1. **Opcije u naredbama. Dugačka lista**

Opcije menjaju način izvršavanja naredbi. Počinju znakom "-", a sama opcija je obično samo jedan karakter. U slučaju da se koristi više opcija oni se mogu pisati i odvojeno i zajedno, a njihov redosled nije bitan.

Isto znače:

ls –al

ls –la

ls –a –l

ls –l –a

Izuzetak:

<naziv naredbe> --help (kratka pomoć: sintaksa naredbe)

man <naziv naredbe> (pomoć u dužem obliku: duži opis o naredbi)

***Dugačka lista***

1. **Zadatak**

Kreirati fajl pod imenom *dugacka*, zatim folder pod imenom *SUMA*! Izlistati sadržaj aktivnog direktorijuma u dugačkom formatu i analizirati dobijene informacije!

[student@drvo1 ~]$ **cat >dugacka**

Pravimo fajl da bi vezbali dugacku listu.

^D

[student@drvo1 ~]$ **mkdir SUMA**

[student@drvo1 ~]$ **ls -l**

total 8

drwxr-xr-x 2 student student 4096 Apr 26 22:59 SUMA

-rw-r--r-- 1 student student 4 Apr 26 23:00 dugacka

U prvom redu se nalazi broj memorijskih blokova koje zauzimaju svi fajlovi u ovom direktorijumu. Ostali redovi daju informacije o fajlovima.

• Prvi karakter je tip fajla (-, d, l)

• Sledećih 9 karaktera od po tri grupa predstavljaju dozvole

• Broj tvrdih veza (linkova)

• Vlasnik fajla

• Kojoj grupi pripada fajl

• Veličina fajla u bajtovima

• Datum i vreme kada je fajl poslednji put izmenjen (kod fajlova koji su napravljeni prethodnih godina tu se nalazi godina umesto vremena)

• Ime fajla

Ako se zadaju više opcija kod naredbe *ls* struktura liste se menja:

-a svi fajlovi su ispisani, i oni koji su skriveni (njihova imena počinju tačkom) (all)

-l dugačka lista (long)

-r obrnuti redosled (reverse)

-R uključuje sadržaj svih poddirektorijuma rekurzivno (recursive)

-i ispisuje broj za I-NODE

-x imena fajlova prikazuje u nekoliko kolona umesto jedne kolone

1. **Dozvole i prava**

Unix/Linux je sistem sa više korisnika, što znači i to da u sistemu može postojati jako mnogo fajlova koje istovremeno poseduju razni korisnici (možda njih više stotina). Istovremeno struktura fajlova Unix sistema jeste samo jedno stablo. Ako se sistem ne bi branio od toga, svaki korisnik bi mogao da pristupi svakom fajlu i za kratko vreme bi nastao potpuni haos. Treba se pobrinuti za to da samo oni koji imaju odgovarajuća prava dobiju i dozvolu da mogu pristupiti fajlovima.

Svaki fajl ima jednog vlasnika, to je izvorno onaj korisnik koji je napravio fajl. Sistem dozvola ima tri nivoa. Prvi nivo je **vlasnik**. Kada administrator sistema pušta novog korisnika u sistem, istovremeno ga i stavlja u neku grupu. Npr. u jednoj grupi mogu biti članovi jednog tima, svi koji slušaju jedan kurs, itd. Suština je to da mogu postojati takvi fajlovi kojima može pristupiti više korisnika. Sa tačke gledišta dozvola drugi nivo u pristupe fajla je **grupa**. Treći nivo sačinjavaju svi **ostali** korisnici koji rade u sistemu.

Postoje tri vrste dozvola: dozvola čitanja, pisanja i izvršavanja, znači za svaki fajl postoji 3x3 dozvole:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Znak nivoa | Nivo | Nivo na engleskom | Znak | Dozvola | Dozvola na engleskom |
| u | vlasnik | **u**ser | r | čitanje | **r**ead |
| g | grupa | **g**roup | w | pisanje | **w**rite |
| o | ostali | **o**thers | x | izvršavanje | e**x**ecute |
| a | svi | **a**ll | - | ova dozvola ne postoji |  |

Ponekad ove dozvole pišemo brojevima u oktalnom brojnom sistemu:

0 nema nikakve dozvole

1 samo za izvršavanje

2 samo za pisanje

3 pisanje i izvršavanje

4 samo za čitanje

5 čitanje i izvršavanje

6 čitanje i pisanje

7 čitanje, pisanje i izvršavanje (sve dozvole)

Lakše se sve ovo pamti ako se zna kako se dobijaju ovi brojevi:

r 400

w 200 vlasnik

x 100

r 40

w 20 grupa

x 10

r 4

w 2 ostali

x 1

Npr. rwxr--r-- 744 vlasnik sve može a grupa i ostali imaju dozvolu samo za čitanje.

Podrazumeva se šta su dozvole jednog običnog fajla, samo se to mora zapamtiti da je brisanje fajla takođe pisanje, kao što je to i svaka promena sadržaja.

Šta podrazumevaju razne dozvole u slučaju direktorijuma? Čitanje jednog kataloga znači da se njegov sadržaj može izlistati pomoću naredbe *ls*. Dozvola za pisanje znači da se može promeniti sadržaj direktorijuma. Promeniti sadržaj direktorijuma znači da se na njegovim fajlovima vrše operacije: npr. kreira se fajl, briše se, menja se njegovo ime. Dozvola za izvršavanje znači da se može pristupiti fajlovima u direktorijumu. Ova dozvola je najvažnija za direktorijume. Da bi bilo šta moglo da se uradi sa fajlom, mora se imati dozvola za izvršavanje za svaki direktorijum na putu do fajla. Takođe, kada ne postoji dozvola za izvršavanje za neki direktorijum, ne može da se pristupi ni celom stablu direktorijuma koji je ispod tog foldera.

1. **Zadatak**

Kreirati novi direktorijum pod imenom *CVECE*, istovremeno poddirektorijum pod imenom *RUZA*! Vlasnik fajla treba da ima sve dozvole a ostali ništa!

Opcije naredbe *mkdir*:

-p Kreira i potrebne roditeljske direktorijume (parent)

-m<dozvole> Novi direktorijum nastaje sa datim dozvolama (mode)

-v Pričljiv režim rada (verbose)

[student@drvo1 ~]$ **mkdir -pv -m700 CVECE/RUZA**

mkdir: created directory `CVECE'

mkdir: created directory `CVECE/RUZA'

[student@drvo1 ~]$ **ls -lR**

total 4

drwxr-xr-x 3 student student 4096 Apr 26 23:12 CVECE

./CVECE:

total 4

drwx------ 2 student student 4096 Apr 26 23:12 RUZA

./CVECE/RUZA:

total 0

Na osnovu izlistavanja direktorijuma se vidi da se folder *CVECE/RUZA*kreirao sa dozvolama *rwx* kod vlasnika. Dozvole grupe i ostalih su označene sa minusom tako da oni nemaju nikakve dozvole.

1. **Zadatak**

Ući u direktorijum /home i napraviti novi folder!

[student@drvo1 ~]$ **cd /home/**

[student@drvo1 home]$ **mkdir LJILJAN**

mkdir: cannot create directory `LJILJAN': Permission denied

Sistem ispisuje da ne postoje dozvole za izvršenje ove operacije.

[student@drvo1 home]$ **ls -l /**

...

drwxr-xr-x 7 root root 4096 Apr 18 16:14 home

Pogledati koje dozvole ima korisnik u direktorijumu /home/: pošto se korisnik nije prijavio kao korisnik root niti je u grupi root, tako da se na njega primenjuju dozvole ostalih: read i execute. Pošto nema dozvolu za pisanje u folderu /home/zato korisnik tamo ne moženapraviti novi direktorijum.

***Promena dozvola***

**chmod** <opcije> <dozvole> <parametri> (change mode)

Ova naredba vrši promene dozvola. Njegova najvažnija opcija je:

**–R** rekurzivno menja dozvole direktorijuma i fajlova u njima

Parametri mogu biti folderi ili obični fajlovi (može se nabrojati i više fajlova/direktorijuma).

**chown** <opcije> <novi vlasnik> <fajlovi> (change owner)

Fajlove predaje novom vlasniku. Ako se predaje pravo na svojinu nekog fajla, svojina tog fajla se više ne može vratiti, samo novi vlasnik može da ga vrati. Novi vlasnik se zadaje pomoću korisničkog imena. Sistem dozvola u najvećem broju sistema Unix/Linux je ograničen jer samo korisnik root ima pravo da menja vlasništvo fajlova.

**chgrp** <opcije> <grupa> <fajlovi> (change group)

Nabrojane fajlove predaje datoj grupi.

1. **Zadatak**

U početnom direktorijumu napraviti fajl pod imenom *danas* zatim promeniti grupu fajla na grupu *root*! Dati dozvole *read* i *write* grupi root tako da se ostale dozvole ne menjaju!

[student@drvo1 ~]$ **cat >danas**

Danas je oblacno vreme.

^D

[student@drvo1 ~]$ ls -l danas

-rw-r--r-- 1 student student 29 Apr 27 00:13 danas

[student@drvo1 ~]$ **chgrp root danas**

chown: changing ownership of `danas': Operation not permitted

U Fedori samo korisnik root može da menja vlasnike fajlova. Fajl može da se sta­vi samo u tu grupu gde pripada vlasnik fajla. Pošto u ovom slučaju korisnik student ni­je pripadnik grupe root, dobija se poruka *Operation not permitted* jer operacija nije dozvoljena.

[student@drvo1 ~]$ **chmod g+rw-x danas**

Dozvole se mogu zadati na dva načina: brojevima ili na gornji način. Ovaj primer pokazuje kako se mogu uključiti/iskjučiti dozvole pojedinim korisnicima (vlasnicima fajla), grupi ili ostalima.

**g+rw-x** znači da se daju (+) dozvole za čitanje i pisanje (rw - read, write) grupi (g - group) i oduzima (-) joj se dozvola za izvršavanje. Ako se želi npr. dati dozvola za pisanje ostalima treba ukucati sledeće: **o+w**.

1. **Zadatak**

Vlasniku fajla *danas* dati dozvole *write* i *execute*, grupi *read* i *write*, a ostalima samo dozvolu *execute*! Odrediti dozvole pomoću brojeva! Izlistati sadržaj fajla *danas*!

[student@drvo1 ~]$ **chmod 361 danas**

[student@drvo1 ~]$ **ls -l**

total 8

**--wx-w---x** 1 student student 29 Apr 27 00:13 danas

[student@drvo1 ~]$ **cat danas**

cat: danas: Permission denied

Pošto korisnik nema dozvolu za čitanje ne može se izlistati sadržaj fajla.

U ovom slučaju sve tri grupe dozvola su određene istovremeno. Prvi broj se od­nosi na vlasnika fajla, drugi na grupu a treći na ostale korisnike sistema. Konačan broj se dobija na način opisan na početku poglavlja: dozvole se podele u tri dela i zamenjuju se brojevima. Svako slovo ima vrednost i brojeve gde nema crtice treba sabrati:

4 2 1 | 4 2 1 | 4 2 1

- w x | r w - | - - x

0 + 2 + 1 | 4 + 2 + 0 | 0 + 0 + 1

= = =

3 6 1

1. **Zadatak**

Vlasniku direktorijuma *CVECE* dati dozvole za pisanje i čitanje a svima ostalima nikakve dozvole! Izlistati sadržaj direktorijuma *CVECE* zatim ući u taj folder! Nakon toga ući u direktorijum *CVECE/RUZA*!

[student@drvo1 ~]$ **chmod 600 CVECE/**

[student@drvo1 ~]$ **ls -l**

total 4

d**rw**------- 3 student student 4096 Apr 26 23:57 CVECE

[student@drvo1 ~]$ **ls -l CVECE/**

total 0

?--------- ? ? ? ? ? CVECE/RUZA

[student@drvo1 ~]$ **cd CVECE/**

-bash: cd: CVECE/: Permission denied

Postoji dozvola za čitanje direktorijuma ali listing nije potpun: ispisuje samo imena fajlova. Ne postoji dozvola da se uđe u direktorijum (tj. izvršavanje - execute), dobija se poruka Permission denied.

[student@drvo1 ~]$ **cd CVECE/RUZA**

-bash: cd: CVECE/RUZA: Permission denied

Ne sme se ući ni u direktorijum CVECE/RUZA iako za folder RUZA postoje sve dozvole. To se dešava zbog toga što negde na putu do tog direktorijuma ne postoje dozvole za izvršavanje.

1. **Zadatak**

Obrisati sadržaj početnog direktorijuma!

Direktorijum se briše pomoću naredbe **rmdir**:

**rmdir** <opcije> <direktorijumi>

**-p** briše i roditeljske direktorijume (parent)

[student@drvo1 home]$ **rmdir –p CVECE/RUZA**

Pomoću ove naredbe se mogu obrisati direktorijumi CVECE i CVECE/RUZA ali samo u slučaju da su oni prazni. Naredba *rm* briše fajlove ali ako zadajemo određene opcije može da izbriše i celo podstablo direktorijuma zajedno sa svim fajlovima (treba da se obazrivo koristi).

**rm** <opcije> <fajlovi>

**-r** ili **–R** rekurzivno briše celo podstablo direktorijuma

**-f** da se izbegne pitanje za svaki fajl "da li sigurno želiš brisati fajl" (force)

[student@drvo1 home]$ **rm –rf** **/home/student/\***

Ova naredba najefikasnije rešava postavljeni zadatak. Početni direktorijum je zadat pomoću apsolutne putanje tako da je isključena mogućnost greške.

[student@drvo1 ~]$ **rm -rf /home/student/\***

rm: cannot chdir from `.' to `/home/student/CVECE': Permission denied

Ipak postoji problem jer ne postoji dozvola za izvršavanje za direktorijum CVECE, tako se on ne može ni obrisati. Najelegantnije rešenje je da se svakom fajlu, direktorijumu i poddirektorijumu vlasniku zadaju sva prava:

[student@drvo1 ~]$ **chmod -R u+rwx /home/student/\***

Opcija *-R* naredbe *chmod* služi za rekurzivno menjanje dozvola u svim poddirektorijumima (recursive). Sada se može izdati naredba:

[student@drvo1 ~]$ **rm -rf /home/student/\***

[student@drvo1 ~]$ **ls -l**

total 0