# POGLAVLJE

# Što je to Linux

(Jack Tackett)

# U ovom poglavlju

Što je Linux? Zbog čega treba koristiti Linux? Linux distribucije Prednosti uporabe Linuxa Nedostaci uporabe Linuxa Komercijalna strana Linuxa Kratka povijest Linuxa Tko je vlasnik Linuxa? Da biste shvatili što je to Linux, najprije trebate znati odgovor na pitanje "Što je to UNIX?" zbog toga jer je Linux projekt koji je nastao s namjerom stvaranja inačice UNIX-a koja bi trebala raditi na strojevima s Intelovim procesorom, odnosno računalima koja se uobičajeno nazivaju IBMuskladiva osobna računala.

UNIX se danas koristi kao najpopularniji operativni sustav na računalima koja se rabe u znanstvene svrhe kao i na nekim radnim stanicama. U ovom ćete poglavlju saznati koji su to razlozi zbog kojih biste mogli odabrati Linux, a ne neki drugi operativni sustav kao što je primjerice MS-DOS, Windows 95/98, Windows NT ili OS/2.

# Što je Linux?

Linux predstavlja operativni sustav za nekoliko vrsta računala, ali u prvom redu se koristi na osobnim računalima. Sustav su razvijali i gradili programeri iz čitavog svijeta. Cilj je bio klonirati UNIX i to bez bilo kakvog komercijalnog softvera koji se koristi u svijetu.

U osnovi je Linux nastao kao plod hobija Linusa Torvaldsa tijekom njegova studija na helsinškom sveučilištu. On je želio stvoriti zamjenu za operacijski sustav Minix (koji je nalik UNIX-u ali se može upotrebljavati na osobnim računalima).

BILJEŠKA U nastavku poglavlja ćemo vam objasniti mnoge od navedenih izraza, stoga nemojte brinuti ako neki od njih trenutno ne razumijete.

Linux je u osnovi klonirani UNIX, što znači da s Linuxom možete iskoristiti brojne prednosti UNIX-a. Linux je u pravom smislu višezadaćni sustav, što znači da istovremeno možete pokrenuti više programa i oni će se neprestano odvijati. Drugi operativni sustavi, kao što je primjerice Microsoft Windows 3.1, omogućavaju pokretanje više programa odjednom, ali čim pređete s jednog programa na drugi, prvi program se obično zaustavi. Windowsi 95 i Windowsi NT su u tom pogledu vrlo nalik Linuxu jer omogućavaju višezadaćni rad. Linux vam omogućava da istovremeno prenosite datoteku, ispisujete dokument, kopirate sadržaj diskete, koristite CD-ROM i igrate igricu.

Linux u potpunosti podržava višekorisnički rad što znači da se istovremeno više osoba može prijaviti na sustav i koristiti Linux. Iako vam se ova značajka Linuxa kod kuće vjerojatno neće učiniti osobito korisnom, ona mnogim korisnicima, primjerice na sveučilištu, nudi istovremeni pristup zajedničkim izvorima informacija, te isključuje potrebu za većim brojem skupih računala. Čak i kod kuće vam može biti korisna mogućnost odvojenog prijavljivanja na tzv. *virtualne terminale*. Osim toga, pomoću Linuxa i nekoliko modema možete stvoriti vlastiti mrežni servis.

Vidi "Upravljanje korisnicima", stranica 107

Linux je besplatan, odnosno gotovo besplatan. U osnovi, učitavanjem jedne od distribucija Linuxa s Interneta dobit ćete potpune operativne sustave s mnoštvom dodatnih programa. Razne distiribucije nude sve što je potrebno za postavljanje Linuxa na vaše računalo i njegovo pokretanje. S Web lokacije Sun Microsystemsa (odnedavno) možete učitati i potpuno besplatan uredski paket programa kojim Sun pokušava uzeti dio tržišta Microsoftovom Officeu.

Linux vam nudi i mogućnost učenja jer se pred vama nalazi kompletan radni operativni

sustav, uključujući i izvorni kod kojim se možete pozabaviti. Traženje trikova u Linuxu je nešto što inače ne možete raditi u običnom UNIX okruženju, a zasigurno takvo što ne možete pokušati s bilo kojim komercijalnim operativnim sustavom jer niti jedan proizvođač ne želi na svjetlo dana iznijeti izvorni kod.

Na kraju, Linux vam daje mogućnost da obnovite – ili možda prvi puta doživite – doživljaje iz ranih početaka razvoja osobnih računala. Sredinom 70-tih godina računala su bila privilegija velikih organizacija, kao što su vlade pojedinih država, velike tvrtke i sveučilišta. Obični ljudi nisu imali pristup takvim čudima. Najprije su osobna računala postala privilegija hakera, odnosno računalnih zanesenjaka koji su prodrli do ranih sustava jer su ti sustavi bili slabo iskorišteni. No, kako su hakeri napredovali u svom istraživanju, a rasle mogućnosti osobnih računala, tako su i ona izbila u prvi plan.

Izraz "haker" je, nažalost, u današnjem tumačenju poprimio negativan smisao. U nastavku ovog BILJEŠKA poglavlja, u odlomku "Hakeri" saznajte nešto više o hakerima.

Isto se danas odnosi i na sustavski softver (odnosno na operativne sustave). Linux predstavlja prijelomnicu u pogledu sustava koje stvaraju ogromne računalne organizacije koje kreativnost i napredak podređuju kretanjima na tržištu softvera.

## Zbog čega treba koristiti Linux?

Linux ćete koristiti jer je to danas jedini operacijski sustav koji u potpunosti podržava višezadaćni i višekorisnički rad na IBM-uskladivim osobnim računalima. Niti jedan drugi operativni sustav ne nudi vam te mogućnosti u toliko moćnom obliku kao Linux. Osim toga, koristeći Linux sigurno nećete postati žrtvom raznih sumnjivih trgovaca. Također nećete svakih par godina morati nadograđivati programe i na to trošiti znatne sume novca. Mnoge Linux aplikacije možete besplatno naći na Internetu, kao i izvorni kod samog Linuxa.

S obzirom da imate mogućnost pristupa izvornom kodu, možete mijenjati i operativni sustav te ga prilagođivati svojim potrebama, što nije slučaj niti s jednim komercijalnim operativnim sustavom kao što su Windowsi NT, Windowsi 95, MS-DOS i OS/2.

Činjenica da ste se riješili trgovaca softverom može biti i negativna strana kod uporabe Linuxa. Kako niti jedna komercijalna kuća ne stoji iza Linuxa, pomoć nećete moći dobiti tako što ćete nazvati određeni telefonski broj. Linux vam može stvarati probleme i tako što neće raditi sa svim mogućim hardverskim komponentama. Postoji i potencijalna opasnost da se oštete ili obrišu datoteke s podacima koje se nalaze na disku, jer se Linux neprestano mijenja, a osim toga ne prolazi kroz stroge kontrole prije objave određene inačice.

Linux nije igračka već sustav stvoren da bi korisnicima dao osjećaj sudjelovanja u novom projektu, baš kao što je to bilo na početku razvoja osobnih računala. Unatoč svemu, Linux je stabilan na mnogim sustavima i kao takav predstavlja jeftinu prigodu za učenje i korištenje jednog od najpopularnijih operativnih sustava današnjice – UNIX-a. Mnoge informatičke kuće i izdavači CD-ova, kao što su Red Hat ili Caldera, sada podržavaju

operativni sustav Linux. Ovaj sustav predstavlja zamjenu drugim UNIX sustavima i može se upotrijebiti umjesto tih, ponekad skupih sustava. Ukoliko vaš program radi na UNIX

sustavima, možda ćete htjeti imati sličan sustav i kod kuće. Jeste li vi možda administrator UNIX sustava u svojoj tvrtci? Ako jeste, tada neke od svojih obaveza možete obaviti i kod kuće, koristeći Linux. Znate li uopće što je to UNIX? Ako ne, onda vam Linux može pred-stavljati gotovo besplatan uvod u jedan od najpopularnijih sustava – UNIX.

Linux također nudi lagani pristup Internetu i drugim informatičkim prometnicama.

# Linux distribucije

Linux distribuiraju mnoge organizacije, a svaka od njih nudi jedinstvenu zbirku programa zajedno s jezgrom koju čine datoteke određenog izdanja. Knjiga se uglavnom temelji na distribucijama Linuxa s inačicom jezgre 2.0.34, dok je u međuvremenu objavljena i inačica jezgre 2.2. Ta distribucija može sadržavati eksperimentalne jezgre s pogonskim programima pojedinih hardverskih komponenti. Pod Red Hat distribucijom, jezgre su dio sustava Red Hat Package Management (RPM) i instalirane su kao dio samog operacijskog sustava. Calderina Open Linux distribucija slijedi iste sheme jer se zasniva na Red Hat distribuciji.

Na svu sreću, kupnjom ove knjige još uvijek imate mogućnost donijeti odluku koju ćete distribuciju koristiti. Knjiga se temelji i detaljnije opisuje Red Hat distribuciju i Calderinu distribuciju, ali na Internetu su vam na raspolaganju još neke distribucije:

- MCC Interim Linux
- TAMU Linux
- LST
- SLS
- Debian Linux
- Yggdrasil Plug-and-Play Linux CD-ROM i Linux Bible
- Trans-Ameritech Linux plus BSD CD-ROM
- The Linux Quarterly CD-ROM
- Caldera
- Red Hat (komercijalna inačica Red Hata sadrži komercijalni X poslužitelj nazvan Metro X).

Distribucija HOWTO nudi popis Linux distribucija. U nastavku ćete ovog poglavlja naučiti kako pristupiti raznim HOWTO-ovima koji se nalaze u svakom izdanju Linuxa.

# Prednosti uporabe Linuxa

Uporaba Linuxa ima mnoge prednosti. U odnosu na mnoge operativne sustave koji se danas koriste, Linux je najpopularniji besplatni sustav. Za IBM-uskladiva osobna računala Linux nudi cjelovit sustav s ugrađenim višekorisničkim i višezadaćnim mogućnostima pomoću kojih se u potpunosti može iskoristiti procesor (386 ili noviji) vašeg računala.

Linux se pojavljuje s potpuno implemetiranim mrežnim protokolom TCP/IP. Uz pomoć Linuxa možete ostvariti vezu na Internet i na njemu koristiti more informacija koje ta globalna mreža sadrži. Također nudi i cjelovit sustav za slanje i primanje e-pošte.

Osim toga, Linux ima grafičko korisničko sučelje (GUI), to jest XFree86, koje se zasniva na poznatom sustavu XWindows. XFree86 predstavlja cjelovitu implementaciju XWindowsa koja se može besplatno distribuirati zajedno s Linuxom. XFree86 sadrži poznate elemente grafičkog sučelja koje možete pronaći i u drugim komercijalnim platformama, kao što su Windowsi i OS/2.

Danas sve to možete dobiti s Linuxom i to praktički besplatno. Sav trošak predstavlja cijena kopiranja programa s Interneta ili putem e-pošte (što je moguće dobiti od nekoliko različitih proizvođača). Naravno, kod odgovarajućih prodavača možete kupiti (cijena nije visoka) cjelovit Linux sustav na CD-ROM-ovima.

## Prenosivost sustava

U nikad dorečenoj priči o standardizaciji, mnoge organizacije se vraćaju počecima razvoja operativnih sustava. U tom slučaju UNIX nije mogao ostati nezapažen. Sada postoji mogućnost standardiziranja mnogih inačica UNIX-a. U idućem ćete odlomku više naučiti o tome koje su inačice dosad razvijene.

Bilo je potrebno učiniti znatan napor prilikom prikupljanja i sastavljanja svih inačica UNIX-a u jedinstveni operativni sustav. Kao i kod mnogih uzvišenih nastojanja, tako je i ovo moglo propasti jer pojedini programeri nisu htjeli žrtvovati dio svojih prijašnjih inačica (tužno, ali istini-to, mnogi programeri se i danas tako ponašaju).

Ipak, postojanje različitih inačica UNIX-a nije nužno i razlog za uzbunu. Unatoč tim razlikama, sve inačice su bitno nadmoćnije od svih drugih operativnih sustava jer sve one sadrže iste elemente koji su prethodno opisani.

*Prenosivost* je mogućnost prenošenja operativnog sustava s jedne platforme na drugu, a da pri tome on ostane u istom obliku. UNIX je u osnovi prenosivi operativni sustav. Izvorno, UNIX može raditi na samo jednoj platformi – mini računalu DEC PDP-7. Danas mnoge inačice UNIX-a mogu raditi u bilo kojem okruženju i na bilo kojoj platformi, od prijenosnih do velikih računala.

Prenosivost u osnovi znači da različita računala mogu pokrenuti UNIX i međusobno točno i efikasno komunicirati. Takvi sustavi mogu izmjenjivati podatke bez posebnih, skupih komunikacijskih dodataka. To ne može omogućiti niti jedan postojeći operacijski sustav.

## Programi

Iako i korištenje samih operativnih sustava ponekad može biti zabavno, to ipak nije razlog zbog kojeg većina korisnika rabi računalo. Većina ljudi želi ubrzati rad koristeći računalo. Linux danas podržava tisuće programa, uključujući proračunske tablice, baze podataka, pisanje dokumenata, aplikacije koje su nastale u raznim programskim jezicima i telekomunikacijske pakete pomoću kojih se možete umrežavati. Uz Linux dolaze i brojne igrice,

i to one tekstualne i grafičke. Želite li se odmoriti od naporna svakodnevnog rada, Linux će vam ponuditi nekoliko minuta (sati) odmora.

## Prednosti za profesionalne korisnike

Ako se profesionalno služite računalom, Linux nudi brojne razvojne alate, među kojima su kompilatori (prevoditelji) za mnoge programske jezike, kao što su C, C++ i Smalltalk. Ako vas ne zanimaju ti programski jezici, Linux vam nudi alate kao što su Flex i Bison, koje možete upotrijebiti za stvaranje vlastitih programskih jezika. Te ćete alate naći na Internetu, no njihove će vas komercijalne inačice koštati koju stotinu dolara. Želite li naučiti neki od prethodno spomenutih programskih jezika, ali da pri tome ne potrošite stotine dolara za neki kompilator, tada će vam svakako dobro doći Linux i njegovi razvojni alati.

Linux vam također omogućava komuniciranje s uredskim računalima u vašoj tvrtci. Ako ste i sami administrator UNIX sustava, Linux vam može pomoći da dio svog posla obavite od kuće. Iako je rad kod kuće za sada samo vizija, možda ćete jednog dana moći upotrijebiti Linux za obavljanje posla u svom domu, a zatim ćete u svoju tvrtku povremeno odlaziti na sastanke.

U informatičkoj industriji često se koriste riječi *otvoreni sustavi i interoperativnost*, a obje se odnose na mogućnost mnogih sustava da međusobno komuniciraju. Većina otvorenih sustava zahtijeva POSIX (Portable Operating System Interface), odnosno određenu formu UNIX-a. Danas Linux udovoljava tim normama. U osnovi je Linux načinjen za prijenos izvornog koda, tako da ako koristite program koji pokrećete u okružju određene inačice UNIX-a, možete isti taj program prenijeti na sustav koji radi pod Linuxom.

Određene korporacije inzistiraju na takvim otvorenim sustavima kako ne bi bile vezane uz samo jednog proizvođača softvera. Sjetite se stare poruke: "Ne stavljate sva svoja jaja u istu košaru". Danas korporacije ne žele da njihove sustave kontrolira samo jedna tvrtka jer u tom slučaju ona može određivati kako će izgledati softver i hardver koji se koristi u određenoj korporaciji. Ako vam ne odgovara smjer u kojem se razvija jedna takva tvrtka, tada ćete naići na ozbiljne probleme jer ćete morati prihvaćati njezine odluke bez obzira odgovaraju li vam ili ne. S UNIX/Linux operativnim sustavima sami upravljate svojom budućnošću. Ako operativni sustav ne odgovara vašim potrebama, tada potražite nekog od brojnih konzultanata koji će vam unijeti potrebne prepravke. To je moguće jer vam je na raspolaganju izvorni kod operativnog sustava.

## Obrazovanje

Studenti, uočite da Linux sadrži programe za pisanje vaših zahtjeva i programe za provjeru pravopisa i gramatike koji će provjeriti te zahtjeve. Pomoću Linuxa ćete se moći prijaviti na

vašu školsku mrežu. Naravno, pristupite li Internetu, imat ćete trenutni pristup neograničenom svijetu informacija. Također ćete moći konzultirati tisuće svjetskih stručnjaka za različita područja koji su u stanju odgovoriti na vaša pitanja. Linux može biti koristan čak i ako vam računala nisu glavna briga.

Linux nudi takve prednosti za tako malo novaca zbog duha i načela zajednice koja ga je stvorila i koja ga nadograđuje. Linux predstavlja jedan veliki eksperiment. Stotine računalnih hakera iz čitavog svijeta sudjeluje u njegovu razvoju. Linus Torvalds je prvi stvorio sustav koji je kasnije postao Linux, a potom svoje djelo prepustio svijetu.

## Hakeri

U osnovi, Linux je sustav koji su stvorili hakeri i koji oni koriste. Današnja definicija pojma haker ima negativni smisao, no računalni hakeri nisu izvorno kriminalci. Definicija hakera određuje kako netko pristupa bilo kojoj životnoj aktivnosti, a ne odnosi se samo na računala. Hakeri osjećaju određenu obavezu i uzbuđenje dok provaljuju sustav, što u osnovi znači da žele saznati sve što mogu o tom sustavu, dovesti sustav do točke raspada, a da pri tome imaju mogućnost popraviti određeni sustav ako se on raspadne.

Hakeri u osnovi žele znati kako radi sustav koji ih zanima. Većinu ne zanima novac ili neka vrsta bolesne osvete, iako su neki hakeri prešli zamišljenu liniju i postali crackeri. Računalni hakeri se osjećaju uvrijeđeno kada ih se uspoređuje s tim vandalima i kriminalcima. Na svu sreću, Linux vam daje osjećaj što to znači biti haker, a u idealnom slučaju nećete poželjeti biti cracker.

Ako ste jednostavno znatiželjna osoba i želite što više naučiti o UNIX-u, tada će vam Linux doista dobro doći, jer se radi o potpunoj inačici UNIX-a kojoj imate besplatan i neograničen pristup. Tako nešto ćete rijetko naći u stvarnom svijetu. Većini korisnika UNIX-a su dodijeljeni računi na UNIX-ovim strojevima koji im daju ograničena prava i privilegije, tako da u tom slučaju običan korisnik ne može koristiti ili provjeravati određene UNIX/Linux naredbe. Ali, na takav način nećete moći naučiti sve o UNIX-u. S Linuxom imate cjelovit pristup i možete učiniti sve što poželite. Naravno, s tako velikom

privilegijom stiže i velika odgovornost: Morat ćete naučiti kako se upravlja stvarnim UNIX sustavom, što i samo po sebi može biti zabavno.

## Nedostaci uporabe Linuxa

Vjerojatno najveći nedostatak uporabe Linuxa leži u činjenici da niti jedna korporacija nije uključena u njegov razvoj. Ukoliko nešto krene po zlu ili ako naiđete na problem, nećete moći upotrijebiti neki od besplatnih telefonskih brojeva kako biste zatražili pomoć. Kada malo bolje o tome razmislite, možete li na sličnim brojevima doista dobiti adekvatnu pomoć i za postojeće komercijalne sustave? Koliko ste se puta nekome obratili tražeći tehničku pomoć i dobili odgovor na svoja pitanja? Koliko ste puta zamoljeni da svoje pitanje pošaljete poštom kako biste dobili na njega odgovor? Pa dobro, kod Linuxa doista ne postoji telefonski broj za tehničku potporu, ali postoje tisuće korisnika koji mogu odgovoriti na vaša pitanja (pogledajte Dodatak A "Izvori informacija" i potražite mjesta na kojima možete dobiti pomoć).

## Nedostatak tehničke potpore

Nema sumnje da nedostatak tehničke potpore može biti problem Linuxa. Isto se odnosi i na Linux programe. Iako postoji nekoliko komercijalnih programa za Linux, većinu programa su razvile male grupe programera i zatim objavili u javnosti. Ipak, mnogi programeri nude potporu i odgovaraju na pitanja.

**BILJEŠKA** Mnoge komercijalne tvrtke sada stvaraju Linux programe koje mogu prodavati. Da bi se korisnici služili njihovim aplikacijama, te tvrtke obično nude i besplatnu kopiju Linux distribucije u paketu s njihovim proizvodom (ta je distribucija usklađena s njihovom inačicom proizvoda).

## Problemi s hardverom

Drugi nedostatak je taj što Linux teško možete instalirati na svim hardverskim platformama. Za razliku od komercijalnih programa, kod kojih se program mjesecima stvara i provjerava uz različiti hardver, tvorci Linuxa su raštrkani po čitavom svijetu i ne postoji mogućnost osiguranja kvalitete programa. Tvorci programa ga objavljuju u trenutku kada im se to učini pogodnim. Također, hardver koji podržava Linux ovisi o hardveru koji koristi svaki pojedini programer dok piše dio programskog koda. Zbog toga Linux ne radi sa svim hardverskim komponentama koje se danas koriste u osobnim računalima.

#### **UPOZORENJE**

Ako vaše računalo ne sadrži hardver koji Linux podržava, imat ćete problema prilikom instalacije i pokretanja operativnog sustava. U 3. poglavlju "Instalacija Red Hata" i 4. poglavlju, "Instalacija OpenLinux Litea", pronaći ćete detaljne upute za hardver koji vam je nužan kako biste koristili Linux.

Ako koristite hardver koji Linux podržava, vjerojatno nećete imati problema prilikom instalacije i uporabe Linuxa. A ako nemate potreban hardver... pa dobro, tvorci Linuxa očekuju da to sami riješite. Ipak se radi o hakerskom operativnom sustavu.

## Nemogućnost korištenja postojećeg softvera

Još jedan nedostatak je taj da postojeće aplikacije za operativne sustave kao što su DOS i OS/2 u većini slučajeva neće raditi pod Linuxom. Na svu sreću, ti se drugi sustavi mogu koristiti zajedno s Linuxom tako da, iako ne možete oba operativna sustava istovremeno pokrenuti, imate mogućnost napustiti Linux i podići drugi operativni sustav da biste koristili svoje aplikacije.

Trenutno se radi na Linux emulatorima koji pokreću DOS i Windows programe. Primjer takvog emulatora je Executor koji pod Linuxom pokreće programe za Macintosh računala. Iako se emulatori DOS-a duže razvijaju od emulatora za Windows i Macintosh računala, svi oni su još uvijek u početnoj fazi. Ipak, u bliskoj budućnosti će Linux moći pokretati programe za Mac, DOS i Windows računala.

Osim toga, organizacija Caldera je objavila sučelje WABI (Windows Applications Binary Interface). WABI omogućuje programima pisanim za Windowse 3.1 da rade pod XWindowsima na Linux računalima. Za razliku od mnogih Linux aplikacija, Caldera prodaje ovaj proizvod zajedno s još nekoliko Linux aplikacija. Caldera besplatno nudi Red Hat distribuciju Linuxa koja služi za pokretanje programa koji su u prodaji. Osim toga, radi se na izradi posebne inačice DOS-a za Linux (DR DOS).

Da biste instalirali Linux, obično ćete trebati preformatirati svoj tvrdi disk (iako to nije uvijek neophodno). Pri tome ćete trebati obrisati particije na disku, zajedno s postojećim programima i podacima. Za sada ne postoji siguran način instaliranja Linuxa bez istovremenog preformatiranja diska. Namjeravate li instalirati Linux, tada biste svakako trebali napraviti sigurnosnu kopiju podataka s diska (najsigurnije je načiniti dvije ili tri kopije). Osim toga, moguće je da nećete imati dovoljno prostora na disku da biste instalirali Linux i sačuvali preostali softver. U tom slučaju sami odlučujete što ćete obrisati a što ne. Bez obzira na sve, morate pohraniti podatke, preformatirati disk, zatim obnoviti stari softver i tek potom instalirati Linux. Sve to skupa će vam oduzeti puno vremena, a moguće su i neke pogreške prilikom instalacije.

**BILJEŠKA** Postoji i alternativa preformatiranju tvrdog diska. Prostor možete podijeliti između Linuxa i DOS-a ili možete upotrijebiti program za formatiranje diska bez brisanja datoteka. Iako se te alternative mogu koristiti, još uvijek ostaje opasnost od gubitka podataka prilikom instalacije sustava. No, nakon preformatiranja možda ćete poboljšati karakteristike svog računala, te imati bolju kontrolu nad prostorom na disku koji koristi Linux.

O programima koje namjeravate instalirati ovisit će i veličina prostora na disku koji vam je potreban za pokretanje Linuxa. Na disku na kojem namjeravate instalirati Linux trebali biste ostaviti barem 120 MB prostora, nakon što prenesete podatke i programe koje ste koristili pod drugim operativnim sustavima. Imate li 200 MB slobodnog prostora, to bi trebalo biti više nego dovoljno za kompletnu instalaciju Linuxa.

## Nedostatak iskustva

Napokon, ako već niste stručnjak za UNIX, morat ćete naučiti kako upravljati Linux sustavom. Za razliku od DOS-a, Windowsa i OS/2, Linuxom i UNIX-om treba upravljati. Onaj koji to čini se obično naziva *administrator sustava* i njegova je uloga da održava sustav. Administrator sustava je odgovoran za dodavanje i brisanje korisničkih računa, zaštitu sustava, instaliranje novog softvera, konfiguriranje sustava i otklanjanje nedostataka (koji se mogu javiti i na komercijalnim inačicama UNIX-a koji se svakodnevno koristi). S obzirom da UNIX ne radi besprijekorno čitavo vrijeme, administrator sustava ga mora održavati. Nudi vam se izuzetna prigoda da naučite kako postati administrator sustava na UNIX-u.

Vidi "Sustavi s centralnim upravljanjem", stranica 166.

## Prevladavanje nedostataka

Najprije ste možda pomislili da vas uporaba Linuxa ostavlja samima na svijetu, nudeći vam borbu za preživljavanje. To je samo djelomice točno jer je Linux nastao kao sustav hakera te su ga hakeri sami održavali i ispravljali nedostatke. No danas, uslijed porasta popularnosti Linuxa, postoji mnoštvo izvora pomoći. Tisuće stranica dokumentacije se nudi s većinom Linux distribucija. Te informacije možete pronaći u odgovarajućim direktorijima svake distribucije (obično su to direktoriji /DOCS ili /DOC). Osim toga, nekoliko časopisa se bavi Linuxom i u njima ćete naći mnoštvo mrežnih izvora informacija i korisnika koji su voljni odgovoriti na vaša pitanja. Ako radite za određenu tvrtku i želite uspostaviti poslovni odnos, danas imate i tu mogućnost. Nakon što instalirate Linux, naći ćete puno informacija o gotovo svakoj Linux naredbi i programu. Pogledajte Dodatak A, "Izvori informacija", i uvjerit ćete se da niste sami na svijetu.

## Nedostaci sve više iščezavaju

Iako još uvijek postoje svi nedostaci koji su spomenuti u prethodnim odlomcima, mnogi od njih polako iščezavaju s pojavom novih rješenja koje neke tvrtke ugrađuju u Linux. Dvije takve tvrtke su Red Hat i Caldera. Kao primarnu distribuciju koju opisujemo u knjizi odabrali smo Red Hat zbog jednostavnosti korištenja i instalacije sustava. Caldera također koristi Red Hat distribuciju za vezu s Linux programima. Obje distribucije nude mogućnost rada na mreži, slanje i primanje faks poruka i e-pošte kao i tehničku potporu za svoje proizvode i njihove inačice Linuxa.

# Komercijalna strana Linuxa

Linux nije samo "igračka" jer mnoge tvrtke koriste Linux kao jeftini Web poslužitelj svojih internih mreža. Linux se također koristi za različite mrežne aplikacije (kao što je DNS) kao usmjernik ili vatrozid (*firewall*). Osim toga, mnogi davatelji Internet usluga koriste Linux kao glavni operativni sustav.

Također postoje mnogi komercijalni programi za Linux koje možete isprobati. Druge organizacije, kao što su NASA i Digital Domain koriste Linux za različite slike kao što su slike planeta visoke razlučivosti (NASA) ili specijalne filmske efekte, primjerice u filmu Titanic (Digital Domain).

## Komercijalni programi za Red Hat distribuciju

Osim što je organizacija Red Hat objavila jednu od najpopularnijih distribucija Linuxa, tvrtka je proizvela i nekoliko komercijalnih programa. Također su i načinili programa za upravljanje Linux paketom (RPM) koji je objavljen tako da ga mogu koristiti i druge distribucije.

Zajedno s GPL inačicama Linuxa i RPM-om, tvrtka Red Hat nudi programsko okruženje nazvano Applixware koji sadrži program za obradu teksta, proračunsku tablicu, program za prikaz grafika, alat za rad s e-poštom i različite razvojne alate. Red Hat također nudi komercijalnu inačicu Motifa – grafičkog sučelja za Linux.

## Komercijalni programi Caldera distribucije

Caldera izvorno nudi mrežno orijentiranu distribuciju koja se zasniva na Red Hat distribuciji i Novellovoj tehnologiji (mnogi stručnjaci su i stigli iz Novella). Druga generacija proizvoda, pod imenom Caldera OpenLinux Base, predstavlja jeftini operativni sustav nalik UNIX-u. Taj sustav koristi Linux jezgru i OpenLinux distribuciju koju je razvila tvrtka Caldera. U distribuciju je uključeno grafičko korisničko sučelje pomoću kojeg se može upravljati sustavom i mrežnim resursima. Ponuđena je instalacija putem izbornika i to na više jezika. Caldera OpenLinux Base uključuje program-klijent za Internet, poslužitelj, komunikacijske protokole i servise. OpenLinux Base također uključuje komercijalni X poslužitelj organizacije MetroLink i kompletnu licencu Linux inačice programa Netscape Navigator.

Caldera također nudi i Corelov WordPerfect za Linux, kao i uredski paket sa svim poslovnim aplikacijama. Ove komercijalne programe, kao i desetke drugih, možete potražiti na CD-ROM-u Solutions koji je izdala tvrtke Caldera. Pomoću Netscapea možete pregledati katalog i zatim slijediti upute za naručivanje koje se nalaze na stranici Ordering.

#### NA WEBU



Calderin katalog potražite na adresi http://www.caldera.com/solutionscd.

Tvrtka Caldera je objavila Sunsoft WABI tehnologiju koja omogućuje krajnjim korisnicima pokretanje popularnih programa za Windowse 3.1 na Linux sustavima.

# Kratka povijest Linuxa

Povijest Linuxa je vezana uz povijest UNIX-a i programa nazvanog Minix. Minix je bio operativni sustav koji je napisao poznati i cijenjeni znanstvenik Andrew Tannebaum. Ovaj operativni sustav je postao popularan za razne PC platforme uključujući i MS-DOS računala. No, o Minixu će biti više riječi u nastavku. Najprije slijedi kratka povijest UNIX-a.

Iako je tvrtka AT&T stvorila operativni sustav UNIX, mnoge druge tvrtke i pojedinci su se trudili poboljšati osnovnu ideju. U idućim ćete odlomcima upoznati nekoliko najpoznatijih inačica koje se danas koriste.

## AT&T

Ken Thompson (programer tvrtke AT&T Bell Laboratories) i grupa ljudi koja radi pod njegovom upravom, razvili su operativni sustav koji je bio prilagodljiv i u potpunosti je odgovarao njihovim potrebama. Govori se da je Ken, koji je koristio operativni sustav MULTICS, dugo držao da je novi proizvod UNIX u osnovi šala njega i njegova razvojnog tima. MULTICS je bio višekorisnički operativni sustav, dok je naziv UNIX nastao od riječi *uni*, koja označava jedninu. Najzanimljivije u cijeloj priči je to što je MULTICS,

kao višekorisnički operativni sustav zapamtilo samo nekoliko korisnika, dok je UNIX postao industrijski standard za višekorisničke i višezedaćne operativne sustave.

## BSD

Berkeley Software Distribution (BSD) – sveučilište Berkeley u Kaliforniji je objavilo prvu inačicu UNIX-a, koja je nastala na osnovu AT&T, inačica 7, 1978. godine. Sustav BSD UNIX, kako se tada nazivao u industriji, je sadržavao poboljšanja koja su uveli programeri s Berkeleya, a koja su se odnosila na lakšu uporabu UNIX-a. Poboljšanja su korisničkog sučelja bila pokušaj da se UNIX približi običnim korisnicima, a ne samo naprednim programerima kojima se sviđala njegova prilagodljivost kojom je odgovarao na njihove programske zahtjeve. Iako BSD UNIX nije bio potpuno uskladiv sa sustavom AT&T, njegovi su stvoritelji postigli svoj cilj: dodali su značajke pomoću kojih su ga približili običnim korisnicima.

BSD je postao akademski standard za UNIX. Programeri koji su ga stvorili objavili su operativni sustav pod imenom BSD koji je mogao raditi na Intel platformi. I tu inačicu možete pronaći na Internetu i kod distributera softvera. Autori su također napisali nekoliko članaka u računalnom časopisu Dr. Dobb's Journal, koji se odnose na stvaranje i razvoj sustava BSD386 ili FreeBSD. Danas je BSDI, komercijalna inačica sustava FreeBSD, još jedan popularni operativni sustav nalik Linuxu.

## USL

UNIX System Laboratories (USL) predstavlja još jednu organizaciju koja je razvijala operativni sustav UNIX tijekom ranih 80-tih. Organizacija USL je objavila izvorni kôd svih inačica UNIX-a System V, ali one nisu bile prodavane u to vrijeme.

Zadnja inačica UNIX-a koju je objavila organizacija USL je bila UNIX System V Release 4.2 (SVR4.2), koja označava početak ulaska organizacije na tržište UNIX-a. U suradnji s Novellom stvorena je tvrtka Univel, a organizacija USL je objavila inačicu SVR4.2 pod nazivom UnixWare. Nakon što ju je preuzeo Novell, tvrtka se umjesto programskim kodom počela baviti proizvodnjom UnixWarea. Nakon toga je tvrtka Novell prodala svoju inačicu UNIX-a organizaciji Santa Cruz Operation (SCO).

Nedavno je organizacija SCO načinila besplatnu licencu za javno korištenje SCO UNIX-a (jedan korisnik). Za razliku od Linuxa, distribucija programa stoji 19 dolara. No, iako organizacija SCO nudi kopije svog operativnog sustava, ona ne nudi i izvorni kôd.

## XENIX, SunOS i AIX

Krajem 70-tih i početkom 80-tih godina, tijekom razvoja osobnih računala, tvrtka Microsoft je razvila svoju inačicu UNIX-a, sustav XENIX. Procesori osobnih računala počeli su sustizati one na postojećim miniračunalima. Pojavom Intelovog mikroprocesora 80386, brzo je postalo očigledno da XENIX, koji je razvijen isključivo za osobna računala, više nije neophodan. Tvrtke Microsoft i AT&T su ujedinile UNIX i XENIX u jedan operativni sustav nazvan System V/386 3.2., koji može raditi na praktički bilo kojoj hardverskoj konfiguraciji. Sustav XENIX i danas možete nabaviti od organizacije Santa Cruz Operation (SCO). Ova organizacija je sustav razvijala u suradnji s Microsoftom, trudeći se da promocijom XENIX-a na tržištu osobnih računala učine ovu inačicu UNIX-a jednom od komercijalno najuspješnijih.

Tvrtka Sun Microsystems je promocijom sustava SunOS i pripadnih radnih stanica učinila veliki iskorak na UNIX tržište. Tvrtka je objavila inačicu UNIX-a koja se bazira na BSD-u. Doista je zanimljivo da je AT&T-ova inačica SVR4 kompatibilna s BSD-om.

Tvrtka IBM je u područje UNIX-a ušla putem proizvoda nazvanog AIX (Advanced Interactive Executive). Iako AIX nije toliko poznat kao neke druge UNIX inačice, ovaj sustav radi dobro, tako da se bez problema zadržao na tržištu operativnih sustava.

## Linux

Linux je djelo studenta računalstva Linusa Torvaldsa. Linux je nastao 1991. godine kao posljedica Linusovog hobija. Linus je tada imao tek 23 godine. Nadao se da će stvoriti robusnu inačicu UNIX-a za Minix korisnike. Sustav Minix, kako je to prije spomenuto, predstavlja program koji je razvio profesor računalnih znanosti Andrew Tannebaum.

Sustav Minix je napisan da bi prikazivao nekoliko znanstvenih koncepata u operativnim sustavima. Torvalds je te koncepte objedinio u samostalan sustav nalik UNIX-u. Do programa su mogli doći studenti računalstva širom svijeta, tako da su ubrzo nastale zasebne novinske grupe. Linus Torvalds se potrudio ponuditi svojim kolegama korisnicima Minixa bolju platformu koju je bilo moguće pokretati na osobnim računalima.

Linus se opredijelio za računala s procesorom 386 jer je taj procesor imao mogućnost višezadaćnog rada.

Slijedi nekoliko tvrdnji koje je izrekao Linus nakon objave svog sustava Linux:

"Nakon toga sve je postalo jednostavnije, još je uvijek trebalo pisati dosta kodova no imao sam neke uređaje tako da je otklanjanje grešaka bilo jednostavno. U to sam vrijeme počeo koristiti C što je bitno ubrzalo razvoj. Tada sam i počeo ozbiljno razmišljati o svojim megalomanskim idejama da načinim "bolji Minix od Minixa". Nadao sam se da ću jednog dana moći koristiti gcc pod Linuxom...";

"Dva je mjeseca trebalo za osnovna podešavanja, a zatim nešto više da bi dobio upravljački program za disk (vjerojatno s greškama, ali radi na mom računalu) i mali datotečni sustav. Tako sam stvorio inačicu 0.01 (krajem kolovoza 1991. godine), koja nije bila dobra jer nije imala upravljački program za disketni uređaj, niti velike mogućnosti. Mislim da je nitko nikada neće upotrijebiti. No, od tada sam doista zaražen idejom tako da ne želim odustati sve dok ne budem mogao izbaciti Minix." **BILJEŠKA** Ove izjave su izdvojene iz knjige "Instalacija Linuxa i vodič za početak rada" koji je objavio Matt Welsh (copyright 1992-94 by Matt Welsh, 205 Gray Street NE, Wilson, NC 27893, **mdw@sunsite.unc.edu**).

Na različitim lokacijama možete pronaći kompletnu knjigu "Instalacija Linuxa i vodič za početak rada", a jedna od njih je i **sunsite.unc.edu**, na kojoj trebate potražiti direktorij /pub/Linux/docs/LDP/install-guide. U 31. poglavlju ove knjige, "Kretanje Internetom pomoću World Wide Weba" naći ćete informacije kako pristupiti arhivama i kopirati datoteke.

Linus je 5. listopada 1991. godine objavio Linux 0.02, odnosno prvu službenu inačicu Linuxa:

"Žalite li za lijepim vremenima Minixa 1.1, kada je čovjek bio čovjek i sam pisao upravljačke programe? Jeste li ostali bez lijepe vrste posla i sada lomite zube pokušavajući neki operativni sustav prilagoditi svojim potrebama? Ne čini li vam se da je frustrirajuće ako sve na Minixu radi? Nema više noćnog rada i rješavanja problema? U tom slučaju vas ovaj sustav može zanimati";

"Kao što sam spomenuo prije nekoliko mjeseci, radim na besplatnoj inačici operativnog sustava koji je nalik Minixu i koristi se na AT-386 osobnim računalima. Taj je projekt napokon dosegao razinu kada ga se može upotrijebiti (iako to možda neće biti ono što želite), te sam odlučio objaviti izvorni kôd. Radi se o inačici 0.02, pod kojom mogu uspješno pokretati naredbe bash, gcc, gnu-make, gnu-sed, compress i slično."

# Tko je vlasnik Linuxa?

IBM posjeduje OS/2, a Microsoft MS-DOS i MS Windowse, ali tko je vlasnik Linuxa? Prvo i osnovno, razni ljudi polažu pravo na različite komponente Linuxa. Linus Torvalds pridržava prava na osnovnu Linux jezgru. Tvrtka Red Hat je vlasnik Red Hat distribucije, a Patrick Volkerding distribucije Slackware. Mnogi Linux uslužni programi spadaju pod licencu o javnom korištenju GNU General Public License (GPL). U osnovi, Linus i većina tvoraca Linuxa su svoje djelo zaštitili pomoću te licence koju možete pronaći na Internetu.

Tu ćete licencu ponekad naći i pod imenom GNU Copyleft (pandan riječi *copyright*). Ta licenca obuhvaća sav softver proizveden od tvrtki GNU i Free Software Foundation. Licenca omogućava programerima stvaranje softvera za bilo koga i osnovna joj je pretpostavka da bi softver trebao pripadati svima tako da bilo tko može prepraviti program prema svojim potrebama. Jedini je uvjet da i ostali korisnici imaju prava na taj novi, promijenjeni programski kôd.

GNU Copyleft, ili GPL, omogućuje programerima očuvanje njihovih zakonskih prava, ali i korisnicima mijenjanje i prodavanje novih programa. Izvorni programeri ne mogu ograničiti ista takva prava na mijenjanje programa ljudima koji kupuju program. Ukoliko prodate program u istom ili izmijenjenom obliku, morate isporučiti i izvorni kôd programa. Zbog toga Linux uvijek možete dobiti zajedno s njegovim izvornim kôdom.

# Odavde...

Linux je nova zamjena za UNIX. Mogućnost čitanja i mijenjanja izvornog koda i pripadnih aplikacija čine Linux pravom alternativom drugim operativnim sustavima za osobna računala. Više toga saznajte u sljedećim poglavljima:

- 3. poglavlje, "Instaliranje Red Hata" sadrži detaljne upute za instaliranje Red Hat distribucije na vaše računalo.
- 4. poglavlje, "Instaliranje OpenLinux Litea" nudi informacije kako instalirati Calderinu distribuciju na vaše računalo.
- 5. poglavlje, "Pokretanje Linux programa" objašnjava vam kako ćete koristiti neke programe koji se isporučuju uz distribucije Linuxa.

# P 0 G L A V L J E 2

# Postupak instalacije Linuxa

(Jack Tackett)

## U ovom poglavlju

Hardverski zahtjevi Početak instalacije Različiti načini instalacije Podjela tvrdog diska na particije Otklanjanje problema U ovom ćete poglavlju naći potrebne informacije za instaliranje gotovo svake Linux distribucije. Upamtite, Linux nije komercijalni proizvod i može vam se dogoditi da naiđete na određene probleme. Iako ova knjiga sadrži dosta informacija, možda ćete trebati potražiti dodatne izvore kao što su razne HOWTO datoteke koje možete pronaći na Internetu.

Autori ove knjige su pretpostavili da već imate određeno radno iskustvo s DOS-om, te da su vam **BILJEŠKA** poznati pojmovi kao što su formatiranje tvrdog diska, particije i veličina sektora. Ako vam sve to zvuči kao strani jezik, potražite neku dobru knjigu o DOS-u ili zamolite nekog stručnjaka za DOS da vam pomogne.

**SAVJET** Imate mogućnost unositi bitne promjene u svoj sustav, stoga budite vrlo pažljivi. Nije loša ideja da kraj sebe uvijek držite olovku i papir kako biste mogli zabilježiti probleme na koje naiđete. Osim toga, prilikom instalacije ćete trebati zapamtiti i određene brojeve.

# Hardverski zahtjevi

Da biste mogli uspješno instalirati Linux, potreban vam je određeni hardver. Odabir prave konfiguracije na kojoj ćete instalirati Linux ovisi o broju korisnika te o vrsti programa koji će se koristiti, jer ti zahtjevi određuju veličinu radne memorije, prostora na tvrdom disku, vrste potrebnih terminala i slično.

#### NA WEBU

Informacije o hardveru kojeg Red Hat podržava (i ne podržava) možete pronaći na Web lokaciji:

#### http/www.redhat.com/support/docs/rhl/intel/rh51-hardware-intel.html.

Danas većina Linux sustava radi na osobnim računalima. Obično su ti sustavi prilagođeni za jednog korisnika, iako mogu biti i sastavni dio većih Linux ili UNIX sustava.

#### UPOZORENJE

Linux se neprestano razvija, a usporedo s time se razvija i podrška za hardver. Red Hat distribucija, koja je jedna od najjačih, je relativno stabilna, no od vremena izdavanja knjige sigurno se pojavila i nova hardverska podrška. Iako većina hardverskih komponenti ima svoju zamjenu, ipak ne mogu sve komponente raditi pod Linuxom. Ako koristite hardver koji je opisan u ovom poglavlju, tada imate odlične šanse da vaš Linux sustav uspješno instalirate i pokrenete. No, ako se vaša oprema razlikuje od opisane u ovom poglavlju, Linux može (ali i ne mora) pravilno raditi.

Koristite li konfiguraciju Linuxa za jednog korisnika (najčešća konfiguracija), tada ste i sami postali administrator sustava. U tom slučaju trebate dobro poznavati sustav kako biste mogli obavljati i administratorove zadaće i svoj operativni sustav održavati u optimalnom stanju. Ove dužnosti se sastoje od održavanja dovoljne količine slobodnog prostora na disku, sigurnosne pohrane i brige da svi pripadni uređaji koriste odgovarajuće upravljačke programe. Osim toga, u održavanje sustava spada i instalacija i podešavanje softvera i slično.

Odabir hardvera koji ćete koristiti uglavnom ovisi o hardveru koji su koristili programeri prilikom stvaranja Linuxa. Za razliku od programera komercijalnog softvera, koji mogu isprobati svoj softver na različitim hardverskim konfiguracijama, Linux programeri obično mogu koristiti samo vlastita osobna računala. Na sreću, s obzirom da postoji puno Linux programera, ipak možete naći podršku za većinu hardverskih komponenti.

### Procesor

Osnovni sustav zahtijeva IBM-uskladivo osobno računalo s procesorom Intel 80386 (ili novijim) uključujući sve tipove procesora kao što su 80386SX, 80486DX/2 i različite Pentium procesore. S Linuxom su također uskladivi i drugi procesori kao što su čipovi 80386 koje su proizvele tvrtke Cyrix i Advanced Micro Devices (AMD).

Procesori 80386 i 80486SX nemaju ugrađen matematički koprocesor, ali on i nije nužan za Linux. Korištenjem softverskih rutina, Linux emulira koprocesor, ali uz znatno smanjenu brzinu rada. Da bi vaš sustav brže radio, trebali biste nabaviti procesor s ugrađenim matematičkim koprocesorom kao što je 80486DX ili Pentium (u svim svojim varijacijama).

Linux jezgra također podržava i druge procesore, kao što su DEC Alpha, PowerPC (Mac), Sun Sparcs, te umetnuti sustavski procesori koje koristi Calderino mrežno osobno računalo.

## Sabirnice

Također, bitna je i vrsta sabirnica koje se koriste za komunikaciju s perifernim jedinicama sustava. Linux radi samo sa sabirnicama ISA, EISA i PCI, a ne podržava sabirnice MicroChannel Architecture (MCA) koje se koriste na IBM-ovim računalima PC/2 (no ta se podrška upravo razvija, možda je već i gotova). Neki noviji sustavi koriste brze sabirnice za pristup disku i grafičkim karticama, tj. koriste *lokalne sabirnice*. Linux podržava lokalne sabirnice VESA, ali možda nećete moći koristiti druge vrste lokalnih sabirnica.

## Memorija

Za samo je pokretanje Linuxa potrebno relativno malo RAM-a, osobito u usporedbi s drugim operativnim sustavima kao što su OS/2 i Windowsi NT. Linux zahtijeva najmanje 2 MB RAM-a, iako vam preporučamo da koristite 4MB. Imate li manje od 4 MB RAM-a, tada trebate koristiti *swap* datoteku. Osnovno je pravilo: što sustav ima više memorije na raspolaganju, to brže radi.

Idući zahtjev za memorijom, koji se javlja pri uporabi Linuxa, predstavlja klon XWindowsa koji se naziva XFree86. XFree86 je inačica XWindowsa koja se besplatno distribuira i uključena je u instalaciju Linuxa. Grafičko korisničko sučelje Xfree86 je nalik Microsoft Windowsima.

Vidi "Instaliranje sustava XFree86", stranica 425.

Da biste mogli koristiti X, potrebno je imati barem 16 MB virtualne memorije. *Virtualna memorija* predstavlja kombinaciju fizičke memorije i prostora na tvrdom disku (*swap*). Opet vrijedi isto pravilo - što više fizičke memorije sadrži sustav, on će biti brži, osobito prilikom uporabe programa XFree86.

## Diskovni pogoni i prostor na disku

Iako je Linux moguće pokretati i s diskete, ipak vam to ne bismo preporučili.

BILJEŠKA

Linux možete podići i s diskete. *Podizanje sustava* označava postupak pokretanja računala i učitavanja operativnog sustava u memoriju.

Kod kućnih računala vam je potreban disketni pogon (5,25 ili 3,5 palca), čak i u slučaju da Linux instalirate i pokrećete s CD-ROM-a.

Da biste dobili bolje karakteristike sustava, trebali biste Linux instalirati na tvrdi disk. Trebate imati IBM AT disk kontroler, što i ne bi trebao biti problem s obzirom da je većina modernih kontrolera AT-uskladiva. Linux podržava sve MFM i IDE kontrolere, kao i većinu RLL i ESDI kontrolera. Linux može (ali i ne mora) podržavati novije IDE pogone sa starijim 8-bitnim IDE kontrolerima.

Linux podržava široki raspon SCSI kontrolera za tvrdi disk. Ako je vaš kontroler doista SCSI, a ne samo slična inačica SCSI kontrolera, tada će ga Linux moći koristiti. Linux danas podržava SCSI kontrolere tvrtki Adaptec, Future Domain, Seagate, UltraStor, SCSI adapter na kartici ProAudio Spectrum 16 i Western Digital. Slijedi popis kartica koje Linux podržava:

Adaptec 152x/1542/1740/274x/284x/294X	Always IN2000
Buslogic	Pro Audio Spectrum 16
EATA-DMA (DPT, NEC, AT&T)	Qlogic
Seagate ST-02	Trantor T128/T128F/T228
Future Domain TMC-8xx, 16xx	UltraStor
Generic NCR5380	7000FASST

NCR 53x7, 8xx

Ako imate odgovarajući disk kontroler, tada jedino trebate voditi brigu o zahtjevima koji se odnose na slobodan prostor na disku. Linux podržava više različitih diskova. Za razliku od drugih operativnih sustava, Linux ne zahtijeva instalaciju na istom pogonu na disku, već pojedini dijelovi mogu biti instalirani na raznim pogonima.

**Prostor na disku** Linux se ne treba čitav nalaziti na istoj particiji kao drugi operativni sustavi kao što su MS-DOS i OS/2. *Particije* su područja na disku, određena tijekom inicijalizacije diska, a prije formatiranja. Da biste disk podijelili na particije, obično se koristi program **fdisk**. Neki komercijalni programi omogućuju vam preparticioniranje diska (kod Linuxa to čini uslužni program pod imenom FIPS). Da biste učinkovito koristili sustav Linux, trebali biste preparticionirati svoj tvrdi disk i ostaviti dovoljno prostora za sustavske datoteke Linuxa i datoteke s podacima.

#### **UPOZORENJE**

Osim ako Linux ne instalirate na potpuno novi disk, trebat ćete preparticionirati i preformatirati disk. Na taj se način uništavaju svi podaci koji su pohranjeni na tom dijelu diska. Zbog toga trebate pohraniti svoje datoteke (i to dvostruko) prije nego instalirate Linux. Ako imate dovoljno prostora tada jedan tvrdi disk možete podijeliti na više particija i datoteke kopirati na jednu od njih.

Potrebna količina prostora na disku ovisi o softveru koji namjeravate instalirati i količini podataka koju će taj softver proizvesti. Linux zahtijeva manje prostora na disku od UNIX-a. Da biste pokrenuli potpuno funkcionalan Linux sustav, bez XWindowsa, potrebno vam je oko 20 MB prostora. No, za cjelovitu instalaciju svega što se nalazi u određenoj distribuciji preporuča se slobodan prostor od 150 do 200 MB.

**Swap prostor** Kao što smo spomenuli u prethodnom odlomku "Memorija", ukoliko je količina vašeg RAM-a ograničena, tada vam je potreban *swap* prostor (prostor na disku koji se ponaša kao dopunska memorija). Za razliku od Microsoft Windowsa i drugih sustava koji stvaraju *swap datoteku* koja se pojavljuje na tvrdom disku kao obična datoteka, Linux omogućuje smještaj *swap datoteke* na posebnoj *swap particiji*. Većina Linux instalacija koriste particije umjesto datoteka. S obzirom da možete smjestiti više particija na isti fizički disk, to znači da na isti disk možete pohraniti i *swap particiju*. No bolje je da je pohranite na poseban pogon.

Linux omogućava stvaranje osam *swap particija* koje ne mogu biti veće od 16 MB. Pravilo je da veličinu *swap datoteke* podesite na dvostruku veličinu RAM-a koji se nalazi u vašem sustavu. Stoga, ako imate 8 MB RAM-a, tada bi vaša *swap particija* trebala biti velika 16 MB.

## Monitor

Linux podržava sve standardne grafičke kartice i monitore kao što su Hercules, CGA, EGA, VGA i SuperVGA. Da biste iskoristili prednost ispisa u boji koje koristi Linux, potreban vam je i monitor u boji. Kod tekstualnih operacija bi trebala raditi bilo kakva kombinacija monitora i kartice.

Veliki se problemi javljaju kada pokrenete XWindowse koji se distribuiraju s Linuxom. Da biste mogli koristiti program XFree86, potrebna vam je grafička kartica (video-adapter) koja koristi neke od čipova iz tablice 2.1. *Čipovi* su u osnovi grupa integriranih krugova koji se koriste za prihvat podataka s računala i njihovo pretvaranje u oblik koji se može prikazati na monitoru. Da biste saznali koje čipove koristi vaša kartica, pročitajte dokumentaciju koju ste dobili, a potom provjerite hoćete li imati problema prilikom uporabe programa XFree86.

Tablica 2.1	Video čipovi koje Linux podržava	
Proizvođač	Čipovi	
Tseng	ET3000, ET4000/W32	
Western Digital	WD90C00, WD90C10, WD90C11, WD90C24, WD90C30, WD90C31	
Trident	TVGA8800CS, TVGA8900B, TVGA8900C, TVGA8900CL, TVGA9000, TVGA9000i, TVGA9100B, TVGA9200CX, TVGA9320, TVGA9400CX, TVGA9420	
ATI	28800-4, 28800-5, 28800-a	
NCR	77C22, 77C22E, 77C22E+	
Cirrus Logic	CLGD5420, CLGD5422, CLGD5424, CLGD5426, CLGD5428, CLGD6205, CLGD6215, CLGD6225, CLGD6235	
OAK	OTI067, OTI077	
S3	86C911, 86C924, 86C801, 86C805, 86C805i, 86C928	
Compaq	AVGA	
Western Digital/	Paradise PVGA1	

BILJEŠKA Uz najnoviju inačicu programa XFree86, koja se distribuira s Linuxom, trebali biste naći popis čipova koje Linux podržava i koje ne podržava (gornji popis s obzirom na novije inačice vjerojatno nije potpun).

Programeri sučelja XFree86 su se suočili s nekim problemima jer im proizvođači grafičkih kartica nisu dostavili potrebne informacije o programiranju kartica. Bez tih podataka programeri nisu mogli načiniti podršku za kartice. Osim toga, neki proizvođači daju te informacije, ali zahtijevaju kraljevski dio kolača (novca) ili čvrsti dogovor da drugi neće koristiti te informacije. Zbog takvih ograničenja nije bilo moguće stvoriti podršku za neke kartice, jer je program XFree86 sastavni dio Linuxa koji se besplatno distribuira.

BILJEŠKA Do sada nema potpore za grafičke kartice Diamond, jednostavno stoga što tvrtka ne želi dati potrebne informacije. U zadnje vrijeme tvrtka Diamond počinje surađivati s XFree timom kako bi izradili potporu za svoje kartice koje bi radile pod Linuxom i programom XFree86.

## **CD-ROM**

Da biste instalirali Linux koji se nalazi na CD-ROM-u koji ste nabavili, morate imati CD-ROM kojeg podržava Linux. Što se tiče CD-ROM-ova koji koriste SCSI kontroler, praktički će svaki SCSI kontroler koji je naveden u prethodnom odlomku "Diskovni pogoni

i prostor na disku", vjerojatno raditi s CD-ROM-om. Sada Linux podržava mnoge novije EIDE i ATAPI CD-ROM-ove koji su se pojavili na tržištu.

Mnogi CD-ROM-ovi koji su sastavni dio multimedijskih paketa mogu (ali i ne moraju) raditi pod Linuxom, ovisno o tome koriste li odgovarajući SCSI kontroler ili ne. U većini slučajeva drugi kontroleri neće raditi. Linux podržava liniju CD-ROM-ova tvrtke Creative Labs SoundBlaster i nudi posebnu instalaciju za tu vrstu uređaja. Sljedeći CD-ROM-ovi provjereno rade pod Linuxom:

NEC CDR-74	Okano
Sony CDU-541	Wearnes CD s interface karticom
Sony CDU-31a ili 33a	
Plextor DM-3024	Većina IDE/ATAPI CD-ROM-ova
Aztech	Mitsumi CD-ROM-ovi
Orchid	SoundBlaster, Panasonic Kotobuki, Matsushita TEAC-55a ili Lasermate

## Mrežne kartice

Operativni sustav Linux možete povezati sa svijetom na nekoliko različitih načina, od kojih su dva najpopularnija: putem mrežne kartice i modema. Najpoznatije vrste mrežnih kartica su Token Ring, FDDI, TAXI i Ethernet, a većina poslovnih mreža koristi Ethernet mrežne kartice.

**Pristup mreži putem Ethernet kartica** Ethernet je protokol koji je uvela tvrtka Xerox i taj je mrežni protokol doživio iznimnu popularnost u svjetskim mrežama. Iako svoje kućno Linux računalo nećete povezivati s globalnom mrežom putem Ethernet kartice, mnoge poslovne i obrazovne ustanove to čine. Tablica 2.2 sadrži popis nekoliko Ethernet kartica koje podržava Linux.

Tablica 2.2 Et	hernet mrežne kartice koje Linux podržava
Proizvođač	Kartica
3Com	3c503, 3c503/16, 3c509
Novell	NE1000, NE2000
Western Digital	WD8003, WD8013
Hewlett-Packard	HP27245, HP27247, HP27250

**Pristup mreži putem modema** Iz svoje kuće se na svjetsku mrežu obično spajate putem modema i komunikacijskih protokola kao što su SLIP ili PPP. Linux podržava gotovo sve vrste modema koje se trenutno nalaze na tržištu, bez obzira radi li se o unutarnjim ili

vanjskim modemima. Ako mreži možete pristupiti putem modema iz MS-DOS-a, nećete imati problema kada to isto budete htjeli učiniti pod Linuxom.

Vidi "Razumijevanje zahtjeva SLIP i PPP", stranica 562

## Periferni uređaji

U nastavku ćete naći popis periferija koje Linux podržava, a u koje spadaju miš, pogon za trake i pisači. Iako ove hardverske komponente olakšavaju korištenje Linuxa, one nisu neophodne u radu.

**Miš** Uporaba Linuxa koji je u osnovi tekstualni operativni sustav ne zahtijeva korištenje miša. Za razliku od mnogih UNIX sustava, Linux vam omogućava da pomoću miša odrežete tekst s jednog dijela ekrana i ulijepite ga u redak s naredbom. Namjeravate li koristiti klon XWindowsa, program XFree86, tada će vam miš svakako biti potreban.

Linux podržava većinu serijskih miševa među kojima se nalaze:

- Logitech
- MM series
- Mouseman
- Microsoft
- Mouse Systems

Linux također podržava Microsoft, Logitech, ATIXL i PS/2 *bus* (sabirničke) miševe. U osnovi, bilo koji uređaj za pokazivanje, kao što je *trackball* ili ekran osjetljiv na dodir, a koji emulira navedene miševe, može raditi pod Linuxom.

**Pogon za trake** Pogoni za trake predstavljaju odličan način sigurnosne pohrane vaših podataka. Linux podržava nekoliko SCSI pogona za trake (kao što je pokazano u tablici 2.3). Osim toga, moći ćete koristiti i popularni pogon Colorado Memory Systems (inačice 120 i 150). Inačice koje se spajaju na priključak za pisač se još uvijek ne mogu koristiti pod Linuxom. Većina pogona koji podržavaju QIC-02 bi također trebali raditi pod Linuxom.

Tablica 2.3	Pogoni za trake koje podržava Linux		
Proizvođač	Model		
Exabyte	Svi SCSI modeli		
Sanko	CP150SE		
Tandberg	3600		
Wangtek	5525ES, 5150ES, 5099EN		

**Pisači** Linux podržava sve paralelne pisače. Konfiguracija Linuxa za podršku serijskih pisača je vrlo opasna i moguće su greške. Podrška za serijske pisače nije dobro dokumentirana niti podržana u osnovnoj Linux instalaciji. Ukoliko koristite serijski pisač, mogu vam se javiti problemi kod rada pod Linuxom. Ako pak koristite paralelni pisač, onda će vam najveći problem biti *stupnjevit* ispis:

Ovo je prvi redak. Ovo je drugi redak.

```
Ovo je treći redak.
```

Evo kako UNIX, pa tako i Linux, tretiraju prelazak u novi redak i vraćanje glave pisača na početak retka. Kod većine UNIX sustava, naredba za pomicanje papira za jedan redak niže, te postavljanje glave pisača na početak retka je predstavljena jednim upravljačkim znakom. U sustavima kao što su MS-DOS i Windowsi, te su naredbe predstavljene odvojenim upravljačkim znakovima. Ako UNIX datoteku ispisujete na pisač koji je prilagođen za

MS-DOS, dobit ćete stupnjevit ispis jer datoteka sadrži samo naredbu za prelazak u novi redak, a ne i za vraćanje glave pisača na početak.

Vidi "Što vam treba za podešavanje postavki pisača?", stranica 406.

# Početak instalacije

Da biste započeli instalaciju Linuxa, potrebne su vam jedna ili dvije (ovisno o instalaciji) diskete 3,5", 1,44 MB. One će biti upotrijebljene za stvaranje *boot* disketa (disketa za pokretanje sustava) za instalaciju Linuxa.

Zatim trebate osigurati dovoljno prostora na tvrdom disku da biste instalirali Linux. Ukoliko želite instalirati sve što distribucija sadrži, tada će vam trebati oko 300 MB prostora, no možda će vam biti dovoljno i manje prostora, osobito ako ne namjeravate instalirati XWindowse. Da biste odredili potrebnu količinu prostora na disku, potrebno je odlučiti koliko prostora želite odvojiti za korisničke račune, odnosno, odrediti prostor koji ćete ponuditi korisnicima. Ukoliko sustav predviđa samo jednog korisnika, tada je 30 MB više nego dovoljno.

Potom odredite koliko vam treba *swap* prostora na disku. Ukoliko vaše računalo ima 8 MB RAM-a ili manje od toga, onda vam je potrebno 16 MB *swap* prostora. No, ako imate 16 MB RAM-a, onda *swap* prostor treba biti veličine memorije.

I na kraju trebate predvidjeti oko 30 MB za korijenski (*root*) direktorij, koji će biti glavni direktorij iz kojeg ćete pristupati svim drugim Linux direktorijima.

Vidi "Standardni direktoriji u Linuxu", stranica 313.

Minimalna instalacija može zauzimati oko 200 MB prostora, dok puna instalacija, s dosta korisničkog prostora, može stati na 500 MB.

**BILJEŠKA** Dio Linuxovog sustava datoteka možete pokrenuti s CD-ROM-a (ako instalaciju obavljate s CD-a) bez instalacije čitavog softvera. Tu odluku možete donijeti tijekom instalacije.

Odlučite li instalirati i podesiti XWindowse (što vam zaista preporučamo), trebali biste također zapisati vrstu čipova koje koristi vaša grafička kartica. Ako imate serijskog miša i modem, zapišite serijske priključke na kojima su spojeni, jer će vam ti podaci trebati tijekom podešavanja sustava.

# Različiti načini instalacije

Autori ove knjige predviđaju da će većina korisnika instalirati Red Hat distribuciju s CD-ROM-a. Ipak, možete upotrijebiti bilo koju od iduće četiri metode za instalaciju Red Hata: s CD-ROM-a, putem NFS-a, FTP-a ili s tvrdog diska.

Da biste sustav instalirali izravno s CD-ROM-a, trebate najprije pokrenuti DOS. U DOS *promptu* upišite naredbu

[cdrom-drive]:dosutils\autoboot gdje [cdrom-drive] zamjenjuje naziv pogona CD-ROM-a.

#### **UPOZORENJE**

Koristeći ovu metodu obrisat ćete sadržaj na svom tvrdom disku, tako da najprije morate zaštititi sve bitne podatke.

Ako na raspolaganju imate još jednu particiju, tada Linux možete instalirati usporedo svom operativnom sustavu, bez brisanja ostalog sadržaja. Da biste to učinili, potreban vam je CD-ROM, prazna particija na disku i *boot* disketa. U nastavku ovog poglavlja ćete naučiti kako se stvara *boot* disketa, te kako možete preparticionirati svoj tvrdi disk.

NFS (Network File System) vam omogućava instalaciju Red Hat distribucije putem mreže. Najprije morate podesiti pogon CD-ROM-a da podržava sustav datoteka ISO-9660 s RockRidge ekstenzijama, a zatim eksportirati sustav datoteka putem NFS-a. Potrebno je znati stazu (*path*) izvezenog sustava datoteka i IP adresu ili, ako je podešen DNS, poznavati ime sustava.

FTP (File Transfer Protocol) je način prijenosa datoteka putem Interneta (u 30. poglavlju, "Pristup mreži pomoću naredbi telnet, ftp i r-" ćete naći detaljniji opis FTP-a). Da biste instalirali Linux putem FTP-a, potrebna vam je *boot* disketa i tzv. *dopunska* disketa (koja će biti objašnjena u nastavku poglavlja).

Instalacija Red Hat distribucije s tvrdog diska zahtijeva *boot* i dopunsku disketu, kao i kod instalacije putem FTP-a. Najprije stvorite direktorij pod imenom RedHat. Zatim kopirajte odgovarajući direktorij s CD-ROM-a, kao i sve poddirektorije. To možete učiniti pomoću ovih DOS naredbi:

```
cd \RedHat
xcopy /s e:\RedHat
```

Kod naredbe c<br/>d podrazumijeva se da se nalazite na tvrdom disku, a kod naredbe x<br/>copy da je e ime pogona CD-ROM-a.

Bez obzira koji način instalacije Linuxa upotrijebite, bit će vam potrebna barem *boot* disketa. No, najprije ćete trebati sakupiti neke informacije.

## Prikupljanje potrebnih informacija

Prije nego započnete s instalacijom, bit će vam potrebne sljedeće informacije o vašem računalu:

- Vrsta grafičke kartice, čipova i monitora
- Serijski priključak koji se koristi za miša
- Serijski priključak koji se koristi za modem
- Ako je vaše računalo spojeno na mrežu, potrebno je znati IP adresu, ime domene, gateway i slično
- Vrstu tvrdog diska i pogona CD-ROM-a, te njihove kontrolere
- Ime koje želite dati svom sustavu

Ako imate vezu na Internet, tada većinu ovih podataka možete dobiti od administratora mreže ili davatelja Internet usluga.

Ako na istom računalu namjeravate koristiti i neki drugi operativni sustav (kao što su Windowsi 95, Windowsi NT ili OS/2), tada na disku trebate stvoriti nužne particije za te operativne sustave. Obično ćete trebati koristiti softver operativnog sustava za podjelu na particije, jer Linux ne može raditi s drugim vrstama particija.

#### NA WEBU



Softver pod imenom System Commander (tvrtka V Communications) omogućuje vam instalaciju i korištenje 32 različita operativna sustava. O tome saznajte nešto više na adresi

#### http://www.v-com.com/

Zatim biste trebali pročitati promjene koje su u zadnji trenutak unesene u Red Hat distribuciju. Ima više razloga za to, no dva glavna razloga su da se Linux svakodnevno mijenja, a ova knjiga opisuje inačice nakon kojih su se pojavile i nove. U međuvremenu su se pojavili novi materijali ili su otklonjene neke greške.

#### NA WEBU



Najnovije materijale potražite na Webu i to na adresi

#### http://www.redhat.com/errata

Tablica 2.4 sadrži popis trenutno dostupnih paketa koji rješavaju poznate probleme u Red Hat distribuciji.

Vidi "Instalacija paketa pomoću RPM-a", stranica 150

Tablica 2.4	Popis paketa za rješavanje problema u Red Hat distribuciji
Datum objave	Paket
01-06-1998	linuxconf
01-06-1998	xosview
01-06-1998	bootp
01-06-1998	metamail
02-06-1998	initscripts
02-06-1998	dheped
05-06-1998	netkit-base
05-06-1998	glint
05-06-1998	fstool, usercfig, cabaret
10-06-1998	xscreensaver
10-06-1998	findutils
10-06-1998	emacs
10-06-1998	inn
10-06-1998	libjpeg, et al
10-06-1998	Netscape
10-06-1998	kernelcfg
10-06-1998	tmpwatch
10-06-1998	patch
11-06-1998	Xfree86
11-06-1998	dhcp
12-06-1998	mailx

Informacije koje nudi tvrtka Caldera pogledajte u datoteci /pub/OpenLinux/updates/README koja se nalazi na adresi

#### ftp://ftp.caldera.com/pub/OpenLinux/ipdates/.

Ti direktoriji sadrže razne ažurirane sadržaje za *boot* diskete i pakete u RPM formatu za Calderinu OpenLinux distribuciju. Tablica 2.5 nudi popis dostupnih ažuriranih paketa.

Tablica 2.5 Topis pareta				
Datoteka/Direktorij	Opis			
update.col-1.2007.12.tgz	Ažurirana skripta za OpenLinux 1.2			
update.col.README	Upute za korištenje ažurirane skripte			
README	Upute za dobivanje ažuriranih materijala			
1.0/	Ažurirani paketi za OpenLinux Base 1.0			
1.1/	Ažurirani paketi za OpenLinux 1.1			
1.2/	Ažurirani paketi za OpenLinux 1.2			

Tablica 2.5 Popis paketa za rješavanje problema u Caldera distribuciji

Ukoliko ne instalirate sustav s CD-ROM-a, tada je potrebno preparticionirati tvrdi disk kako biste napravili mjesta za Linux. Tu se mogu pojaviti problemi, jer preparticioniranje diska uništava sve podatke na odabranoj particiji. Nakon što napravite mjesta za Linux, potrebno je podići Linux sustav te stvoriti nove particije i sustave datoteka. Obično je potrebno načiniti primarnu particiju za pohranu datoteka i particiju za *swap* datoteku, osobito ukoliko koristite računalo s 8MB RAM-a (ili manje).

**BILJEŠKA** Sustav datoteka je u osnovi dio tvrdog diska posebno formatiran za pohranu određenih vrsta datoteka. UNIX i Linux koriste sustave datoteka za prikaz stabla direktorija. U tome se razlikuju od MS-DOS-a koji poddirektorije postavlja u stablo direktorija na istom logičkom disku. UNIX sustavi koriste format stabla direktorija radi sigurnijeg smještaja poddirektorija na različite pogone. Ukoliko se javi greška na jednom pogonu, tada je potrebno zamijeniti samo podatke s tog pogona.

Vidi "Sustavi datoteka", stranica 266.

Nakon što ste stvorili sustav datoteka, potrebno je instalirati operativni sustav Linux, njegove popratne datoteke i razne pakete s aplikacijama. Da biste instalirali Linux, najprije trebate pokrenuti skraćenu inačicu operativnog sustava tako da stvorite *boot* i dopunsku disketu sa sažetim operativnim sustavom.

## Stvaranje boot i dopunske diskete

Pomoću programa *rawrite* trebate stvoriti *boot* i dopunsku disketu. Ovaj program možete naći na odgovarajućem CD-ROM-u u poddirektoriju /dosutils. Da biste stvorili *boot* i dopunsku disketu, potrebne su vam dvije formatirane diskete od kojih je jedna označena kao *boot*, a druga kao *supp. Boot* disketu stavite u disketni pogon i upišite sljedeće:

```
E:\dosutils>rawrite
```

```
Enter disk image source file name: e:\images\boot.img
Enter target diskette drive: A:
Please instert a formatted diskette into drive A: and press -ENTER-
```

Ukoliko želite prekinuti proces, jednostavno pritisnite tipku <Ctrl>. Ako naiđete na probleme, pokušajte s novom disketom. Ako se problemi nastave, tada provjerite svoj hardver.

Nakon što stvorite *boot* disketu, trebate kreirati i *dopunsku* disketu. Jednostavno u prethodnoj naredbi upišite supp.img kao ime datoteke.

## Podjela tvrdog diska na particije

Nakon što ste načinili sigurnosnu kopiju svog sustava i stvorili *boot* i dopunsku disketu, morate pripremiti tvrdi disk za sustav Linux.

#### UPOZORENJE

Ovo je najopasniji dio instalacije jer se mogu izgubiti mnogi podaci. Ako još niste načinili sigurnosnu kopiju svog sustava, onda to svakako učinite sada. Čak ako i koristite eksperimentalni program pod imenom FIPS i komercijalne programe kao što su Partition Magic koji ne brišu postojeće podatke, ipak vam preporučamo da načinite zaštitnu kopiju podataka na disku.

## Što su particije

U početku razvoja osobnih računala postojalo je nekoliko tvrdih diskova koji su bili međusobno udaljeni. Većina računala su koristila diskete za podizanje operativnog sustava, spremanje programa i podataka. Pojavom IBM-ovih osobnih računala XT, IBM je uveo tvrdi disk veličine 10 MB. Raniji operativni sustavi kao DOS su imali pristup samo ograničenom prostoru na tvrdom disku. Potom su proizvođači tvrdih diskova počeli povećavati prostor na disku puno brže nego što su tom prostoru mogli pristupiti operativni sustavi. Da bi se izbjegao taj problem, korisnici su trebali dijeliti tvrdi disk u sekcije nazvane *particijama*. Te particije mogu sadržavati programske datoteke, druge operativne sustave ili podatke.

Tipični MS-DOS sustavi imaju samo jednu particiju koja se označava slovom C. Ako pogon podijelite u dvije particije, one će dobivati imena prema abecednom redu – D, E i tako dalje. MS-DOS vam također omogućuje instalaciju većeg broja tvrdih diskova tako da se idući disk u tom slijedu nazove slovom F.

UNIX i Linux ne koriste slova za označavanje particija već umjesto toga upotrebljavaju imena direktorija. Osim toga, kao što je prethodno naznačeno, Linux korisnici mogu postaviti različite direktorije na razne particije i čak na razne diskove. Također možete smjestiti različite operativne sustave na razne particije.

Particije su određene u dijelu tvrdog diska koji se naziva *boot record* u kojem se nalazi *tablica particija*. Tu tablicu koriste različiti operativni sustavi za određivanje onog sustava kojeg treba pokrenuti i mjesta na kojem se njegove datoteke fizički nalaze na tvrdom disku. *Boot record* se koristi za pokretanje operativnog sustava računala. Linuxov program za pokretanje, LILO (LInux LOader), i drugi slični programi koriste taj dio na tvrdom disku za određivanje sustava kojeg se pokreće (taj se dio obično nalazi na prvim sektorima diska).

Tablica particija sadrži informacije o lokacijama i veličini različitih particija na tvrdom disku. Postoje tri vrste particija: primarne, proširene i logičke. DOS i drugi operativni sustavi moraju se pokretati s primarnih particija. Tvrdi disk može sadržavati samo četiri primarne particije. Proširena particija ne sadrži podatke već omogućuje korisniku definiranje drugih, logičkih particija na disku. Da biste izbjegli ograničenje od četiri primarne particije, možete definirati proširenu particiju i potom stvoriti druge logičke particije unutar proširene particije.

Neki operativni sustavi, kao što su MS-DOS i OS/2 (inačice koje su prethodile 2.0) moraju biti instalirane na primarnu particiju, ali mogu pristupati logičkim pogonima u proširenim particijama. Ovo je važno zapamtiti ukoliko namjeravate koristiti DOS i Linux na istom disku. DOS mora ići na primarnu particiju.

## **Program FDISK**

Načinili ste particije i uništili postojeće koristeći program koji se naziva FDISK. Svaki operativni sustav koristi vlastitu inačicu tog programa, tako da obavezno trebate provjeriti imate li pravu inačicu. Ukoliko koristite DOS ili ga namjeravate koristiti, najprije morate preparticionirati DOS pogon pomoću DOS naredbe FDISK, a kasnije možete upotrijebiti Linux inačicu istog programa i stvoriti Linux particije. Ako koristite sustav OS/2, trebate koristiti i njegovu inačicu programa FDISK.

**Zahtjevi particija** Najprije biste trebali isplanirati koje su vam sve particije potrebne. DOS traži primarnu particiju, Linux i OS/2 mogu biti smješteni u drugim particijama. Osim toga, trebate znati da, ukoliko smanjite DOS particiju da biste načinili mjesta za Linux, istovremeno će vam nedostajati prostora da biste obnovili sve DOS datoteke na manjoj particiji.

# **BILJEŠKA** Pomoću Linuxa možete pristupiti DOS particijama, odnosno možete premještati, spremati i uređivati DOS datoteke putem Linuxa.

Dvije Linux komponente (u fazi su provjere) omogućuju vam emuliranje DOS-a pod Linuxom i instalaciju Linuxa pod DOS-om. Oba sustava se tek uvode i puno su prikladniji za Linux hakere. Također, jedna od metoda - UMSDOS - nije uskladiva s Red Hat distribucijom tako da je uopće ne možete upotrijebiti. O toj ćete temi naći pregršt informacija.

Vidi "Pokretanje DOS programa pod Linuxom", stranica 127.

Zatim biste trebali odrediti broj particija koji vam je potreban i prostor koji one trebaju zauzimati.

**Zahtjevi DOS-a** Želite li pokrenuti DOS, morate ga smjestiti na primarnu particiju. Takva DOS inačica ne zahtijeva puno prostora, već samo onoliko koliko je potrebno za sustavske datoteke, COMMAND.COM, CONFIG.SYS i bilo koje pogonske datoteke koje se koriste za pokretanje sustava. Primjerice, ja sam odredio DOS particiju od 5 MB na mom prvom disku i s nje pokrećem DOS.

Nakon što pokrenete DOS, moći ćete pristupiti bilo kojem proširenom i logičkom pogonu na vašem računalu. Nažalost, bez obzira što Linux može pristupiti DOS datotekama u DOS particijama, DOS ne može pristupiti Linux datotekama u Linux particijama.

**Zahtjevi sustava OS/2** OS/2 2.0 i novije inačice ne zahtijevaju primarnu particiju. Sustav OS/2 se može instalirati i pokrenuti s neke proširene particije. DOS možete instalirati na primarnoj particiji i stvoriti proširene particije za OS/2 i Linux. Potreban prostor za OS/2 ovisi o namjeni i inačici sustava, tako da to morate provjeriti u dokumentaciji sustava. Želite li koristiti program koji upravlja pokretanjem sustava (OS/2 *boot manager*), tada trebate osloboditi 1 MB prostora.

**Zahtjevi Linuxa** Kao što smo prethodno objasnili, Linux sprema datoteke u sustave datoteka koji se nalaze na različitim particijama (zbog sigurnosnih razloga). Linux zahtijeva jednu particiju za svaki sustav datoteka. Idući se zahtjev odnosi na *swap* particiju. Linux, kao i većina operativnih sustava koji prostor na disku koriste kao proširenje memorije (što se naziva *virtualna memorija*), traže *swap* datoteku ili *swap* particiju. Linux obično koristi *swap* particiju.

Veličina *swap* particije ovisi o količini fizičke memorije (RAM-a) na vašem računalu. Pravilo je da *swap* particija treba biti dvostruko veća od RAM-a. To znači da ako imate 8 MB RAM-a, trebate stvoriti *swap* particiju veličine 16 MB. Ako vaše računalo ima 4 MB RAM-a ili čak manje, tada morate aktivirati *swap* particiju.

Linux *swap* particije mogu biti veličine do 128 MB, tako da ćete ukoliko vam treba još prostora, morati stvoriti više *swap* particija. Stoga, ukoliko imate Linux sustav koji zahtijeva dvije particije za Linux (jednu za sustavske datoteke i jednu za korisničke datoteke) plus *swap* particiju, tada trebate definirati dvije Linux particije i jednu *swap* particiju veličine 32 MB.

## Preparticioniranje DOS pogona

U ovom odlomku smo pretpostavili da vam je potrebno preparticioniranje DOS pogona. Najprije pokrenite naredbu FDISK tako da u DOS *prompt* upišete **fdisk**. Pojavit će se ekran FDISK Options (vidi sliku 2.1).



Pomoću ekrana FDISK Options možete potražiti postojeće particije, stvoriti nove i obrisati stare particije.



Ekran koji vidite na slici 2.1 može izgledati i nešto drukčije, ovisno o inačici MS-DOS-a koju koristite. Odaberite opciju 4, Display Partition Information. Pojavit će se ekran Display Partition Information koji je prikazan na slici 2.2. Zapišite podatke s ekrana,

jer će vam trebati podaci o postojećoj particiji ukoliko odlučite prekinuti instalaciju Linuxa i vratiti stari operativni sustav koji ste imali prije početka instalacije.

#### Slika 2.2 U MS-DOS-u 6.x, na ekranu Display Partition Information ćete vidjeti podatke o postojećoj particiji.



#### Alternativa za preparticioniranje tvrdog diska

Možda nije potrebno preparticionirati tvrdi disk, unatoč činjenici da ćete tako najkvalitetnije instalirati Linux. Možete upotrijebiti program FIPS za preparticioniranje tvrdog diska bez gubitaka podataka na njemu. FIPS je kratica za First non-destructive Interactive Partition Splitting (Prvo nedestruktivno dijeljenje particija). Program je razvio Arno Schaefer u sklopu Linux projekta. FIPS se koristi za premještanja unutar DOS particija kako bi se napravilo mjesta za Linux particije.

Kompletne upute za korištenje programa FIPS potražite u dokumentu fips.doc koji se uvijek nalazi u direktoriju u kojem i sam program. Taj program može pomoći samo u slučaju da imate dovoljno slobodnog prostora na disku, a u protivnom ćete trebati obrisati nepotrebne datoteke ili koristiti prethodno opisani postupak za preparticioniranje diska.

Kod Slacware Linuxa, koristeći UMSDOS, možete sustav instalirati na istoj particiji na kojoj se nalazi DOS. UMSDOS omogućava Linuxu da se nalazi na istoj particiji s DOS-om, a vama omogućava da stvorite Linux korijenski (root) sustav datoteka u sklopu postojećeg DOS direktorija. Nažalost UMSDOS ne možete koristiti u Red Hat distribuciji.

**Brisanje particija** Nažalost, program FDISK vam ne omogućava jednostavnu promjenu veličine particije, tako da najprije trebate obrisati particiju a zatim je ponovno dodati (i odabrati željenu veličinu). S ekrana FDISK Options odaberite opciju 3, Delete Partition or Logical DOS Drive, kojom brišete odabrane particije. Pojavit će se ekran Delete Partition or Logical DOS Drive (vidi sliku 2.3).

Odaberite odgovarajuću opciju s obzirom na vrstu particije koju brišete, primjerice primarnu DOS particiju. Opcija broj 1, Delete Primary DOS Partition, omogućava brisanje primarnih DOS particija.

Odaberete li opciju 1, pojavit će se ekran Delete Primary DOS Partition (vidi sliku 2.4). U tom ekranu trebate unijeti ime particije i zatim potvrditi brisanje particije. S obzirom da će svi podaci na odabranoj particiji biti uništeni, program FDISK traži apsolutnu potvrdu da želite brisati primarnu DOS particiju.

#### Slika 2.3

Pomoću ekrana Delete DOS partition možete obrisati određenu particiju ili logički pogon.



Slika 2.4 MS-DOS vas upozorava da ćete obrisati primarnu DOS particiju.

Partition Status Type Volume Label Mbytes System Usage C: 1 A PRI DOS OPUS_DOS 5 FAT12 42 2 Non-DOS 8 62 3 Non-DOS 376 1062 4 Non-DOS 114 962 Iotal disk space is 127 Mbytes (1 Mbyte = 1048576 bytes)	Current fi	ked disk	drive: 1				
Total disk space is 127 Mbytes (1 Mbyte = 1048576 bytes)	Partition C: 1 2 3 4	Status A	Type PRI DOS Non-DOS Non-DOS Non-DOS Non-DOS	Volume Label OPUS_DOS	Mbytes 5 8 376 114	System FAT12	Usage 4% 6% 100% 90%
	Total disk	space i	s 127 Mb	ytes (1 Mbyte	= 104857	6 bytes)	
	WARNING! D	ata in ti	he delete	d Primary DOS	Partitio	n_will be	lost.

**Dodavanje particija** Nakon što obrišete sve potrebne particije morate dodati odgovarajuće particije za svoj DOS sustav tako što ćete odabrati opciju Create a DOS Partition koja se nalazi na ekranu FDISK Options. Slika 2.5 pokazuje ekran Create a DOS Partition or Logical DOS Drive.

#### Slika 2.5

Većina operativnih sustava zahtijevaju primarnu aktivnu particiju za pravilno podizanje sustava.

Create DOS Partition or Logical DOS Drive
Current fixed disk drive: 1
Choose one of the following:
1. Create Primary DOS Partition 2. Create Extended DOS Partition 3. Create Logical DOS Drive(s) in the Extended DOS Partition
Enter choice: [1]
Press Esc to return to FDISK Options

BILJEŠKA Ne možete dodati Linux ili OS/2 particije pomoću DOS programa FDISK. Podjela tvrdog diska na particije je objašnjena u odlomku "Uporaba Linux programa fdisk".

Sav raspoloživ prostor na disku je na raspolaganju particiji koja postaje aktivna particija (izvorno podešenje naredbe FDISK).

Slika 2.6	
Sav prostor na disku možete pretvoriti u jednu particiju ili ga podijeliti na nekoliko particija.	Create Primary DOS Partition Current fixed disk drive: 1 Do you wish to use the maximum available size for a Primary DOS Partition and make the partition active (Y/N)? [Y]
	Purse For to usture to PDIOK Antions

Izraz aktivna particija označava particiju s koje se pokreće sustav. Da biste pokrenuli DOS, morate najprije odrediti primarnu particiju kao aktivnu. Odaberite N (ne) kako biste mogli odrediti točnu količinu prostora na disku koji će tvoriti DOS particiju. Ako odgovorite negativno na pitanje koje je prikazano na slici 2.6, pojavit će se ekran Specify Disk Space for the Partition. Odredite veličinu predviđenog prostora za DOS particiju u megabajtima ili u postocima slobodnog prostora. Nakon toga pritisnite tipku <Return>.

Zatim tu particiju trebate podesiti da bude aktivna. S ekrana FDISK Options odaberite opciju 2, Set Active Partition i jednostavno slijedite upute na ekranu Set Active.

**Formatiranje particije** Nakon što preparticionirate svoj tvrdi disk, novu je particiju potrebno pripremiti za sustav DOS i obnoviti odgovarajuće datoteke u DOS particiji. Podignite sustav pomoću boot diskete koju ste prethodno pripremili. Zatim formatirajte odgovarajući pogon i prenesite sustavske datoteke koristeći ovu DOS naredbu:

#### format c: /s

> Nakon što ste formatirali particiju možete prenijeti kopiju datoteka na novi pogon. Upamtite, ako ste smanjili veličinu particije, tada neće sve datoteke stati na novi pogon. Možda ćete preostale datoteke (koje ne stanu taj pogon) trebati kopirati na druge DOS pogone ili particije.

## Uporaba Linux programa fdisk

U fdisk promptu utipkajte m da biste dobili popis naredbi. Tablica 2.6 sadrži popis naredbi koje su vam na raspolaganju.

Tablica 2.6	Linux naredbe u programu fdisk
Naredba	Opis
a	Uključivanje/isključivanje zastavica za podizanje sustava
С	Uključivanje/isključivanje zastavica za DOS-uskladivost
d	Briše particiju
1	Popis poznatih vrsti particija
m	Prikaz naredbi
n	Dodaje novu particiju
р	Prikaz tablice particije
q	Izlazak bez spremanja promjena
t	Promjena identifikacijskog broja sustava particija
u	Promjena jedinica za prikaz/unos
V	Potvrda tablice particije
W	Zapis tablice na disk i izlazak iz programa
Х	Omogućuje dodatne funkcije (samo za stručnjake)

Za početak podjele diska na particije pokrenite naredbu p (pritisnite tipku i zatim <Return>) tako da se na ekranu pojavi tablica postojeće particije, koja bi trebala prikazati pogon koji ste prethodno podijelili na particije pomoću DOS programa fdisk. Ispis 2.1 prikazuje mogući ispis naredbe p.

Ispis 2.1	Primje	r tablice p	articije					
Disk /dev/ha Units = cyli	d: 15 h nders c	neads, 17 of 255 *	sector 512 byt	s, 1024 es	cylinders	5		
Device	Boot	Begin	Start	End	Blocks	Id	System	
/dev/hda1	*	1	1	41	5219	1	DOS 12-bit	FAT
/dev/hda2		1024	1024	4040	384667+	51	Novell?	
Partition 2	Partition 2 has different phisical/logical endings:							
phys=(967,	14,17)	Logica	1=(4096	5,14.17	)			

**BILJEŠKA** Na vašem računalu ispis može izgledati nešto drukčije od ispisa 2.1 jer se razlikuju vrijednosti svake vrste pogona i particija koje su već definirane na tom pogonu.

U ispisu 2.1 možete vidjeti da su već definirane različite particije, početak i kraj particije i veličina blokova. U ispisu možete uočiti i vrstu particije. Tablica 2.7 prikazuje sve vrste particija koje možete definirati pomoću Linux programa fdisk. Najčešće ćete koristiti 83-Linux Native i 82-Linux Swap. Sličan ispis možete dobiti i ako pokrenete naredbu *l*.

Tablica 2.7   Vrste Linux particija	
Referentni broj	Vrsta
0	prazna
1	DOS 12-bit FAT
2	XENIX root
3	XENIX usr
4	DOS 16-bit<32M
5	proširena
6	DOS 16-bit>=32M
7	OS/2 HPFS
8	AIX
9	AIX za pokretanje sustava
a	OS/2 Boot Manager
40	Venix 80286
51	Novell?
52	Microport
63	GNU HURD
64	Novell
75	PC/IX
80	Old MINIX
81	MINIX/Linux
82	Linux Swap
83	Linux Native
93	Amoeba
94	Amoeba BBT
a5	BSD/386
b7	BSDI fs
b8	BSDI swap
c7	Syrinx
db	CP/M

nastavlja se
Tablica 2.7 Nas	stavak
Referentni broj	Vrsta
e1	DOS access
e3	DOS R/O
f2	DOS secondary
ff	BBT

U ispisu 2.1, Linux na dnu ekrana daje napomenu o različitim fizičkim i logičkim završecima. Ta razlika postoji zbog sustava na kojem je pisano ovo poglavlje – particija koja sadrži D pogon (DOS) nije mijenjana, dok je C pogon preparticioniran na manju veličinu kako bi se načinilo mjesto za Linux. Zbog toga postoji prostor između C i D pogona na kojem će biti stvorene potrebne Linux particije.

U ispisu 2.1 su vrlo važni brojevi u stupcima begin, start i end i njih biste trebali zapisati na papir, jer će vam kasnije trebati kod određivanja potrebne veličine particija.

### Dodavanje potrebnih particija

Kako ste preparticionirali pogon za DOS, ne trebate brisati niti jednu particiju za Linux. Potrebno je samo dodavati particije. Da biste dodali particiju, pokrenite naredbu n i na ekranu će se pojaviti ovo:

```
Command Action
e extended
p primary (1 -4)
```

Pritisnite tipke i <Return>, a program fdisk će tražiti da unesete broj particije. Unesite potreban podatak i pritisnite tipku <Return>. Ako odaberete broj particije koja se trenutno koristi, program će vas na to upozoriti i tražiti da obrišete particiju prije nego je pokušate dodati u tablicu particija. U tom primjeru unesite broj 3 da biste dodali treću primarnu particiju /dev/hda3.

Zatim će program zatražiti da unesete lokaciju prvog cilindra. Obično je to prvi slobodni cilindar. U osnovi program fdisk prikazuje raspon koji ste odabrali, primjerice:

```
First cylinder (42 - 1024)
```

Uočite da prva particija završava na 41. cilindru i da iduća particija počinje na 1024. cilindru. Iduću particiju možete započeti bilo gdje između 42. i 1024. cilindra. Dobro bi bilo da particiju ne postavite bilo gdje na disku već da odaberete iduću slobodnu lokaciju (u ovom slučaju 42. cilindar). Upišite broj 42 i pritisnite tipku <Return>.

BILJEŠKA Linux može imati problema kod pokretanja s particija čiji je početak iznad 1024. cilindra.

Ako stvarate Linux particiju samo u tom rasponu, tada Linux morate pokrenuti s diskete. U nastavku ovog poglavlja ćete naučiti kako se stvara *boot* disketa sustava (razlikuje se od *boot* diskete za instalaciju). Razlika je u tome što pokretanje s diskete traje nešto duže od pokretanja Linuxa s tvrdog diska. Nakon toga program fdisk traži da odredite koliko prostora želite dodijeliti toj particiji. Veličinu potrebnog prostora možete izraziti u cilindrima ili u broju bajtova (+size) ili MB (+sizeM). Kako biste već trebali znati približnu veličinu prostora koji vam je potreban za *swap* datoteku, najprije definirajte *swap* particiju, a potom ostatak prostora na disku ostavite za ostale particije. Tako, primjera radi, ukoliko vaše računalo ima 8MB RAM-a,

pa trebate odrediti particiju veličine 16 MB, upišite sljedeće:

Last cylinder or +size or +sizeM or +sizeK (42-1023): +16M

Nakon toga biste trebali pokrenuti naredbu p i potražiti novu particiju koju ste definirali. U ovom primjeru bi nova tablica particije trebala ovako izgledati:

Disk /dev/had: 15 heads, 17 sectors, 1024 cylinders Units=cylinders of 255 \* 512 bytes System Device Boot Begin Start End Blocks Ιd /dev/hda1 \* 1 1 41 5219 1 DOS 12-bit FAT /dev/hda2 1024 1024 4040 384667+ 51 Novell? Partition 2 has different phisical/logical endings: phys=(967, 14,17) Logical=(4039,14.17) 42 42 170 16447 +83 Linux native /dev/hda3 Izvorno, program fdisk stvara novu particiju tipa Linux Native. Da biste stvorili swap

particiju, potrebno je pokrenuti naredbu *t*. Utipkajte *t* i zatim unesite broj particije koju želite promijeniti (u našem primjeru je to broj 3). Program fdisk traži da zatim unesete heksadecimalnu vrijednost koja označava vrstu particije (pogledajte tablicu 2.7). Ako tablicu nemate pri ruci, upišite broj **1** i dobit ćete popis oznaka. S obzirom da želite *swap* particiju, trebate unijeti broj **82**.

Kao što možete vidjeti, program *fdisk* će vas izvijestiti o novoj vrsti particije, ali možete pokrenuti i naredbu *p* da biste provjerili da je particija 3 postala Linux *swap* particija.

Nakon toga možete dodavati Linux particije. Primjerice, dodajte samo jednu particiju. No ako iz bilo kojeg razloga želite više particija, to možete odmah učiniti. Da biste dodali particiju, pritisnite tipku <n>, odredite p za drugu primarnu particiju i zatim odaberite broj te particije (4). Da biste izbjegli "ispreplitanje" raznih particija na disku, započnite novu particiju na kraju prethodne, na 171. cilindru. S obzirom da želite ostatak prostora koristiti za Linux particije, tada odredite broj zadnjeg cilindra umjesto točnog broja bajtova. Upišite **1023**, kao što je pokazano u nastavku:

```
Command (m for help): n

Command action

e extended

p primary particion (1 - 4)

p

Partition number (1 - 4): 4

First cylinder (171 - 1024): 171

Last cylinder or +size or +sizeM or sizeK (171 - 1023): 1023

Sada pomoću naredbe p možete provjeriti nove particije. Ako želite unijeti još neke

promjene, to možete odmah učiniti.
```

Ako ste zadovoljni s izgledom svojih particija, tada možete pomoću naredbe w zapisati tablicu particija na tvrdi disk. Nijedna od promjena koje ste načinili nije trajna sve dok ne pokrenete naredbu w, tako da, ukoliko mislite da ste načinili neke pogrešne promjene, možete upotrijebiti naredbu q i izaći bez spremanja tablice particija. Ako upotrijebite naredbu w, Linux će vas obavijestiti da je zapamćena tablica particija i zatim će se disk preurediti tako da odgovara novoj tablici particija. Ako se vaš sustav u tom trenutku zaglavi, ponovno ga pokrenite pomoću instalacijske diskete dok ne ugledate # prompt.

### **UPOZORENJE**

Nemojte koristiti Linux program fdisk za stvaranje ili mijenjanje particija za druge operativne sustave, jer se može dogoditi da vam tvrdi disk bude neupotrebljiv za oba operativna sustava.

# Otklanjanje problema

Nakon što ste pokrenuli računalo, trebao bi se na ekranu pojaviti LILO *prompt*. Provjerite možete li pokrenuti svoj stari operativni sustav (ukoliko ste ga sačuvali na tvrdom disku). Ako je DOS vaš stari operativni sustav, pritisnite tipku <Shift> i zatim upišite kraticu koju ste koristili za identifikaciju DOS particije. Upišete li krivo, tada pritisnite tipku <Tab> da biste vidjeli popis važećih operativnih sustava. Imate li i nadalje problema, umetnite DOS *boot* disketu i ponovno pokrenite sustav.

Trebali biste računalo podići pomoću *boot* diskete. Nakon što pokrenete sustav pod DOS-om, isprobajte Linux *boot* disketu koju ste stvorili tijekom instalacije (a ne instalacijsku disketu čitavog sustava). Ukoliko ta *boot* disketa ne radi, možda ćete trebati ponovno instalirati Linux. Potencijalni problemi se mogu javiti kod jezgri (kernela) i hardvera. Prije nego počnete s radom, provjerite je li vaš hardver odgovarajući. Ako ste tijekom instalacije zapisivali određene podatke, onda provjerite koju ste jezgru predvidjeli za vaš hardver. Provjerite podržava li Linux vaš hardver.

U nastavku su dati odgovori na najčešće probleme koji se nalaze na Web lokaciji tvrtke Red Hat.

### P: Mogu li koristiti tvrdi disk koji ima vi{e od 1023 cilindra?

**O:** Gadno pitanje. Odgovor je: da, ali ne za pokretanje Linuxa. Linux možete instalirati na particije koje imaju više od 1023 cilindara, ali za pokretanje Linuxa, *root* direktorij i posebno direktorij /boot moraju biti instalirani na prvom tvrdom disku s manje od 1024 cilindra.

### P: Kako u prompt mogu dodati argumente za LILO?

**O:** Neke hardverske komponente zahtijevaju te dodatne parametre prije nego jezgra prepozna hardver. Postupak možete pojednostaviti ako uredite datoteku /etc/lilo.conf tako da se automatski zadaju potrebni parametri ili ako tijekom pokretanja sustava sami upisujete parametre. Potražite LILO HOWTO i naći ćete još primjera LILO parametara.

### P: Za{to LILO zapinje za LI?

**O:** To je simptom problema s cilindirma koji je prethodno opisan. Ukoliko ste instalirali pokretanje sustava na particiji s više od 1023 cilindra, LILO neće moći pokrenuti sustav. Sustav pokušajte pokrenuti s diskete koju ste načinili tijekom instalacije ili pokušajte preparticionirati tvrdi disk i ponovo instalirati Linux.

### P: Tijekom instalacije sustav ne mo`e na}i SCSI karticu.

O: Da biste otklonili problem, trebate dodati argument tijekom podizanja sustava:

```
LILO: linux qlogicfas=0x230,11,5
```

Ovu opciju možete trajno podesiti tako da je ne trebate ponovno unositi. Pogledajte LILO opciju append koja se nalazi na lilo.conf stranici.

### P: Kako mogu deinstalirati LILO?

**O:** Želite li deinstalirati LILO i ponovno instalirati originalni *boot record*, pokušajte pokrenuti ovu naredbu

lilo -u /dev/had

koja predstavlja *boot record* na prvom IDE pogonu. Parametri ovise o vašem računalu. Primjerice, ukoliko je prvi tvrdi disk SCSI disk, tada trebate koristiti naredbu /dev/sda.

### P: Mogu li koristiti LILO i Windowse 95 na istom ra~unalu?

**O:** Da. Najprije instalirajte Windowse 95, a zatim Linux. Tijekom instalacije zadajte Linuxu da postavi LILO u MBR. Osim toga, možete upotrijebiti i komercijalni program kao što je System Commander.

### P: Kako mogu instalirati CD-ROM?

**O:** Instalacijom Red Hat 5.1 distribucije trebali biste postaviti odgovarajuće unose u datoteku /etc/fstab, kao što je prikazano u nastavku:

```
#
# /etc/fstab
#
# You should be using fstool (control-panel) to edit this!
#
# <device> <mountpoint>
                        <filesystemtype> <options> <dump>
                                                             <fsckorder>
                                             ext2 defaults 1 1
/dev/sda1
                  /
                                             ext2
/dev/sda5
                   /home
                                                         defaults 1 2
                                             iso9660
/dev/cdrom
                   /mnt/cdrom
                                                         noauto, ro 0 0
/dev/fd0
                   /mnt/floopy
                                             ext2
                                                         noauto O O
/dev/sda6
                   /var
                                             ext2
                                                         defaults 1 2
/dev/sda2
                                                         0 0 0
                    none
                                              ignore
none
                    /proc
                                              proc
                                                         defaults
/dev/sda7
                    none
                                              swap
                                                         SW
```

Uočite uporabu podešenja *nonauto* za unos cdroma. Bez tog podešenja Linux će prilikom pokretanja sam pokušati stvoriti CD-ROM, što i nije problem osim ako u pogon niste umetnuli CD. Ukoliko u datoteci fstab ne postoji niti jedan unos, tada možete ili urediti datoteku /etc/fstab ili upotrijebiti alat XWindow Control Panel za dodavanje potrebnih informacija. Također provjerite postoji li /mnt/cdrom. Ako je unos ispravan, tada na to mjesto možete prijeći pomoću naredbe cd i pokrenuti sljedeće naredbe:

```
cd /mnt
mount cdrom
```

# P: Koristim Red Hat 5.0 distribuciju nadogra | enu s ld.so RPM paketom, ali mi programi libc5 jo{ uvijek javljaju gre{ku. [to ne valja?

O: Prekid rada programa libc5 može prouzročiti nekoliko stvari.

Prije ili nakon nadogradnje druga inačica programa može biti instalirana na način koji nije predviđen ili programi libc5 mogu biti postavljeni na mjesto koje stvara probleme.

Da biste to otkrili, pokrenite ovu naredbu:

rpm -qa | grep libc

Na ekranu bi se trebao pojaviti ovaj ispis:

```
glibc-devel-2.0.5c-12
libc-5.3.12-24
glibc-debug-2.0.5c-12
rmp-2.4.10-1glibc
rmp-devel-2.4.10-1glibc
glibc-profile-2.0.5c-12
glibc-2.05c-12
```

Vidite li retke u kojima piše nešto nalik na *libc-debug-5.3.12-18* ili *libc-5.4.44-2*, morat ćete ukloniti te pakete (pomoću naredbe rpm –e libc-debug) i pokrenuti naredbu ldconfig –v.

Vaša datoteka /etc/ld.co.conf je promijenjena i sadrži optimalna podešenja. Za optimalno učitavanje datoteku /etc/ld.co.conf trebate ovako podesiti:

```
/usr/i486-linuxaout/lib
/usr/i486-linux-libc5/lib
/usr/openwin/lib
/usr/X11R6/lib
```

### P: Neke moje starije aplikacije imaju pode{eno neto~no vrijeme.

**O:** Neki programi libc5 traže datoteku /usr/lib/zoneinfo. Možete programe prekompilirati za libc6 ili omogućiti vezu pomoću iduće naredbe i stvar će funkcionirati:

ln -s ../share/zoneinfo /usr/lib/zoneinfo

# P: Instalirao sam sve najnovije nadogradnje, ali moji programi jo{ uvijek pokazuju neto~no vrijeme.

**O:** Ako ste instalirali sve najnovije nadogradnje sustava a vaši programi i dalje pokazuju netočno vrijeme, provjerite podešenja u datoteci /etc/sysconfig/clock, koja vjerojatno ovako izgledaju:

UTC=true ARC=false To znači da će Linux pretpostaviti da je vaš BIOS sat podešen na vremensku zonu GMT. Vjerojatnije je da bi sat trebao biti podešen na vašu lokalnu vremensku zonu, tako da prethodni redak trebate promijeniti tako da izgleda ovako:

UTC=false

# P: Tijekom podizanja sustava pojavljuje se poruka u kojoj pi{e da imam nepoznati PCI hardver. [to to zna~i?

**O:** Poruka "unknown PCI device" može se pojaviti iz više razloga. Prvi i najbezbolniji je da PCI nije na razumljiv način odgovorio na Linuxov upit, a Linux je nastavio s podizanjem sustava. Mnogo se češće događa da se sustav zaglavljuje i ne može nastaviti učitavanje.

S obzirom da se radi o hardverskom problemu u jezgri, Red Hat vas jedino može uputiti na onoga tko je stvorio taj dio jezgre. Ta bi vam osoba mogla objasniti što se događa (možda ćete joj trebati objasniti koji hardver koristite u svom sustavu, kako se ubuduće te stvari više ne bi događale). Kontakt osobu potražite na adresi:

### linux-pcisupport@cck.uni-kl.de.

U poruku dodajte i datoteku

/proc/pci

koja sadrži točan opis hardvera koji koristite. Pokušajte saznati koji uređaj nije prepoznat. Možda se radi o matičnoj ploči, vezi PCI-CPU ili vezi PCI-ISA. Ukoliko ne možete naći pravu informaciju u uputama koje ste dobili uz računalo, pokušajte pročitati podatke na čipu koji se nalazi na ploči.

### P: Linux nije uo~io moju mre`nu karticu koja je kompatibilna s NE2000.

**O:** Uočeno je da neke kartice NE2000 koje rade sa starijim jezgrama ne mogu raditi s inačicama 2.0.x. Za neke kartice trebate provesti sljedeću proceduru kako bi one radile.

Karticu pokušajte podesiti na sljedeći način:

```
insmod 8390
insmod ne io=OXXXX irg=Y
```

(Napomena: Zamijenite XXXX i Y s vašom IO adresom i IRQ. Najčešće vrijednosti za IO adrese su 0x300 i 0x310. IRQ može biti bilo koji).

Nakon toga pomoću naredbi *ifconfig* ili netcfg podesite karticu. Ponekad, čak i ako sustav prepozna karticu, ona neće moći prenositi pakete putem TCP/IP protokola.

Ako su gornja podešenja dobra i rade, dodajte u datoteku /etc/conf.modules tako da sve skupa izgleda ovako:

```
alias eth0 8390
alias eth0 ne
options eth0 io=0xXXX irg=Y
```

# P: Instalirao sam Linux i izgleda kao da se sustav po~inje pokretati, no kada do | e do ne~ega {to se naziva *sendmail*, stroj se zaglavi. [to ste to doga | a i {to bi trebalo u~initi?

**O:** Ako se nakon instalacije računalo zaglavljuje kada dođe do određenog procesa kao što je *sendmail, apache* ili SMB, vjerojatno se radi o problemu s mrežom. Najčešći je uzrok taj što Linux ne može naći ime računala (ako ste mrežu podesili da radi s imenima računala). Računalo će se zaustaviti i čekati određeno vrijeme i zatim će se eventualno pojaviti *login prompt*. Kada dobijete prompt, prijavite se i provjerite uobičajene izvore problema.

Ako ste izravno spojeni na mrežu s DNS poslužiteljem, provjerite nalaze li se u datoteci etc/resolv.conf ispravne vrijednosti za vaš DNS poslužitelj. Kod svog administratora sustava provjerite ispravne vrijednosti.

Ako koristite Linux na mreži bez DNS poslužitelja, tada trebate urediti datoteku /etc/hosts tako da sadrži ime *hosta* i IP adresu radi ispravnih pretraživanja. Datoteka /etc/hosts/ bi otprilike ovako trebala izgledati (izraz mymachine treba zamijeniti imenom računala):

127.0.0.1 localhost localhost.localdomain 192.168.200.1 mymachine mymachine.mynetwork.net

# Odavde...

Sada kada ste pokrenuli sustav koji radi možete pročitati iduća poglavlja i saznati još neke informacije o Linuxu:

- 5. poglavlje, "Pokretanje Linux programa" omogućuje vam da ubrzate različite programe koje ste upravo instalirali.
- 6. poglavlje, "Nadogradnja i instaliranje softvera pomoću RPM-a" sadrži informacije kako ponovno instalirati pakete koje ste trebali ukloniti tijekom početnog podešavanja Linux sustava.
- 21. poglavlje, "Instaliranje XWindowsa" nudi vam informacije za instaliranje XWindowsa ukoliko naiđete na probleme kod instaliranja Red Hat distribucije. Iako su XWindowsi predviđeni za Slackware distribuciju, softver XFree86 je isti za bilo koju inačicu Linuxa.
- 22. poglavlje, "Uporaba XWindowsa" je zanimljivo ako ste instalirali XWindowse.
- 31. poglavlje, "Kretanje Internetom pomoću World Wide Weba" nudi vam osnovne upute kako ćete dospjeti do Interneta.

# POGLAVLJE **3**

# **Instaliranje Red Hata**

(Jack Tackett)

# U ovom poglavlju

Početak instalacije Različiti načini instalacije Instalacijaoperativnog sustava Linux Podešavanje sustava Podešavanje mreže Instalacija Red Hata na sustavima DEC Alpha Vraćanje na početak

Otklanjanje problema 86

U ovom ćete poglavlju naći potrebne informacije za instaliranje Red Hat Linux distribucije. Upamtite, Red Hat Linux nije komercijalni proizvod i može vam se dogoditi da naiđete na određene probleme. Iako ova knjiga sadrži dosta informacija, možda ćete trebati potražiti dodatne izvore kao što su razne HOWTO datoteke koje možete pronaći na niz adresa na Internetu. No, Red Hat distribuciju je vrlo jednostavno instalirati, stoga se nemojte bojati.

BILJEŠKA Osnovne podatke o hardverskim zahtjevima, te particioniranju diska potražite u 2. poglavlju.

# Početak instalacije

Da biste započeli instalaciju Linuxa potrebne su vam jedna ili dvije (ovisno o instalaciji) diskete 3,5", 1,44 MB. One će biti upotrijebljene za stvaranje *boot* disketa (disketa za pokretanje sustava) za instalaciju Linuxa.

Zatim trebate osigurati dovoljno prostora na tvrdom disku da biste instalirali Linux. Ukoliko želite instalirati sve što distribucija sadrži, tada će vam trebati oko 300 MB prostora, no možda će vam biti dovoljno i manje prostora, osobito ako ne namjeravate instalirati XWindowse. Da biste odredili potrebnu količinu prostora na disku, potrebno je odlučiti koliko prostora želite odvojiti za korisničke račune, odnosno, odrediti prostor koji ćete ponuditi korisnicima. Ukoliko sustav predviđa samo jednog korisnika, tada je 50 MB više nego dovoljno.

Potom odredite koliko vam treba swap prostora na disku. Ukoliko vaše računalo ima 8 MB RAM-a ili manje od toga, onda vam je potrebno 24 MB *swap* prostora. No, ako imate 16 MB RAM-a, onda *swap* prostor treba biti veličine memorije.

I na kraju trebate predvidjeti oko 50 MB za korijenski (root) direktorij, koji će biti glavni direktorij iz kojeg ćete pristupati svim drugim Linux direktorijima.

Vidi "Standardni direktoriji u Linuxu", stranica 313.

Minimalna instalacija može zauzimati oko 200 MB prostora, dok puna instalacija, s dosta korisničkog prostora, može stati na 500 MB. Tablica 3.1 pokazuje prednosti i nedostatke svakog od načina instalacije.

Tablica 3.1	Za i protiv određenih načina instalacije				
Metoda	Za	Protiv			
CD-ROM	brzo i pouzdano	distribucije brzo zastarijevaju			
FTP	najnoviji softver	nepouzdano i sporo			
	mogućnost pristupa iz čitavog svijeta				
NFS	prikladno ako nemate CD-ROM	nužna mreža i sporo			
SMB	prikladno u MS Windows mreži	nužna mreža i poznavanje Sambe			
Tvrdi disk	koristi se kada nema drugog načina	potrebno puno dodatnog prostora			

**BILJEŠKA** Dio Linuxovog sustava datoteka možete pokrenuti s CD-ROM-a bez instalacije čitavog softvera. Tu odluku možete donijeti tijekom instalacije.

Odlučite li instalirati i podesiti XWindowse (što vam zaista preporučamo), trebali biste također zapisati vrstu čipova koje koristi vaša grafička kartica. Ukoliko imate serijskog miša i modem, zapišite serijske priključke na kojima su spojeni, jer će vam ti podaci trebati tijekom podešavanja sustava.

# Različiti načini instalacije

Autori ove knjige predviđaju da će većina korisnika instalirati Red Hat distribuciju s CD-ROM-a. Ipak, možete upotrijebiti bilo koju od iduće četiri metode za instalaciju Red Hata: s CD-ROM-a, putem NFS-a, FTP-a ili s tvrdog diska.

Da biste sustav instalirali izravno s CD-ROM-a, trebate najprije pokrenuti DOS. U DOS promptu upišite naredbu

[cdrom-drive]:dosutils\autoboot

gdje [cdrom-drive] zamjenjuje naziv pogona CD-ROM-a.

### **UPOZORENJE**

Koristeći ovu metodu obrisat ćete sadržaj na svom tvrdom disku, tako da najprije morate zaštiti sve bitne podatke.

Ukoliko na raspolaganju imate još jednu particiju, tada Linux možete instalirati usporedo svom operativnom sustavu, bez brisanja ostalog sadržaja. Da biste to učinili, potreban vam je CD-ROM, prazna particija na disku i *boot* disketa. U nastavku ćete ovog poglavlja naučiti kako se stvara *boot* disketa, te kako možete preparticionirati svoj tvrdi disk.

Oni koji imaju sustav koji se može pokrenuti s CD-ROM-a (provjerite podešenja u BIOS-u), mogu pokrenuti i instalirati Red Hat s CD-ROM-a.

NFS (Network File System) vam omogućava instalaciju Red Hat distribucije putem mreže. Najprije morate podesiti pogon CD-ROM-a da podržava sustav datoteka ISO-9660 s RockRidge ekstenzijama, a zatim eksportirati sustav datoteka putem NFS-a. Potrebno je znati stazu (*path*) eksportiranog sustava datoteka i IP adresu ili, ako je podešen DNS, poznavati ime sustava.

FTP (File Transfer Protocol) je način prijenosa datoteka putem Interneta (u 30. ćete poglavlju, "Pristup mreži pomoću programa telnet, ftp i r-", naći detaljniji opis FTP-a). Da biste instalirali Linux putem FTP-a, potrebna vam je *boot* disketa i tzv. *dopunska* disketa (koja će biti objašnjena u nastavku poglavlja).

Instalacija Red Hat distribucije s tvrdog diska zahtijeva *boot* i dopunsku disketu, kao i kod instalacije putem FTP-a. Najprije stvorite direktorij pod imenom RedHat. Zatim kopirajte odgovarajući direktorij s CD-ROM-a, kao i sve poddirektorije. To možete učiniti pomoću ovih DOS naredbi:

```
cd \RedHat
xcopy /s e:\RedHat
```

Kod naredbe c<br/>d se podrazumijeva da se nalazite na tvrdom disku, a kod naredbe xcopy da j<br/>eeime pogona CD-ROM-a.

Bez obzira koji način instalacije Linuxa upotrijebite, trebat ćete sakupiti neke informacije.

# Prikupljanje potrebnih informacija

Prije nego započnete s instalacijom, biti će vam potrebne sljedeće informacije o vašem računalu:

- Vrsta grafičke kartice, čipova i monitora
- Serijski priključak koji se koristi za miša
- Serijski priključak koji se koristi za modem
- Ukoliko je vaše računalo spojeno na mrežu, potrebno je znati IP adresu, ime domene, gateway i slično
- Vrstu tvrdog diska i pogona CD-ROM-a te njihove prilagodnike
- Strukturu direktorija koju želite koristiti; primjerice: direktorij /home može biti na posebnom tvrdom disku, a direktorij /var na posebnoj particiji, u odnosu na *swap* datoteku.
- Ime koje želite dati svom sustavu

Ako imate vezu na Internet, tada većinu ovih podataka možete dobiti od administratora mreže ili davatelja Internet usluga.

Ako na istom računalu namjeravate koristiti i neki drugi operativni sustav (kao što su Windowsi 95, Windowsi NT ili OS/2), tada na disku trebate stvoriti nužne particije za te operativne sustave. Obično ćete trebati koristiti softver operativnog sustava za podjelu na particije, jer Linux ne može raditi s drugim vrstama particija.



### NA WEBU

Softver pod imenom System Commander (tvrtka V Communications) omogućuje vam instalaciju i korištenje 32 različita operativna sustava. O tome saznajte nešto više na adresi

### http://www.v-com.com/

Zatim biste trebali pročitati promjene koje su u zadnji trenutak unesene u RedHat distribuciju. Ima više razloga za to, no dva su glavna razloga da se Linux svakodnevno mijenja, a ova knjiga opisuje inačice nakon kojih su se pojavile i nove. U međuvremenu su se pojavili novi materijali ili su otklonjene neke greške.



### NA WEBU

Najnovije materijale potražite na Webu i to na adresi

http://www.redhat.com/errata

Ukoliko ne instalirate sustav s CD-ROM-a, tada je potrebno preparticionirati tvrdi disk kako biste napravili mjesta za Linux. Tu se mogu pojaviti problemi, jer preparticioniranje diska uništava sve podatke na odabranoj particiji. Nakon što napravite mjesta za Linux, potrebno je podići Linux sustav te stvoriti nove particije i sustave datoteka. Obično je potrebno načiniti primarnu particiju za pohranu datoteka i particiju za *swap* datoteku, osobito ako koristite računalo s 8MB RAM-a (ili manje).

**BILJEŠKA** Sustav datoteka je u osnovi dio tvrdog diska posebno formatiran za pohranu određenih vrsta datoteka. UNIX i Linux koriste sustave datoteka za prikaz stabla direktorija. U tome se razlikuju od MS-DOS-a koji poddirektorije postavlja u stablo direktorija na istom logičkom disku. UNIX sustavi koriste format stabla direktorija radi sigurnijeg smještaja poddirektorija na različite pogone. Ukoliko se javi greška na jednom pogonu, tada je potrebno zamijeniti samo podatke s tog pogona.

Vidi "Sustavi datoteka", stranica 266.

Nakon što ste stvorili sustav datoteka, potrebno je instalirati operativni sustav Linux, njegove popratne datoteke i razne pakete s aplikacijama. Da biste instalirali Linux, najprije trebate pokrenuti skraćenu inačicu operativnog sustava tako da stvorite *boot* i dopunsku disketu sa sažetim operativnim sustavom.

### Stvaranje boot i dopunske diskete te sigurnosne kopije sustava

Pomoću programa rawrite trebate stvoriti *boot* i dopunsku disketu. Ovaj program možete naći na odgovarajućem CD-ROM-u u poddirektoriju /dosutils. Da biste stvorili *boot* i dopunsku disketu, potrebne su vam dvije formatirane diskete od kojih je jedna označena kao *boot*, a druga kao *supp. Boot* disketu stavite u disketni pogon i upišite sljedeće:

```
E:\dosutils>rawrite
Enter disk image source file name: e:\images\boot.img
Enter target diskette drive: A:
Please instert a formatted diskette into drive A: and press -ENTER-
```

Ukoliko želite prekinuti proces, jednostavno pritisnite tipku <Ctrl>. Ako naiđete na probleme, pokušajte s novom disketom. Ukoliko se problemi nastave, tada provjerite svoj hardver.

Nakon što stvorite *boot* disketu, trebate kreirati i *dopunsku* disketu. Jednostavno u prethodnoj naredbi upišite supp.img kao ime datoteke.

Zatim biste trebali stvoriti sigurnosnu kopiju sustava. Za razliku od prethodnih inačica Red Hat distribucije, kod inačice 5.1 ne možete pomoću instalacijske diskete podići sustav. Zbog toga je uvedena i disketa za sigurnosnu kopiju sustava (rescue.img). Da biste je stvorili, u prethodnoj proceduri upišite rescue.img kao ime datoteke.

Vidi "Podjela tvrdog diska na particije", 42. stranica

# Instalacija operativnog sustava Linux

Da biste pokrenuli instalaciju Linuxa, najprije postavite boot disketu u disketni pogon te isključite i nakon toga uključite računalo. Ako se na početku pokretanja obavljaju provjere hardvera i BIOS-a, onda ćete na ekranu vidjeti sljedeću poruku:

Wellcome to Red Hat Linux

(Dobrodošli u Red Hat distribuciju Linuxa)

To install or upgrade a system running Red Hat 2.0 or later, press the <ENTER> key. (Za instalaciju ili nadogradnju Red Hat distribucije 2.0 ili novije pritisnite tipku <ENTER>)

To enable expert mode, type expert <ENTER>. Press <F3> for more information about expert mode. (Želite li raditi u načinu rada expert, upišite riječ expert i pritisnite tipku <ENTER>. Više informacija o tom načinu rada potražite pritiskom na tipku <F3>). This disk can no longer be used as a rescue disk. Press <F4> for more information on new rescue disk. (Ovu disketu više ne možete koristiti kao sigurnosnu kopiju sustava. Pritisnite tipku <F4> da biste saznali nešto više o tome). Use the Function Key listed below for more information (F1-Main) (F2-General) (F3-Expert) (F4-Rescue) (F5-Kickstart) (F6-Kernel)

(Navedene funkcijske tipke pritisnite kada želite dobiti dodatne informacije)

Tablica 3.2 sadrži podatke o funkcijskim tipkama i njihovoj namjeni. Obično ćete pritisnuti tipku <ENTER> i nastaviti instalaciju.

Tablica 3.2 Fun	ablica 3.2 Funkcijske tipke koje se koriste tijekom instalacije					
Funkcijska tipka	Opis					
F1	Služi za prikaz glavnog ekrana. Pritiskom na tipku <f1> uvijek se vraćate na taj ekran.</f1>					
F2	Nudi općenite podatke o instalaciji.					
F3	Služi za opis <i>expert</i> načina rada. Linux tijekom instalacije provjerava instalirani hardver i ponekad se može sustav zaglaviti. Ako se to dogodi, morate ući u način rada <i>expert</i> i odabrati hardverske komponente koje koristite u svom sustavu.					
F4	Služi za prikaz uputa namijenjenih stvaranju i uporabi diskete sa sigurnosnom kopijom sustava.					
F5	Red Hat 5.1 vam omogućuje instalaciju sustava pomoću datoteke za podešavanje. Na ovom ćete ekranu naći potrebne infomacije za takav način instalacije.					
F6	Ako ne možete pravilno pokrenuti Linux, možda ćete trebati dodati još neke parametre koji se odnose na jezgru sustava. Na ovom ekranu ćete naći potrebne upute.					

BILJEŠKA Više informacija o parametrima za pokretanje sustava potražite u odgovarajućim direktorijima distribucije (poput /doc/HOWTO/BootPrompt-HOWTO).

Na ekranu će započeti inicijalizacija sustava:

Loading initrd.img.... loading vmlinuz....

Nakon pokretanja sustava ugledat ćete poruku dobrodošlice (vidi sliku 3.1).



Pritisnite tipku <Enter> za nastavak. U idućem ekranu trebate odabrati jezik koji ćete koristiti tijekom instalacije (vidi sliku 3.2).



**BILJEŠKA** Kretanje u okvirima za dijalog je vrlo jednostavno jer instalacijski program nudi podsjetnike u dnu ekrana. Da biste prelazili s jednog elementa (polja) na drugi, pritisnite tipku <Tab> ili <Alt-Tab>. Pomoću razmaknice možete odabrati element s popisa ili potvrditi opciju. Da biste odabrali gumb (obično OK ili Cancel), pritisnite tipku <Enter>. Pomoću tipki sa strelicama možete klizati duž popisa.

U idućem okviru za dijalog (vidi sliku 3.3) trebate odabrati vrstu tipkovnice koju koristite.

SAVJET Za promjenu tipkovnice koristite datoteku /usr/sbin/kbdconfig.

Slika 3.3

instalacije.



Na idućem ekranu trebate odabrati želite li PCMCIA (PC kartica) podršku sustava (vidi sliku 3.4). Odgovor odaberite pomoću tipke <Tab> i zatim pritisnite tipku <Enter>.

Slika 3.4	E Enxterm
Red Hat Linux nudi	Helcome to Red Hat Linux
potporu za PCMCIA (PC)	
kartice.	
	PCMCIA Support
	Do you need POMCIA support2
	Bo gou need richter supporter

Potom trebate odabrati instalacijsku metodu. Odaberite način instalacije i pritisnite tipku <Enter>. Instalacijski program traži da umetnete Red Hat CD-ROM, a potom možete pritisnuti tipku <Enter>.

<Tab>/<Alt-Tab> between elements | <Space> selects | <F12> next screen



BILJEŠKA U nastavku ovog poglavlja podrazumijeva se da ste odabrali instalaciju s CD-ROM-a. Ako ste ipak odabrali neki drugi način instalacije, tada pomoć potražite na Red Hat Web lokaciji http://www.redhat.com.

Nakon toga, instalacijski program nastoji provjeriti pogon CD-ROM-a. Ukoliko provjera uspije, instalacija će se nastaviti, a ako ne, morat ćete odabrati drugi tip CD-ROM-a:

SCSI	Ovaj odabir koristite za SCSI uređaje
Other CD-ROM	Ovu opciju upotrijebite za NON-IDE CD-ROM-ove
	ili one koji su uskladivi sa zvučnim karticama.

Odaberete li opciju Other CD-ROM, pojavit će se okvir za dijalog koji je prikazan na slici 3.6. Odaberite odgovarajuću vrstu pogona i pritisnite tipku <Enter> kako biste nastavili instalaciju.

Slika 3.6 Program za instalaciju Red Hat distribucije traži unos vrste CD- ROM-a koji koristite u svom računalu.	Helcowe to Red Hat Linux
	(Tab)/(Alt-Tab) between elements   <space> selects   <f12> next screen</f12></space>

U grupi Other CD-ROM naći ćete CD-ROM-ove tvrtke Creative Labs (SoundBlaster) i druge CD-ROM-ove za multimediju:

Aztech Cd	Sanyo
Goldstar R420	Sony CDU-31A
Mitsumi	Sony CDU-5xx
Optics Storage 8000	SoundBlaster/Panasonic
Phillips CM206/CM260	

Ovisno o odabiru, instalacijski program može tražiti da upišete još neke parametre, kao što su IRQ ili DMA adrese, ili će program sam odrediti potrebne parametre. Najbolje je da program pokuša sam odrediti parametre.

SAVJET

BILJEŠKA Svaki puta kada instalacijski program provjerava sustav, on se može zaglaviti. Ako se to dogodi, morat ćete ponovo pokrenuti sustav i ponoviti instalaciju. Prikupite potrebne informacije, kao što su IRQ i DMA adrese, i ponovite instalaciju.

Nakon što odredi vrstu CD-ROM-a, sustav će započeti instalaciju. Najprije ćete morati odabrati želite li instalirati novi sustav ili nadograditi postojeći Red Hat sustav. Red Hat 5.1 vrlo jednostavno nadograđuje inačice 2.0 ili novije, no nije lako nadograditi jednu Linux distribuciju na različitu distribuciju. Stoga ako imate prethodnu inačicu distribucije, kao što je Slackware, jednostavno instalirajte novu inačicu i zaboravite onu prije nje. Ukoliko želite nadograditi prethodnu inačicu Red Hata, instalacijski program će spremiti sve postojeće datoteke za podešavanje s dodatkom imenu .rpmsave.

Vidi "Ažuriranje paketa pomoću RPM-a", 152. stranica

Svi postupci tijekom instalacije pohranjeni su u datoteci /tmp/upgrade.log.

Da biste vidjeli što radi program za instalaciju, pritisnite tipku <Alt-F3> i pojavit će se virtualni terminal u kojem možete pratiti sve što se događa.

Potom će instalacijski program provjeravati sve SCSI uređaje. Možda ćete trebati odabrati SCSI prilagodnik, a zatim će se pojaviti okvir za dijalog Configuration u kojem trebate navesti imate li SCSI adaptere. Odaberite odgovarajući gumb i nastavite instalaciju.

Imate li SCSI adapter, tada ćete ugledati okvir za dijalog u kojem možete odabrati neki od ponuđenih SCSI pogonskih programa:

Adaptec 152x	Iomega PPA3 (Parallel port Zip)
Adaptec 1542	NCR 5380
Adaptec 1740	NCR 53c406a
Adaptec 2740, 2840, 2940	NCR 53C810/53C820 PCI
AdvanSys Adapters	Pro Audio Spectrum/Studio 16
Always IN200	Qlogic FAS
Buslogic Adapters	Qlogic ISP
DTC 3180/3280	Seagate ST01/02
EATA DMA Adapters	Trantor T128/T128F/T228
EATA PIO Adapters	UltraStor 14F/34F
Future Domain TMC-885, TMC-950	UltraStor 14F/24F/34F
Future Domain TMC-16x0	Western Digital wd7000



### NA WEBU

Želite li saznati najnovije podatke o tome koje hardverske komponente Red Hat 5.1 distribucija podržava, potražite Red Hat Web lokaciju

### http://www.redhat.com/support/docs/rhl/intel/rh51-hardware-intel.html

Potom trebate particionirati svoje diskove ili odabrati particije koje ste prethodno stvorili. Instalacijski program će prikazati okvir za dijalog Disk Setup koji je prikazan na slici 3.7. Možete odabrati program fdisk ili program s grafičkim sučeljem Disk Druid. Želite li koristiti program fdisk, odaberite gumb fdisk i pritisnite tipku <Enter>, a ako biste radije upotrijebili program Disk Druid, odaberite gumb Disk Druid i potražite odlomak "Uporaba programa Disk Druid".



# Uporaba Linux programa fdisk

U fdisk promptu upišite m da biste dobili popis naredbi. Tablica 3.3 sadrži popis naredbi koje su vam na raspolaganju.

### UPOZORENJE

Budite pažljivi kod uporabe programa *fdisk* jer se on razlikuje od programa s istim imenom koji radi u drugom operativnom sustavu kao što su MS-DOS, Windows 98/95 i OS/2. Te programe ne možete koristiti kao zamjenu! Primjerice, ne možete koristiti Linuxov program *fdisk* za mijenjanje particije u DOS particiju. Iako program *fdisk* možete koristiti za stvaranje particija, obavezno morate odabrati odgovarajuću inačicu za određeni operativni sustav kako biste mogli podesiti vrste datoteka.

Tablica 3.3	Linux naredbe u programu fdisk		
Naredba	Opis		
a	Uključivanje/isključivanje zastavica za podizanje sustava		
С	Uključivanje/isključivanje zastavica za DOS-uskladivost		
d	Briše particiju		

nastavlja se

Tablica 3.3	Nastavak
Naredba	Opis
1	Popis poznatih vrsti particija
m	Prikaz naredbi
n	Dodaje novu particiju
р	Prikaz tablice particije
q	Izlazak bez spremanja promjena
t	Promjena identifikacijskog broja sustava particija
u	Promjena jedinica za prikaz/unos
V	Potvrda tablice particije
W	Zapis tablice na disk i izlazak iz programa
Х	Omogućuje dodatne funkcije (samo za stručnjake)

Za početak podjele diska na particije pokrenite naredbu p (pritisnite tipke p > i zatim Return>) tako da se na ekranu pojavi tablica postojeće particije, koja bi trebala prikazati pogon koji ste prethodno podijelili na particije pomoću DOS programa fdisk. Ispis 3.1 prikazuje mogući ispis naredbe p.

Ispis 3.1	Primje	r tablice	particije					
Disk /dev/h Units = cyl	ad: 15   inders (	neads, 11 of 255 *	7 sectors 512 byte	, 1024 s	cylinders			
Device /dev/hda1 /dev/hda2	Boot *	Begin 1 1024	Start 1 1024	End 41 4040	Blocks 5219 384667+	Id 1 51	System DOS 12-bit Novell?	FAT
Partition 2 has different phisical/logical endings: phys=(967, 14,17) Logical=(4096,14.17)								

BILJEŠKA Na vašem računalu ispis može izgledati nešto drukčije od ispisa 3.1 jer se razlikuju vrijednosti svake vrste pogona i particija koje su već definirane na tom pogonu.

U ispisu 3.1 možete vidjeti da su već definirane različite particije, početak i kraj particije i veličina blokova. U ispisu možete uočiti i vrstu particije. Tablica 3.4 prikazuje sve vrste particija koje možete definirati pomoću Linux programa fdisk. Najčešće ćete koristiti 83-Linux Native i 82-Linux Swap. Sličan ispis možete dobiti i ako pokrenete naredbu *l*.

Tablica 3.4         Vrste Linux particija			
Referentni broj	Vrsta		
0	prazna		
1	DOS 12-bit FAT		
2	XENIX root		
3	XENIX usr		
4	DOS 16-bit<32M		
5	proširena		
6	DOS 16-bit>=32M		
7	OS/2 HPFS		
8	AIX		
9	AIX za pokretanje sustava		
a	OS/2 Boot Manager		
40	Venix 80286		
51	Novell?		
52	Microport		
63	GNU HURD		
64	Novell		
75	PC/IX		
80	Old MINIX		
81	MINIX/Linux		
82	Linux Swap		
83	Linux Native		
93	Amoeba		
94	Amoeba BBT		
a5	BSD/386		
b7	BSDI fs		
b8	BSDI swap		
c7	Syrinx		
db	CP/M		

nastavlja se

Tablica 3.4 Nas	stavak
Referentni broj	Vrsta
el	DOS access
e3	DOS R/O
f2	DOS secondary
ff	BBT

U ispisu 3.1, Linux na dnu ekrana daje napomenu o različitim fizičkim i logičkim završecima. Ta razlika postoji zbog sustava koji se koristi za pisanje ovog poglavlja.

Particija koja sadrži D pogon (DOS) nije mijenjana, dok je C pogon preparticioniran na manju veličinu kako bi se načinilo mjesto za Linux. Zbog toga postoji prostor između C i D pogona, na kojem će biti stvorene potrebne Linux particije.

U ispisu 3.1 su vrlo važni brojevi u stupcima begin, start i end i njih biste trebali zapisati na papir, jer će vam kasnije trebati kod određivanja potrebne veličine particija.

### Dodavanje potrebnih particija

S obzirom da ste preparticionirali pogon za DOS, ne trebate brisati niti jednu particiju za Linux. Potrebno je samo dodavati particije. Da biste dodali particiju, pokrenite naredbu *n*, i na ekranu će se pojaviti ovo:

```
Command Action
e extended
p primary (1 -4)
```

Pritisnite tipke i <Return> a program fdisk će tražiti da unesete broj particije. Unesite potreban podatak i pritisnite tipku <Return>. Ako odaberete broj particije koja se trenutno koristi, program će vas na to upozoriti i tražiti da obrišete particiju prije nego je pokušate dodati u tablicu particija. U tom primjeru unesite broj 3 da biste dodali treću primarnu particiju /dev/hda3.

Zatim će program zatražiti da unesete lokaciju prvog cilindra. Obično je to prvi slobodni cilindar. U osnovi, program *fdisk* prikazuje raspon koji ste odabrali, primjerice:

```
First cylinder (42 - 1024) :
```

Uočite da prva particija završava na 41. cilindru i da iduća particija počinje na 1024. cilindru. Iduću particiju možete započeti bilo gdje između 42. i 1024. cilindra. Dobro bi bilo da particiju ne postavite bilo gdje na disku već da odaberete iduću slobodnu lokaciju (u ovom slučaju 42. cilindar). Upišite broj 42 i pritisnite tipku <Return>. **BILJEŠKA** Linux može imati problema kod pokretanja s particija čiji je početak iznad 1024. cilindra. Ako stvarate Linux particiju samo u tom rasponu, tada Linux morate pokrenuti s diskete. U nastavku ćete ovog poglavlja naučiti kako se stvara *boot* disketa sustava (razlikuje se od *boot* diskete za instalaciju). Razlika je u tome što pokretanje s diskete traje nešto duže od pokretanja Linuxa s tvrdog diska.

Nakon toga program *fdisk* traži da odredite koliko prostora želite dodijeliti toj particiji. Veličinu potrebnog prostora možete izraziti u cilindrima ili u broju bajta (+size), kB (+sizeK) ili u MB (+sizeM). S obzirom da biste već trebali znati približnu veličinu prostora koji vam je potreban za *swap* datoteku, najprije definirajte *swap* particiju, a potom ostatak prostora na disku ostavite za ostale particije. Tako, primjera radi, ukoliko vaše računalo ima 8MB RAM-a, pa trebate odrediti particiju veličine 16 MB , upišite sljedeće:

Last cylinder or +size or +sizeM or +sizeK (42-1023): +16M

Nakon toga biste trebali pokrenuti naredbu p i potražiti novu particiju koju ste definirali. U ovom primjeru bi nova tablica particije trebala ovako izgledati:

```
Disk /dev/had: 15 heads, 17 sectors, 1024 cylinders
Units=cylinders of 255 * 512 bytes
Device
           Boot
                   Begin
                           Start
                                   Fnd
                                          Blocks
                                                    Ιd
                                                           System
/dev/hda1
           *
                   1
                            1
                                   41
                                          5219 1
                                                    DOS
                                                           12-bit FAT
/dev/hda2
                   1024
                           1024
                                   4040
                                          384667+
                                                     51
                                                           Novell?
Partition 2 has different phisical/logical endings:
phys=(967, 14,17) Logical=(4039,14.17)
/dev/hda3
                   42
                            42
                                   170
                                          16447 +
                                                     83
                                                           linux native
```

Izvorno, program *fdisk* stvara novu particiju tipa Linux Native. Da biste stvorili *swap* particiju, potrebno je pokrenuti naredbu *t*. Upišite *t* i zatim unesite broj particije koju želite promijeniti (u našem primjeru je to broj 3). Program fdisk traži da zatim unesete heksadecimalnu vrijednost koja označava vrstu particije (pogledajte tablicu 3.5). Ako tablicu nemate pri ruci, upišite broj **1** i dobit ćete popis oznaka. S obzirom da želite *swap* particiju, trebate unijeti broj **82**.

Kao što možete vidjeti, program *fdisk* će vas izvijestiti o novoj vrsti particije, ali možete pokrenuti i naredbu p da biste provjerili da je particija 3 postala Linux *swap* particija.

Nakon toga možete dodavati Linux particije. Primjerice, dodajte samo jednu particiju. No, ako iz bilo kojeg razloga želite više particija, to možete odmah učiniti. Da biste dodali particiju, pritisnite tipku <n>, odredite p za drugu primarnu particiju i zatim odaberite broj te particije (4). Da biste izbjegli "ispreplitanje" raznih particija na disku, započnite novu particiju na kraju prethodne, na 171. cilindru. S obzirom da želite ostatak prostora koristiti za Linux particije, tada odredite broj zadnjeg cilindra umjesto točnog broja bajtova. Upišite **1023**, kao što je pokazano u nastavku:

```
Command (m for help): n
Command action
e extended
p primary particion (1 - 4)
p
```

Partition number (1 - 4): 4 First cylinder (171 - 1024): 171 Last cylinder or +size or +sizeM or sizeK (171 - 1023): 1023

Sada pomoću naredbe p možete provjeriti nove particije. Ako želite unijeti još neke promjene, to možete odmah učiniti.

Ako ste zadovoljni s izgledom svojih particija, tada možete pomoću naredbe *w* zapisati tablicu particija na tvrdi disk. Nijedna od promjena koje ste načinili nije trajna sve dok ne pokrenete naredbu *w*, tako da, ukoliko mislite da ste načinili neke pogrešne promjene, možete upotrijebiti naredbu *q* i izaći bez spremanja tablice particija. Ako upotrijebite naredbu *w*, Linux će vas obavijestiti da je zapamćena tablica particija i zatim će se disk preurediti tako da odgovara novoj tablici particija. Ako se vaš sustav u tom trenutku zaglavi, ponovo ga pokrenite pomoću instalacijske diskete dok se ne prikaže *# prompt*.

### **UPOZORENJE**

Nemojte koristiti Linux program fdisk za stvaranje ili mijenjanje particija za druge operativne sustave, jer se može dogoditi da vam tvrdi disk bude neupotrebljiv za oba operativna sustava.

### Stvaranje swap particije

Neke Linux distribucije tijekom instalacije automatski stvaraju i aktiviraju *swap* datoteku, tako da ne trebate brinuti o tome. No, ako koristite drukčiju distribuciju, možda ćete trebati stvoriti i aktivirati *swap* datoteku prije nego nastavite s instalacijom.

**BILJEŠKA** Ako tijekom instalacije na ekranu ugledate poruku "out-of-memory", tada trebate povećati *swap* datoteku. Ukoliko ste već dosegli maksimum od 16 MB, tada, prema datim uputama, trebate stvoriti i aktivirati još jednu *swap* particiju. Upamtite da instalacijski program Red Hat distribucije omogućuje samo jednu particiju.

Da biste stvorili *swap* particiju, upotrijebite naredbu *mkswap* i zadajte particiju koju ćete koristiti, kao i veličinu virtualne memorije. Primjerice, da biste stvorili *swap* prostor na particiji /dev/hda3 (koju ste prethodno definirali), trebate upisati sljedeću naredbu u #prompt:

```
# mkswap -c /dev/hda3 16447
```

Broj 16447 predstavlja 16 MB a možete ga pronaći u stupcu *block* ako pokrenete naredbu fdisk p. Opcija –c upućuje naredbu mkswap na provjeru loših sektora na particiji.

Zatim trebate swap sustav aktivirati pomoću naredbe swapon:

```
# swapon /dev/hda3
```

Ukoliko koristite Red Hat CD-ROM, ne trebate voditi brigu o aktiviranju *swap* sustava ako ste stvorili *swap* particiju. Tijekom će instalacije program uočiti *swap* particiju i automatski pokrenuti sustav za instalaciju.

Nakon što ste stvorili particije na nekoliko tvrdih diskova i vratili se u okvir za dijalog Partitioning Disks, odaberite gumb Done za nastavak instalacije.

Zatim ćete trebati odabrati aktivni *swap* prostor, odnosno particiju koju ste označili kao Linux Swap (82) (vidi prethodni odlomak). Odaberite tu particiju i zatim odaberite gumb OK. Program će inicijalizirati *swap* prostor.

Nakon što stvorite *swap* prostor, pojavit će se okvir za dijalog Select Root Partition. *root* particija predstavlja glavni sustav datoteka u Linuxu i u njoj se nalaze sve datoteke za pokretanje sustava. Odaberite tvrdi disk na kojem će se nalaziti *root* particija i zatim pritisnite tipku <Enter>. Potom možete načiniti druge particije. U okviru za dijalog Partition Disk možete stvoriti DOS i OS/2 sustave datoteka, kojima možete pristupati pomoću Linuxa. U popisu odaberite particiju i pritisnite tipku <Enter>. U okviru za dijalog Edit Mount Point možete odrediti direktorij (*mount point*) u kojem želite

stvoriti takvu particiju.

Vidi "Postavljanje i uklanjanje sustava datoteka", 269. stranica

Nakon što odaberete *root* direktorij i tzv. *mount point* direktorije na raznim particijama, program će ih fomatirati. Particije koje želite formatirati odaberite u okviru za dijalog Format partition.

# Uporaba programa Disk Druid

Slika 3.8.

Program Disk Druid omogućava podjelu diska na particije.

Odaberete li gumb Disk Druid u okviru za dijalog Disk Setup (slika 3.7), pojavit će se glavni ekran programa Disk Druid koji je prikazan na slici 3.8.

ed Hat Linux (C)	1998 Red Hat	Software		Setup file	esyste
	Cu	rrent Disk Part:	itions -	-	
Mount Point	Devi	ce Requester	Actual	lype	
	nua	L 1016	E07H	Linux nacive	
	nua hda	5 005H	1279M	Linux native	1
	hda	7 509M	509M	Linux native	
	hda	3 125M	1251	linux swap	
	hda	3 768M	768M	Linux native	
	hda	10 768M	768M	Linux native	
hda [	551/255/631	4165N 4157	1 Free 1 8H	[**********	
Add	Edit	Delete		Back	<
F1-Add F2-6	add NFS F3	Edit F4-Dele	te F <del>5-R</del> es	et F12-0k v	v 1.0

Program Disk Druid omogućava podjelu diska na particije, postavljanje direktorija (*mount point*) pojedinih uređaja (diskova i particija), određivanje veličine particija i odabir sustava datoteka. U tablici 3.5 naći ćete popis različitih polja u glavnom ekranu programa Disk Druid te opis zadaća pojedinih gumbiju.

Vidi "Sustavi datoteka", 266. stranica.

Tablica 3.5 Zna	čenje polja/gumbiju u programu Disk Druid
Polje/Gumb	Opis
Mount Point	Drugi izraz za direktorij; mjesto na koje će biti smješten određeni direktorij zajedno s poddirektorijima.
Device	Fizički tvrdi disk i particija kojem pripada direktorij (mount point).
Requested Size	Možete odabrati bilo podrazumijevanu veličinu (default) particije ili početnu veličinu koja će se kasnije povećavati.
Actual Size	Trenutna veličina pojedine particije.
Туре	Vrsta sustava datoteka na pojedinoj particiji.
Drive Summaries	Ovaj dio glavnog ekrana sadrži podatke o pojedinom uređaju (tvrdom disku/particiji) uključujući i veličinu raspoloživog prostora. Upamtite, tvrdi disk ima nekoliko particija.
Add	Dodavanje nove particije.
Edit	Uređivanje odabranog direktorija (mount point).
Delete	Brisanje odabranog direktorija (mount point).
Ok	Potvrda unesenih promjena i nastavak instalacije.
Back	Povratak na prethodni okvir za dijalog u instalacijskom programu te zanemarivanje svih unesenih promjena.

Da biste dodali novu particiju, kliknite na gumb Add ili pritisnite funkcijsku tipku <F2>. Pojavit će se okvir za dijalog Edit New Partition koji je prikazan na slici 3.9.

### Slika 3.9

Ne trebate pamtiti sve vrste sustava datoteka, jer ćete njihov popis naći u okviru za dijalog Edit New Partition.

Mount Point:	/usr/local_	
Size (Megs): Growable?:	100 [#]	Type:Linux Swap Linux Native DOS 16-bit <32M DOS 16-bit >=32M
Allowable Drives	: [*] hda	
	<u>Ck</u>	Cancel

U polje Mount Point unesite direktorij (*mount point*) za tu novu particiju, primjerice *root* (/) ili var particiju (/var). Potom unesite veličinu particije (u MB) te odredite želite li da se particija povećava kada budete dodavali i brisali druge particije. Nakon toga trebate u popisu Type odabrati sustav datoteka koji će se koristiti na toj particiji. Na kraju, iz popisa Allowable Drives možete odabrati tvrdi disk za smještaj particije.

**BILJEŠKA** Odaberete li veličinu koja je prevelika u odnosu na raspoloživi prostor na disku, program Disk Druid će vas na to upozoriti i tražiti da smanjite količinu prostora, a osim toga će vas upozoriti i na druge moguće probleme i ponuditi vam njihova rješenja.

# Instalacija softverskih komponenti

Čestitamo! Vaše je računalo sada spremno za Linux, ali ste ipak tek na pola puta. Sada morate odabrati različite softverske komponente te ih instalirati i podesiti.

U programu za instalaciju se pojavljuje okvir za dijalog Components to Install, u kojem možete odabrati razne pakete (opis svakog paketa potražite u tablici 3.6).

Tablica 3.6 Softverske komponente		
Komponenta	Opis	
Printer Support	Omogućuje ispis na pisač.	
X Windows System	Nudi grafičko sučelje za sve UNIX/Linux radne stanice (nalik Windowsima 95 i OS/2).	
Mail/WWW/News alati	Programi za korištenje e-pošte, kretanje po Webu te čitanje i slanje Usenet vijesti.	
DOS/Windows Connectivity	Omogućuje pristup DOS datotekama, pokretanje DOS programa, kao i nekih Windows programa (ali uz ograničen uspjeh).	
File Managers	Alati za rad sa sustavom datoteka (primjerice Midnight Commander).	
BRU Backup Util	Inačica poznatog programa za pohranu traka (za jednog korisnika).	
BRU X11 Front End	Grafičko sučelje za program BRU.	
Real Media Client	Omogućava pristup programima Real Media koji se nalaze na Internetu.	
Real Media Server	Omogućava Linux poslužitelju objavljivanje na Internetu sadržaja Real Media.	
Graphics Manipulation	Nudi programe za rad sa slikama, kao što su xv i popularni The GIMP.	
X Games	Nudi popularne strategijske i arkadne igre koje rade u okružju XWindowsa.	
Console Games	Nudi igrice koje se pokreću s konzole.	
X Multimedia Support	Multimedijska potpora za X Windowse.	
	nastavlja se	

Tablica 3.6 Softverske kompone	nte
Komponenta	Opis
Console Multimedia Support	Multimedijska potpora za konzole.
Printer Server	Omogućuje da vaše Linux računalo postane poslužitelj za pisač.
Networked Workstation	Nudi mrežne aplikacije i SNMP potporu.
Dialup Workstation	Omogućuje vam pristup Internetu putem modemske veze.
News Server	Omogućuje da se vaše računalo ponaša kao poslužitelj za vijesti, tako da svojim korisnicima možete ponuditi vijesti.
NFS Server	Omogućuje izvoz i uvoz sustava datoteka.
SMB (Samba) Connectivity	Nudi SMB servise.
IPX/NetWare Connectivity	Omogućuje pristup Novell NetWare mrežama.
Anonymous FTP/Gopher Server	Omogućuje vam da sustav podesite tako da mu i drugi mogu pristupiti putem anonimnog FTP-a.
Web Server	Sadrži najpoznatiji današnji Web poslužitelj – Apache.
DNS Name Server	Sadrži softver koji vam je potreban za pokretanje vlastitog Domain Name Servera.
PostGress (SQL) Server	Omogućuje vam pokretanje sustava PostGress SQL.
Network Management Workstation	Uslužni programi i alati koji služe kao pomoć u radu na mreži i praćenju mreže, uključujući i SNMP servise.
TeX Document Formatting	Niz programa koji se koriste za dodavanje formata u dokumente.
Emacs	Instaliranje programa za uređivanje teksta (s emacsom možete učiniti sve, barem tako tvrde njegovi zagovornici).
Emacs sa XWindowsima	X Windows sučelje za moćni program emacs.
C Development	Sadrži GNU gcc kompajler i alate.
Development Libraries	Sadrži razne biblioteke s razvojnim alatima (kao što su gcc i g++).
C++ Development	Instalacija GNU C++ kompilatora, gcc.

### Ovaj fajl je skinut sa sajta www.tutoriali.org

Komponenta	Opis
X Development	Alati, biblioteke i dodatni elementi (primjerice pisma) za razvoj X aplikacija.
Extra Documentation	Dokumentacija za Linux zajedno s važnim HOWTO datotekama i drugim korisnim informacijama.
Everything	Instalira sve s CD-ROM-a; potrebno vam je oko 350 MB prostora, ne računajući prostor za vaše datoteke s podacima.

**BILJEŠKA** Možete odabrati pojedine pakete tako da potvrdite odgovarajuća polja za potvrdu koja se nalaze u okviru za dijalog ili sve instalirati tako da odaberete čitav popis. Da biste odabrali paket, jednostavno označite željenu komponentu i pritisnite razmaknicu. Nakon što odaberete sve potrebne komponente, prijedite na gumb OK i pritisnite tipku <Enter>.

SAVJET

Možete upotrijebiti program RPM koji je opisan u 6. poglavlju i naknadno instalirati bilo koji paket.

U idućem okviru za dijalog saznat ćete da možete vidjeti popis instaliranih datoteka u datoteci /tmp/install.log. Za nastavak instalacije pritisnite tipku <Enter>.

Sada slijedi teži dio – čekanje. Prijenos i raspakiravanje gotovo 350 MB programa može potrajati poprilično vremena. Program će najprije instalirati sustav datoteka na odabrane particije, a potom će se instalirati softver. Sustav vas u okviru za dijalog Install Status obavještava o tijeku instalacije pojedinih datoteka koje ste odabrali. Trajanje instalacije ovisi o odabranim programima i brzini vašeg računala. Opustite se i naručite pizzu.

# Podešavanje sustava

Nakon instalacije softvera instalacijski program započinje s podešavanjem sustava. Najprije treba podesiti miša u okviru za dijalog Configure Mouse. Jednostavno odaberite onu vrstu miša koja najbolje odgovara mišu što ga koristite. Upamtite da mnogi miševi mogu emulirati Microsoftovog serijskog miša, ukoliko je to nužno. Postoji i opcija Emulate 3 Buttons jer mnogi miševi imaju samo dva gumba, a XWindowsi obično za kretanje i odabir koriste tri gumba. Potvrdite li tu opciju, sustav će pritisak na oba gumba istovremeno prihvatiti kao pritisak na srednji gumb miša s tri gumba. Odaberite potrebnu opciju i zatim odaberite gumb OK.

**SAVJET** U bilo kojem trenutku možete promijeniti podešenje miša ako upotrijebite sljedeću naredbu: /usr/sbin/mouseconfig. Zatim morate odrediti serijski priključak na koji je miš spojen. Nakon što priključak odaberete iz popisa, odaberite gumb OK i pritisnite tipku <Enter>. Program će tražiti da odaberete onu grafičku karticu koja se nalazi u vašem računalu.

### **UPOZORENJE**

Nastojte odabrati ispravnu grafičku karticu jer su grafička kartica i monitor jedine komponente koje softver može uništiti. Odaberete li pogrešnu grafičku karticu, možete "spaliti" monitor, što može biti vrlo neugodno. Stoga, budite oprezni.

Nakon toga instalacijski program će pokušati instalirati odgovarajući XFree86 poslužitelj.

### Vidi "Instaliranje sustava XFree86", 425. stranica

Zatim morate odabrati monitor i pri tome biti što je moguće određeniji. Nakon što odaberete monitor, instalacijski program će zatražiti da upišete količinu grafičke memorije koja je instalirana na kartici. Odaberite točnu vrijednost i za nastavak odaberite gumb OK.

Jeste li upamtili sva upozorenja vezana za monitor? Sada ga doista možete "ispeći", stoga budite pažljivi. U idućem prozoru trebate odabrati frekvenciju s kojom radi grafička kartica (*clockchips*). Radi se o čipovima koji se koriste za pogon video signala. Ako njihov rad nije usklađen, signali mogu, to već pogađate, spaliti monitor (poneki monitori će eksplodirati, a većina će se samo zadimiti). Molimo budite pažljivi! Ako nemate pojma koje čipove koristi vaša kartica, tada izaberite podrazumijevano podešenje -No Clockchip Setting i odaberite gumb OK.

Nakon što ste odabrali frekvenciju rada monitora, sustav će ga automatski isprobati i pokušati podesiti XWindowse. Kod provjere, sustav bi se mogao zaglaviti, no ako se ne radi o ozbiljnijem problemu (ako, primjerice, niste odabrali previsoku frekvenciju za vašu karticu), moći ćete jednostavno ponovno podići sustav i nastaviti s instalacijom. Postoji mogućnost da preskočite automatsku provjeru i nastavite s instalacijom.

**BILJEŠKA** Mnogo sam puta instalirao Red Hat distribuciju i nikada do sada instalacijski softver nije pravilno podesio sustav XWindows. Možda ćete imati više sreće nego ja, stoga nemojte brinuti da ćete "zabrljati" instalaciju. Svaki puta sam uspio naknadno podesiti XWindowse koristeći razne programe za podešavanje.

Vidi "Podešavanje sustava XFree86", 430. stranica

Nakon što prođe automatska provjera sustava, pojavit će se ekran u kojem trebate odabrati razlučivost ekrana. Možete odabrati više raznih razlučivosti ako ih grafička kartica i monitor mogu koristiti. Na kraju će vas instalacijski program uputiti kako ćete pokrenuti i zaustaviti sustav XWindows.

# Podešavanje mreže

Nakon podešavanja XWindowsa instalacijski program nastavlja s instalacijom mreže. Ukoliko vam je računalo spojeno (ili će to biti) na Internet te ako ste instalirali mrežne komponente, tada samo odaberite Yes za nastavak.

Prvo ćete u okviru za dijalog Load Module trebati odabrati odgovarajući pogonski program za Ethernet karticu i zatim odabrati OK.

Instalacijski program će provjeriti vaš hardver te odrediti potrebne vrijednosti kartice. Tijekom provjere sustav se može zaglaviti pa ćete ga morati ponovno pokrenuti. Najprije provjerite jeste li odabrali pravi upravljački program. Zatim pogledajte trebate li podesiti i posebne parametre na uređaju, kao što su IRQ ili DMA adrese. To možete učiniti odabirom opcije Specify Parameter (umjesto opcije automatske provjere).

**BILJEŠKA** Ethernet je danas najpopularniji izbor. Druge tehnologije, kao što su Token Ring, ISDN i ATM, također imaju podršku, ali još se uvijek ne mogu koristiti pod Linuxom, jer se mnoge provjeravaju.

Ako sustav može odrediti o kojoj se mrežnoj kartici radi, onda možete nastaviti s podešavanjem TCP/IP mreže.

# Podešavanje TCP/IP mreže

Instalacijski softver koristi okvir za dijalog Configure TCP/IP za prikupljanje potrebnih podataka. Mrežni administrator ili davatelj Internet usluga bi vam trebao dati sljedeće podatke: IP adresu vašeg računala, mrežnu masku i adresu.

Nakon toga sustav mora podesiti vašu mrežu. Podatke unosite u okvir za dijalog Configure Network. Trebate odrediti ime *hosta* i domene (to su obično zadnja dva dijela Internet adrese, primjerice ako se radi o adresi www.netwarf.com, onda je netwharf.com ime domene, a www ime hosta).

Zatim vam mrežni administrator treba dati vrijednosti podrazumijevanog *gatewaya* i primarnog poslužitelja imena. Vaša mreža također može imati i sekundarni poslužitelj imena, a u tom slučaju trebate unijeti i tu vrijednost na odgovarajuće mjesto.

BILJEŠKA Budite pažljivi s imenom hosta jer će se ono pojavljivati u svim porukama i u log datotekama. Želite li doista da vaš šef dobije poštu s adrese uradork.netwharf.com?

# Podešavanje sata

Zatim morate odrediti način računanja vremena i vremensku zonu u kojoj se nalazite. To možete učiniti u okviru za dijalog Configure Timezones. Odlučite želite li koristiti lokalno vrijeme ili GMT, a zatim odaberite vremensku zonu iz odgovarajućeg popisa. Na kraju odaberite gumb OK.

# Odabir servisa koji se automatski pokreću kod podizanja sustava

Sada trebate odrediti servise (programe i *demone*) koje će računalo automatski pokretati prilikom podizanja sustava. Možete odabrati neke od servisa koji su prikazani u okviru za dijalog Services, koji je prikazan na slici 3.10. Tablica 3.7 sadrži popis raspoloživih servisa te njihovu namjenu. Servisi koji su označeni sa zvjezdicom (\*) podrazumijevano se pokreću prilikom pokretanja sustava.

Vidi "Pokretanje sustava", 206. stranica

### htsyny 0.9.3 - (C) 1998 Red Hat Software legequequequequequequeque What services should be automatically started? Cancel What services should be automatically started? W

Tablica 3.7         Popis servisa koji se mogu pokrenuti tijekom podizanja sustava	
Servis	Opis
amd	pokreće automount demon
apmd*	praćenje stanja baterija; mogućnost gašenja sustava ako su baterije prazne
atd*	pokreće naredbu at prema utvrđenom rasporedu
autofs	automatski se stvara sustav datoteka
bootparamd	omogućuje poslužiteljima Sun podizanje iz Linuxa pomoću naredbe bootp
crond*	pokreće cron demon
dhcpd*	pokreće DHCP servise
gated	pokreće gate demon za usmjerivačke servise za BGP i druge protokole
gpm*	pokreće programe koji sadrže Linux podršku za miša
httpd*	pokreće Web poslužitelj Apache
Inet*	pokreće internet super demon (inted) koji sadrži sve servise odabrane u datoteci /etc/inet.conf
inmd	pokreće poslužitelj za Usenet vijesti innd
kerneld*	pokreće kerneld demon koji prema potrebi učitava/uklanja module jezgre

Linux vam omogućava odabir programa koji se pokreću tijekom pokretanja sustava, što je nalik mapi Startup kod Microsoft Windowsa.

Slika 3.10

Ovaj fajl je skinut sa sajta www.tutoriali.org

Servis	Opis
keytable*	učitava odgovarajuću mapu za tipkovnicu
lpd*	sadrži Linux servise za ispis
mars-new	učitava datoteku MArs NetWare i demon ispisnog poslužitelja
mcserv	sadrži servise za daljinsko upravljanje datotekama
named*	sadrži DNS servise
network*	omogućuje upravljanje svim mrežnim priključcima
nfs*	sadrži servise za Network File System poslužitelj
nfsfs*	postavlja/uklanja sve NFS direktorije (mount point) odabrane u datoteci /etc/exports
pcmcia	omogućuje pristup PCMCIA (PC) servisima
pnserver	pokreće Real Media servise
portmap*	nudi RPC (Remote Procedure Call) potporu za druge protokole kao što je NFS
postgresql	pokreće bazu podataka postgres i SQL servise
random*	sprema i obnavlja slučajnu vrijednost koja pomaže kod stvaranja boljih slučajnih brojeva (koji se koriste u raznim sigurnosnim sustavima)
routed	omogućuje automatsko ažuriranje tablice usmjernika pomoću protokola RIP
rusersd	omogućuje korisnicima da nađu jedan drugoga na mreži
rwalld	omogućuje korisnicima uporabu naredbe rwall za pisanje poruka na udaljenim terminalima
rwhod	protokol rwho omogućuje udaljenim korisnicima da dobiju popis svih trenutno prijavljenih korisnika
sendmail*	pokreće demona za rad s e-poštom
smb*	sadrži SMB (Samba) klijent/poslužitelj servise
snmpd*	Linux podrška za Simple Network Management Protocol
sound*	omogućava pristup zvučnim karticama
squid*	pokreće proxy Web poslužitelj squid
syslog*	omogućuje prijavu na vaš Linux sustav
xntpd	pokreće NTPv3 demon
ypbind	povezuje YP/NIS klijente s poslužiteljem za žute stranice (yellow pages)
yppasswd	omogućuje korisnicima promjenu lozinke na YP/NIS sustavima
ypserv	sadrži funkcije YP/NIS poslužitelja



SAVJET lako ručno možete mijenjati servise koji se automatski pokreću prilikom podizanja sustava, tako da mijenjate datoteke rc.d (vidi 9. poglavlje), naredba /usr/sbin/ntsysv vraća vas u okvir za dijalog Services te vam omogućava ponovno podešavanje tih servisa u grafičkom sučelju.

# Određivanje root lozinke

Sada morate odrediti svoju root lozinku. Radi se o ključu vašeg sustava, stoga je vrlo pažljivo birajte. Super korisnik u Linux/UNIX sustavu može učiniti doista bitne stvari, ali isto tako i veliku štetu. Odaberite sigurnu lozinku i pazite kome je dajete. Okvir za dijalog Root Password je prikazan na slici 3.11, a u njemu možete dva puta unijeti lozinku kako biste potvrdili ono što ste prvi puta napisali.



Iako imate mogućnost mijenjati lozinke svojim korisnicima, ako zaboravite root lozinku, morat ćete ponovo instalirati sustav. Ipak, pokrenete li sustav s diskete i preuredite li datoteku s lozinkom, možda ćete moći spasiti stvar.

Vidi "Zaštita pomoću lozinke", 237. stranica.

# Instalacija programa LILO

Zatim ćete trebati stvoriti boot disketu, kao što je prikazano u okviru za dijalog Bootdisk, na slici 3.12. Doista vam preporučamo da načinite jednu takvu disketu, za slučaj ako ne budete mogli podići svoj sustav. Boot disketa je u takvim slučajevima prva pomoć, a potom slijedi disketa sa sigurnosnom kopijom sustava (rescue).

Nakon što stvorite takvu disketu, instalacijski će program zatražiti da instalirate LILO. LILO je kratica od LInux LOader i predstavlja program koji se pokreće prilikom podizanja sustava i omogućava odabir operativnog sustava koji će se koristiti za pokretanje računala. Program LILO možete koristiti za pokretanje nekoliko različitih operativnih sustava, kao što su Linux i MS-DOS. Pritisnite tipku <Tab> da biste dobili popis operativnih sustava koje LILO može pokrenuti.

Pomoću programa LILO također možete odrediti izvorni operativni sustav kojim će se podizati računalo kao i vremensko ograničenje čekanja prije nego se pokrene taj sustav. Primjerice, ukoliko na svom računalu imate MS-DOS i Linux, onda pomoću programa LILO možete bilo koji odabrati za podizanje računala. Možete zadati programu da pokrene MS-DOS ukoliko nema intervencije unutar 30 sekundi. Prije nego istekne tih 30 sekundi, korisnik može odabrati i drugi operativni sustav (umjesto osnovno postavljenog). Možete pritisnuti neku od tipki <Ctrl>, <Alt> ili <Shift> i prekinuti odbrojavanje.



Sve te podatke trebate zadati tijekom podešavanja programa LILO. Kasnije možete izravno uređivati datoteku lilo.conf, koja se nalazi u direktoriju /etc. Ako Linux ne želite pokretati automatski, onda trebate odabrati gumb Skip za nastavak. U protivnom odaberite tvrdi disk za instalaciju programa LILO i pritisnite tipku <Enter>.

Čestitamo! Nakon što učitate program LILO, vaš će sustav proraditi - nadamo se, bez problema.

# Instalacija Red Hata na sustavima DEC Alpha

Za razliku od drugih distribucija, Red Hat nudi Linux za sustave DEC Alpha. Detaljnije informacije o tome kako ćete nabaviti tu distribuciju potražite na Red Hat Web lokaciji **http://www.redhat.com/products/rhl-alpha.html.** Ukoliko imate pravu distribuciju, možete koristiti iduće upute za instalaciju Red Hat Linuxa na Alphu.

BILJEŠKA Tvrtka Red Hat je izdala distribuciju za Sun Sparc procesore. Više podataka potražite na Red Hat Web lokaciji.

Prije instalacije Linux distribucije na sustav Alpha trebali biste pročitati upute iz prvog dijela ovog poglavlja, jer se većina koraka ponavlja. Također trebate potražiti računalo koje ima mogućnost čitanja i pisanja na diskete za MS-DOS, jer ćete trebati stvoriti instalacijsku disketu.

### Uporaba hardverskih komponenti koje podržava sustav Alpha

Red Hat podržava razne Alpha hardverske komponente koje proizvodi tvrtka Digital Equipment Corporation (DEC), ali i drugi proizvođači. Podržane su sljedeće komponente:

- AlphaPC64 (Cabriolet, Aspen Telluride)
- AxpPCI33 (Noname)
- EB64+ (Aspen Alpine)
- EB66 (NekoTek Mach 1)
- EB66+
- Jensens (DEC PC 150, 2000 model 300, Cullean)

- Universal Desktop Box (UDB, aka Multia)
- AlphaStation 200, 250, 255 i 400 (računala Avanti)
- EB164 (Aspen Avalanche, Timerline, Summit)
- Kineticova računala Platform 2000
- PC164 (Durango)
- Alcor AlphaStations 500, 600 (Maverick, Brett)
- Alpha XL
- Alpha XLT (XL 300, XL 366)
- Mikasa AlphaServer 1000 (ne podržava 1000A)

Sva ta računala imaju SCSI sustave koje podržava Red Hat Linux. Također bi trebali raditi i video sustavi, iako izvorno nije uključena S3 potpora za Jensen sustave. Da biste pokrenuli XWindowse s Jensen sustavom, trebate s mreže kopirati X poslužitelj (adresa **ftp://ftp.azstarnet.com/pub/linux/axp/jensen**). Podržavaju se sve Ethernet kartice, a jezgre za ta računala podržavaju adaptere Token-Ring.

Popis hardverskih komponenti se često mijenja, stoga provjerite najnoviji popis na Red Hat Web lokaciji http://www.redhat.com.

### Stvaranje boot i root disketa

Trebate stvoriti *boot* i *root* disketu za instalaciju Alphe. *Boot* disketa sadrži programsku sliku koja omogućava pokretanje sustava. *Root* disketa sadrži sliku jezgre Linuxa koju sustav koristi tijekom instalacije. Za računala s Intel procesorom možete koristiti program rawrite i stvoriti te diskete.

Sadržaj *boot* diskete ovisi o vrsti Alpha sustava koji koristite. Slike programa su smještene u odgovarajućem direktoriju (/images) zajedno s datotekom README koja sadrži i dodatne informacije o svakoj slici koja je opisana u tablici 3.8

Tablica 3.8	Slike programa i datoteka za sustav DEC Alpha	
Slika (Image)	Opis	
cab.img	AlphaPC64, Cabriolet	
noname.img	AxpPCI33, Noname, Universal Desktop Box (Multias)	
eb64p.img	EB64+, Aspen Timberlines	
eb66.img	EB66	
eb66p.img	EB66+	
jensen.img	Jensens	
avanti.img	AlphaStation 200, 250 i 400	

Slika (Image)	Opis
xl.img	Alpha XL
xlt.img	Alpha XLT
eb 164.img	EB164-računala
p2000.img	Platform 2000
alcor.img	Alcor-računala
mikasa.img	Mikasa računala

Da biste stvorili boot disketu za Universal Desktop Box, trebate upotrijebiti ovu naredbu:

E:\dosutils\rawrite -f E:\images\noname.img -d a: -n

gdje E: predstavlja ime CD-ROM pogona. Nakon što stvorite *boot* disketu, morate stvoriti i *root* disketu koja sadrži sliku jezgre Linuxa. Tu disketu možete stvoriti pomoću naredbe:

```
E:\dosutils\rawrite -f E:\images\ramdisk.img -d a: -n
```

### Instaliranje glavne Red Hat distribucije

Nakon što ste stvorili pokretačku disketu, možete instalirati Linux. Postupak instalacije prilično je sličan onome koji je opisan u odlomku "Instalacija Linux sustava". Instalacijski program vas vodi kroz instalaciju, tražeći da odaberete neke od ponuđenih opcija.

Za početak umetnite boot disketu u disketni pogon i pokrenite sustav. Unesite sljedeću naredbu:

```
boot fd0:vmlinux.gz root=/dev/fd0 load_ramdisk=1
```

Možda ćete vidjeti nekoliko poruka na ekranu. Neka vas to ne zabrinjava, ukoliko ne vidite poruku scsi : 1, što upućuje da postoji problem koji treba otkloniti prije nastavka instalacije. Ako sve prođe kako treba, trebali biste ugledati poruku VFS: Insert Root floppy to be loaded into ramdisk. Umetnite *root* disketu koju ste prethodno stvorili i pritisnite tipku <Enter> za nastavak instalacije.

# Vraćanje na početak

Nakon što ste završili podešavanje sustava, trebali biste ponovno podići sustav tako da sustav prihvati sva podešenja.

Ponovno pokretanje Linuxa je puno složenije od pokretanja DOS-a. Ne možete samo isključiti napajanje i ponovno ga uključiti. Učinite li to u Linuxu, oštetit ćete strukturu datoteka. Sustav Linux se pokušava "oporaviti" tijekom podizanja. Ne isključujte napajanje dok radite s Linuxom.

Vidi "Sigurnosna pohrana podataka i obnova datoteka", 229. stranica

Za izlazak iz Linuxa koristite sljedeću naredbu: shutdown [-r] time
Zastavica –r označava da bi se sustav trebao ponovno pokrenuti nakon isključivanja. Parametar *time* označava vrijeme nakon kojeg se sustav treba isključiti. Umjesto parametra *time* možete upotrijebiti *now*, ako želite trenutno isključiti sustav. Linux će također prepoznati kombinaciju tipki za pokretanje računala koja se koristi u DOS-u (<Ctrl-Alt-Delete>), i to kao naredbu:

shutdown -r now

Provjerite jeste li izvadili sve diskete iz pogona i ponovno pokrenite svoje novo Linux računalo.

# Otklanjanje problema

Nakon što ste pokrenuli računalo, trebao bi se na ekranu pojaviti LILO *prompt*. Provjerite možete li pokrenuti svoj stari operativni sustav (ako ste ga sačuvali na tvrdom disku). Ako je DOS vaš stari operativni sustav, pritisnite tipku <Shift> i zatim upišite kraticu koju ste koristili za identifikaciju DOS particije. Upišete li krivo, tada pritisnite tipku <Tab> da biste vidjeli popis važećih operativnih sustava. Imate li i nadalje problema, umetnite DOS *boot* disketu i ponovno pokrenite sustav.

Vidi "Otklanjanje problema", 52. stranica

Trebali biste računalo podići pomoću diskete sa sigurnosnom kopijom sustava (*rescue*). Nakon što pokrenete sustav pod DOS-om, isprobajte disketu sa sigurnosnom kopijom sustava koju ste stvorili tijekom instalacije (a ne instalacijsku disketu čitavog sustava). Ukoliko ta disketa ne radi, možda ćete trebati ponovno instalirati Linux. Potencijalni problemi se mogu javiti kod jezgri (kernela) i hardvera. Prije nego počnete s radom, provjerite je li vaš hardver odgovarajući. Ako ste tijekom instalacije zapisivali određene podatke, onda provjerite koju ste jezgru predvidjeli za vaš hardver.

#### Odavde...

Sada kada ste pokrenuli svoj sustav, možete pročitati i sljedeća poglavlja i saznati nešto više o Linuxu:

- 2. poglavlje "Postupak instalacije Linuxa" sadrži detaljne upute za instalaciju Linuxa. Poglavlje također nudi informacije o mogućim problemima i njihovom rješavanju.
- 5. poglavlje, "Pokretanje Linux programa" omogućuje vam da ubrzate različite programe koje ste upravo instalirali.
- 6. poglavlje, "Nadogradnja i instaliranje softvera pomoću RPM-a" sadrži informacije kako ponovno instalirati pakete koje ste trebali ukloniti tijekom početnog podešavanja Linux sustava
- 21. poglavlje, "Instaliranje XWindowsa" nudi vam informacije za instaliranje XWindowsa ukoliko naiđete na probleme kod instaliranja Red Hat distribucije. Iako su XWindowsi predviđeni za Slackware distribuciju, softver XFree86 je isti za bilo koju inačicu Linuxa.
- 22. poglavlje, "Uporaba XWindowsa", je zanimljivo ako ste instalirali XWindowse.

# Instaliranje OpenLinuxa Lite

(Jack Tackett)

#### U ovom poglavlju

Što vam je potrebno za instalaciju Linuxa? Instalacija Potrebne pripreme za instalaciju Priprema instalacijskih disketa Instalacija Linuxa Instalacija Linux softverskog sustava Podešenje sustava Instalacija LILO Vraćanje na početak Rješavanje problema U ovom ćete poglavlju naći informacije koje vam trebaju za instalaciju Caldera OpenLinux distribucije. Kao i Red Hat i Slackware distribucije, tako je i Open Linux kompletna distribucija višekorisničkog, višezadaćnog operativnog sustava koji se zasniva na jezgri Linux 2.0. Nekomercijalna inačica Caldera OpenLinux distribucije označava se s *lite*. Ta je inačica skraćena varijanta komercijalne OpenLinux Base distribucije. U tablici 4.1 ćete naći popis razlika između komercijalnog proizvoda i nekomercijalne inačice.

Tablica 4.1         Usporedba inačica OpenLinux Base i OpenLinux Lite				
Komponenta	<b>OpenLinux Base</b> (komercijalna)	<b>OpenLinux Lite</b> (nekomercijalna)		
Linux OS i uslužni programi	Ima	Ima		
Systen Admin i instalacijski alati	Ima	Ima		
Upute na 250 stranica	Ima	Nema		
Tehnička potpora	Ima	Nema		
Netscape Navigator	Ima	Nema		
Looking Glass Desktop	Ima	Ima (proba 30 dana)		
CrispLite Graphical Editor	Ima	Ima (proba 30 dana)		



#### NA WEBU

Želite li saznati nešto više o Calderinim proizvodima, onda svakako posjetite njihovu Web lokaciju koja se nalazi na adresi

http://www.caldera.com

# Što vam je potrebno za instalaciju Linuxa?

Tvrtka Caldera ima ugovore s tvrtkom Red Hat za korištenje njihove distribucije, tako da su mnoge instalacijske procedure iste za obje distribucije. Za pomoć prilikom instalacije OpenLinux distribucije trebali biste pročitati 3. poglavlje "Instalacija Red Hat distribucije".

Da biste uspješno instalirali OpenLinux Lite, vaš bi sustav trebao sadržavati sljedeće komponente:

- procesor 80386 (ili jači) za PC (Intel, jer Caldera za sada ne podržava druge procesore)
- disketni pogon 3,5 palca
- najmanje 8 MB RAM-a
- oko 250 MB prostora na disku iako minimalni sustav, bez Xwindowsa, zahtijeva samo 50 MB. Kompletna instalacija sa svim komponentama traži gotovo 690 MB.
- miš i grafičku karticu koji podržava paket XFree86.

Da ukratko ponovimo: najprije je potrebno tvrdi disk podijeliti na particije, potom stvoriti *boot* disketu i zatim instalirati i podesiti sustav.

Vidi "Podjela tvrdog diska na particije", stranica 42

#### **UPOZORENJE**

Budite oprezni jer ćete u svom sustavu načiniti velike promjene.

#### Instalacija

Najprije je nužno da imate Caldera OpenLInux distribuciju koju, pretpostavljeno, imate na CD-ROM-u. Da biste pokrenuli instalaciju, potrebne su vam dvije formatirane diskete.

Osim toga, trebate odlučiti na koji ćete način pokrenuti Linux. Na raspolaganju su vam dvije mogućnosti:

- Možete Linux pokrenuti s diskete. U tom vam slučaju treba još jedna formatirana disketa (ukupno tri diskete).
- Možete upotrijebiti program LILO (LInux LOader). LILO vam omogućuje odabir operativnog sustava koje ćete pokrenuti. Sličnu mogućnost nude i sustavi OS/2, Windows 98 i Windowsi NT.

Zatim biste trebali provjeriti imate li dovoljno prostora na disku za instalaciju Linuxa. Većini ljudi će biti dovoljno 200 MB, pogotovo ako ne namjeravate koristiti takve programe kao što su TeX i XWindows.

Dobro je uvijek u blizini imati papir i olovku tako da možete zapisati neke stvari, u slučaju da se instalacija zakomplicira. Osim toga, bit će potrebno zapisati i neke brojeve. Kod podešavanja programa XFree86, XWindows programa koji se distribuira s Linuxom, trebali biste zapisati vrstu čipova koje koristi vaša grafička kartica. Koristite li serijskog miša ili modem, zapišite serijske priključke koje svaki od ova dva uređaja koristi, jer će vam te informacije trebati kasnije, tijekom podešavanja sustava.

#### Potrebne pripreme za instalaciju

Imate li sasvim novo računalo, ili ako vas nije briga što će se dogoditi s podacima koji su spremljeni na računalu, tada možete preskočiti većinu idućih odlomaka. No, ako već duže vremena koristite računalo i jednostavno želite dodati Linux, neke stvari trebate isplanirati jer je Linux operativni sustav, a ne obična zbirka programa.

Općenito, ako instalirate Linux (odnosno novi operativni sustav), tada morate učiniti sljedeće:

Stvoriti diskete za pokretanje Linuxa (boot diskete). Morate načiniti dvije diskete jer Linux morate pokrenuti kao novi sustav.

- Preparticionirati tvrdi disk da biste načinili mjesta za Linux. Tu mogu nastati problemi jer se tim postupkom uništavaju svi podaci spremljeni na odabranim particijama.
- Pokrenuti Linux. Nakon što načinite mjesta za sustav Linux, trebate ga pokrenuti da biste mogli pristupiti alatima koji su vam potrebni za stvaranje novih particija i sustava datoteka.
- Stvoriti Linux particije. Obično Linux sustav zahtijeva primarnu particiju za spremanje datoteka i *swap* particiju (osobito ako koristite računalo koje ima manje od 8 MB RAM-a).
- Stvoriti sustave datoteka. Sustav datoteka je sekcija na tvrdom disku tako formatirana da može sadržavati datoteke. UNIX i Linux koriste sustave datoteka za prikaz stabla direktorija. U tom se razlikuju od MS-DOS-a, koji smješta poddirektorije u stablo direktorija na istom logičkom pogonu. UNIX računala koriste strukturu sustava datoteka jer je smještaj poddirektorija na različite pogone puno sigurniji.

Pokvari li se jedan disk, samo će trebati zamijeniti podatke s tog diska.

Instalirati Linux sustav i potrebne programe. Nakon što ste stvorili sustave datoteka, trebate instalirati Linux operativni sustav, njegove prateće datoteke i razne programske pakete koji su sastavni dio distribucije (primjerice, igrice i paketi za mrežnu potporu).

#### Priprema instalacijskih disketa

Morate načiniti sustavsku disketu za svoje računalo. Da biste instalirali Linux, potrebno je preparticionirati tvrdi disk kako bi bilo mjesta za novi operativni sustav. Nažalost, nemate mogućnost jednostavnog kopiranja datoteka putem MS-DOS-a, OS/2 ili Windowsa NT.

Ako s računala možete pročitati CD-ROM ili ako imate slobodnih particija na tvrdom disku, odnosno ako vas nije briga što ćete uništiti postojeći sustav datoteka, Linux možete instalirati izravno s CD-a, tako da se ne morate gnjaviti s instalacijskim disketama. Preporuča se da ipak načinite disketu za module jer će vam možda zatrebati jedan od upravljačkih programa s te diskete.

BILJEŠKA Želite li izravno s CD-a pokrenuti instalaciju Linux sustava, najprije pročitajte odgovarajuće upute koje ćete dobiti s distribucijom. Nemojte zaboraviti stvoriti disketu za module, o kojoj će biti riječi u idućem odlomku.

#### Stvaranje instalacijske i diskete za module

Trebate načiniti instalacijske diskete za Linux. Potrebne su vam dvije diskete: instalacijska i disketa za module. Te ćete diskete stvoriti pomoću MS-DOS programa *rawrite*, koji možete dobiti uz gotovo sve Linux distribucije.

Program rawrite zapisuje sadržaj datoteke izravno na disketu bez obzira na format.

**BILJEŠKA** Primjeri koji se spominju u ovom poglavlju pretpostavljaju da koristite pogon D kao CD-ROM pogon. Ako to nije točno, trebate slovo D zamijeniti odgovarajućim slovom. Da biste kreirali instalacijsku disketu, iz odgovarajućeg direktorija pokrenite program rawrite.

Na taj ćete način pokrenuti program, a potom možete slijediti upute. Kada budete trebali unijeti ime datoteke, odaberite odgovarajuću instalacijsku datoteku. Kod disketnih pogona od 1.44MB, koristite datoteku install.144, a kod pogona od 2.88 MB koristite datoteku install.288.

Za disketu modula trebat ćete odabrati datoteku modules.144 (nije važno o kojoj se veličini diskete radi).

#### Instalacija Linuxa

Da biste upotrijebili diskete koje ste načinili, jednostavno umetnite instalacijsku disketu u računalo i pokrenite sustav.

Bez obzira na koji način pokrećete sustav, instalacijski program će prikazati uvodni ekran. Zatim ćete trebati podići sustav.

Ako je potrebno unijeti neke parametre da biste došli do jezgre prije nego se sustav podigne, upišite parametre u *boot*: prompt. Ako ne trebate unositi parametre, pritisnite tipku <Return> i nastavite s instalacijom.

Nakon toga sustav će pokušati odrediti koje se vrste uređaja koriste u vašem računalu. Zatim se pokreće program Linux Instalation and System Administration (LISA). Tijekom instalacije morate se služiti različitim okvirima za dijalog i pomoću tipki za kretanje odabrati ponuđene opcije u raznim popisima. Pomoću tipke <Tab> možete se u okviru za dijalog premještati s jednog elementa na drugi, odnosno s popisa na gumbe i obratno. U bilo kojem trenutku možete pritisnuti tipku <Escape> i poništiti odabir. Da biste neki element odabrali, trebate pritisnuti tipku <Return>.

Tijekom instalacije možete koristiti kombinacije tipki <Alt-F6> da biste vidjeli tijek instalacije.

**SAVJET** Prvi ekran vam omogućuje odabir jezika kojim ćete se služiti tijekom instalacije (engleski, njemački, francuski, talijanski, španjolski ili portugalski). Nakon što odaberete potrebne elemente, program LISA traži da odaberete tipku koju ćete koristiti. Operativni sustav Linux će koristiti podatke koje ste odabrali za konfiguraciju sustava i pridijeliti određene tipke određenim znakovima.

#### Uporaba prethodne konfiguracije

Zatim vas program LISA pita želite li koristiti prethodno spremljenu konfiguraciju. Caldera distribucija vam omogućuje stvaranje i spremanje nekoliko konfiguracija. Želite li naknadno mijenjati instalaciju, možete ponovno koristiti prethodno definiranu konfiguraciju tako da ne morate ponavljati isti postupak više puta. Kako prvi puta instalirate sustav LinuxOpen, jednostavno odgovorite negativno.

#### Podešavanje programa LISA

Nakon toga trebate podesiti program LISA. Obično je dovoljno prihvatiti zadana podešenja u okviru za dijalog Change LISA Setup. Ako želite unositi promjene, odaberite sljedeće opcije i načinite odgovarajuće promjene.

- Disable Plug and Play ova opcija isključuje podešenja u BIOS-u koja se odnose na Plug and Play kartice. Tijekom testiranja i podešavanja te kartice mogu stvarati probleme.
- Automatic Network Configuration with bootp opcija omogućuje drugom računalu konfiguriranje mreže na lokalnom uređaju pomoću protokola bootp. Barem inicijalno možete konfigurirati vlastite mrežne opcije umjesto da pokušavate koristiti bootp. Ipak, tu opciju Caldera OpenLinux Lite još uvijek ne podržava u potpunosti.
- Automatic Network Configuration with Netprobe opcija služi za mrežna podešavanja pomoću programa Netprobe. Netprobe je Calderin proizvod koji omogućuje daljinsko konfiguriranje sustava.
- Use Selection and Continue upućuje program da prihvati odabranu opciju i nastavi s instalacijom.

#### Provjera hardvera

Zatim će sustav sam odrediti sav hardver koji se koristi. Ukoliko program LISA ne može locirati sve hardverske komponente na vašem sustavu, morat ćete upotrijebiti disketu za module, koju ste prethodno stvorili i učitati potrebne upravljačke programe. Sustav najprije provjerava IDE i ATAPI uređaje. Pregledajte popis komponenti koji je dat u okviru za dijalog Hardware Found (IDE/ATAPI) kako biste ustanovili je li sustav pronašao sav hardver. Kliknite na gumb Continue i program LISA će vas pitati je li sav hardver pronađen. Ako to nije slučaj, odgovorite negativno i nastavite s provjerom hardvera.

Ako i nadalje ne budete mogli naći sve hardverske komponente, program LISA će prikazati okvir za dijalog Kernel Module Manager koji sadrži ove opcije:

- Nastavak instalacije (Continue with installation)
- Analiza modula jezgre (Analyze kernel modules)
- Učitavanje modula jezgre (Load kernel modules)
- Uklanjanje modula (Remove modules)

Tijekom instalacije obično ćete trebati učitati i analizirati module jezgre. Ovisno o hardveru koji instalirate, morat ćete više puta prolaziti kroz razne okvire za dijalog koji su dio okvira Kernel Module Manager.

**Analiza modula jezgre** Odaberite opciju Analyze Kernel Modules ukoliko želite saznati koje je komponente sustav pronašao. Ovu opciju možete upotrijebiti da biste vidjeli koje module trebate dodati tijekom instalacije. Također možete vidjeti poruke koje su nastale tijekom pokretanja sustava. Tablica 4.2 sadrži opis svake funkcije u okviru za dijalog Analyze Kernel.

Funkcija	Opis
Return to Previous Screen	Vraća vas u okvir za dijalog Kernel Modules.
Show Hardware That Has Already Been Found	Prikaz svih hardverskih komponenti koje su dosad nađene u sustavu.
Show Loaded Kernel Modules	Prikaz svih modula jezgre koji su dosad instalirani.
Verbose System Analysis	Detaljne informacije tijekom instalacije.
Display Boot Process Messages	Nalik informacijama koje se mogu vidjeti pomoću tipki <alt-f6>. Pojavit će se sve informacije nastale tijekom pokretanja sustava. Ti podaci uključuju rezultate provjere sustava i instaliranje raznih modula jezgre.</alt-f6>
Store Information on a DOS Floppy	Omogućuje vam stvaranje kopije svih podataka koje ste dobili pomoću funkcija Analyze Kernel Modules.

lablica 4.2 Funkcije u okviru za dijalog Analy	vze kernei
--	------------

**Učitavanje i uklanjanje modula jezgre** Funkcija Load Kernel Modules omogućava učitavanje raznih upravljačkih programa uređaja. Ti se programi nalaze na disketi za module koju ste prethodno stvorili. Tablica 4.3 nudi popis raznih podfunkcija koje možete koristiti iz okvira za dijalog Load Kernel Modules.

Tablica 4.3         Funkcije za učitavanje modula jezgre				
Funkcija	Opis			
Return to Previous Menu	Vraća vas u okvir za dijalog Kernel Modules			
Load Driver for CD-ROM	Omogućuje odabir upravljačkog programa za CD-ROM (s diskete za module)			
Load Driver for SCSI Adapter	Omogućuje učitavanje upravljačkog programa za SCSI adapter			
Load Driver for Network Card	Omogućuje vam odabir upravljačkog programa za Ethernet karticu			

Da biste učitali upravljački program koji trenutno nije dostupan, morate najprije izvaditi instalacijsku disketu iz pogona i zatim umetnuti disketu za module. Primjerice, na instalacijskoj se disketi nudi nekoliko SCSI adaptera, uključujući Adaptec 2940s. No, instalacijska disketa ne sadrži upravljačke programe za adaptere Buslogic. Da biste mogli koristiti Buslogic adapter, potrebna vam je disketa s modulima. Na ekranu će se pojaviti niz okvira za dijalog pomoću kojih trebate odabrati odgovarajući upravljački program. U tim ćete okvirima naći i dodatne informacije za podešavanje uređaja. Program LISA nudi pomoć za vrijeme instalacije. Jednostavno pritisnite tipku <F1> i uvijek ćete dobiti pomoć. Instalaciju biste trebali nastaviti prema popisu funkcija i najprije instalirati podršku za CD-ROM, zatim za SCSI uređaje i potom za mrežnu karticu. Ako uočite da ste instalirali pogrešan upravljački program ili je to učinio program LISA, odaberite opciju Remove Kernel Module i zatim uklonite pogrešno odabrani modul.

#### Priprema tvrdih diskova

Obratite pozornost na prvi redak kod ispisa programa *fdisk: Using /dev/had as default device!* Upamtite da se u sustavu MS-DOS većina particija i tvrdih diskova poziva pomoću slova C ili D. U Linuxu se one pozivaju na bitno drukčiji način. Linux koristi slične oznake za sve – uređaje, datoteke i slično.

Linux i MS-DOS komuniciraju s hardverom putem nekoliko programa nazvanih *upravljačkim programima uređaja*. Dok upravljački programi u MS-DOS-u obično imaju dodatak imenu .SYS i mogu se nalazi bilo gdje na disku, Linux sprema sve upravljačke

programe u direktorij /dev. Linux upravljačke programe koristi tijekom instalacije koja je prethodno opisana. Važno je upamtiti da, s obzirom da tvrdi disk, disketni pogon i CD-ROM predstavljaju hardverske komponente, Linux koristi upravljačke programe pohranjene u direktoriju /dev za pristup tim uređajima. Osim toga, u sustavu Linux su pogoni označeni pomoću odgovarajućih poddirektorija, a ne pomoću slova. Tablica 4.4 prikazuje Linux direktorije za razne uređaje.

Tablica 4.4 Linux uređaji	
Uređaj	Ime
Disketni pogon A	/dev/fd0
Disketni pogon B	/dev/fd1
Prvi tvrdi disk	/dev/hda
Prva primarna particija na tvrdom disku A	/dev/hda1
Druga primarna particija na tvrdom disku A	/dev/hda2
Prva logička particija na tvrdom disku A	/dev/hda4
Drugi tvrdi disk	/dev/hdb
Prva primarna particija na tvrdom disku B	/dev/hdb1
Prvi SCSI tvrdi disk	/dev/sda

Uočite da se svaki tvrdi disk označava s /hds*lovo*. Primarnim particijama su dodijeljena iduća četiri broja, a iza toga slijede logičke particije. Logičke particije uvijek počinju s /dev/hda4. SCSI tvrdi diskovi i CD-ROM-ovi slijede isto pravilo, osim što je *hd* zamijenjeno sa *sd*.

#### Uporaba Linux programa fdisk

U fdisk promptu utipkajte m da biste dobili popis naredbi. Tablica 4.5 prikazuje popis naredbi koje su vam na raspolaganju.

Tablica 4.5	Linux naredbe u programu fdisk
Naredba	Opis
a	Uključivanje/isključivanje zastavica za podizanje sustava
С	Uključivanje/isključivanje zastavica za DOS-uskladivost
d	Briše particiju
1	Popis poznatih vrsti particija
m	Prikaz naredbi
n	Dodaje novu particiju
р	Prikaz tablice particije
q	Izlazak bez spremanja promjena
t	Promjena identifikacijskog broja sustava particija
u	Promjena jedinica za prikaz/unos
V	Potvrda tablice particije
W	Zapis tablice na disk i izlazak iz programa
Х	Omogućuje dodatne funkcije (samo za stručnjake)

Za početak podjele diska na particije pokrenite naredbu p tako da se na ekranu pojavi tablica postojećih particija, koja bi trebala prikazati pogon koji ste prethodno podijelili na particije pomoću DOS programa fdisk. Ispis 4.1 prikazuje mogući ispis naredbe p.

Ispis 4.1 Primjer tablice particije Disk /dev/hda: 15 heads, 17 sectors, 1024 cylinders Units = cylinders of 255 \* 512 bytes Device Start System Boot Begin End Blocks Ιd /dev/hda2 1024 1024 4040 384667+ 51 Novell? Partition 2 has different phisical/logical endings: phys=(967, 14,17) Logical=(4096,14.17)

**BILJEŠKA** Na vašem računalu ispis može izgledati nešto drukčije od ispisa 4.1 jer se razlikuju vrijednosti svake vrste pogona i particija koje su već definirane na tom pogonu.

U ispisu 4.1 možete vidjeti da su već definirane različite particije, početak i kraj particije i veličina blokova. U ispisu možete uočiti i vrstu particije. Tablica 4.6 prikazuje sve vrste particija koje možete definirati pomoću Linux programa fdisk. Najčešće ćete koristiti 83-Linux e i 82-Linux Swap. Sličan ispis možete dobiti i ako pokrenete naredbu *l*.

Tablica 4.6	Vrste Linux particija
Referentni broj	Vrsta
0	prazna
1	DOS 12-bit FAT
2	XENIX root
3	XENIX usr
4	DOS 16-bit<32M
5	proširena
6	DOS 16-bit>=32M
7	OS/2 HPFS
8	AIX
9	AIX za pokretanje sustava
a	OS/2 Boot Manager
40	Venix 80286
51	Novell?
52	Microport
63	GNU HURD
64	Novell
75	PC/IX
80	Old MINIX
81	MINIX/Linux
82	Linux Swap
83	Linux Native
93	Amoeba
94	Amoeba BBT
a5	BSD/386
b7	BSDI fs

Referentni broj	Vrsta
b8	BSDI swap
c7	Syrinx
db	CP/M
e1	DOS access
e3	DOS R/O
f2	DOS secondary
ff	BBT

U ispisu 4.1 Linux na dnu ekrana daje napomenu o različitim fizičkim i logičkim završecima. Ta razlika postoji zbog sustava koji se koristi za pisanje ovog poglavlja.

Particija koja sadrži D pogon (DOS) nije mijenjana, dok je C pogon preparticioniran na manju veličinu kako bi se načinilo mjesto za Linux. Zbog toga postoji prostor između C i D pogona, na kojem će biti stvorene potrebne Linux particije.

U ispisu 4.1 su vrlo važni brojevi u stupcima begin, start i end i njih biste trebali zapisati na papir, jer će vam kasnije trebati kod određivanja potrebne veličine particija.

#### Dodavanje potrebnih particija

S obzirom da ste preparticionirali pogon za MS-DOS, ne trebate brisati niti jednu particiju za Linux. Potrebno je samo dodavati particije. Standardan set particija bi trebao sadržavati sljedeće:

- / (root) particiju za čitav sustav
- swap particiju za swap datoteku
- /usr particiju za softver
- /home particiju za korisničke direktorije
- /var particiju za log datoteke.

Da biste dodali particiju, pokrenite naredbu n i na ekranu će se pojaviti ovo:

```
Command Action
e extended
p primary (1 -4)
```

Pritisnite tipke i <Return>, a program fdisk će tražiti da unesete broj particije. Unesite potreban podatak i pritisnite tipku <Return>. Ako odaberete broj particije

koja se trenutno koristi, program će vas na to upozoriti i tražiti da obrišete particiju prije nego je pokušate dodati u tablicu particija. U tom primjeru unesite broj *3* da biste dodali treću primarnu particiju /dev/hda3.

Zatim će program zatražiti da unesete lokaciju prvog cilindra. Obično je to prvi slobodni cilindar. U osnovi program fdisk prikazuje raspon koji ste odabrali, primjerice:

```
First cylinder (42 - 1024) :
```

Uočite da prva particija završava na 41. cilindru i da iduća particija počinje na 1024. cilindru. Iduću particiju možete započeti bilo gdje između 42. i 1024. cilindra. Dobro bi bilo da particiju ne postavite bilo gdje na disku već da odaberete iduću slobodnu lokaciju (u ovom slučaju 42. cilindar).

BILJEŠKA Linux može imati problema kod pokretanja s particija čiji je početak iznad 1024. cilindra. Ako stvarate Linux particiju samo u tom rasponu, tada Linux morate pokrenuti s diskete. U nastavku ovog poglavlja ćete naučiti kako se stvara boot disketa sustava (razlikuje se od boot diskete za instalaciju). Razlika je u tome što pokretanje s diskete traje nešto duže od pokretanja Linuxa s tvrdog diska. Ako imate IDE pogon s više od 1024 cilindra, pročitajte odlomak o rješavanju problema koji se nalazi u 2. poglavlju.

#### Vidi "Otklanjanje problema", stranica 52

Nakon toga program fdisk traži da odredite koliko prostora želite dodijeliti toj particiji. Veličinu potrebnog prostora možete izraziti u cilindrima ili u broju bajtova (+size), kB (+sizeK) ili u MB (+sizeM). S obzirom da biste već trebali znati približnu veličinu prostora koji vam je potreban za *swap* datoteku, najprije definirajte *swap* particiju, a potom ostatak prostora na disku ostavite za ostale particije. Tako, primjera radi, ukoliko vaše računalo ima 8MB RAM-a, pa trebate odrediti particiju veličine 16 MB , upišite sljedeće:

Last cylinder or +size or +sizeM or +sizeK (42-1023): +16M

Nakon toga biste trebali pokrenuti naredbu p i potražiti novu particiju koju ste definirali. U ovom primjeru bi nova tablica particije trebala ovako izgledati:

```
Disk /dev/had: 15 heads, 17 sectors, 1024 cylinders
Units=cylinders of 255 * 512 bytes
Device
            Boot
                    Begin
                            Start
                                    End
                                           Blocks
                                                      Ιd
                                                               System
/dev/hda1
                    1
                            1
                                    41
                                           5219 1
                                                      DOS
                                                           12-bit FAT
/dev/hda2
                    1024
                            1024
                                    4040
                                           384667+
                                                      51
                                                              Novell?
Partition 2 has different phisical/logical endings:
phys=(967, 14,17) Logical=(4039,14.17)
/dev/hda3
                    42
                            42
                                    170
                                           16447 +
                                                      83
                                                              Linux native
```

Prema osnovnim postavkama, program fdisk stvara novu particiju tipa Linux Native. Da biste stvorili *swap* particiju, potrebno je pokrenuti naredbu *t*. Utipkajte *t* i zatim unesite broj particije koju želite promijeniti (u našem primjeru je to broj 3). Program fdisk traži da zatim unesete heksadecimalnu vrijednost koja označava vrstu particije (pogledajte tablicu 4.6). Ako tablicu nemate pri ruci, upišite broj **1** i dobit ćete popis oznaka. S obzirom da želite *swap* particiju, trebate unijeti broj **82**.

Kao što možete vidjeti, program fdisk će vas izvijestiti o novoj vrsti particije, ali možete pokrenuti i naredbu *p* da biste provjerili da je particija 3 postala Linux *swap* particija.

Nakon toga možete dodavati Linux particije. Primjerice, dodajte samo jednu particiju. No, ako iz bilo kojeg razloga želite više particija, to možete odmah učiniti. Da biste dodali particiju, pritisnite tipku <n>, utipkajte p za drugu primarnu particiju i zatim odaberite broj te particije (4). Da biste izbjegli "ispreplitanje" raznih particija na disku, započnite novu particiju na kraju prethodne, na 171. cilindru. S obzirom da želite ostatak prostora koristiti za Linux particije, tada odredite broj zadnjeg cilindra umjesto točnog broja bajtova. Upišite **1023**, kao što je pokazano u nastavku:

```
Command (m for help): n
Command action
e extended
p primary particion (1 - 4)
p
Partition number (1 - 4): 4
First cylinder (171 - 1024): 171
Last cylinder or +size or +sizeM or sizeK (171 - 1023): 1023
```

Sada pomoću naredbe *p* možete provjeriti nove particije. Ako želite unijeti još neke promjene, to možete odmah učiniti. Ako ste zadovoljni s izgledom svojih particija, tada možete pomoću naredbe *w* zapisati tablicu particija na tvrdi disk.

Nijedna od promjena koje ste načinili nije trajna sve dok ne pokrenete naredbu w, tako da ukoliko mislite da ste načinili neke pogrešne promjene, možete upotrijebiti naredbu q i izaći bez spremanja tablice particija. Ako upotrijebite naredbu w, Linux će vas obavijestiti da je zapamćena tablica particija i zatim će se disk preurediti tako da odgovara novoj tablici particija. Ako se vaš sustav u tom trenutku zaglavi, ponovno ga pokrenite pomoću instalacijske diskete.

#### Stvaranje swap particije

Nakon što ste podesili particije, LISA traži da podesite *swap* particiju. Iz okvira za dijalog Configure Swap Space odaberite *swap* particiju i zatim kliknite na gumb Continue. LISA će zatim konfigurirati, formatirati i aktivirati *swap* područje.

### Instalacija Linux softverskog sustava

Sada kada je računalo particionirano za Linux, moći ćete instalirati različite softverske pakete koji čine softverski sustav OpenLinux. LISA na ekranu pokazuje okvir za dijalog Installation Source Selection u kojem možete odabrati medij s kojeg ćete instalirati softver: CD-ROM, tvrdi disk ili NFS (Network File System).

Najjednostavnije je instaliranje s CD-ROM-a, ali isto tako možete njegov sadržaj kopirati na tvrdi disk i pristupiti mu odande. Ako ste priključeni na mrežu, možete se povezati s drugim računalom i instalirati softver s tog računala. Takav način instalacije može biti koristan ukoliko OpenLinux ne može prepoznati vaš CD-ROM.

Nakon što izaberete način instalacije, trebat ćete slijediti odgovarajuće upute (upamtite da pomoć uvijek možete dobiti ako pritisnete tipku F1). Primjerice, ukoliko ste odabrali instalaciju s CD-ROM-a, trebat ćete odabrati odgovarajući hardverski uređaj. Program obično samo odabere pravi uređaj.

Potom trebate odabrati korijensku particiju (/) na koju će sav softver biti kopiran. Nakon što je odaberete iz popisa particija, LISA formatira i priprema korijensku particiju za instalaciju. Za to vrijeme ćete na ekranu vidjeti brojeve i čuti disk kako radi, no neka vas to ne brine.

Nakon što je pripremljena korijenska (/) particija, LISA traži da upišete druge direktorije (ispod *root* direktorija) u njihove vlastite particije. U ovom slučaju budite pažljivi: ponuđeni je odgovor negativan, ali vi želite postaviti direktorije /usr, /home i /var u particije koje ste prethodno stvorili. Stoga odgovorite potvrdno i taj postupak ponovite za svaki direktorij.

Nakon što ste stvorili oslonce svog sustava datoteka, LISA traži da unesete pakete koje namjeravate instalirati. Osnovni sustav zahtijeva oko 50 MB prostora, a puna instalacija oko 700 MB. Možete odabrati nešto s ovog popisa:

Minimalni sustav bez XWindowsa (49 MB) Minimalni sustav s XWindowsima (68 MB) Mali standardni sustav (121 MB) Standardni sustav (349 MB) Svi paketi (688 MB)

Nakon što odaberete veličinu instalacije, LISA će generirati popis paketa i potom započeti automatsku instalaciju paketa. Možete se opustiti, nasloniti u naslonjaču i gledati kako se instalira sustav.

# Podešavanje sustava

Nakon instalacije paketa možete pristupiti podešavanju sustava. Ako ste spojeni na mrežu, trebat ćete saznati određene podatke od svog davatelja usluga ili administratora mreže. Također trebate odrediti *root* lozinku. Radi se o važnoj lozinki - stoga je pažljivo odaberite i nemojte je zaboraviti.

Za početak će vas LISA tražiti ime *hosta*. Radi se o imenu pod kojim drugi korisnici u mreži vide vaše računalo. Također ćete trebati odrediti ime *domene*, koje je obično nalik imenu tvrtka.com. Pomoću imena *hosta* i *domene* stvorit ćete puno ime računala (primjerice opus.netwharf.com).

#### Vidi "IP adrese", stranica 469

Također trebate upisati IP adresu svog računala zajedno s njegovom mrežnom maskom (*netmask*) i podešenim *gatewayom*. Nakon što unesete te podatke, trebate specificirati ime svog mrežnog poslužitelja ili DNS računala.

Nakon što podesite mrežu, podesite svoj sat i vremensku zonu u kojoj radite. Preporuka je da koristite lokalno vrijeme (iako mnogi poslužitelji koriste GMT), jer većina osobnih računala ima podešene BIOS satove na lokalno vrijeme. Ako vrijeme na osobnom računalu podesite prema GMT-u, mogu vam se javiti problemi, pogotovo ako na svom stroju koristite i druge operativne sustave. Odaberite lokalno vrijeme ili GMT vrijeme i zatim odredite vremensku zonu, kao što je primjerice EST. Zatim morate zadati miša koji koristite. Većina ATX sustava koriste PS/2 miša. Ako vi koristite serijskog miša, provjerite na koji je priključak priključen. Zatim odaberite pisač koji želite koristiti.

Vidi "Osnovna terminologija koja se koristi u ovom poglavlju", stranica 146.

Sada odaberite *root* lozinku, koja omogućava bilo kome da radi bilo što na vašem računalu (*super korisnik*). Nikome nemojte pokazati ovu lozinku! I, naravno, nemojte je zaboraviti. Ako zaboravite lozinku, vrlo je vjerojatno da ćete trebati ponovo instalirati sustav.

BILJEŠKA Pročitajte 12. poglavlje "Poboljšanje sigurnosti sustava" i saznat ćete nešto više o lozinkama i mogućim opcijama ako zaboravite lozinku.

Nakon što ste podesili svoju *root* lozinku, trebate stvoriti korisnički račun prvog korisnika. LISA koristi izvorno ime *col* (Caldera OpenLinux), ali vi možete odabrati bilo koje ime, čak i svoje vlastito. Taj vam prvi korisnički račun omogućava uporabu sustava poput regularnog korisnika, a ne kao dosad kada ste bili tzv. *super korisnik*. Ne biste trebali koristiti pristup *super korisnika* za svakodnevne zadaće koje obavljate na računalu. Nakon što odredite ime novog korisnika, jednostavno prihvatite ponuđene vrijednosti koje se traže u nastavku.

Te su informacije objašnjene u 10. poglavlju "Upravljanje korisničkim računima".

Vidi "Dodavanje korisnika", stranica 220

Potom odaberite kako želite da se vaš sustav pokreće.

## Instalacija LILO

LILO je kratica od LInux LOader i predstavlja program koji se pokreće prilikom pokretanja sustava i omogućava odabir operativnog sustava koji će se koristiti za pokretanje računala. Program LILO možete koristiti za pokretanje nekoliko različitih operativnih sustava, kao što su Linux i MS-DOS. Pomoću programa LILO možete odrediti i osnovni operativni sustav kojim će se podizati računalo kao i vremensko ograničenje čekanja prije nego se pokrene taj sustav. Primjerice, ako na svom računalu imate MS-DOS i Linux, onda pomoću programa LILO možete odabrati bilo koji za pokretanje računala. Možete zadati programu da pokrene MS-DOS ukoliko nema intervencije unutar 30 sekundi. Prije nego istekne tih 30 sekundi, korisnik može odabrati i drugi operativni sustav (umjesto osnovno zadanog). Možete pritisnuti neku od tipki <Ctrl>, <Alt> ili <Shift> i prekinuti odbrojavanje. Pritisnite tipku <Tab> da biste dobili popis operativnih sustava koje možete zadati programu LILO.

Sve te podatke trebate zadati tijekom podešavanja programa LILO. Iako možete izravno uređivati datoteku lilo.conf, koja se nalazi u direktoriju /etc, LILO INSTALLATION nudi bolje sučelje za uređivanje datoteke.

Nakon što podesite sustav, Setup vam omogućava instalaciju programa LILO s mogućnošću naknadnog podešavanja.

#### Deinstalacija programa LILO

Ako koristite LILO inačicu 0.14 ili noviju, tada LILO možete deinstalirati pomoću ove naredbe:

opus: ~# lilo -u

Ako imate prethodnu inačicu, onda najprije morate ukloniti ili isključiti LILO s njegove primarne particije. Možete upotrijebiti Linuxov ili MS-DOS-ov program fdisk kako biste aktivirali drugu particiju.

Ako ste LILO smjestili unutar MBR-a (*master boot record*), tada ga morate zamijeniti s drugim MBR-om iz drugog operativnog sustava. U MS-DOS-u 5.0 (ili novije), naredbom c:\>fdisk /mbr obnovit ćete MS-DOS-ov MBR.

Nakon što ste LILO uklonili iz aktivne particije ili MBR-a, možete slobodno ukloniti datoteke iz direktorija etc/lilo.

Vidi "Uklanjanje datoteka ili direktorija", stranica 323

# Vraćanje na početak

Nakon što ste završili podešavanje sustava, program Setup vas vraća u glavni izbornik, u kojem možete odabrati opciju EXIT i napustiti program. Ako želite mijenjati opcije, ovdje to možete učiniti (u slučaju da tijekom instalacije niste mijenjali opcije, onda u 6. poglavlju, "Nadogradnja sustava i instalacija softvera pomoću RMP-a", potražite informacije koje se odnose na ažuriranje i instalaciju softvera nakon početne instalacije sustava). Odaberite EXIT i napustite program Setup.

Odaberete li EXIT, vratit ćete se u sustavski *prompt*, koji možete prepoznati po znaku #. Sada se nalazite u sustavu Linux i možete jednostavno pokrenuti neke naredbe kao što je *ls* (za ispis datoteka u trenutnom direktoriju). Nakon toga biste mogli ponovno pokrenuti sustav kako bi sve promjene bile prihvaćene.

Ponovno pokretanje Linuxa je bitno složenije od pokretanja DOS-a. Ne možete samo isključiti napajanje i ponovno uključiti računalo. Učinite li to u Linuxu, oštetit ćete strukturu datoteka. Sustav Linux se pokušava "oporaviti" tijekom podizanja. Ne isključujte napajanje dok radite s Linuxom. Za izlazak iz Linuxa koristite sljedeću naredbu:

```
shutdown [-r] time
```

Zastavica –r označava da bi se sustav trebao ponovno pokrenuti nakon isključivanja. Parametar *time* označava vrijeme nakon kojeg se sustav treba isključiti. Umjesto parametra *time* možete upotrijebiti *now*, ako želite trenutno isključiti sustav. Linux će također prepoznati kombinaciju tipki za pokretanje računala koja se koristi u DOS-u (<Ctrl-Alt-Delete>), i to kao naredbu:

shutdown -r now

Vidi "Isključivanje sustava, stranica 215

Provjerite jeste li izvadili sve diskete iz pogona i ponovno pokrenite svoje novo Linux računalo.

# Rješavanje problema

Nakon što ponovno pokrenete svoje računalo, pojavit će se LILO *prompt*. Provjerite možete li pokrenuti svoj stari operativni sustav (ako ste ga sačuvali na tvrdom disku). Ako se radi o DOS-u, pritisnite tipku <Shift> i potom upišite kraticu koju ste koristili za identifikaciju DOS particije (prilikom instalacije LILO-a). Unesete li pogrešnu riječ, pritisnite tipku <Tab> da biste dobili popis raspoloživih vrsta operativnih sustava. Ukoliko se u tom trenutku pojave problemi, postavite svoju DOS *boot* disketu u odgovarajući pogon i pokrenite sustav.

# Odavde...

Nakon što ste pokrenuli sustav koji radi, možete pročitati iduća poglavlja i saznati još neke informacije o Linuxu:

- 3. poglavlje, "Instaliranje Red Hata" jer se dobar dio instalacije Caldere zasniva na starijoj inačici Red Hat distribucije, uključujući i uporabu RPM-a
- 5. poglavlje, "Pokretanje Linux programa" omogućuje vam da ubrzate različite programe koje ste upravo instalirali
- 6. poglavlje, "Nadogradnja i instaliranje softvera pomoću RPM-a" sadrži informacije kako ponovno instalirati pakete koje ste trebali ukloniti tijekom početnog podešavanja Linux sustava
- 12. poglavlje, "Poboljšanje sigurnosti sustava" sadrži savjete za dodjelu lozinki i upute kako otkloniti greške.

# POGLAVLJE 5

# Pokretanje Linux programa

(Jack Tackett)

#### U ovom poglavlju

Rad s Linuxom Upravljanje korisnicima Uporaba osnovnih naredbi Rad s DOS datotekama pod Linuxom Izlazak iz Linuxa Pokretanje Linux programa Igre **124** Pokretanje DOS programa pod Linuxom Pokretanje Windows programa pod Linuxom Sada kad ste instalirali operativni sustav Linux, možete pročitati sažete upute za podešavanje korisničkog računa te savladati osnovne naredbe za kretanje u novom sustavu. Radi se o vašem vlastitom višezadaćnom, višekorisničkom sustavu u kojem je dozvoljeno eksperimentiranje. Stoga, krenite hrabro naprijed i počnite raditi u Linuxu, jer takvu prigodu možda nikada nećete imati u UNIX sustavu.

No, samo igranje s operativnim sustavom i nije toliko zabavno, a i na taj način nećete obaviti svoje svakodnevne zadaće. Osim toga, niti DOS ne koristite čitav dan, zar ne? Upotrebljavate samo njegove aplikacije. Linux omogućava pristup tisućama programa. Neke ste već vjerojatno instalirali. No, još vam ih je puno na raspolaganju. Za operativni sustav Linux već postoje programi koji inače za PC platforme koštaju i stotine dolara.

#### Rad s Linuxom

Nakon što ste instalirali i ponovno pokrenuli Linux, uočit ćete *prompt* s imenom sustava (koje mu je dodijeljeno tijekom instalacije). *Prompt* bi trebao izgledati slično ovome:

```
Red Hat Linux release 5.0 (Hurricane)
Kernel 2.0.31 on an I486
web login:
```

Naravno, ova poruka može prikazivati različitu inačicu jer se radi o sustavu koji se razvija.

Potom trebate unijeti korisničko ime i lozinku. Pomoću korisničkog imena će vas sustav prepoznati (Linux može podržavati više korisnika i to istovremeno ili u različitim vremenima). Osim toga, svaki korisnik dobiva svoj direktorij, nazvan *polazni (home)* direktorij. Mnogi korisnički računi su tako podešeni da ograničavaju korisnike na uporabu određenih direktorija kako bi ih spriječili da pokreću određene naredbe (prvenstveno da jedan korisnik ne bi mogao gledati datoteke drugog korisnika).

#### Unos naredbi

Unos naredbi u Linuxu je vrlo sličan unosu u DOS-u i drugim operativnim sustavima kod kojih se naredbe unose nakon *prompta* (retka za naredbe). Linux je, kao i UNIX, osjetljiv na velika i mala slova. Stoga, ako sustav nije prepoznao naredbu koju ste unijeli, provjerite jeste li je ispravno napisali, uključujući i velika i mala slova. Većina naredbi se izvodi nakon što pritisnete tipku <Return>.

#### Ponovni poziv prethodnih naredbi

Sustav Linux omogućava i ponovno pozivanje prethodno upisanih naredbi. Možete pritisnuti tipku sa strelicom prema gore i pozvati prethodnu naredbu, a potom pritisnuti tipku <Return> i pokrenuti je. Da biste vidjeli čitav popis naredbi koje ste unijeli, pokrenite naredbu history:

```
[tackett@web~]$ history
1 clear
```

```
2 adduser
3 history
```

Nakon što pogledate prethodni popis naredbi, možete neku od naredbi ponoviti tako da pritisnete tipku sa strelicom prema gore i tražite odgovarajuću naredbu ili da pritisnete tipku <!> i unesete broj naredbe koju želite ponovno izvesti. Primjerice, ako želite

ponoviti naredbu adduser (pogledajte prethodni popis), trebali biste upisati sljedeće:

```
[tackett@web~]$ !2
```

Broj elemenata u popisu prethodno pokrenutih naredbi definiraju sami korisnici u datoteci za podešavanje korisničkog profila. Pogledajte 18. poglavlje "Što su Linux ljuske" i saznajte nešto više o datoteci .profile.

BILJEŠKA Linux nudi mnoge različite ljuske od koji neke ne nude mogućnost korištenja naredbe history.

#### Kako nešto odabrati

Ako u sustavu imate miša, te ako ste instalirali program selection, moći ćete miša koristiti za kopiranje teksta s drugih dijelova zaslona u redak za naredbe. Da biste odabrali željeni tekst, jednostavno pritisnite i zadržite lijevu tipku miša te pokazivač (koji se pojavljuje čim kliknete na lijevu tipku miša) povucite preko određenog teksta, a zatim pritisnite desnu tipku miša da biste tekst kopirali u redak za naredbe. Ovo je izuzetno korisno ako želite dugačka imena datoteka kopirati u redak za naredbe.

#### Dovršetak unosa naredbi

Linux nudi i mogućnost dovršavanja naredbi prilikom njihova upisivanja. Možete započeti pisati ime datoteke i zatim pritisnuti tipku <Tab>. Linux će pretražiti direktorij kako bi našao datoteku koja počinje sa slovima koja ste upisali, te, ako je nađe, dovršit će upis u redak za naredbe. Ako Linux ne može naći jedinstveno ime datoteke, čut ćete zvučni

signal, a ime će biti dovršeno do zadnjeg zajedničkog znaka. Primjerice, ako želite kopirati datoteku koja se zove todo Monday u datoteku todo today, u prompt upišite cp to i potom pritisnite tipku <Tab>. Čut ćete zvučni signal, a redak za naredbe će izgledati ovako:

[tackett@web~]\$ cp todo\_

Ako nakon toga upišete slovo *m* i pritisnete tipku <Tab>, Linux će u redak za naredbe upisati ime datoteke todo monday.

# Upravljanje korisnicima

U mnogim sustavima osobe koje su odgovorne za stvaranje korisničkih računa nazivaju se administratorima sustava i podešavaju korisničke račune te obavljaju druge zadaće. Više informacija o raznim aspektima adminstriranja sustava pročitajte u drugom dijelu, "Administrator sustava". U vašem Linux sustavu vi ste administrator sustava i vaša je zadaća podešavanje svog računa, kao i računa članova obitelji i prijatelja.

Kao administrator sustava morate sebi otvoriti račun. Administratori sustava se ponekad nazivaju i *super korisnicima* zbog toga jer imaju tolike ovlasti u sustavu. Da biste započeli s korištenjem Linuxa, najprije se trebate prijaviti kao super korisnik.

#### Prijava i odjava

Najprije u prijavno polje (login prompt) trebate upisati **root**, nakon čega će Linux zatražiti lozinku.

Uporabom lozinke sprečavate neovlaštene osobe da se prijavljuju na bilo koji račun. Putom korisničkog imena Linux provjerava radi li pravi korisnik za računalom. Svoje lozinke ne biste trebali nikome davati. Linux štiti lozinke koje upisujete tako da ih ne prikazuje na zaslonu - stoga trebate biti pažljivi kod unosa lozinke.

Upišete li neispravno korisničko ime ili lozinku, pojavit će se sljedeća poruka:

```
web login: jack
Password: password
Login incorrect
```

```
web login:
```

S obzirom da se sada prvi puta prijavljujete za rad nakon instalacije, *root* račun još nema podešenu lozinku, te će se, nakon što upišete **root**, odmah pojaviti redak za naredbu. Nakon toga možete pokrenuti neku naredbu u Linuxu. Naredbe se unose na sličan način kao u DOS-u: upiše se naredba s potrebnim parametrima i pritisne se tipka <Return>.

Da biste se odjavili, upišite **logout**. Pomoću ove naredbe vraćate se u postupak prijave. Ako naredba *logout* ne radi, pokušajte s naredbom *exit*.

#### Dodavanje korisnika u Slackware distribuciji

Nakon što ste se prijavili u *root* prompt, trebali biste sami sebi otvoriti račun za dodavanje računa, upišite ovu naredbu:

[root@web~]# adduser
Adding a new user. The user name should be not exceed 8 characters in
length, or you many run into problems letter.

Enter login name for new account (^C to quit):

Pogledajte na trenutak zaslon i uočite redak za unos naredbi nakon što ste upisali naredbu, a koji počinje s imenom računala (*host*), odnosno s imenom koji ste unijeli prilikom instalacije. Nakon imena računala slijedi znak tilde (~) koji se odnosi na polazni direktorij korisnika (bit će opisan kasnije). U ovom slučaju taj znak predstavlja direktorij u kojem se korisnik trenutno nalazi. Ako iz direktorija /usr/bin pokrenete naredbu adduser, *prompt* će izgledati ovako:

[root@web~]# /usr/bin#

Iduća oznaka (#) predstavlja superkorisnički račun. Normalni korisnički računi obično imaju oznaku \$.

Zatim ste možda mogli uočiti pogrešno napisane riječi koje ne utječu na sustav, ali naglašavaju činjenicu da Linux, iako sjajan i funkcionalan sustav, ipak nije komercijalni proizvod.

Sada upišite korisničko ime i pritisnite tipku <Return> (ime smije imati najviše 8 znakova). Slijedi primjer otvaranja računa za Jacka Tacketta:

```
Enter login name for new account (^C to guit) : jack
Editing information for new user [jack]
Full Name: Jack Tackett, Jr.
GID[100] :<Return>
Checking for an available UID after 500
501...
First unused uid is 502
UID [502] :<Return>
Home Directory [/home/jack]:<Return>
Shell [/bin/bash] :<Return>
Password: opus
Information for new user [jack]:
Home directory: [/home/jack] Shell: [/bin/bash]
Password: [opus] uid: [502] gid: [100]
Is this correct? [y/N] :y
Adding login [jack] and making directory [/home/jack]
Adding the files from the /etc/skel directory:
. / .kermc -> /home/jack/ . / .kermc
. / .less -> /home/jack/ . / .less
. / .lessrc -> /home/jack/ . / .lessrc
. / .term -> /home/jack/ . / .term
    .term/termrc -> /home/jack/ . / .termrc
. / .emacs -> /home/jack/ . / .emacs
[root@web ~ ]#
```

Tijekom ovog postupka trebate upisati puno ime korisnika kako biste kasnije mogli odrediti kome račun pripada. Zatim trebate upisati identifikacijski broj grupe i korisnika. No neka vas to ne zabrinjava jer te brojeve Linux koristi za određivanje direktorija i datoteka kojima je moguće pristupiti. Možete prihvatiti i predložene vrijednosti (one su u zagradama) tako da jednostavno pritisnete tipku <Return>.

Zatim trebate unijeti polazni (*home*) direktorij korisnika, odnosno direktorij u kojem će se korisnik automatski naći nakon prijave. To je korisnikovo područje u kojem može spremati datoteke. Linux nudi direktorij čiji naziv određuje prema imenu korisnika. Ako vam je to ime prihvatljivo, tada samo pritisnite tipku <Return>, a u protivnom upišite drugo ime i pritisnite <Return>. Za sada prihvatite predloženo podešenje koje nudi naredba adduser.

Potom treba odrediti ljusku korisnika. Ta je ljuska nalik datoteci COMMAND.COM u DOS-u i služi za izvođenje naredbi. Ljuska može prihvatiti ulazne parametre i pokrenuti određene naredbe. Tijekom instalacije Linuxa koristili ste ljusku *bash*. Za sada jednostavno prihvatite ponuđenu opciju *bash*.

#### Vidi "Što su ljuske", 339. stranica

Zadnji parametar koji treba unijeti predstavlja lozinka računa. Izričito vas upozoravamo da biste trebali predvidjeti lozinku za svaki račun. Nakon toga će Linux prikazati na zaslonu sve informacije koje ste unijeli i zatražiti da ih potvrdite. Ako je neki podatak netočan, upišite **n** (ili jednostavno pritisnite tipku <Return>, jer se podrazumijeva negativni odgovor). Sada se morate vratiti i ispraviti pogrešku. Ako je sve u redu, onda upišite **y**.

Na zaslonu će se pojaviti popis datoteka koje su kopirane iz direktorija ./etc/skel u novi korisnički direktorij. Radi se o datotekama za podešavanje, primjerice, korisničkog terminala ili načina pokretanja programa *emacs* i *less*. Korisnik može u bilo kojem trenutku promijeniti te datoteke i na taj način utjecati na "ponašanje" programa.

Nakon što ste dodali račun, na dva načina možete provjeriti da li on zaista postoji: brži je način da pokrenete uslužni program *finger* i provjerite ima li pojedini korisnik odgovarajući račun. Sintaksa naredbe finger je: finger *name*. Slijedi primjer kako možete provjeriti račun koji ste upravo stvorili:

```
[root@web~]# finger jack
Login: jack Name Jack Tackett, Jr.
Directory: /home/jack Shell: /bin/bash
Never logged in.
No Mail.
No Plan.
[root@web~]#
```

Ako korisnik ima račun, na zaslonu ćete ugledati odgovarajuće informacije, a u protivnom ćete vidjeti poruku da takav korisnik ne postoji.

Drugi način provjere stvorenog računa jest da se doista prijavite za rad i vidite da li vam Linux to omogućava. To možete učiniti na sljedeće načine:

- Možete se odjaviti i zatim prijaviti kao novi korisnik.
- Možete pokrenuti naredbu *su* (koja je kratica od *switch user*).
- Možete pokrenuti naredbu login
- Možete upotrijebiti jedan od šest virtualnih terminala i prijaviti se na novi račun (upamtite, Linux je višekorisnički sustav).

U tablici 5.1 naći ćete pregled navedenih metoda.

Naredba	Opis
logout	Odjava iz root računa i povratak u postupak prijave (login). Više nemate pristup root računu sve dok se ponovno ne prijavite kao super korisnik.
su korisničko ime	Odjava s root računa. Za prijavu je potrebno unijeti samo lozinku jer vas sustav neće tražiti korisničko ime. Ako ne upišete korisničko ime, sustav podrazumijeva da se želite prijaviti u root i traži da unesete root lozinku.
login korisničko ime	Gotovo identična naredbi su, osim u slučaju kada ne upišete korisničko ime. Tada će vas sustav vratiti u normalan postupak za prijavu (login).
<alt-fx></alt-fx>	Omogućeno korištenje virtualnih terminala. Virtualnom terminalu možete pristupiti pritiskom na tipku <alt> i jedne od funkcijskih tipki (F1 do F6). Ugledat ćete još jedan prijavni prompt u kojem se možete prijaviti kao novi korisnik. Kod korištenja virtualnih terminala zgodno je što se još uvije nalazite i u prvobitnom računu tako da možete bez odjave prelaziti s jednog računa na drugi pritiskom na tipke <alt-fx>.</alt-fx></alt>

#### Tablica 5.1 Prijava za rad na novom korisničkom računu

Ako korisnika želite naknadno dodati s računa koji već postoji, nećete moći koristiti BILJEŠKA naredbu adduser jer neke naredbe može koristiti samo super korisnik (a naredba adduser je jedna od njih). Naiđete li na problem tijekom dodavanja novog korisnika, provjerite jeste li se prijavili kao super korisnik.

#### Dodavanje korisnika u Red Hat distribuciji

Red Hat distribucija automatski obavlja mnoge funkcije naredbe adduser. Da biste dodali novog korisnika, trebate u redak za naredbu upisati sljedeće:

[root@web~]# adduser jack

Ova naredba predstavlja shell skriptu koja se nalazi u direktoriju /usr/sbin. Da biste pokrenuli ovu naredbu, morate biti super korisnik.

Skripta, obična ASCII datoteka, sadrži direktorije i datoteke koje su potrebne novom korisniku. Ostalo je samo da podesite korisnikovu lozinku i on se odmah može prijaviti. Promjena lozinke je naknadno objašnjena u odlomku "Mijenjanje lozinki".

**Vidi** "Pregled sadržaja datoteke", **325. stranica.** 

#### Uporaba Control Panela za upravljanje korisnicima

Ako ste s Red Hat distribucijom instalirali paket XFree86, tada možete koristiti prozor User/Group Manager (slika 5.1) za dodavanje korisnika, mijenjanje podešenja i brisanje ili deaktiviranje korisnika. U popisu odaberite ime korisnika čiji račun želite mijenjati i kliknite na odgovarajući gumb. Tablica 5.2 opisuje ulogu svakog gumba.

Slika 5.1	REER	HS Linux User/Gr	oup Manager		同居
Okvir za dijalog RHS	UserCfg				
Linux User/Group	Name	Passwd Group	Full Name	Home directory	
Manager omogućuje pregled i mijenjanje informacija u direktoriju /etc/passwd.	root bin daemon adm lp sync shutdown halt mail	exists root locked bin locked daemon locked adm locked Ip locked root locked root locked root locked mail	root bin daemon adm lp sync shutdown halt mail	/root /bin /sbin /var/adm /var/spool/lpd /sbin /sbin /var/spool/mail /var/spool/mail	
	Add	Deactin	vate Reactivate	Remove Edit	Exit

Tablica 5.2	Gumbi u prozoru RHS Linux User/Group Manager
Gumb	Opis
Add	Pojavljuje se okvir za dijalog Add User koji omogućuje podešavanje određenih korisničkih atributa, kao što su polazni (home) direktorij i lozinka.
Deactivate	Omogućuje deaktiviranje korisničkog računa koji će se kasnije ponovno koristiti. Možda ćete trebati deaktivirati račun nediscipliniranog korisnika i njegove datoteke komprimirati kako bi zauzimale što manje prostora na disku.
Reactivate	Omogućuje ponovno aktiviranje korisničkog računa.
Remove	Brisanje korisnika iz sustava. Uklanjaju se korisnikove datoteke i direktoriji (možete ih pohraniti na disketu prije nego ih obrišete).
Edit	Omogućuje uređivanje korisničke lozinke (ako je korisnik zaboravio svoju lozinku), grupe ili ljuske koja se koristi.
Exit	Izlazak iz prozora RHS Linux User/Group Manager.

Kliknite na gumb Add i na zaslonu će se pojaviti okvir za dijalog Add User, koji je prikazan na slici 5.2, u kojem možete u određena polja unositi potrebne informacije. U tablici 5.3 su opisana polja koja se pojavljuju u okviru Add User.

Tablica 5.3	Opcije u okviru za dijalog Add User
Polje	Opis
Username	Ime korisnika
Password	Korisnička lozinka se unosi tako da kliknete na strelicu kraj polja i u izborniku odaberete Edit. Na zaslonu će se pojaviti novi okvir

	za dijalog u kojem možete unijeti novu lozinku. Također imate mogućnost izbrisati lozinku tako da u popisu odaberete none.
UID	Polje koje stvara sustav. Više informacija o ovom polju potražite u 10. poglavlju "Upravljanje korisničkim računima".
Primary Group	Grupa kojoj korisnik pripada. Grupe omogućuju svrstavanje korisnika prema ovlaštenjima.
Full Name	Puno ime korisnika.
Home	Polazni (home) direktorij korisnika koji se obično nalazi u direktoriju /home ili /usr/home.
Shell	Izvorna korisnička ljuska. Klikom na strelicu kraj polja možete odabrati neku od ljuski koje nudi Red Hat distribucija.

Slika 5.2	Add user	同居
Pomoću grafičkih alata Red Hat distribucije, unos novog korisnika je stvarno jednostavan.	Username: Password: S08 Primary Group: Full name: Home:	Secondary groups
	Shell: //bin/ash	Add Remove
	Add	Cancel

#### Promjena lozinki

Ubuduće ćete možda trebati promijeniti lozinku ili je dodati računu koji je nije imao (primjerice *root* računu). *root* račun biste svakako trebali zaštiti lozinkom.

Da biste, u bilo kojoj inačici Linuxa ili UNIX-a, promijenili lozinku, trebate pokrenuti naredbu *passwd*, upisati staru i novu lozinku te potvrditi novu lozinku. Ako nemate (ili što je još gore, ako ste zaboravili) staru lozinku, nećete moći upotrijebiti naredbu *passwd* da biste promijenili lozinku. Evo kako izgleda promjena lozinke:

```
[tackett@web~]$ passwd
Changing password for jack
Enter old password: password
Enter new password: new- password
Re type new password: new- password
```

Ako načinite grešku, Linux će vas upozoriti da lozinka neće biti promijenjena. Operativni sustav Linux zahtijeva lozinku od barem šest znakova.

#### **UPOZORENJE**

Nemojte zaboraviti svoje lozinke! Ako zaboravite korisničku lozinku, morat ćete promijeniti račun, a ako zaboravite *root* lozinku, sustav ćete morati podići pomoću diskete koju ste stvorili tijekom instalacije. Nakon toga trebate promijeniti lozinku. Postoji mogućnost da ukinete lozinku tako da u okviru za dijalog RHS Add/Edit User odaberete opciju none. Na taj će način korisnik sam moći odabrati lozinku pomoću naredbe *passwd*. Također možete otvoriti datoteku /etc/passwd i obrisati skrivenu lozinku iz korisničkog zapisa.

Vidi "Dodjela lozinki", 221. stranica

#### Uporaba osnovnih naredbi

Potrebno je poznavati neke osnovne naredbe da biste se znali služiti operativnim sustavom. U idućim odlomcima ćete se upoznati s nekim naredbama koje se koriste u Linuxu. Mnoge od "naredbi" koje su u nastavku predstavljene u osnovi su uslužni programi koje Linux koristi da bi proširio svoj spektar naredbi i možete ih naći u direktorijima /bin, /sbin i /usr/bin.

#### Dobivanje pomoći pomoću naredbe man

Da biste dobili pomoć za razne Linux naredbe, trebate upisati naredbu **man** i na zaslonu će se pojaviti sve informacije o pojedinoj naredbi. Ako niste sigurni koju naredbu trebate upotrijebiti, poslužite se parametrom -k i upišite jednostavnu ključnu riječ koja predstavlja temu koja vas zanima. Program *man* će potom pretražiti datoteke za pomoć i pronaći zadanu riječ. Linux nudi i zamjenu za tu naredbu, pod imenom *apropos*.

Upišete li naredbu **man ls**, Linux će vam ponuditi pomoć za naredbu ls (kao i sve raspoložive parametre). Ako pak upišete **man –k cls**, na zaslonu ćete ugledati popis naredbi koje u datoteci za pomoć imaju i riječ *cls*. Naredba *apropos cls* ima isti učinak kao i naredba *man –k cls*.

#### Korištenje naredbi za rad s direktorijima

Linux nudi mnoge naredbe za rad s direktorijima. Kao i drugi operativni sustavi koje ste do sada koristili, Linux vam omogućuje stvaranje, brisanje i premještanje direktorija, kao i prikaz informacija o pojedinom direktoriju.

**Mijenjanje tekućeg radnog direktorija pomoću naredbe cd** Linux, kao i DOS i drugi operativni sustavi, sprema datoteke u direktorije koji čine stablo direktorija. Datoteku možete odabrati ako upišete stazu (*path*) od *root* direktorija (koji se označava s kosom crtom /) do same datoteke. Prema tom načelu, datoteka za podešenje programa *emacs*, koju koristi koris-

nik jack, može se odabrati na sljedeći način:

/home/jack/.emacs

Ako vam je poznato DOS-ovo ograničenje na osam znakova za ime datoteke i tri znaka za dodatak imenu, onda će vas ugodno iznenaditi spoznaja da kod Linuxa nema takvih ograničenja.

Vidi "Imena datoteka i staza", 304. stranica

Linux također koristi koncept polaznog (*home*) direktorija koji je određen prilikom otvaranja pojedinog korisničkog računa. Korisnički polazni direktorij je obično označen znakom tilda (~). Taj znak možete upotrijebiti umjesto imena direktorija ako želite kopirati datoteku iz trenutnog direktorija /usr/home/jack u polazni direktorij:

cp .emacs ~

Da biste u Linuxu prelazili iz jednog u drugi direktorij, morat ćete upotrijebiti naredbu za promjenu direktorija (*cd*). Upišete li samo **cd** bez ikakvih parametara, Linux će vas vratiti u polazni direktorij. Za prelazak iz jednog u drugi direktorij naredbu *cd* morate koristiti više puta (baš kao u DOS-u). Osim toga, Linux koristi jednu točku (.) koja predstavlja trenutni direktorij i dvije točke (..) koje predstavljaju naddirektorij. U osnovi, DOS je stvari preuzeo od UNIX-a, a ne obratno.

**BILJEŠKA** Pažljivo odaberite odjelnik među direktorijima. DOS kao odjelnik među direktorijima koristi oznaku (–), dok je Linux upotrebljava kao znak za nastavak naredbe u drugom retku. Za odjeljivanje direktorija se u Linuxu koristi znak (/).

Osim toga, u DOS-u je nevažno ako pogriješite kod uporabe razmaka prilikom unosa parametara i .., dok je to u Linuxu itekako važno. Linux ne razumije naredbu cd.., već samo naredbu cd .., dakle s obaveznim razmakom. U Linuxu se razmakom odvajaju naredbe i parametri.

**Ispis informacija o datotekama i direktorijima pomoću naredbe** *Is* Naredba *ls* (kratica od list – popis) se koristi za ispis datoteka u direktoriju i vrlo je slična naredbi *dir* u DOS-u (Linux prihvaća i naredbu *dir*). Pod operativnim sustavom Linux naredba *ls* u boji prikazuje sve glavne datoteke u direktoriju. Prema osnovnim postavkama plava boja upućuje da se radi o direktoriju, a zelena da se radi o izvršnim programima. Podešene boje možete promijeniti u datoteci /etc/DIR\_COLORS.

Vidi "Ispis sadržaja direktorija", 318. stranica.

Naredba *ls* koristi mnoge parametre koji određuju ne samo način prikaza datoteka, već i koje će datoteke biti prikazane na zaslonu. Najčešće se koristi parametar -la, koji određuje prikaz informacija u dugačkom formatu.

Naredba ls - la dat će ispis svih informacija o svim datotekama u direktoriju u kojem se trenutno nalazite. Pomoću naredbe ls .emacs na zaslonu će se pojaviti datoteka .emacs, dok ćete uporabom naredbe ls - l .emacs dobiti sve informacije o datoteci .emacs.

Opcija *—ltar* (koristi se kao *ls —ltar*) daje popis istih informacija kao i gornja naredba *ls*, osim što su datoteke poredane od najstarijih prema najnovijima.

**Stvaranje novog direktorija pomoću naredbe mkdir** S obzirom da se u Linuxu sustav datoteka temelji na direktorijima, naredba *mkdir* se koristi za stvaranje novih direktorija. Za razliku od DOS-a, kod kojeg se naredba *mkdir* može pozvati i ako se upiše *md*, Linux zahtijeva unos punog imena naredbe. Morate zadati ime svakog novog direktorija, kao što je prikazano u idućem primjeru:

mkdir backup

**BILJEŠKA** Linux omogućava, preko ljuske s naredbama, dodjelu dodatnih imena naredbama. Stoga, ako "ne možete živjeti" bez DOS naredbe *md* i smeta vam stalno upisivanje naredbe *mkdir*, možete naredbi dati ime *md*.

#### Vidi "Zamjenske naredbe", 364. stranica

**Brisanje direktorija pomoću naredbe** *rmdir* Naredba *rmdir* briše direktorije u Linuxu. Uz naredbu je potrebno upisati i ime direktorija koji se želi obrisati, a sam direktorij treba biti prazan, jer ga Linux inače ne može ukloniti.

Primjerice, ako direktorij /backup sadrži dva poddirektorija, neće se moći izvršiti naredba *rmdir* /backup. Naredbom *rmdir* /backup/jack/\* uklonit će se sve datoteke u direktoriju /backup/jack, a zatim će se pomoću naredbe *rmdir* /backup/jack ukloniti i sam direktorij (koji je sada prazan).

#### **UPOZORENJE**

Pomoću naredbe *rmdir* ne možete obrisati direktorij koji sadrži datoteke, već morate upotrijebiti naredbu *rm* s parametrom –*r*. Primjerice, naredbom

rm -r\*

ćete obrisati sav sadržaj trenutnog direktorija uključujući i njegove poddirektorije. Prilikom uporabe ove naredbe trebate biti vrlo pažljivi jer ćete u trenutku obrisati direktorij čiji sadržaj kasnije nećete moći vratiti. Stoga načinite zaštitnu kopiju.

#### Korištenje naredbi za rad s datotekama

Kako Linux na sličan način tretira direktorije i datoteke, to su i naredbe vrlo slične.

**Kopiranje datoteka pomoću naredbe cp** Naredba *cp* je slična DOS naredbi *copy* i možete je koristiti za kopiranje jedne ili više datoteka iz jednog u drugi direktorij na sljedeći način:

cp iz-datoteke u-datoteku

Umjesto izraza *iz-datoteke* i *u-datoteku*, trebate unijeti imena stare i nove datoteke. Ako ne želite unijeti ime datoteke umjesto parametra *u-datoteku*, tada upišite znak (.). U tome se Linux razlikuje od DOS-a jer u DOS-u ne morate unositi parametar *u-datoteku*.

Naredbom *cp fred1 fred1.old* kopira se datoteka pod imenom fred1 u datoteku s imenom fred1.old (zaštitna kopija datoteke), dok pomoću naredbe

*cp* ~*fred1.old /backup/jack* kopirate datoteku fred1.old iz polaznog (*home*) direktorija u direktorij /backup/jack (znak tilda ~ predstavlja korisnikov polazni direktorij).

**Premještanje datoteka pomoću naredbe mv** Naredba mv, koja je slična DOS naredbi move, omogućuje premještanje datoteka iz jednog u drugi direktorij. Premještanje datoteka ima isti efekt kao da ste kopirali datoteke u novi direktorij i zatim obrisali datoteke u starom direktoriju. Naredba mv ne stvara kopije datoteka.

Sintaksa naredbe mv je identična onoj naredbi cp:

mv iz-datoteke u-datoteku

Naredbom *mv fred1 fred1.old* kopira se datoteka pod imenom fred1 u datoteku s imenom fred1.old (zaštitna kopija datoteke), a datoteka fred1 se briše, dok pomoću naredbe

*mv ~fred1.old /backup/jack* premještate datoteku fred1.old iz polaznog (*home*) direktorija u direktorij /backup/jack.

**Brisanje datoteka pomoću naredbe** *rm* Da biste u Linuxu obrisali datoteke, morate upotrijebiti naredbu *rm*. Ova naredba je opasna jer čim obrišete datoteku ona se više ne može vratiti. Iz sigurnosnih razloga trebali biste koristiti ovaj oblik naredbe *rm*:

rm -i imedatoteke

Parametar -i traži potvrdu za brisanje datoteke. Primjerice, pomoću naredbe *rm fred1* ćete obrisati datoteku fred1, dok ćete uporabom naredbe *rm* -i *fred1* obrisati istu tu datoteku, ali tek nakon potvrde.

#### **UPOZORENJE**

Čim u Linuxu obrišete datoteku, ona će trajno nestati. Ne možete je vratiti, kao što možete u DOSu. Nakon brisanja datoteke jedina vam je nada njezina zaštitna kopija.

**Ispis sadržaja datoteka pomoću naredbe more** Naredba *more* ispisuje sadržaj datoteke s tekstom. Imate mogućnost pregledati sadržaj datoteke, ali je ne možete uređivati, ispisati ili zaustaviti njezin ispis na zaslonu. Da biste vidjeli sadržaj datoteke emacs, trebate upisati ovu naredbu:

more .emacs

**BILJEŠKA** Ako pomoću naredbe *mor*e pokušate pregledati binarnu datoteku, dobit ćete neželjene efekte, primjerice, terminal može biti blokiran. Ako se to dogodi, pokušajte pritisnuti kombinaciju tipki <Ctrl-q> ili <Ctrl-s>.

Nedostatak ove naredbe jest nemogućnost vraćanja i pregleda prethodnih redaka. No, naredbe koje će biti objašnjene u idućem odlomku imaju riješen i taj problem.

**Uporaba naredbe less** (*koja je bolja od naredbe more*) Naredba *less* prikazuje sadržaj datoteke na zaslonu. Naziv naredbe je u osnovi igra riječi s naredbom *more* (više).

Kao i naredba *more* (više), tako i naredba *less* (manje) prikazuje na zaslonu sadržaj datoteke, ali s tom razlikom što imate mogućnost vraćanja na već pregledane retke. Pomoću ove naredbe možete pregledati datoteku tipa *readme* koja se nalazi u direktoriju info:

less /info/readme

**Brisanje sadržaja zaslona pomoću naredbe** *clear* Ponekad će vam dobro doći da sa svog zaslona obrišete nepotrebne informacije i dobijete čisti, prazan zaslon. U DOS-u to možete učiniti pomoću naredbe *cls*, dok u Linuxu trebate koristiti naredbu *clear*.

# Rad s DOS datotekama pod Linuxom

Tijekom instalacije mogli ste bilo koju DOS particiju učiniti vidljivom za Linux i smjestiti je u direktorij koji ste odabrali tijekom podešavanja.

Vidi "Dodavanje potrebnih particija", 70. stranica.

Ako te datoteke želite kopirati na disketu, tada uporaba naredbe *cd* može izazvati probleme jer UNIX i Linux tretiraju tekstualne datoteke nešto drukčije od DOS-a, osobito zbog oznaka za prelazak u novi redak. Da biste izbjegli taj problem, razvijena je grupa programa koja vam pomaže u radu s MS-DOS datotekama pod Linuxom. Radi se o *m*- naredbama, među kojima se nalaze *mcopy* i *mdir* koje rade na isti način kao i DOS naredbe *copy* i *dir*. Kao što možete uočiti, ove se naredbe razlikuju od onih koje rade pod DOS-om samo po prefiksu *m*-, pa otuda ime "m-" naredbe. One su dio paketa *mtools* koji predstavlja zbirku javnih programa koji UNIX-u olakšavaju rad s DOS datotekama.

Pomoću tih naredbi možete vrlo lagano načiniti kopiju datoteke na disketi jer imate mogućnost korištenja DOS oznaka, primjerice disketnog pogona A, umjesto Linux oznake /dev/fd0. Da biste saznali još nešto o *m*- naredbama, upišite sljedeće:

man mtools

Tablica 5.4 n	n- naredbe
Naredba	Opis
mattrib	prikaz atributa određene datoteke (ili datoteka)
mcd	promjena direktorija
mcopy	kopiranje datoteka u novi direktorij
mdel	brisanje odabranih datoteka
mdir	ispis sadržaja direktorija
mformat	formatiranje diskete
mlabel	označavanje DOS sustava datoteka
mmd	stvaranje direktorija
mrd	uklanjanje direktorija (direktorij mora biti prazan kao u DOS-u)
mren	promjena imena postojeće DOS datoteke
mtype	ispis teksta u DOS datoteci

Tablica 5.4 sadrži kratak popis različitih m- naredbi.

**BILJEŠKA** lako pomoću Linuxa možete vidjeti DOS datoteku, pa čak i uređivati neke tekstualne datoteke koje se nalaze u DOS particijama, ipak nećete moći pokrenuti DOS ili Windows programe pod Linuxom. Međutim, na Internetu možete pronaći potrebne emulatore za DOS i Windows programe. Iako će to ubuduće sigurno moći, zasad emulatori ne mogu koristiti sve funkcije. U nastavku ćete ovog poglavlja naći uvodne informacije o oba emulatora.

# Izlazak iz Linuxa

Kada završite rad na računalu s operativnim sustavom DOS, jednostavno isključite napajanje računala i posao je gotov. Isto biste mogli učiniti i nakon rada s Windowsima, iako u tom slučaju postoji mogućnost da se datoteka ošteti. Kod Linuxa postoji opasnost da se ošteti čitav sustav, čak i hardver i sustavi datoteka. Stoga se iz Linuxa izlazi na točno određen način, inače vam se može dogoditi da operativni sustav nećete moći opet pokrenuti.

Linux dosta informacija o samom sustavu i datoteke drži u memorijskim spremnicima (*bufferi*), a tek ih naknadno sprema na tvrdi disk. Na taj način sustav postiže bolje karakteristike i mogućnost upravljanja hardverskim komponentama, što je bitno za višezadaćne operativne sustave (na taj način jedan korisnik ne može istovremeno koristiti hardver koji već koristi netko drugi). Ako isključite napajanje, podaci se mogu izgubiti i tako možete oštetiti sustav datoteka.

#### Vidi "Isključivanje sustava", 215. stranica

Kako je Linux višekorisnički i višezadaćni operativni sustav, prije isključivanja sustava potrebno je da svi korisnici završe i spreme sve što rade, kako se podaci ne bi izgubili. Svaki korisnik koji se prijavio za rad imat će vremena i da se odjavi. Da biste pravilno napustili Linux, trebate pokrenuti naredbu shutdown:

shutdown [-r] time-to-shutdown [message]

Parametar -r upućuje na to da bi se Linux trebao odmah ponovno pokrenuti nakon njegova gašenja, što može biti korisno ako želite napustiti Linux i pokrenuti neki drugi operativni sustav.

Opcija *time-to-shutdown* određuje u kojem trenutku se računalo gasi. Vrijeme gašenja se upisuje u obliku 0-24, tako da, primjerice, za izlazak iz Linuxa u 11 sati navečer trebate upisati sljedeću naredbu:

shutdown 23:00

Parametar *message* predstavlja poruku koja se šalje svakom korisniku koji je prijavljen za rad. Tu poruku možete iskoristiti da biste korisnicima objasnili zbog čega gasite računalo. Primjerice, ako se spremate za tjednu sigurnosnu pohranu podataka, možete upisati sljedeću poruku kako bi se svi korisnici na vrijeme odjavili:

[root@web /root]# shutdown -r 23:00 Gasenje sustava u 11:00 zbog radova

Upamtite da ne smijete jednostavno isključiti računalo ili pritisnuti gumb Reset ako radite u Linuxu.

#### **UPOZORENJE**

Na nekim računalima Linux može prepoznati kombinaciju tipki <Ctrl-Alt-Del> kao naredbu za izlazak iz Linuxa (kao da je korisnik upisao naredbu shutdown). No, kod nekih računala to nije slučaj.

Ako greškom isključite računalo i oštetite strukturu datoteka, možete upotrijebiti naredbu *fsck* i pokušati obnoviti sustav datoteka.

Vidi "Uporaba naredbe fsck", 279. stranica.

#### Pokretanje Linux programa

Sada kada ste savladali osnove Linuxa i način pokretanja osnovnih naredbi, možete iskušati nekoliko programa koje ste instalirali tijekom podešavanja sustava. Ti programi pokrivaju veliki broj uslužnih radnji, počev od kalkulatora do kompletnih C i C++ kompilatora (prevoditelja). Neki od navedenih programa stoje stotine dolara, no mnogi od njih su gotovo besplatni, jer je jedini trošak njihovo kopiranje s Interneta.

Srećom, mnogi Linux programi se nalaze i na lokalnim BBS-ovima kojima možete pristupiti pomoću telekomunikacijskih programa što predstavljaju sastavni dio Slackware i Red Hat distribucije. Također, mnogi izdavači CD-ova nude na njima stotine UNIX programa (u izvornom kodu). Te programe trebate kopirati s CD-a i prevesti ih pomoću gcc i g++ kompilatora (koji se isporučuju s Linuxom), a zatim i pokrenuti na svom osobnom računalu (čak i ako nikada prije niste prevodili programe).

Napokon, ti su programi većinom tekstualnog tipa tako da ne zahtijevaju sustav XWindows. Istina, oni ne koriste privlačno grafičko sučelje, ali rade s većinom Linux distribucija.

#### Uporaba programa workbone

Program *workbone* je instaliran sa Slackware distribucijom, a radi se o *CD playeru* koji je napisao Thomas McWilliams. Ako vaš CD-ROM uređaj ima mogućnost izvođenja audio-diskova, moći ćete isprobati ovaj program.

McWilliams je napisao program koji njemu odgovara, služeći se XWindowsima, tako da se može dogoditi da ovaj program ne radi ispravno sa svim CD-ROM pogonima.

Koristite li ovaj program, diskove ćete trebati kontrolirati brojčanim tipkama na tipkovnici, stoga provjerite jeste li uključili <NumLock>. U tablici 5.5 naći ćete popis i namjenu tipki koje se koriste u programu *workbone*.

Tablica 5.5	Tipke koje se koriste u programu workbone
Tipka	Opis
0	Izlazak iz programa workbone i nastavak sviranja glazbe
DEL	Zaslon za pomoć

Tipka	Opis
1	Vraćanje za 15 sekundi
2	Izlazak iz programa workbone i isključivanje glazbe
3	Prelazak za 15 sekundi unaprijed
4	Vraćanje na prethodni odabir
5	Ponovno sviranje zadnjeg odabira
6	Prelazak na idući odabir
7	Stop
8	Pauza/nastavak
9	Sviranje

Dok *workbone* svira, na zaslonu se ažurira vrijeme i trenutni odabir. Ako želite nastaviti raditi dok glazba svira, onda su vam na raspolaganju dvije opcije:

- Možete izaći iz programa i ostaviti da glazba svira (tipka 0)
- Ako želite zadržati ispis na zaslonu, tada jednostavno otvorite drugi virtualni terminal pomoću tipke <Alt> i prijavite se na drugi račun. Kada budete htjeli provjeriti zaslon CD *playera*, samo otvorite određeni virtualni terminal i pročitajte stanje.

Isto tako, možete zaustaviti program pritiskom na tipku  $\theta$  i kasnije ga ponovno pokrenuti kako biste vidjeli koja se pjesma izvodi. Više ćete informacija dobiti ako pokrenete naredbu man workbone.

#### Uporaba proračunske tablice sc

Prodaju li računala softver ili softver prodaje računala? Radi se o prastarom pitanju čiji odgovor glasi da pravi program može prodati tisuće računala. Kad se na tržištu pojavio program VisiCalc, odmah je porasla uporaba osobnih računala u poslovne svrhe. Zašto? Jer su se godinama poslovni ljudi mučili s praćenjem poslovanja na komadićima papira ili blokovima. Program VisiCalc je bila elektronička inačica tablica nacrtanih na papiru i predstavljao je revoluciju u planiranju poslovanja. Danas nasljednici programa VisiCalc, kao što su Microsoft Excel i Lotus 1-2-3, još uvijek održavaju slavu koju je stekao VisiCalc. U Linux sustavu program *sc* predstavlja proračunsku tablicu.

Program *sc* sadrži ćelije poredane u retke i stupce. Svaka ćelija može sadržavati brojčanu vrijednost, oznaku ili izraz, tj. formulu koja računa brojčane vrijednosti ili ispisuje oznake. Pomoću oznaka mogu se povezati i druge ćelije i na taj se način može stvoriti složena povezanost u tablici.

Ako ste do sada već radili s drugim sličnim programima, neće vam biti problem koristiti i program *sc.* Zatreba li vam pomoć, možete pokrenuti program koji će vas naučiti kako se koriste pojedine naredbe. Da biste pokrenuli taj program, upišite sljedeće:

```
sc /usr/lib/sc/tutorial.sc
```

Ove upute sadrže izvrstan uvod za uporabu programa *sc*. Ako vam trebaju brza objašnjenja pojedinih naredbi, možete ih ispisati pomoću sljedeće naredbe:

scqref | 1pr

Kod uporabe ove naredbe, u osnovi prosljeđujete rezultate jedne naredbe (*scqref*) drugoj (*lpr*).

BILJEŠKA Pročitajte 20. poglavlje, "Ispis", i otklonit ćete probleme koji se javljaju prilikom ispisa pod Linuxom. Najveći problem s kojim se možete susresti, osim da uopće ne možete ništa ispisati, jest pojava tzv. stupnjevitog ispisa koji je nastao zbog razlike u načinu na koji UNIX/Linux sustavi "razumiju" znakove za prelazak u novi redak i znakove za vraćanje na početak retka, a u odnosu na to kako iste znakove "razumiju" MS-DOS editori.

Želite li saznati nešto više o programu sc, jednostavno upišite man sc.

#### Uporaba kalkulatora bc

Program be je kalkulator za brza računanja. Radi se o sofisticiranom programskom jeziku koji vam omogućuje i zadavanje aritmetičkih izraza.

Na početku izvođenja kalkulator *bc* se "javi" kratkom porukom o pravima na kopiranje, a zatim redak za unos naredbe ostane prazan. Potom možete unijeti jednostavne funkcije za zbrajanje i oduzimanje, a što se množenja i dijeljenja tiče, znajte da u inačici programa *bc* koja se distribuira s Linuxom često može doći do pogrešaka (to je jedan od rizika koji nosi takva vrsta softvera). Kalkulator *bc* je izvrstan za jednostavne proračune, sve dok ne naiđu problemi s množenjem i dijeljenjem.

Druga sjajna odlika programa je mogućnost spremanja vrijednosti (rezultata) iz jednog proračuna i njihovo korištenje u drugom računu. Sintaksa je vrlo jednostavna *varijabla – ime = izraz*. U narednom primjeru će se računati umnožak 125\*5, a rezultat će se spremiti kao var1. Želite li vidjeti koja je izračunata vrijednost, dovoljno je upisati ime varijable i program će njezinu vrijednost upisati u sljedećem retku. U nastavku se, kao varijabla

var2, sprema vrijednost koja se dobije kada se var1 podijeli s 5.

```
var1 = 125 * 5
var1
625
var2 = var1 / 5
var2
```

#### Uporaba telekomunikacijskog paketa minicom

Nadam se da ćete nakon što pročitate poglavlja u 6. dijelu knjige, "Uporaba Interneta", svoj Linux sustav moći pokrenuti i raditi na Internetu (globalnoj informacijskoj autocesti). Dok to ne savladate, trebali biste ipak naučiti kako se povezati s ostatkom svijeta (naravno ako imate modem i telekomunikacijski paket). Linux sadrži paket koji se naziva *minicom*. Taj paket možete upotrijebiti za softversko povezivanje modema i jednog od vaših serijskih priključaka.
Paket *minicom*, kao i većina softvera za Linux, napisala je jedna osoba uz pomoć mnogih korisnika s Interneta. Glavni autor paketa *minicom* je Miquel van Smoorenburg. Radi se o velikoj aplikaciji koja se može uspoređivati s drugim komercijalnim aplikacijama. Pomoću paketa *minicom* možete se povezati s raznim BBS-ovima, načiniti popis telefonskih brojeva, kopirati datoteke s mreže i na mrežu. Na stranici *man* možete naći pomoć za većinu funkcija koje se koriste u *minicom* paketu.

Prvo što trebate upamtiti jest da *minicom* koristi kombinaciju tipki <Ctrl-Shift-a> za pristup različitim funkcijama, kao što su automatsko biranje telefonskih brojeva i kopiranje datoteka. Za dobivanje pomoći pritisnite kombinaciju tipki <Ctrl-a><z> i pojavit će se zaslon s kratkim opisom naredbi. Tablica 5.6 sadrži popis nekih naredbi.

Tablica 5.6 S	6 Sazetak naredbi koje se koriste u paketu <i>minicom</i>	
Tipka	Opis	
D	Pristup direktoriju za biranje telefonskih brojeva	
S	Slanje datoteka	
Р	Popis komunikacijskih parametara	
L	Uključivanje/isključivanje pohrane u datoteku	
F	Šalje naredbu BREAK na drugi terminal	
Т	Podešavanje emulatora - vt100, Minix ili ANSI	
W	Uključivanje/isključivanje prelaska u novi redak.	
G	Pokretanje minicom skripte	
R	Prijam datoteke	
А	Dodavanje znaka za prelazak u novi redak	
Н	Prekidanje telefonske linije	
М	Inicijalizacija modema	
K	Pokretanje protokola kermit	
E	Uključivanje/isključivanje lokalnog odziva (echo)	
С	Brisanje lokalnog zaslona	
0	Podešavanje minicoma	
J	Prelazak u novu ljusku	
Х	Isključivanje i ponovno podešavanje modema	
Ι	Način rada s pokazivačem	
Z	Prikaz zaslona za pomoć	
В	Vraćanje (klizanje unatrag) u prozoru	

#### 124 5. poglavlje Pokretanje Linux programa

Dok se nalazite u prozoru za pomoć, možete jednostavno pritisnuti pojedino slovo i pokrenuti pripadnu naredbu, dok u programu *minicom* trebate prije slova pritisnuti kombinaciju tipki <Ctrl-a>.

Paket *minicom* sadrži četiri protokola za prijenos datoteka: *zmodem, ymodem, xmodem* i *kermit*. Ako je moguće, trebali biste koristiti protokol *zmodem* jer ima najveće mogućnosti za otkrivanje grešaka prilikom prijenosa podataka. Ako se taj protokol ne koristi u sustavu s kojim trenutno komunicirate, pokušajte s nekim drugim protokolom (prema navedenom redoslijedu). Nije rečeno da je *kermit* loš protokol (jer to i nije) već je samo sporiji od većine drugih. Dobra strana protokola *kermit* je ta što ga koristi i većina drugih sustava.

Drugo što bi vas moglo zabrinuti jest da paket *minicom* koristi prednosti nekih naredbi koje mu omogućavaju pristup onim dijelovima sustava kojima inače pristupa samo super korisnik. Stoga, svatko tko pokrene *minicom* će imati pristup određenim dijelovima Linuxa, što vam vjerojatno ne odgovara.

Vidi "Zaštita datoteka", 240. stranica

# lgre

Ako ste instalirali paket y, moći ćete koristiti mnoštvo igara. Većina tih igara su tekstualne igrice, tako da ne trebate pokretati XWindowse da biste ih mogli igrati. Da biste vidjeli koje sve igre možete igrati, potražite direktorij /usr/games. Pregledom datoteka saznat ćete koje su vam igre na raspolaganju. Ako niste sigurni što koja igra predstavlja, potražite pomoć putem naredbe *man*. Još bolje, jednostavno pokrenite igricu i sami je istražite. Ugodan dan!

### **Tetris**

Tetris je nastao u bivšem Sovjetskom Savezu. Tijekom igre likovi raznih oblika padaju s neba, a igrač ih slaže u dnu zaslona. Cilj igre je da slažete oblike tako da zauzimaju što manje mjesta i da se prozor ne popuni. Kada jedan redak potpuno popunite, on će nestati sa zaslona zajedno sa svim recima ispod njega. On što je u taktici nezgodno su oblici likova koji padaju, tako da biste popunili redak morate rotirati oblike i smještati ih lijevo ili desno prije nego dotaknu drugi lik. Čim jedan dotakne drugi lik, on će ostati na tom mjestu i nećete ga moći pomaknuti.

Ova igra je prilagođena za mnoge platforme, pa ako ste Tetris već igrali na računalu, neće vam biti problem igrati igru pod Linuxom.

Ova inačica igre je priređena za igranje putem tipkovnice, a nemojte očekivati ni raskošnu grafiku. Najveći je nedostatak igre taj što u drugim operativnim sustavima možete koristiti tipke sa strelicama, što kod Linuxa nije slučaj, već možete koristiti samo one tipke koje su navedene u tablici 5.7.

Tablica 5.7 Naredbe u Te	etrisu
Naredba	Tipka
Pomicanje u lijevo	<,>
Pomicanje u desno	
Rotiranje	<,>
Zakucavanje	razmaknica
Pauza	<\$>
Kraj	<q></q>
Osvježenje zaslona	<ctrl-l></ctrl-l>

## Dungeon

Dungeon je tekstualna igrica koja se bazira na starim igrama Adventure, ali umjesto spilja bavite se podzemnim prostorijama tražeći blago i pustolovine. Ako ste dosada igrali druge slične igrice, onda znate o čemu se radi. Ako ste do sada igrali samo grafičke igrice, naslonite se i počnite koristiti mozak. U igri sudjelujete tako da pokrećete naredbe i

izdajete zahtjeve u obliku glagola i imenica. Primjerice, na početku ćete igre saznati da se nalazite u dvorištu ispred velike bijele kuće s ulaznim vratima. Kraj vas se nalazi mali poštanski sandučić. Ako želite pročitati pismo u sandučiću, tada možete pokrenuti sljedeću naredbu:

```
There is a small mailbox here.
> open box
Opening the mailbox reveals:
a leaflet.
> read leaflet
```

Igra sadrži i kratak pregled pravila igre i podataka o talentiranim programerima koji su je načinili. Zadnji redak u pismu vas upućuje da pomoć možete dobiti ako pokrenete naredbu *help* ili *info*.

# Trek

Trek je tekstualna igrica prema popularnoj TV seriji *Zvjezdane staze*. Cilj je preživjeti krvave bitke s Klingoncima i sačuvati svoj dio zvjezdanog prostora od njihove najezde. Igru pokrećete tako da upišete **trek** i zatim odgovorite na nekoliko pitanja radi podešavanja igre:

- Trebate unijeti trajanje igre koju želite igrati.
- Možete ponovno pokrenuti igre koje ste prethodno igrali, tako da upišete ime odgovarajuće log datoteke u redak za unos naredbe. Ime datoteke postaje ime spremljene igre.
- Trebate unijeti razinu igre.
- Možete postaviti lozinku tako da drugi ne mogu igrati vašu igru. Lozinka vam je potrebna kako biste bili sigurni da nitko osim vas neće sjesti u vaš svemirski brod.

U bilo kojem trenutku, tijekom podešavanja ili za vrijeme igranja, možete upisati znak ? i dobiti potrebnu pomoć. U tablici 5.8 dat je popis radnji koje su vam na raspolaganju.

Tablica 5.8.	aredbe u igri Trek	
Naredba	Opis	
abandon	Kraj igre	
damages	Popis oštećenja koje je pretrpio vaš svemirski brod	
impulse	Pokretanje impulsne snage	
ram	Povećanje brzine	
srscan	Pokretanje skanera kratkog dometa	
undock	Napuštanje baze	
captures	Hvatanje Klingonaca	
destruct	Samouništenje	
lrscan	Pokretanje skanera dugog dometa	
dump	Tko zna?	
visual	Pregled rasporeda Klingonaca	
cloak	Prekrivanje broda	
dock	Ulazak u bazu	
move	Crtanje putanje leta	
rest	(ostaje za ubuduće)	
terminate	Izlazak	
Warp	Pokretanje motora	
computer	Traženje dodatnih informacija	
help	Traženje pomoći od baze	
phasers	Paljenje lasera	
shields	Navlačenje oklopa	
torpedo	Ispaljivanje torpeda	

Na početku igre ćete saznati koliko se Klingonaca nalazi u vašem sektoru, te broj i smještaj baza. Ulaskom u bazu možete popraviti svoj brod. Nažalost, nećete unaprijed znati gdje se nalaze ratni brodovi Klingonaca. Stalno provjeravajte potrošnju goriva jer ćete u protivnom doživjeti neugodna iznenađenja.

Iako ova tekstualna igrica na koristi grafiku, korištenjem naredbe *srscan* pokrenut ćete skaner kratkog dometa i ugledati svoj sektor i sve poznate objekte. Pomoću ove naredbe

također možete saznati neke informacije o stanju vašeg broda. Koordinate svih objekata zapisane su u Cartesianovoj matrici i možete ih prepisati na papir kako ne biste trebali često pokrenuti naredbu *srscan*.

# Pokretanje DOS programa pod Linuxom

Nakon što vam dosadi pokretanje raznih Linux aplikacija, vjerojatno ćete htjeti pokrenuti neki DOS ili Windows program. U tijeku je razvoj programa koji emuliraju (zamjenjuju) različite operativne sustave pod Linuxom. DOSEMU je program koji omogućuje pokretanje pod Linuxom programa koji izvorno rade pod MS-DOS-om (i raznim inačicama kao što je PC-DOS). DOSEMU je kratica od DOS EMUlator.

**BILJEŠKA** Neke Linux distribucije sadrže i naredbu *simply* DOS, koja služi za pokretanje DOS programa za uređivanje teksta. Komercijalne inačice Red Hat Linuxa 5.1 (i novije) sadrže i tu naredbu.

Osim toga, upravo se razvija emulator koji korisnicima omogućuje pristup Windows programima pod Linuxom. Njegovo je ime Wine, a bit će objašnjen u odlomku "Pokretanje Windows programa pod Linuxom".

#### Instalacija emulatora DOS-a

Najnoviju inačicu emulatora DOSEMU možete, naravno, pronaći na Internetu, a postupak raspakiravanja je sličan dolje opisanom. Komprimirane datoteke treba prenijeti u direktorij /usr/src i potom dekomprimirati na sljedeći način:

```
[root@web src]# gzip -d dosemu_5.tgz
[root@web src]# tar -xvf dosemu_5.tar
```

Nakon toga trebate stvoriti razne datoteke koristeći sljedeće naredbe:

[root@web src]# make config [root@web src]# make depend [root@web src]# make most

Pomoću tih naredbi ćete instalirati DOSEMU datoteke u direktorij /var/lib/dosemu. Morate se prijaviti kao super korisnik te imati barem 10 MB slobodne virtualne memorije.

```
BILJEŠKA Potrebno je imati instaliran paket d koji služi za razvijanje programa. U tom se paketu nalaze razni kompilatori i alati koji su vam potrebni za stvaranje emulatora DOS-a.
```

# Podešavanje emulatora DOS-a

Nakon što ste instalirali emulator, trebate podesiti sustav. Za početak načinite DOS *boot* disketu i kopirajte sljedeće DOS datoteke na nju: command.com, fdisk.exe i sys.com.

Zatim kopirajte DOSEMU datoteke iz poddirektorija na disketu: emufs.sys, ems.sys, cdrom.sys i exitemu.com. Možete upotrijebiti *m*- naredbe, koje su spomenute u odlomku "Rad s DOS datotekama pod Linuxom", za kopiranje datoteka s Linux particija na disketu.

Ako ne možete naći Linux datoteke, tada upotrijebite naredbu find. Evo primjera te naredbe: find -name emufs.sys -print Na zaslonu će se pojaviti mjesto na kojem se nalazi ta datoteka, naravno ako uopće postoji.

Da bi ispravno radio, emulator DOSEMU zahtijeva datoteku za podešenje, dosemu.conf. Tu datoteku trebate prilagoditi svom sustavu. Primjer jedne takve datoteke možete naći u direktoriju s primjerima, pod imenom config.dist. Ispis 5.1 prikazuje datoteku config.dist. Napomene su označene znakom #, a većina opcija je napisana *kurzivom*. Ako parametar može poprimiti više raznih vrijednosti, one su napisane u vitičastim zagradama {}.

#### Ispis 5.1 Primjer datoteke dosemu.conf

```
# Linux dosemu 0.51 configuration file.
# Updated to include QuickStart documentation 5/10/94 by Mark Rejhon
# James MacLean, jmaclean@fox.nstn.ns.ca, 12/31/93
# Robert Sanders, gt8134b@prism.gatech.edu, 5/16/93
# NOTICE:
#

    Although QuickStart information is included in this file, you

#
    should refer to the documentation in the "doc" subdirectory of the
#
    DOSEMU distribution, wherever possible.
  - This configuration file is designed to be used as a base to make
#
#
    it easier for you to set up DOSEMU for your specific system.
#
  - Configuration options between lace brackets { } can be split onto
#
    multiple lines.
#
  - Comments start with # or ; in column 1. (beginning of a line)

    Send Email to the jmaclean address above if you find any errors.

#
# QuickStart:
# This section is of interest mainly to programmers. This is useful if
# you are having problems with DOSEMU and you want to enclose debug info
#
  when you make bug reports to a member of the DOSEMU development team.
# Simply set desired flags to "on" or "off", then redirect stderr of
  DOSEMU to a file using "dos 2>debug" to record the debug information
# if desired. Skip this section if you're only starting to set up.
debug { config off disk off
                             warning off
                                          hardware off
port off read off
                                  IPC
                                       off
                     general off
video off
          write off
                       xms
                           off
                                  ems
                                       off
serial off keyb off
                      dpmi off
printer off mouse off
}
      #
```

```
# Want startup DOSEMU banner messages? Of course :-)
dosbanner on
# timint is necessary for many programs to work.
timint on
# QuickStart:
# With the "layout" keyword, you can specify your country's keyboard
#
  layout. The following layouts are implemented:
#
    finnish us dvorak
                                 sf
#
    finnish_latin1 uk
                                   sf latin1
                           sg
#
    gr
             dk
                    sg latin1 es
#
    gr_latin1
                 dk latin1 fr
                                   es_latin1
                     fr latin1
#
    be
            no
#
  The us-layout is selected by default if the "layout" keyword is omitted.
#
  The keyword "keybint" allows more accurate keyboard interrupts,
# It is a bit unstable, but makes keyboard work better when set to "on".
#
# The keyword "rawkeyboard" allows for accurate keyboard emulation for
# DOS programs, and is only activated when DOSEMU starts up at the
# console. It only becomes a problem when DOSEMU prematurely exits
#
  with a "Segmentation Fault" fatal error, because the keyboard would
# have not been reset properly. In that case, you would have to reboot
# your Linux system remotely, or using the RESET button. In reality,
# this should never happen. But if it does, please do report to the
# dosemu development team, of the problem and detailed circumstances,
# we're trying our best! If you don't need near complete keyboard
# emulation (needed by major software package), set it to "off".
keyboard { layout us keybint on rawkeyboard on }
# keyboard { layout gr-latin1"keybint:on"rawkeyboard on }
# If DOSEMU speed is unimportant, and CPU time is very valuable to you,
# you may want to set HogThreshold to a non-zero value. This means
# the number of keypress requests in a row before CPU time is given
  away from DOSEMU. A good value to use could be 10000.
#
# A zero disables CPU hogging detection via keyboard requests.
HogThreshold 0
        #
# QuickStart:
# You can specify up to 4 simultaneous serial ports here.
# If more than one ports have the same IRQ, only one of those ports
# can be used at the same time. Also, you can specify the com port,
# base address, irq, and device path! The defaults are:
#
    COM1 default is base 0x03F8, irq 4, and device /dev/cua0
    COM2 default is base 0x02F8, irq 3, and device /dev/cua1
    COM3 default is base 0x03E8, irq 4, and device /dev/cua2
    COM4 default is base 0x02E8, irq 3, and device /dev/cua3
# If the "com" keyword is omitted, the next unused COM port is assigned.
```

#### Ispis 5.1 nastavak

```
# Also, remember, these are only how you want the ports to be emulated
# in DOSEMU. That means what is COM3 on IRQ 5 in real DOS, can become
# COM1 on IRQ 4 in DOSEMU!
# Also, as an example of defaults, these two lines are functionally equal:
# serial { com 1 mouse }
# serial { com 1 mouse base 0x03F8 irq 4 device /dev/cua0 }
# If you want to use a serial mouse with DOSEMU, the "mouse" keyword
# should be specified in only one of the serial lines. (For PS/2
# mice, it is not necessary, and device path is in mouse line instead.)
# Uncomment/modify any of the following if you want to support a modem
# (or any other serial device).
#serial { com 1 device /dev/modem }
#serial { com 2 device /dev/modem }
#serial { com 3 device /dev/modem }
#serial { com 4 device /dev/modem }
#serial { com 3 base 0x03E8 irq 5 device /dev/cua2 }
# If you have a non-PS/2 mouse, uncomment/modify one of the following.
#serial { mouse com 1 device /dev/mouse }
#serial { mouse com 2 device /dev/mouse }
# What type is your mouse? Uncomment one of the following.
# Use the 'internaldriver' option with ps2 and busmouse options.
#mouse { microsoft }
#mouse { logitech }
#mouse { mmseries }
#mouse { mouseman }
#mouse { hitachi }
#mouse { mousesystems }
#mouse { busmouse }
#mouse { ps2 device /dev/mouse internaldriver }
# The following line won't run for now, but I hope it will sometime
#mouse { mousesystems device /dev/mouse internaldriver cleardtr }
#
# Turn the following option 'on' if you require IPX/SPX emulation.
# Therefore, there is no need to load IPX.COM within the DOS session.
# The following option does not emulate LSL.COM, IPXODI.COM, etc.
# NOTE: MUST HAVE IPX PROTOCOL ENABLED IN KERNEL !!
ipxsupport off
# Enable Novell 8137->raw 802.3 translation hack in new packet driver.
#pktdriver novell hack
# !!WARNING!!: A LOT OF THIS VIDEO CODE IS ALPHA! IF YOU ENABLE GRAPHICS
# ON AN INCOMPATIBLE ADAPTOR, YOU COULD GET A BLANK SCREEN OR MESSY SCREEN
# EVEN AFTER EXITING DOSEMU. JUST REBOOT (BLINDLY) AND THEN MODIFY CONFIG.
```

```
# QuickStart:
# Start with only text video using the following line, to get started.
# then when DOSEMU is running, you can set up a better video configura-
# tion.
# video { vga console }
                           # Use this line, if you are using VGA
# video { cga console } # Use this line, if you are using CGA
# video { ega console } # Use this line, if you are using EGA
# video { mda console } # Use this line, if you are using MDA
# Even more basic, like on an xterm or over serial, use one of the
# following :
# For Xterm
# video { vga chunks 25 }
# For serial at 2400 baud
# video { vga chunks 200 }
# QuickStart Notes for Graphics:
#
  - If your VGA-Bios resides at E000-EFFF, turn off video BIOS shadow
#
   for this address range and add the statement vbios seg 0xe000
#
   to the correct vios-statement, see the example below.
# - Set "allowvideoportaccess on" earlier in this configuration file
#
  if DOSEMU won't boot properly, such as hanging with a blank screen,
#
   beeping, or the video card bootup message.
# - Video BIOS shadowing (in your CMOS setup) at C000-CFFF must be dis-
#
   abled.
#
   *> CAUTION <*: TURN OFF VIDEO BIOS SHADOWING BEFORE ENABLING GRAPHICS!
# It may be necessary to set this to "on" if DOSEMU can't boot up properly
# on your system when it's set "off" and when graphics are enabled.
# Note: May interfere with serial ports when using certain video boards.
allowvideoportaccess on
# Any 100% compatible standard VGA card _MAY_ work with this:
#video { vga console graphics }
# If your VGA-BIOS is at segment E000, this may work for you:
#video { vga console graphics vbios seg 0xe000 }
# Trident SVGA with 1 megabyte on board
#video { vga console graphics chipset trident memsize 1024 }
# Diamond SVGA
#video { vga console graphics chipset diamond }
# ET4000 SVGA card with 1 megabyte on board:
#video { vga console graphics chipset et4000 memsize 1024 }
# S3-based SVGA video card with 1 megabyte on board:
#video { vga console graphics chipset s3 memsize 1024 }
```

```
Ispis 5.1 Nastavak
```

```
#
# QuickStart:
# For "mathco", set this to "on" to enable the coprocessor during DOSEMU.
# This really only has an effect on kernels prior to 1.0.3.
# For "cpu", set this to the CPU you want recognized during DOSEMU.
# For "bootA"/"bootC", set this to the bootup drive you want to use.
# It is strongly recommended you start with "bootA" to get DOSEMU
#
  going, and during configuration of DOSEMU to recognize hard disks.
mathco on # Math coprocessor valid values: on off
cpu 80386 # CPU emulation valid values: 80286 80386 80486
bootA
            # Startup drive valid values: bootA bootC
# QuickStart:
# These are memory parameters, stated in number of kilobytes.
# If you get lots of disk swapping while DOSEMU runs, you should
# reduce these values. Also, DPMI is still somewhat unstable.
# (as of early April 1994) so be careful with DPMI parameters.
xms 1024
          # XMS size in K, or "off"
ems 1024
          # EMS size in K, or "off"
dpmi off
           # DPMI size in K, or "off". Be careful with DPMI!
# !!WARNING!!: GIVING ACCESS TO PORTS IS BOTH A SECURITY CONCERN AND
# SOME PORTS ARE DANGEROUS TO USE. PLEASE SKIP THIS SECTION, AND
# DON'T FIDDLE WITH THIS SECTION UNLESS YOU KNOW WHAT YOU'RE DOING.
# ports { 0x388 0x389 } # for SimEarth
# ports { 0x21e 0x22e 0x23e 0x24e 0x25e 0x26e 0x27e 0x28e 0x29e } # for
# jill
# These keywoards are allowable on the "speaker" line:
# native Enable DOSEMU direct access to the speaker ports.
# emulated Enable simple beeps at the terminal.
# off Disable speaker emulation.
speaker native
              # or "off" or "emulated"
#
# !!WARNING!!: DAMAGE MIGHT RESULT TO YOUR HARD DISK (LINUX AND/OR DOS)
# IF YOU FIDDLE WITH THIS SECTION WITHOUT KNOWING WHAT YOU'RE DOING!
# QuickStart:
# The best way to get started is to start with a boot floppy, and set
# "bootA" above in the configuration. Keep using the boot floppy
# while you are setting this hard disk configuration up for DOSEMU,
# and testing by using DIR C: or something like that.
```

```
# If you want DOSEMU to be able to access a DOS partition, the
# safer type of access is "partition" access, because "wholedisk"
#
  access gives DOSEMU write access to a whole physical disk,
# including any vulnerable Linux partitions on that drive!
 !!! IMPORTANT !!!
#
#
  You must not have LILO installed on the partition for dosemu to boot
# off.
# As of 04/26/94, doublespace and stacker 3.1 will work with wholedisk
# or partition only access. Stacker 4.0 has been reported to work with
# wholedisk access. If you want to use disk compression using partition
# access, you will need to use the "mkpartition" command included with
#
 dosemu to create a partition table datafile for dosemu.
#
 Please read the documentation in the "doc" subdirectory for info
# on how to set up access to real hard disk.
  "image" specifies a hard disk image file.
#
# "partition" specifies partition access, with device and partition
#
   number.
  "wholedisk" specifies full access to entire hard drive.
#
  "readonly" for read only access. A good idea to set up with.
#disk { image "/var/lib/dosemu/hdimage" } # use diskimage file.
#disk { partition "/dev/hda1" 1 readonly } # 1st partition on 1st IDE.
#disk { partition "/dev/sda2" 1 readonly } # 1st partition on 2nd SCSI.
#disk { wholedisk "/dev/hda" }
                                         # Entire disk drive unit
# Use the following option to boot from the specified file, and then
# once booted, have bootoff execute in autoexec.bat. Thanks Ted :-).
# Notice it follows a typical floppy spec. To create this file use
# dd if=/dev/fd0 of=/var/lib/dosemu/bdisk bs=16k
#bootdisk { heads 2 sectors 18 tracks 80 threeinch file /var/lib/dosemu/#bdisk }
# Specify extensions for the CONFIG and AUTOEXEC files. If the below
# are uncommented, the extensions become CONFIG.EMU and AUTOEXEC.EMU.
# NOTE: this feature may affect file naming even after boot time.
# If you use MSDOS 6+, you may want to use a CONFIG.SYS menu instead.
#EmuSys EMU
#EmuBat EMU
# QuickStart:
# This part is fairly easy. Make sure that the first (/dev/fd0) and
# second (/dev/fd1) floppy drives are of the correct size, "threeinch"
# and/or "fiveinch". A floppy disk image can be used instead, however.
# FOR SAFETY, UNMOUNT ALL FLOPPY DRIVES FROM YOUR FILESYSTEM BEFORE
# STARTING UP DOSEMU! DAMAGE TO THE FLOPPY MAY RESULT OTHERWISE!
```

```
Ispis 5.1
           Nastavak
```

```
floppy { device /dev/fd0 threeinch }
floppy { device /dev/fd1 fiveinch }
#floppy { heads 2 sectors 18 tracks 80
#
     threeinch file /var/lib/dosemu/diskimage }
# If floppy disk speed is very important, uncomment the following
# line. However, this makes the floppy drive a bit unstable. This
  is best used if the floppies are write-protected.
#FastFloppy on
       #
# QuickStart:
# Printer is emulated by piping printer data to a file or via a unix
# command such as "lpr". Don't bother fiddling with this configuration
# until you've got DOSEMU up and running already.
#printer { options "%s" command "lpr" timeout 20 }
#printer { options "-p %s" command "lpr" timeout 10 } # pr format it
#printer { file "lpt3" }
```

Za promjenu podešenja u ovoj datoteci trebate upotrijebiti program za uređivanje teksta. Nakon toga bi datoteka trebala odgovarati vašem sustavu, što je posebno važno za elemente kao što su procesor i grafička kartica.

BILJEŠKA Emulator DOSEMU možete pokrenuti i s tvrdog diska, umjesto s diskete. Za pristup tvrdom disku jednostavno podesite pogon/particiju u datoteci dosemu.conf.

# Pokretanje emulatora DOS-a

Da biste pokrenuli emulator DOSEMU, jednostavno upišite DOS u bilo koji Linux prompt, a za napuštanje upotrijebite naredbu exitmenu. Tablica 5.9 sadrži popis opcija naredbi koje možete koristiti kod pokretanja emulatora. Da biste dobili čitav i ažuriran popis opcija, upotrijebite opciju -?.

Tablica 5.9	blica 5.9 Parametri naredbi za emulator DOSEMU	
Parametar	Opis	
-A	Pokretanje s pogona A	
-C	Pokretanje s tvrdog diska	
-c	Optimizacija grafičkih značajki virtualnih terminala	
-D	Podešavanje opcija za otklanjanje grešaka (debug)	
-е	Određivanje količine EMS memorije	
-F#	Označava broj (#) disketnih pogona koji se koriste	

Parametar	Opis	
-f	Zamjena A i B disketnog pogona	
-H#	Označava broj (#) tvrdih diskova koji se koriste	
-k	Koristi se konzola koja je definirana u parametru rawkeyboard (datoteka dosemu.conf)	
-P	Kopiranje u datoteku podataka o otklonjenim greškama	
-t	Prosljeđuje vremenski prekid (time interrupt) 9	
-V	Aktivira se VGA emulacija	
-X	Određuje količinu XMS memorije	
-?	Prikaz sažete pomoći za sve naredbe	
-2	Emulacija 286-tice	
-3	Emulacija 386-tice	
-4	Emulacija 486-tice	

Iz DOS *prompta* (kojeg podržava DOSEMU) možete pokrenuti većinu DOS programa, isključujući one koji zahtijevaju sučelje DPMI (DOS Protected Mode Interface). Jednostavno upišite ime programa i nadajte se da će DOSEMU naći program, a zatim ga pokrenuti.

Tablica 5.10 prikazuje neke programe za koje se zna da rade pod Linuxom, a svakodnevno se isto može ustanoviti za sve veći broj programa. Provjerite u datoteci EMUsuccess.txt koja se nalazi u istom direktoriju s emulatorom DOSEMU, koji su novi programa na popisu. U tablici 5.11 ćete naći programe koji ne rade pod Linuxom.

Tablica 5.10. Program	). Programi koji rade s emulatorom DOSEMU	
Ime	Namjena	Potvrda pravilnog rada
1st Wordplus	GEM tekst procesor	jan@janhh.hanse.de
4desc	4dos desc editor	piola@di.unito.it
4DOS 4.2	Interpret. naredbi	rideau@clipper.ens.fr
4dos 5.0c	Interpret. naredbi	J1MCPHER@VAXC. STEVENS-TECH.EDU
ack3d	3-D paket	martin5@trgcorp.solucorp.qc.ca
ACU-COBOL	Kompilator	fjh@munta.cs.mu.OZ.AU
Alite 1.10		ph99jh42@uwrf.edu
AmTax 93 & 94	Financ. softver	root@bobspc.canisius.edu

Tablica 5.10. Nasta	avak	
Ime	Namjena	Potvrda pravilnog rada
ansi.sys	Upravljački program za zaslon/tipkovnicu (prikaz funkcija)	ag173@cleveland. Freenet.Edu
arj v2.41a	Program za komprim.	tanner@winternet.mpls.mn.us
As Easy As 5.01	Prorač. tablica	ph99jh42@uwrf.edu
Autoroute Plus	Auto-karta	hsw1@papa.attmail.com
Axum	Grafika	miguel@pinon.ccu.uniovi.es
battle chess	Šah	jvdbergh@wins.uia.ac.be
Binkley 2.50eebd	Program za poštu	stub@linux.rz.tu-clausthal.de
Blake Stone_	Igra	owaddell@cs.indiana.edu
bnu 1.70	Fossil (Fido)	stub@linux.rz.tu-clausthal.de
Borland C++2.0	86/286 C/C++IDE	rideau@clipper.ens.fr
Boston Business EDT+		keegstra@csdr2.fsfc. nasa.gov
Cardbox Plus	Baza podataka	hsw1@papa.attmail.com
Castle Wolfenstein	3-D igra	gt8134b@prism.gatech.EDU
Checkit diagnostics clipper 5.1	dBASE kompilator	jvdbergh@wins.uia.ac.be
COMPRESS	Kompr. diska	rideau@clipper.ens.fr
CCM (Crosstalk)	Program za modeme	
cshow 8.61	Pregled slika	jvdbergh@wins.uia.ac.be
cview	Pregled slika	lotov@avarice.ugcs.caltech.edu
d86/a86		
DataPerfect 2.1	Baza podataka	fbennett@uk.ac.ulcc.clus1
Dbase		corey@amiganet.xnet.com
Derive 1.2	Paket za matematiku	miguel@pinon.ccu.uniovi.es
Disk Freedom4.6	Uslužni program za disk	
diet 1.45f	Komprimiranje datoteka	stub@linux.rz.tu-clausthal.de
dosnix 2.0	UNIX uslužni program	miguel@pinon.ccu.uniovi.es
Dosshell task	Swapper	jmaclean@fox.nstn.ns.ca
dtmm	Molekularni modeli	miguel@pinon.ccu.uniovi.es

Ime	Namjena	Potvrda pravilnog rada
Dune 2	Igra	COLIN@fs1.in.umist.ac.uk
dviscr	EMTEX dvi pregled	ub9x@rz.uni-karlsruhe.de
Easytrax	Program za uređivanje teksta	maehler@wrcd1.urz. uni-wuppertal.de
Elvis	Klon vi editora	miguel@pinon.ccu.uniovi.es
Epic Pinball	Igra	krismon@quack.kfu.com
Eten 3.1	Kineski terminal	tyuan!root@mp.cs.niu.edu
Eureka 1.0	Paket za matematiku	miguel@pinon.ccu.uniovi.es
Falcon 3.0	Simulator borbi	rapatel@rockypc.rutgers.edu
FastLST 1.03	FidoNdlst kompilator	stub@linux.rz.tu-clausthal.de
FormGen II		root@bobspc.canisius.edu
freemacs 1.6d	Program za uređivanje teksta	ph99jh42@uwrf.edu
Frontier (Elite II)	Igra	COLIN@fs1.in.umist.ac.uk
FW3		Sebastian.Bunka@vu-wien.ac.at
MS Flight Simulator 5	Igra (radi sporo!)	newcombe@aa.csc.peachnet.edu
Foxpro 2.0	Baza podataka	
Framework 4		corey@amiganet.xnet.com
Framework 4		corey@amiganet.xnet.com
Freelance Graphics 2.1	Grafika/crtanje	jwest@jwest.ecen.okstate.edu
GEM/3	GUI	jan@janhh.hanse.de
GEM Draw	GEM prog. za crtanje	jan@janhh.hanse.de
GEM Paint	GEM prog. za crtanje	jan@janhh.hanse.de
gmouse	Upravljački prog. za miša	tk@pssparc2.oc.com
God of Thunder	Igra	ensor@cs.utk.edu
Gravity	Paket za simulaciju	miguel@pinon.ccu.uniovi.es
GWS for DOS	Konverter grafičkih datoteka	bchow@bchow.slip
Gzip 1.1.2	Kompresija datoteka	miguel@pinon.ccu.uniovi.es
Harpoon	Igra	wielinga@physics.uq.oz.au
Harvard Graphics 3.0	Paket za grafiku/ crtanje	miguel@pinon.ccu.uniovi.es
		nustavij

Tablica 5.10. Nasta	avak	
Ime	Namjena	Potvrda pravilnog rada
Hero's Quest I	Igra	lam836@cs.cuhk.hk
Hijaak 2.0	Konverter grafičkih datoteka	bchow@bchow.slip
hocus pocus	Apogee igra	kooper@dutiws.TWI.TUDelft.NL
Image Alchemy Pro	Konverter grafičkih datoteka (-v ne radi)	J1MCPHER@VAXC STEVENS-TECH.EDU
Incredible Machine	Igra (sporo)	sdh@po.cwru.edu
Key Spreadsheet Plus	Prorač. tablica	jwest@jwest.ecen.okstate.edu
Lemmings		sdh@po.cwru.edu
less 1.7.7	Više od pograma more	miguel@pinon.ccu.uniovi.es
LHA	Komprimiranje datoteka	
Lotus Manuscript	Tekst procesor	miguel@pinon.ccu.uniovi.es
Managing Your Money	Financijski program	newcombe@aa.csc.peachnet.edu
Manifest	("umire" kod provjere memorije)	hsw1@papa.attmail.com
Mathcad 2.01	Paket za matematiku	root@bobspc.canisius.edu
MathCad 2.06	Paket za matematiku	miguel@pinon.ccu.uniovi.es
mcafee 9.23 v112	Antivirus	jvdbergh@wins.uia.c.be
Microemacs	Obrada teksta	hjstein@MATH.HUJI.AC.IL
MicroLink Yaht 2.1		root@bobspc.canisius.edu
Microsoft C 6.0	Kompilator	ronnie@epact.se
Microsoft Assembler 5.0	Assembler	ronnie@epact.se
Microsoft Library 2.0		root@bobspc.canisius.edu
Microsoft Make		ronnie@epact.se
MicrosoftMouse Drv 8.2	Uprav. program za miša	hsw1@papa.attmail.com
MoneyCounts 7.0	Program za račune	raeburn@cygnus.com
mscmouse	Uprav. program za miša	tk@pssparc2.oc.com
nnansi.com	ANSI upravljački program	mdrejhon@undergrad. math.uwaterloo.ca

Ime	Namjena	Potvrda pravilnog rada
Netzplan	GEM vođenje projekata	jan@janhh.hanse.de
NHL Hockey	Igra	krismon@quack.kfu.com
NJStar 2.1	Kineski tekst procesor	aab2@cornell.edu
Norton Utils 4.5	Uslužni programi za disk	rideau@clipper.ens.fr
Norton Utils 7.0	Uslužni programi za disk	rideau@clipper.ens.fr
PAF	Paket za geneaologiju	geek+@CMU.EDU
Paradox	Baza podataka	hp@vmars.tuwien.ac.at
PC Paintbrush IV	Program za crtanje	bchow@bchow.slip
Pctools 4.20	Uslužni programi za disk	rideau@clipper.ens.fr
pcwdemo		vinod@cse.iitb.ernet.in
PC-Write 3.0	Tekst procesor	
pcxlab 1.03	PCX preglednik	miguel@pinon.ccu.uniovi.es
peachtree complete 6.0	Računovodstvo	stjeanp@math.enmu.edu
Pinball Dreams	Igra	ronnie@lysator.liu.se
Pkzip/unzip	Komprimiranje datoteka	
pklite 1.15	Komprimiranje datoteka	stub@linux.rz-tu-clausthal.de
Pong Kombat	Igra	ensor@cs.utk.edu
PrintShop	Paket sa čestitkama	geek+@CMU.EDU
Procomm Plus 2.0	Komunikacija	newcombe@aa.csc.peachnet.edu
Procomm 2.4.3	Komunikacija	hsw1@papa.attmail.com
Pspice 5.0	Kružna simulacija	root@bobspc.canisius.edu
Q&A	Tekst proces./Baza podataka	newcombe@aa.csc.peachnet.edu
Qbasic/edit	Interpretator (za DOS 5.0)	
Qedit	Program za uređivanje teksta	
QuickC	Kompilator	martin@trcsun3.eas.asu.edu
Quicken 4.0 for DOS	Knjigovodst. paket	juphoff@nrao.edu
Quicken 6.0 for DOS	Knjigovodst. paket	
Quicken 7.0 for DOS	Knjigovodst. paket	juphoff@astro.phys.vt.edu
Railroad Tycoon		juphoff@astro.phys.vt.edu

Tablica 5.10. Nastavak		
Ime	Namjena	Potvrda pravilnog rada
Red Baron	Igra	wielinga@physics.uq.oz.au
RM/COBOL	Kompilator	fjh@munta.cs.mu.OZ.AU
Rpro 1.6		root@bobspc.canisius.edu
scan109	Antivirus	miguel@pinon.ccu.uniovi.es
scan112	Antivirus	ppiola@di.unito.it
Scorch	Igra	geek+@CMU.EDU
Shez94	Arcer-Shell	stub@linux.rz.tu-clausthal.de
sled	Obrada teksta	piola@di.unito.it
Space Quest IV	Igra	lam836@cs.cuhk.hk
Spell Casting 301		mancini@phantom.com
SPSS/PC+4.0	Statistički paketi	jr@petz.han.de
Squish 1.01	Fido Scan/Tosser	stub@linux.rz.tu-clausthal.de
Stacker 3.1	Kompr. diska	mdrejhon@undergrad math.uwaterloo.ca
Stacker 4.00	Kompr. diska	J1MCPHER@VAXC. STEVENS-TECH.EDU
StatPhys	Simulacijski paket	miguel@pinon.ccu.uniovi.es
STSORBIT	Simulacija orbite	troch@gandalf.rutgers.EDU
Stunts	Igra?	gt8134b@prism.gatech.EDU
Superstor	Kompr. diska	rideau@clipper.ens.fr
TAG 2.02	Poljski tekst procesor	rzm@oso.chalmers.se
TASM 2.51	MACRO assembler	rideau@clipper.ens.fr
Telix	Program za modele	jou@nematic.ep.nctu.edu.tw
Thelp from BC++2.0	Iskočna pomoć	rideau@clipper.ens.fr
TimED/beta	Fido MSGeditor	stub@linux.rz.tu-clausthal.de
TLINK 4.0	LINKER	rideau@clipper.ens.fr
Topspeed Modula-2	Kompilator	mayersn@hermes. informatik.uni-stuttgart.de
Turbo Debugger 2.51	Realmode debugger	rideau@clipper.ens.fr
Turbo Pascal 5.5	Kompilator	

Ime	Namjena	Potvrda pravilnog rada
Turbo Pascal 6.0	Kompilator	t2262dj@cd1.Irz-muenchen.de
Turbo Pascal 7.0	Kompilator	mdrejhon@undergrad. math.uwaterloo.ca
Turb-opoly 1.43		root@bobspc.canisius.edu
Ultima 6	Igra	mpshil@birds.wm.edu
Vpic 6.1		root@bobspc.canisius.edu
warlords II	Igra	buckel@cip. informatik.uni-wuerzburg.de
Warrior of destiny	Igra	msphil@birds.wm.edu
WITWI Carmen Sandiego	Igra	tillemaj@cae.wisc.edu
Windows 3.0	Windows (real mode)	cjw1@ukc.ac.uk
Wolf3d	Igra	owaddell@cs.indiana.edu
WordPerfect 5.1	Tekst procesor	sdh@po.cwru.edu
WordPerfect 6.0	Tekst procesor (treba više od 1Mb RAM-a)	lujian@texmd.minmet.mcgill.ca
Xtpro 1.1	Uslužni programi za disk	root@bobspc.canisius.edu
Xwing	Igra (vrlo spora)	ronnie@lysator.liu.se
Zarkov 2.6	Šah	a-acero@uchicago.edu
Z00	Komprimiranje datoteka	

Tablica 5.11.	Programi	koji ne I	rade s	emulatorom	DOSEMU
---------------	----------	-----------	--------	------------	--------

Ime	Namjena	Potvrda
4D-box	Igra	jvdbergh@wins.uia.ac.be
Apple ĆŠ emulator	Emulator	ph99jh42@uwrf.edu
Borland C++ 3.1 IDE	Kompilator	juphoff@uppieland.async.vt.edu
brief	Obrada teksta	bchow@bchow.slip
Chuck Yeager Aircombat	Simulator leta	jvdbergh@wins.uia.ac.be
CIVILIZATION	Igra	miguel@pinon.ccu.uniovi.es

-

Tablica 5.11. Nastava	k	
Ime	Namjena	Potvrda
DesqView 2.51	(Tipka Alt ne radi)	hsw1@papa.attmail.com
doom	Igra	rideau@clipper.ens.fr
dpms from Stacker 4.0		J1MCPHER@VAXC. STEVENS-TECH.EDU
dxma0mod.sys	Token-ring uprav. program	adjihc4@cti.ecp.fr
dxmc0mod.sys	Token-ring uprav. program	adjihc4@cti.ecp.fr
ELDB	Baza podataka za ekonomiste	hjstein@math.huji.ac.il
FIPS 0.2.2	Uslužni program za disk	
Howitzer	Igra	geek+@CMU.EDU
Lahey Fortran	Fortran kompilator	hjstein@math.huji.ac.il
Maple V2	Matematički paket	ralf@ark.btbg.sub.de
MSDOS 5/6 QBASIC/EDIT	Obrada teksta	bchow@bchow.slip
NORTON UTILITES 7.0	Uslužni programi za disk	bchow@bchow.slip
Quattro Pro 4.0	Proračunska tablica	jwest@jwest.ecen.okstate.edu
Raptor	Igra	ensor@cs.utk.edu
Silent Service II	Igra s podmornicom	jvdbergh@wins.uia.ac.be
thunderByte scan	Antivirus	jvdbergh@wins.uia.ac.be
Ventura Publisher 3.0	Stolno izdavaštvo	niemann@swt.ruhr-uni-bochum.de
wildunix	Zamjenski znakovi	miguel@pinon.ccu.uniovi.es
Windows 3.1		juphoff@uppieland.async.vt.edu

Pokretanje programa pomoću emulatora DOSEMU može izazvati nekoliko problema, većinom stoga što računalo emulira DOS i pripadno računalo, a stvarno ne pokreće DOS. Emuliranje usporava sustav, što može postati neodrživo, osobito ako istovremeno pokrenete druge Linux programe na drugim virtualnim terminalima. Također je s emulatorom DOSEMU usporeno i ažuriranje slike (zaslona).

Ovaj fajl je skinut sa sajta www.tutoriali.org

Mnogi DOS programi "okupiraju" procesor jer im se čini da jedino oni trenutno rade. Na taj je način drugim Linux programima zapriječen pristup procesoru. Da bi riješio nastali problem, gospodin Thomas G. McWilliams je napisao program nazvan *garrot* koji oslobađa pristup procesoru. Program *garrot* možete naći na FTP lokaciji sunsite.unc.edu, u direktoriju /pub/linux/alpha/dosemu.

# Pokretanje Windows programa pod Linuxom

Emulator DOSEMU ne može pokrenuti Microsoft Windows programe, tako da su korisnici Linuxa morali stvoriti program koji će Linux korisnicima omogućiti pokretanje Windows programa. Emulator Windowsa se zove *Wine* i predstavlja kraticu od *WIN*dows *E*mulator, a može značiti i *W*ine *I*s *N*ot a Windows *E*mulator jer je Wine stvoren kao biblioteka (library), a ne kao emulator.

S obzirom da Wine nije tako dobro razvijen kao emulator DOSEMU, morat ćete posezati za odgovorima na često postavljana pitanja (FAQ). Osim što se Wine nalazi u eksperimentalnoj fazi, te ima dosta grešaka, mnogi Windows programi niti ne mogu raditi s programom Wine. Da biste koristili Wine, potrebno je da na posebnoj particiji imate instalirane Windowse, te da Linux ima pristup toj particiji. Osim toga, trebate instalirati i pokrenuti XWindowse da biste mogli koristiti Wine.

Da biste provjerili program Wine, potrebno vam je sljedeće:

- Linux jezgra, inačica 99.13 ili novija
- Izvorni kod programa Wine (jer ga takvog jedino i možete nabaviti)
- Paket d koji je instaliran za kompilatore izvornog koda
- Najmanje 8 MB RAM-a i 12 MB *swap* prostora
- Najmanje 10 MB prostora na tvrdom disku
- Instalirane i podešene XWindowse
- Pokazivački uređaj (miš primjerice)
- Microsoft Windowsi trebaju biti instalirani na particiji kojoj se može pristupiti iz Linuxa

S obzirom da je Wine još u razvoju, gotovo svaki tjedan se pojavljuju nove inačice. Najnoviji izvorni kôd se nalazi na adresi **sunsite.unc.edu** (i na drugim glavnim FTP lokacijama) u direktoriju /pub/Linux/ALPHA/wine/development. Ime datoteke je povezano s datumom kada je objavljena, primjerice wine-961201.tar.tgz.

Vidi "Uporaba FTP-a za daljinski prijenos datoteka", 580. stranica

S obzirom da se Wine brzo mijenja i da je nestabilan, budite oprezni. Želite li provjeriti program Wine, slobodno s Interneta kopirajte najnovije datoteke i pročitajte FAQ i HOWTO datoteke koje sadrže informacije koje su vam potrebne za kompilaciju, instalaciju, podešavanje i uporabu programa Wine.

Instalacija programa Wine je vrlo slična instalaciji emulatora DOSEMU, s tom razlikom da izvornu *tar* datoteku možete smjestiti bilo gdje. Pomoću naredbe *tar* možete raspakirati datoteku u direktorij:

[root@web wine]# gzip -d 950606.tar.gz [root@web wine]# tar -xvf 950606.tar

Stvaranje emulatora Wine je nešto drukčije od stvaranja emulatora DOSEMU i u osnovi je nalik kompiliranju nove jezgre (kernela). Najprije trebate odgovoriti na nekoliko pitanja koja se odnose na podešenje programa. Čitav postupak stvaranja emulatora Wine možete pronaći u Wine HOWTO dokumentu.

Zatim trebate odgovoriti na nekoliko pitanja koja se odnose na podešavanje parametara za pokretanje programa. Ti se parametri nalaze u datoteci /usr/local/etc/wine.conf. Tu datoteku možete i ručno preurediti, no ipak je najbolje koristiti program za podešavanje.

Nakon što ste podesili kompilacijske datoteke i parametre za pokretanje programa, možete stvoriti Wine pomoću jednostavne naredbe *make*, što može trajati nekoliko minuta. Da biste koristili Wine, trebate pokrenuti emulator i unijeti stazu (*path*) do Windows programa:

[tackett@web ~]\$wine /docs/windows/winmine.exe

Programi koje Wine podržava su calc.exe, clock.exe, notepad.exe i winmine.exe. Taj se popis programa stalno širi, stoga pročitajte FAQ i HOWTO dokumente i provjerite je li još neki program na popisu.

**BILJEŠKA** MS-DOS i Microsoft Windows nisu jedini operativni sustavi koji se mogu emulirati pod Linuxom. Postoje emulatori za stari sustav Apple II, CPM i novije operativne sustave za Macintosh računala. Te emulatore potražite na FTP lokacijama u direktoriju /pub/Linux/system/emulators.

# Odavde...

U ovom poglavlju smo se tek dotakli rada s Linuxom i raznim programima. Više informacija potražite u ovim poglavljima:

- 6. poglavlje, "Nadogradnja i instaliranje softvera pomoću RPM-a" sadrži informacije koje su vam potrebne da biste instalirali novi softver s CD-ROM-a ili Interneta.
- 8. poglavlje, "Uporaba vi editora" vas upućuje kako ćete koristiti poznati Linux program za obradu teksta.
- 22. poglavlje, "Uporaba XWindowsa" bavi se grafičkim korisničkim sučeljem XWindowsa.

# POGLAVLJE 6

# Nadogradnja i instaliranje softvera pomoću RPM-a

(Jack Tackett)

# U ovom poglavlju

Ključni izrazi koji se koriste u ovom poglavlju Što je nadogradnja Instalacija softvera Uporaba RPM-a Nadogradite svoju jezgru Osnovni operativni sustav Linux sadrži samo najnužnije uslužne programe i datoteke s podacima. Prema potrebi, administrator sustava instalira dodatne naredbe, korisničke programe i razne datoteke s podacima. Programe treba povremeno ažurirati, jer se sustavski softver mijenja kako se pojavljuju dodatne mogućnosti i otklanjaju postojeće greške u programima. Administrator sustava je odgovoran za dodavanje, podešavanje, stvaranje i brisanje softvera s Linux računala.

Riječ *instaliranje* znači kopiranje pripadnih programskih datoteka na tvrdi disk i *podešavanje* aplikacije (odnosno dodjela resursa) za pravilan rad na određenom računalu. Podešavanjem programa smatra se "upućivanje" programa na to gdje se nalaze pojedini njegovi dijelovi, te kako treba funkcionirati unutar određenog sustava.

Red Hat i Caldera distribucije Linuxa koriste naredbu (program) *rpm*, koja olakšava instalaciju i nadogradnju softvera. No, možda ćete trebati instalirati softver koji se ne nalazi u *rpm* formatu (mnogi softverski paketi na Internetu se nalaze u komprimiranom *tar* formatu).

Na velikim sustavima obično administrator instalira aplikacije, jer većina korisnika nema pristup trakama ili disketnim pogonima. Za instalaciju pojedinih komponenti u sustavske direktorije obično je potrebna administratorova dozvola, jer komponente mogu sadržavati dijeljene biblioteke i uređaje koji se trebaju smjestiti u direktorije kojima običan korisnik ne može pristupiti.

# Ključni izrazi koji se koriste u ovom poglavlju

Kao što ste mogli primijetiti čitajući uvod u ovo poglavlje, upute za instalaciju programa sadrže i neke dodatne pojmove. U tablici 6.1 nalazi se popis novih izraza i njihovi opisi koji vam pomažu da ih bolje razumijete.

Tablica 6.1         Izrazi koji se odnose na instalaciju programa		
Izraz	Definicija	
super korisnik	Korisnik s najvišim privilegijima u sustavu. Naziva se i root korisnik.	
Administrator sustava	Osoba koja je zadužena za održavanje Linux sustava. Administrator sustava ima sva prava super korisnika i mogućnost instaliranja novog softvera.	
Instalacija programa	Prva instalacija ili nadogradnja programa koji se koristi u operativnom sustavu UNIX. Taj postupak obično zahtijeva prava super korisnika kao i pristup trakama ili pogonu za diskete.	
Podešavanje	Podešavanje aplikacije tako da može raditi pod određenim sustavom. Podešavanjem programa se omogućuje njegovo korištenje od strane većeg broja korisnika, postavljanje u direktorije kojima korisnici mogu pristupiti ili dijeljenje programa putem mreže.	

# Što je nadogradnja?

Koji softver trebate nadograđivati? Koliko često to trebate činiti? Odgovori na ta pitanja lako se mogu odrediti ako se zna namjena vašeg računala – je li ono poslovno i osobno – te zahtjevi samog korisnika. Programi se svakodnevno mijenjaju. Razni dijelovi Linux sustava se stalno ažuriraju. Ako biste pratili svaku novu inačicu softvera, ne biste ga imali vremena koristiti.

Najčešće ne trebate ponovno instalirati čitav Linux sustav ako nadograđujete sustavski softver, jer se obično samo manji dio softvera mijenja s njegovim novim izdanjem. Možda trebate nadograditi jezgru ili biblioteke sustava, ali ga vrlo vjerojatno nećete trebati ponovno instalirati. No, ako nadograđujete neke softverske pakete, obično trebate ponovno instalirati kompletan sustav, pogotovo ako "kasnite" nekoliko inačica.

Dobro je načiniti sigurnosnu kopiju sustava prije nego krenete u nadogradnju softvera. BILJEŠKA Na taj način ćete, ako naiđete na probleme, uvijek moći vratiti prvobitni sustav.

Vidi "Savjeti koji se odnose na sigurnosne kopije podataka", 227. stranica

Općenito trebate nadograđivati svoj sustav s novom inačicom sustava ili programa ako ste naišli na ozbiljne probleme s postojećim programom ili on ne udovoljava vašim potrebama. No, vi odlučujete što je to "ozbiljan problem". Ako novo izdanje softverskog paketa rješava neki od problema s kojim se često susrećete ili ako je s novim izdanjem otklonjena greška u programu koja bi inače mogla oštetiti vaš sustav, onda je vjerojatno vrijeme da instalirate novu inačicu.

Ne pokušavajte tražiti sve inačice i dijelove softvera, jer nadogradnja radi same nadogradnje uzima puno vremena i truda. Ako se potrudite, uvidjet ćete da vam BILJEŠKA je sustav u dobrom stanju te ažurirajte samo one dijelove koji su vam neophodni.

# Instalacija softvera

Instaliranje glavnog programa u sustav Linux je znatno složenije od instaliranja sličnog programa na operativne sustave kao što su MS-DOS ili Apple Macintosh System 7.6 koje koristi samo jedan korisnik. Mogućnost višekorisničkog rada Linuxa ujedno znači da svakom programu na računalu istovremeno želi pristupiti više korisnika.

Instaliranje dodatno komplicira i to što većina programa, s iznimkom onih jednostavnih, zahtijeva podešavanje prema pojedinom računalu. Administrator sustava treba instalirati softver te identificirati pojedine elemente sustava i program podesiti tako da pravilno radi.

Primjerice, jedan korisnik se može služiti starijim terminalom (koji prihvaća samo znakove), dok drugi može imati novi terminal s XWindows okruženjem. Super korisnik mora program podesiti tako da pravilno radi na starom terminalu (slanjem ASCII znakova) i da istovremeno XWindows terminal može koristiti sve prednosti programa (boja i slike). Administrator sustava upravlja sustavom i odgovoran je za optimalan rad programa (svi programi trebaju biti ažurni, računi korisnika podešeni i slično). Kao što smo već napomenuli, učitavanje programa u sustav Linux je složenije od iste operacije na osobnim računalima. Administrator koji instalira program će trebati stvoriti nove direktorije za smještaj datoteka koje se nalaze u pojedinim softverskim paketima. Neki paketi zahtijevaju i podešavanje hardvera koji se koristi u sustavu. Dok krajnji korisnik brine samo o svladavanju novih mogućnosti programa i njegovim naredbama, super korisnik vodi brigu o tome da su svi resursi sustava pravilno podešeni i ugrađeni

u program (i da, naravno, ne ometaju postojeće aplikacije).

Instalacija softvera putem izbornika ili naredbi je doista jednostavna za čovjeka, dok je ista zadaća puno složenija za sustav. Programi za osobna računala (jedan korisnik), kao što su DOS programi, obično pokreću samo jednu svoju kopiju u određenom vremenu i ne "susreću" se s drugim programima. Čak i najjednostavniji Linux program, sa samo jednim prijavljenim korisnikom, istovremeno obavlja više zadaća. Pomnožite li tu aktivnost s nekoliko korisnika koji istovremeno rade na programu, uvidjet ćete da čitava stvar postaje vrlo složena.

Operativni sustav Linux istovremeno radi s više korisnika, programa i perifernih hardverskih komponenti. Da bi "preživjela" u Linux okruženju, aplikacija mora biti pravilno učitana. Programi koji se čudno ponašaju ili nisu pravilno instalirani mogu prouzročiti *raspad* sustava (pri kojem se određeni proces ili program nekontrolirano ponaša te zaključa CPU, koji na taj način gubi kontrolu nad svim ostalim programima koji trenutno rade). Sustav pada, svi korisnici će izgubiti vezu, a njihovi programi će prestati s radom.

S obzirom da je administrator sustava zadužen za učitavanje novog programa, on je i odgovoran za to da program bude kompatibilan s operativnim sustavom. Isto tako administrator treba provjeriti program nakon što ga je instalirao. Da biste razumjeli učitavanje programa u sustav Linux, najprije trebate naučiti osnovne stvari o pravima i obavezama administratora sustava.

# Posao administratora sustava

Ako Linux upotrebljavate u malom sustavu, tada ste sami administrator sustava te instalirate i pokrećete vlastite programe. Sami ste odgovorni za stvaranje sigurnosnih kopija podataka, oslobađanje potrebne količine prostora na tvrdom disku, optimalno upravljanje memorijom, kao i za sve ostale postupke koji su potrebni za efikasan rad sustava. Ako ste pak samo korisnik u većem sustavu računala, tada je određena osoba zadužena da se bavi vašim sustavom. U nastavku je dat sažetak poslova koje obavlja administrator sustava:

- Prema potrebi pokreće i isključuje sustav.
- Provjerava ima li dovoljno slobodnog prostora na disku te da li sustavi datoteka imaju kakvu grešku.
- Podešava sustav tako da maksimalan broj korisnika ima pristup hardverskim i softverskim izvorima, a da istovremeno sustav radi što brže i efikasnije.

- Štiti sustav od upada neovlaštenih korisnika i njihovih eventualnih destruktivnih radnji.
- Podešava veze s drugim računalnim sustavima.
- Otvara ili zatvara korisničke račune u sustavu.
- Surađuje s dobavljačima softvera i hardvera, te omogućuje obuku korisnika.
- Instalira terminale, pisače, diskove i druge dijelove sustava i perifernog hardvera, te otklanja probleme vezane uz njih.
- Instalira i podešava nove programe, ažurira operativni sustav i otklanja greške u softveru.
- I ne čini ništa više. Često se korisnici prijavljuju kao super korisnici i sav posao obavljaju s tom lozinkom, no na taj način mogu stvoriti brojne poteškoće u radu sustava. Prednosti super korisnika upotrebljavajte za upravljanje sustavom, a svoj korisnički račun koristite za svakodnevne poslove.

# Uporaba RPM-a

Red Hat i Caldera OpenLinux distribucije koriste pakete za instalaciju softvera. Paketi sadrže cjelovite, potpuno provjerene i podešene programe i obično sadrže izvorni kôd paketa tako da programeri i korisnici znaju što su dobili. Da biste upravljali tim paketima, tvrtka Red Hat Software je razvila RPM (Red Hat Package Manager).



#### NA WEBU

Pogledajte http://www.rpm.org i saznajte nešto više o programu RPM.

Program RPM ima šest načina rada, od kojih se pet mogu koristiti ili iz retka za zadavanje naredbi ili pomoću alata Glint (temeljen na XWindowsima). Radi se o sljedećem: instaliranje, deinstaliranje, ažuriranje, postavljanje upita, verifikacija i stvaranje (jedino stvaranje RPM paketa pokrećete isključivo iz retka za zadavanje naredbi).

Pokretanje programa RPM iz retka za zadavanje naredbi je sljedeće:

rpm [opcije] ime paketa

gdje izraz *opcije* predstavlja mnoge parametre koje program RPM koristi za upravljanje paketima, a izraz *ime paketa* upućuje na softverski paket koji se koristi. Ime paketa obično ovako izgleda: quota-1.55-4.i386.rpm, što u osnovi znači:

ime	quota
inačica	1.55
izdanje	4
računalo	i386
ekstenzija	. rpm (najčešće)

No ime paketa može biti bilo kakvo jer se informacije o samom paketu nalaze u samoj datoteci.

#### Gdje se nalaze paketi?

Većina softverskih paketa koje ste dobili zajedno s distribucijom nalaze se na pripadnom CD-ROM-u, i to u direktoriju /RedHat/RPMS. Da biste postavili CD-ROM i pregledali popis raspoloživih paketa, trebate upotrijebiti sljedeće naredbe:

```
cd /mnt
mount CD-ROM
cd CD-ROM/RedHat/RPMS
ls ¦ more
```

Većina je tih paketa instalirana na vaše računalo tijekom instalacije Linuxa. Ipak, ako niste instalirali određeni paket, to možete učiniti sada.

```
Vidi "Instalacija softverskih komponenti", 75. stranica.
```

RPM također omogućuje instalaciju softverskih paketa s drugih računala, korištenjem protokola FTP (o čemu ćete saznati u idućem odlomku).

## Instalacija paketa pomoću RPM-a

Da biste instalirali softverski paket, potrebno je upotrijebiti opciju -i, i to na sljedeći način:

rpm -i quota-1.55-4.i386.rpm

Pomoću ove naredbe ćete instalirati paket *quota* na svoje računalo. Uporabom opcije -i instalirat ćete paket quota-1.55-4.i386.rpm na lokalno računalo. Prilikom instalacije paketa RPM obavlja sljedeće radnje:

*Provjera međuzavisnosti.* Svaki paket može ovisiti o drugom softveru koji je prethodno instaliran.

*Provjera konflikata.* RPM provjerava da li je određena komponenta već instalirana te je li određena komponenta na računalu starija od one koju instalirate.

*Pohrana datoteka s podešenjima*. RPM nastoji pronaći odgovarajuću datoteku s podešenjima, a ako je nađe program je sprema za buduću uporabu.

*Instaliranje datoteka.* RPM dekomprimira razne komponente paketa i sprema ih u odgovarajuće direktorije.

*Postupak nakon instalacije*. Nakon instaliranja raznih komponenti, RPM obavlja potrebna podešavanja sustava.

Ažuriranje baze podataka. RPM zapisuje sve radnje u bazu podataka.

Tijekom instalacije na zaslonu nećete vidjeti nikakve poruke (odziv), ali možete upotrijebiti opciju -v i dobiti više informacija. Tablica 6.2 sadrži popis opcija koje možete upotrijebiti tijekom instalacije.

Tablica 6.2         Opcije za instalaciju paketa		
Opcija	Opis	
- V V	Prikazuje vrlo opširne informacije.	
- h	Tijekom instalacije ispisuje oznaku (#), što vam omogućuje da vidite da RPM nešto radi i da se nije zaglavio.	
percent	Umjesto oznake #, ova opcija ispisuje postotak dovršenosti instalacije.	
test	Ne instalira softverski paket, ali omogućuje provjeru instalacije i javlja moguće greške.	
replacefiles	Zamjenjuje datoteke iz drugih paketa.	
force	Upućuje RPM da zanemari određene moguće greške pri instalaciji, te da paket u svakom slučaju instalira.	

Da biste instalirali paket koji se nalazi na drugom računalu, morate upotrijebiti URL adresu:

rpm -i ftp://ftp.netwarf.com/pub/RPMS/quota-1.55-4.i386.rpm

U ovoj naredbi je pretpostavljeno da udaljeno računalo prihvaća anonimnu prijavu.

Vidi "Uporaba FTP-a pomoću WEB pretraživača", 601. stranica

Ako trebate upisati korisničko ime i lozinku da biste instalirali datoteku, tada pokrenite sljedeću naredbu:

```
rpm -i ftp://mark@ftp.netwharf.com/pub/RPMS/quota-1.55-4.i386.rpm
Password for mark@ftp.netwharf.com: <ovdje unesite svoju lozinku>
```

BILJEŠKA Svoje korisničko ime i lozinku možete unijeti i na ovaj način:

rpm -i ftp://mark: password@ftp.netwharf.com/pub/RPMS/ quota-1.55-4.i386.rpm

No, to nije najsigurniji način za unos naredbe jer nepozvana osoba može vidjeti koju ste lozinku upisali ili kasnije pozvati naredbu iz datoteke *history*.

# Deinstalacija paketa pomoću RPM-a

Jedna od prednosti korištenja RPM-a je ta što omogućuje laganu instalaciju novih programa. Ako ste čuli za novi program koji se pojavio na Internetu, možete instalirati paket i provjeriti novi program. Što će se dogoditi ako ustanovite da vam softver ne odgovara te ga se želite riješiti? Na sreću, pomoću RPM-a je deinstalacija programa jednako lagana kao i njegova instalacija. Da biste uklonili program, trebate upotrijebiti opciju –*e*:

```
rpm -e quota-1.55-4.i386.rpm
```

Kada brišete paket sa svog računala, RPM obavlja sljedeće radnje:

*Provjerava međuzavisnost.* RPM provjerava u bazi podataka ovisi li neki drugi paket o onome koji želite obrisati. U tom slučaju program ne briše odabrani paket bez izričitog zahtjeva.

Provodi slijed naredbi (skripte). RPM će provesti slijed naredbi prije deinstalacije.

*Provjerava datoteke s podešenjima*. RPM će spremiti kopiju svih promijenjenih datoteka s podešenjima.

Briše datoteke. RPM briše sve datoteke koje se nalaze u određenom paketu.

Provodi slijed naredbi (skripte). RPM će provesti slijed naredbi poslije deinstalacije.

*Ažurira bazu podataka*. RPM iz baze podataka uklanja sve naredbe koje pozivaju obrisani paket.

Kao i kod opcije –*i*, moći ćete upotrijebiti opcije –*v* i –*vv* da biste dobili povratne podatke od naredbe *erase*. Također možete upotrijebiti i opciju - *- test*, koja vas upućuje na moguće probleme ako obrišete određeni paket. Na kraju možete iskoristiti i opciju - *- nodeps* kako biste naredili programu RPM da zanemari međuzavisnosti paketa koji želite obrisati s drugim programima, te da nastavi s brisanjem paketa.

#### **UPOZORENJE**

Budite pažljivi pri uporabi opcije - - nodeps. Ako uklonite paket o kojem ovisi neki drugi program, može se dogoditi da taj program ubuduće ne radi ispravno.

# Ažuriranje paketa pomoću RPM-a

Nakon što ste instalirali paket, trebat ćete ga možda nadograditi radi otklanjanja grešaka ili zbog novih mogućnosti programa. RPM ovaj, inače neugodan zadatak, rješava uporabom opcije -U (uočite veliko slovo U!). Recimo da je netko dodao nekoliko novih mogućnosti u program *quota* i objavio novi paket koji se zove quota-1.55-4.i386.rpm. Za nadogradnju tog paketa trebate pokrenuti ovu naredbu:

rpm -U quota-1.55-4.i386.rpm

Tijekom nadogradnje program RPM instalira odabrani softverski paket i potom briše sve stare inačice paketa (ako one postoje). RPM troši znatan dio vremena za podešavanje pripadnih datoteka, stoga ćete tijekom nadogradnje vidjeti, primjerice, ovakvu poruku (koja upućuje da je datoteka s podešenjima pohranjena pod novim imenom):

Saving syslog.conf to syslog.conf.rpmsave

Takva poruka upućuje na to da je RPM stvorio novu datoteku s podešenjima koja će biti usklađena s vašim sustavom. Poslije nadogradnje trebate usporediti dvije datoteke s podešenjima i načiniti potrebne preinake u novoj datoteci.

# Postavljanje upita pomoću RPM-a

Da biste vidjeli koji su paketi instalirani na vašem računalu, trebate upotrijebiti sljedeću naredbu:

rpm -qa

Na taj ćete način dobiti popis paketa koji su trenutno instalirani na vašem računalu. Da biste dobili informacije o pojedinom paketu, dovoljno je upotrijebiti parametar -q. Tablica 6.3 sadrži razne opcije koje možete koristiti uz naredbu rpm -q.

Tablica 6.3 Opcije za upite			
Opcija	Opis		
-q <i>ime</i>	Prikazuje ime, inačicu i broj izdanja paketa.		
-qa	Daje popis svih instaliranih paketa.		
-qf <i>datoteka</i>	Upit za paket u kojem se nalazi datoteka.		
-qp <i>paket</i>	Upit za paket.		
-qi <i>paket</i>	Prikazuje ime, opis, inačicu, veličinu, datum nastanka, datum instalacije i druge informacije o paketu.		
-ql <i>paket</i>	Popis datoteka koje se nalaze u paketu.		

#### **UPOZORENJE**

Razne -q opcije ne rade dobro ako odaberete simbolički povezane datoteke. Da biste dobili bolje rezultate, najprije pokrenite naredbu *cd* u odgovarajućem direktoriju, a tek potom pokrenite neku od -q opcija.

Vidi "Veze", 308. stranica

Primjerice, ako nađete novi paket i želite saznati što više informacija o njemu, možete upotrijebiti sljedeću naredbu:

rpm -qip quota -1.55-4.i386.rpm

Na zaslonu bi se trebao pojaviti približno ovakav ispis:

```
Name
             : guota
                                  Distribution: Manhattan
Version
            : 1.55
                                        Vendor: Red Hat Software
Release
             : 9
                                    Build Date: Thu May 7 22:45:48 1998
Install date : (not installed) Build Host: porky.redhat.com
Group
            : Utilities/System
                                   Source RPM: quota-1.55-9.src.rpm
            : 82232
Size
Packager
            : Red Hat Software <bugs@redhat.com>
            : Quota administration package
Summary
Description
            :
Quotas allow the system administrator to limit disk usage by user and/or
group per filesystem. This package contains the tools which are needed to
enable, modify, and upadate quotas.
```

### Verifikacija paketa pomoću RPM-a

Na kraju RPM verificira paket. Možda ćete trebati provjeriti konzistentnost datoteka u vašem računalu. Pretpostavimo da sumnjate da je slučajno oštećena neka datoteka.

Trebate usporediti postojeće datoteke s njihovim originalima. RPM vam omogućuje da to učinite koristeći opciju -V (uočite veliko slovo V). Verifikacijom paketa uspoređuje se veličina, MD5 *checksum*, dozvole za pristup datoteci, vrsta datoteke i vlasnik datoteke, kao i podešenje grupe. Da biste provjerili jesu li se datoteke promijenile od trenutka kada su instalirane, pokrenite naredbu *rpm* -V *imepaketa*. Da biste provjerili paket *quota*, trebate upisati ovu naredbu:

rpm -V quota

Ako nisu nastale nikakve promjene, na zaslonu nećete vidjeti nikakvu poruku. Ako se nešto promijenilo, RPM će prikazati oznaku od osam znakova koja upućuje na nastale promjene te na ime datoteke koja je promijenjena. Nakon toga trebali biste provjeriti različite datoteke u paketu te odlučiti treba li ponovno instalirati oštećeni paket. Tablica 6.4 sadrži popis mogućih oznaka.

Tablica 6.4	Popis oznaka za verifikaciju paketa
Oznaka	Značenje
с	Datoteka je konfiguracijska.
5	Datoteka nije prošla provjeru MD5.
S	Od instalacije se promijenila veličina datoteke.
L	Problem sa simboličnim vezama.
Т	Vrijeme mijenjanja datoteke (modification time) ne odgovara originalu.
D	Atribut uređaja.
U	Promijenilo se korisničko podešenje.
G	Promijenilo se podešenje grupe.
М	Promijenila se vrsta datoteke ili dozvola pristupa.

# Instalacija softvera koji ne pripada Linuxu

Nažalost, većina softvera s kojim ćete se susresti nije u formatu RPM. Obično se programi kopiraju preko anonimnog FTP s neke lokacije s arhivama. Instalacija softvera može varirati od iznimno jednostavne do gotovo nemoguće, a sve to ovisi o skriptama naredbi koje su napisali autori softvera i o dokumentaciji za instaliranje.

**Objašnjenje formata softverskih paketa.** Softverski paketi koje dobivate putem anonimnog FTPa će obično biti u obliku komprimirane datoteke *tar*, koje se mogu stvoriti na nekoliko različitih načina. Obično stablo direktorija sadrži datoteke s kodom, izvršne datoteke i druge potrebne datoteke koje su pohranjene u datoteci *tar*. Najčešće je ova datoteka komprimirana kako bi se uštedio prostor. Softverski paket će obično imati i dodatak imenu datoteke, koji upućuje na njegov format. Ako datoteka završava s .gz, to znači da je komprimirana pomoću programa *GNU gzip*. Radi se o najčešće korištenom formatu za komprimiranje Linux paketa. Ako ime arhive završava sa .Z, ona je nastala pomoću programa *compress*. Primjerice, softverski paket foo.tar.gz je *tar* arhiva koja je komprimirana pomoću programa gzip.

BILJEŠKA Ponekad datoteka *tar* može biti komprimirana pomoću programa gzip i imati dodatak imenu .tgz, umjesto .tar.gz.

Vidi "Uporaba programa tar", 229. stranica

**Instalacija softvera** Iduće što morate učiniti nakon što ste pronašli softverski paket jest da odredite gdje ćete spremiti izvorne datoteke. Neki softverski paketi su prilično veliki, tako da je dobro smjestiti ih na onaj sustav datoteka koji ima dovoljno slobodnog prostora. Neki korisnici stvaraju poseban sustav datoteka za izvorne datoteke, primjerice /usr/local/src ili /src. Gdje god da odlučite postaviti svoj softverski paket, najprije provjerite ima li dovoljno prostora na disku kako bi se softver mogao uspješno kompilirati.

Sada možete krenuti dalje i softverski paket premjestiti na izvorno stablo koje ste prethodno podesili, a zatim paket trebate dekomprimirati. Ako je datoteka komprimirana pomoću naredbe *gzip*, možete je dekomprimirati koristeći naredbu *gzip* –*d*. Primjerice, naredba

gzip -d foo.tar.gz

Tablica 6.5	Parametri za naredbu gzip		
Parametar	Ime	Opis	
- a	ascii	ASCII tekst; pretvara oznake za kraj retka koristeći lokalna pravila.	
- C	stdout	Standardni ispis; originalne datoteke ostaju nepromijenjene.	
- d	decompress	Dekomprimiranje.	
- f	force	Forsirano brisanje stare i pisanje nove izlazne datoteke i komprimiranje veza.	
- h	help	Daje pomoć u radu.	
- 1	list	Popis sadržaja komprimirane datoteke.	
- L	license	Licenca softvera.	
- n	no-name	Niti sprema niti obnavlja originalna imena i markicu vremena.	
- N	name	Sprema i obnavlja originalno ime i markicu vremena.	
- q	quiet	Onemogućuje sva upozorenja.	

dekomprimira datoteku foo.tar.gz i zamjenjuje je s *tar* arhivom pod imenom foo.tar. U tablici 6.5 možete vidjeti parametre naredbe *gzip*.

Tablica 6.5	Nastavak	
Parametar	Ime	Opis
-S suffix	suffix . <i>suf</i>	Uporaba suffix. suf na komprimiranim datotekama.
-t	test	Provjera integriteta komprimirane datoteke.
- V	verbose	Promjena načina rada u verbose.
- V	version	Broj inačice.
- 1	fast	Ubrzano komprimiranje.
- 9	best	Bolje komprimiranje – manje datoteke.
file		Određuje ime datoteke (datoteka) za komprimiranje/dekomprimiranje, ako se ne zada, koristi se standardni unos.

Za datoteke koje su komprimirane pomoću naredbe *compress* trebate upotrijebiti naredbu *uncom*press. Primjerice, naredba

uncompress for.tar.Z

dekomprimira datoteku foo.tar.gz i zamjenjuje je s tar arhivom pod imenom foo.tar.

Nakon što dekomprimirate datoteku, trebate datoteku *tar* ekspandirati u stablo direktorija. Trebate postaviti izvorne datoteke svakog pojedinog softvera u zaseban direktorij. Prije nego dekomprimirate datoteku *tar*, trebali biste provjeriti njezin ispis i vidjeti da li je stvorena tako da direktorij čini prvi unos. Upotrijebite naredbu

tar -tvf tarfile-name | more

da biste vidjeli je li direktorij prvi unos u datoteci *tar*. Ako je to točno, datoteka *tar* prilikom dekomprimiranja stvara direktorij. Ako nema direktorija na vrhu datoteke *tar*, sve datoteke koje se nalaze na vrhu se instaliraju u trenutni direktorij. U tom slučaju trebate načiniti direktorij i premjestiti datoteku *tar* u njega, a zatim je dekomprimirati.

**BILJEŠKA** Uvijek provjerite ima li na vrhu datoteke nekakav direktorij. Mogao bi nastati veliki nered ako dekomprimirate datoteku *tar* u trenutni direktorij umjesto u odgovarajući poddirektorij (radi se o stotinama datoteka).

Nakon što ste odredili mjesto na koje ćete ekspandirati datoteku *tar*, možete to i učiniti pomoću naredbe:

tar -xvf tarfile-name

Idući korak ovisi o tome kako je napisan softverski paket koji instalirate. Najprije trebate potražiti datoteku čije je ime nalik ovome: README.1ST.

U ishodišnom direktoriju bi se trebalo nalaziti nekoliko datoteka s uputama za instaliranje programa. **BILJEŠKA** U većini inačica Linuxa možete odjednom dekomprimirati i ekspandirati datoteku *tar*. Jednostavno dodajte parametar *z* u naredbu tar (primjerice *tar –zxvf.tar.gz*).

Obično tijekom instalacije treba preurediti datoteku Makefile i upisati ciljni direktorij u koji softver postavlja svoje binarne datoteke. Potom trebate pokrenuti naredbu *make*, a zatim *make install*.

Naredba *make* varira ovisno o softverskom paketu koji instalirate. Kod nekih ćete paketa možda trebati odgovoriti na pitanja koja se odnose na podešavanje sustava, no prije toga obavezno pročitajte dokumentaciju koju ste dobili s paketom.

# Pregled dozvola za pristup datotekama

Podešavanje dozvola za pristup softverskom paketu obično se automatski obavlja tijekom instalacije. Skripta s instalacijskim naredbama, koja se isporučuje s programom, obično svaku datoteku instalira s imenom vlasnika i dozvolama pristupa. Jedino ako nastanu problemi, te ako korisnik koji bi trebao imati pristup datoteci ne može to učiniti, trebate potražiti direktorij u koji je program kopiran i provjeriti dozvole pristupa.

Obično svi korisnici imaju dozvolu pristupa izvršnim datotekama koje služe za pokretanje programa, te ih mogu pokrenuti, a samo super korisnik ih može obrisati ili zamijeniti drugom datotekom. Programi su obično instalirani u direktoriju koji ima dozvolu čitanja i izvođenja programa, ali ne i dozvolu pisanja.

**Vidi** "Dozvole pristupa datotekama", **310. stranica.** 

# Rješavanje problema

Dobro napisana aplikacija se instalira na vaše računalo uz minimalne zahtjeve za unosom informacija, jer se automatski postavljaju podešenja tako da je sve što trebate učiniti provjera rada programa i informiranje drugih korisnika (obično putem e-pošte) da ste instalirali određeni program. Ponekad, naravno, krenu i problemi. Ako se, iz bilo kojih razloga, program nije do kraja učitao ili ako nakon instalacije program loše radi, tada sami morate ustanoviti o čemu se radi i riješiti problem.

Ako program nije u potpunosti instaliran, tada trebate pročitati dokumentaciju iz datoteke README koju ste dobili uz program, te potražiti popis iznimaka ili problema i njihovih rješenja. No, nitko ne očekuje da postanete stručnjak za Linux softverske pakete. U tom vam je slučaju potrebna nečija pomoć.

Ako problem ne možete riješiti pomoću dokumenata koje ste dobili uz paket, trebali biste potražiti odgovarajuću novinsku (Usenet) grupu da biste vidjeli vodi li se rasprava o paketu s kojim imate problema. Mnoge probleme ćete riješiti ako postavite pravo pitanje u određenoj Linux grupi na Usenetu. Ako i na Internetu ne možete naći pomoć, pokušajte kontaktirati s programerom koji je izradio program (obično putem e-pošte). Upamtite da je Linux besplatan, kao i većina softverskih paketa za Linux. Zato ne očekujte opsežne knjige s uputama niti 24-satnu tehničku potporu. No, da niste skloni avanturama, ne biste niti koristili Linux, zar ne?

# Uklanjanje aplikacija

Ako je neki program zamijenjen boljim softverskim paketom ili ako ga duže vrijeme nitko od korisnika ne koristi, tada je odlična ideja da taj program uklonite. Prostor na disku je uvijek nedovoljno velik, tako da vam doista nije potreban stari, beskorisni program koji onemogućava instalaciju novih aplikacija.

Kao i instalacija, tako je i uklanjanje programa u Linuxu nešto složenije od iste radnje u jednokorisničkim operativnim sustavima. Ponekad nije dovoljno samo obrisati datoteke koje čine softverski paket i zatim ukloniti njegov direktorij. Upravljački programi i druge softverske veze moraju biti prekinute kako bi se izbjegli eventualni problemi. Vodite li bilješke i pratite li poruke tijekom instalacije, moći ćete i sami uočiti što se sve promijenilo na disku prilikom instalacije softvera. Potom ćete moći odrediti one datoteke koje trebate obrisati, a koje zamijeniti kako biste uspješno uklonili nepotrebni softverski paket.

# Nadogradite svoju jezgru

Usporedo s nadogradnjama drugog softvera povremeno se pojavljuju nove inačice jezgre, u kojima su otklonjene greške iz prethodnih inačica ili dodane nove mogućnosti. Osim toga, možda ćete se odlučiti za nadogradnju jezgre jer je trebate drukčije podesiti ili dodati nove upravljačke programe nekih uređaja. U svakom slučaju postupak je uvijek isti. Trebali biste provjeriti imate li sigurnosnu kopiju sustavskog softvera i disketu za pokretanje (*boot*) Linuxa prije početka nadogradnje. U slučaju da se vaš sustav tijekom nadogradnje ošteti moći ćete ga popraviti i vratiti u normalno stanje. Cjelovit opis nadogradnje Linux jezgre potražite u 13. poglavlju, "Podešavanje jezgre Linuxa".

Vidi "Stvaranje nove jezgre", 254. stranica.

Postupak za nadogradnju jezgre sustava je detaljno opisan u dokumentu Kernel HOWTO, koji je postavljen u Linux novinskim grupama na Internetu, kao i na raznim FTP lokacijama (među kojima je i lokacija **sunsite.unc.edu**). Provjerite imate li kopiju tog dokumenta i pročitajte ga pažljivo prije nego započnete nadogradnju jezgre.

Prvi korak u nadogradnji jezgre jest pribavljanje novih izvora podataka koji se nalaze na raznim FTP lokacijama. Kada imate izvorni kôd, trebate zamijeniti postojeće izvorne kodove i to tako da premjestite direktorij /usr/src/linux u, primjerice /usr/src/linux.old. Raspakirajte izvorni program u direktorij /usr/src i stvorite potrebne poddirektorije. Premjestite se u direktorij linux i potražite datoteke README. Mnogo se toga može promijeniti od trenutka kada su programi objavljeni, stoga čitajte dokumentaciju.

Od ovog trenutka postupak može donekle varirati. Obično ćete unijeti naredbu *make config* i pokrenuti skriptu s naredbama za podešavanje te odgovoriti na postavljena pitanja o vašem sustavu. Nakon uspješnog završetka podešavanja trebate upisati naredbu nalik *make dep* te provjeriti je li nova jezgra našla sve potrebne datoteke.

Nakon što obavite i tu provjeru obično trebate unijeti naredbu *make clean* i obrisati sve stare datoteke koje se nalaze u direktoriju s izvornim kodom jezgre. Ako i ovo obavite uspješno, tada upišite naredbu *make* i kompilirajte novu jezgru, a potom je možete instalirati pomoću programa za pokretanje sustava LILO.
No, i prije ovoga koraka pročitajte dokument Kernel HOWTO jer ćete tako saznati više detalja vezanih uz podešavanje jezgre (što vam može uštedjeti sate nerviranja).

# Odavde...

U idućim poglavljima možete naći dodatne informacije o instalaciji i nadogradnji softvera:

- 3. poglavlje, "Instaliranje Red Hata" sastoji se od detaljnih uputa za instaliranje i podešavanje Red Hat distribucije Linuxa.
- 4. poglavlje, "Instaliranje OpenLinuxa Lite" daje vam detaljne upute kako instalirati i podesiti OpenLinux distribuciju Linuxa.
- 5. poglavlje, "Pokretanje Linux programa" nudi vam osnovne upute za podešavanje sustava Linux.
- 11. poglavlje, "Sigurnosna pohrana podataka" objašnjava postupak sigurnosnog spremanja sustava.
- 13. poglavlje, "Podešavanje Linux jezgre" sadrži sve detalje koji su potrebni za instalaciju nove jezgre u Linuxu.

# 

# Smisao administratora sustava

(Jack Tackett)

# U ovom poglavlju

Važnost pravilnog upravljanja sustavom 164 Višekorisnički koncepti 165 Sustavi s centralnim upravljanjem 166 Sustavi s distribuiranim upravljanjem 168 Model klijent/poslužitelj 171 Upravljanje sustavom u mrežnom okružju 172 Uloga mrežnog administratora 172 Svaki bi operativni sustav Linux trebao imati barem jednu osobu koja je zadužena za upravljanje i nadogradnju sustava, tj. trebao bi imati administratora sustava. Ta je osoba odgovorna za praćenje rada sustava. Osim toga, administrator sustava treba znati koga može nazvati ako probleme ne može (ili ne zna) riješiti, te koje sve hardverske i softverske mogućnosti može ponuditi postojećim i novim korisnicima.

Sustav Linux zahtijeva početno podešavanje, ali i neprekidnu pozornost kako bi se osiguralo da on radi te da mogu koristiti svi korisnici. Administrator sustava je osoba koja je odgovorna za "praćenje i udovoljavanje svim potrebama Linux sustava". Prema tome, ta je osoba odgovorna za mnogo različitih zadaća.

U ovom će poglavlju biti riječi o najvažnijim zadaćama administratora sustava koje on obavlja u mrežnom okružju. Pretpostavka je da ste ovu knjigu kupili kako biste naučili sve o instaliranju i korištenju Linuxa. No, sigurno ćete se naći i u ulozi administratora sustava, i to na samom početku rada. Neke od tema koje se spominju u ovom poglavlju odnose se na posao administratora u velikim organizacijama. No čak i ako ste pojedinačni (neumreženi) korisnik koji se Linuxom bavite u slobodno vrijeme, trebali biste se upoznati s ovim temama kako ne biste trebali čitati debela izdanja namijenjene administratorima sustava.

U mnogim se slučajevima Linux računalo nalazi u mreži s drugim računalima koja ne rade pod Linuxom. Ta druga računala možda koriste druge vrste UNIX-a ili neki potpuno drukčiji operativni sustav. S obzirom da je Linux nastao prema UNIX-u, mnogi se podaci iz ovog poglavlja odnose na Linux i na UNIX.

# Važnost pravilnog upravljanja sustavom

Svi UNIX sustavi se međusobno razlikuju ali su jedinstveni po tome da se njima treba upravljati. Po tome se niti Linux ne razlikuje. Dužnosti administratora variraju ovisno o broju korisnika, vrsti perifernih komponenti (pisači, pogoni za trake i slično), vezama u mreži i potrebnoj razini sigurnosti.

Administrator sustava, zajedno s drugim osobljem za održavanje, mora osigurati sigurno i efikasno sučelje za korisnike sustava. Administrator ima mogućnost i obavezu uspostave i održavanja sustava. U višekorisničkom radu postoji nekoliko razina pristupa i prioriteta. Tijekom rada administrator stječe potrebno iskustvo da bi održavao sustav.

Administratorova nadležnost varira od sustava do sustava. U velikim sustavima te su zadaće podijeljene na nekoliko ljudi. Naprotiv, neki mali sustavi ne zahtijevaju administratora s punim radnim vremenom, već ih može održavati neki od korisnika. Ako radite u mrežnom okružju, onda administrator može održavati vaše računalo putem mreže.

Svaki Linux sustav ima jednog korisnika koji može obavljati gotovo sve zadaće. Taj se korisnik naziva *superkorisnik*, a ima i posebno korisničko ime *root*. Polazni direktorij tog korisnika je obično direktorij / (*root*) ili /home/root/.

Administrator sustava se prijavljuje kao superkorisnik kako bi obavio zadaće koje zahtijevaju posebne dozvole. Kod normalnog rada taj se isti korisnik prijavljuje kao obični korisnik. Korisničko se ime superkorisnika, *root*, koristi samo za ograničene namjene. Broj korisnika koji se mogu prijavljivati kao superkorisnici treba svesti na minimum (dvojica, a najviše trojica). Ako se bilo tko prijavi kao superkorisnik, tada ta osoba ima sve potrebne ovlasti za rad i mijenjanje sustava. S tom dozvolom pristupa, superkorisnik može mijenjati atribute datoteke, zaustaviti sustav, pokrenuti sustav, načiniti sigurnosnu kopiju podata-ka i obaviti još neke zadaće.

Administrator mora poznavati mnoge tehničke značajke računalnog sustava, potrebe korisnika i osnovnu namjenu sustava. Svako računalo predstavlja ograničeni resurs i stoga se uporaba resursa na neki način treba limitirati. Baš zbog toga, administrator treba odrediti način korištenja resursa i sustav tehnički prilagoditi tim zahtjevima. Sve spomenute zadaće traže odgovornu, vještu i taktičnu osobu.

Precizan opis posla administratora sustava ovisi o lokalnoj organizaciji. I sami ćete se naći u ulozi administratora svoga sustava, ili ćete barem trebati obaviti neke od njegovih zadaća, od određivanja prioriteta do instaliranja softvera. No, većina administratora treba obavljati neke od ovih zadaća:

- Upravljanje korisnicima. Dodavanje novih korisnika, brisanje korisnika i mijenjanje mogućnosti i prava postojećih korisnika.
- Podešavanje uređaja. Potrebno je omogućiti pristup i dijeljenje uređaja kao što su pisači, terminali, modemi i pogoni za trake.
- Sigurnosna pohrana podataka. U slučaju da se sistemske datoteke izgube ili oštete potrebno je imati njihovu kopiju.
- Isključivanje sustava. Sustav se isključuje prema određenom rasporedu kako bi se izbjegle nedosljednosti u sustavu datoteka.
- Obuka korisnika. Održavanjem obuke korisnika postiže se njihovo efikasno korištenje sustava.
- Sigurnost sustava. Drži korisnike na "pristojnom rastojanju" kako bi se spriječile slučajne ili namjerne štetne radnje.
- Zapis promjena sustava. U posebnoj knjizi treba voditi sve značajnije radnje koje se odnose na sustav.
- Savjetovanje korisnika. Pruža pomoć "običnim" korisnicima u uporabi sustava.

# Višekorisnički koncepti

Višekorisnički sustav koristi dva glavna koncepta: višezadaćne i višekorisničke usluge. Linux ima mogućnost višezadaćnog rada tako da možete čitati e-poštu dok kompilirate program.

Svaka zadaća, bez obzira radi li se o jednostavnoj naredbi koja je upisana u redak za naredbe ili o složenom programu, pokreće jedan ili više procesa. Procesi predstavljaju sve što se pokreće na Linux računalu. S obzirom da Linux ima mogućnost istovremenog pokretanja procesa, znači da se radi o višezadaćnom operativnom sustavu. Postoji više načina kako se možete spojiti na računalo koje radi pod UNIX-om (koje se naziva *poslužitelj*). Možete koristiti terminal ili računalo, odnosno možete se nalaziti u blizini poslužitelja (fizički) ili biti spojeni putem kabla, a možete raditi na drugoj strani planeta i biti spojeni putem brzih veza ili običnih telefonskih linija. Ovisno o tome koristite li terminal ili računalo, te načina na koji ste spojeni na poslužitelj, mijenjaju se i osobine računalne mreže, odnosno razlikuje se je li mreža centralizirana ili distribuirana.

Operativni sustav za jednog korisnika, kao što je DOS, prilagođen je da ga može koristiti samo jedna osoba, jer se svi poslovi odvijaju na jednom računalu koje ima pristup svim resursima, kao što su pisači, spremnici podataka i procesori.

Višekorisnički sustavi koriste centralizirane i distribuirane modele koji omogućuju ugodan rad većeg broja korisnika.

*Sustavi s centralnim upravljanjem* omogućuju mnogim korisnicima (veliki sustavi mogu imati stotine korisnika) pristup resursima na jednom računalu (diskovi, pisač, memorija i procesor) na kojem se obavljaju sve zadaće.

*Sustavi s distribuiranim upravljanjem* omogućuju da se, primjerice, proračuni obavljaju na radnoj stanici jednog korisnika, a da se središnji procesor koristi za distribuciju programa i podataka. Pisači i spremnici podataka (diskovi) mogu se nalaziti na korisničkim radnim stanicama ili na glavnom poslužitelju.

# Sustavi s centralnim upravljanjem

S razvojem tehnologije 50-tih i 60-tih godina operativni sustavi su počeli omogućavati višekorisnički rad putem kojeg su korisnici mogli dijeliti zajedničke resurse, radeći s odvojenih terminala. Dva korisnika su istovremeno mogli pokrenuti odvojeni skup naredbi koristeći isti procesor, spremnik za podatke (disk) i izlaz (pisač).

Pojavom mreže koja se ostvarivala putem telefonskih veza računala počinju upotrebljavati telefonske resurse a računalne se mreže počinju širiti i izvan određenih zemljopisnih granica. U tom modelu svaki procesor je koristio komunikacijske veze za uspostavu kontakta s udaljenim terminalima. Stvorila se potreba za boljim načinom komuniciranja između računala i terminala. To je razlog pojave sustava s centralnim upravljanjem i glavnih procesora koji obavljaju komunikacijske zadaće.

Od vremena kada su osobna računala postala jeftina, moćna i svugdje prisutna, većina UNIX sustava je počela koristiti model s centralnim upravljanjem. Kod tog modela glavno računalo upravlja svim procesima, a korisnici se spajaju na glavno računalo i dijele njegove resurse. Ovaj se model sve manje i manje koristi, iako je još uvijek koristan ako su korisnici međusobno dislocirani.

Primjerice, vaša banka ima jedan glavni računalni centar, a sve poslovnice mogu pristupiti bazi podataka putem računala koje se u njima nalaze. Na svakom stolu se nalazi po jedan terminal koji se sastoji od tipkovnice, monitora i izravne veze s glavnim računalom. Pomoću terminala korisnik može pristupiti središnjim resursima: upravljanje, ispis i pohrana podataka (vidi sliku 7.1). Sustav sa centralnim upravljanjem sastoji se od

(sve što se Grana A Grana B upiše na terminalu prosljeđuje Terminali model centraliziranog se na glavno računalo u uredu) Pisač Pohrana podatak Veza (dodijeljena telef. linija) Glavno računalo Glavni Grana C Grana D proceso fcp

mnogih elemenata: poslužitelj, glavni procesor, terminali, modemi i adapteri s više priključaka.

Kada korisnik pristupi podacima, zahtjev se prosljeđuje na glavno računalo u banci. Dobiveni se rezultati zatim šalju na terminal u poslovnici. Svi se podaci obrađuju i spremaju na glavnom računalu.

## Elementi sustava s centralnim upravljanjem

Slika 7.1

računala.

Ova slika prikazuje

sustava umreženih

Da bi sustav s centralnim upravljanjem radio, potrebni su vam mnogi elementi među kojima se nalaze poslužitelj, glavni procesor, terminali, modemi i adapteri s višestrukim priključcima.

Poslužitelį (server) se može definirati kao računalo podešeno za dijeljenje resursa (procesora, diskova, pisača i slično). Primjerice, u ulozi poslužitelja možete koristi IBM-kompatibilno osobno računalo, naravno ako imate dovoljno prostora na disku i dovoljno RAM-a.

Glavni procesor povezuje komunikacijske kanale i poslužitelj, a služi za upravljanje komunikacijom tako da poslužitelj može samo obrađivati podatke.

Postoje dvije vrste terminala koje se danas koriste u svijetu: glupi i pametni terminali. Tradicionalno se uz operativni sustav UNIX koriste glupi terminali koji se sastoje isključivo od tipkovnica i monitora. Najvažnija činjenica koja se odnosi na ovu vrstu terminala je to što oni nemaju nikakvu mogućnost upravljanja podacima. Komunikacijski priključak terminala je izravno ili putem modema spojen s poslužiteljem. Dok pritiskate tipke na tipkovnici takvog terminala, signali se prenose na poslužitelja gdje se obrađuju. Pametni terminali mogu obaviti minimalnu obradu podataka na lokalnoj razini. Umrežene registar blagajne i drugi slični uređaji su primjeri pametnih terminala. Kod lokalnih uređaja spremaju se pojedini signali s tipkovnice i zatim prenosi cjelovit zahtjev na glavno računalo (umjesto prenošenja signala svake pojedine tipke).

Da biste svoj terminal povezali s telefonskom linijom, morate upotrijebiti modem koji digitalne signale s terminala i računala pretvara u analogne signale (što je zahtjev kod telefonskih linija). Modemi se uvijek koriste u paru, od kojih jedan povezuje vaš terminal s telefonskom linijom, a drugi povezuje poslužitelj i telefonsku liniju. Da biste uspostavili vezu, birajte telefonski broj na terminalu. Kada modem na drugom kraju (spojenom na poslužitelj) odgovori na poziv, vaš će terminal moći komunicirati s poslužiteljem.

Da biste povećali broj raspoloživih priključaka na koje se korisnici mogu spojiti, morate instalirati *adapter s više priključaka*. Obično osobna računala imaju samo dva serijska priključka: COM1 i COM2. Ako namjeravate koristiti osobno računalo kao poslužitelj za više od dva korisnika, tada vam treba i više priključaka. U tom se slučaju adapter s više priključaka sastoji o kartice koju trebate instalirati u računalu i male kutije s osam ili više priključaka, te kabela koji povezuje kutiju i karticu. Uz adapter je nužan i softver kako bi omogućio da pravilno funkcioniraju "dodatni" serijski priključci.

# Sustavi s distribuiranim upravljanjem

Kod sustava s distribuiranim upravljanjem terminal je zamijenjen radnom stanicom, to jest računalom koje obično radi pod DOS-om ili UNIX-om. Programi mogu biti smješteni i pokrenuti s poslužitelja ili s radne stanice. Slično tomu, datoteke se mogu nalaziti na oba ova sustava. Stvarate li datoteku sa svoje radne stanice, pohranite je na poslužitelju kako bi joj i ostali korisnici mogli pristupiti. Datoteku možete ispisati na lokalnim pisačima koji su spojeni na radnoj stanici ili na pisačima spojenim na poslužitelj.

S obzirom da su radne stanice u čestoj uporabi, vaša banka zasigurno koristi sustav s distribuiranim upravljanjem umjesto centraliziranog upravljanja (koji je opisan u prošlom odlomku). Slika 7.2 prikazuje sustav s distribuiranim upravljanjem.

Slika 7.2 Ova slika prikazuje model distribuiranog upravljanja umreženim računalima.



# Elementi sustava s distribuiranim upravljanjem

U mrežama s distribuiranim upravljanjem se koriste poslužitelji, radne stanice, mrežne kartice, koncentratori, *repeateri*, mostovi i *gatewayi*. Uloga poslužitelja je distribuiranje datoteka i programa radnim stanicama, ispis na pisač i upravljanje vezom među radnim stanicama. Više od 90 posto zadaća se obavlja na razini radnih stanica, a 5 do 10 posto poslova se odnosi na administrativne zadatke u mreži.

Osim što osobno računalo možete koristiti kao poslužitelj, možete ga upotrijebiti i kao Linux radnu stanicu. Linux se može pokrenuti na vrlo slabim hardverskim konfiguracijama, čak i na računalima s procesorom 386SX i 4 MB RAM-a! Kako je većina današnjih računala znatno moćnija od te minimalne konfiguracije, ne biste trebali imati problema s procesorom. Količina prostora na tvrdom disku ovisi o softveru koji želite instalirati. Želite li raditi s CD-ROM-om, trebat će vam barem 5 MB prostora na tvrdom disku. Minimalna instalacija zauzima 10 do 20 MB prostora, a puna instalacija i više od 100 MB.

Općenito, resurse bi trebalo rasporediti na razini radnih stanica, gdje se i obavlja većina zadaća. Ovisno o ciljevima koje želite postići, trebat ćete instalirati i dodatne resurse. Primjera radi, programi za pisanje teksta zahtijevaju minimalne resurse (tvrdi disk, RAM, kvaliteta monitora) u usporedbi s programima za crtanje ili onima koji se izvođe u multimedijskom okružju. Za aplikacije koje uključuju i CAD, potreban vam je vrlo veliki disk (svakako veći od 1 GB), dosta RAM-a (16 MB, 32 MB ili 64 MB) te monitori i grafičke kartice velike razlučivosti (1280x1024 ili veće). Možda će vam trebati i pogon za trake kako biste uvijek imali sigurnosnu kopiju sustava, te CD-ROM za pokretanje većih aplikacija.

*Mrežna kartica* umeće se u utor na matičnoj ploči i predstavlja fizičku vezu između računala i mreže. Možete koristiti koaksijalno ili parično ožičenje.

*Koncentrator* služi kao točka spajanja mrežnih kabela kao što su 10BaseTEthernet, a može biti pasivan i aktivan. Pasivni koncentrator obično ima četiri konektora, a aktivni obično ima barem osam priključaka te pojačalo signala.

*Repeateri* pojačavaju signal u mreži kako kod umrežavanja ne biste bili ograničeni udaljenostima.

Mostom se povezuju dva slična tipa mreže.

*Usmjernici* se koriste u velikim, složenim mrežama koje sadrže mnoge staze za mrežne signale koji putuju prema istoj destinaciji. Usmjernik određuje koji je najefikasniji put i njime šalje signale.

*Gateway* koristite kada želite povezati različite tipove mreža koje upotrebljavaju razne protokole. *Gateway* obavlja potrebnu konverziju protokola tako da dvije mreže mogu međusobno komunicirati. Primjerice, ako želite mrežu sa SNA protokolom povezati s TCP/IP mrežom, onda vam je nužan *gateway*.

# Topologija mreže

Izraz *topologija* se odnosi na način povezivanja radnih stanica i poslužitelja u zajedničku mrežu. Imena se raznih topologija izvode iz oblika koji čine kabeli prilikom povezivanja terminala, radnih stanica i poslužitelja. Najčešće se koriste topologije zvijezde, prstena i sabirničke topologije. Ako se više različitih topologija koristi u istoj računalnoj mreži, onda se takova topologija naziva *hibridna topologija*.

**Topologija zvijezde** Kod ove topologije su sve radne stanice povezane sa središnjim poslužiteljem ili koncentratorom (*hubom*) (vidi sliku 7.3). U ovom modelu možete koristiti pasivne ili aktivne koncentratore.



Pasivni koncentrator predstavlja samo točku u kojoj se povezuju radne stanice, a aktivni koncentrator nudi i mogućnost pojačavanja signala. primjer mreže u kojoj se koristi ova topologija je AT&T's StarLan.

**Sabirnička topologija** U ovom modelu (vidi sliku 7.4) sve radne stanice i poslužitelji dijele istu sabirnicu i izravno su spojeni. Ova se tehnologija koristi u Ethernet mrežama.

**Topologija prstena** Računala spojena u ovom modelu mreže nalikuju velikom kotaču i nemaju koncentrator (vidi sliku 7.5). Poslužitelj je povezan s radnim stanicama putem sabirnica, s tim da je spojeno i zadnje računalo kako bi se zatvorio prsten. Ova vrsta mreža koristi uređaj nazvan *repeater*, odnosno jedinica za pristup više stanica (MAU). Primjer takve mreže je IBM Token-Ring Network.

**Hibridna topologija** 70-tih i 80-tih godina tvrtke s decentraliziranim načinom prodaje počele su uvoditi nešto drukčije topologije umreženih računala. Računovodstvo u tim tvrtkama je koristilo sabirničku vrstu mreže, odjeli prodaje prstenastu mrežu, a proizvodnja Ethernet

sabirnice. Uprava je u takvim tvrtkama obično koristila model s velikim glavnim računalom. Počele su se razvijati računalne mreže koje su objedinjavale više raznih tipova mreže: prstenaste, zvjezdaste i sabirničke. Na taj su način nastale hibridne mreže računala.



# Model klijent/poslužitelj

Rezultat razvoja distribuiranog upravljanja je razvoj modela klijent/poslužitelj. U tom modelu se danas Linux može koristiti kao klijent, poslužitelj ili oboje.

Da biste shvatili podešenje klijent/poslužitelj, pretpostavite da je nekoliko Linux radnih stanica (klijenata) sabirnički povezano s poslužiteljem (osobno računalo s puno prostora na disku, koje također koristi operativni sustav Linux). Poslužitelj ima direktorije za svakog klijenta i u tim direktorijima se nalaze datoteke koje se spremaju i pohranjuju putem sigurnosnih kopija (noćna pohrana). Poslužitelj sadrži i direktorije u kojima se nalaze

dijeljene datoteke. Na poslužitelj je spojen brzi laserski pisač kojemu svi mogu pristupati, kao i pogon za trake koji služi za sigurnosnu pohranu sadržaja diska. Također, nekolicina klijenata ima vlastite, nešto sporije laserske pisače koji su lokalno spojeni na radne stanice.

# Upravljanje sustavom u mrežnom okružju

UNIX mreža se obično sastoji od više računala, velikih i malih, izravno spojenih mrežnim kabelima ili putem telefonske mreže. Upravljanje mrežom je zadaća jedne ili više osoba koje se na jednoj od mrežnih lokacija.

Većina ljudi može savladati Linux i potom upravljati mrežom. Što se tiče produktivnosti, bilo bi zgodno odmah potražiti kvalificiranu osobu, no takvi su ljudi prava rijetkost (a i treba ih dobro platiti). Uz malo vježbe i upornosti, čak i ljudi s ograničenim znanjem o računalima mogu naučiti kako se upravlja UNIX/Linux računalima.

# Uloga mrežnog administratora

U slučaju da koristite mrežu u kojoj je spojeno više UNIX/Linux računala, trebala bi postojati i osoba kojoj je zadaća upravljanje mrežom. Potrebno je imati i određenog iskustva pri odlučivanju kako će se računala spojiti (LAN ili modemska mreža), koja je razina zaštite koja će se koristiti, kako se raspoređuju zajedničke periferne komponente (pisači, trake za pohranu podataka i slično). Administrator svakodnevno stvara popise korisničkih imena u sustavu, mrežnih adresa i općenito se brine da mreža pravilno funkcionira.

Tvrtke s računalnim mrežama u kojima se nalaze stotine računala trebaju zaposliti nekoliko administratora i neprekidno ih obučavati za pojedine zadaće. To može biti nužno ako korisnici, primjerice, često upotrebljavaju pisače. Upravljanje pisačima i ispis dokumenata može zahtijevati i dodano znanje o pojedinim pisačima i njihovim vezama s Linux računalima.

# Hardverski i softverski zahtjevi

Ako kao administrator sustava trebate odabrati mrežni softver i hardver koji ćete ugraditi na računala u svojoj mreži, onda svakako vodite brigu o narednim problemima, ali najvažnije je pronaći optimum između onoga što trebate i onoga što si možete priuštiti.

Ako se umrežena računala nalaze na jednom mjestu, u istoj zgradi, onda je lokalna mreža sasvim dovoljna za umrežavanje računala. Lokalna mreža je, osim toga, jeftina i radi na zadovoljavajućim brzinama. U svako računalo umetnite jednu mrežnu karticu (Ethernet) i instalirajte mrežni softver (TCP/IP protokol je sastavni dio Linux distribucija).

Za povezivanje na većim udaljenostima možete koristiti modeme (prijenos na manjim brzinama) i protokole PPP (Point-to-Point Protocol) ili SLIP (Serial Line Internet Protocol) kako biste ostvarili sinkronu TCP/IP vezu. Osim toga, koristite UUCP softver za e-poštu, novinske grupe i prijenos datoteka (pri čemu treba znati da softver UUCP ima ograničenja). Želite li prijenos informacija na većim brzinama, tada koristite ISDN ili posebne brze telefonske linije (zatražite ih od telefonske kompanije). Ne kupujte stari mrežni hardver. Iako mnoge mrežne komponente dolaze zajedno s pogonskim programima koji služe za njihovo pokretanje pod DOS-om, to nije slučaj s operativnim sustavom Linux. Linux računala imaju ugrađene mnoge mrežne pogonske programe. Tablica 7.1 prikazuje neke Ethernet kartice koje Linux podržava. Potražite dokument Ethernet HOWTO i ažurirajte taj popis (u dodatku A ćete naći informacije o dokumentima HOWTO).

Tablica 7.1 Neke Ethernet kartice koje mozete koristiti pod Linuxom	
Proizvođač	Kartice
3Com	3c503, 3c503/16, 3c509, 3c579
SMC (Western Digital)	WD8003, WD8013, SMC Elite, SMC Elite Plus, SMC Elite 16 ULTRA
Novell Ethernet	NE1000, NE2000, NE1500, NE2100
D-Link	DE-600, DE-650, DE-100, DE-200, DE-220-T
Hewlett-Packard	27245A, 27247B, 27252A, 27247A, J2405A
Digital	DE200, DE210, DE202, DE100, DEPCA (rev. E)
Allied Telesis	AT1500, AT1700
PureData	PDUC8028, PDI8023

- . .. • •• • • • •

Programi koji nisu sastavni dio mrežnih proizvoda mogu se koristiti u mrežnom okruženju. Primjerice, možete instalirati određeni program na Linux računalo i dozvoliti korisnicima s drugih računala da ga upotrebljavaju. Oni samo trebaju pokrenuti daljinske naredbe za izvođenje programa koje su ugrađene u UNIX. Program možete dijeliti i tako da ga postavite u zajednički sustav datoteka, a svaki ga korisnik potom može koristiti na svom lokalnom računalu.

# Redovite zadaće mrežnog administratora

Upravljanje se mrežom može promatrati na više načina. Većina mreža se obično neprestance razvija. U idealnom slučaju administratori sudjeluju u nabavljanju računala i softvera tako da su od samog početka uključeni u stvaranje mreže.

**Podešavanje sustava** Mrežni softver treba instalirati i prilagoditi za mrežni rad. Ako ste upotrijebili Ethernet kartice za umrežavanje računala, dobro je provjeriti međusobnu povezanost računala. Također trebate testirati i telefonske linije ako ih koristite u mreži. Isto tako, provjerite ožičenje i terminala. Obično bi instaliranje komponenti trebalo biti

po pravilu Plug and Play (umetni-i-radi), ali to gotovo nikada nije slučaj jer uvijek postoje problemi s vezama.

Prednost kupnje računala bez instaliranog operativnog sustava je ta što sami možete svojim potrebama prilagoditi sustave datoteka. U tom slučaju morate znati koji će se softver koristiti na računalu, broj korisnika koji će ga koristiti i učestalost uporabe računala.

SAVJET Odvojite nešto vremena (i novca) i podesite svoju mrežu. Odmah načinite sigurnosnu kopiju datoteka za podešenje sustava.

Tek kada sustav kompletno radi možete instalirati ostale aplikacije. Softver na Linux računalima je obično bitno složeniji od onoga koji se instalira na osobnim računalima, stoga odvojite dovoljno vremena za instalaciju, podešavanje i pokretanje programa jer čitav posao može trajati nekoliko sati, dana, pa i duže.

Sada možete dodavati korisnike u sustav. Dodajte korisnička imena nekolicine ključnih korisnika i dodijelite im početne lozinke , kao što je *temp01*. Na taj ćete način omogućiti početnu sigurnost sustava, te određene ljude uputiti u rad sustava.

Nakon instalacije računalo možete spojiti na mrežu. Provjerite možete li uspostaviti vezu između svih čvorova u mreži. Veze provjerite seljenjem velikih i malih datoteka s jednog na drugo računalo. Elektronička se pošta treba poslati s bilo kojeg računala. Sva računala moraju "prepoznati" novo računalo u mreži, što znači da računalo trebate dodati u bazu podataka koju koriste sva računala u mreži. Ako lokalno koristite DNS (Domain Name System) sustav, morate *host* ime dodati u bazu podataka.. Ako ne koristite DNS, dodajte ime u datotekama /etc/hosts koje se nalaze na svim računalima u mreži.

**Periferne komponente** Ispis datoteka predstavlja jedan od najvažnijih zadataka administratora sustava. Praćenje i upravljanje pisačima je bitan zadatak i administrator može izgubiti dosta vremena dok ga ne obavi. Potrebno je poznavati način ispisa (*spool* datoteke i slično), alate sučelja i osobine pisača.

Modemi predstavljaju najjeftiniji način povezivanja udaljenih računala. Modemi i PPP protokol omogućavaju manjem broju ljudi da održava veliki broj računala. Kao i kod pisača, tako se i kod modema javljaju problemi koji se teško otklanjaju. Uključite još jednu ili dvije osobe da vam pomognu pri instalaciji i provjeri modema.

**Praćenje sustava** Na kraju instalacije možete podesiti UNIX alate za praćenje rada sustava i osjetit ćete kako vaša mreža radi.

Praćenje rada računala u mreži je proces koji se neko vrijeme razvija, ali će se stabilizirati nakon što prestanete dodavati nova računala i periferne komponente. Dobar administrator sustava treba znati je li problem nastao zbog hardvera ili softvera.

**Nadogradnja softvera** Neke softverske pakete neprekidno treba nadograđivati. Ovo se posebice odnosi na Linux jer se većina softvera nalazi na Internetu i često se mijenja. Dobra je stvar što se na taj način otklanjaju greške u programu, a loša je ta što treba ažurirati svako računalo u mreži. Sa svakom nadogradnjom treba očekivati i nove izazove.

Savjet je da ne stavljate odmah najnovije inačice programa na sva računala, ali svakako te inačice provjerite na jednom od sigurnijih računala. Kada se uvjerite da je s novom inačicom sve u redu, nadogradite i programe na ostala računala. Dobar administrator bi trebao instalirati sve nove inačice programa s jednog računala. Možda vam se čini da je to nemoguće, ali ćete se uvjeriti da mnogi UNIX alati pomažu pri instalaciji nadogradnji i "zakrpa".

# Uvježbavanje administratora

Obuka administratora se u većini organizacija svodi na princip pokušaja i promašaja. Možda pojedina osoba i ima određeno predznanje iz nekih područja u računalstvu, ali se vrlo malo ulaže u obuku te osobe kako bi mogla biti administrator. Upravljanje sustavom zahtijeva znanje o sljedećem:

- Linux/UNIX dizajn i uporabu. Administrator mora razumjeti pojmove kao što su preusmjeravanje, posao u pozadini i slično.
- vi editor. Ovaj program se nalazi na gotovo svakom UNIX računalu unazad 10 godina (uključivo i Linux računala). Mnogi ga ljudi kritiziraju i zamjenjuju drugi programima za pisanje teksta, ali preporučamo da administrator sustava nauči raditi s vi editorom jer se on najčešće koristi u UNIX sustavima.
- Programiranje skripti. Mnogi ključni programi za upravljanje UNIX-om su napisani u obliku skripti te ih prema potrebi treba modificirati. Brojni alati koje smo spomenuli u ovom poglavlju zahtijevaju znanje o stvaranju skripti. Gotovo svaki korisnik ima svoju omiljenu ljusku.
- Ljuska *bash* (Bourne Again shell), predstavlja kloniranu ljusku Bourne i izvorno je podešena u Linuxu. U ovoj distribuciji Linuxa možete naći i Z i T ljuske. Osim toga, gotovo svi programi su napisani u ljusci Bourne. Trebali biste istraživati jezik za upravljanje Perl jer on sadrži veliki skup alata za upravljanje sustavom.
- Komunikacije. Da biste mrežu računala podesili tako da efikasno radi, potrebno je poznavati protokol TCP/IP, dok je za stvaranje anonimne veze na Internet potrebno poznavati protokol PPP. Idealno bi bilo te protokole i njihove opcije provjeriti prije nego se uspostavi "prava" mreža. No, ako to nije moguće, tada bi administrator sustava trebao pohađati određene tečajeve ili pročitati knjige u kojima su opisani navedeni protokoli. Ipak, upamtite da ćete trebati potrošiti dosta vremena na provjere.
- Pravila UNIX-a. Pravila koja se odnose na operativni sustav UNIX se spominju na svim tečajevima o UNIX-u. Primjerice, vjerojatno ste naučili da se binarne izvršne datoteke općenito spremaju u *bin* direktorije, kao što su /usr/bin, /bin i /usr/local/bin. Vlastite programe isto tako možete spremati u direktorij /usr/local/bin. Slično tome, *lib* direktoriji, kao što je /usr/lib, koriste se za baze podataka tako da vlastitu bazu možete pohraniti u direktorij /usr/local/lib. Budete li poštivali standarde operativnog sustava Linux/UNIX, uštedjet ćete vrijeme kada budete rješavali određene probleme.

Nekoliko renomiranih tvrtki održavaju tečajeve za upravljanje sustavima. No, takvi tečajevi se ne održavaju isključivo za Linux sustav. Nekoliko tvrtki koje prodaju Linux distribucije nude i tečajeve koji se organiziraju po pojedinim temama. Također biste trebali potražiti novinske grupe i provjeriti raspored na Usenetu (**comp.os.linux**).

Tečajeve je najbolje održavati u manjim grupama. Trebali biste proći određenu obuku i potom stečeno znanje primijeniti na vlastitoj mreži. Linux ima alate kojima možda nikada nećete u potpunosti ovladati, ali ćete barem znati gdje možete potražiti pomoć.

# Odavde..

O upravljanju sustavom možete naći više informacija u idućim poglavljima:

- 9. poglavlje, "Pokretanje i isključivanje sustava" detaljno opisuje različite radnje koje se događaju tijekom pokretanja ili isključivanja sustava.
- 10. poglavlje, "Upravljanje korisničkim računima" prikazuje kako se dodaju, brišu korisnički računi te kako se njima upravlja.
- 14. poglavlje, "Upravljanje sustavom datoteka" nudi prikaz stvaranja, nadograđivanja i uporabe sustava datoteka pod Linuxom.



# Uporaba vi editora

(Jack Tackett)

# U ovom poglavlju

Upoznavanje s vi editorom Korištenje vi editora Naredbe u vi editoru Podešavanje sučelja vi editora

# Upoznavanje s vi editorom

U prethodnim poglavljima ste naučili koje su prednosti ako u datoteku pohranite seriju naredbi ili programe ljuski. Vjerojatno ćete htjeti napisati određeni tekst, poruku za e-poštu, popis, poruku na memorandumu, izvješće i slično. To možete učiniti jedino uz uporabu programa za uređivanje teksta (editora). U operativnom sustavu Linux imate na raspolaganju nekoliko editora ili programa za pisanje teksta. Da biste niz naredbi spremili

u datoteku, potreban vam je editor koji datoteku s tekstom može spremiti u ASCII formatu. Linux dolazi sa standardnim editorom koji se naziva *vi* editor, koji možete koristiti za složene tehnike pisanja i uređivanja teksta.

*vi* editor je vrlo koristan administratorima sustava jer se može koristiti na svakoj UNIX platformi. Stoga, kada jednom savladate *vi* editor, moći ćete ga upotrijebiti na svakom UNIX sustavu. Program je koristan i zbog toga jer zauzima vrlo malo mjesta na disku i u memoriji, što znači da ga možete koristiti i kada drugi programi ne mogu raditi zbog hardverskih ili drugih sistemskih problema.

*vi* editor i *ex* editor koji se distribuiraju s Red Hat distribucijom su kratice za editor pod imenom *vim* (VI iMproved). Imena *vi* i *ex* se simbolično povezuju u *vim*, stoga kada budete pisali **vi**, pokrenut ćete program *vim*. Pročitajte datoteku /usr/share/vim/vim\_diff.txt i saznat ćete koje su razlike između *vim* i *vi* editora.

Operativni sustav Linux ima i druge programe za pisanje teksta: grafički editor koji se koristi pod sustavom XFree86 te dva standardna editora (nisu grafički) *ed* i *ex*. Oba ova editora su linijski editori, to jest u određenom trenutku možete raditi samo s jednim retkom. Još jedan editor, nazvan *emacs*, koristi se u mnogim Linux distribucijama. Editori *vi* i *emacs* predstavljaju editore za čitav zaslon, što znači da na čitavom zaslonu vidite tekst koji možete mijenjati. U ovom poglavlju neće biti govora o programima *ed* i *ex*, jer je *vi* editor puno jednostavnije koristiti, a osim toga ga možete naći na svakom UNIX računalu.

Da biste upoznali *vi* editor (izgovara se *vi-aj*), potrebno je poznavati dio povijesti programa. Iako današnji sustavi (uključujući i Linux) koriste puno pristupačnije i obimnije editore, trebali biste naučiti kako se koristi *vi* editor jer svako UNIX računalo sadrži taj program. Ponekad je *vi* editor jedini program za uređivanje teksta koji u kritičnim trenutcima možete upotrijebiti. Stoga je potrebno naučiti neke njegove osnovne naredbe.

Operativni sustav UNIX je razvijen u okružju u kojem je korisnički terminal bio neka vrsta sporog terminala, a monitori s video prikazom se praktički nisu niti koristili. U tom sustavu je bilo normalno koristiti linijske editore kod kojih jedan korisnik može vidjeti i raditi samo na jednom retku teksta. Dva takva programa i danas postoje na UNIX računalima (editori *ed* i *ex*).

U ranoj fazi UNIX-a taj je operativni sustav bio dostupan samo na sveučilištima (i to besplatno). Studenti i profesori s nekoliko sveučilišta su unijeli broje promjene u sučelje UNIX-a. Nekoliko značajnih poboljšanja je stiglo i s kalifornijskog sveučilišta Berkeley,

a među njima je bio i editor za uređivanje teksta na čitavom zaslonu. Program je nazvan *vi* editor, što je kratica od *vizualan*. Došlo je vrijeme za početak rada na čitavom zaslonu i korisnici su mogli početi raditi na video terminalima.

**SAVJET** Nije nužno biti stručnjak da bi se koristio vi editor. Jednostavno upišite **man vi** i dobit ćete pomoć. Također pomoć možete dobiti ako pritisnete tipku <Esc> i zatim upišete naredbu **:help.** 

**BILJEŠKA** U ovom poglavlju nećete naći sve mogućnosti *vi* editora, jer bi to zahtijevalo puno prostora (u stvari čitavu bi knjigu trebalo posvetiti *vi* editoru). Umjesto toga ćete upoznati naredbe koje se koriste za osnove uređivanja. Želite li saznati nešto više o uporabi vi editora, potražite stranice *man* koje ste dobili zajedno s Linuxom.

# Što je vi?

S obzirom da se radi o standardnom UNIX okruženju, *vi* editor savladava i koristi više milijuna korisnika UNIX-a. Uočit ćete da se program lako pokreće te da ga možete koristiti i za jednostavne i za složene zadaće. Kao što vjerojatno i očekujete, ovaj se program koristi za unos, mijenjanje ili brisanje teksta, traženje i zamjenu određenog skupa znakova, te kopiranje, lijepljenje i rezanje dijelova datoteke. Također ćete vidjeti da program možete prilagoditi tako da odgovara vašim potrebama. Pokazivač možete postaviti na bilo koje mjesto na zaslonu i premještati ga dok budete uređivali datoteku. Iste metode možete upotrebljavati kod bilo koje datoteke s tekstom, bez obzira na njihov sadržaj.

*vi* editor ne predstavlja program za uređivanje teksta ili sustav za stolno izdavaštvo. U njemu nećete vidjeti izbornike, a čak nemate niti mogućnost za dobivanje pomoći.

BILJEŠKA Originalna inačica vi editora ne nudi mogućnost za dobivanje pomoći, ali novije inačice programa, kao što su vim, koja se koristi u Red Hat distribuciji, nude i pomoć.

Sustavi za uređivanje datoteka obično nude formatiranje te ispis datoteke na pisač, ali s vi editorom to nije slučaj. Navedene zadaće obavljaju druge naredbe u Linuxu, primjerice naredba lp može ispisati datoteku, a naredba nroff formatirati tekst. Neki programi, kao što su TeX (izgovara se tek) i LaTeX mogu izvršavati i umetnute naredbe kao što su naredbe za masna i podvučena slova.

vi editor radi u dva načina rada:

- Način rada za unos naredbi interpretira vaše tipke na tipkovnici kao naredbe koje program treba izvršavati. Neke od tih naredbi vam omogućavaju spremanje datoteke, izlazak iz programa, premještanje pokazivača na drugo mjesto, mijenjanje, brisanje i traženje teksta.
- U načinu rada za unos teksta, program tipke na vašoj tipkovnici shvaća kao tekst. U ovom se načinu rada program ponaša kao pisača mašina.

Tijekom rada možete prelaziti iz jednog u drugi način. Morate upamtiti u kojem se načinu rada trenutno nalazite, te znati na koji ga način možete promijeniti. Nekim korisnicima će se ovakav način rada učiniti kompliciranim, no u nastavku poglavlja ćete upoznati opciju showmode, koja prikazuje način rada u kojem se trenutno nalazite. Uz malo vježbe ćete uvidjeti da je *vi* editor izuzetno pogodan za uređivanje Linux datoteka, i to osobito datoteka s podešenjima i skripti s naredbama.

## Uređivanje teksta

Tekst uređujete ili tako što dodajete novi tekst ili tako da mijenjate postojeći. Kada dodajete novi tekst, onda ga spremate u datoteku s običnu Linux datoteku. Ako pak mijenjate postojeći tekst, tada koristite postojeće ime datoteke i otvarate njezinu kopiju. U oba slučaja se tekst nalazi u memoriji sustava, odnosno u *spremniku*. Uporabom spremnika se sprečava izravno mijenjanje sadržaja datoteke (sve dok ne odlučite spremiti sadržaj spremnika na tvrdi disk). Na taj način imate mogućnost biranja: želite li zanemariti ili prihvatiti unesene promjene.

Dok mijenjate tekst i upisujete dodatni tekst, promjene se događaju samo u spremniku, a ne i u datoteci na disku. Ako ste zadovoljni s unesenim promjenama, pokrenite naredbu za spremanje teksta na disk. tek nakon toga promjene postaju trajne. Promjene možete spremati na disk u proizvoljnim vremenskim intervalima (dobro je to činiti što češće kako biste izbjegli gubitak podataka u slučaju nestanka struje ili pada sustava). Nije potrebno napuštati program kada spremate promjene. U ovom poglavlju ćete naučiti nekoliko načina za napuštanje programa, od kojih neki istovremeno spremaju sadržaj spremnika na disk.

*vi* editor je interaktivan program jer i sami možete utjecati na njegov rad. Program s vama komunicira tako što se u statusnoj traci pojavljuju poruke, na zaslonu možete vidjeti poruke o greškama, a ponekad će sve nestati sa zaslona (karakteristično za Linux). Zadnji redak na zaslonu se naziva *statusna traka* i ona sadrži poruke programa. Promjene koje unesete ćete vidjeti na zaslonu.

Program koristite za mijenjanje, premještanje, brisanje, zamjenu i pretraživanje teksta. Ove operacije pokrećete iz načina rada za unos naredbi. U nekim slučajevima naredbu pokrećete pritiskom na određenu tipku koja odgovara prvom slovu naredbe. Primjerice, slovo *i* predstavlja naredbu insert (umetanje), a slovo *r* naredbu replace (zamjena).

Većina naredbi kao objekt djelovanja koristi jedan ili više redaka teksta. Reci su označeni brojevima od 1 (redak na vrhu datoteke) pa sve do zadnjeg retka u spremniku. Ako dodajete ili brišete retke, njihovi brojevi se automatski ažuriraju. Broj retka je, u osnovi, njegova adresa u spremniku. Raspon adresa čine dva broja međusobno odvojena zarezom. Želite li odabrati raspon redaka između trećeg i osmog retka, koristite brojeve 3 i 8.

Položaj pokazivača predstavlja mjesto u spremniku na kojem se trenutno nalazite i na kojem možete pokrenuti određenu naredbu. Neke od naredbi se odnose na znak na kojem se nalazi pokazivač. Sve dok ne premjestite pokazivač, promjene će se odnositi na isti znak. Naravno, da *vi* editor ima nekoliko naredbi za premještanje pokazivača unutar spremnika.

Sada vam je jasno da se radi o *full-screen* editoru. *vi* editor ima mogućnost premještanja i mijenjanja teksta na vašem *host* terminalu, kao i na drugim vrstama terminala. Koji terminal koristite, te koje su grafičke mogućnosti vašeg sustava, saznat ćete ako provjerite varijablu TERM. Linux koristi varijablu TERM za određivanje mogućnosti vašeg terminala (značajke su podvlačenje slova, brisanje zaslona, uporaba funkcijskih tipki i podešavanje boja).

### RJEŠAVANJE PROBLEMA

**Moj vi editor prikazuje čudne znakove na zaslonu**. Možda nije pravilno podešena varijabla TERM. Druga je mogućnost da je tipkovnica nepravilno podešena te da sprečava pisanje teksta preko postojećih znakova. Naredba \$TERM će vam prikazati trenutno podešenje vašeg terminala. Da biste provjerili vrijednost varijable TERM, upišite **echo \$TERM**. Koristite li terminal vt100, onda ćete na zaslonu ugledati sljedeće (naredbu unesite na terminalu, a ne u *vi* editoru):

#### vt100

Ukoliko pravilno podešeni terminal ne daje odziv, promijenite vrijednost varijable TERM pomoću ove naredbe (ako koristite ljusku bash):

```
TERM=vt100
export TERM
```

Ukoliko koristite ljusku C, upišite sljedeće (ne zaboravite razmake oko znaka =):

```
setenv TERM = vt100
export TERM
```

Ako se vaš terminal razlikuje od vt100, tada koristite istu metodu, ali s drugim terminalom.

Vidi "Podešenje okružja ljuske", 344. stranica.

**Pokrenuo sam** *vi* **editor, ali nisam dobio očekivani odziv**. Provjerite je li vaš terminal pravilno podešen. Vrsta terminala se razlikuje od njegova imena i on možda odgovara nekom od terminala u direktoriju /usr/lib/terminfo.

# Uporaba vi editora

Da biste pokrenuli *vi* editor, jednostavno u redak za unos naredbi upišite **vi**. Ako znate ime datoteke koju želite stvoriti ili uređivati, možete njezino ime upotrijebiti kao argument kod unosa naredbe vi. Primjerice, želite li stvoriti datoteku pod imenom *mojadat*, upišite *vi* mojadat.

Nakon što pokrenete *vi* editor, na zaslonu će se pojaviti oznake č na početku svakog retka, osim prvog. Ta oznaka predstavlja zastavicu praznog spremnika. Slijedi skraćeni pregled zaslona (radi uštede prostora prikazano je samo prvih pet redaka):

```
- ~ ~ ~ ~
```

Pokazivač se nalazi na početku prvog retka (predstavljan je crticom za podvlačenje slova). Vjerojatno ćete na svom zaslonu vidjeti 20 ili 22 istih oznaka 8č). Ako se to ne dogodi, provjerite vrijednost varijable TERM (kao što je opisano u prethodnom odlomku) i porazgovarajte sa svojim administratorom sustava. Ako vaš zaslon izgleda kako smo ga prethodno opisali, to znači da ste uspješno pokrenuli *vi* editor, te da se program nalazi u načinu rada za unos naredbi.

BILJEŠKA Za razliku od mnogih programa za uređivanje teksta, program vi najprije pokreće način rada za unos naredbi. Prije nego počnete unositi tekst, morate promijeniti način rada pomoću naredbi <a> ili <i>, koje će biti objašnjene u idućem odlomku. ■

## Dva načina rada

Kao što je prethodno spomenuto, *vi* editor radi u dva načina rada: za unos naredbi i unos teksta. U načinu rada za unos naredbi, *vi* editor pritisak na tipku predstavlja kao zadavanje naredbe koje možete upotrebljavati za spremanje datoteke, premještanje pokazivača ili mijenjanje, brisanje ili traženje teksta. Naredbu možete proslijediti u ljusku.

Ukoliko određeni znak unesete kao da se radi o naredbi, a taj znak ne predstavlja niti jednu naredbu, program će se javiti zvučnim signalom. Ne brinite. Zvučni signal vas samo upozorava da trebate provjeriti ono što radite i otkloniti moguću grešku.

Tekst možete unositi u načinu rada za unos teksta tako da znakove dodajete iza mjesta na kojem se nalazi pokazivač, ili ispred njega. Na početku retka ionako nema nikakve razlike. Da biste iz unosa naredbi prešli u unos teksta, pritisnite jednu od sljedećih tipki:

- <a> Dodavanje teksta iza pokazivača.
- <i>Umetanje teksta ispred pokazivača.

Ovaj način rada koristite isključivo za unos teksta. Većina programa za obradu teksta nakon pokretanja ostaju u ovom načinu rada, ali se *vi* editor u tome razlikuje. Koristite li neki od postojećih programa za uređivanje teksta, tada možete normalno unositi tekst, a naredbe trebate pokretati korištenjem funkcijskih tipki ili onih tipki koje se ne koriste kod unosa teksta. *vi* editor ne radi na takav način. Prije početka unošenja teksta morate pokrenuti naredbu <a> ili <i>, a na kraju unosa morate pritisnuti tipku <Esc> i vratiti se u način rada za zadavanje naredbi.

# Stvorite svoju prvu datoteku u vi editoru

Najbolji je način da naučite nešto o *vi* editoru da ga počnete koristiti. U ovom odlomku ćete vidjeti jedan primjer tipa korak-po-korak, koji vam pomaže u stvaranju prve datoteke. U svakom od navedenih koraka ćete vidjeti radnju koju trebate obaviti, a zatim i tipke koje trebate pritisnuti. U ovom primjeru nemojte očekivati potpunu preciznost. U ovom ćete primjeru naučiti koje se sve naredbe koriste u stvaranju datoteke, kako se prelazi iz unosa naredbi u unos teksta i kako se spremaju rezultati. Naiđete li na probleme, moći ćete izaći iz programa i ponovno ga pokrenuti koristeći naredbe <Esc> i potom :q!.

- 1. program *vi* pokrenite tako da upišete **vi**. Pojavit će se zaslon s oznakom č na početku svakog retka.
- 2. Pokrenite način rada za unos teksta kako biste mogli unositi znakove u prvi redak na zaslonu. Pritisnite tipku <a> i nemojte pritisnuti tipku <Return>. Nakon toga možete unositi znakove u prvi redak. Znak a ne biste trebali vidjeti na zaslonu.

3. Upišite sljedeći tekst:

Things to do today. Practice vi. Sort sales data and print the results.

Za ispravljanje grešaka u tekućem retku možete upotrijebiti i tipku <BackSpace>. U ovom primjeru ne trebate biti jako precizni jer se radi samo o vježbi. U nastavku ovog poglavlja ćete naučiti još neke metode za promjenu teksta.

- Pritisnite tipku <Esc> za prelazak u način rada za zadavanje naredbi. Ako tipku pritisnete dva puta, čut ćete zvučni signal koji vas upućuje da se već nalazite u načinu rada za unos naredbi.
- 5. Spremite sadržaj spremnika u datoteku pod imenom vipract.1 tako da upišete naredbu :w vipract.1. Naredba :w vipract.1 će se pojaviti u dnu zaslona, u statusnom retku, a ne u tekstu. Pomoću naredbe :w se sadržaj spremnika zapisuje u određenu datoteku. U ovom primjeru naredba sprema, odnosno zapisuje sadržaj spremnika u datoteku vipract.1.
- 6. Pogledajte izgled statusne trake:

"vipract.1" [New File] 3 lines, 78 characters

Potvrđeno je da ste stvorili novu datoteku koja se sastoji od 3 retka i 78 znakova. Izgled vašeg zaslona se može donekle razlikovati ako ste upisali nešto drukčiji tekst.

7. Naredbom : q napustite vi editor.

Nakon što upišete :**q** i dalje ćete se nalaziti u načinu rada za unos naredbi, sve dok ne pritisnete tipku <Return>. Nakon toga se program *vi* gasi, *vi* ćete biti vraćeni u ljusku za prijavu.

Iste upute (ili upute s malim promjenama) možete koristiti za svaku vrstu uređivanja teksta.

#### Što treba zapamtiti o vi editoru

- vi editor se pokreće u načinu rada za unos naredbi.
- Za prelazak iz načina rada za unos naredbi u način za unos teksta pritisnite tipku <a> (dodavanje teksta) ili tipku <i> (umetanje teksta).
- Tekst dodajete kada se nalazite u načinu rada za unos teksta.
- Naredbe zadajete kada ste u načinu rada za unos naredbi.
- Naredbe za spremanje datoteke i izlazak iz programa također zadajete u načinu rada za unos naredbi.
- Za prelazak iz unosa teksta u unos naredbi pritisnite tipku <Esc>.

# Pokretanje vi editora pomoću postojeće datoteke

Želite li uređivati ili pregledati datoteku koja već postoji u vašem direktoriju, upišite naredbu *vi, a* iza nje, u istom retku, upišite ime datoteke. Iskušajte navedenu metodu koristeći datoteku koju ste kreirali u prethodnom odlomku:

vi vipract.1

Na zaslonu ćete ugledati sljedeće (broj prikazanih redaka je nešto manji od onoga koji ćete ugledati na svom zaslonu):

```
Things to do today.
Practice vi.
Sort sales data and print the results.
"vipract.1" 3 lines, 78 characters
```

kao što je prethodno spomenuto, oznaka tilda (č) se nalazi na početku praznog retka. Pogledajte statusnu traku koja sadrži ime datoteke koju upravo uređujete, te broj redaka i znakova.

### **OTKLANJANJE PROBLEMA**

Upisao sam ime za koje sam siguran da postoji, ali vi editor je stvorio potpuno novu datoteku. Svatko od nas griješi kod pisanja i vjerojatno ste upisali ime datoteke koja ne postoji u direktoriju u kojem se trenutno nalazite. Pretpostavimo da ste upisali vi vipract.1, ali u direktoriju nema datoteke s tim imenom. Program će se pokrenuti, ali će stvoriti datoteku s novim imenom.

Pokušavam urediti datoteku, ali vi editor traži dozvolu pristupa i ponovno me vraća u prompt jezgre. Pokušavate uređivati datoteku koju smijete samo čitati. osim toga ne smijete uređivati direktorij, to jest upišete li vi ime direktorija, program će javiti da ste pokušali otvoriti direktorij i neće vam dozvoliti da ga uređujete. Ako u vi editor pokušate učitati izvršnu datoteku u binarnom kodu, na zaslonu ćete ugledati mnoštvo čudnih znakova koje nećete moći čitati niti uređivati. Program vi može uređivati samo datoteke čiji su podaci svrstani u retke.

Otvorio sam datoteku u vi editoru, ali se pojavila poruka da je redak predugačak. Pokušavate koristiti vi editor za datoteku koja se sastoji od jednog dugačkog skupa znakova. Možete modificirati datoteku, ali ćete na taj način vjerojatno oštetiti podatke u njoj.

Otvorio sam datoteku u vi editoru, ali sam ugledao čudne znakove na zaslonu. Možda vi editor koristite za pregled datoteka nastalih u drugom programu za obradu teksta.

U svakom od navedenih slučajeva najbolje je napustiti vi editor pritiskom na tipku < Esc>, a zatim upisati :q!. Pomoću naredbe :q! ćete napustiti vi editor bez spremanja promjena.

# Napuštanje vi editora

vi editor možete napustiti na više različitih načina. Tablica 8.1 sadrži popis naredbi koje možete upotrijebiti za izlazak iz vi editora.



Upamtite da se morate nalaziti u načinu rada za zadavanje naredbi kako biste izašli iz vi BILJEŠKA editora. za prelazak u taj način rada pritisnite tipku <Esc> (ako se već nalazite u tom načinu rada, onda ćete čuti zvučni signal nakon što pritisnete tipku <Esc>).

Tablica 8.1	Kako napustiti vi editor
Naredba	Opis
:q	Napuštanje kada nije bilo promjena, odnosno izlazak nakon što su promjene već pohranjene.
:q!	Izlazak bez spremanja promjena iz spremnika u datoteku na disku.
:wq, :xiliZ	Z Pohrana promjena iz spremnika u datoteku na disku i izlazak iz programa.

Kao što možete uočiti u tablici 8.1, nekoliko naredbi postiže isti učinak. Za vježbu upotrijebite datoteku vipract.1 koju ste prethodno stvorili. Želite li uređivati spomenutu datoteku, upišite *vi* vipract.1. Na zaslonu će se pojaviti sljedeći tekst:

Podvučeno slovo predstavlja položaj pokazivača. Kada prvi puta otvorite navedenu datoteku, pokazivač će se nalaziti na prvom znaku u datoteci (slovo T). S obzirom da niste unosili nikakve promjene od kada ste otvorili datoteku, možete vi editor napusiti pomoću naredbe :**q**. Vidjet ćete prompt ljuske. također možete upisati i naredbu :**wq!**"i prije nego

se pojavi prompt ljuske, na zaslonu ćete ugledati ovu poruku:

```
"vipract.1" 3 lines, 78 characters
```

Ta se poruka pojavljuje jer program najprije sprema sadržaj spremnika u datoteku na disku, a potom se gasi.

Ponovno otvorite istu datoteku (upišite vi vipract.1), i opet ćete ugledati isti zaslon:

```
Things to do today.

Practice vi.

Sort sales data and print the results.

~

~

~

"vipract.1" 3 lines, 78 characters
```

Iako se program pokreće u načinu rada za unos naredbi, za svaku sigurnost pritisnite tipku <Esc>. Nakon toga pritisnite razmaknicu onoliko puta koliko je potrebno da pokazivač premjestite na točku koja se nalazi na kraju prvog retka. Želite li taj znak zamijeniti uskličnikom, tada pritisnite tipku <r> (naredba replace) i upišite !. Prvi redak će nakon toga ovako izgledati:

Things to do today!

S obzirom da ste u spremnik unijeli promjene, program *vi* vam neće dopustiti da izađete dok te promjene ne pohranite ili izričito ne zadate naredbu za izlazak bez pohrane promjena. Pokušate li izaći tako da upišete naredbu :**q**, pojavit će se poruka koja vas upozorava da sadržaj spremnika niste spremili na tvrdi disk: No write since last change (:quit! overrides).

Ovaj fajl je skinut sa sajta www.tutoriali.org

Da biste zanemarili promjene koje ste pohranili u spremnik, upišite naredbu :q!, a za spremanje promjena pokrenite naredbu :wq ili jednu od naredbi ZZ i :x.

- **BILJEŠKA** Program vi ne stvara sigurnosne kopije datoteka. Nakon što pokrenete naredbu **:wq**, originalna datoteka će biti promijenjena i nećete je moći vratiti u prethodno stanje. Sami možete stvarati sigurnosne kopije datoteka.
  - Vidi "Sigurnosna pohrana podataka i obnova datoteka" 229, stranica.

#### **UPOZORENJE**

Naredbu :q! koristite vrlo obazrivo, jer kada ju unesete sve će promjene koje ste načinili biti izgubljene.

Umjesto da koristite naredbu :q!, sigurnije je datoteku pohraniti pod drugim imenom, o čemu će biti govora u kasnijem odlomku "Spremanje datoteka pod novim imenom".

## Poništavanje naredbe

U *vi* editoru možete poništiti prethodnu naredbu ili pohranu u spremnik, ukoliko promjene niste pohranili na disk. Poništavanje naredbi se obavlja u načinu rada za unos naredbi. Pretpostavimo da ste neoprezno obrisali jedan redak u tekstu, promijenili nešto što nije

trebalo ili nepravilno dodali određeni tekst. Pritisnite tipku <Esc> za prelazak u način rada za zadavanje naredbi, a potom pritisnite tipku <u>. Na taj ćete način biti u istoj situaciji kao što ste bili prije nego ste napunili spremnik.

Slijedi primjer poništavanja naredbe. Pokrenite *vi* editor i učitajte datoteku vipract.1 (upišite *vi* **vipract.1**) i na zaslonu ćete ugledati tekst nalik ovome:

```
Things to do today!

Practice vi.

Sort sales data and print the results.

~

~

~

"vipract.1" 3 lines, 78 characters
```

Da biste na kraju drugog retka dodali frazu "for 60 minutes", pritisnite tipku <Return> i premjestite pokazivač na početak drugog retka. Zatim pomoću razmaknice premjestite pokazivač na točku koja se nalazi na kraju drugog retka. Unesite frazu "for 60 minutes" tako da najprije pritisnete tipku <i> i potom upišete željeni tekst. Nakon toga pritisnite tipku <Esc> i vratite se u način rada za zadavanje naredbi. Zaslon bi trebao izgledati ovako:

```
Things to do today.
Practice vi for 60 minutes.
Sort sales data and print the results.
~
```

Vježbanje u vi editoru sat vremena? Možda je to dobra ideja, ali i ne mora biti. Za poništavanje prethodno unesene promjene najprije provjerite nalazite li se u načinu rada za unos naredbi (pritisnite tipku <Esc>) i odmah potom pritisnite tipku <u>. Drugi redak u datoteci će ponovno izgledati kao i prije:

Practice vi.

A možda je ipak dobra ideja vježbati sat vremena u *vi* editoru? Ponovno pritisnite tipku  $\leq u \geq (vec$ se ionako nalazite u načinu rada za unos naredbi) i ponovno će se pojaviti fraza "for 60 minutes". Želite li ili ne toliko dugo raditi u *vi* editoru? To morate sami odlučiti. Poništavanjem naredbi poništavate i promjene koje ste pomoću njih unijeli (ali i poništavate poništenje naredbi) i istu stvar možete ponavljati koliko god želite. Čak i ako ste spremnik odlučili ostaviti u originalnom obliku, program će shvatiti kao da se spremnik mijenjao, te ga morate napustiti pomoću naredbe :q! (zanemarenje promjena) ili pomoću naredbe :wq (spremanje promjena).

Ako želite pohraniti gore navedene promjene, spremite ih u drugu datoteku pomoću naredbe :w vipract.2.



Tipku <Backspace> možete upotrijebiti za ispravljanje pogrešaka u istom retku u kojem SAVJET se nalazi pokazivač. Nažalost, pomoću ove tipke ćete obrisati onoliko znakova koliko puta pritisnete tipku. Koristite li tipku s lijevom strelicom, onda nećete brisati znakove prilikom premještanja pokazivača. Tipke sa strelicama su objašnjene u nastavku poglavlja.

# Pisanje datoteka i spremanje sadržaja spremnika

Vidjeli ste kako možete sadržaj spremnika spremiti u datoteku i izaći iz vi editora. Ponekad ćete trebati pohraniti sadržaj spremnika u datoteku bez napuštanja vi editora. Ovo biste trebali činiti više puta tijekom uređivanja datoteke. Ukoliko sustav padne zbog nestanka napajanja ili neke druge greške, izgubit ćete sve promjene koje ste unijeli nakon zadnjeg spremanja. Da biste sadržaj spremnika pohranili u datoteku, pokrenite naredbu :w.

Prije nego pokrenete naredbu :w (write), pritisnite tipku <Esc> kako biste pokrenuli način rada za BILJEŠKA unos naredbi. Ako se otprije nalazite u tom načinu rada, čut ćete zvučni signal.

Postoje razni načini spremanja datoteke. Oblik naredbe koju ćete pokrenuti ovisi o određenom slučaju, a u nastavku su navedena tri najčešća slučaja. Tablica 8.2 sadrži opis naredbe write.

Tablica 8.2	Naredbe za spremanje datoteka
Naredba	Opis
:w	Zapisivanje sadržaja spremnika u datoteku koju trenutno uređujete
:w imedatoteke	Zapisivanje sadržaja spremnika u određenu datoteku
:w! imedatoteke	Zapisivanje sadržaja spremnika u određenu datoteku i brisanje njezina starog sadržaja.

**Spremanje nove datoteke** Ako pokrenete *vi* editor, a da u naredbi niste naveli ime datoteke, morat ćete ime datoteke odabrati prilikom spremanja datoteke na disk. U tom slučaju ćete trebati pokrenuti naredbu za spremanje koja ima ovakav oblik:

```
:w imedatoteke
```

Pomoću ove naredbe se zapisuje sadržaj spremnika u datoteku pod imenom *imedatoteke* Ako se sve oba*vi* kako treba, na zaslonu ćete vidjeti ime datoteke i broj redaka i znakova koji se u njoj nalaze. Odaberete li ime već postojeće datoteke, u statusnoj traci ćete ugledati sljedeću poruku:

File exists - use "w! filename" to overwrite.

Ovaj slučaj je opisan u odlomku "Spremanje preko postojeće datoteke".

**Spremanje u trenutno otvorenu datoteku** Možda ćete htjeti sadržaj spremnika pohraniti u datoteku koju trenutno uređujete. Primjerice, pokrenete li *vi* editor i u njemu otvorite jednu od postojećih datoteka, a potom unesete određene promjene koje želite pohraniti u izvornoj datoteci, tada trebate upisati **:w**.

Povremeno spremajte promjene koje unosite u datoteku. Svakih, primjerice 15 minuta, SAVJET koristite naredbu :w jer nikada ne znate kada može nestati napajanja.

Naredba :w sprema sadržaj spremnika u datoteku koja je trenutno otvorena (radna datoteka). Na statusnoj traci možete vidjeti ime datoteke i broj redaka i znakova koji su u njoj zapisani.

**Spremanje sadržaja pod novim imenom** Možda ćete trebati sadržaj spremnika pohraniti u novu datoteku, koja će imati različito ime od datoteke iz koje ste započeli uređivanje. Primjerice, ako pokrenete *vi* editor i otvorite datoteku vipract.1, a potom unesete neke

izmjene koje želite spremiti pod novim imenom (a da pri tome ne izgubite datoteku vipract.1), tada datoteku trebate spremiti pod novim imenom. Evo kao izgleda naredba za spremanje datoteke pod novim imenom:

:w imedatoteke2

Ovaj oblik naredbe je u osnovi jednak onome koji je opisan u odlomku "Spremanje nove datoteke". Sadržaj spremnika će biti spremljen u datoteku s imenom *imedatoteke2*. Ako se sve obavi kako treba, na zaslonu ćete vidjeti ime datoteke i broj redaka i znakova koji se u njoj nalaze. Odaberete li ime već postojeće datoteke, u statusnoj traci ćete ugledati sljedeću poruku:

File exists - use ! to overwrite.

U nastavku će biti objašnjeno kako možete prebrisati postojeću datoteku.

**Spremanje preko postojeće datoteke** Pokušate li spremiti sadržaj spremnika u postojeću datoteku, znajte da morate izričito naglasiti da želite prebrisati sadržaj postojeće datoteke. Ako prilikom spremanja sadržaja spremnika navedete ime postojeće datoteke, *vi* editor će prikazati sljedeću poruku na zaslonu:

File exists - use ! to overwrite.

Ukoliko doista želite spremiti sadržaj spremnika i prebrisati sadržaj postojeće datoteke, onda trebate upotrijebiti ovaj oblik naredbe write:

:w! postojećadatoteka

Gdje oznaka *postojećadatoteka* predstavlja ime datoteke čiji sadržaj želite prebrisati. Budite pažljivi jer nakon što jednom obrišete sadržaj datoteke, nećete je više moći vratiti u izvorni oblik.

# Pomicanje pokazivača

Kada uređujete tekst, pokazivač trebate postaviti na mjesto gdje želite dodati tekst, obrisati riječ, ispraviti grešku, promijeniti znak ili nastaviti pisati tekst. Naredbe koje se koriste za postavljanje pokazivača na određeno mjesto se zadaju u načinu rada za unos naredbi.

**Tipke sa strelicama** U mnogim, ali ne u svim sustavima možete koristiti tipke sa strelicama za postavljanje pokazivača na pravo mjesto. Lako je provjeriti rade li te tipke: Pokrenite *vi* editor i otvorite neku od postojećih datoteka, a zatim pritisnite neku od tipki sa strelicama. Također ćete moći koristiti tipke <Page Up> i <Page Down>, ukoliko ste pravilno podesili vrstu terminala (varijabla TERMCAP).

Da biste stvorili novu datoteku pod imenom vipract.3 koja sadrži popis datoteka i direktorija koji se nalaze u direktoriju usr, pokrenite sljedeću naredbu:

ls /usr > vipract.3

Potom ovu datoteku možete koristiti za provjeru rada naredbi za postavljanje pokazivača na određeno mjesto.

Nakon što ste stvorili datoteku, pokrenite *vi* editor i učitajte datoteku vipract.3 (upišite *vi* **vipract.3**), te pokušajte koristiti tipke sa strelicama i tipke <page Up> i <Page Down>.

Može se dogoditi da, iako ove tipke rade, one istovremeno upisuju čudne znakove u vašu datoteku. Da biste provjerili upisuju li ove tipke znakove umjesto da služe za premještanje pokazivača, pritisnite tipku <Esc> za prelazak u način rada za unos naredbi i potom upišite :q. Ako vam program *vi* dozvoli da izađete iz njega, a pri tome vas ne upozorava da ste promijenili sadržaj datoteke, onda je sve u redu.

SAVJET U vi editoru možete ukloniti nepotrebne znakove sa zaslona ako pritisnete kombinaciju tipki <Ctrl-l>.

**Druge tipke za premještanje pokazivača** Pokazivač u *vi* editoru možete premjestiti bez uporabe tipke sa strelicama. Trebate se upoznati s tim načinom rada u slučaju da ne možete ili ne želite koristiti tipke sa strelicama. U ovom odlomku ćete također vidjeti neke od načina za postavljanje pokazivača koji mogu biti efikasniji od uporabe tipki sa strelicama.

Dok je *vi* editor tek nastajao, mnoge tipkovnice nisu sadržavale tipke sa strelicama, te su se druge tipke koristile (i još se uvijek koriste) za postavljanje pokazivača na željeno mjesto. *Vi* editor koristi tipke <h>, <j>, <k> i <l> za premještanje pokazivača, jer se one nalaze na tipkovnici jedna do druge. Potrebno je malo vježbe da biste ih mogli spontano koristiti, no neki od iskusnijih korisnika*vi*editora radije upotrebljavaju ove tipke nego tipke sa strelicama.

U nastavku su date još neke tipke koje služe za premještanje pokazivača:

- Pritisnite razmaknicu ili tipku <l>da biste pokazivač pomaknuli za jedno mjesto u desno.
- Pritisnite tipku <Return> ili <+> za premještanje pokazivača na početak idućeg retka (uočite da tipka <j> služi za premještanje pokazivača za jedan redak niže, ali ne i za vraćanje na početak retka).
- Pritisnite tipku <-> za povratak na početak prethodnog retka (uočite da tipka <k> služi za premještanje pokazivača za jedan redak više, ali ne i za vraćanje na početak retka).

- Pritisnite tipku <h> i pokazivač pomaknite jedan znak ulijevo.
- Pritisnite tipku <0> i vratite se na početak retka.
- Pritisnite tipku <\$> za premještanje pokazivača na kraj retka.

Neke naredbe omogućuju postavljanje pokazivača na dijelove riječi u određenom r etku. *Riječ* predstavlja skup znakova koji su od drugih znakova odvojeni razmakom ili interpunkcijskim oznakama kao što su:

.?,-

Radi se o sljedećim naredbama:

Tipka	Opis
<w></w>	Prelazak na iduću riječ
<b></b>	Prelazak na početak tekuće riječi
<e></e>	Prelazak na kraj tekuće riječi.

U idućem primjeru ćemo pokazati neke od tih naredbi. Pokrenite *vi* editor i otvorite datoteku vipract.1 (upišite *vi* vipract.1). Sada pokušajte upotrijebiti jednu od naredbi za premještanje pokazivača, koje smo upravo opisali, i pokazivač (koji simbolizira podvučeno slovo) premjestite na slovo t u riječi "data" koja se nalazi u trećem retku dokumenta. Treći će redak, nakon toga, izgledati ovako:

```
b. Sort sales data and print the results.
```

Za prelazak na početak iduće riječi pritisnite tipku <w> i pokazivač će se premjestiti na slovo, a u riječi "and" za premještanje pokazivača na kraj riječi pritisnite tipku <e> i pokazivač će se naći na slovu d u riječi "and". Pritisnete li tipku <b>, pokazivač će ponovno biti naslovu, a u riječi "and".

Želite li pokazivač premjestiti na početak riječi koja je nekoliko riječi udaljena od ona na kojoj se trenutno nalazi pokazivač, onda pritisnite neki broj, a zatim tipku <w>. Primjerice, da biste pokazivač s trenutne pozicije (slovo, a u riječi "and") postavili na početak treće riječi (slovo r u riječi "results"), pritisnite tipke <3><w>. Na sličan način se možete vratiti ako pritisnete tipke <4><b> ili pokazivač možete premjestiti na kraj druge riječi (u desno) pomoću tipki <2><e>.

Brojeve možete dodavati i ispred tipki  $\langle h \rangle$ ,  $\langle j \rangle$ ,  $\langle k \rangle$ ,  $\langle l \rangle$ ,  $\langle + \rangle$  i  $\langle - \rangle$ . Primjerice, pritisnite tipke  $\langle 1 \rangle \langle 5 \rangle \langle j \rangle$  i pokazivač će se naći 15 redaka niže od trenutne pozicije. Ako 15. redak ne postoji, čut ćete zvučni signal, a pokazivač će ostati na istom mjestu.

**Tipke za velika premještanja** Pokazivač jednostavno možete postaviti na početak i kraj zaslona te u sredinu zaslona. U svim tim slučajevima pokazivač će se naći na početku retka. Slijede naredbe koje omogućuju postavljanje pokazivača na određeno mjesto na zaslonu:

- Pritisnite kombinaciju tipki <Shift-h> i pokazivač premjestite na prvi redak koji se vidi na zaslonu (ovaj se položaj pokazivača ponekad naziva *početna pozicija*).
- Pritisnite tipku <Shift-m> i pokazivač će se naći u srednjem retku na zaslonu.
- Pritisnite tipku <Shift-l> i premjestite pokazivač na najniži redak na zaslonu.

Želite li pregledati datoteku premještajući se za čitav zaslon (što je zgodnije nego da stalno pritišćete tipku <Return> ili <j>), upotrijebite naredbe koje služe za klizanje duž datoteke. Pritiskom na kombinaciju tipki <Ctrl-f> pokazivač će biti premješten za jedan redak unaprijed., dok se pritiskom na tipke <Ctrl-b> vraćate za jedan zaslon.

Za brzi prelazak na zadnji redak u datoteci pritisnite tipke <Shift-g>, a za prvi redak datoteke pritisnite <1><Shift-g>. Želite li pokazivač postaviti na početak određenog retka, tada najprije pritisnite broj željenog retka i zatim tipke <Shift-g>. Primjerice, za prelazak u 35. redak datoteke (naravno kada on postoji) pritisnite tipke <3><5><Shift-g>.

**BILJEŠKA** Potrebno je malo vježbanja da biste savladali postavljanje pokazivača na određeno mjesto u dokumentu. Upamtite da se uvijek morate nalaziti u načinu rada za unos naredbi ako želite pokazivač premjestiti na drugo mjesto. Zato uvijek najprije pritisnite tipku <Esc>.

## Dodavanje teksta

Da biste tekst dodali u spremnik, morate prijeći iz načina rada za unos naredbi (command mode) u način rada za unos teksta (input mode). Svaki znak koji unesete s tipkovnice je odmah dodan u spremnik. Ako u načinu rada za unos teksta pritisnete tipku <Return>, program *vi* "otvara", tj. dodaje redak u spremnik. Prije nego počnete dodavati tekst postavite pokazivač na mjesto na koje želite dodavati tekst. Pritisnite tipku <a> i pređite

u način rada za unos teksta, a zatim dodajte tekst iza pokazivača. Pritisnite tipku <i>za prelazak u način rada za unos teksta i umetnite tekst ispred pokazivača teksta. Nakon što završite s dodavanjem teksta pritisnite tipku <Esc i vratite se u zadavanje naredbi.

Slijede dva primjera pisanja teksta. Pokazivač se nalazi na mjestu na kojem je podcrtano slovo. U ova dva slučaja prikazano je ono što je bilo prije i tekst nakon unosa.

Primjer pokazuje uporabu naredbe <i> za dodavanje teksta.

Prije:

This report is important.

Pritisnite tipku <i> da biste umetnuli tekst ispred riječi "important", upišite very i pritisnite razmaknicu, a potom pritisnite tipku <Esc>.

Nakon:

This report is very important.

Uočite da se pokazivač nalazi iza zadnjeg slova riječi koju ste dodali (u ovom slučaju je to razmak između riječi very i riječi important).

Primjer pokazuje uporabu naredbe <a> za dodavanje teksta.

Prije:

This report is important.

Pritisnite tipku <a> da biste dodali tekst nakon riječi "is", zatim pritisnite razmaknicu, upišite **very**, a potom pritisnite tipku <Esc>.

Nakon:

This report is very important.

Uočite da se pokazivač nalazi na zadnjem slovu riječi koju ste dodali (u ovom slučaju je to riječ very).

Kada tekst želite dodati na kraju retka, onda pokazivač možete postaviti na kraj retka i pritisnuti tipku <a>. Ili pokazivač možete smjestiti na bilo koje mjesto u određenom retku i pritisnuti kombinaciju tipki <Shift-a>. Pokazivač će se naći na kraju retka, promijenit će se način rada tako da možete dodavati tekst. Na sličan način, uporabom naredbe <Shift-i>, možete dodavati vati tekst na početku retka u kojem se nalazi pokazivač.

Želite li tekst dodati ispod ili iznad retka u kojem se trenutno nalazi pokazivač, pritisnite tipku <o> ili kombinaciju tipki <Shift-o>. U oba slučaja ćete "otvoriti" redak u spremniku i moći ćete dodavati tekst. U iduća dva primjera je prikazano kako se dodaje redak u postojeći tekst.

Primjer prikazuje uporabu naredbe <o> za dodavanje retka ispod tekućeg retka.

```
Prije:
All jobs complete
please call
if you have any questions.
```

Pokazivač se nalazi u drugom retku. Pritisnite tipku <o> i dodajte jedan ili više redaka ispod tekućeg retka. Nakon toga upišite sljedeće:

```
Jack Tackett, Jr.
555-1837
```

Pritisnite tipku <Esc>.

Nakon:

```
All jobs complete
please call
Jack Tackett, Jr.
555-1837
if you have any questions.
```

Primjer prikazuje uporabu naredbe <Shift-o> za umetanje redaka iznad tekućeg retka.

Prije:

```
All jobs complete
please call
if you have any questions.
```

Pokazivač se nalazi u trećem retku. Pritisnite tipku <Shift-o> i dodajte jedan ili više redaka ispod tekućeg retka. Nakon toga upišite sljedeće:

```
Jack Tackett, Jr.
555-1837
Pritisnite tipku <Esc>.
```

Nakon:

```
All jobs complete
please call
Jack Tackett, Jr.
555-1837
if you have any questions.
```

U oba primjera, nakon što pritisnete tipku <Esc>, pokazivač se nalazi ispod zadnjeg znaka koji ste upisali (brojka 7 u telefonskom broju). Na isti način kao što ste dodali dva retka,

mogli ste dodati i više tako da na kraju retka pritisnete tipku <Return>. No, obično se dodaje samo jedan redak bez pritiskanja tipke <Return>.

U tablici 8.3 se nalazi kratak pregled naredbi za dodavanje teksta. Pritisnite tipku <Esc> kako biste prešli u način rada za zadavanje naredbi, a zatim pokrenite željenu naredbu.

Tablica 8.3	Naredbe za dodavanje teksta
Tipka	Opis
<a></a>	Umetanje teksta iza pokazivača
<shift-a></shift-a>	Vraćanje u način rada za unos teksta i dodavanje teksta na kraju tekućeg retka
<i></i>	Umetanje teksta prije pokazivača
<shift-i></shift-i>	Vraćanje u način rada za unos teksta i dodavanje teksta na početku tekućeg retka
<0>	Otvaranje novog retka ispod pokazivača
<shift-o></shift-o>	Otvaranje novog retka iznad pokazivača

## Brisanje teksta

Mijenjate li tekst ili ispravljate pogreške, vjerojatno ćete dio teksta trebati i obrisati. Znakove možete brisati iz načina rada za unos naredbi. Ukoliko radite u načinu rada za unos teksta, onda ćete prilikom pokretanja naredbi za brisanje znakova ustvari u spremnik upisivati određene znakove (slova koja služe za pokretanje naredbi). Ako se to dogodi, onda pritisnite tipku <Esc> i pređite u način rada za unos naredbi, te pokrenite naredbu <u> i poništite prethodnu pogrešku.

U vi editoru možete brisati znak, riječ, nekoliko riječi u nizu, sav tekst do kraja retka ili čitav redak. S obzirom da ovaj program predstavlja vizualni editor, to znači da se znakovi, riječi ili reci odmah uklanjaju s zaslona čim ih obrišete. U tablici 8.4 ćete naći naredbe za brisanje.

Tablica 8.4	Naredbe za brisanje teksta
Tipka	Opis
< <u>x</u> >	Brisanje znaka s mjesta na kojem se nalazi pokazivač
<d><w></w></d>	Brisanje teksta od mjesta na kojem se nalazi pokazivač do početka iduće riječi
<d>&lt;\$&gt;</d>	Brisanje teksta od mjesta na kojem se nalazi pokazivač do kraja retka
<shift-d></shift-d>	Isto kao i naredba <d>&lt;\$&gt;: brisanje dijela retka iza pokazivača</d>
<d><d></d></d>	Brisanje čitavog retka, bez obzira na položaj pokazivača u retku

Pri uporabi ovih naredbi bitan je položaj pokazivača. Postavite pokazivač na određeni znak, riječ ili u pojedini redak i zatim pokrenite odgovarajuću naredbu. Pokušajte ih pokrenuti kako biste uočili njihov učinak i vidjet ćete da se radi o vrlo korisnim naredbama.

Ove naredbe možete primijeniti na različite objekte: znakove, riječi ili retke. Dovoljno je prije naredbe upisati cijeli broj (ovaj način rada je prethodno opisan u odlomku za postavljanje pokazivača). Evo i nekoliko primjera:

- Pritisnite tipke <4><x> i obrišite 4 znaka.
- Pritisnite tipke <3><d><w> i obrišite 3 riječi.
- Pritisnite tipke <8><d><d>i obrišite 8 redaka.

**SAVJET** Da biste u *vi* editoru vidjeli redni broj retka, pritisnite tipku <Esc> (za prelazak u način rada za zadavanje naredbi) i potom upišite **:se number.** Kada ovu opciju želite isključiti, obavezno upišite **:se noumber.** 

Također imate mogućnost unijeti raspon redaka koje želite obrisati i to na ovaj način: pritisnite tipke <Shift-;>, zatim upišite dva broja koja označavaju raspon (brojeve odvojite zarezom), pritisnite tipku <d> i zatim tipku <Return>. Primjerice, želite li obrisati retke između 12. i 36., upišite **:12,36d** i pritisnite tipku <Return>.

Ako brišete dva ili više redaka, na statusnoj traci ćete vidjeti koliko ste redaka obrisali. Pritiskom na tipku <u> možete poništiti naredbu za brisanje.

## Pretraživanje

Traženje riječi, fraze ili broja može biti naporno ako morate pročitati sav tekst koji ste napisali (i koji je, pretpostavimo, dosta dugačak). Kao i većina programa za obradu teksta, program *vi* koristi naredbu za traženje određenog skupa znakova. Smjer traženja može biti prema kraju datoteke ili prema njezinu početku. *vi* editor, nakon što dođe do kraja datoteke, započinje traženje od početka spremnika (a vrijedi i obratno). U tablici 8.5 vidite naredbe za pretraživanje. U bilo kojem slučaju, program traži skup znakova koje ste odredili, i to u odabranom smjeru, a čim nađe određene znakove pokazivač se postavlja na njihovom početku.

Tablica 8.5	Naredbe za pretraživanje teksta
Naredba	Opis
/string	Traženje određenog skupa znakova (string) – prema kraju spremnika
?string	Traženje određenog skupa znakova (string) – prema početku spremnika
<n></n>	Ponovno traženje u istom smjeru
<shift-n></shift-n>	Ponovno traženje u suprotnom smjeru

Nakon što upišete naredbu za traženje, ona će se pojaviti u statusnoj traci. Primjerice, želite li prema kraju spremnika tražiti frazu sales > 100K, tada najprije trebate provjeriti da se nalazite u načinu rada za zadavanje naredbi i potom trebate upisati sljedeće:

#### /sales > 100K

Upisana naredba će se pojaviti u statusnoj traci. Ako se traženi skup znakova nalazi u spremniku, editor će postaviti pokazivač ispod prvog znaka u riječi *sales* (slovo "s"). Ukoliko zadanog skupa znakova nema u tekstu, u statusnoj traci će se pojaviti poruka

"Pattern not found". Za nastavak traženja znakova pritisnite tipku <n> i pokazivač će se naći na sljedećem pojavljivanju određene fraze (ako više takve fraze nema u tekstu, pokazivač će ostati na istom mjestu).

## OTKLANJANJE PROBLEMA

Upisao sam riječ za koju sigurno znam da postoji u tekstu, ali je vi editor ne može naći. Najčešći uzrok takve greške je da ste pogrešno upisali riječ koju tražite. *vi* editor (i općenito računala) nije savršen i treba mu puno vremena da otkrije što ste "mislili" kada ste upisali određenu riječ. Ako tražite riječ "vegi-burger", a napisali ste "vigi-burger", program je neće uspjeti pronaći (osim ako i u tekstu niste načinili istu pogrešku). Pažljivo provjerite skup znakova koje tražite prije nego pritisnete tipku <Return>.

Tražio sam frazu koja u sebi sadrži interpunkcijsku oznaku i program je pronašao krivu riječ. Pretraživanje u *vi* editoru možda neće dati željene rezultate ako tražite posebne znakove. Primjerice, ako želite naći riječ koja se nalazi na kraju rečenice, tada nemojte unositi i točku jer ona prilikom traženja označava "bilo koji znak", a ne "kraj rečenice". Upišete li /kraj. i zatim pritisnete tipku <Return>, program će naći i riječi *krajnji* ili *krajolik*. Da biste našli riječ "kraj" iza koje slijedi točka, trebate upisati slijedeće: /kraj\..

Pretraživanje je "osjetljivo" na velika i mala slova. Tražite li riječ "tigar", tada trebate upisati /**Tigar**, a ne /tigar.

# Mijenjanje teksta i zamjena drugim tekstom

Mijenjanje teksta ili njegova zamjena drugim tekstom je još jedna od čestih zadaća prilikom uređivanja teksta (između ove dvije radnje i nema neke bitne razlike). Naredbe u *vi* editoru koje služe za promjenu teksta omogućuju vam zamjenu jedne riječi ili ostatka retka (od mjesta na kojem se nalazi pokazivač, do kraja retka), dok kod zamjene teksta drugim tekstom možete mijenjati pojedinačne znakove ili više njih odjednom. U tablici 8.6 ćete naći popis naredbi za mijenjanje teksta. Nakon što pokrenete jednu od tih naredbi, jednostavno upišite no*vi* tekst.

Tablica 8.6	Naredbe za mijenjanje teksta	
Tipka	Opis	
< <u>r</u> >	Zamjena pojedinačnog znaka	
<shift-r></shift-r>	Zamjena više znakova odjednom	
<c><w></w></c>	Zamjena riječi na kojoj se nalazi pokazivač, od položaja pokazivača do kraja riječi	
<c><e></e></c>	Zamjena riječi na kojoj se nalazi pokazivač, od položaja	
	pokazivača do kraja riječi (isto kao i <c><w>)</w></c>	
<c><b></b></c>	Zamjena riječi na kojoj se nalazi pokazivač, od početka riječi do jednog znaka prije pokazivača	nastavlja se

Tablica 8.6	Nastavak
Tipka	Opis
<c>&lt;\$&gt;</c>	Zamjena retka, od mjesta na kojem se nalazi pokazivač do kraja retka
<shift-c></shift-c>	Zamjena retka, od mjesta na kojem se nalazi pokazivač do kraja retka (isto kao i <c>&lt;\$&gt;)</c>
<c><c></c></c>	Zamjena čitavog retka



SAVJET Da biste odjednom zamijenili više riječi, trebate upisati cijeli broj (koji predstavlja broj riječi koje želite zamijeniti), prije nego pritisnete tipke <c><w>

Slijede tri primjera uporabe naredbi zamjene teksta.

Primjer prikazuje kako se koristi naredba <c><e> za promjenu teksta do kraja riječi

Prije:

The report demonstraits thw, strenghts of are apporach.

Pokazivač se nalazi na mjestu nepravilno napisane riječi. Da biste riječ ispravili, pritisnite tipke <c><e>, upišite tes, i pritisnite <Esc>.

Poslije:

The report demonstrates thw, strenghts of are apporach.

Primjer prikazuje kako se koristi naredba <Shift-r> za promjenu skupa znakova

Prije:

The report demonstrates thw, strenghts of are apporach.

Pokazivač se nalazi na mjestu nepravilno napisane riječi. Da biste riječ ispravili, pritisnite tipke <Shift-r>, upišite e, pritisnite razmaknicu i zatim pritisnite <Esc>.

Poslije:

The report demonstrates the strenghts of are apporach.

Primjer prikazuje kako se koristi naredba <c><w> za promjenu teksta počev od riječi na kojoj se nalazi pokazivač do kraja rečenice.

Prije:

The report demonstrates the strenghts of are apporach.

Pokazivač se nalazi na početku prve nepravilno napisane riječi. Da biste ispravili zadnje dvije riječi u retku, pritisnite tipke <2><c><w>, upišite our approach i pritisnite <Esc>.

Poslije:

The report demonstrates the strenghts of our approach.

Za povratak u zadavanje naredbi pritisnite tipku <Esc>.

## Kopiranje, rezanje i lijepljenje

Kada brišete ili režete znakove, riječi ili retke, odnosno dijelove redaka, obrisani objekt se sprema u tzv. spremnik s općom namjenom. Samo ime spremnika nemojte niti pamtiti, ali je važno znati da njegov sadržaj možete staviti, ili zalijepiti bilo gdje unutar teksta koji uređujete. To možete učinit tako da pokrenete naredbe ili <Shift-p>. Naredbom lijepite objekt desno od mjesta na kojem se nalazi pokazivač, a naredbom <Shift-p> lijevo od pokazivača (prije njega).

Slijedi nekoliko primjera rezanja i lijepljenja teksta.

Primjer prikazuje uporabu naredbe za lijepljenje sadržaja spremnika iza mjesta na kojem se nalazi pokazivač.

Prije:

Carefully carry these out instructions.

Obrišite riječ"out" i razmak iza nje tako da pokrenete naredbu <d><w>. Zatim premjestite pokazivač na kraj riječi "carry" i pritisnite tipku .

Poslije:

Carefully carry out these instructions.

Primjer prikazuje uporabu naredbe <Shift-p> za lijepljenje sadržaja spremnika ispred mjesta na kojem se nalazi pokazivač.

Prije:

Carefully carry these out instructions.

Obrišite riječ"these" i razmak iza nje tako da pokrenete naredbu <d><w>. Zatim premjestite pokazivač na početak riječi "instructions" i pritisnite tipke <Shift-p>.

Poslije:

Carefully carry out these instructions.

SAVJET Da biste zamijenili redoslijed dvaju znakova, postavite pokazivač na prvi znak i pritisnite tipke <x>. Pokušajte zamijeniti riječ "dva" u riječ "dva".
U prethodnim ste primjerima vidjeli kako tekst možete lijepiti nakon brisanja. No, nije nužno obrisati tekst da biste ga mogli lijepiti na drugom mjestu. Možete upotrijebiti naredbu *yan*, koja se u drugim programima za uređivanje teksta naziva *copy*. Oblik naredbe yank je vrlo sličan obliku naredbe za brisanje teksta. Ideja je da kopirate dio teksta i zatim ga zalijepite na drugo mjesto pomoću naredbe ili <Shift-p>. U ovom popisu se nalaze naredbe za kopiranje (uočite da većina tih naredbi koristi malo slovo y):

Tipka	Opis
<y>&lt;<sub>W</sub>&gt;</y>	Kopiranje od mjesta na kojem se nalazi pokazivač do početka iduće riječi.
<y>&lt;\$&gt;</y>	Kopiranje od mjesta na kojem se nalazi pokazivač do kraja retka.
<shift-y></shift-y>	Isto kao i naredba <y>&lt;\$&gt;: kopiranje ostatka retka.</y>
<y><y></y></y>	Kopiranje čitavog retka u kojem se nalazi pokazivač.

Sve navedene naredbe možete primijeniti na više objekata: znakove, riječi ili retke, ako prije same naredbe upišete cijeli broj.

Da biste kopirali četiri uzastopna retka, trebate učiniti sljedeće:

- 1. Postavite pokazivač na početak prvog retka.
- 1. Pritisnite tipke <4><y><y> za kopiranje tekućeg i još tri iduća retka. Na zaslonu nećete opaziti nikakve promjene.
- 3. Postavite pokazivač na mjesto na koje želite zalijepiti tekst iz spremnika.
- Pritisnite tipku i ulijepite odabrani tekst ispod retka u kojem se nalazi pokazivač.

Također možete tražiti i mijenjati riječi u tekstu (u čitavoj datoteci ili u određenom dijelu teksta). Oblik naredbe je ovaj:

:[raspon]s/staritekst/novitekst/g

gdje je

raspon	Označava dio teksta koji pretražujete. Primjerice, možete koristiti znak % ako želite pretraživati čitavu datoteku, ili brojeve pojedinih redaka (primjerice 1,4 za pretraživanje u recima od 1 do 4).
S	Naziv naredbe za traženje i zamjenu teksta.
staritekst	Skup znakova koje treba zamijeniti (stari znakovi).
novitekst	Skup znakova koji će zamijeniti stari skup znakova (novi znakovi).

Primjerice, da biste zamijenili nepravilno napisanu riječ "primljneo" s pravilnom riječi "primljeno", i to u čitavoj datoteci, onda trebate pokrenuti ovu naredbu:

:%s/primljneo/primljeno/g

#### Ponavljanje naredbi

Ne samo što *vi* editor sprema tekst koji ste obrisali ili kopirali, tako da isti tekst možete koristiti više puta, već program sprema i naredbu koju ste zadnju koristili. Zadnje korištenu naredbu možete ponavljati pritiskom na tipku<.>.

Pretpostavimo da ste dovršili svoj izvještaj, ali ste se potom dosjetili da bi bilo zgodno na ključnim mjestima u izvještaju dodati ova dva retka:

Retke možete dodati na ovaj način:

- 1. Postavite pokazivač na mjesto na kojem se reci trebaju prvi puta pojaviti.
- 2. Umetnite prazne retke pomoću naredbe <o> i upišite željeni tekst.
- 3. Pritisnite tipku < Esc> i pređite u način rada za zadavanje naredbi.
- Potražite iduće mjesto u tekstu na koje želite dodati ista dva retka i pritisnite tipku <.>. Ovaj korak ponavljajte proizvoljan broj puta.

#### Naredbe u vi editoru

Do sada ste stekli osnovna znanja za uporabu *vi* editora. U tablici 8.7 se nalazi popis naredbi i tipki koje se koriste u *vi* editoru.

Tablica 8.7	Popis naredbi u vi editoru	
Tipka/Naredba	Opis	
<i></i>	Umetanje teksta prije pokazivača	
<i></i>	Unos teksta na početku retka	
<a></a>	Umetanje teksta iza pokazivača	
<a></a>	Unos teksta na kraju retka	
<0>	Otvaranje novog retka ispod pokazivača	
<0>	Otvaranje novog retka iznad pokazivača	
<d>&lt;&lt;</d>	Brisanje riječi	
<d><d></d></d>	Brisanje čitavog retka	
<d></d>	Brisanje do kraja retka	
<_X>	Brisanje znaka na kojem se nalazi pokazivač	
<c><w></w></c>	Mijenjanje riječi	
<c><c></c></c>	Mijenjanje retka	nastavlja se

Tablica 8.7	Nastavak
Tipka/Naredba	Opis
<c></c>	Mijenjanje do kraja retka
<r></r>	Zamjena znaka na kojem se nalazi pokazivač
<j></j>	Objedinjavanje redaka
<e></e>	Premještanje pokazivača na kraj riječi
<w></w>	Premještanje pokazivača na iduću riječ
<\$>	Premještanje pokazivača na kraj retka
<l></l>	Premještanje pokazivača za jedno mjesto u desno
<k></k>	Premještanje pokazivača za jedan redak gore
<j></j>	Premještanje pokazivača za jedan redak dolje
<h></h>	Premještanje pokazivača za jedno mjesto u lijevo
<f><x></x></f>	Premještanje pokazivača na prvu pojavu oznake x
<f><x></x></f>	Premještanje pokazivača na zadnju pojavu oznake x
<;>	Ponavljanje zadnje naredbe f/F
broj <đ>	Premještanje pokazivača na određeni stupac
<h></h>	Premještanje pokazivača na najviši redak na zaslonu (a ne u datoteci)
<l></l>	Premještanje pokazivača na najniži redak na zaslonu
<m></m>	Premještanje pokazivača na srednji redak na zaslonu
<g></g>	Premještanje pokazivača na najniži redak u datoteci
broj <g></g>	Premještanje pokazivača na određeni redak (isto kao i <esc>:broj)</esc>
<^>	Premještanje pokazivača na početak retka
<m>x</m>	Obilježavanje trenutne pozicije oznakom x
<ctrl-d></ctrl-d>	Klizanje za pola zaslona unaprijed
<ctrl-u></ctrl-u>	Klizanje za pola zaslona unatrag
<ctrl-f></ctrl-f>	Klizanje za čitav zaslon unaprijed
<ctrl-b></ctrl-b>	Klizanje za čitav zaslon unatrag
<ctrl-l></ctrl-l>	Obnavljanje sadržaja zaslona
<ctrl-g></ctrl-g>	Prikaz imena datoteke, te retka i stupca u kojem se nalazi pokazivač
< <u>z</u> >< <u>z</u> >	Obnavljanje sadržaja zaslona (s tekućim retkom u sredini zaslona)
<y><y></y></y>	Pohrana retka u spremnik

Tipka/Naredba	Opis
	Unos sadržaja spremnika ispod pokazivača
<p></p>	Unos sadržaja spremnika iznad pokazivača
x"[ <i>broj</i> ]" <y><y></y></y>	Pohrana određenog broja redaka u spremnik s imenom $x$ ( $x$ može biti bilo koji znak od a- $z$ )
x>	Unos sadržaja spremnika x iza pokazivača
:w [datoteka]	Pohrana na disk pod imenom datoteka.
:q	Izlazak iz vi editora
:q!	Izlazak iz vi editora bez spremanja promjena
:wq	Pohrana promjena i izlazak iz vi editora
:r datoteka	Učitavanje određene datoteke u program
:e datoteka	Uređivanje određene datoteke
:!naredba	Pokretanje određene naredbe
:broj	Premještanje na određeni broj retka
:f	Ispis retka u kojem se nalazi pokazivač i imena datoteke (kao i <ctrl-g>)</ctrl-g>
/string	Traži određeni skup znakova (prema kraju datoteke)
?string	Traži određeni skup znakova (prema početku datoteke)
:x,ys/oldstring/newstring	Zamjenjuje skup znakova <i>oldstring</i> sa znakovima <i>newstring</i> , i to od retka x do retka y (upišete li umjesto y oznaku \$, bit će zamijenjeno sve do kraja datoteke)
<esc><u></u></esc>	Poništavanje zadnje naredbe
<n></n>	Traženje idućeg mjesta pojavljivanja određenog znaka
	Ponavljanje zadnje naredbe
~	Promjena znaka
<esc></esc>	Prelazak u način rada za zadavanje naredbi

## Podešavanje sučelja vi editora

*vi* editor sadrži nekoliko opcija koje možete, ali i ne morate koristiti. Neke od tih opcija može podesiti administrator sustava. Sučelje programa možete podesiti pomoću nekoliko opcija navedenih u tablici 8.8. Za podešavanje opcija koje se odnose na sučelje programa (kao što je to opisano u idućem poglavlju) možete upotrijebiti kratice iz prvog stupca tablice ili puno ime koje se nalazi u drugom stupcu.

Tablica 8.8	Opcije za podešavanje sučelja u vi editoru
Kratica opcije	Puno ime i namjena opcije
ai	autoindent služi za uvlačenje svakog retka prema retku iznad (korisna je kod pisanja programa). Izvorno je opcija isključena.
ap	autoprint se koristi za ispisna zaslon onog retka u kojem se nalazi pokazivač (ukoliko se redak mijenja). Izvorno je opcija uključena.
eb	errorbells služi za zvučno javljanje u slučaju greške. Opcija je izvorno isključena.
nu	number prikazuje brojeve redaka kod uređivanja datoteke. Opcija je izvorno isključena.
redraw	redraw ažurira izgled zaslona i izvorno je opcija uključena.
report	report podešava broj mogućih promjena koji je potreban da bi se u statusnoj traci pojavila poruka. Primjerice, ako je opcija podešena tako da je report=3, na zaslonu će se pojaviti poruka nakon što obrišete tri retka (ali ne i ako obrišete manje redaka). Izvorno podešenje je report=5.
sm	showmatch prikazuje broj otvorenih zagrada i obično je koriste programeri za pisanje koda. Izvorno je opcija isključena.
smd	showmode služi za prikaz načina rada u statusnoj traci (INPUT, REPLACE ili CHANGE). Izvorno je opcija isključena.
warn	warn prikazuje poruku prilikom izlaska iz programa, ukoliko niste spremili sadržaj spremnika na disk. Opcija je izvorno uključena.
wm=n	wrapmargin definira položaj desne margine. Oznaka n je označava cijeli broj, a ukoliko je taj broj veći od 0, onda se sve riječi koje su dulje od n znakova premještaju u novi redak čim pokazivač naiđe na kraj retka. Primjerice, ako je wn=5, sve riječi koje imaju više od pet slova, a nalaze se na kraju retka, će biti premještene u novi redak. Izvorno je opcija podešena na wm=0 (isključeno).
WS	word search (u nekim sustavima se opcija naziva wrapscan) služi za prelazak s kraja datoteke ( <eof>) na njezin početak (<bof>) prilikom pretraživanja teksta. Opcija je izvorno uključena.</bof></eof>

#### Uporaba naredbe set za pregled i podešavanje opcija

Želite li vidjeti opcije koje se trenutno koriste u vašem sustavu, u načinu rada za unos naredbi upišite **:set**, i u statusnoj traci će se pojaviti trenutno podešene opcije, koje se mogu razlikovati od računala do računala, ovisno o izvornim opcijama i podešenjima koje ste sami dodali. Slijedi primjer onoga što možete vidjeti ako pokrenete naredbu set:

```
autoprint errorbells redraw report=1 showmatch showmode term=vt100 wrap margin=5 \,
```

**BILJEŠKA** Pokrenete li naredbu set bez argumenata, na zaslonu će se pojaviti samo opcije koje je podesio korisnik (odnosno vi). Želite li podesiti broj opcija koje se prikazuju u istom retku, pokrenite naredbu i opcije odvojite s jednim razmakom (kao u idućem primjeru):

:se ap eb redraw report=1 sm smd warn wm=5 ws

Uočite da se na početku retka nalazi dvotočka, što upućuje da iza nje slijedi naredba.

Da biste vidjeli popis svih opcija i njihova podešenja, upišite naredbu **:set all.** Na zaslonu ćete vidjeti popis koji je dat u tablici 8.7.

#### Podešavanje opcije showmode

Jedna od najčešće korištenih opcija je opcija showmode. Želite li nešto više saznati o opciji showmode, pokrenite *vi* editor i otvorite datoteku vipract.1 (upišite *vi* vipract.1).

Prilikom prvog pokretanja programa vi niste mogli odrediti nalazite li se u načinu rada za unos teksta ili ne. Pomoću opcije showmode možete podesiti program da vas upozori ako se nalazite u takvom načinu rada, te da način rada bude upisan u statusnoj traci.

Nakon što podesite opciju showmode, *vi* editor će prikazivati u kojem se načinu rada nalazite: INPUT MODE (za unos teksta), APPEND MODE (za dodavanje teksta), REPLACE 1 CHAR (zamjena jednog znaka) i slično. Da biste uključili ovu opciju, najprije pritisnite tipku <Esc> za prelazak u način rada za unos naredbi i potom upišite :set showmode. Potom se vratite u način rada za unos teksta (pritisnite tipku <i>), i u statusnoj traci biste trebali vidjeti poruku INPUT MODE. Pritisnite tipku <Esc> i pokušajte pokrenuti naredbu za zamjenu teksta.

#### Podešavanje opcija za uključivanje/isključivanje

Bilo koja opcija koja ne koristi broj kao argument je opcija za uključivanje/isključivanje. Primjerice, kao što ste u prethodnim odlomcima naučili, možete uključiti opciju showmode tako da upišete naredbu:

:se showmode

Da biste istu opciju isključili, jednostavno dodajte no ispred riječi showmode:

:se noshowmode.

#### Mijenjanje opcija kod svakog pokretanja vi editora

Podešavanje opcije tijekom rada s vi editorom opciju ćete promijeniti samo dok ne napustite program. No način rada programa možete podesiti tako da promjene unesete u datoteku .excr koje se nalazi u vašem polaznom direktoriju. Da biste vidjeli postoji li takova datoteka u vašem direktoriju, upišite sljedeće naredbe:

cd vi .exrc

Pomoću prve naredbe ćete otvoriti polazni direktorij, a pomoću druge naredbe ćete u vi editoru otvoriti datoteku .excr. Ako datoteka postoji, ona će se pojaviti na zaslonu, a ako ne postoji, program će vas upozoriti da se radi o novoj datoteci.

Naredbe set, u datoteci .excr, započinju s riječi "set", ali bez zareza. U idućem primjeru ćete vidjeti kao se podešavaju opcije number i showmode:

set number showmode

BILJEŠKA Datoteka –excr se učitava kod pokretanje vi editora. Ukoliko je stvorite dok se nalazite u vi editoru, morate ponovno pokrenuti programa da biste vidjeli unesene promjene.

Opcije koje podešavate i njihov rezultat ovise o načinu na koji uređujete dokument. Pokušajte i sami mijenjati opcije ili upitajte nekog od iskusnijih korisnika.

## Odavdje...

Iako u ovom poglavlju nismo mogli objasniti sve opcije ili mogućnosti vi editora, jasno vam je gdje trebate početi i kako možete upotrijebiti osnovne značajke vi editora. Ovim programom svakako trebate ovladati jer se on nalazi na svakom Linux/UNIX sustavu, a osim toga se lako učitava i ne zahtijeva puno memorije tako da ga možete upotrijebiti i onda kada ne možete učitati niti jedan drugi program za uređivanje teksta. Administratori sustava koriste vi editor kod unosa brzih (i neurednih) promjena. Više informacija potražite u idućim odlomcima:

- 11. poglavlje, "Sigurnosna pohrana podataka" objašnjava vam kako ćete zaštiti svoje datoteke s tekstom od neželjenih posljedica. Osim toga ćete naučiti kako stvoriti sigurnosnu kopiju važnijih datoteka koje ste napisali pomoću vi editora.
- U 16. poglavlju, "Sustav datoteka i direktorija" ćete naučiti kako možete manipulirati s datotekama i kako Linux tretira datoteke. Koristite li vi editor ili neki drugi program za obradu teksta, onda vam je svakako potrebno određeno predznanje o sustavu datoteka.
- 20. poglavlje, "Ispisivanje na pisač" sadrži prikaz ispisa na pisač datoteka koje ste stvorili pomoću vi editora.
- U 28. poglavlju, "Uporaba editora emacs" objašnjen je još jedan program za uređivanje teksta, koji se koristi pod Linuxom. Program emacs nudi puno više od vi editora, a osim toga daje vam mogućnost čitanja e-pošte i novosti iz novinskih grupa.

# poglavlje 9

## Pokretanje i isključivanje sustava

(Jack Tackett)

## U ovom poglavlju

Pokretanje sustava 206 Pokretanje sustava s diskete 212 Pokretanje sustava iz Boot Managera 213 Što je Linux Loader? 214 Isključivanje sustava 215 Dva najčešća zadatka koje obavlja administrator sustava u Linuxu su pokretanje i isključivanje sustava. Kao što već zasigurno pogađate, pokretanje i isključivanje Linuxa se obavlja pod posebnim uvjetima.

Da biste koristili Linux, morate pokrenuti operativni sustav. Iako to može zvučati jednostavno, trebate znati da većina ljudi koristi barem još jedan operativni sustav osim Linuxa. To znači da morate odrediti način odabira operativnog sustava, što možete učiniti na dva načina: Linux možete pokrenuti s diskete ili s tvrdog diska uz uporabu posebnog programa za odabir načina pokretanja računala.

#### Pokretanje sustava

Red Hat i većina novijih distribucija Linuxa koriste SysV init postupak za pokretanje, umjesto starog postupka BSD style init. Program init je prvi program koji jezgra izvršava prilikom pokretanja sustava i taj program nosi identifikacijski broj (PID) 1. Radi se o procesu koji prethodi svim drugim procesima u Linuxu.

BILJEŠKA Broj PID je brojka koju operativni sustav koristi za identifikaciju procesa. Mnoge Linux naredbe koriste taj broj kao parametar.

Tijekom pokretanja Linuxa se događa sljedeće:

- 1. Jezgra pokreće program init koji se nalazi u direktoriju /sbin.
- 2. Program init pokreće skriptu /etc/rc.d/rc.sysinit.
- 3. Skripta rc.sysinit podešava razne varijable sustava i obavlja početnu inicijalizaciju.
- 4. Program init pokreće sve skripte koje su podešene za pokretanje tijekom pokretanja sustava.
- 5. Program init pokreće skriptu /etc/rc.d/rc.local.
- Vidi "Što su procesi u Linuxu", 349. stranica

Ovaj program pokreće razne procese i zapisuje podatke na ekran i u datoteku /var/log/messages (spremaju se podaci o stanju svakog procesa koji se pokreće). Ispis 9.1 prikazuje primjer jedne takve datoteke.

U datoteci /var/log/message možete naći puno toga što vam može pomoći u otklanjanju SAVJET problema kod pokretanja sustava, jer u nju jezgra sprema sve poruke o greškama tako da te poruke ne trebate zapisivati kada ih ugledate na ekranu.

#### Ispis 9.1 Tipični zapisi u log datoteci

```
May 22 23:23:42 ns syslogd 1.3-3: restart.
May 22 23:23:43 ns kernel: klogd 1.3-3, log source = /proc/kmsg started
May 22 23:23:45 ns kernel: Loaded 4189 symbols from /boot/System.map.
May 22 23:23:45 ns kernel: Symbols match kernel version 2.0.31.
May 22 23:23:45 ns kernel: Loaded 2 symbols from 3 modules.
May 22 23:23:45 ns kernel: Console: 16 point font, 400 scans
May 22 23:23:45 ns kernel: Console: colour VGA+ 80x25, 1 virtual console
⇒ (max 63)
```

Ovaj fajl je skinut sa sajta www.tutoriali.org

May 22 23:23:45 ns kernel: pci\_init: no BIOS32 detected May 22 23:23:45 ns kernel: Calibrating delay loop.. ok - 49.97 BogoMIPS May 22 23:23:45 ns kernel: Memory: 30816k/3268k aviable (736k kernel code, ⇒ 384k reserved, 832k data) May 22 23:23:45 ns kernel: This processor honours the WP bit even when in supervisor mode. Good. May 22 23:23:45 ns kernel: Swansea University Computer Society NET3.035 for ⇒Linux 2.0 May 22 23:23:45 ns kernel: NET3: Unix domain sockets 0.13 for Linux NET3.035. May 22 23:23:45 ns kernel: Swansea University Computer Society TCP/IP for ➡ NET3.034 May 22 23:23:45 ns kernel: IP Protocols: IGMP, ICMP, UDP, TCP May 22 23:23:45 ns kernel: VFS: Diskquotas version dquot\_5.6.0 initialized May 22 23:23:45 ns kernel: May 22 23:23:45 ns kernel: Checking 386/387 coupling... Ok, fpu using ⇒ exception 16 erroe reporting. May 22 23:23:45 ns kernel: Checking 'hlt' instruction... Ok. May 22 23:23:45 ns kernel: Linux version 2.0.31 (root@porky.redhat.com) ⇒ (gcc version 2.7.2.3.) #1 Sun Nov 9 21:45:23 EST 1997 May 22 23:23:45 ns kernel: Starting kswapd v 1.4.2.2 May 22 23:23:45 ns kernel: Serial driver version 4.13 with no serial options ⇒ enabled May 22 23:23:45 ns kernel: tty00 at 0x03f8 (irq =4) is, a 16550A May 22 23:23:45 ns kernel: tty01 at 0x02f8 (irg =3) is, a 16550A May 22 23:23:45 ns kernel: Real Time Clock Driver v1.07 May 22 23:23:45 ns kernel: Ramdisk driver initialized . 16 ramdisks of ₩ 4096K size May 22 23:23:45 ns kernel: hda: Micropolis 2217A, 1551MB w/508kB Cache, ➡ CHS=3152/16/63 May 22 23:23:45 ns kernel: had: Maxtor 72700 AP, 2583MB w/128kB Cache, ➡ CHS=20746/15/17 May 22 23:23:45 ns kernel: ide0 at 0x1f0-0x1f7, 0x3f6 on irg 14 May 22 23:23:45 ns kernel: Floppy drive(s): fd0 is 1.44M May 22 23:23:45 ns kernel: FDC 0 is an 8272A May 22 23:23:45 ns kernel: md driver 0.35 MAX\_MD\_DEV =4, MAX\_REAL=8 May 22 23:23:45 ns kernel: scsi : 0 hosts. May 22 23:23:45 ns kernel: scsi: detected total. May 22 23:23:45 ns kernel: Partition check: May 22 23:23:45 ns kernel: hda: hda1 May 22 23:23:45 ns kernel: hdb: hdb1 hdb2 May 22 23:23:45 ns kernel: VFS: Mounted root (ext2 filesystem) readonly. May 22 23:23:45 ns kernel: Adding Swap: 3300k swap-space (priority -1) May 22 23:23:45 ns kernel: sysctl: ip forwarding off May 22 23:23:45 ns kernel: Swansea University Computer Society IPX 0.34 ➡ for NET3.035 May 22 23:23:45 ns kernel: IPX Portions Copyright 1995 Caldera, Inc May 22 23:23:45 ns kernel: Appletalk 017 for Linux NET3.035 May 22 23:23:45 ns kernel: eth0: 3c509 at 0x300 tag 1, 10baseT port, → address 00 60 97 13 30 e1, IRQ 10. May 22 23:23:45 ns kernel: 3c509.c:1.12 6/4/97 becker@cesdis.gsfc.nasa.gov May 22 23:23:45 ns kernel: eth0: Setting Rx mode to 1 addresses. May 22 23:23:50 ns named [243]: starting. named 4.9.6-REL Thu Nov 6 23:29:57 ⇒ EST 1997 ^ Iroot@porky.redhat.com:/usr/src/bs/BUILD/bind-4.9.6/named

Program init pokreće sve procese koji su nužni da bi operativni sustav obavljao svoje zadatke, kao što su mrežni programi, uporaba miša te osnovne funkcije kao što je ulaz/izlaz terminala. Program SysV init zna koje procese treba pokrenuti jer ima uvid u datoteke s podešenjima koje se nalaze u direktoriju /etc/rd.d. Te su datoteke dalje podijeljene prema razini pokretanja određenoj putem direktorija.

Razina pokretanja procesa određuje način rada, od jednog korisnika (razina 1) do višekorisničkog, višezadaćnog načina rada (razina 3). Tablica 9.1 prikazuje razine pokretanja koje se koriste u Linuxu.

Tablica 9.1	Razine pokretanja Linuxa
Razina	Opis
0	zaustavljanje
1	jedan korisnik
2	više korisnika, bez NFS
3	potpuni višekorisnički rad
4	neiskorišteno
5	X11
6	ponovno pokretanje

Program init koristi ovu strukturu direktorija:

init.d rc0.d rc1.d rc2.d rc3.d rc4.d rc5.d rc6.d

Razni brojevi u imenima direktorija odgovaraju razinama pokretanja iz tablice 9.1. Svaki direktorij sadrži razne skripte koje pokreću ili zaustavljaju potrebne servise. Pomoću ovih skripti se također inicijalizira sustav datoteka te zaključavaju datoteke.

Vidi "Rad sa skriptama", 365. stranica

Svaki direktorij sadrži različite skripte, a ime svake skripte počinje sa slovom S ili K (Start ili Kill), a zatim slijedi dvoznamenkasti broj. Brojevi se upotrebljavaju kako bi se poredale sekvence i nemaju drugo značenje.

Svaka skripta obično koristi i argument za pokretanje ili zaustavljanje, ali može prihvatiti i druge parametre. Program *init* pridjeljuje skriptama samo parametre *start* ili *stop*, ovisno o

tome je li pozvan program *rc* za promjenu razine pokretanja. Skripte također možete pokretati i ručno ako trebate promijeniti servis. Primjerice, možete koristiti program *sendmail* sa sljedećim naredbama (morate se prijaviti kao superkorisnik ako želite izvoditi init skripte):

```
/etc/rc.d/init.d/sendmail stop
/etc/rc.d/init.d/sendmail stop
/etc/rc.d/init.d/sendmail start
```

Kod ove naredbe trebate uočiti dvije stvari. Prvo, naredba se dva puta ponavlja s parametrom *stop*, kako bi sustav imao dovoljno vremena da zaustavi proces. Nakon toga se pokreće naredba s parametrom *start*. Zatim uočite da se skripta izvodi u direktoriju init.d, a ne iz direktorija za pokretanje. Osim toga, skripta ne sadrži slovo (S ili K) ili broj. Ispišete li popis datoteka u direktoriju za razinu pokretanja, uočit ćete da su povezane s datotekama u direktoriju init.d (pogledajte ispis 9.2). Nakon ispisa slijedi tablica 9.2 s popisom nekoliko najbitnijih skripti u tom direktoriju.

Ispis 9.2		Ispis	direkt	orija	u ti	pično	m rc3.d	direktoriju
lrwxrwxrwx	1	root	root	16	Jan	25	21:56	KO8autofs -> . /init.d/autofs
lrwxrwxrwx	1	root	root	18	Dec	14	12:17	K10pnserver -> ./ init.d/pnserver
lrwxrwxrwx	1	root	root	17	Dec	14	12:17	K2Oruserds -> ./ init.d/ ruserds
lrwxrwxrwx	1	root	root	15	Dec	14	12:17	K2Orwhod -> ./ init.d/rwhod
lrwxrwxrwx	1	root	root	15	Dec	14	12:17	S15nfsfs -> ./ init.d/nfsfs
lrwxrwxrwx	1	root	root	16	Dec	14	12:17	S2Orandom -> ./ init.d/random
lrwxrwxrwx	1	root	root	16	Dec	14	12:17	S3Osyslog -> ./ init.d/syslog
lrwxrwxrwx	1	root	root	13	Dec	14	12:17	S40atd -> ./ init.d/atd
lrwxrwxrwx	1	root	root	15	Dec	14	12:17	S40crond -> ./ init.d/crond
lrwxrwxrwx	1	root	root	14	Dec	14	12:17	S50inet -> ./ init.d/inet
lrwxrwxrwx	1	root	root	15	Dec	14	12:17	S55named -> ./ init.d/named
lrwxrwxrwx	1	root	root	13	Dec	14	12:17	S60lpd -> ./ init.d/lpd
lrwxrwxrwx	1	root	root	13	Jan	31	20:17	S72ams -> ./ init.d/amd
lrwxrwxrwx	1	root	root	18	Dec	14	12:17	S75keytable -> ./ init.d/keytable
lrwxrwxrwx	1	root	root	18	Dec	14	12:17	<pre>S80sendmail -&gt; ./ init.d/sendmail</pre>

Tablica 9.2	rc.3 init skripte	
Ime skripte	Demon	Opis
S15nfsfs	nfs	Upravlja mrežnim servisima za datoteke (NFS)
S30syslog	syslog	Omogućuje zapisivanje sustavskih poruka u datoteku /var/log/messages
S40atd	atd	Omogućuje korisnicima obavljanje određene zadaće u naznačeno vrijeme
S40crond	cron	Linux program za praćenje rada (raspored rada)
S50inet	inetd	Super poslužitelj (PID 1)
S55named	Name server	Omogućuje rad s DNS imenima
S60lpd	lpd	Demon koji omogućuje ispis na pisač

Program init učitava datoteke u određenom direktoriju za razinu pokretanja i postavlja parametre start i stop, ovisno o prvom znaku u imenu datoteke.

#### Vidi "Veze", 308. stranica.

Direktorij rc.d sadrži tri datoteke: *rc, rc.local* i *rc.sysinit*. Skripta *rc* je zadužena za pokretanje sustava u drugoj razini i sadrži jedan parametar, to jest broj koji određuje novu razinu pokretanja. Datoteka *rc.local* se pokreće nakon svih drugih skripti i u nju možete unijeti bilo kakve upute za lokalnu inicijalizaciju. Datoteka *rc.local* (čiji sadržaj vidite u popisu 9.3) sadrži primjer pokretanja lokalnog procesa nazvanog *secure shell*, koji omogućuje siguran daljinski pristup sustavu.

```
Ispis 9.3 Primjer skripte rc.local
```

```
#!/bin/sh
#This script will be executed *after* all the other init scripts.
#You can put your own initialization stuff in here if you don't
#want to do the full Sys V style init stuff.
if [ -f /etc/redhat-release ]; then
       R=$(cat /etc/redhat-release)
else
       R="release 3.0.3"
if
arch=$(uname -m)
a="a"
case "_$arch" in
       a*) a="an" ;;
       _ i*) a="an" ;;
esac
# This will overwrite /etc/issue at every boot. So, make any changes you
# want tomake to /etc/issue here or you will lose them when you reboot.
echo " " > /etc/issue
echo "Red Hat Linux $R >> /etc/issue
echo "Kernel $(uname -r) on $a $(uname -m) " >> /etc/issue
cp -f /etc/issue /etc/issue.net
echo >> /etc/issue
## Start sshd
### Added By Lance Brown 1/29/1998
/usr/local/sbin/sshd
```

Datoteka *rc.sysinit* je prva datoteka koju pokreće datoteka *init*, a sadrži naredbe, primjerice, za podešavanje varijable *wide* (slična imenu *hosta*), provjeru sustava datoteka, povećanje korisničkih kvota i stvaranje sustava datoteka /proc. Skripta u ispisu 9.3 pokreće i lokalni proces pod imenom *sshd*, (*demon* za sigurnost ljuske) koji osigurava sigurnost Telneta i daljinskih naredbi u Linuxu.

BILJEŠKA sshd nije dio standardne Red Hat distribucije zbog ograničenja izvoza određenih proizvoda (vlada SAD je programe za kodiranje svrstala u istu kategoriju kao i nuklearno oružje). No, taj uslužni program ipak možete instalirati. Samo potražite Web lokaciju: http://www.cs.hut.fi/ssh/.

Osnovno podešena razina je određena u datoteci /etc/inittab, naredbom:

id:3:initdefault:

Pomoću ove naredbe podesit ćete sustav da se pokreće u razini 3 (višekorisnički i višezadaćni način rada). Ispis 9.4 prikazuje primjer datoteke /etc/inittab.

```
Ispis 9.4
               Primjer /datoteke /etc/inittab
# inittab
              This file describes how the INIT process should set up
               the system in, a certain run-level.
#
# Author:
              Miguel van Smoorenburg, <miguels@drinkel.nl.mugnet.org>
#
              Modified for RHS Linux by Marc Ewing and Donnie Barnes
#
# Default run level. The run levels used by RHS are:
# 0 - halt (Do NOT set initdefault to this)
# 1 - Single user mode
# 2 - Multiuser, without NFS (The same as 3, if you do not have networking)
# 3 - Full multiuser mode
# 4 - unused
# 5 - X11
# 6 - reboot (Do NOT set initdefault to this)
#
id:3:initdefault:
# System initialization.
si: :sysinit: /etc/rc.d/rc.sysinit
10:0:wait:/etc/rc.d/rc 0
I1:1:wait:/etc/rc.d/rc 1
I2:2:wait:/etc/rc.d/rc 2
I3:3:wait:/etc/rc.d/rc 3
I4:4:wait:/etc/rc.d/rc 4
I5:5:wait:/etc/rc.d/rc 5
I6:6:wait:/etc/rc.d/rc 6
# Things to run in every run level.
ud: :once: /sbin/update
# Trap CTRL-ALT-DELITE
ca::ctrlaltdel: /sbin/shutdown -t3 -r now
# When our UPS tells us power has failed, assume we have, a few minutes
# of power left. Schedule, a shutdown for 2 minutes from now.
# This does, of course, assume you have powerd installed and your
# UPS connected and working correctly.
pf::powerfail: /sbin/shutdown -f -h +2 "Power Failure; System Shutting Down"
```

nastavlja se

#### Ispis 9.4 Nastavak

# If power was restored before the shutdown kicked in, cancel it. pr:12345:powerokwait: /sbin/shutdown -c "Power Restored; Shutdown Cancelled" # Run gettys in stsndard run levels 1: 12345:respawn: /sbin/mingetty tty1 2: 2345:respawn: /sbin/mingetty tty2 3: 2345:respawn: /sbin/mingetty tty3 4: 2345:respawn: /sbin/mingetty tty4 5: 2345:respawn: /sbin/mingetty tty5 6: 2345:respawn: /sbin/mingetty tty6 # Run xdm in run level 5 x:5:respawn: /usr/bin/X11/xdm -nodeamon

Kao izvornu razinu nemojte određivati niti razinu 0 niti razinu 6 jer će vam onesposobiti sustav. Ako se, iz bilo kojih razloga, ošteti vaša datoteka inittab, moći ćete sustav pokrenuti kao osobno računalo (jedan korisnik) i riješiti problem. Da biste to učinili, u promptu LILO boot unesite parametar "Linux single":

LILO boot: Linux single

LILO je program za učitavanje Linuxa i biti će objašnjen u nastavku poglavlja.

Vidi "Što je Linux Loader?", 214. stranica

#### Pokretanje sustava s diskete

Mnogi ljudi koristite disketu za pokretanje Linuxa. Disketa sadrži kopiju Linux jezgre usmjerene na *root* sustav datoteka (koji se nalazi na odgovarajućoj particiji na tvrdom disku). Instalacijski programi Red Hat i Caldera Linux distribucije vam omogućavaju stvaranja takve diskete tijekom same instalacije sustava.

#### **UPOZORENJE**

Disketu za pokretanje sustava biste trebali načiniti tijekom instalacije, čak i ako namjeravate sustav pokretati sa tvrdog diska. Ako se jave problemi s diskom, onda bi disketa mogla biti jedini način za pokretanje sustava! Osim toga, pokušate li u takvoj situaciji koristiti disketu s drugog računala, ona vjerojatno neće raditi.

U slučaju nužde možete upotrijebiti instalacijske diskete. U *boot promptu* proslijedite jezgri opciju *rescue*. Nakon što odgovorite na par pitanja, sustav će zatražiti da umetnete disketu kako biste dovršili pokretanje sustava.

Nakon pokretanja sustava, on će vam ponuditi minimalnu upravljačku ljusku *ash*, te nekoliko uslužnih programa. Tablica 9.3 nudi popis tih programa, koji bi trebali biti dovoljni za popravak sustava.

Tablica 9.3 Uslu	žni programi za popravak sustava
Program	Opis
cat	Prikazuje sadržaj datoteke
chmod	Mijenja dozvole za pristup datotekama
cpio	Kopira datoteke iz arhiva
e2fsck	Provjerava drugi prošireni sustav datoteka u Linuxu
fdisk	Program za rad s tablicama particija
gzip/gunzip	Komprimira i dekomprimira datoteke
insmod	Instalira module jezgre
l s	Daje popis datoteka u direktoriju
mkdir	Služi za stvaranje direktorija
mke2fs	Stvara drugi prošireni sustav datoteka u Linuxu
mount	Stvara sustav datoteka
rm	Briše datoteku
rmmod	Uklanja modul jezgre

**SAVJET** Kod Intel računala s *root* particijom na IDE tvrdom disku, disketu za pokretanje (*boot*) možete upotrijebiti za pokretanje Linuxa ako upišete sljedeću naredbu:

```
Linux single root=/dev/hda1 initrd=
```

Pazite da odaberete odgovarajući disk za svoj *root* sustav datoteke ako se on nalazi na IDE disku drugačijem od /dev/hda1. Pomoću ove naredbe stvara se *root* particija i odmah možete raditi u sustavu s jednim korisnikom, a tek potom nastaviti s pokretanjem sustava. Nažalost, ovakav postupak ne možete koristiti ako se vaša *root* particija nalazi na SCSI disku.

#### Pokretanje sustava iz Boot Managera

Linux se isporučuje s programom za upravljanje pokretanjem sustava LILO (što je kratica od LInux LOader), koji mijenja glavni *boot* sektor na vašem tvrdom disku i omogućuje vam odabir operativnog sustava koji ćete koristiti.

#### Programi za upravljanje pokretanjem sustava

Uporaba programa za pokretanje sustava ima svoje prednosti i nedostatke. Uz takav program vam ne treba disketa za pokretanje sustava. Također imate mogućnost izbora operativnog sustava tijekom pokretanja računala.

Nedostaci su mu što uvodi još jednu razinu u pokretanju sustava. Program morate mijenjati ili čak ponovno instalirati kada dodajete, brišete ili nadograđujete bilo koji operativni sustav na disku. Program mijenja glavni *boot* zapis na tvrdom disku, stoga ako nešto krene naopako, možda računalo nećete moći podići nikako osim uporabom diskete (a potom ćete morati preformatirati disk). Osim toga, program za pokretanje sustava koji ste odabrali možda neće moći raditi sa svim operativnim sustavima.

Dobro razmislite prije nego se odlučite želite li koristiti disketu ili program za upravljanje pokretanjem sustava.

Program LILO možete podesiti tako da ga možete pokrenuti iz OS/2 programa za upravljanje načinom pokretanja računala.

## Što je Linux Loader?

LILO je program za pokretanje sustava koji je sastavni dio Red hat i Caldera Linux distribucija. Može ga se instalirati u glavni *boot* zapis, na formatiranu disketu ili na super blok *boot* particije za pokretanje sustava OS/2..

Vidi "Instalacija programa LILO", 82. i 101. stranica

Nakon što instalirate program LILO, možete upotrijebiti glavni *boot* zapis za odabir operativnog sustava koji želite koristiti. Ovisno o podešenju programa, LILO će neko vrijeme čekati, a zatim pokrenuti podešeni operativni sustav (*default*).

Lakši način instaliranja programa LILO je da to učinite tijekom instalacije Red Hat ili Caldera Linux distribucije, jer ćete na taj način većinu toga obaviti automatski.

#### UPOZORENJE

Preporučamo vam da program LILO instalirate tijekom instalacije Red Hat ili Caldera distribucije, jer se radi o iznimno opasnom postupku kojim možete oštetiti podatke na tvrdom disku.

#### Podešavanje LILO

Program LILO čita datoteku s podešenjima /etc/lilo.conf i koristi je za određivanje operativnih sustava koji su instalirani na vašem računalu i mjesta na kojem su smješteni potrebni podaci za pokretanje računala. Na početku datoteke /etc/lilo.conf nalaze se neke informacije koje općenito određuju način rada programa LILO. Zatim slijedi nekoliko

sekcija s podacima koji su značajni za pojedine operativne sustave. Program LILO je podešen za pokretanje jedne od sekcija za svaki operativni sustav na vašem računalu.

Slijede dvije sekcije u datoteci s podešenjima:

```
# Section for the Linux partition
image=/vmlinuz
label=Linux
root=/dev/hda1
# Section for MS-DOS
other=/dev/hda3
table=/dev/had
label=msdos
```

Prva sekcija sadrži podatke za operativni sustav Linux. Redak s naredbom image upućuje LILO na mjesto na kojem se nalazi jezgra. Redak s naredbom label, koji se pojavljuje u obje sekcije, sadrži imena operativnih sustava koja ćete ugledati u izborniku tijekom pokretanja sustava. Redak s naredbom root određuje mjesto *root* sustava datoteka.

U dijelu koji se odnosi na MS-DOS naredba other upućuje da se particija dodatnog operativnog sustava nalazi na disku *hda3*. Naredba *table* određuje mjesto na kojem se nalazi tablica particija za disk /dev/hda3.

#### Uporaba programa LILO

Kada instalirate program LILO, obično ćete trebati podesiti vrijeme čekanja i osnovni operativni sustav koji se pokreće ako u zadanom vremenu sami ne odaberete operativni sustav.

Pokrećete li računalo s instaliranim programom LILO, pojavit će se prompt LILO:. U tom trenutku pred vama je nekoliko mogućnosti. Možete čekati neko vrijeme kako bi Linux pokrenuo izvorni operativni sustav ili možete pritisnuti tipke <Ctrl>, <Alt> ili >Shift> da bi se odmah pokrenuo osnovni operativni sustav. Osim toga, možete upisati i ime jednog od operativnih sustava i program LILO će ga učitati. Imate mogućnost pritisnuti tipku <Tab> kako bi program LILO prikazao popis raspoloživih operativnih sustava.

## Isključivanje sustava

Kod računala s operativnim sustavom Linux trebate biti pažljivi kod isključivanja sustava. Ne možete samo isključiti napajanje i otići od računala. Linux sprema određene podatke u memorijske spremnike, te ako jednostavno isključite računalo može doći do ošetećenja u sustavu datoteka.

#### **UPOZORENJE**

Nikada ne smijete isključiti računalo ako pravilno niste napustili Linux, jer je prethodno potrebno uskladiti sustave datoteka. Isključite li sustav napajanja, možete oštetiti sustav datoteka.

Najbolji način da isključite Linux sustav jest da pokrenete naredbu *shutdown*, čiji je oblik ovakav:

/sbin/shutdown [parametri] vrijeme [poruka-upozorenje]

Oznaka [*poruka-upozorenje*] predstavlja poruku koja se šalje svim korisnicima koji su trenutno prijavljeni na mreži, a argument *time* je vrijeme nakon kojeg se sustav gasi (i može biti napisan u više oblika):

- možete upisati apsolutno vrijeme u obliku *hh:mm*, gdje *hh* predstavlja sat (jedna ili dvije znamenke), a *mm* minute (koje trebaju biti opisane s dvije znamenke).
- vrijednost *time* možete upisati i u obliku +m, gdje m označava broj minuta nakon kojih nastaje isključenje sustava. Riječ now možete zamijeniti parametrom +0.

Tablica 9.4	Parametri koji se koriste uz naredbu shutdown
Parametar	Poruka Linuxu
-t sec	Čeka određeni broj sekundi prije nego pošalje upozorenje i prekine sve procese. To vremensko zatezanje ostavlja procesima dovoljno vremena da završe započeti rad.
- k	Ne isključuje sustav već samo šalje upozorenje svim korisnicima.
- r	Ponovno pokretanje nakon isključenja.
- h	Zaustavljanje nakon isključenja.
- n	Ne usklađuje diskove prije ponovnog pokretanja ili zaustavljanja.Ovaj parametar vrlo oprezno koristite jer možete oštetiti podatke na disku.
- f	"Brzo" ponovno pokretanje. Stvara se datoteka /etc/fastboot. Skripta rc provjerava tu datoteku i ne pokreće naredbu fsck.
- C	Prekida postupak isključenja sustava. Uz ovu opciju ne možete postaviti parametar time.

Tablica 9.4 sadrži popis parametara koji se mogu koristiti uz naredbu *shutdown*.

Naredba *shutdown* sprečava bilo kojeg korisnika da se u tom trenu prijavi za rad, zapisuje sve korisnike koji trenutno rade na sustavu, čeka određeno vrijeme i zatim svim procesorima šalje signal SIGTERM. Nakon toga se poziva opcija *halt* ili *reboot*, ovisno o tome koju ste od njih odabrali.

#### UPOZORENJE

Sustav možete ponovno pokrenuti ili zaustaviti ako naredbu reboot ili halt izravno upišete. No, ako upotrijebite neku od navedenih naredbi, korisnici neće dobiti nikakvu poruku i sustav će se odmah isključiti. Ove naredbe koristite jedino u slučaju da ste vi jedini korisnik u sustavu. Da biste vidjeli tko se osim vas prijavio za rad u mreži, pritisnite tipku <w> ili upotrijebite naredbu who.

### Odavde...

Očito je da se radi o više upravljačkog posla nego što je samo pokretanje i isključivanje sustava. Više o upravljanju sustavom potražite u ovim poglavljima:

- 7. poglavlje, "Smisao administratora sustava" nudi kratak opis postupaka i procedura koji se koriste kod podešavanja i održavanja jednog Linux sustava.
- 10. poglavlje, "Upravljanje korisničkim računima" prikazuje kako se dodaju i brišu korisnički računi te kako se njima upravlja.
- 14. poglavlje, "Upravljanje sustavom datoteka" se odnosi na stvaranje i upravljanje sustavom datoteka.
- 18. poglavlje, "Što su Linux ljuske" vas uvodi u pisanje različitih skripti u nekoliko Linux ljuski.

# 

## Upravljanje korisničkim računima

(Steve Burnett)

#### U ovom poglavlju

Rad s korisnicima Rad s grupama korisnika Početni direktoriji Upravljanje sustavom putem Weba

## Rad s korisnicima

Kao administrator sustava ste nadležni za upravljanje korisnicima. Pod izrazom *upravljanje* se podrazumijeva dodavanje korisnika tako da se mogu prijavljivati na mrežu, podešavanja prava korisnika, stvaranje i dodjelu početnih direktorija korisnika, udruživanje korisnika u grupe te brisanje korisnika u slučaju potrebe. U ovom poglavlju ćete naučiti koji se alati i tehnike koriste za upravljanje računima korisnika.

Svaki korisnik bi trebao imati jedinstveno korisničko ime koje omogućuje njegovu identifikaciju i sprečava da neki korisnik obriše tuđe datoteke.

Svaki korisnik mora imati i lozinku. Jedina je iznimka ako na sustavu radi samo jedan korisnik, te računalo nema vezu s drugim računalima niti putem modema niti putem mreže.

Vidi "Zaštita pomoću lozinke", 237. stranica

Ako želite nekome onemogućiti pristup vašem sustavu, morate biti sigurni da se osoba ne može prijaviti za rad. Njezino korisničko ime trebate obrisati zajedno s datotekama koje ostalim korisničma nisu potrebne.

#### Dodavanje korisnika

Ako dodajete novog korisnika, u datoteci /etc/passwd će se pojaviti još jedan zapis u kojem su polja odvojena dvotočkom:

```
login_name:encrypted_password:user_ID:group_ID:user_information:
    login_directory:login_shell
```

Tablica 10.1 sadrži popis polja.

Polje	Opis
login_name	Ime koje se koristi kod prijave
encrypted_password	Lozinka koja je nužna za identifikaciju korisnika, tzv. "prva crta obrane" sigurnosti sustava.
user_ID	Jedinstveni broj kojig operativni sustav koristi za identifikaciju korisnika.
group_ID	Jedinstveni broj koji operativni sustav koristi za identifikaciju primarne grupe pojedinog korisnika. Ako je korisnik član više grupa, onda uz dozvolu administratora može zamijeniti jedno članstvo drugim.
user_information	Opis korisnika (ime i zvanje).
login_directory	Početni direktorij korisnika (u kojem se korisnik počinje s radom nakon prijave).
login_shell	Ljuska koju korisnik koristi nakon što se prijavio za rad (primjerice /bin/bash).

 Tablica 10.1
 Polja u zapisu koji se nalazi u datoteci /etc/passwd

Pomoću naredbe adduser možete dodati korisnika u vaš sustav. Prilikom pokretanja naredbe trebate unijeti ime korisnika kojeg želite dodati. U idućem odlomku ćete naći detaljnije upute o toj naredbi.

Vidi "Čemu služe lozinke u sjeni?", 248. stranica

#### Naredba adduser

Dodajete li u svoj sustav novog korisnika, onda jednostavno pokrenite naredbu *adduser* i upišite ime korisnika kojeg želite dodati (vidi ispis 10.1).

	Ispis 10.1	Primjer dodavanja korisnika
#	./adduser	jschmoe
#		

Naredba *adduser* kopira datoteke čije ime počinje točkom (.) iz direktorija /etc/skel u početni direktorij korisnika. Direktorij etc/skel bi treba sadržavati predloške koji su potrebni svakom korisniku, u što obično spadaju "osobne" datoteke za podešavanje komponenti kao što su .profile,.cshrc, i .login za podešavanje ljuske, .mailrc za e-poštu, .emacs za korisnike koji koriste emacs kao program za pisanje teksta, i slično.

Naredba adduser predstavlja skriptu ljuske Bourne smještenu u direktoriju /usr/sbin. Skriptu možete prilagoditi svojim potrebama ako obavljate dodatne poslove prilikom stvaranja korisničkog računa. Obično se u prompt stavlja puno ime korisnika umjesto da se izvorno ime korisnika zapisuje u datoteku s lozinkom. Ako skriptu niste promijenili tako da se zahtijeva unos imena korisnika, morat ćete je ručno mijenjati pomoću naredbe chfn:

```
# chfn jschmoe
Changing finger information for jschmoe.
Name [RHS Linux User]: Joseph A. Schmoe
Office [ ]:
Office Phone [ ]:
Home Phone [ ]:
Finger information changed.
#
```

Pomoću naredbe *adduser* ne možete podesiti lozinku, već to trebate učiniti pomoću naredbe *passwd*.

#### Dodjela lozinki

Lozinku korisniku dodjeljujete pomoću naredbe *passwd*. Trebali biste svakom korisniku dodijeliti lozinku, a nakon prve prijave korisnik je može promijeniti. Slijede upute za postavljanje lozinke pomoću naredbe *passwd*:

 Upišite naredbu i korisničko ime (primjerice passwd jschmoe) te pritisnite tipku <Return>.

- 2. Na prompt New password unesite lozinku (nećete je vidjeti na ekranu).
- 3. Ponovno upišite istu lozinku:

```
New password (again): novalozinka
```

Sustav je zapamtio lozinku i pohranio je u datoteku /etc/passwd.

Važno je da lozinka udovoljava sljedećim pravilima:

- Treba se sastojati od barem šest znakova (najbolje osam znakova).
- Može sadržavati velika i mala slova te brojeve i interpunkcijske oznake.
- Vidi "Zaštita pomoću lozinke", 237. stranica

Dodajete li veći broj korisnika, vjerojatno ćete poželjeti zadavati kratke i lagane lozinke, ali nemojte to činiti. Dobre lozinke su "prva crta obrane" od neželjenih upada u sustav. Recite svojim korisnicima zbog čega ste odabrali baš takvu lozinku. Nadalje, dobro je redovito mijenjati lozinke, ali svakako korisnike treba naučiti kako se zadaje sigurna lozinka.

Nakon što je korisniku dodijeljena lozinka, njegov zapis će ovako izgledati:

```
jschmoe:Zoie.89&^OgW*:123:21:Joseph, a Schmoe:/users/jschmoe:/bin/bash
```

Drugo polje predstavlja lozinku, ali ne u ovom obliku (lozinka je skrivena).

**BILJEŠKA** Korisnici obično zaborave svoje lozinke. Nemate mogućnosti saznati njihove lozinke i reći im kada ih zaborave. Možete obrisati zaboravljenu lozinku koja se nalazi u datoteci /etc/passwd tako da obrišete drugo polje u zapisu određenog korisnika. Zatim možete podesiti novu korisnikovu lozinku pomoću naredbe *passwd*. Za takve situacije biste trebali ustanoviti određeni postupak i svoje korisnike upoznati s njime.

#### Brisanje korisnika

Postoji nekoliko različiti razina uklanjanja korisnika. Uklanjanje korisnika iz sustava ne predstavlja gotov čin, jer ipak postoji nekoliko mogućnosti:

Ukidanje mogućnosti prijave. Ova mogućnost je korisna ako je korisnik neko vrijeme odsutan i planira se vratiti. Korisnički direktoriji, datoteke i informacije o grupama su ostale sačuvane. Preuredite datoteku s lozinkom (/etc/passwd) i u drugo polje stavite oznaku \*:

jschmoe:\*:123:21:Joseph A. Schmoe:/users/jschmoe:/bin/bash

- Uklanjanje korisnika iz datoteke s lozinkom, ali ne i njegovih datoteka na računalu. Ovaj način je koristan ako drugi korisnici rabe iste datoteke ili ako će nova osoba preuzeti dužnosti prethodnika. Obrišite zapis u datoteci s lozinkom koji se odnosi na starog korisnika, što možete učiniti pomoću programa za uređivanje teksta ili naredbom userdel korisničko\_ime. Nakon toga trebate promijeniti pristup i mjesto korisnikovih datoteka koristeći naredbe chow i mv.
- Ukloniti korisnika i obrisati sve njegove datoteke. Radi se konačnom brisanju korisnika. Morate obrisati zapis koji se odnosi na određenog korisnika i koji se nalazi u datoteci

s lozinkom, te sve korisnikove datoteke na računalu. To možete učiniti koristeći naredbu find:

find user's-home-directory -exec rm { } \;

zatim uklonite direktorij pomoću naredbe rmdir user's-home-directory, a potom trebate ukloniti odgovarajući zapis iz datoteke s lozinkom.

BILJEŠKA Koristite li druge datoteke za podešavanje, kao što su datoteke s alias imenima za e-poštu, morat ćete korisnika obrisati i iz tih datoteka.

#### Rad s grupama korisnika

Svaki korisnik je član neke grupe korisnika. Pojedinim grupama možete dodijeliti različite resurse i ovlasti. Evo primjera, grupi korisnika koji mrežu koriste za analizu prodaje proizvoda razumno je dati pristup jednoj skupini datoteka, a korisnicima koji se bave razvojem novih proizvoda možete dodijeliti pristup drugim datotekama.

Datoteka s lozinkom sadrži informacije o pojedinim korisnicima. Podaci o grupama se nalaze u datoteci /etc/group. Evo jedno zapisa iz te datoteke:

```
sales:21:tuser, jschmoe, staplr
```

U navedenom primjeru se grupa zove sales, identifikacijski broj grupe je 21, a njezini članovi su tuser, jschmoe i staplr. Datoteke i direktoriji imaju dozvolu pristupa vlasniku, grupi i drugima. Korisnik može biti član više grupa, a može i mijenjati članstvo u grupama.

#### **Dodavanje grupe**

Novu grupu dodajete tako da preuredite datoteku /etc/group, odnosno upišete podatke o novoj grupi.

Svaka grupa u datoteci /etc/group ima jedinstveni broj grupe. U Linuxu se grupe vode po brojevima, a ne po imenima, tako da ako dvjema grupama dodijelite isti broj, sustav će ih tretirati kao jednu grupu.

#### **Brisanje grupe**

Grupu brišete tako da uklonite njezin zapis iz datoteke /etc/group, a sve datoteke možete pridijeliti drugoj grupi pomoću naredbe find:

find / -gid group-id find user's-home-directory -exec chgrp newgroup { } \;

## Početni direktoriji

Namjeravate li u svom sustavu imati veći broj korisnika, trebali biste logički grupirati početne direktorije. Općenito trebate postaviti sve početne direktorije ispod jednog "krovnog" direktorija. Na taj način će direktoriji biti grupirani bez obzira na to što ćete kasnije s njima učiniti.

Vidi "Stvaranje i razgradnja sustava datoteka", 269. stranica

Primjerice, možete odrediti da direktorij /home bude naddirektorij svim korisničkim direktorijima, a zatim u poddirektorijima možete grupirati korisnike prema odjelima. Korisnici iz prodaje bi trebali imati račune u direktoriju /home/prodaja, oni iz razvoja u direktoriju home/razvoj i tako dalje. U idućoj razini direktorija bi se trebali naći početni direktoriji. S obzirom da korisnički direktoriji znaju zauzeti dosta mjesta na disku, možda biste trebali razmisliti o smještanju raznih grupa korisnika na odvojene diskove. Treba li vam dodatnog prostora na disku, možete jednostavno stvoriti dodatnu kategoriju početnih direktorija i ugraditi je u sustav datoteka kao mount point u direktoriju /home.

#### Upravljanje sustavom putem Weba

Red Hat distribucija Linuxa sadrži alat Jacquesa Gelinasa za administratore sustava koji se naziva Linuxconf i koji vam omogućuje obavljanje mnogih administratorskih zadaća, uključujući i rad s korisnicima i grupama. Osim poznatog tekstualnog unosa naredbi i pristupa putem XWindowsa, Linuxconf podržava upravljanje Linux sustavom putem World Wide Weba. Ako ste u programu Linuxconf potvrdili opciju "Linuxconf HTML Access Control", možete unijeti URL adresu http://<hostname>:98/ i pojavit će se Web stranica alata Linuxconf.

BILJEŠKA Želite li raditi bilo što vezano uz sustav, morat ćete otići na jednu od podstranica, a zatim ćete trebati unijeti lozinku superkorisnika.

### Odavde...

Kao administrator sustava odgovorni ste za upravljanje sustavom i potporu korisnicima koji se prijavljuju za rad na vašem sustavu. Odgovarajući program za upravljanje korisnicima pomaže da jednostavno dodajete i brišete korisnike. Linux sadrži kompletne alate za rad s korisničkim računima i grupama korisnika. Važno je da korisnički računi budu logično grupirani.

Podešavanjem strukture direktorija tako da odražava

Više informacija o upravljanju sustavom potražite u idućim poglavljima:

- 7. poglavlje, "Smisao administratora sustava" predstavlja uvod u zadaće administratora sustava.
- 11. poglavlje, "Sigurnosna pohrana podataka" objašnjava kako treba planirati sigurnosnu pohranu podataka.
- U 14. poglavlju, "Upravljanje sustavom datoteka" je opisan način podešavanja i upravljanja sustavima datoteka na Linux računalima.

# POGLAVLJE **1**

## Sigurnosna pohrana podataka

(Jack Tackett)

#### U ovom poglavlju

Pravila za sigurnosnu pohranu podataka 226 Savjeti za sigurnosnu pohranu podataka 227 Raspored sigurnosne pohrane podataka 227 Sigurnosna pohrana podataka i obnova datoteka 229 Različiti problemi mogu prouzročiti gubitak podataka, odnosno greškom možete obrisati datoteke tako da pojedine hardverske komponente neće raditi, a nećete moći niti pristupiti važnim podacima koji su spremljeni u tim datotekama. U takvim je slučajevima jako korisno imati pohranjene podatke na sigurnom mjestu.

Budućnost vaše tvrtke, a i vaša budućnost u njoj, može ovisiti o sigurnosnoj pohrani podataka. U takvim situacijama ćete i vi i ostali korisnici biti vrlo zadovoljni ako ste prethodno odvojili malo vremena i truda za kopiranje datoteka na određene medije za spremanje podataka. Sigurnosna pohrana datoteka nije neki osobiti posao, ali nema administratora sustava koji ga može izbjeći.

### Pravila za sigurnosnu pohranu podataka

Slijedi nekoliko pravila za sigurnosnu pohranu podataka:

- Potpuna ili djelomična pohrana podataka. Kod potpune pohrane spremaju se sve datoteke. Je li to potrebno svakodnevno činiti? Takva vrsta pohrane podataka obično zahtijeva dosta vremena i znatnu količinu medija za pohranu (traka, disketa i slično). Djelomična pohrana podataka kopira samo one datoteke koje su promijenjene u odnosu na zadnju pohranu.
- Pohrana sustava datoteka. Naravno da stalno trebate spremati sustave datoteka, a drugi podaci se mogu povremeno spremati. Provjerite imate li najnovije kopije svih sustava datoteka.
- Vrsta medija za sigurnosnu pohranu. Ovisno o uređajima koje koristite u svom računalu, možda ćete trebati upotrebljavati vrpcu s devet traka, traku sa spremnikom ¼ incha., trake 4 mm ili 8 mm (DAT) ili diskete. Svaki od navedenih medija ima određene prednosti glede kapaciteta i cijene uređaja i medija. Odaberite medij koji najviše odgovara vašim mogućnostima, ali imajte na umu da najjeftiniji mediji mogu potrošiti najviše vašeg vremena.
- Sigurnosna pohrana podataka utječe na korisnike sustava. Tijekom pohrane podataka korisnici se ne mogu prijavljivati za rad na sustavu. Hoće li ih takva pojava uznemiriti? Osim toga, datoteke koje su korisnici mijenjali tijekom pohrane podataka se mogu izgubiti što je osobito nezgodno ako ste, primjerice, radili na bazi važnih podataka. Ne bi li se pohrana podataka trebala obavljati kada niti jedan korisnik nije prijavljen za rad?
- Naredbe koje se koriste za sigurnosnu pohranu. Postoje relativno jednostavne i brze naredbe za sigurnosnu pohranu podataka kao što su naredbe tar i cpio. Jesu li one dovoljne za taj posao?
- Dokumentacija za pohranjene datoteke. Sve medije (trake i diskete) trebate pravilno označiti tako da ih možete upotrebljavati za povrat podataka kada to bude potrebno. Neki postupci i naredba omogućuju stvaranje sadržaja ili popisa pohranjenih podataka.

S gledišta administratora sustava, sustav datoteka treba automatski pohraniti, uz što manji utjecaj administratora. To bi trebalo učiniti kada je prijavljen mali broj korisnika, tako da pohranjeni podaci budu što je moguće potpuniji. Kod ovakvog razmišljanja treba u obzir

uzeti učinak i njegovu vrijednost. Treba li administrator ili operator ostati na radnom mjestu sve do ponoći kako bi pohrana podataka bila potpuna? Pametnije je potrošiti 2000 dolara, koliko košta pogon za DAT traku, tako da se pohrana podataka može obaviti u 3 sata noću (bez intervencije operatora).

Razmotrite sve mogućnosti, procijenite troškove i donesite pravu odluku. Općenito je puno jednostavnije i jeftinije obnoviti pohranjene podatke nego ih ponovno stvarati.

## Savjeti za sigurnosnu pohranu podataka

Cilj sigurnosne pohrane podataka je da što brže i što jednostavnije možete obnoviti pojedine datoteke ili čitave sustave datoteka. Što god radili navedeno će ostati glavni zadatak.

Načinite plan sigurnosne pohrane podataka. Odredite datoteke koje će biti pohranjene, učestalost spremanja datoteka i način njihovog obnavljanja. Svi korisnici trebaju znati raspored sigurnosne pohrane podataka. Isto tako korisnike treba upoznati s načinom obnove pojedinih datoteka.

Provjeravajte pohranjene podatke, odnosno čitajte sadržaj spremljenih podataka nakon pohrane i nakon obnove datoteka. Upamtite da mediji za pohranu, diskete ili trake, mogu imati prekide.

Pohranu podataka načinite tako da datoteke možete obnoviti na bilo kojem sustavu datoteka ili bilo kojem računalu. Koristite uslužne programe za sigurnosnu pohranu i arhiviranje podataka koji se mogu koristiti na bilo kojem Linux ili UNIX računalu.

Označavajte medij na koji ste pohranili podatke (trake, diskete i slično). Koristite li više traka ili disketa, pravilno ih označite kako biste kasnije mogli utvrditi njihov redoslijed.

Načinite kopije datoteka koje se nalaze na vašem računalu kako biste ih sve mogli obnoviti u razumnom vremenu. Pohranite kopije na diskete ili trake. Slijedi nešto vrlo važno! Barem jednu kopiju pohranjenih podataka biste trebali spremiti izvan svog sustava. U slučaju nepogode (primjerice vatre) možete se desiti da vaše računalo bude potpuno uništeno. Mnoge tvrtke iznajmljuju prostor za čuvanje njihovih traka i disketa. Na isto mjesto trebate pohraniti kompletan popis hardvera kako biste mogli ponovno naručiti iste komponente, ako to bude potrebno.

U određenim vremenskim razdobljima ponavljajte postupak sigurnosne pohrane podataka.

Postoji nekoliko alata koji vam pomažu da automatizirate sigurnosnu pohranu podataka. Potražite arhive na adresi **sunsite.unc.edu** i saznajte nešto više o tome. Osim toga, Linux podržava i FTPE format koji vam omogućava pohranu podataka na magnetskim trakama QIC-80. Detaljnije upute potražite u datoteci FTAPE HOWTO.

#### Raspored sigurnosne pohrane podataka

Važno je odrediti raspored sigurnosne pohrane podataka tako da odgovara vašim potrebama i da omogućuje korištenje najnovijih kopija datoteka. Nakon što utvrdite raspored, potrebno je pridržavati ga se.

#### 228 11. poglavlje Sigurnosna pohrana podataka

Idealno bi bilo obnoviti bilo koju datoteku u bilo koje vrijeme. Osim tog krajnjeg zahtjeva, koji nije moguće ostvariti, postoji i drugi, a taj je svakodnevna obnova datoteka. Da biste to ostvarili, trebate koristiti kombinaciju potpune pohrane podataka i pohrane kod koje se spremaju promijenjene datoteke (u odnosu na prethodnu pohranu). *Potpuna sigurnosna pohrana podataka* predstavlja pohranu svih datoteka na računalu, a kod *sigurnosne pohrane promijenjenih datoteka* spremaju se samo one datoteke koje su promijenjene od prethodne pohrane. Postoje dvije razine pohrane promijenjenih datoteke: spremanje onih datoteka koje su promijenjene u odnosu na prethodnu *potpunu* ili na posljednju *djelomičnu* pohranu. Sigurnosnu pohranu podataka možete zamisliti u tri razine:

Razina 0:	Potpuna pohrana podataka
Razina 1:	Pohrana datoteka koje su promijenjene u odnosu zadnju potpunu pohranu
Razina 2:	Pohrana datoteka koje su promijenjene u odnosu zadnju pohranu razine 1

Slijedi nekoliko primjera rasporeda za sigurnosnu pohranu podataka:

Jedan dan potpuna pohrana, a drugi djelomična

Prvi dan	razina 0, potpuna pohrana
Drugi dan	razina 1, djelomična pohrana
Treći dan	razina 1, djelomična pohrana
Četvrti dan	razina 1, djelomična pohrana
Peti dan	razina 1, djelomična pohrana

Jednom mjesečno potpuna, jednom tjedno djelomična i dnevno djelomična pohrana (u primjeru je odabran utorak, ali može biti bilo koji dan u tjednu).

Prvi utorak	Razina 0, potpuna pohrana
Bilo koji drugi utorak	Razina 1, djelomična pohrana
Bilo koji drugi dan	Razina 2, djelomična pohrana

Da biste prema tom rasporedu obnovili pojedinu datoteku, potrebno je načiniti potpunu pohranu (ako se datoteka nije mijenjala tijekom mjeseca), djelomičnu pohranu - razina 1 (ako je datoteka mijenjana prošli tjedan) ili djelomičnu pohranu – razina 2 (ako je datoteka mijenjana ovaj tjedan). Raspored je puno složeniji od onoga u prvom primjeru, ali dnevna pohrana oduzima manje vremena.

U slučaju da želite obnoviti starije podatke (stariju inačicu datoteke), onda kopiju pohranjenih datoteka trebate čuvati duže vrijeme. Obično se jedna tjedna potpuna pohrana koristi u tri iduća tjedna. Ako se raspored radi za vremensko razdoblje duže od četiri tjedna, onda se jednom u dva tjedna načini potpuna pohrana koja se koristi u iduća dva i pol mjeseca (kvartalno razdoblje).

## Sigurnosna pohrana podataka i obnova datoteka

Postoji nekoliko različitih uslužnih programa koji služe za sigurnosnu pohranu podataka i kasniju obnovu podataka na Linux računalima. Neki od njih su vrlo jednostavni, a drugi nešto složeniji. Jednostavnija rješenja imaju svoja ograničenja, tako da trebate odabrati ono koje vam najviše odgovara.

S obzirom da se radi o vrlo važnoj zadaći, postoji nekoliko softvera koji služe isključivo za pohranu i povrat podataka. U narednim odlomcima ćete upoznati dva takva softvera:

- Program tar možete koristiti na svim Linux ili UNIX računalima. Lako se koristi, a potrebno vam je samo nekoliko traka ili disketa.
- Uslužni program cpio predstavlja softver opće namjene koji služi za kopiranje datoteka. Program se lako koristi i puno je robusniji od programa tar.

#### Uslužni program tar

Program *tar* je izvorno zamišljen za stvaranje arhive na trakama, to jest za kopiranje datoteka ili direktorija na traku i kasniju njihovu obnovu iz arhive. Program možete koristiti za kopiranje na bilo koji uređaj. Evo koje su prednosti programa:

- jednostavno se koristi
- program je pouzdan i stabilan
- arhive se mogu pročitati na gotovo svakom Linux/UNIX sustavu.

Nedostaci su:

- Neke inačice programa tar zahtijevaju da arhiva mora stati na jedan disk ili traku, što znači da se svi podaci se mogu izgubiti ako je dio medija neispravan (loši sektori na disketi ili loši blokovi na traci).
- Program tar ne može pohraniti specijalne datoteke (primjerice datoteke pojedinih uređaja).
- Program može obaviti samo potpunu pohranu podataka. Ako vam je potrebna djelomična pohrana, trebate se upustiti u programiranje ljuske.
- Vidi "Rad sa skriptama", 365. stranica

Tablica 11. prikazuje popis opcija koje se najčešće koriste s programom tar. Postoje i drugi parametri koje možete upotrijebiti, a njihov popis potražite na stranici za pomoć (man).

Tablica 11.1	Najčešće opcije koje se koriste u programu tar	
Орсіја	Opis	
С	Stvaranje arhive.	
Х	Raspakiranje ili povrat podataka koji su pohranjeni na izvornom uređaju za pohranu ili uređaja koji je određen uporabom opcije f.	

nastavlja se

Tablica 11.1	Nastavak
Орсіја	Opis
f ime	Stvaranje arhive ili čitanje arhive s uređaja ime (parametar ime predstavlja ime datoteke ili uređaja koji je odabran u direktoriju /dev, primjerice uređaj /dev/rmt0).
Z	Komprimiranje ili dekomprimiranje arhive.
Z	Komprimiranje ili dekomprimiranje arhive pomoću programa gzip.
М	Stvaranje sigurnosne kopije na više disketa ili traka.
t	Stvaranje kazala svih datoteka koje su pohranjene u arhivi.
V	Uporaba načina rada verbose (opširan način rada).

Pogledajte nekoliko primjera korištenja programa tar za pohranu i povrat datoteka. Pomoću iduće naredbe možete kopirati direktorij /home na disketu /dev/fd0:

tar -cf /dev/fd0 /home

U ovom slučaju opcija f određuje da se arhiva stvara na disketnom pogonu /dev/fd0.

Iduća naredba također služi za arhiviranje direktorija /home:

tar -cvfzM /dev/fd0 /home | tee homeindex

Opcija v upućuje na opširan način rada, opcija z služi za komprimiranje arhive kako bi zauzimala manje prostora na disku, opcija M upućuje program tar na sigurnosnu pohranu na više disketa. nakon što se jedna disketa napuni, program će zatražiti da umetnete iduću. Popis kopiranih datoteka se nalazi u datoteci homeindex. Zgodno je otvoriti tu datoteku i pogledati što je sve arhivirano.

Naredba find je korisna za traženje datoteka koje trebate mijenjati u nekom vremenskom razdoblju (ta datoteka ulazi u raspored povremene sigurnosne pohrane podataka). U idućem primjeru možete vidjeti uporabu naredbe find prilikom stvaranja popisa svih datoteka koje su promijenjene tijekom zadnjeg dana:

Da biste taj popis upotrijebili kao ulaz za naredbu tar, u navodnicima upišite naredbu cat bkuplst jer ćete na taj način uputiti ljusku da naredbu izvršava kao podljusku, a izlaz naredbe će zamijeniti originalnu naredbu u navodnicima.

Pomoću iduće naredbe možete obnoviti datoteku /home/dave/notes.txt s uređaja /dev/fd0 (uočite da je potrebno upisati čitavo ime datoteka kada je želite obnoviti):

tar xv /usr2/dave/notes.txt

**SAVJET** Bilo koju od navedenih naredbi možete automatizirati ako je spremite u *root* datoteku crontab. Primjerice, u datoteku crontab možete upisati sljedeći redak (kako bi se sigurnosna pohrana podataka obavila svaku noć u 1 sat i 30 minuta):

01 \* \* \* tar cvtz /def/fd0 /home > homeindex

Ako su vam potrebne složenije pohrane podataka, onda ćete trebati stvoriti skripte koje služe za upravljanje sigurnosnom pohranom podataka. Te skripte možete pokretati preko datoteke cron.

Vidi "Raspored pokretanja naredbi u datoteci cron i crontab", 388. stranica

Naredbu tar možete koristiti za stvaranje arhiva u Linux sustavu datoteka, umjesto da arhive spremate na neki od uređaja za pohranu. Na taj način grupu datoteka, kao i direktorij u kojem se nalaze možete pohraniti u jednu datoteku. Da biste to učinili, jednostavno mjesto imena uređaja kao parametar upišite ime datoteke. Slijedi primjer arhiviranja direktorija i njegovih poddirektorija:

tar cvf /home/backup.tar /home/dave

Pomoću ove naredbe stvorit ćete datoteku /home/backup.tar koja sadrži sigurnosnu kopiju direktorija /home/dave i svih datoteka i poddirektorija koji se u njemu nalaze.

**BILJEŠKA** Naredba tar izvorno ne komprimira datoteke. Da biste komprimirali datoteke, koristite opciju z ili poseban program za komprimiranje kao što je program gzip.

Ako program tar koristite za stvaranje arhive, dobro je na početak arhive postaviti direktorij, jer će na taj način, kod povrata podataka, sve datoteke biti smještene u određenom direktoriju koji se nalazi unutar tekućeg radnog direktorija. U protivnom, ćete u svom direktoriju naći stotine datoteka koje su kopirane iz arhive.

Pretpostavimo da se trenutno nalazite u naddirektoriju direktorija data, koji sadrži nekoliko stotina datoteka. Postoje dva načina za stvaranje tar datoteke toga direktorija. Možete promijeniti direktoriju u direktorij data i na tom mjestu stvoriti arhivu (kao u sljedećem primjeru):

```
$ pwd
/home/dave
$ cd data
$ pwd
/home/dave/data
$ tar cvf ../data.tar *
```

Na taj ste način stvorili datoteku u direktoriju /home/dave koja sadrži sve datoteke iz direktorija data, ali ne i sami direktorij. Budete li raspakirali ovu tar datoteku, neće se stvoriti direktorij, već će nekoliko stotina datoteka biti raspakirano u direktorij u kojem se tog trenutka budete nalazili.

Drugi način je stvaranje datoteke tar u naddirektoriju te pohranu i samog direktorija zajedno s podacima. Evo naredbe kojom to možete postići:

```
$ pwd
/home/dave
$ tar cvf data.tar data
```

Na taj ćete način stvoriti arhivu direktorija s podacima, s time da će se direktorij naći na prvom mjestu u arhivi. Na taj način, prilikom povrata podataka, prvo će nastati direktorij, a sve datoteke će biti smještene u poddirektoriju.

BILJEŠKA Želite li stvoriti tar datoteku sa svim datotekama unutar direktorija, dobro je odrediti drugu lokaciju za tar datoteku (a ne je pohraniti u tekući direktorij). Na taj način, ako budete htjeli pohraniti sve datoteke iz tekućeg direktorija, neće doći do zabune ( program tar neće pokušati dodati samu tar datoteku u arhivu).

#### Naredba cpio

Naredba cpio se koristi za kopiranje datoteka-arhiva i možete je koristiti za stvaranje sigurnosnih kopija (opcija -o) ili obnavljanje datoteka (opcija -i).

Prednosti naredbe cpio se sastoje u sljedećem:

- Može načiniti sigurnosnu kopiju bilo koje vrste datoteka.
- Može načiniti sigurnosnu kopiju specijalnih datoteka.
- Efikasnije sprema podatke nego što to čini program tar.
- Prilikom obnove podataka zanemaruje loše sektore i blokove.
- Zaštitne kopije je moguće obnoviti na gotovo svim Linux i UNIX računalima.

Nekim se korisnicima čini da je sintaksa naredbe cpio kompliciranija od sintakse naredbe tar. Osim toga, trebat ćete se upustiti i u programiranje ljuske kako biste stvorili zaštitne kopije (incremental backups).

Tablica 11.2 prikazuje najčešće korištene opcije naredbe cpio. Kompletan opis opcija možete potražiti na man stranici naredbe cpio.

Tablica 11.2		Opcije koje se najčešće koriste uz naredbu cpio.	
	Орсіја	Opis	
	- 0	Stvara arhivu na standardnom izlazu.	
	- B	Blokira ulaz i izlaz na 5,120 bajta po zapisu. Opcija služi za pohranu podataka na magnetsku traku.	
	- i	Dekomprimira datoteke sa standardnog ulaza i obično se koristi nakon što ste drugom naredbom cpio stvorili arhivu na standardnom ulazu.	
	- t	Stvara sadržaj ulaza.	

U nastavku ćete vidjeti nekoliko primjera koji pokazuju kako se koristi naredba cpio za pohranu i obnavljanje datoteka:

Pomoću ove naredbe kopiraju se datoteke iz direktorija /home na uređaj /dev/fd0:

ls /home | cpio -o > /dev/fd0

Pomoću ove naredbe možete raspakirati datoteke s uređaja /dev/fd0 i stvoriti kazalo u datoteci bkup.indx:

```
cpio -it < /dev/fd0 > bkup.indx
```

Slijedi primjer uporabe naredbe find za stvaranje popisa svih datoteka u direktoriju /home koje su mijenjane tijekom tekućeg dana:

find /home -mtime 1 -type f -print | cpio -oB > /dev/fd0

Izlaz naredbe find je iskorišten u naredbi cpio za stvaranje arhive na uređaju /dev/fd0, na kojem su podaci spremljeni u obliku 5,120 bajta po zapisu.

Putem ove naredbe možete obnoviti datoteku /home/dave/notes.txt s uređaja /dev/fd0: echo "/home/dave/notes.txt" | cpio -i < /dev/fd0.</p>

Da biste datoteku raspakirali pomoću naredbe cpio, morate navesti puno ime datoteke.

#### BILJEŠKA

Svaku od navedenih naredbi možete automatizirati ako ih smjestite u datoteku

#### SAVJET crontab

koja se nalazi u *rootu*. Primjerice, možete upisati slijedeće u root datoteku cron (svaki dan će podaci biti pohranjeni u 01:30, na direktorij /home):

01 \* \* \* ls /home | cpio -o > /dev/fd0

Ako trebate obaviti nešto zahtjevniju pohranu, možete načiniti i skripte koje upravljaju spremanjem podataka. Te skripte također trebate pokretati preko datoteke cron.

#### Odavdje..

- 7. poglavlje, "Smisao administratora sustava" nudi kratak opis dužnosti administratora Linux sustava.
- 10. poglavlje, "Upravljanje korisničkim računima" prikazuje kako se dodaju, brišu korisnički računi te kako se njima upravlja.
- 14. poglavlje, "Upravljanje sustavom datoteka", nudi prikaz stvaranja, nadograđivanja i uporabe sustava datoteka pod Linuxom.

# poglavlje 12

## Poboljšanje sigurnosti sustava

(Steve Burnett)

#### U ovom poglavlju

Fizička zaštita sustava 236 Zaštita pomoću lozinke 237 Zaštita korisničkih računa 238 Zaštita datoteka 240 Izbjegavajte pretjeranu uslužnost 242 Praćenje uporabe naredbe su 243 Razvoj sigurnosnog sustava 243 PAM - Pluggable Authentication Modules 246 Čemu služe lozinke u sjeni 248
Ako računalo niste zaključali u ormar s ključem koji stalno držite oko vrata, sigurno ćete se susresti s problemom sigurnosti sustava. To je stvarnost, a ne šala. Postoji li više korisnika, ako je računalo povezano s ostalim svijetom putem mreže ili modema ili ako postoji vrijeme kada računalo nije pod nadzorom, onda sigurno postoji i opasnost da netko neovlašten pristupi vašem računalu.

Ponekad neovlašteni pristup nije opasan, no i u tom slučaju će vas sigurno iznervirati. Ako netko ima vremena "šarati" po vašem računalu, tada sigurno ima i mogućnost kopirati podatke koje smatrate strogo povjerljivima ili može neovlašteno koristiti vaše resurse, odnosno mijenjati ili brisati podatke.

U većini organizacija administratori sustava su odgovorni za sigurnost sustava. Zaštita sustava vas ne bi trebala zabrinjavati, ali biste svakako trebali poduzeti potrebne korake i zaštititi sustav. Kod takvih stvari se treba ponašati odgovorno i profesionalno.

U ovom poglavlju ćete upoznati metode za pojačanu sigurnost sustava, od kojih su neke metode beskorisne za kućna računala i odnose se na velike sustave. No, neke druge metode opisane u ovom poglavlju su vrlo prihvatljive i onima koji računalo koriste i kod kuće.

**BILJEŠKA** U zadnjih nekoliko godina masovni mediji su promijenili smisao riječi *haker* od "računalni zanesenjak" do "netko tko upada u tuđe sustave". Među računalcima je uvriježen izraz *kraker* za nekoga tko neovlašteno upada u tuđa računala, tako da ćemo i mi koristiti taj izraz.

### Fizička zaštita sustava

U masovnim medijima se puno govori i piše o virusima, ispadima računalnih sustava i *krakerima* koji putem modemskih i mrežnih veza nasilno ulaze u sustave, ali se zaboravlja na fizičku sigurnost sustava koji je izrazito osjetljiv na različite uvjete u kojima radi.

Vatra i dim mogu brzo uništiti informatičku opremu. U prostorijama u kojima se nalazi računalna oprema trebali bi biti ugrađeni protupožarni detektori

Uz vatru i dim, prašina je veliki neprijatelj informatičke opreme. Prašina skraćuje životnu dob magnetskih medija za pohranu podataka kao i optičkih uređaja. Ona se može skupljati na ventilatoru, koji ima svako računalo, te smanjiti protok zraka što je uzrok pregrijavanja računala. Osim toga, prašina je i električki vodljiva tako da može prouzročiti neželjene spojeve na raznim pločama.

Elektricitet predstavlja posebnu temu kada je riječ o fizičkoj sigurnosti informatičke opreme. Računala su vrlo osjetljiva na prenapone koji se pojavljuju u električnoj mreži. Sva oprema bi trebala biti zaštićena od prenapona koji dolaze iz električne mreže. To se odnosi i na modeme spojene na telefonsku liniju. Mnoga područja u svijetu imaju nestabilan napon što može prouzročiti štete na opremi. **BILJEŠKA** lako odvodnici prenapona mogu zaštiti opremu od prenapona iz električne mreže, oni su gotovo sasvim nemoćni kada su u pitanju atmosferska pražnjenja. Ako grom pogodi dalekovod koji napajanje dovodi do vašeg stana ili ureda, jednostavni odvodnik prenapona neće moći zaštiti vašu opremu. U slučaju oluje je najbolje isključiti napajanje iz zida i čekati da ona prođe.

Računala su česta meta lopova. Ona su obično mala i skupa tako da ih se lako može ukrasti i prodati. Dobro razmislite koliko su vaša računala sigurna na mjestima gdje se nalaze, te ih pokušajte zaštiti od krađe.

Idući aspekt sigurnosti računala je sprečavanje pristupa neovlaštenim osobama. Ako bilo tko može ušetati u sobu u kojoj se nalazi računalo, sjesti za tipkovnicu i početi raditi, a da pri tome nije imao baš nikakve zapreke, onda to predstavlja ozbiljan problem. Jer ako kontrolirate pristup vašem računalu, otežat ćete "posao" nekome tko ima namjeru ukrasti ili oštetiti podatke ili opremu. Zbog toga je potrebno odrediti dozvole pristupa računalima i naučiti korisnike da se pridržavaju pravila rada sa sustavom.

Evo nekih koraka koje trebate poduzeti da biste osigurali fizičku sigurnost sustava:

- Nikada ne ostavljajte dulje vremena bez nadzora sustav, pogon za trake i diskete, terminale ili radne stanice. Dobro bi bilo uvesti određena ograničenja pristupa u sobe u kojima se nalazi vaš glavni sustav zajedno s pogonima za diskete i trake. Zaključavajte vrata u slučaju da namjeravate dulje vremena biti odsutni. Neovlaštena osoba može uzeti medije za sigurnosnu pohranu podataka.
- Nikada ne ostavljajte konzolu ili terminal bez nadzora ako ste se prijavili za rad kao superkorisnik. Poznaju li korisnici sustav, lako će moći prava superkorisnika pridijeliti sebi, promijeniti važan softver ili obrisati neke podatke.
- Upoznajte korisnike sustava s važnošću fizičke sigurnosti sustava. Potaknite ih da vam odmah prijave nepravilnosti koje uoče. Slobodno odstranite bilo koga nepoznatog sa sustava.
- Ako je moguće, nemojte važne podatke ostavljati na računalima koja imaju modemske ili mrežne veze.
- Sigurnosne kopije držite na sigurnom mjestu i ograničite broj osoba koji ih može koristiti.

### Zaštita pomoću lozinke

Prva "crta obrane" od neovlaštenog pristupa sustavu je zaštita pomoću lozinke, ali je lozinka ponekad najlakši put u nepristupačni sustav. U ovom odlomku su objašnjeni neki koraci koje trebate poduzeti kako biste sačuvali tajnost lozinki.

Činjenica je da ljudi vole koristiti jednostavne lozinke koje se lako pamte. Ne žele mijenjati lozinke. Obično ih zapisuju kako ih ne bi zaboravili. Na nesreću vas, administratora sustava, sve su to negativne stvari glede sigurnosti sustava. Sigurnost lozinki zahtijeva gotovo neprestanu pozornost.

Lozinka superkorisnika (*root*) je posebna lozinka. Svatko tko je poznaje može pristupiti svemu na vašem sustavu i možda čak i na druge sustave s kojima je vaše računalo povezano putem mreže. Često mijenjajte lozinku superkorisnika. U većini organizacija je najbolje da dvojica znaju tu lozinku (ali ne i više od toga).

Lozinke bi trebale biti dužine od barem šest znakova, a samo prvih osam znakova svake lozinke je prepoznatljivo. Ako program za otkrivanje lozinke nasumce pogađa neku lozinku, teže će je otkriti ako je ona dulja.

Računala su vrlo zgodna za ponavljanje stvari, kao što je, primjerice, kodiranje riječi iz rječnika i uspoređivanje s vašom lozinkom s ciljem provale u sustav. Nikada nemojte odabrati lozinku koja se nalazi u nekom od rječnika. Nastojite ne koristiti lozinke koje se lako povezuju s vama. Ime, adresa, nadimak, ime djeteta, kućnog ljubimca, telefonski broj, broj vozačke dozvole i slično ne predstavljaju dobar odabir za lozinku.

Pa kako ćete odabrati dobru lozinku, ako je poznato da se jednostavne lozinke jednostavno i otkrivaju? Jedna od tehnika je da odaberete dvije kratke riječi i povežete ih interpunkcijskom oznakom. Na taj ćete način dobiti seriju slučajnih znakova, koje je lako zapamtiti, ali teško slučajno pogoditi. Slijedi nekoliko primjera takvi lozinki:

ivo&dan auto!san lav:plav

Drugi način je da lozinku stvorite od prvih slova fraze koju možete lako zapamtiti. Opet ćete dobiti niz slučajnih znakova, ali ćete ga lako zapamtiti. Evo primjera: fraza "Dame i gospodo, Elvis je napustio zgradu", pretvara se u lozinku D&gEjnz.

Važno je da lozinku možete zapamtiti. Nigdje je ne treba zapisivati. Ako korisnici žele svakako zapisati svoje lozinke, dajte im savjet da to učine tako da lozinku upišu neki popis ili rečenicu. Primjerice, ako ste kao lozinku odabrali auto!san, onda na mali komad papira zapišite rečenicu: "Kupit ću novi auto! San mi je voziti Ferarija". Lozinka je dobro sakrivena u ovoj rečenici.

# Zaštita korisničkih računa

Svaki račun na vašem Linux sustavu predstavlja "vrata" za ulazak na vaše računalo. No da bi netko "ušao kroz ta vrata" potreban mu je pravi "ključ" to jest lozinka. Ako ste ustanovili dobar sustav zadavanja lozinki, to je već dobar početak da načinite siguran sustav. Jedan od aspekata sigurnosti računala, to jest zaštita korisničkih računa, usko je povezana sa zaštitom lozinki.

Zaštita korisničkih računa se sastoji u traženju računa na sustavu koji mogu ugrožavati sigurnost sustava te u rješavanju nekoliko vrsta problema koji su vezani uz takve račune.

### Računi bez lozinke

Mnogi *krakeri* nasilno ulaze u sustave tako da jednostavno traže račun koji nije zaštićen lozinkom. Povremeno trebate provjeravati datoteku s lozinkama i tražiti takvu vrstu računa. Ako ih nađete, obavezno ih zatvorite. Lozinka je pohranjena u drugom polju unutar datoteke s lozinkama. Prazna polja za upis lozinke možete potražiti koristeći neki od alata, kao što su grep, awk ili perl. Računu bez lozinke možete onemogućiti pristup tako da u datoteci s lozinkama, u polju za lozinku upišete zvjezdicu (\*). Na taj ćete način spriječiti da se bilo tko prijavi za rad pod tim korisničkim imenom.

Vidi "Dodjela lozinki", 221. stranica.

### Računi koji se ne koriste

Ako se određeno korisničko ime više ne koristi, trebali biste obrisati taj račun kako vam ne bi stvarao probleme. Ili biste barem trebali preurediti datoteku s lozinkama i kao lozinku za pristup takvom računu trebali biste upisati zvjezdicu (\*). Na taj način ćete spriječiti bilo koga da se prijavi za rad preko tog računa. Želite li obrisati račun, trebate pokrenuti naredbu find i pronaći sve datoteke koje mu pripadaju te im promijeniti vlasnika ili ih obrisati.

- Vidi "Brisanje korisnika", 222. stranica.
- BILJEŠKA Koristite li druge datoteke za podešavanje, kao što su popisi s adresama za slanje pošte, onda trebate ukloniti račun i iz tih datoteka.

### Osnovni (default) računi

Linux sadrži nekoliko računa koji su potrebni da bi sustav pravilno radio. Primjerice, nakon instalacije Linuxa račun superkorisnika nema lozinku. Čim završite instalaciju sustava trebali biste provjeriti datoteku s lozinkama i podesiti lozinke na osnovnim računima (neki računi će biti isključeni i imat će zvjezdicu (\*) umjesto lozinke.

Neki softverski paketi automatski stvaraju račune prilikom instalacije. Naknadno trebate postaviti lozinke i za takve račune ili račune trebate isključiti.

### Anonimni (gost) računi

Neki informatički centri nude račune za anonimni pristup onih posjetitelja koji privremeno koriste sustav. Takvi računi obično nemaju lozinku ili je ona jednaka korisničkom imenu. Primjerice, korisničko ime *gost* može služiti kao lozinka. Kao što možete pretpostaviti, ovakvi računi bitno ugrožavaju sigurnost sustava.

S obzirom da su računi i njihove lozinke svugdje poznati, nepoželjna osoba može lako pristupiti određenom sustavu i pokušati ostvariti pristup kao superkorisnik. Također vaše računalo može postati objekt s kojeg se ugrožava sigurnost drugih računala u mreži.

Anonimni računi nisu jako poželjni u sustavu. Ako ih doista trebate postaviti, tada ih svakako isključite kada vam nisu potrebni. Nikada lozinku nemojte slati e-poštom.

### Računi za pokretanje naredbi

Računala obično imaju nekoliko *računa za naredbe*, koji služe za pokretanje naredbi, nakon čega se zatvaraju. Jedan od takvih primjera je i naredba finger koja ne koristi lozinku. Kada se korisnik prijavi pod imenom finger, pokreće se istoimeni program koji traži sve ostale korisnike koji trenutno rade na mreži i potom se gasi. Drugi slični računi su sync i date koji također ne zahtijevaju uporabu lozinke. Iako takvi računi ne pokreću ljusku i koriste se za pokretanje samo određenih naredbi, ipak predstavljaju određenu nesigurnost sustava.

Ako u svom sustavu koristite račune za naredbe, tada dobro pazite da niti jedna od tih naredbi ne dozvoljava unos podataka s tipkovnice. Osim toga, takve naredbe ne smiju pozivati ljusku jer je to način da korisnik uspostavi kontakt s drugim naredbama.

Drugi razlog zašto se ne bi trebali koristiti takvi računi je taj što oni daju određene podatke o sustavu. Pomoću programa finger nepoželjni korisnik može saznati korisnička imena i pod nekim od njih se može pokušati prijaviti za rad. Upamtite da kombinacija korisničkog imena i lozinke štiti vaše račune. Ako je nepozvanom gostu poznato legalno korisničko ime, to je gotovo pola podatak koji su mu potrebni za prijavu na određeni račun.

### Grupni računi

Grupni račun je onaj račun za koji više osoba zna lozinku i sve se te osobe prijavljuju za rad pod istim korisničkim imenom. Već pogađate da je to prilično opasno. Imate li račun koji dijeli više osoba, onda ćete u slučaju da se otkrije lozinka, teško moći odrediti tko je od legalnih korisnika kriv za nastalu štetu. Ako pet osoba dijeli isti račun, možda će i drugih dvadeset znati njegovu lozinku. To se nikada ne zna.

Vidi "Rad s grupama korisnika", 223. stranica

Linux omogućava pristup dijeljenim datotekama putem članstva u određenoj grupi. Na taj način više osoba može koristiti iste datoteke, ali ne moraju dijeliti i isti račun (i lozinku). Umjesto da stvarate grupne račune, radije korisnike podijelite u grupe i zadržite logiku "jedno korisničko ime, jedna osoba".

### Zaštita datoteka

Sustav datoteka u Linuxu se sastoji od direktorija i datoteka i ima oblik stabla. U sustavu datoteka je pohranjeno nekoliko informacija o svakoj datoteci:

- ime datoteke
- vrsta datoteke
- veličina datoteke

- fizički smještaj na disku
- vrijeme promjene datoteke
- vlasnik datoteka (pojedinačni i grupni)
- dozvole pristupa datoteci

Ako korisnik može mijenjati neke podatke o određenim datotekama, moći će i narušiti sigurnost sustava. Stoga sustav datoteka igra važnu ulogu u zaštiti kompletnog sustava.

### Dozvole

Dozvole služe za određivanje koji korisnici mogu pristupiti određenim datotekama i naredbama. Njima se određuju prava koje prilikom pristupa ima vlasnik datoteke, članovi vlasnikove grupe i drugi korisnici. Koristeći naredbu ls –l možete pogledati popis datoteka i pripadnih dozvola (prvo lijevo polje u zapisu). Polje u kojem su prikazane dozvole može ovako izgledati: -rw-r- -r- -. Prva crtica označava vrstu datoteke i obično je prisutna kod većine datoteka.

Idućih devet znakova (grupirani u grupe po tri znaka) predstavljaju dozvole za vlasnika, grupu i "svijet". Svaka grupa znakova se može sastojati od znaka r (dozvola čitanja datoteke), w (dozvola pisanja u datoteku) i x (dozvola izvođenja datoteke). Bilo koji od navedenih devet znakova može biti prisutan.

Pojavi li se neki od spomenutih znakova, to znači da je izdana određena dozvola pristupa, a ako dozvola ne postoji, na ekranu ćete ugledati crticu (-). Evo primjera: ako polje izgleda ovako –rw-r- -r- -, to znači da se radi o običnoj datoteci (prvi znak je crtica), vlasnik ima dozvolu čitanja i pisanja, ali ne i izvođenja datoteke (rw-), a grupa kojoj pripada vlasnik i svi ostali korisnici imaju mogućnost čitanja datoteka (r- -), ali ne i pisanja ili izvođenja. Dozvole se mijenjaju pomoću naredbe chmod.

- Vidi "Dozvole pristupa datotekama", 310. stranica.
- **BILJEŠKA** Pomoću naredbe chmod možete dodijeliti dozvole datotekama tako da upišete oktalne vrijednosti umjesto oznaka rwx. Ta tri znaka u polju jednostavno smatrajte kao oktalni broj (ako postoji određeni znak, umjesto njega upišite 1, a ako ne postoji tada upišite 0). Kod takvog načina dodjele dozvola pristupa, polje –rw-r- -r- možete zadati kao 644 (oktalni brojevni sustav).

### Programi SUID i SGID

Uz datoteku se dodaju i dvije dodatne dozvole: SUID i SGID. SUID je kratica od Set User ID, a SGID je kratica Set Group ID.

Iako programi SUID i SGID mogu biti vrlo korisni, oni predstavljaju veliku rupu u sigurnosti sustava. Ovi programi se obično koriste kada neki program zahtijeva posebnu dozvolu, kao što je dozvola superkorisnika, da biste ga mogli pokrenuti. Ako ste pokrenuli neki SUID program, korisničko ime koje on koristi je podešeno tako da odgovara vlasniku, bez obzira tko je doista pokrenuo program. Program SGID je sličan, osim što koristi identifikacijski broj grupe. Najveća opasnost nastaje kada SUID programi izvođe naredbe koje su zadane u retku za unos naredbi, aktiviraju ljusku ili otvaraju datoteke koje korisnik može mijenjati i u njih upisivati vlastite naredbe. Iako su ovi programi ponekad korisni, nastojite njihovu uporabu svesti na minimum. Povremeno, pomoću naredbe find, provjerite datoteke na svom računalu kako biste bili sigurni da nema novih SUID programa.

# Izbjegavajte pretjeranu uslužnost

Uz sve mjere predostrožnosti koje se koriste da bi se očuvala sigurnost sustava, najveću pozornost treba obratiti na same korisnike jer oni već imaju legalne i valjane račune.

No, kakve to veze ima s uslužnošću korisnika? O čemu se tu zapravo radi? Kod velikog broja ljudi, ako zaigrate na kartu ljubaznosti, moći ćete doći do određenih podataka. Ljudi u osnovi vole biti korisni, te ako im ponudite priliku, oni će se potruditi da vam pomognu što je god više moguće. *Krakeri* koji imaju smisla za dodvoravanje ljudima mogu lako saznati potrebne podatke.

Zamislite jednog korisnika s korisničkim imenom Jones, koji ne predstavlja velikog stručnjaka za računala, već ima sasvim prosječno znanje o informatici. Jednog dana ga netko nazove u ured i započne ovakav razgovor:

g. Jones:	Halo?
Nepoznati:	Dobar dan gospodine Jones. Fred Smith iz tehničke službe. Zovem vas kako bi vas obavijestio da u 5,30 premještamo neke korisničke direktorije, a među njima i vaš početni direktorij. U to vrijeme nećete moći raditi na računalu.
g. Jones:	U redu. Ja ću do tada već biti kod kuće.
Nepoznati:	Dobro. No svakako se prije toga odjavite. No, moram provjeriti nekoliko stvari. Vaše korisničko ime je jones, zar ne?
g. Jones:	Da. No moje se datoteke neće izgubiti prilikom te promjene, zar ne?
Nepoznati:	Ni slučajno. No ja ću provjeriti vaš račun kako biste bili bez brige. Koja je vaša lozinka?
g. Jones:	Lozinka mi je <i>jutro</i> .
Nepoznati:	U redu. Hvala vam na pomoći. Pogledat ću vaš račun i provjeriti jesu li sve datoteke na mjestu.
g. Jones:	Hvala. Doviđenja!

Da vidimo što se upravo dogodilo. Netko je nazvao jednog od korisnika vašeg sustava i od njega saznao sve što mu je potrebno za pristup mreži: korisničko ime i lozinku. Nastavak možete pretpostaviti. Nikakav Fred Smith ne radi u tehničkoj službi zamišljene tvrtke. Kako spriječiti da se takve stvari događaju? Upozorite i naučite korisnike sustava da budu oprezni. Lozinka se nikada ne daje putem telefona, niti se šalje e-poštom. *Krakeri* upravo koriste takve neoprezne korisnike da bi ostvarili svoj naum. I čak neće niti izgubiti puno vremena.

# Praćenje uporabe naredbe su

Linux provjerava vaš identitet pomoću kombinacije korisničkog imena i lozinke. Čim se prijavite za rad, dobit ćete identifikacijski broj po kojem vas sustav prepoznaje. Radi se o korisničkom identifikacijskom broju koji služi za pristup datotekama i direktorijima.

Linux nudi mogućnost promjene korisničkog identifikacijskog broja tijekom rada. Čim korisnik pokrene naredbu su može postati superkorisnik ili neki drugi korisnik. ali pod uvjetom da zna njegovu lozinku. Primjerice, da bi neki korisnik zamijenio svoj identifikacijski broj s brojem korisnika ernie, mora pokrenuti naredbu:

su ernie

Nakon toga treba upisati lozinku koju koristi korisnik ernie.

Da bi svoj broj zamijenili identifikacijskim brojem superkorisnika, trebate upisati:

su root

Nakon toga se unosi lozinka superkorisnika.

Obično se svi pokušaji korištenja naredbe su zapisuju u sustavsku datoteku, primjerice /var/adm/syslog. Povremeno provjerite sadržaj ove datoteke.

## Razvoj sigurnosnog sustava

Moć donosi odgovornost. Ako obazrivo ne koristite Linux, njegova mogućnost dijeljenja informacija, resursa i perifernih komponenti može ugroziti sigurnost podataka i samog sustava. Vaša je zadaća tako podesiti sigurnosni sustav tako da samo pravi korisnici i računala mogu ostvariti vezu s vašim računalom, te da drugi korisnici mogu upotrebljavati samo dijeljeni dio vašeg sustava.

### Naredba ps

Da biste saznali tko sve koristi vaš sustav, te na koji način to čini, upotrijebite naredbu ps koja služi za praćenje sustava.

Posebnu pozornost obratite na poslove koji dugo traju te na korisnike koji koriste veći broj resursa od uobičajenog. To može biti naznaka neovlaštenog pristupa korisnika koji pokreće neki program kako bi saznao lozinke.

### **Pristup rootu**

Pristup rootu može imati jedino administrator sustava. Osoba koja se prijavi za rad kao superkorisnik ima mogućnost brisati bilo koju datoteku, ograničiti rad nekom od korisnika ili, slikovito rečeno, načiniti zbrku među korisnicima. To je tamna strana medalje. Linux je zamišljen kao sustav koji superkorisnicima nudi alate pomoću kojih će obaviti svoj posao bolje nego u nekim drugim operativnim sustavima.

Mnogi operativni sustavi koriste blokade koje su uspostavili njihovi kreatori, a koje služe da bi se spriječilo brisanje datoteka i nanošenje štete operativnom sustavu. Ljudi koji su zamislili UNIX i Linux ponudili su administratorima različite alate. Naći ćete alate koji vam omogućuju da se povežete s bilo kojim uređajem na računalu. Naći ćete softver koji služi za praćenje karakteristika računala. Možete upotrijebiti pravo more programa i prilagoditi ih za gotovo sve moguće zadaće koje vas očekuju.

Korisnike možete prisiliti da rade samo određene stvari na računalu, ili im možete dati ograničena prava sve dok ne ovladaju sustavom. Superkorisnik, administrator ima takve ovlasti.

S obzirom da je pristup rootu toliko važan, neke tvrtke omogućuju samo nekolicini BILJEŠKA ljudi da budu super korisnici.

### Praćenje modema i krakera

Ako je za pristup sustavu dovoljan običan modem, sličan onome kakvog mnogi korisnici imaju u svojoj kući, tada nepozvana osoba može "upasti" u sustav i uništiti važne podatke. Zbog toga mnoge tvrtke insistiraju na obimnim mehanizmima zaštite, što može otežati rad na računalu. Neke tvrtke koriste mogućnost telefonskog pozivanja sustava na taj način da morate sačekati odziv prije nego budete u mogućnosti uspostaviti vezu.

No u većini slučajeva se preporuča tradicionalni način pristupa koji se koristi kod mnogih Linux/UNIX sustava. Provjerite imaju li svi vaši korisnici svoje lozinke. Ograničite broj računala koja mogu ostvariti vezu s vašim sustavom. Dozvole pristupa držite u posebnim datotekama. pažljivo podesite UID programe (koji korisnicima daju mogućnost da rade pod tuđim imenom). Najčešće se neovlašteni ulasci u sustav događaju "jer je netko ostavio otvorena vrata".

Za sigurnost sustava su veći problem ljudi nego računala. Lozinke ne biste trebali stavljati na oglas-

nu ploču da ih svi mogu pročitati, niti biste trebali koristiti komunikacijske

BILJEŠKA programe s upisanom lozinkom.

### Ostavljanje slobodnih terminala

Korisnici se, na kraju radnog dana, mogu odjaviti ili pokrenuti neki od programa za zaključavanje terminala. Većina UNIX sustava imaju takve programe koji gase terminale ako se oni ne koriste neko određeno vrijeme.

### Pojačana sigurnost

Potpuno je razumljiv zahtjev za zaštitom podataka u vojnim tvrtkama. Tvrtke koje proizvode vrlo osjetljive proizvode također traže dodatnu sigurnost svog informatičkog sustava. No, uposlenici u maloprodaji opreme za vodoinstalaterske radove teško će razumjeti zabrinutost za vlastiti sustav računala.

Djelatnike bi trebalo brzo podučiti o važnosti podataka na računalu, jer ponekad poslovanje tvrtke ovisi upravo o tim podacima. Gubitak važnih podataka može stvoriti kaos. Radnici koji nisu motivirani za sudjelovanje u zaštiti sustava mislit će da je zaštita potpuno nevažna.

No, za administratora sustava je zaštita itekako važna. Ako ste odgovorni za sigurnost podataka na mreži, na koji način možete biti sigurni da su datoteke i direktoriji potpuno osigurani od provale? Na svu sreću, postoji dovoljno alata koji vam u tome pomažu. Primjerice, programi umask, cron, pa i sami operativni sustav Linux.

Dozvole predstavljaju značajan povod za zabrinutost mnogih administratora. Novi administratori su obično škrti na dozvolama, pa im se javljaju korisnici koji ne mogu pristupiti određenim datotekama i programima koji su im neophodni. Potom takvi administratori ukinu dozvole tako da svi mogu pristupati svim podacima. No, ponekad je doista naporno odrediti pravu mjeru između sigurnosti sustava i omogućavanja određenim ljudima da se domognu potrebnih alata.

### Pukotine u sigurnosti sustava

Sigurnost računala zahtijeva da korisnici pomalo budu i "detektivi". Pogledajte sljedeće:

tty02	Jan	7	08:35	old Ofc #2
ttym1d	Jan	7	13:20	. Payroll #1
ttyp0	Jan	7	08:36	8:25 Warehouse
ttyp2	Jan	7	07:05	9:45 CEO Ofc
ttyp4	Jan	7	08:36	. Modem #1
7 19:	18:21	CST	1997	
	tty02 ttym1d ttyp0 ttyp2 ttyp4 7 19:	tty02 Jan ttym1d Jan ttyp0 Jan ttyp2 Jan ttyp4 Jan 7 19:18:21	tty02 Jan 7 ttym1d Jan 7 ttyp0 Jan 7 ttyp2 Jan 7 ttyp4 Jan 7 7 19:18:21 CST	tty02Jan708:35ttym1dJan713:20ttyp0Jan708:36ttyp2Jan707:05ttyp4Jan708:36719:18:21CST1997

Pretpostavimo da znate da je Martha napustila ured u 17 sati. Je li netko pronašao njezinu lozinku ili nije napustila sustav prije nego je otišla kući? Možete uočiti da se za rad prijavila u 13:20. Sada je 19:18 i netko tko je aktivan u sustavu koristi njezinu lozinku. Je li ugrožena sigurnost sustava?

No, što ćete učiniti ako netko doista upadne u vaš sustav? Najprije se uvjerite da stvarno imate uljeza. Mnogo puta razlog za paniku znaju biti ljudske pogreške. Ako doista ustanovite da u sustavu imate uljeza, tada vam na raspolaganju stoji nekoliko mogućnosti. Trebate ustanoviti je li nastala kakva šteta i kolika je ona. Hoćete li tužiti odgovorne osobe za načinjenu štetu, ako ih ulovite?

Morate odlučiti kako ćete ubuduće štititi sustav i otkloniti nastalu štetu. Vjerojatno najvažnije stvar je dokumentirati ono što se zbilo. Odmah počnite bilježiti podatke.

Sve ispise koje prikazuju popis osoba koje su radile na računalu sakupite, potpišite i odložite na jedno mjesto.

Dvije preventivne mjere koje biste trebali poduzeti je da načinite ispise osnovnih datoteka s podešenjima sustava, kao što je datoteka /etc/fstab, te da uspostavite jedinstveni način zaštite sustava. Morate biti sigurni da se svi vaši korisnici pridržavaju propisanih mjera sigurnosti te da ih redovito primjenjuju.

Još jedna od mjera je i praćenje odlaska i dolaska djelatnika na posao. Nakon što djelatnik napusti tvrtku, osoblje zaduženo za osiguranje bi trebalo o tome obavijestiti informatičare tako da oni mogu provjeriti da li se osoba odjavila sa sustava.

Uz sve navedene mjere ostaje pitanje, koja je dovoljna razina sigurnosti? Može li sustav biti "previše" siguran? Iznenadit ćete se, ali odgovor je potvrdan, sustav može biti previše siguran. Općenito govoreći, ako je cijena otklanjanja štete nakon nasilnog upada u sustav niža od troškova sigurnosti, trebali biste smanjiti razinu sigurnosti sustava. Pod pojmom "cijena" se ne smatra samo novčana vrijednost. Među svim drugim stvarima, u razmatranje biste trebali uzeti i sadržaj vaših datoteka, vrijeme i novac koji bi se potrošio za njihovu prepravku, smanjenu efikasnost tvrtke zbog nastale štete, te ugled u javnosti ako se sazna za problem sigurnosti u računalnom sustavu vaše tvrtke.

### Sigurnosna pohrana podataka

Samo nekoliko zadaća administratora sustava su toliko važne kao sigurnosna pohrana ili arhiviranje podataka. Administrator može dobiti otkaz i tvrtka može izgubiti važne podatke. Disk ili diskovi na računalu su elektromehanički uređaji i nakon nekog vremena se mogu pokvariti.

Većina novih tvrdih diskova mogu raditi 150.000 sati bez kvara, odnosno više od pet godina. No, statistika onoga što se događa u tih pet godina može biti opasna. Disk se može pokvariti i nakon 50.000 sati, ili može raditi više od 10 godina. Kockate se s podacima ako ih samo povremeno spremate na neki od medija. Veće su vam šanse da izgubite ako rijetko provjeravate svoje trake za pohranu podataka.

Vidi "Raspored sigurnosne pohrane podataka", 227. stranica

## **PAM – Pluggable Authentication Modules**

Korisnici trebaju imati mogućnost obavljati bilo kakve zadaće, čak i ako im je cilj složiti pasijans. S obzirom na tu tezu, korisnici će se, prema svome stupnju informatičkog znanja, različito odnositi prema sustavu i sadržajima koje on nudi. Općenito govoreći, korisnici bi trebali pokretati programe i stvarati, mijenjati i brisati datoteke koje nemaju utjecaj na sami sustav, ali ne bi trebali imati mogućnost mijenjanja datoteka koje pripadaju drugim korisnicima (naravno ako im ovi to ne dozvole). Jedan od načina kako se može autorizirati rad na takvom jednom sustavu je uporaba korisničkog imena i lozinke. Čim se prijavite za rad, sustav traži da unesete korisničko ime i lozinku. Polazeći od pretpostavke da ste se pravilno predstavili, sustav vam omogućuje da činite gotovo sve što želite da biste ostvarili svoj naum, ali vas ograničava da mijenjate onaj dio sustava za koji niste nadležni. Postoje i drugi načini da se provjeri identitet korisnika, osim unosa imena i lozinke. Arhitektura PAM (Pluggable Authentication Modules) omogućuje promjenu načina identifikacije bez mijenjanja samih programa. U ovom odlomku je predstavljena struktura i međusobni odnosi modula u arhitekturi PAM.

Postoje četiri vrste PAM modula:

- Auth obavlja autorizaciju korisnika.
- Account određuje jeli autorizacija dozvoljena. Pretpostavimo da postoji korisnik koji sustav može koristiti samo u radno vrijeme, a nema dozvolu rada poslije podne niti vikendom. Ovaj modul će otkriti korisnika ako određenu radnju pokušava obaviti u ponoć.
- Password podešava lozinke.
- Session nudi korisničke usluge nakon što modul account "dozvoli" modulu auth provjeru korisnikovog identiteta.

Module možete poredati tako da omogućite više načina pristupa sustavu ili da ograničite pristup onim korisnicima koji "uspješno ne prođu sve module".

### Datoteke za podešavanje PAM-a

Datoteke za podešavanje PAM-a se nalaze u direktoriju /etc/pam.d.

BILJEŠKA U starijim Linux sustavima datoteka /etc/pam.conf sadrži podešenja. Ta se datoteka i dalje koristi radi uskladivosti sustava, ali je njezina uporaba nebitna.

Najbolji je način da upoznate sintaksu datoteke za podešavanje PAM-a, da otvorite jednu datoteku, primjerice datoteku passwd. Ako ste instalirali PAM tijekom instalacije Linuxa, onda bi vaša datoteka /etc/pam.d/passwd trebala ovako izgledati:

```
#%PAM-1.0
auth required /lib/security/pam_pwdb.so shadow nullok
account required /lib/security/pam_pwdb.so
password required /lib/security/pam_cracklib.so retry=3
password required /lib/security/pam_pwdb.so use_authtok nullok
```

Prvi redak predstavlja komentar (označen znakom #). U drugom retku se nalazi upozorenje korisniku da unese lozinku koju će sustav provjeriti. Treći redak čini isto što i drugi redak ako se koriste *lozinke u sjeni* (o njima će biti više riječi nešto kasnije). U četvrtom retku se poziva program za provjeru nove lozinke, a u petom retku je odabran modul koji se koristi za promjenu lozinke.

### Označavanje modula

Možete uočiti da su sva četiri modula označena kao "required", što znači da se određeni modul poziva bez obzira je li prethodni modul uspješno ili neuspješno okončao svoj rad. Zbog sigurnosti se pozivaju sva četiri modula, tako da će u slučaju greške u bilo kojem od modula, poruka izgledati isto. Skrivanjem mjesta gdje je nastala greška, neovlaštenim koris-

od modula, poruka izgledati isto. Skrivanjem mjesta gdje je nastala greska, neovlastenim korisnicima se otežava "posao" ulaska u sustav. Ako su svi moduli označeni sa "required" (obavezan), onda je njihov redoslijed nebitan. No, umjesto opcije "required" mogu stajati i drugi parametri:

- optional
- sufficient
- requisite

Parametar "optional" se postavlja uz sekundarne (rezervne) module, tako da uspješnost/neuspješnost "opcijskog" modula ne utječe na uspješnost prijave, AKO postoji i drugi modul u datoteci s podešenjima. Ako je pak "optional" modul jedini definirani modul za provjeru autentičnosti, onda njegov rezultat određuje uspješnost/neuspješnost prijave. Modul s parametrom "sufficient" ponaša se kao i "opcijski" modul s tom razlikom što može prebrisati pojedine (ili sve) "opcijske" module. Moduli koji su označeni kao "required" ili "requisite" predstavljaju "nadređene" module u odnosu na modul s oznakom "sufficient". Ako modul s oznakom "requisite" uspješno ne obavi zadaću, upravljanje se izravno prenosi na program. Ako želite da se PAM zaustavi kod određenog modula, morate preurediti datoteku s podešenjima i promijeniti parametar "required" u "requisite".

Više informacija o PAM-u potražite na Web lokaciji: http://www.redhat.com/linux-info/pam/.

# Čemu služe lozinke u sjeni?

Na Linux računalu koje nema instaliran softver Shadow Suite, informacije o korisniku (uključujući i lozinke) su pohranjene u datoteci /etc/passwd. Lozinka je spremljena u skrivenom formatu, pa, iako se korisniku čini nerazumljivom, može se lako dekodirati pomoću naredbe crypt (ako se opcija text postavi na vrijednost [null], a lozinka upotrijebi kao ključ).

Teško je, ali ne i nemoguće, iskoristiti dekodiranu lozinku i otkriti originalnu riječ. No, s obzirom da su ljudi ponekad lijeni, na svakom sustavu s više korisnika, neke će lozinke biti jednostavne i odmah će ih se moći otkriti. Može se i stvoriti neka vrsta rječnika za usporedbu s lozinkama u datoteci /etc/passwd. Ovo je jedan od najjednostavnijih načina za upad u sustav. Osim lozinki, datoteka /etc/passwd sadrži podatke kao što su korisnički identifikacijski broj i broj za identifikaciju grupe, koje čitaju mnogi sustavski programi.

Skrivanjem lozinki ustvari premještate lozinke u drugu datoteku, obično datoteku /etc/shadow, koju može čitati samo superkorisnik. Premještanjem lozinki u drugu datoteku možete spriječiti neovlaštenu osobu da dekodira lozinke i načini rječnik pomoću kojeg može upasti u vaš sustav.

Paket Shadow Suite je sastavni dio većine standardnih Linux distribucija.

U idućim slučajevima ne bi bilo dobro instalirati Shadow Suite:

- ako sustav ne sadrži korisničke račune
- računalo koristi lokalnu mrežu (LAN) i mrežne informacijske servise (NIS) za dodjelu korisničkih imena i lozinki drugim računalima u mreži

- računalo se koristi za provjeru korisnika putem mrežnog sustava datoteka (NFS), NIS-a ili slično
- računalo koristi druge programe za provjeru korisnika, kod kojih nisu dozvoljene lozinke u sjeni, a i nije vam dostupan osnovni kod tih programa.

### Datoteke /etc/password i /etc/shadow

Lozinke (koje nisu u sjeni) u datoteci /etc/passwd su zapisane u sljedećem obliku:

```
username:Npje044eh3mx8e:507:200:Full Name:/home/username:/bin/csh
```

Zapis u datoteci /etc/passwd bi trebao ovako izgledati (ako su lozinke u sjeni:

```
username:x:507:200:Full Name:/home/username:/bin/csh
```

Parametar x u drugom polju zamjenjuje stvarnu lozinku koje je pohranjena u datoteci /etc/shadow. U datoteci /etc/shadow su podaci zapisani u ovom obliku:

username:passwd:last:may:must:warn:expire:disable:reserved

U tablici 12.1 se nalazi opis polja koja se koriste u zapisima u datoteci /etc/shadow.

	ja u zapiolina u udototoli /oto/onauon
Polje	Opis
username	Korisničko ime pod kojim se korisnik prijavljuje
password	Kodirana lozinka
last	Broj dana od zadnje promjene lozinke (1.1.1970.)
may	Broj dana do iduće promjene lozinke
must	Broj dana do kada lozinka mora biti promijenjena
warn	Broj dana do upozorenja korisniku da mora promijeniti lozinku
expire	Broj dana nakon upozorenja korisniku nakon kojeg se račun zatvara
disable	Broj dana od 1.11970. kada je račun bio nedostupan
reserved	Rezervirano polje

### Tablica 12.1 Polja u zapisima u datoteci /etc/shadow

### Dodavanje, mijenjanje i brisanje korisnika

Program Shadow Suite sadrži naredbe za dodavanje, mijenjanje i brisanje korisnika: useradd, usermod i userdel.

**useradd** Naredba useradd se koristi za dodavanje korisnika ili za mijenjanje osnovnih podešenja računa.

Prvo je što trebate učiniti da provjerite osnovna podešenja i unesete promjene koje su se odnose na vaš sustav. Pokrenite ovu naredbu:

useradd -D

**usermod** Naredba usermod služi za mijenjanje podataka o korisniku. Ova naredba je vrlo slična naredbi useradd.

userdel Naredba userdel omogućuje brisanje korisničkih računa:

userdel -r username

Koristeći opciju –r, obrisat ćete sve datoteke u korisnikovom početnom direktoriju, kao i sami direktorij. Drugi način, nešto "nježniji", za uklanjanje korisnika iz sustava je uporaba naredbe passwd kojom možete zaključati korisnikov račun.

**passwd** Osim podešavanja i mijenjanja lozinki, super korisnik može upotrijebiti naredbu passwd na jedan od sljedećih načina:

- zaključavanje i otključavanje računa (opcije –l i –u)
- podešavanje maksimalnog trajanja važeće lozinke (-x)
- podešavanje minimalnog broja dana koji mora proteći između promjena lozinke (-n)
- podešavanje trajanja upozorenja da će lozinka isteći (-w)
- podešavanje broja dana između isteka lozinke i blokiranja računa (-i)

**pwck** Program pwck omogućuje provjeru konzistentnosti datoteka /etc/passwd i /etc/shadow. On provjerava svako korisničko ime tako da svaki zapis mora imati:

- točan broj polja
- jedinstveno korisničko ime
- ispravan identifikacijski broj korisnika i grupe
- važeću primarnu grupu
- važeći početni direktorij
- važeću ljusku za prijavu

Osim toga, naredba pwck vas upozorava na račun koji nema lozinku.

# BILJEŠKA Korisno je pokrenuti naredbu pwck nakon što ste instalirali Shadow Suite, te naredbu povremeno pokrenuti (jednom tjedno ili jednom mjesečno). Upotrijebite li opciju –r, moći ćete koristiti naredbu cron i dobiti odgovarajući izvještaj.

**grpck** Program grpck služi za provjeru konzistentnosti datoteka /etc/group i /etc/gshadow, to jest za provjeru broja polja, jedinstvenih imena grupe i važećeg popisa članova i administratora.

Osim toga, koristite li naredbu s opcijom –r, dobit ćete izvještaj. Ovu provjeru možete automatski pokrenuti koristeći program cron.

**Lozinke za uspostavu telefonske veze.** Želite li ograničiti broj osoba koje se spajaju na vaš sustav, onda možete uvesti i lozinke koje koriste oni što daljinski pristupaju sustavu. Da biste to mogli učiniti, najprije u datoteci /etc/login.defs provjerite je li opcija DIALUPS CHECK ENAB postavljena na Yes.

Dvije datoteke sadrže podatke za uspostavu telefonske veze:

- Datoteka /etc/dialups sadrži *tty*-ove (na početku nema /dev/). Ako se određeni *tty* nalazi u popisu, moći će se uspostaviti telefonska veza.
- Datoteka /etc/d\_passwd sadrži kompletnu stazu ljuske i lozinku (proizvoljno).

Ako se korisnik prijavi na uređaj koji se nalazi u popisu u datoteci /etc/dialups, a njegova se ljuska nalazi u popisu u datoteci /etc/d\_passwd, biti će mu omogućen pristup (naravno ako upiše pravu lozinku).

Naredba dpasswd dodjeljuje lozinke ljuskama u datoteci /etc/d\_passwd.

### Odavde...

Dodatne podatke o sigurnosti sustava možete naći u ovim poglavljima:

- 10. poglavlje, "Upravljanje korisničkim računima" prikazuje kako se dodaju, brišu korisnički računi te kako se njima upravlja.
- U 11. poglavlju, "Sigurnosna pohrana podataka" je objašnjeno kako podesiti i obaviti sigurnosnu pohranu podataka.

# POGLAVLJE 13

# Podešavanje Linux jezgre

(Jack Tackett)

### U ovom poglavlju

Pripreme za stvaranje nove jezgre 254 Podešavanje nove jezgre 255 Kompiliranje nove jezgre 258 Stvaranje modularne jezgre 259 U ovom poglavlju ćete naći informacije koje su vam potrebne za podešavanje i instaliranje nove Linux jezgre. Jezgra predstavlja srce operativnog sustava Linux i nudi osnovne sustavske usluge ostalom dijelu Linuxa. Upamtite da Linux nije komercijalni proizvod, tako da kod novih inačica možete naići na probleme, ili netko može otkriti ozbiljne sigurnosne rupe u jezgri. Takve se stvari događaju i kod komercijalnih i kod "besplatnih" operativnih sustava. Razlika je u tome, što kod Linuxa imate na raspolaganju osnovni kod tako da možete otkloniti problem čim ga uočite. Ne trebate čekati da isporučitelj softvera objavi novo izdanje softvera u kojem je vaš problem otklonjen.

Osim toga, nove značajke u postojećim izdanjima Linux jezgre vam omogućuju pokretanje određenih uređaja i programa unutar same jezgre, a da pri tome podršku ne trebate ponovno kompajlirati. Na taj će način u memoriju biti učitani samo oni dijelovi jezgre koji su vam potrebni. Moduli omogućuju mijenjanje jezgre i otklanjanje problema ili dodavanje novih mogućnosti bez ponovnog kompiliranja čitavog sustava.

### Pripreme za stvaranje nove jezgre

Ponekad problem nameće samo jedno rješenje – novu jezgru. Jezgra je "srce" operativnog sustava Linux. Iako niste kardiolog, ponekad ćete biti u situaciji da s Interneta kopirate novu jezgru i instalirate je na svoje računalo. Imate li barem malo programerskog iskustva, te ako znate koristiti programski jezik C, tada ćete moći stvoriti i instalirati novu jezgru. Ako nemate iskustva, tada preskočite ovaj dio knjige.

Novu jezgru trebate instalirati iz dva razloga:

- Objavljena je nova "zakrpa" za pokretanje neke nove hardverske komponente.
- Potrebno je ukloniti neke značajke jezgre koje inače ne koristite kako biste oslobodili memoriju.

Za početak odredite inačicu jezgre koju trenutno koristite. Inačicu možete saznati ako pokrenete naredbu:

uname -e

Saznat ćete koja se jezgra trenutno koristi i kada je nastala. Broj inačice jezgre ima sljedeći oblik:

BrojGlavneInačice.BrojPodInačice.BrojZakrpe

BILJEŠKA Prije nego počnete stvarati i instalirati novu jezgru, svakako pročitajte najnovije informacije o jezgrama, jer se može dogoditi da vaš sustav neće moći raditi ako nešto zabrljate.
 Također biste morali zadržati stariju kopiju jezgre koja vam može poslužiti ako naiđete na probleme. U tom slučaju ćete moći pokrenuti stariju jezgru umjesto novije, ali beskorisne jezgre.

### Podešavanje nove jezgre

Da biste stvorili novu jezgru najprije trebate podesiti datoteke s osnovnim kodom koje bi se trebale nalaziti u direktoriju /usr/src/Linux. Također morate pokrenuti softverski paket C kompilator. Ako taj paket niste instalirali tijekom instalacije sustava, pokrenite RPM i učinite to sada:

```
rpm -i kernel-source-2.0.34-0.6.i386.rpm
rpm -i gcc-2.7.2.3-11.i386.rpm
```

Možda ćete trebati instalirati zaglavlja jezgre i različite zbirke kompilatora.

Vidi "Instalacija paketa pomoću RPM-a", 150. stranica

Najprije morate naći osnovni kod nove jezgre. Na Internetu potražite lokaciju **sunsite.unc.edu** i naći ćete najnovije i najbolje jezgre (ako samo mijenjate postojeću jezgru, onda vam ovaj korak nije potreban). Datoteke s osnovnim kodom su obično

komprimirane i treba ih dekomprimirati.



SAVJET

Načinite rezervnu kopiju postojeće jezgre koristeći ove naredbe:

```
cd /usr/src
cp Linux linux.sav
```

Pomoću ovih naredbi ćete kopirati sadržaj direktorija s osnovnim kodom Linuxa u drugi direktorij pod imenom linux.sav.

Zatim trebate pokrenuti naredbu patch i primijeniti jednu od *patch* datoteka. Nakon što pripremite datoteke s osnovnim kodom moći ćete podesiti i pokrenuti novi sustav. Ovisno o vašim željama i hardverskim mogućnostima, na raspolaganju su vam tri načina za podešavanje jezgre: program s tekstualnim zadavanjem naredbi, program s izbornicima, te program koji radi u sučelju X Windowsa (naravno, prethodno trebate instalirati ovo sučelje).

Da biste koristili module u jezgri, tijekom podešavanja jezgre morate potvrditi opciju za korištenje *demona* kerneld i potvrditi uporabu modula u jezgri (CONFIG\_MODVERSIONS).

### Interaktivni programi bez izbornika

Koristite li tekstualne interaktivne programe, za početak upišite sljedeću naredbu u direktoriju /usr/src:

#make config

Program make će vam postaviti razna pitanja o upravljačkim programima koje želite instalirati ili podesiti. Želite li prihvatiti osnovno podešenu vrijednost, tada pritisnite tipku <Return>, a u protivnom upišite potreban odgovor. Neka od pitanja su prikazana u tablici 13.1. Možda ćete trebati odgovoriti i na druga pitanja, ovisno o inačici jezgre koju instalirate. Ovaj popis opcija podržavaju svi programi za podešavanje jezgre koji su opisani u ovom poglavlju.

ladiica 13.1 Neke opcije za podesavanje jezgre			
Орсіја	Opis		
Code Maturity Level	Koristi se za eksperimentalne komponente u jezgri		
Loadable Module Support	Nužna ako namjeravate koristiti module umjesto cjelovite jezgre		
General Setup	Niz pitanja o općeniti komponentama kao što su podrška za matematički koprocesor i PCI BIOS		
Floppy, IDE and Other Block Devices	Niz pitanja o vrstama IDE tvrdih diskova i drugih uređaja za ulaz/izlaz		
Networking Options	Niz pitanja o nekim značajkama mreže kao što su barijere i IP Masquerading		
SCSI Support	Omogućuje uporabu SCSI kontrolera		
SCSI Low-Level Support	Omogućuje uporabu SCSI kontrolera niže razine i nudi različite statistike.		
Network device support	Omogućuje uporabu raznih mrežnih kontrolera i procesa.		
ISDN subsystem	Omogućuje potporu za ISDN (Integrated Services Digital Network)		
CD-ROM drivers (ne SCSI ili IDE/ATAPI)	Podrška za odgovarajuće pogone CD-ROM-a		
Filesystems	Omogućuje podešavanje raznih sustava datoteka i kodnih stranica za razne strane jezike (DOS)		
Character Devices	Omogućuje uporabu raznih znakovnih i sličnih programa kao što je watchdogs		
Sound	Omogućuje potporu za različite zvučne kartice		
Kernel Hacking	Podrška za rad s jezgrom		

Tablica 13.1	Neke	onclie 7a	nodesavan	IA IA70rA
	HUCKE	opoije zu	poucouvan	

### Uporaba programa s izbornicima

Koristite li tekstualne interaktivne programe, za početak upišite sljedeću naredbu u direktoriju /usr/src/linux:

#make menuconfig

Zatim će se pojaviti glavni ekran, koji je prikaza na slici 13.1.



Prednosti uporabe grafičkog sučelja je ta što trebate podesiti samo one dijelove jezgre koji su vam potrebni. Interaktivni program koji radi u tekstualnom načinu rada vodi vas kroz čitav postupak podešavanja jezgre.

### Uporaba programa koji rade pod X Windowsima

Koristite li tekstualne interaktivne programe, za početak upišite sljedeću naredbu u direktoriju /usr/src/linux:

#make xconfig

Zatim će se pojaviti glavni ekran, koji je prikaza na slici 13.2.



Alati za podešavanje (koji rade u okružju X Windowsa) omogućuju mijenjanje samo onih komponenti jezgre koje nužno treba mijenjati, baš kao i alati koji rade u tekstualnom

podršku za module.

načinu rada. Nakon što kliknete na neki gumb, pojavit će se drugi okvir za dijalog koji služi za podešavanje raznih komponenti. primjerice, slika 13.3 prikazuje okvir za dijalog Loadable Module Support u kojem možete podesiti da čitava jezgra podržava module.



Element odabirete tako da kliknete na određenu opciju. Zatrebali vam pomoć, možete kliknuti na odgovarajući gumb Help koji se nalazi na desnoj strani okvira za dijalog. Pojavit će se novi okvir za dijalog koji sadrži potrebnu pomoć (vidi sliku 13.4).

Slika 13.4		CONFIG_MODULES:
Ekran za pomoć	i	Kernel modules are small pieces of compiled code which can be inserted in or removed from the running kernel, using the programs insmod and mmod. This is described in the file Documentation/modules.txt. Modules can be device drivers, file
CONFIG_MODULES.		
Korisne informacije se		you may want to make use of modules with this kernel in the future, then say Y here. If unsure, say Y.
nalaze na dohvat ruke.		ОК

Nakon što odgovorite na postavljena pitanja, podešenja morate spremiti. Jednostavno kliknite na gumb Save and Exit i spremit ćete podešenja koja se odnose na novu jezgru i napustiti podešavanje sustava.

### Kompajliranje nove jezgre

Nakon što ste odgovorili na različita pitanja koja se odnose na podešenja nove jezgre, morate jezgru kompajlirati. Pokrenete li iduće naredbe, stvorit ćete novu jezgru:

```
make dep
make clean
make
```

Ovisno o vašem hardveru, čitav će posao trajati ili nekoliko minuta ili par sati. Opustite se i naručite pizzu!

Po završetku kompajliranja potrebno je podesiti sustav tako da koristi novu jezgru. Nova jezgra se zove /usr/src/linux/arch/i386/boot/zImage i trebate je kopirati u direktorij *boot*. No prije toga, za svaki slučaj, načinite kopiju "stare" jezgre. Staru jezgru spremate pomoću naredbe: mv /boot/vmlinuz.old /boot/vmlinuz.old

Nakon toga možete kopirati novu jezgru:

cp /usr/src/linux/arch/i386/boot/zImage /boot/vmlinuz

Vidi "Podešavanje LILO", 214. stranica

Da biste promijenili osnovno podešenu jezgru Linuxa, trebate preurediti datoteku /etc/lilo.conf i dodati još jedan zapis za novu jezgru. Idući primjer (ispis 13.1) pokazuje dodatak stare jezgre u popisu operativnih sustava koje računalo može pokrenuti. Najprije morate promijeniti ime jezgre /boot/vmlinuz.u /boot/vmlinuz.old (koristite prethodnu naredbu) i potom promijeniti njezinu oznaku u datoteci lilo.conf (kao u ispisu 13.1).

```
Ispis 13.1 Primjer datoteke /etc/lilo.conf
```

```
boot=/dev/had
map=/boot/map
install=/boot/boot.b
prompt
timeout=50
image=/boot/vmlinuz
label=linux
initrd=/boot/initrd
root=/dev/hda1
read-only
image=/boot/vmlinuz.old
label=old
root=/dev/hda1
read-only
```

Nakon što ste u datoteku /etc/lilo.conf unijeli izmjene, pokrenite ovu naredbu

/sbin/lilo -v

i ažurirani podaci će biti zapisani u uređaj za pokretanje sustava. Ubuduće će se vaše računalo pokretati u novoj jezgri (linux), umjesto u staroj jezgri, a pauza od 50 sekundi ostaje za odabir stare jezgre ako u njoj želite podići sustav.

### Stvaranje modularne jezgre

Uvođenjem modula u jezgru Linux 2.0.x dogodile su se značajne promjene u stvaranju korisničkih jezgri. Prije toga je bilo potrebno kompajlirati podršku u samu jezgru, ako ste željeli pristupiti određenoj hardverskoj komponenti ili sustavu datoteka. Za neke hardverske komponente veličina jezgre je brzo mogla poprimiti kritične razmjere, stoga je priprema za uporabu resursa koji se rjeđe koriste često puta usporavala čitav sustav. Pojavom nove jezgre 2.0.x stvari se mijenjaju, tako da, ako samo povremeno koristite određeni sustav datoteka ili hardversku komponentu, moći ćete učitati ili isključiti pripadni upravljački modul. Da biste vidjeli koji ste moduli trenutno koriste, upišite naredbu:

lsmod

U ispisu, koji je niže prikazan, možete vidjeti koji su moduli učitani, kako se učitavaju, te koliko stranica memorije koriste.

ModulePagesUsed byisofs51 (autoclean)ne2k-pci11 (autoclean)83902[ne2k-pci]0 (autoclean)BusLogic204

jedino distribucije Red Hat Linux/Intel i Red Hat Linux/SPARC podržavaju modularne jezgre, dok kod distribucije Red Hat/Alpha korisnici moraju stvoriti monolitnu jezgru, kao što je opisano u prethodnom odlomku "Stvaranje nove jezgre". U tim ćete uputama saznati sve što je potrebno da biste iskoristili prednosti koje nude modularne jezgre.

**BILJEŠKA** Nužno je najprije instalirati pakete koji se koriste u zaglavljima i izvoru jezgre (*kernel-headers i kernel-source*). Također morate pokrenuti sve naredbe koje se nalaze u direktoriju /usr/src/linux.

Da biste načinili module, potražite direktorij /usr/src/linux i pokrenite sljedeću naredbu:

make modules

Potom pokrenite naredbu za instalaciju modula:

make modules-install

### Rad s modulima jezgre

Sada kada ste kompajlirali i instalirali module, možete proširiti jezgru tako da module učitate. Tablica 13.2 prikazuje osnovne naredbe koje vam stoje na raspolaganju.

Tablica 13.2	Naredbe za pokretanje modula u Linuxu
Naredba	Opis
lsmod	Ispis modula koji su trenutno učitani u jezgru
insmod	Umetanje određenog modula u jezgru
rmmod	Uklanjanje određenog modula iz jezgre
depmod	Stvaranje datoteke koju koristi naredba modprobe
modprobe	Učitavanje modula s popisa načinjenog pomoću naredbe depmod

Koristite li X Windowse, moći ćete iskoristiti prednosti koje vam nudi demon *kerneld* koji se nalazi u Control Panelu (prikazan na slici 13.5). Program služi za rad s modulima u grafičkom okruženju (umjesto zadavanja naredbi u retku za unos naredbi). Kliknete li na taj gumb, pojavit će se okvir za dijalog Kernel Configurator koji je prikazan na slici 13.6.

Da biste vidjeli popis trenutno aktivnih modula, pokrenite naredbu lsmod. Da biste modul dodali u jezgru, možete upotrijebiti naredbu

insmod module-name

ili kliknuti na gumb Add u okviru za dijalog i nakon toga odabrati modul (vidi sliku 13.7).

Slika 13.5

Gumb programa kerneld. Control Panel nudi više različitih administrativnih funkcija, među kojima je i program kerneld.

Gumb kerneld-



### Slika 13.6

Okvir za dijalog Kernel Configurator. Rad s modulima je bitno jednostavniji u ovakvom sučelju.

🔵 Kerne	l Configura	tor		
Туре	Module		Arguments	$\Box$
ethO	3c509			X
	Add Ed	it Remove	Restart kerneld	Quit

#### Slika 13.7

Okvir za dijalog Choose Module Type. Pomoću X Windowsa možete jednostavno dodavati module.

Choose Module Type			- 🗆 ×
Module Type			
Module Types eth			
	Ok	Cancel	

Želite li obrisati modul iz jezgre, pokrenite sljedeću naredbu

rmmod module-name

ili u popisu koji je prikazan na slici 13.6 odaberite modul i potom kliknite na gumb Remove.

### Ponovno pokretanje programa kerneld

Promjene koje ste unijeli pomoću alata Kernel Daemon Configuration se nalaze u datoteci /etc/conf.modules, koju program kerneld čita svaki puta kada se pokrene. U ispisu 13.2 se nalazi primjer jedne takve datoteke.

Ispis 12.2 datoteka /etc/conf.modules

alias scsi\_hostadapter BusLogic alias ethO ne2k-pci

Da biste ponovno pokrenuli kerneld, možete upotrijebiti alat prikazan na slici 13.6 i kliknuti na gumb Restart Kerneld. Također *demon* možete pokrenuti i iz retka za zadavanje naredbi, i to na ovaj način:

```
/etc/rc.d/init.d/kerneld stop
/etc/rc.d/init.d/kerneld start
```

Ponovnim pokretanjem programa kerneld neće se ponovno učitati moduli koji se već koriste, ali će podešenja biti prihvaćena kod budućih učitavanja modula.

### Odavde...

U ovom poglavlju ste nešto naučili o podešavanju i stvaranju nove jezgre za vaš sustav. Također ste naučili kako možete poboljšati funkcionalnost sustava dodavanjem modula. Dodatne informacije potražite u idućim poglavljima:

- 3. poglavlje, "Instaliranje Red Hata" sastoji se od detaljnih uputa za instaliranje inačice Red Hata.
- 5. poglavlje, "Pokretanje Linux programa" nudi vam osnovne upute za pokretanje raznih aplikacija na vašem Linux sustavu, kao što su kompajler i X Windowsi.
- 6. poglavlje, "Nadogradnja i instaliranje softvera pomoću RPM-a" sadrži informacije koje su vam potrebne da biste instalirali nove softverske pakete, među kojima su i oni potrebni za stvaranje koda nove jezgre, te neki kompajleri.
- 22. poglavlje, "Uporaba XWindowsa" sadrži sve potrebne informacije za korištenje XWindowsa pod operativnim sustavom Linux.

# 

# Upravljanje sustavom datoteka

(Jack Tackett)

### U ovom poglavlju

Sustavi datoteka 266 Postavljanje i uklanjanje sustava datoteka 269 Mrežni sustav datoteka 274 Održavanje sustava datoteka 278 Uporaba naredbe fsck 279 Stvaranje i formatiranje sustava datoteka 280 Uporaba swap datoteka i particija 288 Sustavi datoteka tvore osnovu za pohranu podataka na Linux računalima. Linux programi, biblioteke, sustavske datoteke i korisničke datoteke se nalaze u sustavima datoteka. Pravilno upravljanje sustavima datoteka je izuzetno bitno jer svi vaši programi i podaci počivaju na sustavima datoteka.

Mnogi koraci u uputama koje smo naveli u ovom poglavlju se automatski obavljaju kada instalirate Linux. No, ipak biste trebali naučiti kako se upravlja sustavom tako da možete stvarati, upravljati i dalje graditi svoj Linux sustav. Za upravljanje sustavom je nužno razumijevanje sustava datoteka. Sustav datoteka mora pravilno raditi da bi uopće radio čitav Linux sustav.

### Sustavi datoteka

U Linuxu je prostor na kojem se nalaze datoteke podijeljen na direktorije u obliku stabla, na vrhu kojeg se nalazi *root.* Slika 14.1 grafički prikazuje primjer jedne takve strukture direktorija.



Korisnici od tog stabla direktorija vide samo direktorije i datoteke. U stvarnosti mnogi direktoriji koji se nalaze unutar stabla su fizički smješteni na raznim dijelovima diska, na različitim diskovima ili čak računalima. Ako je jedna od tih particija na disku dodana u stablo u direktorij koji je poznat pod nazivom *mount point*, onda se takav direktorij i sve što se u njemu nalazi naziva sustav datoteka.

Operativni sustav Linux se sastoji od nekoliko direktorija i mnogo različitih datoteka. Ovisno o tome što ste odabrali tijekom instalacije sustava, ti direktoriji mogu predstavljati različite sustave datoteka. Obično se većina operativnih sustava sastoji od dva sustava datoteka: *root* (/) i sustav koji je izgrađen u direktoriju /usr.

Ako direktorij promijenite u direktorij *root* (pomoću naredbe cd /) i zatražite ispis direktorija, tada ćete ugledati nekoliko direktorija. Na taj ćete se način upoznati sa osnovnim sustavom datoteka (*root*) u kojem možete naći druge direktorije koji mogu predstavljati početni direktorij za razne sustave datoteka.

Direktorij /bin sadrži binarne programe koji predstavljaju osnovne sustavske datoteke. Mnoge Linux naredbe, primjerice naredba ls, su u osnovi programi koji se nalaze u tom direktoriju. Direktorij /sbin se koristi za spremanje binarnih sustavskih datoteka. Većina datoteka u ovom direktoriju se koristi za upravljanje sustavom.

Direktorij /etc je vrlo važan direktorij, jer sadrži mnoge datoteke za podešavanje sustava. U osnovi te datoteke operativnom sustavu Linux "daju osobnost". Datoteka za lozinke, passwd se nalazi u tom direktoriju, kao i datoteka fstab koja sadrži popis sustava datoteka koji se stvaraju prilikom pokretanja Linuxa. U tom se direktoriju nalaze skripte za pokretanje Linuxa, popis *hostova* s IP adresama koje često posjećujete, te još neke datoteke s podešenjima.

Dijeljene biblioteke koje koriste određeni programi su spremljene u direktoriju /lib. Mnogi programi višestruko koriste isti kod, a biblioteke služe sa pohranu takvog koda na zajedničkom mjesto te za uštedu u memoriji.

Direktorij /dev sadrži posebne datoteke koje se nazivaju datoteke uređaja i koriste za pristup različitim hardverskim komponentama u računalu. Primjerice, datoteka /dev/mouse služi za čitanje ulaza s miša. S obzirom da je u Linuxu na takav način organiziran pristup hardverskim komponentama, one izgledaju kao da se radi o softveru. To znači da u većini slučajeva možete koristiti isti oblik naredbe za softver i za hardver. Primjerice, da biste na traci stvorili arhivu svog početnog direktorija koji se nalazi na disketi, morate pokrenuti sljedeću naredbu:

tar -cdf /dev/fd0 ~tackett

Parametar /dev/fd0 upućuje na to da bi naredba tar trebala koristiti pogon za disketu pod imenom fd0.

### Vidi "Uslužni program tar", 229. stranica

Mnogi uređaji u direktoriju /dev su podijeljeni u logične grupe. U tablici 14.1 naći ćete popis najčešće korištenih uređaja u direktoriju /dev.

Datoteka uređaja	Opis
/dev/console	Konzola sustava je ustvari monitor računala.
/dev/hd	Upravljački program za IDE tvrde diskove. Uređaj /dev/hda1 predstavlja prvu particiju tvrdog diska hda. Uređaj /dev/hda predstavlja čitav disk hda.
/dev/sd	Upravljački program za SCSI tvrde diskove. Vrijedi isti način dodjele imena kao i za IDE diskove.
/dev/fd	Upravljački program za pogone disketa. /dev/fd0 je prvi pogon za diskete, a /dev/fd1 drugi pogon.
/dev/st	Upravljački program za SCSI pogon za trake.
/dev/tty	Upravljački programi koji sadrže nekoliko različiti korisničkih konzola. <i>nastavlja se</i>

Tablica	14.1	Uređaji u	direktoriiu	/dev

Tablica 14.1 Nastavak				
Datoteka uređaja	Opis			
	Ime dolazi iz vremena kada su terminali (teletypes) bili fizički povezani s UNIX sustavom. Pod Linuxom te datoteke pomažu virtualne konzole kojima možete pristupiti ako pritisnete neku od funkcijskih tipki <alt-f1> do <alt-f6>. Virtualne konzole dozvoljavaju odvojeni istovremeni pristup sustavu.</alt-f6></alt-f1>			
/dev/pty	Upravljački programi za pseudo terminale koji se koriste za daljinski rad (kao Telnet).			
/dev/ttyS	Serijski priključak na računalu. Uređaj /dev/ttyS0 odgovara priključku COM1 pod MS-DOS-om. Imate li serijskog miša, onda uređaj /dev/mouse predstavlja simboličku vezu s odgovarajućim uređajem na koji je spojen vaš miš.			
/dev/cua	Posebni uređaji koji se koriste s modemima.			
/dev/null	Poseban uređaj, u osnovi crna rupa. Svu podaci koje zapišete na uređaj /dev/null su zauvijek izgubljeni. Ovaj uređaj može biti koristan ako želite pokrenuti neku naredbu i isprobati standardni izlaz. Također, ako uređaj /dev/null upotrijebite kao ulaznu datoteku, stvorit ćete datoteku veličine nula.			

Direktorij /proc predstavlja virtualni sustav datoteka i koristi se za čitanje podataka iz memorije.

Direktorij /tmp se koristi za spremanje privremenih datoteka koje stvaraju programi tijekom rada. Ako koristite program koji stvara puno velikih privremenih datoteka, možda ćete trebati sagraditi direktorij /tmp kao zaseban sustav datoteka, a ne ga koristiti kao običan podirektorij u direktoriju *root.* Ako direktorij /tmp nije poseban sustav datoteka, može se dogoditi da, uslijed velikog broja privremenih datoteka, napunite sustav datoteka root.

Osnovni direktorij za korisničke početne direktorije je direktorij /home. On se uglavnom gradi kao zaseban sustav datoteka tako da korisnici imaju dovoljno mjesta za svoje datoteke. U osnovi, ako imate puno korisnika u svom sustavu, možda ćete direktorij /home trebati podijeliti u nekoliko sustava datoteka. Da biste to učinili, najprije direktorij /home podijelite na podirektorije /home/staff i home/admin (za osoblje i administratore sustava). Sagradite ta dva sustava datoteka i potom u njima stvorite korisničke početne direktorije.

Direktorij /var sadrži datoteke koje često mijenjaju svoju veličinu. Obično se u taj direktorij spremaju *log* datoteke raznih programa. Direktorij /var/spool i njegovi podirektoriji se upotrebljavaju za pohranu podataka koji nisu trajne naravi, primjerice pošta i novosti koje često učestalo dobivate s neke lokacije.

SAVJET

Ako želite možete stvoriti još jednu polaznu točku (mount point) u direktoriju root. Možda ćete trebati stvoriti direktorij /cdrom ako u sustavu koristite CD-ROM. Direktorij /usr i njegovi podirektoriji su vrlo važni za rad operativnog sustava Linux. Sadrže nekoliko direktorija s najvažnijim programima. Obično podirektoriji u direktoriju /usr sadrže velike softverske pakete koje ste instalirali na svoje računalo. Tablica 14.2 opisu neke od navedenih direktorija. Direktorij /usr gotovo uvijek predstavlja samostalni sustav datoteka.

Tablica 14.2	Važni podirektoriji u sustavu datoteka /usr
Podirektorij	Opis
/usr/bin	Sadrži mnoge izvršne datoteke programa.
/usr/etc	Sadrži mnoge dodatne datoteke za podešavanje sustava
/usr/include	U ovom direktoriju i njegovim podirektorijima možete pronaći sve datoteke za C kompajler u kojima se definiraju konstante i funkcije. Te su datoteke neophodne u programiranju.
/usr/g++include	Ovaj direktorij sadrži datoteke za C++ kompajler.
/usr/lib	U ovom se direktoriju nalaze razne biblioteke za programe koji se koriste za povezivanje.
/usr/man	U ovom ćete direktoriju naći mnoge upute za programe koji se koriste u Linux sustavu. Ispod direktorija /usr/man nalazi se nekoliko direktorija koji odgovaraju različitim skupinama man stranica.
/usr/src	U ovom direktoriju se nalazi osnovni kod raznih programa. Ako dobijete softverski paket kojeg želite instalirati, onda osnovni kod smjestite u direktorij /usr/src/imepaketa prije nego ga instalirate.
/usr/local	Direktorij je predviđen za prilagodbu vašeg sustava. Općenito u ovom direktoriju je instalirano mnogo lokalnog softvera. Oblik ovog direktorija varira ovisno o operativnom sustavu. Jedno od podešanja je da odaberete direktorij /usr/local/bin za binarne programe, /usr/local/etc za konfiguracijske datoteke, /usrl/local/lib za biblioteke i /usr/local/src za osnovni kod. Ako vam za direktorij /usr/local treba puno prostora, možete ga izdvojiti u zasebni sustav datoteka.

### Postavljanje i uklanjanje sustava datoteka

Do sada ste stekli uvid u to što je sustav datoteka. Pa kako onda možete direktorij podesiti kao odvojeni sustav datoteka?

Da biste uspostavili sustav datoteka u stablu direktorija, morate imati fizičku particiju, CD-ROM ili disketu. Također morate provjeriti postoji li doista direktorij kojem želite dodati sustav datoteka. Postavljanjem sustava datoteka nećete stvoriti takav direktorij (*mount point*), već on treba postojati prije nego počnete stvarati sustav datoteka. Pretpostavimo da želite u direktoriju /mnt stvoriti sustav datoteka za CD-ROM (pogon /dev/sr0). Mora postojati direktorij pod imenom /mnt, inače će postupak biti prekinut. Nakon što ste pod tim direktorijem stvorili sustav datoteka, sve datoteke i poddirektoriji s CD-ROM-a će se pojaviti u direktoriju /mnt. U protivnom je direktorij /mnt prazan.

**SAVJET** Pokrenite naredbu df. ako želite saznati kojem sustavu datoteka pripada direktorij u kojem se trenutno nalazite. Na ekranu ćete ugledati naziv sustava datoteka i slobodan prostor.

### Interaktivna uspostava sustava datoteka

Kao što već pogađate, za uspostavu sustava datoteka trebate upotrijebiti naredbu mount. Njezin oblik je ovakav:

mount device mountpoint

Parametar *device* predstavlja fizički uređaj (disk ili particija) na kojem trebate stvoriti sustav datoteka, a *mountpoint* je točka (direktorij) u stablu direktorija na kojem stvarate sustav datoteka.

BILJEŠKA Naredbu mount mogu koristiti isključivo superkorisnici jer je u protivnom narušen sustav sigurnosti. Postoji nekoliko softverskih paketa koji korisnicima omogućuju izradu posebnih sustava datoteka, kao što su sustavi na disketama.

Naredba mount koristi i još nekoliko argumenata osim onih koji su prethodno spomenuti (vidi tablicu 14.3). Ako nije navedena naredba koja vam je potrebna, program mount će je potražiti u datoteci /etc/fstab.

Tablica 14.3	Argument za naredbu mount
Argument	Opis
- f	Biti će obavljene sve zadaće osim pozivanja aktualnog sustava datoteka. Ova naredba "glumi" stvaranje sustava datoteka.
- V	Opširni način rada koji sadrži dodatne informacije o onome što naredba mount čini.
- W	Stvaranje sustava datoteka s dozvolom za čitanje i pisanje.
- r	Stvaranje sustava datoteka s dozvolom samo za čitanje.
- n	Stvaranje sustava datoteka bez zapisivanja u datoteku /etc/mtab.
-t <i>type</i>	Određivanje vrste sustava datoteka (minux, ext, ext2, xiafs, msdos, hpfs, proc, nfs, umsdos, sysv i iso9660 (osnovni tip).
- a	Stvaranje svih sustava datoteka koji se nalaze u datoteci /etc/fstab.

Argument	Opis
-o popis_opcija	Primjena većeg broja opcija odvojenih zarezom. Osim opcija koje se nalaze u ovoj tablici, možete upotrijebiti i dodatne o kojima nešto više možete saznati na man stranici programa mount.

**BILJEŠKA** Uobičajeno se koristiti nekoliko oblika naredbe mount. Primjerice, naredba mount /dev/hdb3 /mnt služi za izgradnju particije /dev/hdb3 pod direktorijem /mnt. Jednako tako, naredba mount -r -t iso9660/dev/sr0 /mnt služi za stvaranje SCSI pogona za CD-ROM /dev/sr0, koji se može samo čitati i čiji sustav datoteka je u obliku ISO 9660. Naredbom mount -vat nfs stvorit ćete sve mrežne sustave datoteka koji su zapisani u datoteci /etc/fstab.

**SAVJET** Ako sustav datoteka nije korektno postavljen, pokrenite naredbu mount –vf *device mount point* i vidjet ćete što naredba mount čini. Na taj ćete način na ekranu dobiti opširan ispis svih naredbi koje se koriste u stvaranju sustava datoteka, a naredba mount će obaviti sve radnje osim stvaranja sustava datoteka. Upotrijebite li ovu metodu, naredba mount će samo "glumiti" stvaranje sustava datoteka, a vi ćete puno saznati o samom procesu stvaranja sustava.

### Uspostava sustava datoteka tijekom pokretanja računala

U mnogim slučajevima sustav datoteka koji koristi Linux se ne mijenja tako često. Zbog toga možete lagano načiniti popis sustava datoteka koje Linux uspostavlja tijekom pokretanja sustava i uklanja na kraju rada. Te sustave datoteka ćete pronaći u posebnoj datoteci za podešavanje, koja se naziva /etc/fstab (što je skraćenica od file system table).

U datoteci /etc/fstab se nalazi popis sustava datoteka koje treba načiniti. U svakom retku datoteke je zapisan po jedan sustav. Polja u svakom retku su odvojena razmacima i tabulatorima. U tablici 14.4 naći ćete popis polja iz datoteka /etc/fstab.

Tablica 14.4 polja u	datoteci /etc/fstab
Polje	Opis
File system specifier	Određuje poseban uređaj ili udaljeni sustav datoteka koji treba postaviti.
Mount point	Određuje točku (direktorij) u stablu direktorija na kojoj će se naći sustav datoteka. Za posebne sustave datoteka, kao što su swap datoteke, koristi se riječ none, što znači da će swap datoteke biti aktivne, ali se neće vidjeti u stablu direktorija.
Туре	Određuje vrstu sustava datoteka. Postoje ove vrste sustava datoteka:
	minix: lokalni sustav datoteka koji podržava imena datoteka od 14 ili 30 znakova.

Tablica 14.4 Nastavak			
Polje	Opis		
	ext: lokalni sustav datoteka s duljim imenima datoteka (ovaj sustav datoteka treba biti zamijenjen sustavom		
	ext2 i više se neće koristiti).		
	ext2: lokalni sustav datoteka s duljim imenima datoteka i drugim značajkama.		
	xiafs: lokalni sustav datoteka		
	msdos: lokalni sustav datoteka za MS-DOS particije		
	hpfs: lokalni sustav datoteka za OS/2 HPFS particije		
	iso 9660: lokalni sustav datoteka koji se koristi za pogone CD-ROM-a		
	nfs: sustav datoteka na udaljenim računalima		
	swap: particija na disku ili posebna datoteka koja se koristi kao virtualna memorija		
	umsdos: UMSDOS sustav datoteka		
	sysv: sustav datoteka System V		
Mount Options	Popis opcija za stvaranje sustava datoteka koje su međusobno odvojene zarezom. Treba upotrijebiti barem opciju koja određuje vrstu sustava datoteka. Više o opcijama saznajte na man stranici za naredbu mount.		
Dump Frequency	Određuje učestalost spremanja sigurnosne kopije sustava (za pohranu se koristi naredba dump). Ako ovo polje nije napisano, naredba dump se neće izvoditi.		
Pass Number	Određuje redoslijed po kojem se provjeravaju sustavi datoteka (koristi se naredba fsck) tijekom podizanja sustava. Vrijednost 1 treba imati root sustav datoteka, a svi ostali sustavi datoteka bi trebali imati vrijednost 2. Ako vrijednost nije zadana, onda se tijekom pokretanja računala sustav datoteka neće provjeravati.		

**SAVJET** Preporučamo vam da gradite sustave datoteka tijekom podizanja računala koristeći datoteku /etc/fstab, a ne da upotrebljavate naredbu mount. Upamtite da to smiju činiti samo superkorisnici.

Slijedi primjer datoteke fstab:

# device	directory	type	options
/dev/hda1	/	ext2	defaults

/dev/hda2	/usr	ext2	defaults
/dev/hda3	none	swap	SW
/dev/sda1	/dosc	msdos	defaults
/proc	/proc	proc	none

U ovom primjeru može uočiti nekoliko različitih sustava datoteka. Najprije uočite da su komentari u datoteci označeni znakom #. U toj datoteci su naznačena dva normalna sustava datoteka: particije /dev/hda1 i /dev/hda2 koje su označene kao tip ext2 i nalaze se pod *root* direktorijem i pod direktorijem /usr.

Parametar default u polju options upućuje na to da sustav datoteka treba stvarati uz pomoć osnovnih opcija, a to su: treba omogućiti čitanje i pisanje u sustavu datoteka, sustav datoteka treba biti interpretiran kao poseban uređaj, ulaz-izlaz datoteka treba obavljati asinkronizirano, dozvoljeno je izvođenje binarnih datoteka, sustav datoteka se može stvoriti pomoću naredbe mount –a, podešeni korisnički broj i grupni identifikacijski broj interpretiraju se na tom sustavu datoteka i obični korisnici ne mogu stvarati taj sustav datoteka. Kao što možete vidjeti, puno je jednostavnije samo upisati parametar **defaults**.

Vidi "Stvaranje swap particije", 72. i 99. stranica

Particija /dev/hda3 je *swap* particija koja se koristi kao virtualna memorija. Niti jedan direktorij nije označen kao polazni (*mount point*) jer ne želite da se ova particija pojavljuje u stablu direktorija. *Swap* particija ipak mora biti u datoteci /etc/fstab kako bi sustav znao gdje se ona fizički nalazi. Swap particije koriste opciju sw.

Sustav datoteka /proc predstavlja virtualni sustav u memoriji. Kao što možete vidjeti, za taj sustav datoteka nije potreban fizički prostor na disku.

SAVJET Za cjelokupne informacije o svim opcijama koje se koriste u datoteci /etc/fstab potražite man stranicu za tu datoteku.

Sustave datoteka koji se koriste pod operativnim sustavom MS-DOS također automatski možete stvarati. Particija /dev/sda1 je prva particija na SCSI tvrdom disku sad. Ona je načinjena kao MS-DOS particija tipa msdos i s polaznim direktorijem (*mount point*) /dosc. Polazni direktorij za MS-DOS sustav datoteka može biti bilo koji direktorij i nije nužno da to bude *root* direktorij.

### Uklanjanje sustava datoteka

Sada kada ste naučili sve o stvaranju sustava datoteka, vrijeme je da se upoznate s njihovim uklanjanjem. Da biste uklonili sustave datoteka, upotrijebite naredbu umount. Sustave ćete trebati ukloniti iz više razloga: da biste mogli provjeriti/popraviti sustav datoteka (koristeći naredbu fsck), da biste uslijed problema na mreži mogli ukloniti NFS sustave datoteka ili sustav datoteka na disketi.

BILJEŠKA Radi se o naredbi umount, a ne "unmount". Pazite kako je pišete.

Postoje tri osnovna oblika naredbe umount:

```
umount device ¦ mountpoint
umount -a
umount -t fstype
```

Parametar *device* predstavlja ime fizičkog uređaja (particije) koji se uklanja, *mountpoint* je početni direktorij (upišite jedan ili drugi parametar). Naredba umount može sadržavati samo dva argumenta: -a koji služi za uklanjanje svih sustava datoteka i –t *fstype* koji djeluje samo na sustave datoteka određenog tipa.

### **UPOZORENJE**

Naredba umount ne uklanja sustav datoteka koji je u trenutnoj uporabi. Primjerice, ako postoje sustavi datoteka unutar direktorija /mnt, a vi želite pokrenuti naredbe:

cd /mnt umount /mnt

tada ste načinili pogrešku. Morate potražiti drugi direktorij u drugom sustavu datoteka da biste uklonili sustav datoteka u direktoriju /mnt.

### Mrežni sustav datoteka

Mrežni sustav datoteka (NFS) je sustav koji vam omogućuje izradu sustava datoteka na raznim računalima pomoću TCP/IP mreže. NFS omogućuje dijeljenje podataka među osobnim računalima, Macintosh, UNIX i Linux računalima. U mrežnom sustavu datoteka, sustav datoteka na udaljenom računalu korisnicima izgleda kao lokalni sustav datoteka. Osjećaj da se radi na lokalnom sustavu (a u tvari se koristi sustav na udaljenom računalu) daje korisnicima mnoge prednosti. Primjerice, u mreži može postojati računalo s puno prostora na disku. Takvo računalo može predstavljati poslužitelj za datoteke i na njemu se mogu nalaziti svi početni direktoriji korisnika. Pomoću NFS-a korisnici mogu pristupiti svoji direktorijima s bilo kojeg računala.

Mrežni sustav datoteka se sastoji od tri osnovne postavke:

- Računala na kojima se nalaze dijeljeni sustavi datoteka moraju međusobno komunicirati putem TCP/IP mreže.
- Računalo s lokalnim sustavom datoteka mora "dozvoliti" da druga računala uspostave taj sustav datoteka. Takvo računalo se naziva poslužitelj, a postupak kojim sustav datoteka postaje dostupan drugima se naziva *izvoz sustava datoteka*.
- Računalo koje želi izgraditi izvezeni sustav datoteka se naziva *računalo-klijent*. Takvo računalo, pomoću NFS-a, može sustav datoteka postaviti tijekom pokretanja računala (koristeći datoteku /etc/fstab) ili interaktivno (pomoću naredbe mount).

U idućim odlomcima ćete upoznati izvoz sustava datoteka i njihovo lokalno stvaranje.
## Izvoz NFS sustava datoteka

Da bi klijenti mogli uspostaviti sustav datoteka, sustav im treba biti dostupan na poslužitelju. Da bi sustav bio dostupan na poslužitelju, morate provjeriti je li taj sustav postavljen na računalu-poslužitelju.

Ako određeni sustav datoteka stalno izvozite, onda se on treba nalaziti u datoteci /etc/fstab na poslužitelju kako bi se automatski stvarao prilikom pokretanja poslužitelja.

Nakon što je sustav datoteka stvoren lokalno, morate omogućiti da on bude svima dostupan putem NFS-a. Riječ je o postupku koji se sastoji iz dva koraka. Najprije provjerite jeste li na poslužitelju pokrenuli *demone* rpc.mountd i rpc.nfsd. Ovi programi se najčešće pokreću pomoću pokretačke skripte /etc/rc.d/init.d/nfs. Provjerite nalaze li se u skripti sljedeći reci:

```
daemon rpc.mountd
daemon rpc.nfsd
```

**BILJEŠKA** Programima rpc.mountd i rpc.nfsd ne se može upravljati putem *demona* inetd jer se oni pokreću tijekom podizanja sustava i registriraju s *demonom* portmap. Stoga programe pokrećite tek nakon što ste pokrenuli *demon* rcp.portmap.

Drugi korak je da upišete NFS sustav datoteka u datoteku /etc/exports. Ova datoteka sadrži podatke o sustavima datoteka koji se izvoze, računalima koja im mogu pristupiti te podatke o vrsti i razini dozvoljenog pristupa.

## Datoteka /etc/exports

Datoteku /etc/exports koriste programi mountd i nfsd za određivanje sustava datoteka koji se trebaju izvoziti te za uvid u njihova ograničenja. Sustavi datoteka su u datoteci /etc/exports poredani jedan ispod drugoga. U svakom retku se najprije nalazi ime ishodišnog direktorija (*mount point*) lokalnog sustava datoteka, a zatim slijedi popis računala koja imaju dozvolu uspostave tog sustava datoteka. Iza svakog imena sustava može se nalaziti zagrada s popisom opcija koje su međusobno odvojene zarezima. U tablici 14.5 vidite popis opcija koje se mogu koristiti u datoteci /etc/exports.

	Opcija	Opis
	insecure	Dozvola neautoriziranog pristupa s pojedinog računala.
	secure	Potrebna autorizacija s pojedinog računala.
	root_squash	Praćenje zahtjeva iz root direktorija, UID 0 na računalu-klijentu, UID NOBODY_UID na poslužitelju.
	no_root_squash	Ne prate se zahtjevi sa UID 0 (osnovno podešenje).
	ro	Stvaranje sustava datoteka samo za čitanje (osnovno podešenje).
	rw	Stvaranje sustava datoteka za čitanje i pisanje. nastavlja se

	Tablica 14.5	Opcije koje se mogu	koristiti u datoteci /etc/exports
--	--------------	---------------------	-----------------------------------

Tablica 14.5 Nastavak			
Opcija	Opis		
link_relative	Pretvaranje apsolutnih veza (veza koja počinje s kosom crtom) u relativne veze (ispred veze se umeće potreban broj oznaka/).		
link_absolute	Ostavlja sve simboličke veze onakvima kakve jesu (osnovno podešenje za Sun NFS poslužitelje). Osnovno podešenje za Linux.		
map_daemon	Praćenje lokalnih i udaljenih imena i identifikacijskih brojeva pomoću demona lname/uid map na računalima-klijentima. Opcija se koristi za praćenje između UID prostora na računalu -klijentu i poslužitelju.		
all-squash	Praćenje svih UID-a i GID-a za anonimnog korisnika. Opcija je korisna za javne direktorije kao što su direktoriji za FTP i vijesti.		
no-all-squash	Suprotna opcija od opcije all-squash (osnovno podešenje u Linuxu).		
squash-uids	Određuje popis UID-a za anonimnog korisnika. Jedan popis identifikacijskih brojeva bi trebao ovako izgledati: squash uids=0-15,20,25-50		
squash-gids	Određuje popis GID-a za anonimnog korisnika. Jedan popis identifikacijskih brojeva bi trebao ovako izgledati: squash uids=0-15,20,25-50		
anonuid	Postavlja korisnički broj (UID) za anonimni račun. Opcija je korisna za PC/NFS klijente.		
anongid	Postavlja grupni korisnički broj (GID) za anonimni račun. Opcija je korisna za PC/NFS klijente.		
noaccess	Koristi se za isključivanje određenih poddirektorija tako da klijent ne može pristupiti sadržaju u pojedinom direktoriju.		

Slijedi primjer datoteke /etc/exports:

```
/home bill.tristar.com(sw) fred.tristar.com(rw)

  george.tristar.com(rw)
/usr/local/bin/bin *.tristar.com(ro)
/projects develop.tristar.com(rw) bill.tristar.com(ro)
/pub (ro,insecure,root_squash)
```

U ovom primjeru poslužitelj izvozi tri različita sustava datoteka. Sustav /home je načinjen s pristupom za čitanje i pisanje i može mu se pristupiti s tri različita računala: bill, fred i george. Prema imenima direktorija reklo bi se da se radi o tri korisnička direktorija. Sustav /usr/local/bin se izvozi samo za čitanje i može mu pristupiti bilo koje računalo iz tristar.com domene.

Sustav datoteka /projects se izvozi s dozvolom čitanja i pisanja za računalo develop.tristar.com, dok računalo bill.tristar.com ima samo dozvolu čitanja sustava datoteka.

Za sustav datoteka /pub ne postoji popis računala kojima je dozvoljen pristup što upućuje na zaključak da se bilo tko može poslužiti tim sustavom datoteka. Sustav se izvozi samo za čitanje i dozvoljen mu je neautorizirani pristup. Poslužitelj prati sve zahtjeve za pristup tome sustavu datoteka, koji su upućeni s direktorija *root* na udaljenom računalu.

## Uspostava NFS sustava datoteka

Uspostava NFS sustava datoteka je slična postavljanju bilo kojeg drugog sustava datoteka. NFS sustav datoteka možete postavljati tijekom pokretanja sustava iz datoteke /etc/fstab ili interaktivno pomoću naredbe mount.

#### **UPOZORENJE**

Prilikom uporabe naredbe mount ili u zapisu unutar datoteke /etc/fstab morate dvotočkom odvojiti ime *hosta* i ime udaljenog sustava datoteka:

```
mailserver:/var/spool/mail
```

Ne odvojite li ime hosta od direktorija, vaše računalo neće moći pravilno stvoriti udaljeni direktorij.

**Uspostava NFS sustava datoteka putem datoteke /etc/fstab** Kada u datoteci /etc/fstab specificirate NFS sustav datoteka, trebate upotrijebiti ovaj oblik

imehosta:/datoteka/sustav/staza

gdje *imehosta* zamjenjuje ime poslužitelja na kojem se nalazi sustav datoteka, a /datoteka/sustav/staza predstavlja sustav datoteka na poslužitelju.

Vrsta sustava datoteka se specificira kao nfs. U tablici 14.6 možete vidjeti popis najčešćih opcija naredbe mount.

Tablica 14.6	vajčešće korištene opcije za uspostavu mrežnih sustava datoteka	
Opcija	Opis	
rsize=n	Određivanje veličine u bajtima koju NFS klijenti koriste za čitanje. Osnovna vrijednost je 1024 bajta.	
wsize=n	Određivanje veličine u bajtima koju NFS klijenti koriste za pisanje. Osnovna vrijednost je 1024 bajta.	
timeo= <i>n</i>	Podešavanje vremenskog intervala čekanja NFS klijenta (u desetinkama sekundi). Osnovna vrijednost je 0.7 sekundi.	
hard	"Tvrda" uspostava sustava datoteka (osnovno podešena).	
soft	"Mekana" uspostava sustava datoteka.	
intr	Omogućuje signalima prekidanje NFS poziva. Korisna opcija za otkazivanje operacije kada NFS poslužitelj ne reagira.	

#### "Tvrda" u odnosu na "mekanu" uspostavu sustava datoteka

"Tvrda", odnosno "mekana" uspostava sustava datoteka određuju način ponašanja NFS klijenta u slučaju kada NFS poslužitelj prestane reagirati na zahtjeve. NFS sustavi datoteka su podešeni za "tvrdu" postavu. Kod bilo koje vrste, ako poslužitelj prestane reagirati na zahtjeve, računalo-klijent čeka određeno vrijeme (zadano pomoću opcije timeo) i zatim ponovno pošalje zahtjev (ovo se naziva *mali prekid*). Ako zahtjevi ne dobiju odgovor u roku 60 sekundi, onda nastupa *veliki prekid*.

Ako je sustav datoteka "tvrdo" postavljen, računalo-klijent će ispisati poruku i početi ponovno slati zahtjeve, ali s dvostruko dužim vremenom prekida. Taj proces može trajati beskonačno dugo. Računalo-klijent pokušava slati zahtjeve za NFS sustavom datoteka sve dok ga ne dobije.

S druge strane, ako se radi o "mekanoj" postavi sustava datoteka, čim nastupi veliki prekid, računalo javlja poruku o grešci i nastavlja raditi na standardni način.

Obično se za važnije softverske pakete i programe koristi "tvrda" postava sustava datoteka i zbog toga je takvo podešenje osnovno postavljeno. Sigurno ne želite da se vaše računalo počne čudno ponašati samo zato što je trenutno prekinuta veza, već naprotiv, Linux treba sačekati i nastaviti s traženjem čim se veza uspostavi. S druge strane, možda ćete za manje bitne podatke trebati "mekanu" postavu sustava datoteka. Ako se u tom slučaju ne bude mogao izgraditi sustav datoteka, nećete izgubiti vezu s poslužiteljem.

Evo kako izgleda jedan od zapisa u datoteci /etc/fstab:

mailserver:/var/spool/mail /var/spool/mail nfs timeo=20,intr

Sustav datoteka /var/spool/mail se nalazi na poslužitelju za poštu i postavlja se na lokalnom direktoriju /var/spool/mail. Vrsta sustava datoteka je nfs, a vrijeme čekanja 2 sekunde (20 desetinki). Operacije na tom sustavu datoteka mogu se prekidati (opcija intr).

**Interaktivna uspostava NFS sustava datoteka** NFS sustavi datoteka mogu se graditi interaktivno, kao i bilo koji drugi sustavi datoteka. No, ipak biste trebali izbjegavati neke opcije naredbe mount ako stvarate NFS sustav datoteka.

Uzmemo li prethodni primjer, onda bi ovako trebala izgledati naredba mount:

# mount -t nfs -o timeo=20,intr mailserver:/var/spool/mail /var/spool/mail

Naredba mount može postati vrlo složena ako želite zadati vremenske intervale za čekanje. Stoga vam preporučamo da NFS sustave datoteka postavljate automatski tijekom pokretanja sustava koristeći datoteku /etc/fstab.

# Održavanje sustava datoteka

Kao administrator sustava vi ste odgovorni za održavanje cjelovitosti sustava datoteka. To u osnovi znači da povremeno trebate provjeriti sustave datoteka te vidjeti ima li oštećenih datoteka. Tijekom podizanja sustava Linux automatski provjerava sustave datoteka ako neki od sustava ima vrijednost veću od 0 u polju *pass number*, koje se nalazi u datoteci /etc/fstab. BILJEŠKA Sustav datoteka ext2, koji se često koristi na Linux računalima, ima posebnu zastavicu *clean bit.* Ako se sustav datoteka korektno sinkroniziran i razgrađen, onda je ta zastavica postavljena na sustav datoteka. Ako je zastavica postavljena dok se Linux pokreće, on neće provjeravati cjelovitost sustava datoteka.

# Uporaba naredbe fsck

Povremeno je dobro provjeriti sustave datoteka kako bi se ustanovila eventualna oštećenja. U Slackware distribuciji Linuxa postoji naredba fsck (provjera sustava datoteka) koja služi za provjeravanje sustava datoteka. Radi se ustvari o seriji naredbi koje služe za provjeravanje sustava. Evo kako izgleda oblik naredbe fsck:

fsck [-A] [-V] [-t fs-type] [-a] [-1] [-r] [-s] filesys

No najčešće se koristi osnovni oblik naredbe:

fsck filesys

Tablica 14.7 opisuje pojedine opcije naredbe fsck

Tablica 14.7	gumenti za naredbu fsck	
Argument	Opis	
- A	U jednom prolazu pokušava provjeriti sve sustave datoteka koji su navedeni u datoteci /etc/fstab. Ova se opcija obično koristi tijekom pokretanja Linuxa za provjeru svih normalnih sustava datoteka. Upotrijebite li opciju –A, nećete moći koristiti argument filesys.	
- V	Opširni način rada. Ispis dodatnih informacija o tome što radi naredba fsck	
-t fs-type	Određivanje vrste sustava datoteka koje treba provjeravati.	
filesys	Određuje sustav datoteka koji se provjerava. Kao argument se može koristiti ime posebnog uređaja, primjerice /dev/hda1 ili početni direktorij (mount point) kao /usr.	
- a	Automatski se otklanjaju problemi u sustavu datoteka (bez dodatnih pitanja). Ovu opciju koristite uz dosta opreza.	
- ]	Popis svih imena datoteka u sustavu datoteka.	
- r	Potrebna je potvrda prije popravljanja sustava datoteka.	
- S	Ispis superbloka prije provjere sustava datoteka.	

Naredba fsck je ustvari program koji poziva naredbu za odabranog sustava datoteka. No, potrebno je zadati vrstu sustava datoteka koji se provjerava. Da biste bili sigurni da program fsck poziva pravu naredbu za određeni sustav datoteka, upišite parametar –t uz naredbu fsck. Bez tog parametra Linux pokušava odrediti sustav datoteka iz datoteke /etc/fstab. Ako u toj datoteci ne naiđe na potrebnu informaciju o tipu sustava datoteka, program će pretpostaviti da koristite sustav datoteka Minix.

#### **UPOZORENJE**

Naredba fsck pretpostavlja da provjeravate sustav datoteka tipa Minix, ako ne zadate drugu vrstu sustava koristeći parametar –t ili pomoću zapisa u datoteci /etc/fstab. S obzirom da su sustavi datoteka na vašem računalu vjerojatno tipa ext2, a ne Minix, morate pažljivo zadati pravu vrstu sustava datoteka. To je osobito važno ako provjeravate sustav datoteka koji nije naveden u datoteci etc/fstab.

Dobro je razgraditi sustav datoteka prije same provjere. Na taj ćete način osigurati da se niti jedna datoteka iz sustava ne koristi tijekom provjere.

BILJEŠKA Upamtite da ne možete razgraditi sustav datoteka ako je jedna od njegovih datoteka aktivna. Primjerice, ako se neki korisnik trenutno nalazi u direktoriju sustava koji namjeravate razgraditi, dobit ćete poruku koja kaže da je sustav datoteka zauzet.

Dodatni problem predstavlja provjera sustava datoteka *root*. Ne možete izravno razgraditi ovaj sustav datoteka jer Linuxu treba ostaviti mogućnost pristupa na taj direktorij. Da biste provjerili sustav datoteka *root*, trebali biste računalo podići s diskete koja sadrži taj sustav datoteka i zatim pokrenuti naredbu fsck na stvarnom *rootu* tako da upišete ime specijalnog uređaja na kojem se nalazi sustav *root*. Ako naredba fsck načini bilo kakve promjene na vašem sustavu datoteka, tada obavezno ponovno pokrenite sustav i to neposredno nakon provjere. Na taj ćete način omogućiti Linuxu učitavanje važnih informacija o vašem sustavu datoteka i spriječiti njegovo oštećenje.

#### **UPOZORENJE**

Odmah nakon što ste pokrenuli naredbu fsck, trebate ponovno podići sustav tako da biste spriječili oštećenje sustava datoteka. Za ponovno podizanje sustava upotrijebite naredbu shutdown –r.

# Stvaranje i formatiranje sustava datoteka

Nakon što dodate novi tvrdi disku u svoje računalo ili ako želite mijenjati particije na starom tvrdom disku, tada na "sirovom" disku trebate stvoriti sustav datoteka. Pretpostavimo da ste dodavali novi disk u računalo. U tom slučaju morate podesiti particije na disku i zatim stvoriti sustave datoteka tako da ih Linux može koristiti. Promjenu podataka vezanih uz particiju na disku obavljate pomoću naredbe fdisk. Nakon što tvrdi disk podijelite na particije, moći ćete pomoću naredbe mkfs stvoriti sustave datoteka.

## Uporaba naredbe fdisk za stvaranje particija na disku

Naredba fdisk se koristi za stvaranje particija na disku i podešavanje atributa koji određuju vrstu sustava datoteka na određenoj particiji. Ako ste Linux instalirali na računalo koje je radilo pod MS-DOS-om, morali ste pokrenuti naredbu fdisk i promijeniti podatke koji se odnose na pojedinu particiju. Nakon toga ste mogli instalirati Linux.

#### **UPOZORENJE**

Uporabom naredbe fdisk možete uništiti sve podatke na disku. S obzirom da ta naredba briše tablicu datoteka, sve se datoteke mogu izgubiti. Stoga provjerite imate li kompletnu kopiju disk prije nego pokrenete naredbu fdisk.

Naredbu fdisk uvijek treba pokrenuti prije nego stvorite sustav datoteka. Ne radi se o pojedinačnoj naredbi, već o interaktivnom programu s izbornicima. Za pokretanje programa fdisk upišite sljedeće:

fdisk [pogon]

Riječ *pogon* predstavlja fizički pogon na disku na kojem namjeravate raditi. Ne odaberete li pogon, sustav pretpostavlja pogon /dev/hda. Evo primjera za pokretanje naredbe fdisk na drugom IDE tvrdom disku:

fdisk /dev/hdb

S obzirom da se u programu fdisk koriste izbornici, postoji nekoliko naredbi koje se mogu koristiti, a čiji je popis dat u tablici 14.8.

lablica 14.8	Tablica 14.8 Linux naredbe u programu fdisk	
Naredba	Opis	
a	Uključivanje/isključivanje zastavica za podizanje sustava	
С	Uključivanje/isključivanje zastavica za DOS-uskladivost	
d	Briše particiju	
1	Popis poznatih vrsti particija	
m	Prikaz izbornika s naredbama	
n	Dodaje novu particiju	
р	Prikaz tablice particije	
q	Izlazak bez spremanja promjena	
t	Promjena vrste sustava datoteka za pojedinu particiju	
u	Promjena jedinica za prikaz/unos	
V	Potvrda tablice particije	
W	Zapis tablice na disk i izlazak iz programa	
Х	Omogućuje dodatne funkcije (samo za stručnjake):	
	b Premještanje početne lokacije podataka	
	C Promjena broja cilindara	I SP

Tablica 14.8	Nastavak	
Naredba	Opis	
	d Ispis neobrađenih podataka u tablicu podataka	
	e Popis proširenih particija na disku	
	h Promjena broja glava na disku	
	r Povratak u glavni izbornik	
	s Promjena broja sektora na disku	

Naredba fdisk služi za podešavanje sustava datoteka koji se nalazi na određenoj particiji. Za stvaranje particija koje se koriste pod Linuxom trebate koristiti isključivo naredbu fdisk. Za MS-DOS particije i OS/2 particije trebate koristiti alate za odgovarajuće operativne sustave, a nakon toga pokrenuti naredbu fdisk za Linux te označiti one particije koje se koriste pod Linuxom.

Tablica 14.9 sadrži popis particija koje možete kreirati pomoću naredbe fdisk. Svaka vrsta particije označena je heksadecimalnim kodom koji trebate upisati u naredbu fdisk prilikom podešavanja particije.

Tablica 14.9 Vrs	te Linux particija
Referentni broj	Vrsta
0	prazna
1	DOS 12-bit FAT
2	XENIX root
3	XENIX usr
4	DOS 16-bit<32M
5	proširena
6	DOS 16-bit>=32M
7	OS/2 HPFS
8	AIX
9	AIX za pokretanje sustava
a	OS/2 Boot Manager
40	Venix 80286
51	Novell?
52	Microport

Referentni broj	Vrsta
63	GNU HURD
64	Novell NetWare
65	Novell Netware
75	PC/IX
80	Old MINIX
81	MINIX/Linux
82	Linux Swap
83	Linux Native
93	Amoeba
94	Amoeba BBT
a5	BSD/386
b7	BSDI sustav datoteka
b8	BSDI swap
c7	Syrinx
db	CP/M
el	DOS access
e3	DOS R/O
f2	DOS secondary
ff	BBT

U nastavku će biti objašnjeno kako se koristi program fdisk. Slijedi primjer uporabe programa za podešavanje particija na tvrdom disku koje će se koristiti pod operativnim sustavom Linux. Prije svega provjerite imate li sigurnosne kopije podataka na disku jer će svi podaci biti uništeni. Ime prvog tvrdog IDE diska je /dev/hda (osnovni disk za Linux).

Pokretanje programa fdisk Program se pokreće pomoću naredbe

# fdisk

Na ekranu će se zatim pojaviti sljedeće:

Using /dev/hda as default device! Command (m for help):

što znači da program fdisk koristi disk /dev/hda kao uređaj na kojem trenutno radite. Možete biti zadovoljni jer ste to i tražili. Uvijek morate provjeriti nalazite li se doista na disku na kojem ste se željeli naći. Linux će zatim prikazati prompt naredbe fdisk.

**Prikaz tekuće tablice particije** Prvo što trebate učiniti je prikazati postojeću tablicu particije. To možete učiniti pomoću naredbe p:

```
Command (m for help): p
Disk /dev/hda: 14 heads, 17 sectors, 1024 cylinders
Units = cylinders of 238 * 512 bytes
Device Boot Begin Start End Blocks Id System
Command (m for help):
```

Uočite da na postojećem disku /dev/hda postoji 14 glava, 17 sektora i 1024 cilindra. Jedinice (cilindri) se sastoje od 238\*512 (121856) bajta. S obzirom da postoji 1024 cilindra sa 121856 bajta, to znači da disk može sadržavati 1024x121856=124780544 bajta ili oko 120 MB. Možete također vidjeti da na disku /dev/hda nema particija.

**Stvaranje nove particije** Pretpostavimo da želite stvoriti particiju veličine 100 MB koja će služiti za korisničke početne direktorije, te *swap* particiju veličine 20 MB. Idući korak je da pokrenete naredbu n za stvaranje nove particije:

```
Command (m for help): n

Command action

e extended

p primary partition (1-4)

p

Partition number (1-4): 1

First cylinder (1-1023): 1

Last cylinder or +size or +sizeM or +sizeK (1-1023): +100M
```

Pokrenete li naredbu n za stvaranje nove particije, na ekranu će se pojaviti drugi izbornik. Trebate odrediti želite li stvoriti proširenu ili primarnu particiju. Obično se radi o primarnim particijama, osim ako na disku nemate više od četiri particije. Potom trebate upisati broj particije koju želite stvoriti. S obzirom da je navedena particija prva na disku, upišite broj 1. Nakon toga unesite broj početnog cilindra nove particije, odnosno odredite mjesto na kojem počinje unos podataka. S obzirom da se radi o prvoj particiji na disku, možete particiju započeti na prvom cilindru.

U idućem retku trebate upisati veličinu particije, što možete učiniti na nekoliko različitih načina. Možete upisati broj koji predstavlja veličinu u cilindrima, bajtima, kB ili MB. S obzirom da želite zadati veličinu particije od 100 MB, onda upišite +100MB.

**Provjera tablice particije** Sada biste opet trebali provjeriti tablicu particije i vidjeti što je dosad napravljeno:

```
Command (m for help): p
Disk /dev/hda: 14 heads, 17 sectors, 1024 cylinders
Units = cylinders of 238 * 512 bytes
                  Begin Start
                                        Blocks Id
Device
            Boot
                                 End
                                                       System
                         1
                                 861 102400
                                                  81
                                                       Linux/MINIX
/dev/hda1
                    1
Command (m for help):
```

Tablica particije pokazuje da imate jednu particiju /dev/hda1, koja počinje od prvog cilindra i završava s cilindrom 861, te koristi 102400 blokova. Tip particije je zadan brojem 81 (Linux/MINIX).

**Stvaranje swap particije** Sada trebate stvoriti *swap* particiju veličine 20 MB. Postupak je sličan kao kod stvaranja prve particije:

```
Command (m for help): n
Command action
e extended
p primary partition (1-4)
p
Partition number (1-4): 2
First cylinder (862-1023): 862
Last cylinder or +size or +sizeM or +sizeK (1-1023): 1023
```

**SAVJET** Obično je bolje veličinu zadnje particije unijeti u cilindrima kako biste bili sigurni da ste iskoristili sav prostor na disku.

Za drugu ste particiju unijeli broj 2. Uočite da se za položaj prvog cilindra nudi područje između 862. i 1023. cilindra. Razlog tome je što prva particija koristi prethodne cilindre. Upišite broj 862 kao početni cilindar duge particije. Vjerojatno želite sav preostali prostor na disku iskoristiti za *swap* particiju. Ostalo vam je negdje oko 20 MB prostora, no ako veličinu zadate u MB, možda će par cilindara ostati neiskorišteno. Upišite broj 1023 koji predstavlja zadnji cilindar druge particije.

BILJEŠKA Možda će se pojaviti poruka koja ukazuje na grešku

```
Warning: Linux cannot currently use the last xxx sectors of this partition.
```

(oznaka xxx je, ustvari, broj sektora). Takve poruke možete zanemariti jer su ostale iz vremena kada Linux nije mogao pristupiti sustavima datoteka većim od 64 MB. ■

**Provjera veličine particije** Do sada ste stvorili obje particije. Trebali biste pogledati tablicu particija i provjeriti imaju li odgovarajuću veličinu:

```
Command (m for help): p
Disk /dev/hda: 14 heads, 17 sectors, 1024 cylinders
Units = cylinders of 238 * 512 bytes
          Boot
                 Begin Start End
Device
                                      Blocks
                                                Ιd
                                                     System
/dev/hda1
                  1
                         1
                               861
                                     102400
                                               81
                                                     Linux/MINIX
                                               81
                    862 862 1023
                                      19159
                                                    Linux/MINIX
/dev/hda2
Command (m for help):
```

Kao što možete vidjeti, particija /dev/hda1 koristi cilindre od 1 do 861 (veličina 102400 blokova) i zauzima pribižno 100 MB. Particija /dev/hda2 se nalazi na području diska od 862. cilindra do 1023. cilindra, i velika je 19156 blokova ili gotovo 20 MB.

**Promjena vrste particije** Iduću stvar koju trebate napraviti je promijenite vrstu particije za svaku opd particija. Naredba t služi za mijenjanje vrste particije. Obično se particija sa standardnim sustavom datoteka podešava tako da se zada broj 83, Linux native, a *swap* particija se podešava uporabom tipa Linux swap, odnosno zadavanjem broja 82.

```
Command (m for help): t
Partition number (1-4): 1
Hex code (type L to list codes): 83
Command (m for help): t
Partition number (1-4): 2
Hex code (type L to list codes): 82
```

Pokrenete li naredbu t, trebat ćete upisati broj particije koju želite mijenjati. Trebate upisati heksadecimalni identifikacijski kod koji označava vrstu particije. Particije s normalnim sustavom datoteka se u Linux označavaju brojem 83, a *swap* particije brojem 82. Želite li vidjeti popis sustava datoteka, upišite broj 1.

**Dovršetak stvaranja particija** Sada kada ste stvorili particije i označili ih, trebali biste još jednom provjeriti tablicu particija i uvjeriti se da je sve u redu.

```
Command (m for help): p
Disk /dev/hda: 14 heads, 17 sectors, 1024 cylinders
Units = cylinders of 238 * 512 bytes
Device Boot Begin Start End Blocks Id System
/dev/hda1 1 1 861 102400 83 Linux/NATIVE
/dev/hda2 862 862 1023 19159 82 Linux/SWAP
Command (m for help):
```

Kao što možete vidjeti, particije se nalaze na pravom mjestu, odgovarajuće su veličine i vrste sustava datoteka su pravilno podešeni. Zadnje što trebate učiniti je pokrenuti naredbu w i za pisati tablicu particija na disk:

Command (m for help): w #

Niti jedna od prethodnih promjena neće imati efekt ako je niste zapisali na disk pomoću naredbe w. Uvijek imate mogućnost napustiti program pomoću naredbe q i ne pohraniti promjene na disk. Uvijek biste trebali imati rezervnu kopiju diska koji mijenjate pomoću programa fdisk.

Nakon što unesete promjene na disk, ponovno pokrenite sustav kako biste bili sigurni da je Linux ažurirao podatke o particijama koji se nalaze u jezgri.

## Uporaba naredbe mkfs za stvaranje sustava datoteka

Nakon što stvorite particiju pomoću naredbe fdisk, morate sagraditi sustav datoteke prije nego što ga počnete koristiti za pohranu podataka. To možete učiniti pomoću naredbe mkfs. Zamislite da uređujete parkiralište. Program fdisk fizički stvara "zamišljeno parkiralište", a program mkfs "iscrtava linije" kako bi vozači znali gdje trebaju parkirati svoje vozilo.

Kao što je program fsck namijenjen za provjeru vrste sustava datoteka, naredba mkfs poziva razne programe za stvaranje sustava datoteka, ovisno o vrsti sustava koji želite stvoriti. Sintaksa naredbe mkfs je sljedeća:

mkfs [-V] [-t fs-type] [fs-options] filesys [blocks]

gdje je *filesys* oznaka particija na kojoj želite načiniti sustav datoteka, primjerice /dev/hda1.

#### **UPOZORENJE**

Naredba mkfs prihvaća kao ime sustava datoteka i ime kao što je /home, no trebate biti iznimno pažljivi kod takve uporabe naredbe. Pokrenete li naredbu mkfs na "živom" sustavu datoteka, vrlo lako se može dogoditi da oštetite neke podatke.

Tablica 14.10 sadrži popis raznih parametara naredbe mkfs.

Tablica 14.10	Parametri za naredbu mkfs		
Орсіја	Opis		
- V	Služi za opsežan izlaz naredbe u kojem se nalaze sve naredbe koje su do tada pokrenute. Odaberete li ovu opciju više puta, zaustavit će se izvođenje bilo koje naredbe koja se odnosi na sustav datoteka.		
-t fs-type	Određuje vrstu sustava datoteka. Ako niste odabrali niti jedan tip sustava, program će pretražiti datoteku /etc/fstab kako bi pronašao odgovarajuću vrijednost varijable filesys. Ne nađe li potrebnu vrijednost, program stvara MINIX datoteku.		
fs-options	Određuju se posebne sustavske opcije koje se koriste tijekom stvaranja sustava datoteka. Iako to nije uvijek slučaj, većinom se koriste ove posebne opcije:		
	- C Prije stvaranja sustava datoteka provjerava se ima li loših blokova,		
	- ] <i>file-name</i> Čitanje popisa loših blokova na disku iz datoteke file-name,		
	- V Program koji stvara sustav datoteka daje opširan izlaz.		
filesys	Ovo je nužan parametar i predstavlja particiju na kojoj se gradi sustav datoteka.		
blocks	Predstavlja broj blokova koji će se koristiti za sustav datoteka.		

Iako je proizvoljna uporaba parametra –<br/>tfs-type, trebali biste steći naviku određivanja vrste sustava datoteka. Kao i na<br/>redba fsck, naredba mkfs pokušava "odgonetnuti" vrstu sustava

datoteka iz datoteke /etc/fstab. Ako to nije moguće, stvorit će se sustav datoteka MINIX. U većini slučajeva ćete ipak odabrati vrstu sustava ext2.

# Uporaba swap datoteka i particija

*Swap* prostor na disku se koristi kao virtualna memorija. O samoj virtualnoj memoriji neće biti govora u ovoj knjizi jer je o tome napisano dosta knjiga koje govore o načinu rada operativnih sustava.

Linux podržava dvije vrste *swap* prostora: particiju i datoteku. *Swap* particija predstavlja stvarnu, fizičku particiju na disku s identifikacijskim brojem sustava datoteka 82. *Swap* datoteka je velika datoteka na normalnom sustavu datoteka koja se koristi kao *swap* prostor.

Bolje je koristiti *swap* particiju, a ne datoteku. Pristup *swap* datoteci je putem normalnog sustava datoteka. U tom slučaju će pojedini blokovi datoteke biti nepovezani, a

karakteristike same datoteke neće biti tako dobre kao kod *swap* particije. Ulazno-izlazna komunikacija sa *swap* particijom se obavlja izravno, a blokovi na disku su međusobno povezani. Osim toga, ako *swap* prostor smjestite izvan normalnog sustava datoteka, smanjit ćete mogućnost da se ošteti sustav datoteka nekom greškom.

## Stvaranje swap particije

Da biste stvorili *swap* particiju, morate najprije stvoriti particiju na disku pomoću naredbe fdsik i označiti je kao tip 82, Linux swap. Nakon što stvorite *swap* particiju, na raspolaganju su vam dva dodatna koraka za aktiviranja particije.

Najprije morate pripremiti particiju na sličan način kao što stvarate sustav datoteka. Umjesto naredbe mkfs, za pripremu particije se korisni naredba mkswap, koja ima sljedeći oblik:

mkswap [-c] particija veličina particije u blokovima

gdje parametar *particija* predstavlja ime *swap* particije, kao što je /dev/hda2, a *veličina u blokovima* predstavlja veličinu particije izraženu u blokovima. Veličinu izraženu u blokovima možete saznati tako da pokrenete naredbu fdisk i pogledate tablicu particija. U prethodnom primjeru veličina datoteke je 19159 blokova. Linux zahtijeva da veličina *swap* particija bude između 9 i 65537 blokova. Parametar –c služi za provjeru sustava datoteka i traženje eventualnih loših blokova.

Sljedeći prethodni primjer swap particije upišite naredbu za podešavanje swap particije:

mkswap -c /dev/hda2 19159

Nakon što ste pokrenuli naredbu mkswap, morate *swap* particiju aktivirati tako da je jezgra Linuxa može koristiti. Naredba za aktiviranje *swap* particije je swapon i ima sljedeći oblik:

```
swapon filesys
```

gdje *filesys* predstavlja sustav datoteka koji želite koristiti kao *swap* prostor. Tijekom pokretanja sustava Linux poziva naredbu swapon –a koja služi za učitavanje svih *swap* particija iz popisa u datoteci /etc/fstab.

BILJEŠKA Ne zaboravite u datoteku /etc/fstab unijeti zapis sa swap particiju ili datoteku koje ste stvorili kako bi im Linux, tijekom podizanja sustava, automatski pristupio.

## Stvaranje swap datoteke

Swap datoteke mogu biti korisne ako trebate proširiti swap prostor, a ne možete odvojiti prostor na disku za stvaranje swap particije. Podešavanje swap datoteke je gotovo identično stvaranju swap particije. Glavna razlika je u tome što morate najprije stvoriti datoteku, a tek potom pokrenuti naredbe mkswap i swapon.

Da biste stvorili *swap* datoteku, upotrijebite naredbu *dd* koja se inače koristi za kopiranje velike količine podataka. Kompletan opis ove naredbe potražite na stranici man za naredbu dd. Ono što trebate znati prije nego stvorite datoteku su ime *swap* datoteke i njezina veličina (u blokovima). U linuxu blok sadrži 1024 bajta. Da biste stvorili *swap* datoteku veličine 10 MB, koja će imati ime /swap, trebate upisati:

# dd if=/dev/zero of=/swap bs=1024 count=10240

Parametar of=/swap određuje ime datoteke, a parametar count=10240 njezinu veličinu (10240 blokova ili 10 MB). Nakon toga pokrenite naredbu mkswap da biste datoteku pretvorili u *swap* prostor:

# mkswap /swap 10240

Upamtite da morate naznačiti veličinu datoteke. Prije nego pokrenete naredbu swapon, trebate provjeriti je li kompletna datoteka zapisana na disku (pokrenite naredbu /etc/sync).

Sada možete aktivirati *swap* datoteku. Kao i kod *swap* particije, pokrenite naredbu swapon i datoteka će postati aktivna:

#swapon /swap

Ako želite obrisati *swap* datoteku, onda ona ne smije biti aktivna. Pomoću naredbe swapoff, datoteka se deaktivira

#swapoff /swap

Nakon toga možete obrisati swap datoteku.

# Odavde...

U ovom poglavlju ste upoznali razne aspekte sustava datoteka koji se koriste u Linuxu. Saznali ste kako možete načiniti osnovnu strukturu direktorija te graditi i razgrađivati sustave datoteke. Upoznali ste način pristupa udaljenim sustavima datoteka pomoću NFS-a i tehniku stvaranja sustava datoteka i njihovu pripremu za uporabu. Osim toga, u ovom poglavlju ste naučili kako se stvaraju *swap* particije i *swap* datoteke.

O administratoru sustava možete saznati nešto više u idućim poglavljima:

- 7. poglavlje, "Smisao administratora sustava" uvodi vas u zadaće koje ima administrator sustava.
- 10. poglavlje, "Upravljanje korisničkim računima" opisuje kako se podešava i upravlja korisničkim računima.
- 11. poglavlje, "Sigurnosna pohrana podataka" objašnjava vam kako treba planirati i provoditi sigurnosnu pohranu podataka.

# POGLAVLJE 15

# **Uporaba Sambe**

(Steve Burnett)

# U ovom poglavlju

Instaliranje Sambe 292		
Podešavanje Sambe na Linuxu 293		
Pokretanje Samba poslužitelja	300	
Uporaba programa smbclient	300	

U ovom poglavlju ćete naći informacije koje su vam potrebne za instaliranje, podešavanje i uporabu paketa Samba (SMB je kratica od Session Message Block). Pomoću sambe ćete moći učiniti sljedeće:

- Dijeliti Linux sustave datoteka s operativnim sustavima Windows 95, 98 ili NT.
- Dijeliti sustave datoteka iz operativnih sustava Windows 95, 98 ili NT s Linuxom.
- Dijeliti pisač koji je spojen na Linux računalo s operativnim sustavima Windows 95, 98 ili NT.
- Dijeliti pisač koji radi u okružju Windowsa 95, 98 ili NT s operativnim sustavom Linux.

Samba je protokol koji koriste Microsoftovi operativni sustavi za dijeljenje datoteka i pisača. Tvrtke Microsoft i Intel su 1987. godine razvili protokol SMB, a nakon toga je Andrew Tridgell sustav prilagodio raznim UNIX i Linux računalima.

Microsoft trenutno nudi drugi standard za dijeljenje resursa - Common Internet File BILJEŠKA System (CIFS). No, taj standard još treba dosta doraditi, uključujući tu i mogućnost dijeljenja resursa s Linux računalima.

Paket samba se sastoji od nekoliko komponenti. Demon smbd sadrži servise za datoteke i ispis koje koriste SMB klijent, kao što su Windows for Workgroups, Windows NT ili LanManager. U datoteci smb.conf su opisana podešenja koja se koriste za taj demon. Demon nmbd služi za rad s NetBIOS imenima i za pretraživanje. Također se može upotrijebiti za upite drugim sličnim programima.

Program smbclient predstavlja program-klijent za slanje datoteka (nalik FTP-u). Koristan je za pristup dijeljenim resursima na drugim poslužiteljima, kao što su Windows računala, a može se upotrijebiti da omogući računalu koje radi pod UNIX-om ispis na pisač povezan s osobnim računalom koje radi u okružju Windowsa 98.

Uslužni program testparm omogućuje vam testiranje datoteke smb.conf, a program smbstatus daje popis trenutnih korisnika smbd poslužitelja.

# Instaliranje Sambe

Sambu možete instalirati tijekom instalacije operativnog sustava ili naknadno pomoću RPM-a. Ako želite instalirati paket Samba, onda najprije kopirajte najnoviju inačicu s Red Hat Web lokacije (http://www.redhat.com). Potom možete instalirati paket (trenutno najnovija inačica je samba-1.9.18p5-1.i386.rpm) koristeći sljedeću naredbu:

```
rpm -ivh samba 1.9.18p5-1.i386.rpm
```

Vidi "Instalacija paketa pomoću RPM-a", 150. stranica

Paket bi trebao sadržavati sve datoteke koje su potrebne za pokretanje Sambe, uključujući i dva primarna programa smbd i nmbd. Ako koristite neku drukčiju distribuciju Linuxa, možda ćete morati ponovno kompajlirati neke programe.

# Podešavanje Sambe na Linuxu

Glavna datoteka s podešenjima se zove smb.conf i smještena je u direktoriju /etc. U ispisu 15.1 vidjet ćete ispis datoteke koja je sastavni dio distribucije Red Hat 5.1.

```
BILJEŠKA Komentari su označeni znakom točka-zarez (;) na samom početku retka. Samba poslužitelj zanemaruje komentare tijekom izvođenja datoteke.
```

```
Ispis 15.1 Primjer datoteke smb.conf koja se koristi za podešavanje Sambe
```

```
; The global setting for a RedHat default install
; smbd re-reads this file regularly, but if in doubt stop and restart it:
; /etc/rc.d/init.d/smb stop
; /etc/rc.d/init.d/smb start
[global]
; workgroup = NT-Domain-Name or Workgroup-Name, eg: REDHAT4
  workgroup = WORKGROUP
; comment is the equivalent of the NT Description field
  comment = RedHat Samba Server
; volume = used to emulate a CDRom label (can be set on a per share basis)
  volume = RedHat4
; printing = BSD or SYSV or AIX, etc.
  printing = bsd
  printcap name = /etc/printcap
  load printers = yes
; Uncomment this if you want a guest account
; guest account = pcguest
  log file = /var/log/samba-log.%m
; Put a capping on the size of the log files (in Kb)
  max log size = 50
; Options for handling file name case sensitivity and / or preservation
; Case Sensitivity breaks many WfW and Win95 apps
  case sensitive = yes
   short preserve case = yes
   preserve case = yes
; Security and file integrity related options
  lock directory = /var/lock/samba
  locking = yes
  strict locking = yes
  fake oplocks = yes
  share modes = yes
; Security modes: USER uses Unix username/passwd, SHARE uses WfW type passwords
        SERVER uses a Windows NT Server to provide authentication services
;
```

nastavlja se

```
Ispis 15.1 Nastavak
```

```
security = user
; Use password server option only with security = server
   password server = <NT-Server-Name>
; Configuration Options*****Watch location in smb.conf for side-effects*****
; Where %m is any SMBName (machine name, or computer name) for which a custom
; configuration is desired
   include = /etc/smb.conf.%m
; Performance Related Options
; Before setting socket options read the smb.conf man page!!
  socket options = TCP NODELAY
; Socket Address is used to specify which socket Samba
; will listen on (good for aliased systems)
   socket address = aaa.bbb.ccc.ddd
; Use keep alive only if really needed!!!!
  keep alive = 60
; Domain Control Options
; OS Level gives Samba the power to rule the roost. Windows NT = 32
       Any value < 32 means NT wins as Master Browser, > 32 Samba gets it
   os level = 33
; specifies Samba to be the Domain Master Browser
   domain master = ves
; Use with care only if you have an NT server on your network that has been
; configured at install time to be a primary domain controller.
   domain controller = <NT-Domain-Controller-SMBName>
; Domain logon control can be a good thing! See [netlogon] share section below!
   domain logons = yes
; run a specific logon batch file per workstation (machine)
   logon script = %m.bat
; run a specific logon batch file per username
   logon script = %u.bat
 Windows Internet Name Serving Support Section
 WINS Support - Tells the NMBD component of Samba to enable its WINS Server
       the default is NO.
   wins support = yes
 WINS Server - Tells the NMBD components of Samba to be a WINS Client
       Note: Samba can be either a WINS Server, or a WINS Client, but NOT both
   wins server = w.x.y.z
 WINS Proxy - Tells Samba to answer name resolution queries on behalf of a non
       WINS Client capable client, for this to work there must be at least one
       WINS Server on the network. The default is NO.
   wins proxy = yes
[homes]
  comment = Home Directories
  browseable = no
  read only = no
  preserve case = yes
  short preserve case = ves
  create mode = 0750
```

```
; Un-comment the following and create the netlogon directory for Domain Logons
; [netlogon]
   comment = Samba Network Logon Service
   path = /home/netlogon
; Case sensitivity breaks logon script processing!!!
   case sensitive = no
   guest ok = yes
   locking = no
;
   read only = yes
.
   browseable = yes ; say NO if you want to hide the NETLOGON share
   admin users = @wheel
; NOTE: There is NO need to specifically define each individual printer
[printers]
  comment = All Printers
  path = /var/spool/samba
  browseable = no
  printable = yes
; Set public = yes to allow user 'guest account' to print
  public = no
  writable = no
  create mode = 0700
;[tmp]
   comment = Temporary file space
   path = /tmp
;
   read only = no
   public = yes
; A publicly accessible directory, but read only, except for people in
; the staff group
;[public]
   comment = Public Stuff
   path = /home/samba
:
   public = yes
   writable = yes
;
   printable = no
;
   write list = @users
:
; Other examples.
; A private printer, usable only by fred. Spool data will be placed in fred's
; home directory. Note that fred must have write access to the spool directory,
; wherever it is.
;[fredsprn]
   comment = Fred's Printer
   valid users = fred
   path = /homes/fred
;
   printer = freds_printer
;
   public = no
   writable = no
   printable = yes
; A private directory, usable only by fred. Note that fred requires write
; access to the directory.
```

nastavlja se

#### Ispis 15.1 Nastavak

```
;[fredsdir]
   comment = Fred's Service
   path = /usr/somewhere/private
   valid users = fred
   public = no
  writable = yes
   printable = no
; a service which has a different directory for each machine that connects
; this allows you to tailor configurations to incoming machines. You could
; also use the %u option to tailor it by user name.
; The %m gets replaced with the machine name that is connecting.
;[pchome]
; comment = PC Directories
; path = /usr/pc/%m
; public = no
; writeable = yes
; A publicly accessible directory, read/write to all users. Note that all files
; created in the directory by users will be owned by the default user, so
; any user with access can delete any other user's files. Obviously this
; directory must be writable by the default user. Another user could of course
; be specified, in which case all files would be owned by that user instead.
;[public]
   path = /usr/somewhere/else/public
   public = yes
  only guest = yes
  writable = yes
   printable = no
; The following two entries demonstrate how to share a directory so that two
; users can place files there that will be owned by the specific users. In this
; setup, the directory should be writable by both users and should have the
; sticky bit set on it to prevent abuse. Obviously this could be extended to
; as many users as required.
;[myshare]
   comment = Mary's and Fred's stuff
   path = /usr/somewhere/shared
  valid users = mary fred
  public = no
  writable = yes
:
   printable = no
   create mask = 0765
```

Datoteka smb.conf se sastoji od nekoliko odlomaka. Svaki odlomak započinje imenom odlomka u uglatim zagradama, primjerice odlomak [global]. Unutar svakog odlomka parametri se određuju pomoću parova *ime=vrijednost*, primjerice commnet=RedHat Samba Server.

Datoteka smb.conf sadrži tri posebna odlomka [global], [homes], [printers], te jedan ili više korisničkih odlomaka.

# Odlomak [global]

Odlomak [global] upravlja parametrima koji se koriste za kompletan smb poslužitelj, a sadrži i osnovna podešenja (default) vrijednosti koje se koriste u drugim odlomcima.

```
[global]
; workgroup = NT-Domain-Name or Workgroup-Name, eg: REDHAT4
   workgroup = WORKGROUP
; comment is the equivalent of the NT Description field
   comment = RedHat Samba Server
; volume=used to emulate, a CDRom label (can be set on, a per share basis)
   volume = RedHat4
```

Prvi redak odlomka ŠglobalĆ (vidi Ispis 15.1) definira radnu grupu kojoj vaše računalo pripada. Zatim je određen komentar za sustav i oznaka diska.

```
; printing=BSD or SYSV or AIX, etc.
    printing = bsd
    printing name = /etc/printcap
    load printers = Yes
```

U idućem retku je opisan sustav za ispis koji se koristi na vašem poslužitelju, a u sljedećem retku je naznačeno mjesto na kojem se nalazi datoteka za podešenje ispisa.

#### Vidi "Datoteka etc/printcap", 411. stranica

U idućem retku se nalazi naredba koja paketu Samba stavlja na raspolaganje sve mrežne pisače koji su definirani u datoteci printcap.

```
; Uncomment this if you want, a guest account
; guest account = pcguest
    log file = /var/log/samba-log.%m
; Put, a capping on the size of the log file (in Kb)
    max log size = 50
```

Naredni redak daje korisničko ime anonimnom računu na vašem poslužitelju. Taj račun se koristi za označavanje korisnika Sambinih servisa koji su se anonimno prijavili na mrežu.

Parametar log file određuje mjesto datoteke log za svakog klijenta koji pristupi poslužitelju samba. Oznaka %m upućuje program da svakom korisniku otvori posebnu datoteku log, a max log size određuje najveću dopuštenu veličinu datoteka log.

# Odlomak [homes]

Odlomak [homes] omogućuje mrežnim klijentima povezivanje s korisničkim početnim direktorijom na vašem poslužitelju, a da pri tome nema izravnog zapisa u datoteci smb.conf. Uputite li zahtjev za određenom uslugom na poslužitelj samba, on će najprije pregledati datoteku smb.conf i potražiti poseban odlomak koji odgovara traženoj usluzi. Ako ne naiđe na odgovarajući zapis u datoteci, program Samba provjerava odlomak [homes]. Ako postoji odlomak [homes], Samba traži datoteku s lozinkama, a u njoj početni direktorij onog korisnika koji je uputio zahtjev. Nakon toga će taj direktorij

postati dijeljeni direktorij na mreži.

```
[homes]
  comment = Home Directories
  browseable = no
  read only = no
  preserve case = yes
  short preserve case = yes
  create mode = 0750
```

Parametar comment se pojavljuje na ekranima korisnika kako bi znali koji su dijeljeni direktoriji. Parametar browseable određuje način kako Samba prikazuje određene direktorije u mrežnom popisu, a parametar read only određuje može li korisnik stvarati i mijenjati datoteke u svom početnom direktoriju kada se taj direktorij dijeli na mreži. Parametri preserve case i short preserve case upućuju poslužitelj da spriječi zapisivanje velikim slovima bilo kakvih informacija. Ovo je važno stoga što imena datoteka u Windowsima ne ovise o velikim i malim slovima, ali se u Linuxu velika i mala slova međusobno razlikuju. U zadnjem zapisu se nalazi parametar koji podešava dozvole

datoteka koje nastaju u dijeljenom direktoriju.

Vidi "Dozvole pristupa datotekama", 310. stranica.

## Odlomak [printers]

Odlomak [printers] određuje način ispisa u slučaju da u datoteci smb.conf nema posebnih zapisa koji se odnose na način ispisa. Kao i kod odlomka [homes], ako nije nađen poseban zapis koji određuje način ispisa na pisač, Samba koristi odlomak [printers] kako bi korisnicima omogućila pristup pisačima koji su definirani u datoteci /etc/printcap.

```
[printers]
  comment = All Printers
  path = /var/spool/samba
  browseable = no
  printable = yes
; Set public = yes to allow user 'guest account' to print
  public = no
  writable = no
  create mode = 0700
```

Parametri commnet, browsable i create mode su objašnjeni u dijelu koji govori o odlomku [homes]. Parametar path određuje mjesto na kojem se nalazi datoteka *spool* koja se koristi kod ispisa putem Sambe.

Vidi "Odabir pisača za rad s Linuxom", 406. stranica

Parametar printable možete podesiti da sadrži vrijednost yes, i u tom slučaju se zadani pisač može koristiti za ispis. Parametar public dodjeljuje mogućnost ispisa korisniku koji se prijavio kao gost.

## Dijeljenje direktorija

Nakon što ste podesili poslužitelj Samba, možete stvoriti posebne dijeljene direktorije kojima mogu pristupiti određene grupe korisnika ili javne direktorije kojima mogu svi pristupati. Primjera radi pretpostavimo da želite načiniti direktorij kojem dozvoljavate

pristup samo određenom korisniku. Trebate stvoriti novi odlomak i unijeti potrebne podatke. Obično ćete trebati zadati korisnika, stazu korisničkog direktorija i još neka podešenja:

```
[jackdir]
  comment = Jack' remote source code directory
  path = /usr/local/src
  valid users = tackett
  browseable = yes
  public = no
  writable = yes
  create mode = 0700
```

Na ovaj ćete način stvoriti dijeljeni direktorij pod imenom jackdir. Staza do direktorija na lokalnom poslužitelju je /usr/local/src. S obzirom da parametar browseable ima vrijednost yes, direktorij jackdir će se pojaviti u mrežnom popisu direktorija. No s obzirom da je parametar public podešen na vrijednost no, moći će mu pristupiti samo korisnik s imenom tackett. Pristup drugim korisnicima možete omogućiti tako da ih navedete u parametru valid users.

## Provjera datoteke smb.conf

Nakon što ste stvorili datoteku s podešenjima, trebate je provjeriti pomoću programa testparm. Radi se vrlo jednostavnom programu za provjeru datoteke s podešenjima /etc/smb.conf. Ako program ne naiđe na probleme tijekom provjere, možete bez brige koristiti datoteku smb.conf.

#### **UPOZORENJE**

Provjera datoteke pomoću programa testparm nije garancija da će programi koji su navedni u datoteci raditi bez greške i biti dostupni korisnicima.

Evo kako izgleda naredba za pokretanje programa testparm:

```
testparm [configfile [hostname hostip]]
```

gdje argument *configfile* upućuje na mjesto na kojem se nalazi datoteka smb.conf (ako se ona ne nalazi na zadanom mjestu, tj. u direktoriju /etc). Proizvoljni parametar *hostname hostip* upućuje program testparm da provjeri ima li *host* mogućnost pristupa servisima koji se spominju u datoteci smb.conf.

U idućem primjeru je prikazan izlaz programa testparm. Ako je u datoteci smb.conf nađena greška, program testparm će izvijestiti o tome posebnom porukom o grešci.

```
# testparm
Load smb config files from /etc/smb.conf
Processingsection "[homes]"
Processingsection "[printers]"
Loaded services file OK.
Press enter to see, a dump of your service definitions
```

Nakon što pritisnete tipku <Enter>, program će otvoriti pojedini odlomak iz datoteke smb.conf.

# Pokretanje Samba poslužitelja

Poslužitelj Samba sadrži dva programa-*demona*: smbd i nmbd. Demon smbd omogućuje dijeljenje datoteka i pisača, a nmbd nudi podršku za NetBIOS.

Poslužitelj Samba možete pokrenuti ili pomoću skripti init koje su detaljnije opisane u 9. poglavlju ili kao sustavsku uslugu pomoću naredbe inetd.

```
Vidi "Pokretanje sustava", 206. stranica.
```

S obzirom da distribucije Red hat i Caldera pokreću paket Samba pomoću skripti init, možete upotrijebiti sljedeću naredbu za pokretanje/zaustavljanje poslužitelja Samba:

```
/etc/rc.d/init.d/samba start|stop
```

# Uporaba programa smbclient

Program smbclient omogućuje korisnicima Linuxa pristup drugim, obično Windows, računalima. Razlog tome je što, ako želite pristupiti datotekama na drugim računalima, možete upotrijebiti razne metode kao što su FTP, NFS i r- naredbe (primjerice rpc).

Vidi "Uporaba r- naredbi", 589. stranica.

Program smbclient koristi FTP sučelje koje vam omogućuje prijenos datoteka putem mreže na drugo računalo koje koristi Sambu. Nažalost, za razliku od NFS-a, program smbclient ne nudi mogućnost građenja dijeljenog direktorija kao da se radi o lokalnom direktoriju.

Program smbclient nudi određene opcije o kojima nešto više možete saznati na stranici man koja se odnosi na taj program. Pomoću sljedeće naredbe dobit ćete popis svih dijeljenih direktorija na računalu win.netwharf.com:

smbclient -L -I win.netwharf.com

Parametar –L predstavlja zahtjev za traženje popisa, a parametar –I upućuje program smbclient da ime računala tretira kao unos DNS, a ne kao unos NetBIOS.

Da biste prenijeli datoteku s jednog na drugo mjesto, morate se najprije povezati sa poslužiteljem Samba pomoću sljedeće naredbe:

smbclient '\\WORKGROUP\PUBLIC' -I win.netwharf.com -U tackett

Parametar '\\WORKGROUP\PUBLIC' određuje servis na udaljenom računalu, odnosno određeni sustav datoteka ili pisač. Opcija –U omogućuje vam upis korisničkog imena pod kojim želite ostvariti vezu. Program samba će zatražiti da upišete lozinku (ako je odabrani račun zahtijeva) i zatim vas postaviti u prompt

smb: ∖

gdje oznaka \ predstavlja radni direktorij.

Sada možete pokrenuti neku od naredbi prikazanih u tablici 15.1.

	Nareque programa smochen	
Naredba	Parmetri	Opis
? or help	[command]	Nudi pomoć za pojedinu naredbu, ili općenitu pomoć (ako se ne naznači naredba).
!	[shell command]	Izvršava se određena naredba iz ljuske ili se korisnik prebacuje u određenu ljusku.
cd	[directory]	Premještanje korisnika na odabrani direktorij na računalu-poslužitelju (ne na lokalnom računalu). Ako se ne odabere direktorij, program smbclient će prijaviti radni direktorij.
lcd	[directory]	Premještanje korisnika na odabrani direktorij na lokalnom računalu. Ako se ne odabere direktorij, program smbclient će prijaviti radni direktorij na lokalnom računalu.
Del	[files]	Brisanje odabranih datotekama poslužitelju (naravno, uz korisnikovu potvrdu). Datoteke mogu sadržavati i specijalne znakove (wildcard characters).
dirorls	[files]	Daje popis naznačenih datoteka.
exit or quit	nijedan	Izlaza iz programa smbclient.
get	[remote file] [local name]	Poziva odabranu datoteku na udaljenom računalu i sprema je na lokalni poslužitelj. Ako ste unijeli ime lokalno ime, kopija datoteke će biti spremljena pod tim imenom (a ne pod originalnim imenom).
mget	[files]	Kopiranje na lokalno računalo svih navedenih datoteka (uključujući i one s posebnim znakovima).
mdilimkdir	[directory]	Stvaranje direktorija na udaljenom računalu.
rdili rmdir	[directory]	Uklanjanje direktorija na udaljenom računalu.
put	[file]	Kopiranje određene datoteke s lokalnog računala na poslužitelj.
mput	[files]	Kopiranje određenih datoteka s lokalnog računala na poslužitelj.
print	[file]	Ispis odabrane datoteke na udaljenom računalu.
queue	nijedan	Prikaz svi poslova na udaljenom pisaču.

#### - --

# Odavde...

Možete pročitati sljedeća poglavlja i saznati nešto više o operativnom sustavu Linux što vam može pomoći u uporabi Sambe:

- 9. poglavlje, "Pokretanje i isključivanje sustava" sadrži postupak i datoteke koje se koriste za pokretanje ili isključivanje raznih servisa u Linuxu.
- 16. poglavlje, "Sustav datoteka i direktorija" sadrži osnovne podatke o datotekama i direktorijima. Kad koristite emacs ili neki drugi program za uređivanje teksta potrebno je dobro razumjeti sustav datoteka. Bez obzira da li mijenjate ili stvarate datoteke, možete im dati željeno ime i pohraniti ih u odabrani direktorij.
- 20. poglavlje, "Ispis na pisač" sadrži informacije o ispisu datoteka s tekstom. Ispis datoteka pod Linuxom zna biti nezgodno. U ovom poglavlju ćete saznati kako se sustav priprema za ispis datoteka na pisač.
- Datoteka SMB-HOWTO sadrži primjere datoteka s podešenjima i skripti za ispis na pisač uz pomoć Sambe. U Dodatku A saznajte kako možete pristupiti raznim datotekama HOWTO.

# POGLAVLJE 16

# Sustav datoteka i direktorija

(Jack Tackett)

# U ovom poglavlju

Imena datoteka i staza **304** Standardni direktoriji u Linuxu **313**  Izraz sustav datoteka u Linuxu ima dvostruko značenje: u jednom slučaju se radi o diskovima na kojima su pohranjeni podaci, a u drugom slučaju se radi o logičkom sustavu datoteka kojeg korisnici vide na ekranu i kojim mogu upravljati. U ovom poglavlju ćemo izraz sustav datoteka koristiti za logički sustav kojim korisnici upravljaju. Ako su vam poznati operativni sustavi za osobna računala, kao što su MS-DOS ili OS/2, iduće teme će vam se učiniti poznatima, jer je struktura datoteka u MS-DOS-u načinjena prema onoj koja se koristi u UNIX-u (odnosno Linuxu).

Svaki fizički i logički element u Linuxu je predstavljen datotekom koja je dio zajedničkog sustava datoteka. Pod fizičke elemente se ubrajaju diskovi, pisači i terminali, a logički elementi su direktoriji i obične datoteke u kojima su pohranjeni dokumenti i programi.

# Imena datoteka i staza

U Linuxu, kao i drugim operativnim sustavima kao što je MS-DOS, treba razlikovati ime datoteke i ime staze. Ime datoteke se sastoji od jednostavnog niza slova, brojeva i interpunkcijskih oznaka. Imena datoteka ne mogu sadržavati razmake ili znakove koji predstavljaju odjelna polja. Primjerice, ispravno ime je johns.letter, a neispravno johns letter.

Imena datoteka ne bi smjela sadržavati posebne znakove kao što su:

!@#\$% ^&\*()[ ]{ }'"\/|; <>

jer oni imaju posebno značenje za ljusku.

Osim toga, u imenima datoteka ne mogu se koristiti kose crte (/) jer se taj znak koristiti u imenima staza (o kojima će biti govora u nastavku ovog poglavlja).

BILJEŠKA U stvari, u imenu datoteka možete koristiti sve znakove ako ime datoteke upišete unutar navodnika:

"! johns.letter"

No, takvim datotekama se teško pristupa iz većine programa i nezgodno ih je prenositi na druge sustave.

Većina ranijih inačica UNIX-a, na osnovu kojih je nastao Linux, ograničava duljinu imena datoteka na 14 znakova, no Linux ipak omogućava ime datoteke u duljini od 256 znakova. Neke novije inačice UNIX-a, kao što je inačica Berkeley (BSD) omogućuje duljinu imena od 64 znaka, ali je od toga značajno samo prvih 14 znakova. S obzirom da je prijenos datoteka s jednog na drugo računalo vrlo bitan, ograničite se na imena datoteke koja neće biti dulja od 14 znakova. Na taj ćete način programe moći svugdje prenositi, a i olakšano vam je pisanje skripti s naredbama.

Ime staze može sadržavati proizvoljan broj znakova. U Linuxu se datoteke ne nalaze u vakuumu nego su pohranjene u direktorijima. Najveći direktorij u LInuxu je direktorij root koji se označava kosom crtom (/). Ako se datoteka pod imenom fred nalazi u direktoriju *root*, njezina staza će biti /fred. Kada dodajete korisnika pomoću naredbe adduser, korisniku će biti dodijeljen i početni direktorij. Prema dogovoru, ovaj početni direktorij se obično nalazi u direktoriju /home. Stoga, ako je korisniku Fredu dodijeljen

direktorij pod imenom /home/fred, sve datoteke koje Fred bude stvorio će se naći u tom direktoriju. Apsolutna staza za jednu od Fredovih datoteka može biti /home/fred/freds.file. Ime apsolutne staze određuje točno mjesto u sustavu datoteka na koje je spremljena pojedina datoteka.

Druga vrsta imena staze se naziva *relativno ime staze* i služi za usmjeravanje naredbi na mjesto na kojem se nalazi određena datoteka u odnosu na tekući direktorij (direktorij u kojem se trenutno nalazite). Ako se korisnik Fred nalazi u svom početnom direktoriju, onda je ime datoteke freds.file i relativno i apsolutno ime. Da biste saznali u kojem se direktoriju trenutno nalazite, pokrenite naredbu pwd. Drugi je način da provjerite vrijednost varijable \$PWD koristeći naredbu echo \$PWD. Na ekranu će se pojaviti naziv direktorija u kojem se trenutno nalazite.

Položaj datoteke u sustavu datoteka možete odrediti pomoću dva pseudonima koje ćete naći u svim direktorijima. Jednostruka točka (.) označava tekući direktorij, a dvije točke (..) naddirektorij tekućeg direktorija. Ista pravila vrijede i u MS-DOS-u i u operativnom sustavu OS/2.

Ako se Fred trenutno nalazi u direktoriju /home/fred, on da može upisati ../../fred i usmjeriti naredbu na direktorij /fred. U tom relativnom imenu staze prve dvije točke označavaju nadirektorij direktorija /home (direktorij *root*), a druge dvije točke predstavljaju direktorij /home (naddirektorij direktorija /home/fred).

Pseudonim za tekući direktorij, odnosno točka (.), dolazi do izražaja kod premještanja datoteka. Ako korisnik Fred želi premjestiti direktorij /fred u tekući direktorij, onda može pokrenuti naredbu move s apsolutnim imenom staze:

mv /fred fred

Drugi je način da umjesto imena direktorija u kojem se trenutno nalazi upotrijebi točku:

mv /fred .

Većina Linux naredbi koristi imena staza. U mnogim slučajevima koristi se ime staze za datoteku koja se nalazi u tekućem direktoriju (radi se o osnovnom podešenju mnogih naredbi). Ako se korisnik Fred nalazi u svom početnom direktoriju (/home/fred), onda sve tri naredbe imaju isto značenje:

```
naredba freds.letter
naredba /home/fred/freds.letter
naredba ./freds.letter
```

**BILJEŠKA** lako postoji razlika između imena datoteka i imena staza, direktoriji su također datoteke. Kada direktorijima dajete imena, morate slijediti ista pravila kao da se radi o običnim datotekama.

Upamtite da, za razliku od operativnih sustava za osobna računala, Linux ne koristi slova kao imena pogona, već samo imena staza. U Linuxu se imena diskova koriste jedino u radu sa sustavskim MS-DOS datotekama, prilikom pokretanja m- naredbi (kao što je naredba mcopy).

Vidi "Sustavi datoteka", 266. stranica.

## Vrste datoteka

Operativni sustav Linux sve podatke sprema u četiri vrste datoteka: obične datoteke, direktorije, veze i posebne datoteke. Postoji nekoliko vrsta običnih datoteka, veza i posebnih datoteka, te velik broj standardnih direktorija. U idućim odlomcima su opisane osnovne vrste datoteka.

Da biste odredili vrstu datoteke, pokrenite naredbu file koja može prepoznati vrstu datoteke koja može biti izvršna, tekstualna, baza podataka i slično. Mnoge UNIX naredbe su u osnovi samo skripte naredbi ili programi slični *batch datotekama* u MS-DOS-u. Naredba find može odrediti je li UNIX naredba binarna izvršna datoteka ili skripta.

Ovaj je naredba korisna i za određivanje sadrži li pojedina datoteka tekst, te može li se pregledati ili uređivati. Slijedi oblik naredbe file:

file [-vczL] [-f namefile] [-m magicfile] filelist

U tablici 16.1 su objašnjeni argumenti naredbe find.

Tablica 16.1 Arg	gumenti naredbe find
Argument	Opis
- C	Ispis datoteke magic (/usr/lib/magic) koja predstavlja broj u prvom dijelu binarne datoteka koja služi za identificiranje vrste datoteke. Ova se opcija obično koristi s argumentom –m za otkrivanje grešaka u novoj "magičnoj" datoteci prije nego je instalirate.
- Z	Određuje vrstu datoteke unutar komprimirane datoteke.
- L	Ukazuje na to da slijedi simbolička veza.
-f namefile	Upućuje program file na to da se popis datoteka nalazi u tekst datoteci namefile. Ovaj je argument koristan ako treba identificirati više datoteka.
-m <i>magicfile</i>	Određuje alternativnu datoteku s "magičnim" brojevima koji se koriste za određivanje vrsti datoteka. Osnovno je podešena datoteka /usr/lib/magic.
filelist	Popis datoteka čiju vrstu želite saznati (datoteke odvojite razmakom).

## **Obične datoteke**

*Obične datoteke* su one datoteke s kojima ćete se najviše baviti. One mogu sadržavati dokumente, osnovni kod u C jeziku, skripte naredbi , odnosno programa koje izvršavaju Linux ljuske, binarne izvršne programe i razne vrste podataka. No, datoteka, kakva god bila, uvijek je datoteka. Linux razlikuje samo one datoteke koje su označene kao izvršne. Takve datoteke se mogu izravno izvršavati (uz pretpostavku da je u datoteci zadano obavljanje određene zadaće i da se datoteka nalazi u odgovarajućoj stazi). U osnovi je potrebno odrediti imena onih staza u kojima Linux traži izvršne datoteke

Vidi "Što su ljuske", 339. stranica

Izvršne datoteke su binarne datoteke, to jest datoteke koje izvođe kôd koji razumije računalo i skripte naredbi. U prethodnom odlomku ste upoznali naredbu file koja služi za traženje podataka u datoteci i na osnovu tih informacija nastoji odgonetnuti vrstu datoteke. Upišete li, primjerice, **file \*.** na ekranu možete vidjeti ovakav ispis:

```
INSTALL:
            symbolic link to /var/adm
ghostvw.txt: ascii text
linux: symbolic link to /usr/src/linux
mbox:
           mail text
mterm.txt: English text
seyon.txt:
            English text
xcalc.txt:
            English text
xclock.txt: English text
xeyes.txt: English text
xgrap.txt:
           English text
xlock.txt: English text
xspread.txt: English text
xtris.txt:
             empty
```

Sve datoteke u prvom stupcu predstavljaju obične datoteke koje sadrže razne vrste podataka i sve se datoteke nalaze u direktoriju iz kojeg je pokrenuta naredba file.

## Datoteke-direktoriji

*Direktoriji* su datoteke koje sadrže imena datoteka i poddirektorija te usmjerivače na te datoteke i poddirektorije. Datoteke-direktoriji su jedino mjesto na koje Linux sprema imena datoteka. Ako pomoću naredbe ls ispišete sadržaj nekog direktorija na ekran, vi ste ustvari ispisali sadržaj datoteke-direktorija.

Ako mijenjate ime datoteke koristeći naredbu mv, a pri tome se datoteka nalazi u tekućem direktoriju, vi ustvari mijenjate zapis u datoteci-direktoriju. Ako datoteku premještate s jednog u drugi direktorij, to znači da premještate njezin opis iz jedne datoteke-direktorija u drugu (naravno uz pretpostavku da se i novi direktorij nalazi na istom fizičkom disku ili particiji). Ako se novi direktorij nalazi na drugom disku, Linux će fizički kopirati sve podatke s jednog na drugi disk.

## Direktoriji i fizički diskovi

Svakoj datoteci u Linuxu je dodijeljen jedinstveni broj nazvan *inode*. Taj je broj spremljen u odgovarajuću tablicu koja je načinjena prilikom formatiranja diska. Svaki fizički disk ili particija ima vlastitu tablicu *inode* brojeva koja sadrži sve informacije o datoteci, uključujući adresu na kojoj se na disku nalaze podaci te vrstu datoteke. Datoteke mogu biti obične, direktoriji ili posebne.

Sustav datoteke u Linuxu direktoriju *root* dodjeljuje broj 1, što predstavlja adresu datoteke-direktorija *root* na disku. Direktorij *root* sadrži popis datoteka i direktorija, te njihove *inode* brojeve. Linux može pronaći bilo koju datoteku tako da pretražuje direktorije (počevši od direktorija *root*). Evo kako može izgledati sadržaj direktorija *root*:

```
etc
dev
```

```
home
.profile
```

Uočite da su u direktoriju prikazane datoteke . (točka) i .. (dvije točke). S obzirom da se radi o direktoriju *root*, to su ova dva direktorija potpuno jednaka. Sadržaja direktorija /home se donekle razlikuje od direktorija *root* i možda će ovako izgledati:

. .. fred

Uočite da je broj *inode* tekućeg direktorija (.) jednak broju direktorija /home koji ste našli u direktoriju *root*, a slično vrijedi i za nadirektorij (..) koji ima isti broj kao i direktorij *root*.

Želite li premjestiti datoteku u direktorij na drugom fizičkom disku, sustav će to otkriti prema broju *inode*. U tom slučaju datoteka se fizički premješta na novi disk i dodjeljuje joj se novi broj *inode* na tom disku, a zatim se briše originalna datoteka.

Kao i kod naredbe mv, ako datoteku brišete pomoću naredbe rm, nećete doći u doticaj sa samom datotekom. Umjesto toga će Linux označiti broj *inode* kao slobodan i ponuditi ga na ponovno korištenje. Briše se i zapis obrisane datoteke u direktoriju.

Vidi "Premještanje i promjena imena datoteka", 323. stranica

### Veze

*Obične veze* ustvari nisu datoteke već elementi u direktoriju koji su usmjereni na njih. Tablica brojeva *inode* sadrži podatak koliko jedna datoteka ima veza, i tek kada se obriše i posljednja veza, onda se taj broj može ponovno upotrijebiti.

Da biste pomoću naredbe ln stvorili vezu, trebate upotrijebiti ovaj oblik naredbe:

```
In [options] source destination
```

Evo primjera: da biste stvorili vezu između datoteke pod imenom mainfile.txt i datoteke tempfile.txt, trebate upisati ovakvu naredbu:

```
ln mainfile.txt tempfile.txt
```

Linux, kao i svi moderniji nasljednici UNIX-a, sadrži i drugu vrstu veza, to jest *simboličke veze*. Takva veza sadrži broj *inode* datoteke koja je i sama usmjerena na drugu datoteku u logičkom sustavu datoteka. Simbolička veza može biti usmjerena na drugu datoteku ili direktorij koji se nalazi na istom disku, ili na drugom disku, ili može biti usmjerena na datoteku ili direktorij na drugom računalu.

Glavna razlika između obične i simboličke veze je ta što kod običnih veza sve veze imaju isti status (to jest, sustav tretira svaku vezu kao da se radi o originalnoj datoteci) i stvarni podaci se ne brišu sve dok nije obrisana i posljednja veza te datoteke. Kod simboličkih veza, čim se obriše originalna datoteka, brišu se i sve simboličke veze s njom. Datoteke povezane simboličkom vezom nemaju isti status kao i originalna datoteka. Da biste stvorili simboličku vezu, upotrijebite naredbu ln s opcijom –s. Želite li stvoriti simboličku vezu iz datoteke named koja se nalazi u direktoriju /etc/rc.d/initd s datotekom S55named, morate pokrenuti ovu naredbu:

ln -s /etc/rc.d/initd/named /etc/rc.d/rc3.d/S55named

Unatoč tim razlikama između veza i datoteka, veze se smatraju datotekama i tako im se i pristupa.

Pomoću naredbe ls –l možete provjeriti je li datoteka u stvari veza. Ako se radi o vezi, na ekranu ćete ugledati ime lokalne datoteke i indikaciju povezane datoteke:

lrwxrwxrwx 1 root root 4 Oct 17 15.27 Info -> info/

Polje s dozvolama pristupa počinje slovom l što upućuje da se radi o datoteci-vezi.

## Posebne datoteke

Svaki fizički uređaj u Linuxu, uključujući diskove, terminale i pisače, na određeni je način predstavljen u sustavu datoteka. Većina, ako ne i svi uređaji se nalaze u direktoriju /dev. Primjerice, ako radite na konzoli sustava, onda se pripadni uređaj naziva /dev/console. Ako radite na standardnom terminalu, možda će ime vašeg uređaja biti /dev/tty01. Terminali, ili serijske linije se nazivaju *tty uređaji* (kratica za prvi UNIX terminal). Pokrenite naredbu tty i saznat ćete koji je naziv vašeg terminala. Na ekranu će se pojaviti ime uređaja na kojem radite.

Pisači i terminali se nazivaju *posebni uređaji koji koriste znakove*. Oni prihvaćaju i proizvode niz znakova. Diskovi, s druge strane, pohranjuju podatke u blokovima koji se nalaze na određenim cilindrima i sektorima diska. Ne možete pristupiti samo jednom znaku na disku, već trebate pročitati čitav blok. Obično se isto odnosi i na magnetske trake. Ovakve vrste uređaja se nazivaju *posebni uređaji koji koriste blokove*. Da bi vam još više zakomplicirali život, spomenimo da se diskovi i trake moraju ponašati kao uređaji koji koriste znakove, tako da svaki uređaj koji koristi blokove ima pripadni uređaj koji koristi znakove. Prijenos podataka iz jedne vrste uređaja u drugi obavlja operativni sustav Linux i to bez vašeg sudjelovanja u procesu.

Morate pokrenuti barem jednu vrstu posebnog uređaja: FIFO (spremnik). Spremnik FIFO izgleda kao obična datoteka, što ga više punite to on više raste. No, ako pročitate spremnik FIFO, njegova veličina će se smanjiti. Spremnik FIFO se uglavnom koristi u sustavskim procesima i omogućuje mnogim programima slanje određenih informacija. Ako datoteku ispisujete pomoću naredbe lp, onda ćete slanjem poruka u spremnik FIFO pokrenuti *demon* lpsched. *Demon* predstavlja sustavski proces koji se obavlja bez sudjelovanja korisnika.

Vrlo je koristan jedan uređaj, odnosno posebna datoteka. /dev/null. Sve što pošaljete na taj uređaj će biti zanemareno što vam može poslužiti ako ne želite vidjeti izlaz pojedine naredbe. Primjerice, ako ne želite vidjeti neki dijagnostički izvještaj, možete ga poslati na uređaj /dev/null i to pomoću sljedeće naredbe:

ls -la> /dev/null

## Dozvole pristupa datotekama

Dozvole pristupa datotekama predstavljaju u Linuxu nešto više od običnih prava koje korisnik ima nad datotekom ili direktorijem. Osim što dozvole služe za određivanje prava na čitanje, pisanje i izvršavanje datoteke, oni određuju i vrstu datoteke i način na koji se ona izvršava.

Na ekranu možete vidjeti dozvole pojedine datoteke ako pokrenete naredbu ls –l. Parametar –l određuje opširniji ispis podataka o datotekama. Upišete li ls -l, na ekranu će se pojaviti ispis koji je nalik ovome:

Drwx 2	sglines	doc	512	Jan	1	13:14	Mail
Drwx 5	sglines	doc	1024	Jan	17	08:22	News
-rw1	sglines	doc	1268	Dec	7	15:01	biblio
drwx 2	sglines	doc	512	Dec	15	21:28	bin
-rw1	sglines	doc	44787	Oct	20	06:59	books
-rw1	sglines	doc	23801	Dec	14	22:50	bots.msg
-rw-r 1	sglines	doc	105990	Dec	27	21:24	duckie.gif

Ovakav ispis prikazuje gotovo sve informacije o datotekama. Prvi stupac sadrži dozvole pristupa datotekama, a drugi stupac broj veza s datotekom (ili dodatnih blokova u

direktoriju). U trećem stupcu je naveden vlasnik datoteke (U Linuxu postoje tri načina vlasništva nad datotekama: vlasnik, grupa vlasnika i svi ostali. U nastavku ovog poglavlja ćete detaljnije upoznati pojam vlasništva nad datotekama). U četvrtom stupcu je navedena grupa datoteka kojima pojedine datoteke pripadaju. U petom stupcu se nalazi veličina datoteke u bajtima, a šesti stupac predstavlja datum i vrijeme nastanka datoteke.

U sedmom stupcu se nalazi ime same datoteke.

Prvi stupac u ispisu predstavlja dozvole pristupa datotekama i moguće ga je podijeliti u četiri segmenta:

- rwx rwx rwx

Prvi segment određuje vrstu datoteke. Obično se na tom mjestu nalazi crtica (-), a direktoriji se označavaju slovom d. U tablici 16.2 naći ćete vrijednosti koje se mogu naći u prvom segmentu dozvole za pristup datotekama.

Tablica 16.2 Vrijednosti koje određuju vrstu datoteke			
značenje			
obična datoteka			
posebna datoteka (blok)			
posebna datoteka (znak)			
direktorij			
simbolička veza			

U iduća tri segmenta dozvole za pristup datotekama nalaze se polja koja određuju dozvolu čitanja, pisanja i izvođenja datoteke. Prvi segment određuje prava vlasnika datoteke, drugi segment određuje prava grupe kojoj pripada vlasnik, a treći segment daje prava svim ostalim korisnicima sustava.
U ta tri segmenta je sadržano nekoliko atributa same datoteke tako da se može reći da oni sadržavaju vrlo bitne informacije o datoteci. Nažalost, značenje tih atributa je određeno inačicom Linuxa koju koristite i ovisi o tome je li određena datoteka izvršna ili ne.

**BILJEŠKA** Obično je vlasnik programa onaj koji ga koristi. Ako je u program uključen i identifikacijski broj korisnika, onda je vlasnik programa osoba koja je vlasnik datoteke, što znači da vlasnik programa dodjeljuje dozvole pristupa i korištenja programa. Ako ste vi jedan običan korisnik, a vlasnik programa je superkorisnik vašeg sustava, program će automatski imati dozvolu čitanja i pisanja u bilo koju datoteku unutar sustava, bez obzira na dozvole koje ste vi dodijelili. Isto pravilo vrijedi i za programe koji u sebi imaju uključen identifikacijski broj grupe.

U navedenim segmentima se može nalaziti argument koji služi za pohranu kopije programa u memoriju. Ako se program često koristi, taj će vam argument uštediti nešto vremena jer se program neće morati ponovno učitavati u memoriju.

Pomoću naredbe chmod možete mijenjati dozvole pristupa svakoj datoteci nad kojom imate dozvolu pisanja. Ta naredba se može koristiti u dva oblika: apsolutni oblik i relativni oblik. Kod apsolutnih dozvola imate mogućnost definirati dozvole koristeći oktalni sustav (brojeve od 0 do 7). Oktalni sustav se koristi stoga što je UNIX nastao na mini računalima DEC koja su koristila oktalni brojčani sustav. Tablica 16.3 sadrži dozvole pristupa datotekama u oktalnom brojčanom sustavu.

Tablica 16.3 Apso	olutne dozvole koje se zadaju pomoću naredbe chmod
Oktalna vrijednost	Dozvola pristupa
0001	Dozvola izvođenja je dodijeljena vlasniku
0002	Dozvola pisanja je dodijeljena vlasniku
0004	Dozvola čitanja je dodijeljena vlasniku
0010	Dozvola izvođenja je dodijeljena grupi
0020	Dozvola pisanja je dodijeljena grupi
0040	Dozvola čitanja je dodijeljena grupi
0100	Dozvola izvođenja je dodijeljena svim ostalim korisnicima
0200	Dozvola pisanja je dodijeljena svim ostalim korisnicima
0400	Dozvola čitanja je dodijeljena svim ostalim korisnicima
1000	Uključeno spremanje programa u memoriju
2000	Ako se radi o izvršnoj datoteci, onda je uključen identifikacijski broj grupe, u protivnom se datoteka zaključava
4000	Ako se radi o izvršnoj datoteci, onda je uključen identifikacijski broj korisnika

Korisnički identifikacijski broj i identifikaciji broj grupe određuju na koga se odnose dozvole korištenje, čitanja ili izvođenja datoteke. Administrator sustava kod stvaranja korisničkog računa dodjeljuje početne dozvole za pristup datotekama. Samo korisnici iz iste grupe mogu pristupiti određenim datotekama čiji je vlasnik grupa korisnika.

Želite li svima dati dozvolu čitanja i pisanja u određenu datoteku, morate zbrojiti sve potrebne dozvole. Evo jednog takvog primjera zbrajanja u oktalnom brojčanom sustavu:

0666	Dozvola čitanja i pisanja je dodijeljena svima
0400	Dozvola čitanja je dodijeljena svim ostalim korisnicima
0200	Dozvola pisanja je dodijeljena svim ostalim korisnicima
0040	Dozvola čitanja je dodijeljena grupi
0020	Dozvola pisanja je dodijeljena grupi
0004	Dozvola čitanja je dodijeljena vlasniku
0002	Dozvola pisanja je dodijeljena vlasniku

Nakon toga pokrenite naredbu chmod u ovom obliku:

chmod 666 file

Relativne dozvole imaju nešto drukčiji oblik i kod njihove dodjele morate znati sljedeće:

- kome dajete dozvolu
- Zelite li dodati, ukinuti i promijeniti dozvolu
- koje dozvole dajete

Ako, primjerice, upišete naredbu **chmod a=rwx file**, dodijelit ćete svim korisnicima dozvolu čitanja, pisanja i izvođenja. U tablici 16.4 nalaze se naredbe za dodjelu relativnih dozvola.

Tablica 16.4	Relativne dozvole koje se koriste s naredbom chmod		
Vrijednost	Opis		
(kome se dozvo	(kome se dozvola daje)		
a	Svim korisnicima (korisniku, njegovoj grupi i ostalima)		
g	Grupi vlasnika datoteke		
0	Svim ostalim korisnicima koji nisu u grupi vlasnika datoteke		
u	Samo vlasniku datoteka		
(operator)			
+	Dodavanje		
-	Uklanjanje		
=	Postavljanje apsolutnih vrijednosti		

Vrijednost	Opis
(dozvola)	
х	Dozvola izvođenja
r	Dozvola čitanja
W	Dozvola pisanja
S	Postavljanje korisničkog identifikacijskog broja
t	Zapisivanje programa u memoriju nakon izvođenja

Ako je datoteka označena kao da ima ugrađen identifikacijski broj korisnika, onda će ispis naredbe ls –l izgledati ovako:

- rws - - - 1 sglines 3136 jan 17 15:42 x

Ako ste dodali identifikacijski broj grupe, dozvola će izgledati ovako:

- rws - - S - - - 1 sglines 3136 jan 17 15:42 x

Ako želite da se program pohrani u memoriju, onda će dozvola ovako izgledati:

- rws - - S - - rws - - S - - T 1 sglines 3136 jan 17 15:42 x

Velika slova S i T određuju status korisničkog identifikacijskog broja (slovo S) i bita koji služi za zapis programa u memoriju (T).

## Standardni direktoriji u Linuxu

Na direktorije ste se već navikli. Odmah nakon prijave za rad na sustavu naći ćete se u svom početnom direktoriju. Varijabla PATH je podešena tako da sadrži druge direktorije u kojima se nalaze izvršni programi. Ti su direktoriji sastavni dio standardne strukture direktorija koja se koristi u Linuxu.

Postoji nekoliko klasičnih direktorija koji su se počeli koristiti u UNIX i nastavili u Linuxu. Oni su opisani u nastavku poglavlja.

### Klasični UNIX direktoriji

Prije pojave UNIX-a V Release 4 (primjerice kod UNIX-a V Release 3.2) većina inačica UNIX-a je koristila regularni sustav organiziranih direktorija, koji je ovako izgledao:

```
/
```

```
/etc
/lib
/tmp
/bin
/usr
/spool
/bin
```

```
/include
/tmp
/adm
/lib
```

Direktorij /etc sadrži većinu posebnih datoteka koje su nužne za pokretanje sustava. U njemu se nalaze datoteke kao što su passwd i inittab koje su neophodne za pravilan rad sustava.

U direktoriju /lib se nalazi biblioteka funkcija potrebnih C kompajleru. Čak i ako na svom računalu nemate C kompajler, ovaj je direktorij itekako bitan jer on sadrži sve dijeljene biblioteke koje programi pozivaju. Dijeljena biblioteka se učitava u memoriju samo kada je poziva određena naredba. Na taj se način izbjegava uporaba velikih programa. U protivnom, svaki program koji se sastoji od mnogo ponavljanja programskog koda zahtijevao bi puno prostora na disku (pohrana programa) i u memoriji (pokretanje programa).

Direktorij /tmp se koristi za prijevremenu pohranu podataka. Programi koji koriste ovaj direktorij sami na kraju rada brišu sve privremene datoteke. Ako ste koristili direktorij /tmp, obrišite sve datoteke u njemu prije nego se odjavite. S obzirom da sustav automatski briše sve datoteke iz ovog direktorija, u njemu ne smije ostati ništa što bi vam kasnije moglo zatrebati.

U direktoriju /bin se nalaze svi programi koji su potrebni za pokretanje sustava kao i većina naredbi koje Linux koristi. Uočite da izvršni program ne mora nužno biti binarni program. Nekoliko manjih programa u direktoriju /bin su ustvari skripte s naredbama.

U direktoriju /usr ćete naći sve ostalo. Varijabla PATH sadrži /bin:/usr/bin jer se u direktoriju /usr/bin nalaze sve Linux naredbe koje nisu smještene u direktoriju /bin. Ovo pravilo ima svoju povijest. U početku razvoja Linuxa tvrdi diskovi su bili vrlo veliki. Za pokretanje Linuxa su nužni direktoriji /etc/tmp i /bin. S obzirom da su na početku diskovi sadržavali samo te direktorije, svi ostale datoteke i direktoriji su se stvarali nakon što je Linux već bio instaliran. S obzirom da je Linux bio relativno mali operativni sustav, postavljanje poddirektorija u direktorij /usr nije imalo smisla. To je omogućavalo da operativni sustav Linux postoji sa samo dva direktorija: */root* i /usr.

Direktorij /usr/adm sadrži sve dijagnostičke informacije koje su potrebne administratoru sustava. Ovaj je direktorij je gotov prazan ako niste pokrenuli niti jedan od dijagnostičkih programa.

U direktoriju /include ćete naći sav osnovni kod kojeg koristi naredba #include u C programima. Za ovaj direktorij možete imati dozvolu samo za čitanje jer on sadrži kod koji definira vaš sustav. Ne biste niti trebali mijenjati niti jednu od datoteka u ovom direktoriju jer su one vlasništvo onoga tko je objavio operativni sustav.

U direktoriju /usr/spool ćete naći sve prelazne podatke koje koriste sustav za ispis, *demon* cron i komunikacijski sustav UUCP. Datoteke koje pošaljete na pisač će biti zadržane u ovom direktoriju prije nego dođu na red za ispis. Ovdje su pohranjeni i svi programi koji čekaju da ih pokrene *demon* cron, kao što su datoteke crontab i slično.

Direktorij /usr/lib sadrži sve ostalo što je dio standardnog operativnog sustava Linux. Općenito govoreći, ovaj direktorij predstavlja organizirani kaos koji je skriven iza relativno discipliniranog Linux sustava. U njemu ćete naći programe koje pozivaju drugi programi iz direktorija /bin i /usr/bin, kao i datoteke s podešenjima terminala i pisača, sustav za e-poštu, program cron i komunikacijski sustav UUCP.

Direktorij /usr sadrži poddirektorije dodijeljene korisnicima. Općenito vrijedi ovo pravilo: Ako je vaše korisničko ime "marija" i početni direktorij će se zvati /usr/marija.

Takav dogovor ima smisla kada su diskovi bili mali i skupi, no pojavom velikih i (relativno) jeftinih diskova, pojavili su se i bolji načini organiziranja Linuxa, kao i nova struktura direktorija koja je objašnjena u idućem poglavlju.

## Direktoriji u Linuxu

Jedan problem s klasičnom strukturom direktorija je taj da je otežana sigurnosna pohrana direktorija koji nisu u sklopu direktorija /usr. Postoje tri razine podataka koje zahtijevaju pohranu iz sigurnosnih razloga: osnovni sustav, podaci u tablicama koje definiraju osnovni sustav (na određenoj lokaciji) i korisnički podaci.

Osnovni sustav je dovoljno jednom pohraniti, a potom spremati samo promjene u upravljačke tablice. Korisnički podaci se učestalo mijenjaju i treba ih povremeno pohraniti na sigurno mjesto. U nastavku je prikazana tipična struktura direktorija u Linuxu. Na vašem računalu se struktura direktorija može donekle razlikovati ovisno o paketima koje ste instalirali na računalo.

```
/ /etc
/passwd (baza podataka korisnika)
/rc.d (skripta naredbi za inicijalizacija sustava)
/sbin
/bin
/tmp
/var
/lib
/home
/ <ovdje treba upisati korisni ko ime> (korisni ki ra uni)
/install
/usr
/bin
/proc
```

Direktoriji /bin, /etc/ i /tmp imaju istu namjenu kao i u klasičnoj strukturi direktorija. Tablica s definicijama sustava su premještene u direktorij /sbin, a svi standardni Linux programi se nalaze u direktoriju /usr/bin, koji je povezan s direktorijem /usr. Radi uskladivosti, svi klasični direktoriji su povezani simboličkim vezama. Direktorij /usr više ne sadrži korisničke podatke, ali da bi se izbjegla zbrka s bivšim direktorijem /usr/lib, i novi sustav prepoznaje direktorij /usr.

# Odavde...

U ovom poglavlju ste naučili kako se u Linuxu koriste datoteke i direktoriji te kako putem sustavskih dozvola možete zaštiti svoje podatke. Također ste naučili kako se mijenjaju dozvole nad datotekama i direktorijima i koje je značenje posebnih datoteka. Na kraju ste upoznali imena i namjenu najčešće korištenih direktorija u Linuxu. Detaljnije informacije potražite u ovim poglavljima:

- U 7. poglavlju, "Smisao administratora sustava" saznat ćete kako novim korisnicima dodijeliti dozvole za rad s datotekama.
- 14. poglavlje, "Upravljanje sustavom datoteka" nudi prikaz sustava datoteka i način njihova organiziranja u Linuxu.
- U 17. poglavlju, "Upravljanje datotekama i direktorijima" ćete saznati kako se mogu organizirati i koristiti datoteke i direktoriji.

# poglavlje **17**

# Upravljanje datotekama i direktorijima

(Jack Tackett)

# U ovom poglavlju

Ispis sadržaja direktorija Organizacija datoteka Kopiranje datoteka Premještanje i promjena imena datoteka Uklanjanje datoteka ili direktorija Pregled sadržaja datoteke Traženje datoteka Mijenjanje markica vremena i datuma Komprimiranje datoteka Velika ve}ina Linux naredbi se odnosi na rad s datotekama i direktorijima. U stvari, skripte s naredbama su prilago | ene za upravljanje datotekama i direktorijima. Upravljanje datotekama je vrlo te{ko u konvencionalnom jeziku (~ak i u C), ali je jednostavno uporabom velikog broja naredbi u Linux ljuskama.

Naredbe za upravljanje datotekama mogu se podijeliti u dvije kategorije:

- Naredbe koje upravljaju datotekama kao objektima
- Naredbe koje upravljaju sadr`ajem datoteka

U ovom poglavlju su nagla{ene naredbe koje upravljaju datotekama kao objektima, to jest naredbe koje premje{taju, mijenjaju ime, kopiraju, bri{u, tra`e i mijenjaju atribute datoteka i direktorija. U ovom poglavlju }ete na}i kratak pregled onih naredbi koje se koriste za rad sa sadr`ajem datoteka.

# Ispis sadržaja direktorija

Naredba ls je osnovna naredba za ispis datoteka u direktoriju. Na~in ispisa datoteka ovisi o na~inu na koji ste zadali naredbu ls. Ako upotrijebite naredbu u osnovnom obliku, svaka }e datoteka biti prikazana u posebnom retku. Isto vrijedi i za neke ina~ice UNIX-a, kao {to je SCO UNIX. Druge ina~ice UNIX-a ispisuju datoteke u nekoliko stupaca. Ispis datoteka u stupcima je u ve}ini slu~ajeva puno prakti~niji, tako da operativni sustavi koji ispisuju jednu datoteku u jednom retku imaju i alternativu naredbu lc.

Na~in prikaza datoteke se mo`e mijenjati uporabom parametara uz naredbu ls. Op}enito, ina~ice naredbe ls se mogu podijeliti u dvije kategorije: naredbe koje su nastale od sustava Linux V i one koje su nastale od ina~ice Berkeley. U ovom poglavlju }e biti govora o parametrima naredbe ls koji se koriste u sustavu Linux V. Ako ste u nedoumici koju ina~icu naredbe ls sami koristite, potra`ite upute za svoj operativni sustav ili pokrenite naredbu man ls.

**BILJEŠKA** Većina stranica za pomoć (stranica *man*), koje se spominju u ovom poglavlju, više ne postoje ili su netočne i nekompletne jer je distribucija Red Hat prešla na grafički prikaz pomoći (HTML ili Texinfo). No, za sada su navedeni podaci točni za inačicu Red Hat Linux 4.0.

Parametri koji se koriste s naredbom ls mogu biti razdvojeni ili zbijeni. To zna~i da idu}e naredbe imaju isti u~inak:

```
IS -I -F
i
IS -IF
```

U tablici 17.1 }ete na}i popis parametara koji se koriste s naredbom ls i to po abecednom redoslijedu.

Parametar	Opis
-a	Ispis svih datoteka. Ako niste unijeli ovu opciju ili opciju –A, ne}e biti prikazane datoteke koje po~inju to~kom (.). Linux mo`e i skrivati datoteke. Sve datoteke koje po~inju to~kom su skrivene jer one u pravilu slu`e za pode{avanje aplikacija. Primjerice, datoteka .profile se koristi za pode{avanje ljuski Bourne i Korn, a datoteka .mailrc za prilagodbu sustava za slanje e-po{te. S obzirom da gotovo svaka glavna naredba ima datoteku za pokretanje, va{ bi po~etni direktorij izgledao glomazno kada biste prikazali na zaslonu sve te datoteke. Ako ih ipak `elite vidjeti, onda upotrijebite parametar -a.
-A	Isti kao parametar -a, jedino {to ne}e biti prikazane datoteke-direktoriji i (u 16. poglavlju, "Sustav datoteka i direktorija" ste nau~ili je pseudonim za teku}i direktorij, a za naddirektorij teku}eg direktorija). S obzirom da ti direktoriji po~inju to~kom, parametar -a }e ih prikazati na zaslonu. Uporabom parametra -A ne}ete ih vidjeti na zaslonu.
-b	Forsirani ispis negrafi~kih znakova u oktalnom prikazu. Parametar -b je korisniji od parametra -q jer vam omogu}uje da saznate o kojim se znakovima radi.
-C	Sortiranje i prikaz prema vremenu kada je datoteka zadnji puta ure   ivana. Linux za svaku datoteku stvara tri markice s vremenom i datumom: vrijeme nastanka datoteke, vrijeme zadnjeg pristupa datoteci i vrijeme unosa zadnje promjene u datoteku. Obi~no se datoteke prikazuju prema ASCII redoslijedu (radi se o abecednom redoslijedu s tim da se prikazuju velika slova prije malih).
-C	Ispis u vi{e stupaca s datotekama poredanim od vrha prema dnu stupca. Ovo je osnovno pode{enje naredbe ls kada se radi o ispisu na zaslon.
-d <i>imedatoteke</i>	Ako je kao argument postavljen direktorij, onda }e naredba prikazati samo njegovo ime (a ne i sadr`aj). Ovaj se parametar ~esto koristi s parametrom -l kako bi se dobio status direktorija. Obi~no se ispisuje sadr`aj direktorija, ako je direktorij izravno zadan ili nazna~en pomo}u posebnih znakova (*, ?). Obi~na naredba ls ispisuje samo imena direktorija, a naredba ls * prikazuje datoteke, direktorije i sadr`aj svih direktorija koji se nalaze u teku}em direktoriju.
-F	Ozna~ava direktorije s kosom crtom (/), izvr{ne datoteke sa zvjezdicom (*), simboli~ke veze sa znakom (@), FIFO-e s oznakom ( ).
-i	U prvom stupcu se ispisuju brojevi inode svih datoteka (brojevi inode su obja{njeni u 16. poglavlju "Sustav datoteka i direktorija"). Ako ugledate povezane datoteke, uo~ite da one imaju isti broj <i>inode</i> .

Tablica 17.1 Parametri koji se koriste uz naredbu /s

nastavlja se

Tablica 17.1	Nastavak
Parametar	Opis
-1	Ispis sadr`aja direktorija u duga~kom formatu u kojem se vidi na~in rada, broj veza, vlasnik, veli~ina u bajtima i vrijeme zadnje promjene. Ako se radi o posebnim datotekama, umjesto veli~ine }ete vidjeti glavne i sporedne brojeve ure aja. Ako je datoteka mijenjana prije vi{e od {est mjeseci, prikazani }e biti mjesec, datum i godina, a u protivnom samo datum i vrijeme zadnje promjene. Ako se radi o simboli~koj vezi, ispisana je i staza do datoteke (ispred koje stoji znak ->). Opciju -l mo`ete kombinirati s drugim opcijama, kao {to je opcija -n koja prikazuje korisni~ki i grupni identifikacijski broj umjesto imena datoteke.
-n	Ispis korisni~kih i grupnih identifikacijskih brojeva (umjesto imena) koji su pridijeljeni svakoj datoteci i direktoriju. Obi~no su prikazana samo imena. Ako pode{avate mre`ni softver, kao {to je primjerice TCP/IP, korisno je znati identifikacijske brojeve da biste pridijelili dozvole na ve}em broju ra~unala.
-q	Prikaz negrafi~kih znakova u imenima datoteka s oznakom ?. Za naredbu ls ovo je osnovno pode{enje ako se ispis {alje na terminal. Ako ste gre{kom stvorili datoteke koje u imenu sadr`e znakove koji se ne daju ispisati, pomo}u parametra -q }ete ih mo}i prikazati.
-r	Promjena redoslijeda prikaza datoteka (obrnuti abecedni redoslijed ili prikaz starijih datoteka na vrhu).
-S	Daje veli~inu (u KB) svake datoteke uklju~uju}i i sve neizravne blokove koji se koriste za mapiranje datoteke. Ako je definirana varijabla POSIX_CORRECT, onda veli~ina bloka iznosi 512 bajta.
-t	Sortiranje prema vremenu mijenjanja datoteke (najnoviji su prikazani prvi), a ne prema imenu. Ako datoteke s najstarijim vremenom `elite prve vidjeti, onda iskoristite kombinaciju –rt.
-u	Koristi se vrijeme zadnjeg pristupa (a ne zadnjeg mijenjanja datoteke) za sortiranje (opcija –t) ili ispis (opcija –l).
- X	Forsira ispis u vi{e stupaca s datotekama poredanim s lijeva na desno a ne odozgo prema dolje.

Ako ste instalirali distribuciju Slackware, mo`ete uo~iti da naredba ls u boji prikazuje razne vrste datoteka. Boje su definirane u datoteci s pode{enjima DIR\_COLORS koja se nalazi u direktoriju /etc. Prema osnovnom pode{enju su izvr{ne datoteke prikazane zelenom bojom, direktoriji plavom, a simboli~ke veze ru`i~astom bojom. Da biste promijenili boje kojima se prikazuju odre | ene vrste datoteka, morat } ete kopirati datoteku DIR\_COLORS u po~etni direk-torij i promijeniti njezino ime u .dir colors. U tablici 17.2 nalaze se boje koje su vam na raspola-ganju. Vi{e informacija potra`ite u stranicama za pomo} (*man* stranice) i u samoj datoteci DIR\_COLORS.

BILJ	EŠKA	Ako koristite distribuciju Red Hat, onda morate upisati Is color da bi dobili ispis
		datoteka u bojama.

Tablica 17.2	Vrijednosti u datoteci DIR_COLORS
Vrijednost	Opis
0	vra}anje osnovnih boja
1	svjetlije boje
4	podcrtani tekst
5	blije{te}i tekst
30	crna boja teksta
31	crvena boja teksta
32	zelena boja teksta
33	`uta (ili sme a) boja teksta
34	plava boja teksta
35	ru`i~asta boja teksta
36	ljubi~asta boja teksta
37	bijela (ili siva) boja teksta
40	crna pozadina
41	crvena pozadina
42	zelena pozadina
43	`uta (ili sme a) pozadina
44	plava pozadina
45	ru`i~asta pozadina
46	ljubi~asta pozadina
47	bijela (ili siva) pozadina

Postoje i druge mogu}nosti koje ovdje nisu prikazane. Potra`ite stranicu za pomo} naredbe ls (*man* stranicu)

# Organizacija datoteka

Ne postoji ~vrsto pravilo za organiziranje datoteka u Linuxu. Datoteke ne zahtijevaju dodatke imenima (primjerice .EXE za izvr{ne datoteke) kao u MS-DOS-u. Morali biste sami stvoriti sustav davanja imena datotekama, no klasi~na organizacija datoteka u Linuxu se zasniva na poddirektorijima.

No, sve vi{e programa dolaze iz DOS-a i "donose" svoja pravila za organizaciju datoteka u Linuxu. Iako to mo`da nije nu`no, ku}e koje izdaju programe upu}uju vas na kori{tenje programa zajedno s njihovim ekstenzijama. Namjeravate li sami pisati naredbe, korisno je tako organizirati direktorije da oni odgovaraju organizaciji u Linuxu (direktoriji /bin /lib i /etc). Stvorite vlastitu strukturu poddirektorija koriste}i navedena imena unutar direktorija /home. Osim toga, zgodno je slijediti tradiciju nastalu u Linuxu i izvr{ne programe spremati u direktorij /bin, naredbe koje se u~estalo koriste u direktorij /lib, a datoteke s pode{enjima u direktorij /etc. Naravno da takva organizacija nije nu`na, ali i to je jedan od na~ina kako mo`ete organizirati svoje datoteke.

Direktorije stvarate pomo}u naredbe mkdir. Oblik te naredbe je vrlo jednostavan:

mkdir *i medi rektorija* 

*Imedirektorija* predstavlja ime koje `elite pridijeliti novom direktoriju. Naravno, da biste u odre | enom direktoriju mogli stvoriti poddirektorij, morate imati dozvolu za pisanje, na ako novi direktorij stvarate u direktoriju /home, onda ne biste smjeli imati nikakvih problema.

Pretpostavimo da ste napisali tri programa pod imenima prog1, prog2 i prog3, te da se oni nalaze u direktoriju \$HOME/bin (to jest u va{em po~etnom direktoriju). Ako svoje privatne programe `elite pokrenuti kao da se radi o standardnim dijelovima Linuxa, morate direktorij \$HOME/bin dodati u varijablu PATH. Da biste to u~inili, trebate u ljusci Bourne ili Korn pokrenuti ovu naredbu:

```
PATH=$PATH: $HOME/bin; export PATH
```

U C ljusci trebate koristiti ovu naredbu:

setenv PATH "\$PATH \$HOME/bin"

BILJEŠKA Upamtite da je \$HOME zamjena za kompletnu stazu do vašeg početnog direktorija. Ako je vaš početni direktorij /home/ams, onda će se izraz \$HOME/bin interpretirati kao /home/ams/bin. ■

Ako va{i programi pozivaju potprograme, mo`da }ete trebati stvoriti poddirektorije unutar direktorija \$HOME/lib. Za svaki program mo`ete stvoriti poseban poddirektorij. Tako pomo}u naredbe pgm1 po`ete pokrenuti, primjerice, program \$HOME/lib/pgm1/pgm1a.

Sli~no, ako va{ program prog1 zahtijeva pokreta~ku tablicu (startup), tu tablicu trebate nazvati \$HOME/etc/pgm1.rc, a va{i podaci se mogu nalaziti u direktoriju \$HOME/data/pgm1.

# Kopiranje datoteka

Naredba koja se koristi za kopiranje datoteka izgleda ovako: cp *iz u.* za kopiranje datoteke potrebna vam je dozvola za ~itanje datoteke koju kopirate kao i dozvola za pisanje u direktorij u koji `elite kopirati datoteku (potrebna vam je i dozvola za pisanje u datoteku ako namjeravate novom datotekom prebrisati postoje}u datoteku koja se nalazi u novom direktoriju). Ne postoje druga ograni~enja glede kopiranja datoteka.

Prilikom kopiranja datoteka potrebno je paziti na nekoliko stvari:

Ako prilikom kopiranja datoteke novoj datoteci date ime datoteke koja ve} postoji i u koju mo`ete pisati (imate dozvolu pisanja), tada }e nova datoteka prebrisati postoje}u.

- Ako kao mjesto kopiranja upi{ete ime direktorija, naredba cp }e kopirati datoteku u odabrani direktorij i to pod originalnim imenom. Ako napi{ete naredbu cp datoteka direktorij, datoteka }e biti kopirana i direktorij pod imenom direktorij/datoteka.
- Mo`ete odjednom kopirati vi{e datoteka u isti direktorij ako pokrenete naredbu cp datoteka1 datoteka2 datoteka3 ... direktorij. Ako se na kraju popisa ne nalazi ime direktorija, program }e javiti gre{ku, kao i u slu~aju da ste u popisu naveli ime direktorija, a ne datoteke.
- Budite pa`ljivi kada u naredbi cp koristite posebne znakove (\*, ?) jer gre{kom mo`ete kopirati vi{e datoteka nego {to ste mislili.
- **BILJEŠKA** S obzirom da mnogi korisnici Linuxa na svojim računalima imaju i MS-DOS datoteke, te da obično i DOS sustav datoteka koriste pod Linuxom, većina Linux naredbi prepoznaje slučaj kada se datoteka kopira u ili iz DOS particije. U tom slučaju Linux mora obaviti neophodni "prijevod" datoteka prilikom njihova kopiranja. "Prijevod" datoteka je nužan stoga što DOS datoteke kraj pojedinog retka označavaju znakovima za prelazak u novi red i vraćanje na početak novog retka. Većina Linux i UNIX sustava za prelazak u novi redak koristi isključivo znak za vraćanje na početak retka, koji se naziva newline.

# Premještanje i promjena imena datoteka

U Linuxu se premje{tanje i promjena imena datoteka obavlja pomo}u iste naredbe: mv. Oblik naredbe kao i pravila njezinog kori{tenja su identi~ni kao kod naredbe za kopiranje cp. Proizvoljan broj datoteka mo`ete premjestiti iz jednog u drugi direktorij uz napomenu da ime novog direktorija mora biti zadnje u popisu te da morate imati dozvolu za pisanje u novi direktorij.

Jedna stvar koju mo`ete raditi s naredbom mv, a ne mo`ete s naredbom cp, je premje{tanje i promjena imena direktorija. Ako premje{tate ili mijenjate ime datoteke, jedino {to se doga|a je promjena zapisa o odre|enoj datoteci (zapis se nalazi u datoteci-direktoriju). Ako se nova lokacija datoteke nalazi na drugom fizi~kom disku ili particiji, tada }e sete fizi~ki premjestiti datoteku te promijeniti sadr`aj datoteke-direktorija.

Ako `elite obrisati (kopirati) direktorij te poku{ate pokrenuti naredbu rm ili cp bez ikakvih parametara, na zaslonu }e se pojaviti poruka koja vas upozorava da poku{avate ukloniti (kopirati) direktorij. Da biste obrisali ili kopirali datoteku-direktorij, morate upotrijebiti parametar –r. Naredba mv mo`e bez ikakvih problema premje{tati direktorije.

# Uklanjanje datoteka ili direktorija

Naredba koja slu`i za uklanjanje datoteka je naredba rm. Da biste brisali datoteku, nije potrebno da budete njezin vlasnik ve} samo da imate dozvolu ~itanja i pisanja. Ako ste vlasnik datoteke, tada je mo`ete i obrisati, uz pretpostavku da sami niste ukinuli dozvolu ~itanja i pisanja u datoteku. Primjerice, ako isklju~ite dozvolu pisanja u odre|enu datoteku tako da upi{ete chmod 000 *datoteka*, morat }ete ponovno otvoriti dozvolu tako da pokrenete naredbu chmod 064 *datoteka*. Tek nakon toga }ete mo}i obrisati datoteku.

Ako gre{kom upi{ete **rm** \*, obrisat }ete sve datoteke za koje imate dozvolu brisanja, a koje se nalaze u teku}em direktoriju. To naredbom se ne bri{u poddirektoriji. Da biste obrisali i njih, morat }ete upisati opciju –r.

Neke ina~ice naredbe rm tra`e potvrdu da doista `elite obrisati datoteke kojima ste vlasnik, ali za koje nemate dozvolu za pisanje. Druge ina~ice ove naredbe tra`e potvrdu za sve datoteke koje ste odabrali koriste}i posebne znakove (\*,?). Osim toga, mo`ete i sami napisati makro naredbu ili skriptu koja vam ostavlja mogu}nost da se predomislite u tijeku brisanja datoteka.

Ako ina~ica naredbe rm koju imate staje u trenutku kada treba obrisati datoteke kojima ste vlasnik, ali za koje nemate dozvolu pisanja, onda se na sljede}i na~in mo`ete djelomi~no za{titi od ne`eljenog brisanja svega {to se nalazi u direktoriju:

- 1. Stvorite datoteku pod imenom 0. Kao ASCII znak, broj 0 se u popisu nalazi prije bilo kojeg slova.
- Uklonite sve dozvole za datoteku pod imenom 0, odnosno upi{ite chmod 000 0. Tom }ete naredbom ukloniti dozvole za ~itanje, pisanje i izvo |enje datoteke svim korisnicima, uklju~uju}i i vas same.
- 3. Upi{ete li naredbu **rm** \*, datoteka pod imenom 0 }e biti prvo koju }e program poku{ati ukloniti.

Ako se va{a ina~ica naredbe rm zaustavlja prije brisanja datoteke 0, nakon {to ste je pokrenuli pomo}u **rm** \*, onda imate prigodu razmisliti `elite li doista obrisati datoteku. Ako vam nije bila namjera obrisati ~itav sadr`aj direktorija, tada pritisnite tipku <Del>

ili  $\langle Ctrl-c \rangle$  i prekinite naredbu rm. Za provjeru mo`ete poku{ati obrisati samo datoteku pod imenom 0. Ako se va{a ina~ica programa ne zaustavlja na datoteci 0, nemojte koristiti naredbu rm \* jer }ete obrisati sve datoteku u direktoriju.

Bolji na~in da se za{titite od ne`eljenog brisanja svih datoteka je da upotrijebite parametar –i (dolazi od rije~i interaktivan) zajedno s naredbom rm. Zadate li naredbu rm –i

*imedatoteke*, program }e vas priupitati `elite li doista obrisati pojedinu datoteku. Morate potvrdno odgovoriti da biste datoteku doista obrisali. Upi{ete li naredbu **rm** –**i** \*, morat }ete potvrditi brisanje svake datoteke koja se nalazi u va{em direktoriju. Na taj }ete na~in imati dovoljno vremena da promislite {to doista `elite u~initi.

#### UPOZORENJE

Razmislite prije nego obrišete datoteke. Za razliku od Windowsa, DOS-a ili MAC-a, nakon što obrišete datoteku (u većini inačica Linuxa), ona će nestati i jedini način da je vratite je sigurnosna kopija podataka, koju ste prethodno načinili, zar ne?

Vidi "Sigurnosna pohrana podataka i obnova datoteka", 229. stranica.

Ako ~esto koristite naredbu rm –i, mo`ete je implementirati na dva na~ina: tako da napi{ete skriptu naredbi ili da stvorite funkciju u ljusci. Ako pi{ete skriptu, upamtite da ljuska tra`i naredbe koja bi se trebale nalaziti u direktorijima zapisanim u varijabli PATH (i to prema redoslijedu kojim su zapisani direktoriji). Ako je posljednji u popisu direktorij \$HOME/bin, onda se nikada ne}e na}i skripta pod imenom rm. Postavite direktorij \$HOME/bin na prvo mjesto u varijabli PATH ili na~inite novu naredbu kao {to je del. Stvorite li skriptu pod imenom del, morate datoteku ozna~iti kao izvr{nu (koriste}i naredbu chmod) da bi je ljuska prepoznala. Ako`elite stvoriti naredbu del, dovoljno je zapisati samo jednu naredbu rm –i \$\*. Ako potom upi{ete naredbu **del** \*, ljuska }e je pretvoriti u naredbu rm –i \*.

Vidi "Uređivanje naredbi u ljusci i njihova zamjena drugim naredbama", 364. stranica

Drugi na~in da obavite isti zadatak je uporaba naredbe *alias*. Radi se o internoj naredbi u ljusci (sli~no kao i naredba doskey koja se pojavila u MS-DOS ina~ici 5).

Da biste dodali naredbu *alias* u C ljusci, morate preurediti datoteku .cshrc. Mo`ete upotrijebiti bilo koji program za ure|ivanje teksta (vidi 8. poglavlje "Uporaba *vi* editora"). Ako koristite C ljusku, tada na po~etak datoteke .cshrc dodajte sljede}e retke:

```
rm ()
{
/bin/rm -i $*
}
```

Za dodavanje naredbe alias u Korn ljusci dodajte ovaj redak u datoteku \$HOME/.kshrc:

```
alias rm'rm -i $'
```

Poku{ate li obrisati direktorij s naredbom rm, ugledat }ete poruku da se radi o direktoriju koji ne mo`ete obrisati. Ako `elite obrisati prazan direktorij, upotrijebite naredbu rmdir, kao {to to ~inite u MS-DOS-u.

Linux nudi drugi na~in brisanja direktorija i njihovih sadr`aja, no ta je metoda bitno opasnija. Naredbom rm –r obrisat }ete sve direktorije i datoteke koji se nalaze u odre | enom direktoriju. Ako imate direktorij ./foo koji u sebi sadr`i druge direktorije i datoteke, onda naredbom rm –r foo brićete kompletan sadr`aj tog direktorija

i datoteke, onda naredbom rm –r foo bri{ete kompletan sadr`aj tog direktorija.

Pokrenete li naredbu rm –i –r, morat }ete potvrditi brisanje svakog direktorija koji se nalazi unutar zadanog direktorija. Prije nego odre |eni direktorij i njegov sadr`aj bude obrisan, morate potvrditi brisanje. Ako ostavite bilo koju datoteku u direktoriju koji namjeravate obrisati, program rm }e prekinuti rad ba{ kao da ste zadali naredbu rm bez ikakvih opcija te s njome poku{ali obrisati direktorij koji nije potpuno prazan.

BILJEŠKA Ne morate svaki parametar odvojeno zadavati, već ih možete kombinirati. Naredbu rm –i –r, možete zapisati i kao rm –ir. ■

# Pregled sadržaja datoteke

Gotovo svaka naredba u Linuxu daje ispis na standardni izlazni ure | aj, odnosno zaslon. Ako naredba ulazne podatke dobije iz neke ulazne datoteke, onda }e se i sadr`aj te datoteke pojaviti na zaslonu. Odabir odgovaraju}e Linux naredbe ovisi o na~inu na koji `elite da se prika`e datoteka. Postoje tri standardne naredbe: cat, more i less.

**BILJEŠKA** Linux, kao i svi UNIX sustavi, na početku rada otvara četiri sistemske datoteke: standardni ulaz, standardni izlaz, standardna greška i dodatni uređaj (AUX). Te četiri datoteke predstavljaju četiri fizička uređaja:

Ime	Alias	Uređaj
Standard input	standard in (stdin)	Tipkovnica
Standard output	standard out (stdout)	Zaslon
Standard error	standard err (stderr)	Zaslon
AUX	auxiliary	Dodatni uređaj

## Uporaba naredbe cat

Za prikaz kratkih ASCII datoteka najbolje je upotrijebiti naredbu cat (kratica od concatenate – spojiti). Naredba cat kao ulaz uzima popis datoteka (ili pojedinu datoteku) i ispisuje njihov sadr`aj na zaslon (bez prekida). Osnovna joj je namjena spajanje datoteka (cat *datoteka1 datoteka2> datoteka3*), ali se mo`e upotrijebiti i za prikaz kratkih datoteka na zaslonu.

Poku{ate li prikazati ve}e datoteke pomo}u naredbe cat, ~itav sadr`aj datoteke }e "projuriti preko" zaslona brzinom kojom zaslon mo`e prikazati niz znakova. Jedini na~in da zaustavite ispis sadr`aja na zaslonu je da pritisnete tipke <Ctrl-s> i <Ctrl-q> te na zaslon po{aljete poruke start i stop. Drugi na~in je da pokrenete jednu od naredbi more ili less.

## Uporaba naredbe more

Naredbe more i less prikazuju sadr`aj datoteke na zaslonu (jedan po jedan zaslon). Iako obje naredbe ~ine istu stvar, one to rade na razli~ite na~ine. Broj redaka koji }e se prikazati na zaslonu ovisi o pode{enju varijabe TERM.

Naredba more je ne{to starija od naredbe less i nastala je od Berkeley ina~ice UNIX-a. Pokazala se toliko korisnom da je, kao i *vi* editor, postala standard u Linuxu. U ovom odlomku }ete na}i osnovne informacije o naredbi more.

Najjednostavniji oblik naredbe je more *imedatoteke*. Na zaslonu }e se pojaviti sadr`aj datoteke. @elite li vidjeti sljede}i zaslon, trebate pritisnuti razmaknicu. Ako pritisnete tipku <Return>, pojavit }e se samo idu}i redak. Ako istovremeno pregledavate sadr`aj ve}eg broja datoteka (more *datoteka1 datoteka2...*), te `elite neku od datoteka ure|ivati, morat }ete upotrijebiti naredbu e ili v. Pritiskom na tipku <e> pokrenut }ete program za ure|ivanje datoteka koji je za pojedinu datoteku odre|en varijablom EDIT, a pritiskom na tipku <v> pokrenut }ete program koji je definiran pomo}u varijable VISUAL. Ako niste definirali sadr`aj tih dviju varijabli, program }e pokrenuti editor ed (ako pritisnete tipku <e>) ili vi editor (ako pritisnete tipku v).

Vidi "Podešavanje okružja ljuske", 344. stranica.

Naredba more ima samo jednu slabu to~ku, ne mo`ete se vra}ati i prikazati sadr`aj koji ste ve} pro{li. No, tu vam mogu}nost nudi naredba less.

## Uporaba naredbe less

Jedini nedostatak naredbe less je {to ne mo`ete upotrijebiti niti jedan program za ure | ivanje datoteke koju vidite na zaslonu. No, ova naredba ima mogu}nost vra}anja na pregledani dio datoteke.

Naredba less se koristi na gotovo isti na~in kao i naredba more. Da biste vidjeli sadr`aj datoteke (jedan zaslon za drugim), upi{ite naredbu **less** *imedatoteke*, i na zaslonu }e se pojaviti po~etak datoteke. Za prelazak na idu}i zaslon pritisnite razmaknicu kao {to to ~initi kod uporabe naredbe more.

Za povratak na prethodni zaslon pritisnite tipku <br/> <br/> @elite li se premjestiti na odre | eno mjesto u datoteci (koje je odre | eno postotkom sadr`aja datoteke), tada najprije pritisnite tipku i zatim upi{ite postotak.

## Pretraživanje datoteke i prelazak u ljusku

Naredbe less i more omogu}uju vam tra`enje niza znakova u datoteci koja je prikazana na zaslonu. Naredba less omogu}uje i tra`enje unatrag, za razliku od naredbe more. @elite li pretra`ivati datoteku od kraja prema po~etku, onda upotrijebite ovaj oblik naredbe less /string. Ako jedna od naredbi less ili more prona|e tra`eni skup znakova, on }e se pojaviti na vrhu zaslona. Pritisnete li tipku <n>, naredba less }e ponoviti prethodno tra`enje znakova.

Naredbe more i less omogu}uju prelazak u ljusku pomo}u tipke !. Nakon {to pokrenete naredbu !, na}i }ete se u podljusci koju mo`ete napustiti na isti na~in kao kad se odjavljujete sa sustava. Ovisno o vrsti ljuske u kojoj se nalazite, mo}i }ete je napustiti pritiskom na tipke <Ctrl-d> ili upisom naredbe **exit**. Program vas vra}a na mjesto u datoteci na kojem ste se nalazili prije pokretanja naredbe !. Ako pritisnete kombinaciju tipki <Ctrl-d> i na zaslonu se pojavi poruka o gre{ci, morat }ete pokrenuti naredbu logout da biste napustili ljusku i vratili se u datoteku.

## Pregled datoteka u drugim oblicima

Druge naredbe prikazuju sadr`aj datoteka na razli~ite na~ine. Primjerice, `elite li pogledati sadr`aj binarne datoteke, prika`ite je pomo}u naredbe od ({to je kratica od *octal dump*). Naredba od prikazuje sadr`aj datoteke u oktalnom (baza 8) zapisu. Kori{tenjem razli~itih parametara datoteku mo`ete prikazati u decimalnom, ASCII ili heksadecimalnom (baza 16) zaSpisu.

#### Oktalni, decimalni i heksadecimalni zapis

Prikazivanje binarnih podataka može predstavljati problem. Ako binarna datoteka sadrži ASCII znakove, onda je nije problem prikazati u ASCII obliku (kakav i očekujete kada pregledavate većinu datoteka). No, ako se radi o datoteci koja sadrži program, onda podaci neće moći biti prikazani u ASCII obliku, već ih treba pregledati u nekom brojčanom obliku.

Ranija mikro računala su koristila 12-bitne riječi. Današnja računala su podešena za rad s 8-bitnim jedinicama u memoriji. Iako podatke možete prikazati u decimalnom obliku (baza 10) koji vam je zasigurno bliži, još uvijek ostaje pitanje što prikazati: bajt, riječ ili 32 bita? Da biste prikazali određeni broj bita, potrebno je bazu 2 povećati na željeni broj bita. Kod starih, 12-bitnih sustava, moglo se pomoću četiri broja prikazati svih 12 bita. S obzirom da se UNIX prije koristio na takvim mikro računalima, većina zapisa u UNIX-u, a time i u Linuxu, je u oktalnom obliku. Svaki bajt se može prikazati u oktalnom kodu s tri znamenke i to na ovaj način: \010

S obzirom da je jedna riječ smještena unutar jednog bajta (8 bita), oktalni sustav više nije bio pogodan način prikaza podataka, već se počeo koristiti heksadecimalni sustav (baza 16 ili 24). Jedan 8-bitni bajt se može prikazati pomoću dvije heksadecimalne znamenke (bajt čija je decimalna vrijednost 10 u heksadecimalnom sustavu se prikazuje kao OA).

Naredba od odre|uje na~in prikaza binarnih podataka. Op}eniti oblik naredbe je sljede}i:

od [opcija]... [datoteka]...

ili od -traditional [datoteka] [[+]offset [[+]oznaka]]

U tablici 17.3 }ete na}i sa`etak parametara koje mo`ete koristiti s naredbom od.

Tablica 17.3	Parametri naredbe od	
Kratki oblik	Puni oblik	Opis
-A	address-radi x= <i>radi x</i>	Odre uje na~in ispisa offseta
– N	read-bytes= <i>bytes</i>	Ograni~ava veli~inu spremnika
-j	skip-bytes= <i>bytes</i>	Preska~e odre eni broj bajtova na po~etku datoteke
-S	<ul><li>-strings[=bytes]</li></ul>	Ispis odre enog broja grafi~kih znakova
-t	format= <i>type</i>	Odabir izlaznog formata
-V	output-duplicates	Spre~ava uporabu znaka (*) za ozna~avanje komentara
– W	<pre>width[=bytes]</pre>	Odre uje {irinu retka u ispisu
	traditional	Prihva}anje argumenata u pre-POSIX obliku
	help	Prikaz ove pomo}i i izlazak iz programa
	version	Prikaz informacija o ina~ici i izlazak iz programa

U tablici 17.4 nalaze se argumenti u pre-POSIX obliku koji se mogu koristiti zajedno s argumentima iz tablice 17.3 ~ime se njihovo djelovanje objedinjuje.

Tablica 17.4	Argumenti u pre-POSI	X obliku
Kratki oblik	POSIX ekvivalent	Opis
-a	-t a	Odabir imenovanih znakova
-b	-t oC	Odabir oktalnih bajtova
-C	-t c	Odabir ASCII znakova ili backslash escape
-d	-t u2	Odabir neozna~enog decimalnog formata (kratki)
-f	-t fF	Odabir oblika s plivaju}im zarezom
-h	-t x2	Odabir heksadecimalnog oblika (kratkog)
- i	-t d2	Odabir decimalnog oblika (kratkog)
-1	-t d4	Odabir decimalnog oblika (dugog)
-0	-t o2	Odabir oktalnog oblika (kratkog)
- X	-t x2	Odabir heksadecimalnog oblika (kratkog)

U starijem obliku naredbe *offset* zna~i –j *offset. Oznaka* predstavlja pseudo adresu prvog bajta koji se ispisuje. Prefiksi 0x ili 0X ozna~avaju da se radi o heksadecimalnom sustavu (kod *offseta* i *oznake*). Sufiks mo`e biti to~ka (.) ako se radi o oktalnom sustavu, ili se mo`e mno`iti s brojem 512.

Parametar type mo`e sadr`avati jednu ili vi{e zastavica koje su prikazane u tablici 17.5.

Tablica 17.5	Vrste zastavica
Zastavica	Opis
а	Imenovani znak
С	ASCII znak ili backslash escape
d[veli~ina]	Ozna~eni decimalni oblik, veli~ina u bajtima po integeru
f[veli~ina]	Oblik s plivaju}im decimalnim zarezom, veli~ina u bajtima po integeru
o[veli~ina]	Oktalni oblik, veli~ina u bajtima po integeru
u[veli~ina]	Neozna~eni decimalni oblik, veli~ina u bajtima po integeru
x[veli~ina]	Heksadecimalni oblik, veli~ina u bajtima po integeru

U tablici 17.5 *veli~ina* se zadaje broj~ano ili se mo`e upisati C za sizeof (char), S za sizeof (short), I za sizeof (int) ili L za sizeof (long). Ako vrijednost varijable *type* iznosi f, onda *veli~ina* mora biti F za sizeof (float), D za sizeof (double) ili L za sizeof (long double).

**BILJEŠKA** sizeof predstavlja funkciju u programskom jeziku C, a rezultat te funkcije je broj bajtova u strukturu podataka. Primjerice, iduću funkciju možete upotrijebiti kako biste odredili broj bajtova u varijablama integer koji se koriste na vašem računalu (taj broj bajtova ovisi o sustavu): si zeof ( int );

Parametar *radix* (tablica 17.3) ima vrijednost d za decimalni sustav, o za oktalni, x za heksadecimalni i n za nijedan broj~ani sustav. Vrijednost *bytes* je heksadecimalna ako ima prefiks 0x ili 0X, a mno`i se s 512 ako ima nastavak b, sa 1024, ako ima nastavak k i sa 1048756 ako ima nastavak m. Parametar –s (ako nema broja) predstavlja brojku 3, -w (bez broja) predstavlja brojku 32. Osnovno pode{enje naredbe od je: od –A o –t d2 –w 16.

## Traženje datoteka

Ako datoteku niste na{li pomo}u naredbe ls, mo`ete upotrijebiti naredbu find. Radi se o vrlo mo}noj naredbi, ali i vrlo slo`enoj za kori{tenje. Naredba find se sastoji od tri glavna dijela, a svaki od njih ima jo{ nekoliko manjih dijelova:

- gdje treba tra`iti
- {to treba tra`iti
- {to treba raditi kada se prona | e datoteka

Ako znate ime datoteke, ali vam nije poznato gdje se ona nalazi, onda }e naredba find imati ovaj oblik:

find / -name imedatoteke -print

#### **UPOZORENJE**

Budite oprezni kada u direktoriju root pokrećete naredbu find. Na velikim sustavima može proći dosta vremena dok program pretraži sve direktorije koji se nalaze unutar direktorija *root* (možda će trebati tražiti i udaljene diskove).

Bilo bi pametno ograni~iti tra`enje na jedan ili dva direktorija. Primjerice, ako znate da se datoteka vjerojatno nalazi u direktorijima /usr ili /usr2, onda upi{ite sljede}u naredbu:

find /usr /usr2 -name imedatoteke -print

Na raspolaganju su vam mnoge opcije koje se koriste s naredbom find, a u tablici 17.6 vidjet }ete samo neke od njih. Da biste vidjeli sve raspolo`ive opcije, trebate pokrenuti naredbu man find.

Tablica 17.6	3 Neke od opcija koje se koriste s naredbom find				
Naredba	Opis				
-name <i>i me</i>	Varijabla ime mo`e predstavljati ime datoteke ili ime uz uporabu specijalnih znakova. Ako se radi o ovom drugome, na zaslonu }ete vidjeti sve datoteke koje udovoljavaju zadanom kriteriju tra`enja.				
-links <i>n</i>	Sve datoteke koje imaju n ili vi{e veza }e biti prikazane na zaslonu (slovo n zamijenite odgovaraju}im brojem).				
-size <i>n</i> [c]	Pronalazi bilo koju datoteku koja zauzima n ili vi{e blokova (512 bajtova). Ako uz broj n upi{ete slovo c, program tra`i datoteke s n ili vi{e znakova.				
-atime <i>n</i>	Odabir svih datoteka kojima ste pristupili u zadnjih n dana. Upamtite da }ete pokretanjem naredbe find promijeniti markicu s datumom i vremenom pristupa pojedinoj datoteci.				
-exec cmd	Nakon {to odaberete odre   ene datoteke, mo`ete pokrenuti Linux naredbu koja te datoteke koristi kao argument. Kod uporabe parametra -exec postoje dva jednostavna pravila: ime odabrane datoteke se nalazi u viti~astim zagradama, a naredba se poni{tava upisom znakova \;. Pretpostavimo da ste se prijavili za rad kao super korisnik i da`elite stvoriti korisni~ki direktorij. Rezultat je da }e vlasnik svih datoteka biti superkorisnik, iako datoteke pripadaju obi~nom korisniku. Pokrenite idu}u naredbu kako biste promijenili vlasnika svih datoteka koje se nalaze u direktoriju /home/jack i u pripadnim poddirektorijima: fi nd /home/j ack –exec chown j ack { } \;				
-pri nt	Ovaj parametar se ~esto koristi i slu`i za ispis imena odabranih datoteka i mjesta na kojem se one nalaze.				

Ovaj fajl je skinut sa sajta www.tutoriali.org

Naredba find vam omogu}uje jo{ neke na~ine odabira datoteka. Primjerice, `elite li prona}i datoteke koje ne mo`ete zbirno prikazati pomo}u posebnih znakova, morate upotrijebiti opciju or (-o) i to na sljede}i na~in:

find /home ( - name datoteka1 -o -name datoteka2 ) -print

Mo`ete birati neograni~eni broj kriterija za tra`enje datoteka. Ako ne upi{ete opciju –o, program podrazmijeva da ste `eljeli unijeti logi~ki operator *and*. Ako, primjerice, zadate naredbu find –size 100 -atime 2, to zna~i da `elite na}i datoteke veli~ine od barem 100 blokova kojima ste pristupali unatrag zadnja dva dana. Ako kombinirate vi{e and/or kriterija, korisno je upotrijebiti zagrade (kao u prethodnom primjeru) kako bi se izbjeglo pogre{no tuma~enje kriterija.

# Mijenjanje markica vremena i datuma

Svaka datoteka u Linuxu koristi tri markice vremena i datuma: datum nastanka datoteke, datum zadnje promjene i datum zadnjeg pristupa datoteci. Datum i vrijeme stvaranja datoteke se ne mo`e mijenjati osim kopiranjem datoteke i promjenom imena. Svaki puta kada se datoteka ~ita ili otvara mijenja se i markica koja ozna~ava vrijeme pristupa datoteci. Kao {to je spomenuto u prethodnom odlomku, uporabom naredbe find promijenit }e se i vrijeme zadnjeg pristupa datoteci.

Ako je na bilo koji na~in promijenjen sadr`aj datoteke, to jest ako ste u datoteku ne{to zapisali, promijenit }e se vremenske markice koje ozna~avaju mijenjanje i pristup datoteci. Markice s datumom su korisne ako `elite odabrati samo one datoteke koje su mijenjane u odre|enom vremenskom razdoblju. Za to vam mo`e poslu`iti naredba find.

@elite li a`urirati vremensku markicu koja ozna~ava vrijeme zadnjeg mijenjanja datoteke (a da pri tome ne mijenjate sadr`aj datoteke), mo`ete to u~initi pomo}u naredbe touch. Osnovno pode{ena naredba touch a`urira vremenske markice tako da im pridjeljuje trenutno vrijeme, a ako ne postoji odre|ena datoteka, naredba touch je sama stvara.

Naredbu touch mo`ete upotrijebiti kako biste zavarali naredbe koje provjeravaju datume i vremena. Primjerice, ako va{ sustav pokre}e naredbu za sigurnosnu pohranu onih datoteka koje su promijenjene nakon odre|enog datuma, mo`ete pokrenuti naredbu touch i "dotaknuti" datoteku koja nije mijenjana u tom razdoblju kako bi i ona bila uvr{tena u pohranu podataka.

Naredba touch koristi sljede}a tri parametra koja mo`ete upotrijebiti za promjenu njezina osnovnog pona{anja:

- -a A`urira samo datum i vrijeme pristupa datoteci
- -m A`urira samo datum i vrijeme mijenjanja datoteke
- -C Spre~ava stvaranje datoteke ako ona ne postoji

Osnovni oblik naredbe touch je: touch -am popisdatoteka.

## Komprimiranje datoteka

Ako na svom ra~unalu nemate dovoljno prostora, ili ako posjedujete velike ASCII datoteke koje rijetko koristite, mo`ete ih komprimirati i tako im smanjiti veli~inu. U Linuxu se za komprimiranje datoteka koristi uslu`ni program gzip. Naredba gzip mo`e ASCII datoteke komprimirati vi{e od 4 puta. Ve}ina UNIX sustava nudi i naredbu compress, koja se obi~no koristi s naredbom tar za komprimiranje i arhiviranje grupe datoteka. Datoteke koje su komprimirane pomo}u naredbe compress zavr{avaju s .Z (primjerice, archive1.tar.Z).

Red Hat distribucija nudi i naredbe zip i unzip za komprimiranje i arhiviranje vi{e datoteka odjednom.

### SAVJET Dobro je komprimirati datoteke koje namjeravate slati poštom.

Ako je datoteka uspje{no komprimirana pomo}u naredbe gzip *imedatoteke*, onda }e njezino novo ime biti *imedatoteke*.gz, a originalna datoteka }e biti obrisana. Da biste obnovili komprimiranu datoteku, morate upotrijebiti naredbu gunzip *imedatoteke*.

BILJEŠKA Kada želite dekomprimirati datoteku, ne trebate upisivati nastavak .gz jer se taj dodatak imenu sam po sebi podrazumijeva.

Ako `elite datoteku zadr`ati u komprimiranom obliku, ali podatke u njoj namjeravate proslijediti drugoj naredbi, ona trebate upotrijebiti naredbu zcat.

Naredba zcat ~ini isto {to i naredba cat, ali kao izlaz daje komprimiranu datoteku. Naredba zcat dekomprimira datoteku i zatim njezin sadr`aj ispisuje na zaslonu.

```
Vidi "Povezivanje procesa", 355. stranica
```

Primjerice, ako ste komprimirali popis imena i adresa u datoteci pod imenom popis, onda komprimirana datoteka nosi ime popis.gz. Ako, kao ulazne podatke za neki program, `elite iskoristiti sadr`aj komprimirane datoteke, onda najprije upi{ite naredbu zcat, a zatim navedite `eljene programe:

zcat popis | program1 | program2

Naredba zcat podlije`e istim ograni~enjima kao i naredba cat: nema mogu}nost povratka na po~etak datoteke. Linux nudi i program pod nazivom zless koji radi na isti na~in kao i naredba less, s tom razlikom da se program zless koristiti za komprimirane datoteke. Iste naredbe koje se mogu koristiti uz naredbu less, dostupne su i pri radu s naredbom zless.

S Linuxom se distribuira besplatni uslu`ni program gzip koji slu`i za komprimiranje datoteka. Taj program ne nailazi na probleme s kojima se suo~ava program compress, tako da su sve instalacijske arhive Linuxa komprimirane pomo}u programa gzip. Ovaj se program mo`e koristiti za sve komprimirane datoteke, pa ~ak i one koje su komprimirane pomo}u starijeg programa compress.

Oni korisnici koji se radije slu`e PKZIP proizvodima mo}i }e koristiti naredbu zip i unzip koje se nalaze u sklopu Red Hat distribucije. Naredba zip komprimira vi{e datoteka i sprema ih u jedinstvenu arhivu (ba{ kao i naredba PKZIP). Naredba unzip se koristi za dekomprimiranje datoteka iz arhive. Na stranicama za pomo} (*man* stranice) potra`ite vi{e informacija o naredbama zip i unzip.

## Odavde...

Upravljanje datotekama i uslu`nim programima je u Linuxu relativno jednostavno. Organiziranje datoteka u direktorije je tako | er lagano. Isto tako, tra`enje, premje{tanje, kopiranje mijenjanje imena i brisanje datoteka i direktorija nije nikakav problem ako upotrijebite neku od naredbi find, mv, cp ili rm. Vi{e informacija potra`ite na sljede}im mjestima:

- U 14. poglavlju, "Upravljanje sustavom datoteka" na}i }ete upute za upravljanje sustavom datoteka.
- Stranice za pomo} (man stranice) naredbi koje su obja{njene u ovom poglavlju: ls, mkdir, mv, cp, rm, rmdir, cat, less, more, find, touch, gzip, compress, tar, zip i unzip.

# POGLAVLJE 18

# Što su Linux ljuske

(Steve Burnett)

## U ovom poglavlju

Prijava za rad Što su ljuske Rastavljanje naredbi u ljusci Procesi u pozadini Odziv naredbe Uređivanje naredbi u ljusci i uporaba zamjenskih naredbi Stvaranje skripti s naredbama Prilagodba ljuski u Linuxu Iako je grafičko sučelje dodano UNIX-u tek nedavno, većina se uslužnih programa koji se koriste u Linuxu (i drugim sličnim operativnim sustavima) pokreću tako da se u redak za zadavanje naredbi upiše ime programa. U Linuxu ljuska interpretira naredbe. U ovom poglavlju je opisano kako možete iskoristiti mogućnosti raznih ljuski za pokretanje Linux programa i rad sa sustavima datoteka.

# Prijava za rad

Kao novi korisnik i administrator sustava na svom Linux sustavu dobili ste korisničko ime i odabrali lozinku. S obzirom da je Linux višekorisnički operativni sustav, on mora razlikovati korisnike te ih svrstati u određene razrede. Linux upotrebljava vaše korisničko ime da bi nakon vaše prijave odredio koje privilegije smijete koristiti. Lozinka služi za autorizaciju vašeg rada.

Kako se teoretski bilo koji korisnik može prijaviti za rad na bilo kojem terminalu (iako postoje i iznimke), operativni sustav na svakom terminalu prikazuje prompt pod kojim se korisnik prijavio za rad. S obzirom da na početnom Linux sustavu možda nećete imati spojeno više terminala (iako je moguće spojiti više terminala na isti sustav), moći ćete koristiti zamjenske, virtualne terminale.

Za prelazak s jednog na drugi virtualni terminal, pritisnite tipku <Alt> zajedno s jednom od prvih šest funkcijskih tipki. Primjerice ako se na virtualni terminal broj jedan želite prijaviti kao superkorisnik, pritisnite tipku <Alt-F1> i ugledat ćete ovakav prompt:

```
Red Hat Linux release 5.1 (Manhattan)
Kernel 2.0.34 on an i686
login:
```

BILJEŠKA U prethodnom ste primjeru vidjeli da se koristi jezgra Linuxa s brojem inačice 2.0.34. S obzirom da se često pojavljuju nove inačice jezgre, možda ćete vidjeti i nešto veći broj inačice na svom računalu.

Upišite svoje korisničko ime (kao super korisnik) te lozinku.

Ako se prijavite na bilo koji terminal, moći ćete na njemu raditi sve dok se ne odjavite. Nakon što se odjavite, Login će postaviti prompt idućeg korisnika. Tijekom vašeg rada (odnosno u vremenu između prijave i odjave) Linux osigurava da možete koristiti sve programe i datoteke kojima ste vlasnik. Osim toga, sustav ne dozvoljava korištenje i čitanje tuđih datoteka, osim u slučaju kada ste za to dobili dozvolu samog korisnika ili administratora sustava. Vaše korisničko ime i lozinka služe sustavu za uspostavu sigurnosnog sustava koji štiti vaše i tuđe datoteke.

Kao administrator sustava dodjeljujete svakom korisniku korisničko ime, privremenu lozinku, identifikacijski broj grupe, početni direktorij i ljusku. Ti se podaci nalaze u datoteci pod imenom /etc/passwd, čiji je vlasnik administrator sustava (koji se naziva i root ili super korisnik). Nakon što se uspješno prijavite za rad moći ćete mijenjati svoju lozinku koja se unosi u skrivenom obliku tako da je nitko drugi ne može pročitati. Zaboravite li svoju lozinku, možete se prijaviti kao super korisnik i unijeti novu lozinku. Osim toga, možete mijenjati vlastitu lozinku uporabom naredbe passwd (ali morate upisati i staru lozinku).

BILJEŠKA Više informacija o osnovnim zadaćama administratora sustava, kao što su dodavanje korisnika i rješavanje problema koji nastaju ako se zaboravi lozinka, potražite u drugom dijelu knjige "Upravljanje sustavom", točnije u 10. poglavlju "Upravljanje korisničkim računima".

# Što su ljuske

Nakon prijave za rad Linux će vas postaviti u vaš početni direktorij i pokrenuti program zvani *ljuska*. Ljuska predstavlja samo program prilagođen za prihvat i izvođenje naredbi. Mnogi programi se mogu koristiti kao ljuske, no u gotovo svim inačicama Linuxa postoji nekoliko standardnih ljuski.

**BILJEŠKA** Ljuske u Linuxu su nalik datoteci COMMAND.COM koja se koristi u MS-DOS-u i služe za prihvat i izvođenje naredbi, pokretanje *batch* datoteka i izvršnih programa.

## Pregled različitih ljuski

Red Hat distribucija nudi sljedeće ljuske: sh, bash (Bourne Again SHell), tcsh, csh, pdksh (Public Domain Korn Shell), zsh, ash i ms. Isprobajte svaku ljusku i odaberite onu koja vam odgovara. U ovom poglavlju je naglašena uporaba ljuski sh i bash jer većina distribucija instalira ljusku bash kao osnovnu ljusku. Osim toga, ljuska sh se nalazi na većini UNIX računala i naći ćete mnoge skripte koje su napisane s naredbama ljuske sh.

S obzirom da ljuska služi kao primarna veza između operativnog sustava i korisnika mnogi korisnici identificiraju ljusku s Linuxom. Oni očekuju da se ljuska može programirati, ali ona nije dio jezgre operativnog sustava. Uz nešto znanja o sistemskom programiranju i poznavanja Linuxa možete pisati programe koji mogu predstavljati ljuske.

Iako su stvorene mnoge različite ljuske, postoji nekoliko najvažnijih: Bourne, C, T i Korn. Najstarija je ljuska Bourne, a druge ljuske imaju neke značajke koje ne postoje u ljusci Bourne. U osnovi Linux koristi inačicu ljuske Bourne, to jest ljusku bash, kao osnovnu ljusku (za početnike treba napomenuti da ljuske Bourne i Korn izgledaju istovjetno jer je ljuska Korn nastala od ljuske Bourne).

BILJEŠKA Distribucija Slackware 96 ne sadrži ljusku Korn, a distribucija Red Hat sadrži inačicu ljuske Korn koja se naziva pdksh (Public Domain Korn Shell).

Ljuska C je nastala na kalifornijskom sveučilištu i pogodnija je za programere od ljuske Bourne. Ljuska T je nastala od ljuske C. Ljuska Korn ima sve značajke ljuske C, ali koristi sintaksu koju upotrebljava ljuska Bourne. Ako vas sve to pomalo zbunjuje, ne brinite. Puno toga možete učiniti čak iako ne poznajete ljusku koju koristite.

U njihovim najjednostavnijim oblicima ljuske Bourne i Korn koriste znak dolara (\$) kao standardni prompt, dok ljuska C koristi znak postotka (%). Na svu sreću ti se znakovi mogu mijenjati tako da ćete možda vidjeti, a možda i ne znak dolara ili postotka kada se prvi puta prijavite za rad na računalu.

Ljuska Bourne je poznata pod imenom sh i predstavlja originalnu UNIX ljusku. Napisao ju je Steve Bourne uz pomoć Johna Masheya (iz tvrtke AT&T Bell Laboratories). Ta se ljuska koristi na svim Linux računalima. Izvršni program te ljuske je u datoteci /bin/sh. S obzirom da je ljuska Bourne dostupna na svim Linux računalima te da ima sve značajke opisane u prethodnim odlomcima, te velike mogućnosti za programiranje, ova ljuska je postala najkorištenija Linux ljuska.

#### BILJEŠKA Mnogi primjeri skripti koji su dati u ovom poglavlju su napisani za ljusku Bourne.

Skripte predstavljaju niz naredbi koje možete zapisati pomoću bilo kojeg programa za unos teksta, kao što je vi editor. Skripte možete zamisliti kao da su batch datoteke u DOS-u.

#### Vidi "Uvod u vi editor", 178. stranica

Ljuska C, poznata pod imenom csh, nastala je na kalifornijskom sveučilištu Berkeley, a razvio ju je Bill Joy. Studenti i profesori s ovog sveučilišta imaju veliki utjecaj na UNIX i njegovog nasljednika Linux. Dva rezultata njihova rada su ljuska C i program za unos teksta *vi*. Ljuska Bourne ima bolje mogućnosti za programiranje, ali ljuska C omogućuje veći utjecaj korisnika. Izvršni program ljuske C se nalazi u datoteci /bin/csh.

Sintaksa za korištenje ljuske C podsjeća na programski jezik C. To je jedan od razloga zbog čega skripte koje su pisane za ljusku C neće raditi pod ljuskom Bourne ili Korn (no programi koje ste kompilirali u ljusci C će raditi normalno) No, ljuska C ima nekoliko značajki koje nedostaju ljusci Bourne: mogu se uređivati naredbe, vraćati prethodno korištene naredbe i zamijeniti jednu naredbu drugom.

Osnovna Linux ljuska je bash koja se nalazi u datoteci /bin/bash i koja sadrži nekoliko značajki podrobnije opisanih u idućih nekoliko odlomaka, kao što su uređivanje naredbi, vraćanje prethodnih naredbi i dovršavanje unosa naredbe.

Sva Linux računala koriste ljusku bash. Možete instalirati i nekoliko drugih ljuski kao što su ljuske C ili T. Da biste odredili koju ljusku koristite, upišite sljedeće:

echo \$SHELL

Naredba echo ispisuje na zaslon sve ono što slijedi iza riječi echo. SHELL je varijabla koju je stvorila ljuska i koja sadrži ime trenutno aktivne ljuske, a izraz \$SHELL predstavlja vrijednost te varijable.

Da biste vidjeli je li vam na raspolaganju ljuska C, upišite naredbu:

csh

Ugledate li znak postotka (%) kao prompt, to znači da ste pokrenuli ljusku C (unesite exit da biste se vratili u prethodnu ljusku). Ako ste se prijavili kao superkorisnik, onda će prompt ljuske C izgledati ovako #. Ako ugledate poruku o grešci, to znači da vam nije na raspolaganju ljuska C.

Ljuska koja se koristi čim se prijavite za rad je zapisana u datoteci s lozinkom. Svaki identifikacijski broj sadrži zapis ili redak u datoteci s lozinkom, a zadnje polje u zapisu sadrži ljusku. Da biste promijenili ljusku u kojoj se prijavljujete za rad, morate promijeniti sadržaj toga polja. To je mijenjanje ljuske relativno lako izvodivo, ali prije nego promijenite ljusku dobro razmislite vrijedi li ponovno učiti nove oblike naredbi i metoda operativnog sustava. Detaljne informacije potražiti na stranicama *man* za svaku pojedinu ljusku.

#### **UPOZORENJE**

Nikada izravno ne mijenjajte datoteku s lozinkom (/etc/passwd). S obzirom na sigurnost sustava, tom datotekom biste trebali upravljati samo pomoću određenih naredbi. Primjerice, u distribuciji Slackware, da biste prešli na uporabu ljuske C upišite **usermod -s /bin/csh user**, gdje *user* predstavlja identifikacijski broj korisnika. Ovo upozorenje je izuzetno bitno ako koristite uslužne programe u sjeni.

Postoji još nekoliko ljuski, od kojih neke možete nabaviti putem Interneta ili iz nekih drugih izvora. Da biste odredili koju ljusku trebate koristiti, jednostavno pročitajte stranice za pomoć (stranice *man*) i isprobajte ljuske. S obzirom da su ljuske programi, možete ih pokretati kao i svaku drugu aplikaciju.

## Podesite okružje u kojem radite

Prije nego vidite prompt ljuske, Linux podešava okružje. Okružje u Linuxu sastoji se od podešenja i podataka koji upravljaju sustavom sve dok ste prijavljeni za rad. Naravno da imate mogućnost mijenjati osnovna podešenja i okružje prilagoditi svojim potrebama.

Okružje se sastoji od dviju komponenti:

- Prva komponenta se naziva okružje terminala i služi za upravljanje terminalom (točnije rečeno, priključkom na koji je spojen terminal).
- BILJEŠKA S obzirom da se Linux koristi na osobnim računalima, pod pojmom "terminal" se podrazumijevaju monitor i tipkovnica. Na računalo koje radi u Linuxu možete, ali i ne morate, imati spojen veći broj terminala. Također vam je na raspolaganju šest virtualnih

terminala na koje se možete prijaviti.

Druga komponenta se naziva okružje ljuske i služi za upravljanje ljuskom i svim programima koje koristite.

Najprije trebate ovladati okružjem terminala.

**Podešavanje okružja terminala** Rad u Linuxu se u osnovi sastoji od dva odvojena programa koji vam daju osjećaj da upravljate sustavom.

Ljuska je program koji prima vaše upute i izvršava ih, no prije nego ljuska primi naredbe sve što upišete mora najprije proći kroz program koji se naziva *pogonski program uređaja*.

Pogonski program upravlja vašim terminalom. On prima znakove koje upisujete i određuje što treba s njima učiniti prije nego ih proslijedi do ljuske. Slično tome, svaki znak koji stvori ljuska mora proći kroz pogonski program prije nego stigne do terminala. U ovom odlomku ćete najprije naučiti kako se upravlja pogonskim programom uređaja.

Linux je jedinstven po tome što svaki uređaj koji je spojen na računalo svim programima "izgleda" kao datoteka. Razni pogonski programi imaju zadaću prilagoditi uređaje da izgledaju poput datoteka. Tvrdi disk se ponaša bitno drukčije od terminala, ali je zadaća njihovih pogonskih programa da oni izgledaju isto za određenu aplikaciju.

Primjerice, disk ima blokove, sektore i cilindre koje treba pravilno adresirati kod čitanja i pisanja podataka. S druge strane, terminal prima neprekinuti niz znakova, ali oni po nekom redoslijedu trebaju biti isporučeni terminalu (i to relativno sporo). Pogonski program prima te podatke i šalje ih vama brzinom od 1200, 2400, 9600 bita u sekundi (ili više), a osim toga u niz znakova umeće bitove *stop, start* i *parity*.

S obzirom da je terminal neprestance spojen na računalo, pogonski program vam omogućuje definiranje posebnih znakova, to jest *upravljačkih znakova* koji služe za označavanje kraja retka i datoteke. Osim toga, pogonski program vam omogućuje

definiranje upravljačkih znakova za procese koji su u tijeku (prekidi u većini slučajeva mogu zaustaviti tekući proces i vratiti vas u ljusku). Slika 18.1 prikazuje na koji se način ponašaju jezgra, ljuska i pogonski program u Linuxu.



Možete podesiti desetke parametara svog terminala, no većina njih se ipak automatski podešava. No, ipak biste se trebali upoznati s nekoliko parametara i načina rada.

Pogonski program uređaja ima dva načina rada *kuhan* i *sirov*. U sirovom načinu rada svi znakovi koje upišete izravno prolaze do ljuske ili do programa koji je pokrenula ljuska. Programi za uređivanje teksta i tablični kalkulatori zahtijevaju sirovi način rada i automatski ga podešavaju. Nakon što takvi programi završe s radom, oni obično vraćaju *kuhani* način rada, ali ne i uvijek. Ako je vaš terminal u sirovom načinu rada, on neće odgovarati na upravljačke tipke.

No, ako je vaš terminal u tzv. *kuhanom* načinu rada, tada će svaku tipku koju pritisnete interpretirati pogonski program. Obične tipke se pohranjuju u spremniku sve dok ne pritisnete tipku koja označava kraj retka. U većini slučajeva je to tipka <Enter> ili <Return> (no te se tipke mogu i promijeniti). Nakon što pogonski program primi znak za kraj retka, on će interpretirati čitav prethodni redak i proslijediti ga ljusci ili izvršnom programu. U tablici 18.1 nalazi se popis najvažnijih upravljačkih tipki.

Tablica 18.1 Upravljačke tipke					
lme tipke	Opis				
Interrupt (prekid)	Prekida izvršenje programa. Ako u Linuxu zadate naredbu i pritisnete tipku za kraj retka, program će raditi do kraja. Ako pritisnete tipku za prekid, to znači da ste programu dali znak da stane. Neki programi takav znak zanemaruju. Ako je terminal u sirovom načinu rada, zahtjev za prekidom će biti izravno proslijeđen programu i možda neće imati željeni učinak. U UNIX-u se za prekide koristila tipka <del>, ali je u Linuxu to promijenjeno pa se koristi kombinacija tipki <ctrl-c>. Promjena je nastala zbog korisnika koji su radili u MS-DOS-u i drugim sustavima koji se koriste istom kombinacijom tipki.</ctrl-c></del>				
Erase (briši)	Briše zadnji znak u spremniku. Obično se koristi tipka Backspace. Princip je isti kao kod brisanja znaka u nekom tekstu. Na nekim terminalima i sustavima nastaje "nesporazum" između tipki <delete> i <backspace>.</backspace></delete>				
Kill (uništi)	Briše (uništava) sve što je u spremniku prije nego se sadržaj spremnika proslijedi ljusci ili aplikaciji. Ova je tipka obično definirana kao znak @. Dok ne pritisnete tipku za prekid, nećete vidjeti prompt nove ljuske iako ste pritisnuli tipku @, već će uređaj čekati da upišete novi tekst.				
End-of-line (kraj retka)	Upućuje pogonski program da ste završili s unosom teksta te da želite tekst interpretirati i proslijediti ga do ljuske ili aplikacije. Linux koristi tipku <enter> ili <return> kao oznaku kraja retka.</return></enter>				
End-of-file (kraj datoteke)	Upućuje ljusku na izlazak i prikaz prijavnog prompta. Ovaj znak se dobiva pritiskom na kombinaciju tipki <ctrl-d>. Linux sve uređaje tretira poput datoteka. S obzirom da je vaš terminal izvor neograničenog broja znakova, Linux koristi znak za kraj datoteke da bi vam naznačio da ste završili s radom i prijavom na sustav.</ctrl-d>				

Za podešavanje i prikaz tih upravljačkih tipki koristi se naredba stty (kratica od set teletype). U "davno prošlim vremenima" je postojao samo terminal teletype, tako da se taj izraz zadržao sve do danas. Stoga se i vaš terminal definira kao tty uređaj (koristi se ime tty14 ili slično). Da biste vidjeli sva podešenja, upišite **stty** –**a** u redak za zadavanje naredbi. Na zaslonu će se pojaviti nešto nalik ovome:

```
speed 38400 baud; rows 25; columns 80; line = 0;
intr = ^C; quit = ^\ ^; erase = ^?; kill = ^U; eof = ^D; eol = <undef>;
eol2 = <undef>; start = ^Q; stop = ^S; susp = ^Z; rprnt = ^R; werase = ^W;
lnext = ^V; flush = ^0; min = 1; time = 0;
-parenb -parodd cs8 hupcl -cstopb cread -clocal -crtscts
```

Ovaj fajl je skinut sa sajta www.tutoriali.org

```
-ignbrk -brkint -ignpar -parmrk -inpck -istrip -inlcr -igncr icrnl ixon ixoff
-iuclc -ixany -imaxbel
opost -olcuc -ocrnl onlcr -onocr -onlret -ofill -ofdel nl0 cr0 tab0 bs0 vt0 ff0
isig icanon iexten echo echoe echok -echonl -noflsh -xcase -tostop -echoprt
echoctl echoke
```

Uočite da je tipka za prekid definirana kao kombinacija tipki <Ctrl-c> (prikazana je kao ^C), a tipka za uništenje pomoću kombinacije <Ctrl-u>. Iako sami možete podešavati sve varijable koje su prethodno prikazane, zbog praktičnosti korisnici obično podešavaju tipke za prekid i za uništenje. Ako ovu drugu tipku želite promijeniti tako da se umjesto ^U koristi ^C, onda trebate upisati sljedeće:

stty kill '^C'

BILJEŠKA Ako se vaš terminal čudno ponaša, vratite "najnormalnija" podešenja tako da pokrenete naredbu stty sane.

SAVJET Ako želite da se određena podešenja koriste svaki puta kada se prijavite za rad, onda upišite naredbu u datoteku .profile koja se nalazi u vašem početnom direktoriju (ako koristite ljusku bash, Bourne ili Korn). Ako se služite ljuskom C, onda naredbu upišite u datoteku .login.

**Podešavanje okružja ljuske** Jedan dio prijave za rad predstavlja i stvaranje okružja u kojem se radi. Svi *procesi* u Linuxu (procesima se nazivaju aktivni programi) imaju vlastito okružje koje je odvojeno od samih programa. Može se reći da se program pokreće unutar svog okružja. Linux okružje se naziva i *okružje ljuske* i sastoji se od brojnih varijabli i njihovih vrijednosti. Te varijable i njihove vrijednosti omogućuju programu, primjerice nekoj ljusci, da se prilagodi izgledu okružja.

*Okružje* se odnosi na stvari kao što su ljuska koja se koristi, početni direktorij, vrsta terminala koji se upotrebljava i slično. Mnoge od tih varijabli se definiraju tijekom prijave za rad i nije ih moguće mijenjati. Ako varijabla nije označena atributom "read-only", moći ćete je mijenjati.

Varijable se zadaju u obliku *VARIJABLA= vrijednost*. Kao ime varijable možete koristiti bilo koji niz znakova. No mnoge varijable imaju definirana imena koja se koriste u mnogim Linux programima. Primjerice, varijabla TERM je definirana za određivanje vrste terminala. Korporacija Digital Equipement je godinama proizvodila poznati terminal VT-100, a značajke tog terminala su kopirali mnogi drugi proizvođači. Ime takvog terminala je vt100, pa za njegovo podešavanje u okružju trebate varijabli TERM pridijeliti vrijednost vt100 (TERM=vt100).

Postoje i mnoge druge varijable koje su unaprijed definirane. Ako koristite ljusku C, onda popis tih varijabli možete dobiti pomoću naredbe printenv, a ako se služite ljuskama Bourne ili Korn, onda morate pokrenuti naredbu set. U tablici 18.2 je dat popis varijabli koje se najčešće koriste. U stupcu varijabla nalazi se ime varijable koje trebate upisati u redak za zadavanje naredbi.

BILJEŠKA Neke sistemske varijable i varijable okružja se mogu mijenjati, a neke ne.

Varijabla	Opis				
HOME=/home/login	HOME podešava vaš početni direktorij, to jest s kojeg započinjete s radom. Zamijenite login s vašim korisničkim imenom. Primjerice, ako je vaše korisničko ime jack, onda varijablu HOME definirajte kao /home/jack.				
LOGNAME=1ogin	Varijabla LOGINAME se automatski podešava prema vašem korisničkom imenu.				
PATH= <i>path</i>	Opcija path predstavlja popis direktorija u kojima ljuska traži naredbe. Primjerice, stazu (path) možete i ovako podesiti: PATH=/usr:/bin:/usr/local/bin.				
PS1=prompt	PS1 je primarni prompt ljuske koji određuje kako će izgledati vaš prompt. Ako ne želite nikakvo posebno podešenje, onda će biti podešen znak dolara (\$). No ako želite, prompt možete promijeniti u nešto kreativnije. Primjerice, podešenje PS1="Upisi naredbu >" daje donekle zanimljiviji prompt.				
PWD= <i>direktorij</i>	Varijabla PWD se automatski podešava i definira mjesto u sustavu datoteka na kojem se nalazite. Primjerice, ako provjeravate varijablu PWD (tako da u redak za unos naredbi upišete echo \$PWD), a Linux prikaže /usr/bin, to znači da se nalazite u tom direktoriju. Naredba pwd prikazuje i direktorij u kojem se trenutno nalazite (tekući direktorij).				
SHELL=shell	Varijabla SHELL identificira mjesto na kojem se nalazi program koji služi kao ljuska. Primjerice, na ovakav način možete podesiti varijablu SHELL u datotekama .profile ili .login: SHELL=/bin/ksh i ljuska Korn će biti ljuska u kojoj se prijavljujete za rad.				
TERM= <i>termtype</i>	Varijabla služi za podešavanje vrste terminala na kojem radite. U datoteke .profile ili .login možete upisati ovakvo podešenje: TERM=vt100.				

Tablica 18.2 Varijable okružja koje se najčešće koriste u ljusci Bourne

BILJEŠKA Ako želite da se određena varijabla okružja definira prilikom svake vaše prijave za rad, onda njezinu definiciju smjestite u datoteku .profile (koja se nalazi u vašem početnom direktoriju) ako koristite ljusku bash ili Bourne. Ako upotrebljavate ljusku C, definiciju upišite u datoteku .login.

Vjerojatno je za okružje u kojem radite najbitnija varijabla PATH.

BILJEŠKA Korisnici DOS-a su već upoznati s varijablom PATH. Ona ima istu zadaću u Linuxu kao i u DOS-u.

Naredba PATH sadrži imena direktorija koji sadrže programe. Redoslijed po kojem su navedeni direktoriji određuje i redoslijed kod traženja programa. Taj je redoslijed bitan na

sustavima koji podržavaju nekoliko različitih oblika iste naredbe. U vašem sustavu možda postoje i naredbe koje ste sami stvorili i kojima želite povremeno pristupati. Primjerice, varijabla PATH može imati ovakav sadržaj:

#### /usr/ucb:/bin:/usr/bin:/usr/local/bin

Ljuska će najprije pretražiti direktorij /usr/ucb. Ako nađe željenu naredbu u tom direktoriju, zaustavlja se daljnje traženje i pokreće se naredba. Direktoriji /bin i /usr/bin sadrže sve standardne Linux naredbe. U direktoriju /usr/local/bin obično se nalaze lokalne naredbe koje ste dodali vi i drugi korisnici. Dodavanje lokalnih naredbi je obično zadaća administratora sustava.

Ako na sustavu radite kao administrator sustava ili želite pristupiti sistemskim naredbama, vjerojatno ćete u varijablu PATH dodati i direktorije /usr/sbin ili /usr/local/sbin

Ako vam je cilj načiniti vlastite naredbe, tada možete mijenjati varijablu PATH i dodati direktorije u kojima se nalaze vaše naredbe. Kako ćete to učiniti ovisi o ljusci koju koristite. Primjerice, ako koristite ljusku Bourne ili Korn, možete direktorij dodati u varijablu PATH ako upišete sljedeću naredbu:

\$ PATH=\$PATH:newpath

Ako ispred imena varijable dodate znak \$, onda će se koristiti trenutna vrijednost varijable. U ovoj naredbi \$PATH predstavlja tekuću vrijednost varijable PATH, a dvotočka i parametar *newpath* služe za dodavanje nove staze tekućoj.

U idućem odlomku je opisano još nekoliko načina upravljanja varijablama koje određuju okružje programa. Za sada je dovoljno reći da okružje ljuske sadrži varijable i funkcije te da se ti objekti koriste u radu s ljuskama i programima. Aplikacije mogu pristupiti okružju i mijenjati ga, no one općenito govoreći upravljaju varijablama unutar programa. Ljuske mogu upravljati jedino varijablama u okružju.

**Korištenje posebnih varijabli ljuske** Ljuska koristi vrijednosti brojnih posebnih varijabli. Te vrijednosti možete vidjeti ako pokrenete naredbu env. Slijedi primjer onoga što možete vidjeti ako upišete **env**:

HOME=/usr/wrev SHELL=/bin/sh MAIL=/usr/mail/wrev LOGNAME=wrev PATH=/bin:/usr/bin: TZ=PST8PDT PS1=\$ TERM=vt100

Bilo koju od tih posebnih varijabli možete upotrijebiti na isti način kao što koristite druge varijable. U tablici 18.3 se nalaze definicije posebnih varijabli.

Tablica 18.3	Posebne varijable koje definiraju okružje
Ime varijable	Značenje
HOME	Puna staza i ime vašeg početnog direktorija
SHELL	Ime ljuske koju trenutno koristite

Ime varijable	Značenje
MAIL	Puna staza i ime vašeg poštanskog sandučića
LOGNAME	Vaše korisničko ime
PATH	Direktoriji u kojima ljuska traži naredbe
ΤZ	Vremenska zona koju koristi naredba date
SECONDS	Broj sekundi potrebnih za pozivanje ljuske
PS1	Prompt sustava
TERM	Vrsta terminala koji koristite

Varijabla HOME Varijabla HOME uvijek označava vaš početni direktorij. Čim se prijavite, naći ćete se u početnom direktoriju. Ako želite, možete pokrenuti naredbu cd i promijeniti direktorij. Da biste se premjestili u direktorij /usr/local/games, trebate upisati cd /usr/local/games. Za povratak u početni direktorij dovoljno je unijeti samo naredbu cd. Kada pišete skripte s naredbama, možete upotrijebiti varijablu HOME prilikom odabira datoteka u početnom direktoriju. Umjesto da napišete naredbu grep \$number /usr/wrev/sales/data.01, bolje je upisati grep \$number \$HOME/sales/data.01 i to zbog sljedećih razloga:

- Naredbu je lakše razumjeti.
- Naredba radi iako premjestite početni direktorij.
- \$HOME uvijek predstavlja početni direktorij bez obzira tko koristi skriptu. Na taj će način i drugi korisnici moći pokrenuti istu naredbu.

Varijabla PATH Varijabla PATH sadrži direktorije u kojima ljuska traži naredbe. Ljuska pretražuje direktorije prema redoslijedu kojim su upisani. Ako varijabla glasi PATH=/bin:/usr/bin/:., svaki puta kada ljuska izvodi naredbu najprije je traži u direktoriju /bin. Ako se naredba ne pronađe u tom direktoriju, ljuska će je tražiti u direktoriju /usr/bin. Na kraju ljuska traži i direktorij (odnosno direktorij u kojem se trenutno nalazite). Unesete li naredbu cat da biste ispisali kalendar za ovaj mjesec, ljuska će najprije pretražiti direktorij /bin. S obzirom da se u njemu ne nalazi tražena naredba, ljuska će pregledati direktorij /usr/bin i naći naredbu.

SAVJET Ako imate i vlastitu naredbu pod imenom cal, ljuska je nikada neće naći jer će se uvijek izvoditi naredba cal iz direktorija /usr/bin. Svojim naredbama morate davati imena koja se razlikuju od sistemskih naredbi.

Možda ćete trebati staviti sve skripte u jedan direktorij i promijeniti varijablu PATH tako da sadrži i taj direktorij. Na taj ćete način moći izvoditi sve svoje skripte bez obzira u kojem se direktoriju trenutno nalazite. Evo kako to možete učiniti:

- 1. Stvorite direktorij u kojem ćete pohraniti skripte. Pomoću naredbe mkdir \$HOME/bin stvorite poddirektorij bin u svom početnom direktoriju.
- Premjestite sve skripte u taj direktorij. Da biste skriptu pod imenom stamp premjestili u poddirektorij bin, pokrenite naredbu mv stamp \$HOME/bin.

3. Dodajte poddirektorij u kojem se nalaze skripte u varijablu PATH pomoću naredbe PATH=\$PATH:\$HOME/bin. Da bi navedene promjene postale trajne, naredbu upišite u datoteku .profile.

Samo jednom trebate stvoriti novi direktorij bin i promijeniti varijablu PATH. U Linuxu takav direktorij već postoji i u njega možete spremati naredbe i skripte koje nisu dio standardnog Linux paketa, već ste ih sami dodali kako bi ih mogli upotrebljavati svi korisnici. U tom slučaju možete očekivati da je i direktorij /usr/local/bin sastavni dio varijable PATH.

**Varijabla MAIL** Varijabla MAIL sadrži ime datoteke koja sadrži vašu e-poštu. Svaki puta kada dobijete poštu, ona se sprema u datoteku koja je određena varijablom MAIL. Ako koristite program koji bilježi pristigle poruke, on uvijek provjerava datoteku određenu varijablom MAIL.

**Varijabla PS1** Varijabla PS1 sadrži niz znakova koji se pojavljuju na zaslonu kao vaš primarni prompt. Prompt je niz znakova koje ljuska prikazuje svaki puta kada očekuje da upišete naredbu. U odlomku "Prilagodba ljuski u Linuxu", koji se nalazi pri kraju ovog poglavlja, naći ćete upute za mijenjanje te varijable kao i mnogih drugih.

**Varijabla TERM** Varijabla TERM se koristi za identifikaciju vrste terminala na kojem radite. Programi koji se koriste na čitavom zaslonu, kao što je program za uređivanje teksta *vi*, trebaju podatak o vrsti terminala.

**Varijabla TZ** Varijabla TZ sadrži niz znakova koji označavaju vremensku zonu. Program date, ako i neki drugi programi koriste taj podatak.

Vaše računalo ima podešeno vrijeme prema meridijanu Greenwich. Ako je varijabla TZ podešena tako da sadrži vrijednost PST8PDT, onda su vrijeme i datum određeni prema standardnom pacifičkom vremenu (PST), osam sati zapadno od GMT-a, s potporom PDT-u (Pacific Daylight Savings Time). Računalo samo prelazi između standardnog vremena i pomaka radi "produženja/skraćenja dana".

**Varijabla LOGNAME** Varijabla LOGNAME sadrži vaše korisničko ime, to jest niz znakova prema kojima vas sustav identificira. Jedan od razloga zbog kojih se koristi varijabla LOGNAME je taj da vas sustav može prepoznati kao vlasnika određenih datoteka, te da vam može dopustiti pokretanje određenih programa, primanje pošte ili poruka koje se šalju naredbom Write.

Slijedi primjer skripte safrm koja služi za sigurno uklanjanje datoteka. Varijabla LOGNAME se koristi za uklanjanje svih datoteka iz direktorija /tmp, a čiji ste vi vlasnik. Za traženje datoteka se koristi naredba find. Ova naredba sadrži brojne opcije, a u skripti ćete naći sljedeće:

find /tmp -user \$LOGNAME -exec rm { } \;

Prvi parametar, /tmp određuje direktorij kojeg se traži. Opcija –user upućuje na traženje samo onih datoteka koje pripadaju određenom korisniku. Prije izvršavanja naredbe ljuska zamjenjuje izraz \$LOGNAME korisničkim imenom onog korisnika koji je pokrenuo skriptu. Opcija –exec označava da će se iduća naredba primijeniti na svaku datoteku koju pronađe naredba find. U ovom slučaju program rm se koristi za uklanjanje svih nađenih datoteka. Vitičaste zagrade predstavljaju položaj svakog imena datoteke koje se prosljeđuje naredbi rm. Zadnja dva znaka \;, koristi naredba find (radi se o primjeru za uporabu obrnute kose crte (\) koja služi da bi se proslijedio određeni znak programu, ali da ljuska ne interpretira taj znak). Opisana naredba je dodana u skriptu koju možete vidjeti u Ispisu 18.1, kako bi se dobio program koji uklanja datoteke i "čisti" sve što korisnik ima u direktoriju /tmp, a što je starije od 10 dana.

#### Ispis 18.1 Skripta safrm

```
# Name:
               safrm
# Purpose:
               copy files to directory /tmp, remove them
#
               from the current directory, clean up /tmp,
#
               and finally send mail to user
# first copy all parameters to /tmp
cp $* /tmp
# remove the files
rm $*
# create, a file to hold the mail message
#
       The file's name is set to msg
#
       followed by process ID number of this proccess
       For example, msg1208
#
msgfile=/tmp/msg$$
# construct mail message
date > $msgfile
echo "These files were deleted from /tmp" >>$msgfile
# get list of files to be deleted from tmp
# -mtime +10 gets all files that haven't been
# modified in 10 or more days, -print displays the names.
find /tmp -user $LOGNAME -mtime +10 -print >> $msgfile
# remove the appropriate files from /tmp
find /tmp -user $LOGNAME -mtime +10 -exec rm {} \ ;
# mail off the message
mail $LOGNAME < $msqfile</pre>
# clean up
rm $msqfile
```

## Što su procesi u Linuxu

Program koji radi u Linuxu se naziva *proces*. S obzirom da se Linux koristi kao višezadaćni operativni sustav, istovremeno se može odvijati više procesa. Da bi razlikovao procese, Linux svakom novom procesu dodjeljuje jedinstveni identifikacijski broj, koji se označava kao *proces ID*.

Taj broj je jedinstvena oznaka svakog pokrenutog programa. Pomoću naredbe ps možete vidjeti sve identifikacijske brojeve procesa koji su trenutno aktivni na vašem sustavu.

Da biste vidjeli većinu trenutno aktivnih procesa, pokrenite naredbu ps s parametrima –guax, i na zaslonu će se pojaviti ovo:

USER	PID	%CPU	%MEM	SIZE	RSS	ΤΤΥ	STAT	START	TIME	COMMAND	
jack	53	3.2	7.0	352	468	р 1	S	02:01	0:01	-bash	
jack	65	0.0	3.5	80	240	р 1	R	02:01	0:00	ps-guax	
root	1	Ø.8	3.1	44	2Ø8	con	S	Ø2:ØØ	Ø:ØØ	init	
root	6	0.0	1.8	24	124	con	S	02:00	0:00	bdflush (daemon)	
root	7	0.0	1.9	24	128	con	S	02:01	0:00	update (bdflush)	
root	40	1.0	3.5	65	240	con	S	02:01	0:00	usr/sbin/syslogd	
root	42	0.2	2.9	36	200	con	S	02:01	0:00	usr/sbin/klogd	
------	----	-----	-----	----	-----	-----	---	-------	------	--------------------	----
root	44	0.5	3.2	68	216	con	S	02:01	0:00	usr/sbin/inetd	
root	46	0.2	3.0	64	204	con	S	02:01	0:00	usr/sbin/lpd	
root	52	0.1	2.0	32	140	con	S	02:01	0:00	selection -t ms	
root	58	0.2	2.4	37	164	р б	S	02:01	0:00	/sbin/agetty 38400	tt

Identifikacijski broj procesa se nalazi u stupcu PID. Uočite redak koji je masno tiskan i koji predstavlja prvi proces koji je sustav pokrenuo – init. Ovaj proces je opisan i u nastavku ovog poglavlja.

Nakon što je Linuxu "rečeno" (naređeno) da pokrene program, odnosno da stvori proces, sustav će to učiniti tako da načini kopiju zadanog programa. Kod jednostavnijih slučajeva zahtjev za pokretanje programa upućujete ljusci, a ljuska Linux jezgri.

**Procesi fork, init i exec** Proces *fork* je kloniranje postojećeg programa. Linux stvara sve nove procese tako da klonira postojeće programe. Kod kloniranja procesa stvara se istovjetna kopija postojećeg procesa (uključujući i okružje programa te sve otvorene datoteke), jedinu razliku između tih dvaju procesa predstavlja zastavica koja određuje koji program je osnovni (*roditelj*), a koji kopija (*dijete*).

S obzirom da svi procesi nastaju na isti način, to znači da svi oni imaju osnovni proces i njegov identifikacijski broj. Svaki proces u Linuxu "vuče korijene" iz procesa init - to jest oca svih procesa. Proces init ima identifikacijski broj 1 i izravno ga pokreće jezgra Linuxa (redi se o jedinom programu kojeg pokreće jezgra, a s kojim vi kao korisnik imate doticaj). Svaki proces koji načinite tijekom rada koristi ljusku u kojoj ste se prijavili za rad.

Nakon što se određeni proces uspješno klonira, kopija procesa poziva rutinu exec koja je transformira u željeni proces. Jedina stvar koja se mijenja nakon toga je "identitet" procesa, okružje novog procesa je točna kopija okružja osnovnog procesa.

**Standardni ulaz i izlaz** Svaki novi proces stvara tri nove "datoteke". S obzirom da Linux na isti način tretira datoteke i uređaje, "datoteka" može biti stvarna datoteka na disku ili uređaj kao što je terminal. Otvaraju se tri datoteke: standardni ulaz (stdin), standardni izlaz (stdout) i standardni izlaz greške (stderr). Sve naredbe i programi u Linuxu prihvaćaju ulazne podatke iz standardnog ulaza te izlazne podatke šalju na standardni izlaz. Sve dijagnostičke poruke se automatski spremaju na standardni izlaz greške.

Kada se prvi put prijavite za rad, vašem terminalu se dodjeljuju ove tri standardne datoteke. Programi (odnosno procesi) "vide" terminal kao tri otvorene datoteke.

## Rastavljanje naredbi u ljusci

Redak za zadavanje naredbi (odnosno ono što u njega upisujete) možete rastaviti na komponente. U Linuxu takvo rastavljanje predstavlja puno više od jednostavne podjele retka za zadavanje naredbi. Niz znakova koji predstavljaju naredbu se dijeli na komponente: ime datoteke (uz koje možete koristiti i posebne znakove – *wild cards*), varijable ljuske, preusmjerivače ulaza i izlaza, podešenja podljuske i grupiranje naredbi te parametre koji mogu biti zamijenjeni drugom naredbom.

Tek nakon što sve to upišete, moći ćete pokrenuti naredbu.

Ako su vam novi izrazi posebni znakovi i preusmjeravanje ulaza i izlaza njihova ćete objašnjenja naći u nastavku ovog poglavlja. No, najprije se morate upoznati s osnovnim oblikom zadavanja naredbi.

## Uporaba naredbi, zastavica i parametara

Da biste pokrenuli neku Linux naredbu, obično trebate upisati i ime određene datoteke. Naredba kojom se ispisuju datoteke u direktoriju je naredba ls. datoteku s tim imenom možete naći u direktoriju /bin. Ako se u varijabli PATH nalazi izraz /bin (a taj bi se izraz i trebao tu nalaziti), ljuska će naći izvršiti naredbu /bin/ls.

Neke naredbe u Linuxu nisu nezavisne datoteke. One su ugrađene u same ljuske. Primjerice, naredba cd (promjena direktorija) je ugrađena u većinu ljuski i izravno se izvodi, a da ljuska pri tome ne traži nikakvu datoteku. Pročitajte upute o ljusci koju koristite i saznat ćete koje se naredbe izvode interno, a koje eksterno. Neke ljuske imaju datoteku s naredbama koje ljuska izravno pokreće.

**Zastavice** Da bi se neka naredba pravilno izvodila, morate je ljusci "predstaviti" na odgovarajući način. Ime se naredbe treba nalaziti na prvom mjestu, a tek potom slijede zastavice i parametri. Zastavica (koja se ponekad naziva i opcija) je označena jednim slovom ispred kojeg se nalazi crtica (-) i služi za promjenu načina rada naredbe.

Primjerice, naredba ls služi za jednostavan ispis imena datoteka u tekućem direktoriju i to prema abecednom redu. Dodavanjem raznih zastavica moći ćete sadržaj direktorija prikazati na više različitih načina. Imena datoteka i sve njihove atribute možete ispisati uporabom zastavice -l:

ls -1

Ako namjeravate koristiti više od jedne zastavice, jednostavno ih dopišite do prve zastavice, primjerice ls -lF. Zastavica -F služi za označavanje izvršnih datoteka, simboličkih veza i poddirektorija. Stranice koje sadrže pomoć za određene naredbe (stranice man) nude popis svih zastavica i njihovog utjecaja na naredbu. Zastavice možete unijeti i odvojeno jer će ih ljuska objediniti prije nego ih proslijedi programu. Primjerice, naredbu ls -lF možete napisati i kao ls -l-F.

Linux nudi mogućnost isticanja imena datoteka u boji. Ako pokrenete naredbu ls,

distribucija Slackware imena datoteka prikazuje u različitim bojama, ovisno o njihovom tipu. Na taj ćete način moći prema boji odrediti koja je datoteka izvršna, koja predstavlja direktorij, a

#### BILJEŠKA

koja vezu s drugim datotekama. Ako preusmjerite izlaz naredbe ls u datoteku, onda će ta datoteka sadržavati i upravljački kod koji se koristi za određivanje boja, što može izazvati probleme nekim programima, kao što su less. U Red Hat distribuciji postoji zastavica - -color, koja

se

koristi na isti način: ls - - color

Jedna vrsta zastavica označava da idući parametar ima posebno značenje. Primjerice, zastavica -t u naredbi sort se koristi da bi uputila sustav da je idući znak odjelnik polja. Ako namjeravate sortirati datoteku /etc/passwd, čija su polja odjeljena dvotočkom (:), tada trebate upisati:

sort -t: /etc/passwd

U slučaju naredbe sort, zastavica –t je nužna samo u slučajevima kada datoteka koristi odjelnik polja koji se razlikuje od osnovnog. Osnovni odjelnik polja je definiran u varijabli okružja IFS (Inter Field Separator). Ljuska koristi ovu varijablu za podjelu retka za zadavanje naredbe. Koristi se standardni odjelnik sve dok se to ne promijeni uporabom zastavice –t.

**Parametri** Zastavice moraju biti zadane prije bilo kojih drugih parametara. *Parametri* su znakovi odvojeni s nekim od znakova definiranim u varijabli IFS. U varijabli IFS su osnovno podešeni razmak, tabulator i oznaka novog retka. Između parametara možete postaviti bilo koliko odjelnika polja jer će ljuska ionako taj broj svesti na samo jedan znak. Primjerice, ako se iza naredbe nalaze tri razmaka, tabulator i tek zatim prvi parametar, ljuska će automatski obrisati tri razmaka i ostat će samo tabulator. Tako sljedeći redak

naredba <razmaknica> <razmaknica> <razmaknica> <Tab> parametar

postati

naredba <Tab> parametar

Parametri su obično imena datoteka koje upućuju naredbe da obave određenu zadaću. Ako parametar sadrži i umetnuti razmak, onda niz znakova trebate staviti u navodnike kako ga ljuska ne bi razdvojila. Iduća naredba sadrži dva parametra. Ljuska pokušava pronaći riječ New u datoteci York:

grep New York

Ako želite naći niz znakova "New York", onda naredbu morate ovako zapisati

grep "New York"

U tom slučaju će "New York" postati parametar naredbe grep.

## Zadavanje imena datoteke (programa)

Većina novijih operativnih sustava (uključujući sve inačice Linuxa i DOS-a) podržavaju uporabu posebnih znakova kod traženja datoteka i niza znakova. Tablica 18.4 sadrži popis posebnih znakova (*wild cards*) i njihova značenja.

Tablica 18.	4 Posebni znakovi za kompletiranje imena datoteke
Znak	Značenje
*	Predstavlja bilo kakav niz znakova osim točke ako je ona prvi znak u imenu datoteke. Primjerice, naredba cat sales* > allsales objedinjuje sve datoteke čija imena počinju riječju sales u datoteku allsales.
?	Predstavlja jedan znak. Primjerice, naredba lp sales.9? služi za ispis svih datoteka čija imena imaju oblik sales.yy, gdje yy označava godinu (jednu od 90-tih).

[ ] Predstavlja pojedini znak u nekom rasponu. Primjerice, ako upišete naredbu rm sales.9Š0-3Ć, obrisat ćete datoteke sales.90, sales.91, sales.92 i sales.93.

**BILJEŠKA** Ako posebne znakove upišete u navodnicima, onda će oni predstavljati ime datoteke. Primjerice, ako upišete ls \*, dobit ćete popis svih datoteka u tekućem direktoriju ali ako upišete ls "\*", vjerojatno će se pojaviti poruka da datoteka nije pronađena jer ste tražili datoteku s imenom \*.

**Znak** \* Zvjezdica (\*) je znak koji se najviše koristi od svih posebnih znakova. Ona jednostavno zamjenjuje sve znakove. Primjerice, oznaka a\* predstavlja sve datoteke koje počinju slovom a. U jednom izrazu možete upotrijebiti proizvoljan broj zvjezdica kako biste odabrali određene datoteke. Primjerice, izraz **xx**.gif znači da će biti odabrane sve datoteke koje imaju dodatak imenu .gif, te koje u imenu sadrže znakove xx. Po tom kriteriju se mogu odabrati datoteke abxx.gif, xxyyzz.gif i xx.gif.

Znak zvjezdice upotrebljavajte kao zamjenu za bilo koji niz znakova. Primjerice, da biste ispisali sve datoteke koje se nalaze u tekućem direktoriju i čija imena završavaju s .txt, upišite

lp \*.txt

Budite pažljivi kod uporabe zvjezdice kao posebnog znaka. Upišete li sljedeću naredbu, ispisat ćete sve datoteke čija imena završavaju znakovima txt:

lp \*txt

Datoteka čije je ime reportxt će biti odabrana drugom naredbom, ali ne i prethodnom. Upišete li iduću naredbu, ljuska će naredbi lp proslijediti imena svih datoteka u vašem direktoriju, uključujući i datoteku txt (ova datoteka će biti dva puta proslijeđena naredbi lp):

lp \* txt

U zadnjem primjeru naredba lp najprije ispisuje datoteke predstavljene zvjezdicom (odnosno ispisuje sve datoteke u direktoriju). Zatim naredba prelazi na drugi element u popisu (Linux razmak između \* i txt "shvaća" kao odjelnik, to jest kao zarez ili sličan znak). Na taj će način biti ispisana datoteka pod imenom txt.

Zvjezdica se može koristiti na bilo kojem mjestu u nizu znakova. Primjerice, ako želite upotrijebiti naredbu ls da biste dobili popis imena svih datoteka koje se nalaze u tekućem direktoriju i koja sadrže niz znakova rep, onda upišite ovu naredbu:

ls \*rep\*

U popisu ćete vidjeti datoteke kao što su frep.data, report i janrep. Postoji i jedna iznimka: ne ispisuju se datoteke koje počinju točkom. Da biste vidjeli popis datoteka koje počinju točkom (još se nazivaju *skrivene datoteke*), morate specificirati točku na početku imena. Primjerice, ako imate datoteku s imenom .reportrc, onda upišite ovakvu naredbu:

ls .\*rep\*

#### **UPOZORENJE**

Budite pažljivi prilikom uporabe zvjezdice tijekom brisanja datoteka. Napišete li naredbu rm \*, obrisat ćete sve datoteke u tekućem direktoriju. Jedna od čestih grešaka nastaje kada obrišete sve datoteke misleći da brišete samo datoteke s određenim nastavkom imena ili prefiksom u imenu. Ako umjesto da napišete naredbu rm \*txt (kojom želite ukloniti sve datoteke čije ime završava s txt), unesete naredbu rm \* txt, Linux će najprije obrisati sve datoteke, a zatim pokušati obrisati datoteku čije je ime txt, no do tada više neće ostati niti jedna datoteka.

Radi sigurnosti upotrijebite opciju –i . Naredba rm –i \*txt zahtjeva potvrdu za brisanje svake datoteke.

**Znak ?** Uporabom upitnika (?) kao posebnog znaka zamijenit ćete pojedini znak u imenu datoteke. Pretpostavimo da imate datoteke report1, reportb, report10, reportb3, report.dft i report.fin. Znate da ćete pomoću naredbe lp rep\* ispisati sve datoteke, ali da biste ispisali samo datoteke report1 i reportb, upišite sljedeće:

lp report?

Želite li dobiti popis datoteka čija imena imaju tri znaka i završavaju znakom x, onda unesite ovu naredbu:

ls ??x

Pomoću te naredbe naći ćete datoteku pod imenom tax, ali ne i trax.

S obzirom da upitnik zamjenjuje samo jedan znak u imenu, niz upitnika ??? predstavlja sve datoteke koje imaju ime od tri znaka. Možete dobiti popis datoteka koje imaju dodatak imenu od tri znaka, ako upišete oznaku \*.???. Primjerice, ako pretražujete direktorij koji sadrži grafike zajedno s drugim podacima, pomoću sljedeće naredbe ćete dobiti popis svih datoteka s dodatkom imenu .tif, .jpg i .gif, te svih drugih datoteka čiji dodaci imaju tri znaka:

ls \*.???

**BILJEŠKA** Upamtite da Linux nije MS-DOS., te da imena datoteka nizu ograničena na osam znakova, a dodaci imenima na tri znaka. Također imajte na umu da se u imenima datoteka razlikuju velika i mala slova.

**Uglate zagrade** [ ] Ponekad trebate biti određeniji nego što vam dopuštaju posebni znakovi (?, \*). Pretpostavimo da želite odabrati datoteke posao1, posao2 i posao3, ali ne i datoteku posaox. Prave datoteke ne možete odabrati ako upotrijebite znak upitnika jer on zamjenjuje sve moguće znakove (pa tako i znak x). No, ove datoteke možete odabrati tako da upišete job[123].

Jedan znak možete zamijeniti i tako da raspon dozvoljenih znakova upišete unutar uglatih zagrada. Da biste dobili popis datoteka koje počinju velikim slovom, upišite sljedeće:

ls [A-Z]\*

Pretpostavimo da imate datoteke s imenima sales.90, sales.91, sales.92 i sales.93 te da prve tri želite kopirati u poddirektorij oldstuff. Uz pretpostavku da direktorij oldstuff postoji, trebali biste upisati sljedeće:

cp sales.9[Ø-2] oldstuff

Kao i znak upitnika, niz znakova unutar ovakvih zagrada [] zamjenjuje samo jedan znak. Imate mogućnost opisati diskretni niz dopuštenih vrijednosti, primjerice [123], koje dozvoljavaju korištenje samo znakova 1, 2 i 3. Osim toga, možete zadati raspon znakova, primjerice [A-Z] predstavlja bilo koji znak između slova veliko, a i slova veliko Z.

Također možete zadati i nekoliko raspona odjednom. Primjerice, ako želite odabrati samo slova, možete upotrijebiti oznaku [A-Z,a-z]. U ASCII skupu znakova postoje i posebni znakovi između znaka veliko Z i malo a, tako da ako upišete oznaku [A-z] odabrat ćete i te posebne znakove.

## Povezivanje procesa

Povremeno ćete izlaz jednog programa htjeti upotrijebiti kao ulaz za drugi program. Umjesto da te dvije naredbe odvojeno pokrećete i rezultate spremate u privremene datoteke, možete spojiti više naredbi u slijed koristeći znak (¦).

Primjerice, ako želite sortirati datoteku pod imenom allsales, i zatim je ispisati, onda upišite sljedeće:

sort allsales ¦ lp

Izlaz programa s lijeve strane znaka (¦) se šalje programu koji se nalazi s desne strane i služi mu kao ulaz. Na sličan način možete povezati nekoliko procesa. Primjerice, da biste ispisali sortirane podatke iz datoteka čije ime počinje sa *sales*, upišite ovu naredbu:

cat sales\* | sort | lp

## Preusmjeravanje ulaza i izlaza

Mnogi programi "očekuju" ulaz s terminala ili tipkovnice, a isto tako mnogi programi šalju svoje izlazne podatke na zaslon. Linux ulazne podatke koje upisujete pomoću tipkovnice označava kao datoteku stdin, a izlazne podatke koje program ispisuje na zaslon operativni sustav označava kao datoteku stdout. Ulaz ili izlaz možete preusmjeriti tako da se umjesto zaslona i tipkovnice koriste izlazna ili ulazna datoteka.

Oznaku < (manje od) koristite da biste ulazne podatke naredbe (ili programa) preusmjerili tako da dolaze iz datoteke, a ne s terminala. Pretpostavimo da datoteku pod imenom info želite poslati putem e-pošte osobi čija je adresa sarah. Umjesto da sadržaj poruke prepisujete iz datoteke, isko-ristite datoteku info kao ulaznu datoteku (stdin) za naredbu mail:

mail sarah < info</pre>

Ako upotrijebite znak > (veće od), preusmjerit ćete izlaz programa s zaslona u datoteku. Naredba date na zaslonu prikazuje trenutno vrijeme i datum. Želite li te podatke pohraniti u datoteku pod imenom now, unesite ovu naredbu:

date > now

#### **UPOZORENJE**

Ako datoteka, koja se nalazi desno od znaka >, već postoji, njezini će podaci biti prebrisani novim sadržajem. Pazite da na taj način ne uništite bitne datoteke.

Želite li podatke dodati u postojeću datoteku, onda upotrijebite ovaj znak >>. Da biste trenutno vrijeme i datum dodali u datoteku pod imenom report, upišite sljedeće:

date >> report

U nešto dužem primjeru smo pretpostavili da se u datoteci sales nalaze podaci o prodaji. Prvo polje u svakom retku sadržava identifikacijski broj kupca. Prva naredba sprema u datoteku sales\_report izlaz naredbe date. Druga naredba sortira datoteku sales i dodaje izlazne podatke naredbe sort u datoteku sales\_report. Zadnja naredba šalje datoteku sales\_report korisnicima sarah i brad kao e-poštu:

```
date > sales_report
sort < sales >> sales_report
mail sarah and brad < sales_report</pre>
```

#### **UPOZORENJE**

Budite pažljivi kako istu datoteku ne biste odredili za ulaz i izlaz pojedine naredbe. Vrlo je vjerojatno da ćete u tom slučaju uništiti njezin sadržaj.

Tablica 18	Tablica 18.5         Oznake za preusmjeravanje ulaza i izlaza				
Oznaka	Značenje	Primjer			
<	Uzimanje podataka iz ulazne datoteke	mail sarah < report			
>	Slanje izlaznih podataka u datoteku	date > now			
>>	Dodavanje podataka u datoteku	date >> report			

Tablica 18.5 prikazuje oznake koje se koriste za preusmjeravanje ulaza i izlaza.

## Zamjena varijabli ljuske

U početku ovog poglavlja ste naučili kako možete proširiti varijable ljuski, primjerice da varijablu PATH proširite na PATH=\$PATH:*newpath*. Ljuska zamjenjuje prijašnju s trenutnom vrijednosti varijable PATH. Ljuske su doista programi koji interpretiraju naredbe, kao BASIC, a varijable predstavljaju objekte kojima naredbe upravljaju. S obzirom da se varijable ljuski često koriste, svaka ljuska nudi određene metode za provjeru i definiranje sadržaja pojedine varijable.

Varijable ljuski su pohranjene u obliku *string*. Ako se dvije varijable nalaze jedna do druge, njihovi se sadržaji ujedinjuju. Primjerice, imate li dvije varijable X=hello i Y=world, onda će izraz koji je označen kao \$X\$Y izgledati ovako:helloworld. Ako upišete iduću naredbu, ljuska će razdijeliti parametre i varijable X i Y će biti razdvojene ako ih unesete u naredbu echo:

echo \$X \$Y

Na zaslonu će se pojaviti ispis: hello world.

BILJEŠKA Pritisnete li i desetak puta tipku Tab između oznaka varijabli, rezultat će i dalje biti isti.

Ako je zamjena varijabli dvosmislena, ljuska će odabrati jednu od mogućnosti, no rezultat takve naredbe može biti nepredvidiv. Primjerice, upišete li naredbu **echo \$XY**, dobit ćete izlaz helloY. Da bi se izbjegli mogući nesporazumi, postoji jednostavan način za upisivanje varijabli. Upišete li **\${X}Y**, ljuska će zamijeniti vrijednost varijable X, a tek potom dodati znak Y.

Ljuske Bourne i Korn sadrže bogatu zbirku metoda za unos varijabli ljuske, a svaka od tih metoda obavlja potrebne provjere prije zamjene varijabli. Više detalja potražite u stranicama za pomoć (man stranice) sh i ksh.

## Zamjena rezultata naredbi

Nakon što ljuska obavi zamjenu varijabli, ona će ponovno pregledati redak u kojem se nalazi naredba te provjeriti nalazi li se u njemu još neka naredba koju treba obaviti prije glavne naredbe. *Zamjena rezultata naredbi* znači da Linux rezultate "pomoćne" naredbe pretvara u parametar "glavne" naredbe. Taj se postupak može i ovako napisati:

naredba - 1 parametar 'naredba - 2'

Budite pažljivi kod uporabe navodnika (') da ih ne biste zamijenili s drugim sličnim znakovima. U tablici 18.6 naći ćete objašnjenje tih znakova.

Tablica 18.6	Oznake koje se koriste kao navodnici
Oznaka	Značenje
"	Dvostruki navodnici onemogućuju stvaranje imena datoteke i sprečavaju širenje parametra, no omogućuju zamjenu varijabli ljuske i zamjenu rezultata naredbi.
د	Apostrofi (jednostruki navodnici) onemogućuju bilo kakvo dijeljenje naredbe. Sve što se nalazi unutar njih predstavlja jedinstveni parametar.
`	Ovaj znak predstavlja zamjenu naredbi. Sve što se nalazi unutar ovakvih navodnika se izvodi kao samostalna naredba. Svi izlazni podaci koje ste poslali na standardni izlaz će zamijeniti naredbu unutar navodnika.

Pretpostavimo da ste upisali sljedeću naredbu:

echo Today  $\setminus$  's date and time are 'date'

Izlaz te naredbe će biti sljedeći:

Today 's date and time are Mon May 18 14.13.09 EST 1994

Da bi se dobio željeni izlaz naredbe echo, upisana je obrnuta kosa crta (\). Gotovo svaki znak (koji ne predstavlja brojku ili slovo) ima određeno značenje za naredbe u ljusci. Da biste neke od posebnih znakova zadali kao običan tekst, morate prije željenog znaka upisati oznaku obrnute kose crte (\), a ako želite ispisati i samu obrnutu kosu crtu, tada je upišite dva puta (\\). Da biste na zaslonu vidjeli znak dolara, trebate upisati \\$.

## Obični izrazi

Obični izraz je niz koji se sastoji od standardnih znakova i posebnih operatora. Obični izrazi su korisni kod traženje određenog skupa znakova u datoteci i obično se koriste s grep alatima: grep, egrep i fgrep, ali i s drugim naredbama u UNIX-u.

Najjednostavnija vrsta običnih izraza su nizovi znakova (string). Naredba grep se zadaje na ovakav način:

grep string imedatoteke

Da biste u datoteci s imenom michael našli riječ hand (ruka), potrebno je upisati ovakvu naredbu:

```
grep hand michael.txt
```

a na zaslonu će se pojaviti sljedeće:

on the other hand, michael has been working hard this past

To je jedini redak u tekstu koji sadrži riječ hand (ruka). Alat grep bi na zaslon ispisao sve retke u kojima se nalazi traženi skup znakova.

Obični izrazi koriste i posebne znakove kao što su točka (.), zvjezdica (\*), uglate zagrade ([]), kosa crta (/), znak (^) i oznaka dolara (&). U tablici 18.7 ćete naći opis tih posebnih znakova i njihovo ponašanje kada se nađu u običnim izrazima.

Tablica 18.7	7 Posebni znakovi u običnim izrazima
Znak	Opis
	Zamjenjuje jedan znak, osim ako to nije znak za povratak na početak retka. Primjerice, b.d može predstavljati riječi bad i bod.
*	Koristi se za traženje bilo kojeg znaka iz običnog izraza (koji je se nalazi ispred zvjezdice).
[]	Koristi se za grupiranje kriterija za traženje. Upamtite da za razliku od DOS-a UNIX je osjetljiv na velika i mala slova. Stoga, da biste našli ime Michael, trebate upisati [Mm]ichael i naći ćete riječi Michael i michael, ali ne i riječ MICHAEL. Ako želite tražiti baš lijevu ili desnu uglatu zagradu,

Znak	Opis
	možete napisati [ [], ili kosom crtom odvojiti traženi znak, primjerice ovako: /[. Crtica unutar zagrada predstavlja raspon tako da je izraz [a-j] identičan izrazu [abcdefghij].
/	Koristi se da bi se izbjeglo "posebno ponašanje" posebnih znakova te da bi se te znakove shvaćalo kao tekst koji treba tražiti. Znak * se koristi za traženje svega, a izraz /* daje samo retke u kojima se nalazi zvjezdica. Dvije kose crte (//) se koriste za traženje redaka s s kosom crtom.
٨	Ako se znak ^ nalazi ispred određenog izraza, onda će sustav tražiti samo one retke u kojima se izraz nalazi na samom početku. Imate li datoteku s telefonskim brojevima koji su sortirani prema pozivnom broju, možete koristiti ovaj znak i izdvojiti sve telefonske brojeve iz određenog grada.
\$	Znak \$ predstavlja zadnji znak u običnom izrazu.

Koristeći vitičaste zagrade sami možete odrediti slijed od nekoliko istih znakova koji se traže u tekstu. Primjerice, naredba

g \ {3,4}

traži sve retke u tekstu koji sadrže znakove ggg ili gggg.

Ako uređujete veliku datoteku sa starom poštom, tada će naredba

```
grep 'whatever' ~/mail/*
```

tražiti izraz *whatever* u direktoriju za poštu. Ovo može biti korisno ako tražite neku poruku, ali niste sigurni u koju ste je mapu spremili. Naredbom

```
grep 'Quigman' ~/mail/*
```

pronaći ćete sva mjesta na kojima se nalazi određeno prezime. Pretpostavimo da ta osoba ima pozivni telefonski broj 408. Pomoću naredbe

```
grep '408. [0-9]\{3\}.[0-9]\{4\}' ~/mail/*
```

će tražiti sve telefonske brojeve koji počinju s 408. Uočite da točka razdvaja izraze  $[0-9] \setminus \{3\}$  i $[0-9] \setminus \{4\}$ . Ta točka zamjenjuje bilo koji znak tako da možete naći telefonske brojeve koji su zapisani u obliku 408-555-1212 ili 408.555.1212, jer neki ljudi brojeve odvajaju točkama.

## Grupe naredbi, podljuske i druge naredbe

Na kraju pojedine naredbe treba pritisnuti tipku <Return>. Ako u redak za zadavanje naredbi želite unijeti više naredbi odjednom, tada ih trebate odvojiti s točka-zarezom (;). Na taj se način stvaraju grupe naredbi. Ljuska će oznaku točka-zarez "shvatiti" kao oznaku kraja retka. Upišete li naredbe na ovakav način, one će se izvoditi jedna za drugom kao da ste svaku od njih upisali u poseban redak:

```
naredba-1; naredba-2; naredba-3
```

Primjerice, možete upisati clear; ls i osvježiti zaslon te na njemu ispisati sadržaj direktorija.

**Grupe naredbi** Ako želite istovremeno preusmjeriti ulaz ili izlaz više naredbi, to možete učiniti ako upišete grupu naredbi u redak za zadavanje naredbi. *Grupa naredbi* se definira tako da se proizvoljan broj naredbi upiše unutar vitičastih zagrada ({ }). Primjerice, pomoću iduće naredbe ćete preusmjeriti izlaz dviju naredbi u datoteku pod imenom *izlazna*:

{naredba-1; naredba-2} > izlazna

Također možete koristiti bilo koji oblik naredbe za preusmjeravanje. Izlaz grupe naredbi možete proslijediti trećoj naredbi:

{naredba-1; naredba-2} / naredba-3

### BILJEŠKA

Naredbe iz iste grupe naredbi se pokreću u tekućoj ljusci. To znači da one mogu mijenjati okružje ili direktorije.

**Podljuske** Ako niz naredbi pokrećete kao grupu naredbi, one će biti pokrenute u tekućoj ljusci. Ako jedna od naredbi mijenja okružje ili direktorij, te će promjene ostati i nakon što grupa naredbi završi svoju zadaću. Da biste izbjegli taj problem, grupu naredbi pokrenite u *podljusci*.

Podljuska predstavlja kloniranu ljusku, ali s obzirom da kopije procesa ne mogu mijenjati okružje osnovnog procesa, sve naredbe koje se pokrenu u podljusci neće imati utjecaja na okružje nakon što naredbe obave svoje zadaće. Da biste grupu naredbi pokrenuli u podljusci, zamijenite vitičaste zagrade običnim zagradama. Tako da će prethodni primjer izgledati ovako:

(naredba-1; naredba-2) | naredba-3

Samo se naredba-3 izvršava u tekućoj ljusci dok se izlaz prvih dviju naredbi koristi kao standardni ulaz naredbe-3.

## Procesi u pozadini

S obzirom da je Linux višezadaćni operativni sustav, na nekoliko načina možete pokrenuti naredbe u pozadini. Najjednostavniji oblik pozadinskih procesa omogućuje vam istovremeno pokretanje više naredbi. Sve druge metode "guraju" naredbe sve dublje i dublje u pozadinu glavnog programa.

## Priprema procesa za rad u pozadini

Ljuska vam omogućuje da pokrenete neki proces, i da, prije nego on završi, pokrenete drugi. Ako to učinite, stavit ćete prvi proces u pozadinu. Da bi se neki proces odvijao u pozadini, morate na kraju retka za unos naredbi upisati znak &. Pogledajte ovu naredbu:

```
sort sales > sales.sorted &
```

Upišete li ovakvu naredbu, na zaslonu ćete jednu brojku koja predstavlja identifikacijski broj procesa koji ste stavili u pozadinu. Radi se o načinu na koji sustav označava procese.

Obično kada pokrenete neku naredbu ljuska prestaje s radom dok se naredba ne izvrši. Ako na kraju retka dodate znak &, onda se naredba izvodi zajedno s ljuskom. Postavljanjem znaka & iza neke naredbe, ljuska nastavlja s radom sve dok je aktivna

naredba u pozadini. Naredba u pozadini i ljuska očekuju ulazne podatke i "isporučuju" izlazne podatke putem terminala, osim u slučaju kada ste preusmjerili ulaz/izlaz naredbe u pozadini. Ako naredba u pozadini sama "ne vodi brigu" o ulazu i izlazu podataka, oblik zadavanja naredbe u pozadini će ovako izgledati:

naredba-string [ulazna-datoteka] izlazna- datoteka &

Primjerice, da biste kopirali više datoteka čija imena završavaju s .txt u poddirektorij oldstuff, a istovremeno ispisali sortirani popis podataka iz svih datoteka koje počinju sa sales, morate upotrijebiti ove naredbe:

```
cp *.txt oldstuff &
cat sales* | sort | lp
```

SAVJET Posao stavite u pozadinu u slučaju da ne želite čekati njegovo izvršenje da biste pokrenuli drugi proces. Programe možete postaviti u pozadinu i u slučaju da imate više zadaća od kojih se barem jedna može neovisno izvršavati. Pokrenite upravo takav program i postavite ga u pozadinu.

Također možete koristiti i virtualni terminal za izvođenje neke naredbe te se istovremeno prijaviti za rad na drugom terminalu.

S obzirom da je proces u pozadini kopija vaše ljuske, on se automatski prekida nakon vaše odjave sa sustava. Sve kopije procesa (djeca) "umiru" ako "umru" njihovi originali (roditelji).

## Uporaba naredbe nohup

Da biste naredbu "gurnuli" dublje u pozadinu nego što to omogućuje operator &, trebate upotrijebiti naredbu nohup (kratica od *no hang up*). Naredba nohup koristi imena naredbi kao svoje argumente. No uz ovu se naredbu trebaju koristiti operatori & ako želite da programi doista rade u pozadini. Ako ste neku naredbu pokrenuli pomoću naredbe nohup, ta se naredba neće prekinuti niti nakon što prekinete vezu (njezina osnovna namjena). Evo kako izgleda sintaksa naredbe nohup:

nohup command-string [input-file] output-file &

## Uporaba demona cron

Ako naredbu pokrećete pomoću naredbe nohup, ona će se odmah izvršiti. Ako naredbu želite naknadno pokrenuti, morat ćete zatražiti pomoć *daemona* cron.

Program *cron* se pokreće u pozadini, ili točnije, pokreće ga init, glavni program Linuxa. Program *cron* nudi raspored izvođenja svih procesa u Linuxu. Pomoću programa *cron* možete odrediti točno vrijeme pokretanja nekog drugom programa, zadati izvršenje programa u pravilnim vremenskim razmacima i slično.

Vidi "Raspored izvršavanja naredbi", 388. stranica.

**Naredba at** Naredba at očekuje da unesete vrijeme ili datum kao parametar te prima proizvoljan broj naredbi kao vlastiti standardni ulaz. Kada naredba at naiđe na oznaku kraja datoteke, ona stvara skriptu koja se izvodi u zadano vrijeme.

Naredba je vrlo fleksibilna glede načina unosa datuma i vremena. Primjerice, ako unesete naredbu **at now +1 day**, idući puta će se naredba pokrenuti sutra u isto vrijeme. Jedan od načina uporabe ove naredbe je pomoću skripte.

*Skripta* predstavlja datoteku koja sadrži sve naredbe koje su potrebne da bi se obavila određena zadaća. Ime datoteke potom postaje vaš dodatak Linuxu. Slijedi jedan primjer za uporabu naredbe at:

```
at now +1 day
naredba-1
naredba-2
```

Ako ove naredbe upišete u odgovarajuću skriptu, one će se izvršavati i idućeg dana. Više različitih naredbi možete pokrenuti ako ih jednostavno upišete nakon retka u kojem se nalazi naredba at (njihov broj nije ograničen).

**Naredba batch** Naredba batch je ekvivalent naredbi at now. Ako pokušate pokrenuti naredbu at now, ugledat ćete poruku o grešci. Naredba batch radi kao naredba at now, ako je to logički moguće, s jednom malom iznimkom: Daemon *cron* stvara odvojene upite za naredbe koje nastaju uz pomoć naredbi at, batch i cron. Pretpostavimo da ste u datoteku pod imenom backup unijeli sljedeće naredbe:

tar -cvf tackettbkup /usr/home/tackett

Nakon toga možete načiniti sigurnosnu kopiju sustava /usr/home/tackett:

batch backup

Vidi "Stvaranje prve datoteke u vi editoru", 182. stranica

**Naredba crontab** Jedan od najboljih načina korištenja programa cron je automatsko stvaranje sustava. Pomoću *crona* vi kao administrator sustava možete postaviti automatsko stvaranje sigurnosne kopije sustava svakog jutra u 4 sata i to od ponedjeljka do subote. Naredbe koje želite koristiti u tom postupku instalirate, brišete i ispisujete pomoću naredbe crontab.

Da biste u vremenskim razmacima pokretali naredbe, morate stvoriti datoteku u formatu crontab, odnosno datoteka se mora sastojati od šest polja odvojenih razmacima ili tabulatorima. Prvih pet polja su integer varijable koje određuju minute (00-59), sate (00-23), dane u mjesecu (01-31), mjesece u godini (01-12) i dane u tjednu (0-6, gdje se s nulom označava subota). U šestom polju se nalazi naredba. U svakom brojčanom polju se može nalaziti i raspon brojeva (kao što je raspon 1-5 koji označava radne dane u tjednu), ili diskretni niz brojeva (kao što su brojevi 0,20,40 koje upućuju da se pojedina naredba obavlja svakih dvadeset minuta). Polje može sadržavati i zvjezdicu \* koja zamjenjuje sve moguće vrijednosti polja.

U idućem primjeru je prikazano kako se naredba calendar pokreće svakih dvadeset minuta, počev od ponedjeljka u ponoć do petka u 11:40 navečer:

0,20,40 \* \* \* 1-5 calendar -

Ako datoteci date ime cronfile, moći ćete je instalirati u program cron tako da pokrenete naredbu crontab cronfile.

Daemon cron vremenske intervale dijeli na minute, što znači da najkraće trajanje intervala, koje možete upotrijebiti, iznosi jednu minutu. Kao administrator sustava možete ograničiti broj naredbi koje možete istovremeno pokretati. To što ste od programa cron tražili da pokrene neku at, batch ili crontab datoteku, ne znači da će ih on pokrenuti u precizno naznačeno vrijeme.

## Odziv naredbe

Linux nudi trenutni odziv za naredbe čiji rad je prekinut iz bilo kojeg razloga. U većini slučajeva greške se događaju zbog krivo upisane naredbe ili krivo formatiranog imena datoteke. Ako pokušate pokrenuti nepostojeću naredbu, Linux će ispisati poruku:

command: command not found (naredba nije nalena)

Ako pokušate koristiti nepostojeću datoteku, ugledat ćete sljedeće:

command: file: No such file or directory

Ako je greška nastala uslijed nekog drugog razloga, vrlo je vjerojatno da ćete na zaslonu dobiti informaciju što je uzrok greške (iako to ne mora biti uvijek razumljivo napisano).

Ako pokušate pokrenuti neku naredbu pomoću naredbe nohup, Linux će automatski poruke o grešci spremati u datoteku pod imenom nohup.out (datoteka se nalazi u direktoriju iz kojeg je naredba pokrenuta).

S obzirom da naredbe koje pokreće cron imaju nižu razinu hitnosti, bilo koje poruke o greškama, u stvari kompletan standardni izlaz iz programa, biti će vam poslan putem e-pošte.

## Uređivanje naredbi u ljusci i uporaba zamjenskih naredbi

Različite ljuske sadrže određene značajke koje omogućuju stvaranje prečica za pokretanje naredbi (programa). *Uređivanje naredbi* omogućuje vam promjenu naredbi koje su prethodno upisane. Upotrijebite li Linuxovu značajku *command history*, moći ćete ponovno pozvati već korištene naredbe. Značajka *aliasing* omogućuje vam stvaranje naredbi koje su zamjena za druge naredbe. *Command completion* omogućuje dopunjavanje ostatka imena datoteke nakon što ste unijeli njegov prvi dio.

## Uređivanje naredbi

Uređivanje naredbi u osnovi znači da nakon što upišete naredbu, te prije nego pritisnete tipku <Return> imate mogućnost promijeniti dio naredbe bez ponovnog upisivanja čitave naredbe. Da biste uređivali naredbu, pritisnite tipku <Esc> i prijeći ćete u uređivački način rada, a zatim koristeći naredbe za kretanje iz *vi* editora, moći ćete promijeniti bilo koji dio naredbe. Tipku <BackSpace> možete koristiti za povratak na dio naredbe koji namjeravate promijeniti, a zatim pomoću tipke x obrisati znak ili ga zamijeniti drugim znakom pomoću naredbe r.

## Pregled prethodno korištenih naredbi

Linux koristi značajku koja omogućuje pregled prethodno unešenih naredbi i njihovo ponovno pozivanje. Na taj ćete način uštediti vrijeme i izbjeći greške koje mogu nastati kod ponovnog upisa naredbe. Objedinite li ovu značajku s onom za uređivanje naredbi, moći ćete lako otkloniti greške u složenijim naredbama te ih ponovno pokrenuti.

U obje ljuske naredba history prikazuje popis prethodno izvršenih naredbi. Naredbe su označene brojevima. Da biste pokrenuli naredbu 10 upišite **! 10**. Ljuska bash može koristiti i tipke za kretanje (sa strelicama) tako da prethodnu naredbu možete pozvati pritiskom na strelicu prema gore.

## Zamjenske naredbe

Zamjenske naredbe vam omogućuju da sami date ime naredbi. Zamislite ovaj primjer: Naredba man prikazuje dokumentaciju Linuxa odnosno stranice *man*. Želite li riječ help koristiti kao zamjenu za pokretanje naredbe man, onda trebate upisati sljedeće:

alias help=man

Nakon toga možete upisati help cp ili man cp i dobit ćete pomoć za naredbu cp.

Zamjenske naredbe možete koristiti i kod naredbi koje imaju opcije ili argumente. Primjerice, ako želite dobiti popis svih datoteka u tekućem direktoriju sortiranih prema vremenu kada su zadnji puta promijenjene (tako da se najnovije datoteke nalaze u dnu popisa), onda trebate upotrijebiti ovu naredbu:

#### ls -art

Naredba ls se koristi za ispis datoteka, opcija –a označava ispis svih datoteka, opcija –r služi za obrnuti redoslijed ispisa, a opcija –t označava sortiranje prema vremenu zadnje promjene u sadržaju datoteke. Previše je tih opcija da biste ih sve pamtili. Tu složenu naredbu možete zamijeniti naredbom time dir, i to tako da upišete:

alias timedir="ls -art"

Navodnici su nužni jer ljuska "očekuje" da iza zamjenske naredbe timedir bude razmak ili <Return>. Ako nakon svega upišete timedir, dobit ćete željeni ispis sadržaja direktorija.

**BILJEŠKA** Podesite li zamjensku naredbu u retku za zadavanje naredbi, ona će vrijediti samo do kraja rada u Linuxu. Da biste trajno mogli koristiti zamjenske naredbe, upišite njihove definicije u datoteku .profile (ako koristite ljusku Bourne), odnosno u datoteku .login ako rabite ljusku C.

## Dovršetak unosa naredbi

Ova značajka vam omogućuje da upišete početak imena datoteke i zatim pritisnete tipku <Tab> kako biste na zaslonu ugledali ostatak imena datoteke. Na taj ćete način uštediti vrijeme i izbjeći greške pri unosu. Ako dvije datoteka imaju isti početak, Linux

će dodati samo zajednički dio imena obaju naredbi i uputiti vam zvučni signal. Vi trebate unijeti nastavak željenog imena datoteke.

## Dodavanje teksta pomoću naredbi Cut i Paste

Distribucije Red Hat Linux i Slackware Linux nude program koji se može pokrenuti tijekom podizanja sustava i koji vam omogućuje uporabu miša kod odabira teksta bilo gdje na zaslonu. Taj tekst zatim možete ulijepiti u redak za zadavanje naredbi. Da biste ugledali pokazivač miša, dovoljno je pritisnuti jednu od njegovih tipki. Tekst odabirete tako da najprije lijevom tipkom miša kliknete na početak teksta koji želite odabrati, a zatim, držeći pritisnutu tipku miša, povučete pokazivač do kraja željenog teksta. Nakon toga kliknite na desnu tipku miša i kopirajte odabrati tekst u redak za zadavanje naredbi.

## Stvaranje skripti s naredbama

Ljuske prihvaćaju naredbe, interpretiraju ih i prilagođavaju tako da ih operativni sustav može izvoditi. U prethodnim odlomcima ste vidjeli kako ljuska interpretira posebne znakove pri dovršavanju imena datoteka, preusmjerava ulaz i izlaz, povezuje procese i u pozadinu postavlja određene procese.

Naredbu možete upisati na terminalu ili u datoteku. *Skripta* predstavlja skup naredbi u zajedničkoj datoteci. Da biste izvršili naredbe, potrebno je upisati samo ime datoteke, to jest skripte. Prednosti ovakvog pristupa su sljedeće:

- Ne morate više puta upisivati naredbe koje se odvijaju prema ustaljenom rasporedu.
- Možete odrediti korake koji su potrebni za obavljanje određene zadaće.
- Možete sebi i drugima pojednostaviti određene operacije.

Koristeći varijable i ključne riječi možete napisati programe koje će zatim ljuska interpretirati. Ovakav način rada je koristan jer omogućuje stvaranje općenitih skripti koje i vi i ostali korisnici možete koristiti u različitim situacijama.

Pretpostavimo da nakon što ste se prijavili za rad želite vidjeti tko još radi na sustavu, ili da želite pokrenuti program calendar koji sadrži vaš dnevni raspored sastanaka, te ispisati datum i vrijeme na zaslonu. Sve to možete učiniti pomoću ovih naredbi:

```
who
calendar
date
```

Ako te tri naredbe pohranite u datoteku pod imenom whatsup, te ako datoteka postane izvršna, dobit ćete skriptu koju pokrećete kao i bilo koju drugu naredbu. Datoteka whatsup mora biti datoteka s tekstom. Pomoću programa za unos teksta vi ili emacs upišite naredbe u datoteku whatsup. Zatim upišite iduću naredbu i datoteka whatsup će postati izvršna datoteka:

```
chmod +x whatsup
```

Naredba chmod služi za promjenu ili postavljanje dozvola vezanih uz određenu datoteku. Opcija +x pretvara datoteku u izvršnu datoteku, odnosno u datoteku koja se pokreće kao i bilo koja standardna naredba u Linuxu. Odjednom možete upisati naredbe u datoteku i datoteku načiniti izvršnom. Od tog trenutka možete u redak za unos naredbi upisati whatsup i pokrenuti naredbe koje su upisane u skripti. Skriptu možete koristiti kao bilo koju drugu naredbu. Primjerice, da biste ispisali rezultate naredbe whatsup, upišite sljedeće:

whatsup | 1p

Želite li rezultate upisati u datoteku pod imenom info, tada trebate pokrenuti ovu naredbu:

whatsup > info

Slijedeći ove upute možete stvoriti skriptu koja se koristi u svakoj prilici:

- 1. Pokrenite program za unos teksta, kao što je vi ili emacs, i upišite naredbe u datoteku (tekst ili ASCII). U prethodnom primjeru su naredbe upisane u datoteku whatsup.
- 2. Datoteci pridružite dozvolu za njezino izvođenje. Upišite chmod +x *imedatoteke* (primjerice, chmod +x whatsup).
- 3. Provjerite naredbu tako da upišete njezino ime i pritisnete tipku <Return>.

Nakon što prethodne upute upotrijebite više puta uvidjet ćete kako je jednostavno stvarati vrlo korisne skripte. Naravno da je najteži dio onaj kada trebate odrediti naredbe koje se koriste i način na koji možete ostvariti svoju zadaću.

Skriptu možete provjeriti ako upišete ovu naredbu:

sh -x ime-skripte

Oznaka *ime-skripte* predstavlja ime datoteke u koju je upisana skripta. Naredba sh –x služi za prikaz svih koraka, odnosno naredbi, koji su zapisani u skripti. Ova mogućnost je korisna za otklanjanje grešaka u skripti.

## Pisanje programa

Da biste napisali programe koji koriste ljuske, morate poznavati varijable i upravljačke strukture. No nemojte dozvoliti da vas ovi izrazi zaprepaste. Varijabla je objekt koja u svakom trenutku ima određenu vrijednost (od više različitih mogućih vrijednosti). Upravljačke strukture određuju način upravljanja naredbama iz skripte. Postoje dvije osnovne vrste upravljačkih struktura: uvjetne strukture (kao što su if...then...else ili

strukture case), te iteracijske strukture ili petlje (kao što su for ili while petlje). Kod uvjetnih struktura odabirete jedan od ponuđenih načina izvođenja naredbi. Odabir obično ovisi o nekoj varijabli ili o izlazu pojedine naredbe. Kod iteracijskih struktura se višestruko ponavljaju pojedine naredbe. U prethodnom odlomku "Podešavanje okružja ljuske" ste se upoznali s varijablama ljuske, a u idućem odlomku "Programiranje pomoću upravljačkih struktura" saznat ćete nešto više o upravljačkim strukturama.

**Uporaba naredbe echo** Naredbu echo možete koristiti za prikaz poruka koje vam daju informacije o tome što se događa u pojedinoj skripti. Naredba echo na zaslonu prikazuje svoje argumente, odnosno sve ono što slijedi iza riječi echo. Ako niz znakova stavite u navodnike, onda će oni biti prikazani na zaslonu. Rezultat naredbe echo možete preusmjeriti u neku datoteku.

Naredba

```
echo "Molim ostanite na vezi..."
```

prikazuje na zaslonu ovakav redak:

Molim ostanite na vezi...

Iduća naredba sprema poruku Molim ostanite na vezi... u datoteku pod imenom messg:

```
echo "Molim ostanite na vezi..." > messg
```

**SAVJET** Uporabom naredbe echo korisnik može provjeriti što se događa s nekom naredbom koju je prethodno pokrenuo, a nije dobio odziv nekoliko sekundi ili čak i duže.

Naredba echo je korisna kada želite provjeriti rad neke skripte. Uporabom ove naredbe na ključnim mjestima u skripti dobit ćete informaciju o tome što se doista događa prilikom izvođenja naredbi zapisanih u skripti. Evo kao izgleda datoteka whatsup u koju je dodano nekoliko naredbi echo:

```
echo" Pogledajmo tko radi na sustavu."
who
echo" Je li dogovoren kakav sastanak?"
calendar
date
echo" Sve je u redu."
```

Nakon što pokrenete datoteku whatsup, na zaslonu ćete ugledati sljedeće:

#### \$ whatsup

```
Pogledajmo tko radi na sustavu.
sarah ttyO1 Dec 20 08:51
brad ttyO3 Dec 20 08:12
ernie ttyO7 Dec 20 08:45
Je li dogovoren kakav sastanak?
```

```
12/20 Sales meeting at 1:45
12/21 party after work!
Mon Dec 20 09:02 EST 1993
Sve je u redu.
$
```

**Uporaba komentara** Uvijek postoji mogućnost da nakon što napišete skriptu koji potom ne koristite duže vrijeme zaboravite način rada skripte. Stavljajte komentare u skripte koji sadrže objašnjenja za uporabu pojedinih naredbi. *Komentar* je bilješka koju možete pročitati vi ili netko tko preuređuje skriptu. Ljuske zanemaruju komentare jer su oni važni samo ljudima, a ne računalima.

Oznaka funte (#) predstavlja početak komentara u skripti. Svaki znak u tekućem retku koji slijedi iza # smatra se komentarom. Evo kako možete "iskomentirati" skriptu whatsup:

```
# Ime: whatsup
# Napisana: 1/19/97, ime i prezime
# Namjena: Prikaz osoba koje su se prijavile, zakazanih sastanaka i datuma
echo " Pogledajmo tko radi na sustavu. "
who
echo " Je li dogovoren kakav sastanak? "
calendar
date
echo " Sve je u redu. "
```

Ponovno pokrenite skriptu i dobit ćete isti rezultat kao i prije. Komentari ne mijenjanju način izvođenja naredbi upisanih u skripti.

**Uporaba varijabli u programima** Da biste koristili varijable u programiranju ljuski, morate poznavati način dodjele vrijednosti određenoj varijabli te način pristupa postojećoj vrijednosti varijable. Korištenje vrijednosti varijabli je jednostavno, ali postoje četiri načina kako varijabli možete promijeniti vrijednost:

- izravnom dodjelom vrijednosti
- pomoću naredbe read
- pomoću parametara naredbi
- zamjenom izlaza pojedine naredbe

*Izravna dodjela vrijednosti varijable* Izravan način dodjele vrijednosti pojedinoj varijabli je da napišete izraz sličan ovome:

myemail=edsgar@crty.com

kojim ćete varijabli myemail dodijeliti vrijednost edsgarŽcrty.com. Nemojte koristiti razmake na bilo kojoj strani znaka jednakosti (=). Izravna promjena vrijednosti pojedine varijable mora imati ovakav oblik:

```
ime varijable=vrijednost varijable
```

Ako *vrijednost varijable* sadrži i razmake, onda je upišite unutar navodnika. Da biste varijabli myoffice dodijelili adresu na kojoj se nalazi vaš ured, napišite sljedeću naredbu: myoffice="Room 21, Suite C"

Ljuska vraća vrijednost varijable svaki puta kada se ime varijable nalazi iza znaka dolara (\$). Pogledajte što se događa kada se pokrenu ove dvije naredbe:

```
echo" Moja adresa za e-postu je $myemail"
echo" Moj ured se nalazi na adresi $msoffice"
```

Pretpostavimo da često kopirate datoteke u direktorij /corporate/info/public/sales. Da biste kopirali datoteku current u taj direktorij, upišite ovu naredbu:

```
cp current /corporate/info/public/sales
```

Stvar će izgledati puno jednostavnija ako dugačko ime direktorija pohranite u varijablu corpsales:

```
corpsales=/corporate/info/public/sales
```

Ako potom budete trebali kopirati datoteku current u taj direktorij, to možete učiniti na ovaj način:

```
cp current $corpsales
```

Ljuska će zamijeniti \$corpsales s vrijednošću varijable corpsales i zatim pokrenuti naredbu za kopiranje.

**Uporaba naredbe read** Naredba read uzima idući redak kao ulaz i pridjeljuje ga varijabli. U idućem primjeru skripte korisnik može sam odrediti ime datoteke koja se kopira u direktorij /corporate/info/public/sales:

```
# Ime: copycorp
# Namjena: kopiranje odrelene datoteke u direktorij
# /corporate/info/public/sales
corpsales=/corporate/info/public/sales
echo " Unesite ime datoteke koju želite kopirati" # upit
read filename # učitavanje imena
cp $filename $corpsales # kopiranje
```

Naredba read zaustavlja izvođenje naredbi iz skripte i čeka unos s tipkovnice. Nakon što pritisnete tipku <Return> nastavlja se izvođenje naredbi iz skripte. Izvođenje se prekida ako pritisnete kombinaciju tipki <Ctrl-d> umjesto odgovora na pitanje koje vam je postavljeno pomoću naredbe read.

**Uporaba parametara u naredbama** Kada ljuska pokreće naredbe, ona pridjeljuje imena svim elementima u retku za zadavanje naredbi. Pod pojmom elementa podrazumijeva se niz znakova odvojenih razmakom ili tabulatorom (pomoću navodnika možete naznačiti da razmak nije ujedno i odjelnik elemenata u retku za zadavanje naredbi). Varijable koje se dodjeljuju elementima u retku za zadavanje naredbi označavaju se kao \$0, \$1, \$2 i tako dalje do \$9. Ova imena odgovaraju položaju elementa u retku za zadavanje naredbi, ime naredbe se označava s \$0, prvi argument ka \$1, i tako dalje. Da biste shvatili uporabu ovakvog koncepta, uočite primjer skripte pod imenom shovars:

```
# Ime: shovars
# Namjena: demonstracija varijabli u retku za zadavanje naredbi
echo $0
echo $2 $4!
echo $3
```

Sada zamislite da naredbu upišete u sljedećem obliku:

shovars -s hello "look at me" bart

Izlaz te naredbe će ovako izgledati:

```
shovars
hello bart!
look at me
```

Možete vidjeti da je u prvom retku napisano ime naredbe (varijabla \$0), a u drugom retku drugi i četvrti argument (varijable \$2 i \$4), dok zadnji redak sadrži treći argument (varijabla \$3).

No, posvetimo se nešto ozbiljnijem primjeru. Iduća skripta briše određenu datoteku no prije toga je kopira u direktorij /tmp tako da je u slučaju potrebe možete vratiti:

Unesete li naredbu safrm abc def, biti će obrisana samo datoteka abc jer skripta safrm briše samo varijablu \$1. No možete u skripti sve parametre prikazati pomoću \$\* i na taj način će skripta služiti za brisanje svih datoteka koje unesete u redak za zadavanje naredbi. Upišete li safrm abc def xx guio, sve četiri datoteke (abc, def, xx i guio) će biti uklonjene iz tekućeg direktorija.

**Zamjena izlaza naredbe** Varijabli možete pridijeliti rezultat jedne od izvršenih naredbi. Da biste u varijablu cwd pohranili ime direktorija u kojem trenutno radite, trebate upisati ovo:

cwd= 'pwd'

Uočite da je naredba pwd napisana unutar dvaju apostrofa, a ne u navodnicima.

Iduća skripta mijenja ime datoteke tako što u ime datoteke dodaje mjesec, dan i godinu:

```
# Ime: stamp
# namjena: promjena imena - dodavanje datuma u ime datoteke
# postavljanje tekućeg datuma (oblik mmddyy) u varijablu td
    td='+%m%d%y'
# promjena imena datoteke
    mv $1 $1.$td
```

U ovom primjeru varijabla td ima vrijednost tekućeg datuma. U zadnjem retku ta se vrijednost dodaje varijabli \$1. Ako je tekući datum 24. veljače 1997, te ako skriptu primijenite na datoteku pod imenom myfile, datoteka će biti premještena (što u osnovi znači da će joj se promijeniti ime) u datoteku myfile.022497.

**Uporaba posebnih znakova u programima** Imali ste priliku vidjeti kako ljuska na poseban način tretira znakove kao što su >, \*, ?, \$ i drugi. No što biste trebali učiniti da ti posebni znakovi ne budu više "posebni"? Postoji nekoliko odgovora na to pitanje.

Možete upotrijebiti apostrofe kako bi ljuska zanemarila te posebne znakove. Niz znakova upišite unutar apostrofa, kao u ovom primjeru:

```
grep '^Mary Tuttle' customers
```

Rezultat je takve naredbe da će biti prikazani reci u datoteci customers koji počinju s imenom Mary Tuttle. Oznaka Č upućuje naredbu grep da traženje započinje na početku retka. No ako se ime Mary Tuttle ne nalazi unutar dvaju apostrofa, naredba će biti sasvim drukčije interpretirana. Osim toga, razmak između riječi Mary i Tuttle neće biti shvaćen kao odjelnik polja ako se obje riječi nalaze unutar apostrofa.

Također možete upotrijebiti i navodnike kako bi ljuska zanemarila većinu posebnih znakova (s iznimkom znaka dolara \$ i znaka '). U idućem primjeru se zvjezdice (\*), razmaci i znak (>) smatraju običnim znakovima jer se nalaze unutar navodnika:

```
echo " ** Molim upi{ite odgovor ->"
```

U idućem primjeru se pravilno koristi vrijednost varijable LOGNAME, no ne postoji vrijednost za varijablu \$5:

echo " >>>Hvala za \$5, \$LOGNAME"

Uporabom obrnute kose crte (Đ) postići ćete da ljuska zanemaruje pojedini znak. Želite li, primjerice, da ljuska zanemari znak dolara ispred brojke 5, tada trebate upisati ovu naredbu:

```
echo " >>>Hvala za $5, $LOGNAME"
```

Dobit ćete ovakav odziv:

>>>Hvala za \$5, wrev

## Programiranje pomoću upravljačkih struktura

Postoje dvije osnovne upravljačke strukture koje se koriste u programiranju ljuski. Kod *uvjetnih struktura* kao što su if...then...else i case, na osnovi vrijednosti određenog izraza (varijable, značajki datoteke, broja parametara u ljusci ili rezultata izlaza naredbe), pokreće se određeni skup naredbi. Kod *iteracijskih struktura*, kao što su petlje for i while, mogu se izvršavati određene naredbe na zbirkom datoteka ili do ispunjenja određenih uvjeta.

U nastavku ćete upoznati primjere koji, iako nisu previše složeni, objašnjavaju uporabu upravljačkih struktura.

**Struktura case** Ova struktura omogućuje vam da na osnovi vrijednosti pojedine varijable odaberete jednu od ponuđenih naredbi. U ispisu 18.2 možete vidjeti kratki program za uvođenje izbornika.

```
Ispis 18.2
               Implementacija izbornika pomoću naredbe case
# Name:
             ShrtMenu
             Allow user to print, a file, delete, a file,
# Purpose:
#
             or quit the program
# Display menu
     echo "Please choose either P, D, or Q to "
     echo " [P]rint, a file"
     echo " [D]elete, a file"
     echo " [Q]uit "
# Get response from user
     read response
# Use case to match response to action
     case $response in
        P(p) echo "Name of file to print?"
             read filename
             lp $filename; ;
        D¦d) echo "Name of file to delete?"
             read filename
             rm $filename; ;
        *)
             echo "leaving now";;
     esac
```

Tvrdnja case ima ovakav oblik:

```
case riječ in
    uzorak) tvrdnja (s);;
    uzorak) tvrdnja (s);;
    ...
esac
```

Parametar *riječ* se uspoređuje s parametrima *uzorak*, počev od uzorka na početku popisa. Ako su parametri jednaki, izvode se odgovarajuće naredbe (koje su odvojeni s dva znaka ;;). Kraj tvrdnje case je označen riječju esac (ili naopako case).

U ispisu 18.2 je prikazana uporaba znaka ¦ koji se koristi da bi vam ponudio mogućnost izbora. Primjerice oznaka P¦p znači da i malo i veliko slovo p zadovoljava uvjet.

Uzorak \* zamjenjuje sve uzorke koji nisu izravno navedeni. Ako korisnici pritisnu bilo koju od tipki <P>, , <D> ili <d>, napustit će izbornik.

Ispis 18.3 koristi tvrdnju case za odabir na osnovu broja parametara koje ljuska predstavlja oznakom \$#.

ISDIS 10.3 Raslavilatile flareupi politocu flareupe ca	Ispis 18.3	Rastavlianie nare	dbi pomoćı	i naredbe	case
--	------------	-------------------	------------	-----------	------

```
# Name: recent
# Purpose: list the most recent files in, a directory
# If user types recent <Return > then the names of
# the 10 most recently modified files are displayed
# If the user types recent n < Return > then the names of
# the n most recently modified files are displayed
# Otherwise, user is notified of incorrect usage
#
```

**Određivanje izlaznog stanja** Nakon izvršenja naredbe u ljusci njezin je status uspješan ili neuspješan. Koristite li naredbu grep "American Terms" customers da biste saznali nalazi li se u popisu korisnika datoteke i American terms, potrebna vam je dozvola za čitanje datoteke. Ako se u datoteci nalazi izraz American Terms, onda je naredba uspješno obavljena. Naredba će biti neuspješna ako nema izraza u datoteci ili ako nemate dozvolu za njezino čitanje.

Ljuska uvijek izvješćuje o stanju naredbe, programa ili skripte. Vrijednost koja opisuje stanje naredbe naziva se vrijednost *izlaznog stanja* i označena je s #?. Upišete li iduće naredbe, dobit ćete vrijednosti izlaznog stanja:

```
grep "American Terms" customers
echo $?
```

BILJEŠKA Ako varijable \$? pokazuje vrijednost 0, naredba je uspješno izvedena, a u protivnom naredba je neuspješna.

Slijedi primjer u kojem je izlazno stanje naredbe who¦grep \$1 iskorišteno kao parametar u tvrdnji case:

Ako upišete just.checking rflame i ako je rflame prijavljen za rad, na zaslonu ćete vidjeti sljedeće:

rflame is logged in. Have, a great day!

Ako korisnik rflame nije prijavljen, onda ćete ugledati ovo:

rflame is not here. Try again later. Have, a great day! **Uporaba if struktura** Struktura if...then...else...fi omogućuje vam da odaberete dvije grupe naredbi koje se izvode ako je ispunjen uvjet (dio else je proizvoljan). Umjesto točkica (...) upisuju se jedna ili više naredbi. Uz pretpostavku da je nula izlazno stanje zadnje naredbe u segmentu if (to jest da je naredba uspješno obavljena), izvodit će se blok naredbi iz segmenta then. U protivnom se izvode naredbe iz segmenta else.

Drugim riječima, izvodi se jedna ili više naredbi. Ako je zadnja naredba uspješno obavljena, obavljaju se naredbe iz segmenta then, a potom naredbe koje se nalaze iz riječi fi (fi je oznaka za kraj upravljačke strukture). Ako je zadnja naredba neuspješna, obavljaju se naredbe iz segmenta else.

Evo poznatog primjera koji je nalik primjeru za uporabu tvrdnje case:

**Uporaba naredbe test** Mnoge skripte koje su prikazane u ovom poglavlju očekuju da se njihovi korisnici "lijepo ponašaju". Ne provjerava se imaju li korisnici dozvolu za kopiranje ili premještanje datoteka, odnosno koriste li oni obične datoteke ili direktorije. Pomoću naredbe test možete provjeriti te i druge stvari. Primjerice, naredba test –f abc će imati izlazno stanje 0 ako postoji obična datoteka abc.

Možete obrnuti značenje naredbe test ako ispred opcije upišete uskličnik. Primjerice, da biste provjerili nemate li dozvolu za čitanje datoteke abc, upišite naredbu test ! –r abc. U tablici 18.7 možete vidjeti nekoliko opcija naredbe test.

Tablica 18.7	Opcije koje se koriste za provjere datoteka pomoću naredbe test
Орсіја	Značenje
- f	Uspješna ako postoji datoteka i ako se radi o običnoj datoteci
- d	Uspješna ako se radi o datoteci-direktoriju
- r	Uspješna ako postoji datoteka i ako se ona može čitati
- S	Uspješna ako postoji datoteka koja nije prazna
- W	Uspješna ako postoji datoteka i ako se u nju može pisati
- X	Uspješna ako postoji datoteka i ako se radi o izvršnoj datoteci.

U ispisu 18.4 dat je primjer korištenja naredbe test.

Ispis 18.4 Primjer skripte u kojoj se koristi naredba test # Name: safcopy # Purpose: Copy file1 to file2 # Check to see we have read permission on file1 # If file2 exists then # if file2 is, a file we can write to # then warn user, and get permission to proceed # else exit # else # copy file # # Check for proper number of arguments case \$# in 2) if test ! -r \$1 # cannot read first file; ; then; ; # exit with non-zero exit status; ; exit (1) fi; ; if test -f \$2 # does second file exist?; ; then; ; echo " \$2 exists, copy over it? (Y/N) "; ; # get permission from user; ; read resp case \$resp in; ; Y¦y) cp \$1 \$2;; # go ahead; ; \*) exit(1); ; # good byel; ; esac; ; else; ; #Second file exists but can't write; ; exit (1) fi # Second file doesn't exists but can't write; ; else fi; ; \*) echo "Usage: safcopy source destination" ; ; exit (1) ; ; esac

Naredbu test možete upotrijebiti i za provjeru brojeva. Da biste odredili je li vrijednost varijable hour veća od 12, upišite naredbu test \$hour -gt 12. U tablici 18.8 naći ćete popis nekih opcija koje možete koristiti za uspoređivanje brojeva pomoću naredbe test.

Tablica 18.8	Opcije naredbe test za usporedbu brojeva
Орсіја	Značenje
-eq	jednak
-ne	različit od
-ge	veći ili jednak
-gt	veći od

nastavlja se

Tablica 18.8	Nastavak
Opcija	Značenje
-le	manji ili jednak
-It	manji od

Ispis 18.5 pokazuje kako se te opcije koriste kod različitih načina pozdravljanja.

	Ispis 18.5 Ispis pozdrava na zaslonu pomoću naredbe test	
#	Name: greeting	
#	Purpose: Display Good Morning if hour is less than12	
#	Good Afternoon if hour less than 5PM	
#	Good Evening if hour is greater than 4PM	
#	Get hour	
	hour='date +%Hsymbol 39 \f "Arial" \s 12'	
#	Check for time of day	
	if test \$hour - It 12	
	then	
	echo "Good Morning, \$LONGNAME	
	else	
	if test \$hour -It 17	
	then	
	echo "Good Afternoon, \$LOGNAME"	
	else	
	echo "Good Evening, \$LOGNAME"	
	fi	
	fi	

**Uporaba iteracijskih struktura** Iteracijske upravljačke strukture omogućuju vam pisanje skripti koje sadrže programske petlje. Dvije osnovne vrste petlji su for i while.

U petljama tipa for određuje se nekoliko datoteka ili vrijednosti koje se koriste u određenom bloku naredbi. Da biste kopirali sve datoteke čija imena završavaju sa znakovima .txt u direktorij pod imenom textdir, upotrijebite naredbu for:

Ljuska interpretira tvrdnju for i in \*.txt i omogućuje da varijabla i poprimi ime svih datoteka u tekućem direktoriju čija imena završavaju s .txt. Potom možete varijablu \$i koristiti u svim tvrdnjama koje se nalaze između ključnih riječi do i done.

Skripta u ispisu 18.6 služi za ispis datoteka na odvojenim *banner* stranicama. Osim toga, skripta šalje poštu onim korisnicima koji očekuju ispis svojih dokumenata. Znakovi \$\* predstavljaju sve parametre koji se pridružuju naredbi u ljusci.

```
Ispis 18.6
               Upravljanje datotekama pomoću naredbe for
# Name:
              Prntel
              Print one more files
# Purpose:
#
              each with own title page
#
              Notify user which files were sent to the printer
#
              and which were not.
#
              Do this for all parameters to the command
for i in $*
do
     if lp -t $i -dlasers $i > /dev/null
     then
           echo $i >> printed
     else
           echo $i >> notprinted
     fi
done
# end of loop
if test -s printed
then
     echo "These files were sent to the printer " > mes
     cat printed >> mes
     mail $LOGNAME < mes</pre>
     rm mes printed
fi
if test -s notprinted
then
     echo "These files were not sent to the printer " > mes
     cat notprinted >> mes
     mail $LOGNAME < mes</pre>
     rm mes notprinted
fi
```

Petlja while, na isti način kao i tvrdnja if, provjerava izlazno stanje određene naredbe. Skripta iz ispisa 18.7 bilježi korisnike koji su primili novu poštu. Pretpostavka je da, ako se poštanski sandučić promijenio, korisnik je dobio novu poštu. Skripta koristi naredbu diff za usporedbu dviju datoteka te izvještava o razlikama između njih. Ako se radi o datotekama istog imena, izlazno stanje naredbe je 0 (naredba je bila uspješna).

	Ispis 18.7	Ponavljanje naredbi pomoću naredbe while	
#	Name:	checkmail	
#	Purpose:	Notify user if their mail box has changed	
#	Suggestion:	Run this in the background	
#	get, a size	of mail box for comparison	
	cp \$MAIL	omail # Get set for first time through	
#	MAIL is, a"	"special" variable indicating the user's mailbox	
#	while omail	and \$MAIL are the same, keep looping	
	while dif	ff omail \$MAIL > /dev/null	
	do		
	(	cp \$MAIL omail	
	9	sleep 30	
		14	astavli

nastavlja se

done # There must be, a change in the files echo "New mail!!" ¦ write \$LOGNAME

Možda ćete htjeti neke naredbe koje koriste oblike if... then... else zamijeniti petljom *while*. Razlika je u tome što se petlja *while* ponavlja više puta uzastopce, bez vašeg utjecaja.

## Prilagodba ljuski u Linuxu

Ljuska se pokreće čim se prijavite za rad u Linuxu. Tablice 18.2 i 18.3 prikazuju posebne varijable čije vrijednosti određuju okružje ljuske. Ljuske podešavaju vrijednosti nekih od tih varijabli. Podešenja možete promijeniti tako da preuredite datoteku .profile (ako koristite ljusku Bourne ili bash), ili datoteku .login (ako koristite ljusku C). Osim toga, možete koristiti i zamjenske naredbe.

Svaki puta kada pokrenete naredbu pokreće se i nova ljuska koja poprima mnoge značajke, to jest postojeće okružje, tekuće ljuske. Uz novu ljusku su vezane dvije stvari:

- Nova ljuska se pokreće u tekućem direktoriju. Naredba pwd vraća istu vrijednost unutar ljuske kao i prije nego je ljuska pokrenuta.
- Nova ljuska poprima mnoge vrijednosti varijabli iz postojeće ljuske. Postoji više načina kako se vrijednosti varijabli u postojećoj ljusci izvoze u novu ljusku.

## Izvoz varijabli u novu ljusku

Stvorite li varijable ljuski ili ako postojećim varijablama pridijelite određenu vrijednost, te će se vrijednosti odnositi na ljusku koju ste pokrenuli. Varijabla koju ste podesili u početnoj ljusci (ljuska koja se koristi čim se prijavite za rad u Linuxu) je na raspolaganju svim naredbama koje zadajete u retku za zadavanje naredbi, dok varijabla koju ste podesili u nekoj drugoj ljusci zadržava svoju vrijednost samo dok koristite tu ljusku. Čim napustite određenu ljusku, vrijednost varijable nestaje ili se postavlja njezina osnovna vrijednost.

Evo primjera. Upišite iduće dvije naredbe:

```
today=Thursday
echo $today
```

Pretpostavimo da naredba echo prikazuje vrijednost Thursday. Zamislite da ste napisali i pokrenuli sljedeću skriptu s naredbama:

```
# Name: whatday
# display the current value of the variable today
    echo "Today is $today."
# set the value of today
    today=Friday
# display the current value of the variable today
    echo "Today is $today."
```

Zatim u redak za unos naredbi upišite sljedeće četiri naredbe:

chmod +x whatday today=Thursday whatday echo \$today

Na zaslonu će se pojaviti sljedeće:

```
Today is .
Today is Friday.
Thursday
```

Vrijednost varijable today u početnoj ljusci je Thursday. Nakon što pokrenete skriptu whatday vidjet ćete da varijabla today inicijalno nije definirana (prikaz na zaslonu je Today is .). Zatim varijabla today u ljusci ima vrijednost Friday. Nakon što ste pokrenuli skriptu whatday i vratili se u početnu ljusku varijabla today će poprimiti njezinu osnovnu vrijednost Thursday.

Da biste varijabli today pridijelili istu vrijednost koju ima u početnoj ljusci, trebate upotrijebiti naredbu export. Pomoću ove naredbe možete "izvesti" odnosno proslijediti varijable iz jedne ljuske u pripadne podljuske:

export today

Sada bilo koja ljuska koju pokrenete iz početne ljuske poprima vrijednost varijable today. U prethodno navedeni set naredbi dodajte i naredbu export:

```
today=Thursday
export today
whatday
echo $today
```

Dobit ćete sljedeći odziv na zaslonu:

```
Today is Thursday.
Today is Friday.
Thursday
```

Uočite da vrijednost koju je varijabla poprimila u ljusci koja se pokreće pomoću skripte whatday nije vraćena u početnu ljusku. Izvoz varijabli je moguć u jednom smjeru, od tekuće ljuske prema novoj ljusci (zapravo podljusci), a nikada obratno. Posljedica je toga da ćete se opet naći na početnom mjestu ako promijenite tekući direktorij unutar jedne ljuske, nakon što ljuska završi s radom.

Bilo koju varijablu možete izvesti iz jedne ljuske u njezinu podljusku koristeći ovu naredbu:

```
export ime-variable
```

U toj naredbi *ime-varijable* je ime one varijable koju izvozite. Da biste promijenili podešenje svog terminala i postavili terminal vt100, upišite sljedeće naredbe kako bi nova vrijednost varijable TERM postala dostupna svim podljuskama ili programima:

```
TERM=vt100
export TERM
```

Kada mijenjate ili podešavate varijable u ljusci bash (to činite u datoteci .profile), nemojte ih zaboraviti izvesti. Primjerice, želite li varijablu PATH podesiti tako da bude PATH=/bin:/usr/local/bin:., upišite je u tom obliku u datoteku .profile, a ispod tog retka unesite naredbu export:

```
export PATH
```

Da biste promijenili prompt ljuske, morate promijeniti vrijednost varijable PS1 u datoteci .profile. Ako želite vrijednost promijeniti tako da prompt umjesto znaka \$ sadrži tekst Ready \$, onda u datoteku .profile upišite ovo:

```
PS1="Ready $"
export PS1
```

BILJEŠKA Promjene koje ste načinili u datotekama .profile ili .login nećete moći koristiti sve dok se ne odjavite i ponovno prijavite za rad u sustavu Linux.

## Definiranje zamjenskih naredbi

Zamjenske naredbe su korisne za one naredbe koje često koristite, ali koje imaju puno detalja koje treba upamtiti i zapisati. Pomoću zamjenskih naredbi možete urediti svoje radno okružje tako da ono izgleda onako kako vi to želite. U idućem primjeru možete vidjeti kako se dade načiniti naredba recent koja služi za ispis 10 najnovijih datoteka u tekućem direktoriju:

alias recent="ls -lat;head"

Da ne biste trebali nakon svake prijave upisivati ovakvu naredbu, unesite je u datoteku .login (ako koristite ljusku C) ili u datoteku .profile (ako koristite ljusku bash ili neku sličnu ljusku). Na taj će vam način zamjenska naredba recent uvijek biti na raspolaganju kada radite u navedenim ljuskama.

## Odavde...

Ljuska je osnovno sučelje između vas i operativnog sustava Linux. Iako ljuska može biti gotovo svaki izvršni program, postoji nekoliko standardnih ljuski u Linuxu. Neke od njih možete naći u osnovnom kodu (napisanomu programskom jeziku C), a druge su već kompilirane za vaše računalo. Sve ljuske u Linuxu su vrlo sofisticirane i sadrže sve uobičajene konstrukcije poznate iz programskog jezika u kojem su pisane. Posebna je namjena programskog jezika u kojem su napisane Linux ljuske da spoji puno malih naredbi i uslužnih programa iz okružja Linuxa. Ljuske vam omogućuju da uz malo napora napišete složene programe koristeći standardni ulaz i izlaz podataka i rad u pozadini glavnog programa. Više podataka potražite u idućim poglavljima:

- 5. poglavlje, "Pokretanje Linux programa" nudi vam osnovne upute za rad s Linuxom.
- 8. poglavlje, "Uporaba vi editora" vas upućuje kako se uređuju datoteke s tekstom.

# poglavlje 19

## Upravljanje većim brojem procesa

(Jack Tackett)

## U ovom poglavlju

Razumijevanje višezadaćnosti Pokretanje većeg broja procesa Uporaba naredbi vremenskog raspoređivanja zadataka Nadziranje višezadaćnog okružja i izvještavanje o trenutnom stanju Nadzor nad većim brojem procesa Linux je višekorisnički i višezadaćni operativni sustav. *Višekorisnički* znači da nekoliko osoba istovremeno može koristiti računalni sustav (za razliku od jednokorisničkog operativnog sustava, poput MS-DOS-a). *Višezadaćni* znači da Linux, poput Windows NT-a, može istodobno raditi na nekoliko zadaća - može započeti rad na jednoj zadaći i preuzeti drugu prije nego je prva zadaća završena.

Vođenje brige o višekorisničkih zahtjeva i višezadaćnosti poslovi su operativnog sustava. Većina sustava ima samo jedan procesor i jedan skup čipova koji čine glavnu memoriju, to jest RAM. Za sekundarnu memoriju sustav može imati više od jednog diska ili tračni pogon te nekoliko ulazno-izlaznih uređaja. Svi ti resursi trebaju biti upravljani i dijeljeni među nekoliko korisnika. Operativni sustav stvara privid da svaki korisnik ima zasebni računalni sustav.

## Razumijevanje višezadaćnosti

Kako je prije spomenuto, posao je Linuxa stvaranje privida da vam sustav, kada izdate zahtjev, posvećuje punu pažnju. U stvarnosti, u vremenu nakon vašeg pritiska na <Return> do odaziva sustava na vašu naredbu može biti obrađeno stotine drugih zahtjeva.

Zamislite da istovremeno trebate pratiti na desetke zadaća. Procesorsku snagu, spremišne kapacitete, ulazne i izlazne uređaje trebate dijeliti između nekoliko korisnika ili nekoliko procesa koji pripadaju jednom korisniku. Linux nadzire popis - poznat i kao *queue* - zadaća koje čekaju da budu izvršene. Te zadaće mogu uključivati korisničke poslove, zadaće operativnog sustava, poštu i pozadinske poslove poput ispisivanja. Linux za svaku zadaću raspoređuje djeliće sustavskog vremena. Prema ljudskim standardima svaki je od tih vremenskih odsječaka izuzetno kratak - djelić sekunde. U računalnom je vremenu taj vremenski odsječak dovoljan da program obradi stotine ili tisuće instrukcija. Dužina vremenskog odsječka za svaku zadaću može ovisiti o razmjernom prvenstvu svake zadaće.

Linux neko vrijeme radi na jednoj zadaći iz rasporeda, tu zadaću stavi na stranu da bi započeo rad na drugoj zadaći i tako dalje. Zatim se vraća do prve zadaće i ponovno radi na njoj. Linux to kruženje nastavlja sve dok zadaću ne završi i izvadi je iz reda ili dok zadaća ne bude zaustavljena. U takvom su načinu rada, poznatom kao *time-sharing (dijeljenje vremena)*, resursi sustava dijeljeni između svih zadaća. Naravno, to dijeljenje vremena treba biti izvršeno na pouzdan i učinkovit način. U UNIX-u se umjesto zadaće koristi pojam *procesa*. Tablica 19.1 prikazuje nekoliko vrsta procesa.

Tablica 19.1	Vrste procesa
Vrsta procesa	Opis
interactive	Pokrenut iz ljuske, a izvršava se u prednjem planu ili u pozadini.
batch	Tipična vrsta procesa raspoređenog za izvršavanje u određenoj točci vremena.
daemon	Obično se pokreće u vremenu pokretanja sustava radi izvršavanja funkcija operativnog sustava prema zahtjevu, poput LPD, NFS i DNS.

Već ste vidjeli da se program može pokrenuti u pozadini. Dok program radi u pozadini, vi možete nastaviti unositi naredbe ili raditi druge poslove. To je značajka višezadaćnosti: Linux za uravnoteženje vaših neposrednih naredbi i onih koje rade u pozadini koristi metodu dijeljenja vremena. U ovom su poglavlju prikazani drugi načini za raspoređivanje procesa tako da se oni mogu izvršavati bez vašeg nadzora (*batch* proces).

#### Vidi "Izvođenje procesa u pozadini", str. 361

Prvenstvena je obaveza operativnog sustava Linux baratanje pojedinostima u radu s nekoliko korisnika i s nekoliko procesa. Kao korisnik imate moć da odredite koji program želite pokrenuti. Neke vam naredbe u Linuxu omogućuju da odredite kada želite pokretanje procesa. Također možete nadzirati vaše procese kao i vidjeti koji se drugi procesi izvršavaju. U nekim slučajevima možete promijeniti njihovo relativno prvenstvo., a ako se ukaže potreba, vaš proces uvijek možete zaustaviti. Ako ste vi administrator sustava, imate sve ove mogućnosti kao i odgovornost i moć da pokrećete, nadzirete i upravljate procesima koji pripadaju operativnom sustavu ili bilo kojem korisniku.

U tablici 19.2 je popis naredbi koje omogućuju upravljanje višekorisničkim i višezadaćnim mogućnostima Linuxa.

Naredba	Radnja
at	Izvršava naredbe u zadano vrijeme
batch	Izvršava naredbe kada opterećenje sustava to dopusti
cron	Izvršava naredbe prema rasporedu
crontab	Održava crontab datoteke za pojedinačne korisnike
kill	Zaustavlja procese
nice	Podešava prvenstvo procesa prije njegovog pokretanja
nohup	Omogućuje nastavljanje procesa i nakon što se vi odjavite
ps	Prikazuje informacije o procesu
renice	Podešava prvenstvo procesa koji se trenutno izvršava
W	Prikazuje tko je prijavljen na sustav i što radi
who	Prikazuje koji su korisnici prijavljeni na sustav

 Tablica 19.2
 Višekorisničke i višezadaćne naredbe

BILJEŠKA Više informacija o naredbama u tablici 19.2 možete potražiti na sljedećoj man stranici:

man command

Također možete koristiti opciju -- help

command --help ■

## Pokretanje većeg broja procesa

Izvršavanje programa možete pokrenuti unošenjem njegovog imena. Programe možete pokrenuti i iz datoteka koje sadrže naredbe ljuske. Izvršavanje programa može uzajamno djelovati s mnogim različitim dijelovima sustava. Program može učitavati ili upisivati iz odnosno u datoteke, upravljati svojim informacijama u RAM-u ili slati informacije u pisače, modeme ili druge uređaje. Također, operativni sustav informacije pridružuje procesu tako da ih sustav može pratiti i upravljati s njima.

Proces je izvršavanje programa, ali je različit od samog programa. Proces je, u određenom smislu, više od programa, budući da je program samo skup instrukcija. Proces je dinamičan, budući da koristi resurse sustava u kojem se izvršava. S druge strane, jedan samostalan Linux program može pokrenuti nekoliko procesa.

Linux te procese prepoznaje i prati ih dodijeljujući svakom procesu identifikacijski broj (PID).

## Pokretanje većeg broja procesa

Već ste vidjeli da je vaša prijavna ljuska uvijek u radu. Svaki put kada unesete naredbu pokrenete najmanje jedan novi proces dok prijavna ljuska nastavlja rad. Ako, na primjer, upišete sljedeću naredbu, datoteka nazvana report.txt je poslana u program *lp*:

```
lp report.txt
```

```
Vidi "Razumijevanje ljuske", str. 339
```

Kada program *lp* dovrši svoju zadaću, ponovno se pojavi naredbeni redak ljuske. Međutim, prije nego se naredbeni redak ljuske ponovno pojavi, prijavna ljuska i naredba lp su se već izvršavale - u tom ste slučaju pokrenuli više procesa. Ljuska je čekala da naredba *lp* završi prije nego je naredbeni redak ljuske ponovno prikazala na zaslonu.

## Pokretanje pozadinskog procesa

Proces možete pokrenuti kao pozadinski posao zadavši naredbu za pokretanje procesa, a iza naredbe stavite znak &. Na primjer, ako unesete naredbu *lp report.txt* &, ljuska odmah odgovara s brojem - PID za taj proces. Naredbeni se redak ljuske ponovno pojavljuje bez čekanja da proces završi. Slijedi primjer onog što biste mogli vidjeti:

```
$ 1p report.txt &
3146
$
```

3146 je u ovom primjeru PID procesa pokrenutog naredbom lp.

Bez obzira da li naredbu *lp* pokrećete u pozadini, proces pridružen naredbi *lp* je pokrenut iz trenutne ljuske. Proces lp je *child* proces (proces dijete) trenutne ljuske. Ovaj primjer ukazuje na uobičajeni odnos između procesa - onaj između roditelja i djeteta. Vaša je trenutna ljuska roditeljski (matični) proces, a proces *lp* koji se trenutno izvršava, je proces *dijete*. Matični proces prije nego nastavi s radom obično pričeka završetak jednog od ili više svojih procesa potomaka. Ako želite da roditelj nastavi bez čekanja da dijete završi, naredbi koja će pokrenuti proces *dijete* pridružite znak &. Dok se potomak izvršava, možete nastaviti s drugim radom ili naredbama.

BILJEŠKA Ako radite iz znakovnog terminala ili preko daljinske prijave na sustav, vaša je trenutna ljuska obično vaša prijavna ljuska. Međutim, ako koristite virtualni terminal ili prozor terminala iz GUI-a (grafičkog sučelja), svakoj je vašoj prijavi na sustav (novi session) pridružena zasebna ljuska.

## Uporaba cijevi za pokretanje višestrukih procesa

Drugi je način za pokretanje višestrukih procesa uporaba jedne ili više cijevi na naredbenom retku. Za ispis dugačkog popisa 10 posljednjih preinačenih datoteka u vašem trenutnom direktoriju unesite ovu naredbu:

```
ls -lt | head | lp
```

Ova naredba istovremeno pokreće tri procesa i svi su oni potomci (djeca) trenutne ljuske. Cijev djeluje na ovaj način: naredbe s obje strane okomite crte (đ) počinju u isto vrijeme. Ni jedna nije roditelj one druge - obje su potomci procesa koji se izvršavao kada su one stvorene. U tom smislu o naredbama s obje strane simbola cijevi možete misliti kao o bratskim procesima.

Neki su programi napisani tako da oni sami stvore nekoliko procesa. Jedan je takav primjer naredba *ispell*, koja popisuje riječi u dokumentu koje Linux nije mogao pronaći u rječniku sustava. Naredba *ispell* pokrene nekoliko zavisnih procesa. Pretpostavimo da unesete ovo:

```
ispell final.rept > final.errs &
```

Prikazat će vam se sljedeći rezultati:

```
1286
$
```

1286 je ovdje PID procesa *ispell*. Naredbeni redak *\$* vam pokazuje da je ljuska spremna obraditi vašu sljedeću naredbu. Premda *ispell* može pokrenuti neke potomke i čekati dok se oni ne završe, vi ne trebate čekati. U ovom je primjeru trenutna ljuska roditelj *ispella*, a potomci *ispella* mogu biti smatrani unucima prijavne ljuske. Premda roditelj može čekati svoju djecu, baka ih ne čeka.

Svi ovi primjeri pokazuju kako korisnici mogu pokrenuti višestruke procese. Možete odlučiti da li ćete prije nastavka rada pričekati završetak procesa *djeteta* ili nećete. Ako nastavite bez čekanja na završetak procesa *djeteta*, time potomke činite pozadinskim procesima. U sljedećem je odjeljku osvrt na neke Linux naredbe koje možete upotrijebiti za sastavljanje rasporeda procesa koji će se izvršavati u točno određena vremena ili pri nižem relativnom\_prvenstvu.

## Uporaba naredbi vremenskog raspoređivanja zadataka

Linux okružje pruža mnogo načina za baratanje izvršenjem naredbe. Linux vam omogućuje stvaranje popisa naredbi i određivanje kada će one biti izvršene. Naredba *at*, na primjer, uzima popis naredbi utipkanih na tipkovnici ili iz datoteke i izvršava ih u vrijeme određeno
naredbom. Naredba *batch* je slična naredbi *at*, osim što *batch* pokreće naredbe kada sustav nađe vremena za njih umjesto da korisniku dopusti određivanje vremena njihovog pokretanja. Naredba *cron* omogućuje periodično izvršavanje naredbi, a naredba *crontab* korisniku omogućuje uređivanje datoteka koje koristi *cron*.

Sve su ove naredbe raspoređivanja korisne za pokretanje zadaća u vremenima kada sustav nije prezauzet. One su dobre i za izvršavanje skripti prema vanjskim uslugama - poput upita u bazama podataka - u vremenima kada se to može najjeftinije učiniti.

#### Pokretanje naredbi u točno određena vremena s at

Za raspoređivanje jedne ili više naredbi za točno određeno vrijeme koristite naredbu *at*. S tom naredbom možete odrediti vrijeme, datum ili obje vrijednosti. Naredba očekuje dva ili više argumenta. Kao minimum, određujete vrijeme kada želite izvršenje (jedne ili više) naredbi i naredbu (ili više njih) koju želite izvršiti.

U sljedećem se primjeru zadani posao obavlja u 01,23 h. Ako radite u sitne jutarnje sate, prije 1,23 h ujutro (to jest, između ponoći i 01,23 h), naredba se izvršava danas, u 01,23 h. Inače će biti izvršena u 01,23 h sljedećeg dana. Ovaj posao ispisuje sve datoteke u direktoriju /usr/sales/reports i korisniku nazvanom *boss* šalje neku poštu objavljujući da je posao ispisivanja izvršen u 01,23 h. Na terminalu upišite sljedeće naredbe, s tim da na kraju svakog reda pritisnete <Return>. Nakon što unesete svaki redak, za dovršenje naredbe pritisnite <Ctrl-d>.

```
at 1:23
lp /usr/sales/reports/*
echo "Files printed, Boss!" ¦ mail -s"Job done" boss
```

BILJEŠKA cron poslovi, o kojima se govori kasnije u ovom poglavlju, su najčešće korišteni

mehanizmi za izvršavanje automatiziranih poslova održavanja sustava pod Linuxom. Međutim, za stvaranje i uređivanje stavki *cron* poslova trebate biti superkorisnik. Naredba *at* svakome omogućuje pokretanje zadaća, čak i ako on ili ona nemaju povlastice superkorisnika.

Vidi "Podešavanje terminalskog okružja", str. 341

Naredbe koje treba staviti u raspored s pomoću *at* se upisuju kao popis naredbi, u istom redu iza naredbe *at*.

Nakon što naredbu at završite, ugledat ćete prikaz sličan sljedećem:

job 756603300.a at Tues Jan 21 01:23:00 1997

Ovaj odaziv pokazuje da će posao biti izvršen u 01:23 h, kako je određeno. Broj posla, 756603300.a označava posao. Ako taj posao odlučite zaustaviti, to učinite korištenjem pridruženog broja posla, poput:

at -d 756603300.a

Ako uporabom naredbe *at* želite rasporediti nekoliko naredbi, najbolje je staviti ih u datoteku. Ako je ime datoteke, na primjer, *getdone* i te naredbe želite rasporediti za 10,00 h, utipkajte ili

at 10:00 < getdone

ili at 10:00 -f getdone

Upamtite da znak manje-od (<) ukazuje na uporabu sadržaja datoteke getdone kao ulazne podatke za naredbu *at*. Opcija *-f* vam omogućuje navođenje imena datoteke s naredbama bez uporabe preusmjeravanja.

Za posao možete odrediti i datum. Na primjer, za raspoređivanje posla u 17,00 h, 24. siječnja, unesite ove naredbe:

```
at 17:00 Jan 24
lp /usr/sales/reports/*
echo "Files printed, Boss!" ¦ mail -s"Job done" boss
```

Poslovi čiji raspored napravite s *at* stavljaju se u red čekanja koji operativni sustav periodično provjerava. Vi ne trebate biti prijavljeni da bi posao bio izvršen. Naredba *at* se uvijek izvršava u pozadini, oslobađajući resurse, međutim ona ipak obavi posao. Svaki se izlaz koji je proizveden s naredbama u vašem *at* poslu automatski poštom šalje vama.

Da biste vidjeli koje ste poslove rasporedili s *at*, unesite *at -l*. Radeći s prethodnim primjerima, ugledat ćete sljedeće rezultate:

```
job 756603300.a at Sat Dec 21 01:23:00 1996
job 756604200.a at Fri Jan 24 17:00:00 1997
```

Ispisani su samo at poslovi.

Za uklanjanje *at* posla iz rasporeda, unesite *at -d* i iza toga broj posla. Na primjer, za uklanjanje drugog posla, koji je upravo prikazan na popisu, unesite ovu naredbu:

```
at -d 756604200.a
```

Tablica 19.3 ukratko prikazuje različite načine uporabe naredbe at.

Tablica 19.3 Sažet pregled at naredbi	
Format naredbe	Radnja
at <i>hh:mm</i>	Raspoređuje posao u navedeni sat (hh) i minute (mm), koristeći 24-satno vrijeme
at hh:mm month day year	Raspoređuje posao u navedeni sat (hh), minute (mm), mjesec, dan i godinu
at -l	Prikazuje popis raspoređenih poslova; to je zamjenski format naredbe atq
at now + <i>broj vremenskih jedinica</i>	Raspoređuje posao odmah sada plus broj vremenskih jedinica za odbrojavanje. Vremenske jedinice mogu biti minute, sati, dani ili tjedni
at -d <i>id_posla</i>	Poništava posao s brojem posla jednakim id_posla; koristi se umjesto naredbe atrm

Kao root korisnik možete koristiti sve ove naredbe. Za druge korisnike datoteke /etc/at.allow i /etc/at.deny određuju dopuštenje za uporabu ovih naredbi. Ako /etc/at.allow postoji, uporaba naredbe at se dopušta samo korisničkim imenima koja su popisana u toj datoteci. Ako datoteka /etc/at.allow ne postoji, sustav provjerava /etc/at.deny i svakom je korisničkom imenu koje nije spomenuto u /etc/at.deny dopuštena uporaba at (drugim riječima, korisnicima navedenim na popisu u /etc/at.deny nije dopuštena uporaba at).

Ako niti jedna od datoteka ne postoji, samo root korisnik (superuser) može koristiti at. Ako je /etc/at.deny prazna, svaki korisnik može koristiti at.

#### Pokretanje dugih zadaća s batch

Linux ima više naredbi za raspoređivanje zadaća. U prethodnom je članku opisana naredba at pomoću koje možete propisati kada će se zadaća izvršiti. Međutim, uvijek postoji mogućnost da sustav bude opterećen s više poslova raspoređenih za isto vrijeme s kojima onda ne bi mogao udobno baratati. Naredba batch operativnom sustavu omogućuje da sam odluči o odgvarajućem vremenu za pokretanje procesa. Kada posao rasporedite pomoću naredbe *batch*, Linux će proces pokrenuti i raditi na njemu kada opterećenje sustava nije preveliko. Poslovi koji rade pod batch izvršavaju se u pozadini, baš poput onih koji se pokreću s at. U stvari, batch je drugo ime za at b u Red Hat Linuxu.

SAVJET Korisno je naredbe koje želite pokrenuti s at ili batch staviti u datoteku tako da naredbe ne morate ponovno utipkati svaki put kada te poslove želite pokrenuti. Ako za raspoređivanje naredbi u datoteku getdone želite upotrijebiti batch, unesite naredbu batch < getdone.

Oblik za *batch* naredbe jest unošenje popisa naredbi u retke koji slijede iza naredbe *batch*. Popis naredbi zatvarate s <Ctrl-d>. Popis naredbi možete staviti u datoteku i zatim ulaz datoteke preusmjeriti na batch. Za razvrstavanje zbirke datoteka, ispis rezultata i obaviještavanje korisnika nazvanog boss da je posao završen unesite sljedeće naredbe:

```
batch
sort /usr/sales/reports/* | lp
echo "Files printed, Boss!" | mailx -s"Job done" boss
```

Sustav vraća sljedeći odaziv:

job 7789001234.b at Fri Feb 21 11:43:09 1997

Ispisani datum i vrijeme su datum i vrijeme kada ste pritisnuli <Ctrl-d> za dovršenje naredbe batch. Kada je posao završen, provjerite vašu poštu; poštom vam je poslano sve što naredbe uobičajeno prikažu.

#### Rasporedivanje naredbi s cron i crontab

I at i batch raspoređuju naredbe na jedno-vremenskoj osnovi. Za raspoređivanje naredbi ili procesa na redovitoj osnovi koristite program cron. Vremena i datume u koje želite pokretati naredbu određujete u crontab datotekama. Vremena mogu biti određena u minutama, satima, danima u mjesecu, mjesecima u godini ili danima u tjednu.

Program cron je pokrenut samo jednom, tijekom podizanja sustava. Pojedinačni korisnici ne bi smjeli imati dopuštenje za izravno pokretanje crona. Također, kao administrator sustava, cron ne biste smjeli pokrenuti utipkavši ime naredbe; cron treba biti na popisu u skripti ljuske, kao jedna od naredbi koja se pokreće tijekom podizanja sustava.

Kada je pokrenut, *cron* (kratica za *chronograph*) provjerava redove čekanja za izvršavanje *at* poslova i također provjerava da li su korisnici ili *root* poslove rasporedili uporabom crontab datoteka. Ako ništa ne radi, *cron* "ide na spavanje" i postaje neaktivan. Međutim, "budi se" svake minute da provjeri ima li naredbi koje treba pokrenuti.

Možete vidjeti koliko je ovo sredstvo korisno i važno; osim toga, *cron* koristi vrlo malo resursa sustava.

*crontab* koristite za instalaciju popisa naredbi koje će se izvršavati u redovitom vremenskom\_rasporedu. Naredbe su raspoređene za pokretanje u određeno vrijeme (poput jednom mjesečno, svaki sat, jednom dnevno itd.). Popis naredbi koje trebaju biti izvršene po zadanom rasporedu mora biti uključen u crontab datoteku, koja je instalirana s naredbom *crontab*. Nakon što crontab datoteku instalirate, *cron* u zadana vremena učitava i izvršava naredbe u popisu. S *crontab* naredbom možete i pregledati popis naredbi uključenih u datoteku i, ako želite, možete ga poništiti.

Prije nego s naredbom *crontab* instalirate vašu crontab datoteku, uporabom tekstualnog editora poput *vi* ili *emacs* stvorite datoteku koja sadrži popis naredbi koje želite staviti u raspored. Naredba *crontab* barata s postavljanjem datoteke. Svaki korisnik ima samo jednu crontab datoteku, stvorenu kada je izdana naredba *crontab*. Ta je datoteka stavljena u direktorij iz kojeg naredba *cron* učitava.

Linux korisničku crontab datoteku sprema u direktorij /usr/spool/cron/crontabs i datoteci daje korisničko ime. Ako je vaše korisničko ime mcn i pomoću tekst editora stvorite datoteku koju nazovete mycron i instalirate ju utipkavši *crontab mycron*, bit će stvorena datoteka /usr/spool/cron/crontabs/mcn. (U ovom je primjeru datoteka mcn stvorena, ili ponovno upisana, sa sadržajem datoteke mycron, koja može sadržavati stavke koje pokreću jednu ili više naredbi.)

Da bi korisnici mogli koristiti naredbu crontab, trebaju biti naveden na popisu u datoteci /etc/cron.d/cron.allow. Ako korisnika u sustav dodate preko naredbenog retka (uporabom naredbe

BILJEŠKA useradd), on ili ona neće biti automatski dodani u datoteku /etc/cron.d/cron.allow.

Kao root korisnik, novog korisnika u datoteku cron.allow trebate dodati s tekstualnim editorom.

Premda vašu crontab datoteku na početku možete stvoriti s tekstualnim editorom, nakon što u jednom stvorite, preinačavate je samo uporabom naredbe *crontab*. Datoteku koju *cron* pregledava (dakle, datoteku /usr/spool/cron/crontabs/user) nemojte pokušavati zamijeniti ili preinačiti nikakvim drugim sredstvom osim uporabe naredbe *crontab*.

Svaki redak u crontab datoteci sadrži vremenski obrazac i naredbu. Naredba se izvršava u zadanom vremenskom obrascu. Vremenski je obrazac podijeljen na pet polja odvojenih razmacima ili tabulatorima. Svaki izlaz koji se obično pojavi - to jest, informacija koja nije preusmjerena u *stdout* ili *stderr* - je poštom poslan korisniku.

U sljedećem je redu prikazana sintaksa za naredbe koje unosite u datoteku koju će koristiti *crontab*:

minuta sat dan-u-mjesecu mjesec-u-godini dan-u tjednu naredba

Prvih pet polja su polja vremenske opcije. Morate odrediti svih pet polja. Ako neko polje želite zanemariti, u tom polju upotrijebite zvijezdicu (\*).

**BILJEŠKA** Tehnički, zvjezdica u crontab polju znači "svaka valjana vrijednost", a ne "zanemari tu vrijednost" - znači da bilo što odgovara. Na primjer, crontab stavka 02 00 01 \* \* date kaže da naredbu *date* treba pokrenuti u dvije minute nakon ponoći, prvog dana u mjesecu. Budući da su u polja *mj*esec i *dan u tjednu* stavljene zvjezdice, ova se stavka izvršava prvog dana svakog mjeseca i u bilo koji dan u tjednu na koji slučajno pada prvi dan tog mjeseca.

Tablica 19.4 prikazuje popis opcija vremenskih polja raspoloživih s naredbom crontab.

Tablica 19.4 Op	cije vremenskih polja za naredbu crontab
Polje	Raspon
minuta	00 do 59
sat	00 do 23 (ponoć je 00)
dan-u-mjesecu	01 do 31
mjesec-u-godini	01 do 12
dan-u-tjednu	01 do 07 (ponedjeljak je 01, nedjelja je 07)

U crontab datoteci možete imati koliko god želite stavaka i možete ih odrediti za pokretanje u bilo koje vrijeme. To znači da u jednoj crontab datoteci možete pokrenuti koliko god želite naredbi.

Za razvrstavanje datoteke nazvane /usr/wwr/sales/weekly i slanje izlaza poštom korisniku nazvanom twool, u 7,30 h svakog ponedjeljka, u datoteku stavite sljedeću stavku:

30 07 \* \* 01 sort /usr/wwr/sales/weekly {mail -s"Weekly Sales" twool

Ova naredba navodi minute kao 30, sat kao 07, zvjezdicu za bilo koji dan u mjesecu, drugu zvjezdicu za bilo koji mjesec u godini i dan-u-tjednu kao 01 (što predstavlja ponedjeljak).

Uočite cijev između naredbi *sort* i *mail* u prethodnom primjeru. Naredbeno polje može sadržavati cijevi, točka-zareze, strelice i sve drugo što možete unijeti u naredbeni redak ljuske. U navedeni datum i vrijeme, *cron* izvršava cijelo naredbeno polje pomoću standardne ljuske (*bash*).

Pri određivanju slijeda vrijednosti za jedno od prva četiri polja za razdvajanje vrijednosti koristite zareze. Pretpostavimo da imate program, *chkquotes*, koji pristupa servisu što pruža cijene dionica i te vrijednosti stavlja u datoteku. Za dobivanje tih cijena u 09 h, 11 h, 14 h i 16 h svakog tjedna u ponedjeljak, utorak i četvrtak - i svakako 10. ožujka i 10. rujna - upotrijebite sljedeću stavku:

\* 09, 11, 14, 16 10 03,09 01,02,04 chkquotes

Naredbene retke u datoteku stavljajte uporabom *vi* ili nekog drugog editora koji vam dopušta spremanje datoteka kao tekstualnih datoteka. Pretpostavimo da svoje naredbe stavljate u datoteku nazvanu *cronjobs*. Za uporabu naredbe *crontab* za stavljanje datoteke tamo gdje će ju *cron* pronaći unesite ovu naredbu:

crontab cronjobs

Svaki put kada *crontab* koristite na ovaj način, ona će tu datoteku upisati preko bilo koje crontab datoteke koju ste možda već pokrenuli.

Naredba crontab ima tri opcije:

- Opcija -e uređuje sadržaj crontab datoteke. (Opcija -e vašu datoteku otvara uporabom editora ed ili nekim drugim editorom ako je taj dodijeljen varijabli EDITOR u vašoj ljusci).
- Vidi "Podešavanje okružja ljuske", str. 344
  - Opcija -*r* trenutnu crontab datoteku uklanja iz crontab direktorija.
  - Opcija -*l* pravi popis sadržaja trenutne crontab datoteke.

U svim ovim slučajevima *crontab* radi s crontab datotekom koja ima vaše prijavno ime. Ako je vaše prijavno ime mcn, vaša je crontab datoteka /usr/spool/cron/crontabs/mcn. Naredba *crontab* ovo radi automatski.

I administrator sustava i korisnici su odgovorni osigurati da je sustav odgovarajuće korišten. Kada raspoređujete neki proces, budite svjesni utjecaja koji on može imati na cjelokupan sustav. Linux vam, kao administratoru sustava, dopušta odobravanje pristupa do naredbi *at, batch* i *cron* svim korisnicima, određenim korisnicima ili niti jednom korisniku (ili ne dopušta pristup pojedinim korisnicima).

#### RJEŠAVANJE POTEŠKOĆA

**Naredbe koje stavljam u svoju crontab datoteku ne rade.** Naredba *cron* vaše crontab stavke pokreće uporabom ljuske Bourne Again (*bash*). Vaše stavke ne uspijevaju ako koristite osobine ljuske koje *bash* ne podržava. Na primjer, ljuska Public Domain Korn (*pdksh*) vam omogućuje ili uporabu tilde (~) za predstavljanje početnog direktorija ili naredbe alias za određivanje zamjenskih imena za neke naredbe.

Kada pokušam upotrijebiti naredbu at, obaviješten sam da mi njena uporaba nije dopuštena. U datoteku /etc/cron.d/at.allow niste dodali svoj prijavni ID (identifikacijski broj).

**Naredbu at sam pokušao upotrijebiti za trenutno pokretanje naredbe.** Bez obzira koliko brzo tipkali, at se sada uvijek odaziva s porukom *ERROR: Too late.* U ovom je slučaju najbolje upotrijebiti naredbu batch, koja će naredbu pokrenuti umjesto vas. Možete, međutim, upotrijebiti i *at now* +5 *min* za pokretanje naredbe nakon 5 minuta. Nakon što pritisnete <Return>, brzo tipkajte tako da naredbu unesete prije isteka pet minuta.

## Nadziranje višezadaćnog okružja i izvještavanje o trenutnom stanju

Poznato vam je da je Linux višekorisnički, višezadaćni operativni sustav. Budući da s ovim sustavom toliko mnogo ljudi može istovremeno uraditi toliko mnogo poslova, korisnici smatraju da je korisno utvrditi tko koristi sustav i koji se procesi izvršavaju, kao i nadziranje procesa. Važno je znati da drugi mogu pratiti koje naredbe unosite. Većina korisnika ne može pristupiti vašim datotekama bez vašeg dopuštenja, međutim oni mogu vidjeti imena naredbi koje unosite. Također, vi (kao administrator sustava) ili netko drugi tko ima *root* lozinku može pomno pregledati sve datoteke u sustavu.

Premda ne trebate imati osjećaj proganjanja zbog privatnosti na Linux sustavu, trebate znati da sustav može biti nadziran od svakog tko na to želi potrošiti vrijeme. Informacije koje možete prikupiti o tome što se događa u sustavu su korisnije od samog zadovoljavanja znatiželje. Vidjevši koji se poslovi izvršavaju, možete odgovarajuće rasporediti svoje zadaće. Također možete vidjeti i da li je vaš proces još uvijek aktivan te da li se ispravno ponaša.

#### Pronalaženje tko je na sustavu s who

Svrha je naredbe *who* otkrivanje tko je prijavljen na sustav. Naredba *who* pruža popis prijavnih imena, linija terminala i prijavnih vremena trenutno prijavljenih korisnika.

Naredba *who* je u mnogim prilikama korisna. Ako, na primjer, uporabom naredbe *write* želite stupiti u vezu s nekim za računalom, s naredbom *who* možete otkriti da li je ta osoba prijavljena na sustavu. Naredbu *who* možete upotrijebiti i da saznate kada su određeni korisnici prijavljeni na računalo radi praćenja njihovog vremena potrošenog na sustavu.

**Uporaba naredbe who za prikaz popisa korisnika prijavljenih na sustav** Da biste vidjeli tko je sve trenutno prijavljen na sustav, unestite *who*. Ugledat ćete prikaz sličan ovome:

\$ who				
root	console	Dec	13	08:00
ernie	tty02	Dec	13	10:37
bkraft	tty03	Dec	13	11:02
jdurum	tty05	Dec	13	09:21
ernie	ttys7	Dec	11	18:49
\$				

Ovaj popis prikazuje da su trenutno prijavljeni root, ernie, bkraft i jdurum. Prikazuje da se root prijavio u 08,00 h, bkraft u 11,02 h i jdurum u 09,21 h. Možete vidjeti i da je ernie prijavljen na dva terminala i da se na jednog prijavio dva dana ranije, u 18,49 h (što bi moglo dati povoda za zabrinutost ili su to možda samo ernijeve uobičajene radne navike).

**Uporaba zaglavlja u popisima korisnika** S *who* imamo nekoliko opcija na raspolaganju, međutim ovo poglavlje opisuje uporabu samo dvije za praćenje procesa u sustavu:

- U Prikazuje samo korisnike koji su sada prijavljeni
- H Prikazuje zaglavlje iznad svakog stupca

S ovim dvjema opcijama možete dobiti više informacija o trenutno prijavljenim korisnicima. Zaglavlja stupaca prikazana s opcijom -H su NAME, TIME, IDLE, PID i COMMENTS. U tablici 19.5 su objašnjenja pojmova koji se pojavljuju kao zaglavlja stupaca.

Tablica 19.5	Izlazni oblik za naredbu who
Polje	Opis
NAME	Popis korisničkih prijavnih imena.
LINE	Popis korištenih linija terminala.
TIME	Popis vremena prijavljivanja korisnika.
IDLE	Popis sati i minuta od posljedne aktivnosti na toj liniji. Točka je prikazana ako se aktivnost dogodila unutar zadnje minute sustavskog vremena. Ako je od posljednjeg korištenja lnije prošlo više od 24 sata, prikazana je riječ old.
PID	Popis identifikacijskog (ID) broja korisničke prijavne ljuske.
COMMENT	Prikaz sadržaja polja COMMENT ako su u /etc/inittab bile uključene napomene ili ako ima mrežnih veza.

**BILJEŠKA** Polje COMMENT u bilo kojem od nedavnih Linux sustava vjerojatno nećete često vidjeti ispunjeno. U starim su danima procesi koji su vam dopuštali prijavu na UNIX (getty ili *uugetty*) bili pokretani izravno iz stavki u datoteci /etc/inittab i obično su osluškivali prijavne zahtjeve s određenog terminala. Polje COMMENT je moglo prepoznati lokaciju tog terminala i moglo vam e prikazati koji su korisnici prijavljeni i za kojim su terminalima sjedili. Danas s procesima koji osluškuju prijavne zahtjeve obično barata Service Access Facility i oni više nisu na popisu u /etc/inittab.

Sljedeći primjer koristi opcije -u i -H i prikazuje odaziv vraćen od Linuxa:

\$ who -ul	Η				
NAME	LINE	TIME	IDLE	PID	COMMENT
root	console	Dec 13 08:00		10340	
ernie	tty02	Dec 13 10:37		11929	Tech-89.2
bkraft	tty03	Dec 13 11:02	0:04	4761	Sales-23.4
jdurum	tty05	Dec 13 09:21	1:07	10426	
ernie	ttys7	Dec 11 18:49	old	10770	oreo.coolt.com
\$					

Iz ovog popisa možete zaključiti da je posljednja *sesija* povezana s imenom ernie stigla s mrežne lokacije nazvane oreo.coolt.com i da na njoj više od 24 sata nije bilo nikakve aktivnosti (što može ukazivati na poteškoće). Na *sesijama* za *root* i prvoj za ernie je bilo pristupa unutar posljednje minute. Posljednja je aktivnost na *sesiji* za bkraft bila prije četri minute, a od poljednje je izvještene aktivnosti na *sesiji* za jdurum prošlo jedan sat i sedam minuta.

Također uočite da ovaj popis sadžava i PID (identifikacijski broj procesa) za prijavnu ljusku svake korisničke *sesije*. Sljedeći članak prikazuje kako PID možete koristiti za daljnje nadziranje sustava.

**Uporaba naredbe finger da saznate više o tome tko je na sustavu** Naredba *finger* služi kao dopuna naredbi *who*. Ako želite vidjeti više informacija o određenom korisniku, možete unijeti *finger imekorisnika* (ili *finger imekorisnika@domena* ako je korisnik na

drugom računalu). Na primjer, za više informacija o korisniku nazvanom tackett, unijet ćete ovu naredbu:

finger tackett

Zatim ćete ugledati sljedeći izlaz:

```
Login: tackett Name: Jack Tackett Jr

Directory: /home/tackett Shell: /bin/tcsh

Office: 2440 SW Cary Parkway 114 Office Phone: 919 555 1212

Home Phone: 919 555 1212

Never logged in.

Mail last read Fri Jul 3 17:42 1998 (EDT)

Plan:

Jack Tackett, Jr.

In the immortal words of Socrates:

I drank WHAT?
```

Ovaj izlaz prikazuje prijavno i stvarno ime pridruženo određenom korisničkom računu. Možete vidjeti i koju ljusku korisnik najradije koristi, njegovu adresu, kada je posljednji put pročitao svoju elektroničku poštu i kada je posljednji put bio prijavljen na sustav. Ako je on i trenutno prijavljen, naredba *finger* vam kazuje koliko je dugo bio prijavljen te koji program trenutno koristi. Naredba *finger* prikazuje i sve informacije koje je korisnik možda stavio u svoju datoteku plan u svom početnom direktoriju.

Kao što možete vidjeti, naredba *finger* prikazuje mnogo informacija o korisniku, što hakeri mogu iskoristiti za razbijanje sustava. To je razlog zbog kojeg mnogi administratori sustava onemogućuju naredbu *finger* tako da drugi ove informacije ne mogu vidjeti.

**BILJEŠKA** Ako na vašem sustavu dopustite uporabu naredbe **finger**, ili ako ju vaš administrator sustava dopusti na sustavu koji vi koristite, možete koristiti naredbu *chfn* s kojom mijenjate informacije prikazane s *finger*. O više informacija o ovome možete pogledati odgovarajuću *man* stranicu (koristite naredbu *man chfn*).

#### Izvještavanje o stanju procesa s ps

Naredba *ps* (process status) izvještava o stanju procesa. Možete je koristiti za utvrđivanje koji se procesi izvršavaju, da li je proces završen, da li je proces zastao ili ima nekih poteškoća, koliko se dugo proces izvršavao, koje resurse proces koristi, relativno prvenstvo procesa te PID (identi-fikacijski broj procesa) potreban prije nego proces možete zaustaviti. Sve su te informacije korisne korisniku i vrlo korisne administratoru sustava. Bez ikakvih opcija *ps* prikazuje PID svakog procesa pridruženog vašoj trenutnoj ljusci. Moguće je vidjeti i detaljan popis svih procesa koji se izvršavaju u sustavu.

**Nadziranje procesa s ps** Uobičajena je uporaba naredbe *ps* nadziranje pozadinskih poslova i drugih procesa u sustavu. Budući da pozadinski procesi u većini slučajeva nisu u vezi s vašim zaslonom i tipkovnicom, za praćenje njihovog napretka koristite *ps*.

*ps* popisi prikazuju četiri unaprijed zadana zaglavlja kao pokazatelje informacija u poljima ispod svakog zaglavlja: *PID*, *TTY*, *TIME*, i *COMMAND*. Ova su zaglavlja objašnjena u talbici 19.6.

Tablica 19.6	Zaglavlja u izlazu naredbe ps
Polje	Objašnjenje
PID	Identifikacijski broj procesa.
ТТҮ	Terminal s kojeg proces potječe.
TIME	Zbirno vrijeme izvršavanja procesa, u minutama i sekundama.
COMMAND	Ime naredbe koja se izvršava.

Pretpostavimo da želite razvrstati datoteku nazvanu sales.dat, kopiju razvrstane datoteke spremiti u datoteku nazvanu sales.srt i razvrstanu datoteku poštom poslati korisnici sarah. Ako ovaj posao također želite staviti u pozadinu, unesite sljedeću naredbu:

sort sales.dat { tee sales.srt { mailx -s"Sorted Sales Data" sarah &

Za praćenje ovog procesa unesite ps i ugledat ćete prikaz sličan ovom:

PID	TTY	TIME	COMMAND
16490	tty02	0:15	sort
16489	tty02	0:00	mailx
16492	tty02	0:00	ps
16478	tty02	0:00	bash
16491	tty02	0:06	tee
16480	tty02	96:45	cruncher

Za svaki proces pokrenut naredbom vidite nakupljeno vrijeme i njegov PID. Vidite i informaciju za vašu prijavnu ljusku (*bash*) i samu *ps.* Uočite da se sve informacije u cijevi izvršavaju istovremeno, baš kao što biste i očekivali (to je način rada procesa s cjevovodom). Posljednja je stavka za naredbu koja se izvršavala dulje od sat i pola. Ako je to problem, možda ćete proces željeti zaustaviti uporabom naredbe *kill* (opisane kasnije u ovom poglavlju). Ako unesete *ps* i ugledate samo sljedeći popis, prethodni je posao koji ste stavili u pozadinu već završen:

```
        PID
        TTY
        TIME
        COMMAND

        16492
        tty02
        0:00
        ps

        16478
        tty02
        0:00
        bash

        16480
        tty02 99:45
        cruncher
```

#### BILJEŠKA

*p*s koristite povremeno za provjeru stanja naredbe. Međutim, ako *p*s koristite svaki čas radi provjere da li je pozadinski posao završen, tada taj posao uopće nema smisla stavljati u pozadinu.

**Uporaba ps za dobivanje više informacija o procesima** Ponekad o vašim procesima trebate znati više nego što unaprijed zadani *ps* popisi pružaju. Za stvaranje dodatnih informacija možete prizvati neke od zastavica s popisa u tablici 19.7.

Tablica 19.7	Cesto koristene zastavice za naredbu ps
Zastavica	Opis
- a	Prikazuje i procese drugih korisnika.
- C	Prikazuje ime naredbe iz okružja task_struct.
- e	Prikazuje okružje nakon naredbenog retka i "and".
- f	Prikazuje "forest" oblik obiteljskog stabla (procesi i potprocesi).
- h	Izostavljanje zaglavlja.
- j	Jobs format.
- 1	Long format.
- m	Prikazuje podatke o memoriji.
- n	Brojčani (numeric) izlaz za USER i WCHAN. WCHAN je ime za funkciju jezgre (kernela) gdje proces spava, s tim da je iz imena funkcije skinuto sys Ako /etc/psdatabase ne postoji, broj je umjesto toga heksadecimalan.
- r	Samo procesi koji se izvršavaju.
- S	Signal format.
- S	Child procesu dodaje vrijeme i greške stranice.
-txx	Samo procesi povezani s ttyxx.
- U	User format. Daje korisničko ime i početno vrijeme.
- V	vm (virtual memory) format.
- W	Wide (široki) izlaz. Naredbene retke ne podrezuje da bi stali u jedan red.
- X	Prikazuje procese bez upravljanja terminalom.

Naredba *ps* daje samo približnu sliku stanja procesa jer se stanje može izmijeniti i mijenja se za vrijeme njenog izvršavanja. Naredba *ps* pruža trenutnu sliku stanja procesa u trenutku njenog izvršavanja. Trenutna slika uključuje i samu naredbu *ps*.

Sljedeći primjeri prikazuju tri naredbe. Prva je naredba prijavna ljuska (*bash*). Druga je naredba *sort*, koja se koristi za razvrstavanje datoteke nazvane *inventory*. Treća je naredba *ps* koju sada izvršavate.

Za otkrivanje koje procese trenutno izvršavate upotrijebite sljedeću naredbu:

\$ps			
ΡID	ΤΤΥ	TIME	COMMAND
65	tty01	0:07	-bash
71	tty01	0:14	sort inventory
231	tty01	0:09	ps

Za dobivanje cjelovitog popisa upotrijebite ovu naredbu:

\$ ps -u	ax						
UID	ΡID	PPID	С	STIME	ΤΤΥ	TIME	COMD
amanda	65	1	0	11:40:11	tty01	0:06	-bash
amanda	71	65	61	11:42:01	tty01	0:14	sort inventory
amanda	231	65	80	11:46:02	tty01	0:00	ps -f

Uočite nekoliko stvari u ovom punom popisu. Pored PID-a, ispisan je i PPID. PPID (parent process ID) je identifikacijski broj matičnog procesa tog procesa. U ovom je primjeru prvi proces na popisu, PID 65, roditelj sljedeća dva. Stavka u četvrtom stupcu (stupac s zaglavljem C) daje iznos procesorskog vremena koje je proces nedavno koristio. U odabiru sljedećeg procesa s kojim će raditi operativni sustav odabire proces s nižom C vrijednosti, dok oni s višom vrijednosti trebaju čekati. Stavka u stupcu *STIME* je vrijeme u koje je proces pokrenut.

Za nadziranje svakog procesa u sustavu i dobivanje cjelovitog popisa unesite *ps -uax*. Ucjevljivanjem te naredbe preko naredbe *grep \$LOGNAME*, bit će prikazani procesi koji pripadaju vašem prijavnom imenu dok će svi ostali biti isfiltrirani. Za prikaz punog popisa svih vaših procesa unesite ovo:

ps -uax ¦ grep \$LOGNAME

Za prikaz popisa procesa za dva terminala (na primjer, tty1 i tty2) koristite sljedeću naredbu:

\$ ps	-t "1 2	**	
ΡID	ΤΤΥ	TIME	COMMAND
32	tty01	0:05	bash
36	tty02	0:09	bash
235	tty02	0:16	vi calendar

U ovom je primjeru opcija -*t* korištena za ograničavanje popisa na procese pridružene terminalima tty01 i tty02. Terminal tty02 izvršava naredbu ljuske (PID 32) i koristi *vi* za uređivanje kalendara (PID 235). Prikazano je i zbirno vrijeme za svaki proces. Ako ljuske koristite iz grafičkog sučelja (naredba *xterm*), za prikaz vaših procesa na njima upotrijebite imena uređaja pts001, pts002 itd s opcijom -*t*.

Ponekad je proces označen kao < *defunct*>, što znači da je proces zaustavljen i da je njegov matični proces o tome obaviješten, međutim matični proces još nije potvrdio da je proces "mrtav". Takav se proces naziva *zombi proces*. Moguće je da je roditelj zauzet s nečim drugim i zombi će uskoro nestati. Ako ugledate više *defunct* procesa ili onih koji već neko vrijeme odugovlače, to je znak neke poteškoće s operativnim sustavom.

## BILJEŠKA Budući da zombi proces nema roditelja, zombi ne možete zaustaviti. Jedini je način za uklanjanje zombi procesa ponovno podizanje sustava vašeg stroja.

## Nadzor nad većim brojem procesa

Linux vam pruža moć za usporedno pokretanje nekoliko procesa. Također korisniku ili administratoru omogućuje nadzor nad procesima koji se izvršavaju. Ovaj je nadzor koristan kada trebate uraditi sljedeće:

- Pokrenuti proces koji se nastavlja i nakon što se njegov roditelj prestane izvršavati (upotrijebite naredbu nohup)
- Staviti u raspored proces s prvenstvom različitim od ostalih procesa (koristite naredbu *nice*)
- Zaustaviti proces (upotrijebite naredbu kill)

#### Uporaba nohup s pozadinskim procesima

U slučaju zaustavljanja matičnog procesa obično se zaustave i potomci tog procesa. To znači da, kada pokrenete pozadinski proces, on se pri vašem odjavljivanju zaustavlja. Za nastavak rada procesa i nakon vaše odjave koristite naredbu nohup - stavite je na početak naredbenog retka:

```
nohup sort sales.dat &
```

Ovaj primjer naredbe kaže naredbi sort da zanemari činjenicu da ste se vi odjavili sa sustava; ona bi trebala nastaviti s radom sve dok proces ne završi. Na taj način možete pokrenuti proces koji će se izvršavati danima ili čak tjednima. Štoviše, za vrijeme njegovog rada vi ne trebate biti prijavljeni na sustav. Naravno, vi želite osigurati da se posao koji pokrenete ponaša uredno, to jest da se na kraju zaustavi i ne stvori preobilan izlaz.

Kada koristite *nohup*, ta naredba sav izlaz i poruke o greškama naredbe, koje se inače pojavljuju na zaslonu, šalje u datoteku nazvanu nohup.out. Razmotrite sljedeći primjer:

```
$ nohup sort sales.dat &
1252
Sending output to nohup.out
$
```

Razvrstana datoteka i sve poruke o greškama stavljene su u datoteku nohup.out. Sada razmotrite ovaj primjer:

```
$ nohup sort sales.dat > sales.srt &
1257
Sending output to nohup.out
```

Sve su poruke o greškama stavljene u datoteku nohup.out, ali je razvrstana datoteka sales.dat stavljena u sales.srt.

BILJEŠKA Kada nohup koristite s cjevovodom, nohup trebate koristiti sa svakom naredbom u cjevovodu:

nohup sort sales.dat ¦ nohup mailx -s"Sorted Sales Data" boss & 💻

#### Raspoređivanje prvenstva naredbi s nice

Naredbu nice koristite za pokretanje naredbe prema prvenstvu koje ste joj dodijelili u rasporedu. Naredba nice vam pruža određeni nadzor nad prvenstvom jednog prema drugom poslu. Ako nice ne koristite, procesi se pokreću prema postavljenom prvenstvu. Pomoću naredbe nice možete prvenstvo procesa spustiti tako da drugi procesi mogu biti raspoređeni za češću uporabu procesora nego nice posao. Superuser (osoba koja se može prijaviti kao root korisnik) može prvenstvo procesa i podići.

BILJEŠKA Naredbe nice —help i nice —version ne rade u GNU primjeni naredbe nice.

Opći je oblik naredbe nice sljedeći:

nice -broj naredba

Razina je prvenstva utvrđena argumentom *broj* (veći broj znači niže prvenstvo). Unaprijed je postavljen broj 10, a *broj* je otklon od unaprijed podešene vrijednosti. Ako je argument *broj* prisutan, prvenstvo je povećano za taj iznos do najviše ukupno 20. Ako unesete sljedeću naredbu, proces *sort* se pokreće s prvenstvom 10:

```
sort sales.dat > sales.srt &
```

Ako želite pokrenuti drugi proces - recimo naredbu *lp* -, ali želite da prvenstvo ostane naredbi *sort*, možete unijeti sljedeće:

nice -5 lp mail\_list &

Ako naredbi lp želite dati najniže moguće prvenstvo, unesite ovo:

nice -10 lp mail\_list &

BILJEŠKA Brojčanoj zastavici iznad prethodi znak "-" koji određuje zastavicu, što ne biste smjeli pomiješati s oznakom negativnog broja.

Samo *root* korisnici mogu povećati prvenstvo procesa. Oni za to koriste negativni broj kao argument za *nice*. Upamtite - što je vrijednost *nice* niža, to je više prvenstvo (do navišeg prvenstva od 20). Kada poslu želi dati najviše prvenstvo, *root* korisnik taj posao pokreće ovako:

nice -10 job &

Znak ampersand (&) nije obavezan. Ako je posao interaktivan, za stavljanje procesa u pozadinu nećete koristiti ampersand.

#### Raspoređivanje prvenstva procesa u izvršavanju s renice

Naredba *renice*, raspoloživa na nekim sustavima, omogućuje vam preinačavanje prvenstva procesa koji se izvršava. Berkeley UNIX sustavi imaju naredbu *renice*. Ona je raspoloživa i u direktoriju /usr/ucb sustava Linux System V radi usklađenosti s Berkeley sustavima. S *renice* možete podešavati prvenstva naredbi tijekom njihovog izvršavanja. Oblik za *renice* jest sličan onom od *nice*:

renice -broj PID

Za promjenu prvenstva procesa u izvršavanju trebate znati njegov PID. Pomoću ove ćete naredbe pronaći PID-ove za sve vaše procese:

ps -e ¦ grep ime

U ovoj naredbi *ime* predstavlja ime procesa koji se izvršava. Naredba *grep* filtrira (izbacuje) sve procese koji ne sadrže ime procesa koji tražite. Ako se izvršava nekoliko procesa tog

imena, željeni ćete proces pronaći pogledavši njegovo vrijeme pokretanja. Ako želite djelovati na sve procese koji pripadaju određenoj skupini ili određenom korisniku, za naredbu *renice* možete navesti GID ili UID procesa koji se izvršavaju.

Stavka u drugom stupcu *ps* popisa jest PID procesa. U sljedećem se primjeru za trenutnog korisnika izvršavaju tri procesa (pored ljuske). Ime je trenutnog korisnika *pcoco*.

```
$ ps -ef ; grep $LOGNAME
pcoco 11805 11804 0 Dec 22 ttysb 0:01 sort sales.dat>sales.srt
pcoco 19955 19938 4 16:13:02 ttyp0 0:00 grep pcoco
pcoco 19938 1 0 16:11.04 ttyp0 0:00 bash
pcoco 19940 19938 142 16:11:04 ttyp0 0:33 find . -name core -exec rm {};
```

Za snižavanje prvenstva procesa s PID-om 19940 (proces find), unesite sljedeće:

renice -5 1994Ø

Kao što biste i očekivali, sljedeće su tvrdnje o renice točne:

- *renice* možete koristiti samo s procesima čiji ste vi vlasnik.
- Root korisnik može *renice* upotrijebiti na bilo koji proces.
- Samo root korisnik može povećati prvenstvo procesa.

#### Zaustavljanje procesa s kill

Proces ponekad trebate ili želite zaustaviti. Evo nekih razloga za zaustavljanje procesa:

- Koristi previše procesorskog vremena.
- Predugo se izvršava, a bez stvaranja očekivanog izlaza.
- Stvara previše izlaza na zaslonu ili u datoteku na disku.
- Cini se da je zamrznuo terminal ili neke druge *sesije*
- Koristi pogrešne datoteke za ulaz ili izlaz zbog greške operatora ili programske greške.
- Više nam nije koristan.

Vjerojatno ćete naići i na brojne druge razloge za zaustavljanje procesa. Ako je proces koji treba zaustaviti pozadinski proces, za izlazak iz takvih situacija koristite naredbu *kill*.

Za zaustavljanje naredbe koja nije u pozadini, pritisnite <Ctrl-c>. Međutim, kada je naredba u pozadini, nećemo je zaustaviti pritiskanjem tipke prekida. Budući da pozadinski proces nije pod nadzorom terminala, unos preko tipkovnice bilo kakve tipke prekida je zanemaren. Jedini je način za zaustavljanje naredbi u pozadini uporaba naredbe *kill*.

**Uobičajeno zaustavljanje pozadinskih procesa** Naredba *kill* šalje signale programu zahtijevajući da se proces zaustavi. Za uporabu naredbe *kill* koristite jedan od ovih oblika:

```
kill PID(ove)
```

kill -signal PID(ove)

ili

Za zaustavljanje procesa čiji je PID 123, unesite *kill 123*. Za zaustavljanje nekoliko procesa čiji su PID-ovi 123, 342 i 73, unesite *kill 123 342 73*.

Uporabom opcije *-signal* možete učiniti više nego samo zaustaviti proces. Drugi signali mogu prouzročiti da proces koji se izvršava ponovno učita konfiguracijske datoteke ili da se proces zaustavi, ali ne i poništi. Za prikaz popisa valjanih signala koristite naredbu *kill -l.* Prosječni će korisnik, međutim, vjerojatno koristiti samo *kill* bez ikakvog signala ili, u najboljem slučaju, sa signalom -9 (signal Ja-to-stvarno-mislim-pa-me-stoga-nemoj--zanemariti, opisan u sljedećem članku).

#### **UPOZORENJE**

Pazite da s naredbom kill koristite ispravan PID. Uporaba pogrešnog PID-a može zaustaviti proces koji ne želite zaustaviti. Upamtite da zaustavljanje pogrešnog procesa ili procesa sustava može imati poguban učinak. Upamtite i da u slučaju kada ste prijavljeni kao administrator sustava možete zaustaviti *svaki* proces.

Ako proces uspješno zaustavite, o tome nećete dobiti obavijest od ljuske, već će se samo pojaviti naredbeni redak ljuske. Ako pokušate zaustaviti proces koji niste ovlašteni zaustaviti ili ako pokušate zaustaviti proces koji ne postoji, ugledat ćete poruku o grešci.

Pretpostavimo da je vaše prijavno ime *chris* i da ste sada prijavljeni na tty01. Želite li pogledati koji se procesi izvršavaju, unesite *ps -f* i ugledat ćete sljedeći odaziv:

UID	ΡID	PPID	С	STIME	ΤΤΥ	TIME	COMMAND
chris	65	1	0	11:40:11	tty01	0:06	-bash
chris	71	65	61	11:42:01	tty01	0:14	total_updt
chris	231	65	80	11:46:02	tty01	0:00	ps -f
chris	187	53	60	15:32:01	tty02	123:45	crunch stats
chris	53	1	0	15:31:34	tty02	1:06	-bash

Uočite da se program *total\_updt* izvršava na vašem trenutnom terminalu. Drugi se program, *crunch*, izvršava na drugom terminalu i vi smatrate da je koristio neuobičajeno veliki iznos procesorskog vremena. Za zaustavljanje bi tog procesa moglo biti dovoljno da unesete *kill 187*. Za zaustavljanje roditelja tog procesa unesite *kill 53*.

Ako se prijavite kao administrator sustava i uočite da su neki terminali ostavljeni bez nadzora (ako ste Linux opremili s daljinskim terminalima), možda ćete htjeti zaustaviti i roditelja i potomka. Možete, dakle, zaustaviti vremenski raspoređeni proces (*child* proces) koji taj korisnik izvršava te njegovu prijavnu ljusku (matični proces), tako da zapostavljeni terminal više nije prijavljen.

Zaustavljanje matičnog procesa nekada zaustavi i *child* proces. Da biste bili sigurni, zaustavite roditelja i njegove potomke radi zaustavljanja svih radnji pridruženih matičnom procesu. Za zaustavljanje oba procesa u prethodnom primjeru, unesite *kill 187 53*.

Ako se vaš terminal zamrzne, prijavite se na drugi virtualni terminal, pritisnuvši kombinaciju Alt+funkcijska tipka (F1-F6), unesite *ps -ef ¦ grep \$LOGNAME* i zatim zaustavite prijavnu ljusku zamrznutog terminala.

**Bezuvjetno zaustavljanje pozadinskih procesa** Izdavanje naredbe *kill* procesu šalje signal. Linux programi mogu slati ili primati više od 20 signala, od kojih je svaki predstavljen s brojem. Na primjer, kada se odjavljujete, Linux svim pozadinskim procesima pokrenutim iz vaše prijavne ljuske šalje signal zaustavljanja (signal broj 1, hang-up).. Ovaj će signal te procese zaustaviti osim ako su pokrenuti s *nohup* (kako je opisano ranije u ovom poglavlju).

Uporabom *nohup* za pokretanje pozadinskog procesa omogućujemo procesu da se ne obazire na signal koji ga pokušava zaustaviti. Možete koristiti i programe ili skripte ljuske napisane za zanemarivanje nekih signala. Ako pri uporabi naredbe *kill* ne navedete neki određeni signal, procesu će biti poslan signal 15. Naredba *kill 1234* šalje signal 15 procesu čiji je PID 1234. Međutim, ako je taj proces podešen da zanemari signal 15, uporabom ga

te naredbe nećemo zaustaviti. Ipak, naredbu kill možete koristiti i na način koji proces "ne može odbiti".

Signal 9 je bezuvjetni signal zaustavljanja. On uvijek zaustavi proces. Za bezuvjetno zaustavljanje procesa koristite sljedeću naredbu:

kill -9 PID

Pretpostavimo da unesete ps -f i ugledate sljedeći odaziv:

UID	ΡID	PPID	С	STIME	ΤΤΥ	TIME	COMMAND
chris	65	1	0	11:40:11	tty01	0:06	-bash
chris	71	65	61	11:42:01	tty01	0:14	total_updt inventory
chris	231	65	80	11:46:02	tty01	0:00	ps -f
chris	187	53	60	15:32:01	tty02	123:45	crunch stats
chris	53	1	0	15:31:34	tty02	1:06	-bash

Za zaustavljanje biste procesa 187 obično unijeli *kill 187*. Ako ponovno unesete *ps -f* i uočite da je proces još uvijek tu, znate da je proces podešen da zanemari naredbu *kill*. Za njegovo bezuvjetno zaustavljanje unesite *kill -9 187*. Kada ponovno unesete *ps -f*, vidjet ćete da tog procesa više nema.

#### **UPOZORENJE**

Nedostatak je uporabe bezuvjetne inačice naredbe *kill* u tome što *kill* -9 procesu ne dozvoljava da prije zaustavljanja završi ono što je radio. Ako *kill* -9 upotrijebite na programu koji upravo obnavlja datoteku, možete izgubiti obnovljeni materijal ili čitavu datoteku.

Moćnu naredbu *kill* -9 odgovorno koristite. Opciju -9 u većini slučajeva ne trebate, Naredba *kill* izdana bez argumenata zaustavlja većinu procesa.

**Zaustavljanje svih pozadinskih procesa** Za zaustavljanje svih pozadinskih poslova unesite *kill 0*. Naredbe koje se izvršavaju u pozadini ponekad pokrenu više od jednog procesa. Praćenje svih PID brojeva pridruženih procesu koji želite zaustaviti može biti zamorno. Budući da *kill 0* zaustavlja sve procese pokrenute s trenutnom ljuskom, to je brži i manje zamoran put za zaustavljanje procesa. Kada želite vidjeti koje se naredbe izvršavaju u pozadini za trenutnu ljusku, unesite naredbu *jobs*.

## Odavde...

U ovom su poglavlju predstavljene naredbe koje trebate za upravljanje većim brojem procesa. Saznali ste da pri svakom stavljanju poslova u pozadinu s ampersandom (&) ili kada koristite cijevi, pokrećete više procesa. Poslove možete rasporediti za pokretanje u točno određeno vrijeme pomoću naredbe *at*, u vrijeme koje sustav smatra odgovarajućim pomoću naredbe *batch* i u redovitom vremenskom rasporedu s *cron* i *crontab*. O više informacija pogledajte sljedeće:

- U poglavljima se 3. dijela, "Upravljanje datotečnim sustavom" raspravlja kako nazdirati i održavati vaš Linux sustav. Administracija sustava kao temu nije lako naučiti i, u stvari, zahtijeva pristup učenju s mnogo prakse. Ovaj vam dio knjige pruža temeljno razumijevanje osnova i zadaća koje se zahtijevaju od administratora sustava (često nazivanih i *sys admin*).
- 18. poglavlje, "Razumijevanje ljuski Linuxa" pruža specifične informacije o ljuskama potrebne pri programiranju skripti za pokretanje, zaustavljanje i nadziranje procesa na vašem Linux sustavu.

Pored toga, pogledajte još i *man* stranice za različite naredbe o kojima se govori u ovom poglavlju.

# P 0 G L A V L J E **20**

## Ispisivanje

(Jack Tackett)

### U ovom poglavlju

Odabir pisača za rad s Linuxom Što vam treba za podešavanje postavki pisača? Kako ispisivanje radi pod Linuxom? Razumijevanje važnih programa za ispis Razumijevanje važnih direktorija Razumijevanje važnih datoteka Razumijevanje datoteke /etc/printcap Stvaranje probne printcap stavke Sastavljanje cjeline Podešavanje pisača za Red Hat Premda smo svi očekivali da će nam računalna revolucija donijeti ured bez papira, to se nije dogodilo. Danas se koristi više papira nego prije 20 godina. Bell Labs su dok je Unix još bio u povojima stvarali - i ispisivali - tehničku dokumentaciju. Posljedica je toga da UNIX, a stoga i Linux, ima vrlo mnogo uslužnih programa za ispis (ili barem za formatiranje podataka za ispis). Ovo je poglavlje usredotočeno na mehaniku stvarnog ispisivanja datoteke.

Ispisni se sustavi uobičajeni za BSD UNIX/Linux nazivaju lpr (Line PrinteR)sustavi.

## Odabir pisača za rad s Linuxom

Ako pisaču možete pristupiti iz MS-DOS-a, tada biste ASCII znakove trebali moći ispisivati i iz Linuxa. Jedini je nedostatak što iz Linuxa nećete moći pristupiti određenim mogućnostima vašeg pisača. Jedan je od glavnih razloga za to da sustav pod Linuxom datoteku koju treba ispisati najprije šalje u drugu datoteku. Linux datoteke na to privremeno mjesto šalje zbog toga što su pisači razmjerno spori vanjski uređaji i sustav vašu *sesiju* ne želi usporiti radi običnog ispisivanja datoteke. Taj je proces nazvan *spooling* (privremeno spremanje datoteka za kasniji ispis) pa su stoga i pisači nazvani *spooling devices*. Kada datoteku ispisujete iz Linuxa, ona ne ide izravno na pisač, nego u red čekanja s ostalim stavkama koje su poslane na ispis. Ako je vaša datoteka prva na redu, ispisivanje će početi gotovo trenutno.

BILJEŠKA Spool je kratica za Simultaneous Peripheral Operation Off Line. Ova je kovanica stvorena u ranim danima velikih IBM mainframe računala, kada su manja računala korištena za ispisivanje izvještaja izvan veze s velikim računalom. Ovakav je postupak omogućavao da skupa mainframe računala nastave sa svojim zadaćama bez gubljenja vremena na priproste poslove poput ispisivanja.

Budući da je Linux u velikoj mjeri naslijedio organizaciju i mogućnosti UNIX-a, on podržava mnoge vrste pisača. Ako vašem pisaču možete pristupiti iz DOS-a (kako je ranije spomenuto), trebali biste mu moći pristupiti i iz Linuxa.

## Što vam treba za podešavanje postavki pisača?

Ovo poglavlje pretpostavlja da znate kako pod Linuxom urediti tekstualnu datoteku i da poznajete osnove vlasništva datoteka i dopuštenja pristupa datotekama. Pretpostavlja se i da ste vaš Linux sustav podesili i da on ispravno radi. Ako ćete koristiti ispis na udaljenom uređaju, osobito je važno da su vaši mrežni podsustavi instalirani i da ispravno rade. Za više informacija pogledajte *man* stranice o naredbama *chmod* i *chown*. Pregledajte i 8. poglavlje, "Uporaba editora *vi*" radi informacije o uporabi editora *vi*, budući da ćete ga trebati za uređivanje nekoliko datoteka pri podešavanju postavki vaših pisača.

## Kako ispisivanje radi pod Linuxom?

Najjednostavniji je način za ispisivanje pod Linuxom poslati podatke izravno na ispisni uređaj. Sljedeća naredba šalje ispis direktorija prvom paralelnom pisaču (u DOS-u je to LPT1):

ls > /dev/lp0

Ovaj način ne koristi prednosti višezadaćnih mogućnosti Linuxa budući da će se za dovršenje ove naredbe zauzeti onoliko vremena koliko je pisaču potrebno da podatke stvarno ispiše. To bi na sporijem pisaču ili pisaču koji nije odabran ili je isključen moglo prilično dugo potrajati. Bolji je način postupni ispis (*spooling*) podataka - to jest sakupiti podatke za ispis u datoteku i zatim pokrenuti pozadinski proces za slanje podataka u pisač.

Privremeno je spremanje datoteka za kasniji ispis suštinski način rada Linuxa. Za svaki je pisač određeno *spool* područje (direktorij s datotekama koje čekaju u redu za ispis). Podaci se za svaki pisač sakupljaju u taj privremeni direktorij, po jedna datoteka na jedan ispisni zadatak. Pozadinski proces (nazvan *printer daemon*) neprestano pretražuje *spool* područja radi novih datoteka za ispis. Kada se neka pojavi, podaci se šalju u odgovarajući pisač i uklanjaju iz reda čekanja. Kada na ispis čeka više datoteka, one se ispisuju redom kojim su dovršene - prva unutra, prva van. Tako je *spool* direktorij zapravo red čekanja i poslovi koji čekaju na ispis su u redu čekanja (*queued* ili *in the print queue*). U slučaju ispisa na udaljenom uređaju podaci su najprije spremljeni lokalno kao za svaki drugi ispisni zadatak, ali je pozadinskom procesu rečeno da podatke pošalje u određni pisač na određenom udaljenom stroju.

Neophodne informacije koje *printer daemon* treba za obavljanje svog posla - fizičku napravu koju treba koristiti, *spool* direktorij koji treba pretražiti, udaljeni stroj i pisač za daljinski ispis itd. - sve je to pohranjeno u datoteku nazvanu /etc/printcap. O pojedinostima se o ovoj datoteci govori kasnije u članku "Razumijevanje datoteke /etc/printcap".

BILJEŠKA Pisač je u Linuxu samo još jedna datoteka. Međutim, budući da je to stvaran uređaj, imat će svoju stavku u direktoriju /dev. Linux se prema stvarnim uređajima odnosi kao da su oni dio datotečnog sustava.

Pojam *pisač (printer)* se ovdje odnosi na pisač koji je naveden u /etc/printcap. Pojam *stvaran (fizički) pisač* se odnosi na uređaj koji stvarno stavlja znakove na papir. U /etc/printcap je moguće imati više stavki koje sve opisuju jedan stvaran pisač, ali to rade na različite načine. Ako vam ovo nije jasno, pročitajte članak o /etc/printcap.

## Razumijevanje važnih programa za ispis

UNIX-ov sustav ispisivanja obuhvaća pet programa. Kako je unaprijed zadano, njihove su lokacije prikazane u tablici 20.1, vlasništvo su *root* korisnika, spadaju u skupinu *daemon*, i imaju dopuštenja koja su prikazana u tablici.

Tablica 20.1 Važni programi z	a ispis
Dopuštenja pristupa datotekama	Lokacije datoteka
-rwsr-sr-x	/usr/bin/lpr
-rwsr-sr-x	/usr/bin/lpq
-rwsr-sr-x	/usr/bin/lpc
-rwsr-sr-x	/usr-bin/lprm
-ľwxľ-S—-	/usr/bin/lpd

Prva se četiri dopuštenja u tablici 20.1 koriste za podnošenje, poništavanje i pregledavanje ispisnih zadaća; /usr/sbin/lpd je *printer daemon*.

BILJEŠKA Lokacije, vlasništva i dopuštenja u tablici 20.1 su pojednostavljena i možda će za vaš sustav biti pogrešna, stoga obratite pažnju na *lpd* datoteke i dopuštenja pristupa datotekama.

Sve ove naredbe imaju *man* stranice koje možete proučiti ako vam treba više informacija. Ovdje je važno da, kako je unaprijed zadano, *lpr*, *lprm*, *lpc*, i *lpq* djeluju na pisaču nazvanom *lp*. Ako definirate varijablu okružja nazvanu PRINTER, umjesto *lp* se koristi ovo zadano ime. *lp* i varijablu okružja PRINTER možete i premostiti ako u naredbenom retku odredite ime pisača koje će se koristiti, na primjer:

lpc - PMYPRINTER

#### Ipd demon

Linux svim poslovima ispisivanja barata preko *lpd demona*. Ako se taj proces ne izvršava, ne može se izvesti nikakvo ispisivanje - datoteke za ispis će ostati u svojim privremenim direktorijima u kojima čekaju red na ispis sve do pokretanja *lpd* procesa (više se informacijama o *spool* direktorijima pojavljuje kasnije u članku "Razumijevanje važnih direktorija").

Vidi "Razumijevanje Procesa", str. 349

Ako vaš sustav pri pokretanju ne učitava *lpd* ili *lpd demon* zbog nekog razloga morate zaustaviti i zatim ponovno pokrenuti, sljedeća naredba pokreće *printer demon*:

lpd [opcije]

*Man* stranica o *lpd* daje popis opcija, međutim, jedna važna opcija pri podešavanj postavki vaših Linux pisača jest *-l*, koja stvara *log* datoteku koja bilježi svaki zahtjev za ispisom na vašem sustavu. Ova *log* datoteka može biti korisna pri otkrivanju i uklanjanju grešaka vašeg sustava za ispis.

#### Naredba Ipr

*lpr* naredba šalje posao pisaču ili ispisni zadatak stavlja u red čekanja - stvarno se zapravo događa da se datoteka koju ste odredili kopira u direktorij s datotekama koje čekaju u redu na ispis *(spool direktorij)*. Svaki pisač koji je uvršten u vaš Linux sustav mora imati svoj vlasitti *spool* direktorij. Veličina je tog direktorija određena u *minfree* datoteci smještenoj u svakom direktoriju. *Minfree* datoteka određuje broj diskovnih blokova koje treba pričuvati za datoteke koje čekaju ispis na tom pisaču. Time se *lpd daemonu* onemogućuje zauzimanje cijelog tvrdog diska kada zahtjeve za ispis stavlja u red čekanja.

*lpd* pronalazi datoteku, koja zatim preuzima brigu o premještanju podataka u stvarni pisač. Ako datoteku ne odredite, *lpr* koristi uobičajeni ulaz.

#### Naredba Ipq

Naredba *lpq* prikazuje sadržaj *spool* direktorija datog pisača. Jedna od važnih informacija koju *lpq* prikazuje jest identifikacijski broj posla po kojem određeni posao prepoznajemo. Ako želite poništiti nedovršeni posao, trebate navesti i taj broj.

*lpq* s brojem ukazuje i na kojem je mjestu u redu svaki posao. *active* znači da se datoteka upravo ispisuje - ili da ju *lpd* barem pokušava ispisivati.

#### Naredba Iprm

Naredba *lprm* posao uklanja iz reda čekanja - to jest neispisane datoteke uklanja iz *spool* direktorija. Možete navesti identifikacijski broj posla (koji dobivate uporabom naredbe *lpq*) ili kao identifikacijski broj navesti "-" za poništavanje svih poslova koji vam pripadaju.

Ako *lpq* izdate kao *root*, svi će poslovi za pisač biti poništeni. Ako ste prijavljeni kao *root* i želite ukloniti sve poslove koji pripadaju određenom korisniku, navedite njegovo korisničko ime.

#### Naredba Ipc

Naredba *lpc* vam omogućuje provjeru stanja pisača i upravljanje nekim vidovima njihove uprabe. Na primjer, *lpc* vam omogućuje pokretanje ili zaustavljanje slanja podataka u pisače, omogućavanje ili onemogućavanje pisača i preuređivanje redoslijeda poslova u redu čekanja. Sljedeće naredbe onemogućuju ispis na pisaču *myprinter*, omogućuju redoslijed čekanja na ispis na pisaču *yourprinter* i posao broj 37 pomiču na vrh reda:

```
lpc down myprinter
lpc enable yourprinter
lpc topg 37
```

Ako *lpc* dozovete bez ikakvih argumenata naredbe, *lpc* miruje, čekajući da izdate naredbe koje treba poduzeti. Neke su od važnijih naredbi prikazane u tablici 20.2. O cjelovitim uputama pročitajte *man* stranicu. Većina naredbi *lpc* kao parametar uzima ime pisača koje je navedeno u /etc/printcap.

Tablica 20.2	Neke uobičaj	ene lpc naredbe
Naredba	Parametar	Opis
stop	pisa <b>č</b>	Zaustavlja pisač, ali zahtjevi za ispis ostaju u redu čekanja.
start	pisač	Pisaču omogućuje da započne ispis datoteka prethodno stavljenih u red čekanja i svih novih datoteka poslanih na ispis tom pisaču.
exit, quit	(Ništa)	Napušta interaktivni način rada naredbe lpc
status	pisač	Prikazuje trenutno stanje pisača. status pruža informacije poput, na primjer, da li je pisač omogućen i pruža broj poslova (ako postoje) u redu čekanja za ispis.

```
BILJEŠKA Upamtite da su neke lpc naredbe ograničene na root korisnika.
```

#### **UPOZORENJE**

*lpc* je u svojoj trenutnoj primjeni pod Linuxom vrlo nestabilna. Neki su korisnici izvijestili da *lpc* može prikazati neispravne poruke stanja, a ponekad čak i sasvim zamrznuti sustav.

## Razumijevanje važnih direktorija

Za ispis postoji samo jedan važan direktorij - *spool* područje (red čekanja) gdje se podaci za ispis nakupljaju prije nego ih /etc/lpd ispiše. Međutim, u sustavu je obično postavljeno više *spool* direktorija, po jedan za svaki pisač, da bi se olakšalo upravljanje pisačem. Na primjer, moj je sustav podešen za uporabu /usr/spool/lpd kao glavno spremište datoteka koje čekaju red na ispis, a svaki zasebni pisač pod ovim ima direktorij imenovan jednako kao i pisač. Tako pisač nazvan ps\_nff ima /usr/spool/lpd/ps\_nff kao svoj direktorij s datotekama koje čekaju red za ispis.

*Spool* direktoriji bi trebali pripadati skupini *daemon*. Korisnik i skupina bi u i iz njih trebali moći učitavati i upisivati, dok bi ostali trebali iz njih moći učitavati - to jest nakon što direktorij stvorite, pobrinite se da s naredbom *chmod* ima dopuštenja -rwxrwxr-x (0775). Za direktorij myprinter bi odgovarajuća naredba trebala izgledati ovako:

```
chmod ug=rwx, o=rx myprinter
chgrp daemon myprinter
```

- Vidi "Dozvole pristupa datotekama", str. 310
- **BILJEŠKA** Lokacije, vlasništva i dopuštenja su ovdje pojednostavljena i možda na vašem sustavu neće biti ispravna, stoga biste trebali pronaći napomene o *lpd* datotekama i dopuštenjima.

## Razumijevanje važnih datoteka

Neovisno o do sada raspravljanim programima svaki *spool* direktorij sadrži datoteke koje imaju dopuštenja -rw-rw-r:

- Datoteka /etc/printcap sadrži specifikacije za svaki imenovani pisač u vašem sustavu.
- Datoteka .seq sadrži brojač posla koji se adresira za *lpr*.
- Datoteka stanja (*status*) sadrži poruku koja će se objaviti s *lpc stat*.
- Datoteku zaključavanja (*lock*) *lpd* koristi za sprečavanje istovremenog pokušaja ispisivanja dva posla na istom pisaču.
- Datoteka grešaka (errs) bilježi neuspjehe pisača.

Linux za potrebe ispisa ne treba datoteku grešaka, međutim, ta datoteka treba postojati da bi *lpd* mogla bilježiti neuspjehe pisača. Datoteka grešaka može biti nazvana kako god želite,

pod uvjetom da je to ime zabilježeno u /etc/printcap. Datoteku grešaka obično stvarate ručno pri postavljanju *spool* područja. O tome ćete više naći u članku "Sastavljanje cjeline ", kasnije u ovom poglavlju.

Datoteka /etc/printcap je od velike važnosti i ona se detaljno opisuje u sljedećem članku.

## Razumijevanje datoteke /etc/printcap

Datoteka /etc/printcap je tekstualna datoteka koju možete uređivati s vašim omiljenim editorom. /etc/printcap treba biti u vlasništvu *root* korisnika i imati dopuštenja -rw-r—r.

Sadržaj datoteke /etc/printcap obično izgleda prilično nerazgovjetno, poput šifre, međutim, kada znate kako ta datoteka radi, tada je sadržaj mnogo lakše razumijeti. U nekim je distribucijama to dodatno zamršeno time što za *printcap* nema *man* stranice, a većinu su printcap datoteka stvorili ili programi ili osobe koje nisu mislile na njihovu čitljivost. Radi očuvanja vlastitog zdravog razuma vašu printcap datoteku napravite što logičnijom i čitljivijom, s mnogo napomena. Ako *man* stranicu još nemate, nabavite ju od *lpd* izvora.

Jedna printcap stavka opisuje jedan pisač. U osnovi, printcap stavka pruža logično ime za fizički uređaj i zatim opisuje kako treba baratati s podacima koji se šalju u taj uređaj. Na primjer, printcap stavka definira koji će fizički uređaj biti korišten, u koji bi se direktorij za čekanje na ispis bilo kakvi podaci za taj uređaj trebali pohraniti, kakav prethodni postupak treba izvršiti na podacima, gdje se trebaju bilježiti greške fizičkog uređaja, i tako dalje. Možete ograničiti količinu podataka koju mogu biti poslani u jedan posao, ili određenim klasama korisnika ograničiti pristup pisaču. Slijedi primjer kako je pisač definiran u printcap datoteci:

```
# Primjer printcap stavke s dva alijasa
myprinter!laserwriter:\
# lp je urelaj za ispis - ovdje je to prvi paralelni pisa~.
:lp=/dev/lp0: \
# sd zna~i spool directory - gdje se sakupljaju podaci za ispis
:sd=/usr/spool/lpd/myprinter:
```

U redu je imati više printcap stavki koje definiraju nekoliko različith načina za baratanje podacima namijenjenih istom fizičkom pisaču. Na primjer, fizički pisač može podržavati formate PostScript i HP LaserJet, ovisno o nekim sekvencama podešavanja koje se fizičkom pisaču šalju prije svakog posla. Ima smisla definirati dva pisača: jedan koji prije obrađuje podatke prethodno dodavši HP LaserJet sekvencu i jedan koji prethodno dodaje Postscript sekvencu. Programi koji proizvode HP podatke šalju ih u HP pisač, dok programi koji proizvode PostScript ispisuju na PostScript pisaču.

Ako preko varijable okružja ne odredite unaprijed zadani pisač ili pisač ne odredite na *lpr* nared-BILJEŠKA benom retku, Linux će ispisni zadatak usmjeriti na pisač *lp*.

Programi koji podatke mijenjaju prije nego su poslani u stvaran pisač nazivaju se *filteri*. Može se dogoditi i da filter stvarnom pisaču ne pošalje nikakve podatke - to jest da filter isključi sve podatke.

### Razumijevanje polja u /etc/printcap

Datoteka printcap ima previše polja da bi se ona u ovom poglavlju mogla u potpunosti objasniti i zato su opisana samo ona najvažnija. Sva su polja u /etc/printcap (osim imena pisača) ograđena dvotočkama i označena dvoslovnim kodom. Iza dvoslovnog koda slijedi vrijednost koja ovisi o vrsti polja. Postoje tri vrste polja: niz znakova (*string*), Boolean

i brojčano. U tablici 20.3 su opisana najčešća i najvažnija polja koja su zatim u sljedećim člancima opisana s više pojedinosti.

Tablica 20.3 Polja u datoteci /etc/printcap				
	Polje	Vrsta	Opis	
	lp	Niz znakova	Određuje uređaj koji će se koristiti za ispis - na primjer, /dev/lp0	
	sd	Niz znakova	Navodi ime spool direktorija za taj pisač	
	lf	Niz znakova	Navodi datoteku u koju će se bilježiti greške s ovog pisača	
	if	Niz znakova	Navodi ime ulaznog filtera	
	rm	Niz znakova	Navodi ime udaljenog računala-domaćina za ispis	
	rp	Niz znakova	Navodi ime udaljenog pisača	
	sh	Boolean	(suppress headers) Ovo navedite za ispuštanje stranice s informacijama o pojedinom ispisu	
	sf	Boolean	(suppress form feed) Isključivanje naredbe za izbacivanje stranice	
	mx	Brojčano	Određivanje najveće dopuštene veličine posla za ispis (u blokovima)	

**Polje** *lp* Ako kao ispisni uređaj navedete /dev/null, svi će se drugi postupci ispravno izvršavati, međutim, završni podaci neće nikamo biti poslani. Ispisivanje u prazno je rijetko korisno osim za probna podešavanja pisača ili s čudnim pisačima. Kada podešavate postavke udaljenog pisača (što znači da ste naveli polja rm i rp), navedite :lp=:.

Polje nemojte ostaviti prazno osim ako koristite udaljeni pisač. *Printer daemon* prigovara ako ne navedete ispisni uređaj.

Polje lf Ovdje treba navesti postojeću datoteku, inače greške neće biti zabilježene.

**Polje if** *Input filters* su programi koji podatke za ispis preuzimaju na njihovom standardnom ulazu i izlaz stvaraju na njihovom standardnom izlazu. Uobičajena je uporaba ulaznih filtera otkrivanje običnog ASCII teksta i njegovo pretvaranje u PostScript - to jest njihov ulaz je sirovi tekst, a njihov izlaz je PostScript.

Kada navedete ulazni filter, *printer daemon* podatke koji su na redu za ispis ne šalje u zadani uređaj. Umjesto toga, on ulazni filter s podacima za ispis vodi kao standardni ulaz, a ispisni uređaj kao standardni izlaz.

**Polja rm i rp** Slanje je vaših podataka za ispis u pisač priključen na drugi stroj jednostavno poput određivanja udaljenog stroja *rm (remote machine)* i udaljenog pisača *rp (remote printer)* i osiguravanja da je polje ispisnog uređaja *lp (local printer) prazno.* 

BILJEŠKA Podaci za ispis su, prije nego se pošalju udaljenom stroju, i dalje spremljeni u lokalnom redu čekanja. Svi se ulazni filteri koje ste naveli također izvršavaju.

**Polja sh i sf** Osim ako vaše računalo koristi mnogo različitih osoba, vi vjerojatno ne trebate posebno dodane stranice s informacijama o pojedinom ispisu, stoga navedite *sh*.

Naredbu za izbacivanje stranice (*form feed*) sprečavamo navođenjem polja *sf*. To je najkorisnije ako se vaš pisač najčešće koristi kao izlaz iz programa za obradu teksta. Većina uobičajenih programa za obradu teksta stvaraju cjelovite stranice podataka pa ako *printer daemon* na kraj svakog posla za ispis još doda naredbu za izbacivanje stranice, nakon svakog posla dobijete i praznu stranicu. Međutim, ako se pisač obično koristi za ispis programskog koda ili direktorija, naredba za izbacivanje stranice će osigurati potpuno izbacivanje posljednje stranice, tako da svaki popis započne na vrhu nove stranice.

**Polje mx** Polje *mx* vam omogućuje ograničavanje veličine podataka koji će se slati u red čekanja na ispis. Broj koji navedete predstavlja *BUFSIZE* blokove (1 KB pod Linuxom). Ako navedete nulu, ograničenje je uklonjeno i poslove za ispis ograničava jedino raspoloživi prostor na disku.

Ograničenje se odnosi na veličinu podataka u redu za čekanje na ispis, ne na količinu **BILJEŠKA** podataka koja se šalje fizičkom pisaču.

Ako korisnik to ograničenje pokuša premašiti, datoteka će biti potkraćena. Korisnik će ugledati sljedeću poruku:

(lpr: ime-datoteke: copy file is too large)

To je ograničenje korisno za ne-PostScript pisače ako imate korisnike ili programe koji namjerno ili slučajno mogu stvoriti neumjereno veliki izlaz. Za PostScript pisače to ograničenje nimalo ne koristi budući da vrlo mala količina PostScript podataka za ispis može stvoriti veliki broj izlaznih stranica.

#### Umetanje varijable okružja PRINTER

Možda ćete u vašu prijavnu skriptu - ili čak u unaprijed zadanu korisničku prijavnu skriptu - htjeti dodati redak koji postavlja varijablu okružja *PRINTER*. Pod ljuskom *bash* je podesan redak za to *export PRINTER=myprinter*. Nakon toga korisnici više neće trebati navoditi - *Pmyprinter* svaki put kada šalju ispisni zadatak.

Za dodavanje je više pisača samo potrebno taj proces ponoviti s drukčijim imenima pisača. Upamtite da možete imati više printcap stavki koje sve koriste isti fizički uređaj. Na taj se način prema istom uređaju možete različito odnostiti, ovisno o tome kako ga nazovete kada mu šaljete ispisni zadatak.

## Stvaranje probne printcap stavke

Sljedeća je skripta ljuske vrlo jednostavan ulazni filter - on svoj izlaz naprosto ulančava na kraj datoteke u /tmp, iza uvodne stranice s informacijama o dotičnom ispisu. Taj filter navedite u printcap stavki i kao ispisni uređaj navedite /dev/null. Tako ispisni uređaj zapravo neće biti korišten, ali ispis trebate usmjeriti na nešto, inače će *printer daemon* prigovarati.

U sljedećoj printcap stavki uočite prihvatljivo čitljiv format i uporabu znaka za nastavak (\) u svakom retku osim posljednjeg:

```
myprinter!myprinter: \
:lp=/dev/null: \
:sd=/usr/spool/lpd/myprinter: \
:lf=/usr/spool/lpd/myprinter/errs: \
:if=/usr/spool/lpd/myprinter/input_filter: \
:mx#0: \
:sh: \
:sf:
```

## Sastavljanje cjeline

Za sastavljanje svih prethodnih dijelova u cjelinu sljedeći vas koraci vode kroz podešavanje postavki jednog pisača na /dev/lp0. To zatim možete proširiti i na druge pisače. (Usput, da biste sve ovo mogli uraditi, trebate biti *root* korisnik).

- 1. Provjerite dopuštenja pristupa i lokacije za *lpr*, *lprm*, *lpc*, *lpq* i *lpd*. Ranije u ovom poglavlju, u tablici 20.1 su prikazane ispravne postavke i direktoriji.
- Stvorite *spool* direktorij za vaš pisač (ovdje nazvan *myprinter*). Osigurajte da su
  i direktorij i pisač u vlasništvu *root* korisnika, da pripadaju grupi *daemon* i imaju
  dopuštenje upisivanja za korisnika i grupu i dopuštenje učitavanja (read-only)
  za druge (-rwxrwxr-x). Koristite sljedeće naredbe:

```
mkdir /usr/spool/lpd
mkdir /usr/spool/lpd/myprinter
chown root.daemon /usr/spool/lpd /usr/spool/lpd/myprinter
chmod ug=rwx,o=rx /usr/spool/lpd /usr/spool/lpd/myprinter
```

3. U direktoriju /usr/spool/lpd/myprinter stvorite neophodne datoteke i dajte im ispravna dopuštenja i vlasnika. Koristite sljedeće naredbe:

```
cd /usr/spool/lpd/myprinter
touch .seq errs status lock
chown root.daemon .seq errs status lock
chmod ug=rw,o=r .seq errs status lock
```

4. U direktoriju /usr/spool/lpd stvorite skriptu ljuske *input\_filter*. Za vaš ulazni filter koristite onaj prikazan prije, u članku "Stvaranje probne printcap stavke". Osigurajte da je datoteka vlasništvo *root* korisnika, da pripada skupini *daemon* i da je svatko može izvršavati. Koristite sljedeće naredbe:

```
cd /usr/spool/lpd/myprinter
chmod ug=rwx,o=rx input_filter
```

- 5. Stvorite datoteku /etc/printcap ako ona još ne postoji. Uklonite sve stavke u njoj i dodajte joj probnu printcap stavku datu u članku "Stvaranje probne printcap stavke". Osigurajte da je datoteka u vlasništvu *root* korisnika i *read-only* (samo za čitanje) za sve ostale. Možete koristiti naredbu *chmod* za uspostavljanje ispravnih dopuštenja pristupa datoteci: -rw-r—r—(ili oktalno 644).
- Uredite datoteku rc.local (možete koristiti bilo koji ASCII editor, npr., vi ili emacs). Na kraj dodajte redak /etc/lpd za pokretanje printer daemona tijekom svakog podizanja sustava. Međutim, sada nije neophodno podizati sustav - možete ga pokrenuti ručno s naredbom lpd.
- 7. Napravite probni ispis unjevši sljedeće:

ls -1 ¦ lpr -Pmyprinter

- Uporabom naredbe *ls* u /tmp potražite datoteku testlp.out. Ona bi trebala sadržavati sadržaj vašeg direktorija, koji možete provjeriti s naredbama *more*, *less* ili *cat*.
   O više informacija o ovim naredbama pogledajte 17. poglavlje, "Upravljanje datotekama i direktorijima".
- Vidi "Pregledavanje sadržaja datoteke", str. 325
  - 9. Uporabom ASCII editora (poput vi) učinite sljedeće promjene u /etc/printcap:
    - U stavci za prvi pisač oba izraza *myprinter* promijenite u *testlp* samo u prvom retku.
    - U drugoj stavci, /*dev/null* promijenite u vaš stvaran ispisni uređaj na primjer, /*dev/lp0*.
    - U drugoj stavci potpuno uklonite redak if.

Sada stavku myprinter kopirajte tako da u datoteci imate dvije jednake stavke.

- 10. Ili ponovno podignite sustav ili zaustavite *printer daemon* i ponovno ga pokrenite. To radite zbog toga što *printer daemon* pri prvom pokretanju gleda samo u datoteku /etc/printcap.
- Uporabom naredbe *ls -l* | *lpr -Pmyprinter* ponovno pokrenite probni ispis. Taj bi trebao izaći na vašem stvarnom pisaču.

#### RJEŠAVANJE POTEŠKOĆA

**Dobivam poruku**: *lpd: connect: No such file or directory.* Printer daemon */etc/lpd* ne radi. Možda ste ga zaboravili dodati u vašu datoteku */etc/rc.local*. Ili ste ga možda dodali, ali od tada još niste ponovno podigli sustav. Dodajte ga i ponovno podignite sustav ili naprosto pokrenite */etc/lpd*. Upamtite da za to trebate biti root korisnik.

**Dobivam poruku**: *Job queued, but cannot start daemon.* To je često javlja odmah nakon poruke *lpd: connect.* Problem je isti kao i prethodni.

**Dobivam poruku:** *lpd: cannot create spooldir/.seq.* Niste stvorili *spool* direktorij naveden u printcap stavci ili ste ga pogrešno nazvali. Drugi bi odgovor bio (premda mnogo manje vjerojatnom) da vam je na disku ostalo premalo prostora.

nastavlja se

#### nastavak

**Dobivam poruku:** *Ipr: Printer queue is disabled.* Kao root korisnik za omogućavanje pisača koristite *Ipc enable ime-pisača.* Imajte na umu da kao root korisnik možete poslove slati čak i onemogućenom pisaču.

Pošaljem posao za ispis i nema poruka o grešci, međutim, na fizički pisač ništa ne izlazi. Za to bi moglo biti više razloga:

- Osigurajte da je fizički pisač uključen, odabran i fizički spojen na uređaj naveden u datoteci /etc/printcap.
- Pomoću naredbe *lpq* provjerite da li je stavka za ispis trenunto u redu čekanja. Ako jest, uređaj je možda zauzet, možda pisač ne radi ili je u njemu kakva greška. U slučaju grešaka provjera datoteke za bilježenje grešaka navedene u printcap stavci može vam pomoći u otkrivanju uzroka.
- Pomoću naredbe /pc status možete provjeriti da li je zastoj u pisaču, a za njegovo oporavljanje možete upotrijebiti /pc up ime-pisača ili /pc restart ime pisača (za to trebate biti root korisnik).

Ako vaši poslovi za ispis ni nakon provjere ne izlaze iz pisača, provjerite da li je svaki ulazni filter koji ste naveli prisutan u ispravnom direktoriju i ima ispravna dopuštenja. Ako pokrenete syslogd, možete u vašim zabilježenim događajima provjeriti poruke od *lpd-a*. Ako među zabilješkama ugledate stavku cannot execv ime izlaznog filtera, gotovo je sigurno da je problem u tome.

Druga je mogućnost da imate PostScript pisač, a vi u njega ne šaljete PostScript. Većina PostScript pisača zanemaruje ne-PostScript podatke. Možda ćete trebati instalirati odgovarajući ulazni filter za pretvaranje teksta u PostScript.

l na kraju (a osjećat ćete se budalasto ako je to slučaj) provjerite da li vaš ulazni filter stvarno proizvodi izlaz i da izlazni uređaj nije /dev/null.

**Moj se pisač izgleda umrtvio. Izgleda da ni jedan od prethodnih načina ne rješava taj problem.** Kada sve drugo ne uspije i pisač i dalje ne ispisuje, prije posljednje mogućnosti možemo još pokušati zaustaviti *lpd daemon* i ponovno ga pokrenuti. Ako ni to ne uspije, kao posljednje vam sredstvo ostaje ponovno podizanje vašeg Linux sustava s naredbom shutdown -r. Prije uporabe ove opcije provjerite da nitko drugi nije prijavljen na sustav i da ste spremili sve datoteke, inače odredite vrijeme zaustavljanja sustava i o tome prije obavijestite sve vaše trenutne korisnike. Pisač možete isprobati i na DOS ili Windows stroju radi provjere da sam fizički uređaj radi ispravno.

## Podešavanje pisača za Red Hat

Ako ste pod Red Hatom instalirali XFree86, za dodavanje i brisanje pisača kao i održavanje datoteke /etc/printcap i datoteka i direktorija za datoteke koje čekaju red na ispis možete koristiti alat za podešavanje postavki pisača prikazan na slici 20.1. Ovaj alat možete pronaći u Control Panelu. Tablica 20.5 opisuje svaku od stavki PrintTool izbornika.



PrintTool	lpd	Tests	Help	
			Printer Queues in /etc/printcap	
lp			REMOTE 1pd queue 1p on tackett.netwharf.com	$\overline{\Lambda}$
lp0			Canon BJC-600 and BJC-4000 on /dev/lp1	
	Edit	1	Add Delete	

	+ Princiool izpornici	
Izbornik	Podizbornik	Opis
PrintTool	Reload	Ponovno u direktoriju traži printcap datoteke
	About	Prikazuje informaciju o PrintToolu
	Quit	Izlaz iz PrintToola
Lpd	Restart	Ponovno pokretanje lpd daemona nakon uvođenja promjena
Tests	Print ASCII test page	Ispis probne stranice u običnom tekstu na odabranom pisaču
	Print PostScript test page	Ispis PostScript probne stranice na odabranom pisaču
	Print ASCII directly to port	Ispis probne stranice izravno na uređaj, a ne preko lpd sustava
Help	General Troubleshooting	Pruža opću pomoć za PrintTool Pruža pomoć o različitim poteškoćama s ispisom

Za dodavanje novog pisača kliknite na gumb Add. Najprije trebate odrediti da li je to lokalan, udaljen ili SMB pisač, kako je prikazano na slici 20.2. *local printer* je spojen na vaš paralelni ili serijski priključak, a *remote printer* je spojen na vašu mrežu. Lan Manager Printer je pisač priključen na drugi sustav putem protokola *Server Message Block* (SMB-Samba) - to je obično Microsoft Windows sustav.

Vidi "Uporaba SAMBE", str. 291

Tablias 00 / DrintTaal Inhormiai

#### Slika 20.2

Za dodavanje pisača trebate odabrati vrstu pisača.



Za dodavanje postojeće konfiguracije pisača odaberite željenu stavku i kliknite na gumb Edit. Oba će postupka otvoriti okvir za dijalog prikazan na slici 20.3. U svako polje u okviru za dijalog treba unijeti potrebne vrijednosti. Tablica 20.5 opisuje svako polje.

#### Slika 20.3

Za ispravno ispisivanje iz Linuxa trebate navesti određene opcije, poput imena pisača i lokaciju fizičkog priključka.

Names (name1 nan	1p0	
Spool Directory		/var/spool/lpd/lp0
File Limit in Kb (0 =	0	
Printer Device	/dev/lp1	
Input Filter	Select	*auto* - BJC600
≡ S	uppress H	eaders
ок	1	Cancel

labilca 20.5 Stavke u poljima za svaki pisac				
Naziv polja	Opis			
Names	Ime pisača i njegov redoslijed. Možete navesti više imena koja razdvajate znakom  .			
Spool Directory	Direktorij za pohranu dokumenata koji čekaju red za ispis na ovom pisaču, poput /usr/spool/lpd/myprinter.			
File Limit	Najveća dopuštena veličina dokumenta (u kilobajtima). Vrijednost 0 znači da nema ograničenja.			
Printer Device	Fizički priključak za vaš pisač, poput lp0.			
Input Filter	Unesite punu stazu i ime datoteke vašeg uobičajenog filtera. Ako trebate podesiti postavke pisača, kliknite na gumb Select.			
Suppress Headers	Ovaj okvir potvrdite ako ne želite da se sa svakim dokumentom ispisuje i posebna stranica s njegovim opisom.			
Remote Host	Ovo polje u okviru za dijalog Remote Host navodi ime udaljenog domaćina na koji je pisač spojen.			
Remote Queue	Ovo polje u okviru za dijalog Remote Host navodi ispisni redoslijed na udaljenom stroju. Unesite punu stazu.			

Za podešavanje postavki ispisnog filtera kliknite na gumb Select koji će otvoriti okvir za dijalog prikazan na slici 20.4. Tablica 20.6 opisuje razna polja u okviru za dijalog Configure Filter.

Tablica 20.6 Stavke	e polja za svaki filter
Naziv polja	Opis
Printer Type	Vrsta pisača za ovaj filter.
Driver Description	Pruža opis odabranog pisača.
Resolution	Odaberite željenu razlučivost za ovaj pisač.

Naziv polja	Opis
Paper Size	Odaberite željenu veličinu papira za ovaj pisač.
Color Depth	Odaberite željenu uporabu boje za ovaj pisač.
Printing Options	Send EOF naređuje pisaču izbacivanje posljednje stranice. Fix stair-stepping text? uklanja nazubljenost teksta. Fast text printing omogućuje veće brzine ispisa za ne-PostScript pisače.
Margins	Odredite slobodne bijele rubove (margine).
Extra GS options	Odredite posebne ghostscript opcije za odabrani pisač.



Nakon što dodate ili promijenite stavku pisača mogli biste ustanoviti da trebate ponovno pokrenuti *lpd daemon.* Za to u RHS Linux Print System Manageru naprosto odaberite stavku izbornika lpd i kliknite na stavku Restart lpd.

## Odavde...

Naredba *lpr* je standardno sučelje Linuxa za ispis datoteka. S naredbenog retka možete *lpr* koristiti za ispis na mnogo različitih vrsta pisača i za traženje mnogih različitih opcija. Kasnije možete stanje vaših ispisnih zadataka provjeriti s naredbom *lpq*. Ako se predomislite i od pojedinog ispisnog zadatka želite odustati, to možete učiniti s naredbom *lprm*. U svakom slučaju, za više biste informacija trebali pročitati posljednje izdanje Printing HOWTO.

Za dodatne informacije o ovim temama pogledajte sljedeće:

- U 3. se poglavlju, "Instaliranje Red Hata" opisuje kako instalirati Red Hat distribuciju Linuxa.
- U poglavljima 3. dijela, "Upravljanje sustavom datoteka" se opisuje što je administrator sustava i pojedinosti o tome što on radi.
- Sustav pomoći PrintToola vam pruža bezbroj informacija o uporabi alata za podešavanje sustava vašeg pisača uporabom X-a.
- U 22. se poglavlju, "Uporaba XWindowsa" opisuje uporaba sustava X Windows.
- U dodatku A, "Izvori informacija" ćete saznati kako pronaći Linux Printing HOWTO.

# P 0 G L A V L J E 2 1

## Instaliranje sustava X Windows

(Steve Burnett)

## U ovom poglavlju

Razumijevanje X Windowsa Instaliranje sustava XFree86 Podešavanje sustava XFree86 Uporaba datoteka resursa X Windowsa Da bi se bilo koji operativni sustav mogao natjecati za prostor na današnjim stolnim računalima, on mora imati lako za korištenje grafičko sučelje. Danas su najomiljeniji sustavi Windows i Macintosh. Za razliku od UNIX-a, ni jedan od njih ne može lako izvršavati grafičke aplikacije preko raznovrsne mreže.

Linux sposobnost izvršavanja prozorskih aplikacija preko raznovrsne mreže postiže pridruživanjem XFree86 primjene X11 standarda X Windowsa stvorenog na Massachusetts Institute of Technology (MIT-u). Taj je sustav mnogo više od grafičkog sučelja korištenog za izvršavanje aplikacija - to je moćan klijent poslužitelj sustav koji aplikacijama omogućuje rad i dijeljenje resursa preko mreže. Premda je XFree86 namijenjen za rad u umreženom okružju, on odlično radi i na jednom stroju. Za izvršavanje XFree86 ili X Windows aplikacija nije vam potrebna mreža.

Za instaliranje, podešavanje sustava i uporabu XFree86 trebate poznavati neke osnovne Linux naredbe, na primjer, kako pokrenuti programe, kretanje kroz direktorije te kopiranje, pregledavanje i brisanje datoteka. Možda ćete i neke datoteke trebati preinačiti s tekstualnim editorom. Ako naiđete na temu koju ne razumijete u potpunosti, ovo vam poglavlje pokušava pružiti naredbu koju trebate za izvršavanje operacije i zatim će vas uputiti na drugo poglavlje da s više pojedinosti naučite kako izvršiti operaciju.

Poput većine dijelova Linuxa, XFree86 također ima dokument HOWTO. XFree86 HOWTO održava Matt Welsh na **mdw@sunsite.unc.edu** i može se naći na Wordl Wide Webu na **http://sunsite.unc.edu/LDP**/.

#### UPOZORENJE

Obično se ne trebate brinuti da bi softver mogao oštetiti vaš hardver. Međutim, na žalost, svaki softver koji izravno barata s vašim grafičkim sustavom - bilo s karticom ili s monitorom - može uzrokovati fizičko oštećenje, naročito ako XFree86 pokušavate koristiti pod Linuxom s nepodržanom grafičkom karticom. Prije isprobavanja rada XFree86 provjerite imate li neophodan hardver. Vrlo je preporučljivo pročitati dokumentaciju koja dolazi s sustavom XFree86, smještenu u direktorij /usr/X386/lib/X11/etc pod Linuxom, kao i XFree86 HOWTO Helmuta Geyera u /usr/doc/faq/howto/XFree86-HOWTO.

## Razumijevanje X Windowsa

Sustav X Windows je moćno grafičko operativno okružje koje podržava mnoge aplikacije preko mreže. Sustav X Windows je razvijen u MIT-u i može se besplatno raspačavati. U ovom se poglavlju govori o X11R6 inačici X Windowsa. Međutim, Linux i XFree86 su pomične mete i na Mreži mogu biti raspoložive novije inačice X-a.

Vidi "Uporaba FTP-a za daljinski prijenos datoteka", str. 580

XFree86, inačica koju koristi Linux, je X11R6 standard na Intel-temeljenim sustavima. XFree86 podržava širok raspon standardnog hardvera za osobna računala.

Sustav X Windows je izvorno nastao iz zajedničkog napora dvaju odjela na MIT-u: odjel odgovoran za mrežni program nazvan Project Athena i odjel nazvan Laboratory for Computer Science. Oba su koristila velike količine UNIX radnih stanica i ubrzo su shvatili
da oba ponovno izmišljaju kotač što se tiče programiranja grafičkih korisničkih sučelja (GUI-a) za UNIX radne stanice. Da bi smanjile količinu koda koji su obje skupine pisale, odlučili su stvoriti jedan otporan, proširiv prozorski sustav - X Windows.

1987. je nekoliko proizvođača - nadajući se stvaranju jedinstvenog prozorskog sustava za UNIX radne stanice - osnovalo organizaciju nazvanu *X Consortium* za promicanje i standardiziranje X Windowsa. Zahvaljujući tom naporu otvoreno je računalstvo postalo stvarnost. X Consortium je sastavljen od veličina poput IBM-a, Digital Equipmenta i MIT-a. Ova skupina velikih organizacija nadzire izradu i izdavanje novih inačica X11.

XFree86 je zaštićeno ime proizvoda tvrtke XFree86 Project, Inc. Prvobitni programeri koji su X Windowse prilagodili platformi 80386 odlučili su projekt osnovati radi stjecanja članstva u X Consortiumu. Postavši članicom X Consortiuma, XFree86 Project je dobio pristup trenutno razvijanim poslovima i tako je u XFree86 mogao prebacivati nove mogućnosti istovremeno s njihovom primjenom u X Windowsima, umjesto da čeka njihov službeni izlazak da bi ih mogao iskoristiti. X Consortium je od 1. siječnja 1997. X prepustio skupini Open Group.

X Windowsi su zapravo niz dijelova koji zajednički rade da bi korisniku predstavili grafičko sučelje (GUI):

- Osnovni je sustav prozora program koji svoje usluge pruža sustavu X Windows.
- Sljedeći je dio protokol za komunikaciju preko mreže X Network Protocol.
- Na programima koji primjenjuju X Network Protocol je niskorazinsko sučelje, nazvano Xlib, između osnovnog sustava/mreže i programa više razine. Aplikacijski programi obično koriste funkcije u Xlibu umjesto funkcija niže razine.
- Sve ove djeliće zajedno spaja program za upravljanje prozorima (*window manager*). *Window manager* je X Windows aplikacija čija je svrha upravljanje načinima na koje su prozori predstavljeni korisnicima.

Različit od većine drugih sustava prozora osnovni prozorski sustav (grafičko sučelje) ne pruža objekte korisničkog sučelja, poput kliznih traka, naredbenih gumba ili izbornika. Stavke korisničkog sučelja su ostavljene komponentama više razine i programu za upravljanje prozorima.

U X Windows aplikacije ne spadaju samo *window manageri*, nego i igre, grafički uslužni programi, programerski alati i mnoge druge poslastice. Gotovo su sve aplikacije koje trebate ili posebno napisane ili prebačene na X Windowse. Podešavanje i uporaba nekoliko uobičajenih X Windows aplikacija je s više pojedinosti pokriveno u 22. poglavlju "Uporaba X Windowsa".

X Windowsi program za upravljanje prozorima primjenjuju za baratanje zadaćama stvaranja i upravljanja sučeljem koje čini vidljivi dio sustava X Windows. To ne treba pobrkati s OS/2 Presentation Managerom ili s Microsoft Windows Program Managerom. Premda *window manager* za X Windowse upravlja ponašanjem i položajem prozora, nećete naći ikonu System Setup ili Control Panel za održavanje postavki vašeg Linux sustava. Za malo hrabrije, XFree86 uključuje i programske biblioteke i datoteke za programere koji žele razviti svoje vlastite aplikacije pod XFree86. Dok tema programiranja ili bilo kakva druga objašnjenja uključena u stvaranje X Windows aplikacija nadilaze doseg ove knjige, na brojnim je lokacijama za distribuciju preko Interneta, poput **prep.ai.mit.edu**, i na mnogim CD-ROM distribucijama raspoloživa iscrpna dokumentacija koja će vam pomoći u stjecanju osnovnih znanja neophodnih za stvaranje aplikacija za XFree86.

# Što je klijent/poslužitelj sustav?

X Windowsi su klijent/poslužitelj (client/server) sustav kojim upravljaju dva zasebna softvera. Jedan se izvršava na klijentu, a drugi na poslužitelju. Klijent i poslužitelj dijelovi ove slagalice mogu biti na različitim sustavima ili, kao što je slučaj na većini osobnih računala, oba dijela mogu prebivati na istom stroju.

*Klijent/poslužitelj* je vrlo često korišten izraz u današnjoj računalnoj industriji. Poput većine osnovnih zamisli u industriji, klijent/poslužitelj je prečesto i pretjerano korišten do mjere da zbunjuje prosječnog korisnika računala. U tradicionalnom je smislu *poslužitelj* stroj koji samo pruža resurse - prostor na tvrdom disku, pisače, modeme itd. - drugim računalima preko mreže. *Klijent* je potrošač tih usluga - drugim riječima, *klijent* koristi diskovni prostor, pisač ili modeme koje mu pruža *poslužitelj*.

Sada kada razumijete što je klijent i što je poslužitelj, vrijeme je da to potpuno obrnete. U svijetu X Windowsa odnos klijent/poslužitelj je suprotan od onog što ste naučili u svijetu osobnih računala. Prihvaćena je ili uobičajena predodžba poslužitelja da on pruža usluge klijentu koji ih koristi. U najosnovnijem obliku klijent prikazuje aplikaciju koja se izvršava na poslužitelju.

Pod X Windowsima poslužitelj prikazuje aplikaciju koja se izvršava na klijentu. To bi isprva moglo biti pomalo zbunjujuće, međutim, imat će smisla kada se više naviknete na sustav X Windows.

Klijent je u X Windowsima izvor koji pruža programe i resurse nužne za pokretanje aplikacije što bi u tradicionalnom smislu bilo nazvano poslužiteljem. Resursi se nalaze na sustavu klijenta (upamtite da i klijent i poslužitelj sustavi mogu biti na istom stroju), dok je aplikacija prikazana i koristi se na sustavu poslužitelja.

Sposobnost X Windows aplikacije, koja je klijent da se izvršava na poslužitelju smještenom na istom ili na nekom drugom računalu, je nazvana *network transparency (transparentnost mreže)*. Tako je X aplikaciji svejedno radi li na lokalnom ili udaljenom stroju. Zahvaljujući toj sposobnosti, dugotrajni se zadaci mogu pokrenuti na drugom poslužitelju pa rasterećeni lokalni klijent može izvršavati druge zadaće.

# Izlazne mogućnosti

Temeljni prozorski sustav X Windowsima pruža mnoštvo grafičkih operacija za bitmape. X Windowsi i X Windows aplikacije te operacije koriste za grafički prikaz informacija prema korisniku. XFree86 pruža preklapajuće prozore, trenutno iscrtavanje grafike, bitmapirane grafike i slike visoke razlučivosti i visoko-kvalitetni prikaz teksta. Dok su raniji X Windows sustavi u osnovi većinom bili jednobojni, današnji X Windowsi i XFree86 podržavaju širok raspon sustava boja. X Windowsi podržavaju i višeprocesne mogućnosti UNIX-a; tako i XFree86 podržava višeprocesne mogućnosti Linuxa. Svaki prozor prikazan pod X Windowsima može može biti zasebna zadaća koja se izvršava pod Linuxom.

# Mogućnosti korisničkog sučelja

X Consortium nije izdao standardna pravila za korisnička sučelja. Premda se to danas čini pomalo kratkovidno, u to se vrijeme tehnologija korisničkog sučelja vrlo malo istraživala pa ni jedno sučelje nije smatrano nedvojbeno najboljim. Kojem se izgledu daje prednost i kakav osjećaj pruža korisničko sučelje, to vrlo ovisi o osobnim sklonostima.

X Consortium je X Windowse želio učiniti standardom za UNIX radne stanice, što je jedan od razloga da su X Windowsi besplatno raspoloživi na Internetu. Učinivši X Windowse besplatno raspoloživima, oni njeguju interoperabilnost (sposobnost rada na različitim okružjima), koja je kamen temeljac otvorenih sustava. Da je X Consortium propisao korisničko sučelje, X Windowsi možda ne bi stekli svoju trenutnu razinu popularnosti.

## Ulazne mogućnosti

Sustavi na kojima se izvršavaju X Windowsi obično imaju nekakav oblik pokazivačkog uređaja, najčešće miša. XFree86 zahtijeva miša ili uređaj, poput trackballa, koji oponaša miša. Bez takvog uređaja, sustav XFree86 ne možete koristiti s Linuxom. X Windowsi signale iz pokazivačkog uređaja i iz tipkovnice pretvaraju u događaje. Zatim X Windowsi na te događaje reagiraju, poduzimajući odgovarajuće radnje.

#### UPOZORENJE

Ako vaš miš ili drugi hardverski pokazivački uređaj nije među onima koje Linux podržava, imat ćete poteškoća s uporabom XFree86 i programa selection.

# Instaliranje sustava XFree86

Uz malo sreće, sustav XFree86 ste instalirali s pratećeg CD-ROM-a pri instaliranju cijelog paketa Linux Slackware. Sustav X Windows je sadržan u distribucijskim paketima *x* i *xap*. Ako sustav X Windows niste tada instalirali, za instaliranje X Windowsa možete upotrijebiti Slackwareov program *pkgtool*.

Distribucija Red Hat također instalira X kao dio instalacijskog postupka.

# BILJEŠKA Instaliranje softvera

Pod Slackwareom je *pkgtool* najlakši način za instaliranje XFree86 i predstojeći članak "Instaliranje X sustava s *pkgtoolom* za Slackware" sadrži upute za instaliranje X-a pomoću programa *pkgtool*. Međutim, ako datoteke trebate ručno instalirati (na primjer, pri

- - --

nadogradnji na noviji sustav), trebate znati da su datoteke smještene na Slackware CD-ROM-u u /slakware/x# direktorijima: /slakware/x1 do x16.

X se sastoji od nekoliko velikih arhivskih datoteka. Trenutna inačica XFree86 za Linux, smještena na CD-ROM-u, jest 3.1.1. Tablica 21.1 prikazuje glavne datoteke. Trebate se prijaviti kao superuser (root) i potrebne datoteke kopirati u /usr/x386. Ako taj direktorij ne postoji, stvorite ga naredbom mkdir, kako je ovdje prikazano:

```
opus#: mkdir /usr/x386
opus#: cd /usr/x386
opus#: cp -r /cdrom/slakware/x1 .
```

Te će naredbe također sve datoteke s CD-ROM-a postavljene u /cdrom kopirati u trenutni direktorij.

Tablica 21.1	Glavne distribucijske datoteke za XFree86
Ime datoteke	Opis
x3270.tgz	Emulacija terminala IBM 3270
x_8615.tgz	Poslužitelj IBM 8514
x_mach32.tgz	Poslužitelj temeljen na čipu Mach32
x_mach8.tgz	Poslužitelj temeljen na čipu Mach8
x_mono.tgz	Poslužitelj za jednobojne monitore
x_s3.tgz	Poslužitelj na osnovi čipa S3
x_svga.tgz	Poslužitelj za većinu SVGA kartica (dobra osnovna postava)
_vga16.tgz	EGA/VGA 16-bojni poslužitelj
xconfig.tgz	Primjeri Xconfig konfiguracijskih datoteka (obavezne)
xf_bin.tgz	Osnovne binarne datoteke potrebne za X (klijenti)
xf_cfg.tgz	XDM konfiguracija i FVWM programi
xf_doc.tgz	Dokumentacija za XFree86
xf_kit.tgz	Linker (povezivački) kit za XFRee86 (1 od 2)
xf_kit2.tgz	Upravljački programi za Linker kit (2 od 2)
xf_lib.tgz	Biblioteke dinamičkih veza (dll) i konfiguracijske datoteke
xf_pex.tgz	PEX distribucija
xfileman.tgz	Program za upravljanje datotekama (file manager)
xfm.tgz	Program xfm za upravljanje datotekama
xfnt.tgz	X Windows pisma

Ime datoteke	Opis
xfnt75	Pisma od 75 točkica za X
xfract	Program xfractint za prikazivanje fraktala
xgames	Igre za X
xgrabsc.tgz	Programi Xgrabsc i Xgrab (Xgrab je korišten za stvaranje većine slika u ovoj knjizi)
xinclude.tgz	Programske header datoteke definicija i deklaracija za programiranje X Windowsa
xlock.tgz	Program xlock za zaštitu lozinkom na zaslonu
xman1.tgz	Man stranice za X
xman3.tgz	Još man stranica za X
xpaint.tgz	Program Xpaint za crtanje u X-u
xpm.tgz	Xpm biblioteke, dijeljene i statičke
xspread.tgz	Program Xspread za tablične kalkulacije
xstatic.tgz	Statičke biblioteke za X
xv.tgz	XV preglednik slika
xxgdb.tgz	X Windows sučelje za GNU debugger

Za raspakiravanje tih datoteka koristite sljedeću naredbu:

opus: gzip -d imedatoteke.tgz
opus: tar -xvf imedatoteke.tar

**BILJEŠKA** U CD-ROM priložen uz ovu knjigu su stavljani što je moguće svježiji materijali, uz neizbježan vremenski zaostatak vezan uz proizvodnju knjige. Ipak, moguće je da je u trenutku kada ovo čitate na Internetu raspoloživa novija inačica XFree86, stoga provjerite lokacije s potrebnim arhivama - to bi vas moglo spasiti od nekih glavobolja.

Vidi "Uporaba FTP-a za prijenos udaljenih datoteka", str. 580

## Osiguravanje hardverske podrške za XFree86

Provjerite imate li odgovarajući hardver za rad X Windowsa, dovoljan iznos memorije i neophodan diskovni prostor.

Za instalaciju sustava XFree86 i pribavljenih X Windows aplikacija trebate oko 21 MB prostora na disku. Za rad X Windowsa trebate najmanje 16 MB virtualne memorije. *Virtualna memorija* je kombinacija fizičkog RAM-a u vašem sustavu i iznosa *swap* (zamjenskog) prostora koji ste dodijelili Linuxu. Da bi XFree86 mogao raditi pod Linuxom, trebate imati najmanje 4 MB fizičkog RAM-a uz koji je potrebna i *swap* datoteka od 12 MB. Što više fizičkog RAM-a imate, to će učinak sustava XFree86 biti bolji. Vidi "Stvaranje swap particije", str. 72. (za Red Hat) i str. 99. (za OpenLinux)

Zatim, trebate grafičku karticu koju podržava XFree86. Prema Matt Welshovom XFree86 HOWTO, izdanim 15. ožujka 1995., XFree86 podržava grafičke kartice sa skupovima čipova (*chipsetovima*) navedenim u tablicama 21.2 i 21.3.

Tablica 21.2 Ne	e-ubrzani chipsetovi podržani od XFree86
Proizvođač	Chipset(ovi)
ATI	28800-4, 28800-5, 28800-6, 28800-a
Advance Logic	AL2101
Cirrus Logic	CLGD6205, CLGD6215, CLGD6225, CLGD6235
Compaq	AVGA
Genoa	GVGA
MX	MX68000, MX680010
NCR	77C22, 77C22E, 77C22E+
OAK	OTI067, OTI077
Trident	TVGA8800CS, TVGA8900B, TVGA8900C, TVGA8900CL, TVGA9000, TVGA9000i, TVGA9100B, TVGA9200CX, TVGA9320, TVGA9400CX, TVGA9420
Tseng	ET3000, ET4000AX, ET4000/W32
Western Digital/ Paradise	PVGA1
Western Digital	WD90C00, WD90C10, WD90C11, WD90C24, WD90C30
Video 7	HT216-32

Tablica 21.3 Ub	rzani chipsetovi podržani od XFree86
Proizvođač	Chipset(ovi)
Cirrus	CLGD5420, GLGD5420, CLGD5422, CLGD5424, CLGD5426, CLGD5428
Western Digital	WD90C31
ATI	Mach8, Mach32
S3	86C911, 86C924, 86C801, 86C805, 86C805i, 86C928

# Instaliranje X Sustava s pkgtoolom za Slackware

Za instaliranje X Windowsa se trebate prijaviti kao *root* korisnik. Zatim trebate zabilježiti lokaciju X Windows paketa koje želite instalirati. Te su datoteke smještene pratećem Slackware CD-ROM-u u direktoriju /slackware. Za pristup X Windows paketima iz priloženog CD-ROM-a pogledajte u sljedeće direktorije: /cdrom/slackware/x1, /cdrom/slackware/x2, itd. Pobrinite se upamtiti gdje su te datoteke smještene.

- **BILJEŠKA** Budući da Linux taj CD-ROM postavlja (*mount*) u direktorij, navedene ćete datoteke pronaći na tom položaju. Tipična Linux instalacija tako CD-ROM obično postavlja (*mounts*) u direktorij pod korijenskim direktorijem nazvanim cdrom.
  - Vidi "Postavljanje (mounting) i uklanjanje sustava datoteka", str. 269

Zatim, u naredbeni redak unesite *pkgtool*. Ta naredba aktivira programski alat Slackware paketa koji vam omogućuje brisanje starih paketa ili instaliranje novih. Za X Windowse su to paketi *x* i *xap* sadržani u distribuciji Slackware. Pojavit će se izbornik s sljedećim opcijama:

Stavka izbornika	Opis
Current	Instalira pakete iz trenutnog direktorija.
Other	Instalira pakete iz nekog drugog direktorija.
<u>F</u> loppy	Instalira pakete s disketa.
<u>R</u> emove	Uklanja pakete koji su trenutno instalirani.
View	Prikazuje popis datoteka sadržanih u paketu.
<u>E</u> xit	Izlaz iz paketnog alata.

Pritisnite <Shift-o> ili pomoću tipki sa strelicama odaberite redak izbornika Other i pritisnite <Return>. *pkgtool* će zatražiti izvorni direktorij. Unesite direktorij koji ste ranije zabilježili kao direktorij prvog *x* paketa (to je obično *x1*). Dakle, unijeli biste */cdrom/slack-ware/x1*.

Nakon dostavljanja početnog direktorija *pkgtool* će najprije potražiti X poslužitelj za vašu grafičku karticu. Možete instalirati samo jedan poslužitelj, stoga, kada prelazite sa zaslona na zaslon, prije nego odaberete Yes pričekajte da dosegnete potreban X poslužitelj.

Upamtite da za svaki paket trebate instalirati odgovarajuće programe. Premda nisu svi paketi potrebni, ako XFree86 instalirate nakon što ste instalirali Linux, trebali biste pregledati sve pojedinosti o paketima koje ćete instalirati. Ako imate 21 MB potrebnih za punu instalaciju, slobodno instalirajte sve pakete osim X poslužitelja - instalirajte samo X poslužitelj za vaš *chipset*.

# BILJEŠKA Ako ste prethodno instalirali X Windowse, najprije biste trebali napraviti sigurnosnu kopiju važnih konfiguracijskih datoteka i zatim izbrisati trenutno instalirane x i xap pakete.

# Podešavanje sustava XFree86

Nakon instaliranja XFree86 trebate ga podesiti za vaš sustav. XFree86 očekuje da će datoteku XF86Config pronaći u jednom od sljedećih direktorija:

- /etc/XF86Config
- /usr/X11R6/lib/X11/XF86Config.imedomaćina
- /usr/X11R6/lib/X11/XF86Config

BILJEŠKA Podešavanje je XFree86 jednako u većini Linux distribucija, uključujući Red Hat, Slackware 96 i Caldera.

Informacije iz konfiguracijske datoteke možete pronaći u direktoriju /etc/X11/etc. Prije podešavanja vašeg sustava trebali biste provjeriti datoteke README.Config i README.Linux. Ako imate standardnu, podržanu opremu navedenu ranije u članku "Osiguravanje hardverske podrške za XFree86", trebali biste provjeriti ogledne Xconfig datoteke iz paketa x3. Te su datoteke pohranjene u direktoroiju /usr/X11/lib/X11/Sample-Xconfig-files. U datoteci Xconfig.Index provjerite da li je i vaša grafička kartica navedena. To možete učiniti uporabom sljedećih naredbi:

```
cd /usr/X11/lib/X11/Sample-Xconfig-files
less Xconfig.Index
```

#### **UPOZORENJE**

Nikada ne biste smjeli koristiti Xconfig datoteku od nekog drugog, niti doslovce uzimati ni jedan prijedlog iz ove knjige ili iz bilo kojeg drugog izvora, bez temeljitog pregledavanja datoteke radi neodgovarajućih vrijednosti. Na primjer, rad vašeg monitora na nepodržanim frekvencijama može oštetiti vašu opremu.

Ako je vaša grafička kartica navedena, odgovarajuću Xconfig.broj datoteku iz oglednog direktorija kopirajte u direktorij /usr/X11/lib/X11. Za to možete koristiti sljedeću naredbu (samo broj u ovdje prikazanom primjeru zamijenite s brojem iz datoteke Xconfig.Index):

cp Xconfig.broj /usr/X11/lib/X11/Xconfig

Te bi ogledne konfiguracijske datoteke mogle biti dobre za standardan hardver. Konfiguracijsku datoteku možete isprobati pokretanjem X Windowsa (unesite startx). Ako se X Windows sustav pokrene i proradi, čestitamo! Ako zbog nekog razloga konfiguracijska datoteka nije ispravna, Linux prijavljuje grešku. Ako se samo "zamrzne", ponovno podignite vaš sustav. Nakon neuspješnog pokretanja, kada se vratite u naredbeni redak, vi sami trebate stvoriti konfiguracijsku datoteku.

BILJEŠKA Ako nešto krene krivo (ali vam monitor ipak ne explodira), pritisnuvši < Ctrl-Alt-Backspace> trebali biste zaustaviti X poslužitelj i vratiti se u naredbeni redak ljuske.

# Pokretanje programa SuperProbe

Ako prethodne instalacijske procedure ne djeluju, možete pokrenuti program za podešavanje vašeg sustava. Slackware pruža program nazvan xf86Config kao pomoć u podešavanju vašeg XFree86 sustava, međutim, taj program zahtijeva da odgovorite na nekoliko pitanja. Ta se pitanja tiču vrste hardvera koji imate na vašem sustavu i netočni podaci mogu prouzročiti da X taj hardver ošteti.

Trebali biste pročitati nekoliko dokumentnih datoteka u direktoriju /usr/X11R6/lib/X11/doc: HOWTO.Config, README.Config, i configxf.doc. Za čitanje ovih datoteka možete koristiti sljedeću naredbu:

```
less imedatoteke
```

T 1 Kan 04 4

Trebali biste prikupiti i sve priručnike proizvođača vaše grafičke kartice i monitora.

Zatim pokrenite uslužni program SuperProbe:

/usr/X11R6/bin/SuperProb

Taj uslužni program pretražuje vaš sustav, pokušavajući prepoznati instalirani video hardver. Dobivenu informaciju trebate zapisati za kasniju uporabu s programom xf86Config. Informaciju koju vam je dostavio SuperProbe trebali biste dodatno provjeriti u dokumentaciji vašeg hardvera. Program SuperProbe će proizvesti informaciju koja će biti stavljena u različite dijelove datoteke XF86Config.

# Razumijevanje odjeljaka u XF86Config

Datoteka XF86Config je obična ASCII tekstualna datoteka koju učitava XFree86 i koristi ju za podešavanje X poslužitelja za ispravan rad u vašem hardverskom sustavu. Datoteka je organizirana u sljedeće dijelove, kako je prikazano u tablici 21.4.

Tablica 21.4 Odjeljci datoteke XF86Config		
Odjeljak	Opis	
Files	Navodi direktorije za pisma i rgb datoteke.	
ServerFlags	Navodi posebne zastavice za X poslužitelj.	
Keyboard	Opisuje vrstu tipkovnice.	
Pointer	Opisuje vaš pokazivački uređaj, obično vašeg miša.	
Monitor	Pruža detaljne opise vašeg monitora. Ovaj je odjeljak vrlo važan budući da netočna informacija može ozbiljno oštetiti monitor.	
Device	Opisuje vašu grafičku karticu.	
Screen	Koristi informacije iz odjeljaka Monitor i Device za opis područja vašeg fizičkog zaslona, uključujući stavke poput broja boja i veličine zaslona u pikselima.	

#### 432 21. poglavlje Instaliranje sustava X Windows

Svaki odjeljak u datoteci ima sljedeći opći oblik:

```
Section "Ime"
unosne vrijednosti
unosne vrijednosti
još vrijednosti prema potrebi...
#ovo je redak za napomene i XFree86 ga zanemaruje
EndSection
```

Takvu biste konfiguracijsku datoteku trebali izraditi uporabom tekstualnog editora poput *vi*, slijedeći date primjere. Nakon stvaranja datoteke pokrenite program *xf86Config* za stvaranje datoteke XF86Config radi usporedbe. Na kraju pokrenite X poslužitelj u posebnom načinu rada radi ispitivanja podešenja vašeg sustava koja inače ne biste mogli utvrditi iz primjera, proizvedene datoteke ili dokumentacije. Te su mjere opreza neophodne radi stvarne opasnosti oštećenja vašeg sustava.

Vidi "Uporaba vi editora", str. 181.

**Odjeljak Files** Ovaj odjeljak pruža popis različitih pisama instaliranih u vaš sustav u direktoriju /usr/X11R6/lib/X11/fonts. Tu će svaka serija pisama imati svoj vlastiti poddirektorij. Sljedećom naredbom možete utvrditi koja su pisma učitana:

ls /usr/X11R6/lib/X11/fonts

Svaki direktorij na popisu treba imati odgovarajuću stavku u odjeljku Files.

Ovisno o vašem odabiru tijekom instalacije, vaše bi datoteke pisama trebale ići u uobičajene direktorije, a vaš će odjeljak *Files* izgledati poput sljedećeg primjera:

```
Section "Files"

RgbPath "/usr/X11R6/lib/X11/rgb"

fontPath "/usr/X11R6/lib/X11/misc/"

fontPath "/usr/X11R6/lib/X11/Type1/"

fontPath "/usr/X11R6/lib/X11/speedo/"

fontPath "/usr/X11R6/lib/X11/75dpi/"

FontPath "/usr/X11R6/lib/X11/100dpi/"

EndSection
```

**Odjeljak ServerFlags** Rijetko ćete trebati uređivati unaprijed zadani odjeljak *ServerFlags*. Taj odjeljak nadzire tri zastavice koje X poslužitelj koristi za upravljanje svojom operacijom.

Zastavica	Opis
NoTrapSignials	Zastavica koja uzrokuje "dump core", to jest da X poslužitelj sadržaj memorije ispiše u datoteku za otkrivanje grešaka koju je za to stvorio - kada X poslužitelj primi signal od softvera operativnog sustava.
DontZap	Onemogućuje uporabu kombinacije tipki <ctrl-alt-backspace> za zaustavljanje X poslužitelja.</ctrl-alt-backspace>
DontZoom	Onemogućuje prebacivanje između različitih grafičkih načina rada.

Ogledni je odjeljak prikazan kako slijedi - svaka je zastavica stavljena iza znaka napomene i tako onemogućena:

```
Section "ServerFlags"
#NoTrapSignals
#DontZap
#DontZoom
EndSection
```

**Odjeljak Keyboard** Odjeljak Keyboard vam omogućuje određivanje nekoliko opcija za vašu tipkovnicu, poput rasporeda tipaka. Najminimalniji bi odjeljak Keyboard izgledao ovako:

```
Section "Keyboard"
Protocol "Standard"
AutoRepeat 500 5
ServerNumLock
EndSection
```

Na raspolaganju su i mnoge druge opcije, kako je prikazano u tablici 21.5, međutim, mnoge od njih nisu nužne za ispravan rad vaše tipkovnice. Za prikaz punog opisa raznih parametara za svaki odjeljak datoteke XF86Config u naredbeni redak ljuske utipkajte *man XF86Config*.

Opcija	Parametar/Opis
Protocol	Može biti Standard (unaprijed zadano) ili Xqueue.
AutoRepeat <i>odgoda učestalost</i>	Određuje vremensku odgodu prije ponavljanja tipke pri zadanoj brzini (učestalosti).
ServerNumLock	Nalog X poslužitelju da odazivom na tipku NumLock barata interno.
VTSysReq	Određuje da X poslužitelj prebacivanje između virtualnih terminala obavlja uporabom tipke >SysRq> umjesto tipke <ctrl>.</ctrl>

Tablica 21.5Opcije odjeljka Keyboard

Za prelazak između različitih virtualnih terminala pod Linuxom obično koristite <Alt-F*x*> (gdje F*x* predstavlja bilo koju funkcijsku tipku). Međutim, kada ste u X-u, za pristup virtualnom terminalu trebate koristiti <Ctrl-Alt-F*x*>. Naravno, ako sumnjate u potrebu za virtualnim terminalima kada koristite GUI, razmislite što će se dogoditi ako se vaša X sesija *zaključa* - tada virtualni terminal možete iskoristiti za potpuno zaustavljanje vaše X *sesije*.

**Odjeljak** *Pointer* Odjeljak *Pointer* se bavi vašim mišem ili drugim pokazivačkim uređajem. XFree86 ovdje spremljenu informaciju koristi za podešavanje vašeg miša za uporabu pod X-om. Tu svakako trebate navesti protokol koji vaš miš koristi i vrstu uređaja. Ako imate serijski miš, uređaj će biti serijski priključak na koji je miš spojen. Slijedi primjer odjeljka *Pointer*:

```
Section "Pointer"
Protocol "Microsoft"
Device "/dev/mouse"
EndSection
```

Različiti protokoli koje Linux podržava su

BusMouse	Microsoft
Logitech	MouseSystems
MM Series	Xqueue
Mouseman	PS/2

Neke su od drugih opcija raspoloživih u odjeljku Pointer prikazane u tablici 21.6, međutim, njih ne biste smjeli dodavati vašoj datoteci XF86Config osim ako ste posve sigurni kakav će učinak imati na vaš sustav.

Tablica 21.6 Opcije odjelika Pointer		
Opcija	Opis	
BaudRate <i>brzina</i>	Navodi brzinu serijskog miša u baudima.	
SampleRate <i>brzina</i>	Potrebno za neke Logitech miševe.	
ClearDTR ili ClearRTS	Potrebno za neke miševe koji koriste protokol MouseSystem.	
ChordMiddle	Potrebno za neke Logitech miševe.	
Emulate3Buttons	Omogućuje mišu s dvije tipke, poput miša Microsoft, oponašanje miša s tri tipke. Treću tipku zamjenjuje pritisak na obje tipke istovremeno. Mnoge X aplikacije za pravilan rad trebaju miša s tri tipke.	

BILJEŠKA Ako imate Logitech miša, naročito onog koji ne oponaša Microsoft miša, možda ćete trebati eksperimentirati s nekim opcijama u tablici 21.6.

**Odjeljak Monitor** Odjeljak Monitor je vjerojatno najvažniji odjeljak datoteke XF86Config - i vjerojatno najopasniji. Pogrešna informacija u ovoj datoteci može prouzročiti katastrofalno oštećenje vašeg sustava. Budite vrlo pažljivi!

U stvaranju će ovog odjeljka od velike pomoći biti program SuperProbe i dokumentacija proizvođača monitora. Za traženje podataka o vašem monitoru možete iskoristiti i datoteke /usr/X11R6/lib/X11/doc/modesDB.txt i /usr/X11R6/lib/X11/doc/monitors.

Ovako izgleda tipičan odjeljak Monitor:

```
Section "Monitor"
Identifier "Sanyo 1450 NI"
VendorName "Sanvo"
```

```
ModelName
            "Moj 14 pal~ani monitor"
Bandwidth
            60
HorizSync
            30-60
VeriRefresh 50-90
         Name
#Modes:
                    dotclock
                              Horizontal Timing
                                                  Vertical Timing
          "640x480"
                    25
                              640 672 768 800
                                                  480 490 492 525
ModeLine
          "800x600"
ModeLine
                      36
                              800 840 912 1024
                                                  600 600 602 625
          "1024x768i"
ModeLine
                      45
                              1024 1024 1224 1264
                                                 768 768 776 816
EndSection
```

U vašem odjeljku *Monitor* može biti definirano više monitora i za svaki monitor trebate dostaviti informacije prikazane u tablici 21.7.

Opcija	Opis
Identifier <i>niz znakova</i>	Identifikator monitora.
VendorName <i>niz znakova</i>	Ime proizvođača.
ModelName <i>niz znakova</i>	Naziv proizvoda i model.
Bandwidth <i>vrijednost</i>	Širina pojasa monitora.
HorizSync <i>raspon</i>	Valjane frekvencije horizontalne sinkronizacije (u kHz). To može biti raspon ako imate multysync monitor ili slijed pojedinačnih vrijednosti za monitor stalne frekvencije osvježavanja.
VertRefresh <i>raspon</i>	Navodi frekvencije vertikalnog osvježavanja. Mogu biti navedene kao raspon ili kao slijed pojedinačnih vrijednosti, poput vrijednosti za HorizSync.
Gamma vrijednost	Vrijednost gama korekcije za vaš monitor.
ModeLine <i>vrijednosti</i>	Navodi slijed vrijednosti za svaku razlučivost koja će se na tom monitoru koristiti.

Tablica 21.7 Opcije odjeljka Monitor

Za svaku razlučivost trebate imati stavku *ModeLine* u odjeljku *Monitor*. Stavka ima sljedeći oblik:

ModeLine "name" *dotclock* Horizontal *Freq* Vertical *Freq* 

Vodoravne i okomite frekvencije su slijed četiri vrijednosti izraženih u kHz. Većinu vrijednosti možete saznati pokretanjem programa *xf86Config* (o kojem se govori kasnije, u članku "Pokretanje programa *xf86Config*" ili iz raznih dokumentacijskih datoteka sadržanih u paketu XFree86). Za prvu je probu najbolje unijeti standardnu konfiguraciju iz dokumentacije i zatim X-u dopustiti isprobavanje vašeg sustava za podesnije vrijednosti.

**Odjeljak** *Device* XFree86 treba odjeljak *Device* radi opisa grafičke kartice u sustavu. Odjeljak *Device* bi za Standard VGA izgledao ovako:

```
Section "Device"

Identifier "SVGA"

VendorName "Trident"

BoardName "TVG89"

Chipset "tvga8900c"

VideoRam 1024

Clocks 25.30 28.32 45.00 36.00 57.30 65.10 50.40 39.90

Option ...

EndSection
```

Jedine vrijednosti koje bi moglo biti teže saznati su vrijednosti radnog takta. Vaša grafička kartica te vrijednosti koristi za stvaranje signala koji pak pružaju različite frekvencije potrebne za prikaz informacija na vašem monitoru. Ako u tim vrijednostima previše pogriješite, mogao bi vam eksplodirati monitor. Tu vrijednost možete dobiti pokretanjem X-a s posebnim parametrom, *-probeonly*, koji X-u omogućuje pretraživanje vašeg sustava vez velike opasnosti od oštećenja sustava (o tom se parametru govori kasnije u ovom poglavlju). X zatim stvara izvještaj s većinom vrijednosti potrebnih za vašu konfiguraciju.

Vaš poslužitelj može zahtijevati i neke dodatne parametre. Te su dodatne stavke u odjeljku *Device* detaljno objašnjene na odgovarajućim *man* stranicama za vaš poslužitelj.

**Odjeljak Screen** Vaša datoteka XF86Config može sadržavati mnoge *Monitor* i *Device* stavke. Te su stavke u odjeljku *Screen* povezane zajedno radi stvaranja vaše X radne površine za vaš X poslužitelj. Slijedi primjer odjeljka *Screen*:

```
Section "Screen"
Driver
           "vga2"
           "SVGA"
Device
Monitor "Sanyo 1450 NI"
Subsection "Display"
         8
Depth
           "1024x768" "800x600" "640x480"
Modes
ViewPort
          0 0
Virtual
           1024 768
EndSubsection
EndSection
```

Odjeljak *Screen* uzima identifikacijska imena iz odjeljaka *Device* i *Monitor*. Vrijednost *Driver* pokazuje koji ste X poslužitelj pokrenuli i može imati jednu od sljedećih vrijednosti:

- Accel
- SVGA
- VGA16
- VGA2
- Mono

Unutar odjeljka *Screen* su prikazani pododjeljci koji opisuju različite *modove* raspoložive za određenu razlučivost. Svaka *Mode* vrijednost upućuje natrag na svaku *ModeLine* vrijednost definiranu u odjeljku *Monitor*.

X se pokreće na položaju određenom s vrijednošću *ViewPort*. Vrijednost 0, 0 X-u nalaže pokretanje na položaju 0,0 u gornjem lijevom kutu zaslona.

S vrijednošću *Virtual* možete definirati virtualni zaslon koji je veći od vašeg stvarnog zaslona. Ako odredite veći zaslon, X će zaslon po potrebi automatski pomicati kada pokazivač pomaknete na položaj izvan raspona stvarnog zaslona.

SAVJET Mnogi programi koje ćete naći na Internetu pretpostavljaju uporabu miša s tri tipke i veličinu zaslona 1152x900. Ta je veličina zaslona uobičajena veličina za zaslone na Sun radnim stanicama. Stoga ćete za oponašanje takvog sustava u odjeljku *Pointer* trebati navesti *Emulate3Buttons*, a u pododjeljku *Display* odjeljka *Screen* navesti *Virtual* 1152 900.

# Pokretanje programa xf86Config

Nakon pokretanja programa SuperProbe i stvaranja osnovne datoteke XF86Config možete pokrenuti program *xf86Config* za stvaranje *config* datoteke za vaš sustav. Najprije provjerite da niste u direktoriju /usr/X11R6/lib/X11, jer je to mjesto na kojem će X najprije potražiti datoteku XF86Config, a vi ne želite da vam podaci u datoteci koju ste upravo stvorili budu prebrisani. Za pokretanje programa *xf86Config* izdajte sljedeću naredbu:

/usr/X11R6/bin/xf86Config

Program *xf86Config* postavlja mnoga pitanja o vašem sustavu, koja koristi za popunjavanje raznih odjeljaka u datoteci XF86Config. Nakon što program završi trebate provjeriti da li te vrijednosti sliče onima koje ste prikupili pri stvaranju vaše inačice te datoteke. Jedine stavke za koje ćete trebati pomoć su vrijednosti radne frekvencije za vaš monitor. Za te vrijednosti možete dobiti pomoć od samog X-a.

# Rad X-a u načinu rada -probeonly

Izvršavanjem X-a u posebnom načinu rada program stvara datoteku s podacima o vašem cijelom sustavu. Informacije u toj datoteci možete koristiti za dovršavanje vaše datoteke XF86Config. Za pokretanje X-a u posebnom, ispitnom (*probe-only*) načinu rada unesite ovu naredbu:

```
X -probeonly > /tmp/x.value 2>&1
```

Ova naredba izlaz iz X-a preusmjerava u datoteku /tmp/x.value. To je ASCII datoteka koju možete uređivati s bilo kojim ASCII editorom, poput *vi* editora. U ovoj datoteci možete provjeriti informaciju o radnom taktu i zatim je kopirati i umetnuti u vašu datoteku XF86Config radi dovršavanja vaše konfiguracijske datoteke za X.

**Vidi** "Kopiranje, rezanje i uljepljivanje", **str. 197.** 

Sada datoteku koju ste stvorili kopirajte u jedan od direktorija koje XFree86 pretražuje. Datoteku ćete vrlo vjerojatno moći kopirati uporabom ove naredbe:

```
cp XF86Config /usr/X11R6/lib/X11/
```

Sada ste spremni za pokretanje vašeg X poslužitelja naredbom startx.

# Uporaba datoteka resursa X Windowsa

Za pokretanje i uporabu X-a trebate barem *startup* datoteku, nazvanu .xinitrem, koja pruža unaprijed zadane postavke koje X koristi tijekom rada. Za premošćenje unaprijed zadanih postavki možete koristiti osobnu datoteku nazvanu .Xresources koju stavljate u vaš polazni direktorij. Linux unaprijed zadanu .xinitre pribavlja u /etc/X11/xinit/xinitre, premda tu datoteku možete naći i u /usr/lib/X11/xinit/xinitre. O više informacija o ovim datotekama pogledajte *man* stranice za *startx* i *xrdb*.

# Odavde...

U ovom ste poglavlju učili o XFree86 primjeni XR11R6 standarda X Windowsa. Saznali ste razliku između klijent i poslužitelj aplikacija i po čemu se one razlikuju od klijent/poslužitelj aplikacija za osobna računala. Sljedeća poglavlja pružaju više informacija:

- 3. poglavlje, "Instaliranje Red Hata" objašnjava kako instalirati Red Hat distribuciju Linuxa, koja X Windowse instalira kao dio instalacijskog postupka.
- 6. poglavlje, "Nadogradnja i instaliranje softvera s RPM" pokazuje kako za instalaciju novih softverskih sustava na vaš Linux okvir koristiti program *pkgtool*. Taj je program koristan za instalaciju X-a ako ga niste instalirali s osnovnim Linux sustavom.
- 8. poglavlje, "Uporaba vi editora" pokazuje kako koristiti vi. Da biste X-u pružili valjane informacije za rad na posebnom hardveru koji koristite, trebate urediti vašu X konfiguracijsku datoteku. vi je izvrstan editor za uređivanje raznih datoteka potrebnih X-u.
- 21. poglavlje, "Instaliranje X Windows sustava" opisuje kako instalirati XFree86 inačicu X-a za Linux, što trebate obaviti prije instaliranja X Windowsa.

# poglavlje 222

# Uporaba X Windowsa

(Steve Burnett)

# U ovom poglavlju

Kretanje u X Windowsima Programi za upravljanje prozorima u Linuxu Uporaba X aplikacija u Red Hatu Uporaba X Windows aplikacija sa Slackwareom 96 Zabavljanje s DOOM-om za Linux Ako poznajete druga grafička korisnička sučelja (GUI), poput Microsoft Windowsa ili korisničkog sučelja za Macintosh, u X Windowsima ne biste trebali naći velike razlike. X Windowsi korisniku prikazuju nekoliko prozora, od kojih svaki prikazuje izlaz iz X Windows aplikacije, nazvane klijent. Klijent se može izvršavati na korisničkom osobnom računalu, što je najvjerojatniji slučaj kod Linuxa, ili na drugoj radnoj stanici na mreži.

BILJEŠKA Upamtite da s X Windowsima izraz klijent/poslužitelj ima obrnuto značenje od uobičajenog značenja klijenta i poslužitelja.

Na koji se način krećete po X Windowsima uvelike ovisi o programima za upravljanje prozorima. Većina prozora koristi pokazivač nazvan kursor, koji kretanjem po zaslonu pokazuje gdje trenutno radite. Kursor može imati mnogo oblika, ovisno o tome što radite i koji program za upravljanje prozorima koristite.

# Kretanje u X Windowsima

X Windowsi, poput većine grafičkih sučelja, omogućuju ulaz preko tipkovnice i pokazivačkog uređaja, obično miša. U pravilu, ulaz može prihvatiti samo aktivan prozor. Aktivan prozor obično ima nešto drukčiji izgled (na primjer, istaknuti rub) nego neaktivni prozori.

Način na koji prozor možemo učiniti aktivnim ovisi o programu za upravljanje prozorima. Neki window manageri prozoru omogućuju da postane aktivan već samim pomicanjem kursora iznad njega; drugi zahtijevaju da mišem kliknete na prozor, kao u Microsoft Windowsima.

# Uporaba izbornika

Mnoga grafička korisnička sučelja na današnjim osobnim računalima pružaju padajuće izbornike i izbornike prečica. Postojanje takvih stavki ponovno ovisi o programu za upravljanje prozorima, uključujući i dostupan izbor naredbi u izbornicima. Većina window managera za X Windowse nema glavnu traku izbornika uzduž vrha monitora. Umjesto toga, oni koriste plutajući izbornik. Taj plutajući izbornik obično dozovete klikom na prazan dio radne površine. Tipku miša zadržite pritisnutom i kursor povlačite preko raznih opcija izbornika. Kada pronađete željenu opciju izbornika, samo otpustite tipku, što je vrlo slično kretanju po izbornicima na Macintoshu i dosta različito od kretanja po izbornicima u Microsoft Windowsima.

# Uporaba virtualnih terminala u X Windowsima

Vaš se X poslužitelj izvršava na virtualnom terminalom dodijeljenom od Linuxa. Taj je terminal dodijeljen sedmom virtualnom terminalu, koji možete dosegnuti kombinacijom <Ctrl-Alt-F22> iz znakovnog terminala. Druge terminale možete iz X Windowsa dosegnuti kombinacijom tipki <Ctrl-Alt-Fx>, gdje x predstavlja broj virtualnog terminala kojem želite pristupiti. Premda i pristupanje drugim virtualnim terminalima može biti spretno, X Windowsi vam omogućuju pokretanje emulatora znakovnih terminala, nazvanih xterm sesije.

Ako se vaš X poslužitelj izvršava, za prelazak s X poslužitelja na virtualni terminal BILJEŠKA trebate koristiti kombinaciju <Ctrl-Alt-Fx>. Kombinaciju <Alt-Fx> možete i dalje koristiti za prelaske između virtualnih terminala.

# Programi za upravljanje prozorima u Linuxu

Kako je ranije u poglavlju rečeno, X Windowsi ne određuju posebno program za upravljanje prozorima (window manager). Izgled X Windowsa i osjećaj rada u njima su posve prepušteni na izbor korisniku. Pod vašim je nadzorom gotovo svaka pojedinost u ponašanju grafičkog korisničkog sučelja. U tom duhu, Linux ne pruža samo jedan program za upravljanje prozorima u X Windowsima, premda je u instalaciji za Red Hat i Slackware unaprijed podešena instalacija programa *fvwm* kao unaprijed zadanog upravljača prozorima. U tablici 22.1 su navedeni neki od raznih *window managera* raspoloživih za Linux.

Ime	Opis
twm	Tomov window manager
fvwm	Virtualni window manager za X11
fvwm95	Virtualni window manager za X11 koji prilično nalikuje Microsoftovim Windowsima 95
mwm	Motif window manager
olwm	Openlookov window manager, temeljen na Sunovom Open Looku
olvwm	Openlookov virtualni window manager
Enlightenment	Omiljen i elegantan prozorski manager
CDE	Common Desktop Environment, X GUI koji je prilagođen mnogim UNIX-ima
KDE	K Desktop Environment, besplatna varijacija CDE-a

#### Tablica 22.1 Neki od window managera raspoloživih za Linux

#### twm

Program *twm* za upravljanje prozorima u sustavu X Windows pruža naslovne trake, oblikovane prozore, nekoliko obrazaca upravljanja ikonama, korisnički definirane makro funkcije, klik za upisivanje i pokazivačem upravljano žarište tipkovnice, korisnički određene tipke i povezivanja tipkom miša. Ovaj je program obično pokrenut programom za upravljanje korisničkom *sesijom* ili skriptom koja se izvodi pri pokretanju sustava. Kada se koristi iz *xdm*-a ili *xinit*-a bez programa za upravljanje *sesijom*, *twm* se često izvršava u prednjem planu kao posljednji klijent. Kada se na taj način izvšava, izlaz iz *twm*-a uzrokuje zaustavljanje *sesije* (to jest, odjavljivanje).

Unaprijed je zadano da su aplikacijski prozori okruženi "okvirom", s naslovnom trakom na vrhu i posebnim rubom oko prozora. Naslovna traka sadrži ime prozora, pravokutnik koji je osvijetljen kada prozor prima unos s tipkovnice, a na lijevom i desnom kraju naslovne trake su funkcijski okviri poznati kao *title buttons (gumbi naslovne trake)*. Klik s Button1 (obično lijeva tipka miša, osim ako je to promijenjeno s *xmodmap*-om) na gumb naslovne trake obično doziva funkciju pridruženu tom gumbu. Prozori su, prema unaprijed zadanim postavkama sučelja, *ikonizirani* (smanjeni na veličinu ikone) klikom na lijevi gumb naslovne trake koji izgleda poput točke. I obratno, prozori su *deikonizirani*, odnosno povećani na najveću veličinu, klikom na pridruženu ikonu ili stavku u upravitelju ikona (*icon manageru*).

#### 442 22. poglavlje Uporaba X Windowsa

Promjena veličine prozora se vrši klikom na desni gumb naslovne trake (na kojem je crtež grupe okvira), povlačenjem pokazivača iznad ruba koji treba pomaknuti i otpuštanjem tipke miša kada obris prozora namjestimo na željenu veličinu. Slično tome, prozori se premještaju klikom na naslovnu traku, povlačenjem obrisa prozora na novi položaj i zatim otpuštanja tipke miša kada je obris na željenom položaju. Za podizanje prozora bez premještanja treba samo kliknuti na naslovnu traku.

Pri stvaranju novih prozora *twm* poštuje svaku informaciju o veličini i položaju prema zahtjevu korisnika. Inače je prikazan obris unaprijed zadane veličine prozora, njegova naslovna traka i crte koje prozor dijele u rešetku tri-sa-tri koja prati pokazivač. Svaka tipka miša obavlja drukčiju radnju:

- Klik na Button1 prozor postavlja na trenutnom položaju i daje mu unaprijed zadanu veličinu.
- Klik na Button2 (obično srednja tipka miša) i povlačenje obrisa prozor ostavlja na njegovom trenutnom položaju, ali omogućuje promjenu veličine stranica kako je gore opisano.
- Klik na Button3 (obično desna tipka miša) prozoru daje njegov trenutni položaj, ali ga pokušava učiniti dovoljno dugačkim da dodirne dno zaslona.

## fwm

Program za upravljanje prozorima *fwm* za X11 jest izvedenica *twm*-a. Program je preoblikovan za smanjenje potrošnje memorije, pružanje trodimenzionalnog izgleda okvirima i pružanje jednostavne virtualne radne površine. Zauzeće je memorije procijenjeno na oko pola do trećine memorije potrebne za *twm*, prvenstveno zahvaljujući preoblikovanju *twm*ovog nedjelotvornog načina pohrane naredbi pridruženih tipkama miša (*mouse bindings*). Osim toga, uklonjene su i mnoge mogućnosti podešavanja postavki *twm*-a.

XFree86 pruža virtualni zaslon čije djelovanje može biti zbunjujuće kada se koristi s virtualnim window managerom *twm*. S XFree86, prozori koji se pojavljuju na virtualnom zaslonu se stvarno iscrtavaju u video memoriji pa je veličina virtualnog zaslona ograničena raspoloživom video memorijom.

S *fwm*-ovom se virtualnom radnom površinom prozori koji se ne pojavljuju na zaslonu ne iscrtavaju stvarno u video RAM. Veličina je virtualne radne površine ogrančiena na 32.000x32.000 piksela. Nije praktično koristiti virtualnu radnu površinu s više od pet veličina vidljivog zaslona u svakom smjeru.

# BILJEŠKA S virtualnom je radnom površinom zauzeće memorije ovisno o broju trenutno otvorenih prozora. Veličina radne površine na to nema utjecaja.

Dok se s *fwm*-om tek upoznajete, preporučljivo je da virtualni zaslon u XFree86 onemogućite podešavanjem veličine virtualnog zaslona na veličinu stvarnog zaslona. Kada *fwm* bolje upoznate, možda ćete virtualni zaslon u XFree86 poželjeti ponovno omogućiti.

*fwm* pruža višestruke virtualne radne površine korisnicima koji ih žele koristiti. Zaslon je samo okvir kroz koji gledamo radnu površinu koja je veća (ili iste veličine) od zaslona. Može se pristupiti i na više udaljenih radnih površina. Temeljni je princip jedna radna

površina za svaki projekt ili po jedna radna površina za svaku aplikaciju kada su aplikacije za prikaz različite. Budući da svaka radna površina može biti veća od stvarnog zaslona, lako se mogu promatrati i prozori koji su veći od zaslona ili velike skupine povezanih prozora.

Veličina svake virtualne radne površine mora biti određena pri podizanju sustava, Unaprijed je podešena na trostruku veličinu stvarnog zaslona. Sve virtualne radne površine moraju biti iste veličine. Ukupan broj zasebnih radnih površina ne treba biti naveden, ali je ograničen na približno ukupno 4 milijarde. Svi prozori na trenutnoj radnoj površini mogu biti prikazani u dojavniku (*pager*), minijaturnom prikazu ili na trenutnoj radnoj površini. Prozori koji nisu na trenutnoj radnoj površini mogu biti stavljeni na popis prozora (sa svojim geometrijama) koji je dostupan kao izbornik prečica. Pod *geometrijama* su navedene koordinate i broj piksela potreban za prozor pod X *window managerom*.

*Sticky windows (ljepljivi prozori)* su prozori koji nadvisuju virtualnu radnu površinu "zalijepivši se na staklo zaslona". Oni uvijek ostaju vidljivi na zaslonu. To je prikladno za programe poput satova i *xbiff*ova, pa trebate pokrenuti samo jedan takav uslužni program i on će uvijek ostati s vama.

BILJEŠKA Aplikacija xbiff vas obaviještava kada pristigne nova pošta.

Geometrije su prozora navedene u odnosu na trenutni okvir prikaza (*viewport*) - to jest *xterm-geometry* +0+0 se uvijek pojavljuje u gornjem lijevom kutu vidljvog dijela zaslona. Dopustivo je navesti i geometrije koje prozore postavljaju na virtualnu radnu površinu, ali izvan onog dijela koji se trenutno vidi na zaslonu. Na primjer, ako je vidljivi zaslon 1000x1000 piksela, veličina je radne površine tri sa tri, a trenutni je okvir prikaza u gornjem lijevom kutu radne površine, pozivanje *xterm-geometry* +1000+1000 prozor postavlja odmah iza donjeg desnog kuta zaslona. Prozor se tada može vidjeti pomicanjem miša u donji desni kut zaslona i čekanjem da se prozor pomakne unutar zaslona i postane vidljiv. Prozor možete mapirati samo na aktivnu radnu povšrinu, dok na neaktivnu ne možete.

Geometrija određena kao *xterm-geometry -5-5* obično donji desni kut prozora postavlja pet piksela od donjeg desnog kuta vidljivog dijela zaslona. Takve negativne otklone geometrije prozora ne podržavaju sve aplikacije.

## fvwm95

Program za upravljanje prozorima *fvwm95* za X11 je "hakerska preinaka temeljena na fvwm2.x". Cilj je razvijatelja bio oponašanje glavnih značajki GUI-ja dobro poznatog operativnog sustava da bi se korisnici ugodnije osjećali u UNIX okružju i radi izbjegavanja napuhavanja jednostavnog i čistog GUI koda fvwm-a. Dodatne informacije potražite na http://mitac11.uia.ac.be/html-test/fvwm95.html.

#### olwm

Program za upravljanje prozorima *olwm* za X Windows sustav primjenjuje dijelove grafičkog korisničkog sučelja Openlook. To je standardan program za upravljanje prozorima za Sunov proizvod Open Windows, ali ispravno radi sa svakim X11 sustavom,

uključujući XFree86. Jedini su zahtjevi za pokretanje *olwm*a da poslužitelj na raspolaganju ima *OPEN LOOK* glyphove i korsorska pisma, što bi trebao biti slučaj ako ste instalirali sva pisma za X Windowse.

# Enlightenment

*Enlightenment* je omiljen i sjajno napisan program za upravljanje prozorima. Stabilan je i brz. Premda je ispočetka temeljen na *rvwm* radu, njegove su novije inačice u potpunosti napisane od nule. Tvorac *Enlightenmenta* ima Web lokaciju na **http://www.rasterman.com** - ona sadrži više informacija nego što je u ovu knjigu moglo biti uključeno.

# CDE

Common Desktop Environment (*CDE*) je komercijalni pokušaj prilagodbe uobičajene radne površine na većinu inačica UNIXa. Na primjer, *CDE* je prilagođen za Sun Solaris, IBM AIX i Hewlett-Packardov HP-UX (između ostalih). Za uobičajenu UNIX radnu površinu *CDE* nije loš. TriTeal je *CDE* prilagodio za Linux i njihovu inačicu (temeljjenu na OSF Motif 1.2.5) preprodaje Red Hat Software u svome paketu za Linux. Za više informacija o *CDE*-u za Linux idite na TriTealovu Web lokaciju na **http://www.triteal.com**. Neke opće informacije o *CDE*-u za Linux možete pronaći na Usenet novinskoj grupi comp.unix.cde, a njihova se datoteka Frequently Asked Questions nalazi na Webu na **http://www.pobox.com/čburnett/cde**/.

# KDE

K Desktop Environment (*KDE*) je veliki *freeware* projekt zamišljen za stvaranje integriranog radnog okružja sliičnog *CDE*-u, ali razvijen i izdan u potpunosti pod GNU Public License (GPL).

Glavne prednosti *KDE*-a uključuju jaku podršku internacionalizaciji, ugrađen sustav pomoći, i standardizirani izgled i osjećaj za najrazličitije aplikacije. O više informacija pogledajte **http://www.kde.org**/.

# Uporaba X aplikacija u Red Hatu

Red Hat nije ništa štedio pri stvaranju distribucije koja je izvrsno prilagođena X-u. U stvari, komercijalna inačica Red Hata sadrži komercijalni X poslužitelj nazvan Metro-X licenciran za jednog korisniika. Kada X pod Red Hatom pokrenete ovom naredbom:

```
startx &
```

ugledat ćete zaslon koji prilično podsjeća na okružje Microsoftovih Windowsa 98 (vidi sliku 22.1).

Gumb Start sadrži stavke izbornika za mnoge korisne Linux programe, sustavske naredbe i procese. Kao i u većini X instalacija, do tih istih naredbi možete doći i klikom s lijevom i/ili desnom tipkom miša iznad radne površine. Slika 22.2 prikazuje stavke izbornika Start i sve su stavke opisane u tablici 22.2.

Slika 22.1 X pod Red Hatom zapanjujuće nalikuje omiljenom korisničkom sučelju Microsoftovih Windowsa 98.



#### Slika 22.2

Red Hat Linux preko izbornika Start pruža laki pristup lokacijama za mnoge usluge.

8	Red Hat Linux	
	<u>N</u> ew shell	
$\mathbb{X}$	<u>Applications</u>	►
Ŷ	<u>U</u> tilities	►
1	Multimedia	►
۲	<u>G</u> ames	►
匂	Hosts	►
,ø	System Utilities	►
匂	Window Operations	►
P	Preferences	►
	Screensaver	►
Ê	<u>L</u> ock Screen	►
2	About Fvwm	
3	Help Fvwm	
	E <u>x</u> it Fvwm	►

Tablica 22.2 S	itavke izbornika Start
Stavka	Opis
<u>N</u> ew Shell	Korisniku pruža novi prozor naredbene ljuske - to jest prozor xterm.
<u>Applications</u>	Pruža pristup do različitih aplikacija, poput pine (e-mail), xpaint (grafika) i irc (chat).

nastavlja se

Tablica 22.2 Nastavak				
Stavka	Opis			
<u>U</u> tilities	Pruža uslužne programe poput kalkulatora, kalendara, xterm emulatora u boji i pristupa man stranicama.			
<u>M</u> ultimedia	Pruža audio CD player i audio mikser.			
<u>G</u> ames	Pruža arkadne igre poput Tetrisa i grafičkih avantura poput DOOM-a.			
H <u>o</u> sts	Pruža grafički pristup do drugih domaćina na vašoj mreži ili na Internetu.			
System Utilities	Pruža pristup do uslužnih programa za pristup sustavu kao root i za valjano upravljanje prozorima.			
Window Operations	Pruža stavke izbornika za zatvaranje, "ubijanje" i premještanje prozora po radnoj površini.			
<u>P</u> references	Omogućuje vam prilagođavanje vaše X radne površine po volji.			
S <u>c</u> reensaver	Pruža vam izbor slika za vaš zaštitnik zaslona koji se aktivira nakon određenog vremena neaktivnosti.			
Lock Screen	Pruža vam izbor uzoraka šara za zaslon koje možete koristiti kada svoj zaslon zaključate. Zaključavanje se zaslona događa kada odaberete jedan od uzoraka. Za otključavanje zaslona samo unesite svoju lozinku.			
A <u>b</u> out Fvwm	Prikazuje okvir za dijalog s informacijama o fvwm window manageru.			
<u>H</u> elp Fvwm	Pomoć za fvwm prikazuje u HTML pretraživaču.			
E <u>x</u> it Fvwm	Omogućuje vam zatvaranje X-a i vraćanje u terminal iz kojeg je X pokrenut ili ponovno pokretanje X poslužitelja.			

## nxterm

Odabirom stavke izbornika New Shell pokrećemo *sesiju xterm*, koja je u Red Hat Linuxu nazvana *nxterm session. xterm* je uobičajena X Windows aplikacija koja oponaša uobičajeni video terminal poput DEC-a vt100. Kada pokrenete sesiju *xterm*, iz nje možete pokrenuti svaki program s naredbenim retkom ili izvršiti bilo koju Linux naredbu baš kako biste to uradili u bilo kojem virtualnom terminalu koji je potekao iz Linuxa. Na slici 22.3 je prikazana *xterm* sesija.

#### XV

*xv* je Red Hatov program za hvatanje zaslona. Za razliku od većine Linux aplikacija, ovaj je program *shareware*. Slika 22.4 prikazuje glavni okvir za dijalog ove aplikacije.



Pod X-om je pokretanje nove ljuske s naredbenim retkom lako izvesti.



#### Slika 22.4

xv pruža cjelovit program za hvatanje zaslona i pretvorbu grafičkog formata datoteka pod X-om.

EE KV CO	ntrols					同居
	Disp	Display 24/8 Bit		t	Algorithms	
	Ro	ot	Window	s	Ima	age Size
xcalc.bmp					습	Next
startscr.bmp						Drev
xaps.bmp						1107
xterm.bmp						Load
						Save
						Print
						Delete
					ۍ	4 files
Successfully	wrote '/ro	ot/xterm.b	mp' (8789	4 byte	s)	
*	la X		5 0	\$	ŧ	Grab
• A	Crop	UniCrop	AutoCrop	Abo	out XV	Quit

BILJEŠKA Shareware programi su programi koje možete besplatno učitati/presnimiti, međutim, ako ustanovite da vam je program koristan i želite ga koristiti, od vas se zahtijeva da ga nakon određenog vremena uredno platite autoru. Ipak, shareware programi su u pravilu prilično jeftini.

Gumbi uzduž desne strane glavnog okvira za dijalog su nam najkorisniji. Njihovo je djelovanje opisano u tablici 22.3. Glavni popis datoteka pruža imena svih grafičkih datoteka sada raspoloživih programu.

Tablica 22.3	xv-ovi naredbeni gumbi
Gumb	Opis
Next	Odabir sljedeće datoteke u okviru s popisom datoteka.
Prev	Odabir prethodne datoteke u okviru s popisom datoteka.
Load	Učitavanje datoteke s diska u program.
Save	Tenutno uhvaćenu sliku sprema u datoteku na disku. Možete birati između sljedećih vrsta slika: GIF, JPEG, TIFF, PostScript, PBM (sirova), PBM (ASCII), X11 bitmapa, XPM, BMP, Sun rasterska datoteka, IRIS RGB (24-bit), Fits i PM.
Print	Ispisivanje trenutno odabrane slikovne datoteke.
Delete	Brisanje trenutno odabrane slikovne datoteke.

Gumb Grab u donjem desnom kutu okvira za dijalog vam omogućuje hvatanje bilo kojeg dijela radne površine. Klikom na taj gumb otvaramo okvir za dijalog *xv grab* sliku 22.5).

Slika 22.5	🔲 🖂 🖾 xv grab	<b>F E</b>
Na raspolaganju su vam različiti načini	Grab: after delay, Left button gral grabs a rectangular area, Right bu	bs a window, Middle button itton cancels.
za hvatanje bilo kojeg dijela zaslona pod	AutoGrab: after delay, grabs the v positioned in. Delay must be nor	window the cursor is n—zero.
xv-om.	Delay: 0 seconds	Hide XV windows
	Grab	AutoGrab Cancel

Objekt na zaslonu koji želite uhvatiti odabirete pomoću miša. Za hvatanje prozora možete kliknuti na gumb Grab i zatim lijevom tipkom miša kliknuti na prozor koji želite zgrabiti. Možete i podesiti vrijeme odgode (Delay), kliknuti na gumb AutoGrab i zatim kursor miša postaviti u prozor. U oba slučaja *xv* hvata sliku i prikazuje je u vlastitom prozoru. Za daljnju obradu i spremanje slike možete koristiti kontrole u glavnom okviru za dijalog.

# Uporaba X Windows aplikacija sa Slackwareom 96

Na Internetu postoji mnoštvo aplikacija za X Windowse. Sljedeći članci pružaju sažet pregled nekoliko X aplikacija dostavljenih ili s Slackware distribucijom ili na Slackware CD-ROM-u u direktoriju /contrib. Nekoliko ih je priloženo i s Red Hat distribucijom ili su na raspolaganju za skidanje s vaše omiljene GNU Web lokacije.

## xterm

Slackvareov *xterm* je isti program kao i Red Hatova naredba New Shell (*nxterm*). *xterm* je uobičajena X Windows aplikacija koja oponaša uobičajeni video terminal poput DEC-a vt100. Nakon pokretanja *xterm* sesije možete pokrenuti bilo koji program s naredbenim

retkom ili izvršiti bilo koju Linux naredbu baš kao što biste to uradili na bilo kojem od virtualnih terminala dostavljenih s Linuxom. Slika 22.6 prikazuje *xterm* sesiju.

Slika 22.6

xterm pruža prikladan pristup do ljuske s naredbenim retkom.

	-					xterm						• •
Ĩ		drwxr-xr-x	7	root	root	1024	Jan 10	18:12	openvin,	1		
		Traxraxrax	1	root	root	13	Jan 10	17:41	preservo	3 ->	/var/pre	serve
		, dr⊌xr−xr−x	2	root	bin	1024	Jan 10	17:49	sbin/			
		drwxr-xr-x	2	root	root	1024	Nov 25	1993	share/			
		lrwxrwxrwx	1	root	root	10	Jan 10	17:41	speel ->	> /va	ar/spool/	
		drwxr-xr-x	6	root	root	1024	Jan 10	18:08	src/			
		Inwxnwxnwx	1	root	root	8	Jan 10	17:41	tap -> ,	/var,	/tap/	
		darkstar:/usr	°#	ps	001040110							
		PID ITY SI	HI	TIME	COMMAND							
		40 V01 S		0:00	-bash	++. 0 004		-1-				
		41 002 5		0:00	/sbin/getty	tty2 3840	JU cons	-1- 016				
		42 003 5		0:00	/sbin/getty	++4 2240	0 cons	olo				
		40 004 3		0.00	/sbin/getty	+++++++++++++++++++++++++++++++++++++++	10 cons					
		44 V00 0		0.00	/shin/getty	++u6 38/0	10 cons	ole				
		52 v01 S		0:00	sh /usr/X11	/hin/start	ty cons	010				
		53 v01 S		0:00	xinit /usr/	(386/1ib/)		it/xin	itrc			
		55 v01 S		0:00	sh /usr/X386	3/1ib/X11	/xinit/	xinitr	c			
		58 v01 S		0:01	fvwm	,,						
		60 v01 S		0:01	/usr/bin/X1:	l/xterm -s	sb -sl	500 -j	-ls -fn	7×14	4	
		61 pp0 S		0:00	-bash	-		-				
		74 pp0 R		_0:00	ps							
		darkstar:/usr	r#	•								

Program *xterm* je emulator terminala za sustav X Windows. On pruža terminale usklađene s DEC-om vt102 i Tektronixom 4014 za programe koji ne mogu izravno koristiti sustav prozora. Ako operativni sustav u kojem radi podržava terminalove sposobnosti promjene veličine, *xterm* te mogućnosti koristi za obavještavanje programa koji se izvršavaju u prozoru nakon svake njegove promjene veličine.

Terminali vt102 i Tektronix 4014 imaju svoje vlastite prozore pa u jednom možete uređivati tekst, a u drugom istovremeno pregledavati grafike. Za održavanje ispravnog odnosa - visine i širine prikaza u pikselima, Tektronixove su grafike ograničene na najveći okvir s Tektronix 4014 odnosom visine i širine koji stane u prozor.

Premda prozori s tekstovima i grafikom mogu biti istovremeno prikazani, prozor koji sadrži tekstualni kursor se smatra "aktivnim" prozorom za primanje ulaza preko tipkovnice i izlaz preko terminala. Aktivni prozor može biti odabran preko Escape sekvenci, izbornika vt Options u prozoru vt102 i izbornika Tek Options u prozoru 4014.

**Emulacije** *\$TERMCAP* stavke koje rade s *xterm*om uključuju *xterm*, *vt102*, *vt100* i *ANSI*. Varijabla okružja *\$TERMCAP* određuje vrstu terminala koji vaš sustav oponaša. *xterm* te stavke automatski traži u termcap datoteci baze podataka prema ovom redoslijedu i zatim postavlja varijable okružja *TERM* i *\$TERMCAP*.

BILJEŠKA O više informacija o podržanim termcap stavkama i Escape sekvencama pogledajte man stranicu za termcap.

Mnoge od posebnih *xterm* značajki mogu biti preinačene pod programskom kontrolom pomoću skupa Escape sekvenci različitih od standardnih *vt102* Escape sekvenci.

Emulacija Tektronix 4014 je također prilično dobra. Podržane su četiri različite veličine pisama i pet različitih vrsta linija. Tektronixove tekstualne i grafičke naredbe *xterm* interno zapisuje, a slanjem Tektronixove Escape sekvence *COPY* mogu biti upisane u datoteku.

**Druge mogućnosti xterma** *xterm* automatski odabire kursor teksta kada pokazivač uđe u prozor i deselektira ga kada pokazivač izađe iz prozora. Ako prozor ima žarište, tekstualni je kursor odabran bez obzira gdje je pokazivač.

U vt102 načinu rada Escape sekvence aktiviraju i deaktiviraju spremnik zamjenskog zaslona, koji je iste veličine kao i prikazna površina prozora. Kada je aktiviran, trenutni je zaslon spremljen i zamijenjen s zamjenskim zaslonom. Spremanje je redaka pomaknutih iznad vrha prozora onemogućeno sve dok se ne vrati normalan zaslon. Stavka *termcap* za *xterm* vizualnom editoru *vi* omogućuje prebacivanje na zamjenski zaslon za uređivanje te povrat zaslona na izlazu.

U vt102 ili Tektronix načinu rada Escape sekvence mijenjaju imena prozora.

**Uporaba miša s xtermom** Kada je prozor vt102 stvoren, *xterm* vam omogućuje odabiranje teksta i njegovo kopiranje unutar istog ili u drugim prozorima.

Funkcije su odabira pozvane kada su gumbi pokazivača korišteni bez *modifikatora* i kada se koriste s tipkom <Shift>. Dodijeljene funkcije tipkama i gumbima mogu biti izmijenjene preko baze podataka resursa.

Tipka Button1 na mišu (obično lijeva tipka) se koristi za spremanje teksta u spremnik isječenog teksta. Kursor pomaknite na početak teksta, pritisnite tipku, kursor odvucite na kraj područja i zatim otpustite tipku. Odabrani je tekst označen i spremljen u globalni spremnik isječenog teksta. Taj je tekst učinjen prvim odabirom u trenutku kada se tipka miša otpusti. Dvostruki klik odabire cijelu riječ, trostruki klik odabire red, četverostruki se klik vraća natrag na znakove i tako redom.

Tipka Button2 na mišu (obično desna tipka) uljepljuje tekst iz primarnog odabira, ako takav postoji. Inače se tekst umeće iz spremnika isječenog teksta, kao ulaz iz tipkovnice.

Rezanjem i uljepljivanjem dijelova teksta bez povlačenja novih redova možete, na primjer, uzeti tekst s nekoliko mjesta u različitim prozorima i oblikovati naredbu ljusci, ili uzeti izlaz iz programa i umetnuti ga u vaš omiljeni editor. Budući da je spremnik isječenog teksta dijeljen globalno između različitih aplikacija, trebali biste ga smatrati datotekom čiji vam

je sadržaj poznat. Emulator terminala i drugi tekstualni programi trebaju se prema spremniku isječenog teksta odnositi kao da je to tekstualna datoteka - to jest tekst je razgraničen s novim recima.

Područje pomicanja unutar prozora koji prikazuje *xterm* pokazuje položaj i količinu teksta trenutno prikazanu u prozoru u odnosu na količinu teksta koja je stvarno spremljena. Što je više teksta spremljeno (sve do najveće količine dopuštene u sustavu), to se veličina označenog područja smanjuje.

Klik na Button1 s pokazivačem unutar područja pomicanja pomiče sljedeći redak na vrh prozora. Klik na Button2 prikaz pomiče na položaj u spremljenom tekstu koji odgovara položaju pokazivača na kliznoj traci. Klikom na Button3 se gornji red u prozoru pomiče dolje na visinu pokazivača.

Za razliku od prozora vt102, prozor Tektronix ne dopušta kopiranje teksta. On, međutim, omogućuje Tektronix GIN način rada, u kojem se kursor mijenja iz strelice u križić. Pritisak na bilo koju tipku taj znak šalje na trenutne koordinate kursora u obliku križića. Pritisak na

Button1, Button2 ili Button3 daje slova *l*, *m* i *r*. Ako je pritisnuta i tipka <Shift>, poslano je odgovarajuće veliko slovo. Za razlikovanje tipke pokazivača od tipke slova postavljen je najviši bit znaka.

## xcalc

Slika 22.7 prikazuje *xcalc*, znanstveni kalkulator kao pomoćni program koji oponaša kalkulator TI-30 ili HP10C. Operacije se mogu izvršavati s tipkom miša Button1 ili, u nekim slučajevima, s tipkovnicom.





Mnoge uobičajene operacije kalkulatora imaju svoje ubrzivače na tipkovnici. Za zaustavljanje možete tipkom miša Button3 na kalkulatoru TI kliknuti na gumb AC ili na kalkulatoru HP kliknuti na gumb OFF. U TI načinu rada brojčane tipke, tipke +/-, =, +, -, \*, / čine točno ono što od njih i očekujete.

**BILJEŠKA** Operacije poštuju uobičajena pravila prvenstva. Tako unos 3+4\*5= daje rezultat 23, a ne 35. Ta pravila možete zaobići uporabom zagrada. Na primjer, unos (1+2+3)\*(4+5+6)= daje rezultat 90 (6\*15).

Broj prikazan na zaslonu kalkulatora se u cijelosti može odabrati radi uljepljivanja rezultata izračuna u tekst. Na tablici 22.4 su prikazane razne funkcije za TI emulaciju.

Tablica 22.4	TI emulacija
Tipka/namjena	Opis
1/x	Broj na zaslonu zamjenjuje s njegovim recipročnim brojem.
x^2	Kvadrat broja prikazanog na zaslonu.
SQR	Uzima kvadratni korijen broja na zaslonu.
CE/C	S prvim klikom briše broj prikazan na zaslonu bez brisanja stanja stroja, omogućujući ponovno unošenje broja ako ste učinili pogrešku. S dva se klika briše i stanje. (Klik na AC također briše zaslon, stanje i memoriju). Klikom na CE/C s tipkom Button3 isključujemo kalkulator i izlazimo iz xcalc.

Tablica 22.4	Nastavak
Tipka/namjena	Opis
INV	Aktiviranje obratne funkcije. Pogledajte pojedinačne funkcijske tipke za pojedinosti o njihovoj obratnoj funkciji.
sin	Izračunava sinus broja na zaslonu, kako je protumačen trenutnim DRG načinom rada (vidi DRG). Ako je obrnut (s uključenom INV - invert funkcijom), izračunava kosekans (arkus sinus).
cos	Izračunava kosinus. Kada je obrnut s tipkom INV, izračunava sekans (arkus kosinus).
tan	Izračunava tangens. Kada je obrnut, izračunava kotangens (arkus tangens).
DRG	Mijenja DRG način rada (MODE), koji je na dnu zaslona kalkulatora prikazan s DEG, RAD ili GRAD. U DEG načinu rada se pretpostavlja da brojevi na zaslonu predstavljaju stupnjeve, u RAD načinu rada su brojevi u radijanima, a u GRAD načinu rada su u gradima. Kada je obrnuta s tipkom INV, tipka DRG ima namjenu pretvaranja stupnjeva u radijane ili grade i obratno. Na primjer, kalkulator stavite u DEG način rada i unesite 45 INV DRG. xcalc će prikazati .2285398, što je 45 stupnjeva pretvoreno u radijane.
e	Konstanta e, čija je vrijednost 2. 718282.
EE	Koristi se za unos eksponencijalnih brojeva. Na primjer, za dobivanje -2.3E-4 unesite 2 . 3 +/- EE 4 +/
log	Izračunava logaritam (s bazom 10) broja na zaslonu. Kada je obrnut, potencira 10.0 na broj prikazan na zaslonu. Na primjer, unos 3 INV log daje rezultat 1000 (3 je tada potencija broja 10.0).
ln	Izračunava logaritam (baze e) broja na zaslonu. Kada je obrnut, potencira e na broj prikazan na zaslonu. Na primjer, unos e ln daje rezultat 1.
y^x	Potenciranje, gdje je y (lijevi broj) baza, a x (desni broj) eksponent. Na primjer unos 2 y^x 3 = daje rezultat 8, što je 2^3.
PI	Konstanta p, koja iznosi 3.14159265.
x!	Izračunava faktorijel broja prikazanog na zaslonu. Broj na zaslonu mora biti cijeli broj u rasponu od 0 do 500, međutim, ovisno o vašoj matematičkoj biblioteci, rezultat već mnogo prije može postati prevelik.
(	Lijeva zagrada.
)	Desna zagrada.
/	Dijeljenje.
*	Množenje.
-	Oduzimanje.
+	Zbrajanje.

Tipka/namjena	Opis
=	Izvršava izračun.
STO	Broj prikazan na zaslonu kopira u memorijsku lokaciju.
RCL	Broj iz memorijske lokacije kopira na zaslon.
SUM	Broj na zaslonu pridodaje broju u memorijskoj lokaciji.
EXC	Broj na zaslonu zamjenjuje s brojem u memorijskoj lokaciji.
+/-	Promjena predznaka.
	Decimalna točka.

U RPN ili HP načinu rada - tipke s brojevima, CHS (*change sign* - promjena predznaka), tipke +, -, \*, / i ENTER čine točno ono što od njih očekujete. Mnoge su od preostalih tipki jednake kao u TI načinu rada. Razlike su opisane u tablici 22.5.

Tablica 22.5	HP emulacija
Tipka/namjena	Opis
<	Tipka Backspace koja se može koristiti ako pri unosu broja učinite grešku. Briše znakove sa zaslona. Ako Backspace invertirate, briše se x registar.
ON	Briše zaslon, stanje i memoriju. Klikom na nju s Button3 isključujemo kalkulator i izlazimo iz xcalca.
INV	Preokreće značenje funkcijskih tipki. To je na HP kalkulatoru tipka f, ali xcalc ne prikazuje višestruke legende za svaku tipku. O više pojedinosti pogledajte pojedinačne funkcijske tipke.
10^x	Potencira 10.0 s brojem na vrhu stoga. Kada je obrnuta (s tipkom INV), izračunava logaritam (s bazom 10) broja na zaslonu.
e^x	Potencira e s brojem na vrhu stoga. Kada je obrnuta (s tipkom INV), izračunava logaritam (s bazom e) broja na zaslonu.
STO	Broj na vrhu stoga kopira u memorijsku lokaciju. Postoji 10 memorijskih lokacija. Željena se memorija određuje brojačnom tipkom koja slijedi iza ove tipke.
RCL	Broj iz navedene memorijske lokacije stavlja na vrh stoga.
SUM	Broj na vrhu stoga pribraja broju u određenoj memorijskoj lokaciji.
x:y	Zamjenjuje položaje dvaju najviših brojeva u stogu, registre x i y.
R v	Pomiče prikaz na niže brojeve u stogu. Kada je obrnuta (s tipkom INV), odmotava stog prema gore.
(prazne tipke)	Te su se tipke na HP-10C koristile za programske funkcije. Ova mogućnost nije prenesena u xcalc.

## xspread

Program *xspread*, prikazan na slici 22.8, je proračunska tablica koja se izvršava pod X Windowsima i koristi se kao javno dobro *(public domain)*. Instalira se s Slackware distribucijom. Za pokretanje ovog programa trebate koristiti X Windows terminal. (Stvaralaci *xspread*a rade na tome da mu omoguće uporabu ASCII zaslona ako za svoj rad ne može pronaći zaslon X Windowsa.) Reference Manual za *xspread* pruža cjelovitu dokumentaciju za ovaj program. LaTeX-ova se izvorna kopija za ovo uputstvo nalazi u datoteci xspread.tex.

#### Slika 22.8

xspread pod XFree86 pruža poznate mogućnosti proračunske tablice za korisnike Linuxa.



xspread podržava mnoge uobičajene mogućnosti proračunskih tablica, uključujući sljedeće:

- Unos u ćelije i uređivanje
- Veličina radnog lista od 2202 stupaca s neograničenim brojem redova
- Učitavanje i upisivanje datoteka
- Šifriranje datoteke
- Apsolutne i relativne reference na ćelije
- Podaci u ćeliji mogu biti brojevi i oznake (to jest, nizovi znakova)
- Lijevo ili desno poravnanje za oznake
- Umetanje i brisanje redova i stupaca
- Skrivanje i otkrivanje redova i stupaca
- Rasponi imena
- Ručno ili automatsko ponovno računanje
- Brojčani operatori (+, -, \*, /, ^, %)
- Relacioni operatori (<, <=, >, >=, =, !=)

- Logički (ili Boolean) operatori (&, ¦, -)
- Reference na funkcije
- Grafikoni (XY, stupci, stožasti stupci, tortni i linijski grafikoni)
- Matrične operacije (transponiranje, množenje, dodavanje, oduzimanje i inverzija)
- Postavljanje kursora s mišem
- Odabir stavki izbornika s mišem
- Reference na vanjske programe, koje su nazvane external functions (vanjske funkcije)

Ustroj i djelovanje proračunske tablice su slični - premda ne isti - omiljenim proračunskim tablicama poput Lotusa 1-2-3 i njegovih klonova. Kao i u drugim proračunskim tablicama, radni je prostor uređen u redove i stupce ćelija. Svaka ćelija može sadržavati broj, oznaku ili formulu koja proračunava broj ili oznaku.

Program možete pokrenuti sa ili bez navođenja datoteke koju treba učitati. Ta datoteka treba biti spremljena kao radni list. Ako je datoteka navedena u naredbenom retku, *xspread* će ju pokušati pronaći i učitati. Ako to uspije učiniti, *xspread* se pokreće sa sadržajem te datoteke u radnom prostoru. Ako datoteku ne uspije pronaći ili učitati ili u naredbenom retku nije navedena nikakva datoteka, *xspread* će se pokrenuti s praznim radnim prostorom.

Kao vodič za program proračunske tablice pokrenite jednu od demonstracijskih datoteka - *demo*, *demo-math* ili *demo-matrix* i u direktoriju doc pogledajte datoteku Sample-Run.

## Seyon

Seyon je, kako je prikazano na slici 22.9, potpuno opremljen cjeloviti komunikacijski paket za sustav X Windows. Neke su od njegovih mogućnosti ovdje navedene i zatim detaljno opisane u sljedećim člancima:

- Telefonski imenik
- Emulacija terminala
- Jezik za pisanje skripte
- Razni protokoli za prijenos, uključujući Zmodem
- Mogućnosti prevođenja

**Telefonski imenik** Telefonski imenik podržava neograničeni broj stavki. Imenikom se u cijelosti može baratati s mišem i ima mogućnost praćenja stanja poziva, vremenskog ograničenja, automatskog ponavljanja poziva, pozivanja više brojeva i cirkularno pozivanje zadanog popisa brojeva. Svakoj se stavci u telefonskom imeniku može podesiti vlastita brzina u baudima, bit maska i datoteka skripte. Telefonski imenik koristi obično ASCII tekst koji možete uređivati i iz Seyona. Seyon podržava i ručno biranje.

**Emulacija terminala** Emulacija terminala podržava DEC vt102, Tektronix 4014 i ANSI. Seyon svoju emulaciju terminala prepušta *xterm*u, tako da su preko Seyonovog prozora emulacije terminala raspoložive sve poznate *xterm* funkcije, poput spremnika (*buffera*) za pregled prijašnjeg teksta, mogućnost rezanja i uljepljivanja i vizualnog zvonca.

#### Slika 22.9

Premda je danas pristup Internetu vrlo važan, mnogi korisnici i dalje trebaju pristup BBS-ovima preko svojih modema.



Uporaba *xterma* znači da Seyon ima cjelovitiju emulaciju vt102 od bilo kojeg drugog UNIX ili DOS telekomunikacijskog programa. Sa Seyonom možete koristiti i druge programe za emulaciju terminala radi zadovoljavanja korisničkih potreba: na primjer, *xterm* u boji se može koristiti za ANSI u boji (omiljen na mnogim BBS sustavima), a *xvt* se može koristiti ako nam je memorija pomalo skučena.

**Jezik za pisanje skripte** Jezik za pisanje skripte možete iskoristiti za automatiziranje dosadnih poslova poput prijavljivanjak na udaljene domaćine. Seyonov prevoditelj skripte koristi obične tekstualne datoteke i sintaksa mu je slična onoj u *sh* s nekim dodatnim mogućnostima. Podržava mnoge poznate naredbe poput uvjetnog grananja s *if...else* i stvaranje petlje s *goto*. Skripte se mogu dodijeliti stavkama u telefonskom imeniku za automatsko izvršenje nakon uspostave veze.

**Prijenos datoteka** Seyon podržava neograničeni broj pretinaca za vanjske protokole prijenosa datoteka. Protokoli se aktiviraju s prijenosne konzole upravljane mišem, koja za podešavanje protokola koristi običnu ASCII tekstualnu datoteku koja se može uređivati iz Seyona. Seyon će od korisnika zatražiti imena datoteka samo ako odabrani protokol zahtijeva imena datoteka ili ako je prijenosna radnja slanje datoteka, za što Seyon prihvaća i zamjenske znakove. Za različite se pretince prijenosa može navesti više *download* direktorija.

Seyon otkriva dolazeće Zmodem potpise i automatski aktivira korisnički definiran Zmodem protokol za primanje dolazećih datoteka. Zmodem prijenosi su tako posve automatski i ne zahtijevaju korisničko posredovanje.

**Mogućnosti prevođenja** Seyon može izvršiti korisna prevođenja korisničkog unosa. Na primjer, može <Backspace> prevesti u <Delete>, oznaku novog reda u oznaku vraćanja

na početak reda, prevođenje *meta* tipaka - to jest, vašu *meta* tipku <Esc> možete zamijeniti s tipkom <Alt>. Ovaj posljednji način oponaša *meta* tipke na domaćinima koji ne podržavaju čiste 8-bitne veze i omogućuju uporabu *meta* tipaka u programima poput *emacs*a.

**Druge mogućnosti Seyona** Seyon vam omogućuje interaktivno podešavanje programskih parametara, ugrađeni sustav pomoći, softversku (XONN/XOFF) i hardversku (RTS/CTS) kontrolu protoka, hvatanje sesije u datoteku i privremeno pokretanje lokalne ljuske u prozoru emulacije terminala.

Seyon je zamišljen da bude jednostavan, a ipak široko podesiv. Gotovo se svaka značajka Seyona pomoću ugrađenih resursa može podesiti prema korisničkom ukusu.

#### xgrab

*xgrab* je interaktivno sučelje za *xgrabsc*, hvatač slika u X Windowsima. *xgrab* je napisao Bruce Schuchardt (**bruceŽslc.com**) i mnogi drugi, koji je zadržao labava autorska prava na program. *xgrab* vam omogućuje hvatanje proizvoljnih pravokutnih slika iz *xserver*a i njihovo upisivanje u datoteke ili naredbe (poput *lpr*) u mnoštvu različitih formata.

Na *man* stranicama za *xgrabsc* ćete naći opise raznih opcija za *xgrab*. Nakon odabira opcija iz različitih kategorija kliknite na OK i *xgrab* će pokrenuti *xgrabsc* i omogućiti vam hvatanje slike na zaslonu. Nakon klika na OK, *xgrab*ov prozor nestaje i *xgrabsc* preuzima nadzor sve do završetka postupka hvatanja slike. Nakon toga se ponovno pojavljuje prozor *xgrab*.

*xgrab* reagira na uobičajene aplikacijske opcije, poput -\_*display*. Cjeloviti ćete popis naći na *man* stranici za X Windowse. Unaprijed zadane postavke za *xgrab* možete premostiti u vašoj datoteci .Xdefaults. O uputama pogledajte članak "Primjeri".

**Resursi** Datoteka resursa za *xgrab*, Xgrab.ad sadrži cjelovitu specifikaciju resursa za sve kontrole (*widgets*) korištene u prozoru *xgrab*. *Widgets* su specifikacije resursa za stavke poput gumba i izbornika. Globalni su resursi, poput unaprijed zadanog pisma i boje, stavljeni na dno datoteke.

**Primjeri** Izlazna opcija *ToCommand* se može koristiti za ucjevljivanje *xgrabsc* izlaza do drugih programa. Najčešće su naredbe *lpr* za PostScript izlaz i *xwud* za izlaz X Windows Dump. Programe koji ne prihvaćaju ucjevljeni ulaz ne treba koristiti s *ToCommand*.

**SAVJET** Možete biti i maštoviti te izlaz ucijeviti preko više naredbi, poput *tee screen.dmp xwud*, za pohranu uhvaćene slike i dobivanje prozora s probnim prikazom (*preview*).

Unaprijed zadane postavke za *xgrab* se mogu napraviti u vašoj datoteci .Xdefaults. Za verziju *xgrab*a iz Athena toolkita stanje gumba-prekidača može preko njihovog atributa .*state* biti podešeno kao uključeno ili isključeno, a nizovi znakova tekstualnih polja mogu biti postavljeni putem njihovog atributa \**string*. Za verziju iz Motif toolkita, čiji su izborni gumbi u obliku dijamanta (romba), gumbi prekidači mogu biti uključeni ili isključeni preko njihovog atributa *.set*, a stringovi tekstualnih polja mogu biti postavljeni putem njihovog atributa \**value*. Na primjer, za podešavanje unaprijed zadane veličine papira za PostScript izlaz u .Xdefaults stavite ove retke (za njihovo učitavanje u poslužitelj koristite *xrdb*):

```
XGrab*.pageWidthText*string: 8.5
XGrab*.pageHeightText*string: 11.0
ili
XGrab*.pageWidthText*value: 8.5
XGrab*.pageHeightText*value: 10.0
```

Za podešavanje unaprijed zadane vrste izlaza u XWD u .Xdefaults stavite ove retke:

XGrab\*.ps.state: 0 XGrab\*.xwd.state: 1

## xlock

Patrick J. Naughton (**naughton@eng.sun.com**) je napisao *xlock* i prepustio ga svijetu. Program *xlock* zaključava lokalni X Windows zaslon sve dok korisnik ne unese lozinku preko tipkovnice. Dok se *xlock* izvršava, odbijaju se sve nove veze s poslužiteljem, zaštitnik zaslona je onemogućen, kursor miša je isključen, zaslon je ispražnjen i na njega je stavljen promjenjivi uzorak. Kada netko pritisne tipku na tipkovnici ili tipku miša, tražit će se unos lozinke korisnika koji je pokrenuo *xlock*.

Ako se unese ispravna lozinka, zaslon će se otključati i X poslužitelj će biti vraćen. Kada utipkavate lozinku, <Ctrl-Shift-u> i <Ctrl-Shift-h> su aktivne kao naredbe zaustavljanja (*kill*) i brisanja. Za vraćanje na zaključani zaslon kliknite na malu ikoniziranu inačicu promjenjivog uzorka.

# Zabavljanje s DOOM-om za Linux

Najbolje se uvijek čuva za kraj. Zašto izvršavati XFree86 pod Linuxom? Zato jer je ID Software, Inc. na raspolaganje stavio njihovu inačicu *shareware* igre DOOM. Premda su je potisnule druge pucačke igre koje se igraju u prvom licu (npr., ID Softwareov Quake), DOOM je očaravajuća krvoločna avantura koja se još uvijek igra u cijelom svijetu. Uporabom realistične 3-D grafike vi u ulozi svemirskog marinca ulazite u bezbožnu, terorom ispunjenu svemirsku koloniju smještenu na jednom od Marsovih mjeseca. Trebate se probijati kroz laboratorije i različite lokacije, tražeći vaše izgubljene drugove. Sve što ćete umjesto toga pronaći su strašna čudovišta i druge svemirske marince koji su se okrenuli protiv vas.

X Windows inačica dostavljena na pratećem Slackware CD-ROM-u u direktoriju /contrib je cjelovita *shareware* inačica. (Red Hat distribucija tijekom instalacije automatski instalira i ovu igru). Premda ova inačica radi i na 386 računalima, ona je napravljena za rad na vrhunskim 486 sustavima. Ako DOOM pokrenete na 386-tici s malom količinom fizičkog RAM-a, budite spremni na razočaranje: igra će biti prespora da bi se u njoj moglo uživati. Za igranje DOOMa pod Linuxom trebate mnogo konjskih snaga.

# Instaliranje DOOM-a

DOOM se automatski instalira s Red Hat distribucijom i može se pokrenuti odabirom stavke xdoom u izborniku Start, Programs, Games.
Pod Slackwareom je DOOM pohranjen u slijedu arhiviranih datoteka u direktoriju ///slackware/y2. Ako ste pri instalaciji Linuxa odabrali i instalaciju paketa igara, DOOM bi već trebao biti instaliran. Ako niste, sada ga možete instalirati uporabom programa *pkgtool* ili možete izvršiti sljedeće korake:

- 1. Arhivirane datoteke iz direktorija ///slackware/y2 kopirajte u područje na vašem tvrdom disku.
- Direktorij promijenite u osnovni direktorij koji želite koristiti. Arhive će datoteke raspakirati u direktorij usr/games/doom pa je možda najbolje sve datoteke u doom direktoriju na pratećem Slackware CD-ROM-u kopirati u /usr sljedećim naredbama:

```
cd /usr
cp /cdrom/contrib/linuxdoom/* .
```

3. Sve datoteke u direktoriju raspakirajte ovom naredbom:

```
gzip -d imedatoteke
```

gdje je *imedatoteke* ime svake datoteke u direktoriju. Ova naredba stvara dvije *tar* datoteke.

4. Obje arhive raspakirajte naredbom *tar* radi stvaranja neophodnih direktorija i datoteka:

```
tar -xfv archive-file
```

Vidi "Uporaba naredbe tar", str. 229.

#### Pokretanje DOOM-a

Prije pokretanja DOOM-a trebate pokrenuti X Windowse, stoga unesite *startx*. Kada X Windowsi prorade, možete pokrenuti *xterm* sesiju ili upotrijebiti kombinaciju tipaka <Ctrl-Alt-Fx> za pristup jednom od virtualnih znakovnih terminala i zatim unesite *linuxxdoom*. Ako to ne uspije, Linux ne može pronaći program DOOM - to jest on nije smješten u vašoj stazi. Ako se to dogodi, samo trenutni direktorij promijenite u direktorij u koji ste instalirali DOOM. Zatim ponovno unesite DOOM naredbu.

Ako ste DOOM pokrenuli iz virtualnog terminala, trebate se vratiti u X Windows sesiju pritisnuvši <Ctrl-Alt-F22>. Ako ste DOOM pokrenuli iz *xterm* sesije, za nekoliko biste sekundi trebali ugledati uvodni zaslon DOOM-a.

Za vrijeme dok se DOOM učitava pratite slijed poruka. Jedna bi mogla biti i obavijest da DOOM i Linux ne mogu pokrenuti zvukovni sustav, stoga ćete DOOM možda trebati igrati bez zvuka. U sadašnjoj prilagodbi DOOM-a zvuk još uvijek nije u cijelosti podržan - takav je život sustava u razvoju poput Linuxa. Upute o igranju DOOM-a potražite u datoteci README.Linux. BILJEŠKA X Windows poslužitelji (upamtite, poslužitelj se izvršava na lokalnom sustavu) su raspoloživi za većinu ne-UNIX sustava. Jedan je od najboljih X poslužitelja MicroImagesova aplikacija MI/X, raspoloživa i za Macintosh i za Windows operativni sustav. Više informacija i besplatni X poslužitelj potražite na http://www.microimages.com/.

# Odavde...

Na mreži ima mnoštvo programa na raspolaganju za Linux. X Windowse možete koristiti za lakši višezadaćni rad s raznim Linux programima uporabom *xterm*a umjesto virtualnih terminala raspoloživih sa znakovnih zaslona. O raznim informacijama o X Windowsima općenito možete provjeriti novinske grupe **comp.windows.x.apps** i **comp.windows.x.intrinsics**. Sljedeća poglavlja također pružaju više informacija:

- 3. poglavlje, "Instaliranje Red Hata" opisuje kako instalirati Red Hat distribuciju Linuxa, koja uključuje instalaciju X-a pod Red Hatom.
- 21. poglavlje, "Instaliranje sustava X Windows" objašnjava kako instalirati XFree86 inačicu X-a za Linux, koju trebate staviti prije instaliranja X Windowsa.
- Ako želite saznati kako na Internet pristupiti s Linuxom, pogledajte 30. poglavlje, "Pristupanje Mreži s naredbama *telnet*, *ftp* i *r*-".
- O pretraživanju Mreže provjerite 31. poglavlje, "Kretanje po Internetu pomoću World Wide Weba".

# poglavlje23

# Razumijevanje protokola TCP/IP

(Steve Burnett)

#### U ovom poglavlju

Povijest TCP/IP-a Internet Terminologija Model Open Systems Interconnection Stog TCP/IP protokola IP adrese Podmreže i podmrežne maske Usmjeravanje (*routing*) Postavljanje Internet mreže Garnitura je široko korištenih protokola poznatih kao Transmission Control Protocol/Internet Protocol (TCP/IP) postala izuzetno važna budući da o njoj ovisi komunikacija mreža svih veličina, uključujući i Internet.

TCP/IP je od svog početnog razvoja kao projekta pod pokroviteljstvom vlade narastao na široko rasprostranjenu uporabu, povezujući mreže svih veličina. Prihvaćen zbog svoje sposobnosti omogućavanja komunikacije među nesličnim strojevima, može ga se naći na doslovce svim radnim stanicama, miniračunalima i *mainframe* računalima. Ovo poglavlje opisuje porijeklo i jezik TCP/IP-a, njegove konvencije adresiranja i imenovanja i osnovne zamisli koje su poslužile kao temelji za stvaranje Interneta.

#### **Povijest TCP/IP-a**

U kasnim 1960-ima je u američkom ministarstvu obrane (U.S. Department of Defense - DOD) uočen narastajući problem s elektroničkom komunikacijom unutar ministarstva. Razmjena sve veće količine elektroničkih informacija među DOD osobljem, istraživačkim laboratorijima, sveučilištima i kooperantima je naišla na ozbiljnu zapreku. Zasebne su jedinice imale računalne sustave od različitih proizvođača računala, koji su koristili različite operativne sustave i različite mrežne topologije i protokole. Kako bi se te informacije mogle dijeliti?

Za rješavanje je problema s različitom mrežnom opremom i topologijom bila zadužena Advanced Research Project Agency (ARPA). ARPA je osnovala savez sa sveučilištima i proizvođačima računala radi razvijanja komunikacijskih standarda. Taj je savez točno odredio i izgradio mređu s četiri čvorišta koja je osnova današnjeg Interneta. Tijekom 1970-tih ta je mreža promijenjena prema nacrtu novog, osnovnog protokola (*core protocol*) koji je postao osnova za TCP/IP.

Spominjanje TCP/IP-a zahtijeva sažeti uvod u Internet, golemu mrežu mreža koja omogućuje međusobnu komunikaciju računala iz cijelog svijeta. Ona raste toliko nevjerojatnom brzinom da bi svaka procjena broja računala i korisnika na Internetu u trenutku tiskanja ove knjige već bila zastarjela! Čvorišta uključuju sveučilišta, velike korporacije, istraživačke laboratorije iz SAD-a i drugih zemalja, škole, velike i male tvrtke i računala u vlasništvu pojedinaca. Eksplozija rasta World Wide Weba u posljednjih nekoliko godina potakla je širenje Interneta. Pored toga, Internet je i riznica za milione *shareware* programa, novosti o svim temama, javnih okupljališta i razmjenu informacija, elektroničku poštu i drugo. Uporabom Telnet protokola moguća je daljinska prijava na bilo koji računalni sustav na mreži. Zahvaljujući velikom broju međupovezanih sustava moguće je dijeliti goleme računalne resurse, omogućivši izvršavanje velikih programa na udaljenim sustavima. Masovno podijeljeni procesni projekti popud dešifriranja Data Encryption Standarda u 1997. su mogući samo zahvaljujući organizaciji Interneta "sve je povezano sa svim ostalim".

# Internet terminologija

Garnitura je Internet protokola sastavljena od mnogih srodnih protokola temeljenih na osnovama stvorenim s TCP-om i IP-om. Radi pojašnjenja odnosa tih komponenata tablica 23.1 pruža neke definicije i napomene.

Tablica 23.1	Mrežni pojmovi
Pojam	Definicija
datagram	Označava jedinicu razmijenjene informacije. Isto značenje imaju i pojmovi data packet i network message.
DNS	Domain Name Service - usluga koju pruža jedno ili više računala na mreži kao pomoć u pronalaženju staze do željenog čvora. Time se svaki sustav na mreži oslobađa potrebe čuvanja popisa svih sustava s kojima želi razgovarati. Koriste je prolazi za poštu.
GOSIP	Government Open System Interconnection Profile - zbirka OSI protokola koji se koriste u računalnim mrežama i projektima vlade SAD-a.
Internet	Računalna mreža temeljena na TCP/IP i srodnim protokolima. Javna mreža koja povezuje druge mreže u sveučilištima, tvrtkama, vladinim uredima i istraživačkim laboratorijima.
FTAM	File Transfer, Access, and Management - protokol za prijenos i upravljanje datotekama, kako je određeno u OSI.
FTP	File Transfer Protocol, koji omogućuje prijenos datoteka između sustava.
IP	Internet Protocol - protokol odgovoran za prijenos datagrama preko Interneta.
NFS	Network File System, mrežni virtualni diskovni sustav koji računalu klijentu omogućuje postavljanje (mount) udaljenih datotečnih sustava i direktorija. Izvorno razvijen u Sun Microsystemsu.
NIC	Network Information Center, odgovoran za administriranje Interneta i TCP/IP adresa, kao i mrežnih imena.
node (čvor)	Računalo na mreži.
OSI	Open System Interconnection, model ISO standarda za definiranje podatkovne komunikacije.
RFC	Request For Comments, dokumentacija koju održava NIC, a koja se odnosi na Internet protokole, adresiranje, usmjeravanje, podešavanje i druge srodne Internet teme.
RIP	Routing Information Protocol, koji se koristi za razmjenu informacija između usmjerivača (routera).
RMON	Remote monitor - udaljeni mrežni monitor koji omogućuje zbirku informacija o mrežnom prometu.
RPC	Remote Procedure Call, koja omogućuje izvršavanje procedura na poslužitelju.
SMTP	Simple Mail Transfer Protocol, koji se koristi za prijenos elektroničke pošte između sustava. <i>nastavlja se</i>

Tablica 23.1	Nastavak
Pojam	Definicija
SNMP	Simple Network Management Protocol - protokol koji se koristi za upravljanje udaljenim mrežnim uređajima i za sakupljanje informacija s udaljenih uređaja u vezi s podešenjima, greškama i alarmima.
ТСР	Transmission Control Protocol – protokol između dviju aplikacija odgovornih za pouzdan prijenos podataka.
Telnet	Protokol za uspostavljanje veza udaljenih terminala.
UDP	User Datagram Protocol - protokol za prijenos podataka između agenata, bez potrebe za uspostavom izravne veze.
VT	Virtual terminal - metoda uporabe Telneta za prijavu na udaljene sustave preko mreže.

## **Model Open System Interconnection**

Danas se koriste mnoge različite vrste računala, s različitim operativnim sustavima, procesorima, mrežnim sučeljima i mnogim drugim značajkama. Zbog tih je razlika komunikacija između raznovrsnih računalnih sustava postala značajan problem. U 1997. je International Organization for Standardization (ISO) stvorio pododbor za razvoj standarda podatkovne komunikacije radi promicanja komunikacije među raznovrsnim sustavima. Tako je stvoren model Open Systems Interconnection (OSI).

Model OSI ne određuje nikakve komunikacijske standarde niti protokole, već pruža smjernice koje komunikacijske zadaće trebaju slijediti.

#### BILJEŠKA

Važno je razumijeti da je model OSI samo model - kostur - koji određuje koja djelovanja treba izvršiti. On ne ulazi u pojedinosti načina izvršenja tih funkcija. ISO, međutim, potvrđuje određene protokole koji odgovaraju OSI standardima za dijelove modela OSI. Na primjer, ISO je protokol CCITT X.25 prihvatio kao primjenu koja pruža većinu usluga Mrežnog sloja modela OSI.

Radi pojednostavljenja, ISO pododbori su problemu pristupili na način "podijeli pa vladaj". Razdijelivši složeni komunikacijski proces u manje podzadaće, problem je postao lakši za baratanje i svaka se podzadaća mogla optimizirati. Model OSI je podijeljen u sedam slojeva:

- Aplikacijski
- Prezentacijski
- Sesijski
- Prijenosni

- Mrežni
- Podatkovni
- Fizički

SAVJET Na engleskom su jeziku ti slojevi: Application, Presentation, Session, Transport, Network, Data Link, Physical i njhov se redoslijed može lakše upamtiti ako se iz početnih slova napravi rečenica: All People Seem To Need Data Processing.

Svakom je sloju dodijeljen određeni skup djelovanja. Svaki sloj koristi usluge sloja ispod njega i pruža usluge sloju iznad njega. Na primjer, mrežni sloj koristi usluge podatkovnog sloja, a prijenosnom pruža svoje mrežne usluge. Tablica 23.2 opisuje usluge koje pruža svaki sloj.

BILJEŠKA Zamisao o sloju koji koristi usluge i pruža usluge susjednom sloju je vrlo jednostavna. Razmotrite djelovanje neke tvrtke: tajnica svoje tajničke usluge pruža predsjedniku (sljedeći sloj prema gore) za pisanje poruke. Tajnica koristi usluge kurira (sljedeći sloj prema dolje) za isporuku te poruke. Razdvajanjem tih usluga tajnica (aplikacija) ne mora znati na koji se način poruka stvarno donosi primatelju. Tajnica samo treba kurira (mrežu) zatražiti da je isporuči. Baš kao što mnoge tajnice na taj način mogu poruke slati uporabom uobičajene kurirske službe, slojevita mreža može pakete slati predavši ih mrežnom sloju za isporuku.

Sloj	Opis
Fizički (1. sloj)	Ovaj sloj pruža fizičku vezu između računalnog sustava i mreže. On određuje priključnicu i zadaće nožica, naponske razine itd.
Podatkovni (2. sloj)	Ovaj sloj "pakira" i "raspakirava" podatke za prijenos. On podatke oblikuje u podatkovne pakete, nazvane frames (okviri). Frame predstavlja točnu strukturu podataka koji se fizički prenose preko žice ili drugog sredstva.
Mrežni (3. sloj)	Ovaj sloj pruža usmjeravanje (routing) podataka kroz mrežu.
Prijenosni (4. sloj)	Ovaj sloj pruža slanje paketa u sekvencama (u slijedu) i potvrdu prijenosa.
Sesijski (5. sloj)	Ovaj sloj uspostavlja i okončava komunikacijske veze.
Prezentacijski (6. sloj)	Ovaj sloj vrši pretvorbu podataka i osigurava razmjenu podataka u univerzalnom formatu.
Aplikacijski (7. sloj)	Ovaj sloj pruža sučelje za aplikaciju koju korisnik izvršava - prolaz između korisničkih aplikacija i mrežnog komunikacijskog procesa.

#### Tablica 23.2 Usluge koje pruža svaki OSI sloj

**BILJEŠKA** Nemojte aplikacijski sloj pobrkati s aplikacijskim programima koje izvršavate na računalu. Upamtite da je *aplikacijski sloj* dio modela OSI koji ne određuje *kako* se ostvaruje veza između korisnika i komunikacijske staze. Aplikacijski je *program* posebni vid primjene tog sučelja. Stvarna aplikacija obično izvršava zadaće zadnja tri sloja, a zadaće prva četiri sloja ostavlja mrežnom operativnom sustavu.

Svaki sloj komunicira sa svojim parom u drugim računalima. Na primjer, 3. sloj u jednom sustavu komunicira s 3. slojem u drugom računalnom sustavu.

Kada se informacija prosljeđuje s jednog sloja na sljedeći, podacima se dodaje zaglavlje koje ukazuje otkuda informacija dolazi i kamo odlazi. Zaglavlje+podatkovni blok informacija s jednog sloja postaju podaci za sljedeći. Na primjer, kada 4. sloj podatke proslijedi 3. sloju, on dodaje svoje vlastito zaglavlje. Kada 3. sloj te informacije prosljeđuje 2. sloju, on zaglavlje+podatke s 4. sloja uzima u obzir kao podatke, dodaje im svoje vlastito zaglavlje i zatim tu kombinaciju dalje prosljeđuje.

U svakom se sloju jedinicama informacija daju drukčija imena (vidi tablicu 23.3). Prema tome, poznavajući pojmove koji se koriste pri upućivanju na podatke, vi točno znadete o kojem se sloju modela govori.

Tablica 23.3	Pojmovi koje OSI slojevi koriste pri upućivanju na informacijske jedinice
OSI sloj	Naziv informacijske jedinice
Aplikacijski	Poruka
Prijenosni	Segment
Mrežni	Datagram
Podatkovni	Okvir
Fizički	Bit

Prije pojave modela OSI Ministarstvo obrane SAD-a je definiralo svoj vlastiti mrežni model, poznat kao *DOD model*. Model DOD je vrlo srodan protokolima TCP/IP, kako je objašnjeno u sljedećem članku.

# Stog TCP/IP protokola

Stog TCP/IP protokola predstavlja mrežnu arhitekturu sličnu mrežnom modelu ISO OSI. Slika 23.1 prikazuje mapiranje TCP/IP slojeva na stog ISO protokola.

TCP/IP ne radi velike razlike između gornjih slojeva stoga protokola kao što to čini OSI. Gornja su tri OSI sloja približno jednakovrijedni procesnim protokolima za Internet. Neki od primjera takvih procesnih protokola su Telnet, FTP, SMTP, NFS, SNMP i DNS.



Sloj prijenosnog modela je odgovoran za pouzdanu isporuku podataka. Na stogu Internet protokola to odgovara *host-to-host (domaćin-domaćin)* protokolima. Primjeri takvih protokola su TCP i UDP. TCP se koristi za prevođenje poruka promjenjive dužine iz protokola s gornjih slojeva i pruža neophodnu potvrdu primitka i kontrolu toka između udaljenih sustava.

UDP je sličan TCP-u, osim što nije usmjeren na vezu i ne potvrđuje primitak podataka. UDP samo prima poruke i prosljeđuje ih duž protokola gornjih slojeva. Budući da UDP nema nikakve nadogradnje srodne TCP-u, on pruža mnogo učinkovitije sučelje za djelovanja poput udaljenih diskovnih servisa.

Internet Protocol (IP) je odgovoran za komunikacije bez izravne veze među sustavima. On se mapira na OSI model kao dio mrežnog sloja koji je odgovoran za kretanje informacija po mreži. Ta se komunikacija ostvaruje pregledavanjem adrese mrežnog sloja, čime se utvrđuju sustavi i staza za slanje poruke.

IP pruža iste mogućnosti kao i mrežni sloj i pomaže u prosljeđivanju poruka među sustavima, ali ne jamči isporuku tih poruka. IP može poruke razdvojiti na djeliće i zatim ih na odredištu ponovno sastaviti. Čak i ako djelići stignu u pobrkanom redu, IP će pakete na odredištu ponovno sastaviti u pravilnom slijedu.

### **IP** adrese

Internet Protocol zahtijeva da svaki uređaj na mreži ima dodijeljenu adresu. Ta je adresa, poznata kao *IP adresa*, organizirana kao slijed od četiri okteta. Svaki od tih okteta definira jedinstvenu adresu, s dijelom adrese koji predstavlja mrežu (neobavezno i podmrežu) i s drugim dijelom koji predstavlja određeni čvor na mreži.

Nekoliko adresa ima posebno značenje na Internetu:

- Adresa koja počinje s nulom upućuje na lokalni čvor unutar njegove trenutne mreže. Na primjer, 0.0.0.23 upućuje na radnu stanicu 23 na trenutnoj mreži. Adresa 0.0.0.0 upućuje na trenutnu radnu stanicu.
- Povratna (*loopback*) adresa, 127, važna je u pronalaženju grešaka i mrežnoj dijagnostici. Mrežna adresa 127.0.0.0 je lokalna povratna adresa unutar radne stanice.
- Adresa ALL (za sve) je predstavljena uključivanjem svih bitova, što po oktetu daje vrijednost 255. Tako 192.18.255.255 šalje poruku svim čvorovima na mreži 192.18. Slično tome, 255.255.255.255 šalje poruku svakom čvoru na Internetu. Te su adrese važne za *multicast* poruke koje se istovremeno šalju na više adresa, kao i za objave usluga.

#### UPOZORENJE

Važno je da pri dodjeljivanju brojeva čvora vašim radnim stanicama ne koristite 0, 127 ili 255, budući da su to rezervirani brojevi i imaju posebna značenja.

#### **Klase IP adresa**

IP adrese su doznačene u rasponima koje nazivamo *klasama*, ovisno o aplikaciji i veličini organizacije. Tri su najčešće klase A, B i C. Te tri klase predstavljaju broj lokalno doznačivih bitova raspoloživih za lokalnu mrežu. Tablica 23.4 prikazuje odnose između različitih klasa adresa, raspoloživi broj čvorova i početne postavke adrese.

Tablica 23.4 Klase IP adresa			
Klasa	Raspoloživi čvorovi	Početni bitovi	Polazna adresa
А	2 <sup>24</sup> =167.772	0xxx	0-127
В	216=65.536	10xx	128-191
С	2 <sup>8</sup> =256	110x	192-223
D		1110	224-239
E		1111	240-255

Adrese klase, A se koriste za vrlo velike mreže ili skupine povezanih mreža. Adrese klase B se koriste za velike mreže koje imaju više od 256 čvorova (ali manje od 65.536 čvorova). Adrese klase C koristi većina organizacija. Za organizaciju je bolje da dobije nekoliko adresa klase C budući da je broj adrsea klase B ograničen. Klasa D je rezervirana za *multicast* poruke na mreži, a klasa E je rezervirana za isprobavanje i razvoj. **Dobivanje IP adresa** Administraciju Internet adresa trenutno vrši Network Information Center (NIC):

Network Solutions

ATTN: InterNIC Registration Services

505 Huntmar Park Drive

Herndon, VA 22070

(703) 724-4777



#### NA WEBU

InterNIC možete pronaći i na Web adresi

#### http://www.internic.net

Kada računalo ili mrežu spojite na Internet, u većini će vam slučajeva registraciju vaše mrežne IP adrese umjesto vas moći organizirati vaš dobavljač Internet usluga.

**Dobivanje RFC-ova** Pored dodjeljivanja adrese NIC može pružiti i druge vrijedne informacije. NIC je riznica za svu tehničku dokumentaciju u vezi Interneta. On ima zbirku dokumenata koji opisuju sve pridružene protokole, metodologije usmjeravanja, smjernice za upravljanje mrežom i metode za uporabu različitih mrežnih tehnologija.

Kako je u tablici 23.1 spomenuto, RFC je kratica za Request For Comments. RFC-ove s Interneta možete dobiti uporabom FTP protokola za spajanje na nekoliko različitih riznica. RFC nizovi su raspoloživi s raznih lokacija na Internetu preko anonimnog FTP-a, poput **ftp.internic.net** u direktoriju /rfc, a može im se pristupiti i putem Telneta na **rs.internic.net**.

U tablici 23.5 je popis RFC-ova primjerenih za uspostavljanje mreže. Neki od tih dokumenata prilično ulaze u pojedinosti o djelovanju različitih protokola, o podložnim specifikacijama i teoriji. Drugi su više općenitiji i pružaju ključne informacije koje mrežnom upravitelju mogu biti od koristi. Upravitelj Internet mreže bi kao minimum trebao znati gdje su ti dokumenti smješteni i kako ih dobaviti. Oni pružaju informacije koje mogu pomoći u osmišljavanju i nadogradnji mreže u organizaciji.

Tablica 23.5	Zanimljivi RFC-ovi
Ime RFC-a	Naslov
RFC791.txt	Internet Protocol DARPA Internet Program Protocol Specification
RFC792.txt	Internet Control Message Protocol
RFC793.txt	Transmission Control Protocol DARPA Internet Program Protocol Specification
RFC950.txt	Internet Standard Subnetting Procedure
RFC1058.txt	Routing Information Protocol nastavlja se

Tablica 23.5	Nastavak
Ime RFC-a	Naslov
RFC1178.txt	Choosing, a Name for Your Computer
RFC1180.txt	A TCP/IP Tutorial
RFC1208.txt	A Glossary of Networking Terms
RFC1219.txt	On the Assignment of Subnet Numbers
RFC1234.txt	Tunneling IPX Traffic Through IP Networks

#### Davanje imena na mreži

Imenovanje mrežnih čvorova zahtijeva nešto planiranja. Pri odabiranju imena imajte na umu upravljanje mrežom i prihvatljivost korisnicima. Mnoge organizacije imaju standarde za imenovanje mreža. Ako i vaša organizacija ima takve standarde, najbolje je slijediti ih radi izbjegavanja zbrke. Ako ih nema, tu ima mnogo prostora za vašu domišljatost. Imena računala i mreža mogu biti i jednostavna poput imenovanja radnih stanica po njihovim korisnicima, primjerice Diana, Bruno, Lara.

Ako imate mnogo sličnih računala, njihovo bi pobrojavanje (na primjer, PC1, PC2... PC128) moglo biti primjereno. Imenovanje mora biti izvršeno tako da računalni sustavi dobiju jedinstvena imena. Ako računalu date ime *računalousjevernomuredu*, nemojte se čuditi ako će korisnici prigovarati. Na kraju, čak i administrator sustava s vremena na vrijeme mora utipkavati imena računala. Također, izbjegavajte imena poput *oiiomfw932kk*. Premda bi takvo ime mrežne uljeze moglo spriječiti u spajanju na vaše računalo, ono bi i vas moglo spriječiti u spajanju na vašu radnu stanicu.

Uvijek je dobro izabrati imena koja je lako razlikovati i koja slijede neku temu. Ona će pomoći pri usklađivanju budućih proširenja i korisnicima pružaju osjećaj povezanosti s njihovim strojeva. Nakon svega, mnogo je lakše imati dobar odnos sa strojem koji je dobio neko ugodno ime nego sa strojem OF1284.

Pri odabiru sheme imenovanja dobro je upamtiti sljedeće točke:

- Odaberite jednostavna i kratka imena najviše šest do osam znakova. Premda Internet Protocol dopušta imena duga do 255 znakova, to trebate izbjegavati budući da neki sustavi ne mogu baratati s dugačkim imenima. (Svaka oznaka može imati do 63 znaka. Oznaka je svaki točkom odvojeni dio punog imena domene.)
- Razmotrite uporabu tema poput zvijezda, cvijeća ili boja, osim ako se na vašoj lokaciji zahtijevaju drukčiji standardi imenovanja.
- Ime ne započinjite s brojevima.
- U imenu ne koristite posebne znakove.
- Pripazite da ne ponavljate ista imena.
- Ustrajno se držite vašeg pravila imenovanja.

Držeći se ovih smjernica možete uspostaviti uspješnu metodologiju imenovanja.

Internet imena predstavljaju organizacije i namjene sustava unutar mreže. Sljedeći su primjeri imena koja možete koristiti:

palcic.razvoj.mojatvrtka.com

nic.ddn.mil

Sljedeći su primjeri imena koja je teško koristiti ili upamtiti:

ovojemojaradnastanica.dugizavojitiodjel.dugoimetvrtke.com

34556nx.m3422.mojatvrtka.com

Posljednji bi primjer mogao biti šifrirana informacija o radnoj stanici u sobi 345 na mreži 56 s izvršnim mrežnim funkcijama, međutim, ovakva se vrsta sheme imenovanja obično smatra slabom praksom jer može dovesti do pometnje i pogrešno usmjerenih poruka.

Internet ime poput Maja@PC28.Programiranje.mojatvrtka.com vam omogućuje da korisnika uputite na određeni čvor.

#### NIC-ovo stablo imenovanja

NIC održava stablo imenovanja na mreži. Ovo se stablo koristi za grupiranje sličnih organizacija pod sličnim granama stabla. Slika 23.2 prikazuje stablo imenovanja. Glavne su organizacije grupirane pod sličnim granama. Ovo je izvor za Internet oznake, poput com, edu i gov, koje vidimo u Internet imenima.





Tablica 23.6 prikazuje neke od čestih imena listova i definicija za NIC stablo. Ispod stabla su i mnogi drugi listovi, međutim, ovi su najčešći.

Tablica 23.6	Česta NIC imena	
Ime	Vrsta organizacija	
edu	Obrazovne ustanove (poput sveučilišta i visokih škola)	
com	Komercijalne (većina korporacija)	
gov	Civilna tijela vlade SAD-a (Bijela kuća, ministarstva)	

nastavlja se

Tablica 23.6	Nastavak
Ime	Vrsta organizacija
mil	Vojne (vojni korisnici i njihovi dobavljači)
net	Internet administracija i mrežno upravljanje
org	Druge vrste organizacija (obično neprofitne)

# Podmreže i podmrežne maske

Subnetting (dijeljenje mreže) je razdjeljivanje velike logičke mreže u manje fizičke mreže. Među razlozima dijeljenja mreže mogu biti električka ograničenja mrežne tehnologije, želja za razdjeljivanjem radi jednostavnosti stavljajući zasebnu mrežu na svaki kat zgrade (ili u svaki odjel i za svaku aplikaciju) i potreba za povezivanjem udaljenih lokacija s vrlo brzom linijom.

Tako nastale mreže su manji dijelovi cjeline i s njima je lakše upravljati. Manje podmreže međusobno komuniciraju preko prolaza i usmjerivača. Organizacija može imati i nekoliko podmreža koje su fizički na istoj mreži radi logičkog razdjeljivanja mrežnih funkcija na radne skupine.

Pojedinačne su podmreže dijelovi cjeline. Pretpostavimo da je mreža klase B podjeljena na 64 zasebne podmreže. Za postizanje se ovog dijeljenja mreže IP adresa promatra u dva dijela: mreža i domaćin (vidi sliku 23.3). Mrežni dio (NETWORK) postaje dodijeljena IP adresa i informacijski bitovi podmreže. Ti su bitovi, u suštini, uklonjeni iz domaćinskog dijela (HOSTS) adrese. Za mrežu klase B je dodijeljeno 16 bitova. Podmrežni dio dodaje 6 bitova, za ukupno 22 bita radi razlikovanja podmreže. Rezultat je tog dijeljenja 64 mreže s 1.024 čvorova u svakoj. Mrežni dio može biti veći ili manji, ovisno o željenom broju mreža ili broju čvorova po mreži.

Postavljanje je podmrežne maske stvar utvrđivanja gdje mrežna adresa završava i adresa domaćina počinje. Podmrežna maska sve jedinice sadrži u mrežnom polju, a sve nule u polju domaćina.

Pretpostavimo da je mreža klase C sastavljena od sljedećeg:

```
N = network (mreža)
H = host (domaćin)
NNNNNNN.NNNNNNN.NNNNNNN.HHHHHHHH
```

Svako mjesto predstavlja zasebni bit u 32-bitnom adresnom prostoru. Ako će ta mreža klase C biti razdijeljena u četiri mreže klase C, uzorak će nalikovati sljedećem:

NNNNNNN.NNNNNNNNNNNNNNNNNNNNN

Podmrežna će maska izgledati ovako:

```
11111111.1111111.11111111.11000000
```



Ako je ta adresa napisana u decimalnom obliku s brojevima odvojenim točkama, maska podmreže je 255.255.192. Ova se maska koristi za komuniciranje između čvorova na svim podmrežama unutar ove mreže.

Ako se iz polja domaćina uzmu tri bita, može se osnovati osam mreža, a nastala će mrežna maska izgledati ovako:

```
11111111.1111111.1111111.11100000
```

Maska ove podmreže je 255.255.255.224. Svaka će od osam mreža imati 29 čvorova jer je raspoloživo pet adresnih bitova. (Bilo bi ih 32, međutim, sve jedinice, sve nule i 127 nisu legalne adrese).

Ovaj se princip može proširiti na mreže klase B i klase A. Jedina je razlika da su preostala polja 0 (nula).

Razmotrite mrežu klase B. Adresni je prostor podijeljen kako slijedi:

NNNNNNN.NNNNNNN.HHHHHHHH.HHHHHHH

Ako se iz polja domaćina uzmu dva bita i dodaju mrežnom dijelu, koristi se sljedeća podmrežna maska:

11111111.1111111.11000000.0000000

Maska je napisana kao 255.255.192.0.

Bitovi potrebni za podmrežnu masku mogu se uzeti s bilo kojeg položaja bita unutar polja domaćina, međutim, to bi dovelo do složenih podmrežnih maski i izuzeća adresa. To biste trebali izbjegavati, ako je ikako moguće.

# Usmjeravanje (routing)

Usmjeravanje je postupak prijenosa informacija između mreža. Usmjerivač (*router*) radi na mrežnom sloju. Podaci se mogu usmjeravati na nekoliko različih načina. Za Internet mrežu je primijenjen postupak usmjeravanja Routing Information Protocol (RIP).

#### **Routing Information Protocol (RIP)**

RIP je napravljen za uporabu u mrežama male do srednje veličine i temelji se na usmjerivačkim protokolima Xerox Network Systems (XNS). RIP utvrđuje put poruke uporabom pomoću algoritma koji pretpostavlja da je svakoj stazi određen trošak. Taj trošak može opisati mrežnu propusnost, vrstu linije ili poželjnost staze. Protokol zatim utvrđuje stazu najnižeg troška preko koje zatim prenosi poruku. (Informacije o usmjeravanju možete dobiti iz nekoliko RFC-ova.)

#### Na koji način radi protokol usmjeravanja

Za održavanje popisa skokova do susjednih čvorova RIP usmjerivač drži tablicu usmjeravanja u usmjerivaču ili računalnoj memoriji. Ta se tablica svakih 30 sekundi obnavlja s informacijama iz susjednih usmjerivača. Te se informacije koriste za preračunavanje najjeftinije staze između sustava. Svaki usmjerivač na mreži šalje (oglašava) i prima informacije za usmjeravanje.

Protokol usmjeravanja je ograničen razdaljinom na koju poruka može biti usmjerena (poslana). Svaki usmjerivač može poruku usmjeriti samo do troška od 16 jedinica. Ako je poruka poslana na žicu koja košta više od 16, smatra se da je domaćin nedostižan. *Trošak* je postupak dodjeljivanja vrijednosti različitim stazama kroz mrežu i to je način osiguravanja učinkovitog smjera do odredišta kada do odredišta postoji više puteva.

Kada se dogodi prekid na mreži, usmjerivači ponovno trebaju saznati koje staze imaju najniži trošak. To traje neko vrijeme, a do tada se poruke šalju po višem trošku. Kada neki čvor prestane raditi, svi usmjerivači moraju ponovno namjestiti svoje tablice usmjeravanja. Tijekom tog vremena poruke mogu biti izgubljene u mreži. Usmjerivači se nakon nekog vremena ponovno usklade i usmjeravanje se nastavlja.

Na pad sustava nekog usmjerivača također treba obratiti pažnju. U slučaju pada sustava usmjerivača susjedni usmjerivači tijekom 180 sekundi obnavljaju svoje susjedstvo s njim. Ako nakon isteka tog vremena s tog usmjerivača nije primljena usmjerivačka informacija, ta se staza uklanja iz baze podataka lokalnih usmjerivača.

RIP ne upravlja s razdaljinama usmjeravanja, već samo s troškovima. Zbog toga RIP možda neće iskoristiti najkraću fizičku stazu između dvije točke. Radi rješavanja ovog problema vršeni su neki radovi i preinake protokola. Razvijen je i ispitan novi protokol usmjeravanja, Open Shortest Path First (OSPF), koji se već počeo prihvaćati i uvoditi u uporabu.

#### Segmentacija mreže

Internet mreže su iz raznih razloga podijeljene na manje dijelove. Neki se od razloga odnose na podložne mrežne tehnologije, neki se odnose na zemljopisne lokacije. Neki se od najboljih razloga za odvajanje dijelova mreže temelje na gustoći prometa na mreži. Ako se mnogo prometa na mreži odvija između nekolicine čvorova, te je čvorove najbolje odvojiti. S tim odvajanjem opada zauzetost mreže i omogućuje se brži odziv mreže za druge korisnike mreže.

Drugi su razlozi za segmentaciju promjena mrežnih tehnologija ili komuniciranje između različitih mrežnih tehnologija. Na primjer, uredski prostor može koristiti Token Ring,

a kat s trgovinama može raditi na Ethernetu. Svaki ima različitu namjenu. Uredima je Token Ring možda potreban za komuniciranje s AS/400. Kat s trgovinama možda ima Ethernet da bi omogućio komunikaciju kontrolera i računala. Informacije iz kata s trgovinama zatim mogu biti poslane u uredsku mrežu radi praćenja narudžbi. Veza između tehnologija se obično uspostavlja preko usmjerivača. Usmjerivači prosljeđuju samo informacije koje je potrebno razmjenjivati s jedne na drugu mrežu. Te se informacije zatim mogu dijeliti između čvorova na svakoj mreži.

Pretjerana uporaba usmjerivača može postati opterećenje za mrežu, zbog čega bi se učinkovitost čak mogla smanjiti. Uporaba je usmjerivača od male koristi ako svi čvorovi na jednoj mreži moraju doprijeti do svih čvorova na drugoj mreži i obratno. U tom bi slučaju prednosti usmjeravanja bile umanjene zbog dotatnog rada u protokolima usmjeravanja. U takvom bi slučaju most bio bolje rješenje.

Most (bridge) omogućuje dijeljenje svih informacija iz dvije mreže. Pristup je na fizičkom, a ne mrežnom sloju. Na taj se način izbjegava prevođenje adresa i dodatni rad pri usmjeravanju. Most omogućuje prijenos svih informacija, uključujući i poruke koje emitira sustav. Ako dvije mreže rijetko dijele informacije, usmjerivač je bolji izbor; u suprotnom je pravilnije odabrati most.

# Postavljanje Internet mreže

Nacrt i konfiguracija Internet mreže su slični izvedbi bilo koje računalne mreže. Ona obuhvaća mnoge vrste čvorova, uključujući radne stanice, poslužitelje, pisače, *mainframe* računala, usmjerivače, mostove, prolaze (gateway), ispisne poslužitelje i terminale. Internet zahtijeva da svaki uređaj ima jedinstvenu IP adresu. Uređaj može imati više od jedne adrese, ovisno o njegovoj namjeni, ali je za komunikaciju s drugim uređajima potrebna najmanje jedna adresa.

#### Razumijevanje vrsta veza

TCP/IP mreža se može sastojati od nekoliko sustava povezanih u lokalnu mrežu (LAN) ili od stotine sustava s vezama do tisuća sustava na Internetu. Svaka organizacija može stvoriti vrstu mreže koja najbolje odgovara njenim potrebama.

Slika 23.4 prikazuje jednostavnu mrežu koja se sastoji od nekoliko radnih stanica i datotečnog poslužitelja. Svakoj je stanici na mreži doznačena mrežna adresa 194.62.23. Svakom je uređaju dodijeljena adresa zasebnog čvora. Ovakva je mreža uobičajena u većini odjela unutar tvrtke ili čak za mali ured. Na mreži ima dovoljno prostora za spajanje pisača i još radnih stanica. Mreža nije opremljena za spajanje na druge lokalne mreže ili mreže za šira područja.

Na slici 23.5 je prikazana složenija mreža. Ona uključuje tri zasebne mreže međusobno povezane kombinacijom usmjerivača i poslužitelja. Svaka radna stanica ili računalo na svakom segmentu može (ali ne mora) biti izdvojena od uporabe informacija na jednoj od druge dvije mreže. To je karakteristika podmrežne maske i sigurnosti omogućene na poslužiteljima i usmjerivačima.



Informacija se s jedne mreže na jednu od drugih mreža usmjerava na osnovi potrebe. Ovakva je postava uobičajena za većinu velikih korporacijskih mreža. Ona se može temeljiti na ograničenjima fizičke duljine podložne mrežne tehnologije ili opterećenju pojedine mreže. Na jednoj ili više mreža može se odvijati gusti promet koji se mora raspodijeliti na nekoliko mreža.

Usmjerivač 1 između mreža 1 i 2 pribavlja informacije usmjeravanja između dviju mreža. Ako poslužitelj 1 koji spaja mreže 2 i 3 ima omogućeno usmjeravanje, informacija s mreže 3 na mrežu 2 je usmjerena. Također, informacija s mreže 3 na mrežu 2 može biti usmjerena posredstvom poslužitelja 1 i s mreže 2 na mrežu 1 posredstvom usmjerivača 1. Poslužitelj 1, koji spaja mreže 2 i 3, ima dvije IP adrese: jednu IP adresu na mreži 2, a drugu adresu na mreži 3. Isto važi i za usmjerivač 1, s adresama na mreži 2 i mreži 1.

Razmotrite slučaj u kojem između mreže 3 i mreže 1 ima mnogo prometa s Internet mreže. U tom bi se slučaju isplatilo između mreže 1 i mreže 3 postaviti dodatni usmjerivač. Dodatni usmjerivač može poslužitelj 1 osloboditi dodatnog posla za usmjeravanje i omogućiti prosljeđivanje informacija između mreža i kada poslužitelj 1 ne radi. Dodatni usmjerivač može podići razinu otpornosti na pogreške na mreži. Ta se otpornost na pogreške temelji na činjenici da se informacija iz mreže 3 još uvijek može usmjeriti na mrežu 2, čak i kada poslužitelj 1 ne radi. Staza bi između mreže 3 i mreže 2 bila kroz mrežu 1 i usmjerivač 1. Slika 23.6 prikazuje dodatni usmjerivač 2.



Mrežna otpornost na pogreške unaprjeđuje cjelovitost mreže i u određenim aplikacijama može biti od osobite važnosti. Ako vremenski kritična informacija treba biti dijeljena između dviju mreža, između tih mreža treba osigurati i neku alternativnu stazu. To se može omogućiti uporabom dodatnih usmjerivača. Budući da te staze mogu biti zaobilazne (kroz treću mrežu), potrebno je koristiti konfiguracijski parametar.

Taj parametar obično spominjemo kao *mrežni trošak*. Cijena se skoka može povećati povećanjem vrijednosti koju paket potroši preko mrežne staze. Prednost se unaprijed daje stazi niže cijene. Zamjenska je staza staza visoke cijene. Takvim se uređenjem onemogućuje prečesto usmjeravanje informacije po stazi visoke cijene.

Slika 23.6 prikazuje dodatni usmjerivač dodan između mreža 1 i 3. Poželjna staza za informaciju iz mreže 3 u mrežu 2 je preko poslužitelja 1. Budući da usmjerivač 2 povezuje mrežu 3 i mrežu 1, informacija se može usmjeriti između tih dviju mreža. Također, budući da je usmjerivač 1 između mreže 1 i mreže 2, informacija se usmjerava preko te staze. Informacija iz mreže 3 koja je namijenjena za mrežu 2 može ići preko jedne od ovih dviju staza: ili kroz poslužitelj 1 ili kroz usmjerivač 2 i usmjerivač 1. Ova druga staza nije poželjna, budući da se informacija može usmjeriti izravno preko poslužitelja 1. Prema tome, viši je trošak dodijeljen stazi kroz usmjerivač 2 i usmjerivač 1. Ovakva se analiza staze mora izvršavati u mreži s više segmenata.

#### Odabir mrežne konfiguracije

Fizčiki medij koji bi koristila Internet mreža može biti gotovo svaka mrežna tehnologija u trenutnoj uporabi. Promet Internet mreže nije ograničen na Ethernet, ARCnet ili Token Ring. On može putovati preko asinkronog RS-232, T1 linija i *frame relaya*. Bez obzira koju mrežnu topologiju odabrali za mrežu, potrebno se pridržavati konfiguracijskih, instalacijskih i operacijskih pravila pridruženih toj mrežnoj tehnologiji.

Imajte na umu propusnost koju aplikacija zahtijeva. Mnoge aplikacije zahtijevaju prijenos megabajta podataka i tu propusnost postaje najvažnija. Na propusnosti obično možete uštedjeti sažimanjem datoteka prije njihovog slanja preko mreže.

#### Vidi "Komprimiranje datoteka", str. 332.

U obzir treba uzeti i fizičku lokaciju mreže. Ako su svi čvorovi u istoj zgradi, dovoljan će biti jedan LAN. Međutim, ako su mreže smještene širom grada, mogla bi biti potrebna T1 veza. Ako su čvorovi smješteni na različitim zemljopisnim lokacijama, može se koristiti *frame relay* ili *packet-switched* mreža.

Pri polaganju mreže trebate uzeti u obzir vrstu informacija koje će se preko mreže prenositi, fizičku lokaciju i opterećenje mreže. U utvrđivanju kapaciteta mreže pomoći će vam pregled radnih stanica, poslužitelja i aplikacija.

Ako se na mreži koriste radne stanice bez diskova, svaki je čvor na mreži opterećen većim prometom zbog toga što udaljene radne stancie bez diskova trebaju cjelokupni sustavski kôd učitavati preko mreže. Budući da su sve aplikacije, uslužni programi i podatkovne datoteke pohranjene na udaljenom računalu, svaka radnja na toj radnoj stanici zahtjeva pristup mreži.

U obzir treba uzeti i količinu NFS prometa koji će se odvijati na mreži. NFS osigurava usluge udaljenih virtualnih diskova pa se na mreži neprestano koriste informacije koje se učitavaju ili spremaju na te udaljene diskove.

Razmotriti treba i velike grafičke prikaze, zamjensku memoriju (*swapping*) i datoteke stranica korištene za virtualnu memoriju, raspodijeljene aplikacije baza podataka, promet pisača i terminalski promet. To sve treba razmotriti na svakoj mreži, međutim, kreatori i korisnici LAN-ova temeljenih na osobnim računalima o tome obično ne trebaju voditi brigu. Kada je mreža spojena na veliku korisničku zajednicu, sve ove pojedinosti mrežnog okružja dobivaju na važnosti.

Treba još razmotriti i potrebu za biranim i daljinskim pristupom. Ako se taj pristup odnosi na terminal i promet na zaslonu, dovoljan će biti serijski priključak s postojećeg sustava. Ako je uspostavljena Point-to-Point Protocol (PPP) veza, potrebno je razmotriti koliko će dodatnog opterećenja mreži biti nametnuto kada će korisnici preko telefonskih linija učitavati uslužni softver, programe i baze podataka. Ovo je potrebno uzeti u obzir budući da IP nije ograničen na veze visoke brzine poput Novela IPX i drugih mrežnih protokola.

#### Razumijevanje smjernica za konfiguraciju mreže

Mrežu treba planirati na temelju smjernica i pravila. Pri planiranju mreže trebate razmotriti sljedeća pitanja:

- Kako bi se mreža koristila danas?
- Kako će se mreža koristiti u sljedećih nekoliko godina?
- Koje će se aplikacije koristiti na mreži?
- Koje će radne skupine unutar organizacije u budućnosti zahtijevati mrežne resurse?
- Koliki će broj i koje vrste radnih stanica biti na mreži?
- Koliko će poslužitelja, miniračunala i drugih domaćina biti na mreži?
- Koji će drugi mrežni uređaji, poput pisača i crtača, biti na mreži?
- Hoće li biti potrebna polja diskova?
- Da li će upravljanje mrežom biti centralizirano?
- Da li će mreža biti spojena na Internet ili druge korporacijske mreže ili će možda poslužiti kao osnova za WAN (mrežu šireg područja)?
- Koji će drugi protokoli koristiti mrežnu tehnologiju (protokoli IPX, DECNET, LAT, OSI i TCP/IP)?
- Gdje će se kritični podaci razmjenjivati (utvrdite nekoliko različitih staza)?
- Kako će mreža rasti i mijenjati se?

Kada odgovorite na sva ova pitanja, možete definirati mrežu. Broj čvorova ukazuje koliko je adresnih prostora klase C potrebno ili da li je potrebna klasa B.

Razmotriti treba i veze s udaljenim instalacijama. Opterećenje se može raspodijeliti preko više mrežnih segmenata. Pokušajte smanjiti promet koji se treba odvijati preko više mreža. Na primjer, ako imate dva sustava koji razmjenjuju mnogo informacija, a za njihovu su komunikaciju potrebni skokovi preko tri mreže, razmislite o premještanju sustava na istu mrežu.

Utvrdite najbjolju mrežnu topologiju radi udovoljavanja zahtjevima navedenima u analizi mreže. Da bi se omogućio rast mreže, najbolji je pristup utvrditi najveće opterećenje i zatim razviti mrežu u kojoj takvo opterećenje predstavlja najmanje opterećenje.

#### Uporaba usmjerivača i mostova

Za osiguravanje veze između mreža i sustava koriste se uređaji posebne namjene. Ponekad se pojmovi *gateway (prolaz)* i *router (usmjerivač)* koriste u istom značenju. Strogo gledavši, *prolaz* opisuje sustav koji poruke šalje između različitih vrsta mreža, dok *usmjerivač* šalje poruke između mreža iste vrste.

*Router (usmjerivač)* se u ovom tekstu redovito koristi za opisivanje bilo kojeg uređaja koji poruke uzima s jedne i prosljeđuje ih u drugu mrežu. Usmjerivač sadrži dovoljno inteligencije da zna treba li primljena poruka biti proslijeđena drugoj mreži ili usmjerivaču.

Usmjerivači rade u mrežnom sloju i obično su udruženi s protokolom, poput IP-a ili IPX-a. Većina usmjerivača koji usmjeruju IPX promet mogu usmjerivati i IP promet. Usmjerivači se koriste za spajanje više lokalnih mreža i mreža šireg područja. Oni pružaju način dijeljenja podataka između mreža. Također, budući da rade u sloju Network, mogu pomoći u smanjenju gustoće prometa na mreži.

Ako jedna mreža koristi mnogo različitih protokola, a druga koristi samo IP, za komunikaciju je tih mreža potreban usmjerivač koji usmjeruje samo IP adrese. Usmjerivač onemogućuje slanje poruka na mrežu koja s njima ne može raditi.

S druge strane, mostovi se mogu koristiti za međupovezivanje lokalnih i mreža šireg područja oni dijele informacije bez obzira na protokol. Most dvjema međupovezanim mrežama omogućuje da svaka istovremeno ima mnogo različitih protokola. Poruke proslijeđene preko mosta obično ne sadržavaju nikakve daljnje informacije usmjeravanja. Poruke se obično ostavljaju netaknute.

Jedna od loših strana mostova jest da se poruke iz svih međupovezanih mreža koje se šalju na više adresa ili cijeloj mreži vide na svim granama povezanim s mostom. To ima za posljedicu mnogo dodatnog zauzeća mreže zbog poruka o najnovijem stanju na mreži. Također, most prosljeđuje poruke samo mrežnim adresama na drugoj strani mosta, međutim, on može prosljeđivati sve mrežne protokole i poruke koje se šalju cijeloj mreži.

Usmjerivači i mostovi se koriste za dijeljenje informacija među mrežama. Prikladnost je svakog utvrđena mrežnim zahtjevima, korištenim protokolima, kapacitetom mreže i zahtjevima korisnika. Pravilan odabir sastavnih dijelova može pomoći učinkovitom radu mreže, omogućiti budući rast i pomoći u osiguravanju stalne pouzdanosti.

BILJEŠKA Mostove koristite samo ako mreže trebaju koristiti pakete s više protokola. Usmjerivač je inače bolji izbor budući da on pomaže u smanjenju nepotrebnog zauzeća mreže.

#### Odavde...

Više informacija o TCP/IP možete naći u sljedećim poglavljima:

- 24. poglavlje, "Podešavanje TCP/IP mreže" opisuje kako postaviti i podesiti vaš mrežni sustav za Linux.
- 25. poglavlje, "Podešavanje DNS-a" objašnjava sustav utvrđivanja Internet imena.
- 29. poglavlje, "Uporaba protokola SLIP i PPP" pokazuje kako podesiti asinkroni TCP/IP preko serijskih linija.

# Podešavanje TCP/IP mreže

(Steve Burnett)

#### U ovom poglavlju

Razumijevanje TCP/IP konfiguracijskih datoteka**484**Inicijalizacija Ethernet sučelja**486**Razumijevanje TCP/IP usmjeravanja**489**Nadziranje TCP/IP mreže s programom netstat**493** 

Podešavanje TCP/IP mreže je jedna od češćih zadaća s kojima ćete se suočavati pri administriranju Linux strojeva. U najosnovnijim slučajevima to nije presloženo, međutim, ipak zahtijeva nešto razmatranja izvedbe vaše mreže i poznavanje manjeg broja programa i konfiguracijskih datoteka.

#### Razumijevanje TCP/IP konfiguracijskih datoteka

TCP/IP umrežavanjem u Linuxu se upravlja skupom konfiguracijskih datoteka u direktoriju /etc. Te datoteke Linuxu govore koja je njegova IP adresa, ime domaćina i ime domene te nadziru mrežna sučelja. Tablica 24.1 vam prikazuje što svaka datoteka radi. Sljedeći odjeljci detaljno opisuju svaku od tih datoteka.

Tablica 24.1Konfiguracijske	datoteke Linux TCP/IP mreže
Datoteka	Opis
/etc/hosts	Imena domaćina mapira IP adresama
/etc/networks	Imena domaćina mapira mrežnim adresama
/etc/rc.d/rc3.d/S10network	Podešava postavke i aktivira vaša Ethernet sučelja pri podizanju sustava

#### Datoteka /etc/hosts

Svako računalo na TCP/IP mreži ima IP adresu, kanonsko (standardno) ime domaćina i nijedno ili više imena domaćina. Datoteka /etc/hosts je osnovna metoda mapiranja imena domaćina IP adresama.

BILJEŠKA Sva imena domaćina, imena domena i IP adrese koje se koriste u ovom poglavlju su izmišljene i ne odražavaju ni jednu stvarnu mrežu na Internetu.

Za primjer, pogledajte izmišljenu mrežu koju je izgradila Burwell, Inc. Ta se mreža sastoji od jedne adrese mreže klase B koju je Burwellu dodijelio InterNIC (organizacija koja upravlja Internet adresama). Ta je mreža podijeljena u dvije podmreže klase C. Slijedi oblik datoteke hosts:

```
# /etc/hosts for linux1.burwell.com
#
# For loopbacking.
127.0.0.1 localhost
# This machine
166.82.1.21 linux1.burwell.com linux1
                                           # the local machine
# Other hosts on our network
166.82.1.20 server.burwell.com server
                                           # the server
166.82.1.22
             wk1.burwell.com
                                             # workstation
```

```
166.82.1.10 netpr1.burwell.com netpr1
                                                                         # networked printer
166.82.1.1netpri.burwell.com netpri# networked pr166.82.1.1gateway.burwell.com gateway# the router166.82.1.1gate-if1# 1st interface on gateway166.82.2.1gate-if2# 2nd interface on gateway
                     linux2.burwell.com linux2
166.82.1.30
                                                                         # Laptop via PLIP
# end of hosts file
```

SAVJET Uočite da prethodni prolaz ima dva imena domaćina za IP adresu 166.82.1.1. Dobra je zamisao svakom sučelju na stroju zadati jednistveno ime. Na taj je način lakše vidjeti što se događa kada koristite naredbe ifconfig i route.

> Format datoteke hosts se sastoji od jedne IP adrese po retku počevši u prvom stupcu, kanonskog imena domaćina pridruženog toj adresi i zatim nijednog ili više zamjenskih imena. Polja su odvojena razmacima ili tabulatorima. Prazni redovi i tekst koji slijedi iza znaka # se smatraju napomenama i zanemaruju se.

IP adresa 127.0.0 je poznata kao local loopback address (lokalna povratna adresa) i rezervirana je za tu svrhu. Njoj se obično dodjeljuje ime localhost. Ako ćete vaš stroj koristiti samo kao samostalan sustav ili ćete koristiti SLIP ili PPP za spajanje na vanjski svijet, u vašoj datoteci hosts trebate samo localhost adresu.

BILJEŠKA Zadaću je datoteke /etc/hosts na strojevima spojenim na Internet ili velike unutarnje mreže uglavnom preuzeo Domain Name Service (DNS). Međutim, DNS nije raspoloživ tijekom podizanja sustava ili kada radite u jednokorisničkom modu, stoga je dobra zamisao informacije za neophodne strojeve poput poslužitelja i prolaza staviti u /etc/hosts.

Na mreži sa samo nekoliko strojeva koji nisu spojeni na Internet je lakše u /etc/hosts držati cjeloviti ispis svih domaćina nego postavljati i održavati DNS.



SAVJET Davanje imena vašim mrežama omogućuje pogodnosti poput statičnog usmjeravanja koje uzima ime domaćina ili ime mreže. Podmreže ne trebate pamtiti po njihovim IP adresama, dovolino je znati njihova imena.

#### Datoteka /etc/networks

Baš kao što domaćini imaju imena i IP adrese, tako se i mrežama i podmrežama mogu dati imena. S tim se imenima barata u datoteci /etc/networks. IP adrese u datoteci networks uključuju samo dio s mrežnom adresom plus podmrežni bajt. U sljedećem je primjeru ogledna datoteka za burwell.com:

```
# /etc/networks for burwell.com
localnet 127.0.0.0 # software loopback network
burwell-c1 166.82.1 # Development Group Network, Class C
burwell-c2
                 166.82.2
                                 # MIS Network, Class C
# end of networks file
```

Prvo je ime localnet i IP adresa, 127.0.0.0. Ako vaš Linux stroj ne spajate na TCP/IP mrežu ili koristite samo SLIP ili PPP, sve što u ovu datoteku trebate staviti jest ime localnet i IP adresu.

Sljedeća dva reda identificiraju dvije podmreže klase C koje je Burwell napravio iz svoje mreže klase B.

# Inicijalizacija Ethernet sučelja

Program ifconfig čini mrežna sučelja poput softverske povratne petlje i Ethernet kartica poznatih Linuxovoj jezgri, tako da ih Linux može koristiti. Program ifconfig se koristi i za nadziranje i promjenu stanja mrežnih sučelja. Jednostavan poziv za ifconfig jest

ifconfig adresa sučelja

koji aktivira navedeno mrežno sučelje i dodjeljuje mu IP adresu. To se naziva *dovođenje sučelja*. Uopćena je pozivna sintaksa za ifconfig sljedeća:

ifconfig sučelje [aftype] [opcije] | adresa

U tablici 24.2 je popis argumenata naredbenog retka za ifconfig.

Tablica 24.2 Algumenti hareubenog retka za irconing		
Argument	Opis	
interface (su <b>č</b> elje)	Ime mrežnog sučelja, obično ime upravljačkog programa uređaja iza kojeg slijedi identifikacijski broj. Ovaj je argument obavezan.	
aftype (apvrsta)	Adresna porodica koju treba koristiti za dekodiranje i prikazivanje svih adresa protokola. Sada su podržane adresne porodice inet (TCP/IP), ddp (Appletalk Phase 2), ipx (Novell) te AX.25 i netrom (obje za amaterski packet radio). Unaprijed je zadana porodica inet.	
up	Aktivira navedeno sučelje.	
down	Deaktivira navedeno sučelje.	
[-]arp	Uključuje ili isključuje uporabu ARP protokola na navedeno sučelje. Znak minus se koristi za isključivanje zastavice.	
[-]trailers	Uključuje ili isključuje repne informacije na Ethernet paketima.	
[-]allmulti	Uključuje ili isključuje promiskuitetni način rada sučelja. Kada je uključen, sučelje jezgri šalje sav promet na mreži, a ne samo onaj koji je adresiran na vaš stroj.	
metric N	Mjere sučelja postavlja u cjelobrojnu vrijednost N. Metrička vrijednost predstavlja "cijenu" slanja paketa na tom smjeru. Linuxova jezgra tu cijenu smjera trenutno ne koristi, ali će ona biti primijenjena u budućnosti.	

Tablica 24.2 Argumenti naredbenog retka za ifconfig

Argument	Opis
mtu N	U cjelobrojnoj vrijednosti N podešava najveći broj bajtova s kojima sučelje može baratati u jednom prijenosu. Trenutni mrežni kôd u jezgri ne barata s IP fragmentacijom, stoga za MTU (Maximum Transmission Unit) podesite dovoljno veliku vrijednost.
dstaddr <i>addr</i>	Postavlja IP adresu drugog kraja point-to-point veze. Ključna riječ pointopoint ga je učinila zastarjelim.
netmast <i>addr</i>	Postavlja IP mrežnu masku za navedeno sučelje.
irq <i>addr</i>	Postavlja prekidni redak koji ovaj uređaj koristi. Upamtite da mnogi uređaji ne podržavaju dinamičko IRQ podešavanje.
[-]broadcast[ <i>addr</i> ]	Postavlja adresu emitiranja za sučelje kada je uključena i adresa. Ako se ne izda nikakva adresa, uključit će se zastavica IFF_BROADCAST za navedeno sučelje.
[-]pointopoint[ <i>addr</i> ]	Uključuje point-to-point način rada na navedenom sučelju. To jezgri govori da je ovo sučelje izravna veza do drugog stroja. Kada je i adresa uključena, ona se dodjeljuje stroju na drugom kraju popisa. Ako adresa nije dana, uključena je zastavica IFF_POINTOPOINT za sučelje. Vodeći znak minus isključuje zastavicu.
hw	Postavlja hardversku adresu za navedeno sučelje. Iza ove ključne riječi mora slijediti ime hardverske klase i ASCII istoznačnica hardverske adrese. Sada su podržani Ethernet (ether), AMPR AX.25 (ax25) i PPP (ppp).
address	Ime domaćina ili IP adresa koja treba biti doznačena navedenom sučelju. Imena domaćina koja se ovdje koriste odgovaraju njihovim istoznačnicama u IP adresama. Ovaj je parametar obavezan.

Obično ne trebate koristiti sve opcije. ifconfig treba samo ime sučelja, *netmask* (mrežnu masku) i dodijeljenu IP adresu da bi podesio sve što je potrebno. Samo u slučaju kada ifconfig ne uspije ili kada imate složenu mrežu, trebat ćete većinu parametara sami postavljati.

#### **UPOZORENJE**

Ako je vaš Linux stroj na mreži, program ifconfig mora biti osiguran od neovlaštene uporabe. Postavljanje mrežnog sučelja u način rada u kojem prima sve pakete bez obzira na odredišnu adresu omogućuje drugim osobama "njuškanje" po vašoj mreži i dobivanje osjetljivih podataka poput lozinki. To je ozbiljno kršenje sigurnosti.

Vidi "Upravljanje fizičkom sigurnošću", str. 236.

#### Uporaba ifconfig za provjeru mrežnog sučelja

Kada se ifconfig pokrene bez argumenata, on će izdati stanja svih mrežnih sučelja za koja jezgra zna. Kada se ifconfig izvršava samo s imenom sučelja u naredbenom retku, on će ispisati stanje tog sučelja:

Ovaj primjer koristi lo, softversko povratno sučelje. Možete vidjeti dodijeljenu IP adresu, (inet addr), adresu emitiranja (Bcast) i mrežnu masku (Mask). Sučelje je UP s MTU-om od 2000. Posljednja dva retka daju statistiku s brojem primljenih (RX) i poslanih paketa (TX), zajedno s brojem paketnih grešaka, ispuštenih paketa i grešaka prestizanja.

#### Podešavanje povratnog sučelja

Svi Linux strojevi s mrežnim slojem instaliranim u jezgri imaju sučelje s povratnom petljom. To se sučelje koristi za ispitivanje mrežnih aplikacija i za pružanje mreže lokalnim TCP/IP servisima kada stroj nije spojen na stvarnu mrežu.

Ime mrežnog sučelja za povratni sustav je lo. Za pokretanje ifconfig unesite sljedeće:

ifconfig lo 127.0.0.1

To će aktivirati povratno sučelje i dodijeliti mu adresu 127.0.0.1. Ova se adresa uobičajeno koristi za povratnu petlju budući da InterNIC za mrežu klase nikome nikada neće dodijeliti adresu 127.0.0.

Da biste *povratni sustav* učinili posve spremnim za rad, trebate za njega dodati smjer s naredom route, koja se objašnjava kasnije u članku "Razumijevanje TCP/IP usmjeravanja".

#### Podešavanje mrežnog sučelja

Za podešavanje je Ethernet mrežnog sučelja potrebno malo više rada, naročito ako koristite podmreže. Osnovni poziv za ifconfig izgleda poput ovog za linux1.burwell.com:

```
ifconfig eth0 linux1
```

Time ćete ifconfig natjerati da aktivira Ethernet sučelje 0, u datoteci /etc/hosts pogleda IP adresu za linux1 i dodijeli je ovom sučelju. Pregledavanje eth0 sučelja na ovoj točki otkriva sljedeći kôd:

```
$ ifconfig ethØ
eth0 Link encap 10Mbps Ethernet Hwaddr 00:00:E1:54:3B:82
inet addr 166.82.1.21Bcast166.82.1.255 Mask 255.255.255.0
UP BROADCAST RUNNING MTU 1500 Metric 0
RX packets 3136 errors 217 dropped 7 overrun 26
TX packets 1752 errors 25 dropped 0 overrun 0
Interrupt:10 Base address:0x300
```

Uočite da je ifconfig na osnovi IP adrese koju je pronašao u /etc/hosts automatski postavio adresu emitiranja (radne stanice) i mrežnu masku. Ako koristite podmreže, trebate sami jasno navesti adresu radne stanice i mrežnu masku. Na primjer, ako imate mrežu klase C i koristite prvi bit iz dijela domaćina u adresi da biste napravili dvije podmreže, pri pokretanju ifconfig trebate navesti adresu emitiranja i mrežnu masku:

ifconfig eth0 linux1 broadcast 166.82.1.127 netmask 255.255.255.128

#### Podešavanje paralelnih IP sučelja

ifconfig s Parallel IP (PLIP), Serial Line IP (SLIP) i Point-to-Point Protocol (PPP) sučeljima upravlja ponešto drukčije. Za dovođenje PLIP sučelja u ifconfig naredbeni redak dodajete opciju pointopoint. Pretpostavimo da je Burwelovo prijenosno računalo linux2 priključeno na prvi paralelni priključak na linux1. Za aktiviranje PLIP veze ovako pozivate ifconfig:

ifconfig plip0 linux1 pointopoint linux2

To će aktivirati plip0 sučelje s IP adresom za linux1, postaviti pointopoint zastavicu i sučelju reći da je linux2 IP adresa za drugi kraj veze. ifconfig će u /etc/hosts potražiti IP adrese za linux1 i linux2 i te će adrese odgovarajuće doznačiti. Na prijenosniku koristite sličan poziv

ifconfig plip0 linux2 pointopoint linux1

Vidi "Razumijevanje zahtjeva za SLIP i PPP", str. 562.

#### Razumijevanje TCP/IP usmjeravanja

Usmjeravanje utvrđuje stazu kojom će paket prolaziti od svog izvora kroz mrežu do svog odredišta. Ta se staza utvrđuje uspoređivanjem odredišne IP adrese s tablicama usmjeravanja u jezgri i slanjem paketa u naznačeni stroj, koji može, ali ne mora biti odredište paketa. Tablica usmjeravanja u jezgri sadrži informacije u obliku "Za dolaženje od stroja Y do mreže X, paket pošalji u stroj Z s troškom 1", zajedno s vrijednostima *time-to-live* (TTL - koliko se dugo paket smije zadržavati) i pouzdanosti za taj smjer.

#### Odlučivanje o pravilima usmjeravanja

Prvi je korak u postavljanju usmjeravanja na vašoj mreži odlučivanje o pravilima usmjeravanja. Za male, nepovezane mreže je dovoljna uporaba naredbe route za postavljanje statičnih smjerova na svakom stroju tijekom podizanja sustava. Velike mreže s mnogo podmreža ili mreža spojenih na Internet trebaju koristiti dinamičko usmjeravanje. Program usmjeravanja će dinamičko usmjeravanje osigurati komuniciranjem s programima usmjeravanja na drugim strojevima i instaliranjem smjerova na temelju onog što je saznao o topologiji mreže.

Vrlo se često koristi kombinacija statičkog i dinamičkog usmjeravanja. Strojevi na svakoj podmreži za dosezanje svojih neposrednih susjeda koriste statičko usmjeravanje. Unaprijed zadani smjer - smjer korišten za pakete različit od svih ostalih smjerova u tablici smjerova - podešen je za *gateway* (prolaz) - stroj koji vrši dinamičko usmjeravanje i svjestan je ostatka svijeta. Na ovaj se način mogu planirati velike mreže, smanjujući gnjavažu s konfiguracijskim datotekama i količinom propusnosti koju koriste programi za dinamičko usmjeravanje.

#### Uporaba programa /sbin/route

Program /sbin/route upravlja s tablicom smjerova (*Kernel routing*) i koristi se za određivanje statičkih smjerova do drugih računala ili mreža preko sučelja koja su podešena i aktivirana s ifconfig. To se obično obavlja skriptom /etc/rc.d/rc3.d/S10network tijekom podizanja sustava. Tablica 24.3 opisuje argumente naredbenog retka za /sbin/route.

	Argument	Opis
	(ništa)	/sbin/route bez argumenata izdaje trenutnu tablicu smjerova.
	- n	Ovaj argument uzrokuje isti izlaz kao i kada nema argumenata, ali imena domaćina zamjenjuje s njhovim brojčanim IP adresama.
	del	
	add	

Tablicca 24.3 Argumenti naredbenog retka za /sbin/route

**Pregledavanje tablice smjerova** Pokretanjem /sbin/routeb bez ikakvih argumenata naredbenog retka ili samo s -n dobivamo ispis tablice smjerova:

/sbin/route Kernel routing table Destination Gateway Genmask Flags Metric Ref UseIface 127.0.0.0 \* 255.0.0.0 U 0 0 100 lo

Ovo je iz stroja u kojem je aktivirano samo povratno sučelje. Tablica 24.4 pruža opis polja u izvještaju tablice smjerova.

Tablica 24.4 Polja u izvještaju tablice smjerova				
Polje	Opis			
Destination	IP adresa odredišta smjera.			
Gateway	Ime domaćina ili IP adresa prolaza koji taj smjer koristi. Ako prolaz ne postoji, izlaz je zvjezdica (*).			
Genmask	Mrežna maska za odredišnu adresu. Jezgra je koristi za generalizaciju puta do odredišne adrese tako da primijeni logičku operaciju I na bitove Genmaska i paketne IP adrese prije usporedbe s odredišnom adresom.			

Polje	Opis
Flags	Zastavice za smjer (U znači gore (up), H znači host (domaćin), G znači gateway (prolaz), D znači dinamički smjer, a M znači modified (preinačen).
Metric	Metrička cijena za smjer. Trenutno nije podržano u mrežnom sloju jezgre.
Ref	Broj drugih smjerova koji se oslanjaju na prisutnost ovog smjera.
Use	Ovaj broj pokazuje koliko je puta stavka tablice smjerova korištena.
Iface	Mrežno sučelje kojem ovaj smjer isporučuje pakete.

Vrativši se na Burwellovu mrežu, sljedeći je primjer s prijenosnika linux2, s uspostavljenom SLIP vezom koja je trenutno u radu:

<pre>\$ /sbin/route</pre>							
Kernel routing	table						
Destination	Gateway	Genmask	Flags	Metric	Ref	Use	Iface
<pre>slip.burwell.c</pre>	*	255.255.255.255	UH	0	0	0	s10
127.0.0.0	*	255.0.0.0	U	0	0	100	10
default	<pre>slip.burwell.c</pre>	*	UG	0	0	1	s10

Stavka tablice za povratnu petlju je ista kao i prije, a tu su i dvije nove stavke. Prva određuje smjer za slip.burwell.com. Druga nova stavka određuje unaprijed zadani smjer preko prolaza slip.burwell.com.

**BILJEŠKA** Svaki stroj spojen na mrežu mora u svojoj tablici smjerova imati unaprijed zadani smjer. Unaprijed zadani smjer se koristi kada se u tablici smjerova ne pronađe odgovarajuća stavka za odredište paketa.

**Dodavanje statičkih smjerova** Smjerove u tablicu smjerova dodajete pokretanjem programa route s argumentom add. Sintaksa argumenta naredbenog retka za naredbu route add jest

route add [ -net ¦ -host ] addr [gw gateway] [metric cost] ⇒[netmask mask] [dev device]

Tablica 24.5 opisuje argumente naredbenog retka koje koristi naredba route add.

lablica 24.5 Argumenti naredbenog retka za naredbu route add				
Argument	Opis			
-net¦-host	Iznuđuje odnos prema navedenoj adresi kao prema adresi mreže ili domaćina.			
addr	Odredišna adresa za novi smjer. To može biti IP adresa, ime domaćina ili ime mreže.			
gw <i>gateway</i>	Određuje usmjeravanje svih paketa za ovu adresu preko navedenog prolaza.	nastavlja se		

Tablica 24.5 Argumenti naredbenog retka za naredbu route add

Tablica 24.5 Nastavak				
Argument	Opis			
metric <i>cost</i>	U tablicu smjerova postavlja polje metric.			
netmask <i>mask</i>	Navodi mrežnu masku dodanog smjera. Program route će je i sam pretpostaviti pa je u uobičajenim prilikama ne trebate navoditi.			
dev <i>device</i>	Prisiljava route da novi smjer pridruži navedenom uređaju mrežnog sučelja. route i ovdje obično ispravno pretpostavi koji uređaj treba koristiti za novi smjer, stoga ovo ne trebate često koristiti.			
	mrežnog sučelja. route i ovdje obično ispravno pretpostavi koji uređaj treba koristiti za novi smjer, stoga ovo ne trebate često koristiti.			

#### **UPOZORENJE**

Kada tablici smjerova dodajete smjer preko prolaza, trebate osigurati da se do navedenog prolaza može doprijeti. Prije dodavanja smjera koji koristi prolaz obično trebate dodati statički smjer za taj prolaz.

**Neki primjeri usmjeravanja** Počinjemo s primjerom za povratno sučelje. Nakon postavljanja povratnog sučelja s ifconfig, trebate mu dodati smjer, kao u sljedećem primjeru:

```
# route add 127.0.0.1
```

Ništa drugo nije potrebno budući da route datu mu adresu upoređuje s adresama za poznata sučelja i povratno sučelje dodjeljuje novom smjeru. Sljedeći primjer pokazuje kako postaviti usmjeravanje za SLIP vezu na Burwellovom stroju linux2 nakon što je SLIP veza uspostavljena i ifconfig upotrebljen za aktiviranje sučelja:

```
# route add slip.burwell.com
# route add default gw slip.burwell.com
```

Prva naredba dodaje statički smjer za domaćina slip.burwell.com. Druga naredba jezgri kaže da slip.burwell.com koristi kao prolaz za sve pakete s nepoznatim odredištima.

#### **UPOZORENJE**

Osigurajte da se sva imena domaćina koja koristite s naredbom route nalaze u datoteci /etc/hosts tako da route može pronaći IP adrese za njih, inače neće moći obaviti zadatak.

Ako u vašoj mreži uspostavljate podmreže dijeljenjem IP adresa u sredini okteta, pri pokretanju ćete programa route trebati navesti traženu mrežnu masku. Na primjer, ako imate mrežu klase C i četiri podmreže koje koriste prva dva bita posljednjeg okteta, route ćete trebati pokretati na sljedeći način:

# route add hostname netmask 255.255.255.192

Ovime osiguravate da će route u stavku tablice smjerova staviti ispravnu mrežnu masku.

Za Ethernet i druga sučelja za mrežno emitiranje trebate dodati smjerove koji jezgri govore koje se mreže mogu dosegnuti preko svakog podešenog sučelja. Nakon uporabe ifconfig za dovođenje eth0 mrežnog sučelja na linux1.burwell.com kao što ste prethodno uradili, trebate pokrenuti route da instalira smjer do mreže na tom sučelju:

# route add -net 166.82.1.0

To se možda ne čini dovoljnim za ispravno podešavanje tablice smjerova, budući da nikakvo sučelje nije naznačeno, međutim, route će sučelje pronaći uspoređivanjem IP adrese u naredbenom retku s IP adresom svakog mrežnog sučelja i smjer doznačuje tako pronađenom sučelju. U ovom je slučaju za eth0 bila dodijeljena adresa 166.82.1.21 s mrežnom maskom 255.255.255.0. To se slaže s mrežnom adresom zadanom u naredbi route, pa će route instalirati smjer do mreže 166.82.1.0 uporabom sučelja eth0, kako slijedi:

```
$ route
Kernel routing table
Destination Gateway Genmask Flags Metric Ref UseIface
166.82.1.0 * 255.255.255.0 UN 0 0 0 eth0
127.0.0.0 * 255.0.0.0 U 0 0 100 lo
```

Da bi linux1 znao kako dosegnuti drugu podmrežu, trebaju vam još dvije stavke u tablici smjerova radi sigurnosti.

```
# route add gateway.burwell.com
# route add -net 166.82.2.0 gw gatway.burwell.com
```

S ovime dodajete statički smjer za gateway.burwell.com i zatim dodajete mrežni smjer za 166.82.2.0 uporabom gatweay.burwell.com kao prolaza za tu mrežu, kako je prikazano u sljedećem:

```
        $ route

        Kernel routing table

        Destination
        Gateway

        Genmask
        Flags

        Metric Ref
        UseIface

        gateway.burwell
        *

        255.255.255.0
        UH
        0
        0

        166.82.1.0
        *
        255.255.255.0
        UN
        0
        0

        166.82.2.0
        gateway.burwell
        255.255.255.0
        UN
        0
        0
        0

        127.0.0.0
        *
        255.0.0.0
        U
        0
        0
        100
        10
```

Ovo prikazuje statički smjer koji ste dodali za gateway.burwell.com i smjer s prolazom do mreže 166.82.2.0.

**Brisanje smjerova s naredbom route** Smjerove brišete pozivanjem routea s opcijom del i navođenjem odredišne adrese smjera koji želite izbrisati. Na primjer,

```
# route del -net 166.82.2.0
```

briše mrežni smjer za mrežu 166.82.2.0.

#### Nadziranje TCP/IP mreže s programom netstat

Program netstat je vrlo koristan alat za praćenje vaše TCP/IP mreže. On može prikazati tablicu smjerova, stanje aktivnih veza na mreži i korisnu statistiku o svakom mrežnom sučelju. Tablica 24.6 opisuje uobičajene argumente naredbenog retka za netstat. Dodano je i nekoliko argumenata namijenjenih naprednim korisnicima. O više informacija pogledajte odgovarajuću *man* stranicu.

Tablica 24.6	Uobicajeni argumenti naredbenog retka za program netstat	
Argument	Opis	
- a	Prikazuje informacije o svim vezama s Internetom, uključujući i one koje se upravo osluškuju.	
- i	Prikazuje statistike za sve mrežne uređaje.	
- C	Prikazuje neprestano osvježavano mrežno stanje. nestat će svake sekunde prikazivati stanje na mreži sve dok ne bude prekinut.	
- N	Prikazuje udaljene i lokalne adrese i informacije o portovima u brojčanom/sirovom obliku umjesto pretvaranja u imena.	
- 0	Prikazuje vremenski zavisna stanja svake mrežne veze.	
- r	Prikazuje tablicu smjerova u jezgri.	
-t	Prikazuje samo TCP informacije , uključujući i one koje se upravo osluškuju.	
- U	Prikazuje samo UDP informacije	
- V	Prikazuje inačicu informacije za netstat.	
- W	Prikazuje sirove informacije s priključnice (socket).	
- X	Prikazuje informacije s priključnice UNIX domene.	

# \_ . . ..

#### Prikazivanje aktivnih mrežnih veza

netstat pokrenut bez argumenata naredbenog retka stvara ispis aktvnih mrežnih veza na vašem stroju. Sljedeći primjer prikazuje unaprijed zadani izlaz iz netstata:

```
$ netstat
Active Internet connections
Proto Recv-Q Send-Q Local Address
                                            Foreign Address
                                                                    (State)
                   0 linux1.burwell.com:1266 server.burwell.:telnet ESTAB-
tcp
           0
LISHED
Active UNIX domain sockets
Proto RefCnt Flags
                        Туре
                                                   Path
                                       State
unix 1
             [ ACC ]
                        SOCK_STREAM
                                       LISTENING
                                                   /dev/printer
unix 2
             []
                        SOCK_STREAM
                                       CONNECTED
                                                   /dev/log
unix 2
              []
                        SOCK STREAM
                                       CONNECTED
unix 1
              [ ACC ]
                        SOCK_STREAM
                                       LISTENING
                                                   /dev/log
```

Prvi dio prikazuje aktivnu TCP protokol vezu korisnika burt između adrese linux1.burwell.com na portu 1266 i adrese server.burwell.com na telnet portu.

Tablica 24.7 opisuje polja u ispisu Active Internet Connections.

Tablica 24.7 Polja	adlica 24.7 Polja u Active Internet Connections			
Polje	Opis			
Proto	Protokol korišten za tu vezu – TCP ili UDP.			
Recv-Q	Broj bajtova primljenih na ovoj priključnici, ali koje korisnički program još nije kopirao.			
Send-Q	Broj bajtova poslar još nije potvrđen.	Broj bajtova poslanih udaljenom domaćinu čiji primitak još nije potvrđen.		
Local Address	Ime lokalnog domaćina i broja porta dodijeljenog ovoj vezi. IP adresa priključnice je pretvorena u kanonsko ime domaćina za tu adresu, a broj porta je pretvoren u ime servisa osim ako je korištena zastavica -n.			
Foreign Address	Ime stranog domaćina i broj porta dodijeljenog toj vezi. Zastavica -n na ovo polje utječe jednako kao i na polje Local Address.			
State	Trenutno stanje pri	ključnice. Moguća su sljedeća stanja:		
	ESTABLISHED	Veza je u potpunosti uspostavljena.		
	SYN_SENT	Socket sada pokušava uspostaviti vezu s udaljenim domaćinom.		
	SYN_RECV	Veza je incijalizirana.		
	FIN_WAIT1	Priključnica je zatvorena i čeka da se veza isključi.		
	FIN_WAIT2	Veza je isključena. Priključnica čeka signal isključivanja iz udaljenog domaćina.		
	TIME_WAIT	Priključnica je zatvorena i čeka retransmisiju signala isključivanja.		
	CLOSED	Priključnica nije u uporabi.		
	CLOSE_WAIT	Udaljeni domaćin je isključio svoju vezu. Lokalni domaćin čeka zatvaranje priključnice.		
	LAST_ACK	Udaljena veza je isključena i priključnica je zatvorena. Lokalni domaćin čeka odobrenje.		
	LISTEN	Priključnica osluškuje dolazeći pokušaj uspostavljanja veze.		
	UNKNOWN	Stanje priključnice nije poznato.		
User	Prijavni ID korisnika u čijem je vlasništvu ta priključnica.			

Drugi dio prikazuje aktivne priključnice UNIX domene. Priključnice UNIX domene su IPC (interprocess communication) mehanizmi (mehanizmi međuprocesne komunikacije) koji kao sustav sastajanja koriste UNIX datotečni sustav. Procesi u datotečnom sustavu stvaraju posebne datoteke koje zatim otvaraju drugi procesi na stroju koji želi komunicirati. Prethodni netstat ispis prikazuje dvije priključnice koje osluškuju: jedna na /dev/printer,

a druga na /dev/log. Tu su i dvije trenutno spojene priključnice: jedna na /dev/log i druga za koju nije navedena pridružena staza. Tablica 24.8 opisuje polja u popisu Active UNIX Domain Sockets.

Tablica 24.8	Polja u popisu Active UN	ija u popisu Active UNIX Domain Sockets			
Polje	Opis	Opis			
Proto	Protokol u uporabi z	Protokol u uporabi za ovu priključnicu. To će obično biti unix.			
RefCnt	Broj procesa pridruž	Broj procesa pridruženih toj priključnici.			
Flags	Zastavice za tu prik SO_ACCEPTON (A (to jest, na njoj trem napravio priključnic	Zastavice za tu priključnicu. Za sada je jedina poznata zastavica SO_ACCEPTON (ACC), koja ukazuje da je priključnica odspojena (to jest, na njoj trenutno nije uspostavljena veza) i da proces koji je napravio priključnicu čeka zahtjev za uspostavu veze.			
Туре	Način pristupa prikl od sljedećih ključnil	jučnici. Ovo će polje sadržavati jednu h riječi:			
	OCK_DGRAM	Datagram, način bez uspostavljene veze.			
	OCK_STREAM	Protočni način rada.			
	OCK_RAW	Sirovi način.			
	OCK_RDM	(Reliably delivered message) Pouzdana dostava poruke.			
	OCK_SEQPACKET	Sekvencijalni paketni način rada.			
	NKNOWN	Način nepoznat programu netstat.			
State	Trenutno stanje prik	ljučnice. Koriste se sljedeće ključne riječi:			
	FREE	Priključnica nije dodijeljena.			
	LISTENING	Priključnica čeka zahtjev za uspostavu veze.			
	UNCONNECTED	Na priključnici trenutno nije uspostavljena veza.			
	CONNECTING	Priključnica trenutno pokušava uspostaviti vezu.			
	CONNECTED	Priključnica trenutno ima uspostavljenu vezu.			
	DISCONNECTING	Priključnica pokušava isključiti vezu.			
	UNKNOWN	Stanje priključnice nije poznato. Ovo nećete vidjeti pod uobičajenim uvjetima djelovanja.			
Path	Ovo je ime staze ko	Ovo je ime staze koju drugi procesi koriste za spajanje na priključnicu.			

Tablica 24.8 Polja u popisu Active UNIX Domain Sockets


SAVJET Mrežno sučelje koje ispušta mnogo paketa ili dobiva mnogo grešaka prebrzog pristizanja podataka, može biti pokazatelj preopterećenog stroja ili mreže. Provjerom statistike mrežnog sučelja možete dobiti brzu dijiagnozu ovog problema.

> Kada se netstat pozove s opcijom -o, on u popis Active internet Connections dodaje informaciju o unutarnjem stanju, kako je prikazano u sljedećem primjeru:

```
$ netstat -o
Active Internet connections
Proto Recv-Q Send-Q Local Address
                                  Foreign Address
                                                  (State)
     0 0 localhost:1121
                                 localhost:telnet ESTABLISHED
tcp
⇔off (0.00/0)
              0 localhost:telnet localhost:1121 ESTABLISHED
tcp
     0
⇒ON (673.69/0)
```

Dodani podaci su na kraju svakog retka i uključuju brojač retransmisije prijemnika, brojač retransmisije predajnika, stanje mjerača vremena (on/off) i vrijednosti vrijeme/odstupanje (u zagradama). Prikazano vrijeme je vrijeme prije isteka mjerača vremena. Odstupanje (backoff) predstavlja broj ponovnih pokušaja za trenutni prijenos podataka. Ti su podaci korisni u dijagnosticiranju mrežnih problema jer je s njima lako vidjeti koja veza ima poteškoća.

Budući da opcija -o izdaje stanje unutarnjih TCP/IP podataka, u kasnijem se izdanju mrežnog softvera format ovih podataka može promijeniti ili će možda ova opcija biti uklonjena.

#### Pregledavanje tablice Kernel routing

netstat pozvan s opcijom -r ispisuje tablicu smjerova Kernel routing. Format je isti kao i za naredbu route.

#### Prikazivanje statistike mrežnog sučelja

Pozivanjem netstata s opcijom -i dobivamo ispis statistike uporabe za svako aktivno mrežno sučelje - još jedan izvrstan alat za otkrivanje uzroka poteškoća na mreži. S ovom je naredbom vrlo lako vidjeti kada su paketi ispušteni, kada se ne mogu obraditi zbog prevelike brzine pristizanja itd.

Slijedi primjer uporabe opcije -i, a u tablici 24.9 su objašnjena sva polja u popisu.

```
$ netstat -i
Kernel Interface table
Iface MTU Met RX-OK RX-ERR RX-DRP RX-OVR TX-OK TX-ERR TX-DRP RX-OVR Flags
10
     2000 0
               0
                     0
                           0 0 1558 1
                                                     0
                                                           0
                                                              I RU
```

Tablica 24.9	Polja u tablici Kernel Interface	u tablici Kernel Interface	
Polje	Opis		
Iface	Ime mrežnog sučelja.		
MTU	Najveći broj bajtova koji se s ovim sučeljem može poslati u jednom prijenosu.		
Met	Metrička vrijednost za ovo sučelje.		
RX-OK	Broj paketa primljenih bez greške.		
RX - ERR	Broj paketa primljenih s greškama.		
RX - DRP	Broj ispuštenih paketa.		
RX-OVR	Broj grešaka prestizanja (overrun).		
ТХ-ОК	Broj paketa poslanih bez grešaka.		
TX-ERR	Broj paketa poslanih s greškama.		
TX-DRP	Broj paketa ispuštenih tijekom prijenosa.		
TX-OVR	Broj paketa ispuštenih zbog grešaka prestizar	ıja.	
Flags	U ovom polju mogu biti prikazane sljedeće z	astavice:	
	A Sučelje prima pakete namijenjene	za više odredišta.	
	B Sučelje prima emitirane pakete (po	oslane svim stanicama).	
	D Aktivirana je mogućnost otkrivanj	a grešaka na sučelju.	
	L (loopback) Ovo je povratno sučelj	e.	
	M Sučelje prima sve pakete bez obzi	ra na odredišnu adresu.	
	N Sučelje ne obrađuje repne informa	icije paketa.	
	0 Address Resolution Protocol je na	ovom sučelju isključen.	
	P Ovo je sučelje korišteno kao PPP	(point-to-point) veza.	
	R Sučelje je trenutno u radu.		
	U Sučelje je aktivirano.		

#### Odavde...

Ovo poglavlje pokriva osnove podešavanja Linux stroja za uporabu na mreži. Dodatne se informacije mogu naći na *man* stranicama za ovdje spomenute naredbe. O više informacija o TCP/IP umrežavanju i podešavanju pogledajte sljedeća poglavlja:

- 18. poglavlje, "Razumijevanje Linux ljuske" pruža više pojedinosti o pisanju skripta ljuske.
- 23. poglavlje, "Razumijevanje protokola TCP/IP" objašnjava pojedinosti o TCP/IP protokolima.
- 25. poglavlje, "Podešavanje DNS-a" vam pokazuje kako Linux podesiti kao DNS klijent i poslužitelj.

## P 0 G L A V L J E **25**

### Podešavanje DNS-a

(Steve Burnett)

#### U ovom poglavlju

Upoznavanje DNS-a Postavljanje resolvera Uporaba demona *named* za podešavanje poslužitelja Rješavanje problema Prvobitno je, kada je Internet tek osnovan, broj domaćina na Mreži bio vrlo malen. Mapiranje je imena/adresa bilo vrlo lako održavati. Svaki je domaćin naprosto u lokalnoj datoteci imao cjelokupni popis svih imena domaćina i adresa. Kako se rast Interneta ubrzavao, taj je sustav brzo postao glomazan i nespretan. Kada je dodan novi domaćin,

bilo je nužno obnoviti svaku datoteku domaćina na svakom računalu. Pored toga, budući da je za svako novo računalo u svakoj datoteci domaćina trebalo stvoriti novi redak, datoteke domaćina su počele narastati na priličnu veličinu. Bilo je očito da je potrebno novo rješenje.

Mapiranje je imena Internet sustava u IP adrese zadatak koji zahtijeva prilično promišljanja. Uz eksplozivni rast Interneta kroz posljednjih nekoliko godina, prvobitni se sustav održavanja imena domaćina za mapiranje IP adresa u lokalnoj čistoj ASCII datoteci

brzo pokazao kao nepraktičan. S tisućama računala na Mreži i sve više novih koji su svakodnevno dodavani bilo je potrebno stvoriti novi sustav. Novi je sustav bio širom mreže distribuirana baza podataka poznata kao BIND - Berkley Internet Domain poslužitelj.

Ovaj sustav, koji još nazivamo Domain Name Service, Domain Name Server ili DNS, pruža učinkovit i dovoljno transparentan način mapiranja imena domaćina na IP adrese.

DNS je ozloglašen po tome što ga je teško postaviti i podesiti, ali kada u tome uspijete, prilično ga je jednostavno održavati. Ovo poglavlje pruža osnovni pregled instalacije i podešavanja DNS sustava. To ni u kom slučaju nisu potpune upute jer su o toj temi napisane cijele knjige.

#### Upoznavanje DNS-a

DNS pruža mehanizam za pretvaranje IP adresa u lakše pamtiva imena koja služe kao zamjenska imena za adrese domaćina, mreža i za poštu. On to čini dijeljenjem cjelokupnog adresnog prostora na Internetu u različite logičke skupine. Svaka skupina ima ovlasti za svoja vlastita računala i druge informacije.

Budući da je DNS zamršena tema, za njega su razvijeni posebni pojmovi. U tablici 25.1 je popis s opisima nekih često korištenih DNS pojmova.

Pojam	Opis		
domena (domain)	Logičko tijelo ili organizacija koja predstavlja dio mreže. Na primjer, unc.edu je ime primarne domene za University of North Carolina u Chapel Hillu.		
ime domene (domain name)	Dio imena domaćina koji predstavlja domenu koju sadržava domaćin. Na primjer, u adresi sunsite.unc.edu ime domene je unc.edu. Za to se često koristi i samo pojam domena.		
domaćin (host)	Računalo na mreži.		
čvor ( <i>node</i> )	Računalo na mreži.		
poslužitelj za imena (name server)	Računalo koje pruža DNS usluge za mapiranje DNS imena na IP adrese (DNS poslužitelj).		

Tablica 25.1 Često korišteni DNS pojmovi

Pojam	Opis
prevođenje (resolve)	Prevođenje DNS imena u njegovu odgovarajuću IP adresu.
resolver	Programska rutina koja iz biblioteke poslužitelja imena vadi DNS informacije.
obratno prevođenje (reverse resolution)	Traženje DNS imena prema zadanoj IP adresi.
prijevara (spoof)	Prijavljivanje na mreži pod lažnom IP adresom ili imenom domene.

DNS može koncepcijski biti podijeljen na sljedeća tri dijela:

- Prostor za imena domena. To je specifikacija za strukturu stabla koja identificira skup domaćina i pruža informacije o njima. Svako čvorište stabla ima bazu podataka s informacijama o domaćinima pod njegovom ovlasti. Upiti pokušavaju iz te baze podataka izvaditi odgovarajuće podatke. Pojednostavljeno, to je samo popis svih mogućih vrsta informacija imena, IP adresa, zamjenskih imena za poštu i ostalog što je raspoloživo u DNS sustavu.
- Poslužitelji za imena (DNS poslužitelji). To su programi koji čuvaju i održavaju podatke smještene u prostoru imena domene. Svaki poslužitelj imena ima cjelovitu informaciju o podskupu prostora imena domena i pohranjene informacije o drugim dijelovima.

Poslužitelj imena ima cjelovitu informaciju za područje u njegovoj ovlasti. Ta je informacija podijeljena na područja poznata kao *zone*, koja mogu biti podijeljena između različitih poslužitelja imena radi pružanja obilja servisa za zone. Svaki poslužitelj imena zna za druge poslužitelje imena koji su odgovorni za druge zone. Ako pristigne zahtjev za informacijom iz zone za koju je dotični poslužitelj imena odgovoran, poslužitelj imena naprosto vraća traženu informaciju. Međutim,

ako stigne zahtjev za informacijom iz druge zone, poslužitelj imena se obraća odgovarajućem poslužitelju koji je nadležan za tu zonu.

Resolveri. To su naprosto programske rutine koje iz biblioteke poslužitelja imena vade traženu informaciju u odazivu na upit o domaćinu u prostoru imena domene.

#### Postavljanje resolvera

Prvi je korak u uporabi DNS-a postavljanje biblioteke resolvera u vaše računalo. Ako namjeravate koristiti DNS pretvaranje imena, u vašem računalu trebate instalirati lokalni resolver čak i ako ne namjeravate pokretati lokalni DNS poslužitelj.

#### Datoteka /etc/host.conf

Lokalne resolver biblioteke se postavljaju preko datoteke host.conf, smještene u direktoriju /etc. Ta datoteka resolveru govori koje servise da koristi i kojim redom. To je čista ASCII datoteka s popisom opcija resolvera, jedna po svakom retku. Polja u ovoj datoteci mogu biti odvojena razmacima ili tabulatorima. Znak # ukazuje na početak napomene.

U datoteci host.conf može biti navedeno nekoliko opcija, kako je prikazano u tablici 25.2.

Tablica 25.2	Konfiguracijske opcije za datoteku /etc/host.conf		
Орсіја	Opis		
order	Navodi po kojem se redoslijedu različiti mehanizmi pretvaranja imena iskušavaju. Resolving servisi se iskušavaju u redoslijedu u kojem su navedeni u popisu. Podržani su sljedeći mehanizmi pretvaranja imena: hosts (pokušaji pretvaranja imena traženjem u lokalnoj datoteci /etc/hosts), bind (upiti za DNS poslužitelja imena radi pretvaranja imena) i nis (ime domaćina pokušava pronaći uporabom NIS protokola (Network Information Service).		
alert	Kao argumente uzima off ili on. Ako je uključen, zadužuje syslog za bilježenje svakog pokušaja prijave lažne IP adrese.		
nospoof	Obrnuti se postupak pronalaženja imena domaćina koje odgovara zadanoj adresi koristi kada je ime domaćina vraćeno radi dodatne provjere da odgovara adresi za koju je poslan upit. Na taj se način sprečava davanje lažnih IP adresa (spoofing). Za omogućavanje ove opcije navedite nospoof on. Upozorenje - uporaba ove opcije uzrokuje primjetno dodatno opterećenje na poslužitelju.		
trim	Kao argument uzima ime domene. trim ime domene uklanja prije postupka pretvaranja imena prema /etc/hosts. To vam omogućuje da u /etc/hosts stavite samo osnovno ime domaćina, bez navođenja imena domene.		
multi	Uzima off ili on kao argumente. Koristi se samo s host upitima pri utvrđivanju da li je domaćinu dopušteno imati više od jedne IP adrese navedene u /etc/hosts. Ova opcija ne utječe na NIS ili DNS upite.		

Slijedi primjer konfiguracijske datoteke /etc/hosts koja koristi ove opcije:

```
#Sample /etc/host.conf file
#
# Lookup names via DNS first then fall back to /etc/hosts
order bind hosts
# We don't have machines with multiple addresses
multi off
#check for IP address spoofing
nospoof on
# and warn us if someone attempts to spoof
alert on
# Trim the tristar.com domain name for host lookups
trim tristar.com
```

Ovaj primjer prikazuje općenitu postavu resolvera za domenu tristar.com. Resolver će IP adrese prema zadanim imenima domaćina najprije tražiti uporabom DNS-a i zatim probati s lokalnom datotekom /etc/hosts.

Ovaj fajl je skinut sa sajta www.tutoriali.org

**BILJEŠKA** U postupku pretvaranja imena dobro je navesti i lokalnu datoteku /etc/hosts. Ako vam zbog nekog razloga vaši poslužitelji imena ne bi bili na raspolaganju, još uvijek možete dobiti imena za domaćine koji su na popisu u vašoj lokalnoj datoteci domaćina. Također biste popis svih vaših lokalnih domaćina u vašim datotekama /etc/hosts trebali držati u svakom od vaših lokalnih računala.

Onemogućeno je da jedan stroj ima više IP adresa. Ovaj domaćin lažiranje IP adresa provjerava ponovnim prevođenjem imena domaćina prema IP adresi dobivenoj obrnutim prevođenjem. To dodatno opterećuje sustav, ali smo tako sigurniji da se nitko ne pretvara da je neki drugi domaćin nego što stvarno jest. Resolver ste također podesili da vas upozori kada otkrije pokušaj lažiranja. Na kraju, resolver uklanja domenu tristar.com iz svih imena domaćina koja se traže u lokalnoj datoteci /etc/hosts.

#### Datoteka /etc/resolv.conf

Sada kada ste podesili osnove ponašanja resolver biblioteke, trebate postaviti i neke informacije za DNS dio resolvera. To trebate uraditi samo ako za pretvaranje imena domaćina koristite DNS - to jest, ako u naredbi order datoteke /etc/host.conf navedete i bind. Ovo poglavlje vjerojatno ne biste ni čitali da DNS niste namjeravali koristiti, zar ne?

/etc/resolv.conf upravlja načinom na koji resolver koristi DNS za pretvaranje imena domaćina. Ona navodi s kojim se DNS poslužiteljima treba spojiti pri i po kojem redoslijedu treba s njima stupiti u vezu. Ona pruža i ime lokalne domene i neke ključeve po kojima se zna kako pogoditi ime domene za domaćine koji su navedeni bez imena domene.

	U	tablici 25.3	je popis	valjanih	opcija za	i daototeku	/etc/resolv.d	onf.
--	---	--------------	----------	----------	-----------	-------------	---------------	------

Tablica 25.3	Opcije za podešavanje datoteke /etc/resolv.conf
Орсіја	Opis
domain	Navodi ime lokalne domene za tog domaćina. Ako nije data, resolver će ime lokalne domene pokušati dobiti iz sustavskog poziva getdomainname().
nameserver	Navodi IP adresu DNS poslužitelja s kojim za pretvaranje imena treba stupiti u vezu. Možete navesti do tri poslužitelja imena ako opciju nameserver upotrijebite više puta. DNS poslužitelji se ispituju prema redoslijedu kojim su navedeni. Najprije navedite vaš najpouzdaniji DNS poslužitelj tako da upitu ne istekne vrijeme na poslužitelju koji možda ne radi.
search	Navodi popis domena koje treba iskušati ako u upitu uz ime domaćina nije navedeno nikakvo ime domene. Ako opcija search nije data, u popis domena će biti stavljena lokalna domena i sve njene matične domene.

Slijedi primjer datoteke /etc/resolv.conf za tristar.com:

```
# /etc/resolv.conf for tristar com
#
# Set our local domain name
```

```
domain tristar.com
# Specify our privary name server
nameserver 166.82.1.3
```

U ovom primjeru s opcijom domain navodite lokalnu domenu i za pretvaranje imena domaćina navodite jedan poslužitelj imena.

BILJEŠKA Kao argument za opciju nameserver trebate navesti IP adresu DNS poslužitelja, a ne ime domaćina. Ako navedete ime domaćina, DNS ne zna s kojim se domaćinom treba spojiti da bi u poslužitelju imena potražio ime domaćina.

U ovoj datoteci niste upotrijebili opciju search za navođenje redoslijeda pretraživanja. U tom će slučaju, kada zatražite adresu stroja, na primjer, skippy, resolver najprije pokušati potražiti skippy. Ako to ne uspije, potražit će skippy.tristar.com i zatim skippy.com.

DNS poslužitelji mogu (a to i čine) neočekivano prestati raditi. Ako se za pretvaranje imena oslanjate isključivo na DNS poslužitelj, u slučaju pada njegovog sustava vi nećete moći raditi. Zbog toga, za svaki slučaj, navedite više poslužitelja i u vašoj datoteci /etc/hosts držite dobar popis domaćina.

#### Uporaba demona *named* za postavljanje poslužitelja

Ovdje tek počinje prava čarolija. Saznali ste kako postaviti osnove konfiguracije resolvera i kako vašem resolveru reći s kojim se poslužiteljima imena treba povezati. U sljedećim ćete člancima učiti tehniku postavljanja poslužitelja imena.

Za pribavljanje je DNS poslužitelja u Linuxu zadužen *daemon* named (name-d). Taj se daemon obično pokreće pri podizanju sustava i potrebne informacije učitava iz skupa konfiguracijskih datoteka. named je obično u radu sve do isključivanja stroja. Kada je named pokrenut i inicijaliziran svojim konfiguracijskim informacijama, on ID svog procesa upisuje u ASCII datoteku /etc/named.pid. Zatim počinje osluškivati DNS zahtjeve na unaprijed zadanom mrežnom sučelju navedenom u /etc/services.

#### Datoteka named.boot

Prva je datoteka koju named učitava pri pokretanju obično /etc/named.boot. Ova je vrlo mala datoteka ključ do svih drugih konfiguracijskih datoteka koje named koristi. Ona sadrži smjernice do raznih konfiguracijskih datoteka i do drugih poslužitelja imena. U datoteci named.boot napomene počinju s točkazarezom i nastavljaju se do kraja retka. U njoj može biti navedeno nekoliko opcija, koje su opisane u tablici 25.4.

lablica 25.4	Konfiguracijske opcije za datoteku named.boot
Opcija	Opis
directory	Navodi direktorij u kojem su smještene datoteke DNS zone. Ponovljenom uporabom opcije directory možete navesti nekoliko različitih direktorija. Možete navesti i imena staza do datoteka koje se odnose na te direktorije.

Opcija	Opis
primary	Kao argumente uzima ime domene i ime datoteke. Opcija primary deklarira da named ima ovlasti za navedenu domenu i potiče named da iz navedene datoteke učita informacije s tog područja.
secondary	Nalaže namedu da djeluje kao sekundarni poslužitelj za određenu domenu. Kao argumente uzima ime domene, popis adresa i ime datoteke. named informacije o zoni pokušava prenijeti iz domaćina navedenih u popisu adresa i zatim ih sprema u datoteku navedenu u retku ove opcije. Ako se named ne uspije spojiti ni s jednim od domaćina, informacije će pokušati pribaviti iz sekundarne datoteke zone.
cache	Određuje spremljene informacije za named. Kao argumente uzima ime domene i ime datoteke. Ime domene je obično navedeno kao . (točka). Datoteka sadrži skup zapisa, poznatih kao <i>server hints (natuknice za poslužitelj)</i> , koji navode informacije o <i>root</i> poslužiteljima imena.
forwarders	Kao argumente uzima popis poslužitelja imena. Lokalnom poslužitelju imena naređuje da se spoji s poslužiteljima u tom popisu ako adresu koju treba pretvoriti ne pronađe u svojim lokalnim podacima.
slave	Pretvara lokalni poslužitelj imena u podređeni ( <i>slave</i> ) poslužitelj. Ako je slave opcija stavljena, lokalni će poslužitelj DNS imena pokušati prevesti pomoću povratnih (rekurzivnih) upita. On će upit naprosto proslijediti jednom od poslužitelja navedenih u retku opcije forwarders.

Pored ovih opcija često se koriste i neke dodatne opcije. O više informacija o ovim opcijama pogledajte *man* stranicu za named.

BILJEŠKA Budući da tristar.com nije priključen na Internet, mnoge su od IP adresa domaćina i mreža u ovim primjerima lažne. Pri podešavanju vašeg vlastitog poslužitelja imena provjerite koristite li valjane adrese koje su vam dodijeljene.

Slijedi primjer datoteke named.boot:

```
; named.boot file
;, a sample named.boot for tristar.com
;
directory /var/named
;
cache . named.ca
primary tristar.com named.hosts
primary 197.198.199.in-addr.arpa named.rev
```

U ovom se primjeru određuje primarni poslužitelj imena za tristar.com. Kao što možete vidjeti, napomene počinju s točka zarezom (;). Naredba directory u ovoj datoteci namedu govori da su sve radne datoteke smještene u direktoriju /var/named. Budući da ni jednoj od drugih datoteka navedenih u datoteci named.boot nije pridružena staza direktorija, one su smještene u /var/named.

Sljedeći redak određuje spremljene informacije za ovaj poslužitelj imena. Ova opcija treba biti prisutna gotovo na svakom stroju koji služi kao poslužitelj imena. Ona namedu naređuje da omogući spremanje i učitavanje informacija o root poslužitelju iz datoteke named.ca.

Stavka cache je vrlo važna. Bez nje na lokalnom poslužitelju nije omogućeno privremeno spreman-BILJEŠKA je. To može ozbiljno smanjiti učinak pri traženju imena. Također, lokalni poslužitelj ne može uspostaviti vezu ni s jednim root poslužiteljem imena, zbog čega ne može prevoditi imena udaljenih domaćina, osim ako je podešen za prosljeđivanje imena drugim poslužiteljima.

Sljedeći redak u datoteci named.boot namedu kaže da ovaj poslužitelj ima prvenstvenu ovlast za domenu tristar.com. Zapisi se informacija o zonama i domaćinima nalaze u datoteci named.hosts. U sljedećem ćete članku saznati pojedinosti o tim zapisima ovlastima nad zonama.

Drugi redak primary u datoteci named.hosts pokazuje da imate prvenstvenu zonsku ovlast i za zonu 197.198.199.in.addr.arpa s informacijama o zoni u datoteci named.rev. Ova je čudna sintaksa način na koji named pribavlja informacije za sparivanje IP adresa s DNS imenima. Budući da je DNS prvobitno određen za pronalaženje IP adresa prema zadanim DNS imenima, za obratno je prevođenje potreban drukčiji redak primary.

Domena in-addr.arpa je upotrebljena za određivanje obratnog prevođenja (IP adrese BILJEŠKA u DNS ime).

#### Datoteke baza podataka i zapisi resursa

Sve su informacije u raznim datotekama named baza podataka pohranjene u formatu poznatom kao resource record (zapis resursa). Svakom je zapisu resursa pridružena vrsta (type) koja ukazuje na namjenu zapisa. Zapis resursa je najmanja količina informacija koju named koristi.

Većina ljudi sintaksu za zapise resursa i glavne (master) datoteke baze podataka općenito smatra pomalo tajnovitom i nejasnom. Pri tom ne pomaže ni činjenica da neki zapisi resursa trebaju biti prisutni na određenim mjestima u određenim datotekama. Nakon ovog, vrijeme je da se u to zaroni i prouči sintaksa za zapise resursa i razne master datoteke.

Unutar master konfiguracijskih datoteka možete navesti apsolutna imena domaćina

BILJEŠKA ili imena domaćina relativna prema toj domeni. Imena domaćina se apsolutnim smatraju ako završavaju s točkom (.), kao u foo.tristar.com.. Imena domaćina koja ne završavaju s točkom se smatraju relativnima prema lokalnoj domeni, poznatoj i kao origin (ishodište). Na samo ishodište možete upućivati uporabom znaka @.

U zapisima resursa se koristi opća sintaksa koja se dosljedno koristi u svim vrstama zapisa resursa. Međutim, da bi se tu ipak uvela zbrka, neki dijelovi zapisa nisu obavezni, ovisno o vrsti zapisa, i ako vrijednost nije navedena mogu pretpostaviti unaprijed zadanu vrijednost. Osnovni je format zapisa resursa

[owner] [ttl] [class] type data

Polja se odvajaju bijelim razmakom - tipkom razmaknicom ili tabulatorom. Tablica 25.5 opisuje značenje različitih polja.

Tablica 25.5	Polja u formatu podataka zapisa resursa.
Polje	Opis
owner	Domena ili ime domaćina na koje se zapis odnosi. Ako ime nije dato, pretpostavlja se ime domene iz prethodnog zapisa.
ttl	Polje time-to-live, koje ukazuje koliko je sekundi informacija u zapisu valjana nakon što je zaprimljena iz DNS poslužitelja. Ako ttl vrijednost nije navedena, koristi se namjanji ttl iz posljednjeg Start of Authority (SOA) zapisa.
class	Navodi klasu mrežne adrese. Za TCP/IP mreže koristiti vrijednost IN. Ako klasa nije data, koristi se ona iz prethodnog zapisa resursa.
type	Navodi vrstu zapisa resursa. Ova je vrijednost obavezna. Različite su vrste zapisa resursa opisane u sljedećoj tablici.
data	Navodi podatke koji su pridruženi ovom zapisu resursa. Ova je vrijednost obavezna. Format polja data ovisi o sadržaju polja type.

Kao što možete vidjeti, format zapisa resursa može postati prilično zbunjujući. Postoji nekoliko neobaveznih polja, a polje data ovisi o vrsti zapisa resursa. Stanje dodatno pogoršava postojanje nekoliko različitih vrsta zapisa resursa. Tablica 25.5 navodi najčešće vrste zapisa resursa. Nekoliko se drugih vrsta rijetko koriste. Ako vas zanimaju i druge vrste, potražite odgovarajuće RFC-ove i *man* stranice za named.

Tablica 25.6	Često korištene vrste zapisa resursa
Polje	Opis
A	Zapis adrese koji imenu domaćina pridružuje adresu. Podatkovno polje sadrži adresu u decimalnom obliku s točkama. Za svaki navedeni domaćin može postojati samo jedan, A zapis, budući da se taj zapis smatra autoritativnom (službenom) informacijom. Svako dodatno ime domaćina ili mapiranja adrese trebaju biti dodana uporabom vrste CNAME.
CNAME	Kanonskom (standardnom) imenu domaćina (koje je navedeno u A zapisu za tog domaćina) pridružuje zamjensko ime.
HINFO	Pruža informacije o domaćinu. Podatkovno polje sadrži informacije o hardveru i softveru za određenog domaćina. To je samo tekstualni niz znakova slobodnog formata pa u njega možete staviti sve što ima smisla o vašem hardveru.
МХ	(mail exchanger) - Postavlja zapis za sustav razmjene pošte. Podatkovno polje sadrži cjelobrojnu odabranu vrijednost nakon koje slijedi ime domaćina. MX zapisi sustavu za prijenos pošte govore da poštu pošalje u drugi sustav koji zna kako je isporučiti na njeno krajnje odredište.

nastavlja se

Tablica 25.6	6 Nastavak
Polje	Opis
NS	( <i>name server</i> ) - ukazuje na poslužitelja imena za drugu zonu. Podatkovno polje NS resursa sadrži DNS ime poslužitelja imena. Trebate svakako navesti i, A zapis radi sparivanja imena domaćina s adresom poslužitelja imena.
PTR	Mapira adrese na imena, kao u domeni in-addr.arpa. Ime domaćina mora biti kanonsko ime domaćina.
SOA	( <i>start of authority</i> ) - poslužitelju imena kaže da su svi sljedeći zapisi resursa službeno važeći za ovu domenu. Podatkovno je polje zatvoreno zagradama i obično se sastoji od više redaka. Podatkovno polje SOA zapisa sadrži sljedeće stavke:
	origin - kanonsko ime primarnog poslužitelja imena za tu domenu. Obično je dato apsolutno ime domene koje završava s točkom (.), dakle nije preinačeno s daemonom named.
	Contact - e-mail veza s osobom odgovornom za održavanje ove domene. Budući da znak @ u zapisima resursa ima posebno značenje, ovdje je zamijenjen s točkom (.). Ako je ime osobe odgovorne za održavanje zonske informacije o tristar.com Dave, tada je adresa za vezu s njim dave.tristar.com.
	serial - broj inačice datoteke informacija o zoni, koji se stavlja kao cijeli broj. Njega koriste sekundarni poslužitelji imena za utvrđivanje kada je datoteka s informacijama o zoni promijenjena. Ovaj biste broj svaki put kada datoteku s informacijama promijenite trebali povećati za 1.
	refresh - vremensko trajanje u sekundama koliko bi sekundarni poslužitelj trebao čekati prije nego pokuša provjeriti SOA zapis primarnog poslužitelja imena. SOA zapisi se ne mijenjaju često pa biste ovu vrijednost trebali otprilike podesiti na jedan dan (84600).
	retry - vrijeme u sekundama koliko sekundarni poslužitelj čeka prije ponovnog pokušaja slanja zahtjeva primarnom poslužitelju ako primarni poslužitelj nije bio raspoloživ. To bi obično trebalo biti nekoliko minuta.
	expire - vrijeme u sekundama koliko bi sekundarni poslužitelj trebao čekati prije odbacivanja zonske informacije ako nije uspio stupiti u vezu s primarnim poslužiteljom. Taj bi broj u pravilu trebao biti vrlo velik - otprilike 30 dana.
	minimum - Unaprijed zadana ttl vrijednost za zapise resursa koji ne navode ttl. Ako se vaša mreža ne mijenja mnogo, taj se broj može podesiti na prilično veliku vrijednost, poput nekoliko tjedana. To uvijek možete premostiti navođenjem ttl vrijednosti u vašim zapisima resursa.

Kao što možete vidjeti, format zapisa resursa u žurbi može postati zamršen. Stvari bi vam trebale postati jasnije uz nekoliko primjera master konfiguracijskih datoteka koje koristi named.

#### Datoteka named.hosts

U vašoj ste datoteci named.hosts naveli named.hosts kao datoteku koja sadrži informacije o vašoj lokalnoj domeni, tristar.com. Tu datoteku možete nazvati kako god želite s tim da isto ime stavite u primary redak datoteke named.boot. Datoteka named.hosts sadrži službene informacije o domaćinima u zoni ovlasti - tristar.com. U ispisu 25.1 je prikazana ogledna datoteka named.hosts koja koristi nekoliko vrsta zapisa resursa.

```
Ispis 25.1
              Primjer datoteka named.hosts
 named.hosts file for tristar.com
0
           ΤN
                  SOA
                         ns.tristar.com. dave.tristar.com (
6 : serial number
86400 ;refresh 24 hrs
300 ; retry 5 minutes
2592000 ; expire 30 days
86400 ; minimum 24 hrs
)
ΙN
       NS
               ns.tristar.com.
; your domain itself tristar.com
0
           ΙN
                  А
                       199.198.197.1
       МΧ
              100 mailhost.tristar.com
ΤN
ΙN
       HINFO
               PC-486
                         linux
;
; your primary nameserver
;
            ΤN
                         199.198.197.1
ns
                   А
             ΙN
                      CNAME
nameserver
                            ns.tristar.com.
;
; other hosts
mailhosts
            ΙN
                   А
                        199.198.197.2
opus
            ΤN
                   Α
                        199.198.197.3
    МХ
                  mailhost.tristar.com
ΙN
            100
skippy
            ΙN
                  A 199.198.1971.4
   MX
            100
ΤN
                  mailhost.tristar.com
; the localhost
                          127.0.0.1
Localhost
              ΙN
                     А
```

BILJEŠKA Imena domaćina u zapisima resursa koja završavaju s . (točkom) se dalje ne prevode.

Ako posljednji znak u imenu domaćina nije točka, named pretpostavlja da se ime domaćina koje ste dali odnosi na ishodišno ime domene na koje upućuje @ i ime domene pripaja imenu domaćina.

Pogledajte pojedinosti datoteke named.hosts u ispisu 25.1. Prvi je zapis na koji ste u ovoj datoteci naišli SOA (start of authority) zapis za oglednu domenu. Prvi redak ovog zapisa počinje znakom @, koji ukazuje na trenutno porijeklo ili domenu (tristar.com). Definicija

#### 512 25. poglavlje Podešavanje DNS-a

porijekla dolazi iz domene navedene u odgovarajućem primary retku u named.boot. Nakon toga vidite kodove IN SOA, koji namedu kažu da ovaj zapis resursa koristi Internet (TCP/IP) adresiranje.

Sljedeće su dvije stavke u retku kanonsko ime primarnog poslužitelja imena za tu domenu (ns.tristar.com) i e-mail veza u kojoj je @ zamijenjen s točkom (dave.tristar.com). Zatim navodite različita polja podataka potrebnih za SOA zapis, jedno po retku. (Za cjelovito objašnjenje svake od ovih stavki pogledajte tablicu 25.6).

Nakon SOA zapisa sljedeći je redak zapis resursa poslužitelja imena, koji kao poslužitelja imena za ovu domenu navodi ns.tristar.com. Budući da je polje za domenu ostalo prazno, pretpostavlja se posljednja navedena domena, a to je bila @, navedena u SOA zapisu. I, naravno, znak @ stvarno obuhvaća i lokalnu domenu, tristar.com. Može li nešto biti jednostavnije razumijeti?

Sljedeća tri retka postavljaju neke informacije o samoj domeni tristar.com. Premda ste ime domene radi jasnoće naveli kao @, budući da je to bilo posljednje ime domene navedeno u datoteci, ovi se zapisi resursa automatski primjenjuju na nju ako ste polje domene ostavili prazno. Redak

@ IN A 199.198.197.1

korisniku omogućuje da se prema tristar.com odnosi kao da je to stvarni stroj. Dodijeljena joj je IP adresa 199.198.197.1, što je, kako ćete vidjeti, stvarno IP adresa za ns.tristar.com. Sljedeći redak postavlja MX (*mail exchanger*) zapis za tristar.com, tako da sva pošta koja na nju dolazi bude proslijeđena na mailhost.tristar.com. Posljednji redak u ovoj grupi postavlja HINFO (*host information*) - zapis informacije o domaćinu za tristar.com, koji svijetu govori da je to PC-486 s operativnim sustavom Linux.

Nekoliko ste redaka ranije u datoteci preko NS zapisa kao vaš poslužitelj imena naveli ns.tristar.com. Da bi named ispravno radio trebate mu pribaviti adresu ili, A zapis koji daje adresu za ns.tristar.com. Sljedeći redak u vašoj datoteci upravo to čini. Nakon "priljepivog zapisa" koji daje adresu poslužitelja imena, imate CNAME zapis resursa. Taj vam zapis kaže da je nameserver.tristar.com drugo ime za ns.tristar.com.

Zatim nastavljate stavljati zapise resursa za tri druga domaćina u vašoj domeni: mailhost, opus i skippy. Uočite da su nakon, A zapisa za opus i skippy stavljeni MX zapisi koji svu poštu zaprimljenu na opusu ili skippyju usmjeravaju na mailhost.tristar.com. Budući da u prvom polju tih MX zapisa nije navedeno nikakvo ime, oni se odnose na prethodna imena - opus i skippy.

**BILJEŠKA** Budući da se u polju vlasnika, ako je ostavljeno prazno, pretpostavlja da se odnosi na posljednje navedeno ime, lako je grupirati zapise koji se primjenjuju na jednog domaćina. Međutim, trebate biti oprezni ako u datoteku dodajete nove zapise za novog domaćina. Ako ih dodate u sredinu datoteke, možete prouzročiti da se za neke postojeće zapise resursa promijeni pretpostavljeni domaćin. Zato dobro pripazite gdje ćete u postojeću datoteku dodavati nove zapise resursa.

Na kraju, posljednji domaćin u ovoj datoteci named.hosts je localhost, koji je mapiran na adresu 127.0.0.1. Kao što možete vidjeti, sintaksa za te datoteke postaje prilično zamršena i pruža vam obilje mogućnosti za pogreške.

#### Datoteka named.rev

Datoteka named.rev je vrlo slična datoteci named.hosts, osim što u biti radi obratno - adrese mapira na imena domaćina. Ispis 25.2 prikazuje oglednu datoteku named.rev za tristar.com.

```
Ispis 25.2
               Ogledna datoteka named.rev
 named.rev file for tristar.com
0
          ΤN
                   SOA
                            ns.tristar.com. dave.tristar.com. (
6 ; serial number
86400 ; refresh 24 hrs
300 ; retry 5 minutes
2592000 ; expire 30 days
86400 ; minimum 24 hrs
)
ΤN
      NS
            ns.tristar.com.
  reverse map your IP addresses
:
                   PTR
1
          ΙN
                            ns.tristar.com.
2
          ΤN
                   PTR
                             mailhost.tristar.com.
                   PTR
3
          ΤN
                             opus.tristar.com.
Δ
          ΙN
                   PTR
                             skippy.tristar.com.
```

U ovom primjeru imate isti SOA zapis koji ste vidjeli u datoteci named.hosts. On samo postavlja službenu informaciju za domenu. U ovom je slučaju @ - vrijednost porijekla, podešena na 197.198.199.in-addr.arpa iz retka primary u datoteci named.boot. Prisjetite se da domena in-addr.arpa upućuje na obratno mapiranje adresa na imena.

BILJEŠKA Adresa navedena kao dio vašeg retka in-addr.arpa je vaša preokrenuta mrežna adresa.

Vaša ogledna mreža za ovo poglavlje ima adresu 199.198.197.0. Kada ju navodite u datotekama obratnog mapiranja, navodite ju kao

1997.1998.199.in-addr.arpa

Ovdje imate NS zapis koji navodi poslužitelj imena za vašu domenu. Nakon njega slijede zapisi za obratno prevođenje adresa. To su PTR zapisi koji daju broj domaćina (dio IP adrese koji nije naveden u vrijednosti in-addr.arpa) i kanonsko ime domaćina koje je njegov par. Ovdje trebate koristiti kanonsko umjesto relativnog imena domaćina. Na primjer, redak

2 IN PTR mailhost.tristar.com.

namedu govori da adresu domaćina 199.198.197.2 mapira na ime domaćina mailhost.tristar.com.

#### Datoteka named.ca

Kako je prije u ovom poglavlju navedeno, *cache* operacija koju provodi named ima veliku važnost. Na sreću, datoteka named.ca koja određuje privremeno spremanje je obično i najjednostavnija od named konfiguracijskih datoteka. Ona samo navodi *root* poslužitelje imena za različite domene s njihovim IP adresama. Ona sadrži nekoliko posebnih pokazatelja polja koji namedu govore da su to *root* poslužitelji.

Vjerojatno možete naprosto kopirati format ogledne datoteke named.ca u ispisu 25.3. Za dobivanje cjelovitog trenutnog popisa *root* poslužitelja imena koristite uslužni program nslookup.

Ispis 25.3	Primje	er datoteke na	amed.ca			
; named.ca f	file					
. 99999	999	IN NS	NS.NI	C.DDN.M	IL.	
99999999	ΙN	NS	NS.NASA.GO	ν.		
99999999	ΙN	NS	KAVA.NISC.	SRI.COM.		
99999999	ΙN	NS	TERP.UMD.E	DU.		
99999999	ΙN	NS	C.NYSER.NE	Τ.		
99999999	ΙN	NS	NS.INTERNI	C.NET.		
;						
NS.NIC.DDN.M	1IL.	99999999	ΙN	А	192.112.36.4	
NS.NASA.GOV.		99999999	9 I N	А	128.102.16.10	
KAVA.NISC.SF	RI.COM.	9999999	9 I N	А	192.33.33.24	
TERP.UMD.EDL	J.	9999999	99 IN	A	128.8.10.90	
C.NYSER.NET.		99999999	9 I N	А	192.33.4.12	
NS.INTERNIC.	NET.	99999999	ΙN	А	198.41.0.4	

Kao što možete vidjeti, datoteka named.ca naprosto NS zapise poslužitelja imena mapira na odgovarajuće adrese za njih.

#### Rješavanje problema

DNS je vrlo složen sustav. Tu mnogo toga možete pogrešno napraviti, što će uzrokvoati da se vaš sustav ne ponaša ispravno. Mnoge će od poteškoća koje se pojavljuju pri postavljanju DNS-a izgledati isto, ali će se pojavljivati zbog različitih uzroka. Većina je poteškoća ipak posljedica grešaka u sintaksi vaših konfiguracijskih datoteka.

Provjerite da li su imena domaćina koja navodite u vašim DNS konfiguracijskim datotekama ispravna. Ako je ime domaćina apsolutno, ono mora završavati s točkom.

Naročito budite pažljivi s imenima koja se koriste u SOA i CNAME zapisima. Ako tu napravite pogrešku, ti zapisi resursa mogu upite za imena domaćina preusmjeriti na računala koja ne postoje.

Pri uvođenju promjena u vaše konfiguracijske datoteke svakako povećajte serijski broj. Ako to zaboravite, DNS neće ponovno učitati tu datoteku. Provjerite da li ste unijeli ispravne IP adrese za, a zapise i provjerite da li se ono slaže s vašom datotekom /etc/hosts (ako ju imate). Također provjerite da se DNS ime i IP adresa slažu s odgovarajućom informacijom za obratno prevođenje u datoteci named.rev.

Vaš je najbolji alat za otkrivanje grešaka naredba nslookup. Pomoću nslookupa možete temeljito ispitati vaš DNS poslužitelj. Napravite obično i obratno prevođenje za svaku adresu u vašoj DNS bazi podataka i tako provjerite da su sva imena i adrese ispravne.

BILJEŠKA Paul Vixie (koji je napisao BIND) je osmislio zanimljiv projekt, S/WAN, koji pruža prikladno šifriranje na razini Ethernet paketa. Više informacija potražite na http://www.cygnus.com/čgnu/swan.html.

#### Odavde...

Ovo poglavlje pokazuje različite dijelove DNS sustava i istražuje različite konfiguracijske datoteke neophodne za rad DNS poslužitelja imena na vašem Linux sustavu. Budući da je sintaksa za zapise resursa prilično nejasna, pri pisanju konfiguracijskih datoteka trebate na to dobro pripaziti.

Više informacija o umrežavanju možete naći u sljedećim poglavljima:

- 24. poglavlje, "Podešavanje TCP/IP mreže" pokazuje kako postaviti i podesiti TCP/IP mrežu.
- 29. poglavlje, "Uporaba SLIP-a i PPP-a" pokazuje kako SLIP i PPP podesiti za pristup na Internet preko birane veze.
- 31. poglavlje, "Kretanje Internetom pomoću World Wide Weba" pruža pregled Interneta.

# poglavlje26

## Podešavanje elektroničke pošte

(Steve Burnett)

U ovom poglavlju

Pregled elektroničke pošte **518** sendmail **524** 

U ovom će se poglavlju najprije govoriti o općim pitanjima elektroničke pošte i općenito o Internetu - opće zamisli i definicije, poštanski standardi kako su definirani u RFC-ovima (Request for Comment) i nekoliko protokola definiranih za uporabu u prometu elektroničkom poštom. Poglavlje zatim opisuje sendmail, UNIX-temeljeni podsustav u najširoj uporabi na Internetu.

### Pregled elektroničke pošte

U ovom je članku pružen širok pregled sustava slanja i primanja elektroničke pošte. Najprije će se raspraviti neki opći principi elektroničke pošte, uključujući dvije temeljne vrste poštanskog softvera i gdje sendmail u toj podjeli pripada. U sljedećem su dijelu ovog članka predstavljeni RFC-ovi (Request for Comment), u kojima su definirani protokoli korišteni za komuniciranje unutar i preko mreža. U posljednjem se dijelu ovog članka objašnjavaju neki od protokola koji se koriste za definiranje elektroničkih poruka.

#### Povijest i opći principi

Jedan je od prvih naširoko korištenih uredskih poštaniskih sustava bio IBM-ov PROFS - sustav koji se temeljio na mainframe računalima. Imao je mogućnosti slične suvremenim sustavima poput Microsoft Exchangea i Lotus Notesa, uključujući:

- Jake alate za administriranje i upravljanje
- Mogućnost prilagođavanja sigurnosti
- Sposobnost vremenskog raspoređivanja zadataka

PROFS i drugi sustavi za baratanje poštom iz tog vremena dijelili su neke sličnosti. Bilo da su namijenjeni za *mainframe* ili UNIX, temeljili su se na tekstu i smatrani su centraliziranim sustavima rada s e-poštom za računala domaćine. Budući da je PROFS bio proširiv i prilagodljiv, IBM je za rad s poštom u svojoj rasprostranjenoj i raznovrsnoj mreži tek nedavno prešao na drugi sustav.

Kako su osobna računala postajala sve prihvatljivija i počela se naširoko koristiti u korporacijama, ljudi su počeli koristiti prednosti iz prijelaza računalne snage s *mainframe* na stolna računala. Među prvim primjenama mreže osobnih računala bilo je dijeljenje datoteka, koje je koristilo središnji poslužitelj datoteka i dijeljeni mrežni diskovni pogon kojem se slobodno pristupalo. Uskoro nakon toga sustav za rad s poštom je počeo koristiti prednosti nove snage na korisničkim stolnim računalima. Tako se rad s poštom temeljen na domaćinima u nekim slučajevima prebacio na LAN-temeljeni rad s poštom.

#### Model rada s e-poštom nad dijeljenom datotekom

cc:Mail je primjer na mreži temeljenog rada s poštom, koje se naziva i *shared-file messaging* (rad s e-poštom nad dijeljenom datotekom). U ovom modelu rada stolno računalo - klijent ima svu moć i svo upravljanje. Klijent poruke šalje u poštanski sandučić u poslužitelju i svoju poštu učitava iz svog zadanog poštanskog sandučića (direktorija) u poslužitelju. Poslužitelj je pasivan, samo pohranjuje poruke. On ne izvršava obradu ili razvrstavanje i nema ovlasti za postavljanje pravila za upravljanje protokom poruka. Rad s poštom dijeljenjem datoteke pruža sljedeće prednosti nad upravljanjem porukama temeljenom na domaćinima:

- Mogućnost dodavanja datoteka čisto tekstualnim porukama
- Potrebni su poslužitelji niže cijene
- Pojednostavljeno podešavanje
- Poboljšan učinak za neke radnje klijenta

Međutim, sustavi rada s poštom dijeljenjem datoteka su doveli do nekih novih problema. Budući da je svaki korisnik trebao puni pristup datotečnom sustavu, uključujući i poštanske sandučiće drugih korisnika, sigurnost je bila upitna. Također, zbog toga što je svaki klijent trebao u poslužitelju provjeravati ima li nove pošte, povećavao se mrežni promet. Mrežna propusnost se češće javlja kao usko grlo nego što je to propusnost poslužitelja ili klijenta.

#### Model rada s e-poštom klijent/poslužitelj

Sustav rada s poštom klijent/poslužitelj podijelio je zadaće u obradi poruka između radnih stanica i posljužitelja. Uporabom modela guranja poruka poštanski klijenti mrežu više nisu preopterećivali neprestanim provjeravanjem za nove poruke. Klijent/poslužitelj upravljanje poštom je poboljšanjem sigurnosti poboljšalo i rad s prvim modelom, tako da bi korisnici teže uspijevali u pokušaju čitanja tuđe pošte. Inteligentniji je poslužitelj omogućio da se razvrstavanje i obrada poruka izvrše prije neko se poruke klijentu pošalju preko mreže.

#### MUA, MTA i MDA

Sustav elektroničke pošte može biti podijeljen na tri elementa: Mail User Agent (MUA), Mail Transport Agent (MTA) i Mail Delivery Agent (MDA).

MUA je korisničko sučelje - softver s kojim korisnik čita svoju poštu, organizira je u direktorije ili mape i šalje poštu. Različite osobe u svojim MUA-ovima žele različite mogućnosti, a nisu ni svi MUA-ovi raspoloživi na svim platformama. Mnogi MUA-ovi mogu istodobno zajedno s drugima postojati na istom stroju. Na primjer, UNIX radna stanica može imati bilo koji ili sve sljedeće MUA-ove raspoložive za uporabu: mailx, elm, pine, mutt, mailtool i dtmail. Korisnik može koristiti bilo koji MUA prisutan na njegovom sustavu budući da su MUA-ovi naprosto lokalne aplikacije. Pored toga MUA mogućnosti su često uključene u višenamjenski softver poput Lotus Notesa i Netscape Maila.

MTA se ne koristi za pisanje poštanske poruke, nego za usmjeravanje pošte iz lokalnog MUA na drugi MTA na drugom sustavu. (sendmail je primjer MTA, koji se ne koristi za izravno čitanje ili pisanje pošte, već je namijenjen samo za ispuručivanje predformatiranih poruka.) Usmjeravanje se pošte može odvijati i lokalno i udaljeno. MTA je u lokalnom prijenosu pošte, gdje i pošiljaoc i odredište imaju račune na istom stroju, odgovoran za prijenos pošte od sebe do lokalnog MDA. U tom je postupku moguće da MTA uređuje protokole, adrese i smjer poštanske poruke. Poruka stvorena na UUCP mreži zahtijeva neke promjene prije nego ju može primiti osoba na TCP/IP mreži. MTA djeluje kao prolaz za poštu pri slanju poruke s jedne mreže na drugu mrežu koja koristi drukčije protokole. U velikoj će većini slučajeva na jednom stroju biti samo jedan MTA.

MDA je treći sastavni dio rutine baratanja poštom. Dok sendmail izravno barata SMPT prijenosom pošte između MTA-ova, u baratanju se lokalnom isporukom iz sendmailovog

reda čekanja u red čekanja koji koristi MUA oslanja na Mail Delivery Agente (MDA). Dva uobičajena MDA-ova za čiju je uporabu sendmail često podešen su /bin/mail i procmail. /bin/mail je gotovo sveprisutan na UNIX sustavima. procmail je naširoko raspoloživ i istovremeno je brži i mnogo sposobniji od uobičajenog /bin/maila, pružajući jake mogućnosti predrazvrstavanja i predobrade pošte.

Za bolje razumijevanje MUA/MTA/MDA odnosa razmotrite primjer iz stvarnog života kada neka osoba pošalje pismo. MUA predstavlja osobu koja šalje pismo. On pismo napiše, stavi ga u omotnicu, napiše adresu i zalijepi marku i zatim ga dostavi u poštanski ured. MTA je poput osoblja poštanskog ureda - oni pismo zaprime, provjere adresu i ako je potrebno preformatiraju je i zatim pismo usmjere ili u poštanski sandučić u istom

poštanskom uredu (ako je pismo lokalno) ili u drugi poštanski ured (za udaljena odredišta). MDA odgovara poštaru koji poštu iz poštanskog ureda isporučuje na namijenjenu lokaciju. Ako je korišten prolaz, ta se sličnost može proširiti - MTA koji zaprimi pismo za odredište u drugoj državi treba tu poruku prenijeti drugom MTA-u koji zna kako pismo isporučiti u ciljnoj državi.

#### **IETF Request for Comment**

*RFC* je formalni opis formata protokola korištenih na Internetu. Tih se protokola drže i mnogi drugi ne-Internet sustavi. Te RFC-ove (Request for Comment) izdaje Internet Engineering Task Force (IETF). RFC-ovi su radi jasnoće označeni brojevima prema kojima se mogu zatražiti - lakše je uputiti na RFC822 nego na "Standard for the Format of ARPA Internet Text Messages". U trenutku pisanja ovog članka postoji više od dvije tisuće RFC-ova, od kojih su neki već zastarjeli izdavanjem novih RFC-ova. Za pronalaženje određenog RFC-a pogledajte stranice IETF-a na World Wide Web adresi **http://www.ietf.org**.

Budući da je pošta jedna od tako često korištenih namjena Interneta, mnogi od RFC-ova određuju standarde za razmjenu pošte. sendmail i drugi MTA-ovi ispunjavaju uvjete i definicije mnogih od tih protokola. Međutim, pokušaj bi detaljnog opisivanja svih RFC-ova od važnosti za prijenos i format pošte mogao potrajati više godina i zauzeo bi tisuće stranica.

Tablica 26.1	RFC-ovi koji se bave sustavo	m rada s elektronskom postom
Broj	Naslov	Napomena
RFC819	Domain Naming	Sadrži konvencije o korisničkim aplikacijama za Internet.
RFC821	Simple Mail Transfer Protocol	Definira SMTP.
RFC822	Standard for the Format of ARPA Internet Text Messages	Definira format (zaglavlja, tijelo i kako ih odvojiti) tekstualnih poštanskih poruka na Internetu.
RFC976	UUCP Mail Interchange Format Standard	Definira UNIX-to-UNIX-Copy-Protocol (UUCP) format poštanskih poruka između dva UNIX sustava.

Tablica 26.1 u kronološkom redoslijedu predstavlja RFC-ove od važnosti za sendmail.

Broj	Naslov	Napomena
RFC1123	Requirements for Internet Hosts - Application and Support	Proširuje i obnavlja RFC822, uglavnom pojašnjavajući dvojbena pitanja u izvornom dokumentu.
RFC1327	Mapping between X.400 (1988) / ISO 10021 and RFC822	Obnavlja RFC822.
RFC1521 i RFC 1522	MIME (Multipurpose Internet Mail Extensions) Parts One and Two	Definiranjem MIME- a pruža dodatno proširenje formata za poštu definiranog uRFC822. MIME,među ostalim, omogućuje umetanje binarnih datoteka (poput grafike i zvuka) u poštanske poruke.RFC2045-2049 je ova dva učinio zastarjelima.
RFC1651	SMTP Service Extensions	Predstavlja ESTMP (Extended Simple Mail Transfer Protocol).
RFC1652	SMTP Service Extension for 8-bit MIME Transport	
RFC1653	MTP Service Extension for Message	
RFC1869	SMTP Service Extensions	Čini RFC1651 zastarjelim.
RFC1870	SMTP Service Extension for Message Size Declaration	Čini RFC1653 zastarjelim.
RFC1891	SMTP Service Extension for Delivery Status Notifications	
RFC1892	The Multipart/Report Content Type for the Reporting of Mail System Administrative Messages	
RFC1893	Enhanced Mail System Status Codes	
RFC1894	An Extensible Message Format for Delivery Status Notifications	
RFC2045-2049	Multipurpose Internet Mail Extensions (MIME) Parts One through Five	Čine RFC1521 i RFC1522 zastarjelima.

#### Internet protokoli

sendmail za premještanje poruka između dva poslužitelja za poštu koristi *Simple Mail Transfer Protocol* (SMTP). Djelujući kao protokol za rad s poštom između dva poslužitelja, SMTP-u za lokalno prikupljanje i obradu poruka te isporuku poruka određenim

korisnicima treba drugi protokol poput POP3. SMTP je komunikacijski protokol koji se uglavnom koristi u UNIX-temeljenim mrežama za poštu preko TCP/IP (Transmission Control Protocol/Internet Protocol) veza. Za razliku od UUCP protokola koji mora imati "kartu" s ubilježenim strojevima koji postoje između pošiljatelja i odredišta, TCP/IP omogućuje "izravan" razgovor jednog s drugim sustavom na mreži međusobnim slanjem paketa s informacijama. SMTP protokol je definiran u IETF-ovom RFC821, s naslovom "Simple Mail Transfer Protokol".

**SMTP i ESMTP** SMTP je TCP-temeljeni klijent/poslužitelj protokol, prvobitno definiran u IETFovom RFC821. SMTP je složen u pojedinostima, ali je u osnovi jednostavan. Nakon uspostave pouzdane veze poštanski klijent (MUA) pokreće kratku sekvencu rukovanja s poslužiteljem za poštu (MTA). Klijent zatim MTA-u šalje jednu ili više poruka koje treba dostaviti. Prije slanja svake poruke poštanski klijent šalje popis lokalnih primatelja te poruke i adresu pošiljatelja. U očitoj sličnosti s papirnatom poštom ta se informacija naziva omotnicom poruke.

Sekvenca se rukovanja i razmjena sadržaja poruke odvija u formalnom jeziku koji čine četveroznakovne naredbe i troznamenkasti brojevi kodova odgovora. Na primjer, zapis ESMTP razmjene pošte bi mogao izgledati ovako:

```
$ /usr/sbin/sendmail -v david@mail.fake.com < message</pre>
david@mail.fake.com... Connecting to localmail.mail.fake.com.via smtp...
220 localmail.mail.fake.com ESMTP Sendmail 8.9/8.9/; Sat. 22 May 1999
08:06:22 -
0700
>>> EHLO gateway.oppositemail.com
250 localmail.mail.fake.com Hello michael@gateway.oppositemail.com
[192.168.0.5],
pleased to meet you
>>> MAIL From:michael@gateway.oppositemail.com
250 <michael@gateway.oppossitemail.com>... Sender ok
>>> RCPT To:david@mai..fake.com
250 Recipient ok
>>> DATA
354 Enter mail, end with "." on, a line by itself
>>> .
250 WAA11745 Message accepted for delivery
david@mail.fake.com... Sent (WAA11745 Message accepted for delivery)
Closing connection to localmail.mail.fake.com.
>>> 0UITT
221 localmail.mail.fake.com closing connection
```

Okvir za dodatne mogućnosti u elektroničkoj pošti je nazvan *Extended Simple Mail Transport Protocol* (ESMTP). ESMTP je mehanizam po kojem se o bilo kojoj ekstenziji koje se koriste s tradicionalnim SMTP-om može pregovarati između klijenta i poslužitelja. Mehanizam, kako je opisano u RFC1651, ima otvoreni kraj. Dvije su moguće ekstenzije definirane u RFC1652 i RFC1653.

RFC1652 definira 8-bitno MIME kodiranje, koje korisniku omogućuje slanje 8-bitnih podataka u poštanskim porukama bez potrebe za prekodiranjem podataka uporabom base64, *quoted-print-able* ili nekog drugog načina kodiranja. Ono također uklanja lomove

koji mogu biti posljedica slanja 8-bitnih podataka SMTP poslužitelju koji radi prema RFC821 i koji ne zna što učiniti s dijelovima koji mu pristižu.

Deklaracija o veličini poruke (definirana u RFC1653) poslužitelju pruža način za ograničavanje veličine poruke za čiji je prihvat pripremljen. S RFC821 SMTP-om, poslužitelj je poruku mogao samo odbaciti nakon što je ona već u cjelini poslana i preko mreže stigla u poslužitelj. To je, međutim, rasipanje propusnosti mreže, a poštanski klijent nema načina saznati da je poruka odbačena zbog svoje veličine.

Drugi nastavci mogući s ESMTP-om uključuju zahtjev za obavijesti o stanju isporuke odlazećih poruka (tako da pošiljatelj može biti obaviješten kada poruke stignu na svoja odredišta) i pregovaranje šifriranja između sigurnosnih poslužitelja za poštu radi sigurnijeg slanja pošte.

#### Formatiranje poštanske poruke

SMTP je definirao kako poštansku poruku prenijeti preko Interneta, ali nije definirao kako poštansku poruku prepoznati. RFC822 definira format poruka elektroničke pošte na Internetu. Format je jednostavan, kako standardu i dolikuje:

- Zaglavlje koje sadrži razne obavezne i neobavezne atribute poruci
- Prazan redak
- Sadržaj poruke

Polja zaglavlja su u ovdje datom primjeru znatno dulja od sadržaja:

```
Return-Path: david@mail.fake.com
Received: from localmail.mail.fake.com (localmail.mail.fake.com
[168.9.100.10])
by gateway.oppositemail.com (8.9/8.9) with ESMTP id WAA01322 for
<robert@oppositemail.com>; Sat, 22 May 1999 18.17.06 -0500
Received: from beta.mail.fake.com (beta.mail.fake.com [207.266.47.2]) by
localmail.mail.fake.com (8.9/8.9) with SMTP id WAA13732 for
<robert@oppositemail.com>; Sat, 22 May 1999 18:22:06 -500
Message-Id: 199802180506.WAA13732@localmail.mail.fake.com
X-Sender: pete@localmail.mail.fake.com
X-Mailer: Amiga Eudora Lite Version 2.1.2
Mime-Version: 1.0
Content-Type: text/plain; charset="us-ascii"
Date: Sat. 22 May 1999 18:22:08 -0500
To: robert@oppositemail.com
From: David Wylie david@mail.fake.com
Subject: Test message
This is, a test message.
David
```

Prazan redak nakon retka "Subject" dijeli zaglavlje od tijela poruke koja slijedi. Svaki je sljedeći prazan redak dio poruke i nema strukturno značenje. Većina su polja zaglavlja kratka i imaju prilično očito značenje (poput polja "Subject"), dok su neka druga podugačka i teže razumljiva (poput "Received..."). O detaljnom objašnjenju mnogih standardnih i manje standardnih polja zaglavlja pogledajte 35 poglavlje Costalesove i Allmanove *sendmail, 2nd edition*.

Svaki se redak zaglavlja sastoji od *ključne riječi* i *vrijednosti* - para koji definira značajku te poruke. Na primjer, primatelj poruke je obavezna značajka poštanske poruke. Ta je značajka definirana s ključnom riječi To:, jednim ili više znakova razmaknice ili tabulatora i zatim vrijednosti koja navodi poštansku adresu primatelja. U poruci iznad je ta značajka definirana u sljedećem retku:

To: robert@oppositemail.com

#### sendmail

sendmail se općenito smatra jednom od nekoliko pravih noćnih mora u upravljanju UNIX sustavom. sendmail je težak za podešavanje i može mu se pristupiti na prilično jednak način na koji novajlije pristupaju UNIX-u. Kada je netko jednom Ericu Altmanu (stvaratelju sendmaila) prigovorio da je upravljanje sendmailom prilično zamršeno, on je odgovorio: "Podešavanje je sendmaila složeno zato što je svijet složen". Dok sendmail može učiniti gotovo sve čega se možete dosjetiti, uputiti ga kako da učini ono što želite može biti naporan zadatak.

Ipak, premda je sa sendmailom teško raditi, nedavne su inačice značajno unaprijedile poslove podešavanja i upravljanja sendmailom. Dodan je veliki skup M4 makroa i mogućnost uporabe shvatljivih imena za opcije, što je pored jednoznakovnih prekidača u konfiguracijskoj datoteci konfiguraciju sendmaila učinilo lakšom zadaćom. sendmail se sada može smatrati razumno zrelim proizvodom. Premda se pogreške još uvijek gotovo mjesečno pronalaze, sendmail se koristi u mrežama velikih korporacija za isporuku pošte preko rasprostranjenog skupa mreža i širokih područja.

#### **Povijest sendmaila**

Eric Allman je u kasnim 1970-ima bio na University of California u Berkeleyu. Tamo je napisao prethodnika sendmaila, nazvanog delivermail, koji je izdan 1979. radi rješavanja problema prijenosa pošte između triju mreža u krugu sveučilišta u to vrijeme. Te su tri mreže bile ARPANET (koja je koristila NCP - Network Control Protocol), UUCP poštanski sustav i unutarnja mreža nazvana BerkNet.

Sljedeće je godine ARPANET započeo prijelaz s NCP-a na TCP (Transmission Control Protocol). Prije toga je pošta isporučivana uporabom FTP-a (File Transfer Protocol), međutim, SMTP je razvijen zbog očekivanja mogućeg porasta poštanskog prometa na mreži u rasponu od nekoliko puta.

U odazivu na te promjene Allman je prihvatio sveobuhvatan pristup formatima poruka elektroničke pošte. Ako poruka nije odgovarala prihvaćenom formatu, sendmail je umjesto odbacivanja poruka radije njen format pokušao prilagoditi. Alman se također odlučio na ograničavanje namjenskog cilja sendmaila na usmjeravanje pošte, umjesto da uključi i mogućnosti poštanske aplikacije za kranjeg korisnika. Inačica 4.1 BSD (Berkeley Software Distribution) UNIX-a je sadržavala i prvo javno izdanje sendmaila.

Drugi su se u međuvremenu, odvojeno od Allmana, bavili proširivanjem mogućnosti sendmaila. Pored različitih neslužbenih nastojanja bilo je i nekoliko komercijalnih

proizvođača poput Suna i Hewlett-Packarda koji su razvili svoje vlastite inačice sendmaila kada su ustanovili potrebu za poboljšanjima koja u trenutnim inačicama nisu bila uključena. Iz tih je usporednih razvoja proizašlo nekoliko inačica sendmaila s različitih razinama usklađenosti. Allman je 1998. godine sendmailu od inačice 8.9 dao komercijalan status, dok je inačicu 8.8.x ostavio besplatnom kakva je uvijek i bila.

#### Arhitektura sendmaila

Kompilacija je i instalacija distribucije sendmaila u pravilu jednostavnija nego se na prvi pogled čini. Paket s materijalima uključuje datoteke s opisom instalacije skrojene za mnoge različite sustave i "build" skriptu koja odabire pravu datoteku za lokalno okružje. Ponekad će administrator možda trebati unijeti manje promjene u *make-description* datoteku koja je najsličinija okružju na kojem on radi, kako bi je prilagodio posebnom lokalnom sustavu.

**sendmail kao demon** Sam je sendmail za rad na UNIX sustavu obično podešen kao *demon* koji osluškuje dolazeću poštu.

BILJEŠKA Demon je program u UNIX sustavu koji se izvršava u pozadini bez upravljačkog prozora terminala.

Kada radi kao *demon*, osim ako mu je pri pokretanju naređeno da to ne radi, sendmail se račva i izvršava u pozadini, osluškujući priključnicu (*socket*) 25 radi dolazećih SMTP veza. Naredba da se sendmail izvršava kao *demon* bi na Berkeley UNIX-temeljenom sustavu mogla otprilike izgledati poput ove:

/usr/lib/sendmail -bd -q30m

Ta se naredba može definirati kao jedna od *startup* naredbi koje se izvršavaju pri pokretanju UNIX sustava. Ovdje je primjer naredbe uzet iz skripte sendmail.init, smještene u direktoriju /etc/rc.d/init.d u Linux sustavu, koja se izvodi pri pokretanju sustava:

```
# Start daemons.
echo -n "Starting sendmail: "
daemon sendmail -bd -q1h
echo
touch /var/lock/subsys/sendmail
;;
```

Zastavica -bd pokreće sendmail kao *demon*, a prekidač -q1h sendmailu naređuje da jedamput u svakih sat vremena provjeri red čekanja. Ogledna je naredba, koja je prethodila ovoj, imala prekidač -q koji sendmailu naređuje da red čekanja provjerava svakih trideset minuta.

sendmail nakon pokretanja najprije učita konfiguracijsku datoteku /etc/sendmail.cf. Datoteka sendmail.cf i njene ovisne konfiguracijske datoteke su predstavljene u sljedećem članku.

**Podešavanje i upravljanje sendmailom uporabom datoteke sendmail.cf** sendmail dio svoje snage dobiva iz pristupa koji je omogućen do sendmailovih konfiguracijskih datoteka. Kako se poštanske poruke slijevaju kroz sendmailove konfiguracijske datoteke, sendmail izvršava sve funkcije usmjeravanja poruka, uključujući analiziranje, prosljeđivanje, isporuku, vraćanje i stavljanje u red čekanja.

Srž sendmailove konfiguracije čini datoteka sendmail.cf. To je složena konfiguracijska datoteka koja se učitava samo jednom na početku rada sendmaila. sendmail.cf sadrži tri važne vrste informacija:

- Opcije poput prekidača za upravljanje operacijama, definicije programa za prijenos pošte i lokacije drugih podkonfiguracijskih datoteka za sendmail
- Makroi za uporabu u skupovima pravila
- Skupovi pravila (Rulesets) za prepisivanje adresa na dolazećim i odlazećim porukama

BILJEŠKA Nitko se neće gnjaviti pisanjem daoteke sendmail.cf ispočetka počevši od prazne stranice u tekstualnom editoru. Ako podešavate postavke sendmaila za novu mrežu, gotovo sigurno možete pronaći datoteku sendmail.cf koja traži tek neznatne promjene. Naravno, nakon što vaš poslužitelj za poštu uspješno proradi, napravite sigurnosnu kopiju radne konfiguracije i stavite negdje gdje će biti sigurna.

V8 sendmail je dodao uporabu m4 makroa preprocesso, koji se koristi za stvaranje datoteka sendmail.cf što sadrže mogućnosti koje ste odabrali. Sendmailovoj m4 kreacijskoj datoteci obično treba dati nastavak imena datoteke .mc (macro configuration), no to nije neophodno za rad procesa. Mnoge su ogledne .mc skripte priložene i u standardnim distribucijama sendmaila.

Na primjer, najmanja bi .mc datoteka za Linux radnu stanicu (s odgovarajućim napomenama) mogla izgledati ovako:

```
OSTYPE(linux)dnl
MAILER(local)dnl
```

Ova su dva jedini obavezni makroi u .mc datoteci. Vi ćete vjerojatno željeti više mogućnosti, međutim, ova se datoteka - nazvana smallest\_linux.mc - može pokrenuti sljedećom naredbom (pretpostavivši da ste u direktoriju /usr/lib/sendmail/cf/cf, u koji standardne distribucije sendmaila stavljaju datoteke m4):

m4 ../m4/cf.m4 smallest\_linux.mc > sendmail.cf

U sljedećem su popisu razdvojeni elementi iz prethodne naredbe:

m4	Poziva m4 predprocesor
/m4/cf.m4	Identificira unaprijed zadanu konfiguracijsku datoteku za m4
smallest_linux.mc	To je dvoredna makro konfiguracijska datoteka
>sendmail.cf	Određuje da se izlaz pohranjuje u datoteku sendmail.cf

Sada kada ste m4 upotrijebili za stvaranje datoteke sendmail.cf koja sadrži točno one značajke koje ste zatražili, još uvijek ćete datoteku sendmail.cf trebati prilagoditi za uporabu na vašoj lokaciji. Ipak, uporaba m4 za stvaranje datoteke sendmail.cf je brza

i točna. Pored uporabe mnogih m4 makroa dostavljenih u sendmail distribuciji možete napisati i svoje vlastite, ako smatrate da je potrebno - i uključiti ih u uporabu.

Za brzu konfiguracijsku datoteku za sendmail ispunite World Wide Web obrazac sučelja za m4 konfiguracijski alat (za V8 sendmail) na http://www.completeis.com/sendmail/sendmail.cgi.

U Web obrascu ispunite željene opcije i bit će vam vraćena datoteka sendmail.cf s odabranim opcijama.

**Konfiguracijske datoteke za sendmail i njihove lokacije** sendmail.cf je prva datoteka koju sendmail učitava pri pokretanju. sendmail.cf sadrži lokacije svih drugih podkonfiguracijskih datoteka koje sendmail koristi. Te su datoteke navedene u tablici 26.2.

Tablica 26.2 Konfiguracijske datoteke za sendmail				
lme i lokacija datoteke	Opis			
/etc/aliases	Popis definiranih alijasa za imena - u ASCII tekstu			
/etc/aliases.db	Baza podataka zamjenskih imena sastavljena iz /etc/aliases			
/etc/sendmail.hf	Datoteka sustava pomoći			
/var/log/sendmail.st	Sakupljene statistike			
/var/spool/mqueue/*	Privremene datoteke za poštu u redu čekanja			
/var/run/sendmail.pid	ID procesa daemona			

To su samo unaprijed zadane lokacije datoteka. Budući da su njihove lokacije definirane unutar sendmail.cf, one se mogu preinačiti u bilo koje ime i stazu direktorija koju želite.

sendmail sadrži daleko previše opcija za podešavanje da bi one sve mogle biti predstavljene u ovoj knjizi. Za te opcije postoje dvije vrste sintakse: vrlo tajnovita i malo manje tajnovita. U tajnovitoj inačici sintakse opcije naredba O (veliko *o*, a ne nula) pokreće opcionu naredbu u datoteci sendmail.cf. Tako sljedeće dvije ogledne naredbe iz datoteke sendmail.cf:

```
08pass8
i
```

```
0 EightBitMode=pass8
```

izvršavaju istu zadaću. One sendmailu kažu da 8-bitno formatirane podatke proslijedi kao 8 bitne, a ne da ih potkraćuje na 7-bitne. Uočite promjenu sintakse - jednoznakovna inačica (O8) ne sadrži razmak između O i znaka koji označuje određenu opciju, dok u inačici s imenima (O EightBitMode) između naredbe O i imena opcije treba biti razmak. Kao i sve druge sendmail naredbe, O mora biti na krajnjem lijevom položaju u retku, što je također 1. stupac.

Ovo ograničenje sprečava pogrešno tumačenje naredbe, poput ovog sljedećeg retka koji se također može naći u datoteci sendmail.cf:

#### DMMONGO

Ta naredba definira (D) makro (M) kojem daje vrijednost MONGO, tako da pri ponovnom pisanju pravila možete umjesto upisivanja "MONGO" koristiti \$M. Bez ograničenja da naredbu označava O u krajnjem lijevom stupcu, O u MONGOu bi se mogao protumačiti kao naredba. Ove upravo predstavljene opcije oslikavaju oblik opcione naredbe za uporabu unutar konfiguracijske datoteke. Međutim, opcije se mogu definirati ili u m4 makro datoteci ili u naredbenom retku. Za pisanje gornjih opcija u inačici naredbenog retka prije opcije treba staviti crticu, zatim malo *o* za označavanje jednoznakovne opcione naredbe i veliko *O* za označavanje imenovane opcione naredbe, kako je u ovim primjerima prikazano:

```
-o8pass8
i
-O EightBitMode=pass8
ili
```

```
-0 EightBitMode=pass8
```

#### Skupovi pravila (Rulesets) u sendmailu

sendmail koristi pravila za ponovno pisanje adresa na dolazeću i odlazeću poštu. Ta su pravila samo središte sendmailovih mogućnosti, kao i njegove složenosti. Sendmailova su pravila preoblikovanja adresa specijalizirani tekstualni programski jezik. Eric Allman je sendmail osmislio tako da skupovi pravila izvršavaju dvije osnovne zadaće:

- Pregledavaju svaku adresu primatelja radi utvrđivanja koji MDA treba koristiti za slanje poruke primatelju (ili u njegovu blizinu).
- Preoblikovanje adresa i na omotnici i u zaglavlju poruke radi olakšavanja isporuke ili odgovora.

Pravila preoblikovanja adresa su organizirana u skupove pravila (*ruleset*). *Ruleset* je podrutina ili modul koji se sastoji od slijeda pravila. Kada je adresa proslijeđena *rulesetu*, podrutina tu adresu prosljeđuje po redu svakom od njegovih pravila. Ako dotična adresa odgovara zadanom uvjetu, pravilo se primjenjuje, adresa je preoblikovana i rezultat se prosljeđuje sljedećem pravilu. Ako se adresa ne slaže s trenutnim pravilom, ostavlja se kakva jest, bez preinaka i šalje se na iskušavanje sljedećem pravilu u skupu.

**Sintaksa sendmail** *ruleseta* Svaki je skup pravila identificiran brojem i svaki skup pravila počinje sa *S* u krajnjem lijevom stupcu nakon čega slijedi njegov identifikacijski broj. Pravila počinju slovom *R* i nisu označena brojevima. Skup pravila završava s ne-*R* naredbom. Na primjer:

Sintaksa je ovih pravila tajnovita, ali prilično jednostavna. Svako pravilo ima lijevu i desnu stranu. Napomena nije obavezna. Dvije su strane i neobavezna napomena razdvojene tabulatorima. Lijeva se strana uspoređuje s adresom kao uzorkom od niza znakova. Ako se uzorak slaže s lijevom stranom, adresa se preoblikuje prema desnoj strani pravila i prosljeđuje se sljedećem pravilu.

U sendmail.cf redak za napomene počinje znakom #. Prazni se redovi zanemaruju. S0 definira početak Ruleseta 0. R u sljedećem retku definira početak pravila. \$\* prihvaća svaku adresu koja mu je proslijeđena, a \$: \$>98 \$1 adresu prosljeđuje Rulesetu 98 na daljnju obradu. Tekst "handle local hacks" je napomena. Budući da su pravila razgraničena tabulatorima, na početku dijela s napomenom nije potrebna oznaka napomene (#).

**Glavni skupovi pravila za sendmail** Postoji nekoliko standardnih skupova pravila i oni se u sendmail.cf mogu pojavljivati u bilo kojem redoslijedu. Kada sendmail učitava konfiguracijsku datoteku, on ta pravila odgovarajuće razvrsta. Prema skupu pravila koji se očekuje, iako nije prisutan odnosi se kao da je prisutan, ali prazan. Ovo su glavni skupovi pravila:

- Ruleset 0 prema učitanoj adresi pronalazi odgovarajući MDA
- Ruleset 1 obrađuje adresu pošiljatelja
- Ruleset 2 obrađuje adresu primatelja
- Ruleset 3 predobrađuje sve adrese
- Ruleset 4 naknadno obrađuje sve adrese
- Ruleset 5 preinačava lokalne korisnike bez zamjenskih imena

Aliasi (zamjenska imena) u sendmailu Zamjensko je ime kratica za jednu ili više punih adresa epošte. Premda zamjensko ime može biti samo nadimak za dulju adresu koju

ne želite svaki put utipkavati (na primjer, "john" za

"john.dagenhamster@someothercompany.com"), zamjensko ime može biti i ime za popis nekoliko primalaca.

Mnogi MUA-ovi održavaju svoj vlastiti popis zamjenskih imena, međutim, ti su popisi obično u formatima koji se ne mogu dijeliti s drugim MUA-ovima. Ako na Linux radnoj stanici obično koristite pine, njegova datoteka zamjenskih imena neće biti raspoloživa za vaš klijent Lotus Notes na Windows 95 radnoj stanici kada pismo pišete s tim alatom. Nasuprot tome, mnogi će mogući popisi zamjenskih imena sadržani u zamjenskim imenima koja se održavaju u send-mailovoj datoteci zamjenskih imena biti prepoznati i prošireni kada će poruku obrađivati send-mail, bez obzira na MUA koji je korišten za stvaranje te poruke. sendmail dopušta više datoteka zamjenskih imena - unaprijed je zadana vrijednost do najviše dvanaest.

#### Odavde...

O dodatnim informacijama u vezi ove teme pogledajte 33. poglavlje, "Uporaba elektroničke pošte", koje pokazuje kako pomoću e-mail sustava komunicirati s drugim osobama.

- Više pojedinosti o sendmailu možete saznati s Weba na http://www.sendmail.org/ ili iz knjige Bryana Costalesa i Erica Allmana, *sendmail, second edition*, izdanje O'Reilly & Associates.
- 33. poglavlje, "Uporaba elektroničke pošte" pokriva uporabu korisničkih poštanskih agenata poput elma, pinea i mutta za čitanje i sastavljanje elektroničke pošte.

## P 0 G L A V L J E **27**

### Podešavanje Usenet novinskog servisa

(Steve Burnett)

U ovom poglavlju

Usenet početnica **532** Podešavanje Usenet klijenata **534** 

#### Usenet početnica

Usenet se često pobrka s Internetom, ali Usenet nije Internet. Usenet nije mreža, već usluga koja se obavlja preko Interneta, a također i mnoga računala koja nisu izravno dio Interneta. Najbolji način koji sam pronašao za opisivanje Useneta jest da su to otprilike dvadeset tisuća oglasnih ploča, svaka s različitim naslovom koji opisuje za koju je temu ta ploča namijenjena. Tu možete potražiti oglasnu ploču s temom za koju mislite da bi vas mogla zanimati i pročitati neke ili sve poruke koje su taj dan na ploči. Ako želite, možete i sami staviti poruku, bilo kao javni odgovor na nečiju drugu poruku ili za pokretanje nove rasprave. Također možete kopirati adresu te osobe i poslati joj osobno pismo koje se neće pojaviti na ploči. Kasnije se možete vratiti i pogledati ima li novih zanimljivih poruka od drugih osoba.

Usenet nije poput društvene telefonske linije budući da s drugim osobama ne komunicirate u stvarnom vremenu. Tu ne možete nekoga prekinuti dok razmišlja što bi napisao na ploču (kasnije ipak njegovu poruku možete ponoviti i citirati ga izvan konteksta, međutim, osim što je to nepristojno, to nije isto kao da ga prekinete i spriječite da drugi čuju njegove riječi). Usenet je ipak umnogome poput domjenka, budući da se slabo može nadzirati što netko može reći. Ako na primjer netko ustraje na tvrdnji da su vjeverice jedine toplokrvne životinje koje ne mogu prenositi bjesnoću, on takvu poruku može poslati. Naravno, osobe koje znaju da to nije točno mogu odgovoriti s ispravnom informacijom. Ako prva osoba nastavi tvrditi da je ona u pravu, ostali će čitatelji te ploče vjerojatno prestati obraćati pažnju na poruke koje dolaze od te osobe.

Za informacije o povijesti Useneta pomoću Netscapea prijeđite na

http://www.yahoo.com/Reference/FAQs/ gdje ima nekoliko FAQ-ova BILJEŠKA (Frequently Asked Questions) u vezi Useneta.

Povijest i porijeklo Useneta

Tamo negdje u srednjem vijeku računalstva (oko kasnih 1970-ih) izdana je inačica UNIX-a s oznakom V7. Jedna je od pridodanih aplikacija bila UUCP (Unix-to-Unix-CoPy). 1979. godine su dva diplomirana studenta na Duke University počela UUCP koristiti za razmjenu poruka između dva sustava na sveučilištu. Zaitm je razvijen niz skripta ljuske za razmjenu poruka između Dukea i mreže na University of North Carolina u Chapel Hillu. Kasnije su ljuske skripte napisane u C jeziku i od tada su još mnogo puta prerađivane i proširivane.

#### Struktura Useneta

Ovdje je na mjestu citat Douglasa Adamsa u The Hitchhikers' Guide to the Galaxy, "Svemir je VELIK, Stvarno VELIK". Usenet (u trenutku dok ovo pišem) ima približno deset tisuća različitih novinskih grupa (newsgroup), s ukupnim brojem od nekoliko milijuna sudionika. Neke su od tih novinskih grupa mrtve i u njih nitko nije nikada poslao poruku. Neke su novinske grupe silno aktivne i vjerojatno će se uskoro razdvojiti u više novinskih grupa

(bilo zbog toga što je već vrlo teško baratati njihovim obujmom ili zbog toga što veliki dio sudionika neke novinske grupe zanima uži podskup tema od ostalih čitatelja te novinske grupe). Primjer novinske grupe iz koje je nastala nova dogodio se u comp.sys.powerpc, novinskoj grupi posvećenoj raspravama o RISC procesoru PowerPC. Kada je tvrtka Be, Inc. najavila BeBox, radnu stanicu s dva procesora i novim operativnim sustavom, značajan se dio te novinske grupe usredotočio isključivo na Beov hardver i softver. Da bi se udovoljilo interesima onih koji su željeli raspravljati o BeOS-u i željama onih koje BeOS nije zanimao, osnovana je nova novinska grupa nazvana comp.sys.be. Povremeno se dogodi da za neku raspravu, poput "undocumented features of the PowerPC processor family" postoji zanimanje cijele novinske grupe i informacija stvorena iz te rasprave može postati veliki dio FAQ-a te novinske grupe ili posve zasebni FAQ.

Premda sam obujam Useneta može izgledati preglomazan, u njegovoj strukturi ima neke logike. U tablici 27.1 su prikazani neki od odsjeka prve razine na Usenetu. Oznaka se prve razine pojavljuje kao krajnji lijevi dio svakog Usenet imena.

Tablica 27.1 Imena hijerarhija na Usenetu			
Naziv hijerarhije	Opis		
biz	(Business) - poslovanje		
comp	Sve u vezi s računalima		
misc	(Miscellaneous) - razno		
news	Usenet pitanja, opće informacije		
rec	(Recreational) - sportovi, vještine, hobiji		
sci	(Scientific) - znanstvena		
soc	(Social) - o ljudima i kulturi		
talk	Razgovaranje o svemu		
alt	Sve ostalo		

Postoje i drugi identifikatori prve razine za novinske grupe, od kojih su mnogi više regionalni. Na primjer, novinska grupa čije ime počinje s de.\* uglavnom ima sudionike iz njemačkog govornog područja i većina se de.\* hijerarhija bavi s njemačkim i evropskim pitanjima.

U hijerarhiji novinskih grupa alt.\* zauzima golemi dio Useneta. Zahtjevi su za stvaranje alt.\* novinske grupe lakši nego zahtjevi za stvaranje novinske grupe u nekoj od drugih glavnih hijerarhija. Pored toga, novinske grupe alt.\* ne prenose uvijek svi davatelji pristupa na Internet, zbog dva razloga. Prvi je razlog propusnost - alt.\* novinske grupe čine znatan dio svih Usenet novinskih grupa, a neke od njih, naročito novinske grupe posvećene binarnim datotekama bilo aplikacija ili slika, mogu zauzimati golemi dio propusnosti. Drugi je razlog za ograničavanje alt.\* novinskih grupa njihova uvredljivost. U alt.\* novinskim grupama se dopušta više krajnosti ili nepristojnog jezika i tema čiji će sadržaj vjerojatnije vrijeđati ljude nego ono što se može vidjeti u drugim novinskih grupama. **BILJEŠKA** Neki davatelji Internet pristupa imaju "korisnički definirana" pravila za pristup Usenetu. Oni pružaju grupe s punog popisa noviniskih grupa na Usenetu, ali će prenositi samo novinske grupe koje su zatražili korisnici te mreže. Ovakav "a la carte" pristup znatno smanjuje zakrčenost mreže zbog potreba Useneta, a korisnički interesi nisu ograničeni niti cenzurirani. Drugi se davatelji pristupa drže složenijeg pravila - oni će od grupa koje korisnici zatraže prenositi one koje mogu odobriti prema svojim unaprijed zadanim kriterijima.

Ako ste novi na Usenetu, na svoj pretplatnički popis svakako stavite i novinsku grupu news.announce.newusers. U ovu se novinsku grupu otprilike svaka dva tjedna obnavljaju osnovni informacijski vodiči o pitanjima Useneta.

#### Podešavanje Usenet kljienata

Usenet djeluje na familijarnom klijent/poslužitelj odnosu. Poslužitelj razmjenjuje poruke s drugim poslužiteljem i pohranjuje ih u lokalni sustav. Da biste mogli čitati novinsku grupu na Usenetu, trebate se na vašoj mreži ili kod davatelja pristupa na Internet raspitati za ime NNTP poslužitelja. Nakon što nabavite ime poslužitelja (koji će obično izgledati nešto poput test.fake.com ili 192.168.2.221), možete početi.

#### NN i TIN

NN i TIN su dva vrlo slična *newsreadera* (programa za čitanje Usenet poruka) i oba su uključena u mnoge uobičajene distribucije Linuxa.

Ako koristite inačicu TIN-a kompiliranu s NNTP opcijama iz korisničkog računa UNIX ljuske, pokušajte ove naredbe:

Ako koristite ljusku ksh ili bash:

\$ NNTPSERVER= test.fake.com tin -r -f .fakenewsrc I .newsnet/index

Ako koristite ljusku C ili ljusku tcsh:

% setenv NNTPSERVER test.fake.com; tin -r -f .fakenewsrc I .newsnet/.index

Podešavanje je slično i za NN:

Ako koristite ljusku ksh ili bash:

\$ NNTPSERVER=test.fake.com nn newsrc=- .fakenewsrc

- Ako koristite ljusku C ili ljusku tcsh:
  - % setenv NNTPSERVER test.fake.com nn newsrc=-/.fakenewsrc

#### **Pine**

Pine je najčešće korišten kao *mail reader* (program za čitanje pošte), ali se može koristiti i za čitanje Usenet poruka. Za podešavanje vašeg klijenta za poštu *pine* za pristup Usenetu pritisnite <S> (Setup) i zatim <C> (Config). Zatim redak "news collections" uredite kako je ovdje prikazano:

```
*{test.fake.com/NNTP}[]
```

Nakon toga pritisnite  $\langle E \rangle$  za izlaz iz pinea i njegovo pononvo pokretanje. Zatim pritisnite  $\langle L \rangle$  (List Folders), pronađite mape s novinskim grupama i za pretplatu na željenu novinsku grupu odaberite  $\langle A \rangle$  (Add).

Nakon toga bi zaslon mogao izgledati slično ovome:

PINE 3.96 FOLDER LIST Folder: INBOX 313 Messages Folder-collection <mail/[]> \*\* Default for Saves \*\* (Local) [ Select Here to See Expanded List ] News-collection <News on test.fake.com> (Remote) [ Select Here to See Expanded List ]

Premda čitanje novosti može biti zabavno i poučno, samo je čitanje (bez slanja) poruka u novinskim grupama obrazac ponašanja nazvan *lurking* (vrebanje) i na takvo se ponašanje korisnici s dužim stažom na Usenetu mršte. Ako u novinske grupe želite poslati nove Usenet poruke ili poslati odgovore elektroničkom poštom, obično trebate ispuniti i sljedeća polja:

- Vaše ime i prezime
- Vaša e-mail adresa (korisničkiid@imedomaćina.domena)
- Računalo koje prosljeđuje vašu poštu (vašeg administratora sustava upitajte za ime SMTP poslužitelja za poštu)

To je obično već podešeno za vaš softver za slanje i čitanje elektroničke pošte. Vjerojatno ćete moći iskoristiti iste unesene stavke. To ipak provjerite kod vašeg administratora sustava.

BILJEŠKA Većina će administratora sustava imati već pripremljenu stranicu s potrebnim informacijama. Ako je vaš nema, sakupljenu potrebnu informaciju sačuvajte na lako dostupnom mjestu, tako da nekoj drugoj osobi kojoj to također treba možete olakšati traženje.

S pineom i drugim programima za slanje i čitanje Usenet poruka, kao i s većinom e-mail aplikacija možete definirati datoteku potpisa - *sigfile* (kratica za *signature file*). To je blok informacija koje želite da budu uključene u svaku poruku koju šaljete u novinsku grupu. Ljudi tu obično uključuju svoje ime, e-mail adresu i (ako je taj korisnički račun povezan s poslom) svoju titulu i rang. Neki tu uključuju i kratki citat koji smatraju domišljatim.
BILJEŠKA Ono što biste u vašu datoteku s potpisom stvarno mogli poželjeti staviti su vaša kućna adresa i telefonski broj. Međutim, čak i bez da u obzir uzimamo nečije obijesno ili pakosno ponašanje, uvijek postoji mogućnost da vas osoba koja s vama obavlja zakonito poslovanje nazove iz Singapura, pritom zaboravivši na razliku u vremenskoj zoni. Upamtite da vam osoba koja preko Interneta čita vašu e-poštu ili Usenet poruku uvijek može odgovoriti na isti način.

Drugo upozorenje ako sastavljate datoteku s potpisom jest da upamtite da ste vaš potpis definirali. Ručno umetanje vaše datoteke s pitpisom na kraj poruke koju šaljete ili dodavanje datoteke s potpisom kao pridružene datoteke kada se ona ionako automatski pridodaje, učinit će vas pomalo smiješnim.

U jednom ćete trenutku pri čitanju novinskih grupa poželjeti ili pridonijeti trenutnoj raspravi ili započeti raspravu o novoj temi. Prije nego počnete aktivno sudjelovati na Usenetu, trebate razumijeti pravila podkulture Useneta. Pri slanju poruka u novinske grupe držite se ovih općih savjeta:

- Nemojte slati poruku ako grupu niste pratili barem tjedan dana i nemate osjećaj za ton rasprave koja se u njoj vodi. Smatrate li da je prosječna rasprava koja se tu vodi previše nepristojna i surova i da li će vaše opaske razdražiti ostale sudionike novinske grupe?
- Podražite i pročitajte FAQ (ako postoji). Mnogi su od FAQ-ova za Usenet pohranjeni na ftp://rtfm.mit.edu/pub/Usenet/.
- Poruke šaljite samo novinskim grupama u kojima one imaju smisla. Ako pokušavate prodati vodeni krevet u Kaliforniji, vaš biste oglas vjerojatno trebali objaviti samo u .forsale novinskih grupama u neposrednoj blizini. Nije vjerojatno da bi čitatelji novinske grupe triangle.forsale koja služi za istočni dio Sjeverne Karoline bili voljni vaš vodeni krevet prevoziti preko cijele zemlje. Slično tome, ako imate pitanje o računalima Amiga, nemojte ga poslati u novinsku grupu koja raspravlja o Macintoshima.

Može se dogoditi da u novinsku grupu pošaljete poruku, ali je kasnije pri pregledavanju te novinske grupe ne možete pronaći. Neke se novinske grupe uređuju (moderiraju), što znači da sve poruke za tu novinsku grupu čita osoba ili skupina osoba koje nepodesne poruke izbacuju i ne šalju ih u novinsku grupu. Je li neka novinska grupa uređivana ili nije, možete saznati čitajući FAQ ili povelju za tu novinsku grupu ili prateći poruke u toj novinskoj grupi tjedan - dva prije slanja poruke, tražeći da li netko za sebe kaže da je moderator te novinske grupe.

# **BILJEŠKA** Ako pri prvom prijavljivanju u novinsku grupu ne uspijete lako pronaći FAQ, pokušajte koristiti jedan od Usenet-sposobnih pretraživačkih servisa, poput

http://www.dejanews.com ili http://www.altavista.digital.com i kao ključne riječi za traženje staviti ime novinske grupe i "FAQ". U vašoj prvoj poruci možete postaviti i pitanje "Postoji li FAQ za ovu grupu?" i spasiti se dodatne tjeskobe.

U pravilu, koristite se zdravim razumom i sve će vjerojatno biti u redu. O više savjeta o ponekad nezgodnim pitanjima *pravila ponašanja na Mreži (Netiquette)* pomoću vašeg Web pretraživača pronađite **http://www.fau.au/rinaldi/netiquette.html**, Web lokaciju s mnogim dobrim materijalima općenito za pridošlice na Internet.

# Odavde...

O više informacija o Usenetu pričitajte 34. poglavlje, "Preživljavanje u novinskim grupama Useneta".

Ono opisuje novinski sustav Useneta koji se koristi za razmjenu javnih poruka.

27. poglavlje, "Podešavanje Useneta" pruža temelje za neke od tema u tom poglavlju.

# P 0 G L A V L J E **28**

# Uporaba editora emacs

(Jack Tackett)

#### U ovom poglavlju

Pokretanje emacsa Uporaba emacsa Upisivanje datoteka i spremanje memorijskog spremnika Pregled osnovih naredbi Prilagođavanje emacsa Ime editora emacs je kratica za Editor MACroS, koji je svoj život započeo kao zamjena za jedan od prvih tekstualnih editora teco. emacs je jedan od najčešće korištenih editora raspoloživih u današnjem UNIX/Linux svijetu. Štoviše, inačice su emacsa raspoložive za gotovo svaku poznatu računalnu platformu, od Linuxa do Microsoft Windowsa.

Puna je inačica emacsa vrlo velika i zauzima nekoliko megabajta diskovnog prostora. To je potpuno opremljen, vrlo moćan editor s proširenim mogućnostima koje nadilaze samo editiranje teksta. U nekim ga instalacijama možete koristiti za uređivanje datoteka, održavanje kalendara, rad s e-poštom, upravljanje datotekama, čitanje UseNet ili mrežnih novosti, crtanje skica, kao kalkulator, pa čak i pretraživati World Wide Web. emacs je

na neki način radno okružje koje sadrži tekstualni editor. Omiljena se inačica emacsa raspačava preko GNU licence. To je inačica emacsa koja se instalira tijekom instalacije Linuxa.

### Pokretanje emacsa

Stvaralac editora emacs je praotac GNU-a Richard Stallman. Izvorni je kôd za emacs u biti besplatno raspoloživ pod GNU licencom. Stallman je osnivač i predlagač Free Software Foundation i GNU (GNU's Not UNIX) projekta. Činjenica da je emacs besplatno raspoloživ odgovara Stallmanovoj filozofiji da bi sav softver trebao biti besplatan i da bi računalni sustavi trebali svima biti otvoreni za uporabu. Korisnike se čak ohrabruje na preinačavanje softvera, ali su te promjene pritom obavezni podijeliti s drugima.

Editor emacs nema dva osnovna načina rada poput editora vi, što znači da sve što pišete stavljate u datotečni spremnik (*buffer*). Za izdavanje naredbi editoru za spremanje datoteka, traženje teksta, brisanje teksta itd. trebate koristiti druge tipke. Za izdavanje različitih naredbi u emacsu koristite tipku <Ctrl> u kombinaciji s raznim znakovima (obično <Ctrl-x> i <Ctrl-c>) i tipku <Esc>. Paleta je uobičajenih naredbi opisana kasnije u ovom poglavlju.

Vidi "Dva načina rada editora vi", str. 182.

Te su emacs naredbe zapravo prečice do punih teksktualnih naredbi. Na primjer,

<Ctrl-x><Ctrl-s>, koja trenunti memorijski spremnik sprema u datoteku, je zapravo prečica koja korisnika oslobađa punog postupka - pritisnuti tipku <Esc> i zatim utipkati stvarnu emacs naredbu: -x save-buffer. Kao što možete vidjeti, uporabu je slijeda

kombinacija tipki <Ctrl-x><Ctrl-s> mnogo jednostavnije i lakše upamtiti neko punu emacs naredbu. Na kraju je ovog poglavlja prikazan sažeti popis osnovnih naredbi.

emacs također omogućuje uređivanje više *buffera* (memorijskih spremnika), ili datoteka, u istoj *sesiji*. To znači da s emacsom možete istovremeno uređivati više od jedne datoteke. Ovo poglavlje pokriva i neke od naredbi za baratanje memorijskim spremnikom. emacs te memorijske spremnike koristi i za držanje izbrisanog teksta kao i za prikaz naredbenog retka za upis naredbi.

Za pokretanje emacsa utipkajte **emacs** i pritisnite <Return> (Enter). Pojavit će se prazan zaslon sa statusnim retkom na dnu.

U ovom se poglavlju ne govori o prečicama na tipkovnici i naredbama koje emacs koristi, međutim, pomoć možete dobiti pritisnuvši <Ctrl-h><h>. Nakon toga, za potpuni izlazak

možete pritisnuti <Ctrl-x><Ctrl-c> ili pritisnite <Ctrl-x><1> za vraćanje u editor. Tako je emacs, za razliku od vi, opremljen sustavom pomoći pa čak i Tutorijalom (vodičem kroz program) kojima možete pristupiti tijekom rada.

Nakon što zatražite ugrađen sustav pomoći, emacs će uvesti još jedan memorijski spremnik i nakon toga je spreman za pružanje pomoći. Ako pritisnete <t>, emacs pokreće odličan vodič. Ako pritisnete <k>, emacs će pružiti pomoć o sljedećoj naredbi/tipki koju unesete. Tako, ako pritisnete <Ctrl-h><k><Ctrl-w>, emacs će pružiti informaciju o brisanju označenog područja.

Za vraćanje radu u editoru pritisnite <Ctrl-x><1> i emacs će se vratiti uređivanju samo jednog memorijskog spremnika.

Cjeloviti je GNU emacs sustav velik, ali se može prilagoditi vašem lokalnom okružju. Neke manje inačice emacsa koje su također raspoložive su Freemacs (autor Russel Nelson) i MicroEmacs (prvobitno od Davea Conroya). Također upamtite da Linux distribucija pruža i neke druge editore slične emacsu - JED i JOVE, koji su po veličini mnogo manji od pune instalacije emacsa.

Ovo poglavlje ne pokriva sve mogućnosti emacsa - to bi zahtijevalo mnogo više prostora od raspoloživog. U stvari, o emacsu su već napisane cijele knjige. Zbog toga ćete ovdje saznati samo najneophodnije naredbe za poslove uređivanja. Ako želite više saznati o naprednim mogućnostima

BILJEŠKA

emacsa i naprednim operacijama uređivanja teksta, pogledajte uputstva dostavljena s vašim sustavom. Ne trebate postati stručnjak za emacs da biste ga koristili. emacs ima vrlo detaljan vodič kao dio svog sustava. O vodiču ćete više saznati kasnije u ovom poglavlju, ali za njegovo pokretanje samo pritisnite <Ctrl-h><t>.

#### Uporaba emacsa

Tekst uređujete stvaranjem novog teksta ili preinačavanjem postojećeg teksta. Kada stvarate novi tekst, stavljate ga u datoteku s uobičajenim imenom Linux datoteka. Kada preinačavate postojeći tekst, za pozivanje kopije datoteke u sesiju editiranja teksta koristite postojeće ime datoteke. U oba slučaja, kada koristite editor, tekst je pohranjen u memoriji sustava u spremišnom prostoru nazvanom buffer (memorijski spremnik).

Uporaba memorijskog spremnika onemogućuje izravno mijenjanje sadržaja datoteke sve dok sadržaj tog memorijskog spremnika ne odlučite spremiti u datoteku. Tako datoteku možete slobodno preuređivati i zatim odlučiti da to ništa ne valja i početi ispočetka, a da spremljena datoteka pritom ostane netaknuta.

emacs vam omogućuje istovremeno uređivanje nekoliko memorijskih spremnika. Na taj način tekst možete isijecati i uljepljivati iz jednog u drugi memorijski spremnik, uspoređivati tekst iz različitih datoteka ili neku datoteku umetnuti unutar druge datoteke. emacs čak koristi posebni memorijski spremnik za prihvat naredbi i prikazivanje informacija korisniku. Taj je memorijski spremnik, mini-spremnik, prikazan na dnu zaslona.

emacs vam također omogućuje prikaz sadržaja različitih memorijskih spremnika u njihovim vlastitim prozorima - tako istovremeno možete vidjeti nekoliko datoteka, čak i ako ne koristite grafičko korisničko sučelje.

#### Pogled na zaslon emacsa

Na slici 28.1 je prikazan uobičajen zaslon emacsa. Njegov gornji dio prikazuje sadržaj različitih memorijskih spremnika, ponekad u više prozora. Na dnu je zaslona prikazan statusni redak (mode line), obično u suprotnoj boji. Taj redak korisnicima pruža informacije o memorijskom spremniku, poput imena memorijskog spremnika, o glavnom i sporednom načinu rada i količini teksta prikazanog u memorijskom spremniku. Ispod statusnog retka se nalazi jedan redak *mini-spremnika* (mini-buffer), u koji unosite emacs naredbe i gdje emacs izvještava o ishodima pojedinih naredbi.

Slika 28.1	this is a test
Uobičajeni zaslon	
emacsa prikazuje	
područja memorijskog	
spremnika i	
mini-spremnika.	
memorijski spremnik	
Mini-sprompik	=* GNU EMACS (> == main == (TEXI Fill) ======= ALL===========================
	Statusni redak

Trenutni je položaj u memorijskom spremniku prikazan pomoću kursora. emacs kursor naziva *point*, naročito u ugrađenom sustavu pomoći, stoga je važno upamtiti taj pojam za kursor.

#### Stvaranje vaše prve emacs datoteke

Sljedeće vam upute pokazuju kako urediti vašu prvu emacs datoteku. Ako naiđete na poteškoće, za izlazak iz programa i ponovno pokretanje pritisnite <Ctrl-x><Ctrl-c>. Slijedite ove korake:

- 1. Pokrenite emacs (utipkajte **emacs** i pritisnite <Return>). Ugledat ćete zaslon prikazan na slici 28.1.
- 2. U memorijski spremnik dodajte sljedeće redove teksta:

```
Poslovi za danas.
a. Vježbati emacs.
b. Razvrstati prodajne podatke i ispisati rezultate.
```

Za ispravljanje grešaka u retku u kojem tipkate možete koristiti tipku <Backspace>. Međutim, ne brinite se previše o preciznosti jer je ovaj primjer samo za vježbu. Druge ćete načine za unošenje promjena saznati u kasnijim člancima ovog poglavlja.

Obratite pažnju na mini-spremnik na dnu zaslona. Ovdje se pojavljuju tipke koje ste pritisnuli zato jer utipkavate naredbe za editor *emacs*. ■

#### BILJEŠKA

3. Vaš memorijski spremnik spremite u datoteku nazvanu emacs-pract.1. Najprije pritisnite <Ctrl-x> i zatim utipkajte emacs-pract.1. Uočite da se emacs-pract.1 pojavljuje na dnu zaslona. Pritisnite <Return>. Ova naredba sadržaj memorijskog spremnika sprema ili upisuje u datoteku emacs-pract.1 (koju ste prije odredili).

U statusnom retku biste trebali ugledati sljedeću potvrdu:

Wrote /root/emacs-pract.1

Ova izjava potvrđuje da je datoteka emacs-pract.1 stvorena i spremljena na disk. Vaš prikaz može biti drukčiji ako ovu informaciju niste utipkali točno kako je navedeno.

- BILJEŠKA Obratite pažnju na broj znakova u imenu datoteke. Za razliku od MS-DOS-a i Windowsa, Linux vam omogućuje unos imena duljeg od osam znakova i više od tri znakova za nastavak imena datoteke.
  - 4. Izađite iz emacsa pritiskom na <Ctrl-x><Ctrl-c> i zatim na <Return>. Ako imate nespremljeni materijal, emacs bi vas mogao upitati želite li spremiti nespremljeni memorijski spremnik/datoteku. Ako vas emacs to upita, za spremanje informacije pritisnite <y> ili pak <n>, ako je ne želite spremiti. emacs se zatim zatvara i vraćate se u naredbeni redak prijavne ljuske.



#### RJEŠAVANJE POTEŠKOĆA

emacs **znakove koje upisujem stavlja u mini-spremnik i s tim znakovima pokušava učiniti čudne radnje.** Ako dvaput pritisnete <Esc>, emacs ulazi u programsko okružje LISP. LISP je izvorni jezik koji je Stallman koristio za programiranje emacsa i programeri kroz LISP mogu proširivati i prilagođavati emacs. Ako pritisnite <Esc><Esc>, emacs ulazi u stanje primanja naredbi i od korisnika očekuje unos LISP naredbe. Za izlazak iz tog načina rada samo pritisnite <Return>.

#### Pokretanje emacsa uporabom postojeće datoteke

Za uređivanje ili pregledavanje datoteke koja već postoji u vašem trenutnom direktoriju upišite *emacs* i zatim ime datoteke. Na primjer, pokušajte to s datotekom koju ste stvorili u prethodnom članku. Unesite ovu naredbu:

```
emacs emacs-pract.1
```

Ugledat ćete sljedeće:

Poslovi za danas. a. Vježbati emacs. b. Razvrstati prodajne podatke i ispisati rezultate.

Pogledajte mini-spremnik. On sadrži ime datoteke koju uređujete.

#### RJEŠAVANJE POTEŠKOĆA

**Utipkao sam ime postojeće datoteke, ali se** emacs **ponaša kao da stvaram novu datoteku**. Možda ime datoteke niste ispravno utipkali ili ste možda utipkali ime koje u vašem trenutnom direktoriju ne postoji. Ako, na primjer, utipkate emacs pract1. i pritisnete <Return>, a u vašem trenutnom direktoriju nema datoteke pract1, emacs će se pokrenuti, međutim, budući da datoteka s tim imenom ne postoji, emacs će se ponašati kao da stvarate novu datoteku.

Pokušavam uređivati datoteku, ali emacs prikaže poruku da je odobrenje za učitavanje uskraćeno i pojavi se naredbeni redak ljuske. Pokušali ste uređivati datoteku za čije učitavanje nemate dopuštenje. Također, ne možete uređivati direktorij - to jest, ako utipkate emacs *ime\_direktorija* - dakle, ako utipkate ime direktorija, emacs će vas obavijestiti da ste otvorili direktorij i neće vam dopustiti da ga uređujete. Ako emacs pokušate koristiti s binarnom, umjesto s ASCII datotekom, ugledat ćete zaslon pun čudnih (kontrolnih) znakova - nešto što ne možete čitati ni uređivati. emacs očekuje da datoteke budu spremljene kao čisti tekst.

Kada u emacsu pokušam otvoriti datoteku, dobijem poruku da je redak predugačak. Znači da emacs pokušavate koristiti na podatkovnoj ili binarnoj datoteci koja je naprosto jedan dugačak niz bajtova.

Kada datoteku pokušam spremiti s kombinacijom tipki <Ctrl-x><Ctrl-s>, terminal se "zamrzne" i ne reagira na naredbe s tipkovnice. Vaš se terminal vjerojatno odaziva na znakove za kontrolu toka <Ctrl-s> i <Ctrl-q>. Za ponovno pokretanje vaše sesije pritisnite <Ctrl-q>.

Datoteku sam otvorio u *emacsu* i na zaslonu su se pojavili neki čudni znakovi. Možda emacs koristite s datotekom stvorenom u nekom programu za obradu teksta.

U svim ovim slučajevima pritisnite <Ctrl-x> <Ctrl-c> za izlazak iz emacsa i vraćanje u naredbeni redak vaše prijavne ljuske. Zatim na upit želite li datoteku spremiti odgovorite s n. Uporaba ovih tipki osigurava izlazak iz emacsa bez preinačavanja postojeće datoteke.

#### Izlazak iz emacsa

Kako je već navedeno, za izlazak iz emacsa pritisnite <Ctrl-x><Ctrl-c>. Ako u datoteku još niste spremili nikakve promjene, emacs će vas upitati želite li spremiti sadržaj memorijskog spremnika. Ako utipkate y, emacs sprema datoteku i vraća vas u Linux ljusku. Ako niste dostavili ime datoteke, emacs će vas upitati za ime datoteke i zatim izaći. Ako na upit za spremanje memorijskog spremnika odgovorite s *n*, emacs će vas upitati jeste li sigurni da želite izaći bez spremanja memorijskog spremnika. Na ovaj upit trebate utipkati potpun odgovor - *yes* ili *no*. Ako odgovorite s yes, emacs vas vraća u Linux bez spremanja bilo kakvih promjena koje ste uradili u memorijskom spremniku. Također, ako ste otvorili više memorijskih spremnika, emacs će vam postavljati upite za svaki memorijski spremnik.

#### **UPOZORENJE**

Prema osnovnim postavkama emacs izvršava periodična spremanja memorijskih spremnik a koje trenutno uređujete. emacs sam po sebi ne pravi sigurnosne kopije datoteka, premda se, kada datoteku prvi put spremate, kopija datoteke sprema u *#filename#*. Nakon što pritisnete <Ctrl-x><Ctrl-s>, izvorna je datoteka promijenjena i ne može se vratiti u prvobitno stanje. Stoga biste prije početka rada u emacsu sami trebali napraviti sigurnosne kopije datoteka na kojima ćete raditi i tako se osigurati da se važna datoteka s automatskim spremanjem neće nenamjerno

#### **UPOZORENJE**

Dobro promislite kada na upit o spremanju promjena pri izlasku iz programa odgovarate s *n*. Kada odgovorite s *n*, sve će promjene koje ste u datoteku unijeli od njenog posljednjeg spremanja biti izgubljene. Ako niste sigurni u vezi vaših promjena u datoteci, bolje se osigurati i datoteku spremiti pod drugim imenom.

Možda još niste posve završili s vašim poslom u emacsu, ali se trebate posvetiti drugim poslovima u Linuxu. U tom slučaju imate nekoliko mogućnosti:

- emacs možete privremeno obustaviti i vratiti se u Linux ljusku.
- Možete se prebaciti na drugi virtualni terminal.
- Naredbu ljuske možete izdati i iz emacsa.

**Privremena obustava emacsa** emacs možete privremeno obustaviti - u stvari, gotovo svaku Linux aplikaciju možete privremeno obustaviti - pritiskom na <Ctrl-z>. Ova kombinacija tipki trenutnu aplikaciju stavlja u pozadinu i pruža vam drugi naredbeni redak ljuske. Naredba za ponovno aktiviranje emacsa ovisi o ljuski koju izvršavate. Možete utipkati naredbu *fg*, što znači vraćanje pozadinske naredbe u prvi plan (*foreground*). Ako ljuska koju koristite ne razumije tu naredbu, utipkajte **exit**, što pononvo aktivira vašu emacs *sesiju* sa svim datotekama i memorijskim spremnicima i dalje netaknutima.

**Prebacivanje između emacsa i drugih virtualnih terminala** Linux korisniku pruža šest virtualnih terminala - dakle, imate šest različitih *sesija*. Dok ste u emacsu, za aktiviranje drugog terminala možete pritisnuti <Ctrl-Alt-Fx>, gdje je Fx jedna od funkcijskih tipki, od F1 do F6. Ako se još niste prijavili u *sesiju* na tom terminalu, sada to trebate učiniti, baš kao da upravo pokrećete Linux. Zatim imate u potpunosti aktivnu Linux *sesiju*. Za prebacivanje natrag u emacs samo ponovno pritisnite <Ctrl-Alt-Fx>. Ako ste zaboravili kojoj *sesiji* emacs pripada, pritiskom na <Ctrl-Alt> i različite funkcijske tipke možete kružiti kroz sve virtualne terminale.

**Vidi** "Upravljanje korisnicima", **str. 107.** 

Možete upotrijebiti i naredbu ps za prikaz svih aktivnih procesa, kako je prikazano u Ispisu 28.1. Izlaz naredbe ps -guax prikazuje u kojem se terminalu svaki proces trenutno izvršava.

Ispi	s 28.1	1 I:	zlaz na	aredbe	ps					
USER	PID	%CPU	%MEM	SIZE	RSS	ΤΤΥ	STAT	START	TIME	COMMAND
root	1	0.5	3.1	44	208	?	S	20:48	0:00	init
root	6	0.0	1.8	24	124	?	S	20:48	0:00	bdflush (daemon)
root	7	0.0	1.9	24	128	?	S	20:48	0:00	update (bdflush)
root	23	0.0	2.9	56	200	?	S	20:48	0:00	/usr/sbin/crond -110
root	36	0.6	3.5	65	240	?	S	20:48	0:00	/usr/sbin/syslogd
root	38	0.1	2.9	36	200	?	S	20:48	0:00	/usr/sbin/klogd
root	40	0.3	3.2	68	216	?	S	20:48	0:00	/usr/sbin/inetd
root	42	0.1	3.0	64	204	?	S	20:48	0:00	/usr/sbin/lpd
root	47	0.1	6.0	259	404	?	S	20:48	0:00	sendmail:accepting c
root	51	0.1	2.0	32	140	?	S	20:48	0:00	selection -t ms

nastavlja se

Ispis	28.1		Nastava	ak								
root	52	1.5	7.2	376	484	v01	S	20:48	0:01	-bash		
root	53	0.3	3.4	88	232	v02	S	20:48	0:00	/sbin/getty	tty2	3840
root	54	0.3	3.4	88	232	v03	S	20:48	0:00	/sbin/getty	tty3	3840
root	55	0.2	3.4	88	232	v04	S	20:48	0:00	/sbin/getty	tty4	3840
root	56	0.3	3.4	88	232	v05	S	20:58	0:00	/sbin/getty	tty5	3840
root	57	0.3	3.4	88	232	v06	S	20:48	0:00	/sbin/getty	tty6	3840
root	67	0.0	3.5	80	240	v01	R	20:49	0:00	ps -guax		

Zatim ovu TTY vrijednost, u rasponu od v01 do v06, možete iskoristiti za odabir odgovarajućeg virtualnog terminala. Na primjer, ako naredba ps ukazuje da emacs trenutno radi na tty v01 i tty v02, pritisak na <Alt-F1> ili <Alt-F2> vas vraća u odgovarajuću emacs *sesiju*.

**Pristupanje Linux naredbama iz emacsa** Ponekad samo trebate na brzinu provjeriti da li neka datoteka postoji ili izvršiti neke druge brze Linux naredbe i za izvršavanje vam te radnje nije potrebna sesija u punoj ljusci. U takvom slučaju naredbe ljuske možete izdati i iz emacsa. Za izvršavanje ljuske unutar emacsa pritisnite <Ctrl-u><Esc><!>. Od vas će se zatražiti unos naredbe ljuske - unesite naredbu i pritisnite <Return>. emacs će naredbu proslijediti Linux ljusci, koja će tu naredbu izvršiti.

Ako ne pritisnete <Ctrl-u>, emacs će izlaz staviti u memorijski spremnik/prozor nazvan \*Shell Command Output\*. O prozorima ćete više saznati kasnije u ovom poglavlju, ali, u osnovi, prozori vam omogućuju da istovremeno vidite više memorijskih spremnika. emacs pruža različite naredbe za kretanje po prozorima i njihovo brisanje (bez brisanja pripadajućih memorijskih spremnika). Za brisanje izlaznog prozora pritisnite <Ctrl-x><1>.

#### Undo - vraćanje u prethodno stanje

U emacsu možete poništiti vašu posljednju radnju ili promjenu memorijskog spremnika tako dugo dok tu promjenu niste spremili u datoteku na disku. Vraćanje u prethodno stanje možete izvesti pritiskom na <Ctrl-x><u>. Ponovljenom uporabom ove naredbe možete poništavati promjene koje ste učinili u memorijskom spremniku.

#### BILJEŠKA

Sve vaše akcije emacs bilježi najprije u memoriji, a zatim i u datoteci na disku, tako da teoretski možete poništiti djelovanje bilo koje naredbe i akcije – jedino je ograničenje

prostor na disku u kojem emacs drži navedene podatke. Međutim, u praksi će naredba undo biti korisna samo za nekoliko posljednjih naredbi ili promjena koje ste učinili.

Na žalost, naredba *undo* je djelotvorna samo za ispravljanje onog što upisujete u memorijski spremnik i ne može poništiti promjene koje ste spremili u datoteku.

Ako datoteku želite ponovno učitati s diska (i tako prebrisati vaše trenutne promjene u memorijskom spremniku), možete pritisnuti Ctrl-x><Ctrl-r>. Ta naredba zadanu datotekuučitava u trenutni memorijski spremnik i pritom briše njegov prethodni sadržaj. Tako će emacs,ako navedete isto ime datoteke, trenutni sadržaj memorijskog spremnika zamijeniti sa sadržajemdatoteke na disku. To je brz način za poništavanje mnogih promjena bez izlaženja i ponovnogpokretanja emacsa.

Međutim, što ako je emacs automatski spremio datoteku ili ste datoteku sami spremili s neželjenim promjenama? Kada datoteku prvi put spremate, emacs stvara i sigurnosnu kopiju te datoteke, ali tek kada tu datoteku spremate. Ime je te sigurnosne kopije datoteke jednako imenu datoteke, osim što je na početku i kraju imena znak #. Dakle, ako je vaša datoteka nazvana emacs-prtc.1, ime sigurnosne kopije datoteke je #emacs-prtc.1#. Ako vašu trenutnu datoteku slučajno spremite s neželjenim promjenama i tako izbrišete njenu raniju inačicu, moći ćete iskoristiti sigurnosnu kopiju datoteke i početi ispočetka.

# Upisivanje datoteka i spremanje memorijskog spremnika

Saznali ste kako memorijski spremnik upisati u datoteku i izaći iz emacsa. Ponekad, međutim, samo želite spremiti njegov sadržaj bez izlaženja iz emacsa. Datoteku biste redovito trebali spremati tijekom rada na njenom uređivanju. U slučaju pada sustava ili nestanka struje izgubili biste vaš rad ako ga niste nedavno spremili. Za spremanje

memorijskog spremnika pritisnite <Ctrl-x><Ctrl-s>.

Ako ste emacs pokrenuli bez navođenja imena datoteke, za spremanje datoteke na kojoj radite trebate navesti njeno ime. U tom slučaju pritisnite <Ctrl-x><Ctrl-s>, utipkajte ime datoteke i pri-tisnite <Return>.

Možda ćete datoteku u memorijskom spremniku željeti spremiti pod drukčijim imenom od datoteke s kojom ste započeli rad. Na primjer, emacs pokrenete s datotekom emacs-pract.1, u datoteci napravite neke promjene i zatim te promjene želite spremiti u novu datoteku bez gubljenja izvorne datoteke emacs-pract.1. Za spremanje datoteke pod novim imenom pritisnite <Ctrl-x><Ctrl-w>. emacs će vas upitati za ime datoteke. Memorijski spremnik će zatim biti upisan u novo imenovanu datoteku. Ako se naredba uspješno izvrši, ugledat ćete ime te datoteke.

Ako navedete ime postojeće datoteke, u mini-spremniku će se pojaviti poruka s pitanjem da li novu datoteku želite upisati preko te postojeće - tada samo trebate odgovarajuće odgovoriti na pitanje.

#### Uporaba datoteka

Ako želite učitati novu datoteku koju želite preurediti, emacs vam može omogućiti učitavanje nove datoteke u trenutni memorijski spremnik ili učitavanje datoteke u novi memorijski spremnik, ne dirajući sadržaj trenutnog memorijskog spremnika. emacs vam omogućuje i umetanje sadržaja datoteke u trenutni memorijski spremnik.

Za zamjenu sadržaja trenutnog memorijskog spremnika sa sadržajem druge datoteke pritisnite <Ctrl-x><Ctrl-v>. emacs će vas u mini-spremniku upitati za ime datoteke. Ako se ne sjećate cijelog imena datoteke ili je ime prilično dugačko, možete koristiti mogućnosti dovršavanja imena u emacsu. Kada vas emacs upita za ime datoteke, možete unijeti samo prvih nekoliko slova u imenu i zatim pritisnuti <Tab>. emacs će to ime zatim proširiti prema imenu bilo koje datoteke s takvim početkom. Ako pronađe više datoteka čija imena odgovaraju upisanim slovima, emacs će prikazati prozor sa svim datotekama koje odgovaraju znakovima koje ste upisali i omogućit će vam odabir željene datoteke. Za učitavanje datoteke u novi memorijski spremnik pritisnite <Ctrl-x><Ctrl-f>. U naredbeni redak mini-spremnika upišite ime datoteke. emacs će memorijski spremnik nazvati prema imenu datoteke, međutim, ime memorijskog spremnika možete i promijeniti - pritisnite <Esc><x>, upišite novo ime za memorijski spremnik i pritisnite <Return>. emacs će vas upitati za novo ime. Upišite novo ime memorijskog spremnika i pritisnite <Return>. Statusni redak će prikazivati novo ime.

Za umetanje datoteke u trenutni memorijski spremnik naprosto kursor pomaknite na željeno mjesto u datoteci i pritisnite <Ctrl-x><i>.

#### Pozicioniranie kursora

Pri uređivanju teksta kursor trebate postaviti na mjesto gdje želite umetnuti dodatni tekst, izbrisati tekst, ispraviti greške, promijeniti riječi ili dodati tekst na kraj postojećeg teksta. Naredbe koje unosite nazivaju se naredbe za pozicioniranje kursora (cursor-positioning commands).

**Tipke sa strelicama** Za postavljanje kursora na željeno mjesto možete koristiti tipke sa strelicama na mnogim, ali ne svim sustavima. Lako je ustanoviti da li su tipke sa strelicama upotrebljive. Pokrenite emacs s postojećom datotekom i pogledajte kakvo djelovanje imaju tipke sa strelicama. Uz njih možete koristiti i tipke <Page Up> i <Page Down>.

Sljedeću naredbu unesite radi stvaranja nove datoteke nazvane emacs-pract.3 koja sadrži popis datoteka i direktorija u direktoriju /usr. Ovu datoteku možete koristi za isprobavanje naredbi za pomicanje kursora.

ls /usr > emacs-pract.3

Ako datoteku stvarate kombinacijom/slijedom tipki <Ctrl-u><Esc><!>, ugledat ćete sljedeću poruku:

(Shell command completed with no output)

BILJEŠKA Ne brinite se zbog ove poruke. Ona ne znači neku poteškoću. Uobičajeni je izlaz preusmjeren u datoteku i emacs tako nema izlaza koji bi uhvatio u memorijski spremnik.

Kada je datoteka stvorena, pokrenite emacs s datotekom emacs-pract.3 (utipkajte emacs emacs**pract.3** i pritisnite <Return>). Sada isprobajte uporabu tipki sa strelicama i tipki <Page Up> i <Page Down> (ako one postoje na vašoj tipkovnici) za pomicanje kursora po sadržaju memorijskog spremnika koji uređujete. Ako tipke rade, vjerojatno ćete ih željeti koristiti za pomicanje kursora.

Može se dogoditi da, premda izgleda da tipke za pomicanje kursora rade, one istovremeno u datoteku unose čudne znakove. Ti su znakovi kodovi koje računalo koristi za predstavljanje različitih tipki umjesto samih znakova. Ako ugledate takve znakove, za pomicanje kursora biste umjesto uporabe samih tipki na tipkovnici trebali koristiti različite naredbe preko tipkovnice.

SAVJET Za čišćenje zaslona od neželjenih i neobičnih znakova u emacsu pritisnite < Ctrl-I>.

**Druge tipke za pomicanje kursora** Kursor u emacsu možete pomicati i na druge načine, bez uporabe tipki sa strelicama. Ove biste načine trebali dobro upoznati u slučaju da tipke sa strelicama ne želite ili ne možete koristiti. Ovdje ćete saznati i načine za učinkovitije pomicanje kursora od uporabe tipki sa strelicama.

1975. godine, kada je emacs razvijen, mnogi terminali nisu imali tipke sa strelicama - za pomicanje su se kursora koristile druge tipke, koje se još uvijek koriste. Za ugodan je rad s tim tipkama potrebno malo vježbe, međutim, neki ih iskusniji korisnici emacsa koriste radije nego tipke sa strelicama. Ovdje je navedeno nekoliko drugih tipki za pomicanje kursora:

- Ctrl-f> pritisnite za pomicanje kursora u desno za jedno mjesto ("forward").
- Ctrl-b> pritisnite za pomicanje kursora za jedno mjesto u lijevo ("back").
- <Ctrl-n> pritisnite za pomicanje na početak sljedećeg reda, sačuvavši vaš položaj u retku ("next").
- <Ctrl-p> pritisnite za pomicanje na prethodni red, sačuvavši vaš položaj u retku ("previous").
- Ctrl-a> pritisnite za pomicanje na početak retka.
- <Ctrl-e> pritisnite za pomicanje na kraj retka ("end").

Neke vam emacs naredbe omogućuju postavljanje kursora u odnosu na riječ u retku. *Riječ* je definirana kao slijed znakova koji su od drugih znakova odvojeni razmacima ili znakovima interpunkcije, poput točaka, upitnika, zareza i povlaka. To su sljedeće naredbe:

- <Esc><f> pritisnite za pomicanje unaprijed za jednu riječ.
- <Esc><b> pritisnite za pomicanje unatrag za jednu riječ.

Neke su od tih radnji prikazane u sljedećem primjeru. Za pokretanje emacsa i otvaranje datoteke emacs-pract.1 utipkajte **emacs emacs-pract.1** i pritisnite <Return>. Sada kursor, koji je prikazan kao podvlaka, uporabom upravo opisanih naredbi za pomicanje kursora pomaknite na v u riječi "Razvrstati" u trećem retku datoteke. Treći će redak izgledati ovako:

b. b. Razvrstati prodajne podatke i ispisati rezultate.

Za pomicanje na početak sljedeće riječi pritisnite  $\langle \text{Esc} \rangle \langle f \rangle$  - kursor će se postaviti ispod p u riječi "prodajne" u prethodnoj rečenici. Pritisnite  $\langle \text{Esc} \rangle \langle f \rangle$  za pomicanje na p u "podatke". Za pomicanje na početak riječi "prodajne" pritisnite  $\langle \text{Esc} \rangle \langle b \rangle$  - kursor će se ponovno postaviti ispod p u riječi "prodajne".

**Tipke za pomicanje velikim koracima** Ako se kroz datoteku želite kretati svaki put za visinu zaslona, što je učinkovitije od pritiska na <Page Down>, koristite naredbe za kretanje kroz datoteku. Naredba <Ctrl-x><]> vas pomiče unaprijed za jednu stranicu. Za kretanje kroz tekst u velikim koracima koristite sljedeće naredbe:

- <Ctrl-v> pritisnite za pomicanje unaprijed za jedan zaslon.
- <Esc><v> pritisnite za pomicanje unatrag za jedan zaslon.
- Ctrl-x><]> pritisnite za pomicanje unaprijed za jednu stranicu.
- <Ctrl-x><[> pritisnite za pomicanje unatrag za jednu stranicu.

Za brzo pomicanje na posljednji redak u datoteci ili memorijskom spremniku pritisnite <Esc><Shift-.>. Za pomicanje na prvi redak datoteke pritisnite <Esc><Shift-,>. U stvari, za pomicanje na točno određeni red u memorijskom spremniku utipkajte naredbu **gotoline** *n*, gdje je *n* broj reda na koji želite prijeći. Za pomicanje u 35. red datoteke (ako 35. red postoji) pritisnite <Esc>, upišite **goto-line 35** i pritisnite <Return>.

Svaku naredbu možete ponoviti pritiskom na <Esc-*n*>, gdje *n* označava koliko puta tu naredbu želite ponoviti, i nakon toga unesite naredbu koju želite ponoviti.

Za pozicioniranje je kursora uporabom naredbi opisanih u prethodnim člancima potrebno malo ježbe. Upamtite da za izdavanje naredbi za pomicanje kursora trebate biti u načinu rada za izdavanje naredbi.

#### Dodavanje teksta

Za dodavanje teksta u memorijski spremnik editora trebate kursor postaviti na mjesto koje ste odredili za početak teksta koji dodajete. Svi se uobičajeni tekstualni znakovi koje utipkavate zatim dodaju u memorijski spremnik. Ako pritisnete <Return>, emacs "otvara" ili dodaje jedan red u memorijski spremnik. Prije nego počnete dodavati tekst, kursor postavite na mjesto na koje taj tekst želite dodati i zatim počnite s tipkanjem teksta.

Za dodavanje retka teksta ispod trenutnog retka koristite naredbu <Ctrl-o>. Ona u memorijskom spremniku "otvara" novi red i omogućuje vam dodavanje teksta. U sljedećem primjeru dodajete red u neki postojeći tekst.

Prije:

Sav je posao zavr{en molim te, nazovi ako imaš kakva pitanja.

Kursor je u drugom redu. Pritisnite <Ctrl-o> za dodavanje jednog ili više redova ispod tog retka i zatim utipkajte sljedeće:

Lee Nashua 555-1837

#### Poslije:

```
Sav je posao završen
molim te, nazovi
Lee Nashua
555-1837
ako imaš kakva pitanja.
```

Premda ste dodali samo dva reda, mogli ste ih dodati i više, pritiskom na <Return> na kraju svakog retka. Bez pritiska na <Return> biste, naravno, mogli dodati samo jedan red.

#### Brisanje teksta

Vršenje ispravki i preinaka teksta može uključivati i brisanje teksta. U emacsu možete izbrisati znak, riječ, određeni broj uzastopnih riječi, sav tekst do kraja retka ili cijeli red. Budući da je emacs vizualni editor poput editora vi, kada znakove, riječi ili redove izbrišete, oni nestaju i sa zaslona.

U tablici 28.1 je popis naredbi za brisanje i opis njihovog djelovanja. One sve djeluju s trenutnog položaja kursora. Znači da kursor trebate pomaknuti do znaka, riječi ili retka koji želite izmijeniti i zatim izdati željenu naredbu za brisanje. Vježbajte njihovu uporabu da vidite kakav imaju učinak. Uvidjet ćete da su za ispravljanje datoteka korisne.

Tablica 28.1 Nai	redbe za brisanje teksta
Kombinacija tipki	Djelovanje
<ctrl-d></ctrl-d>	Briše znak na položaju kursora
< <u>Esc</u> >< <u>d</u> >	Briše riječ u kojoj je kursor
<ctrl-k></ctrl-k>	Brisanje od položaja kursora do kraja reda
<esc><k></k></esc>	Briše rečenicu u kojoj je kursor
<ctrl-w></ctrl-w>	Briše označeno područje (za naredbe označavanja teksta pogledajte Označeni tekst u tablici 28.4)

Ako koristite naredbu <Ctrl-k>, informacija ne nestaje odmah u košari za bitove. Izbrisani se znakovi dodaju u *pričuvni spremnik* iz kojeg ih s naredbom <Ctrl-y> u bilo koje vrijeme možete istresti natrag.

#### Traženje i zamjena teksta

Pronalaženje bi riječi, izraza ili broja u datoteci bilo naporno kada biste svaki redak morali sami provjeriti. Poput većine editora i programa za obradu teksta, emacs ima naredbe koje vam omogućuju traženje niza znakova i, ako želite, njihovu zamjenu s drugim znakovima. Možete tražiti prema naprijed ili unatrag od vašeg trenutnog položaja u memorijskom spremniku. Traženje također možete ponoviti ili nastaviti. Kada s traženjem stigne do kraja datoteke u memorijskom spremniku, emacs će započeti od početka i obratno. Popis naredbi za traženje je prikazan u tablici 28.2. U svakom sluačju emacs će tražiti niz znakova koji ste zadali, u smjeru koji ste zadali i kursor će postaviti na početak pronađenog niza znakova.

Tablica 28.2 Nareda	e za traženje i zamjenu
Naredba	Djelovanje
<ctrl-s></ctrl-s>	Traži prema naprijed od trenutnog položaja
<ctrl-r></ctrl-r>	Traženje unatrag od trenutnog položaja
<ctrl-x><s></s></ctrl-x>	Ponavljanje traženja prema naprijed
<ctrl-x><r></r></ctrl-x>	Ponavljanje traženja unatrag
<esc><r></r></esc>	Zamjena svih instanci prvog utipkanog niza znakova u mini-spremniku s drugim utipkanim nizom znakova, završavajući svaki niz znakova s <esc></esc>
<esc><ctrl-r querry=""></ctrl-r></esc>	Prije izvršavanja zamjene, u mini-spremniku odgovorite s jednim od sljedećeg: <ctrl-g>: odustajanje od radnje <!-- -->: zamjena preostalog</ctrl-g>
	: dobivanje popisa s opcijama nastavlja se

Tablica 28.2	Nastavak
Naredba	Djelovanje
	<.>: zamjena i vraćanje na mjesto od kojeg je naredba pokrenuta <,>: zamjena i čekanje bez pitanja <y>: ili razmaknica: zamjena i nastavak s operacijom zamjene <n>: trenutna se instanca ne mijenja, ali se nastavlja s operacijom zamjene</n></y>

**Traženje** Kada utipkate naredbu za traženje, ona će se pojaviti u mini-spremniku. Za traženje prema naprijed niza znakova prodaja > 100K u datoteci, koristite ovu naredbu:

<Ctrl-s>prodaja > 100K

Ova naredba pokreće kontinuirano traženje kroz memorijski spremnik. Uočite da pri utipkavanju niza znakova za traženje emacs postavlja kursor na slijed znakova. Ako emacs ne pronađe zadani tekst, prikazat će poruku search failed. Ako se taj niz znakova nalazi u memorijskom spremniku, emacs će kursor postaviti ispod slova p u riječi prodaja. Kada pronađete prvu pojavu zadanog niza znakova, za zaustavljanje traženja trebate pritisnuti <Esc>, inače će emacs nastaviti s traženjem dok unosite drugi tekst. emacs takvu vrstu traženja naziva *kontinuirano traženje (incremental searches)* - emacs traži već pri unošenju niza znakova.

emacs može izvršavati i nekontinuirana traženja ako prije upisivanja niza znakova za traženje pritisnete tipku <Esc> i na kraju pritisnete <Return>, kako je ovdje prikazano:

<Ctrl-s><Esc>prodaja > 100K

Ako pretražujete veliku datoteku i shvatite da ste za traženje unijeli pogrešan niz znakova, emacs će pretraživati cijelu datoteku. Za zaustavljanje traženja pritisnite <Ctrl-g>.

#### RJEŠAVANJE POTEŠKOĆA

Utipkao sam niz znakova za koji znam da u datoteci postoji, ali ga emacs ne može pronaći. Najčešći je uzrok za ovu grešku netočan upis niza znakova. emacs - i računala općenito - nije baš dobar u razmišljanju - on nikako neće uspjeti shvatiti što ste zapravo mislili. Ako tražite niz znakova "vegi-burger", ali utipkate "vigi-burger", emacs ne može pronaći što želite (osim ako ste pogriješili i pri upisivanju u memorijski spremnik i upisali vigi umjesto vegi). Prije nego pritisnete <Return> pažljivo provjerite niz znakova za traženje.

Zamjenjivanje Premda vam traženje teksta može pomoći u pronalaženju određene riječi ili dijela teksta, mnogo puta pronađeni tekst želite i zamijeniti. Na primjer, ustanovili ste pravopisnu grešku koju stalno ponavljate i umjesto da ispravljate jednu po jednu riječ, istovremeno želite ispraviti cijeli memorijski spremnik. Ako, na primjer, svaku pojavu riječi "softwer" želite zamijeniti sa "softver", pritisnite <Esc><r>. Mini-spremnik će zatražiti niz znakova za traženje - unesite softwer. emacs će vas zatim zatražiti niz znakova za zamjenu - unesite softver. Program će zatim pregledati datoteku, tražeći niz znakova "softwer"

i zamjenjujući ga sa "softver". emacs će pritom strogo poštovati mala i velika slova. Tako, ako se "softwer" pojavljuje kao "Softwer", emacs će ga zamijeniti sa "Softver".

Možda ne želite zamijeniti svaku pojavu niza znakova za traženje - u tom ćete slučaju emacs uputiti da vam prije zamjene niza znakova postavi upit. Ako želite da vam emacs prije zamjene postavi upit, pritisnite <Esc><Ctrl-r>.

Na primjer, ako ime vašeg operativnog sustava, Linux, na nekim mjestima u tekstu želite zamijeniti s imenom njegovog prethodnika, UNIX, pritisnite <Esc><Ctrl-r>. emacs će u mini-spremniku postaviti upit Query replace:. Sada unesite niz znakova za traženje - Linux. Pojavit će se zahtjev Query replace Linux with:. Unesite niz znakova za zamjenu - UNIX. emacs će započeti traženje i prikazati - Query replacing Linux with UNIX. Ako ovu operaciju traži-i-zamijeni želite zaustaviti, pritisnite <Ctrl-g>.

Kada emacs u tekstu pronađe riječ Linux, na njoj će se zaustaviti i postaviti vam upit za daljnje djelovanje. Mogući su sljedeći odgovori:

Kombinacija tipki	Djelovanje
<ctrl-g></ctrl-g>	Odustajanje od zamjene
	Zamjena preostalih riječi bez postavljanja upita
	Prikaz popisa s opcijama
<.>	Zamjena trenutne instance i prestanak traženja
<,>	Zamjena trenutne instance, ali bez prelaska na sljedeću
<y> ili razmaknica</y>	Zamjena i nastavak s operacijom zamjene
<n></n>	Trenutna se instanca ne dira, ali se nastavlja s operacijom zamjene

**Mijenjanje teksta** Druga zadaća uređivanja s kojom se često suočavate je mijenjanje teksta ili zamjena jednog tekstualnog niza znakova s drugim (između ovih dviju radnji nema velike razlike). Naredbe za zamjenu koristite za zamjenu jednog znaka ili slijeda znakova. Naredbe za mijenjanje znakova možete koristiti i za ispravljanje jedne od najčešćih grešaka pri tipkanju zamjena mjesta dvaju susjednih slova. Tablica 28.3 prikazuje naredbe za mijenjanje znakova.

Tablica 28.3 Na	redbe za mijenjanje
Kombinacija tipki	Djelovanje
<ctrl-t></ctrl-t>	Zamjena mjesta dvaju susjednih slova
<esc><t></t></esc>	Zamjena mjesta susjednih riječi
<ctrl-x><ctrl-t></ctrl-t></ctrl-x>	Zamjena mjesta susjednih redaka
<esc><c></c></esc>	Pravilno postavljanje velikog slova na početak riječi
<esc><l></l></esc>	Pretvaranje svih slova u riječi u mala slova
<esc><u></u></esc>	Pretvaranje svih slova u riječi u velika slova

Te su promjene vezane za trenutni položaj kursora. Prije uporabe ovih naredbi kursor postavite na mjesto u memorijskom spremniku koje želite ispravljati.

#### Kopiranje, isijecanje i uljepljivanje

Kada kopirate ili odsijecate znakove, riječi, redove ili dijelove retka, izbrisani se objekt sprema u takozvani *pričuvni spremnik (kill-buffer)*. Ime nije toliko važno, već je važno da sadržaj tog spremnika možete umetnuti ili ulijepiti na bilo koje mjesto u tekstu koji uređujete. To činite s naredbom *yank* (izbaciti) - <Ctrl-y>. Naredba <Ctrl-y> objekt uljepljuje desno od ili iza položaja kursora.

Sljedeći primjer prikazuje uporabu <Ctrl-y> za uljepljivanje sadržaja pričuvnog spremnika iza kursora:

Prije:

Carefully carry these out instructions.

Riječ "out" i razmak izbrišite pritiskom na <Esc-d>. Zatim kursor pomaknite na razmak iza *y* u "carry" i pritisnite <Ctrl-y>.

Poslije:

Carefully carry out these instructions.

Za kopiranje slijeda od četiri reda na drugi dio teksta najprije ta četiri reda teksta trebate označiti, izbrisati ih u pričuvni spremnik i zatim ih izvaditi natrag na odgovarajuće mjesto. Slijedite ove korake:

- 1. Kursor postavite na početak prvog od ta četiri reda.
- 2. Pritisnite <Ctrl-razmaknica> za postavljanje oznake.
- Kursor pomaknite na kraj četvrtog retka. Time je stvoreno označeno područje koje emacs naziva region.
- 4. Označeni tekst izbrišite pritiskom na <Ctrl-w>.
- Budući da te redove želite kopirati, izbrisani tekst trebate vratiti. To učinite s naredbom <Ctrl-y>.
- 6. Kursor pomaknite na mjesto u memorijskom spremniku na koje tekst želite kopirati.
- Pritisnite <Ctrl-y> za uljepljivanje kopiranih redova ispod retka u kojem se nalazi kursor.

#### RJEŠAVANJE POTEŠKOĆA

Izdao sam naredbu za brisanje označenog područja, ali područje koje sam označio nije

**izbrisano.** Na žalost, GNU emacs koji se isporučuje s Linuxom ne pokazuje nikakvu vrstu oznaka pa je oznake lako zaboraviti postaviti ili ih postaviti na neodgovarajuća mjesta. Za provjeru položaja oznake koristite naredbu <Ctrl-x><Ctrl-x>. Ta naredba zamjenjuje položaj kursora i oznake. Ako se kursor pomakne na mjesto na kojem mislite da je oznaka postavljena, tada ste sigurni da ste oznaku ispravno postavili. Za vraćanje kursora i oznake na ispravna mjesta ponovno izdajte naredbu <Ctrl-x><Ctrl-x>.

# Pregled osnovnih naredbi

Tablica 28.4 vam pruža sažet popis glavnih naredbi raspoloživih u emacsu. <Esc><c> znači da pritisnete i otpustite *meta tipku* - na PC tipkovnici je to obično tipka <Esc>, premda na nekim tipkovnicama možete koristiti tipku <Alt> - i zatim pritisnete sljedeću, tipku <c>. <Ctrl-c> znači za pritisnete i zadržite tipku <Ctrl> i istovremeno pritisnete tipku <c>. Upamtite da pritisak na <Ctrl-g> u bilo koje vrijeme zaustavlja izvršavanje trenutne naredbe.

Tablica 28.4 Osn	ovne naredbe u emacsu
Kombinacija tipki	Opis
Spremanje na disk	
<ctrl-x><ctrl-s></ctrl-s></ctrl-x>	Sadržaj trenutnog memorijskog spremnika sprema na disk
<ctrl-x><ctrl-w></ctrl-w></ctrl-x>	Trenutni memorijski spremnik sprema na disk, pitavši za ime datoteke
<ctrl-x><n></n></ctrl-x>	Mijenja ime datoteke trenutnog memorijskog spremnika
<esc><z></z></esc>	Spremanje svih preinačenih memorijskih spremnika na disk i izlazak iz emacsa
Učitavanje s diska	
<ctrl-x><ctrl-f></ctrl-f></ctrl-x>	Pronalazi datoteku i učitava je u novi memorijski spremnik stvoren prema imenu datoteke
<ctrl-x><ctrl-r></ctrl-r></ctrl-x>	Datoteku učitava u trenutni memorijski spremnik, izbrisavši prethodni sadržaj
<ctrl-x><ctrl-i></ctrl-i></ctrl-x>	Umetanje datoteke u trenutni memorijski spremnik na položaj kursora
Pomicanje kursora	
<ctrl-f></ctrl-f>	Pomicanje prema naprijed za jedan znak
<ctrl-b></ctrl-b>	Pomicanje unatrag za jedan znak
<ctrl-a></ctrl-a>	Pomicanje na početak trenutnog retka
<ctrl-e></ctrl-e>	Pomicanje na kraj trenutnog retka
<ctrl-n></ctrl-n>	Pomicanje u sljedeći redak
<ctrl-p></ctrl-p>	Pomicanje u prethodni redak
<esc><f></f></esc>	Pomicanje prema naprijed za jednu riječ
<esc><b></b></esc>	Pomicanje unatrag za jednu riječ
<esc><a></a></esc>	Prelazak u zadani redak
<esc><shift></shift></esc>	Pomicanje na početak memorijskog spremnika
<esc><shift-,></shift-,></esc>	Pomicanje na kraj memorijskog spremnika
Brisanje i umetanje	
<ctrl-d></ctrl-d>	Brisanje sljedećeg znaka

nastavlja se

Tablica 28.4 Nasta	vak
Kombinacija tipki	Opis
Brisanje i umetanje	
<ctrl-c></ctrl-c>	Umetanje razmaka
<esc><d></d></esc>	Brisanje sljedeće riječi
<ctrl-k></ctrl-k>	Brisanje do kraja trenutnog retka
<return></return>	Umetanje novog retka
<ctrl-j></ctrl-j>	Umetanje novog retka i uvlake (indenta)
<ctrl-o></ctrl-o>	Otvaranje novog retka
<ctrl-w></ctrl-w>	Brisanje područja između oznake i kursora
<esc><w></w></esc>	Kopiranje područja u pričuvni spremnik
<ctrl-x><ctrl-o></ctrl-o></ctrl-x>	Brisanje redaka oko kursora
Traženje i zamjenjivanje	9
<ctrl-s></ctrl-s>	Traženje prema naprijed od trenutnog položaja
<ctrl-r></ctrl-r>	Traženje unatrag od trenutnog položaja
<ctrl-x><s></s></ctrl-x>	Ponavljanje traženja prema naprijed
<ctrl-x><r></r></ctrl-x>	Ponavljanje traženja unatrag
<esc><r></r></esc>	Zamjena svih instanci prvog utipkanog niza znakova u mini-spremniku s drugih utipkanim nizom znakova, završavajući svaki niz znakova s <esc></esc>
<esc><ctrl-r></ctrl-r></esc>	Upiti prije izvršavanja zamjene. U mini-spremniku odgovorite s jednim od sljedećeg: <ctrl-g>: odustajanje od radnje <!-- -->: zamjena preostalog <? >: dobivanje popisa s opcijama &lt;.&gt;: zamjena i vraćanje na mjesto od kojeg je naredba pokrenuta <y>: ili razmaknica: zamjena i nastavak s operacijom zamjene <n>: trenutna se instanca ne mijenja, ali se nastavlja s operacijom zamjene</n></y></ctrl-g>
Označeni tekst	
<ctrl><razmaknica></razmaknica></ctrl>	Postavlja oznaku na trenutni položaj kursora
<ctrl-x><ctrl-x></ctrl-x></ctrl-x>	Zamjena oznake i kursora
<ctrl-w></ctrl-w>	Brisanje označenog područja
<esc-w></esc-w>	Kopiranje označenog područja u pričuvni spremnik
<ctrl-y></ctrl-y>	Umetanje pričuvnog spremnika na trenutni položaj kursora

Kombinacija tipki	Opis					
Memorijski spremnici (k	pufferi)					
<ctrl-x><b></b></ctrl-x>	Prebacivanje na drugi memorijski spremnik					
<ctrl-x><x></x></ctrl-x>	Prebacivanje na sljedeći memorijski spremnik u popisu memorijskih spremnika					
<esc><ctrl-n></ctrl-n></esc>	Promjena imena trenutnog memorijskog spremnika					
<ctrl-x><k></k></ctrl-x>	Brisanje memorijskog spremnika koji nije prikazan					

# Podešavanje emacsa

Vašu inačicu emacsa možete prilagoditi stavljanjem posebnih funkcija u datoteku nazvanu .emacs. Ta se datoteka mora nalaziti u vašem polaznom direktoriju. Ona sadrži funkcije napisane u emacs LISP-u, s kojima emacs možete prilagoditi vlastitim željama. Slijedi primjer LISP funkcije:

```
(keyboard-translate ?\C-h ?\C-?)
```

Ova je funkcija korisna ako vaš terminal tipku <Backspace> prevodi u znakove <Ctrl-h>. Prema unaprijed zadanom podešenju, ta se kombinacija tipki koristi za pozivanje pomoći unutar emacsa. Određivanjem nove funkcije i povezivanjem te funkcije s tipkom možete podesiti kako će se emacs odazivati na kombinacije tih tipki.

U prethodnom primjeru, ?\C-h predstavlja pritisak na tipke <Ctrl-h>. ?\C-? predstavlja tipku <Delete>. Gotovo na svim ASCII tipkovnicama obje tipke predstavljaju istu ASCII vrijednost, to jest 8. Nakon što u vašu datoteku .emacs umetnete redak s ovom funkcijom i spremite je, kada sljedeći put pozovete emacs moći ćete znakove brisati uporabom tipke <Backspace>.

Naravno, to također znači da preko tipkovnice više nećete imati pristup sustavu pomoći. Za ublažavanje ove poteškoće možete funkciju sustava pomoći povezati s novom kombinacijom tipki, kao što ste to napravili s funkcijom brisanja. Naprosto u vašu datoteku .emacs dodajte sljedeći redak, s tim da umjesto *key* stavite vašu odabranu tipku:

```
(keyboard-translate ?\C-key ?\C-h)
```

# Odavde...

Više informacija o drugom editoru i o datotečnom sustavu u Linuxu možete naći u sljedećim poglavljima:

8. poglavlje, "Uporaba vi editora " govori o osnovama uporabe ovog omiljenog editora. vi je važan budući da se nalazi na svim Linux/UNIX sustavima. Ako znate kako koristiti vi, trebali biste znati kako urediti datoteku na bilo kojem sustavu. Administratori sustava također koriste vi za mnoge zadaće upravljanja sustavom.

#### 558 28. poglavlje Uporaba editora emacs

- 11. poglavlje, "Sigurnosna pohrana podataka" vam pokazuje kako ispravno zaštiti vaše tekstualne datoteke od slučajnog brisanja.
- 16. poglavlje, "Razumijevanje sustava datoteka i direktorija" govori o osnovama u vezi datoteka i direktorija. Za uporabu emacsa ili bilo kojeg drugog editora trebate poznavati osnove datotečnog sustava. Pri stvaranju i preinačavanju datoteka u editoru na vama je da im dajete odgovarajuća imena i stavljate ih u odgovarajuće direktorije.
- 20. poglavlje, "Ispisivanje" pruža informacije o ispisivanju vaših tekstualnih datoteka pod Linuxom. Ispisivanje datoteka pod Linuxom može skrivati mnoge zamke. Ovo vam poglavlje pomaže u pripremanju vašeg sustava za ispisivanje.

# P 0 G L A V L J E **29**

# Uporaba SLIP-a i PPP-a

(Steve Burnett)

#### U ovom poglavlju

Razumijevanje zahtjeva za SLIP i PPP Uporaba *dipa* za automatiziranje SLIP operacija Uporaba *diplogina* za pružanje SLIP usluga Uporaba PPP-a Jezgra Linuxa podržava dva serijska protokola za prijenos IP (Internet Protocol) prometa: SLIP (Serial Line Internet Protocol) i PPP (Point-to-Point Protocol). Ti su protokoli razvijeni kao alternativa siromašnih ljudi umjesto skupih postava za povezanost s Internetom preko iznajmljenih linija. Svatko s modemom razumno visoke brzine i davateljem pristupa koji podržava te protokole može na svom Linux stroju dobiti IP vezu za vrlo nisku cijenu u usporedbi sa sustavima preko iznajmljene linije. SLIP upravljački programi za Linux su bili raspoloživi uskoro nakon prvog izdanja Linuxa, a podrška za PPP je dodana uskoro nakon toga. Premda je PPP danas prevladavajući na ovom polju, SLIP konfiguracija je još uvijek korisna kao osnova.

# Razumijevanje zahtjeva za SLIP i PPP

Potrebno je osigurati podešenja nekih pojedinosti u vašoj Linux jezgri ili konfiguracijskim datotekama. TCP/IP umreženje treba biti omogućeno i potrebno je podesiti sučelje s povratnom petljom.

Vidi "Podešavanje softverskog povratnog sučelja", str. 448.

Poželjno je da IP adresa vašeg DNS (Domain Name Service) poslužitelja bude uključena u vašu datoteku /etc/resolv.conf radi prikladnijeg pristupa i drugim strojevima osim vašeg domaćina za biranu vezu. Ako je vaša birana veza spora ili sklona smetnjama, možda ćete u vašoj Linux kutiji željeti pokrenuti i poslužitelj imena za privremeno spremanje DNS popisa i smanjenja količine DNS IP prometa na vašoj biranoj vezi.

- Vidi "Datoteka /etc/resolv.conf", str. 505.
- Vidi "Uporaba demona named za podešavanje poslužitelja", str. 506.

# Uporaba dipa za automatiziranje SLIP operacija

Linux pruža brojne programe za upravljanje vašim SLIP operacijama. Upravljački program dip (Dial-Up IP Protocol) je jedan od najsvestranijih alata. On pruža jezik za pisanje skripti za automatiziranje upravljanja modemom i automatski podešava SLIP mrežno sučelje i *Kernel routing* tablice. dip možete koristiti za pokretanje SLIP veza ili pružanje biranih SLIP usluga drugim strojevima. Sintaksa za dip je sljedeća:

```
dip [-tvi] [-m mtu] [datotekaskripte]
```

Tablica 29.1 opisuje dipove najčešće argumente naredbenog retka.

Tablica 29.1	uobičajeni argumenti naredbenog retka za dip
Argument	Opis
- a	Od korisnika traži upis korisničkog imena i lozinku.
-t	Pokreće dip u načinu rada s naredbenim retkom. Način rada s naredbenim retkom vam pruža puni pristup do svega što dip može uraditi, omogućujući vam ručno pokretanje SLIP veza.
- V	Koristi se s -t za prikaz trenutne greške.

Argument	)pis	
- i	dipu naređuje ulazni način rada. Ova se zastavica koristi kada dip pruža SLIP usluge za druge koji biranom vezom pristupaju vašem stroju.	
-m <i>mtu</i>	Prisiljava dip da koristi navedenu MTU vrijednost.	
datotekaskripte ( <i>scriptfile</i> )	Navodi ime dip skripte koju treba izvršiti.	

#### Uporaba dipa u načinu rada s naredbenim retkom

Pozivanje dipa s opcijom -t stavlja ga u način rada s naredbenim retkom. Taj vam način rada omogućuje izravno upravljanje s dipom i to je odličan alat za razvoj i ispravljanje dip skripta. Sljedeći vam primjer pokazuje kako dip izgleda u načinu rada s naredbenim retkom:

```
$ /sbin/dip -t
DIP: Dialup IP Protocol Driver version 3.3.7i-uri (17 Apr 95)
Written by Fred N. van Kempen, MicroWalt Corporation.
```

DIP>

Iz DIP> naredbenog retka možete pokrenuti bilo koju dip naredbu - samo je utipkajte i pritisnite <Return>. Naredba help prikazuje popis raspoloživih naredbi. Ako naredbu pozovete s nevaljanim argumentima, bit će prikazan kratki opis uporabe za tu naredbu. Tablica 29.2 opisuje naredbe raspoložive za uporabu u načinu rada s naredbenim retkom ili u dip skriptama.

Tablica 29.2         Naredbe raspoložive za dip		
Naredba	Opis	
chatkey <i>ključnariječ</i> [kôd]	Dodaje ključnu riječ i kôd greške skupu kodova grešaka koje vraća naredba dial. Naredba chatkey se može koristiti za otkrivanje kada vaš modem vraća BUSY, VOICE ili druge posebne greške.	
config [ <i>argumenti</i> ]	Omogućuje vam izravno upravljanje SLIP sučeljem koje vam dip pruža. Ova je naredba obično onemogućena budući da se s njome opasno ugrožava sigurnost. Za omogućavanje ove naredbe treba malo preinačiti datoteku izvornog koda command.c.	
databits bitova	Određuje broj bitova koji se mogu koristiti kao podaci u svakom bajtu u svrhu prilagodbe za 6- i 7-bitne veze.	
default	Nalaže dipu da odredi unaprijed zadani smjer u Kernel routing tablici usmjeren na udaljenog domaćina.	
dial num	Bira navedeni telefonski broj (num).	
echo on off	Uključuje i isključuje jeku. Ako stavite echo, on dip će prikazivati što šalje i prima iz modema.	

nastavlja se

Tablica 29.2 Nastavak		
Naredba Opis		
flush	Odbacuje sve odazive iz modema koji još nisu pročitani.	
get \$var	Određuje varijablu \$var ili kao konstantu ask ili kao navedenu udaljenu konstantu, od korisnika traži da unese vrijednost ili uzima sljedeću riječ iz serijske linije i dodijeli je varijabli \$var.	
goto <i>oznaka</i>	Prelazi na navedenu oznaku u dip skripti.	
help	Prikazuje popis raspoloživih naredbi u načinu rada s naredbenim retkom.	
if \$var op <i>broj</i>	Izvršava uvjetno grananje u skripti goto label. \$var mora biti jedna od \$errlvl, \$locip ili \$rmtip. Broj mora biti cijeli. Raspoloživi su sljedeći operatori koji imaju uobičajena značenja iz C jezika: ==, !=, <, >, <= i >=.	
init <i>initstring</i>	Postavlja incijalizacijski niz znakova koji se modemu šalje ponovnim određivanjem (resetiranjem) naredbe na <i>initstring</i> .	
mode SLIP CSLIP	Određuje protokol za vezu i uzrokuje da dip prijeđe u <i>daemon</i> način rada. Ova naredba obično uzrokuje prelazak dipa u <i>daemon</i> način rada bez vraćanja upravljanja na skriptu ili naredbeni redak DIP>.	
modem HAYES	Određuje vrstu modema. Sada je podržan samo HAYES modem (HAYES treba napisati s velikim slovima).	
netmask <i>maska</i>	Određuje mrežnu masku za smjerove koje dip instalira u maska.	
parity E O N	Određuje paritet serijske linije: <i>even</i> (paran), <i>odd</i> (neparan) ili <i>odd</i> (ne koristi se).	
password	Od korisnika traži unos lozinke koju vraća na siguran način. Ova naredba ne prikazuje znakove lozinke dok je upisujete.	
print	Ispisuje tekst na konzolu s koje je dip pokrenut. Varijable uključene u tekst su zamijenjene s njihovim vrijednostima.	
port <i>uređaj</i>	Određuje uređaj koji dip koristi.	
quit	Izlaženje iz programa dip.	
reset	Šalje incijalni niz znakova u serijsku liniju.	
send <i>tekst</i>	Navedeni tekst šalje u serijsku liniju. S uobičajenim se sekvencama s obrnutim kosim crtama u C-stilu ispravno barata.	
sleep <i>broj</i>	Obradu odgađa za navedeni broj sekundi.	
speed <i>broj</i>	Određuje brzinu serijske linije.	

Naredba	Opis
stopbits bitovi	Određuje broj zaustavnih bitova koje serijsko sučelje koristi.
timeout broj	Određuje unaprijed zadani istek vremena na cjelobrojnu vrijednost <i>num</i> - mjeri se u sekundama.
term	Nalaže dipu prelazak u način rada emulacije terminala. To vam omogućuje izravno sučelje na serijsku vezu. Pritisak na <ctrl-]> vraća vas u naredbeni redak DIP&gt;.</ctrl-]>
wait riječ broj	Čini da dip čeka dolazak navedene riječi na serijskoj liniji s istekom vremena za broj sekundi.

dip vam za uporabu pruža i brojne varijable. Neke, poput lokalnih i udaljenih IP adresa, možete sami odrediti, dok su druge samo za čitanje i koriste se za dijagnosticiranje i za informiranje. Na početku je svake varijable znak dolara, a ostatak treba napisati malim slovima. U tablici 29.3 je prikazan popis varijabli i njihove uporabe.

Tablica 29.3	/arijable koje vam pruža <i>dip</i>	
Varijabla	Opis	
\$local	Ime domaćina na lokalnom stroju.	
<pre>\$locip</pre>	IP adresa dodijeljena lokalnom stroju.	
<pre>\$remote</pre>	Ime domaćina na udaljenom stroju.	
\$rmtip	IP adresa udaljenog stroja.	
\$mtu	MTU vrijednost za vezu.	
\$modem	Korištena vrsta modema (samo-za-čitanje).	
\$port	Ime serijskog uređaja koji <i>dip</i> koristi (samo-za-čitanje).	
\$speed	Podešena brzina serijskog uređaja (samo-za-čitanje).	
\$errlvl	Kôd koji je nastao izvršenjem posljednje naredbe (samo-za-čitanje). Nula označava uspjeh, a sve druge vrijednosti znače grešku.	

SAVJET Stavljanje imena domaćina u varijablu \$local ili \$remote uzrokuje da dip ime domaćina pretvara u njegovu IP adresu i pohranjuje je u odgovarajuću varijablu za IP adresu. Time ste u skripti koju pišete uštedjeli jedan korak.

BILJEŠKA Varijable koje su samo za čitanje ne možete odrediti izravno uporabom naredbe get.

#### Uporaba dipa sa statičkim IP adresama

Dodjeljivanje je zasebnih IP adresa svakom stroju koji koristi SLIP davatelja usluga prilično uobičajeno. Kada vaš stroj pokrene SLIP vezu s udaljenim domaćinom, dip će SLIP sučelje podesiti s tom poznatom adresom. Ispis 29.1 prikazuje dip skriptu koja koristi statične IP adrese za pokretanje SLIP veze od linux2.burwell.com do linux1.burwell.com.

```
Ispis 29.1
              Ogledna dip skripta za uporabu statičnih IP adresa u SLIP vezi
# Connect linux2 to linux1 using static IP Addresses
# Configure Communication Parameters
port /dev/cual # use modem on /dev/cual serial line
speed 38400
modem HAYES
reset # Send initialization string to modem
flush # Throw away modem response
                                 # Set local IP address
get $local linux2
get $remote linux1
                                  # Set remote IP address
# Dial number for linux1 modem
dial 555-1234
if $errlvl != 0 goto error
                                # If the dial command fails, error out
wait CONNECT 75
if $errlvl != 0 goto error
                             # If we don't get, a CONNECT string
# from the modem, error out
send \r\n
                                 # Wake up login program
wait ogin: 30
                                # Wait 30 seconds for login prompt
if $errlvl != 0 goto error# Error out if we don't get login promptsend Slinux2\n# Send SLIP login name for linux2
wait ssword: 5
                                # Wait 5 seconds for password prompt
# Send password
send be4me\n
wait running 30
                                 # Wait for indication that SLIP is up
if $errlvl != 0 goto error # Otherwise error out
# We're in, print out useful information
print Connected to $remote with address $rmtip
                                  # Make this link our default route
default
mode SLIP
                                  # Turn on SLIP mode on our end
# Error routine in case things don't work
error:
print SLIP to $remote failed.
```

SAVJET Praćenje SLIP korisničkih računa može biti teško. Uobičajeno je da se UNIX korisnički računi prijavnim imenima dodjeljuju sa svim malim slovima. Uporabom imena klijent stroja s velikim S dodanim na početak, kao prijavnog imena za SLIP račun tog stroja, olakšava se njegovo praćenje i izbjegavaju se sukobi prijavnog imena s uobičajenim korisničkim računima.

Skripta u Ispisu 29.1 incijalizira modem i određuje lokalnu i udaljenu IP adresu za SLIP vezu. Ako ovdje koristite imena domaćina, dip će ih pretvoriti u njihove istoznačne IP adrese. Skripta zatim naziva modem i probija se kroz prijavnu sekvencu. Kada je prijavljena na udaljenom domaćinu i sigurna da je SLIP veza uspostavljena, skripta dipu prepušta podešavanje tablice smjerova i zatim serijsku liniju prebacuje u SLIP način rada.

Ako se dogodi greška, rutina za grešku na kraju skripte ispisuje poruku upozorenja i zaustavlja skriptu. dip je odličan za napuštanje serijske linije u prihvatljivom stanju kada s njom završi.

#### Uporaba dipa s dinamičkim IP adresama

Kako je SLIP postajao sve omiljeniji, zadaća je upravljanja s IP adresama za SLIP klijente postajala sve teža. Taj se problem pogoršao kada su u uporabu došli terminalski poslužitelji koji su podržavali SLIP. Sada vam može biti dodijeljena bilo koja iz niza IP adresa, ovisno na kojem je sučelju terminalski poslužitelj zaprimio vaš poziv. To je dovelo do promjena u dipu koji je iz dolazećih podataka na serijskoj liniji hvatao informaciju o IP adresi. Ispis 29.2 prikazuje dip skriptu koja iz serijske linije hvata lokalne i udaljene IP adrese.

```
Ispis 29.2
               Ogledna dip skripta za dinamičke IP adrese
# Connection script for SLIP to server with dynamic IP address
# assignment. The terminal server prints out:
#
# remote address is XXX.XXX.XXX.XXX the local address is YYY.YYY.YYY.YYY
# Set the desired serial port and speed.
port /dev/cual
speed 38400
# Reset the modem and terminal line.
Reset
flush
# Prepare for dialing.
dial 555-1234
if $errlvl != 0 goto error
wait CONNECT 60
if $errlvl != 0 goto error
# We are connected. Login to the system.
login:
wait name: 10
                                        # Log in to system
if $errlvl != 0 goto error
send Slinux2\n
                                        # Send user ID
wait ord: 10
if $errlvl != 0 goto error
send be4me∖n
                                        # Send password
if $errlvl != 0 goto error
get $remote remote 10
                                        # Get remote IP address
```

nastavlja se

```
Ispis 29.2
               Nastavak
if $errlvl != 0 goto error
                                        # Get local IP address
get $local remote 10
if $errlvl != 0 goto error
done:
print CONNECTED to $remote with address $rmtip we are $local
default
                                        # Set routing
                                        # Go to SLIP mode
mode SLIP
goto exit
error:
print SLIP to $host failed.
exit:
```

Skripta u ispisu 29.2 koristi get \$remote remote 10 za nadgledanje serijske linije i hvatanje prve informacije koja liči na IP adresu u varijablu \$remote. Naredba završava s greškom ako IP adresu ne ugleda u vremenu od 10 sekundi.

### Uporaba diplogina za pružanje SLIP usluga

Program dip automatizira pokretanje SLIP veza iz klijent stroja. Linux podržava i dolazeće birane SLIP veze. Za to je već raspoloživo nekoliko paketa. Ovdje ćete koristiti program diplogin, što je u stvari samo drugo ime za dip.

Davanje SLIP usluga drugima zahtijeva da u vašoj Linux kutiji stvorite poseban račun za svaku osobu i taj račun ispravno podesite. Trebat ćete napisati i datoteku /etc/diphosts s odgovarajućim informacijama o svakom domaćinu za koji pružate SLIP usluge.

#### Stvaranje SLIP korisničkih računa

SLIP korisnički račun možete ručno stvoriti uporabom skripte adduser s odgovarajućim odgovorima na svako pitanje. Ovdje je primjer stavke /etc/passwd za linux2.burwell.com u datoteci passwd na linux1.burwell.com:

Slinux2:IdR4gDZ7K7D82:505:100:linux2 SLIP Account:/tmp:/sbin/diplogin

Preporučljivo je da kao polazni direktorij za SLIP račune bude korišten /tmp radi povećanja sigurnosti sprečavanjem SLIP korisnika od upisivanja datoteka u osjetljiva područja vašeg datotečnog sustava prema unaprijed zadanim postavkama. Provjerite da li koristite ispravnu stazu za program diplogin.

#### Uporaba datoteke /etc/diphosts

Datoteka /etc/diphosts upravlja pristupom do SLIP-a na vašem stroju i sadrži parametre veze za svaki korisnički račun kojem je dopuštena uporaba SLIP-a. Ona sadrži retke koji izgledaju slično sljedećem:

Slinux2::linux2.burwell.com:linux2 SLIP:SLIP,296

Polja u ovoj datoteci su: identifikacijski broj korisnika, sekundarna lozinka, ime domaćina ili IP adresa stroja koji poziva, informacijsko polje koje se trenutno ne koristi i parametri

veze za ovaj račun. Polje parametara veze sadrži protokol (SLIP ili CSLIP) i MTU (Maximum Transmission Unit) vrijednost za ovaj korisnički račun.

Ako drugo polje nije prazno, diplogin će tražiti unos vanjske sigurnosne lozinke kada će se navedeni korisnički račun prijavljivati na vaš stroj. Ako odaziv iz udaljenog domaćina ne odgovara nizu znakova u ovom polju, pokušaj prijave se odbacuje.

#### **UPOZORENJE**

Program diplogin za preinačavanje *Kernel routing* tablice zahtijeva povlastice *root* korisnika. Ako ne izvršavate dip setuid root, ne možete koristiti vezu između *dipa* i *diplogina*. Trebate napraviti zasebnu kopiju *dipa* nazvanu *diplogin* i imati njegov setuid root.

To je sve što treba. Vaš ste sustav podešavanjem SLIP korisničkog računa i datoteke /etc/diphosts u potpunosti podesili za podršku dolazećim SLIP vezama.

#### **Uporaba PPP-a**

Point-to-Point Protocol (PPP) je još jedan protokol za slanje *datagrama* (podatkovnih paketa) preko serijske veze. Razvijen nakon SLIP-a, PPP sadrži brojne mogućnosti koje SLIP-u nedostaju. PPP može automatski pregovarati opcije poput IP adresa, veličine podatkovnih paketa i autorizacije klijenata. On također može prenositi pakete i iz drugih protokola osim IP-a.

#### Automatiziranje PPP veza s pppdom i chatom

PPP djeluje u dva dijela: upravljački program PPP u Linux jezgri i program nazvan pppd koji sam korisnik treba pokrenuti. Najosnovniji je način uporabe PPP-a ručno prijavljivanje na udaljenog domaćina uporabom komunikacijskog programa i zatim ručno pokretanje pppda na udaljenom i lokalnom domaćinu. Mnogo je lagodnije s pppdom koristiti chat skriptu koja barata s modemom, prijavljuje se na udaljenog domaćina i pokreće udaljeni pppd. Prije uranjanja u pppd, pogledajmo na brzinu chat.

**Uporaba programa chat** chat je program za automatiziranje međudjelovanja između vašeg računala i modema. Koristi se uglavnom za uspostavljanje modemske veze između lokalnog i udaljenog pppd *daemon* procesa. Sintaksa za chat je sljedeća:

chat [opcije] skripta

Tablica 29.4 prikazuje popis opcija naredbenog retka za program chat.

Tablica 29.4         opcije naredbenog retka za program chat	
Opcija	Opis
-f imedatoteke	Koristi chat skriptu u navedenoj datoteci.
-l lockfile	Stvara zaključanu datoteku prema UUCP stilu uporabom navedene zaključane datoteke.

nastavlja se

Tablica 29.4	Nastavak	
Opcija	Opis	
-t broj	Navedeni broj koristi kao vrijeme isteka u sekundama za svaki očekivani niz znakova.	
- V	Sve što chat šalje i prima bilježi se u syslog.	
script	Navodi koju chat skriptu treba koristiti.	

Ne možete istovremeno koristiti opciju - f i navesti chat skriptu - te se opcije međusobno isključuju. Ako za chat koristite opciju -l, s pppdom nemojte koristiti opciju lock, budući da zaključana datoteka koju je stvorio chat u pppdu uzrokuje grešku, jer on tada misli da je modem već u uporabi.

SAVJET Pri uklanjanju grešaka u chat skriptama, na jednoj virtualnoj konzoli pokrenite tail -f /var/adm/messages, a na drugoj pokrenite chat s opcijom -v. Zatim možete promatrati razgovor koji chat vodi kada dolazi na prvu virtualnu konzolu.

> Stvaranje chat skripti chat skripte se sastoje od jednog ili više parova nizova znakova očekivani tekst - odgovor, odvojenih razmacima. Program chat čeka očekivani tekst i kada ga primi šalje tekst odgovora. U očekivani se dio mogu uključiti neobavezni potparovi očekivani tekst-odgovor, odvojeni crticom.

Ovdje je uobičajena chat skripta za prijavljivanje na Linux stroj:

ogin:-\r\n-ogin: abbet1 word: costello

Ova skritpa kaže da chat treba čekati pojavu niza znakova ogin:. Ako chatu prije prijema ovog niza znakova istekne vrijeme, trebao bi poslati oznake za kraj retka i dodavanje novog retka i zatim ponovno čekati niz znakova ogin:. Kada chat ugleda niz znakova ogin:, pošalje abbet1, zatim očekuje word: i u odazivu šalje costello.

SAVJET U očekivanje niza znakova uključite samo tekst koji je neophodan za sigurno prepoznavanje onog što očekujete radi smanjenja mogućnosti za neuspjeh prepoznavanja i ispada skripte zbog iskrivljenog teksta. Na primjer, koristite ogin: umjesto login: i word: umjesto password:.

chat obično nakon svakog niza znakova u odgovoru šalje i oznaku za kraj retka, osim ako niz znakova završava slijedom znakova, a \c. Oznaka za kraj retka se ne traži u očekivanom nizu znakova osim ako je ona u očekivanom nizu znakova izričito zatražena slijedom znakova r.

Većina modema može izvijestiti zašto poziv nije uspio kada dobiju znak zauzetosti ili ne mogu pronaći noseću frekvenciju. Uporabom očekivanog niza znakova abort možete naložiti chatu da u slučaju primanja zadanog niza znakova ne učini ništa. Moguća je uporaba više parova za odustajanje. Sljedeća je skripta primjer uporabe očekivanog

abort niza znakova:

abort 'NO CARRIER' abort 'BUSY' ogin:-ogin: ppp word: be4me

Ova skripta uzrokuje odustajanje chata ako u bilo kojoj točci tijekom izvršavanja skripte primi signale NO CARRIER ili BUSY.

chat prepoznaje brojne znakovne i Escape sekvence, kako je prikazano u tablici 29.5.

Tablica 29.5	Znakovne i Escape sekvence koje chat prepoznaje	
Sekvenca	Opis	
BREAK	Koristi se kao niz znakova za odgovor. Nalaže chatu da modemu pošalje prekid. Ovaj poseban signal obično kod udaljenog domaćina uzrokuje promjenu brzine prijenosa.	
، ،	Šalje prazni niz s jednim znakom za vraćanje na početak reda.	
\b	Znak za Backspace.	
\c	Sprečava slanje novog reda nakon niza znakova za odgovor i mora biti na kraju niza znakova za odgovor.	
\d	Uzrokuje čekanja chata u trajanju od jedne sekunde.	
\ K	Još jedan način određivanja signala prekida.	
\n	Šalje znak za novi redak.	
\ N	Šalje NUL znak.	
\p	Stanka od 1/10 sekunde.	
/d	Sprečava da string u kojem je uključena bude prikazan u syslog datoteci.	
\r	Šalje ili očekuje znak za vraćanje na početak reda.	
\s	Šalje ili očekuje znak razmaka.	
\t	Šalje ili očekuje znak tabulatora.	
$\setminus \setminus$	Šalje ili očekuje znak obrnute kose crte.	
\ddd	Navodi ASCII znak u oktalnom sustavu.	
^C	Navodi kontrolni znak koji je predstavljen s C.	

SAVJET abort niz znakova možete koristiti za sprečavanje poziva male brzine na vašem modemu velike brzine. Vaš modem podesite da nakon uspostavljanja veze vrati niz znakova CARRIER 14400, a vašoj chat skripti dodajte abort CARRIER 2400. Na taj će način chat prekinuti vezu ako se vaš modem spoji na 2400 bps umjesto na 14400 bps.

Uporaba PPP-a s chatom Program pppd ima opcije naredbenog retka koje upravljaju svim pojedinostima PPP veze. Sintaksa je za pppd sljedeća:

```
pppd [opcije] [tty_ime] [brzina]
```

Tablica 29.6 opisuje najčešće korištene opcije.

Tablica 29.6         Cesto korištene opcije pppd naredbenog retka		
Орсіја	Opis	
device	Koristi navedeni uređaj. Ako je potrebno, pppd u niz znakova dodaje /dev/. Kada uređaj nije naveden, pppd koristi upravljački terminal.	
speed	Određuje brzinu modema.	
asyncmap <i>mapa</i>	Određuje kartu (mapu) asinkronih znakova. Ta karta određuje koji kontrolni znakovi ne mogu biti poslani preko veze i potrebno ih je slati kroz escape sekvencu. Karta je 32-bitni heksadecimalni broj u kojem svaki bit predstavlja znak. Nulti bit (00000001) predstavlja znak 0 x 00.	
auth	Zahtijeva da udaljeni domaćin dokaže svoju autentičnost.	
connect <i>program</i>	Za uspostavljanje veze koristi program ili naredbu ljuske. Tu se koristi chat.	
crtscts	Koristi hardversku kontrolu toka.	
xonxoff	Koristi softversku kontrolu toka.	
defaultroute	Nalaže pppdu da u vašoj Kernel routing tablici odredi unaprijed zadani smjer do udaljenog domaćina.	
disconnect <i>program</i>	Pokreće određeni program nakon što pppd svoju vezu prekine.	
escape <i>c1, c2</i> ,	Uzrokuje izbjegavanje nekih znakova pri prijenosu. Znakovi su navedeni uporabom ASCII heksadecimalnih istoznačnica.	
file <i>imedatoteke</i>	Učitava pppd opcije iz navedene datoteke.	
lock	Koristi zaključavanje serijskog uređaja u UUCP stilu.	
mru <i>broj</i>	Određuje najveću prijemnu jedinicu prema navedenom broju.	
netmask <i>maska</i>	Određuje mrežnu masku PPP mrežnog sučelja.	
passive	Čini da pppd čeka valjanu vezu umjesto da odustane ako je ne uspije odmah uspostaviti.	
silent	pppdu ne dopušta da započne pokušaj uspostave veze već mora čekati da je započne udaljeni domaćin.	

Više od 40 drugih argumenata naredbenog retka upravljaju svim mogućnostima PPP-a na svim razinama. Informaciju o njima možete potražiti na glavnoj stranici.

**BILJEŠKA** Program pppd zahtijeva postojanje datoteke /etc/ppp/options, čak i ako je prazna. pppd tu datoteku učitava i ona je odlično mjesto za stavljanje opcija koje želite da pppd koristi tijekom svakog pokretanja.

pppd i chat možete sjedinjavati na razne načine. U naredbenom retku možete navesti sve argumente naredbenog retka za oba programa, pppd opcije stavite u datoteku ili u datoteku stavite chat skriptu. Slijedi jednostavan primjer sa svime u naredbenom retku: \$ pppd connect 'chat "" ATDT5551234 ogin: linux2 word: be4me' \
/dev/cual 38400 mru 296 lock debug crtscts modem defaultroute

Ovo pokreće pppd s jednostavnom chat skriptom koja bira telefonski broj i korisnika linux2 prijavljuje na udaljenom domaćinu. Uključene su i device, speed, MRU i brojne druge opcije.

U drugom krajnjem slučaju možete većinu opcija za pppd staviti u datoteku i naložiti chatu da učita datoteku skripte. Sljedeći je primjer poziv za pppd:

pppd /dev/cual 38400 connect 'chat -f linux1.chat'

Sljedeći redovi prikazuju sadržaj datoteke na koju se upućuje:

# Global PPP Options File	
mru 296	# Set MRU value
lock	# Use UUCP locking
crtscts	# Use hardware handshaking
modem	# Use modem control lines
defaultroute	# Make PPP set up default route

pppd učitava ovu datoteku i obrađuje opcije koje u njoj pronađe. Tekst koji slijedi iza znaka # smatra se napomenom i zanemaruje se.

Sljedeća chat skripta određuje nekoliko abort nizova znakova, bira telefonski broj, čeka prijavni naredbeni redak i ppp korisnika prijavljuje na udaljenom domaćinu s lozinkom ppp-word:

```
abort 'NO CARRIER'
abort 'BUSY'
abort 'VOICE'
abort 'CARRIER 2400'
** ATDT555-1234
CONNECT '\c'
ogin:-BREAK-ogin: ppp
word: ppp-word
```

#### Pružanje PPP usluga

Podešavanje je vašeg Linux stroja da bude PPP poslužitelj čak lakše od podešavanja SLIP poslužitelja. Ono zahtijeva samo jedan novi korisnički račun i skriptu ljuske koja ispravno pokreće program pppd.

Stvorite korisnički račun nazvan ppp sa stavkom /etc/passwd koji izgleda poput ovog:

\$ ppp:\*:501:300:PPP Account:/tmp:/etc/ppp/ppplogin

i odgovarajuće odredite lozinku. Brojevi uid (501) i gid (300) ne smiju biti isti. Također, ako želite, možete svakom PPP klijentu koji imate dodijeliti jedan korisnički račun. Datoteka /etc/ppp/ppplogin treba biti izvršna skripta poput sljedeće:

```
#!/bin/sh
# PPP Server Login Script
# Turn off messages to this terminal
mesg n
# Turn off echoing
```

```
stty -echo
# Run pppd on top of this sh process
exec pppd -detach silent modem crtscts
```

Ova skripta izvršava pppd s argumentom -detach, kako bi ga spriječio od samoisključivanja s terminala. Ako se pppd odijeli, skripta izlazi, uzrokujući isključivanje birane veze. Opcija silent čini da pppd čeka da udaljeni pppd *daemon* započne uspostavljanje veze. Opcije modem čine da pppd nadzire kontrolne retke modema, a crtscts čini da pppd koristi hardversku kontrolu toka.

To je sve što treba. Kada se korisnik prijavljuje na vaš stroj s valjanim korisničkim ID-om i lozinkom, PPP veza se u vašem računalu automatski uspostavlja.

#### Čuvanje sigurnosti vaše PPP veze

Očuvanje je sigurnosti vaše PPP veze vrlo važno. Dopustiti bilo kome da vaš stroj spoji na PPP poslužitelj ili da se sa svojim strojem spoji na vaš PPP poslužitelj jest jednako loše kao da bilo kome dopustite da svoj stroj izravno spoji na vašu mrežu. PPP pruža izravnu IP vezu jer strojeve na oba kraja veze zapravo stavlja na istu mrežu.

Za povećanje su sigurnosti PPP-a razvijena dva protokola za provjeru autentičnosti - Password Authentication Protocol (PAP) i Challenge Handshake Authentication Protocol (CHAP). Za vrijeme uspostavljanja PPP veze svaki stroj može od drugog zahtijevati da dokaže svoju autentičnost. To vam omogućuje da pouzdano znate tko koristi vaš PPP servis. CHAP je sigurniji protokol i ovdje se o njemu govori.

CHAP koristi garnituru *tajnih ključeva* - to su tekstualni nizovi znakova koje u tajnosti drže vlasnici strojeva koji koriste CHAP i šifrirani sustav provjere za međusobnu provjeru autentičnosti. Korisna je osobina CHAP-a da u redovitim vremenskim razmacima izdaje zahtjeve za provjeru, sve dok se PPP veza ne prekine. Tako se, na primjer, mogu otkriti uljezi koji su se zamjenom telefonskih linija ubacili umjesto pravovaljanog korisnika.

Tajni su ključevi za CHAP spremljeni u /etc/ppp/chap-secrets. Za uporabu provjere autentičnosti na vašoj PPP vezi u poziv za pppd dodajete opciju auth i u datoteku chap-secrets dodajete odgovarajuću informaciju za domaćina čija se autentičnost provjerava. Slijedi primjer ogledne datoteke chap-secrets za linux2.burwell.com:

```
# linux2.burwell.com CHAP secrets file
# client/server/secret/IP addr
linux2.burwell.com linux1.burwell.com "It's Full of Stars"
[ccc]linux2.burwell.com
linux1.burwell.com linux2.burwell.com "three stars" linux1.burwell.com
* linux2.burwell.com
```

Svaki redak sadrži do četri polja: ime klijenta domaćina, ime poslužitelja domaćina, tajni ključ, i neobavezan popis IP adresa koje ovaj klijent može zahtijevati da mu budu dodijeljene. Određenja klijenta i poslužitelja u ovoj datoteci utvrđuje domaćin koji daje zahtjev za provjeru autentičnosti (poslužitelj).

Ova datoteka definira različite CHAP tajne. Prvi se redak koristi kada linux1.burwell.com zatraži CHAP dokaz autentičnosti od linux2.burwell.com. Drugi se koristi za obratan slučaj.
Posljednji redak definira univerzalnog klijenta. Time se omogućuje da bilo koji stroj koji poznaje valjani tajni ključ može uspostaviti PPP vezu s linux2.burwell.com. Zamjenski se znak (\*) može koristiti u polju klijenta ili poslužitelja.

Pažljivo vam održavanje datoteke chap-secrets omogućuje potpuni nadzor nad strojevima koji mogu pristupiti vašem PPP poslužitelju i strojevima kojima vi možete pristupiti s PPP-om.

## Odavde...

SLIP i PPP su jeftine zamjene za skupo IP povezivanje preko unajmljene linije. Ovdje ste saznali uvjete za izvođenje SLIP-a i PPP-a i kako SLIP i PPP veze automatizirati uporabom dip i chat naredbi. Naučili ste kako Linux podesiti kao SLIP ili PPP poslužitelj i kako sigurnost PPP-a pojačati uporabom CHAP protokola. Cjelovitu dokumentaciju za dip, chat i pppd možete naći na *man* stranicama.

- 10. poglavlje, "Upravljanje korisničkim računima" pokazuje kako dodati i izbrisati korisničke račune.
- 23. poglavlje, "Razumijevanje protokola TCP/IP" objašnjava što je TCP/IP i kako ti protokoli rade.
- 24. poglavlje, "Podešavanje TCP/IP mreže" vam pokazuje kako Linux stroj podesiti za uporabu na mreži.
- 25. poglavlje, "Podešavanje DNS-a" opisuje kako Linux osposobiti za uporabu DNS-a.

# poglavlje30

## Pristupanje mreži s naredbama telnet, ftp i r-

(Steve Burnett)

## U ovom poglavlju

Uporaba telneta za pristup udaljenim računalima **578** Uporaba FTP-a za daljinski prijenos datoteka **580** Uporaba naredbi r- **589**  Glavna je prednost koju dobivate s umrežavanjem računala mogućnost dijeljenja resursa i informacija i pristupanje tim informacijama iz udaljenih lokacija. Linux za tu svrhu pruža moćan skup alata. Dok vam World Wide Web omogućuje pristup mnoštvu informacija u hipertekstualnom formatu, dodatni vam alati omogućuju prijavljivanje na udaljena računala, prijenos datoteka i daljinsko izvršavanje naredbi.

## Uporaba telneta za pristup udaljenim računalima

Naredba telnet je osnovni alat za prijavljivanje na daljinu pod Linuxom. telnet vam omogućuje terminalsku *sesiju* na udaljenom računalu. Premda neki sustavi pružaju mogućnost prijave kao gosta, takva je mogućnost rijetka zbog sigurnosnih pitanja.

Zamisao koja stoji iza ta dva okružja za goste jest osigurati sigurnost računala i zaštiti sustav od zlonamjernih ili nepažljivih nepoznatih korisnika. Ograničena ljuska korisniku onemogućuje izvršavanje određenih naredbi, a sustav izbornika pruža izbor samo z prethodno određenog skupa izbornika, u potpunosti spriječivši pristup ljuski.

telnet korisnicima omogućuje i prijavljivanje na svoja vlastita računala iz udaljenih lokacija, unošenjem svog korisničkog imema i lozinke. Na taj način korisnici mogu provjeravati e-poštu, uređivati datoteke i pokretati programe na svojim uobičajenim računalima kao da su na njih prijavljeni lokalno. To ćete, međutim, trebati učiniti u okružju koje se temelji na terminalima jer za to ne možete koristiti sustav X Windowsa. telnet pruža samo emulaciju terminala za uobičajene terminale poput DEC VT-100, koji ne podržavaju grafička okružja poput X Windowsa.

#### Pregled naredbi u telnetu

Osnovna je sintaksa za telnet sljedeća:

telnet [hostname]

*hostname* je ime udaljenog računala. Ako udaljenog domaćina ne navedete, telnet se pokreće u svom interaktivnom načinu rada. Ako ime udaljenog domaćina navedete, telnet odmah pokušava započeti *sesiju*.

telnet prihvaća nekoliko argumenata naredbenog retka, navedenih u tablici 30.1.

Tablica 30.1 Arg	umenti naredbenog retka za naredbu telnet
Argument	Opis
- d	Uključuje otkrivanje grešaka.
- a	Pokušava automatsko prijavljivanje.
-n <i>tracefile</i>	Uključuje praćenje i podatke iz praćenja sprema u datoteku pod tracefile.

Argument	Opis
-e escape_char	Određuje da znak za izlazak iz sesije bude escape_char.
-l user	Korisničko ime <i>user</i> šalje udaljenom sustavu za automatsko prijavljivanje. Ovaj argument automatski uključuje i argument -a.
port	Ukazuje na broj <i>porta</i> koje treba koristiti za spajanje na udaljeni sustav. Ovaj se argument koristi za određivanje različitih mrežnih programa. Ako nije naveden, telnet se spaja na unaprijed zadani telnet <i>port</i> .

## Ogledna telnet sesija

Pogledajmo sada jednu oglednu telnet *sesiju*. telnet *sesiju* pokrećete utipkavši **telnet**, nakon čega slijedi ime domaćina - računala na koje se želite spojiti. telnet vam zatim vraća poruku Trying *neka IP adresa* (gdje je *neka IP adresa* adresa računala koje ste naveli). Ako se telnet uspješno spoji na računalo (što znači da je to računalo uključeno i mreža nije isključena), Linux će izvijestiti Connected to *ime računala* i zatim će vam reći da je znak za izlazak iz *sesije* neka određena kombinacija tipki, gotovo uvijek <Ctrl-Ć>. Znak za izlazak određuje i kombinaciju tipki koju trebate utipkati za prelazak iz vaše terminalske sesije u telnetov interpretator naredbi. To ćete uraditi ako naredbe želite izravno slati programu telnet, a ne u vašu *sesiju* na udaljenom računalu.

Nakon što se telnet uspješno spojio na udaljeni sustav bit će prikazana prijavna informacija i sustav će vas zatražiti unos korisničkog imena i lozinke. Pod pretpostavkom da imate valjano korisničko ime i lozinku, uspješno ćete se prijaviti i sada možete interaktivno raditi na udaljenom sustavu.

Slijedi primjer telnet sesije iz Linux računala koje se spaja na Linux računalo:

```
$ telnet server.somewhere.com
Trying 127.0.0.1...
Connected to server.somewhere.com.
Escape character is '^]'.
"Red Hat Linux release 4.0 (Colgate)
kernel 2.018 on an I486
login:bubba
Password: password
Last login: Mon Nov 11 20:50:43 from localhost
Linux 2.0.6. (Posix).
server:~$
server:~$
server:~$
logout
Connection closed by foreign host.
$
```

Kada s udaljenom *sesijom* završite, svakako se trebate odjaviti. telnet zatim izvijesti da je udaljena *sesija* zatvorena i vraćate se u vaš naredbeni redak lokalne ljuske.

## Uporaba FTP-a za daljinski prijenos datoteka

File Transfer Protocol (FTP) je jednostavno i djelotvorno sredstvo za prijenos datoteka između računala koja su spojena na TCP/IP mrežu. FTP korisnicima omogućuje prijenos ASCII i binarnih datoteka.

Tijekom FTP sesije spajate se na drugo računalo uporabom FTP klijent programa. Od ove se točke možete kretati prema gore ili dolje u stablu direktorija, možete pregledavati sadržaje direktorija, kopirati datoteke iz udaljenog u vaše računalo i datoteke iz vašeg računala slati u udaljeno računalo. I ovdje se primijenjuju uobičajene zaštite datoteka - ako za određenu datoteku nemate valjano dopuštenje, ne možete ju dobiti iz udaljenog računala niti ju poslati u njega.

Pri uporabi FTP-a za prijenos datoteka trebate znati valjano korisničko ime i lozinku na udaljenom računalu. Ta se kombinacija korisničko ime/lozinka koristi za provjeru valjanosti vaše FTP sesije i utvrđivanja kakvo imate dopuštenja pristupa datotekama za prijenos. Očito je da trebate znati i ime računala na kojem namjeravate sprovesti FTP sesiju.

Trebate biti svjesni da FTP klijenti imaju različit skup naredbi, ovisno o dotičnom operativnom sustavu. Ovo poglavlje pokriva Linux FTP klijent, međutim, kada FTP sesiju pokrenete na udaljenom sustavu, udaljeni bi sustav mogao očekivati drukčije naredbe. Rijetkost je da su FTP sustavi međusobno posve neusklađeni. Uobičajeno je da su naredbe koje obično koristite ili nešto drukčije ili nisu raspoložive.

## **Anonimni FTP**

Zahvaljujući eksplozivnom rastu Interneta, mnoge su organizacije svoje goleme riznice informacija stavile na raspolaganje preko FTP-a. Te FTP lokacije imaju sve, od tekstualnih datoteka do softvera svih zamislivih vrsta. Međutim, kako tom ogromnom skladištu podataka pristupiti ako na udaljenom računalu nemate svoj korisnički račun? Trebate li radi pristupa tim datotekama na svakoj od tih FTP lokacija tražiti korisnički račun?

Ukratko, odgovor je ne. Prema uobičajenom dogovoru na Internetu korisnici mogu dobiti FTP pristup do riznica datoteka u svojstvu gosta, što im omogućuje prijenos datoteka. Takav se pristup u svojstvu gosta naziva anonymous FTP. Za uporabu anonimnog FTP-a pokrenite FTP sesiju prema udaljenom sustavu i za korisničko ime koristite anonymous,

a kao lozinku stavite vašu e-mail adresu. Na primjer, u sljedećem primjeru korisnik nazvan smith na linux.somewhere.com želi pokrenuti FTP sesiju s uobičajenom FTP lokacijom:

\$ ftp ftp.uu.net ftp.uu.net (login:smith): anonymous Password: smith@linux.somewhere.com

Mnoge lokacije ne dopuštaju anonimni FTP. Dopuštanje korisnicima spajanje na vaše BILJEŠKA računalo u svojstvu gosta uključuje i određeni rizik. U slučajevima gdje anonimni FTP

nije dopušten, naredba ftp se neće uspješno izvršiti i prikazat će poruku sličnu Login failed - User "anonymous" unknown. Lokacije koje anonimni FTP dopuštaju obično korisnika stavljaju u ograničeno stablo direktorija s pristupom samo-za-čitanje. Ako vam se dopušta i slanje datoteka u udaljeno računalo, obično ih možete stavljati samo u jedan direktorij.

## Pregled ftp naredbi

Linux ftp naredba pruža vrlo opširan skup naredbenih opcija u interaktivnom načinu rada. Kako je ranije napomenuto, neki udaljeni domaćini možda neće podržavati sve te naredbe. Mnoge od njih vjerojatno ionako nećete ni trebati koristiti. Tablica 30.2 prikazuje popis naredbi raspoloživih u FTP-u.

Tablica 30.2 ftp	naredbe raspoložive u interaktivnom načinu rada
Naredba	Opis
!	Izlazak u ljusku
\$	Izvršava makro
account	Udaljenom poslužitelju šalje naredbu account
append	Dodaje se datoteci
ascii	Određuje da se prijenos datoteka vrši u ASCII načinu
bell	Nakon izvršenja naredbe oglašava se zvučni signal
binary	Određuje da se prijenos datoteke vrši u binarnom načinu rada
bye	Zaustavljanje i izlaz i FTP sesije
case	Izmjenjuje mget mapiranje imena datoteka velikim ili malim slovima
cd	Mijenja radni direktorij na udaljenom računalu
cdup	Mijenja udaljeni radni direktorij u matični direktorij
chmod	Mijenja dopuštenja pristupa udaljene datoteke
close	Zaustavlja FTP sesiju
cr	Uključuje/isključuje znak vraćanja na početak reda pri primanju ASCII datoteke
delete	Briše udaljenu datoteku
debug	Uključuje/isključuje način rada s ispravljanjem grešaka
dir	Ispisuje sadržaj udaljenog direktorija (s veličinama i dopuštenjima)
disconnect	Zaustavlja FTP sesiju (isto kao i close)
exit	Zaustavljanje i izlaz i FTP sesije
form	Određuje format za prijenos datoteka
get	Učitavanje datoteke iz udaljenog računala
glob	Uključuje/isključuje proširenja lokalnih imena datoteka sa zamjenskim znakom (*)
hash	Uključuje/isključuje ispisivanje znaka # za svaki preneseni sadržaj privremenog spremnika
help	Ispisuje informaciju lokalnog sustava pomoći

nastavlja se

Tablica 30.2	ftp naredbe raspoložive u interaktivnom načinu rada
Naredba	Opis
idle	Očitava ili podešava dopušteno vrijeme mirovanja na udaljenom računalu
image	Određuje da se prijenos datoteka vrši u binarnom načinu (isto kao i binary)
lcd	Mijenja lokalni radni direktorij
ls	Prikazuje popis sadržaja udaljenog direktorija (s veličinama i dopuštenjima datoteka)
macdef	Definira makro
mdelete	Brisanje više datoteka na udaljenom računalu
mdir	Prikazuje sadržaj više udaljenih direktorija
mget	Učitavanje više datoteka iz udaljenog računala
mkdir	Stvara direktorij na udaljenom računalu
mls	Prikazuje sadržaj više udaljenih direktorija
mode	Određuje način prijenosa datoteka
modtime	Prijazuje vrijeme posljednje preinake na udaljenoj datoteci
mput	Slanje više datoteka u udaljeno računalo
newer	Učitava udaljenu datoteku ako je udaljena datoteka novija od podudarne lokalne datoteke
nmap	Određuje predloške za unaprijed zadano mapiranje imena datoteka
nlist	Prikazuje sadržaj udaljenog direktorija
ntrans	Određuje tablicu prijevoda za unaprijed zadano mapiranje imena datoteka
open	Spajanje na udaljenu FTP lokaciju
passive	Unos pasivnog načina prijenosa
prompt	Iznuđuje interaktivne upite za više naredbi
proxy	Izdaje naredbu na zamjenskoj vezi
put	Slanje jedne datoteke u udaljno računalo
pwd	Ispis radnog direktorija na udaljenom stroju
quit	Zaustavljanje i izlaz i FTP sesije
quote	Šalje bilo koju ftp naredbu
recv	Prima datoteku
reget	Učitavanje datoteke na kraj lokalne datoteke
rstatus	Prikazuje stanje udaljenog stroja

Naredba	Opis
rhelp	Dobivanje pomoći iz udaljenog računala
rename	Preimenovanje datoteke
reset	Briše spremljene odzive naredbi
restart	Pokreće ponovni prijenos datoteke od zadanog broja bajtova
rmdir	Uklanja direktorij iz udaljenog stroja
runique	Dodjeljuje jedinstveno ime datoteke svakoj zaprimljenoj datoteci kada se u isti direktorij zaprima više datoteka s istim imenom
send	Slanje jedne datoteke u udaljeno računalo
site	Šalje jednu od lokacijski-specifičnih naredbi u udaljeni poslužitelj - jednu od umask, idle, chmod, help, group, gpass, newer ili minfo
size	Prikaz veličine udaljene datoteke
status	Prikaz trenutnog stanja
struct	Određivanje strukture prijenosa datoteke
system	Prikaz vrste udaljenog sustava
sunique	Pri slanju više datoteka istog imena u isti direktorij, svakoj se poslanoj datoteci dodjeljuje jedinstveno ime datoteke
tenex	Određivanje vrste prijenosa datoteka tenex
tick	Uključuje/isključuje brojač ispisne veličine u bajtovima tijekom prijenosa
trace	Uključuje/isključuje praćenje paketa
type	Određuje vrstu prijenosa datoteka
user	Slanje informacija o novom korisniku
umask	Očitava ili postavlja umask na udaljenom računalu
verbose	Uključivanje/isključivanje prikazivanja poruka u engleskom tekstu umjesto kao sažeti kôd
?	Ispisivanje informacija iz lokalnog sustava pomoći

Kao što možete vidjeti, ftp ima priličan broj naredbi. Međutim, stvarno trebate upamtiti samo one koje najčešće koristite.

**Pokretanje FTP sesije** Za otvaranje FTP *sesije* na udaljenom domaćinu koristi se naredba open. Njena je sintaksa sljedeća:

open *hostname* 

Ovu naredbu obično trebate samo ako ćete se tijekom FTP *sesije* spajati na više od jedne lokacije. Ako se tijekom *sesije* želite spojiti samo na jedno računalo, dovoljno je u naredbeni redak kao argument za naredbu ftp navesti ime udaljenog domaćina. **Zaustavljanje FTP sesije** Za zaustavljanje se FTP *sesije* na udaljenom računalu koriste naredbe close, disconnect, quit ili bye. Naredbe close i disconnect imaju isto značenje - one zatvaraju vašu vezu s udaljenim računalom, ali vas ostavljaju u programu ftp na vašem lokalnom računalu. Naredbe quit, exit i bye zatvaraju vašu vezu s udaljenim računalom (ako je aktivna) i zatim zatvaraju program ftp na vašem računalu.

**Promjena direktorija** Naredba cd [*direktorij*] se koristi za promjenu direktorija na udaljenom računalu tijekom vaše FTP *sesije*. Naredba cdup vas vodi u matični direktorij trenutnog direktorija. Naredba lcd mijenja vaš lokalni direktorij tako da možete navesti gdje pronaći ili staviti lokalne datoteke.

**Ispis udaljenog direktorija** Naredba ls prikazuje sadržaj udaljenog direktorija, poput ls-a iz interaktivne ljuske. Sintaksa za ls je sljedeća:

```
ls [directory] [local_file]
```

Ako je kao argument naveden direktorij, ls prikazuje sadržaj tog direktorija. Ako ste dali ime lokalne datoteke, ispis sadržaja direktorija se stavlja u navedenu datoteku u vašem lokalnom računalu.

Naredbe dir i ls pružaju popise s dugim sadržajem, prikazujući zaštite, veličine, vlasnike i datume. Sintaksa je naredbe dir sljedeća:

dir [directory] [local\_file]

Slijedi primjer ispisa direktorija s naredbom dir:

-rw-r—r—	1	root	archive	2928	May	17	1993	README
-rw-r-r-	1	root	archive	1723	Jun	29	1993	README.NFS
dr-xr-xr-x	2	root	wheel	8192	Jun	6	12:16	bind
-rwxr-xr-x	5	root	wheel	8192	Aug	2	06:11	decus
drwxr-xr-x	19	root	archive	8192	Feb	7	1994	doc
drwxr-xr-x	6	root	wheel	8192	Jun	15	15:45	edu
dr-xr-xr-x	7	root	wheel	8192	Sep	28	09:33	etc

Vidi "Upravljanje sustavom datoteka" str. 265.

**Uzimanje datoteka iz udaljenog sustava** Naredbe get i mget se koriste za učitavanje datoteka iz udaljenog sustava. Naredba get učitava datoteku koju ste naveli kao argument (*imedatoteke*). Slijedi sintaksa za naredbu get:

get filename [remote\_filename]

Možete, dakle, dati i lokalno ime datoteke, to jest, ime koje će se datoteci dati pri njenom spremanju u vaše lokalno računalo. Ako ne date lokalno ime datoteke, koiristit će se *remote\_filename*.

Naredbom mget istovremeno dobavljate više datoteka. Sintaksa za mget je sljedeća:

mget filename\_list

Te ćete datoteke navesti upisivanjem njihovih imena odvojenih razmacima ili uporabom zamjenskog znaka (\*). Za svaku će se datoteku pojaviti posebni upit. Za isključivanje tih upita, prije naredbe mget upotrijebite naredbu prompt. Datoteke se u oba slučaja prenose kao ASCII datoteke, osim ako ste odredili drukčiji način prijenosa. **Slanje datoteka u udaljeni sustav** Za slanje se datoteka u udaljeno računalo koriste naredbe put i mput. Naredba put šalje lokalnu datoteku koju navedete kao argument. Sintaksa je sljedeća:

put filename

Naredba mput šalje niz lokalnih datoteka. Sintaksa za mput je ovdje prikazana:

mput filename\_list

Imena datoteka koja navodite odvajate razmacima. Možete koristiti i zamjenski znak (\*). Pri uporabi naredbe mget pojavljivat će se upiti za svaku datoteku. Za isključivanje tih upita koristite naredbu prompt. Datoteke se u oba slučaja prenose kao ASCII datoteke, osim ako ste odredili drukčiji način prijenosa.

**Promjena načina prijenosa datoteka** ftp će datoteke prenositi kao ASCII datoteke osim ako odredite neki drugi način. To je u redu za običan tekst, ali binarne podatke čini beskorisnima. Naredbe ascii i binary određuju način prijenosa, tako da možete spriječiti oštećenja vaših binarnih datoteka.

**NAPOMENA** Mnoge su datoteke koje ćete htjeti prenjeti spremljene u binarnom formatu. Datoteke s nastavkom .tar su arhive stvorene naredbom tar. Datoteke s nastavcima .Z i .gz su komprimirane s naredbom compress, odnosno s GNU naredbom gzip. Datoteke koje završavaju s .zip su komprimirane arhive stvorene s PKZIP-om. Kada niste sigurni, koristite binarni način prijenosa. Uporabom ASCII načina kvarite datoteke s binarnim podacima.

**Provjera stanja prijenosa** Pri prijenosu će vam velike datoteke vjerojatno biti korisno da vas ftp obaviještava o napredovanju prijenosa. ftp će s naredbom hash nakon svakog uspješnog prijenosa sadržaja podatkovnog spremnika na zaslonu ispisati znak #. Ta se naredba može koristiti i pri slanju i pri primanju datoteka.

**Lokalne naredbe iz FTP-a** Znak ! se koristi za prosljeđivanje naredbi u naredbenu ljusku u vašem lokalnom računalu za vrijeme rada u FTP-u. To može biti vrlo korisno ako nešto trebate obaviti usred rada u FTP *sesiji*. Pretpostavimo da trebate stvoriti direktorij za pohranu zaprimljenih datoteka. Ako unesete !mkdir novi\_dir, Linux će u vašem trenutnom lokalnom direktoriju stvoriti direktorij s imenom novi\_dir.

## **Primjer FTP sesije**

Ispis 30.1 prikazuje kratku FTP sesiju.

```
Ispis 30.1 Uspostavljanje FTP veze i dobivanje ispisa direktorija

$ ftp opus
Connected to opus.
220 opus FTP server (Linux opus 2.0.6 #4 Mon Nov 11 16:01:33 CDT 1996) ready.
Name (opus:smith): smith
Password (opus:smith): password
331 Password required for smith.
230 User smith logged in.
```

nastavlja se

```
Ispis 30.1 Nastavak
```

```
Remote system type is UNIX.
 Using ASCII mode to transfer files.
 ftp> dir
 200 PORT command successful.
 150 Opening ASCII mode data connection for /bin/ls.
 total 8

      -rw-r-r-
      1
      root
      ude...

      dr-xr-xr-x
      2
      root
      wheel
      512
      Jun 24
      11:00

      dr-r-r-
      2
      root
      wheel
      512
      Jun 24
      11:18
      dev

      dr-r-r-
      2
      root
      wheel
      512
      Jun 24
      11:18
      dev

      dr-r-r-
      2
      root
      wheel
      512
      Jun 24
      11:24
      etc

      dr-xr-xr-x
      4
      root
      wheel
      512
      Sep 29
      15:37
      pub

      dr-xr-xr-x
      3
      root
      wheel
      512
      Jun 24
      11:15
      usr

      - n n-
      1
      root
      daemon
      461
      Jun 24
      13:46
      weld

                                                                                   1525 Sep 29 15:37 README
                                                                                    461 Jun 24 13:46 welcome.msg
 433 bytes received in 0.027 seconds (16 Kbytes/s)
 ftp> get README
 200 PORT command successful.
 150 Opening ASCII mode data connection for README (1525 bytes).
 226 Transfer complete.
 local: README remote: README
 1561 bytes received in 0.0038 seconds (4e+02 Kbytes/s)
 ftp> guit
 221 Goodbye.
 $
```

U prethodnom primjeru korisnik otvara FTP *sesiju* u domaćinu opus i prijavljuje se kao smith. Udaljeni FTP poslužitelj traži lozinku koju korisnik zatim utipka (lozinka se ne pojavljuje na zaslonu). ftp zatim korisnika smith prijavljuje na udaljenom sustavu i prikaže naredbeni redak ftp> za unos naredbi u interaktivnom načinu rada. Korisnik je unio naredbu dir s kojom ftp-u kaže da ispiše sadržaj udaljenog direktorija i zatim koristi naredbu get za prijenos (učitavanje) datoteke README. Nakon završetka rada u FTP *sesiji*, neustrašivi se korisnik odjavljuje s naredbom quit i vraćen je u naredbeni redak lokalne Linux ljuske.

#### Primjer anonimne FTP sesije

U prethodnom je primjeru korisnik pokrenuo FTP *sesiju* unutar sustava i pogledao neke direktorije. Korisnik ima valjano korisničko ime i lozinku na udaljenom sustavu. Sada pogledajte anonimnu FTP *sesiju* na Internet lokaciji s velikom arhivom softvera. Ispis 30.2 je vrlo sličan ispisu 30.1, ali ima i nekih zanimljivih razlika.

Ispis 30.2 Uspostavljanje i rad u anonimnoj FTP vezi

```
$ ftp ftp.uu.net
Connected to ftp.uu.net.
220 ftp.UU.NET FTP server (Version wu-2.4(1) Wed Nov 13 15:45:10 EST 1996)
ready.
Name (ftp.uu.net:bubba): anonymous
331 Guest login ok, send your complete e-mail address as password.
Password: your_e-mail_address
```

230-230-Welcome to the UUNET archive. 230-A service of UUNET Technologies Inc, Falls Church, Virginia 230-For information about UUNET, call +1 703 204 8000. 230or see the files in /uunet-info 230-230-Access is allowed all day. 230-Local time is Wed Nov 13 15:53:02 1996. 230-230-All transfers are logged with your host name and email address. 230-If you don't like this policy, disconnect now! 230-230-If your FTP client crashes or hangs shortly 230after login, try using a 230dash (-) as the first character of your password. 230-This will turn off the informational messages which may 230be confusing your ftp client. 230-Please read the file /info/README.ftp 230- it was last modified on Mon Nov 11 17:39:53 1996 - 2 days ago 230 Guest login ok, access restrictions apply. ftp> ftp> dir 200 PORT command successful. 150 Opening ASCII mode data connection for /bin/ls. total 4149 34 512 Jul 26 1992 drwxr-sr-x 2 0 .forward 1 34 uucp 0 Jul 26 1992 -rw-r-r-.hushlogin 1 34 archive 59 Jul 31 1992 -rw-r—r— .kermrc 1 34 archive 0 Jul 26 1992 -rw-r-r-.notar drwx-s-x 5 34 512 Jul 23 19:00 admin archive lrwxrwxrwx 1 34 archive 1 Jul 26 1992 archive -> . 4 0 archive 512 Apr 20 16:29 drwxrws—x bin lrwxrwxrwx 1 34 archive 23 Sep 14 1993 by-name.gz -> ⇒index/master/by-name.gz lrwxrwxrwx 1 34 archive 23 Sep 14 1993 by-time.gz -> ⇒index/master/by-time.gz -rw-r-r- 1 34 archive 90112 Apr 26 1991 compress.tar 1 0 9 Jul 23 lrwxrwxrwx archive 18:50 core -> /dev/null archive 512 Jul 26 1992 dev 2 0 drwxrws-x drwxrwsr-x 21 34 archive 1024 Sep 29 15:18 doc 512 Apr 14 16:42 etc 60 archive drwxrws—x 1 34 archive 31 Dec 8 1993 faces -> lrwxrwxrwx ►/archive/published/usenix/faces drwxrwsr-x 2 34 512 Jul 26 1992 ftp archive 4 34 512 Sep 29 10:34 government drwxrwsr-x archive drwxrwsr-x 18 34 archive 1024 Sep 29 10:28 graphics -rw-rw-r— 1 27 archive 798720 Jul 11 20:54 gzip.tar 1 34 17 Jul 26 1992 help -> info/archive-help lrwxrwxrwx archive drwxrwsr-x 20 34 archive 1024 Dec 2 1993 index drwxrwsr-x 19 34 archive 512 Sep 29 10:30 inet 4 34 archive 512 Sep 29 15:36 drwxrwsr-x info drwxrwsr-x 25 34 archive 512 Sep 29 10:29 languages drwxrwsr-x 4 34 512 Sep 29 10:28 library archive 2 0 8192 Jul 26 1992 lost+found drwx—s—x 0 nastavlja se

```
Ispis 30.2
               Nastavak
1rwxrwxrwx
            1 34
                      archive
                                    20 Aug 2 1992
                                                      ls-1R.Z ->
⇒index/master/ls-lR.Z
                      archive
                                    21 Sep 14 1993
                                                      ls-1R.gz ->
lrwxrwxrwx
            1 34
⇒index/master/ls-lR.gz
                                    21 Aug 2
                                               1992
                                                      ls-ltR.Z ->
lrwxrwxrwx 1 34
                      archive
⇒index/master/ls-ltR.Z
                      archive
                                    22 Sep 14 1993
                                                      ls-ltR.gz ->
lrwxrwxrwx
           1 34
⇒index/master/ls-ltR.gz
drwxrwsr-x 24 34 archive 1024 Sep 29 15:10 networking
                               512 Aug 10 09:26 package
512 Sep 26 12:29 private
           2
drwxrwsr-x
                34
                      archive
                                               09:26 packages
                 34 archive
d—xrws—x 17
drwxrwsr-x 25 34 archive 1536 Sep 29 15:30 pub
drwxrwsr-x 17 34 archive 1024 Sep 29 15:38 published

        34
        archive
        10
        Jul 2b
        1992
        01

        24
        archive
        512
        Sep 29
        04:18
        systems

        20
        15:36
        tmp

           1
                                                      sco-archive -> vendor/sco
lrwxrwxrwx
drwxrwsr-x 20 34 archive
drwxrwxrwx 14 34 archive 1536 Sep 29 15:36 tmp
lrwxrwxrwx 1 34 archive 17 Jul 26 1992
                                                      unix-today ->
⇒vendor/unix-today
lrwxrwxrwx 1 34 archive
                                    17 Jul 26 1992
                                                      unix-world ->
⇒vendor/unix-world
drwxrwsr-x 36 34 archive
                                  1024 Sep 29 15:29 usenet
drwxrws-x 6 0
                                   512 Oct 22 1992
                      archive
                                                      usr
lrwxrwxrwx 1 34 archive
                                    16 Aug 2 1992
                                                      uumap -> networking/uumap
-rw-rw-r 1 34 archive 3279895 Sep 28 21:05 uumap.tar.Z
drwxrwsr-x 3 210 archive
                                  2560 Sep 29 15:36 uunet-info
drwxrwsr-x 64 34
                      archive
                                  1536 Sep 29 10:29 vendor
226 Transfer complete.
3257 bytes received in 0.76 seconds (4.2 Kbytes/s)
ftp>
ftp> cd systems/unix/linux
250-Files within this subtree are automatically mirrored from
250-tsx-11.mit.edu:/pub/linux
250-
250 CWD command successful.
ftp>
ftp> binary
200 Type set to I.
ftp> get sum.Z
200 PORT command successful.
150 Opening BINARY mode data connection for sum.Z (80959 bytes).
226 Transfer complete.
local: sum.Z remote: sum.Z
80959 bytes received in 5.6 seconds (14 Kbytes/s)
ftp> guit
221 Goodbye.
$
```

Ovdje je FTP *sesija* pokrenuta na **ftp.uu.net** - glavnoj lokaciji s FTP arhivom na Internetu. Na upit za korisničko ime je upisano anonymous, jer je to anonimni FTP. Za lozinku je korištena puna e-mail adresa. **ftp.uu.net** zatim prikazuje poruku dobrodošlice s nekim informacijama o arhivi. U ovom slučaju možete vidjeti da korisnik mijenja direktorije, određuje binarni prijenos datoteka, dobiva komprimiranu binarnu datoteku i izlazi.

#### RJEŠAVANJE POTEŠKOĆA

Prenio sam binarnu datoteku, ali ona ne radi ispravno. Ne uspijevam je raspakirati ni s jednim od programa za sažimanje. Što bih trebao uraditi? Najvažnije je da podesite binarni način prijenosa. To možete učiniti u ftp> naredbenom retku s naredbom binary.

**Upravo mi je u tijeku prijenos velike datoteke i želim provjeriti kako napreduje.** Upotrijebite naredbu hash. Za svaki obrađeni podatkovni spremnik ftp će na zaslonu prikazati znak #. Veličina podatkovnog spremnika ovisi o vašoj inačici Linuxa - obično je 1.024, 4.096 ili 8.192 bajtova.

Na jednu sam se FTP lokaciju pokušao prijaviti kao anonymous, međutim, s lokacije je stigla obavijest da je korisnik anonymous nepoznat i prijavljivanje nije uspjelo. Ili ste "anonymous" pogrešno napisali ili lokacija ne dozvoljava anonimni FTP. U drugom biste slučaju na udaljenom računalu trebali imati valjano korisničko ime i lozinku.

Želim prenjeti nekoliko datoteka, ali ne želim da mi FTP za svaku postavlja upit. Upotrijebite naredbu prompt, koja te upite uključuje i isključuje.

Pokušao sam koristiti anonimni FTP, ali sam iz lokacije upozoren da za lozinku nisam unio valjanu e-mail adresu. Prije je dogovor bio da se tijekom anonimnog FTP spajanja kao lozinka unese guest (gost). Sada je dogovor da se unese vlastita e-mail adresa. Mnoge FTP lokacije imaju poseban softver za FTP poslužitelje, koji provjerava lozinku i osigurava da je napisana u obliku user@host.somewhere.domain. Pokušajte ponovno i pobrinite se da ispravno unesete vašu punu e-mail adresu.

## Uporaba -r naredbi

Pored naredbi ftp i telnet postoje i neke druge koje vam omogućuju pristup na udaljena računala i razmjenu datoteka preko mreže. Te su naredbe zbirno poznate kao *-r* naredbe.

-r naredbe zaslužuju poseban osvrt budući da jedna od njihovih osobina u slučaju nepažnje može prouzrokovati ozbiljnu sigurnosnu rupu. Kada izdate r- naredbu, udaljeni sustav provjerava da li je vaš lokalni domaćin naveden u datoteci /etc/hosts.equiv. Ako vašeg lokalnog domaćina ne pronađe, traži datoteku .rhosts u vašem polaznom direktoriju na udaljenom stroju. Zatim r- naredba provjerava da li se ime vašeg lokalnog domaćina nalazi u datoteci .rhosts. Ako je vaš lokalni domaćin naveden u bilo kojoj od ovih datoteka, naredba se izvršava bez provjeravanja lozinke.

Premda može biti vrlo lagodno ako vašu lozinku ne morate upisivati svaki put kada trebate pristup na udaljeno računalo, to očito može uzrokovati ozbiljne sigurnosne probleme. Preporučljivo je da ovaj utjecaj r- naredbi na sigurnost pažljivo razmotrite prije nego datoteke /etc/hosts.equiv i .rhosts postavite na vašem lokalnom sustavu.

### rlogin

Naredba rlogin je vrlo slična naredbi telnet, budući da vam omogućuje pokretanje interaktivne *sesije* na udaljenom računalu. Sintaksa za rlogin je sljedeća:

rlogin [-8EKLdx] [-e char] [-k realm] [-l user-name] hostname

Međutim, najčešće se koristi ovako jednostavno:

rlogin *hostname* 

Tablica 30.3 objašnjava različite opcije za rlogin.

Tablica 30.3	Opcije naredbenog retka za naredbu rlogin
Орсіја	Opis
- 8	Dopušta unos 8-bitnih podatkovnih staza u bilo koje vrijeme, čime se omogućuje slanje formatiranih ANSI znakova i drugih posebnih kodova. Ako se ova mogućnost koristi, bitovi pariteta se uklanjaju osim kada su znakovi za daljinsko pokretanje ili zaustavljanje drukčiji nego <ctrl-s> i <ctrl-q>.</ctrl-q></ctrl-s>
- E	Sprečava prepoznavanje bilo kojeg znaka kao <i>escape</i> znaka. Kada se koristi s opcijom -8, to omogućuje potpunu transparentnost veze.
- K	Isključuje svu Kerberos provjeru autentičnosti. Koristi se samo pri spajanju na domaćina koji koristi protokol Kerberos za provjeru autentičnosti.
- L	Omogućuje izvršavanje rlogin <i>sesije</i> u načinu litout. Za više informacija pogledajte <i>man</i> stranice za tty.
- d	Uključuje ispravljanje grešaka na TCP priključnici koja se koristi za vezu s udaljenim domaćinom. Za više informacija pogledajte <i>man</i> stranicu za setsockpot.
- e	Koristi se za određivanje escape znaka za rlogin <i>sesiju</i> . Unaprijed je zadano da je escape znak ~. Možete odrediti i neko slovo ili oktalnu vrijednost u obliku \nnn.
- k	Od rlogina zahtijeva nabavu Kerberos ulaznica za udaljenog domaćina u zadano područje umjesto u područje udaljenog domaćina kako je utvrđeno s krb_realmoofhost(3).
- ]	Dopušta navođenje udaljenog imena. Ako je raspoloživa, koristi se Kerberos provjera autentičnosti.
- X	Uključuje DES šifriranje za sve podatke prosljeđene preko <i>sesije</i> rlogin. Ovo može utjecati na vrijeme odaziva i zauzetost procesora, ali pruža povećanu sigurnost.

### rsh

Naredba rsh, kratica za *remote shell*, pokreće ljusku na zadanom udaljenom domaćinu i izvršava naredbu koju ste možda naveli u naredbenom retku rsh. Ako ne navedete naredbu koju treba izvršiti, na udaljeni se stroj prijavljujete uporabom naredbe rlogin.

Sintaksa je za naredbu rsh sljedeća:

rsh [-Kdnx] [-k realm] [-l username] hostname [command]

Međutim, najčešće se koristi na ovaj način:

rsh hostname [command]

Argument *command* može biti praktički svaka Linux naredba koja se može unijeti iz naredbenog retka ljuske. Tablica 30.4 objašnjava opcije naredbenog retka za rsh.

Tablica 30.4	Opcije naredbenog retka za naredbu rsh
Орсіја	Opis
- K	Isključuje svu Kerberos provjeru autentičnosti. Koristi se samo pri spajanju na domaćina koji koristi Kerberos.
- d	Uključuje ispravljanje grešaka na TCP priključnici koja se koristi za vezu s udaljenim domaćinom. Za više informacija pogledajte <i>man</i> stranicu za setsockpot.
- k	Od rsha zahtijeva nabavu Kerberos ulaznica za udaljenog domaćina u zadano područje umjesto u područje udaljenog domaćina kako je utvrđeno s krb_realmoofhost(3).
- 1	Dopušta navođenje udaljenog imena. Ako je raspoloživa, koristi se Kerberos provjera autentičnosti, a autorizacija se utvrđuje kao i za naredbu rlogin.
- n	Preusmjerava ulaz iz posebnog uređaja /dev/null.
- X	Uključuje DES šifriranje svih prosljeđenih podataka. To može utjecati na vrijeme odaziva i zauzetost procesora, ali pruža povećanu sigurnost.

Linux uzima standardni ulaz u naredbu rsh i kopira ga u standardni ulaz u udaljenu naredbu koja se izvršava. Standardni izlaz udaljene naredbe kopira u standardni izlaz za rsh. Također kopira udaljenu standardnu grešku u lokalni *file descriptor* standardnih grešaka. Svi se signali izlaženja, zaustavljanja i prekidanja šalju udaljenoj naredbi. Sa svim se posebnim znakovima ljuske koji nisu zatvoreni navodnicima, kao u ">>", lokalno barata. Ako jesu zatvoreni navodnicima, tada s tim znakovima rukuje udaljena naredba.

#### rcp

Naredba rcp (kratica za *remote copy*) je posljednja od r- naredbi koje trebate poznavati. Koristi se za kopiranje datoteka između računala. rcp možete koristiti za kopiranje datoteka s jednog na drugo udaljeno računalo, a pritom ni izvor ni odredište ne trebaju biti na lokalnom stroju.

Naredba rcp ima dva oblika. Prvi se oblik koristi za kopiranje datoteke u datoteku. Drugi se oblik koristi pri kopiranju datoteka ili direktorija u direktorij. Prema tome, možete koristiti jednu od sljedećih sintaksi:

```
rcp [-px] [-k realm] filename1 filename2
rcp [-px] [-r] [-k realm] file(s) directory
```

Kao argument za datoteku ili direktorij može se staviti udaljeno ime ili lokalno ime. Udaljena imena datoteka imaju oblik *rname@rhost:path*, gdje je *rname* udaljeno korisničko ime, *rhost* udaljeno računalo, a *path* staza do datoteke. Ime datoteke mora sadržavati dvotočku.

Tablica 30.5 objašnjava argumente za rcp

Tablica 30.5	Argumenti naredbenog retka za naredbu rcp
Opcija	Opis
- r	Rekurzivno kopiranje stabla izvornog direktorija u odredišni direktorij. Za uporabu ove opcije, odredište mora biti direktorij.
- p	Pokušava sačuvati vremena preinaka i modove izvorne datoteke, zanemarujući umask
- k	Od rcpa zahtijeva nabavu Kerberos ulaznica za udaljenog domaćina u zadano područje umjesto u područje udaljenog domaćina kako je utvrđeno s krb_realmoofhost(3).
- X	Uključuje DES šifriranje svih podataka prosljeđenih s rcp-om. To može utjecati na vrijeme odaziva i zauzetost procesora, ali pruža povećanu sigurnost.

Ako staza zadana u imenu datoteke nije potpuna, ona je protumačena kao da se odnosi na prijavni direktorij navedenog korisnika u udaljenom računalu. Ako korisničko ime za udaljeno računalo nije navedeno, koristi se vaše trenutno korisničko ime. Ako staza u udaljenom domaćinu sadrži posebne znakove ljuske, ona može biti stavljena pod navodnike, uporabom \, " ili ', ovisno o uvjetima. To uzrokuje da svi metaznakovi ljuske bivaju interpretirani na udaljenom računalu.

rcp ne traži unos lozinke. On svoje kopije izvršava preko naredbe rsh.

#### BILJEŠKA

#### ssh

ssh (kratica za *secure shell*) je, poput naredbe rsh, program za prijavljivanje na udaljeni stroj i izvršavanje naredbi na tom udaljenom stroju. ssh je napravljen kao zamjena za rsh i rlogin pružajući sposobnost definiranja šifrirane *sesije* između dva nepouzdana sustava preko nesigurne mreže. Jedna od poteškoća s telnetom jest, kada se prijavite na udaljeni sustav, lozinka se u ASCII obliku šalje preko mreže. Netko bi promatrajući Ethernet pakete mogao pokupiti vaše prijavno ime i lozinku za udaljeni sustav. rsh to sprečava uporabom RSAtemeljene provjere autentičnosti. Zbog njegove sigurnosti, ssh danas često koriste administratori sustava. ssh klijenti su raspoloživi i za druge operativne sustave, uključujući Macintosh i Windows. Naredba ssh je vrlo slična naredbi telnet, budući da vam omogućuje pokretanje interaktivne *sesije* na udaljenom sustavu. Sintaksa za ssh je sljedeća:

```
ssh [-a] [-c idea|blowfish|des|3des|arcfour|tss|none] [-e escape_char]

       [-I identity_file] [-l login_name] [-n] [-k] [-V] [=o option] [-p port]
       [-q] [-P] [-t] [-v] [-x] [-C] [-L port"host:hostport] [-R port:host:hostport]
       hostname [command]
```

Međutim, najčešće se koristi ovako:

ssh *hostname* 

Tablica 30.4 objašnjava različite opcije za ssh.

Орсіја	Opis
- a	Onemogućuje prosljeđivanje agenta za provjeru autentičnosti.
- C	Odabir šifre za šifriranje <i>sesije</i> . Idea je unaprijed zadana, arcfour je najbrža, a none je jednaka uporabi rlogina ili rsh-a (bez šifriranja).
- e	Određuje escape znak za sesiju.
- f	ssh stavlja u pozadinu nakon uspostavljanja provjere autentičnosti i prosljeđivanja.
-i	Odabire datoteku iz koje se čita privatni ključ RSA autentifikacije
- k	Onemogućuje prosljeđivanje Kerberos ulaznica.
- 1	Određuje prijavno ime za uporabu na udaljenom stroju.
- n	Preusmjerava stdin iz /dev/nulls kada ssh radi u pozadini.
- 0	Koristi se za korisnički definirane opcije prema formatu u konfiguracijskoj datoteci.
- p	Određuje priključak koji se koristi za spajanje na udaljenog domaćina.
- q	Aktivira tihi način rada, obustavljajući sve poruke osim fatalnih grešaka.
- P	Koristi neprilegirani priključak.
- t	Iznuđuje pseudo-tty dodjeljivanje.
- V	Aktivira verbose način rada (koristan za ispravljanje pogrešaka)
- X	Onemogućuje X11 prosljeđivanje.
- C	Zahtijeva sažimanje svih podataka.
- L	Određuje lokalni priključak za prosljeđivanje do određenog udaljenog domaćina i priključka.
- R	Određuje udaljeni priključak za prosljeđivanje do lokalnog domaćina i određenog priključka.

#### Tablica 30.4 Opcije naredbenog retka za naredbu ssh

## Odavde...

Više informacija o Internetu možete pronaći u sljedećim poglavljima:

- 31. poglavlje, "Kretanje Internetom pomoću World Wide Weba" opisuje različite vrste informacija raspoloživih na Internetu i alate za pristup tim informacijama.
- 33. poglavlje, "Uporaba elektroničke pošte" pokazuje kako e-poštu slati i primati preko Interneta.

## poglavlje31

## Kretanje Internetom pomoću World Wide Weba

(Steve Burnett)

## U ovom poglavlju

Predstavljanje World Wide Weba Uporaba FTP-a s Web pretraživačem Uporaba archiea s Web pretraživačem Uporaba telneta s Web pretraživačem Uporaba gophera s Web pretraživačem Pristup Usenet novostima s Web pretraživačem Prijava na poštanske popise (Mailing lists) Uporaba Wide Area Information Servera (WAIS) Čuli ste da su na Internetu raspoložive sve vrste informacija. To je istina. Tu ćete naći sve, od najnovije fotografije s meteorološkog satelita do softvera, statistika i *online* kupovine.

U ovom se poglavlju govori o uslugama koje možete koristiti za dobivanje informacija na Internetu. Možete koristiti World Wide Web, FTP, gopher, WAIS ili archie. Najvjerojatnije ćete koristiti neku kombinaciju. U ovom se poglavlju opisuju sve glavne vrste usluga i prikazuju neke osnovne informacije o načinu uporabe tih usluga. Budući da Web pruža tako lagan pristup informacijama na Internetu i budući da se iz vašeg Web pretraživača mogu koristiti i mnogi drugi servisi, ovo se poglavlje usredotočuje na uporabu tih servisa iz Web pretraživača.

## Predstavljanje World Wide Weba

Internet je u potpunosti raspršena mreža, što znači da je vaše računalo spojeno ne samo s računalom na drugoj strani hodnika, nego s tisućama drugih širom svijeta. Vaše se računalo spaja s drugim računalom, koje je pak spojeno s drugim računalima i tako dalje.

Da bi sve bilo još složenije, Internet ima međunarodni opseg. Doslovce svaka zemlja na svijetu ima neki oblik pristupa na Internet. I prije je, već godinama bilo mnogo servisa za dobivanje informacija (FTP, gopher itd.), ali ni jedan nije bio lak za uporabu. Trebali ste imati sav odgovarajući softver, zatim ste trebali znati koji servis koristiti i kada, i tako dalje. Da bi se korisnicima olakšalo dobivanje informacija na Internetu, bilo je potrebno nešto poput World Wide Weba, kao oblik "informacijskog navigatora".

Web je započet 1989. godine u CERN-u, Evropskom laboratoriju za fiziku čestica, kao projekt mreže i hiperteksta. Istraživači su uvidjeli potrebu da ljudi mogu dijeliti i razmjenjivati informacije i dokumente u stvarnom vremenu iz bilo koje lokacije i iz bilo kojeg računala. Također su željeli jednostavan i pouzdan način baratanja tim informacijama. Iz toga je nastao Web.

Web koristi skup hipertekstualnih veza koje korisnicima omogućuju lako kretanje među dokumentima, grafikama, datotekama, zvukovnim isječcima itd., iz lokacija bilo gdje na Internetu. Kada u dokumentu odaberete hipertekstualnu vezu, sadržaj (bez obzira kakav) na koji veza ukazuje se automatski učitava. Skakanjem s jedne na drugu vezu, korisnici Interneta brzo pronalaze put do različitih informacija koje trebaju.

## Razumijevanje strukture Weba

Web se temelji na modelu klijent/poslužitelj. Softverski paket klijenta (to je Web pretraživač u vašem računalu) stupa u vezu s računalom poslužiteljem (softver za Web poslužitelja) i s tim računalom razmjenjuje poruke pomoću skupa pravila koja i klijent i poslužitelj razumiju. Takav je skup pravila poznat kao *protokol*. Web polužitelji i klijenti komuniciraju pomoću protokola nazvanog Hypertext Transfer Protocol (HTTP). Kada Web klijent iz Web poslužitelja prima dokument, programi vjerojatno komuniciraju uporabom HTTP-a. Kao što ćete vidjeti kasnije u ovom poglavlju, Web poslužitelji podržavaju i druge Internet protokole.

#### O klijentima i poslužiteljima

Odnos klijent/poslužitelj je vrlo važan u mrežnom sustavu, a naročito u kretanju po Webu. *Poslužitelj* (server) je računalo koje pruža usluge koje druga računala mogu koristiti. *Usluge* (servisi) mogu biti bilo kakva vrsta programa, rutina ili podataka koje poslužitelj pruža. Na primjer, poslužitelj vas može opskrbiti informacijom iz baze podataka u koju vi nemate izravan pristup.

*Klijent* je računalo koje koristi usluge iz poslužitelja. Klijent uspostavlja vezu s poslužiteljem i traži neku vrstu usluga. U mnogim slučajevima klijent računalo koristi poseban softver napravljen za međudjelovanje s posebno napravljenim programom poslužiteljem na računalu poslužitelju.

U tom modelu klijent/poslužitelj, osobe s različitim računalima na različitim lokacijama mogu pristupiti informacijama na istom poslužitelju. Moguće je složiti različita računala-poslužitelje s različitim vrstama podataka. Budući da korisnici za komunikaciju s poslužiteljom koriste klijent program, za svaku je računalnu platformu koju koriste moguće razviti drukčiji klijent program. Na taj način, osobe koje koriste Windowse ili Macintosh mogu koristiti klijent softver za pristup informacijama na UNIX ili Linux poslužitelju jednako lako kao što to mogu korisnici UNIX-a ili Linuxa.

Za pristup Webu trebate klijent softver poznat kao *Web pretraživač* (Web browser). Web pretraživač je program koji razumije kako s Web poslužiteljem komunicirati putem HTTP protokola, prikazivati informacije i kako pružiti način za prikaz hipertekstualnih veza. Na r aspolaganju ima mnogo pretraživača. Trenutno su najčešće korišteni pretraživači Netscapeov Navigator i Microsoftov Internet Explorer. Pretraživač možete nabaviti na brojne načine - od vašeg davatelja Internet pristupa, možete ga kupiti u trgovini, učitati s Interneta itd. Nakon što instalirate pretraživač i softver podesite s informacijama za vaš pristup na Internet, sve e spremno za početak.

#### Razumijevanje URL-ova

Informacije s Weba dobivate uporabom opisne adrese poznate kao *uniform resource locator* (URL). URL možete zamisliti kao usmjerivač na objekt na Internetu koji vam ne kaže samo gdje se objekt nalazi, već i kako je nazvan i kako do njega doći. Sve čemu pristupate na Webu ima URL.

Sintaksa za URL-ove možda izgleda zastrašujuće, ali je zapravo prilično jednostavna. Evo primjera:

#### http://www.ncsa.uiuc.edu/SDG/Software/Mosaic/Docs/whats-new.html

Zastrašujuće? To stvarno nije toliko strašno. Dio lijevo od dvotočke određuje način pristupa za dobivanje podataka. Taj način pristupa definira protokol koji se koristi za vezu s poslužiteljem, a također se lako može zaključiti i kakva će se vrsta interakcije odvijati. Tablica 31.1 pruža popis nekoliko valjanih načina pristupa.

Tablica 31.1	Valjane metode pristupa za URL-ove
Metoda	Opis
http	Protokol za pristupanje većini Web stranica. Pruža interaktivne hipermedijske veze do stranica napisanih u HTML jeziku (Hypertext Markup Language).
wais	Koristi se za pristupanje WAIS lokaciji (Wide Area Information Service).
gopher	Koristi se za pristup gopher poslužitelju.
ftp	Pruža anonimnu FTP vezu
telnet	Otvara telnet vezu do lokacije
news	Koristi se za čitanje Usenet novosti.

#### Prije Weba, bilo je...

Mnogi su servisi i izvori informacija postojali i prije Weba. Ti servisi koriste drukčije protokole od HTTP-a. Ipak, mnogi vam Web klijenti poput Netscape Navigatora omogućuju pristup tim servisima iz samog pretraživača. Na primjer, možete datoteke u vaše računalo prenositi uporabom FTP protokola, učitavati dokumente iz gopher poslužitelja, vršiti traženja teksta s WAIS-om (Wide Area Information Service) i čitati Usenet novosti.

Nakon :// u URL-u slijedi ime domaćina na računalu poslužitelju na koje se želite spojiti. Iza imena poslužitelja slijedi staza direktorija do dokumenta koji želite pregledati ili učitati. Ova staza u potpunosti ovisi o tome gdje je datoteka smještena u udaljenom poslužitelju. (U nekim slučajevima možda nećete trebati stazu, ako je datoteka u unaprijed zadanom direktoriju). Na kraj se stavlja ime datoteke dokumenta. Taj dokument može biti tekst, hipermedijski dokument, zvukovna datoteka, grafika ili neka druga vrsta datoteke.

Dakle, pogledajte još jednom ovaj primjer. Sljedeći URL

#### http://www.ncsa.uiuc.edu/SDG/Software/Mosaic/Docs/whats-new.html

koristi HTML protokol za vezu s računalom poslužiteljem **www.ncsa.uiuc.edu** i kaže mu da vas zanima dokument nazvan whats-new.html, smješten u direktoriju /SDG/Software/Mosaic/Docs. Nastavak .html u imenu dokumenta vašem Web klijentu (na primjer, Netscape Navigatoru) kaže da je dokument napisan u HTML jeziku (Hypertext Markup Language). HTML je posebna sintaksa koja se koristi za pisanje hipertekstualnih stranica za Web. Za više pojedinosti o HTML-u pogledajte 32. poglavlje, "Stvaranje Web dokumenata s HTML-om".

## Pretraživanje Weba

Web je golem i svakog dana postaje sve veći. Umjesto klikanja kroz tisuće stranica, pretraživački servisi vam mogu pomoći u bržem pronalaženju informacija. Pretraživački servis (*search engine*) je program koji u svojoj bazi podataka traži informaciju koja odgovara vašem zahtjevu. Neki pretraživački servisi, poput AltaVista i Infoseeka, pretražuju cijeli Web i svoje informacije pohranjuju u goleme baze podataka. Drugi pretraživački servisi pretražuju samo određenu Web lokaciju. Kada na uobičajenoj Web lokaciji ugledate gumb Search, on obično služi samo za tu Web lokaciju. Kada želite pretražiti cijeli Web, za to vam treba općenitiji alat za pretraživanje. Sljedeći popis opisuje neke od mnogih pretraživačkih servisa koji pretražuju Web lokacije po cijelom Internetu. Neki vam čak omogućuju pretraživanje drugih izvora informacija na Internetu, poput onih na Usenet i FTP lokacijama.

- Alta Vista (http://www.altavista.digital.com), za Web i Usenet. Tu možete naći sve, bilo gdje na Webu ili Usenetu. Svoje ćete traženje trebati što je moguće više suziti, jer natrag suviše lako dobivate previše rezultata!
- Yahoo! (http://www.yahoo.com), za Web, Usenet, e-mail adrese, trenutne vijesti, traženje osoba, planove gradova i stanje na burzi. Yahoo! nije pravi pretraživački servis. On je u osnovi veliki popis Web lokacija razvrstanih u kategorije, koje su predložili korisnici. Koristan je za uobičajene informacije i za dobivanje pregleda koliko mnogo različitih informacija na Webu postoji. Yahoo! pruža i veze do drugih pretraživački servisa.
- Infoseek (http://www.infoseek.com) za Web, Usenet, FAQ-ove, trenutne vijesti, e-mail adrese, karte, burzovne izvještaje i popise tvrtki. Infoseek sadrži alat za pretraživanje kao i golemi popis i dobar je kada želite pretražiti i više od Weba ili Useneta. Infoseek koristi jezik za pretraživanje koji je različit od drugih pretraživačkih servisa.
- Open Text Index (http://index.opentext.net) za Web, Usenet, trenutne vijesti i e-mail adrese. Može poslužiti kao laka-za-uporabu zamjena za Alta Vistu i dobar je kada tražite neke nejasne teme. Moguće je pretraživati i na drugim jezicima, poput japanskog i španjolskog.
- Excite (http://www.excite.com) za Web, Usenet i Excite preglede Web lokacija. Excite vrši koncepcijsko pretraživanje Weba i dobar je kada niste sigurni za pojam koji tražite. Excite je besplatan, budući da koristi jednolokacijske pretraživačke alate na mnogim Web lokacijama.
- Lycos (http://www.lycos.com/) za Web, FTP lokacije i gopher lokacije. Lycos je sličan Yahoo!-u. Dobar je za jednostavna pretraživanja o uobičajenim temama. Možete tražiti i zvuk, grafiku ili neki subjekt.
- Search.Com (http://www.search.com/) za Web i Usenet. Ovaj vam pretraživački servis omogućuje i pretraživanje drugih pretraživačkih servisa poput Alta Viste, HotBota ili Infoseeka. Search.com pruža A-Z popis drugih pretraživačkih servisa i ima zgodan pomoćni program koji predlaže koji će pretraživački servisi pronaći ono što trebate.
- Inference Find! (http://www.inference.com/) samo za Web. On sam nije pretraživački servis. Inference Find! grupira rezultate drugih pretraživačkih servisa i uklanja ponavljanja. Trenutno naziva WebCrawler, Yahoo!, Lycos, Alta Vistu, InfoSeek i Excite.
- HotBot (http://www.hotbot.com) za Web i Usenet. HotBot je dobar za pronalaženje lokacija koje koriste određenu tehnologiju, poput JavaScripta ili VRML-a. Svoje traženje možete suziti i na određena geografska područja (poput Evrope), klasu domena (poput edu) ili jednu Web lokaciju (poput www.apple.com).

Dobre ključne riječi vaše traženje čine djelotvornijim. Zadajte riječi koje su jedinstvene za ono što stvarno želite pronaći. Pokušajte izbjeći često korištene pojmove, poput *www*,

*Internet, computer* itd. Ako ih baš trebate, koristite ih s drugim posebnijim pojmovima i Boolean operatorima koji će vam pomoći u sužavanju vaše potrage, kao u ovom primjeru:

WWW and "Search Engines"

BILJEŠKA Većina pretraživačkih servisa dopušta i uporabu navodnih znakova ("") za traženje izraza. Za određene pojedinosti provjerite sustav pomoći vašeg pretraživačkog servisa. ■

Vjerojatno ćete ustanoviti da vam čak i pretraživački servis daje preveliki popis lokacija koje biste trebali pogledati. Broj pronađenih lokacija možete smanjiti sužavanjem vašeg traženja. Ispravnom uporabom nekih jednostavnih pojmova - AND, OR i NOT možete tisuće pronađenih lokacija suziti na tek nekoliko.

AND (i), OR (ili) i NOT (ne ili nije) nisu one iste riječi koje koristite u svakodnevnom životu. One dolaze iz simboličkog logičkog sustava koji je u 19. stoljeću razvio matematičar George Boole. Boolean pretraživanje koristi osnovnu sintaksu izvedenu od operatora i pojmova za traženje. Budući da ti pojmovi nemaju točno isto značenje kao u engleskoj gramatici, pobrinite se da ih pravilno shvatite. Tablica 31.2 pokazuje primjere uporabe AND, OR i NOT.

Tablica 31.2	Korisni Boolean izrazi
Izraz	Opis
AND ili +	Vraća stranice koje sadrže sve vaše pojmove za traženje. Ako sve te riječi nisu na stranici, stranica se ne prikazuje. AND ili + koristite kada imate različite pojmove i pronađeni rezultat želite suziti na nekoliko preciznih pogodaka. Na primjer, BMW AND roadster ili BMW + roadster će prikazati samo stranice koje sadrže i BMW i roadster.
OR	Vraća stranice koje sadrže bilo koji od traženih pojmova. OR koristite za vraćanje stranica koje sadrže bilo koji od pojmova navedenih u potrazi. Na primjer, BMW OR roadster će prikazati sve stranice koje sadrže ili BMW ili roadster ili oba pojma.
NOT	Vraća stranice koje ne sadrže riječi navedene za traženje (ne podržavaju ga svi pretraživački servisi).

Nemojte se bojati isprobavati. Iskušajte nekoliko različitih vrsta traženja s istim ciljem na umu tako da dobijete bolji osjećaj za rezultate koje neki od izraza i vaše riječi ili fraze za traženje vraćaju. Ustanovit ćete da isprobavanjem pojmova traženja postajete sposobniji u sužavanju vašeg odabira na prihvatljivu veličinu.

Sada kada smo obradili URL-ove i pretraživanje Weba, pogledajmo s više pojedinosti neke druge načine pristupa navedene ranije u tablici 31.1. Svaki članak pruža opis servisa, kako mu pristupiti sa i bez pretraživača i uzorak informacija koje takav servis vraća.

## Uporaba FTP-a s Web pretraživačem

FTP ili File Transfer Protocol je metoda koju Internet koristi za razmjenu datoteka između računala. Bez obzira što tražite - softver, dokumentaciju, FAQ popise, programe ili bilo što drugo, kopiju toga vjerojatno možete dobiti preko anonimnog FTP-a.

Anonimni FTP je servis koji vam omogućuje prikupljanje podataka širom Interneta bez potrebe da na tom stroju imate korisnički račun. Uporabom anonimnog FTP-a možete pristupiti svim datotekama koje su administratori sustava na udaljenim računalima stavili na javno raspolaganje.

**Vidi** "Uporaba FTP-a za daljinski prijenos datoteka", str. 580.

FTP podržava ASCII način prijenosa za tekstualne datoteke i binarni način prijenosa za druge vrste datoteka. Na sreću, većina Web klijenata za vas automatski utvrđuju vrstu datoteka pa se o tome ne trebate brinuti. Vrstu arhive ili programa za sažimanje koji je za neku datoteku korišten obično možete utvrditi prema nastavku imena datoteke. Tablica 31.3 pruža popis najčešćih nastavaka imena datoteka na koje ćete naići.

Tablica 31.3	Uobičajeni nastavci imena binarnih datoteka raspoloživih preko FTP-a
Nastavak	Opis
.Ζ	Komprimirana s UNIX programom compress
.Z	Vjerojatno komprimirana s GNU programom gzip ili s UNIX programom compress
.gz	Komprimirana s GNU programom gzip
.tar	Arhiva od nekoliko datoteka stvorena u UNIX programu tar
.zip	Arhiva od nekoliko datoteka stvorena s programom pkzip

Ponekad ćete naići na datoteke koje su stvorene na više ovih načina. Na primjer, datoteka programs.tar.Z je arhiva od nekoliko datoteka stvorena s uslužnim programom tar i zatim komprimirana s uslužnim programom compress.

Ako Web klijent poput Netscape Navigatora (koji ima ugrađenu FTP podršku) želite koristi za izvršavanje anonimnih FTP prijenosa, dio za protokol u URL-u zamijenite s "ftp". Na primjer, za pokretanje anonimne FTP *sesije* na **sunsite.unc.edu** koristite sljedeći URL:

#### ftp://sunsite.unc.edu

Ovaj će URL vašeg Web klijenta navesti da pokuša uspostaviti FTP vezu s **sunsite.unc.edu** i prijaviti vas kao anonimnu FTP *sesiju*. Nakon uspostavljanja FTP *sesije*, klikanjem na prikazane hiperveze možete pretraživati direktorije i prenositi datoteke.

**BILJEŠKA** Većina anonimnih FTP poslužitelja zahtijeva da kao lozinku koristite vašu e-mail adresu. Ako imate problema, provjerite da li su vaše e-mail postavke ispravno podešene

u vašem pretraživaču.

Za određivanje ne-anonimne FTP sesije u Netscapeu, unesite

ftp://username@ftp.startup.com

gdje je *username* vaše korisničko ime, a *ftp.startup.com* adresa na koju želite ići. Program će zatim od vas zatražiti unos vaše lozinke.

BILJEŠKA Kada odaberete tekstualnu datoteku za prijenos iz udaljenog poslužitelja u FTP sesiju, većina će klijenata datoteku prikazati na zaslonu. Datoteku trebate spremiti na disk, što ćete učiniti odabirom odgovarajuće naredbe u izborniku. Neki vam Web pretraživači omogućuju da odaberete učitavanje datoteke na vaš disk umjesto na zaslon.



#### NA WEBU

Za dobivanje popisa FTP lokacija preko Weba pogledajte sljedeću stranicu:

http://www.yahoo.com/Computers and Internet/Internet/FTP Sites/

## Uporaba archiea s Web pretraživačem

Baš kao i na Webu, jedan je od glavnih problema na anonimnom FTP-u ustanoviti gdje su datoteke koje vas zanimaju smještene na Internetu. Za pomoć korisnicima u pronalaženju datoteka stvoren je sustav archie, archie je u osnovi pretraživački alat za anonimne FTP lokacije.

archie je program za pretraživanje baza podataka koji se spaja na anonimne FTP lokacije širom svijeta i od svake lokacije traži potpun popis svih njenih datoteka. archie zatim te informacije indeksira u svojoj vlastitoj internoj bazi podataka. U toj bazi podataka možete tražiti lokacije datoteka na Internetu. Budući da je osvježavanje archievih baza podataka očito postupak za koji treba vremena, te se baze podataka obično obnavljaju jednom mjesečno. Stoga je, premda malo vjerojatno, moguće da informacija koju vam je archie pružio nije točna.

archie je omiljen servis. Razni archie poslužitelji širom svijeta mogu biti vrlo opterećeni i tada za rješavanje zahtjeva treba više vremena. Neke lokacije postavljaju ograničenja na broj istovremenih veza i tako sprečavaju preveliko usporavanje rada poslužitelja. Ako isprobate neki archie poslužitelj i ustanovite da je preopterećen, možete ili pokušati na drugom poslužitelju ili pričekati nekoliko minuta i ponovno pokušati.

#### NA WEBU



Mnogi archie poslužitelji sada podržavaju upite putem Web obrazaca. Za pretraživanje archiea idite na sljedeći URL za popis archie prolaza na Web. Od te se stranice možete povezati na mnoge od zrcalnih stranica archie baze podataka. Obično je najbrže povezati se na vama najbližu lokaciju.

#### http://www.nexor.co.uk/public/archie/servers.html

Tablica 31.4 prikazuje popis nekih od širom svijeta raspoloživih archie poslužitelja.

Tablica 31.4 Aktivni archie poslužitelji				
Poslužitelj	IP adresa	Lokacija		
archie.unl.edu	129.93.1.14	SAD (NE)		
archie.internic.net	198.49.45.10	SAD (NJ)		
archie.rutgers.edu	128.6.18.15	SAD (NJ)		
archie.ans.net	147.225.1.10	SAD (NY)		
archie.sura.net	128.167.254.179	SAD (MD)		
archie.au	139.130.4.6	Australija		
archie.uni-linz.ac.at	140.78.3.8	Austrija		
archie.univie.ac.at	131.130.1.23	Austrija		
archie.cs.mcgill.ca	132.206.51.250	Kanada		
archie.uqam.ca	132.208.250.10	Kanada		
archie.funet.fi	128.214.6.102	Finska		
archie.univ-rennes1.fr	129.20.128.38	Francuska		
archie.th-darmstadt.de	130.83.128.118	Njemačka		
archie.ac.il	132.65.16.18	Izrael		
archie.unipi.it	131.114.21.10	Italija		
archie.wide.ad.jp	133.4.3.6	Japan		
archie.hana.nm.kr	128.134.1.1	Koreja		
archie.sogang.ac.kr	163.239.1.11	Koreja		
archie.uninett.no	128.39.2.20	Norveška		
archie.rediris.es	130.206.1.2	Španjolska		
archie.luth.se	130.240.12.30	Švedska		
archie.switch.ch	130.59.1.40	Švicarska		
archie.twnic.net	192.83.166.10	Tajvan		
archie.ncu.edu.tw	192.83.166.12	Tajvan		
archie.doc.ic.ac.uk	146.169.11.3	Velika Britanija		
archie.hensa.ac.uk	129.12.21.25	Velika Britanija		

Za spajanje na jedan od ovih poslužitelja koristite telnet i prijavite se kao archie. Svaki je poslužitelj nešto drukčiji, ali ih je većina u osnovi ista. Nakon prijavljivanja na poslužitelj pojavit će se naredbeni redak poput

archie>

\_

u koji možete unjeti vaše naredbe za traženje. Različiti poslužitelji imaju različite unaprijed zadane vrijednosti za traženje. Unaprijed zadana podešenja za poslužitelj na koji ste spojeni možete utvrditi uporabom naredbe show search. Naredba show search vraća jednu od sljedećih vrijednosti:

regex	archie vaš niz znakova za traženje tumači kao običan UNIX izraz.
exact	Niz znakova koji ste zadali mora točno odgovarati imenu datoteke.
sub	Pretraga za nizom znakova bit će uspješna ako se vaš zadani niz pronađe kao podniz u imenu datoteke.
subcase	Slično sub vrsti traženja, osim što i zadane veličine slova moraju biti jednake pronađenima.

Željenu vrstu traženja možete odrediti uporabom naredbe set search kako je ovdje prikazano:

archie> set search search-type

Kada ste vaše traženje podesili na željeni način, nakon toga za traženje prema imenu datoteke koristite naredbu prog. Na primjer, sljedeći par naredbi

```
archie> set searach sub
archie> prog linux
```

u archie bazi podataka izvršava traženje svih datoteka koje sadrže podniz linux. Za svaku pronađenu datoteku koja odgovara zadanim uvjetima traženja, archie izvještava računalo domaćin koje tu datoteku drži, zajedno s punom stazom do datoteke u tom domaćinu.

Ako vas uporaba archiea zbunjuje ili zatrebate pomoć, samo u naredbeni redak archie> utipkajte help. Dobit ćete informaciju o dobivanju pomoći u archieu. U naredbeni redak help> utipkajte ? za prikaz popisa podtema o kojima možete dobiti pomoć.

Nakon što pronađete informaciju koju tražite, za izlazak iz archiea u naredbeni redak archie> utipkajte **exit** ili **quit**.

## Uporaba telneta s Web pretraživačem

telnet postoji gotovo od samog nastanka Interneta. Uporabom telneta možete se spajati na baze podataka, kataloge biblioteka i druge izvore informacija širom svijeta. Želite li vidjeti kakvo je vrijeme u Vermontu? Provjeriti stanje usjeva u Azerbajdžanu? Dobiti više informacija o nekome čije ste ime vidjeli negdje na Internetu? telnet vam omogućuje sve to pa i više. Kada se s telnetom želite priključiti na drugo računalo, prelazite preko Interneta i prijavljujete se na taj stroj. Tu nećete naći grafiku kao na Webu - telnet radi samo s tekstom.

gopher je još jedan od ranih Internet alata i mnoge se telnet lokacije najlakše pronalaze preko BILJEŠKA gopher izbornika. Pogledajte sljedeći članak o gopheru. Za pokretanje telneta iz vašeg pretraživača unesite URL telnet lokacije na koju želite ići. Na primjer,

telnet://pac.carl.org

će pokrenuti program telnet i odvesti vas na lokaciju koju ste upisali. Tu ste izašli izvan pretraživača i ušli u "zemlju izbornika".

Vidi "Uporaba telneta za pristup udaljenim račuanlima", str. 578.

#### Podešavanje Netscapea za rad s telnetom

telnet vjerojatno nije ugrađen u vaš pretraživač. Program telnet trebate nabaviti, instalirati ga u vaše računalo i zatim vaš pretraživač podesiti za njegovu uporabu. Ovdje je primjer uputa za podešavanje telneta za Netscape:

- 1. U Netscape Navigatoru otvorite izbornik Options i odaberite Preferences.
- 2. Na raspoloživim karticama odaberite Applications i Directories.
- 3. Uz prozor Telnet Application odaberite Browse.
- 4. Pronađite i odaberite izvršnu datoteku telneta.
- 5. Pritisnite <Return>. Netscape je sada podešen.

Windowsi 95 i Windowsi NT aplikaciju Telnet imaju u svojoj polaznoj mapi (Windows). Za Macove i starije inačice Windowsa je omiljen izbor NCSA Telnet.

Većina je telnet lokacija prilično laka za uporabu i imaju ugrađene sustave pomoći. Također, većina ih radi najbolje -, a u nekim slučajevima i samo - s emulacijom VT100. Mogli biste ustanoviti i da su mnogi od tih resursa sada raspoloživi i na Webu.

## Uporaba gophera s Web pretraživačem

gopher je servis na Internetu koji vam pristup informacijama omogućuje vršenjem odabira u nizu izbornika. gopher je bio jedan od prvih Internet servisa koji su napravili ozbiljan pokušaj u ponudi korisnički-prijateljskog sučelja.

Kada se spajate na lokaciju koja pruža gopher usluge, dobivate izbornik s raspoloživim mogućnostima. Svaka je opcija u izborniku ili datoteka ili drugi izbornik. Vaš izbor možete odabrati iz izbornika bez potrebe za poznavanjem imena ili IP adrese odredišne lokacije ili imena direktorija i datoteka za određenu informaciju koju tražite. gopher se tim pojedinostima bavi umjesto vas.

**BILJEŠKA** Ni jedan informacijski resurs na Internetu nije napravljen isključivo za gopher. Sve što možete dobiti preko gophera je pristupačno i na druge načine, poput HTML Web stranice, FTP-a ili telneta. U nekim su slučajevima, zbog sigurnosnih razloga, neke lokacije odlučile resurse staviti na raspolaganje samo preko gophera. Za pristup gopher poslužitelju s Web pretraživačem, protokolni dio u URL-u promijenite tako da umjesto http stoji gopher. Na primjer, URL za gopher poslužitelj na sunsite.unc.edu je

#### gopher://sunsite.unc.edu

gopher omogućuje lako kretanje po Internetu. Na žalost, informacija koju gopher može vratiti možda neće biti dobro organizirana, stoga za pronalaženje onog što želite trebate imati i nešto pustolovnog duha. Budući da su stavke u Gopherprostoru predstavljene kao skupovi izbornika, da biste došli do datoteke koju tražite ponekad se trebate probijati kroz mnoge različite izbornike. Međutim, ako to stavimo na stranu, preko gophera je raspoloživo mnoštvo korisnih informacija.

Jedan od nedostataka gophera je nepostojanje standardnih popisa tema za različite gopher poslužitelje. Administratori su za svaki gopher poslužitelj svoje informacije organizirali prema vlastitim sklonostima. To znači da svaki gopher poslužitelj kojem pristupite ima različite teme. Ako neki gopher poslužitelji slučajno imaju i neke od istih tema, one vjerojatno nisu imenovane na isti način.



Budući da Gopherspace ("Gopherprostor") postoji dulje od World Wide Weba, a slabije je organiziran, prevelik je za nasumično pretraživanje. veronica je poput archiea, osim SAVJET što ona pretražuje gopher poslužitelje. Za više informacija o veronici pogledajte gopher://gopher.scs.unr.edu/00/veronica/veronica-fag.

#### NA WEBU

Za popis gopher lokacija pogledajte



http://www.yahoo.com/Computers and Internet/Internet/Gopher/

## Pristup Usenet novostima s Web pretraživačem

Prema najjednostavnijoj bi definiciji Usenet novosti, koje se nazivaju i netnews ili naprosto news (novosti), bile forum za rasprave na Internetu. Mnoga računala širom svijeta razmjenjuju male količine informacija, nazvane articles (članci) o gotovo svakoj zamislivoj temi. Ta računala nisu fizički spojena na istu mrežu. Ona su logički povezana njihovom sposobnošću razmjenjivanja podataka. Za potpuniju informaciju o Usenet novostima pogledajte 34. poglavlje, "Preživljavanje Usenet novosti".

Clanci s novostima na Usenetu su raspodijeljeni u novinske grupe (newsgroups) prema temama. Te su grupe zatim podijeljene u hijerarhije na temelju vrlo općenitih razlika u temama.

Vidi "Struktura Useneta", str. 659.

U novinskim se grupama vode razgovori i rasprave o gotovo svakoj temi koju možete zamisliti. To je vrlo dobar način za pronalaženje i razmjenu informacija.

## Prijava na poštanske popise (Mailing lists)

Još jedna široka tema za raspravu o Internetu dolazi iz poštanskih popisa (Mailing lists) za e-poštu. Poštanski popisi se od Useneta razlikuju po tome što se razne poruke i članci u raspravama šalju putem e-pošte, a ne putem medija za Usenet novosti.

Zašto koristiti poštanske popise umjesto Usenet novinskih grupa? Poštanski su popisi namijenjeni za manji broj osoba. Na Usenetu je prilično teško postaviti novu novinsku grupu jer je potrebno vrijeme za predlaganje, raspravu i glasanje. Svaki administrator sustava može sastaviti poštanski popis. Također, budući da se svaki poštanski popis o država na jednom računalu, administrator sustava ima veću mogućnost nadzora. On može odlučivati tko može biti na popisu, a problematičnih se korisnika može riješiti na djelotvorniji način. Neki su poštanski popisi, poput onih koji raspravljaju o pitanjima računalne sigurnosti, ograničeni na određene osobe. Ako trebate biti na jednom od tih popisa, od upravitelja popisa trebate zatražiti dopuštenje za pretplatu.

## Pronalaženje poštanskih popisa

Kao i Usenet novosti, poštanskih popisa također ima za najrazličitije teme. Cjeloviti se popis javno raspoloživih poštanskih popisa redovito šalje u Usenet novinsku grupu **news.answers**.



#### NA WEBU

Poštanske popise možete potražiti i preko Weba na

http://www.liszt.com/

## Uporaba poštanskih popisa

Poštanski popisi su obično postavljeni uporabom *mail reflectora. Mail reflector* je posebna e-mail adresa koja je podešena tako da se sva pošta koja je na nju poslana *reflektira* (odbija) natrag prema skupini ljudi. Poštanskom popisu su obično pridružene dvije e-mail adrese - jedna od osobe koja taj popis održava, a druga od samog popisa.

Pretpostavimo da postoji e-mail adresa za korisnike nekih uređaja - te ćemo uređaje ovdje nazvati *widgets*. E-mail adresa bi za taj popis mogla biti nešto poput widgets@somewhere.com. Ako pošaljete e-mail poruku na adresu tog popisa, ona će biti reflektirana svima koji su na taj popis "pretplaćeni".

Prema dogovoru, poštanski popisi na Internetu za administrativne zahtjeve, poput "pretplaćivanja" na popis, koriste posebnu e-mail adresu. Ta je adresa sastavljena dodavanjem -request imenu popisa. Tako bi e-mail adresa za zamišljeni poštanski popis widgets bila widgets-request@somewhere.com. Sva pošta u vezi administrativnih pitanja treba biti poslana na administrativnu adresu.

Svaki poštanski popis (i Usenet novinska grupa) ima svoja vlastita pravila i podkulturu. Prije slanja pošte na takav popis trebali biste upoznati njihove običaje. Obično pri "pretplaćivanju" na popis dobijete uvodnu poruku i popis često postavljanih pitanja (FAQ). Uvodna poruka sadrži i sva posebna pravila koja se odnose na taj popis.

Najprije svakako pročitajte i FAQ zato da ne postavljate ista pitanja koja su već postavljena stotine puta prije vas.

Vidi "Pravila ponašanja na Usenetu", str. 666.

## **Uporaba Wide Area Information Servera (WAIS)**

WAIS (Wide Area Information Servers) je sustav za traženje informacija u velikom skupu baza podataka. Po pojmu *wide area* (široki prostor) se može naslutiti da se može koristiti u velikim mrežama, poput Interneta, za provođenje traženja uporabom klijent/poslužitelj softvera.

Uporabom WAIS-a možete pronaći tekstualne ili multimedijske dokumente koji su pohranjeni u bazama podataka širom Interneta. WAIS možete zamisliti kao nešto slično gopheru, samo što WAIS sam obavlja traženje za vas.

Poput gophera, za uporabu WAIS-a trebate klijent softver ili trebate koristiti telnet za spajanje na lokaciju koja pruža javni pristup za WAIS klijente. Postoji interaktivni UNIX WAIS klijent poznat kao *swais*. Za uporabu tog sustava možete se telnetom spojiti na **sunsite.unc.edu** i prijaviti kao swais. Zatim dobijete izbornik baza podataka koje možete pretraživati.

## Odavde...

World Wide Web je divan način za istraživanje Interneta i napajanje na njegovim golemim resursima. Različiti su Web pretraživači raspoloživi kao besplatan softver ili kao komercijalni proizvodi. Uporaba Web pretraživača olakšava kretanje po Mreži i pronalaženje informacija koje trebate. Više o Webu i Internetu možete saznati u sljedećem poglavlju:

32. poglavlje, "Izrada Web dokumenata pomoću HTML-a", pruža uvod u Hypertext Markup Language koji se koristi za stvaranje stranica za Web.

## poglavlje32

## Izrada Web dokumenata pomoću HTML-a

(Steve Burnett)

## U ovom poglavlju

Razumijevanje HTML-a Rad s HTML-om Uporaba osnovnih HTML elemenata Razumijevanje sintakse HTML-a U 31. poglavlju, "Pretraživanje Interneta pomoću World Wide Weba", naučili ste kako pristupiti World Wide Webu i upoznali ste različite tipove dostupnih informacija. Možete kliknuti na hipertekst veze i skakati s jednog mjesta na drugo onoliko brzo koliko vam dopušta brzina prijenosa. Možete vidjeti Web stranice s privlačnom grafikom i zvukom, popisima, oblicima i razne druge zgodne stvari. Ali kako zapravo stvoriti Web stranicu na koju drugi Web korisnici mogu doći? To nije tako teško kao što mislite. Treba vam pristup Web poslužitelju koji rabi Hypertext Transport Protocol (HTTP) i skup dokumenata napisanih u Hypertext Markup Laguageu (HTML).

U ovom poglavlju pregledat ćemo što je potrebno za izradu Web stranica pomoću HTML-a.

## Razumijevanje HTML-a

Hypertext Markup Language (HTML) je jezik rabljen za razvoj Web stranica i dokumenata. HTML nije programski jezik kao što su C++, Java, Pascal ili Perl; u osnovi to je CPML (*cross-platform markup*) jezik dizajniran dovoljno prilagodljivo da prikaže tekst i druge elemente (npr. grafiku) na različitim preglednicima.

HTML dokument čine posebni *tagovi* umetnuti u ASCII dokument. Te *tagovi* tumače programi Web pretraživača, koji oblikuju i prikazuju dokument.

**BILJEŠKA** HTML je podskup Standard Generalized Markup Languagea (SGML), što je međunarodni standard (ISO 88791) za razmjenu elektroničkih dokumenata. SGML je meta-jezik za definiranje i standardiziranje strukture dokumenata. SGML također opisuje gramatiku koju možete rabiti za dizajniranje drugih markup jezika. Svaki valjan HTML dokument je također valjan SGML dokument. Kao i svaki jezik izveden iz SGML-a, HTML-ova gramatika je opisana Document Type Definition (DTD) datotekom.

HTML govori Web pretraživaču kako prikazati Web dokumente; unatoč tome, oblikovane informacije koje HTML daje su prilično općenite. Za Internet su dostupni mnogi različiti pretraživači kao što su Netscape Navigator, Microsoft Internet Explorer ili NCSA Mosaic. Mnogi se izvode pod grafičkim sučeljima kao što su X Windows system ili MS Windows. Neki, kao na primjer Lynx, su ANSI pretraživači i ograničeni su s obzirom na grafičke karakteristike koje mogu prikazati.

Dok pišete HTML dokumente imajte na umu da će izgledati drugačije ovisno koji pretraživač čitatelj rabi. Svi dostupni Web pretraživači oblikuju HTML dokumente koliko god mogu; međutim, ono što vidite možda neće biti ono što vidi netko tko rabi drugačiji pretraživač ili čak isti pretraživač pod drugim operativnim sustavom.

## Rad s HTML-om

Kako se alati za Web razvijaju, stvaranje HTML dokumenata postaje sve jednostavnije. Novi alata sakrivaju od vas stvarni HTML kôd. Vi samo trebate napisati riječi, oblikovati svoj dokument i sačuvati ga na prikladnoj lokaciji. Neki pretraživači, kao Netscape Navigator Gold, imaju i editor koji vam omogućuje izradu HTML stranice uz pomoć alata označi-i-klikni. Druge označi-i-klikni alate imaju Microsoft FrontPage, Adobe PageMill, Allaire HomeSite i Macromedia DreamWeaver.

Ako ne želite rabiti HTML-ov editor ili već imate dokumente koje želite staviti na Web, postoji nekoliko softverskih aplikacija koje možete uporabiti za pretvaranje obrade tekstova, stolnog izdavaštva, proračunskih tablica ili drugih dokumenata u HTML. Novije inačice programskih paketa, kao što je Adobe Framemaker će pri spremanju dokumenta uključiti HTML kao opciju.



#### NA WEBU

Slijedeća Web lokacija sadrži mnogo dostupnog softvera za pretvaranje:

#### http://www.yahoo.com/computers\_and\_Internet/Sofware/Internet/World\_Wide\_Web /HTML\_Convertors/

Jednom ćete se htjeti okušati u izradi vlastitog HTML-a. Iako ima mnogo alata koji pomažu pri pisanju HTML-a, vjerojatno ćete otkriti kako vam neće dozvoliti da učinite sve što želite.

Također ćete utvrditi da je s HTML-om relativno lako raditi. Kako se HTML temelji na ASCII markup jeziku, treba vam editor koji će omogućiti čuvanje datoteka u ASCII formatu i Web pretraživač koji možete rabiti za gledanje svojih Web stranica dok ih razvijate. Za razvoj Web dokumenata ne treba vam veza s Internetom. Bilo koji Web pretraživač bi vam trebao omogućiti otvaranje lokalne HTML datoteke i uvid u nju kao

da ste je preuzeli s Interneta.

#### BILJEŠKA

Možda ćete ustanoviti kako je uporaba HTML editora korisno. Neki editori omogućuju odabir teksta koji želite oblikovati i potom primjenjuju HTML *tagova* iz izbornika, pa ne morate sami upisivati *tagove*. Drugi editori izgledaju više kao programi za obradu teksta – odaberete tekst i način oblikovanja koji želite iz traka s alatima. U oba slučaja vjerojatno ćete morati direktno uređivati HTML kako bi dobili točan izgled i dojam kakav želite.

#### NA WEBU

Editore kao što su Sausage Software HotDog/HotDog Pro i Macromedia možete pronaći na slijedećim URL adresama:

#### http://www.sausage.com

#### http://www.macromedia.com

Popis HTML editora naći ćete na slijedećoj adresi:

#### http://www.yahoo.com/Comuters\_and\_Internet/Software/Internet/World\_Wide\_Web /HTML\_Editors/

Prije nego se dalje upoznate sa sintaksom HTML-a pogledajte Web stranicu i njen izvorni kôd HTML-a. Slika 32.1 prikazuje jednostavnu Web stranicu.

Popis 32.1 prikazuje HTML kôd za tu Web stranicu. Ovdje možete vidjeti osnovne elemente HTML stranice.
stranica



**Popis 32.1** Izvorni kôd za jednostavnu HTML stranicu

```
<HTML>
<HEAD>
   <TITLE>Hello Web!</TITLE>
</HEAD>
<BODY>
<H1>Hello!</H1>
This is the body section of a simple HTML page.
<P>
<IMG src="example.gif ">
</BODY>
</HTML>
```

Utvrdit ćete da je HTML prilično neposredan:

- Svi HTML *tagovi* su zatvoreni u kutne zagrade (znakovi < i >). Uzmimo <HTML> *tag* na početku Popisa 32.1 kao primjer.
- Većina HTML tagova zahtjeva uporabu početnog taga (<tag\_name>) i završnog taga (</tag\_name>). Oblikovanje započinjete i završavate postavljanjem tih tagova. Ako ispustite potreban završni tag, neće biti nikakvog oblikovanja; možda čak nećete vidjeti ni tekst.
- Mnogi HTML *tagovi* imaju atribute koji se mogu prilagoditi. *Atribut* vam dozvoljava mijenjanje zadano ponašanje *taga*. Na primjer, ako želite tablicu bez okvira mijenjate je postavljanjem atributa (BORDER=some\_number) unutar <TABLE> *taga*.
- Sve HTML datoteke zahtijevaju <HTML> *tag* na početku datoteke i zatvarajući <HTML> *tag* na kraju. Ti *tagovi* govore Web pretraživaču da je dokument koji obrađuje napisan u HTML-u.

HTML dokument se sastoji od dva logična dijela: glave i tijela. *Glava* sadrži informacije o dokumentu; *tijelo* sadrži informacije samog dokumenta. Kao što ste mogli pretpostaviti, zaglavlje je zatvoreno s <HEAD> i </HEAD> *tagovima*, a tijelo je zatvoreno s <BODY> i </BODY> *tagovima*.

Glava HTML stranice sadržava informacije vezane uz sam dokument. U zaglavlje možete smjestiti različite *tag*ove, ali <TITLE> *tag* je u najširoj upotrebi. Naslov dokumenta – sve što je obuhvaćeno <TITLE> i </TITLE> *tagovima* – pojavljuje se u naslovnoj traci prozora na Web pretraživaču.

Tijelo HTML stranice sadržava većinu elemenata koje vidite kad stranicu gledate pomoću Web pretraživača. Tu unosite sve elemente koje želite da ljudi vide, kao što je tekst, grafika, veze (URL-ovi), popisi i tablice.

**BILJEŠKA** Element <TITLE> identificira ime vaše stranice. Kad netko napravi knjižnu oznaku na stranicu, to je ime koje se rabi. Iako HTML ne ograničava dužinu <TITLE> elementa, trebalo bi dati stranici kratak opisni naslov koji se lako može prikazati. Dobar osjećaj za mjeru dužine naslova je ne više od jedne fraze i ne duže od 60 znakova.

Naslov dokumenta je prikazan u odvojenom prozoru i nije dio samog dokumenta, pa zbog toga tekst naslova mora biti čisti tekst i ne smije imati hiperveze ili oblikovanje teksta.

SAVJET Jako zgodna karakteristika izvornog kôda HTML-a je mogućnost gledanja. Jeste li ste ikad našli zanimljivu stranicu i htjeli znati kako je napravljena? Zavirite! Gledanjem tuđih izvornih kôdova možete puno toga naučiti.

Izvorne kôdove obično možete gledati preko izbornika vašeg pretraživača. Na primjer, da bi vidjeli izvorni kôd HTML-a za stranicu pod Netscapeom, izaberite Document Source iz View izbornika.

Kada ste vidjeli osnove HTML-a (da, to je stvarno to!) vrijeme je da naučite nešto o uporabi *tagova*. Iako ima mnogo različitih *tagova*, uporabom samo nekoliko čestih *tagova* (i njihovim kombiniranjem na različite načine) možete stvoriti izvrsne Web stranice.

## Uporaba osnovnih HTML elemenata

Osnovna HTML sintaksa sastoji se od tri komponente: *tagovi*, atributi i URL-ovi. Svaka od tih komponenti detaljno opisuje kako se ti elementi trebaju oblikovati i prikazati, osobitosti određenih akcija te smještaj drugih datoteka i dokumenata.

## Tagovi

Osnovni građevni element HTML-a su *tagovi* koji govore Web pretraživaču kako prikazati tekst i grafiku zajedno s drugim oblikovanim informacijema. Kao što se sjećate, *tagovi* se pišu u oštrim zagradama (<*tag\_name*>), a većina *tagova* treba i završni *tag* (<*/tag\_name*>). Na slici 32.1 vidjeli ste uporabu <TITLE> *taga*:

```
<TITLE>Hello Web!</TITLE>
```

Ovaj red govori pretraživaču kako oblikovati tekst koji počinje s Hello Web! kao <TITLE>. Vidite li početne (<TITLE>) i završne (</TITLE>) *tagove*? Svi elementi unutar tih *tagova* – u ovom slučaju tekst – su prikazani u naslovnoj traci prozora Web pretraživača. Drugi *tagovi* na sličan način utječu na elemente koje obuhvaćaju.

## Atributi

Ponekad *tagovi* trebaju bolje opisati točnu informaciju, na primjer, gdje se nalazi datoteka. *Atributi* se rabe zajedno sa *tagovima* kako bi dali više detalja o izvršavanju *taga*. Pogledajte, na primjer, slijedeći *tag*:

<IMG src="example.gif">

Ovaj *tag* je <IMG> (slika) *tag* koji govori vašem Web pretraživaču kako prikazati sliku. Ali koju sliku? Ovdje u igru ulazi atribut. U ovom primjeru, polje atributa je SRC="example.gif", što daje detaljne informacije kako protumačiti <IMG> *tag*. U ovom slučaju, example.gif datoteka će biti prikazana kao slika.

SAVJET Korištenje atributa za širinu i visinu s <IMG> *tagom* pomaže pretraživaču da brže prikaže sliku. Ako je u Netscapeu slika u GIF i JPG formatu, možete otvoriti samo tu sliku. Iz View izbornika zatim izaberite View Document Info kako bi vidjeli veličinu slike.

## **URL-ovi**

Web resursima se pristupa preko opisnih adresa poznatim kao *uniform resource locators* (URLovi). Sve čemu pristupate na Webu ima URL. HTML rabi URL-ove za određivanje lokacije potrebne datoteke ili druge Web stranice koje su povezane preko hipertekst veza.

## Razumijevanje sintakse HTML-a

Kao što ste vidjeli, HTML je podijeljen na tri osnovne komponente: *tagove*, atribute i URL-ove. Osnovni građevni element je *tag. Tagovi* se rabe za davanje naredbi Web pretraživaču, dok se atributi i URL-ovi rabe za detaljni opis naredbi

*Tagovi* mogu biti grupirani u nekoliko kategorija, ovisno o njihovoj funkciji. Neki *tagovi* daju informacije o dokumentu kao cjelini, neki se rabe za oblikovanje teksta, a neki za grafiku i hipertekst veze prema drugim dokumentima.

#### NA WEBU

Za popis HTML-ovih tagova i njihovih atributa pogledajte mrežnu inačicu Queovog HTML Quick Reference i Queovog Special Edition Using HTML na sljedećim adresama:

http://www.mcp.com/que/developer\_expert/htmlqr/toc.htm

http://www.mcp.com/que/et/se\_html2/toc.htm

## Uporaba tagova za dokumente

<HTML> i </HTML> tagovi se rabe za obavještavanje Web pretraživača da je dokument koji obrađuje napisan u HTML-u. HTML> tag bi trebao biti prvi tag vašeg dokumenta, a </HTML> tag zadnji.

HTML dokument je podijeljen za zaglavlje i tijelo. Zaglavlje sadržava informacije o dokumentu, a tijelo informacije samog dokumenta. Kao što ste mogli pretpostaviti, zaglavlje je okruženo <HEAD> i </HEAD> *tagovima*, a tijelo <BODY> i </BODY> *tagovima*. Nekoliko *tagova* može biti smješteno u zaglavlje, ali je samo <TITLE> *tag* 

u širokoj uporabi. Naslov dokumenta – što god se nalazi između <TITLE> i </TITLE> *tagova* – se pojavljuje u naslovnoj traci prozora Web pretraživača.

#### **Oblikovanje teksta**

HTML omogućava nekoliko različitih načina oblikovanja teksta za prikazivanje. Nemojte zaboraviti da stvarno oblikovanje teksta vaše Web stranice kontrolira Web pretraživač koji se rabi za gledanje stranice

**Naslovi** HTML podržava šest razina naslova koji mogu biti prikazani u dokumentu korištenjem <H1> do <H6> *tagova*. Slika 32.2 pokazuje kako ti naslovi mogu biti prikazani na Web pretraživaču (u ovom slučaju Netscapeu). Upamtite da su ti naslovi drugačije prikazani na različitim pretraživačima –pismo i veličina ne moraju biti isti na vašem pretraživaču. Popis 32.2 pokazuje izvor.

```
Popis 32.2 Primjer upotrebe zaglavlja
```

```
<HTML>
<HEAD>
<TITLE>Heading example</TITLE>
</HEAD>
<BODY>
<H1>Heading 1<H1>
<H2>Heading 2
</H2>
<H3>Heading 3
```

nastavlja se





Pogledajte lokaciju razine 2 i 3 naslovnih *tagova* u Popisu 32.2. HTML ne zanima nalaze li se *tagovi* na kraju reda ili negdje drugdje. Ti *tagovi* govore pretraživaču da je sav tekst između njih u određenoj razini naslova koju definiraju. Također, uočite da izreka This is plain text nije uokvirena *tagovima*. Ovu frazu pretraživač će prikazati kao obični tekst.

**SAVJET** Obično je dobro rabiti samo tri razine naslova na Web dokumentu. Ako su vam potrebne više od tri razine naslova, razmislite o dodavanju stranica.

**Uobičajeni tekst** HTML omogućuje nekoliko načina oblikovanja normalnog teksta u dokumentima. Web pretraživači ne obraćaju pažnju na kraj reda u HTML datoteci. Isto je i s prijelomima redaka, pa morate rabiti posebne *tagove* koji označuju gdje red završava i počinje odlomak. <BR> *tag* naređuje pretraživaču koji prikazuje dokument umetanje prekida retka. To je poput prijeloma retka na pisaćem stroju. Tekst koji slijedi je pomaknut u slijedeći red.

Želite li napraviti novi odlomak, uporabite <P> *tag*. <P> *tag* ima isti učinak kao i <BR>, osim što većina pretraživača umeće prazan red kao dio odlomka za vizualno odvajanje jednog bloka teksta od drugog. Zbog toga što pretraživači nadziru prikaz teksta, stvarno ponašanje <P> *taga* može varirati. Kada rabite <P> *tag* obično ne trebate </P> završni *tag*. Međutim, potreban je ako rabite bilo koji od <P> atributa.

Ponekad ćete željeti vizualno odvojiti različite dijelove iste stranice. HTML nudi mogućnost povlačenja vodoravne crte preko cijelog prikaza dokumenta. <HR> *tag (horizontal rule)*, se rabi za povlačenje vodoravnih crta. On umeće razmak prije linije pa nije potreban <P> *tag*. Kao i <P> *tag*, ni <HR> ne zahtijeva završni *tag*.

Vidjeli ste nekoliko načina kontroliranja oblikovanja teksta na HTML stranici. Sada pogledajte još jedan kratak HTML primjer da bi vidjeli rezultate prekida retka, oznaka odlomaka i vodoravne crte na prikazu teksta. Popis 32.3 pokazuje primjer HTML-a u kojem su korišteni ti t*agovi* za oblikovanje

#### Popis 32.3 Primjer HTML-a koji pokazuje osnovno oblikovanje teksta

```
<HTMI>
<HEAD>
<TITLE>A Sample Text Formatting Page</TITLE>
</HEAD>
<BODY>
<H2>Text Sample1</H2>
Here is some sample text that is written on separate lines without using
line breaks.<P>
<H2>Text Sample 2</H2>
This sample text has a<BR>
line break in the middle.<P>
<H2>Text Sample 3</H2>
Text before is a paragraph mark.<P>
Text after a paragraph mark.<HR>
Text after a horizontal rule mark.
</BODY>
</HTMI>
```

Slika 32.3 pokazuje ovaj primjer prikazan u Mosaicu



Pretpostavimo da želite prikazati neki tekst, npr. tablicu i želite da prijelom redaka i razmaci ostanu onakvi kakve ste vi unijeli. Možete rabiti <PRE> i </PRE> *tagove* za definiranje predformatiranog teksta. Bilo koji tekst okružen tim *tagovima* prikazan je u pismu jednolikog razmaka među znakovima, a prijelomi i razmaci su ostali isti.

Prikaz stranice možete urediti do određenog stupnja. HTML osigurava *tagove* koji govore pretraživaču da prikaže tekst masnim slovima, podcrtanim i nakošenim. Ti *tagovi* su znani kao *fizički stilovi. Tagovi* za masna slova su <B> i </B>, *tagovi* za podcrtani tekst su <U> i </U> i *tagovi* za nakošena slova su <I> i </I>. Tekst koji želite oblikovati ogradite početnim i završnim *tagom* stila koji želite.

HTML također omogućuje neke *logičke stilove* za oblikovanje teksta. <EM> i </EM> *tagovi* se rabe za označavanje naglašenog teksta koji je obično nakošen. <STRONG> i </STRONG> *tagovi* se rabe za dobivanje jačeg naglaska, obično su otisnuti masno.

Slika 32.4 pokazuje kako se mogu pojaviti različito oblikovani tekstovi.

Tablica 32.1 je kratak popis nekih od najčešćih HTML tagova za oblikovanje teksta.

Tag	Akcija
<b></b>	Čini tekst <b>masnim</b>
<blockquote></blockquote>	Oblikuje tekst sa lijevim i desnim uvlakama
<font></font>	Kontrolira različite aspekte teksta s atributima – na primjer boju teksta (COLOR=rgb value) i veličinu (SIZE=number)

 Tablica 32.1
 Odabrani HTML tagovi za oblikovanje teksta

Tag	Akcija
<i></i>	Pretvara tekst u nakošen.
<pre></pre>	Ostavlja tekst oblikovan jednako kako izgleda.
<strike></strike>	Oblikuje tekst kao <del>preertan</del> .
<u></u>	Podcrtava tekst.
<em></em>	Logički stil; naglašava tekst (obično prikazan nakošeno).
<kbd></kbd>	Logički stil; prikazuje tekst kao stil tipkovnice (većinom prikazan u monospaced pismu)
<strong></strong>	Logički stil; naglašava tekst (obično otisnut <b>masno</b> )

**SLIKA 32.4** Primjeri oblikovanog teksta u Netscapeu

ile Edit View Go	o Bookmarks	Options	Directory	Window	Help
the second sec	Edit Relo	ext_f.htm	Open Print	Find Stop	N
What's New? What's	Cool? Destina	tions Net	Search Pe	ople Software	
Examples of common te	xt formatting used	in HTML.			
Text formatting:					
BOLD					
STRIKE This is strike-	thru formatting.				
Font larger and Font smi	aller				
Bold, italic (Note that t	his is a bold and ar	d italic com	bined.)		
BLOCKOUOTE	. This formatting c	reates left a	nd right indente	ed text. See how it	
looks? This is on	a type of formattir	g that you w	ould probably	apply to text that goes	
for more than on	ie line.				
PRE. When you want t	to preserve the	exact form	atting of a c	thunk of text,	
you can use the pre i	tag. All returns	, spaces,	etc. are kept	t displayed as they	
that text within the	pre tage does n	ot resize.	Jour promet	HARRING, MOLACE	
Logical styles:					
Strong					
Emphasis					
Keyboard					
(a)					

Ovaj fajl je skinut sa sajta www.tutoriali.org

**Raznoliki tekst** *Tagovi* za oblikovanje koji se ne uklapaju u druge kategorije su <ADRESS> i </ADRESS> *tagovi*. Oni se rabe za označavanje adresa, potpisa i ostalog unutar dokumenta. Obično je ovako oblikovan tekst smješten na kraju dokumenta, iza oznake vodoravne crte. Precizno oblikovanje <ADRESS> teksta određuje pojedini Web pretraživač.

## Organiziranje informacija pomoću popisa

Ponekad trebate isporučiti informaciju koja je na neki način logički grupirana . Na primjer, želite prikazati popis grafičkih slika, ili numerirani popis. HTML omogućuje nekoliko različitih načina oblikovanja i prikazivanja popisa informacija. Uporaba popisa u HTML-u je moćan način isporučivanja informacija, jer korisnikov Web pretraživač oblikuje sav tekst u popisu na homogen način. Vi samo trebate odlučiti uklapaju li se te informacije

**Prikazivanje nepobrojanih popisa** *Nepobrojani popis* je tekst odvojen grafičkim znakovima ili drugim znakovima za oblikovanje teksta. Svaki tekst unesen u nepobrojani popis može biti dug nekoliko redaka.

Za stvaranje nepobrojanog popisa rabe se dva skupa *tagova*. <UL> i </UL> *tagovi* određuju početak i kraj popisa, a <LI> *tag* se rabi za označavanje svakog njegovog pojedinog dijela. Popis 32.4 pokazuje HTML za jednostavan nepobrojani popis. Slika 32.5 pokazuje taj popis u Mosaicu.

Popis 32.4 Nepobrojani popis

```
<HTML>
<HEAD>
<TITLE>An Unordered List</TITLE>
</HEAD>
<BODY>
<LI>This is the list item 1.
<LI>This is the list item 2.
<LI>This is the list item 3.
</UL>
</BODY>
</HTML>
```

#### **SLIKA 32.5**

Nepobrojani popis prikazan u Mosaicu.

Document Title: An Unordered List Document URL: file://localhost/html/i  This is listlem 1. This is listlem 2. This is listlem 3.	ila Ontiona	Movidate	Appotato	Ho
Document Title: An Unordered List Document URL: file://localhost/html/:  This is listRem 1. This is listRem 2. This is listRem 3.	ne Options	Mavigate	Annotate	<u></u>
Document URL: file://localhost/html/: • This is listtem 1. • This is listtem 2. • This listtem 3.	ocument Title:	An Unord	ered List	
Document URL: file://localhost/html/c • This is listhem 1. • This is listhem 2. • This is listhem 3.				- 9
This is listitem 1.     This is listitem 2.     This is listitem 3.	ocument URL:	file://l	ocalhost/html	12 -

**Predstavljanje pobrojanog popisa** *Pobrojani popis* predstavlja popis informacija po brojčanom redu. Svaki put kada se otkriva novi stavak popisa, broj stavka popisa se povećava.

Pobrojani popisi su protumačeni <OL> i </OL> *tagovima*, a isti <LI> *tag* rabljen u nepobrojanim popisima se također rabi u pobrojanim popisima za označavanje svakog elementa popisa

Popis 32.5 pokazuje HTML za jednostavni pobrojan popis. Slika 32.6 pokazuje kako je taj popis prikazan u Mosaicu.

#### Popis 32.5 Pobrojan popis

```
<hrmsl>
<HEAD>
<TITLE>An Ordered List</TITLE>
</HEAD>
<BODY>
<OL>
<LI>This is list item 1.
<LI>This is list item 2.
<LI>This is list item 3.
</OL>
</BODY>
</HTML>
```



**Uporaba popisa definicija** Sjetite se kako izgleda rječnik stručnih pojmova u knjizi: obično je svaka riječ ili pojam zasebno izvučena, a zatim slijedi odlomak gdje je njegova definicija. HTML popisi definicija (ili rječnika) omogućuju da to učinite sa Web stranicama. Popisi definicija sastoje se od pojma – ovo može biti jedna riječ ili niz riječi – popraćenih definicijom. Definicija je obično tekst koji objašnjava pojam.

Premda su popisi definicija djelomično korisni za rječnike, možete ih rabiti za predstavljanje bilo koje vrste informacija gdje trebate naziv i objašnjenje.

Popisi definicija zahtijevaju <DL> i </DL> *tagove* za označavanje početka i završetka popisa. Oni radije koriste dvojne *tagove*: <DT> označava stavku rječnika, a <DD> označava definiciju. Popis 32.6 pokazuje HTML jednostavnog popisa definicija. Slika 32.7 pokazuje taj popis u Mosaicu.

```
Popis 32.6 Jednostavan definicijski popis
```

```
<HTML>
<HEAD>
<TITLE>A Simple Glossary List</TITLE>
</HEAD>
<BODY>
<DL>
<DD>This is the definition field for list item 1.
<DT>Item 2
<DD>This is the definition field for list item 2.
<DD>This is the definition field for list item 3.
</DL>
</BODY>
</HTML>
```

#### **SLIKA 32.7**

Jednostavan popis definicija u Mosaicu



**Kombinirani popisi** Kao što vidite, različiti popisi u HTML-u pružaju vam nekoliko načina predstavljanja informacija korisniku. U stvari, HTML vam dozvoljava kombinirati vrste popisa da svoje informacije možete bolje kontrolirati. Lako možete ugnijezditi jednu vrstu popisa unutar druge.

Pretpostavimo da želite izraditi odsječak vaše polazne stranice kako biste korisnicima pokazali svoje omiljene filmovi i pjesme. Moguće je i ugnijezditi dva rječnika popisa unutar nepobrojanog popisa. Popis 32.7 pokazuje primjer HTML-a koji upotrebljava ugniježdene popise (*nested lists*).

#### Popis 32.7 Izrada prilagođenog popisa ugniježdenjem različitih vrsta popisa

```
<HTML>
<HEAD>
<TITLE>A Custom List</TITLE>
</HEAD>
<BODY>
This list shows some of my favorite musicians and movies.
It uses two definition lists nested in an unordered list.
It also uses some text formatting tags.<P>
I hope that you enjoy it.<HR>
```

```
<LI>Here are some of my favorite movies<P>
 <DL>
 <DT>
 < [] [] >
 <DT>
 <DD>
 </D|>
<P>
<LI>Here are some of my favorite musical groups<P>
 <DL>
 <DT>Coil
 <DD>A European experimental and electronic project.
 <DT>Philip Glass
 <DD>An incredible modern composer.
 <DT>Ozone Quartet
 <DD>Instrumental prograssive quartet.
</D|>
</UL>
</BODY>
</HTML>
```

Ovaj kombinirani popis je malo složeniji od prijašnjih, ali još uvijek koristi samo tehnike koje smo obradili u ovom poglavlju. Obratite pažnju da su rječnički popisi uvučeni u HTML kôd. Uvlaka se koristi samo za lakše čitanje koda (sjetite se da Web pretraživač zanemaruje prekide redaka i dodatne razmake pri prikazivanju stranice). Slika 32.8 pokazuje prilagođene popise u Netscapeu.

Tablica 32.2 pokazuje HTML tagove za izradu ovih popisa.

Tablica 32.2	Popis HTML tagova
Tag	Akcija
<ul></ul>	Stvara nepobrojani (grafički označen) popis; stavke popisa počinju <li> <i>tagom</i></li>
<0L> 0L	Stvara pobrojani (numeriran) popis; stavke popisa počinju <li> tagom</li>
<dl></dl>	Stvara definicijski popis; stavci popisa počinju <dt> <i>tagom</i> za definiranje izraza ili <dd> <i>tagom</i> za definiciju</dd></dt>

## Povezivanje stranica sa sidrom

U ovom odjeljku ćete naučiti kako spojiti više Web stranica zajedno i izraditi hipertekst veze za skakanje s mjesta na mjesto. Hipertekstovi veze u HTML-u se zovu *sidra (anchors)* i za njihovo definiranje koriste se <A> i </A> *tagovi*. Ti *tagovi* se stavljaju oko riječi koju želimo uporabiti za hipertekst veze. Web pretraživači automatski podvlače hipertekst veze i prikazuju ga u drugoj boji.



HREF atribute i URL-ove sa sidrom uporabite kako biste Web pretraživaču rekli koji dokument treba dozvati kad kliknemo na hipertekst veze. Pretpostavimo da želite izraditi hipertekst vezu prema polaznoj stranici NCSA Mosaica. Ako želite uključiti rečenicu

Check here to go to NCSA Mosaic home page, a riječ here je hipertekst veze, trebaju vam slijedeće HTML naredbe:

Click <A HREF=>here</A> to go to the NCSA Mosaic home page.

Sidra okružuju hipertekst veze – u ovom slučaju riječ here. HREF atribut je umetnut u početni *tag* za sidro.

## BILJEŠKA U HREF atribut možete staviti bilo koji URL. Možete se povezati s Web stranicom, FTP lokacijom, Gopher poslužiteljem ili bilo kojom drugom lokacijom.

Osim izrade hipertekst veze, možete davati i imena vezama koristeći atribut NAME. Imenovane veze su vrlo korisne za skakanje na određene lokacije u dokumentu. Na početku duljeg dokumenta možete navesti sadržaj i svaku stavku u tablici sadržaja oblikovati kao hipertekst veze prema nekom mjestu u dokumentu. Kombiniranjem s atributom HREF korisnike možete poslati na određenu lokaciju u drugom dokumentu.

Pretpostavimo da imate dulji dokument – npr. Frequently Asked Questions (FAQ) popis koji raspravlja o DVD-ROM-u. Možete izraditi hipertekst vezu od sadržaja do odlomka "Kako koristiti DVD-ROM". Najprije morate dati ime sidru u odlomku "Kako koristiti DVD-ROM". HTML za to mora izgledati ovako:

```
<A NAME="howtouse">Kako koristiti DVD-ROM</A>
DVD-ROM je odličan uređaj za reprodukciju DVD filmova.
Nažalost, u videotekama još nema dovoljno DVD naslova.
Kako se ne odnose na HTML, ne treba o njima dalje govoriti
u ovom poglavlju.
```

Sada trebate postaviti hipertekst vezu iz sadržaja do tog mjesta. Za povezivanje sa sidrom i davanje imena sidru ispred kojeg ide znak # uporabite atribute HREF. Sadržaj izgleda ovako:

```
<A HREF="#howtouse">Kako koristiti DVD-ROM</A>
```

Kada netko klikne na Kako koristiti DVD-ROM u sadržaju, pretraživač skače na sidro howtouse koje se nalazi malo dalje u tekstu.

Možete imenovati sidro koje je hipertekst veza prema drugoj lokaciji. Uporabite atribute NAME i **BILJEŠKA** HREF u istom sidru. Na primjer,

<A HREF=>Go to the end</A>

će vas odvesti na drugi dokument. Također će prikazati datoteku koja počinje tamo gdje je u datoteci smješteno sidro <#end>. ■

## Uporaba grafika

Karakteristika koja je Web učinila toliko popularnim jest mogućnost uključivanja grafike i teksta u jednostavan format. HTML olakšava umetanje grafike u dokument. Mnogi ljudi pristupaju Webu sporim telefonskim linijama i grafika se pri tim brzinama prijenosa jako dugo prebacuje. Također, neki ljude rabe pretraživače na osnovi teksta, na primjer Lynx, koji uopće ne mogu prikazivati grafiku. Trebali biste pripaziti da se po vašem Web dokumentu svi mogu jednostavno kretati, čak i ako im je grafika isključena ili njihov pretraživač ne podržava grafiku.

U HTML dokumentu grafiku možete rabiti na dva načina: uporabom hipertekst veza ili uporabom slika.

- Hiperveze možete napraviti prema grafičkim datotekama. Ta metoda zahtijeva od korisnika helper program (pomoćni program) koji je sposoban prikazati grafičku datoteku. Vrsta grafike koja se može prikazati ovisi o helper aplikaciji. Veze za ovu metodu su hipertekstovi veze koje grafičku sliku imenuju kao odredišni dokument.
- Grafiku možete umetnuti direktno u vaš HTML dokument. Tako umetnute grafike zovu se <u>inline slike</u>. Mnogi pretraživači podupiru grafiku u JPG, GIF ili X Bitmap formatima kao inline slike.

HTML koristi <IMG> *tag* za ukazivanje na inline sliku. Taj *tag* se kombinira s atributom SRC="*filename*" za određivanje koja slikovna datoteka je prikazana. HTML također osigurava atribut ALIGN= koji govori Web pretraživačima kako trebaju poravnati grafički prikaz s tekstom. Valjane vrijednosti za ALIGN su TOP, MIDDLE i BOTTOM.

Vaše Web stranice morali bi prikazivati i pretraživači koji ne podupiru grafiku. Zato svaki puta kada koristite grafički prikaz trebate osigurati text reference. HTML osigurava način za definiranje prikaza teksta ukoliko pretraživač nije u stanju prikazati grafiku. Atribut ALT="text about the graphic" određuje neke zamjenske tekstove u tom slučaju.

Popis 32.8 je dio HTML-a koji prikazuje inline GIF sliku. Ako pretraživač ne može prikazati sliku, umjesto nje dobije se opis slike.

Popis 32.8 Umetanje grafike u skriptu

```
<HTML>
<HEAD>
   <TITLE>Image example</TITLE>
</HFAD>
<BODY>
<H2>Images</H2>
<P>
<P>
<IMG src="venus.jpg"
  align="right"
  height="160" width="82"
   alt="Statue of Aphrodite">
<|||>
<LI>Statue of Aphrodite, known as the Venus de Milo".
<LI>Carved out of marble in Greece, circa 100 B.C.
<LI>Currently in the Louvre in Paris
<LI>Part of the Louvre's Greek, Etruscan and Roman Antiques Collection
</UL>
<HR>
To go to the Louvre's web site click <A HREF=>HERE</A>
</BODY>
</HTML>
```

Slika 32.9 pokazuje kako je ova stranica prikazana u Netscapeu. Uočite povezivanje različitih elemenata (zaglavlja, popisa, vodoravne crte, grafika i veza).



Grafičke prikaze i sidra možete kombinirati kako bi izradili grafičke hipertekst veze. Oružite <IMG> *tag* sidrom koje određuje koji dokument treba učitati. U slijedećem primjeru slika služi kao hipertekst veza s drugim HTML dokumentom:

```
Click the picture to find out more about this statue.
<A HREF="statue.html"><IMG SRC="statue.gif"
ALIGN=BOTTOM ALT="(Photo of statue)">
</A>
```

## Odavde...

U ovom poglavlju upoznali ste HTML za izradu Web stranice. Naučili ste uporabiti *tagove* koji daju naredbe Web pretraživaču, te uporabiti atribute i URL-ove koji opisuju naredbe. Na Internetu je dostupno puno vodiča za HTML koji obrađuju sve, od sintakse do dizajna. Za pronalaženje tema o HTML-u koje vas zanimaju iskoristite neke od mnogih pretraživačkih servisa.

#### NA WEBU

Za početak, pogledajte Yahoo-ove popise HTML-a:

## http://www.yahoo.com/computers\_and\_Internet/Information\_and\_Documentation /Data\_Formats/HTML/

Više o World Wide Webu i Internetu možete doznati u 31. poglavlju "Kretanje Internetom pomoću World Wide Weba", koje vas upoznaje s Webom i kako ga pretraživati.

# poglavlje33

## Uporaba elektroničke pošte

(Steve Burnett)

## U ovom poglavlju

Elektronička pošta Slanje e-pošte pomoću maila Čitanje pošte Ispis poruka Dobivanje pomoći za rad s e-poštom Spremanje e-pošte u datoteke Brisanje i vraćanje obrisanih poruka Odgovaranje na e-poštu Usmjeravanje pošte drugim korisnicima Prilagođavanje okolice programa mail Napuštanje programa mail Uporaba programa elm Uporaba Mutt klijenta za e-poštu Elektronička pošta ili e-pošta na juriš je osvojila svijet. Milijuni korisnika računala širom svijeta imaju pristup elektroničkoj pošti. Brojne profesionalne mreže ili ISP-i danas omogućuju pristup e-pošti.

## Elektronička pošta

E-pošta je je svaki program koji se rabi za slanje i primanje elektroničkih poruka. Najmanje što vam treba je program s adresom primatelja i poruka koju šaljete. U adresi je uključen prijavno ime osobe koja treba primiti poštu. Ako korisnik rabi drugi sustav u mreži, adresa uključuje i sredstvo za prepoznavanje sustava odredišnog računala. Poruku pripremate dok rabite program za e-poštu ili prije uz pomoć programa za uređivanje teksta, kao što je vi.

#### Vidi "Uporaba vi editora", str. 181

Uporaba elektroničke pošte ima nekoliko prednosti:

- Možete slati izvješća, podatke i dokumente koji na odredište stižu za nekoliko sekundi ili minuta.
- Ne trebate se brinuti hoćete li nekog prekinuti u nečemu ako mu pošaljete poruku, ni da će vas netko prekinuti ako primate poruku – time upravlja računalo.
- Ne trebate se dogovarati za mjesto i vrijeme kada ćete stupiti u vezu s nekim.
- Možete slati i primati poruke kada to vama odgovara.

Kada pošaljete poruku, računalni sustav će je poslati na mrežu i isporučiti na neku drugu lokaciju. Tada kažemo da je pošta poslana. Uskoro poruka dolazi na primateljevo računalo.

Ako su pošiljatelj i primatelj na istom računalnom sustavu, slanje i primanje se zbiva na istom stroju. Sustav e-pošte na odredišnom računalu provjerava postoji li naslovnik, a poruka se dodaje datoteci koja čuva svu e-poštu za tog korisnika (ako Mreža nije potrebna, lokalno računalo provjerava naslovnika). Datoteka u koju se pohranjuje pošta naziva se korisnički *poštanski sandučić sustava (system mailbox)* i ima isto ime kao i korisnik koji prima poštu. Na primjer, ako je vaše prijano ime george, vaš poštanski sandučić sustava je datoteka nazvana george u direktoriju /var/spool/mail. Kada se poruka "isporuči" u poštanski sandučić, kažemo da je pošta primljena.

**BILJEŠKA** Postoji zajednička e-pošta zvana Post Office Protocol (POP) gdje se e-pošta pohranjuje na udaljeni sustav, a potom je vaše računalo učitava. U ovom poglavlju se pretpostavlja da vi izvodite puni sustav za e-poštu na Linux računalu, pomoću programa sendmail koji upravlja pozadinskim poslovima slanja i primanja e-pošte.



Na slici 33.1 je veza između slanja i primanja pošte.

#### Stiže li pošta uvijek na odredište?

Kada pošaljete e-poštu, na zaslonu ćete dobiti poruku Mail Sent! To znači da je pošta poslana – ne znači da je isporučena ili primljena. Obično će vas sustav e-pošte upozoriti ako se poruka ne može isporučiti.

Poruke e-pošte ne stižu uvijek na odredište. Za to postoji nekoliko razloga. Ako pošta ide na Mrežu, adresa mreže može biti točna, ali ime korisnika na toj mreži ne mora biti točno. Ili je adresa točna, ali zbog problema s odobrenjima ili quotas (kvotama?) poruka se ne može smjestiti u korisnikov poštanski sandučić sustava. U oba slučaja, pošta je poslana, ali se ne može isporučiti. Potom, e-pošta može biti isporučena, ali korisnikov poštanski sandučić je uništen ili oštećen. Konačno, korisnik možda ne obraća pažnju na e-poštu ili se ne prijavljuje nekoliko dana, tjedana ili mjeseci.

Računalnu sustav vas upozorava kada imate poštu. E-poštu možete čitati poruku po poruku. S poštom možete napraviti i ovo:

- Obrisati poruke nakon što ste ih pročitali ili čak nepročitane (recimo, kada dobijete bezvrijednu e-poštu).
- Zadržati neke poruke u poštanski sandučić sustava.
- Zadržati neke poruke u osobnom poštanski sandučić.
- Zadržati poruke u pojedinačnim datotekama ili mapama.
- Odgovoriti izravno pošiljatelju poruke.
- Odgovoriti skupini korisnika koji svi dobiju istu poruku.
- Proslijediti poštu drugima.
- Ispisati poštu.

Svoju poštu možete urediti tako da ne zauzima više prostaora na disku nego je to nužno. Sigurno ne trebate spremiti sve poruke koje primite. Vidjet ćete da je lakše čitati poštu koja vam stiže ako redovito brište ili uklanjate poruke iz poštanskog sandučića sustava.

Za Linux postoji nekoliko različitih programa za e-poštu, uključujući i one koji su ugrađeni i Web pretraživače poput Netscapea. Najčešće sučelje za e-poštu, dostupno u svakoj UNIX okolici, je mail. Uz pomoć programa mail možete:

- Pregledati i prikazati e-poštu.
- Uključiti zaglavlje poruke na e-poštu koju šaljete.
- Proslijediti e-poštu drugima.
- Postaviti poštanske popise.

U ovom odjeljku su primjeri iz maila. Nakon toga obrađen je drugi program za e-poštu – elm mailer.

## Slanje e-pošte pomoću maila

E-poštu možete poslati pojedincu, skupini pojedinaca ili poštanskom popisu. Kao i kod obične pošte, morate znati adresu primatelja. Kadšto ćete poruku pisati ili sastavljati dok šaljete e-poštu; drugi puta ćete poslati već pripremljene poruke; možete poslati i ispis naredbi ili program s e-poštom. Kada rabite programe mail ili elm, poruka koju šaljete mora biti datoteka s tekstom – tj. ASCII datoteka.

BILJEŠKA Simple Mail Transport Protocol (SMTP) rabi se za prijenos pošte među računalima. Za sada podupire samo ASCII datoteke. Želite li poslati binarnu datoteku putem e-pošte, trebate je pretvoriti u ASCII datoteku uporabom pomoćnog programa uuencode. ■

Bez obzira kakvu poruku ste pripremili, poštu šaljete naredbom:

mail address

Ovom naredbom pokreće se mail sustav. Potom možete sastaviti poruku i poslati je na neku adresu. U ovoj sintaksi *address* je adresa e-pošte osobe koja prima poruku. Adresa može imati nekoliko različitih oblika. Ako šaljete e-poštu nekome tko ima korisnički ID na računalu koje i vi rabite, uporabite korisnički ID te osobe. Na primjer, želite li poslati e-poštu nekome na vašem sustavu čije je prijavno ime george, unesite naredbu:

mail george

Ako je george na drugom sustavu kojem možete pristupiti putem Mreže, morate uključiti ime pod kojim je taj sustav poznat na Mreži. Pretpostavimo da je george ime korisnika na računalnom sustavu čija se Mreža zove apples.startup.com. E-poštu možete poslati ovako:

mail george@apples.startup.com

Točan oblik adrese ovisi o vrsti mreže i lokalnim pravilima ili dogovorima. O obliku adrese možete doznati od svog mrežnog administratora.

Želite li istu poruku poslati nekolicini korisnika, njihove adrese stavite u redak s naredbom mail:

```
mail fred bill george@apples.startup.com
```

#### Pisanje poruke za vrijeme slanja e-pošte

Puno korisnika sastavlja ili piše poruku dok je u programu za e-poštu. To je obično najbrži – ali ne i najjednostavniji – način slanja pošte. Taj način nije najjednostavniji zato što imate ograničene mogućnosti uređivanja teksta poruke. Općenito, možete se istovremeno baviti samo jednim retkom. Najprije, upišete naredbu za slanje e-pošte, odredite adresu, a potom pritisnete <Return>. Potom upišete poruku, te na kraju stavite točku u poseban redak. Za završetak poruke možete uporabiti i <Ctrl-d>. Evo primjera kako poslati e-poštu korisniku koji se zove lynn, pri čemu pokrećete sustav e-pošte i određujete lynnovu adresu na svojem sustavu:

```
mail lynn
Subject: Congratulations! Lunch Thursday?
```

Upišite poruku, pri čemu za kraj retka pritisnete na <Return>. Evo primjera poruke koju možete poslati lynn (na kraju retka možete pritisnuti <Return> kako biste razmaknuli odlomke poruke):

Lynn,

```
Just wanted to tell you that I thought you did a great
job at the meeting yesterday! It seems as if we're
finally turning this problem around.
Want to get together for lunch Thursday?
Give me a call.
joe
```

Umjesto točke, poruku možete završiti i sa <Ctrl-d>. Računalo odgovara sa EOT, što znači *kraj prijenosa (end of transmission)*.

## Otkazivanje poruke

Poruku možete otkazati za vrijeme pisanja, ali ne i nakon slanja. Želite li otkazati poruku dok je pišete, pritisnite tipku na vašem sustavu koja je za to određena (obično <Ctrl-c> ili <Del>). Kada je poruka otkazana, spremljena je u datoteku dead.letter. Tu datoteku možete obrisati ili kasnije uređivati. Tipke <Ctrl-c> morate pritisnuti dva puta pri

otkazivanju poruke (kao zaštitu ako ste <Ctrl-c> ili <Del> pritisnuli slučajno). Nakon otkazivanja poruke dobit ćete naredbeni redak. U sljedećem primjeru pokazano je otkazivanje poruke:

```
Mail lynn
Subject: Congratulations! Lunch Thursday?
Lynn,
Just wanted to tell that I thought you did a great
job<Ctrl-c>
  (Interrupt - - one more to kill letter)
```

Sada morate odlučiti hoćete li dalje pisati pismo ili ga uništiti. Ako odlučite nastaviti, nastavite pisati tekst:

```
at the meeting yesterday! It seems as if we're finally turning this problem around.
```

Ponovno odlučite otkazati pismo, pa pritisnete <Ctrl-c> ili <Del>. Sustav odgovara s (Interrupt -- one more to kill letter). S obzirom da želite uništiti poruku, pritisnite još jednom <Ctrl-c> ili <Del>; program mail završava i dobijete *prompt* ljuske (shell *prompt*).

## Slanje pripremljene poruke

Za pisanje poruke možete uporabiti program za uređivanje teksta, npr. vi. Tako možete rabiti alate za uređivanje teksta, provjeru pravopisa itd. Nije važno koji program rabite za izradu teksta, samo morate spremiti kao ASCII datoteku ili tekstualnu datoteku.

Recimo da se datoteka koju želite poslati zove report.txt, a primateljeva adresa je top@kite.fish.com. Postoje tri načina za slanje datoteke. U slijedećim primjerima naredbi, program mail rabi opciju -s, a niz koji služi kao subject heading (naziv predmeta?) ima navodnike:

use a pipe. Želite li report.txt poslati pomoću naredbe mail, unesite slijedeće:

```
cat report.txt | mail -s" Sales Report" top@kite.fish.com
```

Preusmjerite ulaz. Želite li report.txt poslati pomoću naredbe mail, s opcijom -s, unesite slijedeće:

```
mail -s" Sales Report" top@kite.fish.com < report.txt</pre>
```

Uporabite čr za uključivanje datoteke u poruku. Želite li uporabiti mail za slanje datoteke (uporabom podrazumijevanog prompta Subject), unesite ove naredbe:

```
mail top@kite.fish.com
Subject: Sales Report
~r report.txt
~.
EOT
```

Kada poruku pošaljete pomoću jedne od te tri metode, javlja se prompt sustava.

**BILJEŠKA** U trećem primjeru čr se rabi za read ili uključivanje datoteke report.txt u poruku e-pošte. To je primjer *tilda naredbe*. Ispred naredbe stavlja se znak tilda (~) dok čitate ili šaljete poštu. Postoji nekoliko takvih naredbi o kojima će biti riječi u ovom poglavlju.

## Slanje rezultata naredbe ili programa e-poštom

Ako izvodite naredbu ili program koji daje rezultate na zaslonu (poznat kao stdout), taj izlaz možete proslijediti u mail naredbu. Pretpostavimo da imate neke informacije u datoteci contrib.1st. Uporabite naredbu sort za razvrstavanje datoteke, a potom si pošaljite rezultate (prijavno ime bkorn) i top. Za to unesite ove naredbe:

Sort contrib.1st | mail -s "Sorted Contrib Info" bkorn top@kite.fish.com

## Čitanje pošte

Većina Linux sustava vas pri prijavi obavješćuje da imate e-poštu. Za čitanje pošte možete uporabiti mail ili neki drugi program za e-poštu. Kad poštu pročitate, program svaku pročitanu poruku označava. Ovisno o naredbama koje rabite i kako napuštate program za e-poštu, pročitane poruke čuvaju se ili u vašem poštanskom sandučiću sustava /var/spool/mail\$LOGNAME ili u prijavni direktorij u datoteci mbox.

## Uporaba maila za čitanje pošte

Želite li čitati poštu iz maila, unesite **mail**. Ako je vaše prijavno ime bkorn, zaslon će vam izgledati ovako (ono što vi upisujete je masno):

```
mail
mail Type ? for help.
"/var/spool/mail/bkorn": 5 messages 2 new 1 unread
        1 sarah Wed Jan 8 09:17 15/363
        2 top@kite.fish.com Thu Jan 9 10:18 26/657 Meeting on Friday
U 3 fred_Fri Jan 10 08:09 32/900 New Orders
> N 4 jones Fri Jan 10 13:22 35/1347 Draft Report
N 5 smith@somewhere.com Sat Jan 11 13:21 76/3103 Excerpt from book
?
```

Uočite slijedeće stavke:

- U prvom retku prepoznaje se program i upućuje na tipkanje upitnika ukoliko se želi dobiti pomoć.
- U drugom retku vidi se da mail čita vaš poštanski sandučić sustava /var/spool/mail/bkorn i da imate pet poruka. Dvije su stigle nakon vaše zadnje provjere pošte, jedna je stigla prije, ali je niste pročitali, a dvije su već pročitane.

U slijedećih pet redaka imate informacije o pošti. Za sada nemojte obraćati pažnju na prvih nekoliko znakova. Svaki redak ima broj poruke, adresu pošiljatelja, datum slanja poruke, broj redaka i slova u poruci te predmet (ako postoji). Pogledajte slijedeći redak:

2 top@kite.fish.com Thu Jan 9 10:18 26/657 Meeting on Friday

Iz ovog retka možemo doznati da je poruka pod rednim brojem 2 stigla od top@kite.fish.com - adresa upozorava da je poruka na vaše računalo došla s računala na drugoj Mreži (pošta od lokalnog korisnika označena je korisnikovim prijavnim ID). Poruka je poslana u četvrtak, 9. siječnja, u 10:18, sastoji se od 26 redaka i 657 znakova. Predmet je *Meeting on Friday*.

- Redak s porukom koji počinje s N upozorava da je to nova pošta pošta koju ste primili nakon posljednje provjere e-pošte. Redak koji počinje s U upozorava na nepročitanu poštu. Ako nema tih oznaka, znači da ste pročitali i spremili poruku u poštanski sandučić sustava.
- Znak veće od (>) na početku retka označava tekuću poruku poruku you'll act on next.
- Upitnik (?) u posljednjem retku je naredbeni redak u programu mail.

**Čitanje tekuće poruke** Tekuća poruka je ona koja je označena znakom veće od (>). Želite li čitati tu poruku, pritisnite tipku <Return>. Kada je otvorite, imate otprilike ovakav tekst:

```
Message 4:
From jones Fri, Jan 10 13:22 EST 1997
Received: by your.system.com
Date: Fri, Jan 10 13:22:01 -0500
From: Carol Jones <jones>
Return-Path: <jones>
To: aborat, lynn, oackerm, bkorn
Subject: Draft Report
Here is a draft of the report I intend to submit next week.
Please take a look at it and let me know your comments.
Thanks.
- - - - - - - - - Report Starts Here- - - -
Opportunities for Expansion
Prepared by Carol Jones
Over past 6 months, we've seen an indication of an increase in the
demand for our services. Current market trends indicate that the demand
will continue for at least 18 months and possibly longer. The manager of
our service staff states "We're up to our necks in new customers and
:
```

Poruka se prikazuje zaslon po zaslon. Svaki puta kada vidite dvotočku možete pritisnuti <Return> pa ćete vidjeti slijedeći zaslon. Želite li prekinuti čitanje poruke možete pritisnuti <q>. Sada pritisnite <Return> pa ćete vidjeti ostatak poruke.

Kada vidite posljednji zaslon, pokaže se EOF: (za *end of file, završetak datoteke*). Pritisnite <q> ili <Return> pa ćete se vratiti na ? *prompt*. Uočite da znak veće od i dalje pokazuje na poruku koju ste upravo pročitali. Poruka koja je tekuća, to i ostaje.

Prije početka poruke prikazano je još nekoliko redaka. To su informacije o zaglavlju – i mogu biti korisne. Informacije o zaglavlju imaju ove podatke:

- Broj poruke
- Tko je poslao poruku
- Kada je poruka poslana
- Ime sustava koji je primio poruku
- Datum primitka poruke
- Pravo ime korisnika i njegov korisnički ID
- Povratnu stazu
- Primatelja ili primatelje poruke
- Predmet

Sve te informacije su sastavni dio poruke e-pošte. Pošiljatelja se uvijek prepoznaje i tako se onemogućava krivotvorenje. Pravo ime koje se pojavljuje u retku Form preuzima se iz polja korisnikovog unosa u datoteku s lozinkom. Ako odgovarate na poruku, sustav rabi informacije Return-Path ili Reply-To. U retku To je adresa ili popis adresa primatelja poruke. (Naš primjer prikazuje skupnu poruku.) Pošiljatelj je popunio redak Subject.

**Čitanje sljedeće poruke** Postoje dva načina čitanja sljedeće poruke (poruke koja je iza tekuće poruke). Za prikaz sljedeće poruke možete pritisnuti <Return> ili <n>. Nakon što ste je pročitali, ta poruka postaje tekuća. Sljedeću poruku čitate na isti način kao i tekuću. Kada pročitate posljednju poruku s popisa, dobijete poruku 'At EOF'.

**Čitanje bilo koje poruke** Sve poruke u vašem poštanski sandučić su numerirane. Ako unesete broj poruke kada dobijete ? *prompt*, poruke možete čitati bilo kojim redoslijedom. Na primjer, želite li pročitati poruku broj 2, upišite **2** i pritisnite <Return>. Sada poruka broj 2 postaje tekuća poruka.

## Čitanje e-pošte iz drugih datoteka

Kada pokrenete mail, čitate poruke u svom poštanskom sandučiću sustava koji ima stazu /var/spool/mail/\$LOGNAME. U 18. poglavlju ste naučili da je LOGNAME varijabla ljuske koja obuhvaća vaše prijavno ime. Ako je vaša prijava bkorn, vaša pošta se čuva u /var/spool/mail/bkorn. Poštu možete čitati iz drugih datoteka koje imaju potpune poruke e-pošte – tj. poruke sa zaglavljima i tekst poruka. Naravno, za te datoteke morate imati pravo čitanja.

Želite li pročitati poruke iz te datoteke, upišite naredbu za pokretanje programa za e-poštu i nakon toga **-f** *filename*, te pritisnite <Return>. Na primjer, za čitanje e-pošte u datoteci mbox, unesite ovu naredbu:

mail -f mbox!

Poštu u datoteci možete pročitati na isti način kako čitate e-poštu iz poštanskog sandučića sustava.

BILJEŠKA Datoteka mbox smještena je u polaznom direktoriju i automatski obuhavća poruke koje ste pročitali, ali niste obrisali. Te poruke spremaju se u mbox kada izađete iz maila.

## Slanje pošte za vrijeme čitanja

E-poštu možete poslati dok čitate poruke u programu mail. To ćete napraviti tako da unesete **m** *address* u ? *prompt*. Slijedite ove korake:

- 1. Pokrenite program mail (upišite mail i pritisnite <Return>).
- Dok radite u njemu, u ? prompt unesite slijedeće naredbe kako biste poslali e-poštu korisniku čije je prijavno ime ernie:

m ernie

3. U promptu za predmet upišite subject heading:

Subject: Game Time

4. Upišite poruku i završite je točkom u posljednjem retku, kao u ovom primjeru:

```
Don't forget we're playing V-ball at 6:30
```

Računalo će odgovoriti ovako:

EOT ?

5. Nastavite rabiti mail.

## Ispis poruka

Pomoću programa mail možete ispisati tekuću poruku na pisač povezan s vašim sustavom. Poruka koju želite ispisati mora biti tekuća. U ? *prompt* unesite **| 1pr**. To znači da prosljeđujete tekuću poruku u program 1pr.

Želite li ispisati više poruka, spremite ih u datoteku, a potom ispišite datoteku. O spremanju poruka bit će riječi u odjeljku "Spremanje e-pošte u datoteke".

## Dobivanje pomoći za rad s e-poštom

Kada upišete naredbu za pokretanje programa za e-poštu, dobit ćete ? *prompt*. U programu mail upisat ćete ? želite li dobiti pomoć. Želite li dobiti popis naredbi i neke informacije o svakoj naredbi, upišite ? i pritisnite <Return>.

```
Kada ste upisali ? i pritisnuli <Return>, dobit ćete otprilike ovakav prikaz:
```

```
Mail
           Commands
t <message list> type messages
      goto and type next message
n
n goto and type next message

e <message list> edit messages

f <message list> give head lines of messages

d <message list> delete messages

s <message list> file append messages to file

u <message list> undelete messages

R <message list> reply to message senders

r <message list> reply to message senders and all recipients

pre <message list> print message

g <message list> make message

m <message list> mil to specific usons
m <user list> mail to specific users
                        quit, saving unresolved messages in mbox
q
                        quit, do not remove system mailbox
Х
h
                       print out active message headers
                        shell escape
1
cd [directory] chdir to directory or home if none given
A <message list> consists of integers, ranges of same, or user names
separated by spaces. If omitted, Mail uses the last message typed.
A <user list> consists of user names or aliases separated by spaces.
Aliases are defined in .mailrc in your home directory.
ጲ
```

U ovom ispisu su naredbe koje možete rabiti iz ? *prompta*. Iako su neke od naredbi objašnjene u ovom poglavlju, uočite slijedeće:

- U svim slučajevima možete rabiti prvo slovo naredbe ili upisati čitavu naredbu.
- Stavke u [] i ⇔ su altrenativne; zagrade ne morate pisati kao dio naredbe.
- Ako uporabite \*, message list odnosit će se na sve poruke. Želite li sve poruke spremiti u datoteku allmail, upišite s \* allmail i pritisnite <Return>.
- Izraz message list može se odnositi na jednu poruku. Želite li spremiti poruku broj 2 u datoteku meeting, upišite s 2 meeting i pritisnite <Return>.
- Izraz message list može se odnositi na skup poruka, s time da brojeve poruka povezujete crticom. Na primjer, 2-4 odnosi se na poruke 2, 3 i 4. Želite li spremiti te poruke u datoteku memos, upišite s 2-4 memos i pritisnite <Return>.
- Izraz print u retku print message ne znači da treba poruku ispisati na pisač. On znači da se poruka prikaže na zaslonu.
- Naredba edit korisna je za mijenjanje poruke prije slanja nekom drugom ili spremanja u datoteku.

## Spremanje e-pošte u datoteke

Neke od primljenih poruka htjet ćete zadržati. Nije praktično svu poštu čuvati u poštanskom sandučiću sustava. Razlozi su slijedeći:

- Morat ćete se probijati kroz previše poruka kada ćete htjeti pročitati poštu.
- Administratori sustava često ograničavaju veličinu vašeg poštanskog sandučića sustava. Ograničenje veličine ovisi o postavkama administratora sustava. Ako dođete do te granice, nećete moći više primati poštu.
- Vaša pošta neće biti organizirana i bit će vam teško pronaći važne poruke ili poruke koje se odnose na određenu temu.

Već ste naučili da se poruke koje ste pročitali spremaju (osim ako ne odredite drugačije) u datoteku mbox. Znate i da te poruke možete pročitati ako upišete **mail -f mbox** i pritisnete <Return>. Poruke možete čitati i iz drugih datoteka.

Postoje dva načina (sa i bez zaglavlja) spremanja tekuće poruke u datoteku kada rabite mail. Možete odrediti datoteku gdje će biti poruka i poruka se dodaje u tu datoteku. Ako ne odredite datoteku, poruka se dodaje u datoteku mbox u vašem polaznom direktoriju. Ako rabite q za napuštanje programa mail, poruke se uklanjaju iz poštanskog sandučića sustava.

Kada vidite ? prompt, možete uporabiti slijedeće metode za spremanje poruka:

- Upišite s kako biste dodali tekuću poruku u mbox u vašem polaznom direktoriju.
- Upišite s *filename* kako biste dodali tekst tekuće poruke u datoteku uz nedirnuta zaglavlja (korisno ako želite uporabiti svoj program za e-poštu za kasnije čitanje poruka).
- Upišite w *filename* kako biste dodali tekst tekuće poruke u datoteku bez informacija iz zaglavlja (korisno kada želite uporabiti samo tekst poruka u datoteci koja se može obraditi u nekom drugom programu).

Želite li nakon čitanja poruka iste čuvati u poštanskom sandučiću sustava, a ne u mbox, uporabite naredbu za čuvanje, pre. Tu naredbu možete rabiti s popisom poruka.



Znate da se poruke koje ste već pročitali automatski spremaju u mbox, osim ako ne uporabite naredbu za čuvanje (*preserve*).

Dobro je pri uporabi naredbe za spremanje, s, odrediti ime datoteke. Ako ga ne odredite, tekuća se poruka dodaje u datoteku mbox. Ako uključite popis poruka, ali ne odredite datoteku, mail rabi popis poruka kao ime datoteke u koju sprema tekuću poruku.

Ako uporabite q za izlazak i programa za e-poštu, spremljene poruke uklanjaju se iz poštanskog sandučića sustava.

## Brisanje i vraćanje obrisanih poruka

Za brisanje poruka iz datoteke s porukama koju čitate, uporabite naredbu d. Ako napustite program mail pomoću naredbe q, sve poruke koje ste obrisali pomoću naredbe d uklanjaju se iz datoteke. Naredbe d ili delete uporabite za označavanje poruka koje treba obrisati dok ste u programu mail. Ako program napustite naredbom q, označene poruke uklanjaju se iz poštanskog sandučića.

Želite li obrisati tekuću poruku, upišite **d** i pritisnite <Return>. Možete odrediti i popis poruka.

Ako označite poruku ili skupinu poruka koje treba obrisati, možete se predomisliti i vratiti obrisanu poruku ili poruke uporabom naredbe u. Naredbu u morate uporabiti prije naredbe q za napuštanje programa; kada unesete q, poruke se brišu. Naredbu u ili undelete rabite na isti način kao i d ili delete.

SAVJET Za vraćanje svih obrisanih poruka, na ? prompt upišite u \*.

## Odgovaranje na e-poštu

Za odgovaranje na e-poštu možete uporabiti adresu u polju zaglavlja Reply-To. Ako to polje ne postoji, uporabite informacije u polju zaglavlja Return-Path. Slijede nepotpuna zaglavlja dvije poruke; jedna ima oba polja zaglavlja, a druga samo Return-Path. Ta polja su u oba zaglavlja otisnuta masno.

Prva poruka:

From server@malte.abc.com Mon Nov 8 18:31 EST 1993 Received: from MALTE.ABC.COM by s850.mwc.edu with SMTP Return-Path: <server@matle.ams.com> Date: Mon, 8 Nov 93 18:17:15 -0500 Comment: From the DuJour List Originator: dujour@mathe.abc.com Errors-To: asap@can.org Reply-To: <dujour@mathe.abc.com> Sender: dujour@mathe.abc.com

#### Druga poruka:

```
From jones Fri, Jan 7 13:22 EST 1994
Received: by your.system.com
Date: Fri, 7 Jan 1994 13:22:01 -0500
From: Carol Jones <jones>
Return-Path: <jones>
To: aborat, lynn, oackerm, bkorn
Subject: Draft Report
```

Żelite li odgovoriti na prvu poruku, uporabite Reply-To adresu dujour@mathe.abc.com. Uočite da se polja Reply-To i Return-Path razlikuju. U drugom primjeru uporabite adresu jones za slanje odgovora.

BILJEŠKA Uvijek rabite adresu Reply-To ako je uključena u zaglavlje jer predstavlja adresu pošiljatelja. Kada ta adresa nije dostupna, u polju Return-Path obično ćete naći odgovarajuću adresu pošiljatelja.

Programu mail može omogućiti određivanje adrese na koju šalje odgovor na poruku. To možete napraviti uz pomoć ovih naredbi:

- R Adresira odgovor na pošiljatelja poruke
- r Adresira odgovor na pošiljatelja i sve primatelje poruke e-pošte

Uz pomoć bilo koje od tih naredbi možete odrediti popis poruka. Inače se naredbe R ili r primjenjuju na tekuću poruku.

Dio zaglavlja pokazuje kako uporabiti te dvije naredbe. Ovo zaglavlje izvučeno je iz poruke Carol Jones u kojoj ona pita skupinu primatelja što misle o skici izvješća koji je pripremila:

```
From jones Fri, Jan 7 13:22 EST 1994
Received: by your.system.com
Date: Fri, 7 Jan 1994 13:22:01 -0500
From: Carol Jones <jones>
Return-Path: <jones>
To: aborat, lynn, oackerm, bkorn
Subject: Draft Report
```

Želite li odgovoriti samo na jones, unesite R u ? prompt. Dobit ćete ovaj odgovor:

To: jones Subject: Re: Draft Report

Redak To govori vam da odgovor ide jednoj osobi. Zaglavlje Subject pokazuje da je poruka odgovor na neku poslanu poruku s istim predmetom.

Želite li da vaše komentare vide svi s popisa, unesite **r** u ? *prompt*. Dobit ćete ovaj odgovor:

To: jones, aborat, lynn, oackerm, bkorn Subject: Re: Draft Report

Redak To govori vam da odgovor ide svima s prvobitnog popisa, kao i autoru. Zaglavlje Subject pokazuje da je poruka odgovor na neku poslanu poruku s istim predmetom.

Sada možete unijeti svoju poruku onako kako je to već objašnjeno u odjeljku "Slanje e-pošte pomoću maila".

#### POZOR

Pripazite kako rabite naredbu r za odgovor. Sve što pošaljete šalje se svima koji dobiju kopiju originalne poruke. Linux sustav osjetljiv je na malo i veliko slovo, pa se može dogodit da nepažnjom pošaljete nekome ono što ne želite.

## **BILJEŠKA** Prije slanja odgovora razmislite o tome što pišete i tko će čitati poruku. Sarkazm ili satira nije preporučljiva u porukama e-pošte. Uporaba e-pošte nije isto što i razgovor, ne vidite i ne čujete reakcije, a ni ta osoba ne vidi i ne čuje vaše reakcije. U porukama e-pošte lakše je i djelotvornije biti uljudan i izravan.

Lako je proslijediti poruku; čim nešto pošaljete nekome ne znate gdje će to završiti i koliko će ljudi vidjeti vašu poruku. Razmislite o tome i uzmite to u obzir.

**Vidi** "Nedostatak vizualne informacije", **str 662** 

## Usmjeravanje pošte drugim korisnicima

E-pošta se raspodjeljuje prema adresama. Zadaće poput prosljeđivanja poruke, slanja kopija (cc:) poruke, izrada jednostavnijih oblika adresa i izrada poštanskih popisa uključuju upravljanje adresama. To ne morate raditi izravno – program mail ima ugrađene te sposobnosti.

#### Prosljeđivanje poruka

Poruku možete proslijediti (u stvari uključujete poruku u onu koju ste vi sastavili) tako da pokrenete program mail kao kada čitate poruku. Potom uporabite naredbe m, r ili R za slanje poruke. Dok pišete poruku, uporabite tilda naredbu, čf, za prosljeđivanje jedne ili više poruka. Opći oblik naredbe čf je čf *msglist*. Evo primjera kako proslijediti poruku:

1. Pokrenite mail (utipkajte mail i pritsnite <Return>. Sustav odgovara otprilike ovako:

```
mail Type ? for help.
"/var/spool/mail/bkorn": 5 messages 2 new 1 unread
1 sarah Wed Jan 8 09:17 15/363
2 top@kite.fish.com Thu Jan 9 10:18 26/657 Meeting on Friday
U 3 fred_Fri Jan 10 08:09 32/900 New Orders
> N 4 jones Fri Jan 10 13:22 35/1347 Draft Report
N 5 smith@somewhere.com Sat Jan 11 13:21 76/3103 Excerpt from
book
?
```

- Pročitajte poruku 5 tako da otipkate 5 i pritisnete <Return>. (Tekst te poruke nije ovdje prikazan.) pretpostavimo da ga želite poslati svojim prijateljima čije su adrese sarah, anglee@hb.com i lynn@netcong.com.
- 3. Uporabite naredbu m za slanje pošte na adrese iz 2. koraka, upišite predmet i napišite početak svoje poruke, kao u primjeru:

```
? m sarah anglee@hb.com lynn@netcong.com
Subject: Forwarding an excerpt from new Que Linux book
Hi!
I'm forwarding an excerpt I came across from a new book by Que.
It's Special Edition Using Linux, Fourth Edition. I'll be getting my
own copy tomorrow.
Do you want me to pick up a copy for you, too?
```

 Uporabite naredbu - f za prosljeđivanje poruke broj 5 (upišite ~f 5 i pritisnite <Return>). Program mail odgovara ovako:

```
Interpolating: 5 (continue)
```

 Pokazivač je sada ispod riječi "continue". Možete nastaviti unositi tekst u poruku ili je završiti tako da otipkate č. i pritisnete <Return>. Ako završite poruku, javit će se ? prompt.

## Slanje kopije pomoću programa mail

Kopiju poruke možete poslati na jednu ili više adresa. Željene adrese unesite u *cc: popis*. Naredba radi ovako: pošta se šalje na osnovnu adresu ili adrese (one u zaglavlju To) kao i na adrese u Cc zaglavlju. Želite li uključiti adrese na cc: popis, uporabite tilda naredbu  $\sim$ c *address*.

U sljedećem je primjeru slanje kratke obavijesti na osnovnu adresu (wjones) i kopije te obavijesti sebi i na još jednu adresu (vaša je adresa bkorn, a druga ecar1st). Sebi ste poslali jednu poruku jer želite imati kopiju obavijesti. Uz pomoć ovih koraka možete poslati poruku primateljima s cc: popisa:

 Pokrenite program mail za slanje poruke na osnovnu adresu, wjones i dajte predmet poruke. To ćete učiniti ovako:

\$ mail fred Subject: Memo - Sales Agreement with Framistan

2. Unesite tekst koji želite poslati. Na primjer:

```
TO: Fred Jones
Date: Oct 31, 1996
From: Henry Charleston
RE: Sales Agreement With Framistan Motors
On October 27, 1996, I held a meeting with the CEO of Framistan
Motors. We concluded and initialed a sales agreement by which Framistan
would purchase 10,000 units of our thermo-embryonic carthurators.
The agreement has been forwarded to the appropriate parties in our
organization and we intend to formally complete the agreement within
two weeks.
```

 Dajte naredbu ~c za dodavanje adresa u cc: popis. Na primjer, da bi sebi (bkorn) i ecar1st poslali kopiju, upišite ovo:

~c ecar1st bkorn

4. Poruku ćete poslati tako da unesete tildu i točku (č.) te pritisnete <Return>. Pojavljuje se poruka EOT i *prompt* ljuske.

Želite li pregledati i možda promijeniti zaglavlja na porukama, unesite ~h dok pišete poruku. Dobit ćete jedno po jedno zaglavlje koje možete mijenjati.

Kada je poruka tako poslana, svi primatelji mogu vidjeti zaglavlja To i Cc. Svatko tko na poruku odgovori r naredbom, poslat će odgovore i na sve adrese u popisima To i Cc, kao i autoru.

Program mail možete i prilagoditi tako da od vas uvijek traži Cc zaglavlje na isti način kako od vas traži zaglavlje Subject (o tome se govori u odjeljku "Prilagođavanje okolice programa mail"). Naravno, ako pritisnete <Return> Cc popis će ostati nepromijenjen.

## Uporaba zamjenskih imena i poštanskih popisa

Program mail, kao i većina programa za e-poštu vam omogućuje izradu zamjenskog imena (*alias*) za adresu i zamjenskog imena grupe za popis adresa. Prema zamjenskom imenu grupe ponašate se kao prema poštanskom popisu. Uporaba zamjensko imena za pojedinačne adrese je lakša od uporabe običnih adresa jer je zamjensko ime obično kraće i lakše ga je zapamtiti.

Želite li postaviti pojedinačno ili zamjensko ime grupe, uporabite naredbu alias na ? *prompt* dok čitate poštu. Želite li zamjenska imena još bolje iskoristiti, stavite ih u datoteku .mailre u svom polaznom direktoriju (kao što je opisano u slijedećem odjeljku).

Evo primjera postavljanja i uporabe zamjenskih imena pomoću programa mail:

1. Pokrenite program mail tako da na ? *prompt* unesete **mail**. Nakon ispisa zaglavlja, dobit ćete ? *prompt*:

```
mail Type ? for help.
"/var/spool/mail/bkorn": 5 messages 2 new 1 unread
   1 sarah Wed Jan 5 09:17 15/363
   2 croster@kite.fish.com Thu Jan 6 10:18 26/657 Meeting on
Friday
U 3 wjones Fri Jan 7 08:09 32/900 Framistan Order
> N 4 chendric Fri Jan 7 13:22 35/1347 Draft Report
N 5 kackerma@ps.com Sat Jan 8 13:21 76/3103 Excerpt from GREAT
new Linux
?
```

2. Za postavljanje pojedinačnog zamjenskog imena, uporabite naredbu alias, a potom zamjensko ime za adresu. Zamjensko ime ros za adresu croster@kite.fish.com napravit ćete ovako:

Alias ros croster@kite.fish.com

 Uporabite zamjensko ime ros za adresu; program mail će to pretvoriti u potpunu adresu. Na primjer, želite li poslati poruku na adresu croster@kite.fish.com, unesete naredbu m ros.

Želite li postaviti zamjensko ime grupe, uporabite naredbu alias i nakon toga zamjensko ime za adrese. U slijedećem primjeru izrađuje se zamjensko ime friends, a potom se grupi šalje poruka:

```
alias friends chedric karlack abc.com!homebase!fran eca@xy.srt.edu
m friends
Subject: Excerpts from new Linux book - get a copy!
~f 5
Interpolating: 5
~.
EOT
?
```

## Prilagođavanje okolice programa mail

Okolicu programa mail možete prilagoditi stavljanjem naredbi ili varijabli okolice u datoteku .mailrc polaznog direktorija. Program mail provjerava tu datoteku kada god ga rabite. Možete postaviti prilično varijabli okolice i naredbi, a različiti programi za poštu će rabiti druge naredbe. Na man page provjerite popis svih opcija iz .mailrc. Neke od naredbi koje prepoznaje program mail su već dane u odjeljku "Dobivanje pomoći u radu s e-poštom"; u ovom odjeljku opisujemo podskup naredbi i varijabli koje se mogu rabiti u datoteci .mailrc. U tablici 33.1 je popis naredbi; u tablici 33.2 je popis varijabli okolice.

Tablica 33.1	Naredbe u programu mail
Naredba	Definicija
#	Označava komentar. Nema akcije.
alias	Postavlja pojedinačno zamjensko ime ili zamjensko ime grupe. Rabi se kao alias <i>alias-name address-list</i> .
set	Postavlja varijable okolice. Rabi se kao set <i>variable-name</i> ili kao set <i>variable-name=string</i> .

Bilo koju naredbu iz tablice 33.1 možete izdati iz ? *prompta* kad god rabite program mail; **SAVJET** one će biti aktivne samo taj puta.

Tablica 33.2	Varijable okolice u programu mail
Varijabla	Definicija
askcc	Nakon unošenja poruke od vas očekuje cc: popis. Podrazumijevano je noaskce.

Varijabla	Definicija
asksub	Prije unošenja poruke od vas očekuje Subject popis. Podrazumijevano je omogućena.
noheader	Ne ispisuje informacije iz zaglavlja na porukama kada pokrenete program mail. Podrazumijevano je isključena.
ignore	Ne obraća pažnju na znakove prekida kada unosite poruke. Korisna je ako su veze preopterećene. Podrazumijevano je noignore.
metoo	Kada imate svoje ime u zamjenskom imenu grupe, poruka vam se ne šalje. Postavljanje ove varijable omogućuje vam primanje poruka poslanih zamjenskom imenu grupe koja sadržava vašu adresu. Podrazumijevano je nometoo.

SAVJET Ako naredbe ili postavljene varijable spremite u datoteku /etc/mail.rc, možete postaviti sustavsku okolicu.

U slijedećem primjeru postavljena je datoteke .mailrc tako da možete uporabiti naredbe i varijable okolice popisane u tablicama 33.1 i 33.2. Znak povisilice (#) rabi se za opisivanje. Ovu datoteku možete napraviti pomoću vi editora ili nekog sličnog koji može napraviti datoteke s tekstom ili ASCII datoteke.

```
# .mailrc file for D. Wayne Love
# make sure interrupts are NOT ignored
set noignore
# set variables so that prompts for Subject and Cc always appear
set asksub
set askcc
# individual aliases
alias billy wcuth
alias ben benjamin@flagstaff.abaced.com
alias me dwlove
# group aliases, mailing list
alias mercs miles@dendarii.net quinn taura
alias research jones brown smith
alias googol djames bkorn cam@googol.org bkorn
```

Ovo smjestite u datoteku .mailrc. Kada kod pokrenete program mail, te naredbe se izvršavaju.

## Napuštanje programa mail

Dok čitate e-poštu u poštanskom sandučiću, možete čitati, preskakati ili brisati poruke. Te se akcije ne zbivaju u samom poštanskom sandučiću, nego u njegovoj privremenoj kopiji. Program za e-poštu možete napustiti tako da promijenite sadržaj svog poštanskog sandučića (promijenjena privremena kopija zamjenjuje sadržaj originalnog poštanskog sandučića),
ili možete izaći tako da vam poštanski sandučić ostane nepromijenjen bez obzira na to što ste radili.

#### Napuštanje i spremanje promjena

Želite li napustiti program mail te spremiti promjene, na ? *promptu* pritisnite <q><Return>. Ponovno ćete dobiti *prompt* ljuske. Kada tako izađete, poruke koje ste pročitali, ali niste obrisali spremljene su u datoteku mbox u vašem polaznom direktoriju.

Pretpostavimo da za čitanje pošte rabite program mail. Vaše je prijavno ime bkorn, a polazni direktorij /home/bkorn. Kada unesete nardbu **mail** za pokretanje programa, dobit ćete podatke:

```
mail Type ? for help.
"/var/spool/mail/bkorn": 5 messages 2 new 1 unread
1 sarah Wed Jan 8 09:17 15/363
2 top@kite.fish.com Thu Jan 9 10:18 26/657 Meeting on Friday
U 3 fred Fri Jan 10 08:09 32/900 Framistan Order
> N 4 jones Fri Jan 10 13:22 35/1347 Draft Report
N 5 smith@somewhere.com Sat Jan 11 13:21 76/3103 Excerpt from book
?
```

Pretpostavimo da čitate tekuću poruku tako da pritisnete <Return>, a potom poruku 1 tako da na ? *prompt* utipkate 1 i pritisnete <Return>. Ako pritisnete <q><Return> za napuštanje programa, dobit ćete ove informacije:

Saved 2 messages in /home/bkorn/mbox held 3 messages in /var/spool/mail/bkorn

Dvije poruke koje ste čitali spremljene su u datoteku mbox u polaznom direktoriju; ostale tri spremljene su u poštanski sandučić sustava /var/spool/mail/bkorn.

Ako često ovako spremate poruke, mbox može postati jako velik. Povremeno možete brisati njegov sadržaj. Također, poštu možete čitati iz te datoteke kao da je ona vaš poštanski sandučić sustava, što ćemo kasnije opisati.

## BILJEŠKA Možete čitati poštu i odrediti da se tekuća poruke čuva u vašem poštanskom sandučiću sustava, /var/spool/mail/bkorn, a ne u datoteci mbox. To ćete napraviti tako da nakon čitanja poruke u ? prompt unesete **pre** (za *preserve*, sačuvaj).

#### Napuštanje programa bez spremanja promjena

Program mail možete napustiti i tako da na ? *promptu* pritisnete <x><Return>. Kada to napravite, izlazite iz programa i ne mijenjate poštanski sandučić sustava ni bilo koju drugu datoteku – kao da uopće niste čitali poštu. Potom dobijete *prompt* ljuske. Tako možete napustiti program mail i spremiti poštu u poštanski sandučić sustava.

## Uporaba programa elm

Za Linux postoji nekoliko programa za e-poštu. Svaki ima i prednosti i nedostatke.

Sa Slackware i Red Hat distribucijom Linuxa isporučuje se program elm za e-poštu. Taj program ne nalikuje linijskom editoru, već tekst procesoru. Osigurava skup interaktivnih izborničkih elelmenata i lagan je za uporabu. Sve što možete napraviti s programom mail, možete napraviti i u programu elm i to puno lakše!

Kako je elm lagan za uporabu, u sljedećim odjeljcima samo se daju natuknice za njegovu uporabu. Više informacija dobit ćete na Mreži, u elmovom sustavu pomoći ili ako pročitate njegovu man page.

#### Pokretanje programa elm

Jednostavno u naredbeni redak upišite elm. Ako prvi puta rabite elm, on će od vas tražiti dozvolu za postavljanje konfiguracijskog direktorija za vaš račun i izradu datoteke mbox, ako ona već ne postoji. Ako prvi puta pokrećete program elm, dobit ćete ove retke:

```
$ elm
Notice:
This version of ELM rquires the use of a .elm directory in your home
directory to stor your elmrc and alias files. Shall I create the
directory .elm for you and set it up (y/n/q)? y
Notice:
ELM requires the use of a folders directory to store your mail folders in.
shall I create the directory /home/gunter/Mail for you (y/n/q)? y
Great! I'll do it now.
```

Kada elm napravi direktorij i datoteku mbox, izvodi glavni program za poštu. To je program za e-poštu koji radi preko čitavog zaslona. Zaslon vam se pročisti i vidite ovakav prikaz:

```
Mailbox is '/var/spool/mail/gunter' with 2 messages [ELM 2.4 PL25)
N 1 Nov 11 Jack Tackett Linux book
N 2 Nov 11 Jack Tackett more ideas
You can use any of the following commands
by pressing the first character;
d)elete or u)ndelete mail, m)ail a message,
r)eply or f)orward mail, q)uit
To read a message, press <return>. j = move down, k = move up, ? = help
```

Command:

Na vrhu zaslona, elm vam govori gdje je smješten vaš poštanski sandučić sustava, koliko je poruka u njemu i koja se inačica elma izvodi. Potom elm u vašem poštanskom sandučiću ispisuje jedan redak za svaku poruku. Slovo N odnosi se na novu poruku, kao i u programu mail. Sumarni redak za svaku poruku govori je li poruka nova, daje datom, pošiljtelja i predmet. (Kao i uvijek, prikaz se neznatno razlikuje ovisno o inačici elma.) Tekuća je poruka označena na popisu (u prethodnom popisu, tekuća je poruka otisnuta masno).

#### Uporaba naredbi iz elma

Tablica 33.3 Naredhe iz programa elm

Na dnu zaslona imate naredbe koje su dostupne za tekući zaslon. Kao što vidite iz prethodnog primjera, možete brisati ili vraćati obrisanu poštu, poslati poruku, odgovoriti na poruku, proslijediti poštu ili napustiti program. Ako pritisnete tipku <j> vraćate se na prethodnu poruku, ako pritisnite tipku <k> idete na slijedeću poruku. Pomoć pozivate pritiskom na tipku <?>. Command: *prompt* na dnu zaslona govori vam da pritisnete neku od tipki za naredbu.

Program elm je lako rabiti zbog velike interaktivnosti i pomoći koja je dostupna na zaslonu. U tablici 33.3 su sve naredbe koje se mogu izvesti iz programa elm.

Naredba/tipka	Opis
<return>, razmaknica</return>	Prikazuje tekuću poruku
1	proslijeđuje poruku sustavskoj naredbi
!	izlazak iz ljuske
\$	resinkronizira mapu
?	Prikazuje pomoć s Mreže
$+, < \mathbb{R} >$	Prikazuje slijedeću stranicu kazala
-, <¬>	Prikazuje prethodnu stranicu kazala
=	Postavlja tekuću poruku na prvu poruku
*	Postavlja tekuću poruku na posljednju poruku
<number><return></return></number>	Postavlja tekuću poruku na broj (number)
/	traži uzorak po redcima
//	Pretražuje tekst čitave poruke i traži uzorak
>	Sprema tekuću poruku u mapu
<	Pretražuje tekuću poruku tražeći unose datuma
a	Premješta se na "alias" način rada
b	Ponovno šalje tekuću poruku
С	Kopira tekuću poruku u mapu
С	Premješta se u drugu mapu
d	Briše tekuću poruku

Naredba/tipka	Opis
<ctrl-d></ctrl-d>	Briše poruke s određenim uzorkom
e	Uređuje tekuću mapu
f	Prosljeđuje tekuću poruku
g	Šalje grupni (svim primateljima) odgovor na tekuću poruku
h	Prikazuje zaglavlje s porukom
J	Povećava tekuću poruku za jedan
j, <->	Ide na sljedeću vraćenu poruku
K	Smanjuje tekuću poruku za jedan
k, <->	Ide na prethodnu vraćenu poruku
1	Ograničava poruke prema nekom kriteriju
<ctrl-l></ctrl-l>	Ponovno crta zaslon
m	Šalje poruku
n	Odlazi do sljedeće poruke i prikazuje tekuću, a potom povećava
0	Mijenja opcije programa elm
р	Ispisuje tekuću poruku
q	Napušta program, uz upozorenje o brisanju, pohrani i čuvanju poruka
Q	Brzi napuštanje programa bez upozorenja
r	Odgovara na tekuću poruku
S	Sprema tekuću poruku u mapu
t	Označava tekuću poruku za dalji rad
Т	Tags tekuću poruku i odlazi na slijdeću poruku
<ctrl-t></ctrl-t>	Označava poruke određenim uzorkom
u	Vraća obrisanu tekuću poruku
<ctrl-u></ctrl-u>	Vraća obrisane poruke koje imaju određeni uzorak
x, <ctrl-q></ctrl-q>	Izlazi ostavljajući mapu nedirnutu; pita želite li izaći ako ste mijenjali mapu
Х	Izalazi ostavljajući mapu nedirnutu, bez uvjeta

## Uporaba Mutt klijenta za e-poštu

Mutt je *freeware* klijent za e-poštu čija popularnost ubrzano raste. Iako ga mogu rabiti i početnici (podrazumijevano Mutt izgleda i radi kao elm klijent), Mutt je posebno popularan kod naprednih korisnika jer ga se može izuzetno dobro prilagođavati.

Osnovne osobine programa Mutt su:

- Podrška za boju
- Podrška za nizanje poruka
- Podrška za MIME uključujući RFC2047 podršku za kodirana zaglavlja i PGP/MIME (RFC2015)
- Podrška za POP3
- Podrška za višestruke formate poštanskog sandučića uključujući mbox, MMDF, MH i maildir
- Povezivanje tipaka s određenim akcijama (podrazumijevano je isto kao u programu elm)
- Sposobnost pretraživanja uz uporabu dozvoljenog izraza
- Podrška za Delivery Status Notification (DSN)
- Sposobnost uključivanja privitaka iz naredbenog retka kada se sastavlja poruka
- Sposobnost odgovaranja ili prosljeđivanja višestrukih poruka odjednom
- .mailrs stil konfiguracijskih datoteka
- U postupku instaliranja rabi se GNU autoconf

#### Gdje dobiti Mutt

Mutt se dijeli pod uvjetima GNU licence na CD-ROM-u za instalaciju Red Hata, a smješten je u direktoriju Mail u dijelu Applications. Međunarodna inačica programa Mutt koja ima podršku za PGP<sup>™</sup> dostupna je na nekoliko FTP poslužitelja, uključujući i **ftp://FTP.gbnet.net/pub/mutt-international**/.

## Još informacija o programu Mutt

Polazna stranica Mutta je http://www.cs.hmc.edu/čme/mutt/index.html. Na toj su stranici veze prema mrežnom priručniku za Mutt koji je napisao Michael Elkins (na http://www.math.fu-berlin.de/čleitner/mutt/faq.html). Polazna stranica također ima informacije o nekoliko poštanskih popisa posvećenih klijent programu za e-poštu.

## Odavdje...

Više informacija o razmjeni informacija preko Interneta možete naći u sljedećim poglavljima:

- 31. poglavlje "Kretanje Internetom pomoću World Wide Weba", opisuje različite vrste informacija dostupnih na Internetu.
- 34. poglavlje "Preživljavanje u novinskim grupama Useneta", opisuje Usenetov sustav novinskih grupa koji se rabi za razmjenu javnih poruka.

# poglavlje 34

## Preživljavanje u novinskim grupama Useneta

(Steve Burnett)

## U ovom poglavlju

Što su Usenetove novinske grupe? Rječnik pojmova Useneta Kratka povijest Kako je Usenet ustrojen Bez središnje vlasti Kultura ponašanja na Usenetu Čitanje i objava novinskih grupa Netiquette na Usenetu Uporaba rn programa za čitanje novinskih grupa Zbog izuzetno brzog rasta Interneta, Usenetove novinske grupe privukle su puno pažnje. Puno servisa na Mreži nudi pristup Usenetu. No, što je Usenet? Usenet je kartica od *User Network*, proto-mreže računala koja razmjenjuju informacije grupirane po hijerarhiji tema. Izraz proto-mreže rabi se zbog toga što Usenet nije fizička mreža u običnom smislu. Napravljena je od svih računala koja razmjenjuju Usenetove novinske grupe.

## Što su Usenetove novinske grupe?

Najjednostavnija definicija Usenetovih novinskih grupa (*Usenet news, netnews* ili jednostavno *news*) je forum za rasprave na Mreži. Puno računala na cijelom svijetu razmjenjuje informacije, nazvane *člancima (articles)* o svakoj mogućoj temi. Ta računala nisu fizički povezana na istu mrežu; ona su logički povezana svojom sposobnošću razmjene podataka. Tako čine logičku mrežu koju nazivamo Usenet.

BILJEŠKA Softver koji pokreće Usenet podijeljen je na dva dijela: program za čitanje novinskih grupa (softver koji korisnici rabe za čitanje i oglašavanje članaka) te softver koji obrađuje članke i prenosi ih između sustava.

Kada se govori o Usenetu, puno ljudi misli na BBS. Iako Usenet na prvi pogled ima neke sličnosti, postoje bitne i važne razlike:

- Različiti novinski članci nisu prisutni na jednom računalu, kao što je to kod BBS-a. Oni se šalju od računala do računala mehanizmom spremanja i daljnjeg slanja. Svaka lokacija koja primi novinske grupe razmjenjuje članke s jednim ili više susjeda u transakcijama. Rezultat je da novinskim člancima treba vremana za širenje od jednog do drugog mjesta.
- Nitko nije zadužen. Usenet nema općeg upravitelja kao što je administrator BBS-a. Svaka lokacija ima veliku autonomiju. Usenet se može opisati kao "organizirana anarhija".

Općenito, Usenet se može podijeliti na dva logička dijela: programi i protokoli koji čine mehanizam za objavljivanje članaka i prijenos novinskih članaka među računalima, te korisničke programe za čitanje i objavljivanje članaka novinskih grupa. U ovom poglavlju uglavnom se govori o korisničkom dijelu.

## Rječnik pojmova Useneta

Usenetove novinske grupe imaju svoj ustroj i kulturu, o čemu se govori u odjeljku "Kultura ponašanja na Usenetu". Usenet ima i svoju terminologiju. Ti izrazi zbunjuju nove korisnike, posebno one koji rabe BBS sustave. U tablici 34.1 je kratki rječnik pojmova uobičajenih izraza iz Useneta.

Izraz	Definicija
article (članak)	Poruka oglašena na novinskoj grupi.
bandwidth (širina pojasa)	Tehnički izraz koji se odnosi na količinu podataka koji može podnijeti sredstvo prijenosa. Često se rabi u izrazu <i>waste of bandwidth</i> ( <i>traćenje prostora</i> ) za članke koji imaju malo korisnih informacija.
BTW	Skraćenica za By The Way (usput).
FAQ	Skraćenica za <i>Frequently Asked Questions (često postavljana pitanja).</i> Puno novinskih grupa ima FAQ koji redovito oglašava. Obično se smatra nepristojnim objaviti pitanje čiji je odgovor u FAQ-u neke grupe.
flame (bijesni odgovor?)	Članak koji je pun grubih, gnjevnih i uvredljivih izjava o nekoj osobi.
FYI	Skraćenica za For Your Information (za vašu informaciju).
hierarchy (hijerarhija)	Usenetov sustav grupiranja novinskih grupa u hijerarhijski ustroj na temelju sadržaja.
IMHO	Skraćenica za In My Humble Opinion (prema mojem skromnom mišljenju).
newsgroup (novinska grupa)	Logička skupina članaka o općoj temi.
news reader (program za čitanje	Korisnički program poput rna koji se rabi za čitanje i oglašavanje članaka na Usenetu.novinskih grupa)
net.personality	Netko tko je poznat na Usent ili Internet zajednici.
net.police	Izmišljena organizacija odgovorna za nametanje pravila o Usenetu. Obično se rabi sarkastično.
netiquette	Pravila ponašanja na Usenetu.
newbie	Novi korisnik Usenetovih novinskih grupa.
quoting (citiranje)	Uključuje dijelove poruke na koju odgovarate. Većina programa za čitanje novinskih grupa omogućuje citiranje članaka. Trebate citirati samo važne dijelove članka radi štednje prostora. Kadkad se naziva i <i>quotebacks</i> .
ROFL	Skraćenica za Rolling On the Floor, Laughing (kotrljati se po podu od smijeha).
RTFM	Skraćenica za <i>Read The Forgotten Manual (pročitajte zaboravljeni priručnik)</i> . Obično se rabi ovako: "Evo kratkog odgovora na vaše pitanje. RTFM pa ćete dobiti više informacija."
post (objaviti, poslati)	Poslati članak novinskoj grupi.

Tablica 34.1 Uobičajeni izrazi iz Useneta

nastavlja se

Tablica 34.1 Nastavak	
Izraz	Definicija
signal-to-noise (odnos signal-šum)	Tehnički izraz koji se odnosi na odnos podataka prema pozadinskom šumu. Na Usenetu se taj izraz odnosi na količinu korisnih informacija u novinskoj grupi u odnosu na brbljanje u pozadini koje nema veze s temom. Veliki odnos signal-šum odnosi se na novinsku grupu koja ima puno korisnih informacija i malo brbljanja koje nema veze s temom. Izraz signal-to-noise može se uporabiti i kao opis neke osobe.
signature (potpis)	Kratka datoteka koja se nalazi na kraju svih vaših oglasa. Obično ima vaše ime, adresu e-pošte, a katkad i neki duhoviti citat.
sig file	Pogledajte signature.
smileys	Česte oznake za osjećaje u oglasu ili poruci e-pošte. Na primjer, : -) i :-( su sretno i tužno lice. (Nakrivite glavu prema lijevom ramenu i pogledajte ih sa strane.)

## Kratka povijest

Krajem 1979. dva sudenta sa Duke University razmišljala su kako povezati UNIX računala tako da mogu razmjenjivati poruke s tekstom. Njihovim se naporima priključio student sa University of North Carolina koji je napisao prvi program za sustav prijenosa novosti, a koji se sastojao od zbirke skripti ljuski. Taj softver je instaliran na prve dvije Usenetove lokacije, unc i duke. Početkom 1980., dodano je još jedno računalo na Duke University pod imenom phs. Softver za novinske grupe napisan je u C jeziku za javnu uporabu. To je postalo poznato kao *A News* softver.

Kako je popularnost tog softvera rasla, tekući sustav prijenosa novinskih grupa nije više mogao pratiti količinu novina. Godine 1982. programeri na University of California na Berkleyu povećali su sposobnosti softvera i tu inačicu nazvali *B News*.

Svo to vrijeme članci novinskih grupa su se prenosili uporabom protokola UNIX-to-UNIX Copy Program (UUCP). Mreži se priključivalo sve više lokacija, pa je ona prerasla mogućnosti. Kako UUCP više nije vrijedio kao protokol za prijenos novinskih grupa, potražila se pomoć od Interneta i TCP/IP protokola. Godine 1986. napravljen je softver koji je uveo Network News Transport Protocol (NNTP). Taj je protokol definiran u RFC 977. NNTP je omogućio razmjenu članaka uporabom TCP/IP protokola umjesto sporog UUCP-a. On omogućuje korisnicima čitanje i objavljivanje novina s udaljenih računala tako da se na svako računalo ne mora instalirati glavni softver za obradu novinskih grupa.

Kada je NNTP postao dostupan na Netu, razvoj Useneta krenuo je ogromnom brzinom. Program B News je postao prespor. Godine 1987, Henry Spencer i Geoff Collyer sa University of Toronto razvili su novi softver za obradu novinskih grupa, *C News*. Potom je Rich Salz razvio sustav za prijenos novinskih grupa poznat kao *INN*, jedan od najčešće upotrebljavanih novinskih poslužitelja na Internetu. Usenetov sustav novinskih grupa razvija se velikom brzinom. Drugi komercijalni ISP-i sada imaju Usenetove novinske grupe kao dio svoje uskluge na Mreži. Čak i neke BBS mreže, kao što su FidoNet imaju Usenetove novinske grupe.

BILJEŠKA O povijesnom razvoju Usenetovih novinskih grupa naći ćete u članku "Usenet software: History and Spurce", koji je napisao dr. Gene Spafford. Taj se čalanak može naći na World Wide Webu na http://www.faqs.org/faqs/usenet/software/part1/.

## Kako je Usenet ustrojen

Postoji doslovce preko 20 000 novinskih grupa. Za sada nitko ne zna njihov točan broj. Postoji tisuće grupa o gotovo svakoj temi, a njihov broj svaki dan raste. Teme se kreću od smiješnih ili besmislenih do jasnih i korisnih.

#### Hijerarhije novinskih grupa

Kako biste lakše našli informacije o onom što vas zanima, novinske grupe su organizirane po hijerarhiji koja se temelji na predmetu. Imena novinskih grupa napravljena su od subnames odvojenih točkom. Ta imena, dok ih čitate s lijeva na desno, kreću se od općih kategorija do određene kategorije. Na vrhu hijerarhije je nekoliko standardnih kategorija grupa, te puno specijaliziranih kategorija. Te standardne kategorije su dobro utemeljene. U tablici 34.2 je popis standardnih kategorija grupa najviše razine na Usenetovom sustavu novinskih grupa.

Tablica 34.2	Standardne kategorije grupa najviše razine u Usenetovoj hijerarhiji
Grupa	Opis
comp	Puno različitih tema povezanih s računalima
misc	Mješovite teme koje se ne uklapaju u druge kategorije
news	Različite teme povezane sa samim Usenetovi sustavom novinskih grupa
rec	Teme o rekreaciji i hobijima
soc	Društvene teme
sci	Različite znanstvene teme
talk	Sadržaji o kojima se razgovara

Kao i kod svega drugog na Internetu, postoje iznimke za pravila iz tablice 34.2. Postoji još i druge hijerarhije najviše razine; najčešće su posvećene različitim predjelima svijeta. Na primjer, **ba** i **triangle** govore o temama tipičnim za područje zaljeva San Franciscoa i parka North Carolina Research Triangle Park.

O jednoj od takvih posebnih hijerarhija grupa ćemo posebno raspravljati. Hijerarhija **alt** ima jako labava pravila za izradu novinske grupa. Pod tom hijerarhijom doslovce svatko može napraviti grupu; međutim, izrada novinske grupe pod ostalim grupama najviše razine je izuzetno teška. Hijerarhija **alt** ima puno novinskih grupa koje raspravljaju o temama koje su izvan glavnih interesa društva. U stvari, puno ljudi u toj hijerarhiji nalazi teme koje su vrijedne prigovora. Pokrenuto je puno rasprava o cenzuriranju jer su lokacije odlučile zabraniti dijelove ili čitave novinske grupe u **alt** hijerarhiji.

#### Distribucija novosti

Osim grupiranja članaka po hijerarhiji, Usenet ima mogućnost ograničavanja širenja članka u sustavu novinskih grupa. Tim se mehanizmom može ograničiti članke na određeno zemljopisno područje. Ako je raspodjela postavljena na određeno područje, članak će dobiti samo lokacije unutar tog područja distribucije. Administrator sustava svake lokacije odlučuje koja distribucija odgovara toj lokaciji.

Zašto ograničavati distribuciju članka? Recimo da živite u North Carolina i objavite oglas o susretu lokalne korisničke grupe. To vjerojatno ne zanima korisnike u Australiji. Ograničavanjem distribucije članka na zemljopisno područje možete uštediti prostor, smanjiti troškove slanja poruke i smanjiti broj ogorčenih korisnika širom svijeta kojima vaša poruka ne znači ništa.

Distribuciju članka možete ograničiti uključivanjem retka Distribution: u zaglavlje članka. Većina programa za čitanje novinskih grupa od vas traži ograničenja kada objavljujete članak. Nakon dvotočke u retku Distrubution: unesite odgovarajuće zemljopisno područje. U tablici 34.3 je popis područja distrubucije.

Tablica 34.3	Vrijednosti distribucije novinskih grupa
Vrijednost	Objašnjenje
local	Obično su članci za mjesnu distribuciju ograničeni na grupu lokalnih novinskih poslužitelja unutar vaše organizacije. Ta se distribucija često rabi za lokalne novinske grupe.
nc	Svaka država ima svoju distribuciju koja je ista kao i poštanska kratica za državu. U primjeru Distribution: nc ograničava se distribucija članka u državi North Carolina.
us	Šalje članak svim Usenetovim lokacijama u USA.
na	Šalje članak svim Usenetovim lokacijama u Sjevernoj Americi.
world	Šalje članak svim Usenetovim lokacijama na svijetu. Obično je to podrazumijevana distribucija ako ništa drugo nije određeno.

Možda će se vaš članak trebati razdijeliti u posebna područja, poput jedne organizacije ili regije. Općenito, trebate izabrati distribuciju koja vaš članak šalje samo u područje interesa.

## Bez središnje vlasti

Usenet nema središnju vlast i to zbunjuje ljude. Vaš lokalni administrator sustava ima ovlast samo nad lokalnim sustavom. Nema središnje grupe ni organizacije koja upravlja i prima žalbe. Usprkos tome, Usenet radi izuzetno dobro. Dapače, ljudi tvrde da radi bolje nego što bi radio da postoji središnja ustanova.

Kako to uspijeva? Usenet se pokreće suradnjom među lokacijama i klijentelom koja ju je razvila.

Usenet sam sebe dobro kontrolira. Ako korisnik počne zluopotrebljavati mrežu, budite sigurni da će on i njegov administrator sustava dobiti tisuće poruka e-pošte i da će ih zvati telefonom. Tako se problem rješava vrlo brzo.

## Kultura ponašanja na Usenetu

Usenet ima svoju kulturu ponašanja. Trebali biste je upoznati prije uporabe Useneta, to će vam olakšati posao.

Posljednjih je godina velik broj *online* servisa svojim uslugama dodalo Usenet. Tako je Usenetove novinske grupe počelo čitati puno novih korisnika. Većina njih se žali da su oni koji sudjeluju u Usenetu neodgojeni. No, kultura ponašanja Useneta se jednostavno razlikuje od ponašanja na svim drugim servisima. Ako je usporedite s ponašanjem na Netu, Usenet će vam se činiti jednostavnijim.

Svaki dan Usenetove novinske grupe čitaju milijuni ljudi širom svijeta. Ti su ljudi jako različiti po svemu. Kako se Usenet pojavljuje na računalima širom svijeta, on je doista dio međunarodne zajednice. Ljudi koje susrećete preko Usenta ne govore iste jezike, imaju različitu kulturu, pripadaju različitim etičkim skupinama, vjerama i imaju drugačije društvene vrijednosti.

Jedna strana Usenetove kulture ponašanja, *flame(bijesna poruka?)*, je obično neugodna za nove korisnike. Flame je surova poruka, obično puna uvreda, koju netko objavi kao odgovor na vaš članak. Tu ne možete učiniti ništa, jednostavno ne obraćajte pažnju. Neki ljudi to vole raditi, jeftinije im je od psihoterapije...

## Nedostatak vizualne informacije

Elektroničkoj komunikaciji nedostaje razgovor oči u oči. Kada ljudi osobno razgovaraju, primaju informacije na svjesnoj i podsvjesnoj razini. Kako ne možete vidjeti ljude koji čitaju i oglašavaju na Usenetu, nema vizualnih dojmova. Kako mi često rabimo jezik tijela i vizualne dojmove za prikazivanje osjećaja, lako se može dogoditi da krivo razumijemo nečiju poruku.

Srećom, postoje dogovori na Usenetu koji djelomično nadomještaju taj nedostatak vizualnih dojmova. Neke izraze možete dodatno naglasiti stavljenjem zvjezdica, "I \*really\* mean it!" Uporaba velikih slova se smatra vikanjem. Ako slučajno zaboravite isključiti tipku <Caps Lock>, pa pošaljete oglas pisan velikim slovima, vjerojatno će vas na to upozoriti.

U poruci možete izraziti i osjećaje. Na primjer, pokraj sarkastične izjave ćete upisati <sarkazam>. *Smileys*, nazvani još i *ikonice (emoticons)*, također objašnjavaju osjećaje. Smješko je ASCII prikaz lica koji gledate sa strane. Npr., :-) je sretno lice, a :-( je tužno lice.



#### NA WEBU

Popis različitih lica dostupan je na Webu. Pregledajte ga, neka su lica doista originalna, no najčešće se rabi sretno i tužno lice. Uporaba nekih rijetkih lica može samo zbuniti ljude. Lica ćete naći na lokaciji:

http://www.eff.org/pub/Net\_culture/Folklore/Arts/smiley2.list

#### Kultura ponašanja unutar novinske grupe

Svaka novinska grupa na Usenetu ima svoja pravila ponašanja. Novinske grupe bave se različitim temama za koje su zainteresirane drugačiji tipovi ljudi. U nekim je grupama puno studenata, u drugim znanstvenika.

Neke grupe bave se tehnikom, npr. **comp** i **sci** i orijentirane su na rasprave o činjenicama. Članovi tih grupa zainteresirani su za činjenice i teme u svezi tehnike. Kada tamo objavljujete, brižno sastavite članak i dodajte reference. Hijerarhija kao što je **rec** više je orijentirana na mišljenja. Najčešće ćete na svoj članak dobiti odgovore koji odražavaju tuđa mišljenja, a koja se mogu prilično razlikovati od vašeg. U grupama **talk** i **misc** razvijaju se vrlo vruće rasprave, recimo o umjetnom pobačaju i kontroli oružja. Ako ste novi na Usenetu, najprije se upoznajte s grupom, a potom oglašavajte. Budite spremni na svakakve odgovore.

BILJEŠKA Kada počnete čitati neku novinsku grupu, najprije se upoznajte s njezinim pravilima ponašanja, a onda oglašavajte. Najmanje nekoliko dana čitajte sadržaj tako da dobijete osjećaj kakvo je poželjno, a kakvo nepoželjno ponašanje. Pregledajte i FAQ (ako novinska grupa nije objavila FAQ, uz pomoć pretraživačkog servisa pokušajte pronaći pomoć).

U nekoliko novinskih grupa ograničena je objava članaka. To su takozvane *modreirane novinske grupe (moderated newsgroups)*. Njima upravlja osoba zvana *moderator*. On mora odobiriti sve članke prije objave. Moderator odlučuje odgovara li članak i ako da, objavljuje ga na grupi. Većina softvera za novinske grupe automatski otkriva je li novinska grupa moderirana, a ako jest, članak šalje moderatoru.

## Čitanje i objava novinskih grupa

U ovom odjeljku možete doznati kako se općenito čitaju i objavljuju novinske grupe; točniji detalji ovise o softveru za čitanje novinskih grupa. Dostupno je puno različitog softvera. Dosta ljudi rabi Web pretraživače s ugrađenim programima za čitanje novinskih grupa, kao što je Netscape. Drugi rabe *linijske* alate, kao što je rn. No, ovi opći pojmovi mogu se primijeniti na sav softver za čitanje novinskih grupa.

#### Pretplata na novinske grupe

Želite li čitati novinske grupe, najprije morate odlučiti koju novinsku grupu želite čitati. Postupak odabira novinske grupe zove se *pretplata* (*subscribing*).

Većina programa za čitanje novinskih grupa nudi vam popis dostupnih novinskih grupa, pa možete odabrati onu koja vas zanima. Postupak pretplate razlikuje se ovisno u softveru, ali obično morate izabrati niz novinskih grupa s popisa. Od tada ćete vidjeti samo one grupe na koje ste se pretplatili. Uvijek se možete pretplatiti na nove grupe ili otkazati pretplatu na stare.

Sjetite se da postoji preko 20 000 novinskih grupa, pa će vašem poslužitelju trebati neko vrijeme dok učita popis svih tih grupa.

## Čitanje novosti

Kada ste se pretplatili na novinsku grupu, možete čitati novosti. Novinsku grupu odaberete s popisa grupa na koje ste se pretplatili. Program za čitanje novinskih grupa će vam prikazati popis tema. Teme mogu biti razvrstane, ali i ne moraju. Neki program iza čitanje novinskih grupa razvrstavaju članke po sadržaju, pokazujući koji su članci odgovori na neke druge. To se zove *nizanje (threading)*.

Kada odberete članak koji ćete čitati, na vrhu ćete dobiti neke informacije. To je *zaglavlje člana-ka (article header)*. Ono sadržava puno informacija o članku, uključujući autora, datum pisanja, novinsku grupu gdje je objavljen i stazu po kojoj je došao na vašu lokaciju. Moguće su i druge informacije, npr ključne riječi ili organizacija kojoj autor pripada.

Kod većine programa za čitanje novinskih grupa članak se označava pročitanim čim ga pogledate. Obično se pri odabiru novinske grupe pokazuju samo novi članci. To znači da ako ste jednom pogledali članak, on se više neće pojaviti na popisu. Ako ga želite zadržati, možete ga spremiti na disk ili ispisati. Možete ga označiti kao da nije pročitan. Većina programa za čitanje novinskih grupa omogućuje vam pristup starim člancima koji još nisu obrisani.

### Odgovaranje putem e-pošte

Kada pročitate članak i odlučite na njega odgovoriti, možda vaš odgovor neće zanimati sve sudionike novinske grupe. Većina programa za čitanje novinskih grupa vam omogućuje odgovoriti na članak putem e-pošte.

Ako na članak odgovarate putem e-pošte, program za čitanje novinskih grupa rabi informacije iz zaglavlje članaka za dobivanje adrese e-pošte autora, te poziva editor za e-poštu. Obično imate mogućnost uključivanja originalnog članka u odgovor. Ako ne uključite cijeli članak, onda barem poruku uredite tako da uključite bitne dijelove.

Zbog lako dostupnih adresa e-pošte na mreži, puno onih koji objavljuju na Usenetu *munge* ili mijenjaju svoje adrese e-pošte kako bi spriječili netraženu, komercijalnu e-poštu nazvanu *neželjena pošta (spam)*. Recimo, adresa e-pošte može namjerno imati krivi unos, poput "mjameson@IHATESPAM.netcrom.com". U sigfile (datoteci s potpisom) mogu biti upute što treba mijenjati u toj adresi. Na primjer, za krivu adresu "sbarnes@sequoia.skytails.org" mogu postojati upute "Zamijeni 'tails' sa 'wings' u odgovoru".

## Objavljivanje članka

Postupak izrade novog članka i slanje preko Usenet sustava zove se *objavljivanje članka (posting an article)*. Kada odlučite objaviti članak, možete ili objaviti slijedni članak na drugi članak ili napraviti novi članak o nečemu. Vaš program za čitanje novinskih grupa obično ima različite naredbe za različite vrste objave.

**Objavljivanje slijednog članka** *Slijedni članak (follow-up article)* je odgovor na drugi članak. Taj članak bavi se istim sadržajem kao i originalni, a prikazan je kao odgovor, dio niti jednog raz-govora.

Kada objavljujete slijedni članak, možete izabrati uključivanje prvobitnog. To je dobar način za pisanje jasnog odgovora jer između objave originalnog članka i vašeg odgovora može proći nekoliko dana. Ako uključite originalni članak, nastojte uključiti ili *citirati (quote)* samo dijelove članka koji su važni za vaš odgovor. Zamorno je među mnoštvom citata tražiti novu informaciju. Neki novinski poslužitelji mogu i odbiti vaš odgovor ako citat zauzima više od određenog postotka vaše poruke.

Trebate provjeriti i Subject – govori li on točno o saržaju vašeg oglasa i promijeniti ga ako se raspravlja o novoj temi. Provjerite i redak Newsgroup – pripada li slijedni članak u odgovarajuću novinsku grupu. Posebno razmotrite odgovara li slanje poruke za više novinskih grupa ili se odnosi samo na jednu ili dvije.

**Objavljivanje novog članka** Ako započnete raspravu o novoj temi, objavit ćete novi članak. Mehanizam objavljivanja novog članka sličan je prethodnom. Programu za čitanje novinskih grupa dajete odgovarajuću naredbu; on vas traži neke informacije, kao što su odredišna novinska grupa, predmet i distribucija. Glavna je razlika da vi stvarate nit umjesto odgovaranja na poruku.

**SAVJET** Dokumentacija o stilu pisanja na Usenetu redovito se objavljuje u novinskoj grupi **news.announce.newusers.** 

Dok pišete članak, trebate razmisliti o mnoštvu stvari. Možete se poslužiti člankom "Savjeti o stilu pisanja na Usenetu". U tim savjetima govori se o formatu i sadržaju članka.

Reci trebaju imati manje od 80 slova jer puno terminala ne može prikazati retke dulje od 80 znakova. Također, članak ne bi trebao imati više od 1000 redaka. Neke lokacije i dalje izvode starije inačice softvera za prijenos novinskih grupa, pa ima dugački članci uzrokuju probleme.

Moguće je uključiti i datoteku s potpisom koja se automatski dodaje na kraju oglasa. Iako većina programa za čitanje novinskih grupa podržava takve datoteke, točan mehanizam se razlikuje. Većina ljudi u datoteku s potpisom stavlja ime i adresu e-pošte, te svoju adresu. Neki ljudi dodaju i šaljive citate ili sličice u ASCII kôdu. Neka vaša datoteka s potpisom bude kratka. Smatra se lošim ponašanjem uključiti ime, nadimke, citate i crtež vašeg auta. Pravilo je da se ograničite na četiri retka. Neki softveri za novinske grupe automatski ograničavaju potpis na četiri retka.

Kada objavljujete članak, on mora imati predmet. Neka to bude kratko i jasno. Puno ljudi pretražuje po predmetima pa im tako dajete mogućnost da pročitaju vaš članak ako ih zanima tema. Brižno odaberite i novinske grupe u kojoj ćete objaviti članak. Većina programa za čitanje novinskih grupa omogućuje objavljivanje članka na više novinskih grupa. Trebali biste odabrati najmanji mogući broj grupa, ionako svaku grupu čita tisuće ljudi.

## Netiquette na Usenetu

U ovom poglavlju naglašava se važnost sadržaja i tona vaše poruke i njezinog tumačenja. Opća pravila ponašanja na Usenetu i Internetu općenito zovu se *netiquette*. Netiquette se odnosi na sva područja Interneta, uključujući elektroničku poštu.

Izraz *netiquette* odnosi se na "prikladno i uljudno" ponašanje. Sve dok imate na umu da je Usenet jako velik i različit, nećete imati problema. Jednostavno, imajte na umu da na Usenetu nemaju svi vaše znanje, uvjerenja ili položaj.

Jasno izrazite svoje ideje. Nedostatak jezika tijela i kašnjenje između objave i odgovora može vrlo lako rezultirati pogrešnim tumačenjem. Također, ne govore svi na Usenetu isti jezik, pa ne poznaju loklane izraze i sarkazam.

Oglašavanje na Usenetu nije dobrodošlo. Postoje novinske grupe koje oglašavaju proizvode ili usluge. Nisu poželjni ni lanci sreće. Ukoliko ih objavite, na sebe ćete navući bijes tisuće ljudi.

Nemojte objavljivati flames, posebno one koji govore o gramatici i pravopisu. Iako se flames čine neizbježni dio Useneta, osobni napadi i bijesne poruke ne postižu ništa. Ako vas netko flame zbog vašeg oglasa, smirite se i brižno razmislite kako ćete mu odgovoriti; najbolje je uopće ne odgovoriti. Katkad možete primiti flame, ali smireni odgovor može rezultirati isprikom osobe koja ga je poslala. Ako bijesno pošaljete još jedan flame, samo povećavate sukob.

Ako vam netko radi velike probleme možete ga dodati u kill file, konfiguracijsku

BILJEŠKA datoteku programa za čitanje novinskih grupa koja ima popis korisnika ili predmeta.

Sve što se pojavljuje u kill file automatski se ne prikazuje pri čitanju novinskih grupa. Većina

programa za čitanje novinskih grupa ima neku inačicu te datoteke. To je prilično bezbolan način rješavanja problema.

Općenito, potrebno je malo zdravog razuma i uljudnosti, pa nećete imati problema na Usenetu. Međutim Usenet je ogromno područje s mnoštvom ljudi i vi ne možete vakog usrećiti. Vjerojatno ćete se susresti s flame porukom.

## Uporaba rn programa za čitanje novinskih grupa

Dostupno je puno različitih vrsta softvera za čitanje novinskih grupa koje ne možemo opisati. Česti takav program je rn. On se može naći na gotovo svakoj varijanti UNIX-a. Razvio ga je Larry Wall i lako je dostupan. Iako taj program nije najlakši za čitanje novinskih grupa, on je među najpopularnijim postojećim. Program rn omogućuje čitanje novinskih grupa preko ASCII sučelja koje je prikladno za lokalni rad na terminalu ili s udaljene mreže.

**BILJEŠKA** Popularan je još jedan program za čitanje novinskih grupa, trn, koji se distribuira s Linuxom. Taj je program gotovo identičan rnu, osim kod podrške za vezane članke (niti). Više informacija o tim mogućnostima trna dobit ćete na Linux man stranici.

Kada prvi puta pokrenete rn, dobit ćete poruku dobrodošlice, te popis novinskih grupa. Odmah se možete pretplatiti na neke od njih. Ako na lokaciji imate puno novinskih grupa, za to može trebati prilično vremena. Program rn sprema vašu pretplatu u polazni direktorij u datoteku .newsrc.

Kada ste završili s pretplatom, rn vas postavlja u način rada za odabir novinske grupe. Ime svake novinske grupe na koju ste se pretplatili prikazuje se jedno po jedno. Možete unijeti ime novinske grupe i pritiskom na <y> početi čitati članke, pritiskom na <n> možete prijeći na slijedeću grupu, ili se vratiti na prethodnu novinsku grupu pritiskom na <q>. Možete dobiti i popis predmeta u novinskoj grupi ako pritisnete <=> na *promptu* novinske grupe. Većina naredbi u rnu i trnu imaju jedan znak, a pomoć je dostupna u naredbenom retku kada pritisnete <h>.

Kada odaberete novinsku grupu, pokrećete način rada za odabir članaka. Tu postoje naredbe koje vam pomažu pri kretanju među člancima u novinskoj grupi. U tablici 34.4 je popis nekih naredbi dostupnih u u ovom načinu rada.

Tablica 34.4         Neke naredbe dostupne u načinu rada za odabir članaka	
Naredba	Opis
<n><razmaknica></razmaknica></n>	Pretražuje prema naprijed i traži slijedeći nepročitani članak. Razmaknica to radi samo na kraju članka.
<razmaknica></razmaknica>	Pokazuje sljedeću stranicu tekućeg članka.
<shift-n></shift-n>	Ide na sljedeći članak.
<ctrl-shift-n></ctrl-shift-n>	Ide na sljedeći članak s istim predmetom.

nastavlja se

Tablica 34.4 Nasta	vak
Naredba	Opis
>	Pretražuje unatrag kako bi se pronašao članak koji nije pročitan; ostaje na tekućem članku ako ne nađe ni jedan.
<shift-p></shift-p>	Ide na prethodni članak.
<ctrl-shift-r></ctrl-shift-r>	Ide na posljednji prethodni članak sa istim predmetom kao i tekući članak.
<h></h>	Prikazuje pomoć za način rada za odabir članaka.
< <u>r</u> >	Odgovara autoru članka putem e-pošte.
<shift-r></shift-r>	Odgovara autoru članaka preko e-pošte, uključujući i tekući članak.
<f></f>	Objavljuje slijedni članak.
<shift-f></shift-f>	Objavljuje slijedni članak, uključujući originalni članak.
<s>filename</s>	Sprema tekući članak u datoteku <i>filename</i> .
<q></q>	Napušta tekuću grupu i vraća se na odabir novinskih grupa.

To su samo neke dostupne opcije u rnu i trnu. Ti programi omogućuju puno korisnički prilagodbi. Više informacija dobit ćete na man stranicama i sustavu pomoći na Mreži.

## Odavde...

U ovom poglavlju istražili smo ustroj Useneta, osnove o čitanju i objavljivanju članaka, hijerarhiju novinskih grupa i opća pravila ponašanja na Usenetu. Uz malo strpljenja, Usenetove novinske grupe će vam biti neizmjeran izvor informacija. O elektroničkoj komunikaciji i Internetu možete naći i u ovim poglavljima:

- 31. poglavlje "Kretanje Internetom pomoću World Wide Weba", u kojem se opisuju različite vrste informacija koje možete naći na Internetu.
- 33. poglavlje "Uporaba elektroničke pošte", u kojem pokazujemo kako komunicirati s drugim ljudima uporabom sustava e-pošte.

# P 0 G L A V L J E **35**

## Početak u Apacheu

(Steve Burnett)

## U ovom poglavlju

Kompiliranje Apachea Utemeljivanje hijerarhije datoteka Osnovne prilagodbe Pokretanje Apachea Provjera postupka pokretanja poslužitelja Postavljanje Apache-SSL Želite li uporabiti Linux sustav kao Web poslužitelj, u svoj sustav morate instalirati poseban softver za poslužitelj. Najpopularniji UNIX-ovi paketi Web poslužitelja su Apache i NCSA. Anketa u lipnju 1998. pokazala je da Apache pokriva više od 53 posto instaliranih Web poslužitelja. Ovo se poglavlje odnosi na Apache poslužitelj, ali rječnik se odnosi i na druge Web poslužitelje. NCSA obitelj poslužitelja ima puno zajedničkog u konfiguracijskim datotekama zato što je Apache potekao od NCSA 3.1 poslužitelja, a trebao je zadržati uskladivost s NCSA poslužiteljima.

U ovom poglavlju govori se o osnovnim postupcima za instalaciju softvera i poslužitelja. Ako ste već instalirali Apache ili NCSA poslužitelje, možete ovo poglavlje preskočiti, iako biste mogli letimice pogledati osnovne razlike.

## Kompiliranje Apachea

Apache je dostupan za sve UNIX-e: Solaris 2.X, SunOS 4.1.X, Irix 5.X i 6.X, Linux, FreeBSD/NetBSD/BSDI, HP-UX, AIX, Ultrix, OSF1, NeXT, Sequent, A/UX, SCO, UTS, ApolloDomain/OS, QNX i neke druge. Napravljen je i *port* za OS/2, te beta inačica za Windows NT 4.0. Prenosivost je izuzetno važna.

Apache binarni programi su uključeni u većinu Linuxovih distribucija. Osiguran je i izvorni kôd za Apache. Apache binarni programi su na CD-ROM-ovima pa možete preskočiti postupak kompiliranja ako ste u žurbi. No, ako želite dodati nove module ili ugađati funkcional-nost, morate ih znati kompilirati.

Kopirajte izvorni kôd u dio vašeg datotečnog sustava. Za kompiliranje poslužitelja trebat će vam nekoliko megabajta prostora na disku. Raspakirajte ga i otiđite na /src poddirektorij. Dobit ćete niz naredbi:

```
cd /CDROM
cp apache_1.3.0.tar.gz /usr/local/apache/
cd /usr/local/apache/
tar -zxvf apache_1.3.0.tar
CD src
```

## 1. korak: uređivanje konfiguracijske datoteke

Konfiguracijska datoteka rabi program Configure kako bi napravio Makefile namijenjen isključivo vašoj platformi s modulima koje ste vi izabrali. On također stvara modules.c koji ima informacije koje module treba povezati u vrijeme kompiliranja.

Morate deklarirati koji C kompilator rabite (najvjerojatnije gcc) i morate ukloniti oznaku komentara te promijeniti odgovarajuću postavku za AUX\_CFLAGS. U Makefile potražite taj unos koji može izgledati ovako:

```
CC=gcc
AUX_CFLAGS = -DLINUX
```

**BILJEŠKA** Za definiciju CFLAG-a, ako želite da se na poslužiteljskoj strani dekodiraju informacije za svaku datoteku s postavljenim "izvršnim" bitom, postavite -DXBITHACK.

Želite li biti sigurniji u ime računala-domaćina, možete postaviti -DMAXIMAL\_DNS. To ćete postaviti ako štitite dijelove svoje lokacije koji se temelje na imenu računala-domaćina. Ovo nije obvezatno, a osigurano je uglavnom za uskladivost s ranijim inačicama NCSA 3.1.

Na dnu datoteke je popis pakiranih modula koji se isporučuju s Apacheom. Svi oni nisu podrazumijevano kompilirani u konačni program. Želite li ugraditi modul, uklonite oznaku komentara za njega. Neki se moduli uzajamno isključuju. Na primjer, nije dobro istovremeno kompilirati podesivi prijavni modul i zajednički prijavni modul.

Neki moduli, kao što je mod\_auth\_dbm, zahtijevaju vanjsku biblioteku i unos koji se dodaje EXTRA\_LIBS retku. Kasnije ćete naučiti više o modulima, za početak uporabite podrazumijevane module.

#### 2. korak: Izvođenje konfiguracijske skripte

Konfiguracijska skripta je jednostavna *Bourne shell* skripta koja uzima konfiguracijsku datoteku i iz nje pravi Makefile te modules.c.

#### 3. korak: Izvođenje naredbe make

Naredba make kompilira poslužitelj. Možete se susresti s upozorenjima o vrstama podataka, posebno ako ste kompilirali s prekidačem -Wall, ali pogreške ne bi trebale biti fatalne.

Ako je sve bilo dobro, u src/ direktoriju imat ćete izvršni program po imenom httpd.

## Utemeljivanje hijerarhije datoteka

Slijedeći korak u postavljanju poslužitelja je donošenje nekih temeljnih odluka u svezi smještaja različitih dijelova poslužitelja na datotečni sustav. Zabilježite svoja rješenja, trebat ćete ih u slijedećem odjeljku "Prikazivanje osnovnog oblikovanja".

Najprije morate odlučiti gdje će biti *server root (korijenski direktorij poslužitelja)*. To je poddirektorij gdje se nalazi poslužitelj i koji vodi do conf/ direktorija, logs/ i CGI-bin/ poddirektorija, te drugih direktorija koji su povezani s poslužiteljem. Podrazumijevani direktorij je as/usr/local/Apache, iako je obično /pub/htdocs. Svoje konfiguracijske datoteke i datoteke prijave možete imati na drugim mjestima. *Server root* je prikladno mjesto za čuvanje svega što je vezano za poslužitelj. Ako se poslužitelj sruši i ostavi datoteku jezgre, ona će biti u ovom direktoriju.

Potom morate odlučiti gdje ćete smjestiti *korijenski direktorij dokumenta*, direktorij gdje su spremljeni svi HTML-ovi. Datoteka myfile.html u korijenskom direktoriju dokumenta zvat će se http://host.com/index.html. Taj direktorij može biti poddirektorij *server roota* i zove se htdocs, ali može biti i samostalan u svom direktoriju. Ako maknete korijenski direktorij dokumenta iz korijenskog direktorija poslužitelja, morate mu dati kratko ime – npr. /home/www ili /www/htdocs. Ako, recimo, stavljate Web poslužitelj na vrh FTP poslužitelja, možete pokazati na korijenski direktorij dokumenta u /home/FTP/pub.

Na kraju morate odlučiti gdje ćete držati datoteke s podacima o prijavi. Taj prostor mora imati prilično veliko radno područje. Na primjer, lokacija sa 100K pristupa dnevno (što je umjereni promet) imat će dnevno 15MB informacija u datotekama s podacima o prijavama. Obično je direktorij s podacima o prijavi najbolje imati na odvojenoj particiji diska ili čak na drugom disku.

## **Osnovne prilagodbe**

U ovom odjeljku govorimo o osnovnim promjenama konfiguracijskih datoteka koje su potrebne za pokretanje jednostavne Web lokacije.

Apache ima tri odvojene konfiguracijske datoteke. Tako je i NCSA-u, a razlog je slijedeći: postoje tri glavna područja upraviteljskog prilagođenja, pa njihovo instaliranje u odvojene datoteke omogućuje Webmasterima davanje različitih dopuštenja za pisanje.

Konfiguracijske datoteke za Apache naći ćete u poddirektoriju conf/ korijenskog direktorija poslužitelja. Svaka ima dometak imenu datoteke -dist; preporučuje se napraviti kopiju bez -dist te urediti te nove datoteke, a inačicu -dist čuvati kao sigurnosnu kopiju i referencu.

Osnovni format konfiguracijskih datoteka je sučelje nalik ljuski i pseudo-HTML-u. Osnovna jedinica je uputa koja može poprimiti brojne argumente:

```
Uputaargument argument....
to jest
Port 80
ili
AddIcon /icons/back.gif ..
```

Upute možete grupirati unutar određenih pseudo-HTML *tagova*. Ti *tagovi* moraju biti u svom retku, kao u primjeru:

```
<Virtualhost www.myhost.com>
documentRoot /www/htdocs/myhost.com
ServerName www.myhost.com
</Virtualhost>
```

#### BILJEŠKA Virtualhost uputa omogućuje poslužitelju da se pretvara kao da je višestruki poslužitelj. Na primjer, računalo-domaćin www.myhost.com ne mora biti na računalu www.myhost.com; može biti na računalu hosts.netwharf.com.

#### httpd.conf

Prva konfiguracijska datoteka http.conf je datoteka koja postavlja osnovne informacije na razini sustava o poslužitelju, npr. s kojim je *portom* povezan, koji korisnici ga pokreću itd. Ako niste administrator sustava lokacije na koju instalirate poslužitelj, potražite pomoć od administratora.

Osnovne su stavke ove datoteke:

port number

Na primjer:

port 80

To je TCP/IP broj *porta* s kojim je poslužitelj povezan. *Port* 80 je podrazumijevani *port* u http: URL-ovima. Drugim riječima, http://www.myhost.com/ ekvivalentan je http://www.myhost.com:80/.

Želite li pokrenuti poslužitelj s drugog *porta*; na primjer, možda se na *portu* 80 već izvodi neki poslužitelj ili taj poslužitelj želite zadržati tajnim. (U tom slučaju trebali biste rabiti pristup koji nadzire računalo-domaćin ili zaštitu lozinkom.)

User #number\_or\_uid

```
Group #number_or_uid
Na primjer:
User nobody
Group nogroup
```

Apache se treba pokrenuti kao root kako bi se povezao s *portom* manjim od 1024. Čim se dočepa *porta*, Apache mijenja svoj korisnički ID u drugi, obično korisnik nobody. To je jako važno zbog sigurnosnih razloga.

Taj korisnički ID treba čitati datoteke u korijenskom direktoriju dokumenta i mora imati dozvolu samo za čitanje konfiguracijskih datoteka. Argument može biti pravo ime korisnika; međutim, želite li imati brojčani ID, postavite znak # prije broja. Upute za grupe slijede isto načelo: odredite skupni ID koji pokreće poslužitelj.

#### 676 35. poglavlje Početak u Apacheu

- BILJEŠKA Pokretanje Web poslužitelja kao roota znači da svaku sigurnosnu rupu na poslužitelju (na samom poslužitelju ili u CGI skripti) može iskoristiti vanjski korisnik za izvođenje naredbe na vašem stroju. Zbog toga je postavka korisnika nobody ili www najsigurnija.
  - ServerAdmin email\_address

Postavlja adresu e-pošte korisnika koji može primati poštu. U slučaju pogreške na poslužitelju, pretraživač koji posjećuje vašu lokaciju dobit će poruku "please report this problem to user@myhost.com". U budućnosti će Apache moći poslati e-poštu ServerAdmin korisniku ako dođe do većih problema na poslužitelju.

ServerRoot directory

Na primjer:

ServerRoot /usr/local/Apache

Postavlja korijenski direktorij poslužitelja koji ste prije odredili. Dajte punu stazu i nemojte završiti kosom crtom.

ErrorLog directory/filename

```
TransferLog directory/filemname
```

Određuje gdje prijaviti pogreške i pristupe na Web. Ako ime datoteke koje ste vi dali ne počinje kosom crtom, pretpostavlja se da se odnosi na korijenski direktorij poslužitelja. Prije sam preporučio da se datoteke s podacima o prijavi šalju kao odvojeni direktorij izvan korijenskog direktorija; tu ćete odrediti direktorij s podacima o prijavi i ime datoteka s podacima o prijavi.

```
ServerName DNS_hostname
```

Povremeno će Web poslužitelju trebati ime računala-domaćina koje poziva, a koje se može razlikovati od pravog imena računala-domaćina. Na primjer, ime www.myhost.com može biti DNS *alias* za gateway.myhost.com. Tada ne želite da URL-ovi koje proizvodi poslužitelj budu http://gateway.myhost.com/. ServerName vam omogućuje da to precizno odredite.

#### srm.conf

Druga konfiguracijska datoteka je srm.conf. U toj su datoteci važne postavke:

```
DocumentRoot directory
```

Kao što je prije opisano, to je početna razina stabla vaših dokumenata koja može biti ili /usr/local/Apache/htdocs ili /www/htdosc. Taj direktorij mora postojati, a korisnik koji pokreće Web poslužitelj (obično nobody) mora ga biti u stanju pročitati.

```
ScriptAlias reguest_path_alias directory
```

ScriptAlias omogućuje vam odrediti da direktoriju izvan korijenskog direktorija

dokumenta bude promijenjeno ime na zahtjev *i* da objekti u tom direktoriju budu izvršeni, a ne samo pročitani iz datotečnog sustava. Na primjer, zadano

```
ScriptAlias /CGI-bin/ /usr/local/apache/CGI-bin/
```

znači da će zahtjev za http://www.myhost.com/CGI-bin/fortune izvršiti program /usr/local/Apache/CGI-bin/fortune. Apache se isporučuje s brojnim CGI skriptima koje su korisne početnicima, jednostavnim skriptama ljuski koje ilustriraju CGI programiranje.

Konačno, direktorij koji ima CGI skripte ne smije biti pod korijenskim direktorijem dokumenta.

#### access.conf

access.conf ustrojena je strože od ostalih konfiguracijskih datoteka; sadržaj se nalazi unutar <Directory></Directory> pseudo-HTML *tagova* koji definiraju opseg direktorija unutar njih.

Vidi "Osnove prilagođavanja", str. 684

Na primjer, upute između

```
<Directory /www/htdocs>
```

i

</Directory>

utječu na sve unutar /www/htdocs direktorija. Mogu se rabiti i zamjenski znakovi. Na primjer:

```
<Directory /www/htdocs/*/archives/>
```

```
</Directory>
```

primjenjuje se na /www/htdocs/list1/archives/, /www/htdocs/list2/archives/ i tako dalje.

## Pokretanje Apachea

Za pokretanje Apachea izvedite binarni program koji ste prije kompilirali s -f zastavicom koja pokazuje na httpd.cong datoteku, kao u primjeru:

/usr/local/Apache/src/httpd -f /usr/local/Apache/conf/httpd.conf

Slijedeća pokazna inicijalizacijska skripta aktivira i Apache Web poslužitelj. Red Hat distribucija automatski instalira tu skriptu ako ste označili instaliranje Web poslužitelja.

```
#!/bin/sh
#
# Startup script for the Apache Web server
```

```
#
# chkconfig: 345 85 15
# description: Apache is a World Wide Web server. It is used to serve \setminus
# HTML files and CGI.
#
#
# Source function library.
. /etc/rc.d/init.d/functions
# See how we were called.
Case "$1" in
   start)
         echo -n "Starting httpd: "
         daemon httpd
         echo
          touch /var/lock/subsys/httpd
          ; ;
   stop)
         echo -n "Shutting down http: "
         kill 'cat /var/run/httpd.pid'
         echo httpd
         rm -f /var/lock/subsys/httpd
         rm -f /var/run/httpd.pid
          ; ;
   status)
         status httpd
         ; ;
   restart)
         $0 stop
         $0 start
          ; ;
   *)
         echo "Usage: httpd.init {start|stop|restart|status}"
         exit 1
esac
exit 0
```

Sada uporabite naredbu ps kako biste vidjeli izvodi li se httpd. Obično je dovoljno napisati ps –aux | grep. Vidjet ćete istovremeno izvođenje brojnih httpd postupaka. Što se zbiva?

Prvi Web poslužitelji, kao CERN i NCSA, oblikovali su jedan od glavnih Web poslužitelja klonirajući se sa svakim zahtjevom koji je stigao. Klon bi odgovorio na zahtjev, a prvobitni poslužitelj bi čekao novi zahtjev s *porta*. Iako jednostavno, *kloniranje* (u UNIX-u *forking*) bila je skupa operacija pod UNIX-om zbog dugog trajanja prijave. Kada je bilo puno klonova, poslužitelju je bilo teško prepoznati koliko ih je aktivno. Tako poslužitelji nisu lako mogli odbiti ili odgoditi veze kojima su nedostajali resursi. Apache – kao i NCSA 1.4+, Netscapeovi Web poslužitelji i još neki drugi Web poslužitelji koji se temelje na UNIX-u, umjesto toga rabe model grupe potomaka (djece) koji se paralelno izvode. Potomke usklađuje roditeljski proces, koji može reći koliko je potomaka živo, kreirati nove potomke, te zaustaviti stare potomke ako postoji puno potomaka u mirovanju. (*Roditelj* i *potomak ili dijete* su izrazi iz UNIX-a.)

Otvorite Web pretraživač i usmjerite ga na lokalni poslužitelj. (Uporabite uobičajeni http:// format i dodajte ServerName parametar koji ste definirali u datoteci httpd.conf.) Radi li? Ako je sve prošlo dobro, trebali biste vidjeti popis kazala direktorija svega u korijenskom direktoriju dokumenta ili ako je u tom direktoriju index.html, vidjet ćete sadržaj te datoteke.

U tablici 35.1 su druge opcije naredbenog retka.

Tablica 35.1 Opcije naredbenog retka za httpd	
Opcija	Rezultat
-d serverroot	Postavlja početnu vrijednost za ServerRoot.
- X	Pokreće poslužitelj u jednoprocesnom načinu rada.
	(To je korisno za postupak provjere, ali nemojte
	u tom načinu pokretati poslužitelj kada radi
	s vanjskim svijetom.)
- V	Ispisuje inačicu poslužitelja, a potom izlazi.
- ?	Ispisuje dostupne argumente naredbenog retka u Apache.

Kada ste provjerili da se Apache ispravno pokreće, možete u skripte za pokretanje sustava dodati naredbu za pokretanje tako da se Apache automatski pokreće kada se pokreće sustav. Obično se naredba za pokretanje smješta u datoteku /etc/rc.d/rc.local.

## Provjera postupka pokretanja poslužitelja

Apache daje dobre poruke o pogreškama, ali neke ćemo sada još detaljnije opisati.

#### Poruke o pogreškama pri otvaranju datoteka

```
httpd: could not open document config file ....
fopen: No such file or directory
```

Te poruke o pogreškama pri otvaranju datoteka obično su posljedica dodjeljivanje relativne staze argumentu -f tako da Apache traži datoteku koja ovisi o kompilaciji poslužitelja (postavljeno u src/httpd.h), umjesto onih koje ovise o direktoriju u kojem jeste. Morate zadati punu stazu.

#### Poruke o pogreškama portova

```
httpd: could not bind to port [X]
bind: Operation not permitted
```

Poruke o pogreškama *portova* najvjerojatnije su uzrokovane pokušajem izvođenja poslužitelja na *portu* broja manjeg od 1024 bez njegovog pokretanja kao "korijenskog". Većina UNIX-ovih operativnih sustava, uključujući i Linux, sprečava one koji nemaju pristup korijenskom direktoriju pokrenuti bilo koju vrstu poslužitelja na *portu* broja manjeg od 1024. Ako pokrenete poslužitelj kao korijenski, poruka o pogreški će nestati.

httpd: could not bind to port bind: Address already in use

Ove poruke o pogreškama *portova* znače da se na vašem stroju već nešto izvodi na *portu* koji ste odredili. Radi li tu neki drugi poslužitelj? Ni jedan standardni UNIX mehanizam ne može odrediti što se izvodi na kojem *portu*; na većini sustava datoteka /etc/services može vam reći koji su najčešći problemi, ali ne daje potpuni popis. Isprobajte i naredbu netstart s različitim opcijama kao što su -a.

### Poruke o krivo prijavljenim korisnicima i grupama

```
httpd: bad user name ....
httpd: bad group name ....
```

Poruke o krivo prijavljenim korisnicima i grupama znače da korisnik ili grupa koju ste odredili u httpd.conf ne postoji u vašem sustavu. Možete vidjeti pogreške koje vam govore da određene datoteke ili direktoriji ne postoje. Ako izgleda da su datoteke tamo, uvjerite se da ih mogu pročitati root i nobody (kako već poslužitelj radi).

## Poruka o pogrešci pri pokretanju poslužitelja

Pretpostavimo da se Apache pokrenuo i prema ps, on se izvodi. Međutim, kada dođete na lokaciju, susrećete se s ovim problemima:

- No connection at all. Uvjerite se da između vas i poslužitelja nema zaštitnih mehanizama (*firewall*) koji će filtrirati pakete za poslužitelj. Drugo, pokušajte uporabiti telnet na *portu* s kojeg ste pokrenuli Web poslužitelj na primjer, telnet myhost.com 80. Ako ne dobijete poruku Connected to myhost.com, niste se uspjeli povezati ni s poslužiteljem.
- 403 Access Forbidden. Vaš korijenski direktorij dokumenta se ne može pročitati ili možda u datoteci access.conf imate nešto što sprečava pristup lokaciji s računala gdje je vaš Web pretraživač.
- 500 Server Error. Je li vaša početna stranica CGI skripta? Skripta je možda pogrešna.

To su najčešće pogreške pri početnom pokretanju poslužitelja. Ako ste sigurni da je veza s poslužiteljem uspostavljena, informacije o pogreškama možete potražiti u ErrorLog.

## Postavljanje Apache-SSL

Na ovom mjestu pogledat ćemo varijantu Apache Web poslužitelja, Apache-SSL, koji može provesti sigurne transakcije preko Secure Sockets Layer protokola. SSL je RSA protokol šifriranja zasnovan na javnim ključevima koji je razvio Netscape Communications za uporabu u Netscape Navigator pretraživaču i na Netscape Web poslužiteljima.

Do nedavno, jedina mogućnost obavljanja SSL transakcija na World Wide Webu bila je uporaba patentiranog poslužitelja kao što su Netscape Commerce ili OpenMarket Secure. Jače šifrirane inačice tih poslužitelja nisu bile dostupne izvan SAD-a zbog izvoznih ograničenja.

Eric Young, autor libdes paketa, s Timom Hudsonom je napisao biblioteku koja implementira SSL, nazvanu SSLeay. SSLeay paket narastao je u tajno pismo i nadzornika certifikata kao biblioteka za sve namjene, a ime je ostalo isto.

Ben Laurie, član Apache Group, uzeo je SSLeay biblioteku i povezao je s Apache poslužiteljem. Svoje zakrpe postavio je na Mrežu. Sameer Parekh iz Community ConneXion, Inc. (ovdje se zove C2), potom je uzeo zakrpe Bena Lauriea i napravio paket koji se može rabiti u SAD-u.

Kako je RSA tehnologija koju rabi SSL u SAD zaštićena patentom koji posjeduje RSA Data Security, Inc. (RSADSI - **www.rsa.com**), SSLeay paket se ne može upotrebljavati proizvoljno. C2 je licencirao RSA tehnologiju kako bi omogućio zakonitu uporabu paketa u SAD-u uporabom "RSAREF" paketa koji proizvodi RSADSI and Consensus Development Corporation (**www.con-sensus.com**).

Zbog izvoznih ograničenja, nitko izvan SAD-a ne može zakonito učitati i instalirati C2 Apache-SSL paket.

Želite li doznati više o SSL-u i Apacheu, otiđite na URL http://www.Apache-SSL.org.

## Odavde...

O postavljanju, prilagodbi i izvođenju Apache Web poslužitelja naučit ćete više u ovim poglavljima:

- 36. Poglavlje "Prilagodba Apachea", gdje se detaljno govori o mogućnostima prilagodbe Apachea.
- 37. Poglavlje "Upravljanje Internet Web poslužiteljem", gdje učite kako svoj poslužitelj učiniti snažnim, uspješnim, automatiziranim i sigurnim.

# poglavlje36

## **Prilagodba Apachea**

(Steve Burnett)

U ovom poglavlju

Osnovne postavke **684** Posebni moduli **693** Napredno funkcioniranje **702** 

## Osnovne postavke

Sada već sigurno izvodite Web poslužitelj, no on je minimalno podešen. U ovom poglavlju detaljnije ćete naučiti o funkcionalnosti poslužitelja. Ovo poglavlje je napravljeno kao niz vodiča tako da, ako ste novi korisnik Apachea, možete sve brzo razumjeti. Na kraju poglavlja imate uvid u neke eksperimentalne Apache module.

Zbog velike brzine razvoja, Apache će vjerojatno u vrijeme vašeg čitanja ove knjige, već imati nove funkcije. Međutim, postojeće funkcije se neće mijenjati, što znači da se zadržava uskladivost s prijašnjim inačicama.

U datotekama srm.conf i access.conf se nalazi većina postavki povezana sa stvarnim objektima na poslužitelju. Datoteka srm.conf se još zove ResourceConfig, što je instrukcija koja se može postaviti u httpd.conf; datoteka access.conf zove se i AccessConfig, što je isto instrukcija u httpd.conf.

Imena *srm.conf* i *access.conf* su povijesna. Kada je poslužitelj bio samo NCSA, datoteka access.conf rabila se samo za postavljanje dozvola, ograničenja, provjere autentičnosti itd. Kada je dodano indeksiranje direktorija, javila se potreba za nadzorom određenih karakteristika na temelju izravnih veza među direktorijima. Datoteka access.conf bila je jedina konfiguracijska datoteka koja je imala neku vrstu ustroja za uski nadzor pristupa: pseudo-HTML <Directory> spremnik.

#### Vidi "access.conf", str. 677

S Apacheovim rutinama za dekodiranje, većina instrukcija može se doslovce pojaviti bilo gdje – na primjer, unutar <Directory> spremnika u access.conf, u <VirtualHost> spremnicima u httpd.conf, itd. Trebali biste sačuvati neke ustroje konfiguracijskih datoteka. Opcije prilagođenja za poslužiteljsku procesnu razinu (kao što su Port i <VirtualHost>) trebali biste staviti u httpd.conf, opće informacije o resursima poslužitelja (kao što su Redirect i AddType) u srm.conf, a ostalo u access.conf.

U dodatku <Directory> spremnika je <Limit> spremnik koji se u <Directory> spremnicima rabi za određivanje nekih HTTP metoda na koje se primjenjuju pojedine instrukcije. Primjeri su dani kasnije u ovom poglavlju.

## **BILJEŠKA** lako je ovo poglavlje korisno, verzija 1.3.0 i kasnije verzije Apachea uključuju GNU klijenta za automatsku prilagodbu koji podupire sve prethodne mogućnosti podešavanja, kao i poboljšanja u 1.3.0 i više.

#### per-directory konfiguracijske datoteke

Prije proučavanja različitih mogućnosti oblikovanja, pogledajmo mehanizam koji nadzire postavke na temelju izravnih veza među direktorijima. To se postiže uporabom konfiguracijske datoteke koja je lokalna u odnosu na direktorij koji želite podesiti. Opcije poddirektorija možete nadzirati u access.conf, kao što je to rečeno u 35. poglavlju, "Početak u Apacheu". Međutim, možete omogućiti da te postavke održavaju korisnici, oni koji nemaju mogućnost ponovnog pokretanja poslužitelja (korisnici koji imaju svoje početne stranice). U tu svrhu napravljena je instrukcija AccessFileName.

```
Vidi "Osnovne prilagodbe", str. 674
```

Podrazumijevano, AccessFileName je .htaccess. Ako želite uporabiti nešto drugo – na primjer .acc, u srm.conf morate napisati:

```
AccessFileName .acc
```

Ako je omogućeno traženje datoteke AccessFileName, a dođe zahtjev koji prevodi u datoteku /www/htdocs/path/path2/file, poslužitelj će potražiti /.acc, /www/.acc, /www/htdocs/.acc, /www/htdocs/path/.acc i /www/htdocs/path/path2/.acc, tim redoslijedom. Ako nađe datoteku, poslužitelj će je analizirati kako bi odredio koje postavke primijeniti. (To analiziranje se treba dogoditi kod svakog pristupa, odvojeno tako da može biti puno pristupa.) Instrukciju AccessFileName možete isključiti postavljanjem sljedeće opcije u konfiguracijskoj datoteci pristupa:

<Directory />
AllowOverride None
</Directory>

Zbog kratkoće i jasnoće pretpostavimo da je opcija AccessFileName postavila ime tih datoteka kao .htaccess. Na što mogu utjecati te opcije? Uputa AllowOverride kontrolira područje dostupnih opcija unutar <Directory> spremnika u AccessConfig datoteci, kao što je već prije rečeno. U tablici 36.1 je popis točnih argumenata za AllowOverride.

Tablica 36.1	AllowOverride argumenti
Argument	Posljedica
AutoConfig	Kada se navede, .htaccess datoteke mogu odrediti svoje upute za prijavu, kao što su AuthUserFile, AuthName, AuthType i require.
FileInfo	Kada se navede, .htaccess može zaobići sve postavke za metainformacije o datotekama uporabom instrukcija kao što su AddType, AddEncoding i AddLanguage.
Indexes	Kada se navede, .htaccess datoteke mogu lokalno postaviti upute koje nadziru indeksiranje direktorija, kao što je provedeno u mod_dir.c modulu – na primjer, FancyIndexing, AddIcon i AddDescription.
Limit	Argument dozvoljava uporabu instrukcija koje ograničavaju pristup na temelju imena računala-domaćina ili IP adrese poslužitelja-domaćina (allow, deny i order).
Options	Ovaj argument dozvoljava uporabu instrukcije Options.
A11	Ovaj argument dozvoljava upotrebu svih prethodnih argumenata.

AllowOverride opcije nisu spojene, što znači da ako se postavka za /path/ razlikuje od postavke za /, /path/ će imati prvenstvo jer je dublja.

#### MIME vrste: AddType i AddEncoding

Osnovni element HTTP protokola i razlog zašto je Web tako prirodan domaćin različitih medijskih formata je da se svakom objektu koji se prenosi putem HTTP-a pridružuje odgovarajući MIME tip.

BILJEŠKA MIME (Multipurpose Internet Mail Extensions) vodi porijeklo iz napora da se standardizira prijenos dokumenata višestrukih medija putem e-pošte. Dio MIME specifikacije je da e-pošta ne smije imati metainformacije u zaglavljima kojima se identificiraju poslani podaci. Jedna vrsta MIME zaglavlja je Content-Type, koja navodi format ili vrstu podataka u kojem j e objekt. Na primjer, HTML ima oznaku "text/html", a JPEG slike imaju oznaku "image/jpeg".



#### NA WEBU

Registar MIME tipova održava Internet Assigned Numbers Authority na lokaciji:

#### http://www.isi.edu/div7/iana/

Kada pretraživač pita poslužitelj za neki objekt, poslužitelj daje taj objekt pretraživaču i izjavljuje koji je njegov Content-Type. Tako pretraživač može odlučiti kako obraditi dokument. Na primjer, on ga može poslati programu za slike, PostScript pregledniku ili VRML pregledniku.

To znači da svaki objekt koji se isporučuje mora imati pridruženu ispravan MIME vrstu. Srećom, postoji dogovor da se vrste podataka izražavaju nastavcima imenu datoteke od dva, tri ili četiri slova - na primjer, foobar.gif je najvjerojatnije GIF slika.

Poslužitelju treba datoteka kako bi unio nastavak u MIME content type. Srećom, Apache se isporučuje s takvom datotekom u konfiguracijskom direktoriju - to je mime.types. Jednostavni format te datoteke sastoji se od jednog zapisa po retku, gdje je zapis MIME tip, te popis prihvatljivih nastavaka. Iako se u određeni MIME tip može unijeti više od jednog nastavka imena, ne možete imati više od jednog MIME tipa po nastavku. Za određivanje zamjenske lokacije datoteke možete uporabiti instrukciju TypesConfig.

Internet se tako brzo razvija da je teško imati potpuno ažurirane mime.types datoteke. Taj problem možete riješiti tako da u srm.conf datoteku upišete posebne, AddType instrukcije:

AddType x-world/x-vrml wrl

Kad god se od poslužitelja traži dap obradi datoteku koja završava s .wrl, on također zna da mora poslati zaglavlje slično ovome:

```
Content-type: x-world/x-vrml
```

Tako ne trebate brinuti hoće li buduće mime.type datoteke biti u skladu s postavkama i instalacijom vašeg računala.

Kao što ćete vidjeti kasnije, AddType se također rabi za određivanje "posebnih" datoteka kojima se upravlja pomoću određenih osobina unutar poslužitelja.

Sestra instrukcije AddType je AddEncoding. Kao što MIME zaglavlje Content-type može odrediti format podataka objekta, Content-Encoding zaglavlje određuje *kodiranje* objekta. Kodiranje je atribut objekta dok se on prenosi ili sprema; pretraživač treba znati da mora "dekodirati" sve što dobije, na temelju navedenog kodiranja. Najčešća uporaba je kod komprimiranih datoteka. Na primjer, ako imate

```
AddEncoding x-gzip gz
```

i ako potom pristupite datoteci myworld.wrl.gz, MIME zaglavlja poslana u odgovoru izgledat će ovako (MIME zaglavlja ovog formata prate svaki prijenos na Webu; ta zaglavlja pretraživač ne prikazuje, ali se rabe za određivanje kako rukovati datotekom koju prime):

```
Content-Type: x-world/x-vrml
Content-Encoding: x-gzip
```

Svaki će pretraživač znati da mora raspakirati datoteku prije obrađivanja na VRML pregledniku.

BILJEŠKA U Apacheu 1.3 neobvezni modul mod\_mime\_magic (ako je uključen) može analizirati sadržaj datoteke i dodijeliti joj bilo koji nastavak ako ga ona nema.

#### Alias, ScriptAlias i Redirect

Alias, ScriptAlias i Redirect instrukcije – koje se nalaze u srm.conf, a izvršava ih mod\_alias.c modul - omogućuju vam određenu prilagodljivost u preslikavanju između URL-prostora na vašem poslužitelju i stvarnog izgleda datotečnog sustava.

U osnovi svaki URL koji izgleda kao http://myhost.com/x/y/z ne zahtijeva preslikavanje u datoteku nazvanu x/y/z u korijenskom direktoriju poslužitelja, ponašajući se kao simbolična veza. Na primjer:

```
Alias /path/ /some/other/path/
```

Prethodne instrukcije primaju zahtjev za objektom iz zamišljenog poddirektorija /path pod korijenskim direktorijem dokumenta i preslikavaju ga u neki drugi direktorij. Na primjer, zahtjev za

```
http://myhost.com/statistics/
```

normalno će ići u korijenski direktorij dokumenta /statistics, osim kada vi želite da pokazuje negdje izvan korijenskog direktorija dokumenta (na primjer, /usr/local/statistics). Za to trebate uporabiti slijedeću naredbu:

```
Alias /statistics/ /usr/local/statistics/
```
Za vanjskog korisnika to će biti skroz nevidljivo. Ako uporabite Alias, nemojte *alias* negdje unutar korijenskog direktorija dokumenta. Nadalje, zahtjev poput

```
http://myhost.com/statistics/graph.gif
```

prevest će se u zahtjev za datotekom

```
/usr/local/statistics/graph.gif
```

ScriptAlias je ista kao i Alias, s time da sve u poddirektoriju podrazumijevano pretvara u CGI skripte. To se može činiti čudnim, ali prije su modeli za izradu Web lokacija imali sve CGI funkcije odvojene u poseban direktorij i pozivali ih putem Web poslužitelja ovako:

```
http://myhost.com/cgi-bin/script
```

Ako u srm.conf datoteci imate

ScriptAlias /cgi-bin/ /usr/local/etc/httpd/CGI-bin/

Onda prethodni URL pokazuje na skriptu na /usr/local/etc/httpd/CGI-bin/script. Kao što ćete uskoro vidjeti, postoji elegantniji način određivanja da je datoteka CGI skripta koju treba izvršiti.

Redirect preusmjerava zahtjev do drugog resursa. Taj resurs može biti na istom stroju ili negdje drugdje na Mreži. Također, analiziraju se i obrađuju i podnizovi, počevši od početka. Na primjer, ako ste unijeli naredbu

Redirect /network http://myhost.com/maps/states/newyork

onda će zahtjev za

http://myhost.com/newyork/index.html

biti preusmjeren na

http://myhost.com/maps/states/newyork/index.html

Naravno, drugi argument za Redirect može biti URL na nekoj drugoj lokaciji. Samo budite sigurni da znate što radite.

#### UPOZORENJE

Čuvajte se slučajne izrade petlji. Na primjer, Redirect /newyork http://myhost.com/newyork/newyork može biti prilično pogubna za poslužitelj.

# Bolji način aktiviranja CGI skripti

Postoji elegantniji način aktiviranja CGI skripti od uporabe ScriptAlias. Možete uporabiti AddType i napraviti korisnički MIME tip, poput ovog:

```
AddType application/x-httpd-cgi cgi
```

Kada poslužitelj dobije zahtjev za CGI datotekom, poslužitelj odredi MIME tip i zna da mora to izvršiti, a ne poslužiti kao obične datoteke. Tako CGI datoteke možete imati u istim direktorijima kao i HTML, GIF i druge datoteke.

#### Indeksiranje direktorija

Kada se Apacheu daje URL direktorija umjesto određene datoteke, kao u ovom primjeru:

```
http://myhost.com/statistics/
```

Apache najprije traži datoteku koji je odredila instrukcija DirectoryIndex u srm.conf. Za osnovne je postavke to datoteka index.html. Možete postaviti popis datoteka koje treba potražiti ili apsolutnu stazu do stranice ili CGI skripte:

DirectoryIndex index.cgi index.HTML /CGI-bin/go-away

Prethodna instrukcija znači da se najprije treba potražiti index.cgi datoteka. Ako se ne može naći, onda treba potražiti index.html datoteku. Ako se ni ona ne može naći, preusmjerava se zahtjev u /CGI-bin/go-away.

Ako Apache poslužitelj ne može naći ništa odgovarajuće, napravit će – potpuno dinamički- HTML popis svih datoteka dostupnih u direktoriju.

Postoji više načina za prilagođenje prikaza indeksiranja direktorija. Najprije trebate odlučiti zanimaju li vas stvari poput ikona i vrijeme posljednje modifikacije. Ako da, onda ćete uključiti

FancyIndexing On

Inače ćete dobiti jednostavni izbornik dostupnih datoteka koji je prikladan iz sigurnosnih ili izvedbenih razloga.

Ako uporabite FancyIndexing opciju, morate pitati trebate li je dalje prilagoditi, a ako da, kako. Podrazumijevane postavke funkcija indeksiranja direktorija već su dobro objašnjene.

Upute AddIcon, AddIconByEncoding i AddIconByType prilagođavaju izbor ikona za datoteke. AddIcon slaže ikone prema imenu prema šabloni

AddIcon iconfile filename [filename] [filename] ...

Na primjer, sljedeći redak

```
AddIcon /icons/binary.GIF .bin .exe
```

znači da svakoj datoteci koja završava na .bin ili .exe treba pridružiti ikonu binary.gif. Imena datoteka mogu biti i zamjenski znakovi, potpuno ime ili jedno od dva "posebna" imena: ^^DIRECTORY^^ za direktorije i ^^BLANKICON^^ za prazne retke.

Vidjet ćete ovakve retke:

```
AddIcon /icons/dir.gif ^^DIRECTORY^^
AddIcon /icons/old.gif *~
```

Konačno, *iconfile* može biti niz koji ima ime datoteke ikone i zamjenski tekst koji treba staviti u ALT atribut. Primjer bi trebao izgledati ovako:

```
AddIcon (BIN,/icons/binary.gif).bin .exe
AddIcon (DIR,/icons/dir.gif ^^DIRECTORY^^
```

Uputa AddIconByType je nešto fleksibilnija i vjerojatno se više preporučuje za uporabu. Ona ikone MIME vrste pridružuje povezanim datotekama. Sintaksa je slična:

```
AddIconByType iconfile mime-type [mime-type] ...
```

*mime-type* može biti ili MIME tip koji se slaže s onim što ste dodijelili datoteci ili šablona koja se slaže. U podrazumijevanim konfiguracijskim datotekama vidjet ćete ove retke:

```
AddIconByType (SND,/icons/sound2,gif) audio/*
```

Slaganje po šabloni je jače od slaganja po nastavcima u imenu datoteke.

AddIconByEncoding obično se rabi za razlikovanje sažetih datoteka od drugih vrsta datoteka. To ima smisla samo ako se uporabi uz AddEncoding upute u srm.conf datoteci. Podrazumijevani srm.conf izgleda ovako:

```
AddEncoding x-gzip gz
AddEncoding x-compress Z
AddIconByEncoding (CMP,/icons/compressed.GIF) x-compress x-gzip
```

Opcija AddIconByEncoding postavlja ikonu pokraj sažete datoteke.

UputaDefaultIcon određuje koju ikonu uporabiti kada se niti jedna šablona ne slaže s datotekom. Naredba je ova:

```
DefaultIcon /icons/unknown.gif
```

Moguće je dodati tekst na vrh i dno popisa kazala direktorija. Ta je sposobnost jako korisna jer pretvara indeks direktorija iz UNIX-ovog sučelja u dinamičko dokumentno sučelje. Dvije instrukcije nadziru taj postupak: HeaderName i ReadmeName koje određuju imena datoteka na vrhu i dnu popisa. Te instrukcije u podrazumijevanoj srm.conf datoteci izgledaju ovako:

```
HeaderName HEADER
ReadmeName README
```

Kada se radi indeksiranje direktorija, Apache će potražiti HEADER.html. Ako je nađe, sadržaj će smjestiti na vrh kazala direktorija. Ako je ne nađe, tražit će samo HEADER. Ako je nađe, pretpostavit će da je datoteka čisti tekst, pa će npr. prevesti znakove < u &lt; niz znakova, a potom ih umetnuti na vrh kazala direktorija. Isti postupak je i kod README, osim što tekst ide na dno kazala direktorija.

U puno slučajeva potrebno je da mehanizam indeksiranja direktorija ne obraća pažnju na neke vrste datoteka, kao što su emacs datoteke sigurnosnih kopija podataka ili datoteke koje počinju s a . (sakrivene datoteke). IndexIgnore uputa omogućuje vam odrediti vrste datoteka koje treba ignorirati kada radite kazalo direktorija. Podrazumijevana postavka je:

IndexIgnore \*/ .??\* \*~# \*/HEADER\* \*/README\* \*/RCS

Taj redak izgleda kao šifra, ali je u stvari popis šablona. Prva šablona odgovara bilo kojoj . datoteci koja je dulja od tri znaka. Tako i dalje može raditi prethodni direktorij u stablu (..). Druga (\*~) i treća (\*#) šablona su zajedničke za slaganje starih emacs datoteka sigurnosnih kopija podataka. Sljedeće šablone služe za izbjegavanje popisa istih datoteka kao kod HeaderName i ReadmeName. Posljednja šablona (\*/RCS) je dana zato što puno lokacija rabi RCS, softver za nadzor nad revizijama, koji sprema svoje informacije u RCS direktorije.

Na kraju, imamo dvije zanimljive instrukcije za kontrolu zadnje postavke opcija koje se tiču indeksiranja direktorija. Prva je AddDescription, koja radi slično AddIcon:

AddDescription description filename [filename] ...

Na primjer,

AddDescription "My cat" /private/cat.gif

Filename može biti šablona, pa možete imati

AddDescription "An MPEG Movie Just For You!" \*.mpg

Na kraju dolazi IndexOptions. Sintaksa joj je jednostavna:

IndexOptions option [option] ...

U tablici 36.2 popisane su sve opcije.

Tablica 36.2         Dostupne opcije za IndexOptions	
Opcija	Objašnjenje
FancyIndexing	Ova opcija je ista kao i uputa FancyIndexing (Zbunjuje, ali u pitanju je uskladivost s ranijim inačicama).
IconsAreLinks	Ako je postavljena ta opcija, ikona će predstavljati vezu prema resursu kojem je pridružen unos za vezu. Drugim riječima, ikona postaje dio hiperveze.
ScanHTMLTitles	Kada dobije popis HTML datoteka, poslužitelj će otvoriti HTML datoteku i analizirati je kako bi odredio vrijednost polja <title> u HTML dokumentu, ako on postoji. To opterećuje poslužitelj jer treba puno prostora na disku i resurse CPU-a kako bi se raspakirao naslov iz HTML-a. Ne preporučuje se, osim ako nemate dovoljno prostora.</title>
SuppressDescription, SuppressLastModified, SuppressSize	Ovo će spriječiti prikaz odgovarajućih polja (Description, Last Modified i Size). Obično su ta polja u ispisu.

Podrazumijevano nije uključena niti jedna opcija IndexOption. Opcije se ne *spajaju* što znači da kada to postavljate uz pomoć access.conf ili .htaccess datoteka, postavljanje opcije za određeniji direktorij zahtijeva ponovno postavljanje svih ispisanih opcija. Na primjer, u konfiguracijskoj datoteci pristupa pogledajte ovo:

```
<Directory /pub/docs/>
IndexOptions ScanHTMLTitles
</Directory>
<Directory /pub/docs/others/>
IndexOptions IconsAreLinks
</Directory>
```

Popisi direktorija u i ispod drugog direktorija, /pub/docs/others, nemaju postavku ScanHTMLTitles. To je zato što administratori moraju biti u stanju onemogućiti opciju koju su postavili općenito u određeni direktorij. To je jednostavnije od pisanja NOT u popisu opcija.

Ako imate problema s indeksiranjem direktorija, pripazite da postavke za Options upute u konfiguracijskim datotekama pristupa dozvoljavaju indeksiranje u tom direktoriju. Options uputa mora uključivati Indexing. Ako rabite .htaccess datoteke za postavljanje AddDescription i AddIcon, instrukcija AllowOverride mora uključivati opciju FileInfo.

# Korisnički direktoriji

Lokacije s puno korisnika katkad mogu korisnicima omogućiti upravljanje dijelovima Web stabla u svojim direktorijima tako da rabe ovakav izgled URL-a:

```
http://myhost.com/~user/
```

gdje je *čuser* u stvari *alias* korisnikovog polaznog direktorija, što se razlikuje od instrukcije Alias koja može preslikati samo određeni pseudodirektorij u stvarni direktorij. U tom slučaju, želite da se *čuser* preslika u /home/user/public\_html. Broj "korisnika" može biti jako velik, pa je ovdje korisna neka vrsta makro naredbe. Ta makro naredba je UserDir.

UserDir određuje poddirektorij u korisnikovim polaznim direktorijima gdje mogu staviti sadržaj koji je preslikan u *čuser* URL. Drugim riječima, podrazumijevani

```
UserDir public_html
```

dat će zahtjev za

```
http://myhost.com/~dave/index.html
```

što će potražiti UNIX datoteku

/home/dave/public\_html/index.html

pretpostavljajući da je /home/dave Daveov polazni direktorij.

# Posebni moduli

Ono što Apache razlikuje od konkurencije su funkcije koje su umetnute kao moduli u Apache API, što je izuzetno korisno zato što se funkcije mogu razvijati odvojeno od poslužitelja, a izvedba se može ugađati. Posebne funkcije opisane su u sljedećim odjeljcima.

## Predprocesiranje na strani poslužitelja

Predprocesiranje se zbiva na strani poslužitelja tako da posjetitelji vaše lokacije ne moraju imati poseban klijent-softver. Format ovih instrukcija izgleda otprilike ovako:

```
<! - -#directive attribute="value" - ->
```

Kadšto dana direktiva može imati više od jednog atributa u isto vrijeme. Sintaksa je malo čudna jer se željelo sakriti tu funkciju u SGML komentar – tako HTML alati za provjeru rade, a vi ne morate učiti nove *tagove*. Sintaksa je jako važna, na primjer, izostavljanje - - uzrokuje pogrešku.

**#include** #include se vjerojatno najčešće rabi. Rabite je za umetanje druge HTML datoteke u HTML dokument. Dozvoljeni atributi za #include su virtual i file. Funkcija file atributa je podskup onog što omogućuje atribut virtual, a postoji većinom uskladivost s ranijim inačicama, pa njezina uporaba nije preporučljiva.

Atribut virtualnom poslužitelju govori da s vrijednosti atributa postupa kao sa zahtjevom za relativnu vezu – znači da možete uporabiti ../ za smještaj objekata iznad direktorija, a na njega će se primijeniti druge pretvorbe, kao što je Alias. Evo primjera:

```
<! - -#include virtual="quote.txt" - ->
<! - -#include virtual="/toolbar/footer.html" - ->
<! - -#include virtual="../footer.html" - ->
```

**#exec** Uputa#exec rabi se za pokretanje skripte na strani poslužitelja i usmjeravanje njezinog izlaza u SSI dokument. Postoje dvije mogućnosti: izvršavanje CGI skripte uporabom cgi atributa ili izvršavanje naredbe ljuske uporabom cmd atributa.

Na primjer,

```
<! - -#exec cgi="counter.cgi" - ->
```

usmjerava izlaz CGI programa counter.cgi u dokument.

BILJEŠKA CGI izlaz ipak treba imati zadanu vrstu sadržaja "text/html", inače će se pojaviti pogreška.

Isto tako,

<! - -#exec cmd="ls -1" - ->

uzima izlaz naredbe ls -1 u direktoriju dokumenta i umeće ga na izlaznu stranicu kao zamjenu za naredbu #exec. Kao i atribut file za instrukciju #include, ova vrsta #exec naredbe je uglavnom tu zbog uskladivosti s ranijim inačicama, jer je u biti sigurnosna rupa.

**BILJEŠKA** Postoje veliki sigurnosni problemi ako se korisnicima dozvoli pristup CGI funkcijama te naredbi #exec cmd, kao što je

cmd="cat /etc/passwd"

Ako administrator lokacije želi korisnicima omogućiti uporabu ovih naredbi, ali ne i uporabu naredbe #exec, može postaviti opciju IncludesNOEXEC u konfiguracijskoj datoteci pristupa.

**#echo** #echo ima jedan atribut – var – čija je vrijednost svaka varijabla iz CGI okoline kao i još neke varijable popisane u tablici 36.3.

tablica 36.3 Vrijednosti za var atribut		
atribut	definicija	
DATE_GMT	Datum u obliku Greenwich Mean Time.	
DATE_LOCAL	Datum u mjesnoj vremenskoj zoni.	
DOCUMENT_NAME	Ime datotečnog sustava SSI dokumenta, ne uključuje imena direktorija ispod njega.	
DOCUMENT_URI	URI je <i>Uniform Resource Identifier</i> . U Uniform Resource Locator (URL) formatu http://host/path/file URI je dio /path/file.	
LAST_MODIFIED	Datum kada je SSI dokument mijenjan.	

Na primjer, naredba

<! - -#echo var="DATE\_LOCAL" - ->

umeće u dokument npr. Wednesday, 05-Mar-97 10:44:54 GMT.

**#fsize**, **#flastmod** Instrukcije #fsize i #flastmod ispisuju veličinu i datum zadnje promjene objekta prema URI popisu u file ili virtual atributima. Na primjer, naredba

<! - -#fsize file="index.html" - ->

vraća veličinu index.html datoteke u taj direktorij.

**#config** Obradu određenih SSI instrukcija možete mijenjati uporabom instrukcije #config. Atribut sizefmt kontrolira obradu instrukcije #fsize vrijednostima bytes ili abbrev. Točan broj bajtova ispiše se kada se daje naredba bytes, a skraćena inačica veličine (u KB za kilobajte ili MB za megabajte) daje se kada se postavi abbrev (podrazumijevano). Na primjer, dio SSI HTML-a

```
<! - -#config sizefmt="bytes" - ->
The index.html file is <! - -#fsize virtual="index.html" - -> bytes
daje The index.html file is 4,522 bytes. Nadalje,
<! - -#config sizefmt="abbrev" - ->
daje The index.html file is 4K bytes.
```

Uputa timefint kontrolira obradu datuma u DATE\_LOCAL, DATE\_GMT i LAST\_MODIFIED vrijednosti instrukcije #echo. Ona rabi isti format kao i strftime. (U stvari, poslužitelj poziva strftime, koji formatira vrijeme u niz određene duljine.) Format niza sastoji se od varijabli koje počinju s %. Na primjer, %H je sat. Upute za konstrukciju strftime formata za datum, naći ćete na strftime stranici.

#### Primjer

```
<! - -#config timefmt="%Y/%m/%d-%H:%M:%S" - ->
```

za dan 2. veljače 1997. 12:30 daje rezultat

1997/01/02-12:30:00

Posljednji atribut koji #config može imati je errmsg, što je jednostavno ispis pogrešaka u dokumentu. Na primjer, podrazumijevano je

<! - -#config errmsg="An error occurred while processing this directive" -->

#### Kolačići

HTTP *kolačići* su metoda održavanja jasne i stalne veze s klijentom. Što to znači? Na HTTP-u veza između klijenta i poslužitelja obično se proteže na puno odvojenih stvarnih TCP veza, što otežava pristup u aplikacije koje zahtijevaju stanje. Kolačići su rješenje. Poslužitelji mogu klijentu dodijeliti kolačić, neku vrstu neprozirnog niza koji je značajan samo poslužitelju, a potom klijent može vratiti taj kolačić poslužitelju na zahtjev.

Modul mod\_cookies upravlja detaljima dodjeljivanja jedinstvenih kolačića svakom posjetitelju, na temelju imena posjetiteljevog računala-domaćina i slučajnog broja. Kolačičima se može pristupiti iz CGI okoline kao HTTP\_COOKIE varijabli okoline. CGI skripte to mogu rabiti kao ključ u bazi podataka o sesijama ili bilježiti da se procijeni broj korisnika koji su posjetili lokaciju, a ne samo broj prijava.

Srećom, ovdje nema igre s postavkama. Jednostavno kompilirajte s mod\_cookies i to je sve.

## Prilagodiv postupak prijave

Podrazumijevani format datoteke s podacima o prijavi (nazvan *Common Logfile Format* ili CLF) ne daje dovoljno informacija za ozbiljnu analizu uspješnosti vaše Web lokacije. On daje broj prijava, pristupe na stranice, pristupe računala-domaćina, vremenske oznake itd., ali ne daje URL-ove, koji se pretraživač i kolačići rabe. Postoje dva načina kako ćete dobiti više podataka: uporabom NCSA-uskladive naredbe za prijavu nekih bitova informacija na odvojenim pretraživačima ili uporabom Apacheovog formata datoteke s podacima o prijavi koji se može oblikovati.

**NCSA-uskladivost** Za usklađivanje s NCSA 1.4 Web poslužiteljem dodana su dva modula. Ti moduli prijavljuju User-Agent i Referer zaglavlja iz HTTP zahtjeva.

User-Agent je zaglavlje koje identificira softver koji pretraživač rabi. Prijavu zaglavlja može aktivirati instrukcija AgentLog u datoteci srm.conf. Ta instrukcija uzima jedan argument, ime datoteke u koju su prijavljeni korisnički agenti, na primjer,

AgentLog logs/agent\_log

Želite li uporabiti AgentLog instrukciju, pripazite da mod\_log\_agent modul bude kompiliran i povezan s poslužiteljem.

Pretraživač šalje zaglavlje Referer kako bi pronašao završne veze. Drugim riječima, kada ste na stranici koja ima URL "A" i na njoj postoji veza koja ima URL "B" i vi slijedite u vezu, zahtjev za stranicom "B" uključuje Referer zaglavlje s URL-om "A". To je jako korisno želite li pronaći koje se lokacije povezuju s vašom i koji je promet na njima.

Prijava Referer zaglavlja aktivira se uputa RefererLog što pokazuje na datoteku na koju se prijavljuje:

RefererLog logs/referer\_log

RefererIgnore je uputa koja vam omogućuje ignoriranje Referer zaglavlja. RefererIgnore korisna je želite li počistiti ove unose sa svoje lokacije. Na primjer, ako je vaša lokacija www.myhost.com, uporabite ovo:

RefererIgnore www.myhost.com

Zapamtite da prijava Referer zaglavlja zahtijeva kompiliranje i povezivanje sa mod\_log\_referer.

**Potpuno prilagodiva prijava** Prethodni moduli uskladivi su s prethodnim inačicama. No, tu ima i nekih problema. Kako nemaju nikakve druge informacije o zahtjevu prema kojem se prijavljuju, gotovo je nemoguće odrediti koja Referer polja odgovaraju kojim objektima na vašoj lokaciji. U idealnom slučaju sve informacije o transakcijama s poslužiteljem mogu se prijaviti u jednu datoteku, proširenjem formata datoteke s podacima o prijavi ili njegovom potpunom zamjenom. Ta mogućnost postoji u modulu mod\_log\_config. Modul mod\_log\_config izvršava instrukcija LogFormat koja kao svoj argument uzima niz, a varijable počinju s % kako bi ukazale na različite vrste podataka iz zahtjeva. U tablici 36.4 je popis varijabli.

	Tablica 36.4	rijable za LogFormat uputu	
	Varijabla	Definicija	
	%h	Udaljeni poslužitelj (host).	
	%1	Daljinska identifikacija preko identd.	
	%u	Udaljeni korisnik, određen prema bilo kakvoj provjeri autentičnosti korisnika. Ako se korisniku ne provjeri autentičnost, a status zahtjeva je 401 (pogreška pri autorizaciji), ovo polje može se izmisliti.	
	%t	Uobičajeni format datoteke s podacima o prijavi za vrijeme.	
	%r	Prvi redak zahtjeva.	
	% S	Status. Za zahtjeve koji su preusmjereni, ovo je stastus prvobitnog zahtjeva; %>s će dati posljednji zahtjev.	
	%b	Poslani bajtovi.	
	%{Foobar}i	Sadržaj Foobara: reci zaglavlja u zahtjevu računala-klijenta poslužitelju.	
	%{Foobar}o	Sadržaj Foobara: reci zaglavlja u odazivu poslužitelja klijentu.	
1			

Na primjer, ako u prijavi želite zadržati samo ime udaljenog poslužitelja, zahtijevani objekt i vrijeme, uporabite ovo:

LogFormat "%h \"%r\" %t"

Što će prijaviti ovo

```
host.outsider.com "GET /HTTP/1.0" [06/Mar/1996:10:15:17]
```

#### Mala uputa

Varijabla zahtjeva mora imati navodnike. Modul za prijavu će automatski protumačiti vrijednosti varijabli, a neće samo pročitati ime varijable. Obrnuta kosa crta s navodnikom, \", rabi se za označavanje da je to znak navodnika, a ne kraj niza. Na primjer, ako želite dodati prijavu User-Agent niza, vaš format prijave bit će:

LogFormat "%h \"%r\" %t \"%{User-Agent}i\""

Budući da User\_Agent polje ima u sebi razmake i oni se trebaju staviti u navodnike. Ako hoćete zadržati Referer polje:

nastavlja se

nastavak

```
LogFormat "%h \"%r" %t %{Referer}i"
```

Ne trebate preskakati navodnike jer **Referer** zaglavlja, kao i URL-ovi nemaju razmake. Međutim, ako radite neku prijavu, razmake trebate staviti u navodnike jer **Referer** isporučuje klijent i nema garancija o formatu.

Podrazumijevani format datoteke s podacima o prijavi je Common Logfile Format (CLF), koji se izražava kao

```
LogFormat "%h %l %u %t \"%r\" %s %b
```

U stvari, većina postojećih alata za analizu datoteke s podacima o prijavi za CLF neće obraćati pažnju na posebna polja na kraju. Želite li dobiti najvažnije informacije, možete uporabiti ovaj format:

LogFormat "%h %1 %u %t \"%r\" %s %b %{Referer}i \"%{User-Agent}i\""

Ako želite još veći nadzor nad onim što se prijavljuje, možete uporabiti prilgodivi prijavni modul i tako umetnuti jednostavni uvjetni test varijabli. Tako ga možete oblikovati da prijavi varijable samo kada se vraća – ili ne vraća - određeni statusni kod. Format za hvatanje statusnog koda je umetnuti popis tih kôdova odvojenih zarezom između % i slova varijable:

```
%404,403{Referer}i
```

Ovaj primjer znači da će se Referer zaglavlje prijaviti samo ako je status koji vrati poslužitelj 404 Not Found ili 403 Access Denied. U ostalim slučajevima je prijavljen. Inače, pogreške prijave 403 i 404 su korisne ako vas zanimaju stare veze koje pokazuju na resurse koji više nisu dostupni.

Negacija Referer statusnog koda je uskličnik (!) na početku popisa kodova. Na primjer,

%!401u

prijavljuje korisnika za svaku provjeru, osim ako provjera ne zadovoljava. U tom slučaju vjerojatno ne želite vidjeti ime korisnika.

Poput drugih funkcija, funkcije prijave mogu se oblikovati po virtualnom domaćinu. Želite li da sve prijave od svih virtualnih domaćina na istom poslužitelju idu na istu prijavu, možete napraviti ovo:

```
LogFormat "hosta ...."
```

u <VirtualHost > odjeljcima za hosta i

LogFormat "hostb ...."

u <VirtualHost > odjeljcima za hostb. Više riječi o virtualnim domaćinima bit će u odjeljku "Virtualni *hostovi* (domaćini)".

BILJEŠKA Kako biste podesili prijavljivanje na ovoj osnovi, morate kompilirati s mod\_log\_config. Pripazite da podrazumijevani mod prijave, mod\_log\_common nije kompiliran; inače će se poslužitelj zbuniti. ■

# Pregovaranje o sadržaju

Pregovaranje o sadržaju je mehanizam pomoću kojeg Web klijent može reći poslužitelju koje vrste podataka zna obraditi i na temelju te informacije poslužitelj može klijentu dati "optimalnu" inačicu zahtjevanog resursa. Ovakvo se pregovaranje može dogoditi kod brojnih različitih tipova podataka, govornom jeziku u kojem su podaci (npr. engleskom ili francuskom), skupu znakova dokumenta i njegovim kodovima.

**Pregovaranje o vrsti datoteke** Ako želite rabiti JPEG slike na stranicama, ali ne želite praviti probleme korisnicima čiji pretraživači ne mogu rukovati JPEG slikama, možete napraviti i GIF inačicu slike. Iako GIF datoteka može biti veća ili biti samo 8-bitna, bolje je i to nego prekinuta veza. Dakle, pretraživač i poslužitelj *pregovaraju* koje podatke poslužitelj šalje računalu-klijentu.

Specifikacije za pregovaranje dio su HTTP-a od samog početka. No, na njih se ne možete osloniti. Na primjer, sadašnji pretraživači koji imaju dodatke u zaglavljima vezanih paketa ne pokazuju za koje vrste medija imaju dodatke. Tako se pregovaranje ne može uporabiti za odlučivanje treba li nekom poslati ShockWave datoteku ili odgovarajuću u Javi. Jedino sigurno mjesto za pregovore za sada su JPEG ili GIF slike na stranici, što danas rabi dosta poslužitelja.

Mod\_negotiation.c datoteka u Apacheu 1.0 ima ove specifikacije u starijim inačicama HTTP/1.0 IETF, što će uskoro dobiti RFC status. Pregovaranje je uklonjeno jer specifikacija nije bila potpuna. Za HTTP/1.1 specifikacije pregovaranja su znatno pojačane, što ne znači da se može sigurno rabiti za odabir umetnutih slika.

Želite li aktivirati pregovaranje, morate uključiti mod\_negotiation.c modul na poslužitelj. Postoje dva načina oblikovanja:

- Uporaba datoteke koja opisuje sve vrste resursa o kojima se pregovara s određenim vrijednostima i karakteristikama sadržaja
- Postavljanje Option vrijednosti zvane MultiViews

Razmotrit ćemo funkcioniranje MultiViews, a ako vas zanima navedena datoteka, dokumentaciju potražite na Apacheovoj Web lokaciji.

U datoteci access.conf pronađite redak u kojem su postavljene opcije za dio lokacije za koji želite omogućiti pregovaranje. (Možete to postaviti i za cijelu lokaciju.) Dakle, imat ćete otprilike ovakav redak

```
Options Indexes Includes Multiviews
```

```
ili
Options All MultiViews
```

Kada se promijeni MultiViews parametar, morate ponovno pokrenuti svoj poslužitelj.

Kada je uključena MultiViews, napravite ovo: smjestite JPEG sliku u direktorij, npr /path/ i nazovite je image.jpg. Potom napravite odgovarajuću sliku u GIF formatu i smjestite je u isti direktorij /path/ kao i image.gif. URL-ovi za te objekte su:

```
http://host/path/image.jpg
i
http://host/path/image.gif
```

Ako sada od svog Web pretraživača zatražite da nađe

http://host/path/image

poslužitelj će otići u direktorij/path/, pogledati dvije slike i odlučiti koji format slike poslati na temelju onoga što može podržati računalo-klijent. Ako računalo-klijent može prihvatiti oba formata, poslužitelj će odabrati inačicu koja je manja i poslati je klijentu. Obično su JPEG slike puno manje od GIF-ova.

Ako vaš HTML izgleda slično ovome

```
<HTML><HEAD>
<TITLE>Welcome to the Gizmo Home Page!</TITLE>
</HEAD><BODY>
<IMG SRC="/header" ALT="GIZMO Logo">
Welcome to Gizmo!
<IMG SRC="/products" ALT="Products">
<IMG SRC="/services" ALT="Correct">
```

Možete imati odvojene GIF i JPEG datoteke za zaglavlje, proizvode i usluge. Računala-klijenti će dobiti ono što mogu podržati.

Ako imate datoteke pod imenom image te image.gif, datoteka image će biti vraćena ako je zahtjev napravljen za image. Slično, zahtjev za image.gif nikad neće vratiti image.jpg iako računalo-klijent zna kako prikazati JPEG slike.

**Pregovaranje o jeziku** Ako je omogućena opcija MultiViews, resurse možete razlikovati prema jeziku, npr. japanskom ili engleskom. To ćete napraviti dodavanjem više unosa kod imena datoteke koji pokazuje na jezike koje poslužitelj želi uporabiti, a potom ih rangirati. U srm.conf datoteci pojavljuju se dvije nove upute - AddLanguage i LanguagePriority. Formati su ovakvi:

AddLanguage en .en AddLanguage it .it AddLanguage fr .fr AddLanguage jp .jp LanguagePrority en fr jp it

Pretpostavimo da želite odabrati jezik u datoteci index.html koja vam je dostupna na engleskom, francuskom, talijanskom i japanskom. Napravit ćete index.html.en, index.html.fr, index.html.it, index.html.jp, a potom pozvati dokument index.html. Kada se poveže višejezično računalo-klijent, u jednom od svojih zaglavlja (npr. Accept-Language) trebat će odrediti koji jezik želi, a pretraživač to pokazuje standardnom kraticom od dva slova. Poslužitelj vidi što klijenti mogu prihvatiti i daje im "najbolje". LanguagePriority organizira odluku o "najboljem". Ako klijent ne prihvaća engleski, pokušat će s francuskim, potom japanskim i na kraju s talijanskim. LanguagePriority također određuje koji jezik prihvatiti ako nema Accept-Language zaglavlja.

Sufiksi za jezik i sadržaj dijele isti prostor u imenu, pa ih možete miješati. Index.fr.html je isto kao i index.html.fr. Jedino pazite da ih povežete s ispravnim resursom.

#### As-is datoteke

Često u dokumentima želite rabiti zaglavlja, kao što su Expires:, ali ne želite da stranica bude CGI skripta. Najlakši način je dodavanje httpd/send-as-is MIME tipa u srm.conf. datoteku.

AddType httpd/send-as-is asis

To znači da svaka datoteka koja završava na .asis može uključiti svoja MIME zaglavlja. Međutim, mora uključiti dva prijeloma retka prije tijela sadržaja teksta. Apache će sam umetnuti dopunu do kraja retka. Dakle, ako želite poslati dokument s korisničkim MIME tipom koje nema na poslužitelju, možete poslati slijedeće:

```
Content-type: text/foobar
This is text in a very special "foobar" MIME type.
```

Najznačajnija aplikacija za ovo je izuzetno uspješan mehanizam za poslužiteljsko *guranje* objekata bez CGI skripti. Razlog zašto CGI skripta treba napraviti guranje obično je taj da sadržaj uključuje odvajatelja (zato što se radi o višedjelnoj poruci). U sljedećim primjerima XXXXXXX ukazuje na granicu između dijelova poruke:

```
Content-type: multipart/x-mixed-replace;boundary= XXXXXXXX

- - XXXXXXXX

Content-type: image/gif

....(GIF data) ....

- - XXXXXXXX

Content-type: image/gif

.... (GIF data)....

- - XXXXXXXX

....
```

Kada niz podataka pretvorite u jednostavnu datoteku s parametrom .asis u AddType uputi, umjesto uporabe CGI skripte, štedite si puno problema.

Ako je uključen MultiViews, .asis možete dodati na kraj imena datoteke, a ni jednu vezu ne morate preimenovati. Na primjer, foobar.html može biti foobar.html.asis, a ipak je možete pozvati kao foobar.html.

Posljednja primjena "asis" je HTTP preusmjeravanje bez pristupa konfiguracijskim datotekama poslužitelja. Na primjer, sljedeća .asis datoteka će preusmjeriti ljude na drugo mjesto:

```
Status 302 Moved
Location: http://some.other.place.com/path/
Content-type: text/html
<HTML>
<HEAD><TITLE>We've Moved!</TITLE></HEAD>
<BODY>
<H1>We used to be here, but now we're
<A HREF="http://some.other.place.com/path/">over there. </A>
</H1>
</BODY></HTML>
```

HTML tijelo postoji samo zbog onih klijenata koji ne razumiju odgovor broj 302.

# Napredno funkcioniranje

Želite li još veću kontrolu nad svojim poslužiteljem ili prilagođavanje svoje operativne okolice, možete oblikovati Apache poslužitelj tako da podupire napredne funkcije kao što je kontrola pristupa i zaštita korisnika.

# Kontrola pristupa na hostu (domaćinu)

Možete kontrolirati pristup poslužitelju ili čak poddirektoriju poslužitelja na temelju imena računala-domaćina, domene ili IP adrese klijentovog računala. To se radi pomoću instrukcija allow i deny, koje se mogu rabiti zajedno s order. allow i deny mogu preuzeti višestruke domaćine:

```
deny from badguys.com otherbadguys.com
```

Obično želite napraviti dvije stvari: ne dozvoliti pristup svojem poslužitelju, osim s nekoliko odabranih računala ili odobriti pristup svima osim nekoliko računala domaćina. Zabrana pristupa svima osim nekoliko računala postiže se ovako:

```
order deny, allow
allow from mydomain.com
deny from all
```

Ova instrukcija znači "Odobri pristup samo računalima domaćinima u domeni mydomain.com". Ta domena može uključivati host1.mydomain.com, ppp.mydomain.com i the-boss.mydomain.com.

Prethodna uputa govori poslužitelju da provjeri deny uvjete prije allow uvjeta kada odlučuje hoće li dozvoliti pristup. Slično tome, isključivanje nekoliko lokacija može se provesti ovako:

```
Order allow, deny
allow from all
deny from badguys.com
```

order je potreban jer poslužitelj treba znati koje pravilo najprije primijeniti. Podrazumijevano za order je deny, allow.

U trećem argumentu mutual-feature, uvjet mora proći allow i deny pravila kako bi uspio. Drugim riječima, mora se pojaviti na allow popisu, a ne smije se pojaviti na deny popisu, slično ovom primjeru:

```
order mutual-failure
allow from mydomain.com
deny from the-boss.mydomain.com
```

U ovom primjeru the-boss.mydomain.com ne može pristupiti resursu, a sva ostala računala na mydomain.com mogu.

#### **UPOZORENJE**

Zaštita resursa pomoću imena računala domaćina je opasna. To je lak postupak za čovjeka koji kontrolira obrnuto DNS mapiranje za svoju IP adresu kako bi podvalio bilo koje ime računala domaćina. Zato se toplo preporučuje uporaba IP adresa za zaštitu povjerljivih podataka. Na isti način, možete jednostavno popisati ime domene koja se odnosi na bilo koje računalo u toj domeni. Možete dati i dijelove IP adresa:

```
allow from 204.62.129
```

To će omogućiti samo ona računala-domaćine čije se IP adrese poklapaju, npr. 204.62.129.1 ili 204.62.129.130.

Obično se te upute rabe u <Limit> spremniku, te u <Directory> spremniku, najčešće u datoteci access.conf. Slijedeći je primjer dobar predložak za zaštitu; on štiti /www/htdocs/private direktorij od svih domaćina osim onih u 204.62.129 IP prostoru:

```
<Directory> /www/htdocs/private>
Options Includes
AllowOverride None
<Limit GET POST>
order allow,deny
```

nastavlja se

nastavak

```
deny from all
allow from 204.62.129
</Limit>
</Directory>
```

## Provjera korisnika

Kada resurs smjestite pod *provjeru korisnika*, ograničavate mu pristup zahtjevom za ime i lozinku. Ime i lozinka čuvaju se u bazi podataka na poslužitelju. Ta baza podataka može različito izgledati; Apache moduli napisani su za stolne baze podataka, sustave za upravljanje bazama podataka (DBM), mSQL baze podataka, Oracle i Sybase baze podataka i druge. U ovom poglavlju govorimo samo o stolnim i DBM-format bazama podataka.

Najprije nešto o osnovnim uputama za oblikovanje. Uputa AuthName postavlja provjeru valjanosti "područja" za stranice koje su zaštićene lozinkom. Područje je ono što se predstavlja računalu-klijentu kada se od njega traži provjera, kao u Please, enter your name and password for the realm.

Instrukcija AuthType postavlja vrstu provjere za područje. U HTTP/1.0 postoji samo jedna vrsta provjere – Basic. HTTP/1.1 imat će ih još nekoliko, kao što je MD5.

Instrukcija AuthUserFile određuje datoteku koja sadržava popis imena i lozinki, po jedan par u retku. Lozinke su šifrirane po UNIX crypt() uputama. Na primjer,

```
joe:D.W2yvlfjaJoo
mark: 21slfoUYGksIe
```

Instrukcija AuthGroupFile određuje datoteku koja sadržava popis grupa i članova tih grupa, npr.:

managers: joe mark
production: mark shelley paul

Konačno, instrukcija require određuje koje uvjete treba ispuniti za prihvaćanje pristupa. Ona može popisati samo korisnike koji se mogu povezati, odrediti grupu ili popis grupa korisnika koji se mogu povezati ili reći da je nekom korisniku iz baze podataka automatski omogućen pristup. Evo primjera:

```
Require user mark paul
(Only mark and paul may access.)
require group managers
(Only people in group managers may access.)
require valid-user
(Anyone in the AuthUserFile database may access.)
```

Konfiguracijska datoteka završava otprilike ovako:

```
<Directory /www/htdocs/protected/>
AuthName Protected
AuthType basic
AuthUserFile /usr/local/etc/httpd/conf/users
<Limit GET POST>
require valid-user
</Limit>
</Directory>
```

Želite li zaštiti direktorij za određenu grupu, konfiguracijska datoteka izgleda otprilike ovako:

```
<Directory /www/htdocs/protected/>
AuthName Protected
AuthType basic
AuthUserFile /usr/local/etc/httpd/conf/users
AuthGroupFile /usr/local/etc/httpd/conf/group
<Limit GET POST>
require group managers
</Limit>
</Directory>
```

## Baza podataka korisnika

Apache možete oblikovati da rabi DBM datoteke za brži pregled lozinke i grupnog članstva. Želite li uporabiti DBM datoteku, morate imati modul kompiliran mod\_auth\_dbm.

DBM datoteke su UNIX datoteke koje brzo pregledavaju *hash* tablice, pa su idealne za rukovanje velikim bazama podataka korisnik/lozinka. Stolni sustavi zahtijevaju analizu datoteke lozinke za svaki pristup sve dok se ne nađe odgovarajuća lozinka. Tablice smjesta znaju postoji li u bazi podataka "ključ" i koja je njegova vrijednost.

Neki sustavi rabe ndbm biblioteke; neki rabe berkeley db biblioteke. Međutim, sučelje kroz Apache je isto.

Želite li uporabiti DBM datoteku za bazu podataka, uporabite instrukciju AuthDBMUserFile umjesto AuthUserFile. Isto tako za datoteku grupe uporabite AuthDBMGroupFile umjesto AuthGroupFile.

# Virtualni hostovi (domaćini)

Apache ima jasan način postupanja s *virtualnim domaćinima*, mehanizmom za posluživanje više od jednog računala-domaćina na određenom računalu. Zbog ograničenja u HTTP-u, posluživanje višestrukih domaćina postignuto je dodjeljivanjem više od jedne IP adrese računalu, a potom različitim povezivanjem Apachea s tim IP adresama. Na primjer, UNIX okvir može imati brojeve 204.122.133.1, 204.122.133.2 i 204.122.133.3 koji pokazuju na njega, s time da je www.host1.com vezan za prvi, www.host2.com za drugi, a www.host3.com za treći broj.



SAVJET Apache 1.2 i više inačice preko HTTP 1.1 specifikacije protokola podupiru i virtualne domaćine koji se ne temelje na IP-u. S tom novom osobinom više ne morate osigurati IP adresu svakom virtualnom domaćinu.

Virtualni domaćini oblikovani su pomoću spremnika u httpd.conf. Izgledaju otprilike ovako:

```
<VirtualHost www.host1.com>
DocumentRoot /www/htdocs/host1/
TransferLog logs/access.host1
ErrorLog logs/error.host1
</VirtualHost>
```

Atribut u VirtualHost tagu je ime računala-domaćina koje poslužitelj pregledava kako bi dobilo IP adresu.

Ako postoji mogućnost da www.host1.com vrati više od jednog broja ili da BILJEŠKA Web poslužitelj ima problema razlučiti ime računala domaćina od IP adrese, uporabite samo IP adresu.

Svaka instrukcija stavljena u VirtualHost spremnik odnosi se samo na zahtjeve za ime tog računala-domaćina. DocumentRoot pokazuje na direktorij koji vjerojatno ima sadržaj za www.host1.com.

Svaki virtualni domaćin može imati svoju prijavu pristupa, svoju prijavu pogreške, svoje izvedenice prijava, svoje Redirect i Alias upute, svoje ServerName i ServerAdmin upute i drugo. Njegov poslužitelj ne može podržati jedino ove upute:

d

ServerType	MaxRequestsPerChil
UserId	BindAddress
GroupId	PidFile
StartServers	TypesConfig
MaxSpareservers	ServerRoot
MinSpareServers	

Ako planirate izvoditi Apache s velikim brojem virtualnih domaćina, trebate pripaziti na ograničenja. Na primjer, neke UNIX platforme dozvoljavaju otvaranje samo 64 opisnih blokova datoteke odjednom. Apache potomak će potrošiti jedan opisni blok datoteke po datoteci s podacima o prijavi po virtualnom domaćinu, 32 virtualna domaćina – svaki sa svojom datotekama s podacima transfera i pogrešaka – brzo će doći do tog ograničenja. Ako vaša datoteka s podacima o pogreški počne prijavljivati pogreške kao npr. unable to fork(), ili vam datoteke s podacima obavljenog pristupa nisu napisane, znači da imate takve probleme. Apache pokušava pozvati setrlimmit() (poziv funkcije sustava koja pokušava ograničiti procese) kako bi sam riješio taj problem, ali je to katkad neuspješno.

## Prilagođene poruke o pogreškama

Apache može dati prilagođene odgovore u slučaju pogreške, što se kontrolira pomoću instrukcije ErrorDocument. Sintaksa izgleda ovako:

ErrorDocument HTTP\_response\_code action

HTTP\_response\_code potiče akciju. Akcija može biti

- Lokalni URL prema kojem se poslužitelj preusmjerava
- Vanjski URL prema kojem se preusmjerava klijent
- Niz tekstualnih znakova koji počinje znakom navodnika ", a %s varijabla sadržava posebnu informaciju

#### Na primjer,

```
ErrorDocument 500 "Ack! We have a problem here: %s
ErrorDocument 500 /errors/500.cgi
ErrorDocument 500 http://backup.myhost.com/
ErrorDocument 401 /subscribe.html
ErrorDocument 404 /debug/record-broken-links.cgi
```

Dvije dodatne CGI varijable proslijeđene su u svaki preusmjereni izvor: REDIRECT\_URL sadržava originalni URL zahtjev, a REDIRECT\_STATUS daje prvobitan status koji je uzrokovao preusmjeravanje. To će pomoći skripti ako je njezin posao pronaći što je uzrokovalo odziv na pogrešku.

## Ostale httpd.conf postavke

Neke opcije ne spadaju ni u koju kategoriju jer je njhovo funkcioniranje različito od ostalih. To su, među ostalima, BindAddress, PidFile i Timeout.

**BindAddress** Pri pokretanju Apache se veže za *port* koji je određen za sve IP adrese u okviru. Uputa BindAddress može se uporabiti za određivanje samo određene IP adrese za povezivanje. Uporabom određene adrese možete izvoditi višestruke kopije Apachea od kojih će svaka posluživati drugog virtualnog domaćina. To je korisno ako želite izvoditi Web poslužitelje s različitim sustavom korisničkih ID-ova zbog sigurnosnih razloga.

Pretpostavimo da imate tri IP adrese: 1.1.1.1, 1.1.1.2 i 1.1.1.3, a 1.1.1.1 je osnovna adresa računala. Želite izvesti tri Web poslužitelja, ali želite jednog od njih izvesti kao drugačiji ID od ostala dva. Trebate dva skupa konfiguracijskih datoteka. Prva konfiguracijska datoteka bit će otprilike ovakva:

```
User web3
BindAddress 1.1.1.3
ServerName www.company3.com
DocumentRoot /www/company3/
```

#### 708 36. poglavlje Prilagodba Apachea

Druga konfiguracijska datoteka će izgledati otprilike ovako:

```
User web1
ServerName www.company1.com
DocumentRoot /www/company1/
<VirtualHost 1.1.1.2>
ServerName www.company2.com
DocumentRoot /www/company2/
</VirtualHost>
```

Ako pokrenete prvu datoteku ona će se vezati samo za IP adresu 1.1.1.3. Druga će se vezati za *port* na svim IP adresama jer nema BindAddress upute. Dakle, želite pokrenuti poslužitelj s prvim skupom konfiguracijskih datoteka, a potom pokrenuti drugu kopiju poslužitelja s drugim skupom konfiguracijskih datoteka. Dva poslužitelja će se tako izvoditi na istom računalu.

**Pidfile** PidFile je mjesto datoteke koja sadržava postupak ID za Apache. Ta datoteka je korisna zbog sposobnosti automatizacije zaustavljanja pogona ili ponovnog pokretanja Web poslužitelja. Podrazumijevano je taj parametar logs/httpd.pid. Na primjer, poslužitelju možete obustaviti rad naredbom:

```
Cat /usr/local/etc/httpd/logs/httpd.pid { xargs kill -15
```

Možete to maknuti iz direktorija s podacima o prijavi u direktorij kao što je /var.

**Timeout** Uputa Timeout određuje koliko vremena će poslužitelj čekati između paketa poslanih prije nego vezu smatra "izgubljenom". Na primjer, 1200 (podrazumijevano) znači da će poslužitelj čekati 20 minuta nakon slanja paketa, a zatim vezu smatrati izgubljenom ako nema odziva. Neki jako aktivni poslužitelj mogu skratiti to vrijeme.

# Odavde...

Možete naučiti više o izvođenje Apache Web poslužitelja u slijedećim poglavljima:

- 35. poglavlje "Početak u Apacheu", u kojem se raspravlja o osnovama instaliranja i konfiguracije Apache Web poslužitelja.
- 36. poglavlje "Prilagodba Apachea", u kojem vam pokazujemo kako poslužitelj učiniti snažnim, uspješnim, automatiziranim i sigurnim.

# P 0 G L A V L J E **37**

# Upravljanje Web poslužiteljem

(Steve Burnett)

# U ovom poglavlju

Kontrola procesa djece (potomaka) na poslužitelju Uporaba procesne datoteke Povećavanje uspješnosti softvera za poslužitelj Automatizacija rotacije prijavne datoteke O sigurnosti Ostale teme vezane uz prilagođavanje Jedna od velikih prednosti Apache Web poslužitelja je visoka prilagodivost. Gotovo svaka osobina koja zahtijeva posebnu vrstu učitavanja poslužitelja je mogućnost, što znači da možete osobine podrediti brzini. To znači da je Apache napravljen tako da bude brz i efikasan. Sve te Apachea osobine rijetko ćete kada morati uporabiti, toliko ih ima.

Apache je napravljen tako da administratorima lokacije omogućuje nadzor nad sigurnosti i funkcionalnosti. Za neke je lokacije, poput ISP-a, vrlo važna mogućnost nadzora vrste funkcija. Također, na Webu je potrebna potpuna fleksibilnost, čak i na račun sigurnosti ili oštećenja koja može nanijeti npr. CGI skripta (puno ljudi smatra da je CGI općenito rupa u sigurnosti).

# Kontrola procesa djece (potomaka) na poslužitelju

Vidi "Pokretanje Apachea", str. 677

Kao što ste naučili u 35. poglavlju, "Početak u Apacheu", Apache rabi *demone*, koji se katkad zovu *djeca (potomci)*, istovremeno izvodeći i odgovarajući na upite. Iako je veličina tih demona promjenljiva, postoje ograničenja veličine i brzine rasta. Ta veličina je kritična; jedna od glavnih problema izvedbe starijih poslužitelja koji su izvršavali fork() sustavski poziv na svaki zahtjev bio je neograničeni ukupni broj istovremenih demona. Kada bi se glavna memorija zasitila, računalo bi se zablokiralo.

Drugi softver za poslužitelj omogućuje vam određivanje točnog broja procesa, s mogućnosti otkazivanja ako su potomci zauzeti kada stigne novi zahtjev. To nije baš najbolji način – ako se izvodi 30 potomaka, a potrebno je samo 5, to će usporiti izvedbu – ali ovim modelom uklanja se potreba zaštite od blokiranja.

Dakle, Apacheov model je početi s određenim brojem stalnih procesa i uvijek osigurati neki broj rezervnih procesa ukoliko se pojave istovremeni zahtjevi. Tako možete pokrenuti nekoliko procesa kako bi se održao minimalni broj rezervi. Ako imate više poslužitelja u mirovanju od najvećeg broja rezervi, suvišni se mogu onemogućiti. Postoji maksimalni broj procesa kako bi se zaštitilo računalo od blokiranja.

Algoritam za zaštitu od previše procesa treba smjestiti u /usr/local/apache/httpd.conf, a izgleda ovako:

```
StartServers 10
MinSpareServers 5
MaxSpareServers 10
MaxClients 150
```

Ti su brojevi podrazumijevani. Upute kažu da se pri pokretanju Apachea automatski pokreće 10 potomaka (StartServer), bez obzira na početno učitavanje. Ako su svi potomci "zauzeti", ostali se zahtjevi odbijaju. To zahtijeva najmanje pet (MinSpareServers), ali ne više od 10 (MaxSpareServers) slobodnih poslužitelja koji se bave preopterećenjem (tj, kada se pojavi puno zahtjeva ). Ta preopterećenja često uzrokuju pretraživači koji otvaraju odvojene TCP veze za svaku sliku na stranici u pokušaju da korisniku poboljšaju izvedbu, često na račun poslužitelja i mreže.

BILJEŠKA Te upute su dio osnovnog skupa osobina Apachea i trebaju biti dostupne u svakoj inačici Apachea.

Obično se postiže stabilan broj procesa potomaka, ali ako dolazi puno zahtjeva, možete dostići MaxClients granicu. Zahtjevi će tada čekati na red. Ako i dalje pristižu, korisnici će vidjeti poruku da je veza odbijena. No, i to je bolje od neograničenog broja istovremenih procesa, jer će poslužitelj stalno pokretati potomke, a nitko neće dobiti odgovor od njega.

Preporučuje se da ne prilagođavate MaxClients jer je 150 dovoljan broj za većinu sustava. No, ako želite provjeriti koliko zahtjeva mogu podnijeti vaši najjači procesori, možete postaviti veći broj MaxClients. Ako izvodite Web poslužitelj na računalu s ograničenom memorijom ili CPUom, postavljanje manje vrijednosti MaxClients ima smisla.

# Uporaba procesne datoteke

Višeprocesni model opisan u prethodnom odjeljku zahtijeva određenu komunikaciju između roditeljskog i procesa-potomka, pa je odabrana metoda procesne datoteke gdje svaki potomak ima komadić prostora u datoteci u koji može pisati. Roditeljski httpd proces iz te datoteke dobiva izvješće o statusu i donosi odluke hoće li pokrenuti još procesa potomaka ili zaustaviti mirujuće procese.

U početku je to bila datoteka smještena u /tmp direktoriju. Zbog problema s postavkama Linuxa kada se brišu /tmp direktoriji, procesna je datoteka smještena u /var/log/ direktorij. Uz pomoć instrukcije ScoreBoardFile možete odrediti mjesto te datoteke.

Program httpd\_monitor u poddirektoriju support/ u Apacheu može pokrenuti procesnu datoteku kako bi dobio sliku o stanju procesa potomaka, jesu li oni tek na početku, aktivni, miruju ili ne rade. To vam može pokazati jesu li vaše postavke za MaxSpare Servers i MinSpareServers u redu. Možete ga smatrati sličnim sustavskoj naredbi iostat.

# Povećavanje uspješnosti softvera za poslužitelj

Standardno izvršenje možete poboljšati na puno načina, uključujući pametnije oblikovanje resursa, osobina koje se mogu isključiti, čak i elemente na razini operativnog sustava i hardvera koji se mogu adresirati. Svi ti činitelji pridonose razlici između običnog Web poslužitelja i Web poslužitelja visokih performansi.

Većina poboljšanja koja nisu povezana s hardverom mogu se grupirati u tri kategorije: ona koja smanjuju učitavanje u CPU, ona koja smanjuju iznos ulazno-izlaznih operacija na disku i ona koja smanjuju zahtjeve za memorijom.

## SSI pretprocesiranje

Već smo pisali o pretprocesiranju. CPU mora analizirati datoteku tražeći pretprocesorske instrukcije; analiza datoteke je detaljnija od čitanja datoteke.

Problemi u pristupu disku javljaju se zbog dva, tri, četiri ili više odvojenih pristupa disku kako bi se sakupile sve stranice. Na primjer, tipični SSI dokument može imati zaglavlje i podnožje koje treba skupiti zajedno u memoriji. To znači da treba tri puta pristupiti disku. Ako su HTML datoteke velike, razlika nije tako očita. No, HTML datoteke su obično male, pa je kašnjenje zbog pristupa disku relativno veliko. Problem je isti i kod CGI skripti; ako imate SSI stranicu s dvije CGI skripte, vjerojatno ćete dva puta pristupati disku.

# Uporaba .htaccess datoteka

Apache za kontrolu pristupa direktorijima rabi posebne datoteke - .htaccess. Te datoteke rade hijerarhijski. Kada se pojavi zahtjev za /path/path2/dir1/dir2/foo, Apache će potražiti .htaccess datoteku u *svakom* poddirektoriju. U navedenom zahtjevu to je najmanje pet poddirektorija – što predstavlja značajno opterećenje diska koje je najbolje izbjeći.

Želite li izbjeći preveliki broj pristupa disku, trebate sve što kontrolirate preko .htaccess datoteka staviti u access.conf konfiguracijsku datoteku ili čak u srm.conf datoteku. Trebate li potražiti .htaccess datoteke u poddirektorijima, a možete to suziti na određeni poddirektorij, uz pomoć AllowOverride možete postići da poslužitelj traži .htaccess datoteke u tom poddirektoriju.

Pretpostavimo da je vaš korijenski direktorij dokumenta u /wwwĐhtdocs i da želite isključiti pretraživanje svih .htaccess datoteka osim onih u /www/htdosc/dir1/dir2. U access.config konfiguracijsku datoteku ćete upisati:

```
<Directory /www/htdocs>
Options All
AllowOwerride None
</Directory>
<Directory /www/htdocs/dir1/dir2>
Options All
AllowOwerride All
</Directory>
```

Važan je redoslijed direktorija tako da drugi <Directory> ne bude ispred prvog.

#### Uporaba .asis datoteka

.asis datoteke razlikuju se prema HTTP zaglavljima koja su izravno umetnuta u samu datoteku. Ona su korisna optimizacija određenih vrsta datoteka, kao što su s poslužitelja gurane animacije koje zahtijevaju sposobnost postavljanja vlastitih zaglavlja i obično

ih uništavaju CGI skripte. Obična gurana CGI skripta ima dodatne troškove dinamičkog slaganja slika, dok se u .asis datoteci sve može povezati u jednu datoteku, smanjujući ulazno-izlazno optererećenje i zauzeće memorije i CPU-a.

Vidi "As-is datoteke", str 701

Jedino što gubite uporabom .asis parametra je sposobnost vremenski raspoređeno guranje, gdje postoji vremenski razmak između okvira umetnutih kao sleep() (sustavski poziv gdje program zastane definirani broj sekundi). No kako guranje ima ograničenu širinu pojasa, smatra se da je vremenska uvjetovanost dvosmislena osobina.

# Automatizacija rotacije prijavne datoteke

Jedan od ciljeva administratora lokacije treba biti automatizacija rotacije prijavne datoteke i datoteke s podacima o pogreški. Čak i slabo opterećeni poslužitelj generirat će nekoliko megabajta prijava tijekom dana.

Osnovni element ove rotacije je da Web poslužitelj prestane pisati u staru datoteku prijave i počne pisati u drugu bez prekidanja usluge drugim korisnicima. Najbolji je način *nez-natno preimenovanje* prijave i slanje SIGHUP signala roditeljskom procesu. To znači preimenovanje u access\_log.0 ili slično na istom disku, na istoj podjeli. Zašto? Svaki potomak ima opis datoteke otvoren u datoteci s podacima o prijavi. Kada preimenujete datoteku, opisni blok datoteke i dalje će pokazivati na istu stvarnu prijavu sve dok potomak ne primi "eho" SIGHUP od roditeljskog procesa. Kada se to dogodi, opisni blok datoteke se zatvara, javlja se novi i stvara novi access\_log. To je jedini način da se ne izgube izvješća o prometu na poslužitelju dok rotirate prijavu.

Evo primjera skripte:

Ova se skripta treba izvoditi kao i korisnik koji je prvobitno pokrenuo HTTP deemon – na primjer "korijenski direktorij". Možete napisati dodatne skripte kako biste smjestili te .0 datoteke u neku arhivu; ja rabim godine i mjesece kao poddirektorije, npr. prijave za 1. siječanj 1997 idu u datoteku 1997/01/01 izvan direktorija gdje ima puno mjesta. Tako se datoteke s podacima o prijavi micanjem direktorija mogu smjestiti negdje drugdje.

# 0 sigurnosti

Sigurnost poslužitelja je sigurno najveća briga administratora Web lokacije. Izvođenje Web poslužitelja je sigurnosni rizik, a također i rad na mreži. No, vaš Web poslužitelj može biti siguran od vanjskih utjecaja (ako ljudi pokušaju provaliti na vašu lokaciju) i unutrašnjih utjecaja (korisnici vaše Web lokacije pogreškom ili namjerno otvaraju sigurnosne rupe).

# **CGI** skripte

Većina CGI skripti se temelje na ljuski i rabe Perl ili interpretirane programe C ljuske, a ne kompilirane programe. Zato se dogodilo puno napada pri korištenju "osobina" u tim ljuskama. U ovom odjeljku govorit ćemo o nekim osnovnim stvarima koje se tiču sigurnosti CGI skripti.

CGI skripta izvodi se s korisničkim ID-om. Podrazumijevano je to "nobody". Želite li se zaštititi, takav korisnik nije povjerljiv, pa on neće imati odobrenje za čitanje vaših privatnih datoteka, kao ni dopuštenje za pisanje u njih. Neke CGI skripte – na primjer, knjiga posjetitelja gdje korisnici mogu pisati komentare o vašoj Web lokaciji – zahtijevat će pristup s mogućnosti pisanja u neke datoteke. Zato je najbolje odrediti direktorij u koji CGI skripte mogu pisati bez ikakvih problema.

Nadalje, administratori lokacije mogu ograničiti uporabu CGI na određene direktorije uporabom upute ScriptAlias. Ako imate .cgi nastavak za CGI datoteke, za bolju kontrolu uporabe CGI datoteka možete uporabiti uputu OptionsExecCGI u access.conf.

U slijedećem primjeru, ispis 37.1, imate kontrolu pristupa uz pomoć ExecCGI, ukoliko želite omogućiti da se CGI rabi svugdje na lokaciji (korijenski direktorij dokumenata /home/htdocs) osim za "korisničke" poddirektorije.

#### Ispis 37.1 Primjer access.conf datoteke koji prikazuje informacije o oblikovanju direktorija

```
<Directory /home/htdocs/>
Options Indexes FollowSymLinks Includes Multiviews ExecCGI
AllowOverride None
</Directory>
<Directory /home/htdocs/users/>
Options Indexes SymLinksIfOwnerMatch IncludesNOEXEC Multiviews
AllowOverride None
</Directory>
```

Kako ExecCGI nije na popisu Options za drugi direktorij, nitko tu ne može uporabiti CGI skripte.

Na žalost, ne postoji sredina između omogućavanja i onemogućavanja CGI skripti. Većina jezika koja se rabi za CGI programe nemaju ugrađene sigurnosne mjere. Zato pravila poput "ne diraj tvrdi disk" ili "ne šalji /etc/passwd datoteku u e-poštu izvan korisnika" trebaju biti ista kao da imate stvarne korisnike Linuxa na koje trebate primijeniti ista ograničenja. To će se možda u budućnosti promijeniti.

# Pretprocesiranje

Iz ispisa 37.1 vidite još jednu promjenu *povjerljivog* dijela poslužitelja i **nesigurnog**: argument Includes u Options promijenjen je u IncludesNOEXEC. Ta uputa omogućuje vašim korisnicima u koje nemate povjerenja uporabu pretprocesorskih instrukcija bez izvođenja #include iz CGI skripte ili #exec naredbe. Naredba #exec je posebno nezgodna u nesigurnoj okolici jer autoru HTML-a daje pristup kroz ljusku.

# Simboličke veze

U nesigurnoj okolici UNIX-ove *simboličke veze* (omogućuju povezivanje preko granica datotečnog sustava) također mogu zabrinuti administratore Web lokacija. Zlobni korisnici lako mogu napraviti simboličke veze iz direktorija gdje smiju pisati do objekta ili resursa, čak izvan korijenskog direktorija dokumenata. Na primjer, korisnik može napraviti vezu prema /etc/passwd datoteci, a potom je poslati na Web izlažući vašu lokaciju potencijalnim provalama – posebno ako vaš sustav ne rabi rezervne lozinke.

**BILJEŠKA** Na Alta Vista pretraživačkom servisu (**www.altavista.digital.com**) pretraživanje po riječima koje imaju datoteke lozinki (*bin, root, FTP* itd.) preokrenulo je upute na stvarne datoteke lozinki koje su ostavljene u javnosti. Te datoteke lozinki imaju šifrirane lozinke koje je lako razbiti za nekoliko sati.

Za zaštitu, administrator lokacije ima dvije mogućnosti: dozvoliti simboličku vezu ako su vlasnik veze i vlasnik resursa s kojim se povezuje isti pomoću SymLinksOwnerMatch ili onemogućiti simboličke veze tako da se ne aktivira FollowSymLinks ili SymLinksIfOwnerMatch.

Uočite da oba <Directory> dijela ispisa 37.1 imaju AllowOverride None. Onemogućavanje simboličkih veza je najsigurnija postavka; ako ipak želite omogućiti usklađivanje nečeg u tim direktorijima, možete to odrediti uputom AllowOverride.

## Prostori u koje se može pisati

Posljednja prijetnja sigurnosti za Web poslužitelje je posluživanje prostora gdje se može pisati preko HTTP-a. Na primjer, puno lokacija omogućuje pristup FTP direktoriju izravno preko Weba. To je sigurnosna rupa ukoliko netko tamo stavi zločestu CGI skriptu koja poziva #exec da napravi neku štetu. Ako vam treba prostor gdje se može pisati, možete se zaštititi na ove načine:

Postavku za instrukciju Options napravite ovako:

Option Indexes

Možete uprabiti i None, ali Indexes ne uzrokuju nikakve sigurnosne probleme tako dugo dok vam ne smeta da drugi mogu učitati sve što pošaljete.

- Postavite AllowOverride None tako da ljudi ne mogu poslati .htaccess datoteku u vaš direktorij i promijeniti vaše postavke.
- Pripazite da FTP demon koji rabite ne dozvoljava postavljanje izvršnog bita. Tako sprečavate izvršenje poslanih CGI skripti. Ako za aktiviranje pretprocesnih instrukcija rabite XBitHack, također možete spriječiti njihovo izvođenje. To je uglavnom sigurnosna pohrana podataka za postavljanje Options kao na ispisu 37.1, što bi vas trebalo zaštititi od prijetnji.

Ista pravila se primjenjuju ako imate CGI skripte koje proizvode svoje HTML ili CGI datoteke. Na primjer, program guestbook.cgi stalno dodaje poslane osobne informacije u datoteku guestbook.html; zato se sadržaj te HTML datoteke na smatra sigurnim. Ako se CGI skripta dvostruko provjerava i otklanja "opasan" kôd, sigurnost je veća.

# Ostale teme vezane uz prilagođavanje

Apache Web poslužitelj više je podređen sigurnosti nego brzini; prema izreci "Možeš promašiti brže nego što druga osoba može pucati". Izvedba je također u redu, jer svaki Web poslužitelj može lako zasititi T1 liniju.

Oni koje zanimaju najnovije mogućnosti poslužitelja mogu ugoditi svoja računala tako da poboljšaju sposobnost odziva. Njajeftinije poboljšanje za svaki Web poslužitelj je dodavanje RAM-a. Treba izbjegavati da Web poslužitelj mora tražiti zamjensku memoriju izvan diska. Jedan od načina izbjegavanja je ograničavanje broja MaxClients unutar dostupnog RAM-a. Najbolji izvor informacija o prilagođavanju izvedbe je na Web lokaciji **http://www.apache.org/docs/misc/perf-tuning.html**.

# Odavde...

Možete nučiti više o postavkama, oblikovanju i izvođenju Apache Web poslužitelja u 35. poglavlju "Početak u Apacheu". U tom poglavlju je detaljan uvod u Apache Web poslužitelj.

#### **D O D A T A K**



# Izvori informacija

(Jack Tackett)

# U ovom poglavlju

Linux Web lokacije Novinske grupe na Usenetu Online dokumentacija HOWTO dokumenti za Linux Časopisi **726** Linux FTP lokacije Za Linux programere Zahtijevani hardver Kako se Linux temelji na UNIX-u, praktički svaka knjiga o UNIX-u pruža neke informacije i o Linuxu. Najbolji je izvor informacija ipak sama zajednica Linux korisnika - ona će vam pružiti kako nove verzije Linuxa, tako i vrlo aktivne Usenet grupe. Linux nudi i online dokumentaciju kroz Linux Documentation Project (LDP) - cjelovite priručnike za Linux. Najnovija izdanja ovog projekta dostupna su putem Interneta.

Sljedeći popisi nižu Internet FTP lokacije, časopise, konferencije i novinske grupe iz kojih možete prikupiti dodatne informacije o Linuxu.

# Linux Web lokacije

S obzirom na činjenicu da je Linux dijete Interneta, na njemu ćete naći niz Web stranica s vrijednim informacijama. Linux je doista popularna tema na Internetu. Tablica A.1 popisuje adrese na Internetu na kojima ćete naći najveći dio potrebnih informacija.

Tablica A.1 Glavne Linux Web lokacije		
URL	Opis	
http://sunsite.unc.edu/mdw	Prava lokacija za informacije o Linuxu i sjedište LDP-a	
http://www.Linux.org.uk	Web lokacija za evropske korisnike Linuxa	
http://www.li.org	Međunarodna Linux Web lokacija	
http://www.redhat.com	Web lokacija tvrtke RedHat	
http://www.slackware.org	Službena Web lokacija Slackwarea	
http://www.caldera.com	Calderina Web lokacija	
http://www.linux.org	Web lokacija Linux Organization	
http://sunsite.unc.edu/linux-source	Linux Source Navigator - pregled Linuxovog izvornog koda	

# Novinske grupe na Usenetu

Ako imate pristup Usenetovim novinskim grupama, uživat ćete u onima koje ćemo navesti jer pružaju niz različitih informacija i diskusija o Linuxu. Moderirane su samo dvije od njih: **comp.os.linux.announce** i **comp.os.linux.answers**.

Vidi "Kultura ponašanja na Usenetu", str. 661

- BILJEŠKA Izvorna novinska grupa koja se odnosi na Linux, comp.os.linux, više ne postoji jer su otvorene nove grupe koje pružaju veću detaljnost.
  - comp.os.linux.announce Moderirana novinska grupa koja se koristi za važne najave, poput ispravaka nekih pogrešaka i slično.
  - comp.os.linux.answers Moderirana novinska grupa koja pruža odgovore na mnoga pitanja, posebno o instalaciji i podešavanju Linuxa. Prije nego što u ovu grupu pošaljete svoje pitanje, odvojite malo vremena i pogledajte Linux dokumentaciju i odgovore na često postavljana pitanja (FAQ).
  - comp.os.linux.development.system Novinska grupa koja je posvećena programerima širom svijeta koji razvijaju Linux.
  - comp.os.linux.development.apps Novinska grupa posvećena mnogim programerima širom svijeta koji razvijaju aplikacije za Linux.
  - comp.os.linux.hardware U ovoj ćete grupi pronaći odgovore na hardverske uređaje uskladive s Linuxom.
  - comp.os.linux.setup Novinska grupa koja će vam pomoći ako imate problema s instalacijom i podešavanjem Linuxa.
  - comp.os.linux.advocacy Odlična grupa za sudjelovanje u diskusijama vezanim uz objašnjenja zašto je Linux najbolji operativni sustav.
  - comp.os.linux.networking Novinska grupa s informacijama o mrežnom povezivanju Linuxa s ostatkom svijeta.
  - comp.os.linux.x Novinska grupa koja pruža informacije o Xwindows sustavu.
  - comp.os.linux.m68k Namjera je ove novinske grupe promovirati interes za prebacivanje Linuxa na Motorolinu seriju procesora 680x0.

Ako su vam potrebne informacije koje ne potpadaju pod jednu od navedenih grupa, pokušajte na comp.os.linux.misc. Isto tako, riječ Linux u svome opisu sadrži još 170 različitih novinskih grupa. Neke od njih su:

alt.linux.sux	alt.os.linux
alt.uu.comp.os.linux.questions	alt.os.linux.slackware
aus.computers.linux	dc.org.linux-users
de.comp.os.linux.hardware	de.comp.os.linux.misc
de.comp.os.linux.networking	de.comp.os.linux.x
de.alt.sources.linux.patches	uk.comp.os.linux
fj.os.linux	fr.comp.os.linux
han.sys.linux	linux.apps.bbsdev
linux.apps.linux-bbs	linux.apps.seyon
linux.apps.seyon.development	linux.apps.flexfax
linux.debian	linux.debian.announce

linux.debian.user	linux.dev.gcc
linux.dev.680x0	linux.dev.admin
linux.dev.apps	linux.dev.bbs
linux.dev.c-programming	linux.dev.config
linux.dev.debian	linux.dev.doc
linux.dev.fido	linux.dev.fsf
linux.dev.fsstnd	linux.dev.ftp
linux.dev.hams	linux.dev.ibcs2
linux.dev.interviews	linux.dev.japanese
linux.dev.laptop	linux.dev.linuxbsd
linux.dev.linuxnews	linux.dev.linuxss
linux.dev.localbus	linux.dev.lugnuts
linux.dev.mca	linux.dev.mgr
linux.dev.msdos	linux.dev.net
linux.dev.new-lists	linux.dev.newbie
linux.dev.normal	linux.dev.nys
linux.dev.oasg	linux.dev.oi
linux.dev.pkg	linux.dev.ppp
linux.dev.qag	linux.dev.scsi
linux.dev.serial	linux.dev.seyon
linux.dev.sound	linux.dev.standards
linux.dev.svgalib	linux.dev.tape
linux.dev.term	linux.dev.uucp
linux.dev.wabi	linux.dev.word
linux.dev.kernel	linux.dev.x11
linux.fido.ifmail	linux.free-widgets.anounce
linux.free-widgets.bugs	linux.free-widgets.development
linux.local.chicago	linux.local.nova-scotia
linux.local.silicon-valley	linux.motif.clone
linux.new-tty	linux.news.groups
linux.ports.alpha	linux.samba
linux.samba.announce	linux.sdk
linux.wine.users	linux.test

# **Online dokumentacija**

Grupa entuzijasta stalno radi na pisanju priručnika za Linux i čini ih dostupnim na Internetu. Zadnje verzije dokumenata mogu se naći na sunsite.unc.edu u direktoriju /pub/Linux/docs. Ranije verzije dokumenata možete naći i na svom sustavu, u direktoriju /docs. Trenutno sjedište LDP-a je na adresi:

#### http://sunsite.unc.edu/mdw

Dostupni dokumenti uključuju:

- "Linux Installation and Getting Started", Matt Welsh
- "The Linux System Administrators' Guide", Lars Wirzenius
- "The Linux Network Administrators' Guide", Olaf Kirch
- "The Linux Kernel Hackers' Guide", Michael K. Johnson
- "The Linux Frequently Asked Questions (FAQ) List", Ian Jackson
- "The Linux META-FAQ", Michael K. Johnson
- "The Linux INFO-SHEET", Michael K. Johnson
- "The Linux Software Map", Aaron Schab

## HOWTO dokumenti za Linux

Kazalo HOWTO dokumenata popisuje sve dostupne HOWTO dokumente u kojima ćete naći detaljna objašnjenja o odgovarajućoj temi. Neki od naslova su:

- The Linux Installation HOWTO (instalacija Linuxa)
- The Linux Hardware HOWTO i The Linux Printing HOWTO (hardver i ispis)

Datoteke HOWTO dokumenata možete pronaći i na svom sustavu, u direktoriju /usr/doc/faq/howto. Mnogi od njih su komprimirani pomoću gzipa kako bi se uštedio prostor na disku. Kako biste pročitali takve dokumente, upotrijebite naredbu zless.

Najnovije verzije HOWTO dokumenata možete pronaći na FTP lokacijama:

ftp://sunsite.unc.edu/pub/Linux/docs/HOWTO

#### ftp://tsx-11.mit.edu/pub/linux/docs/HOWTO

Mnoga često postavljana pitanja i odgovore na njih možete pronaći u direktoriju /usr/info.

#### man stranice

Operativni sustav Linux pruža i odličnu ugrađenu pomoć kojoj ćete pristupiti pomoću naredbe *man*. Kako biste otvorili ugrađeni sustav pomoći, unesite *man* i nakon toga naziv ili temu o kojoj želite saznati više. Pogledajte 114. stranicu knjige.

Ovaj fajl je skinut sa sajta www.tutoriali.org
## Za Linux programere

Nakon čitanja ove knjige zaključili ste da je Linux upravo sjajan i da želite pridonijeti njegovom daljem razvoju. Imate sreće. Aktivni skup poštanskih popisa na Internetu posvećen je različitim temama koje se odnose na razvoj Linuxa. Radi se o višekanalnom poštanskom popisu, što znači da se poruke s različitim temama šalju različitim grupama zainteresiranih. Morate se pretplatiti na svaki kanal koji vas zanima. Ako se želite uključiti u čitav projekt razvoja Linuxa, više ćete informacija saznati ako pošaljete poruku e-pošte na adresu:

## majordomo@vger.rutgers.edu

s riječju lists u tijelu poruke ako želite dobiti popise, odnosno riječju help ako želite dobiti standardnu datoteku pomoći u kojoj ćete saznati kako se pretplatiti i kako ukinuti pretplatu.

## Zahtijevani hardver

Kao što ste mogli pročitati još na početku knjige, postoji popis hardvera koji je uskladiv s Linuxom. Kako se taj popis brzo mijenja, a vjerojatno biste željeli najnoviju verziju, savjetujemo vam da na niže navedenoj Internet adresi pronađete najnoviji popis cjelokupnog hardvera kojeg Linux podržava:

## http://users.bart.nl/~patrickr/hardware-howto/Hardware-HOWTO.html