

Osnove rada na
UNIX
OPERATIVNIM SISTEMIMA
(od login do logout)

Autor:
Srđan Rajčević
s.rajcevic@city.academic.gr
City Liberal Studies
University of Sheffield

Sadržaj

1. UVOD	3
1.1 ISTORIJA I UNIX-A	3
1.2 DALJNJA EVOLUCIJA I POJAVA RAZNIH VERZIJA	4
1.3 ŠTA SADRŽI JEDAN UNIX SISTEM ?	5
2. KORISNIČKO OKRUŽENJE	6
2.1 PRIJAVLJIVANJE NA SISTEM	6
2.2 OSNOVNE KOMANDE	7
2.3 FAJLOVI I DIREKTORIJUMI	9
2.3.1 ORGANIZACIJA UNIX FILE SISTEMA	9
2.3.2 PREUSMERAVANJE	10
2.3.3 PRETRAGA ZA ODREĐENIM TEKSTOM – GREP	11
2.3.4 RAD SA ARHIVAMA – TAR, GZIP, RPM	11
2.3.5 KOMPILIRANJE PROGRAMA	13
2.4 RAD U MREŽNOM OKRUŽENJU	14
2.4.1 PRENOS FAJLOVA – FTP	14
2.4.2 SURFANJE PO MREŽI - LYNX	15
2.4.3 RAZGOVOR U ŽIVO – IRC	15
2.4.4 ČITANJE I SLANJE E-MAILA – MAIL	16
2.4.5 KONEKTOVANJE NA UDALJENI SISTEM – TELNET	17
2.5 UNIX PROGRAMIRANJE	18
2.5.1 JAVA	18
2.5.2 PERL	18
2.5.3 TCL	19
2.5.4 PYTHON	19
2.5.5 C/C++	19
2.6 RAD SA PROGRAMIMA	21
2.6.1 EKRANSKI EDITOR PICO	21
2.6.2 MAIL KLIJENT PINE	22
2.7 ODJAVA SA SISTEMA	23
O AUTORU	24
KORIŠTENA LITERATURA	25

1. Uvod

1.1 Istorijat UNIX-a

UNIX je jednostavan, ali ipak moćan višekorisnički operativni sistem prvenstveno dizajniran za interaktivnu upotrebu. Iako ne sasvim portabilan, u smislu da se izvršni programi mogu prenositi sa jedne platforme (arhitekture procesora) na drugu, dostupan je za skoro sve tipove računara u današnjoj upotrebi. U UNIX svetu i ovaj problem je rešiv korišćenjem raznih alata i biblioteka koji dolaze uz UNIX servise.

Istorija ovog sistema je veoma bitan faktor u procesu evolucije računara i računarstva uopšte. Većina operativnih sistema koje danas susrećemo u svakodnevnoj upotrebi manje ili više koristi filozofiju UNIX-a i neka njegova idejna rešenja. Možemo čak reći da koncept interaktivnog korisničkog okruženja nikad ne bi bio primenjiv da nije bilo tada pionirskog rada na nečemu što se zvalo UNIX.

Prva verzija UNIX-a je kreirana u Bell System laboratorijama tokom 1969. i 1970. godine od strane Ken Thompsona, programera čija je ideja bila da napravi sistem koji će mu biti od koristi za njegov rad. Ideje korištene za dizajn su preuzete iz njegovog ranijeg projekta koji je radio za MIT (Massachusetts Institute of Technology) nazvanog Multics. Drugi Bell istraživač, Brian Kernighan je novi sistem nazvao UNIX i, evolucija je počela.

Thompson je kreirao prvu verziju na Digital PDP-7 računaru. Zbog činjenice da je PDP-7 mali računar (uobičajena konfiguracija u to vreme se sastojala od 32K memorije, 320K hard diska i minijaturnog sistema za arhiviranje koji je koristio traku kao medij za smeštanje podataka), operativni sistem je morao biti mali i efikasan. Prvi UNIX je napisan u PDP-7 mašinskom jeziku.

U to doba, pojam “personalni računar” nije bio poznat. Uopšte gledano, rad sa računarom je bio izuzetno težak. Iako su velike radne stanice bile opslužene sa nekom vrstom korisničkog okruženja koje je bilo lako razumeti, mali računari su imali primitivne operativne sisteme (ako su ih uopšte i imali) i nisu bili dizajnirani za interaktivnu upotrebu. Bell Laboratories su ubrzo shvatili od kolikog je značaja ovaj Thompsonov projekat, jer su u mogućnosti izdavanja jednog operativnog sistema za personalnu upotrebu videli veliku materijalnu korist.

Thompsonu se ubrzo pridružio Denis Ritchie, još jedan Bell istraživač sa projekta Multics. Njegove zasluge za dalji razvoj sistema su ogromne, jer je on kreirao novi jezik C koji je trebao biti upotrebljen za lako prebacivanje sistema sa jedne platforme na drugu (tzv. *porting*). Njih dvojica su zajedno sve ispočetka napisali u C-u i portovali sistem na malo moderniju mašinu – Digital PDP-11. Iako je prvobitni UNIX zamišljen kao single-user sistem, sa razvojem na PDP-11 je dobijena nova ideja – multikorisnički sistem, koja je ubrzo i realizovana. PDP-

11 je za razliku od zastarelog PDP-7 omogućavao uvođenje podrške za multiprocessing i multitasking (mogućnost kreiranja više procesa u isto vreme). Ove promene su proslavile UNIX unutar Bell laboratorija, a PDP-11 je uveden u mnoge škole, koledže i univerzitete širom USA. AT&T je besplatno dostavio kod UNIX-a za mnoge programere, pa se sistem ubrzo proširio, tako da je do kraja 1978. u upotrebi bilo više od 600 instalacija UNIX-a.

1.2 Daljnja evolucija i pojava raznih verzija

Računarski stručnjaci sa Univeziteta Kalifornija iz Berkeleya su kreirali alternativnu verziju UNIX-a, nazvanu BSD (Berkeley Software Distribution) koja je koristila sopstvene ljuske (shells). Zbog činjenice da je UNIX napisan u C-u, bilo je relativno lako integrisati nove ideje i portovati sistem na različite tipove računara, tako da je BSD postala izuzetno popularna distribucija, čak i u komercijalnim poslovima, npr. za kompanije Sun i Hewlett-Packard. BSD izdanja su unela mnogo novina, kao npr. *vi* editor, C ljosku, prvu UNIX podršku za virtuelno spremanje podataka, dugačke nazive fajlova, mnogo aplikativnog softvera i ono što je najvažnije, integrisan je TCP/IP protokol koji dominira današnjim mrežama i koji će učiniti od UNIX-a ono što danas jeste – najrasprostranjenijim i najpuzdanijim mrežnim sistemom.

Finski student Linus Torvalds je imao nameru da se bolje upozna sa onim što se dešava “ispod haube” operativnog sistema. Model na kome se učio je bio MINIX, mala akademska varijanta UNIX-a kreirana kao alat koji je dolazio uz knjigu koja se bavila operativnim sistemima. Do novembra 1991. Linus je završio radnu verziju UNIX baziranog kernela koji je postavio na Internet pod imenom Linux, a platforma koju je koristio za rad bila je x86.

Torvalds je pozvao i ostale programere zainteresovane za Linux da mu se priključe. Od tada, hiljade dobrovoljaca su učestvovali u izradi dizajna, kodiranju, testiranju i otklanjanju grešaka. U Linux su integrisane podrške za ogroman broj uređaja, portovan je i na druge platforme, Usenet forumi su postali okupljalište zainteresovanih za Linux a istovremeno se dobijala i tehnička pomoć, pisani su prva uputstva za instalaciju i rad sa Linux-om. Iako je Linux deklarisan kao besplatan (Open Source) softver, postoje mnoge komercijalne distribucije koje naplaćuju uglavnom svoju tehničku podršku, i aplikacije iz domaće proizvodnje.

Takođe, bilo je mnogo nastojanja da se sve verzije UNIX-a spoje u jednu, ali do danas je učinjeno malo toga na standardizaciji, tako da se možemo susresti sa velikim brojem distribucija od kojih svaka sadrži svoje prednosti i mane. Kao dodatak Open Group standardu, set IEEE pravila nazvan POSIX je donekle uspeo da označi minimalan skup komandi, programa i servisa da bi se osigurala kompatibilnost između različitih sistema. Iako POSIX ne uključuje neke finese pojedinačnih distribucija, on ipak garantuje da će program razvijen po standardu moći da se bezbolno prenese i izvrši na drugom sistemu.

1.3 Šta sadrži jedan UNIX sistem ?

UNIX ili Linux sistem danas se može opisati pomoću sledećih karakteristika:

- Podrška za multitasking, sa automatskim raspoređivanjem i izvršavanjem pozadinskih procesa.
- Jedna ili više ljuski koje obezbeđuju interaktivno i za korisnike prijateljsko okruženje sa jednostavnim, ali ipak moćnim i fleksibilnim komandama, alatima, i programskim jezicima koji kombinovani mogu da izgrade programe nazvane skripte (shell scripts).
- Podrška za kreiranje i interaktivno izvršavanje novih procesa, mogućnost kontrole istih, kao i mogućnost određivanja prioriteta u izvršavanju, memorijskih lokacija i ostalog.
- Logično i jednostavno uređen fajl sistem koji podržava i lokalni i pristup preko mreže, sa mogućnošću zaštite fajlova na dva nivoa – nivou grupe i korisnika.
- Jezgro (kernel) koji pruža pristup hardveru na jednostavan i elegantan način, uključujući podršku za virtuelno spremanje podataka i deljenje resursa.

2. Korisničko okruženje

2.1 Prijavljivanje na sistem

Kao što smo naglasili, UNIX je višekorisnički sistem pa prema tome mora sadržati način kako da identificira pojedinačnog korisnika. Svaki korisnik na UNIX-u mora da poseduje svoje korisničko ime (username) i lozinku (password). Ono sa čime se svaki korisnik mora susresti pri prijavljivanju na sistem je tzv. login prompt čime sistem stavlja do znanja da je spreman za proces prijave, a on otprilike izgleda ovako

```
Welcome to ZEUS.rsrđjan.com

zeus login: rsrdjan
Password:

Last login: Thu Feb 24 16:55:04 +0200 2002 on tty1.
No mail.
rsrdjan@zeus:~$ _
```

Sistem prvo očekuje da se unese tačno korisničko ime a zatim i lozinka. Ako su podaci ispravno uneseni i ako je proces verifikacije protekao bez problema, naćete se u ljusti i sa svojim promptom sistem vam stavlja do znanja da je spreman za rad. U suprotnom, dobićete poruke o netačnom unosu. Ne budite zbunjeni pri unosu lozinke jer ne vidite karaktere, to je jedna od mera predostrožnosti.

Uobičajeno UNIX okruženje pri prijavi je tzv. komandna linija, ali start-up rutina u mnogim modernim UNIX sistemima omogućava i automatsko podizanje X Windows-a, odnosno grafičkog okruženja (GUI – Graphical User Interface). Ipak, mnogim iskusnim UNIX korisnicima jedini vid “potpune kontrole” nad sistemom predstavlja komandni prompt. Ono što daje prepoznatljivost i moć UNIX-u je tzv. remote prijavljivanje, odnosno prijavljivanje i rad na udaljenom računaru preko mreže.

Na tipičnom sistemu, korisnik ima mogućnost izbora nekoliko ljusti. Najpoznatije UNIX ljuste su Bourne shell, C shell i Korn shell, ali Linux npr. podržava i set sličnih ljusti – *bash*, *tosh* i *pdksh*, tako da možete izabrati ljustu koja vam najviše odgovara. Takođe, sve ljuste sadrže i tzv. start-up fajlove koji se izvršavaju odmah nakon prijavljivanja, a u kojima možete podesiti i urediti vaše okruženje.

2.2 Osnovne komande

Svaki UNIX sistem sadrži komande za rad sa fajlovima, alate za sortiranje, programe za rad u mrežnom okruženju - čitanje e-maila, web pretraživanje itd. Ono što sledi je tabela sa osnovnim komandama koje svaki UNIX sadrži:

Rad sa fajlovima i direktorijumima		Rad sa tekstom i dokumentima	
<i>cat</i>	Ispis tekstualnih fajlova	<i>emacs</i>	Screen editor
<i>cd</i>	Promena aktivnog direktorijuma	<i>lp</i>	Štampanje teksta
<i>chmod</i>	Određivanje dozvole pristupa	<i>more</i>	Prikaz stranica jednu po jednu
<i>cp</i>	Kopiranje fajlova	<i>nroff</i>	Formatiranje teksta za prikaz ili štampu
<i>ln</i>	Pridruživanje alias-a fajlu	<i>spell</i>	Provera spelovanja
<i>ls</i>	Izlistavanje sadržaja direktorijuma	<i>tbl</i>	Table preprocessor
<i>mkdir</i>	Kreiranje novog direktorijuma	<i>vi</i>	Screen editor
<i>mv</i>	Premestanje ili promena imena fajla ili direktorijuma	<i>wc</i>	Prebrojavanje reči, linija i karaktera
<i>pwd</i>	Ispis aktivnog direktorijuma	Alati za programiranje	
<i>rm</i>	Brisanje fajlova	<i>cb</i>	Formatiranje C izvornog koda
<i>rmdir</i>	Brisanje direktorijuma	<i>cc</i>	C kompajler
Komande za pretragu i sortiranje		<i>g++</i>	C++ kompajler
<i>find</i>	Traženje fajlova u sistemu	<i>lint</i>	Provera kompatibilnosti C izvornog koda
<i>grep</i>	Traženje određenog teksta u tekstualnim fajlovima	<i>make</i>	Ažuriranje fajlova za povezivanje i kompajliranje
<i>sort</i>	Sortiranje ili spajanje fajlova po redovima	<i>sdb</i>	Symbolic debugger
<i>strings</i>	Traženje određenog teksta u binarnim fajlovima		
Rad u mrežnom okruženju		Opšte komande za rad sa sistemom	
<i>ftp</i>	Prenos fajlova između sistema	<i>at</i>	Izvršavanje komandi u određeno vreme
<i>login</i>	Prijava na lokalni sistem	<i>cpio</i>	Kopiranje fajlova u/iz arhive
<i>mail</i>	E-mail	<i>crontab</i>	Automatsko izvršavanje komandi
<i>rlogin</i>	Prijava na udaljeni sistem	<i>date</i>	Prikaz ili podešavanje datuma
<i>talk</i>	Razgovor sa korisnicima	<i>du</i>	Prikaz iskorištenosti diska
<i>telnet</i>	Konektovanje na udaljeni sistem	<i>finger</i>	Informacija o korisnicima
<i>write</i>	Slanje poruka korisnicima	<i>kill</i>	Gašenje aktivnog procesa
Razne komande		<i>nice</i>	Podešavanje prioriteta izvršavanja
<i>bc</i>	Kalkulator	<i>nohup</i>	Izvršavanje procesa i posle odjave sa sistema
<i>cal</i>	Kalendar	<i>passwd</i>	Promena lozinke
<i>man</i>	On-line pomoć	<i>tar</i>	Arhiver
<i>news</i>	Čitanje on-line novosti	<i>who</i>	Prikaz prijavljenih korisnika
		<i>su</i>	Prijavljivanje kao root

Većina gore navedenih komandi zahteva dodatne parametre. U slučaju da ne znate šta da unesete, uvek možete otkucati *man ime_komande* da bi ste dobili kratak opis.

Da bi smo prikazali kako sve radi u praksi, uzećemo jedan primer. Recimo da želimo da promenimo svoju lozinku na sistemu *zeus*

```
rsrdjan@zeus:~$ passwd
Changing password for rsrdjan
Old password:
New password:
Password changed.
rsrdjan@zeus:~$ _
```

Komandom *passwd* pozvali smo program koji nas prvo pita za staru lozinku a onda traži da unesemo novu. Ako smo sve uneli kako treba, sistem ispisuje poruku da je lozinka uspešno promenjena.

NAPOMENA: Pažljivo radite sa komandom *passwd* i uvek imajte na umu da je vaša lozinka vaša privatna stvar. Pri promeni lozinke budite posebno obazrivi jer u slučaju da zaboravite novu lozinku nećete moći pristupiti sistemu. Dobar savet je da za odabir lozinke koristite kombinaciju malih i velikih slova i brojeva – radi vaše sigurnosti.

2.3 Fajlovi i direktorijumi

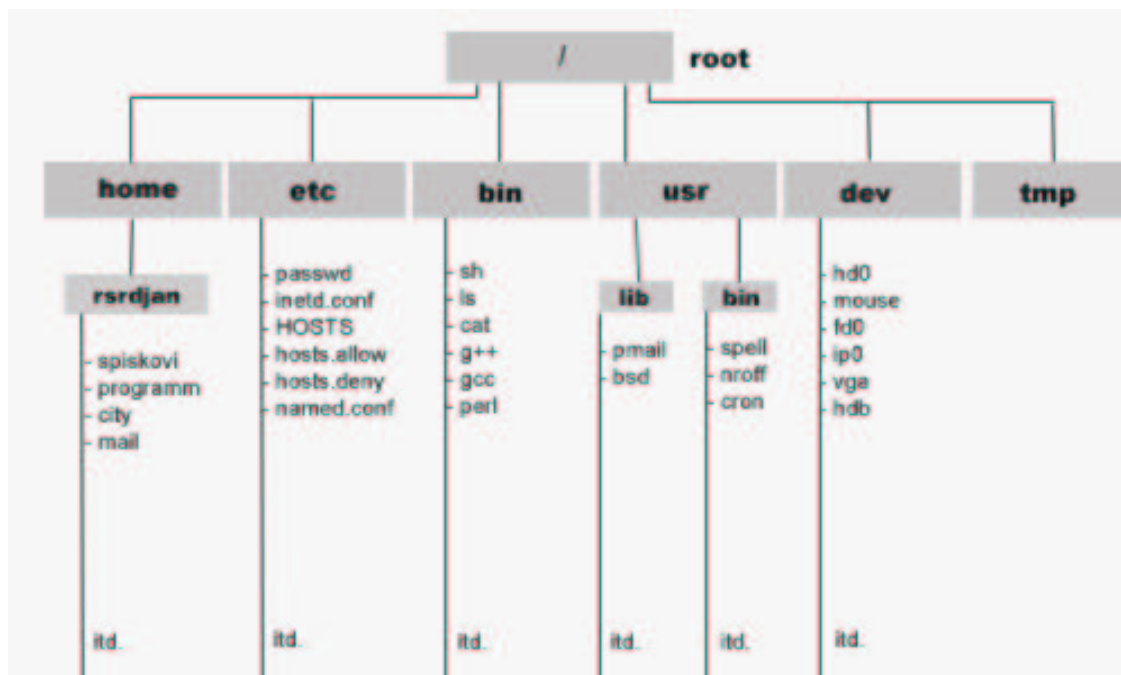
2.3.1 Organizacija UNIX file sistema

UNIX fajl sistem je odgovoran za organizaciju, prepoznavanje, i smeštanje svakog fajla na računaru. UNIXova definicija fajla je veoma jednostavna - on je definiše kao skup bajtova, i on ne zna šta se unutar svakog fajla nalazi, odnosno unos podataka u fajl prepušta programu koji radi sa tim fajlom.

Svaki fajl je označen sa logičnim imenom koji se sastoji od niza ASCII karaktera. Jedini karakter koji ne može biti korišten za davanje naziva fajlovima je slash znak (/), koji je određen kao separator za putanje u sistemu direktorijuma. Poželjno je da se nazivi fajlovima određuju sa karakterima koji se mogu odštampati, ali UNIX tu nije striktan. Svi UNIX sistemi današnjice dozvoljavaju 14 karaktera za ime fajla, mada mnogi moderni UNIXi dozvoljavaju i više. Naziv fajla ne predstavlja ništa za sistem, ali određeni programi zahtevaju da fajl sadrži odgovarajuću ekstenziju da bi se mogao ophoditi sa njim (npr. GNU C kompajler zahteva da se fajlovima sa izvornim kodom da ekstenzija .c).

Jedini fajlovi koje UNIX sistem razlikuje od ostalih su oni koji su korišteni od strane jezgra (kernela) za specijalne namene, kao npr. naziv uređaja ili link koji ukazuje na neki drugi fajl. Svi ostali mogu biti baze, slike, tekstualni i izvršni fajlovi, jednom rečju svi podaci koje korisnik smešta na računar.

Na slici je prikazana organizacija tipičnog UNIX sistema



UNIX fajl sistem je organizovan po hijerarhalnoj direktorijumskoj strukturi. Osnovni direktorijum (root) je uvek označen sa / (slash), a najvažniji poddirektorijumi su:

- */home* služi za smeštanje korisničkih direktorijuma i njihove fajlove. Svaki korisnik dobija svoj direktorijum i njemu se na raspolaganje stavlja dalja organizacija;
- */bin* sadrži ljsuske i najčešće upotrebljivane komande i programe;
- */etc* sadrži konfiguracione, start-up i skripte za administraciju sistema;
- */user* se koristi za smeštanje niza poddirektorijuma koji sadrže biblioteke i ostale fajlove neophodne za pravilno funkcionisanje sistema;
- */dev* sačinjavaju fajlovi koji označavaju hardverske uređaje na računaru;
- */tmp* se koristi za privremeno čuvanje podataka koji su u upotrebi pri radu na sistemu;

Iako svaki administrator može konfigurisati sistem po svojim prohtevima, preporučuje se standardizovana struktura kako bi organizacija bila poznata svakom korisniku koji se često prebacuje sa jednog računara na drugi.

Linux-ov fajl sistem, iako gotovo identičan sa uobičajenim UNIX-ovim sadrži malu razliku u vidu da on poseduje svoj sopstveni *Linux native* fajl sistem koji je sposoban da prepozna i Windowsove FAT 32 sisteme na istom hard disku recimo. On je dakle sposoban da mountuje (podigne,prepozna) različite alternativne fajl sisteme kao i bilo koji drugi uređaj na računaru, a ovo mu sve omogućava tzv. VFS odnosno virtuelni fajl sistem.

2.3.2 Preusmeravanje

Svakako jedna od moćnih osobina UNIX-a je da podržava tzv. *redirecting* odnosno preusmeravanje rezultata koji nam daje izvršavanje određenog procesa na bilo koji uređaj na računaru. Ilustrovaćemo proces preusmeravanja na komandi *cat* za koju smo rekli da služi za ispis sadržaja fajla

```
rsrdjan@zeus:~$ cat spisak1.dat
```

```
-----  
Spisak korisnika u mesecu Januaru 2002. godine  
-----
```

```
rsrdjan  
stelios  
kwstas  
alexandros  
jelena.r  
-----
```

```
rsrdjan@zeus:~$ _
```

Šta smo uradili ? Naredili smo sistemu da nam na ekranu prikaže sadržaj tekstualnog fajla *spisak1.dat*.

Želimo li da taj ispis preusmerimo u novi fajl i sačuvamo ga uradićemo sledeće

```
rsrdjan@zeus:~$ cat spisak1.dat > spisak2.dat
rsrdjan@zeus:~$ cat spisak2.dat
-----
Spisak korisnika u mesecu Januaru 2002. godine
-----
rsrdjan
stelios
kwstas
alexandros
jelena.r
-----

rsrdjan@zeus:~$ _
```

Jasno se vidi da je rezultat (output) comande *cat* preusmeren u novi fajl pod nazivom *spisak2.dat*.

2.3.3 Pretraga za određenim tekstom – *grep*

Želimo li pretražiti određeni direktorijum ili čitav sistem za fajlovima koju u sebi sadrže određeni tekst, učinićemo to sa alatom *grep*. Evo i primera

```
rsrdjan@zeus:~$ grep -l 'Spisak korisnika' /users/rsrdjan/*
/users/rsrdjan/spisak1.dat
/users/rsrdjan/spisak2.dat
rsrdjan@zeus:~$ _
```

Kao argument smo dali *-l 'Spisak korisnika'* tj. označili smo tekst za kojim ćemo tragati, a kao lokaciju smo dali home direktorijum i naveli da se pretraži celokupan sadržaj uključujući i fajlove unutar poddirektorijuma. Sistem nam je odgovorio da je našao dva fajla koji sadrže tekst koji smo tražili.

2.3.4 Rad sa arhivama – *tar*, *gzip*, *rpm*

Rad na bilo kojoj platformi bi gotovo bio nezamisliv bez postojanja arhiva, tj. skupa fajlova koji su radi lakše manipulacije “zbijeni” u jedan. Najpoznatiji arhiver za UNIX-e je *tar* (Tape Archiver) koji gotovo uvek dolazi u kombinaciji sa programom za kompresiju koji se zove GNU Zip ili skraceno *gzip*.

U UNIX svetu, ako želimo da instaliramo neki program, najčešće ćemo se sresti sa arhivom koja sadrži izvorni kod programa koji će biti potrebno kompajlirati. U sledećem primeru pokazaćemo kako odkompresovati i raspakovati arhivu u lokalni direktorijum.

Recimo da imamo arhivu koja sadrži program *ddclient* i da je puni naziv arhive *ddclient.tar.gz*. Proces otpakivanja je sledeći:

```
rsrdjan@zeus:~$ gunzip ddclient.tar.gz
rsrdjan@zeus:~$ ls -l
-rw-r--r--  1 rsrdjan    rsrdjan  143360 Jan 25 21:53 ddclient.tar
-rwxr-xr-x  1 rsrdjan    rsrdjan  16299 Feb  5 01:56 fawx*
-rw-r--r--  1 rsrdjan    rsrdjan   5173 Feb  5 01:55 fawx.c
rsrdjan@zeus:~$ tar -xvf ddclient.tar
ddclient-3.6.2/
ddclient-3.6.2/COPYRIGHT
ddclient-3.6.2/COPYING
ddclient-3.6.2/README.cisco
ddclient-3.6.2/README
ddclient-3.6.2/sample-etc_ddclient.conf
ddclient-3.6.2/sample-etc_rc.d_init.d_ddclient
ddclient-3.6.2/sample-etc_rc.d_init.d_ddclient.redhat
ddclient-3.6.2/sample-etc_dhcpd_dhcpd-eth0.exe
ddclient-3.6.2/sample-etc_ppp_ip-up.local
ddclient-3.6.2/sample-etc_dhclient-enter-hooks
ddclient-3.6.2/sample-etc_cron.d_ddclient
ddclient-3.6.2/ddclient
rsrdjan@zeus:~$ cd ddclient-3.6.2
rsrdjan@zeus:~/ddclient-3.6.2$ _
```

Dakle sa komandom *gunzip* smo prvo odkompresovali arhivu, a zatim je i otpakovali komandom *tar -xvf* (-x extract, -v verify, -f force). Nakon toga dobili smo poseban direktorijum *ddclient-3.6.2* u kome su smešteni fajlovi koji su sačinjavali arhivu. Naglašavamo da UNIX podržava proces *pipinga* tj. nadovezivanja jednog procesa na drugi, tako da smo gorepomenute dve komande mogli pozvati iz jedne linije sa sintaksom *gunzip ddclient.tar.gz | tar -xvf*.

Više verzija Linux-a koristi i poseban format arhive, tzv. *rpm* paket koji je promovisan od strane Red Hat-a, jedne od najpoznatijih distribucija Linux-a. Napominjemo da svi fajlovi unutar rpm arhive dolaze već kompajlirani, odnosno u binarnoj formi što nije slučaj sa *.tar.gz* koji se najčešće koristi za pakovanje izvornog koda (najčešće su programi pisani u C/C++, pa se prema tome za kompajliranje koristi GNU C/C++ kompajler gcc ili g++).

Korištenje rpm-a je više nego jednostavno. Pretpostavimo da imamo arhivu pod nazivom *ddclient.rpm* i želimo da je instaliramo

```
rsrdjan@zeus:~$ rpm -i --nodeps --force ddclient.rpm
rsrdjan@zeus:~$ _
```

2.3.5 Kompajliranje programa

U ovoj sekciji ćemo obratiti pažnju isključivo na C/C++ izvorni kod i GNU C/C++ kompajler. Recimo da se u našem home direktorijumu nalazi fajl *testprogram.c* koji sadrži kod pisan u C-u.

```
rsrdjan@zeus:~$ gcc -o izvrsniprogram testprogram.c
rsrdjan@zeus:~$ _
```

Pod pretpostavkom da kod ne sadrži nikakve sintaksičke greške, naš program je uspešno kompajliran i u home direktorijumu možemo pronaći fajl *izvrsniprogram* kojeg pozivamo navođenjem imena.

U slučaju da se radi o C++ kodu i da je ime fajla *testprogram.cpp* (ekstenzija *.cpp* označava da fajl sadrži C++ izvorni kod), proces kompajliranja je gotovo identičan, s tim da ne pozivamo C, nego C++ kompajler

```
rsrdjan@zeus:~$ g++ -o izvrsniprogram testprogram.cpp
rsrdjan@zeus:~$ _
```

NAPOMENA: U slučaju da kod nije pravilno napisan, odnosno da sadrži sintaksičke greške, kompajler će vas obavestiti i ispisati na ekran u kojoj liniji je pronašao grešku. Editujte fajl, ispravite kod i ponovo pokrenite kompajler. Preporučujemo da ako želite da kompajlirate čitav niz fajlova koji se sadrže u arhivi prvo pročitate *README* ili *INSTALL* fajl u kojima su po običaju objašnjene procedure kompajliranja.

2.4 Rad u mrežnom okruženju

Zbog svoje pouzdanosti, stabilnosti i fleksibilnosti, UNIX se smatra odličnim izborom u radu u mreži. Pošto većina današnjih distribucija dolazi sa brojnim servisima (ftp, www, e-mail) koji su uz to još besplatni, odnosno open-source, to ga čini posebno pogodnim kao server za male i velike mreže. Treba uzeti u obzir da većina današnjih Internet servera (oko 80 %) ima instaliran neki od UNIX sistema, što dokazuje sve gore navedeno. U sledećih nekoliko poglavlja ukratko ćemo opisati rad sa nekim mrežno-orijentisanim alatima.

2.4.1 Prenos fajlova – *ftp*

Gotovo svaka poznatija UNIX distribucija sadrži klijent i server softver za prenos fajlova među udaljenim sistemima. U zavisnosti od brzine prenosa, možete lako da prenesete fajlove i direktorijume sa jednog računara na drugi.

Primer: želimo da prenesemo fajl *spisak1.dat* iz našeg home direktorijuma na udaljeni računar *aqua.rsdjan.com* na kome imamo pristup odnosno korisničko ime i lozinku, i da taj isti fajl smestimo u direktorijum *spiskovi* na računaru *aqua*:

```
rsrdjan@zeus:~$ ftp aqua.rsdjan.com
Connected to aqua.rsdjan.com.
220 ProFTPD 1.2.2rc3 Server (aqua.rsdjan.com)
Name (aqua.rsdjan.com:rsrdjan): rsrdjan
331 Password required for rsrdjan.
Password:
230 User rsrdjan logged in.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> cd spiskovi
250 CMD command successful.
ftp> put spisak1.dat
local: spisak1.dat remote: spisak1.dat
200 PORT command successful.
150 Opening BINARY mode data connection for spisak1.dat.
226 Transfer complete.
6609 bytes sent in 0.0256 secs (2.5e+02 Kbytes/sec)
ftp>
```

Komandom *ftp* pozvali smo program i kao argument mu dali ime računara na koji želimo da prenesemo fajl, u ovom slučaju *aqua.rsdjan.com*. Nakon toga dobili smo poruku o uspešnoj konekciji na udaljeni host, informaciju o verziji servera i prompt za korisničko ime. Nakon što smo se prijavili, komandom *cd spiskovi* prešli smo u taj direktorijum, a komandom *put* započeli smo proces prenosa fajla. Nakon toga dobili smo poruku o završenom prenosu i podatke o veličini fajla i brzini prenosa.

U slučaju da želimo da preuzmemo fajl sa udaljenog računara koristimo komandu *get* koja kao parametar zahteva ime fajla. Takođe, možemo kreirati direktorijum za komandom *mkdir* ili brisati fajlove sa *del* i direktorijume sa *rmdir*. Sa udaljenog sistema se odjavljujete komandom *bye*. Više o ftp komandi možete saznati sa *man ftp*.

2.4.2 Surfanje po mreži - *lynx*

Ako želite na brzinu da pregledate e-mail, pročitate novosti ili ostale usluge koje vam WWW protokol nudi, koristite mali tekstualni WWW browser *Lynx*. Pošto se nalazite u comand prompt modu, ne očekujte čudo – *Lynx* vam neće prikazati slike, a nećete čuti ni muziku. On je prvenstveno namenjen za brzu pretragu. Ipak, grafiku, zvuk i multimediju uopšte možete imati ako koristite X-Windows GUI (više u sekciji *X-Windows*).

Recimo da želimo pregledati stranicu *http://www.freebsd.org*. Koristićemo sintaksu *lynx http://www.freebsd.org*. Nakon toga, naćete se u *Lynx*-u. Ako želite da idete na neki drugi sajt, jednostavno otkucajte slovo *G* i *Lynx* će vas pitati za novu adresu. Sa slovom *D* započinjete proces download-a. Iz programa izlazite sa *Q*.

2.4.3 Razgovor u živo – *irc*

Pored World Wide Web-a, jedan od najpopularnijih Internet servisa je svakako Internet Relay Chat (IRC). Za UNIX postoje mnogi IRC klijenti ali najpopularniji među njima su *BitchX* i *Epic IrcII*. Izbor jednog od ova dva je na korisniku. U sledećem primeru ćemo ukratko opisati sintaksu za pozivanje *BitchX*-a:

```
rsrdjan@zeus:~$ BitchX nana.irc.gr 6667 -c Aggeloi -n techX
BitchX: Auto Response is set to - techX
Connecting to port 6667 of server nana.irc.gr [refnum 0]
[irc] *** Looking up your hostname...
[irc] *** Checking Ident
[irc] *** Found your hostname
BitchX: For more information about BitchX type /about
Welcome to Greek IRC Network techX!tcx@zeus.rsrdjan.com
Your host is nana.irc.gr[@0.0.0.0], running version
    bahamut-1.4(32)+greeklish
This server was created Tue Feb 12 2002 at 09:33:37 EET
nana.irc.gr bahamut-1.4(32)+greeklish oiesadlkfirut etpLDsskfdf
[local users on IRC(229)] 19 %
```

Kao parametar komandi *BitchX* dali smo ime servera *nana.irc.gr* i port na koji se konektujemo (6667), a sa *-c Aggeloi* želimo da se automatski pridružimo kanalu *Aggeloi*. Sa *-n techX* označili smo naš nadimak (nickname).

2.4.4 Čitanje i slanje e-maila – *mail*

U ovom primeru pokazaćemo kako pročitati i poslati e-mail sa standardnim programom *mail*. Iako je *mail* sve manje u upotrebi jer postoje jednostavniji i pregledniji programi za rad sa e-poštom (Pine i Elm), on je ipak standard na svim UNIXima, i ne zavisi od volje administratora da li će ga instalirati ili ne.

Želimo da pošaljemo poštu na adresu *techx@aqua.rsrdjan.com* :

```
rsrdjan@zeus:~$ mail -s Test techx@aqua.rsrdjan.com
Ovo je
Test poruka
Srdjan
.
Cc:
rsrdjan@zeus:~$ _
```

Sa *-s Test* naglasili smo da je naslov ove poruke *Test*, i označili smo adresu na koju želimo da pošaljemo poštu. Nakon toga, *mail* nas je stavio u edit mod, odnosno dao nam je prostor da otkucamo svoju poruku. Kada smo završili sa kucanjem, treba da pošaljemo tačku (.) za kraj edit moda. Nakon toga *mail* nas pita da li još nekome želimo da pošaljemo istu poruku sa *Cc*: što je u ovom slučaju bilo odgovoreno sa ne.

Korisnik *techx* na računaru *aqua.rsrdjan.com* bi trebao da upiše sledeće da bi pročitao tu poruku:

```
techx@aqua:~$ mail
Mail version 8.1 6/6/93. Type ? for help.
"/var/mail/techx": 1 message 1 unread
>U 1 rsrdjan@zeus.rsrdjan.c Sun Feb 24 04:23 20/497 "Test"
&t 1
Message 1:
From rsrdjan Sun Feb 24 04:23:41 2002
Date: Sun, 24 Feb 2002 04:23:41 +0200
From rsrdjan <rsrdjan@zeus.rsrdjan.com>
To: techx@aqua.rsrdjan.com
Subject: Test
X-IMAPbase: 1014517431 2
X-Status:
X-Keywords:
X-UID: 1
Ovo je
Test poruka
Srdjan
&q
Saved 1 message in mbox
techx@aqua:~$ _
```


Kada smo pozvali program, ispisao nam je svoju verziju i pokazao nam da čuva jednu poruku u svom direktorijumu, koja nije pročitana. Sa *t l* je pročitana PRVA poruka u inbox-u i pokazan nam je pošiljalac, vreme kada je poruka poslana i kada je primljena i naslov poruke. Sa *q* smo izašli iz programa i pre nego nas je vratio u sistemski prompt, pokazao nam je da je sačuvao tu poruku. Više informacija o programu *mail* dobijate sa *mail -help* kao i sve dodatne parametre.

2.4.5 Konektovanje na udaljeni sistem – *telnet*

Jedan od najvažnijih faktora popularosti UNIX-a je mogućnost administriranja i remote rada na udaljenom sistemu. Za prijavljivanje na udaljeni sistem koristimo klijent *telnet*.

Pretpostavimo da se želimo prijaviti na računar *thigritza.office.rsrdjan.com* a sa njega na *macedonia.uom.gr*

```
rsrdjan@zeus:~$ telnet
telnet> open thigritza.office.rsrdjan.com
Trying 192.168.0.7...
Connected to thigritza.office.rsrdjan.com.
Escape character is '^]'.

*****
Welcome to THIGRITZA.office.rsrdjan.com
*****

thigritza login: techx
Password:

FreeBSD 4.5
Last login: Wed Feb 23 01:13:26 +0200 2002 on tty3.
$> telnet macedonia.uom.gr
Trying 195.251.213.104...
Connected to macedonia.uom.gr.
Escape character is '^]'.

Digital UNIX (macedonia.uom.gr) (ttyp0)

login: rsrdjan001
Password:

No mail for user rsrdjan001.
rsrdjan001@macedonia$ _
```

NAPOMENA: Budite pažljivi pri korištenju telnet-a. jer telnet server ne koristi kriptovanu konekciju, pa prema tome nije ni siguran. U poslednje vreme *telnet* se nastoji izbaciti iz većine UNIX distribucija zbog svoje nesigurnosti. Kao dobra zamena pokazao se *SSH* (Secure Shell).

2.5 UNIX programiranje

Sa većinom današnjih UNIX-a dolazi mnoštvo kompajlera, biblioteka i razvojnih okruženja, tj. sve što je potrebno za open source programiranje. Nabrojaćemo samo neke jezike podržane od strane UNIX-a: Ada, Assembler, AWK, BASIC, C, C++, Caml, CLisp, Dylan, Eiffel, Erlang, Forth, Fortran, Gofer, Guile, Haskell, Icon, Intercal, Java, Lua, Lisp, Mercury, Modula-2, Modula-3, Oberon, Objective C, Pascal, Perl, Pike, Postscript, Prolog, Python, REXX, Sather, Scheme, sh, Simula, Smalltalk, SQL i Tcl.

Takođe, ljuske na UNIX-u sadrže moćnu sintaksu za kreiranje tzv. shell skripti, odnosno programa koji vam znatno olakšavaju rad u ljusci. Ono što sledi je kratak opis nekih programskih jezika.

2.5.1 Java

Java je pravi objektno orjentisani programski jezik sa primenom Smalltalk koncepta. Ovaj jezik je potpuno nezavistan što se tiče platformi na kojima se programi razvijaju, sadrži jednostavnu sintaksu i standardne biblioteke. Međutim, Java pati od nekoliko nedostataka:

- Jezik je razvijen od strane jedne kompanije (Sun) koja nije zainteresovana za standardizovanje i globalno razvijanje, nego se drži komercijalnih interesa ;
- U biblioteke i jezik se konstantno ubacuju nova rešenja, što primorava programera da uči i usvaja ta rešenja;

Bilo kako bilo, većina programera smatraju da Java predstavlja pravi izbor za budućnost zamenjujući druge objektno orjentisane jezike kao npr. C++. Ono što je važno napomenuti je to da je današnja primena Jave veoma široka, pogotovo za WWW jer se u njoj razvijaju tzv. applet – programi napisani u Javi koji se ubacuju u HTML kod, prenose na korisnikov računar i izvršavaju uz pomoć tzv. *Java Virtual Machine* (JVM). Više o Javi možete pronaći na oficijelnom Internet sajtu proizvođača Sun Microsystems-a (<http://java.sun.com>).

2.5.2 Perl

Perl (Practical Extraction and Report Language) je jezik optimizovan za rad sa tekstualnim fajlovima koji iz njih uzima neophodne informacije i prikazuje ih. Takođe, primenjen je u poslovima oko administracije sistema i nezamenjiv je za mnoge administratore. Perl kombinuje najbolje osobine C-a, sed-a, awk-a i sh-a, pa programeri kojima je sintaksa ovih jezika poznata neće imati problema u radu sa njim. Za razliku od mnogih UNIX jezika, Perl ne poseduje tzv. memorijsko ograničenje, odnosno oslanja se na količinu memorije koju računar poseduje. Tako naprimer čitav jedan tekstualni fajl možete definisati kao string. Perl je primenjen i u poslovima oko sortiranja, jer je sposoban da pretraži ogromne količine podataka za veoma kratko vreme. Primenu nalazi i u Web programiranju

jer se u njemu programiraju tzv. CGI skripte koje olakšavaju rad sa bazama podataka i generisanjem HTML koda. Više informacija o Perlu pronađite na sajtu Perl zajednice na adresi <http://www.perl.com>.

2.5.3 Tcl

Tcl i *Tk* su kreirani od strane John Ousterhouta, i za razliku od drugih jezika uglavnom je primenjen kao dodatno rešenje za velike projekte. Odlikuje se jednostavnom sintaksom i portabilan je – verzije Tcl-a možete naći na gotovo svakom sistemu današnjice. Tk je grafičko razvojno okruženje koje omogućava brzo kreiranje lepo dizajniranih grafičkih aplikacija koristeći Tcl sintaksu. Kao i Perl, Tcl je zbog nedostatka standarda mnogo propatio u svojoj karijeri jer je godinama odbacivan kao rešenje za ozbiljan rad. Više o Tcl-u na <http://www.scriptics.com/software/tcltk/>.

2.5.4 Python

Python je jednostavan za učenje i moćan programski jezik. Odlikuje se high-level struktuiranjem podataka i jednostavnim pristupom objektno orjentisanom programiranju. Elegantna sintaksa ga čini idealnim jezikom za skriptovanje i brz razvoj aplikacija za mnoge platforme. Pythonov preodilac i biblioteke su besplatno dostupne na mnogim lokacijama po Internetu, kako u izvornom kodu tako i u izvršnom obliku. Prevodilac je u novijim verzijama dodatno proširen sa novim funkcijama i tipovima uglavnom preuzetih iz C/C++, što ga čini lakim za učenje i korišćenje mnogim UNIX programerima. Zanimljivo je da Python nudi najopširniju moguću dokumentaciju za početnike i iskusne programere, u vidu raznih vodiča po Internetu kao i knjiga objavljenih od strane mnogih računarskih stručnjaka. Više o Pythonu na <http://www.python.org> .

2.5.5 C/C++

Kao što smo naglasili, celokupan UNIX je baziran na jeziku C koji je specifično razvijen od strane Denis Ritchia za potrebe portovanja ovog operativnog sistema. U kasnim sedamdesetim, C je evoluirao u ono što je sada poznato pod imenom “tradicionalni C”, “klasični C” ili “Kerninghan i Ritchie C”. Publikacija koja je objavljena od strane Kerninghana i Ritchia “*The C Programming Language*” privukla je veliku pažnju računarskog sveta na ovaj jezik. Ogromna raširenost C-a na različitim tipovima računara je nažalost dovela do mnogih varijacija i odstupljanja od izvornog koda, koji je bio jako sličan ali ipak nekompatibilan. Ovo je bio ozbiljan problem za mnoge programere koji su hteli da aplikaciju napisanu na jednoj platformi prebace na drugu. Uskoro je postalo jasno da je neophodno uvođenje jedinstvenog, standardnog oblika C-a.

1983. ANSI (American National Standards Committee on Computers and Information Processing, poznatiji i kao X3) je osnovao poseban tehnički komitet kome je bio zadatak da odredi platformski nezavisni skup pravila po kojima će C biti uređen. Standard je konačno usvojen 1989. godine, i dokument koji je sadržao pravila je objavljen 1990. Drugo izdanje Kerninghanove i Ritchijeve knjige opisuje tzv. ANSI C, verziju koja je usvojena i koja je u upotrebi širom sveta.

C++, proizašao iz C-a, je kreiran od strane Bjarne Strostrupa u ranim osamdesetim u Bell laboratorijama. C++ sadrži niz novina koje proširuju C, ali ono što je najvažnije je da je C++ definisan kao *objektno orjentisan jezik*. Za razliku od dotadašnjeg *strukturnog* programiranja u C-u, C++ je uveo novi oblik, tzv. *objekte*, tj. softverske komponente koje emuliraju modele iz svakodnevnog života. Programeri su otkrili da se do željenog rezultata korištenjem novog modularnog, tj. objektno orjentisanog pristupa (OOP) dolazi mnogo brže i tačnije. OOP programi su jasniji za razumevanje, i jednostavniji za korekciju i modificiranje.

Od tada, razvijeni su mnogi OOP programski jezici, kao npr. *Smalltalk* u kome objekti predstavljaju gotovo sve, dok je C++ hibridni jezik, odnosno programi se mogu pisati po dotadašnjem C modelu, OOP-u, ili kombinacijom jednog ili drugog.

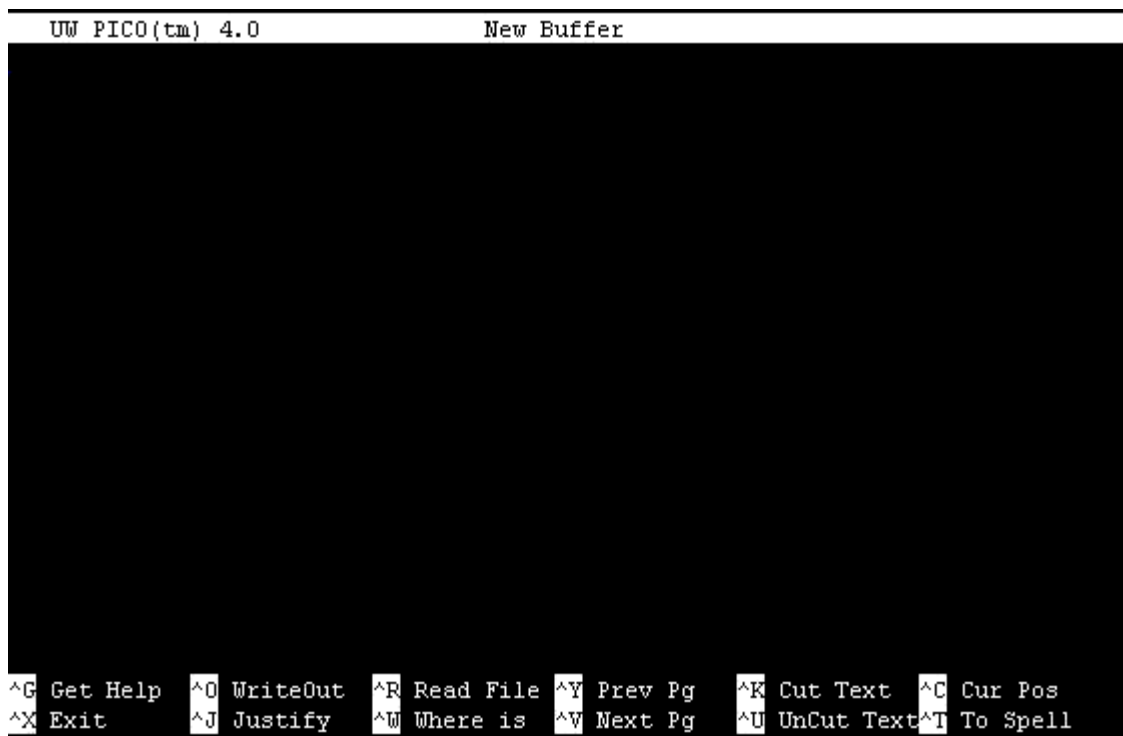
Sve verzije UNIX-a danas sadrže C odnosno C++ kompajler, a najrasprostranjenija verzija je svakako GNU kompajler (<http://www.gnu.org>) koji je dostupan za besplatan download i upotrebu.

2.6 Rad sa programima

2.6.1 Ekranski editor *Pico*

Iako *vi* i dalje slovi za neospornog vladara na području editora na UNIX-u, većina današnjih distribucija sadrži mnogo jednostavniji i za početnike razumljiviji ekranski editor pod imenom *Pico*. *Pico* dolazi u sklopu mail programa *Pine* (opisan u sledećem poglavlju) i primarna uloga mu je da služi za pisanje i čitanje pošte, mada, kako ćemo videti može poslužiti i za druge poslove.

Program se poziva iz ljsuke komandom *pico* koja kao argument može primiti i ime fajla u koji želimo da editujemo. U suprotnom, program pretpostavlja da želimo da kreiramo novi fajl, i dobijamo ekran sličan ovome



```
UW PICO(tm) 4.0          New Buffer
[Black editing area]
^G Get Help  ^O WriteOut  ^R Read File  ^Y Prev Pg   ^K Cut Text   ^C Cur Pos
^X Exit      ^J Justify    ^W Where is  ^V Next Pg   ^U UnCut Text ^T To Spell
```

Pico nas automatski stavlja u edit mod, i možemo početi sa kucanjem. Nakon što smo završili sa unosom teksta, fajl snimamo pritiskom na dugme CTRL u kombinaciji sa slovom O nakon čega će nas program pitati za ime fajla. Ako tekst sadrži više od jedne strane, možemo se kretati brzo sa kombinacijom CTRL+Y za prethodnu ili CTRL+V za sledeću stranu, linije možemo brzo brisati sa CTRL+K a u slučaju da smo pogrešili možemo ih vraćati sa CTRL+U. Takođe omogućeno je brza pretraga teksta za određenom rečju ili tekstem pomoću CTRL+W, a pomoć možemo dobiti pritiskom na CTRL+G. Sa CTRL+X izlazimo iz programa i vraćamo se u ljsuku.

2.6.2 Mail klijent *Pine*

U prethodnim poglavljima naučili smo kako koristiti jednostavni alat *mail* za čitanje i slanje elektronske pošte. Sada ćemo malo olakšati posao i pokrenuti moćan i jednostavan program koji se zove *Pine*. Nakon što smo pozvali program sa *pine* dobijamo uvodni ekran

```

PINE 4.33  MAIN MENU  Folder: INBOX 2 Messages

?  HELP          -  Get help using Pine
C  COMPOSE MESSAGE -  Compose and send a message
I  MESSAGE INDEX  -  View messages in current folder
L  FOLDER LIST   -  Select a folder to view
A  ADDRESS BOOK   -  Update address book
S  SETUP          -  Configure Pine Options
Q  QUIT          -  Leave the Pine program

Copyright 1989-2001.  PINE is a trademark of the University of Washington.
[Folder "INBOX" opened with 2 messages]
? Help          P PrevCmd          R RelNotes
O OTHER CMDS  [ListFldrs]  N NextCmd          K KBlock

```

Kao što vidimo, obavešteni smo da u inboxu imamo 2 poruke i program je spreman za rad. Po menijima se krećemo upotrebom strelica, a ukoliko želimo da pošaljemo novu poruku, automatski će se startovati Pico kojeg smo upoznali u prethodnom poglavlju, sa razlikom da u ovom slučaju Pico dobija novu funkciju *Send message* koju aktiviramo pritiskom na CTRL+X. Takođe, poruku možemo ostaviti neposlanu da CTRL+O, a ukoliko želimo da izbrišemo neke od poruka koje smo prethodno sačuvali moramo ih prvo obeležiti pritiskom na slovo D, a po izlasku iz menija program će nas upitati da li želimo da izbrišemo poruke koje smo obeležili.

U svakom trenutku, meniju za pomoć možemo pristupiti pritiskom na CTRL+? a za povratak na prethodni ekran pritiskom na slovo P. Ako smo ušli u meni *Setup* videćemo da možemo promeniti skoro sve, od imena i prezimena pošiljaoca do editovanja potpisa koji će biti umetnut na kraju svake nove poruke koju pošaljemo.

Jako korisna opcija je i *Adress book* u kojoj možemo sačuvati podatke o osobama sa kojima se najčešće dopisujemo, te tokom narednog slanja poruke iskoristiti prečice koje smo prethodno odredili.

2.7 Odjava sa sistema

Sa sistema se odjavljujete na dva načina: možete koristiti *logout* program koji će se pobrinuti da vas na lep način odjavi sa sistema, često pitajući da li želimo da sačuvamo rad koji nismo završili ili u kojem smo bili prekinuti, ili jednostavno upotrebom komande *exit*.

NAPOMENA: Prilikom procesa odjave, sistem će ugasiti sve vaše prethodno aktivirane procese (osim ako komandom *nohup* niste aktivirali neki pozadinski proces, odnosno proces koji će ostati aktivan i nakon što se odjavite), zato je dobra ideja da ipak proverite da li trebate nešto trajno sačuvati.

O autoru

Srđan M. Rajčević je student prve godine računarskih nauka na britanskom univerzitetu u Sheffieldu. Rođen je 10. oktobra 1982. godine u Banja Luci, Republika Srpska, a trenutno živi i radi u Solunu, Republika Grčka.

Tokom svog dugogodišnjeg rada sa računarima, objavio je mnoštvo radova na temu UNIX-a, Linuxa, baza podataka, Web dizajna, arhitekture računarskih sistema i programiranja u objektno orjentisanim jezicima od kojih mu je specijalnost C++.

Član je Udruženja Linux korisnika Grčke (HELLUG) i dopisnik Linuks.org-a – oficijelnog sajta Udruženja Linux korisnika Republike Srpske. Jedan je od osnivača i koordinator TCML-a, projekta razvoja mašinskog simulatora, a trenutno radi na poslovima administriranja UNIX sistema na odeljenju računarskih nauka City univerziteta u Solunu.

Govori engleski jezik, a služi se i grčkim.

Osnove rada na UNIX operativnim sistemima je priručnik preveden na srpski i engleski jezik, i predstavlja upustvo za početnike koji se prvi put susreću sa UNIX sistemom. U pripremi je i *C++ - Osnove programiranja*.

Za sva pitanja ili sugestije, autor je dostupan na e-mail adresi s.rajcevic@city.academic.gr a možete posetiti i njegov lični web sajt na adresi <http://www.rsdjan.com> .

Korištena literatura

- *The Architecture of Computer Hardware and System Software: An information technology approach*, Irv Englander, John Willey & Sons, Inc. 2002
- *C++ How to Program*, H.M. Deitel & P.J. Deitel, Prentice Hall 2001
- *Advanced UNIX Development*, Bernard Tucker, Krex Publications 2000