

Programska podrška mjernih i procesnih sustava

vježba br. 1: OSNOVE RADA U LINUX-U

NAPOMENE

- Operacijski sustav Linux se nalazi na CD-u koji je potrebno ubaciti u CD drive odmah nakon uključivanja (ili restartanja) računala.
- Naredbe generalno imaju oblik `naredba opcije argumenti`
- Većinu vježbe odrađujete koristeći terminal odnosno komandnu liniju OS-a.
- Za pojašnjenje pojedinih naredbi u komandnoj liniji upišite naredbu:
`ime_naredbe --help` (ne radi za sve naredbe)
MAN(unal) stranice daju više informacija o pojedinoj naredbi i mogu se dobiti ukoliko se u komandnoj liniji upiše (izlazi se sa tipkom `q`):
`man ime_naredbe`

VJEŽBA:

1. KNOPPIX LIVECD

Knoppix je distribucija Linux operacijskog sustava koja je osmišljena za pokretanje pomoću bootabilnog CD-a, tzv. *LiveCD*. Time je omogućeno korištenje Linux-a bez potrebe za njegovim instaliranjem na tvrdi disk na kojem je obično MS Windows OS.

S obzirom da ne koristi tvrdi disk, za mijenjanje i spremanje datoteka služi *ramdisk*. Zauzme se komad radne memorije računala koji se onda ponaša slično kao particija tvrdog diska. Pristup mu je znatno brži nego disku ali se sav njegov sadržaj gubi resetom računala.

Kada se podigao Linux OS naći ćete se u grafičkom okruženju. Standardni GUI na Linuxu je *X Window System* (skraćeno samo X). Temelji se na klijent-server modelu. Klijenti su grafičke aplikacije a server je zadužen za iscrtavanje prozora na ekran, njihovo pomicanje te hvatanje akcija od miša i tipkovnice. Tako je moguće imati X server i na MS Windows OS-u a aplikacije pokretati sa udaljenog Linux računala. Obično radi lakšeg korištenja, većina Linux distribucija dolaze još i sa nekim od Desktop rješenja. Desktop je set alata i aplikacija koje omogućavaju lakši rad, npr. alat analogan Windows exploreru i sl. Knoppix koristi KDE desktop.

Bitne ikone koje će vam trebati su:

Terminal  te Internet browser .

Ostali alati KDE-a se dohvaćaju preko start ikone .

2. KORISNIČKI RAČUNI

Svaki korisnik Linux operacijskog sustava mora imati svoje korisničko ime i zaporku (password). Knoppix već ima jednog predefiniiranog korisnika imenom "knoppix" uz standardnog administratorskog korisnika imenom "root". Oba korisnika nemaju postavljen password tako da je moguće slobodno se prebacivati iz jedne uloge u drugu. Na samom početku rada ste "knoppix" korisnik.

Pokrenite terminal. Ono što vidite je *prompt ljuske*. Podešen je tako da je na njemu vidljivo ime trenutno aktivnog korisnika, ime računala te put do trenutnog direktorija u kojem se nalazite.

```
knoppix@1[knoppix]$
```

U gornjem primjeru ime korisnika je "knoppix" na terminalu broj 1 u trenutnom direktoriju knoppix (puni put direktorija je /ramdisk/home/knoppix).

U kojem se direktoriju nalazite možete saznati i upisivanjem naredbe `pwd` (*print working directory*). Ime trenutnog korisnika dobijete naredbom `whoami`.

```
$ pwd
$ whoami
```

Za razliku od "običnih" korisnika operacijskog sustava, postoji i superkorisnik `root` i ima najveća prava od svih korisnika sustava. Kako biste promijenili korisnika koristite naredbu `su` (*switch user*) koja, ako ne primi kao parametar ime korisnika, po defaultu prebacuje u `root` korisnika. Na taj način koristite administratorske privilegije kada su potrebne.

```
$ su
```

Uočite da prompt za `root` korisnika sada završava sa znakom `#`. U daljnjem tekstu taj znak ispred naredbe će biti korišten kao oznaka da su potrebne administratorske ovlasti za izvršavanje naredbe. Povratak u ulogu prethodnog `knoppix` korisnika je sa naredbom `exit`.

Popis svih korisnika i njihovih parametara zapisani su u tekstualnoj datoteci /etc/passwd. Možemo je pogledati korištenjem programa `less` ili `cat`.

```
$ less /etc/passwd
$ cat /etc/passwd
```

`less` program izlistava stranicu po stranicu (novi ekran dobijete space tipkom a tipkom `q` izlazite iz njega). `cat` daje cijeli ispis datoteke. Evo primjer nekoliko redaka iz izmišljene /etc/passwd datoteke:

```
root:x:0:0:root:/root:/bin/bash
.
.
student:x:1001:300:Student:/home/student:/usr/bin/zsh
riba:x:2101:2101:Ivica Ribic:/home/riba:/bin/sh
ivek:x:2000:300:Ivica Ivic:/home/ivica:/bin/sh
pero:x:1999:1111:Petar Perovic:/home/pero:/bin/sh
```

Prvo polje (do dvotočke) predstavlja korisničko ime, treće identifikacijski broj korisnika (user ID - uid), četvrto identifikacijski broj grupe (group ID - gid), peto komentar, šesto home direktorij i sedmo ljusku (shell) koja se pokrene nakon što se korisnik logira. Uočite da root ima za uid i gid nulu.

Naravno, korisnik se može i dodati "ručno" tako da se u `/etc/passwd` datoteci doda još jedna linija za novog korisnika.

Koje grupe korisnika postoje možete vidjeti ispisom datoteke `/etc/group`.

```
$ cat /etc/group
```

Primjer dijela ispisa:

```
masoni:x:200:riba,ivek
```

Broj 200 ovdje predstavlja identifikacijski broj grupe. Za svakog korisnika obično postoji istoimena grupa u kojoj je samo on član. Editiranjem te datoteke (`root` privilegije su potrebne) mogu se stvarati nove grupe.

3. DATOTEČNI SUSTAV

Korištenjem naredbi `pwd` (*print working directory*), `cd` (*change directory*), `ls` (*list* - izlista sadržaj direktorija) upoznajte se sa organizacijom direktorija.

Naredbom `ls -l` u nekom direktoriju moguće je dobiti npr. slijedeće:

```
total 5414
drwxr-xr-x  2 root root      512 Aug 28 1998 direktorij1
-rw-r--r--  1 ivek masoni   1595 Nov 24 1997 file3
-rw-rw-r--  1 root root 4574357 Oct 15 1998 audio.wav
```

Na prvoj liniji `total` daje broj u *kilobyte*-ovima koliko zauzima trenutni direktorij.

U ostalim linijama moguće je uočiti stupce. U prvom stupcu je oznaka tipa i prava pristupa datoteci. Direktoriji su isto datoteke. Ukoliko je prvo u nizu slovo `d`, radi se o direktoriju. Ukoliko je prvi znak `-`, radi se o običnoj datoteci.

Slijedećih 9 znakova čitaju se u grupama po tri. Prva grupa od tri slova označava prava pristupa vlasnika dotične datoteke (ili direktorija). Druga grupa od tri slova odnosi se na grupu koja je vlasnik datoteke (ili direktorija). Posljednja se tri slova odnose na sve ostale (osim `roota`, naravno).

Korištena slova `r`, `w`, `x` označavaju dopušteni način pristupa datoteci (ili direktoriju).

Ukratko: `r` - dozvoljeno čitanje

`w` - dozvoljeno mijenjanje

`x` - dozvoljeno pokrenuti izvršne datoteke (ili napraviti `cd` na direktorij)

Vidimo da naš "file3" može mijenjati samo vlasnik, a svi ga ostali mogu samo čitati. Kako možemo vidjeti vlasnik mu je `ivek` i pripada grupi korisnika `masoni`.

Zatim slijedi broj koliko drugih datoteka pokazuje na dotičnu. Kod direktorija je to povezano sa brojem neposrednih poddirektorija. Za običnu datoteku je taj broj obično 1. Nakon toga ide ime vlasnika te ime grupe kojoj datoteka pripada. Na kraju je veličina datoteke u *byte*-ovima, datum kreiranja te njen naziv.

Datoteke koje počinju sa '.' obično su konfiguracijske datoteke nekih programa i smatraju se skrivenim tako da ih `ls -l` neće prikazati. Da biste samo njih vidjeli koristite `ls -a` ili sve zajedno sa `ls -la`.

Isprobajte različite opcije naredbe `ls`:

```
ls -a           //prikazuje i skrivene datoteke (počinju sa .)
ls -l          //detaljniji ispis radnog direktorija
ls -l /etc     //detaljniji ispis /etc direktorija
ls -la        //kombinacija l i a opcije
```

Ako pogledate sadržaj direktorija sa `ls -la` primijetit ćete da na vrhu postoje dvije datoteke '.' i '..'. Prva predstavlja link na vlastiti direktorij a druga na prvi naddirektorij. Tako ako se želimo vratiti jedan direktorij pliće u stablu dovoljno je otipkati `cd ..`.

Korijen stabla direktorija se zove `root` (ne brkati sa imenom `superusera`) i označava sa znakom `/`. On je potreban i za definiranje apsolutnih putova. Ako smo u direktoriju `/ramdisk/home/knoppix` onda do npr. direktorija `/etc` možemo doći bilo apsolutnim putem bilo relativnim.

```
$ cd /etc           // apsolutni put
$ cd ../../../../etc // relativni put
```

Da li koristiti relativni ili apsolutni put najčešće ovisi o položaju željenog direktorija u stablu. Za direktorije koji su dublje u istoj grani stabla prikladniji je relativni put. Za razliku od MS Windowsa gdje svaka diskovna particija ima svoje stablo u Linuxu postoji samo jedno stablo i samo jedan korijen `/` unutar kojeg su integrirani sadržaji svih particija. Tako da sam broj i veličina particija nije odmah vidljiva.

3.1. Tipični raspored direktorija kod Linuxa

FSSTND ili Linux filesystem standard je standard koji preporučuje kako bi trebalo izgledati tipično datotečno stablo Linux sustava. Time je i administriranje Linux strojeva znatno olakšano. Evo nekih najvažnijih direktorija:

`/bin` - naredbe koje trebaju prilikom podizanja sustava, ali koje su korištene i od običnih korisnika

`/sbin` - slično kao i `/bin` samo su naredbe uglavnom namijenjene za administratora

`/etc` - (*everything to configure*) konfiguracijske datoteke specifične za dotično računalo

`/dev` - *device* datoteke

`/tmp` - privremene datoteke

`/usr` - sadrži sve naredbe, biblioteke, man stranice. Uglavnom, sve datoteke koje se ne mijenjaju. Pošto se u ovom direktoriju ne nalaze datoteke specifične za dotično računalo moguće je ovaj direktorij dijeliti preko mreže s drugim računalima i time znatno uštedjeti na zauzeću ostalih tvrdih diskova. Pronađite gdje se u ovom direktoriju nalaze naredbe, a gdje man stranice.

`/var` - sadrži podatke koji se mijenjaju za vrijeme rada sistema.

`/home` - matični direktoriji korisnika se nalaze ovdje.

Pregledajte sami sadržaj tih direktorija. Uočiti ćete kako mnoge datoteke nemaju nikakvu ekstenziju. Za razliku od MS Windowsa gdje na temelju ekstenzije se određuje tip datoteke Linux može automatski odrediti tip gledajući njen sadržaj a ne ime tako da ekstenzija nije neophodna.

4. RAD S DATOTEKAMA

U svojem korisničkom direktoriju kreirajte poddirektorij naredbom `mkdir` (briše se sa `rmdir`).

```
$ mkdir moj_novi_direktorij
$ cd moj_novi_direktorij
```

Korištenjem tekstualnog editora `joe` napišite nekoliko redova teksta:

```
$ joe mojtekst
```

```
^K H dobije se popis naredbi (^ je oznaka za Ctrl tipku)
^K X Save and Exit
^K C Exit bez da se sačuvaju promijene
```

Nakon što ste snimili vašu tekstualnu datoteku `mojtekst` naredbom `ls -l` ispitajte kojem vlasniku pripada i kojoj grupi:

```
$ ls -l mojtekst
```

Isti postupak (sa drugim nazivom datoteke) ponovite kao `root`. Sada imate dvije slične datoteke, ali sa drugačijim pravima pristupa. Tko smije pristupati kojoj datoteci?

Evo nekih naredbi za baratanje datotekama: `cp` (*copy*), `mv` (*move*), `rm` (*remove*, brisanje). Pomoću slijedećih naredbi ćemo kopirati datoteku `mojtekst` u datoteku `tekst2`, preimenovati datoteku `tekst2` u `tekst3` i obrisati datoteku `tekst3`:

```
$ cp mojtekst tekst2
$ mv tekst2 tekst3
$ rm tekst3
```

Isprobajte slijedeće naredbe za kontrolu prava pristupa (rezultat vidjeti sa `ls -l`):

```
$ chmod 777 mojtekst
$ chmod 755 mojtekst
```

```
$ chmod 711 mojtekst
```

Brojčani kod predstavlja prava pristupa koja će se primijeniti na datoteku `mojtekst`. Prvi se broj odnosi na vlasnika, drugi na grupu, treći na sve ostale.

Objašnjenje: 751 --> 111 101 001 --> `rwxr-x--x`

Vlasništvo nad nekom datotekom i grupa kojoj ta datoteka pripada može se promijeniti slijedećim naredbama:

```
# chown novi_vlasnik ime_datoteke
# chgrp grupa ime_datoteke
```

Naredbom `ls -l` pogledajte promjene.

Evo kako omogućiti da svi mogu izvršavati datoteku1:

```
$ chmod +x datoteka1
```

Slijedećom naredbom grupa koja je vlasnik datoteke2 dobit će pravo pisanja po njoj:

```
$ chmod g+w datoteka2
```

Koriste se slijedeće oznake: `u` za korisnika, `g` za grupu i `o` za ostale.
Što rade slijedeće naredbe?

```
$ chmod -x datoteka1
$ chmod ug+rx datoteka2
$ chmod o+rw datoteka3
```

Traženje datoteke je moguće naredbom `find`. Prvi argument je ishodišni direktorij od kuda počinje traženje zatim slijedi sam upit. Primjer ako tražimo datoteku `mojtext` počevši od trenutnog direktorija.

```
$ find . -name 'moj*'
```

5. LJUSKA

Da bi se komande mogle izvršavati, zaslužna je ljuska (engl. *shell*) koja interpretira i izvršava komande. Postoji više vrsta shellova:

```
sh - Bourne shell
bash - Bourne Again shell
csh - C shell
ksh - K shell
tcsh - poboljšan csh
zsh - najnoviji shell
```

Ukoliko vas zanima u kojoj se ljusci trenutno nalazite izdajte naredbu:

```
$ echo $SHELL
```

Većina ljuški podržava dohvat starih napisanih naredbi pomoću **tipki sa strelicama gore/dolje** te dopunjavanje riječi.

Dopunjavanje riječi. Vjerojatno ste do sada primijetili koliko je naporno tipkati naredbe u komandnoj liniji. Kako ne biste morali toliko tipkati evo jednoga trika. Upišite slijedeće bez da pritisnete Enter tipku:

```
$ cd /usr/lo
```

Sada pritisnite Tab tipku. Vaša naredba će se dopuniti do punog imena postojećeg direktorija: `/usr/local/`. Važno je samo da napišete dovoljno slova tako da postoji samo jedan mogući izbor datoteke ili direktorija.

Ako naredba `cd` ne dobije nikakav argument onda će se vratiti na korisnikov `home` direktorij. Npr. slijedeći niz naredbi će vas prvo maknuti iz matičnog direktorija a zatim vratiti nazad. Također `home` direktorij se još zove i

```
$ cd /etc
$ cd
```

Preusmjeravanje ulaza i izlaza iz programa. Evo kako preusmjeriti izlaz iz `ls` naredbe u tekstualnu datoteku:

```
$ ls -la > datoteka1
```

Ukoliko sa `less` programom pogledate sadržaj datoteke, vidjet ćete da se u njoj nalazi ispis koji je dala `ls` naredba. Ukoliko `datoteka1` već postoji, njezin će sadržaj biti prebrisan novim sadržajem, a ukoliko ne postoji bit će stvorena. Ponekad je potrebno samo dopuniti postojeću datoteku. To se postiže korištenjem operatora `>>`:

```
$ ls -la /etc >> datoteka1
```

Isto tako, izlaz iz jednog programa možemo preusmjeriti na ulaz drugog. To se postiže operatorom `|`.

```
$ du | sort -rn
```

Ovime smo izlaz programa `du` (engl. disk usage) koji daje podatak koliko koja datoteka zauzima blokova u trenutnom direktoriju, preusmjerili na ulaz programa koji sortira podatke po numeričkom redosljedu (opcija `-n`) i zatim ih prikaže suprotnim redosljedom (opcija `-r`).

Ovaj se operator zove cjevovod (engl. pipe). Ukoliko teksta ima previše, možemo ga pogledati programom `less`:

```
$ du | sort -rn | less
```

Naredbe. Naredbe u Linux-u su (u većini slučajeva) obične datoteke. Na primjer, naredba `ls` je binarna datoteka koja se nalazi u direktoriju `/bin`. Dakle, mogli smo je izvršiti i ovako:

```
$ /bin/ls
```

Da bi se naredbu moglo pokrenuti bez upisivanja imena direktorija u kojem se ona nalazi, potrebno je da je ljuška automatski traži u tom direktoriju. Popis direktorija u kojima ljuška traži naredbe naziva se put (engl. path). Evo kako se može vidjeti kako izgleda put na računalu:

```
$ echo $PATH
```

Na ekranu ćete dobiti popis direktorija odvojenih dvotočkom, nešto slično ovome:

```
/sbin:/bin:/usr/bin:/usr/bin/X11:/usr/local/sbin:/usr/local/bin:  
/usr/X11R6/bin:/usr/games:.
```

Ova će naredba dodati još jedan direktorij vašem putu:

```
$ export PATH=$PATH:/usr/sbin
```

Trenutni direktorij '.' je u PATH-u na zadnjem mjestu iz sigurnosnih razloga. Ako imamo slučaj da se program u trenutnom direktoriju zove isto kao i neki drugi program koji je dohvatljiv preko PATH-a onda je potrebno dodatno obilježiti ga sa apsolutnim ili relativnim putem. Npr. da smo sami napisali program koji se zove 'ls' u lokalnom direktoriju onda da ga pokrenemo bi ga trebali pozvati kao:

```
$ ./ls // relativni put
```

Postoje i neke naredbe koje nisu datoteke. Jedna od njih je `cd`. One su dio ljuške i zato se nazivaju ugrađenim (engl. built-in) komandama.


6. ZAVRŠETAK RADA NA RAČUNALU

Na završetku svake vježbe potrebno je ugasiti računalo. To se može napraviti na dva načina.

Prvi je korištenjem naredbi iz komandne linije. To se postiže naredbom (kao `root` korisnik):

```
# shutdown -h now
```

`-h` označava `halt` a ako želimo `restart` onda treba staviti opciju `-r`.

Drugi način je korištenjem KDE desktopa. Pritisne se ikona za start menu  i na samom dnu je `logout` tipka.