

Linux

2

Osnovi rada sa datotekama i direktorijumima (drugi deo)

Sadržaj:

1. Osnovni pojmovi Linux sistema datoteka
2. O aktivnom stablu i aktiviranju fajl sistema
3. Kopiranje datoteka
4. Pomeranje i promena imena datoteka
5. Brisanje datoteka
6. Rad sa direktorijumima
7. Linkovi

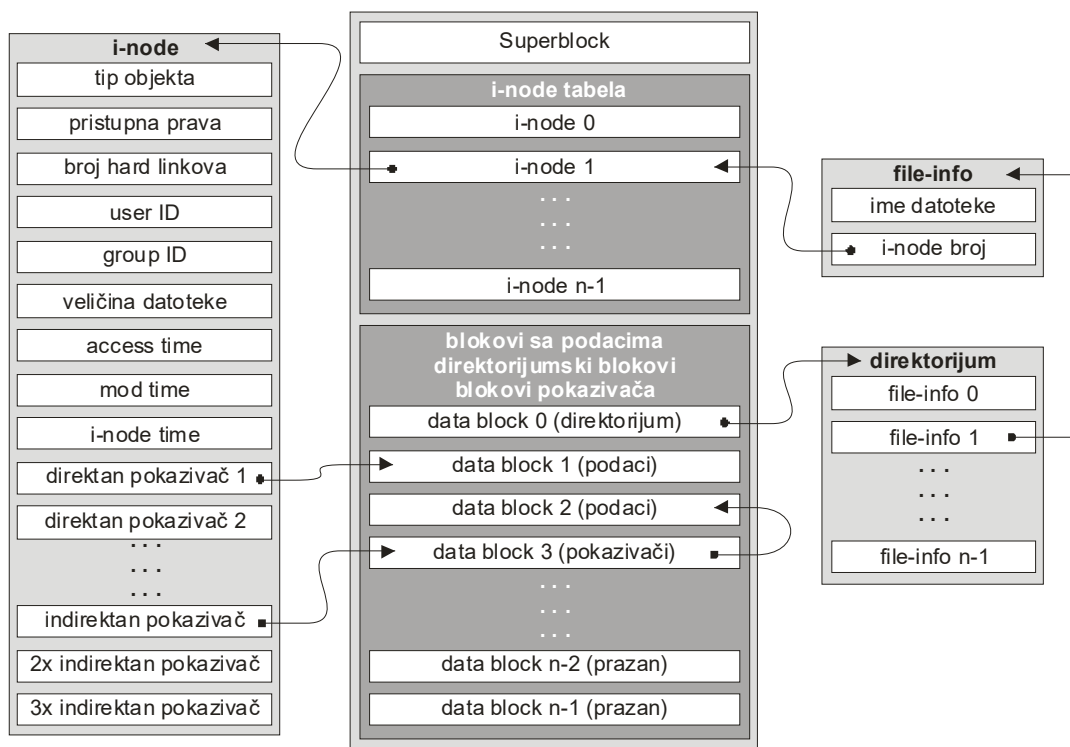
1. Osnovni pojmovi Linux sistema datoteka

NAPOMENA: Pročitajte ovo više informativno i pokušajte da shvatite suštinu bez zalaženja u detalje!

Da bi particija mogla da se koristi za čuvanje podataka potrebno kreirati fajl sistem. Fajl sistem 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.

Linux sistem datoteka čine:

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



Slika 1. Skica Linux sistema datoteka

Superblok je zaglavlje sistema datoteka i sadrži informacije o sistemu datoteka u celini. Sadržaj regularnih datoteka čini određeni broj blokova podataka. Direktorijumi se posmatraju kao specijalne datoteke koje sadrže direktorijumske blokove. Direktorijumski blokovi su tabele sastavljene od određenog broja file-info struktura (FCB, File Control Block). File-info struktura sadrži ime objekta i pokazivač na indeksni čvor (i-node) za taj objekat.

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

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

Ove informacije dobijate kada zadate komandu koja prikazuje tzv. long-listing. Na primer:

```
nmacek2@lrs41:~$ ls -l /var
drwxr-xr-x  2 root root    4096 jan 12 17:00 backups
drwxr-xr-x 21 root root    4096 дец 19 13:26 cache
drwxrwsrwt  2 root whoopsie 4096 jan 12 16:47 crash
drwxr-xr-x 76 root root    4096 дец 22 19:37 lib
drwxrwsr-x  2 root staff   4096 аnp 11 2014 local
drwxrwxr-x 14 root syslog   4096 маp 14 12:34 log
drwxrwsr-x  2 root mail    4096 јyn 22 2014 mail
drwxrwsrwt  2 root whoopsie 4096 јyn 23 2014 metrics
```

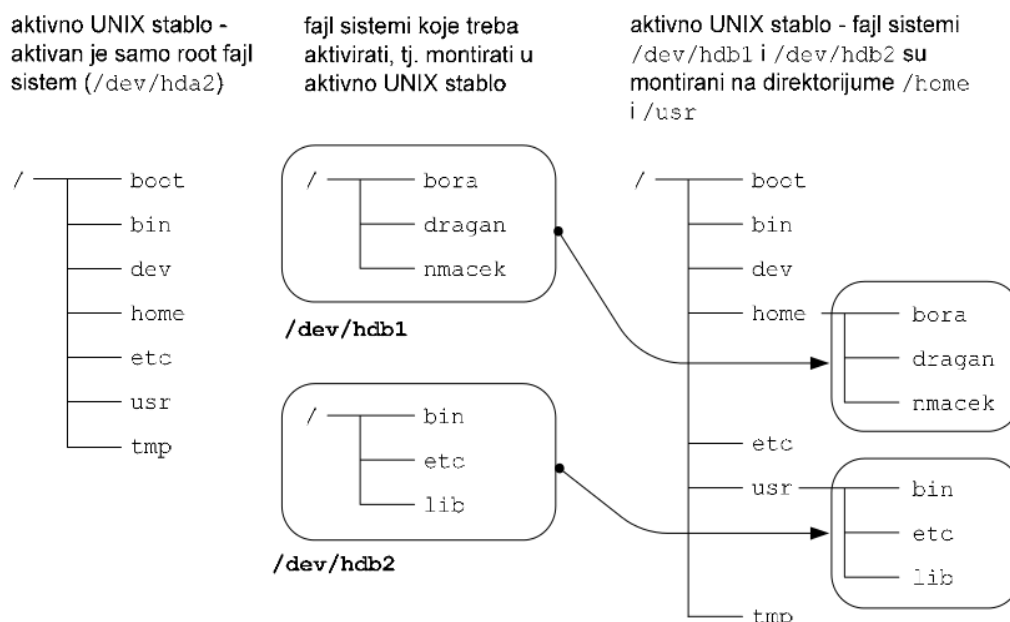
2. O aktivnom stablu i aktiviranju fajl sistema

Da bi sprečili moguće zbunjivanje. Komande C:, D:, E: i slične na operativnom sistemu Linux NE RADE! Vašim fajl sistemima ne pristupate preko logičkih diskova. Fajl sistemi se pod Linux operativnim sistemom aktiviraju montiranjem (engl. mounting) na aktivno stablo, tj. na mount-point direktorijume.

Početak stabla je koreni (root) direktorijum na koji se root fajl sistem koji sadrži osnovnu direktorijumsku strukturu, operativni sistem, instalirani softver, neke privremene datoteke i korisnički direktorijum root korisnika. U root fajl sistemu administrator kreira poddirektorijume na koje "montira" tj lepi druge fajl sisteme, uključujući i DVD medijume, USB fleš i hard diskove itd.

Da pojasnimo: kad uđete USB fleš u računar, grafičko radno okruženje će montirati fleš na neki direktorijum (npr /media/transcend). Ako radite u komandnoj liniji, onda morate ručno da kreirate neki direktorijum i montirate fleš na taj direktorijum.

O svemu ovome više priče će biti u operativnim sistemima 2, ovo je više informativnog karaktera. Pogledajte sliku 2 i biće vam jasno kako ovo radi.

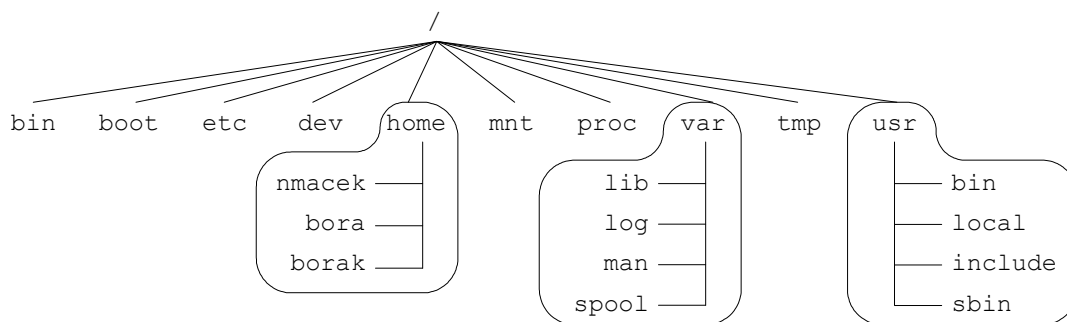


Slika 2. Princip aktiviranja na aktivno stablo

2.1. Pregled značajnijih direktorijuma u aktivnom stablu

Root fajl sistem sadrži osnovnu direktorijumsku strukturu stabla. Značajniji pod-direktorijumi u aktivnom stablu su:

- `/bin` - najčešće korišćene komande koje mogu koristiti regularni korisnici
- `/boot` - datoteke koje koristi boot loader (često se realizuje kao zasebni FS)
- `/dev` - specijalne datoteke koje predstavljaju uređaje (nodes), koje se kreiraju prilikom instalacije Linux sistema
- `/etc` - konfiguracione datoteke u tekstualnom obliku
- `/home` - lični direktorijumi korisnika, npr ako je username george, onda je home direktorijum `/home/george`
- `/mnt`, `/media` - direktorijumi u kome se nalaze mount-point direktorijumi za USB diskove i DVD uređaje (zavisno od distribucije, može biti i neki drugi, npr: `/var/run/nmacek/media`)
- `/tmp` - privremene datoteke
- `/root` - lični direktorijum korisnika root
- `/usr` - većina korisničkih komandi, prateće biblioteke i dokumentacija
- `/var` - spool direktorijumi, log datoteke, slične stvarčice



Slika 3. Značajniji direktorijumi u aktivnom stablu

3. 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 (`SOURCE`) u odgovarajuće odredište (`DESTINATION`). Argumenti `SOURCE` i `DESTINATION` moraju se navesti. Ukoliko korisnik želi da iskopira datoteku u tekući direktorijum, kao odredišni argument može navesti tačku.

U zavisnosti od argunemata postoje tri osnovna oblika korišćenja komande `cp`:

- kopiranje jedne datoteke
- kopiranje grupe datoteka iz istog direktorijuma
- kopiranje direktorijumskog stabla.

3.1. Kopiranje jedne datoteke

Opšti oblik komande:

```
cp source_file destination
```

- Argument `source_file` je ime izvorišne datoteke
- Argument `destination` je odredišni direktorijum ili ime odredišne datoteke.
- Argumenti mogu, po potrebi, sadržati apsolutne ili relativne putanje.

Primeri:

1. Kopiranje datoteke `a.a` u `b.b`

```
cp a.a b.b
```

2. Kopiranje datoteke `a.a` iz direktorijuma `/home` u `/tmp` sa promenom imena u `b.b`

```
cp /home/a.a /tmp/b.b
```

3. Kopiranje datoteke `a.a` iz tekućeg direktorijuma u direktorijum `/tmp`

```
cp a.a /tmp/a.a
```

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

Aktivnost 1: jednostavno kopiranje.

Pozicionirajte se na svoj lični direktorijum:

```
cd
```

Kreirajte komandom touch 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 komandom rm:

```
rm original kopija
```

Aktivnost 2: kopiranje datoteke iz drugog direktorijuma u tekući, sa promenom imena datoteke.

Iskopirajte jednu datoteku čiji je vlasnik korisnik root u svoj lični direktorijum:

```
cp /bin/cp mycp
```

Pogledajte šta piše u i-nodeovima originala i kopije (proverite kome pripada original, a kome kopija):

```
ls -l /bin/cp mycp
```

Obrišite datoteku mycp:

```
rm mycp
```

3.2. Kopiranje grupe datoteka iz istog direktorijuma

Opšti oblik komande:

```
cp source_files dest_dir
```

- Argument `source_files` je grupa izvorišnih datoteka koji se formira pomoću džoker karaktera. Podsetnik: džokeri mogu da zamene jedan karakter (?), više karaktera (*) ili opseg ([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:

1. Kopiranje svih datoteka koje počinju slovom "a" iz tekućeg direktorijuma u /tmp
`cp a* /tmp`
2. Kopiranje svih datoteka koje sadrže dva karaktera (prvi karakter je "a") iz tekućeg direktorijuma u /tmp
`cp a?.old /tmp`
3. Kopiranje svih datoteka iz direktorijuma /etc koji počinju karakterom "a", "b", "c" ili "d" u tekući direktorijum
`cp /etc/[a-d]* .`

Napomene:

- Kopira se grupa datoteka iz istog direktorijuma.
- Nije moguće kopirati grupu datoteka sa promenom imena kopija. Na primer:
`cp a* /tmp/b*`
- Originali i kopije moraju se nalaziti na različitim direktorijumima. Na primer:
- `cp a* .`

Aktivnost 3: upotreba džokera za kopiranje datoteka iz jednog direktorijuma u drugi.

1. Kreirajte dva direktorijuma u svom home direktorijumu:

```
mkdir src dst
```

2. Kreirajte nekoliko datoteka u direktorijumu src zadajući redom sledeće komande:

Predite 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
```

3. Iskopirajte u direktorijum dst sve datoteke koje počinju na slovo "a":

```
cp src/a* dst/
```

4. 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?

5. 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
```

Aktivnost 4: još malo priče o džokerima (da se podsetite).

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 komandom rm:

```
rm -rf src dst
```

3.3. Kopiranje direktorijumskog stabla

Opšti oblik komande:

```
cp -r source dest_dir
```

Primeri:

1. Kopiranje direktorijuma /etc sa svim poddirektorijumima i datotekama u direktorijum /tmp/oldconfig/etc

```
cp -r /etc /tmp/oldconfig
```

Obratite pažnju: datoteka /etc/passwd kopira se u /tmp/oldconfig/etc/passwd

2. Kopiranje kompletnog sadržaja direktorijuma /etc u direktorijum /tmp/oldconfig

```
cp -r /etc/* /tmp/oldconfig
```

Obratite pažnju: datoteka /etc/passwd kopira se u /tmp/oldconfig/passwd

3. Kopiranje svih datoteka i poddirektorijuma iz tekućeg direktorijuma, čije ime počinje karakterom "a" u direktorijum /tmp/mybackup.

```
cp -r a* /tmp/mybackup
```

Napomene:

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

Pitanje 1.

Pronađite uljeza (pogrešno zadatu komandu) i objasite u čemu je greška:

- a. `cp /etc/passwd`
- b. `cp /etc/passwd ~`
- c. `cp /etc/passwd /tmp`
- d. `cp /etc/passwd /tmp/copy_of_passwd`

4. 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`

Komanda `mv` pomera izvorišni objekat (`SOURCE`) u odgovarajuće odredište (`DESTINATION`). Ukoliko je izvorišni objekat datoteka komandom `mv` joj se može promeniti ime. Argumenti `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 i
- pomeranje grupe datoteka iz istog direktorijuma.

4.1. Promena imena i/ili pomeranje jedne datoteke

Opšti oblik komande:

```
mv source_file destination
```

- Argument `source_file` je ime izvorišne datoteke
- Argument `destination` može je odredišni direktorijum i/ili novo ime datoteke.

Primeri:

1. Promena imena datoteke

```
mv a.a b.b
```

2. Pomeranje datoteke iz jednog direktorijuma u drugi

```
mv /home/a.a /tmp/a.a
```

3. Promena imena datoteke i pomeranje u drugi direktorijum

```
mv /home/a.a /tmp/b.b
```

4.2. Pomeranje grupe datoteka iz istog direktorijuma

Opšti oblik komande:

```
mv source_files dest_dir
```

- Argument `source_files` je grupa izvorišnih datoteka koja se formira pomoću džoker karaktera (*, ?, [a-z], [A-Z], [0-9]).
- Argument `dest_dir` je odredišni objekat koji u ovom slučaju mora biti direktorijum (apsolutna ili relativna putanja).

Primeri:

1. Pomera sve fajlove koji počinju na "a" u /tmp

```
mv a* /tmp
```

2. Korišćenje džokera

```
mv a?[0-9].old /usr/tmp
```

3. Pomera fajlove koji počinju na "a", "b", "c" ili "d" u direktorijum ./backup

```
mv /etc/[a-d]* ./backup
```

Napomene:

- 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*`

Aktivnost 5: pomeranje jedne datoteke, grupe datoteka i direktorijuma

1. kreirajte dva direktorijuma u svom ličnom direktorijumu.

```
mkdir src dst
```

2. Kreirajte nekoliko datoteka u direktorijumu src zadajući redom sledeće komande:

Predite 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
```

2. Pomerite datoteku a1 u direktorijum dst:

```
mv src/a1 dst/
```

```
ls src
```

```
ls dst
```

3. Pomerite datoteku a2 u direktorijum dst i promenite joj ime:

```
mv src/a2 dst/a2_v1.01
```

```
ls src
```

```
ls dst
```

4. Promenite ime datoteke a3 u c6:

```
mv src/a3 src/c6
```

```
ls src
```

5. Pomerite grupu datoteka kojima ima počinje na "b" u direktorijum dst:

```
mv src/b* dst/
```

```
ls src
```

```
ls dst
```

6. Probajte da pomerite grupu datoteka i da im promenite ime:

```
mv src/c* dst/d*
```

```
ls -l dst
```

Da li ste uspeali u tome?

7. Probajte da pomerite ceo poddirektorijum:

```
mv src dst/  
ls dst
```

Da li ste uspeali u tome?

8. Obrišite poddirektorijume src i dst:

```
rm -rf src dst
```

Pitanje 2.

Pronađite uljeza (pogrešno zadatu komandu) i objasite u čemu je greška:

- a. `mv a* /tmp`
- b. `mv a?[0-9].old /tmp/a?[0-9].new`
- c. `mv /etc/[a-d]* ./backup`
- d. `mv /etc/?[a-d] ./backup/etc`

5. 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:

- brisanje jedne datoteke
- brisanje grupe datoteka iz istog direktorijuma
- rekurzivno brisanje direktorijumskog stabla

5.1. Brisanje jedne datoteke

Opšti oblik komande:

```
rm filename
```

- Argument `filename` je ime datoteke koja će biti obrisana, a opciono može sadržati i apsolutnu ili relativnu putanju datoteke.

Primeri:

1. Brisanje datoteke a.a iz tekućeg direktorijuma

```
rm a.a
```

2. Briše passwd.old iz direktorijuma /etc

```
rm /etc/passwd.old
```

5.2. Brisanje grupe datoteka iz istog direktorijuma

Opšti oblik komande:

```
rm group_of_files
```

- Argument group_of_files je grupa izvorišnih datoteka koja se formira pomoću džoker karaktera (*, ?, [a-z], [A-Z], [0-9]).

Primeri:

1. Brisanje svih datoteka čije ime počinje karakterom "a" iz tekućeg direktorijuma

```
rm a*
```

2. Brisanje svih datoteka čije ime se završava sa .conf.old iz direktorijuma /etc

```
rm /etc/*.conf.old
```

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

5.3. Rekurzivno brisanje direktorijumskog stabla

Opšti oblik komande:

```
rm -r directory
```

Primeri:

1. Brisanje direktorijum backup i kompletnog sadržaja

```
rm -r /etc/backup
```

2. Brisanje sadržaja direktorijuma backup ali ne i samog direktorijuma

```
rm -r /etc/backup/
```

Aktivnost 6: brisanje datoteka i poddirektorijuma

1. Pozicionirajte se na svoj lični direktorijum

`cd`

2. Kreirajte dve datoteke, direktorijum i u njemu još dve datoteke:

`touch a1 a2 b1 b2`

`mkdir dir1`

`touch dir1/c1`

`touch dir1/c2`

3. Probajte da obrišete dve datoteke sa fiegom -i:

`rm -i a*`

4. Probajte da obrišete dve datoteke sa flegom -f:

`rm -f b*`

5. Probajte da obrišete direktorijum (koji nije prazan) komandom `rmdir`:

`rmdir dir1`

Jeste li uspeli u tome?

`ls -l`

Ako niste, probajte ovako:

`rm -rf dir1`

Jeste li sada uspeli?

Pitanje 3. Objasnite čemu služe opcije -i i -f komande `rm`. Koristite man stranicu!

6. Rad sa direktorijumima

Direktorijumi se kreiraju komandom `mkdir` (make directory), čija je sintaksa:

`mkdir dir`

Ukoliko se kao parametar komandi navede -p, komanda `mkdir` će napraviti sve roditeljske direktorijume koji su neophodni.

Na primer, komanda:

`mkdir -p pera/kojot/super/genije`

radi isto što i sledeće komande:

`mkdir pera`

`mkdir pera/kojot`

`mkdir pera/kojot/super`

`mkdir pera/kojot/super/genije`

Direktorijum se može obrisati komandama `rm` i `rmdir` (remove directory).

- Komandom `rmdir` može se obrisati isključivo prazan direktorijum.
- Komanda `rm` je moćnija – za razliku od komande `rmdir`, rekurzivnim brisanjem komandom `rm` mogu se ukloniti cela direktorijumska stabla sa datotekama.

Primeri:

1. Brisanje direktorijuma `backup` (samo ako je prazan!)

```
rmdir /backup
```

2. Brisanje direktorijuma `backup` i kompletnog sadržaj

```
rm -r /backup
```

Aktivnost 7: kreiranje direktorijuma i brisanje direktorijuma

1. Pređite na lični direktorijum:

```
cd
```

2. Kreirajte jedan direktorijum:

```
mkdir newdirectory
```

3. Kreirajte stablo:

```
mkdir -p 1/11/111
```

4. Proverite šta je prethodna komanda napravila:

```
ls
```

```
cd 1
```

```
ls
```

```
cd 11
```

```
ls
```

```
cd
```

Obrišite direktorijum i stablo

```
rmdir newdirecory
```

```
rm -rf 1
```

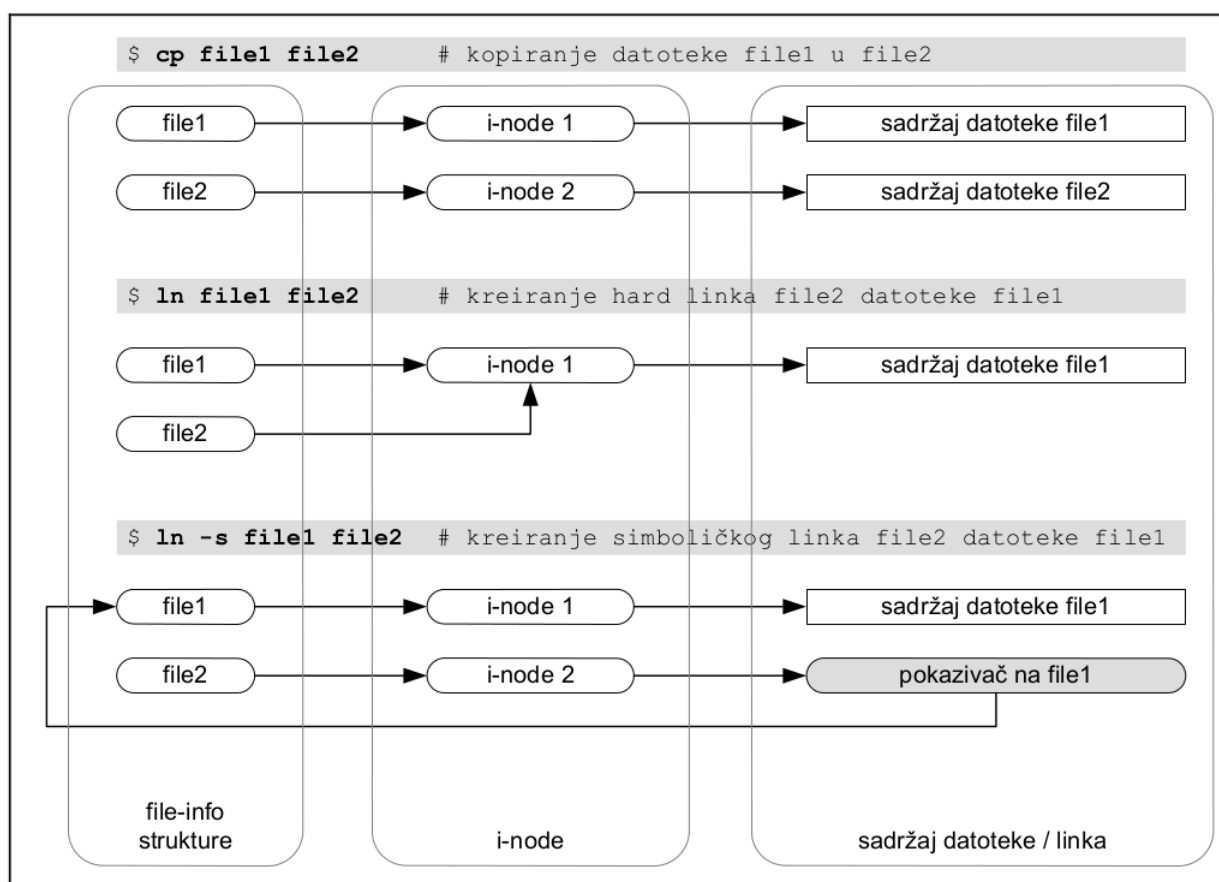
7. Linkovi

Datoteke se pod UNIX sistemom mogu povezati (linkovati). Svaka referenca na linkovanu datoteku (link) uvek se odnosi na originalnu datoteku (osim ukoliko se link navodi kao argument komandi `rm`). Razlozi za korišćenje linkova su:

- lakše pretraživanje i korišćenje datoteka (svakom korisniku je lakše da otkuca `less ph` nego `less /tmp/jsmith/adresses/business/phonebook.2004`, pod uslovom da u tekućem direktorijumu postoji link na odgovarajuću datoteku),
- ušteda prostora na disku (većina Linux distribucija koristi linkove na određene datoteke i direktorijume radi kompatibilnosti sa starijim varijantama UNIX sistema).

Na UNIX sistemima se mogu kreirati dve vrste linkova na datoteke:

- hard link, i
- simbolički link (symbolic link).



Slika 4. Simbolički i hard linkovi

7.1. Hard linkovi

Kada korisnik pozove datoteku po imenu (na primer: `cat tekst1`), UNIX prevodi simboličko ime datoteke koje je naveo korisnik u interno ime, koje koristi operativni sistem. Zbog posebne interne reprezentacije, korisnici mogu datotekama dodeliti veći broj imena. Hard link je jedno od tih imena, odnosno alternativno ime datoteke. Na primer, korisnik koji u svom home

direktorijumu ima datoteku file1 može kreirati hard link file2, odnosno referencu na tu datoteku .

```
ln file1 file2
ls file*
file1 file2
```

Hard linkovi se mogu kreirati na dva načina: pomoću komande ln i pomoću komande cp. Sintaksa ovih komandi je:

```
ln original linkname
cp -l original linkname
```

Hard linkovi su alternativna imena datoteka, i kao takve karakterišu ih sledeće osobine:

- link i original imaju isti i-node, tako da se moraju nalaziti na fizički istom sistemu datoteka (hard link se ne sme nalaziti na drugoj particiji ili na drugom disku);
- ne mogu se linkovati datoteke sa mrežnog sistema datoteka (NFS);
- ne može se linkovati direktorijum niti nepostojeća datoteka;
- vlasnik, grupa i prava pristupa su isti za link i za original;
- slobodan prostor na disku neznatno se umanjuje (jedna dir-info struktura više za alternativno ime datoteke);
- broj linkova originalne datoteke uvećava se za jedan nakon linkovanja;
- datoteka sa hard linkovima se ne može obrisati sa diska sve dok se ne uklone svi hard linkovi koji upućuju na tu datoteku.

Aktivnost 8: kreiranje hard linkova

1. Postavite umask na 027:

```
umask 027
```

2. Napravićete hard link datoteke /etc/passwd u svom home direktorijumu.

Pogledajte broj linkova originalne datoteke:

```
ls -l /etc/passwd
```

Napravite hard link:

```
cp -l /etc/passwd hard_passwd
```

3. Pogledajte i-nodove za obe datoteke:

```
ls -l /etc/passwd hard_passwd
```

Uporedite, vlasnika, grupu i prava pristupa hard linka i originalne datoteke.

Pogledajte broj linkova datoteke i linka.

4. Obrišite link:

```
rm hard_passwd
```

5. Pogledajte broj linkova originala nakon brisanja linka:

```
ls -l /etc/passwd
```

7.2. Simbolički linkovi

Simbolički link (symbolic link, symlink) je prečica ka objektu u sistemu datoteka. Princip korišćenja simboličkih linkova je identičan principu korišćenja hard linkova: svaka referenca na link odnosi se na originalnu datoteku, osim ukoliko se link poziva iz komande `rm`.

- Sa tačke gledišta korisnika, simbolički link pruža veću fleksibilnost jer dozvoljava linkovanje na direktorijume, nepostojeće datoteke i datoteke koje se nalaze na drugom sistemu datoteka (lokalnom ili mrežnom, domaćem ili stranom).
- Sa tačke gledišta Linux administratora, simbolički link je jednostavna referenca, odnosno zaseban objekat sistema datoteka koji koristi jedan i-node i jedan blok podataka u kom je zapisana lokacija originalnog objekta. Kao takav, simbolički link zauzima određen prostor na disku i jedno mesto u i-node tabeli.

Simbolički linkovi se mogu kreirati na dva načina: pomoću komande `ln` i pomoću komande `cp`.

Sintaksa ovih komandi je:

```
ln -s original linkname
```

```
cp -s original linkname
```

Argument `original` je ime originalne datoteke, a argument `linkname` ime simboličkog linka.

Simbolički linkovi su objekti sistema datoteka koji ukazuju na druge objekte (prečice).

Simboličke linkove karakterišu sledeće osobine:

- svaki simbolički link koristi poseban i-node i jedan blok podataka u sistemu datoteka,
- simbolički linkovi se mogu kreirati nalaziti na fizički istom ili različitom sistemu datoteka, odnosno na istoj ili drugoj particiji (disku),
- mogu se linkovati datoteke sa mrežnog sistema datoteka (NFS),
- može se linkovati direktorijum, kao i nepostojeća datoteka,
- u odnosu na `original`, link može imati različitog vlasnika, grupu i prava pristupa. Na korisnika koji datoteci ili direktorijumu pristupa putem simboličkog linka primenjuje se unija restrikcija (presek dozvola) linka i datoteke.
- slobodan prostor na disku se umanjuje (za jedan blok podataka). Takođe, simbolički link troši jedan i-node iz i-node tabele;

- broj linkova originalne datoteke se ne uvećava za jedan nakon linkovanja, već ostaje isti kao pre linkovanja;
- s obzirom da simbolički link može ukazivati na nepostojeći objekat, originalna datoteka se može obrisati sa diska bez obzira na broj simboličkih linkova koji upućuju na nju.

Aktivnost 9: kreiranje simboličkih linkova

1. Napravićete simbolički link na datoteku /etc/passwd u svom home direktorijumu.

Pogledajte broj linkova originalne datoteke:

```
ls -l /etc/passwd
```

Napravite simbolički link :

```
cp -s /etc/passwd sym_passwd
```

2. Pogledajte i-nodove za obe datoteke:

```
ls -l /etc/passwd sym_passwd
```

Obratite pažnju na deskriptor simboličkog linka `lrwxrwxrwx`. Da li je umask "ukinuo" neka prava? Uporedite, vlasnika, grupu i prava pristupa hard linka i originalne datoteke.

Pogledajte broj linkova datoteke i linka:

```
ls -l /etc/passwd sym_passwd
```

7.3. Upotreba opcije -d komande cp (no-dereference)

Prilikom kopiranja simboličkih linkova vrši se ukidanje reference (dereference), odnosno kopira se originalna datoteka na koju link ukazuje. Ukoliko postoji potreba da se iskopira link, a ne originalna datoteka, komanda `cp` se izvršava sa opcijom `-d`. U tom slučaju se ne vrši dereferenciranje. To je moguće izvesti i prilikom drugih operacija (kao što je kreiranje arhive), a opcije kojim se dereferenciranje isključuje zavise od konkretne komande.

Aktivnost 10: no-dereference

1. Probajte kopiranje sa `-d` flegom:

```
cp -d sym_passwd d_passwd
```

2. Probajte kopiranje bez `-d` flega:

```
cp sym_passwd no_d_passwd
```

3. Pogledajte kontekst za originalnu datoteku, simboličku link i dve kopije koje ste dobili i utvrdite šta se dešava kada kopirate simbolički link sa `-d` flegom, a šta se dobija kada kopirate link bez `-d` flega.

```
ls -l /etc/passwd sym_passwd d_passwd no_d_passwd
```

4. Obrišite datoteke i linkove koje ste napravili:

```
rm sym_passwd file_d file_no_d passwd
```