

Laboratorijska vežba br.2 – Fajl sistemi i rad sa datotekama

Teoretski deo

Fajl sistemi

Da bi particija mogla da se koristi za čuvanje podataka potrebno kreirati fajl sistem. FS čine: **zaglavlje** (u kome se nalazi najmanje podataka, ali su ti podaci neophodni za funkcionisanje sistema datoteka), **strukture za organizaciju podataka na medijumu** (meta-data area) i sami **podaci**, odnosno datoteke i direktorijumi.

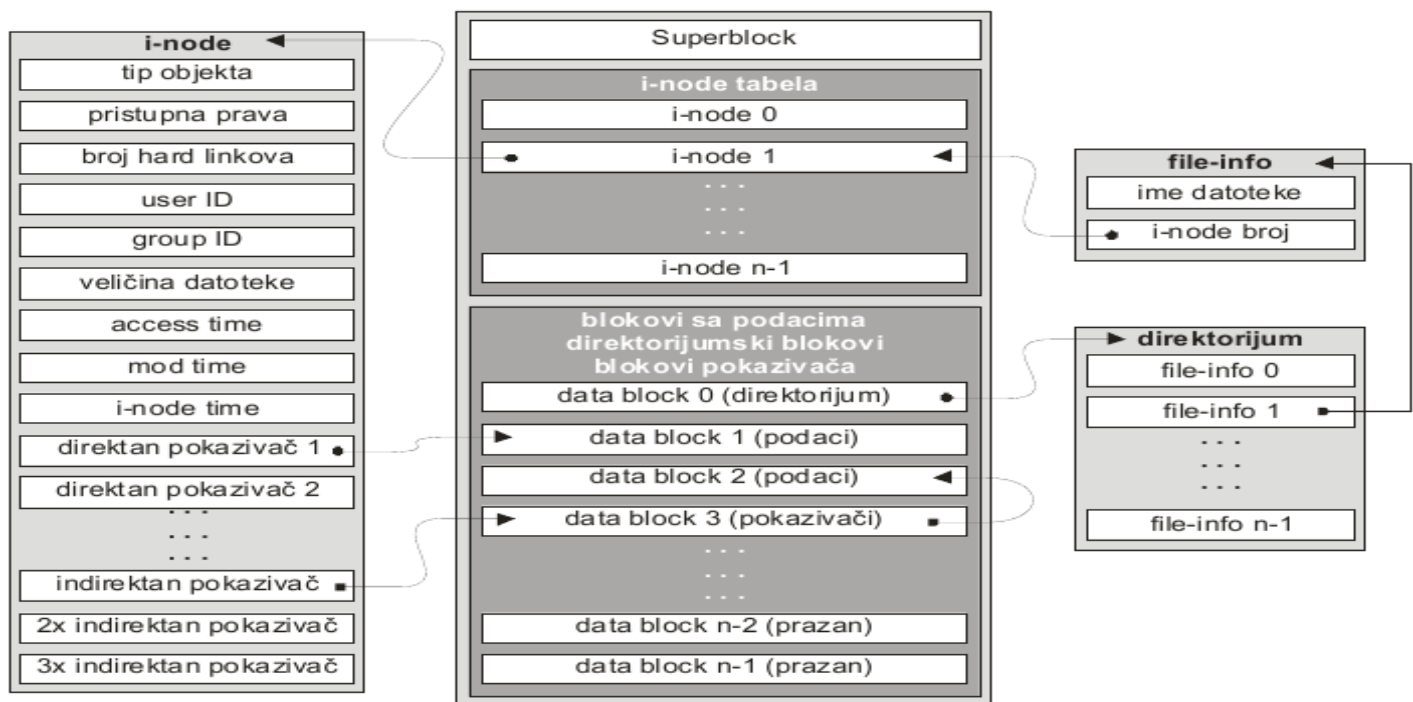
UNIX sistem datoteka čine:

- zaglavlje (engl. *superblock*)
- tabela indeksnih čvorova (*i-node* tabela)
- blokovi sa podacima (engl. *data blocks*)
- direktorijumski blokovi (engl. *directory blocks*)
- blokovi indirektnih pokazivača (engl. *indirection block*)

I-node sadrži sve informacije o objektu koji opisuje osim imena:

- tip objekta (npr. regularna datoteka, direktorijum) i pristupna prava za tri vlasničke kategorije
- broj hard linkova na dati objekat
- user ID, group ID
- veličinu objekta izraženu u bajtovima
- vreme zadnjeg pristupa objektu (access time), vreme zadnje modifikacije objekta (mod time) i vreme zadnje modifikacije indeksnog čvora objekta (i-node time)
- listu pokazivača: direktnih, na blokove sa podacima, koja je dovoljna da se adresiraju prvih 10- 12 blokova podataka koji čine početak datoteke (broj zavisi od tipa sistema datoteka) i indirektnih (na jednostruke, dvostruke i trostruke indirektne blokove)

Pogledajte sličicu, prilično je jasna.



UNIX posmatra svaki sistem datoteka kao nezavisnu hijerarhijsku strukturu objekata (direktorijuma i datoteka) na čijem se vrhu nalazi root direktorijum (/). Objektima se pristupa pomoću relativne ili apsolutne putanje i imena objekta.

Datoteke i direktorijumi

U objekte UNIX sistema datoteka spadaju:

- **regularne datoteke**,
- **direktorijumi** (mogu se posmatrati kao specijalne datoteke koje sadrže objekte sistema datoteka, uključujući i poddirektorijume),
- hard linkovi (alternativna imena datoteka) i simbolički linkovi (prečice)
- blok i karakter specijalne datoteke (opisuju uređaje, odnosno drajvere u kernelu).
- imenovani pipeline.

Imena datoteka

Dozvoljeni karakteri su: sva velika slova (A-Z), sva mala slova (a-z), cifre (0-9), crta (-), underscore (_) i tačka (.).

- Opcije UNIX komandi počinju jednom (-) ili dvema crtama (--), tako da imena objekata ne treba počinjati tim karakterima.
- Ukoliko ime objekta počinje tačkom (.), on se smatra skrivenim i može se videti pomoću komande ls samo ako je zadata s parametrom -a (all). Datoteke koje služe za inicijalizaciju i podešavanje radnog okruženja najčešće su skrivene (.bash_profile, .bash_history);

Zabranjeni karakteri su: ! “ ‘ ` ; : / \ \$ < > () [] { } ~

Ukoliko se umesto imena direktorijuma u apsolutnoj putanji navede ~ korisnik će biti preusmeren na home direktorijum.

- Na primer, komanda less ~/myfile.txt prikazaće sadržaj datoteke myfile.txt koja se nalazi u home direktorijumu korisnika koji je komandu zadao, bez obzira na tekući direktorijum.

Vrste datoteka

UNIX u komandnoj liniji nema pojam ekstenzije, već samo imena objekta. Datoteke nmap.tar.gz i sample.mpg posmatraju se kao regularne datoteke, bez obzira na sadržaj, pri čemu se tip datoteke može odrediti pomoću komande file.

Pojam ekstenzije se uvodi pod grafičkim radnim okruženjem, u smislu asociranja datoteka sa programom na osnovu ekstenzija (npr. datoteka sa ekstenzijom doc biće otvorena u Open Office Writeru, a ne u C kompajleru). Direktorijumi najčešće nemaju ekstenzije, iako ne postoji neko pravilo kojim bi se to eksplicitno zabranilo.

Na UNIX sistemima postoji nekoliko osnovnih tipova regularnih datoteka:

- **tekstualne datoteke** - ASCII (neformatiran tekst), English text (tekst sa interpunkcijskim karakterima) i izvršni shell programi
- **izvršne (binarne)** datoteke
- **datoteke u koje su smešteni podaci** (na primer, Open Office Writer dokument)

Prilikom određivanja tipa datoteke komanda file izvršava sledeće testove: test specijalnih datoteka (filesystem test), test magičnih brojeva (magic number test) i jezički test (language test).

Sledeći primeri ilustruju upotrebu komande file:

\$ file /etc/hosts

/etc/hosts: ASCII English text

\$ file /bin/cp

/bin/cp: ELF 32-bit LSB executable, Intel 80386, version 1 (SYSV),
dynamically linked (uses shared libs), stripped

\$ file /dev/hda

/dev/hda: block special (3/0)

\$ file /dev/lp0

/dev/lp0: character special (6/0)

\$ file /sbin/rmt

/sbin/rmt: symbolic link to /usr/sbin/rmt

\$ file /etc

/etc: directory

Aktivno stablo

Fajl sistemi se pod Linuxom aktiviraju montiranjem (engl. *mounting*) na aktivno UNIX stablo, tj. na mount-point direktorijume. Početak stabla je root direktorijum na koji se “lepi” tzv ROOT fajl sistem (sadrži instalaciju Linuxa). U njemu admin kreira poddirektorijume na koje “montira” tj lepi druge fajl sisteme – uključujući i DVD uređaje, fleš diskove itd.

Npr kad uđete u fleš u komp i hoćete da vidite sadržaj, KDE će “montirati” fleš npr na direktorijum /media/KINGSTON. Slično, ako ubacite DVD, KDE će DVD sa imenom movies montirati na /media/movies. Ako radite u shellu, onda morate ručno da kreirate neki direktorijum i zalepite tj “montirate” fleš ili DVD na taj direktorijum.

Okvirna analiza značajnijih direktorijuma u aktivnom stablu

Root FS sadrži osnovnu direktorijumsku strukturu stabla i sve datoteke neophodne za podizanje operativnog sistema. Sadržaj root sistema datoteka je najčešće dovoljan za jednokorisnički rad (singleuser mode) koji se koristi u svrhe administracije ili oporavka sistema, tako da se u okviru njega nalaze i osnovni administrativni alati. Root direktorijum ne sadrži datoteke, osim simboličkog linka /vmlinuz na sliku jezgra (*boot image*).

Sledi opis značajnih direktorijuma root sistema datoteka.

- /bin - najčešće korišćene komande koje mogu koristiti regularni korisnici.
- /boot - datoteke koje koristi boot loader, uključujući i slike kernela. Ako se OS podiže sa neke “egzotike” npr neki raid realizovan preko device mappera, onda se ovo OBAVEZNO realizuje kao zasebni FS.
- /dev - specijalne datoteke koje predstavljaju uređaje (nodes), koje se kreiraju prilikom instalacije Linux sistema, a imena dobijaju na osnovu specijalnih konvencija.
- /etc - konfiguracione datoteke, sve u tekstualnom obliku. Izvolite, probajte tako da radite administraciju linuxa =)
- /home - lični direktorijumi korisnika, npr ako je username stojko, onda je home direktorijum /home/stojko
- /lib - deljene biblioteke neophodne za rad programa iz root sistema datoteka, /lib/modules - kernel moduli.
- /mnt - direktorijum u kome se nalaze mount-point direktorijumi (npr. /mnt/winc, /mnt/floppy, /mnt/cdrom). U nekim distribucijama Linux sistema ovaj direktorijum je preimenovan u /media.
- /tmp - privremene datoteke.
- /root - lični direktorijum korisnika root (superuser).

- /sbin - komande namenjene superuseru, ali ih po potrebi mogu koristiti i obični korisnici ukoliko im se za to daju dozvole. /sbin se ne nalazi u putanji regularnih korisnika, ali se nalazi u putanji superusera.
- /usr - većina korisničkih komandi, prateće biblioteke i dokumentacija (man pages).
- /var - spool direktorijumi, log datoteke, slične stvarčice.

Shell basics and workaround

NAPOMENA: Ovo je jako bitno da shvatite. Neophodno je za dalji rad u konzoli.

Komandni interpreter je proces koji obavlja sledeće funkcije u cilju obezbeđivanja interfejsa između korisnika i sistema: interpretaciju komandne linije, pokretanje programa, redirekciju ulaza i izlaza, povezivanje komandi u pipeline, zamenu imena datoteka, ...

Tekst unet u shell prompt naziva se **komandna linija** (command line), čiji je opšti oblik:

\$ command [opcije] [argumenti]

- \$ je odzivnik (**prompt**) i prikazuje se svaki put kad je shell spreman da od korisnika prihvati novu komandu.
- Command može biti **interna** komanda (ugrađena u shell) ili **eksterna** (realizovana kao poseban program koji se nalazi u sistemskoj putanji).
- Opcije i argumenti su parametri koje shell prenosi komandi, pri čemu su argumenti najčešće obavezni i predstavljaju ime neke datoteke, direktorijuma, korisnika ili, na primer, identifikator procesa.

Ime komande, opcije i argumenti su osetljivi na velika i mala slova (case-sensitive) i razdvajaju se razmakom.

\$ **free** # samo komanda

\$ **df /dev/hda1** # komanda (fd) i argument (/dev/hda1)

\$ **cp 1.txt 2.txt** # komanda (cp) i dva argumenta (1.txt i 2.txt)

\$ **date -u** # komanda (date) i opcija (-u)

\$ **ls -l /etc** # komanda (ls), opcija (-l) i argument (/etc)

Nakon interpretacije komandne linije shell inicira izvršenje zadate komande. Ukoliko komanda nije interna (ugrađena u shell, poput komande cd) shell traži izvršnu datoteku koja odgovara imenu komande u direktorijumima navedenim u sistemskoj putanji (promenljiva \$PATH). Nakon toga shell pokreće program i prosleđuje mu argumente i opcije navedene u komandnoj liniji. Ukoliko se izvršna datoteka nalazi u tekućem direktorijumu ili u nekom direktorijumu koji nije u sistemskoj putanji, ime komande se mora zadati sa putanjom:

\$ **./myscript** # pokretanje programa iz tekućeg direktorijuma

\$ **/usr/sbin/useradd** # pokretanje programa iz /usr/sbin

Više UNIX komandi mogu se navesti u istoj komandnoj liniji ukoliko su razdvojene znakom tačkazarez.

\$ **clear ; date** # dve komande koje se izvršavaju jedna za drugom

Redirekcija ulaza i izlaza

UNIX komande primaju podatke sa **standardnog ulaza** (stdin, najčešće tastatura) a rezultate izvršenja šalju na **standardni izlaz** (stdout, najčešće ekran). UNIX omogućava da se ulaz i izlaz komande preusmere: Ulaz komande preusmerava se pomoću znaka < (manje od). Na primer, moguće je standardni ulaz sa tastature zameniti nekom tekstualnom datotekom:

\$ **wc -l < /tmp/jsnmith.dat**

Za redirekciju izlaza se koristi znak > (veće od). Ukoliko se redirekcija vrši u postojeću datoteku datoteka se briše, a zatim se kreira nova u koju se smešta rezultat izvršenja komande. Ukoliko korisnik želi da se rezultat izvršenja doda na postojeću datoteku bez brisanja njenog sadržaja, za redirekciju izlaza koristi se znak >>.

```
$ sort kyuss.txt > /dev/lp0      # redirekcija izlaza na štampač
$ ls -l /home/jsmith > myfile   # redirekcija izlaza u novu datoteku
$ ls -l /tmp/jsmith >> myfile    # redirekcija izlaza u postojeću datoteku
```

Povezivanje komandi u pipeline

Pipeline funkcioniše na sledeći način: standardni izlaz komande sa leve strane znaka pipe – cev (|) postaje standardni ulaz komande sa desne strane znaka. Na primer, pretpostavimo da korisnik želi da zna koliko datoteka ima u direktorijumu u /etc. Korisnik može izlistati sadržaj direktorijuma u takozvanom long-listing formatu komandom `ls -l /etc` i izvršiti redirekciju izlaza u privremenu datoteku `/tmp/files_in_etc`. Dalje, korisnik može prebrojati broj linija u privremenoj datoteci pomoću komande `wc -l /tmp/files_in_etc` (word count).

```
$ ls -l /etc > /tmp/files_in_etc
$ wc -l < /tmp/files_in_etc
145
```

Prva komanda koristi datoteku `/tmp/files_in_etc` kao standardni izlaz, a druga kao standardni ulaz, što znači da se prebrojavanje datoteka u direktorijumu `/etc` može realizovati pomoću pipeline sprege komandi `ls` i `wc`:

```
$ ls -l /etc | wc -l
145
```

U pipeline se može povezati veći broj komandi. Sledeći primer ilustruje pipeline koji prebrojava sve korisnike koji su prijavljeni na sistem a ime im počinje slovom `a` i prikazuje ih sortirane po abecednom redu.

```
$ finger | grep a* | wc -l | sort
```

I pipeline i redirekcija kao funkcije komandnog interpretera imaju svoje prednosti. Preusmeravanje se koristi kad se koristi podatak iz datoteke koja će se i dalje koristiti. U suprotnom se koristi pipe, čijom se upotrebom sprečava nepotrebno gomilanje podataka na diskovima.

Džokeri

Imena datoteka mogu se zameniti džoker karakterima (joker) `*`, `?` i `[]`. Argument komande koji sarži džoker karakter zamenjuje se odgovarajućom listom datoteka shodno pravilima zamene. Komandni interpreter izvršava ovu zamenu pre izvršavanja same komande, odnosno pre pokretanja programa. karakter `*` menja bilo koji niz znakova proizvoljne dužine

```
# ls -d /var/s*
/var/spool /var/stat
karakter ? menja bilo koji znak (tačno jedan znak)
# ls -d /*bin
/sbin
opseg [poc-kraj] menja tačno jedan znak koji pripada tom opsegu. Opseg se ne sme zadati uopadajućem redu.
# ls -d /etc/[a-d][a-d]*
```

```
/etc/adduser.conf /etc/bash.bashrc /etc/bash_completion.d
/etc/adjtime /etc/bash_completion /etc/calendar
```

Kompletiranje imena datoteka

Bourne-again shell sadrži mogućnost za kompletiranje imena datoteka. Korisnik može da unesesamo nekoliko prvih karaktera imena datoteke, a zatim da pritisne karakter <Tab> kojim se naznačava shellu da završi ime datoteke. Na primer:

```
$ ls -l /etc/pas<Tab>
$ ls -l /etc/passwd
```

Ukoliko shell u tekućem direktorijumu pronade više od jedne datoteke čije ime počinje tim karakterima, korisnik će morati da unese još nekoliko karaktera u imenu datoteke, a zatim da ponovo pritisne taster <Tab>.

Rad sa datotekama

NAPOMENA: *Midnight commander je blagodet za rad sa datotekama. Ukoliko nije instaliran i sistem nema GUI, onda preostaje da se sa fajl managementom izborite pomoću ovih komandi.*

Kopiranje datoteka

Komanda cp služi za kopiranje datoteka i direktorijuma. U opštem slučaju sintaksa komande cp je:
cp SOURCE DESTINATION

Komanda cp kopira izvorišni objekat sistema datoteka (SOURCE) u odgovarajuće odredište (DESTINATION).

Napomena: parametri SOURCE i DESTINATION moraju se navesti. Ukoliko korisnik želi da iskopira datoteku u tekući direktorijum, kao odredišni argument može navesti tačku.

U zavisnosti od parametara SOURCE i DESTINATION postoje tri osnovna oblika korišćenja komande cp:

- kopiranje jedne datoteke: **\$ cp source_file destination**

Argument source_file je ime izvorišne datoteke, dok argument destination može biti odredišni direktorijum ili ime odredišne datoteke. Argumenti sourcefile i destination (ukoliko je destination ime datoteke) opciono mogu sadržati ili apsolutne ili relativne putanje.

Primeri:

```
$ cp a.a b.b # kopira fajl a.a u b.b
```

```
$ cp /home/a.a /tmp/b.b # kopira fajl a.a iz /home u /tmp kao fajl b.b
```

```
$ cp a.a /tmp/a.a # kopira fajl a.a iz tekućeg direktorijuma u tmp
```

Napomena: ne može se kreirati istoimena kopija datoteke na istom direktorijumu (datoteka se ne može kopirati u samu sebe).

- kopiranje grupe datoteka iz istog direktorijuma: **\$ cp source_files dest_dir**

Grupa izvorišnih datoteka navodi se kao argument source_files koji se formira pomoću džoker karaktera (*, ?, [a-z], [A-Z], [0-9]). Džokeri mogu da zamene jedan ili više karaktera ili opseg (pogledati funkcije komandnog interpretera - zamena imena datoteka). Odredišni objekat u ovom režimu rada mora biti **direktorijum** (apsolutna ili relativna putanja) ispecificira se argumentom dest_dir.

Primeri:

```
$ cp a* /tmp # kopira sve fajlove koji počinju na a u /tmp
```

```
$ cp a?.old /tmp # kopira sve fajlove od dva karaktera koji počinju na a u /tmp
```

```
$ cp /etc/[a-d]* . # kopira sve fajlove koji počinju sa a,b,c ili d u tekući dir
```

Napomene: 1. kopira se grupa datoteka iz iste grane. 2. Nije moguće kopirati grupu datoteka sa promenom imena kopija (npr cp a* /tmp/b*). 3. Originali i kopije moraju biti na različitim direktorijumima.

- rekurzivno kopiranje direktorijumskog stabla.

\$ **cp -r(R) source_dir dest_dir**

\$ **cp -r source_files dest_dir**

Npr:

- kopiranje direktorijuma /etc sa svim poddirektorijumima i datotekama u direktorijum /tmp/ oldconfig/etc (datoteka /etc/passwd kopira se u /tmp/oldconfig/etc/passwd),

\$ **cp -r /etc /tmp/oldconfig**

- kopiranje kompletnog sadržaja direktorijuma /etc u direktorijum /tmp/oldconfig (datoteka / etc/passwd kopira se u /tmp/oldconfig/passwd),

\$ **cp -r /etc/* /tmp/oldconfig**

- kopiranje datoteka čije ime počinje sa a iz tekućeg direktorijuma i svih poddirektorijuma u direktorijum /tmp/mybackup.

\$ **cp -r a* /tmp/mybackup**

Napomene: 1. Originali i kopije moraju imati ista imena. Nije moguće izvesti kopiranje sa promenom imena kopija (na primer cp -r /etc/a* /tmp/b*). 2. Originali i kopije moraju biti na različitim direktorijumima.

Pomeranje i promena imena datoteka

Komanda mv služi za pomeranje i promenu imena datoteka i direktorijuma. U opštem slučaju, sintaksa komande mv je:

mv SOURCE DESTINATION

što znači da komanda mv pomera izvorišni objekat sistema datoteka (SOURCE) u odgovarajuće odredište (DESTINATION). Ukoliko je izvorišni objekat datoteka komandom mv joj se može promeniti ime. **Napomena:** parametri SOURCE i DESTINATION moraju se navesti. Ukoliko korisnik želi da pomeri datoteku u tekući direktorijum, kao odredišni argument može navesti tačku.

U zavisnosti od parametara SOURCE i DESTINATION postoje dva osnovna oblika korišćenja komande mv:

- · promena imena i/ili pomeranje jedne datoteke,

\$ **mv source_file destination**

Argument source_file je ime izvorišne datoteke, dok argument destination može biti odredišni direktorijum i/ili novo ime datoteke.

Primeri:

\$ **mv a.a b.b** # promena imena datoteke

\$ **mv /home/a.a /tmp/a.a** # pomeranje datoteke iz jednog direktorijuma u drugi

\$ **mv /home/a.a /tmp/b.b** # promena imena i pomeranje u drugi direktorijum

- pomeranje grupe datoteka iz istog direktorijuma.

\$ **mv source_files dest_dir**

Grupa izvorišnih datoteka navodi se kao argument source_files koji se formira pomoću džoker karaktera (*, ?, [a-z], [A-Z], [0-9]). Odredišni objekat u ovom režimu rada mora biti direktorijum (apsolutna ili relativna putanja) i specificira se argumentom dest_dir.

Primeri:

\$ **mv a* /tmp** # pomera sve fajlove koji počinju na a u /tmp

\$ **mv a?[0-9].old /usr/tmp** # korišćenje džokera

\$ **mv /etc/[a-d]* ./backup** # pomera fajlove koji počinju na a,b,c ili d u ./backup

U ovom režimu rada komanda mv pomera grupu datoteka iz iste grane isključivo u drugi direktorijum. Pri tom originali i kopije moraju imati ista imena. Nije moguće istovremeno izvesti pomeranje grupe datoteka sa promenom imena (na primer mv a* /tmp/b*).

Brisanje datoteka

Komanda **rm** služi za brisanje datoteka i direktorijuma. U opštem slučaju, sintaksa komande **rm** je:

rm OBJECT

što znači da komanda **rm** briše objekat sistema datoteka (OBJECT). Parametar OBJECT se mora navesti, a u zavisnosti od njega postoje tri osnovna oblika korišćenja komande **rm**:

- brisanje jedne datoteke

\$ rm filename

gde je argument **filename** ime datoteke koja će biti obrisana, a opciono može sadržati i apsolutnu ili relativnu putanju datoteke.

Primeri:

\$ rm a.a # briše a.a iz tekućeg direktorijuma

\$ rm /etc/passwd.old # briše passwd.old iz direktorijuma /etc

- brisanje grupe datoteka iz istog direktorijuma

\$ rm group_of_files

Grupa izvorišnih datoteka navodi se kao argument **group_of_files** koji se formira pomoću džoker karaktera (*, ?, [a-z], [A-Z], [0-9]).

Primeri:

\$ rm a* # briše sve fajlove koji počinju na a iz tekućeg dir

\$ rm /etc/*.conf.old # briše iz /etc sve fajlove koji se završavaju sa .conf.old

Napomena: u ovom režimu rada komandom **rm** mogu se obrisati isključivo datoteke koje se nalaze u istom direktorijumu.

- rekurzivno brisanje direktorijumskog stabla

\$ rm -R directory

Primer:

\$ rm -r /etc/backup # briše kompletan direktorijum backup sa svim sadržajem

\$ rm -r /etc/backup/ # briše sadržaj direktorijuma backup ali ne i sam dir.

Direktorijumi

Apsolutna putanja tekućeg direktorijuma može se prikazati pomoću komande **pwd** (print working directory):

\$ pwd

/home/jsmith/mydoc

Tekući direktorijum se menja komandom **cd** (change directory). Sintaksa komande je krajnje jednostavna:

\$ cd dir

Argument **dir** je apsolutna ili relativna putanja direktorijuma na koji korisnik želi da se pozicionira. Na primer, ako se korisnik nalazi u direktorijumu /tmp/backup i zada komandu **cd ../data**, biće prebačen u direktorijum /tmp/data.

Korisnik koji zada komandu bez argumenta **dir** biće premešten u home direktorijum.

Primeri:

\$ cd # prelazak na home direktoriju,

\$ cd ~ # prelazak na poddirektorijum /data u home direktorijumu

\$ cd .. # prelazak na roditeljski direktorijum

\$ cd backup # prelazak na poddirektorijum backup u tekućem direktorijumu

\$ cd / # prelazak na root aktivnog stabla

\$ cd /backup # prelazak na direktorijum backup u rootu aktivnog stabla

Korisnik može na ekranu pomoću komande ls (list) prikazati sadržaj bilo kog direktorijuma aktivnog UNIX stabla. Sintaksa komande ls je:

\$ ls [options] [dir][filespec]

Komanda ls prikazaće na ekranu spisak objekata direktorijuma dir definisanih argumentom filespec. Argument filespec formira se pomoću džoker karaktera i nije obavezan. Ukoliko se ne navede, podrazumevaju se svi objekti u direktorijumu. Argument dir takođe nije obavezan, i ako se ne navede prikazuje se sadržaj tekućeg direktorijuma. Ukoliko se ne navedu dodatne opcije ls prikazuje samo imena objekata sortiranih u abecednom redu. Imena skrivenih objekata (objekti čije ime počinje tačkom) se ne prikazuju, a imena rezervnih kopija datoteka (datoteke čije se ime završava znakom ~) se prikazuju.

Opcije:

- -a prikazuju se i imena skrivenih objekata
- -d prikazuje se kontekst direktorijuma umesto sadržaja
- -i prikazuje se i-node broj datoteke
- -R rekurzivno se prikazuje sadržaj svih poddirektorijuma
- -h veličine datoteka se prikazuju u čitljivom formatu (1K, 234M, 2G)
- -l prikazuju se i informacije upisane u i-node (prava pristupa, vlasnik, grupa, ...)
- -l prikazuje samo imena objekata (jedno ime u jednoj liniji)

Primeri:

```
$ ls -a                # prikazivanje imena svih objekata tekućeg direktorijuma uključujući skrivene
. .alias .bash_profile .cshrc d1 f2
.. .bash_history .bashrc .gnupg f1 l1
```

```
$ ls -l                # prikazivanje konteksta svih objekata tekućeg direktorijuma
total 16
drwxr-xr-x 2 nmacek nmacek 4096 Sep 21 d1
-rw-r--r-- 1 nmacek nmacek 315 Sep 21 f1
-rw-r--r-- 1 nmacek nmacek 4125 Sep 21 f2
lrwxrwxrwx 1 nmacek nmacek 2 Sep 21 l1 -> f1
```

```
$ ls -R                # listanje tekućeg direktorijuma i svih poddirektorijuma
.:
d1 f1 f2 l1
./d1:
f1 l1
```

```
$ ls -dl /etc           # prikazivanje konteksta direktorijuma /etc
drwxr-xr-x 53 root root 3072 May 21 09:58 /etc
```

Direktorijumi se kreiraju komandom mkdir (make directory), čija je sintaksa: **\$ mkdir dir**

Direktorijum se može obrisati komandama rm i rmdir (remove directory). Komandom rmdir može se obrisati isključivo prazan direktorijum. Ukoliko se u argumentu dir ne navede putanja, briše se direktorijum dir iz tekućeg direktorijuma. Komanda rm je moćnija – za razliku od komande rmdir, rekurzivnim brisanjem komandom rm mogu se ukloniti cela direktorijumska stabla sa datotekama.

```
$ rmdir /backup      # briše direktorijum backup samo ako je prazan
$ rm -r /backup      # briše direktorijum backup i kompletan sadržaj
```

Rad sa tekstualnim datotekama – vieweri i editori

NAPOMENA: Ovo je jako bitno da shvatite. Neophodno je za dalji rad što se tiče programiranja u C jeziku na linuxu. Pokušajte da iskoristite i neki editor iz grafičkog okruženja – npr Kate ili Kwrite.

Cat & less

Sadržaj tekstualne datoteke najlakše se može pregledati pomoću programa **cat** i **less**. U zavisnosti od veličine datoteke korisnik će odrediti koji će program koristiti - ukoliko je datoteka kraća i može se prikazati na jednom ekranu, može se koristiti **cat**. U suprotnom, koristi se program **less** koji prikazuje datoteku sa pauzom nakon svakog punog ekrana.

\$ cat /etc/resolv.conf

```
search internal.vets.edu.yu
nameserver 172.16.32.1
```

Komanda **less** pravi pauzu posle svakog ekrana i navigaciju po datoteci unapred i unazad. Nakon pokretanja programa **less** prikazuje prvi ekran sadržaja datoteke, a zatim se koriste interaktivne komande za navigaciju po sadržaju datoteke i napuštanje programa:

- h ili H prikazuje pomoć pri korišćenju interaktivnih komandi
- <Space> prikazuje sledeći ekran sadržaja datoteke
- <DOWN> prikazuje sledeću liniju teksta
- <UP> prikazuje prethodnu liniju teksta
- <RIGHT> horizontalno pomeranje ekrana udesno (ukoliko se ne koristi word wrap)
- <LEFT> horizontalno pomeranje ekrana ulevo (ukoliko se ne koristi word wrap)
- q ili Q napuštanje programa less.

Primer:

```
$ less /etc/passwd
$ ls /etc/ | less
```

Traženje teksta u datoteci

PRIMER: Jedan od mogućih načina kojim se može utvrditi da li određeni korisnik ima nalog na sistemu ili ne jeste pretraživanje datoteke `/etc/passwd`. Ukoliko se datoteka proverava programom **less**, a na sistemu postoji veliki broj korisničkih naloga, ova procedura može potrajati. Jednostavniji i brži način kojim se to može izvesti je korišćenje komande **grep**. Komanda **grep** traži navedeni uzorak teksta u svim linijama datoteke i na ekranu prikazuje one linije u kojima je uzorak nađen. U najjednostavnijem obliku, sintaksa komande **grep** je

\$ grep text filename

gde je text uzorak koji se traži u datoteci filename.

Primer:

\$ cat myfile

Primer ilustruje traženje stringa "only" u datoteci myfile.

\$ grep only myfile

Joe (Joe's Own Editor)

Editor **joe** se pokreće sledećom komandom:

\$ joe [filename]

Ukoliko korisnik zada komandu joe bez argumenata, ili kao argument navede ime nepostojeće datoteke, joe će početi rad na novoj datoteci. Ukoliko se kao argument navede ime postojeće datoteke, joe će datoteku učitati, nakon čega korisnik može da vrši izmene. Sadržaj datoteke na disku ažurira se sadržajem modifikovanim editorom joe tek kada korisnik zada komandu za snimanje datoteke.

Ovaj editor ne koristi režime: tekst se jednostavno unosi (korisnik se nalazi u insert režimu), a komande se zadaju korišćenjem kombinacije tastera <Ctrl-slovo>. U svakom trenutku korisnik može dobiti pomoć kombinacijom tastera <Ctrl-K>, a zatim pritiskom na taster H. Ekran pomoći je prikazan u nastavku teksta:

Help Screen turn off with ^KH more help with ESC . (^.)

CURSOR GO TO BLOCK DELETE MISC EXIT

^B left ^F right ^U prev. screen ^KB begin ^D char. ^KJ reformat ^KX save

^P up ^N down ^V next screen ^KK end ^Y line ^T options ^C abort

^Z previous word ^A beg. of line ^KM move ^W >word ^R refresh ^KZ shell

^X next word ^E end of line ^KC copy ^O word< ^@ insert FILE

SEARCH ^KU top of file ^KW file ^J >line SPELL ^KE edit

^KF find text ^KV end of file ^KY delete ^_ undo ^[N word ^KR insert

^L find next ^KL to line No. ^K/ filter ^^ redo ^[L file ^KD save

Za navigaciju po tekstu mogu se koristiti kursorski tasteri (kao i PgUp i PgDn).

Za rad sa blokovima (clipboard) koriste se sledeće komande:

- <Ctrl-K>, zatim B markiranje početka bloka
- <Ctrl-K>, zatim K markiranje kraja (blok je obeležen ako se markiraju i početak i kraj)
- <Ctrl-K>, zatim M pomeranje bloka na tekuću poziciju
- <Ctrl-K>, zatim C kopiranje bloka na tekuću poziciju
- <Ctrl-K>, zatim Y brisanje bloka

Za snimanje datoteke i napuštanje editora joe koriste se sledeće komande:

- <Ctrl-K>, zatim X snimanje datoteke na disk i napuštanje editora joe
- <Ctrl-C> napuštanje editora bez snimanja datoteke

Praktični zadaci**Saveti**

- Znak “pipe” (|) dobijate kombinacijom AltGr+W ukoliko je raspored tastature srpski.
- Ako niste sigurni u parametre neke komande, pomoć možete dobiti pomoću takozvanih stranica uputstva (manual pages, man pages). Stranica uputstva se poziva komandom: **man useradd** i prikazuje pomoć za komandu koju ste naveli kao argument. Za navigaciju koristite kursorske tastere (strelice). Man stranicu napuštate pomoću tastera q (quit). Ukoliko radite u alfanumeričkom modu i pokušate da napustite man page kombinacijom tastera Alt+F4, preći ćete u drugu konzolu (terminal). Vratite se nazad sa Alt+F1, pa napustite man page tasterom q.
- Program less za pregledanje teksta se napušta pomoću tastera q (quit) a ne kombinacijom tastera Ctrl-C.

Zadaci i uputsva za rad

1. Očistite tragove koje su napravili prethodni učenici, ukoliko postoje.

- Prijavite se sistem
- Zadajte sledeće komande kako biste pripremili sistem za rad. Obratite pažnju: komande zadajete kako bi ste sa sistema “očistili” eventualne tragove koje su za sobom

ostavili studenti koji su vežbali pre vas. Neke od ovih komandi mogu vratiti grešku (npr. korisnik je već obrisani) – ukoliko ste komande zadali kako treba, ignorišite poruke i nastavite sa radom.

userdel korisnik

rm -rf /home/korisnik

• Kreirajte novi korisnički nalog kositeći YaST, KUser ili odgovarajuće shell komande. Parametri su sledeći. username: **korisnik**, password: **123456**, puno ime: **Probni korisnik**, home direktorijum: /home/korisnik, shell:/bin/bash

- Odjavite se sa sistema. Prijavite se na sistem kao novokreirani korisnik. Otvorite program Konsole i nastavite da radite vežbu u njemu.

2. Redirekcija ulaza i izlaza

- Šta radi sledeća komanda: ls -l /etc
- Probajte da izlaz prethodne komande preusmerite u datoteku etclist: ls -l /etc >etclist
- Pogledajte sadržaj datoteke etclist: cat etclist
- Šta radi sledeća komanda: mount
- Probajte da izlaz prethodne komande preusmerite u datoteku mountlist: mount >mountlist
- Pogledajte sadržaj datoteke mountlist: cat mountlist
- Obrišite datoteke koje ste kreirali: rm etclist mountlist

3. Pipeline

- Potrebno je prebrojati sve datoteke u osnovnoj grani direktorijuma /etc. Jedan od načina na koji se to može učiniti je sledeći:
- preusmerite izlaz komande ls -l /etc (koja ispisuje jednu datoteku direktorijuma /etc u svakoj liniji na ekranu) u neku datoteku: ls -l /etc >etc1.txt
- prebrojite linije datoteke: wc -l etc1.txt
- Drugi način je da iskoristite komandni pipeline: ls -l /etc | wc -l
- Drugi primer gde je pipeline neophodan je bilo koja Linux komanda koja proizvodi veliki output. Komanda koja prikazuje sadržaj etc direktorijuma (ls -l /etc) je dobar primer. Izlaztakve komande možete videti stranu po stranu ako uradite sledeće stvari:
- Preusmerite izlaz komande ls -l /etc (koja ispisuje jednu datoteku direktorijuma /etc u svakoj liniji na ekranu) u neku datoteku: ls -l /etc >etc2.txt
- Pregledajte datoteku viewer-om less: less etc2.txt
- Drugo rešenje je da povežete te dve komande u pipeline: ls -l /etc | less
- Obrišite datoteke koje ste kreirali: rm etc1 etc2

4. Upotreba džokera

- Pređite na direktorijum /bin: `cd /bin`
- Prikažite sve datoteke čije ime počinje na slovo a: `ls -l a*`
- Prikažite sve datoteke čije ime počinje slovima ch: `ls -l ch*`
- Šta prikazuje: `ls -l ??mod`
- Šta prikazuje: `ls -l [a-d]*`
- Vratite se na home direktorijum: `cd`

5. Ponavljanje i editovanje komandi

- Namerno pogrešno otkucajte neku komandu, na primer lls: `lls -l /etc`
- Vratite komandu iz bafera (strelica na gore) pa je ispravite i izvršite: `ls -l /etc`
- Gde se čuvaju komande otkucane u prošlosti?
- Pogledajte sadržaj datoteke `.bash_history` u home direktorijumu: `less ~/.bash_history`

6. Kompletiranje imena datoteke

- Otkucati “`cat /etc/pas`” pa pritisnite `<tab>`
- Od Linuxa dobijate kompletirano ime `/etc/passwd`
- Sta znači ako tab ne radi: ili ima vise opcija, ili nema nijedne. Npr: “`cat /etc/pa`” `<tab>`

7. Ispitivanje sadržaja datoteke

Odredite tipove sledećih datoteka:

- `/etc/hosts`: file `/etc/hosts`
- `/bin/cp`: file `/bin/cp`
- `/dev/hda`: file `/dev/hda`
- `/dev/lp0`: file `/dev/lp0`

8. Jednostavno kopiranje

- Kreirajte jednu datoteku u svom home direktorijumu: `touch original`
- Iskopirajte datoteku: `cp original kopija`
- Pogledajte šta piše u i-nodeovima originala i kopije: `ls -l original kopija`
- Obrišite obe datoteke: `rm original kopija`

9. Kopiranje datoteke iz drugog direktorijuma u tekući, sa promenom imena datoteke

- Iskopirajte jednu datoteku čiji je vlasnik korisnik root u svoj home: `cp /bin/cp my_cp`
- Pogledajte šta piše u i-nodeovima originala i kopije: `ls -l /bin/cp my_cp`
- Uporedite vlasničke odnose originala i kopije. Kom korisniku i kojoj grupi je dodeljena kopija ?
- (hint: pogledajte koja je primarna grupa korisnika “korisnik”)
- Obrišite datoteku `my_cp`: `rm my_cp`

10. Upotreba džokera za kopiranje datoteka iz jednog direktorijuma u drugi

- Kreirajte dva direktorijuma u svom home direktorijumu: `mkdir src dst`
- Kreirajte nekoliko datoteka u direktorijumu `src` zadajući redom sledeće komande:
- pređite na direktorijum `src`: `cd src`
- kreirajte datoteke: `touch a1 a2 a3 a4 a5 b1 b2 b3 b4 b5 c1 c2 c3 c4 c5`
- vratite se na svoj home direktorijum: `cd`

- Iskopirajte u direktorijum dst sve datoteke koje počinju na slovo a: `cp src/a* dst/`
- Sada probajte da Iskopirate u direktorijum dst sve datoteke koje počinju na slovo “b”, ali tako da im promenite prvo slovo u “d” (npr b1 se kopira u d1, b2 u d2...): `cp src/b* dst/d*`
- Da li ste u tome uspeli?
- Ako niste, probajte da kopirate datoteku po datoteku i da joj pri tom menjate ime:
- `cp src/b1 dst/d1`
- `cp src/b2 dst/d2-RC5`
- `cp src/b3 dst/d31`
- `cp src/b4 dst/d41`
- `cp src/b5 dst/d5-tux`

11. Još malo priče o džokerima i njihovoj primerni na komandu ls

- Prikažite sve datoteke u direktorijumu dst: `ls -l dst`
- Prikažite sve datoteke u direktorijumu dst čije je ima dužine dva karaktera: `ls dst/?`
- Prikažite sve datoteke u direktorijumu dst čije ime počinje na “d”: `ls dst/d*`
- Prikažite sve datoteke u direktorijumu dst čije ime počinje na “d” i dužine je 2 karaktera: `ls dst/d?`
- Prikažite sve datoteke u direktorijumu dst u čijem je imenu drugi karakter “2”: `ls dst/?2*`
- Obrišite kompletne direktorijume dst i src: `rm -rf src dst`

12. Pomeranje jedne datoteke, grupe datoteka i direktorijuma

- Kreirajte dva direktorijuma u svom home direktorijumu: `mkdir src dst`
- Kreirajte nekoliko datoteka u direktorijumu src zadajući redom sledeće komande:
- pređite na direktorijum src: `cd src`
- kreirajte datoteke: `touch a1 a2 a3 a4 a5 b1 b2 b3 b4 b5 c1 c2 c3 c4 c5`
- vratite se na svoj home direktorijum: `cd`
- Pomerite datoteku a1 u direktorijum dst: `mv src/a1 dst/`
- Pomerite datoteku a2 u direktorijum dst i promenite joj ime: `mv src/a2 dst/a2_v1.01`
- Promenite ime datoteke a3 u c6: `mv src/a3 src/c6`
- Pomerite grupu datoteka kojima ima počinje na “b” u direktorijum dst: `mv src/b* dst/`
- Probajte da pomerite grupu datoteka i da im promenite ime: `mv src/c* dst/d*`
- Da li ste uspeli u tome?
- Probajte da pomerite ceo poddirektorijum: `mv src dst/`
- Da li ste uspeli u tome?
- Obrišite poddirektorijume src i dst: `rm -rf src dst`

13. Komanda rm – brisanje datoteka i poddirektorijuma

- Kreirajte dve datoteke, direktorijum i u njemu još dve datoteke: `touch a1 a2 b1 b2; mkdir dir1; touch dir1/c1 dir1/c2`
- Probajte da obrišete dve datoteke sa flegom -i: `rm -i a*`
- Probajte da obrišete dve datoteke sa flegom -f: `rm -f b*`

U ČEMU JE RAZLIKA IZMEĐU OVA DVA FLEGA?

- Probajte da obrišete direktorijum (koji nije prazan) komandom `rmdir`: `rmdir dir1`
- Jeste li uspeli u tome? Probajte ovako: `rm -rf dir1`

14. Rad sa direktorijumima – parametri komande ls

- Šta prikazuju sledeće dve komande:
 - ls /etc
 - ls -l /etc
- Opcija -a (prikazuje skrivene datoteke): pređite na home direktorijum: cd
- izlistajte sadržaj: ls -l
- prikažite i skrivene datoteke: ls -al
- Opcija -d (prikazuje kontekst direktorijuma umesto sadržaja):
- izlistajte sadržaj: ls -l /tmp
- prikažite kontekst: ls -ld /tmp
- Prikažite i-nodove sledećih datoteka: ls -li /bin/cp /tmp
- Probajte sadržaj direktorijuma po dubini (recursively): ls -R /usr

15. Navigacija po aktivnom stablu – komanda cd

- Pozicioniranje na home direktorijum: cd Proverite gde ste pozicionirani: pwd
- Pozicioniranje na roditeljski direktorijum: cd .. Proverite gde ste pozicionirani: pwd
- Pozicioniranje na direktorijum /etc: cd /etc Proverite gde ste pozicionirani: pwd
- Relativna putanja do /usr: cd ../usr Proverite gde ste pozicionirani: pwd

16. Kreiranje direktorijuma – komanda mkdir

- Pređite na home: cd
- Kreirajte jedan direktorijum: mkdir newdirectory
- Kreirajte stablo: mkdir -p 1/11/111
- Proverite šta je prethodna komanda napravila, tj pokušajte da objasnite čemu služi fleg -p
- Obrišite direktorijum: rmdir newdirecory
- Obrišite stablo: rm -rf 1

17. Cleaning and housekeeping

Pijavite se na sistem kao root korisnik i obrišite korisnika “korisnik” i njegov home direktorijum.

Uputstvo:

- Otvorite konzolu
- Zadajte sledeće dve komande:

userdel korisnik

rm -rf /home/korisnik

- Zatvorite konzolu tj program Konsole.