

1. Istorijat Unix-a

1.1 Multics

Sredinom 60ih MIT, Bell Labs i General Electric radili su na pravljenju operativnog sistema Multics (Multiplexed Information and Computer Service). Cilj je bio da više korisnika radi na jednom računaru, interaktivno. Glavni nedostatak sistema bio je da je puno vremena trošio na donošenje odluka, a malo vremena ostavljalo korisnicima. Vremenom su se firme povlačile sa projekta pa Multics nikada nije doživeo širu primenu. Međutim, sistem je uključivao mnogo inovativnih ideja koje su imale veliki uticaj na računarstvo.

U osnovi je bila modularna organizacija hardvera i softvera. Ideja je bila da se sistem može jednostavno proširivati dodavanjem novih komponenti (procesora, memorije, diskova itd). Ovo je bilo moguće uraditi bez prekida rada sistema. MIT je koristio ovo da u nekom trenutku odvoji deo hardvera, sklopi drugi računar koji će obaviti neki novi zadatak, i po obavljenom zadatku vrati delove u glavni računar koji je u međuvremenu normalno radio.

Multics je bio prvi veći sistem koji je omogućio hijerarhijski sistem datoteka. Napisan je na višem programskom jeziku (PL/1). Odvojena lista za kontrolu pristupa za svaku datoteku omogućuje fleksibilno deljenje podataka i potpunu privatnost kada je to potrebno. Svi ovi koncepti uticali su na razvoj Unix-a.

1.2 Počeci razvoja Unix-a

Bell Labs, nezadovoljan što Multics postaje sve veći i kompleksniji a ne postiže zacrtane ciljeve, polako se povlači sa projekta. Poslednji istraživači koji su napustili projekat odlučili su da probaju da ponovo naprave sistem mnogo manje veličine. Osnovni koncepti novog sistema bili su višeproceno izvršavanje, interaktivni rad, višekorisnički sistem i jedinstvena hijerarhija sistemskih datoteka. Na čelu tima bio je Ken Thompson. On je kreirao sistem datoteka i sistem za straničenje kao i igru Space Travel. Igra je zahtevala moćniji računar da bi radila, pa je za njeno pokretanje upotrebljen u Belovim laboratorijama retko korišćen računar PDP7.

U to vreme, rad na Multics-u je dolazio do kraja i menadžment firme nije hteo da kupuje novu mašinu za ciljeve programiranja. Na mašini na kojoj je razvijen Space Travel, tim je počeo na razvoju novog operativnog sistema. Projekat je dobio ime Unics (Uniplexed information and Computer Service) po uzoru na Multics. Unics je kasnije podržao rad sa više korisnika pa mu je ime promenjeno u Unix. Da bi koristili Unix na jačoj mašini, Tompson i kolega iz tima Dennis Ritchie obećavaju razvoj programa za obradu teksta. U tu svrhu dobijaju na korišćenje računar PDP-11/20. Tokom 1970. godine Unix je napisan na assembleru ovog računara i tada je prvi put zvanično pokrenut. Unix Programmer's Manual je objavljen 1971. godine.

1.3 Prebacivanje na C

Godine 1973. Unix je ponovo napisan na programskom jeziku C koji je dizajnirao Dennis Ritchie. To je bilo u suprotnosti sa tadašnjom idejom da softver tako kompleksan kao operativni sistem i koji strogo mora da vodi računa o efikasnosti mora biti napisan na assembleru. Ovo je omogućilo da softver bude mnogo prenosiviji, pa je Unix mogao da se instalira na različitim arhitekturama. Pre toga Unix se isporučivao sa mašinom.

1.4 Širenje Unix-a

U početku su ga koristili odeljci AT&T za obradu dokumenata i štampanje, pa su prvi alati na Unix-u imali upravo tu namenu. Firma je u to vreme imala zabranu prodaje softvera, pa je Unix počeo slobodno da se deli zajedno sa izvornim kodom i knjigama sa odgovarajućim uputstvom. Univerziteti su bili veoma zainteresovani, pogotovo za kurseve operativnih sistema. Sistem se brzo širio, korisnici su uočavali šta ne valja, prijavljivali greške i sami pravili zakrpe. Unix je počela da koristi i vlada SAD. Krajem 70ih napravljen je računar WAX, na njemu je instaliran Unix i zajedno su počeli da se šire.

Sredinom 80ih sistem počinje da se koristi i u poslovne svrhe. Mnoge firme počinju da nude komercijalne verzije Unix-a za mini kompjutere i radne stanice. Unix je bio pogodan za prilagođavanje raznim potrebama, pa su ga koristile mnoge kompanije. Dve najrasprostranjenije verzije Unix-a u to vreme bili su upravo njihov *System V* i verzija Univerziteta u Berkliju *BSD*. BSD je sadržao pakete za mrežnu komunikaciju i komunikaciju među procesima koji se i danas koriste u Linux-u. Nešto kasnije Sun Microsystems razvija sistem *SunOS* (kasnije po uzoru na njega nastao Solaris).

1.5 PC računari i nekompatibilnost Unix-a

Početkom 80ih PC računari počinju da se šire velikom brzinom. Prodaju se po znatno nižoj ceni od dotadašnjih računara i namenjeni su za kućnu upotrebu. Verovalo se da će Unix sa svojom prenosivošću i mogućnostima postati standardan sistem na ovim računarima. Međutim, u to vreme su prodavci različitih Unix sistema želeli da vežu kupce za svoj proizvod, pa su namerno pravljene nekompatibilne varijacije sistema. Ovo je znatno umanjilo prenosivost Unix programa. Zbog toga se osamdesete godine zovu godine ratova Unix-a. U to vreme je AT&T ponovo dobio pravo prodaje softvera, pa je firma pokrenula niz sudskih procesa koji su zamalo uništili Unix. Microsoft je imao dobru početnu poziciju i sistemom DOS kome je kasnije dodao grafički korisnički interfejs zauzima vodeću poziciju na tržištu operativnih sistema.

1.6 Standardizacija

Osamdesetih godina prošlog veka počinju razgovori o standardizaciji. Cilj je da se napravi standard kako bi različiti sistemi zasnovani na Unix-u bili kompatibilni. U okviru IEEE (Udruženje inženjera elektronike i elektrotehnike) formira se telo za standardizaciju. Razvijen je standard POSIX 1988 (Portable operating system interface, X u imenu jer je standard namenjen za Unix). POSIX propisuje koji su sistemski pozivi neophodni da postoje, koji shell programi, koji header fajlovi za C itd, da bi se sistem zvao Unix.

I Windows se sa nekoliko promena može izmeniti da postane POSIX kompatibilan sistem. Jedan način je Cygwin biblioteka koja implementira Unix sistemske pozive. Drugi način je softverski paket koji je proizveo Microsoft: Windows Services for Unix koji se isporučuje ugrađen u sistemu, a u novijim verzijama Windows-a zove se Subsystem for UNIX-based applications (SUA).

Organizacija koja se danas bavi standardizacijom naziva se Open group, a standard koji se koristi se zove SUS – Single Unix Specification. Postoji i poseban standard koji se bavi Linuxom i zove se Linux Standard Base (LSB).

1.7 Noviji razvoj Unix-a

Krajem 80ih godina prošlog veka, AT&T i Sun Microsystems zajedno su razvili System V Release 4 (SVR4). U ovo vreme je počela proizvodnja Unixa za vojne i pravne službe, a ove verzije su zahtevale veliku bezbednost i pouzdanost.

Problem koji je nastao u razvoju Unix-a bio je taj što sistemi više nisu bili besplatni i njihov kod više nije bio javno raspoloživ, što su upravo bili razlozi prvobitnog uspeha Unix-a. Sredinom 80ih Richard Stallman pokrenuo je GNU projekat pod okriljem organizacije Free Software Foundation, za koju je napisao i licencu *GNU General Public License*. Napisao je i editor *emacs* kao i GNU C kompajler, ali je najveća prepreka za razvoj GNU operativnog sistema bila nepostojanje kernela (kernel je program koji povezuje aplikativni softver i hardver računara). Linus Torvalds je početkom 90ih napravio kernel i nazvao ga Linux. Tako je sklopljen sistem GNU/Linux (kraće se zove Linux) koji je bio namenjen za PC računare. Linux nije pravi Unix već je Unix-nalik (Unix-like) operativni sistem.

U to vreme razvijeno je i više verzija BSD-a, tako da su dominirale dve nezavisne verzije Unix-a. U BSD-u su svi delovi sistema činili celinu, dok je ideja bila da u korisnici Linux-a sami prikupe pakete i sklope sistem koji može da funkcioniše. Pojavile su se firme sa idejom da prikupe pakete, prekompiliraju, zapakuju i prodaju tako sastavljen sistem. Jedna od prvih takvih firmi je Slackware. Tako nastaju *Linux distribucije*.

Danas su popularne distribucije Mint, Ubuntu, Red Hat, Debian, Suse, Fedora, itd. a od verzija Unix-a koriste se još Solaris, Mac OS i BSD. Linux je danas vodeći operativni sistem kada su u pitanju serveri, radne stanice i superračunari.

2. Najčešće korišćene komande

Generalno, Unix komande se sastoje iz tri dela (poslednja dva su opciona):

1. ime komande
2. opcije komande (slova ili reči kojima prethodi -)
3. argumenti (nazivi datoteka, direktorijuma, proizvoljni tekst)

Npr. `ls -l /tmp`

2.1 Osnovne naredbe

strelica gore dole	– za ponavljanje prethodno zadatih komandi
<code>Ctrl+R</code> <i>niska</i>	– traženje prethodno izvršenih komandi koje sadrže <i>niska</i>
<code>history</code>	– pregledanje istorije korišćenih komandi
<code>tab</code>	– za napisanu nisku lista sve komande koje počinju tom niskom
<code>tab</code>	– dovršava ime fajla; ako postoji više fajlova sa napisanim početkom, ispisaće se i onaj deo imena tih fajlova koji je zajednički za sve njih.
<code>Ctrl+c</code>	– prekid za većinu programa
<code>sudo -s</code>	– dobijanje administratorskih privilegija na sistemu

copy se u terminalu radi obeležavanjem levim tasterom, a *paste* klikom na srednji taster **miša**.

2.2 Naredbe za pomoćne stranice

<code>man 2 write</code>	– pomoćna stranica za write <i>sistemski poziv</i>
<code>man scanf</code>	– pomoćna stranica za funkciju <code>scanf</code>
<code>man man</code>	– pomoćna stranica za komandu <i>man</i>
<code>man -k <string></code>	– prikazuje sve komande koje u svojim kratkim opisima sadrže <i>string</i>
<code>info gcc</code>	– dobija se opis <code>gcc</code> naredbe
<code>man ls</code>	– pomoćna stranica za komandu <i>ls</i>

Kada se pri pregledanju man stranice otkuca */examples* onda se na toj stranici markiraju sva pojavljivanja niske *examples*. Ovo se odnosi i na bilo koju drugu nisku i omogućava brzu pretragu man stranice za tom niskom.

2.3 Naredbe za kretanje, kreiranje, listanje, brisanje

<code>pwd</code>	– štampa tekući direktorijum
<code>cd direktorijum</code>	– promena tekućeg direktorijuma
<code>cd ..</code>	– vraćanje u prethodni (roditeljski) direktorijum
<code>cd /home/mr02685</code>	– prelazak u direktorijum čija je putanja navedena
<code>cd ~</code>	– prelazak u home direktorijum
<code>ls opcija direktorijum</code>	– listanje sadržaja direktorijuma uz korišćenje opcija. Opcije:
<code>-R</code>	– rekurzivno listanje svih poddirektorijuma navedenog direktorijuma
<code>-l</code>	– listanje sa dodatnim informacijama
<code>-a</code>	– listanje svih fajlova, tj. i onih koji počinju sa '.', a inače se ne prikazuju
<code>-s</code>	– ispis datoteka i njihove veličine u kilobajtima
<code>-lh</code>	– ispis datoteka i čitljiv ispis njihove veličine
<code>du -sch direktorijum</code>	– ispis veličine navedenog direktorijuma
<code>cp fajl lokacija_nove_kopije</code>	– kopiranje datoteka
<code>mkdir direktorijum</code>	– kreiranje direktorijuma
Pri brisanju biti naročito oprezan!!! U Unix-u ne postoji recycle bin tako da je preporučljivo praviti rezervne kopije podataka.	
<code>rm fajl</code>	– brisanje datoteke
<code>rmdir direktorijum</code>	– brisanje praznog direktorijuma
<code>rm -r direktorijum</code>	– brisanje direktorijuma koji nije prazan
<code>mv fajl nova_lokacija</code>	– pomeranje datoteke

2.4 Dobijanje informacija o korisnicima

<code>users</code>	– spisak korisnika ulogovanih na sistem
<code>w</code>	– izveštaj u vidu spiska o trenutnom statusu sistema i spisak korisnika ulogovanih na sistem zajedno sa procesima koji im pripadaju
<code>w user</code>	– kao i prethodno, ali se prikazuju samo podaci za zadatog korisnika
<code>who</code>	– spisak korisnika ulogovanih na sistem
<code>who am i</code>	– ispis ličnih podataka za tekućeg korisnika
<code>finger</code>	– traži zadate podatke o korisniku i prikazuje informacije iz inicijalne konfiguracije korisnikovog naloga, kao i ostale sadržaje iz korisnikovih .plan i .project fajlova (ukoliko postoje). Prikazuje se i da li je korisnik prijavljen i koliko dugo traje sesija korisnika.
<code>finger user</code>	– kao i prethodno, ali se prikazuju samo podaci za zadatog korisnika
<code>last</code>	– lista ko je kad radio na serveru počev od nekog datuma.
<code>last user</code>	– isto to samo sa određenu osobu.
<code>groups user</code>	– prikazuje grupe kojima pripada korisnik

2.5 Komunikacija sa serverom i drugim korisnicima

telnet – pristup udaljenom računaru, nepreporučljivo koristiti zbog slabije sigurnosti
ssh *računar* – pristup udaljenom računaru, koristi kriptovanje pa se smatra sigurnim. Npr:
ssh mi11002@alas.matf.bg.ac.rs – prijavljivanje na alas; odjavljivanje se vrši komandom exit
hotssh mi11002@alas.matf.bg.ac.rs – prijavljivanje na alas (možda se program hotssh mora dodatno instalirati); može se koristiti grafičko okruženje za prebacivanje fajlova
scp *fajl nova_lokacija* – skraćeno od secure copy, kriptovano kopiranje podataka. Npr:
scp -r mi11002@alas.matf.bg.ac.rs:BB /home/ja - kopiranje direktorijuma BB sa alasa u direktorijum /home/ja (-r se koristi za rekurzivno kopiranje svih poddirektorijuma)
scp mi11002@alas.matf.bg.ac.rs:datoteka /home/ja/ - kopiranje jedne datoteke
pine – provera pošte
write – slanje poruke drugom korisniku, npr
write mi11002 tekst_poruke (ctrl+c za kraj)

Na Matematičkom fakultetu postoje 2 servera: *alas* (studentski) i *poincare* (nastavnički).

2.6 Naredbe za rad sa datotekama i tekstom

stat *fajl* – prikazuje informacije o navedenoj datoteci
emacs r.c.& – pokretanje editora emacs; & označava da se proces pokreće u pozadini, tj. shell se može koristiti za druge namene dok se proces izvršava
less – pregled teksta stranu po stranu. Akcije u komandi less:
/rec Osvetljava sve pojave niske *rec*. Ako se *rec* ne navede, to znači find next.
q Izlaz.
cat – povezuje fajlove i koristi se za ispis sadržaja fajlova
cat -b *lista_fajlova* numerisanje svih ne-blanko linija izlaza naredbe cat i to počev od 1
cat -n *lista_fajlova* numerisanje svih linija izlaza naredbe cat i to počev od 1
cat -s *lista_fajlova* zamena svih višestrukih blanko linija jednom linijom
cat > *fajl* upis karaktera koji se unose sa tastature u fajl do pritiska Ctrl+C, ali se pritom prethodni sadržaj fajla gubi
cat >> *fajl* dopisivanje karaktera koji se unose sa tastature na kraj fajla
wc *lista_fajlova* – brojač redova, reči i karaktera u fajlovima ili standardnom ulazu (ako nije navedena lista)
wc -L ispisuje dužinu najduže linije
wc -l broji prelaske u novi red
wc -w broji reči
wc -c broji samo karaktere
head *fajl* – ispisuje početak fajla (prvih 10 redova)
head -num *fajl* – ispisuje prvih num redova fajla
tail *fajl* – ispisuje kraj fajla (poslednjih 10 redova)
tail -num *fajl* – ispisuje poslednjih num redova
tail -f *fajl* – ispisuje kraj fajla, kao i redove koji se naknadno (npr. iz drugog procesa) dopisuju u fajl
sort *lista_fajlova* – prikazuje sortirane redove jednog ili više fajlova; ako nema zadatih fajlova, onda se sortiraju podaci sa standardnog ulaza i prikazuju se na standardni izlaz

sort f *lista_fajlova* – pri sortiranju se ne razlikuju velika i mala slova (ignore case)
 find *direktorijum -name ime_fajla* – pronalazi rekurzivno sve fajlove u direktorijumu sa navedenim imenom
 locate – efikasna pretraga fajla u sistemu

2.7 Naredba grep

Grep pretražuje navedene datoteke i ispisuje sve linije koje odgovaraju navedenom obrascu.

Primeri:

grep *opcija uzorak lista_fajlova* – ispisuje sve linije u kojima se nalazi uzorak. Opcije:
 -c – ispisuje broj pronađenih linija u kojima se nalazi *uzorak*
 -h – ispred linije ispisuje i naziv datoteke gde je nađen *uzorak*
 -i – zanemaruje se veličina slova pri pretrazi
 -E *regularni_izraz* – prikazuje sve linije koje sadrže *regularan_izraz*

Regularan izraz može da sadrži:

^	početak linije
\$	kraj linije
.	bilo koji karakter
[<i>lista_karaktera</i>]	bilo koji karakter iz skupa u zagradama, npr [1-2abc9-D]
[^ <i>lista_karaktera</i>]	tačno jedan karakter koji nije među navedenim
<i>izraz</i> {n}	<i>izraz</i> se pojavljuje n puta
<i>izraz</i> *	<i>izraz</i> se pojavljuje 0, 1 ili više puta
<i>izraz</i> +	<i>izraz</i> se pojavljuje bar jednom
<i>izraz</i> ?	<i>izraz</i> se pojavljuje 0 ili jednom

2.8 Još neke česte naredbe

clear – brisanje ekrana
 ps – prikazivanje procesa koji se trenutno izvršavaju u ime korisnika koji je pozvao ovu komandu
 ps aux – prikazivanje svih procesa na sistemu
 pstree – prikazivanje procesa koji se trenutno izvršavaju u obliku drveta
 top – spisak procesa sortiran po procentu zauzeća procesora; spisak se stalno ažurira (izlaz sa q)
 top -u mi11002 – spisak procesa samo korisnika mi11002
 id – identifikacija
 echo *poruka* – ispisuje na ekran *poruka*
 echo *poruka* > *fajl* – piše poruku u fajl; prethodni sadržaj fajla biva obrisan.
 echo *poruka* >> *fajl* – dopisuje poruku na kraj fajla.
 date – ispisuje trenutni datum i vreme
 cal – prikazuje kalendar za tekući mesec
 cal -y – prikazuje kalendar za tekuću godinu
 alias – koristi se za pravljenje skraćenice za neku komandu koja se koristi; skraćenice su aktivne dok se korisnik ne izloguje
 alias listaj="ls -l" – pravimo skraćenicu listaj ako koristimo shell bash.
 listaj – sad je ovo isto kao da smo otkucali ls -l.
 alias – spisak svih alijasa.
 unalias *skraćenica* – brisanje definicije za skraćenicu
 df *opcije fajl_sistemi* – ispisuje slobodan i zauzet prostor na montiranim particijama

`/etc/fstab` – datoteka koja sadrži trenutno montirane sisteme datoteka
`which` – određuje lokaciju na disku gde se nalazi uneta komanda
`tar` – naredba za komprimovanje (dekomprimovanje) podataka
`tar -pczf ime.tar.gz .` – pravljenje arhive `ime.tar.gz` koja sadrži sve iz tekućeg direktorijuma
`tar -xvzf ime.tar.gz` – raspakivanje arhive `ime.tar.gz` u tekući direktorijum
`tar -xvzf ime.tar.gz -C direktorijum` – raspakivanje arhive `ime.tar.gz` u navedeni direktorijum
`cat /etc/*version*` – ime i verzija distribucije koja se koristi
`uname` – informacije o verziji Linuxa kao i o procesorima
`uname -r` – prikaz verzije kernela
`lsb_release -a` – informacije o standardu kao i o samoj verziji sistema
`cat /proc/cpuinfo` – ispisuje informacije o procesoru i jezgrima procesora
`cat /proc/meminfo` – ispisuje informacije o memoriji

2.9 Nadovezivanje naredbi

Prednost Unix-a u odnosu na Windows je da se neki zadaci mogu lakše i brže obaviti, bez instalacije novih programa. Osnovna ideja je da se pravi veliki broj malih programa i da se njihovim nadovezivanjem obavljaju složeniji zadaci. Za nadovezivanje naredbi koristi se `|` (pipe), izlaz jedne naredbe prosleđuje se kao ulaz druge naredbe.

PIPE primeri:

`w | sort` – sortira izlaz naredbe `w`
`man -k copy | grep string` – nalaženje komandi koje sadrže `copy` i `string` u svojim kratkim opisima
`man 2 write | grep return` – prikazuje sve linije u pomoćnoj stranici `write` sistemskog poziva koje sadrže `return`
`last mi11002 | grep Mon` – kada je korisnik ulogovan na `alas`, prikazuje sva logovanja ponedeljkom korisnika sa korisničkim imenom `mi11002`