcd0*** je na sporednom čvoru 0, sa slijedećim uređajima na sporednim čvorovima 1, 2, 3 itd.
- /dev/pt0*** *tape* uređaji (magnetne trake) na paralelnom *portu*. Trake nemaju particije pa su jednostavno numerisani sekvencijalno. To su znakovni uređaji na glavnom čvoru 96. Sporedni brojevi čvorova počinju od 0, za ***/dev/pt0***, 1 za ***/dev/pt1*** itd.
- /dev/parport0*** *raw* (direktni?) paralelni *portovi*. Većina uređaja koji su priključeni na paralelne *portove* imaju sopstvene *driver*-e. Ovo je uređaj koji omogućava direktan pristup paralelnom *portu*. To je znakovni uređaj na glavnom čvoru 99. Sporedni čvorovi se numerišu od nula pa naviše.
- /dev/random***,
- /dev/urandom*** Ovo su kernel generatori slučajnih brojeva. ***/dev/random*** je nedeterministički generator, što znači da vrijednost slijedećeg broja ne može biti pogođena na osnovu njegovog prethodnika. Koristi entropiju hardvera sistema da generira brojeve. Kada nema dovoljno entropije onda mora čekati prije nego što generira još brojeva. ***/dev/urandom*** radi slično. Inicijalno i on koristi entropiju, ali kada nema više entropije nastaviće generirati brojeve koristeći pseudo generator slučajnih brojeva. To se smatra manje sigurnim za vitalne namjene kao što je generiranje parova kriptografskih ključeva. Ukoliko je sigurnost vaša glavna briga koristite ***/dev/random***, ukoliko je to brzina koristite ***/dev/urandom***. To su znakovni uređaji na glavnom čvoru 1, sa sporednim čvorom 8 za ***/dev/random*** i 9 za ***/dev/urandom***.
- /dev/zero*** Ovo je jednostavan način da dobijete mnogo nula. Svaki put kada čitate sa ovoga uređaja on vraća nule. Ovo ponekad može biti korisno , npr.

kada želite datoteku fiksne veličine, ali vas ne zanima njen sadržaj. To je znakovni uređaj na glavnom čvoru 1 i sporednom 5.

Poglavlje 6

Korištenje diskova i ostalih medija za smještaj podataka

”On a clear disk one can seek forever.”

Kada instalirate ili pravite *upgrade* vašeg sistema imate poprilično posla oko vaših hard diskova. Morate kreirati datotečne sisteme na diskovima tako da se na njih mogu snimati datoteke i trebate rezervirati prostor za različite dijelove vašeg sistema.

Ovo poglavlje objašnjava sve ove početne aktivnosti. Kada jednom podesite vaš sistem obično više nema potrebe da ponovo prolazite kroz ove iste postupke, osim ukoliko želite koristiti *floppy* diskove. Vratit ćete se na ovo poglavlje ukoliko dodate ukoliko želite dodati novi disk ili ukoliko želite fino podesiti korištenje diskova na vašem sistemu.

Osnovni zadaci u administraciji diskova su:

- Formatirajte disk. To uključuje više stvari, kao što su provjera loših sektora. (Za većinu novijih diskova formatiranje nije potrebno.)
- Podijelite disk na particije, ukoliko ga želite koristiti za aktivnosti koje ne bi trebale utjecati jedna na drugu. Jedan od razloga za podjelu diska je ukoliko želite koristiti više operativnih sistema na jednom disku. Drugi razlog je to da odvojite korisničke datoteke od sistemskih, što pojednostavljuje *backup* i pomaže u sprječavanju oštećenja sistemskih datoteka.
- Kreirajte datotečni sistem (odgovarajućeg tipa) na svakom disku ili particiji. Sama egzistencija diska nema neki značaj za **Linux** ukoliko na njemu nema datotečnog sistema; tek onda se mogu kreirati datoteke i možete im pristupiti.
- Montirajte različite datotečne sisteme tako da formiraju jedinstvenu strukturu (stablo), ili automatski, ili ručno, u zavisnosti od potreba. (Datotečne sisteme koje ručno montirate obično ćete morati i ručno demontirati.)

Poglavlje 7 sadrži informacije o virtuelnoj memoriji i keširanju kod diskova, na koje također morate obratiti pažnju prilikom korištenja diskova.

6.1 Dvije vrste uređaja

UNIX, a prema tome i **Linux**, sve uređaje svrstava u dvije klase: blok uređaji sa slučajnim pristupom (kao što su npr. diskovi) i znakovni uređaji (kao što su trake i serijske linije), koji mogu biti serijski ili također sa slučajnim pristupom. Svaki podržani uređaj je predstavljen datotečnom sistemu u obliku datoteke uređaja (*device file*). Kada čitate ili pišete u datoteku uređaja, podaci dolaze/idu uređaju kojeg predstavlja datoteka uređaja. Na ovaj način je izbjegnuta potreba za specijalnim programima (ili posebnom metodologijom pisanja korisničkih programa, kao što su hvatanje *interrupt-a* ili *polling* serijskog porta); na primjer, da pošaljete datoteku na štampanje mogli biste otkucati:

```
$ cat filename > /dev/lp1
$
```

I sadržaj datoteke je odštampan (naravno, datoteka mora niti u obliku u kojem je štampač razumije). Međutim, budući da nije dobra ideja da više ljudi istovremeno šalje datoteke na printanje, obično se koristi specijalni program koji šalje datoteke štampaču (obično se zove **lpr**). Ovaj program osigurava da se u jednom trenutku vremena štampa samo jedna datoteka i automatski šalje druge datoteke na štampanje čim se završi štampanje prethodne. Sličan postupak je potreban za većinu drugih uređaja. Ustvari, rijetko morate uopće brinuti o datotekama uređaja.

Budući da su datoteke uređaja prikazane kao stvarne datoteke (unutar /dev direktorija), lako je vidjeti koje datoteke uređaja postoje koristeći **ls** ili neku drugu sličnu komandu. U *output-u* komande **ls -l**, prva kolona sadrži tip datoteke i njene dozvole pristupa. Na primjer listing serijskog uređaja mogao bi dati:

```
$ ls -l /dev/ttyS0
crw-rw-r--  1 root    dialout   4,  64 Aug 19 18:56 /dev/ttyS0
$
```

Prvi znak u prvoj koloni tj. 'c' u **crw-rw-r--**, govori informiranom korisniku tip datoteke, u ovom slučaju datoteke znakovnog uređaja (op.p. (c)haracter = znak). Za obične datoteke prvi znak je '-', za direktorije 'd', a za blok uređaje 'b'; pogledajte **man** stranicu komande **ls** za više informacija.

Primjetite da obično postoje datoteke uređaja za sve moguće uređaje, čak i ako sam uređaj nije fizički instaliran. Zbog toga, činjenica da imate datoteku */dev/sda*, to ne znači da stvarno imate **SCSI** hard disk. Postojanje svih ovih datoteka pojednostavljuje dodavanje i instalaciju novog hardvera (nema potrebe da znate ispravne parametre za datoteke uređaja niti morate ručno kreirati datoteke uređaja za novi uređaj).

6.2 Hard diskovi

Ovaj odjeljak nas uvodi u terminologiju koja se odnosi na hard diskove. Ukoliko već poznajete pojmove i koncepte možete ga preskočiti.

Pogledajte Sliku 6-1 za šematski prikaz važnih dijelova hard diska. Hard disk se sastoji od jedne ili više kružnih *ploča* [12] na kojima su jedne ili obje strane presvučene magnetnom supstancom koja se koristi za snimanje podataka. Za svaku površinu postoji glava za *čitanje-pisanje*. Ploče se okreću na zajedničkoj osovinu; tipična brzina rotacije je 5400 ili 7200 rotacija u minuti (**rpm** - *rotations per minute*), premda postoje vrhunski hard diskovi sa većim brzinama (op.p. npr. *Seagate*

Cheetah 10000 rpm) ili stariji hard diskovi sa manjom brzinom rotacije. Glave se pomijeraju duž radijusa ploča; ovo kretanje, kobinovano sa rotacijom ploča omogućava glavama da pristupe svim dijelovima površina.

Procesor (CPU) i stvarni disk komuniciraju preko *disk kontrolera*. Ovo oslobađa ostatak računara od obaveze da poznaje korištenje diska, budući da se mogu napraviti kontroleri za različite tipove diskova, ali sa istim interfejsom prema ostatku računara. Zbog toga, računar može jednostavno reći "hej disk, daj mi ono što želim", umjesto duge i komplikovane serije električnih signala da bi se pomijerila glava na ispravnu lokaciju i čekanja da se korektan dio površine pozicionira ispod glave. (U stvarnosti, interfejs prema kontroleru je još uvijek kompleksan, ali mnogo manje nego da kontroler uopšte ne postoji). Kontroler također može raditi i druge stvari kao što su keširanje ili automatsko zamjenjivanje loših sektora.

Obično i ne trebate znati više stvari o hardveru (tj. hard disku) od onoga što smo rekli u prethodnom pasusu. Također postoje i druge stvari, kao što je motor koji okreće ploče i pomijera glavu i elektronika koja kontrolira funkcionisanje mehaničkih dijelova, ali one najvećim dijelom nisu relevantne za razumijevanje principa rada hard diska.

Površine su obično podijeljene na koncentrične prstenove, nazvane *staze*, a one su podijeljene na *sektore*. Ova podjela se koristi da se specificiraju lokacije na hard disku i da bi se alocirao prostor za datoteke. Da bi našli određeno mjesto na hard disku možete reći "površina 3, staza 5, sektor 7". Obično je broj sektora isti za sve staze, ali neki hard diskovi stavljaju više sektora na vanjske staze (svi sektori su iste fizičke veličine, tako da ih više stane na vanjske staze). Tipično, sektor sadrži 512 bajta podataka. Sam disk ne može rukovati manjom količinom podataka od jednog sektora.

* Slika 6-1 Šematski prikaz hard diska *

Svaka površina je podijeljena na staze (i sektore) na isti način. Ovo znači da kada je glava jedne površine na nekoj stazi, glave svih drugih površina su također na odgovarajućim stazama. Sve odgovorajuće staze uzete zajedno čine *cilindar*. Da bi se glave pomijerile od jedne staze (cilindra) do druge, potrebno je vrijeme. Postavljanjem podataka kojima se pristupa istovremeno (npr. datoteku) tako da se nalaze unutar jednog cilindra nije potrebno pomijerati glave da bi se sve pročitalo. Ovo poboljšava performanse. Nije moguće uvijek postaviti datoteke na ovaj način; za datoteke koje su razučene na više mjesta u hard disku kažemo da su *fragmentirane*.

Broj površina (ili glava, što je ista stvar), cilindara i sektora mnogo varira. Specifikacija koja sadrži sve ove brojeve naziva se *geometrija* diska. Geometrija se obično snima na specijalnoj, baterijski napajanoj memorijskoj lokaciji nazvanoj **CMOS RAM**, odakle ga može pročitati operativni sistem prilikom podizanja ili inicijalizacije *drivera*.

Nažalost, **BIOS** [13] ima ograničenja u dizajnu, što čini nemogućim specificiranje broja staze koji je veći od 1024 u **CMOS RAM**-u, previše malo za veliki hard disk. Da bi se ovo prevazišlo, kontroler hard diska laže o geometriji i *prevodi adrese* koje je dao računar u nešto što odgovara stvarnosti. Na primjer, disk može imati 8 glava, 2048 staze i 35 sektora po stazi [14]. Njegov kontroler može lagati računaru i tvrditi da ima 16 glava, 1024 staze i 35 sektora po stazi, ne prekoračujući limit na broj staza, prevođenjem adresa koje da računar polovljenjem broja glave i udvostručavanjem broja staze. Ova matematika u stvarnosti može biti još komplikovanija, zato što brojevi nisu tako "fini" (ali, detalji nisu toliko važni za razumijevanje detalja). Ovo prevođenje remeti viđenje operativnog sistema o organizaciji diska, čineći nepraktičnim trik smještanja svih podataka unutar jednog cilindra da bi se poboljšale performanse.

Ovaj prijevod adresa je problem samo za **IDE** hard diskove. **SCSI** diskovi koriste sekvencijalne brojeve sektora (tj. kontroler prevodi sekvencijalni broj sektora u trojku glava, cilindar i sektor) i pot-

puno drugačiju metodu komunikacije između CPU i diska, tako da nemaju te probleme. Međutim, primjetite da računar također ne mora znati ni pravu geometriju SCSI diska.

Budući da **Linux** ne poznaje stvarnu geometriju diska, njegovi datotečni sistemi ni ne pokušavaju smještati datoteke unutar jednog cilindra. Umjesto toga, pokušavaju dodijeliti sekvencijalne brojeve datotekama, što gotovo uvijek rezultira sličnim performansama. Ovo pitanje se dalje komplikuje sa keš memorijom na kontrolerima i automatskim dohvatom podataka (*prefetch*) od strane kontrolera.

Svaki hard disk je predstavljen različitom datotekom uređaja. Obično može postojati samo dva ili četiri **IDE** hard diska. Oni su poznati pod nazivima */dev/hda*, */dev/hdb*, */dev/hdc* i */dev/hdd* respektivno. **SCSI** hard diskovi se nazivaju */dev/sda*, */dev/sdb* itd. Slične metode imanovanja se koriste i za druge tipove hard diskova, pogledajte poglavlje 5 za više informacija. Primjetite da datoteke uređaja za hard diskove omogućavaju pristup cijelom hard disku, bez obzira na particije (o kojima ćemo raspravljati u nastavku), te da je lako pogriješiti i "zabrljati" sa particijama i podacima koji se na njima nalaze ukoliko niste oprezni. Datoteke uređaja za hard diskove su obično potrebne jedino da dobijete pristup **MBR** (*Master Boot Record*) o čemu ćemo raspravljati u nastavku.

6.3 Floppy diskovi

A

6.4 CD-ROM

CD-ROM drive optički čita plastikom presvučen disk. Informacija je snimljena na površini diska [15] u malim "rupama" poredanim duž spirale koja ide od centra prema krajevima diska. Uređaj upravlja kretanjem laserske zrake duž spirale i na taj način čita podatke sa diska. Kada zraka lasera prelazi preko rupe reflektuje se na jedan način, a kada prelazi preko glatke površine reflektuje se na drugi način. na ovaj način se mogu kodirati biti, a prema tome i informacije. Ostatak je jednostavan, prosta mehanika.

CD-ROM uređaji su spori u odnosu na hard diskove. Tamo gdje tipični hard disk ima prosječno vrijeme traženja (*seek time*) manje od 15 ms, brzi **CD-ROM drive** može trebati i par desetinki sekunde. Stvarna protočnost podataka je prilično visoka, u stotinama kilobajta u sekundi. Ova sporost znači da **CD-ROM** uređaji nisu toliko ugodni za korištenje kao hard diskovi (Neke **Linux** distribucije obezbjeđuju "live" datotečne sisteme na **CD-ROM-u**, čineći tako kopiranje datoteka na hard disk nepotrebnim, te olakšavajući instalaciju i smanjujući zahtjeve za prostorom na hard disku). **CD-ROM** je izvrstan (instalacioni) medij za distribuciju softvera budući da pri instalaciji softvera maksimalna brzina nije važna.

Postoji više načina kako da se urede podaci na **CD-ROM-u**. Najpopularniji je onaj koji specifi-cira međunarodni standard **ISO 9660**. Ovaj standard definiše minimalan datotečni sistem, koji je čak oskudniji od datotečnog sistema koji koristi **MS-DOS**. S druge strane, zbog toga što je minimalan, svaki operativni sistem bi ga trebao moći razumjeti i mapirati u svoj datotečni sistem.

Za normalnu upotrebu pod **UNIX-om ISO 9660** datotečni sistem nije upotrebljiv, tako da je on proširen sa tzv. **Rock Ridge** ekstenzijama. On dozvoljava duža imena datoteka, simboličke linkove i ima mnogo drugih "finih" osobina, čineći tako **CD-ROM** usporedivim sa bilo kojim drugim **UNIX** datotečnim sistemom. Ono što je još bolje, **Rock Ridge** je još uvijek validan **ISO 9660** datotečni sistem¹ što ga čini upotrebljivim i na drugim operativnim sistemima osim **UNIX-a**. **Linux** podržava

¹op. p. Tehničkim rječnikom on je *superset* (nadskup) **ISO 9660**.

oba, i **ISO 9660** i **Rock Ridge** ekstenzije, ekstenzije se prepoznaju i koriste automatski.

Međutim, datotečni sistem je samo pola bitke. Većina **CD-ROM** uređaja sadrži podatke kojima može pristupiti samo specijalan program, a većina ovih programa ne rade pod **Linux**-om (osim možda pod **dosemu**, **Linux MS-DOS** emulatorom ili pod **wine** - Windows emulatorom [16]). Postoji i komercijalni produkt, **WMWare** koji može softverski emulirati čitav **x86** baziran računar [17]).

I **CD-ROM drive** ima svoju datoteku uređaja. Postoji više načina na koji se može povezati **CD-ROM** uređaj sa računarom: preko **SCSI** interfejsa, preko zvučne kartice, ili preko **EIDE** interfejsa. Bavljenje hardverom je izvan opsega ove knjige, ali dovoljno je da znate da tip interfejsa određuje koja datoteka uređaja će se koristiti.

6.5 Trake

Tape drive koristi trake slične [18] audio kasetama. *Tape* uređaj je po svojoj prirodi serijski, što znači da ukoliko želite pristupiti nekom dijelu trake morate proći/pregledati sve dijelove prije traženog dijela. S druge strane disk je uređaj sa slučajnim pristupom, tj. možete direktno "skočiti" na bilo koji dio diska. Serijski karakter *tape* uređaja ih čini (veoma) sporim.

S druge strane, trake je relativno lako proizvesti, budući da ne trebaju biti brze. Također se mogu lako učiniti prilično dugim i proporcionalno tome mogu sadržavati veliku količinu informacija. Ovo ih čini veoma pogodnim kao medij za arhiviranje i *backup*, jer za te svrhe nije potrebna velika brzina, ali su bitni niski troškovi i veliki kapaciteti.

6.6 Formatiranje

Formatiranje je proces upisivanja oznaka na magnetni medij koje služe za označavanje staza i sektora. Prije nego što je disk formatiran njegova magnetna površina je neuređena gomila magnetnih signala. Kada je formatiran, određeni red je unešen u taj haos u osnovi crtanjem linija gdje idu staze i gdje su podijeljene u sektore. Stvarni detalji nisu baš tačno ovakvi, ali to je irelevantno. Ono što je važno je to da se disk ne može koristiti ukoliko nije formatiran.

Terminologija je pomalo zbunjujuća: u **MS-DOS** i **MS Windows** svijetu riječ formatiranje se odnosi na proces kreiranja datotečnog sistema (o čemu će biti riječi ispod). Također se ta dva procesa često kombinuju, posebno kod *floppy* disketa. Kada je neophodno napraviti razliku, stvarno formatiranje se naziva formatiranje niskog nivoa (*low-level format*), a kreiranje datotečnog sistema formatiranje visokog nivoa (*high-level format*). U **UNIX** svijetu ovo se naziva formatiranje i kreiranje datotečnog sistema, respektivno, tako da će ta konvencija biti korištena i u ovoj knjizi.

Za **IDE** i neke **SCSI** diskove formatiranje se u stvari obavlja u fabrici i ne treba se ponavljati; zbog toga se većina ljudi rijetko treba brinuti oko toga. Ustvari, formatiranje hard diska može uzrokovati da on radi lošije, npr. zato što disk treba biti formatiran na specijalan način da bi radila automatska zamjena oštećenih sektora.

Diskovi koji se trebaju ili mogu formatirati često svakako zahtijevaju poseban program zato što je interfejs prema logici koja upravlja formatiranjem različit od diska do diska. Program za formatiranje često dolazi uz **BIOS** ili kao **MS-DOS** program. Ni prvo ni drugo se ne mogu lako koristiti iz **Linux**-a.

Tokom formatiranja može se naići na oštećena područja na disku koja se nazivaju loši blokovi (*bad blocks*) ili loši sektori (*bad sectors*). O ovome se ponekad brine i sam hard disk, ali čak i tada,

ukoliko postoji više loših sektora nešto treba učiniti da bi se izbjeglo korištenje ovih dijelova diska. Logika koja ovo radi ugrađena je u datotečni sistem, a sam način kako se ove informacije dodaju datotečnom sistemu opisan je ispod. Kao alternativa, može se kreirati mala particija koja zauzima samo oštećene dijelove diska, ovaj pristup bi mogao biti dobra ideja ukoliko su oštećena područja velika budući da datotečni sistemi ponekad mogu imati problema sa veoma velikim oštećenim područjima.

Floppy diskovi se formatiraju sa komandom **fdformat** kojoj kao parametar prosljedimo ime datoteke uređaja datog *floppy*-a. Npr. slijedeća komanda bi formatirala 3.5-inčni **HD floppy**, na prvom *floppy drive*-u:

```
$ fdformat /dev/fd0H1440
Double-sided, 80 tracks, 18 sec/track. Total capacity 1440 kB.
Formatting ... done
Verifying ... done
$
```

Primjetite da ukoliko želite koristiti uređaj sa automatskom detekcijom (npr. */dev/fd0*), morate prvo podesiti parametre istog sa **setfdprm**. da bi postigli isti efekat kao iznad morali biste uraditi slijedeće:

```
$ setfdprm /dev/fd0 1440/1440
$ fdformat /dev/fd0
Double-sided, 80 tracks, 18 sec/track. Total capacity 1440 kB.
Formatting ... done
Verifying ... done
$
```

Obično je lakše odmah izabrati ispravnu datoteku uređaja koja odgovara tipu uređaja. Primjetite da nije mudro formatirati *floppy* uređaje da sadrže više informacija nego zašto su predviđeni².

fdformat će također provjeriti ispravnost *floppy*-a, tj. provjeriti postojanje loših blokova. Pokušavaće provjeriti loše blokove više puta (obično ovo možete i čuti, buka koju stvara *drive* značajno se promjeni). Ukoliko je *floppy* neznatno oštećen (ako je glava za čitanje-pisanje zaprljana možemo dobiti lažne poruke o greškama) **fdformat** se ne žali, ali prava greška će prekinuti proces provjere. Kernel će ispisati *log* poruke za sve **I/O** greške koje nađe; one se preusmjeravaju na konzolu ili, ukoliko se koristi **syslog** u datoteku */usr/log/messages*. **fdformat** neće reći tačnu lokaciju oštećenja (niti se za to obično treba brinuti, *floppy* diskovi su dovoljno jeftini tako da se oni oštećeni mogu jednostavno odbaciti).

```
$ fdformat /dev/fd0H1440
Double-sided, 80 tracks, 18 sec/track. Total capacity 1440 kB.
Formatting ... done
Verifying ... read: Unknown error
$
```

Možete koristiti komandu **badblocks** u potrazi za diskom (uključujući *floppy*) ili particijom koja sadrži loše blokove. Ona ne formatira disk tako da se može koristiti za provjeru, čak i ako na disku već postoji datotečni sistem. Primjer ispod provjerava 3.5-inčni *floppy* sa dva loša bloka.

²op. p. Pogledajte malo */dev/fd** uređaje! Čak i obični 3.5" 1440kB *floppy* diskovi mogu, poslije specijalnog formatiranja, sadržavati više podataka, npr. 1720kB.

```
$ badblocks /dev/fd0H1440 1440
```

```
718
```

```
719
```

```
$
```

badblocks ispisuje brojeve loših blokova koje nađe. Većina vrsta datotečnih sistema mogu izbjeci loše blokove. Održavaju listu poznatih loših blokova koja je inicijalizirana prilikom kreiranja datotečnog sistema i može se poslije izmijeniti. Početnu potragu za lošim blokovima može obaviti komanda **mkfs** (koja kreira datotečni sistem), ali kasnije provjere bi trebalo obavljati sa **badblocks** i novi loši blokovi bi se trebali dodavati na "crnu listu" sa **fsck**. **mkfs** i **fsck** ćemo opisati kasnije.

Mnogi moderni hard diskovi automatski registruju loše blokove i pokušavaju ih popraviti zamjenjujući ih specijalnim, unaprijed rezervisanim ispravnim blokom. Ovo bi trebalo biti dokumentovano u priručniku vašeg diska. Naravno, ukoliko vas zanima šta se to ustvari dešava. Čak i ovakvi diskovi mogu otkazati, ukoliko broj loših blokova postane prevelik, premda postoje velike šanse da će do tada biti toliko oštećen da će biti neupotrebljiv.

6.7 Particije

Hard disk može biti podijeljen u više *particija*³. Svaka particija funkcioniše kao zaseban hard disk. Ideja je da ukoliko imate jedan hard disk, a želite imati dva (ili više) operativna sistema na njemu, možete ga podijeliti na dvije ili više particija. Svaki operativni sistem koristi svoju particiju kako želi i ne dira "tuđu". Na ovaj način dva operativna sistema mogu mirno koegzistirati na istom disku. Bez particija bi morali kupiti hard disk za svaki operativni sistem koji koristite.

Floppy diskovi obično nisu podijeljeni u particije. Ne postoji tehnički razlog zašto ne bi mogli, ali, budući da su veoma malog kapaciteta, particije bi bile od koristi u samo u iznimnim slučajevima. Također, **CD-ROM** obično nisu izdijeljeni u particije jer ih je lakše koristiti kao jedan veliki disk i rijetko postoji potreba za postojanjem više operativnih sistema na **CD-u**.⁴

6.7.1 MBR, boot sektori i tabela particija

Informacije o particijama na nekom hard disku smještene su u/na prvom sektoru diska (tj. na prvom sektoru, provoj stazi i prvoj ploči diska). Ovaj prvi sektor se naziva **MBR** (*Master boot record*) diska. **BIOS** učitava **MBR** prilikom podizanja sistema i pokušava pokrenuti jedan mali program⁵. Taj program čita tabelu particija, zatim provjerava koja particija je označena kao aktivna i čita prvi sektor te particije, tj. *boot* sektor date particije (i **MBR** je *boot* sektor, ali ima specijalan status i prema tome i specijalno ime). *Boot* sektor sadrži drugi mali program (*bootstrap* program) koji učitava prvi dio operativnog sistema smještenog na toj particiji i pokreće ga.

Ova šema particija nije ugrađena u hardver, čak ni u **BIOS**. To je samo konvencija koju mnogi operativni sistemi prate. Oni koji ne prate su iznimke općeg pravila. Neki operativni sistemi podržavaju particije, ali zauzimaju samo jednu particiju na hard disku i upotrebljavaju svoju šemu podjele unutar

³op. p. Prijevod ove čitave rečenice je malo čudan jer riječ particija na bosanskom jeziku također znači podjela.

⁴op.p. Svi **BIOS**-i koji podržavaju standard **El-Torito** mogu pokrenuti operativni sistem sa **CD-ROM**-a.

⁵op.p Dosta česti su **LILLO** ili **GRUB** s kojima ste se vrlo vjerovatno već susreli, međutim moguće je i da se na **MBR** nalazi samo *bootstrap* program koji direktno učitava kernel operativnog sistema radnu memoriju. Za više informacija pogledajte **BOOTDISK-HOWTO**.

te particije⁶. Ovi zadnji mirno koegzistiraju sa drugim operativnim sistemima (uključujući i **Linux**) i ne zahtijevaju bilo kakve posebne mjere, ali operativni sistem koji ne podržava particije ne može koegzistirati na istom hard disku sa drugim operativnim sistemom.

Kao mjera predostrožnosti, dobra ideja je zapisati tabelu particija na papir, tako da ukoliko dođe do oštećenja tabele ne morate izgubiti sve datoteke. (Oštećena tabela particija može se popraviti programom **fdisk**). Relevantne informacije možete dobiti pomoću komande: **fdisk -l**

```
$ fdisk -l /dev/hda
Disk /dev/hda: 15 heads, 57 sectors, 790 cylinders
Units = cylinders of 855 * 512 bytes

Device Boot   Begin    Start    End    Blocks    Id    System
/dev/hda1          1        24      24    10231+    82    Linux swap
/dev/hda2         25        25      48    10260    83    Linux native
/dev/hda3         49        49     408   153900    83    Linux native
/dev/hda4        409       409     409   163305     5    Extended
/dev/hda5        409       409     744   143611+    83    Linux native
/dev/hda6        745       745     790   19636+    83    Linux native

$
```

6.7.2 Proširene (*extended*) i logičke particije

Originalna šema particioniranja za **PC** dozvoljavala je samo četiri particije. Vrlo brzo se ispostavilo da je ovo premalo u stvarnom životu, dijelom zbog toga što ljudi žele više od četiri operativna sistema (**Linux**, **MS-DOS**, **OS/2**, **Minix**, **FreeBSD**, **NetBSD** ili **Windows NT**, da nabrojimo samo neke), ali primarno zbog toga što je ponekad dobra ideja imati nekoliko particija za jedan operativni sistem. Npr. *swap* prostor je najbolje rezervisati na posebnoj particiji pod **Linux**-om umjesto na primarnoj **Linux** particiji zbog bržeg rada (pogledajte ispod).

Da bi se prevazišao ovaj problem (greška u dizajnu), "izumljene" su *proširene* particije. Ovaj trik dozvoljava podjelu primarne particije u više podparticija. Takva primarna particija se naziva proširena (*extended*) particija, a podparticije logičke particije. Ponašaju se kao obične primarne particije, ali su kreirane na drugačiji način. Ne postoji razlika u brzina rada između primarnih i logičkih particija.

Struktura particija hard diska mogla bi ličiti onoj na slici 6-2. Disk je podijeljen na tri primarne particije, od kojih je druga primarna particija podijeljena na dvije logičke particije. Dio diska ne pripada nijednoj particiji. Disk kao cjelina i svaka primarna particija imaju *boot* sektor.

* SLIKA 6.2 *

6.7.3 Tipovi particija

Tabele particija (ona u **MBR** i one u logičkim particijama) sadržavaju jedan bajt na svaku particiju koji identifikuje tip date particije. Ovo je pokušaj označavanja operativnog sistema koji koristi tu particiju i za u koju svrhu je upotrebljava. Svrha toga je da se spriječi slučajna upotreba iste particije

⁶op. p. **FreeBSD** (i ostali **BSD UNIX**-i na PC-računarima) unutar jedne **PC** hard disk particije koriste svoju posebnu šemu particija, tzv. *slices*.

od strane dva različita operativna sistema. Međutim, u stvarnom životu, operativni sistemi ne mare mnogo o ovom bajtu koji označava tip particije. Npr. za **Linux** uopšte nije važan. A ono što je još gore, neki⁷ ih koriste pogrešno npr. neke vrste **DR-DOS**-a ignoriraju najznačajniji bit ovog bajta dok druge ne.

Ne postoji agencija za standardizaciju koja odlučuje koji tip particije označava svaka bajt vrijednost ovoga bajta, ali neke široko prihvaćene vrijednosti su prikazane u Tabeli 6-1. Kompletan lista je dostupna uz **Linux fdisk** program.

Tabela 6-1. Tipovi particija

0	Empty	40	Venix 80286	94	Amoeba BBT
1	DOS 12 - bit FAT	51	Novell?	a5	BSD/386
2	XENIX root	52	Microport	b7	BSDI fs
3	XENIX usr	63	GNU Hurd	b8	BSDI swap
4	DOS 16 - bit FAT < 32M	64	Novell	c7	Syrinx
5	Extended	75	PC/IX	db	CP/M
6	DOS 16 - bit FAT >= 32M	80	Old MINIX	e1	DOS access
7	OS/2 HPFS	81	Linux/MINIX	e3	DOS R/O
8	AIX	82	Linux swap	f2	DOS secondary
9	AIX bootable	83	Linux native	ff	BBT
a	OS/2 boot manager	93	Amoeba		

6.7.4 Dijeljenje hard diska na particije

Postoji mnogo programa za kreiranje i uklanjanje particija. Većina operativnih sistema posjeduje sopstveni i može biti dobra ideja koristiti upravo taj, za svaki slučaj, ako radi nešto neobično što drugi programi ne mogu. Mnogi od ovih programa, uključujući i onaj koji dolazi uz **Linux** se naziva **fdisk**. Detalje korištenja **fdisk** programa pogledajte u odgovarajućoj **man** stranici. **cfdisk** program je sličan **fdisk**-u, ali ima ugodniji korisnički interfejs.

Prilikom korištenja **IDE** diskova⁸, *boot* particija mora se kompletna nalaziti unutar prva 1024 cilindra. Razlog tome je što se ovaj disk koristi od strane **BIOS**-a tokom *boot*-a (prije nego što je sistem *protected* modu), a **BIOS** ne može prepoznati više od 1024 cilindra. Rade i particije koje su prelaze ovu granicu, ali samo su datoteke koje učitava **BIOS** sigurno unutar 1024 cilindra. Budući da je to teško ostvariti, jako je loša ideja to raditi. Nikada neznate kada će *upgrade* kernela ili defragmentacija diska dovesti do neupotrebljivog sistema. Zbog svega toga, osigurajte da je vaša *boot* particija stala čitava unutar prva 1024 cilindra diska. [19]

Neke novije verzije **BIOS**-a i **IDE** diskova mogu rukovati sa diskovima koji imaju više od 1024 cilindra. Ukoliko posjedujete takav sistem, zaboravite na ovaj problem, ukoliko niste sigurni igrajte po pravilima.

Svaka particija bi trebala imati paran broj sektora, budući da **Linux** datotečni sistemi koriste datotečne sisteme sa veličinom bloka od 1K, tj. dva sektora. Neparan broj sektora znači da zadnj

⁷op. p. Ovde svakako spadaju i Windows NT 4.0 čiji je alat *Disk Administrator* prije upotrebe označi svaku particiju, ali, ovaj, poslije te "bezopasne" operacije drugi programi uopće ne prepoznaju tip particije.

⁸op. p. Ovo se odnosi samo na **PC**. Druge arhitekture imaju drugačija ograničenja (ili uopšte nemaju **BIOS**)

ostaje neupotrebljen. Ovo ne urokuje probleme, ali je ružno i neke verzije **fdisk**-a će prikazati prikladno upozorenje.

Mijenjanje veličine particije obično prvo zahtijeva *backup* particije (ili cijelog diska, za svaki slučaj), brisanje particije, kreiranje nove particije, i onda snimanje spašenih podataka na novu particiju. Ukoliko raste veličina particije možda ćete morati podesiti i veličine drugih susjednih particija na disku.

Pošto je mijenjanje veličine particije bolan posao, bolje je pravilno podesiti particije odmah na početku ili imati efekivan i lako upotrebljiv *backup* sistem. Ako instalirate sa medija koji ne zahtijeva mnogo ljudskih intervencija (npr. CD-ROM-a, za razliku od *floppy* disketa), na početku je često olakšano "igranje" sa različitim konfiguracijama. Budući da nemate podataka koji zahtijevaju *backup* i restauraciju, nije posebno teško više puta modificirati veličine particija.

Postoji **MS-DOS** program, nazvan **fips** [20] koji mijenja veličinu **MS-DOS** particije bez potrebe za ponovnim kreiranjem datotečnog sistema (tj. formatiranjem, kako se to naziva u DOS-ovskom žargonu).

6.7.5 Datoteke uređaja i particije

Svaka particija i proširena particija ima svoju datoteku uređaja. Konvencija po kojoj se imenuju ove datoteke je da je broj particije dodan na ime diska, gdje su brojevima 1-4 označene primarne particije (bez obzira na to koliko primarnih particija postoji), a brojevima 5 ili više logičke particije (bez obzira na kojoj primarnoj particije se nalaze). Npr. */dev/hda1* je prva primarna particija prvog **IDE** diska, */dev/sdb7* je treća proširena particija na drugom **SCSI** hard disku.

6.8 Datotečni sistemi

A

6.8.1 Šta su to datotečni sistemi?

A

6.8.2 Mnoštvo datotečnih sistema

A

6.8.3 Koje datotečne sisteme treba koristiti?

A

6.8.4 Kreiranje datotečnog sistema

A

6.8.5 Montiranje i demontiranje

A

6.8.6 Provjera integriteta datotečnog sistema sa *fsck*

A

6.8.7 Provjera ispravnosti diska sa *badblocks*

A

6.8.8 Fragmentacija

A

6.8.9 Drugi alati za sve datotečne sisteme

A

6.8.10 Drugi alati za *ext2* datotečni sistem

A

6.9 Diskovi bez datotečnog sistema

A

6.10 Dodjela prostora na disku

A

6.10.1 Šeme za kreiranje particija

A

6.10.2 Zahtjevi za prostorom na disku

A

6.10.3 Primjeri dodjele prostora na hard diskovima

A

6.10.4 Dodavanje prostora na disku za Linux

A

6.10.5 Savjeti za štednju prostora na disku

A

Poglavlje 7

Upravljanje memorijom

7.1 Šta je virtualna memorija?

A

7.2 Kreiranje *swap* prostora

A

7.3 Korištenje *swap* prostora

A

7.4 Dijeljenje *swap* prostora sa drugim operativnim sistemima

A

7.5 Dodjela *swap* prostora

A

7.6 Međuspremnik (*buffer cache*)

A

Poglavlje 8

Pokretanje i gašenje

8.1 Pregled

A

8.2 *Boot* proces izbliza

A

8.3 Više o gašenju

A

8.4 Restart

A

8.5 Jednokorisnički mod

A

8.6 *Boot* diskete za podizanje sistema u poslije katastrofa

A

Poglavlje 9

init

9.1 *init* je prvi

A

9.2 Podešavanje *init*-a za pokretanje *getty*-a: datoteka */etc/inittab*

A

9.3 Izvršni nivoi (*run levels*)

A

9.4 Specijalne postavke u */etc/inittab*

A

9.5 Pokretanje sistema u jednokorisničkom modu

A

Poglavlje 10

Prijavljivanje i odjavljivanje

10.1 Prijave preko terminala

A

10.2 Prijave preko mreže

A

10.3 Šta to radi *login*

A

10.4 *X* i *xdm*

A

10.5 Kontrola pristupa

A

10.6 Pokretanje *shell*-a

A

Poglavlje 11

Upravljanje korisničkim računima (*account*-ima)

11.1 Šta je to *account*

A

11.2 Dodavanje novog korisnika

A

11.2.1 */etc/passwd* i druge datoteke sa informacijama

A

11.2.2 Odabir numeričkih identifikacija za korisnike i grupe korisnika

A

11.2.3 Inicijalne postavke: */etc/skel/*

A

11.2.4 "Ručno" kreiranje korisnika

A

11.3 Izmjene atributa korisnika

A

11.4 Uklanjanje korisnika

A

11.5 Privremeno onemogućavanje korisnika

A

Poglavlje 12

Backup

12.1 Značaj *backupa*

A

12.2 Odabir medija za *backup*

A

12.3 Odabir alata

A

12.4 Jednostavni *backup*

A

12.4.1 Korištenje komande *tar* za *backup*

A

12.4.2 Restauracija datoteka sa *tar*

A

12.5 *Backupi* u više nivoa

A

12.6 Šta zaslužuje *backup*

A

12.7 Kompresovani *backup*-i

A

Poglavlje 13

Ukorak s vremenom

13.1 Vremenske zone

A

13.2 Hardverski i softverski sat

A

13.3 Prikazivanje i podešavanje vremena

A

13.4 Kada je sat netačan

A

Poglavlje 14

Traženje pomoći

14.1 *News* grupe i *mailing* liste

A

14.1.1 Nalaženje pravog foruma

A

14.1.2 Prije oglašavanja

A

14.1.3 Kako napisati oglas (*post*)

A

14.1.4 Format oglasa

A

14.1.5 ??? Follow Up ???

A

14.1.6 Dodatne informacije

A

14.2 IRC

A

14.2.1 Boje

A

14.2.2 Budite ljubazni

A

14.2.3 Tipkajte ispravno, na engleskom jeziku

A

14.2.4 Skeniranje portova

A

14.2.5 Ograničite se na kanal

A

14.2.6 Ne skrećite s teme

A

14.2.7 CTCPs

A

14.2.8 Hacking, cracking, phreaking, warezing

A

14.2.9 Suma sumarum

A

14.2.10 Daljnje čitanje

A

Poglavlje 15

Dodatak A: GNU Free Documentation License

Op. p. Nema potrebe ovu prevoditi licencu. U slučaju prijevoda ovu licence verzija na engleskom se MORA zadržati i ona ima prednost nad prijevodom u slučaju bilo kakvog spora.

15.1 PREAMBLE

The purpose of this License is to make a manual, textbook, or other written document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondly, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

15.2 APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you".

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (For example, if the Document is in part a textbook of mathematics, a Secondary Section

may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, whose contents can be viewed and edited directly and straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup has been designed to thwart or discourage subsequent modification by readers is not Transparent. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, \LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML designed for human modification. Opaque formats include PostScript, PDF, proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

15.3 VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

15.4 COPYING IN QUANTITY

If you publish printed copies of the Document numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front

cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a publicly-accessible computer-network location containing a complete Transparent copy of the Document, free of added material, which the general network-using public has access to download anonymously at no charge using public-standard network protocols. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

15.5 MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

- A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.
- B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has less than five).
- C. State on the Title page the name of the publisher of the Modified Version, as the publisher.
- D. Preserve all the copyright notices of the Document.
- E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.

- F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.
- G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.
- H. Include an unaltered copy of this License.
- I. Preserve the section entitled "History", and its title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.
- J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.
- K. In any section entitled "Acknowledgements" or "Dedications", preserve the section's title, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.
- L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.
- M. Delete any section entitled "Endorsements". Such a section may not be included in the Modified Version.
- N. Do not retitle any existing section as "Endorsements" or to conflict in title with any Invariant Section.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties—for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not

add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

15.6 COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections entitled "History" in the various original documents, forming one section entitled "History"; likewise combine any sections entitled "Acknowledgements", and any sections entitled "Dedications". You must delete all sections entitled "Endorsements."

15.7 COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

15.8 AGGREGATIONS WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, does not as a whole count as a Modified Version of the Document, provided no compilation copyright is claimed for the compilation. Such a compilation is called an "aggregate", and this License does not apply to the other self-contained works thus compiled with the Document, on account of their being thus compiled, if they are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one quarter of the entire aggregate, the Document's Cover Texts may be

placed on covers that surround only the Document within the aggregate. Otherwise they must appear on covers around the whole aggregate.

15.9 TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License provided that you also include the original English version of this License. In case of a disagreement between the translation and the original English version of this License, the original English version will prevail.

15.10 TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided for under this License. Any other attempt to copy, modify, sublicense or distribute the Document is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

15.11 FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See <http://www.gnu.org/copyleft/>.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation.

15.12 How to use this license for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

Copyright (c) YEAR YOUR NAME. Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.1 or any later version published by the Free Software Foundation; with the Invariant Sections being LIST THEIR TITLES, with the Front-Cover Texts being LIST, and with the Back-Cover Texts being LIST. A copy of the license is included in the section entitled "GNU Free Documentation License".

If you have no Invariant Sections, write "with no Invariant Sections" instead of saying which ones are invariant. If you have no Front-Cover Texts, write "no Front-Cover Texts" instead of "Front-Cover Texts being LIST"; likewise for Back-Cover Texts. If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.

Poglavlje 16

GLOSSARY

”The Librarian of the Unseen University had unilaterally decided to aid comprehension by producing an Orang-utan/Human Dictionary. He’d been working on it for three months. It wasn’t easy. He’d got as far as Oook.” (Terry Pratchett, Men At Arms”)

Ovo je kratka lista definicija riječi za koncepte koji se odnose na **Linux** i administraciju sistema.

CMOS

account

application program

bad block

bad sector

boot sector

booting

bootstrap loader

cylinder

daemon

daylight saving time

disk controller

emergency boot floppy

filesystem

formatting

fragmented

full backup

geometry

high level formatting

incremental backups

inode

kernel

local time

logical partition

low level formatting

mail transfer agent

mail user agent

master boot record

network file system

operating system

partition

password file

platters

power on self test

print queue

read-write head

root filesystem

run level

sectors

shadow passwords

single user mode

spool

system call

swap space

system program.