Linux

3

Kontrola pristupa na nivou sistema datoteka

Sadržaj

doc. dr Nemanja Maček Beograd, 2018

1. Vlasništvo i prava pristupa

Kontrola pristupa na nivou sistema datoteka obuhvata:

- vlasničke odnose (pripadnost objekata korisnicima i grupama) i
- prava pristupa (dodeljuju se svakoj datoteci i direktorijumu i određuju šta korisnici mogu da rade)

Komanda `ls -l` ("long listing") prikazuje sve informacije o objektu koje (osim imena) čita iz i-nodova objekata. Na osnovu toga možete da odredite šta ko može da uradi sa nekom datotekom ili direktorijumom.

Podsetnik:

`ls -la`	prikazuje sve datoteke, uljučujući i skrivene
`ls -ld`	prikazuje informacije iz i-nodea direktorijuma a ne njegov sadržaj

Primer:

```
ls -la
-rwxr-xr-x      1    root       root       93     2012-02-27 22:52   cssh.sh
drwxr-xr-x      2    korisnik   korisnik   4096   2012-02-27 22:53   .data
----------      -    -------    --------   -----  ----------------   --------
Tip + AR        HL   OWNER      GROUP      SIZE   DATE and TIME      NAME
PRAVA PRISTUPA       VLASNIČKI ODNOSI      OSTALI PODACI O DATOTECI
```

1.1. O vlasniku, grupi i ostatku sveta

Polja **OWNER** i **GROUP** određuju vlasnika i grupu datoteke.

- Vlasnik (OWNER) je korisnik koji je kreirao objekat, odnosno korisnik kome je root dodelio vlasništvo.
- Grupa (GROUP) je primarna grupa korisnika koji je objekat kreirao ili grupa kojoj je root dodelio vlasništvo. Primarna grupa korisnika je grupa koju MORATE da navedete kada kreirate korisnika. Navodi se u datoteci `/etc/group`.
- Svi ostali korisnici koji nisu ni vlasnik objekta, niti pripadaju grupi kojoj objekat pripada spadaju u kategoriju ostali, tj ostatak sveta (others).

Prava pristupa nekog korisnika određuju se prema vlasničkoj kategoriji kojoj korisnik pripada.
Dakle, korisnik može biti ili:

- vlasnik datoteke
- član grupe kojoj je dodeljena datoteka
- ostatak sveta u odnosu na datoteku.

Rezultujuće pravo vlasnika koji pripada grupi kojoj je objekat dodeljen određuje se:

- kao unija prava vlasnika i grupe (kod starijih UNIXa)
- kao pravo vlasnika (kod POSIX-compliant UNIXa, kao što je Linux)

1.2. Prava pristupa

Polje **Tip + AR** označava prava pristupa za tri vlasničke kategorije:

- karakter 1:
    - “d” direktorijum
    - “-” datoteka
    - “l” simbolički link
    - “b” blok uređaj
- karakteri 2-10: prava pristupa objektu za vlasnika, grupu i ostatak sveta.

Potpuni skup prava za svaku vlasničku kategoriju čine tri prava:

- pravo čitanja (“r” – Read)
- pravo upisa (“w” – Write)
- pravo izvršavanja (“x” - eXecute)

Uvek je prva pozicija read, druga write i treća execute - “rwx”.
Ukoliko se na nekoj poziciji umesto slova r, w ili x nalazi crtica (“-”), pravo je ukinuto.

Šta pristupna prava omogućuju da uradite sa datotekom?

- “r“ – čitanje sadržaja datoteke (prikazivanje na ekranu, štampanje, kopiranje)
- “w” – izmena sadržaja datoteke (ne znači da korisnik može da obriše datoteku !)
- “x” – izvršavanje datoteke (ako je shell program ili binarna izvršna datoteka)

Šta pristupna prava omogućuju da uradite sa direktorijumom?

- “r” – čitanje sadržaja direktorijuma, tj. file-info struktura (korisnik može da izvrši

komandu `ls`)

- "w" – izmena sadržaja direktorijuma (dodavanje novih i brisanje postojećih objekata u njemu)
- "x" – pozicioniranje na dir. (`cd`), puni listing sadržaja (`ls -l`) i pretraživanje direktorijuma (`find`)

NAPOMENA 1: da bi direktorijum na Linuxu bio praktično upotrebljiv korisnicima treba dati prava "r" i "x".
NAPOMENA 2: pravo "w" dato nad datotekom ne znači da možete da je obrišete. Da bi ste obrisali datoteku, potrebno je pravo "w" nad direktorijumom u kome se nalazi.

1.3. Primer

Na primer, korisnik zadaje sledeću komandu:

```
ls -l /home/pepe/lemeow
```

i kao rezultat dobija:

```
-rwxr-x---   1   pepe   mouffette   509   Mar 10   17:21   lemeow
```

Karakteri na pozicijama od 1 do 10 (ch 1 - ch 10) imaju značenje:

- ch 1 "-"
  datoteka je obična, tj. regularna
- ch 2,3,4 "rwx"
  vlasnik datoteke (korisnik pepe) može čitati, modifikovati i izvršavati datoteku
- ch 5,6,7 "r-x"
  korisnici koji pripadaju grupi mouffette mogu čitati i izvršavati datoteku
- ch 8,9,10 "---"
  ostatak sveta ne može ništa da radi sa datotekom

Aktivnost 1.

Pokrenite terminal. Prikažite kontekst datoteke i odredite vlasničke odnose za

- datoteku `/etc/passwd` i
- direktorijum `/etc`.

Zadajte sledeće komande:

```
ls -l /etc/passwd
ls -ld /home/korisnik
```

Za prethodno pomenute objekte odrediti:

1. ko je vlasnik?
2. kojoj grupi su objekti dodeljeni?
3. koja su prava pristupa vlasnika, grupe i ostatka sveta?
4. šta vlasnik, grupa i ostatak sveta mogu da rade sa tim objektima?

1.4. Zašto je “rwxr-x-rx” isto što i 755?

Ako je

- “r” = 4
- “w” = 2
- “x”=1

onda je:

```
7  =  4 + 2 + 1  =  rwx          3  =  0 + 2 + 1  =  -wx
6  =  4 + 2 + 0  =  rw-          2  =  0 + 2 + 0  =  -w-
5  =  4 + 0 + 1  =  r-x          1  =  0 + 0 + 1  =  --x
4  =  4 + 0 + 0  =  r--          0  =  0 + 0 + 0  =  ---
```

Primeri:

644 = `rw-r--r--` (vlasnik ima pravo upisa i čitanja, grupa i ostali pravo čitanja)

750 = `rwxr-x---` (vlasnik ima sva prava, grupa prava čitanja i izvršavanja, ostali ništa)

Aktivnost 2: Uticaj prava na neke operacije.

Pristupna prava ne dotiču root korisnike. Zato se ovaj zadatak radi bez korišćenja sudo, pod običnim, neprivilegovanim nalogom.

Proverite i **OBJASNITE** zašto sledeće komande možete, odnosno ne možete da izvršite:

1. prelazak na `/root` direktorijum

```
cd /root
```

savet – odredite prava za svoju vlasničku kategoriju:

```
ls -ld /root
```

2. pregledanje sadržaja datoteke `/etc/shadow`:

```
cat /etc/shadow
```

savet – odredite prava za svoju vlasničku kategoriju:

```
ls -l /etc/shadow
```

3. pregledanje sadržaja datoteke `/etc/passwd`:

```
cat /etc/passwd
```

savet – odredite prava za svoju vlasničku kategoriju:

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

4. upis u direktorijum `/bin`:

```
touch /bin/myfile
```

savet – odredite prava za svoju vlasničku kategoriju:

```
ls -ld /bin
```

5. brisanje datoteke iz `/bin` direktorijuma:

```
rm /bin/cp
```

savet – odredite prava za svoju vlasničku kategoriju:

```
ls -ld /bin
```

2. Podrazumevana prava pristupa

Linux pri kreiranju dodeljuje objektu vlasnika i grupu i postavlja podrazumevana prava pristupa:

- Korisnik koji kreira objekat postaje njegov vlasnik
- Objekat se dodeljuje primarnoj grupi tog korisnika.
- Podrazumevana prava pristupa zavise od promenljive umask.

2.1. Promenljiva umask

Inicijalna prava pristupa dodeljuju se na osnovu vrednosti promenljive umask.
Promenljiva umask je specifična za svakog korisnika i postavlja se prilikom prijavljivanja na sistem (uobičajena vrednost je 022).
Trenutna vrednost može se videti i promeniti pomoću istoimene komande.

Na primer:

```
umask           # ispisuje trenutnu vrednost promenljive umask
0022            # trenutna vrednost promenljive umask
umask 027       # menja vrednost umask u 027
umask           # ispisuje trenutnu vrednost promenljive umask
0027            # trenutna vrednost promenljive umask
```

Prva nula u vrednosti promenljive umask označava da je ona prikazana u oktalnoj formi.
Probajte da ukucate komandu “umask -S” i protumačite rezultat izvršenja komande.

Namena ove promenljive je da ukida prava pristupa novokreiranim direktorijumima i datotekama.
Na primer:

- Ako je umask 022, ukida se pravo upisa za grupu i ostatak sveta.
- Ako je umask 027, ukida se pravo upisa za grupu i sva prava za ostatak sveta.

Inicijalna prava pristupa pri kreiranju se razlikuju za direktorijume i datoteke:

- za direktorijum se vrednost umask oduzima od 777 (`rwxrwxrwx`)
- za datoteku se vrednost umask oduzima od 666 (`rw-rw-rw-`), zato što se podrazumevano NE DODELJUJE pravo izvršavanja!

Primer: vrednost promenljive umask je 022. Podrazumevana prava za datoteku su:

	`rw-rw-rw-`	666	(podrazumevano pravo)
oduzmi:	`----w--w-`	022	(umask)
dobija se:	`rw-r-----`	644	(rezultat)

U ovom slučaju je tačno 666-022=644, ali to nije ispravan način razmišljanja! ZAŠTO?

Ako je umask 027, podrazumevana prava za datoteku NISU 666-027=639! Vrednost 9 ne postoji kao skup prava! Drugim rečima:

	`rw-rw-rw-`	666	(podrazumevano pravo)
oduzmi:	`----w-rwx`	027	(umask)
dobija se:	`rw-r-----`	640	(rezultat)

Dakle, ako je umask 027, rezultujuća prava za novokreiranu datoteku su 640!

Primer:

```
umask 027
mkdir mydir
touch myfile
ls -la
drwxr-x---   2   nm   nm   4096   dec 23   14:33   mydir
-rw-r-----   1   nm   nm      0   dec 23   14:33   myfile
```

Aktivnost 3: Podrazumevana pristupna prava i vlasništvo novih datoteka i direktorijuma.

1. Proverite vrednost umask promenljive
   ```
   umask
   ```
2. Pređite na home direktorijum
   ```
   cd
   ```
3. Kreirajte novu datoteku
   ```
   touch myfile
   ```
4. Pokušajte sami da odredite, a zatim proverite pristupna prava nove datoteke komandom
   ```
   ls -l myfile
   ```
5. Obrišite datoteku komandom
   ```
   rm myfile
   ```
6. Kreirajte novi direktorijum
   ```
   mkdir mydir
   ```
7. Pokušajte sami da odredite, pa onda proverite pristupna prava novog direktorijuma
   ```
   ls -ld mydir
   ```
8. Uklonite direktorijum komandom
   ```
   rmdir mydir
   ```

Aktivnost 4: Promena vrednosti promenljive umask i uticaj na kreiranje objekata.

1. Promenite vrednost umask promenljive
   ```
   umask 027
   ```
2. Pređite na home direktorijum
   ```
   cd
   ```
3. Kreirajte novu datoteku
   ```
   touch myfile
   ```
4. Probajte da odredite sami, pa onda proverite pristupna prava nove datoteke
   ```
   ls -l myfile
   ```
5. Obrišite datoteku
   ```
   rm myfile
   ```
6. Kreirajte novi direktorijum
   ```
   mkdir mydir
   ```
7. Probajte da odredite sami, pa proverite prava novog direktorijuma
   ```
   ls -ld mydir
   ```

8. Uklonite direktorijum

```
rmdir mydir
```

PONOVITE AKTIVNOST ZA UMASK VREDNOST 077!

3. Promena vlasništva i pristupnih prava

Vlasnik, grupa i prava pristupa se dodeljuju svakom objektu prilikom kreiranja, a kasnije se mogu promeniti.

- samo VLASNIK objekta ili ROOT mogu promeniti prava pristupa
- samo ROOT može promeniti vlasnika (sprečava “maltretiranje” korisnika prepunjenjem kvote)

3.1. Promena pristupnih prava (simbolički režim)

Opšti oblik komande:

```
chmod [-R] categories operator permissions [...] filename
```

Argumenti i opcije imaju sledeće značenje:

- categories: vlasnik “**u**”, grupa “**g**”, others “**o**”, sve vlasničke kategorije “**a**”
- operator: dodela prava “**+**”, ukidanje prava “**–**“, definisanje skupa prava “**=**”
- permissions: čitanje “**r**”, modifikacija “**w**”, izvršavanje “**x**”
- Parametar **-R** inicira promenu pristupnih prava direktorijuma i svih njegovih objekata (poddirektorijuma i datoteka)

Vlasnik može dodeliti ili ukinuti prava koja želi, bez poznavanja trenutnih prava.
Ovaj režim je pogodan za dodelu ili oduzimanje prava većem broju datoteka sa različitim trenutnim skupom prava.

Primer:
Datoteka myfile trenutno ima postavljena prava 666 (`rw-rw-rw-`).

```
ls -l myfile
-rw-rw-rw-   1   nm   nm   0   dec 23   15:25   myfile
```

1. Sledeća komanda postavlja `r-x` vlasniku i oduzima `w` grupi i ostalima:

```
chmod u=rx,go-w myfile
```

```
ls -l myfile
-r-xr--r--   1   nm   nm   0   dec 23   15:25   myfile
```

2. Sledeća komanda dodaje pravo upisa vlasniku:

```
chmod u+w myfile
ls -l myfile
-rwxr--r--   1   nm   nm   0   dec 23   15:25   myfile
```

Aktivnost 5: promena pristupnih prava (simbolički režim)

1. Pozicionirajte se na svoj lični direktorijum.

```
cd
```

2. Postavite vrednost promenljive umask tako da vlasnik ima prava “rw-” a grupa i ostali “r--” za sve datoteke koje korisnik kreira.

```
umask 022
```

3. Kreirajte datoteku

```
touch betatest
```

4. Proverite prava pristupa

```
ls -l betatest
```

5. Koristeći chmod u simboličkom režimu

– dodajte grupi i ostalima pravo upisa:

```
chmod go+w betatest
ls -l betatest
```

– oduzmite kategoriji others sva prava:

```
chmod o-rwx betatest
ls -l betatest
```

– dodajte kategorijama owner i group pravo izvršavanja:

```
chmod ug+x betatest
ls -l betatest
```

– dajte svima sva prava:

```
chmod a+rwx betatest
ls -l betatest
```

6. Obrišite datoteku

```
rm betatest
```

3.2. Promena pristupnih prava (numerički režim)

Opšti oblik komande:

```
chmod [-R] mode filename
```

Parametri su sledeći:

- **mode** su nova pristupna prava, navode se kao tri cifre (0-7) koje predstavljaju redom pristupna prava za vlasnika, grupu i ostatak sveta.
- **filename** je ime objekta kome se menjaju prava
- parametar **-R** inicira promenu pristupnih prava direktorijuma i svih njegovih objekata

Primer:

Datoteka myfile trenutno ima postavljena prava 666 (`rw-rw-rw-`).

```
ls -l myfile
-rw-rw-rw-   1   nm   nm   0   dec 23   15:25   myfile
```

1. Dodela prava “rwx” korisniku i “r-x” grupi i ostatku sveta.

```
chmod 755 myfile
ls -l myfile
-rwxr-xr-x   1   nm   nm   0   dec 23   15:25   myfile
```

Aktivnost 6: promena pristupnih prava (numerički režim)

1. Pređite na home direktorijum:

```
cd
```

2. Kreirajte na svom home direktorijumu novu datoteku:

```
touch betatest
```

3. Proverite prava pristupa:

```
ls -l betatest
```

4. Postavite svima pravo čitanja i izvršavanja:

```
chmod 555 betatest
ls -l betatest
```

5. Dajte svima sva prava u odnosu na datoteku `betatest`:

```
chmod 777 betatest
ls -l betatest
```

6. Dajte vlasniku sva prava, grupi prava “r” i “w”, a ostalima ništa:

```
chmod 760 betatest
ls -l betatest
```

7. Obrišite datoteku:

```
rm betatest
```

3.3. Promena vlasnika i grupe

Opšti oblici komandi za promenu vlasništva i grupe:

```
chown [-R] user filename          (promena vlasnika)
chgrp [-R] group filename         (promena grupe)
```

Parametri su sledeći:

- **user** i **group** su novi vlasnik i nova grupa
- **filename** je ime objekta kome se menja vlasništvo
- Parametar **-R** inicira promenu vlasništva direktorijuma i svih njegovih objekata

NAPOMENA: U Ubuntu Linuxu, chown i chgrp mogu da koriste samo sudo-eri i root korisnik (ukoliko mu je dodeljena lozinka). Drugim rečima, unosi se prvo sudo pa ime komande.

Primer:
Vlasnik datoteke myfile je korisnik “nm”, a datoteka je dodeljena grupi “staff”, kao što se vidi iz listinga komande:

```
ls -l myfile
-rw-r--r--   1   nm       staff   0   Apr 28   12:07   myfile
```

1. Sledeća komanda dodeljuje datoteku korisniku “jsmith”. NAPOMENA: sudo zahteva da unesete svoju lozinku!

```
sudo chown jsmith myfile
```

Provera:

```
ls -l myfile
-rw-r--r--   1   jsmith   staff   0   Apr 28   12:07   myfile
```

2. Sledeća komanda dodeljuje datoteku grupi “users”.

```
sudo chgrp users myfile
```

Provera:

```
ls -l myfile
-rw-r--r--   1   jsmith   users   0   Apr 28   12:07   myfile
```

Aktivnost 7: promena vlasnika

1. Pređite na home direktorijum:

```
cd
```

2. Kreirajte novu datoteku:

```
touch myfile
```

3. Kreirajte korisnika “korisnik2” iz grafičkog okruženja ili jednostavno zadajte sledeću komandu:

```
sudo adduser korisnik2
```

Napomena: sudo traži da unesete svoju lozinku!

4. Probajte da poklonite datoteku drugom korisniku:

```
chown korisnik2 myfile
```

5. Da li ste uspeli to da uradite? Proverite komandom

```
ls -l myfile
```

6. Probajte da kao administrator, upotrebom mehanizma sudo, dodelite tu datoteku drugom korisniku:

```
sudo chown korisnik2 myfile
```

7. Da li ste sada uspeli to da uradite? Proverite komandom

```
ls -l myfile
```

8. Obrišite datoteku:

```
sudo rm myfile
```

4. Uticaj kontrole pristupa na rad sa datotekama

Vlasništvo i pristupna prava određuju da li ćete uspeti da izvršite komandu za kopiranje, pomeranje i brisanje datoteka i direktorijama, da izvršite datoteku, kreirate direktorijum I izlistate njegov sadržaja. Za to kontrola pristupa i služi!

4.1. Kopiranje datoteka

Da bi korisnik mogao da iskopira datoteku <file> iz direktorijuma <dir1> u direktorijum <dir2> potrebno je da ima sledeća dva prava:

- pravo r nad datotekom <file> (čime se omogućava čitanje sadržaja originalne datoteke).
- pravo w nad direktorijumom <dir2> (čime se omogućava izmena sadržaja odredišnog direktorijuma, odnosno kreiranje nove datoteke).

Ovaj skup prava je minimalan - neke varijante UNIX sistema zahtevaju dodatna prava. Dovoljni uslovi za kopiranje fajla na svim UNIX sistemima uključuju potrebne uslove i pravo x nad direktorijumima <dir1> i <dir2>.

Šta se dešava sa kopijama?

- vlasnik kopije je korisnik koji je pokrenuo komandu cp,
- datoteka se dodeljuje primarnoj grupi korisnika koji je pokrenuo komandu cp (grupa koja je navedena u datoteci `/etc/passwd` za datog korisnika),
- pristupna prava kopije su najčešće sužena u odnosu na pristupna prava originala, a dobijaju se logičkim množenjem bitova pristupnih prava originala i vrednosti promenljive umask. Na primer: ako su pristupna prava originalne datoteke 666, a promenljiva umask 002, pristupna prava kopije biće 664.

4.2.Pomeranje datoteka

Da bi korisnik mogao da pomeri datoteku <file> iz direktorijuma <dir1> u direktorijum <dir2>, potrebno je da ima sledeća prava:

- pravo r nad datotekom <file> (čime se omogućava čitanje sadržaja originalne datoteke),
- pravo w nad direktorijumom <dir1> (čime se omogućava izmena sadržaja izvorišnog direktorijuma, odnosno brisanje datoteke),
- pravo w nad direktorijumom <dir2> (čime se omogućava izmena sadržaja odredišnog direktorijuma, odnosno kreiranje nove datoteke).

Dovoljni uslovi za pomeranje datoteka na svim UNIX sistemima uključuju potrebne uslove i pravo x nad direktorijumima <dir1> i <dir2>.

4.3. Promena imena datoteke

Da bi korisnik mogao da promeni ime datoteke <file> koja se nalazi u direktorijumu <dir1>, potrebno je da ima:

- pravo w nad direktorijumom <dir1> (čime se omogućava izmena sadržaja izvorišnog direktorijuma, odnosno promena imena datoteke).

Dovoljni uslovi za promenu imena datoteka na svim UNIX sistemima osim ovog prava uključuju i pravo x nad direktorijumom <dir1>.

4.4. Brisanje datoteke

Da bi korisnik mogao da obriše datoteku <file> iz direktorijuma <dir1>, potrebno je da ima:

- pravo w nad direktorijumom <dir1> (čime se omogućava izmena sadržaja direktorijuma, odnosno brisanje datoteke).

Dovoljni uslovi za brisanje datoteka na svim UNIX sistemima osim ovog prava uključuju i pravo x nad direktorijumom <dir1>.

Aktivnost 8: pristupna prava originala i kopije – uticaj promenljive umask

1. Kreirajte jednu datoteku u svom home direktorijumu

```
touch original
```

2. Postavite prava pristupa za datoteku “original” na 775

```
chmod 775 original
```

3. Postavite vrednost promenljive umask tako da odseca prava write i execute kategorijama group i others sledećom komandom

```
umask 033
```

4. Iskopirajte datoteku

```
cp original kopija
```

5. Pogledajte šta piše u i-nodeovima originala i kopije

```
ls -l original kopija
```

Uporedite pristupna prava originala i kopije. Koji su bitovi ukinuti ?

6. Postavite vrednost promenljive umask na vrednost 022

```
umask 022
```

7. Obrišite obe datoteke:

```
rm original kopija
```

Aktivnost 9: vlasnički odnosi originala i kopije

1. Pređite na svoj home direktorijum

```
cd
```

Proverite vlasničke odnose datoteke `/etc/hosts`

```
ls -l /etc/hosts
```

Kreirajte kopiju datoteku u svom home direktorijumu

```
cp /etc/hosts myhosts
```

Proverite vlasničke odnose originala i kopije

```
ls - l /etc/hosts myhosts
```

Kojoj grupi je dodeljena datoteka myhosts?

Koja je primarna grupa korisnika “korisnik” koji je inicirao kopiranje? U listingu pronađite korisnika I proverite koja je grupa. Viewer “less” napuštate tasterom “q”:

```
less /etc/hosts
```

Obrišite datoteku myhosts

```
rm hosts
```

Aktivnost 10: potrebni i dovoljni uslovi za kopiranje datoteke

Odredite potrebne i dovoljne uslove da nešto iskopirate.

Kreirajte dva direktorijuma u svom home direktorijumu:

```
mkdir dir1 dir2
```

Kreirajte datoteku u direktorijumu dir1:

```
cp /etc/passwd dir1/file1
```

1. Ako su sva prava data, kopiranje je izvodljivo:

- Dodelite svima sva prava nad direktorijumima:

```
chmod 777 dir1 dir2
```

- Dodelite svima sva prava nad datotekom:

```
chmod 777 dir1/file1
```

- Probajte da iskopirate datoteku

```
cp dir1/file1 dir2/file2
```

Ovog puta prolazi sigurno, jer su vam data sva prava.

- Obrišite kopiju:

```
rm dir2/file2
```

2. Ukidajte redom prava nad datotekom dir1/file1 i utvrdite koja su vam prava neophodna nad izvorišnom datotekom da bi ste mogli da izvršite kopiranje:

2.1. pravo čitanja “r” izvorišne datoteke:

- ukinite pravo “r”:

```
chmod u-r dir1/file1
```

- probajte da iskopirate:

```
cp dir1/file1 dir2/file2
```

- da li je kopiranje uspešno? proverite komandom:

```
ls dir2
```

- ukoliko je kopiranje uspešno, obrišite kopiju

```
rm dir2/file2
```

- ako nije uspešno, znači da je to pravo (read) bilo potrebno za kopiranje.
- dodelite ponovo sebi pravo “r”

```
chmod u+r dir1/file1
```

2.2. Na sličan način, proverite da li je za kopiranje potrebno pravo upisa “w” u izvorišnu datoteku.

- ukinite pravo “w”:

```
chmod u-w dir1/file1
```

- probajte da iskopirate:

```
cp dir1/file1 dir2/file2
```

- da li je kopiranje uspešno? proverite komandom:

```
ls dir2
```

- ukoliko je kopiranje uspešno, obrišite kopiju

```
rm dir2/file2
```

- ako nije uspešno, znači da je to pravo (write) bilo potrebno za kopiranje.
- dodelite ponovo sebi pravo “w”

```
chmod u+w dir1/file1
```

2.3. Na sličan način, proverite da li je za kopiranje potrebno pravo izvršavanja “x” izvorišne datoteke.

- ukinite pravo “x”:

```
chmod u-x dir1/file1
```

- probajte da iskopirate:

```
cp dir1/file1 dir2/file2
```

- da li je kopiranje uspešno? proverite komandom:

```
ls dir2
```

- ukoliko je kopiranje uspešno, obrišite kopiju

```
rm dir2/file2
```

- ako nije uspešno, znači da je to pravo (execute) bilo potrebno za kopiranje.
- dodelite ponovo sebi pravo “x”

```
chmod u+w dir1/file1
```

3. Ukidajte redom prava nad direktorijumom `dir1` i utvrdite koja su vam prava neophodna nad direktorijumom u kome se nalazi izvorišna datoteka da bi ste mogli da
izvršite kopiranje:

3.1 pravo čitanja “r” izvorišnog direktorijuma

- ukinite pravo “r”

```
chmod u-r dir1
```

- probajte da iskopirate:

```
cp dir1/file1 dir2/file2
```

- da li je kopiranje uspešno? proverite komandom:

```
ls dir2
```

- ukoliko je kopiranje uspešno, obrišite kopiju

```
rm dir2/file2
```

- ako nije uspešno, znači da je to pravo (read) bilo potrebno za kopiranje.
- dodelite ponovo sebi pravo “r”

```
chmod u+r dir1
```

3.2. Na sličan način, proverite da li je za kopiranje potrebno pravo upisa “w” u izvorišni direktorijum.

- ukinite pravo “w”

```
chmod u-w dir1
```

- probajte da iskopirate:

```
cp dir1/file1 dir2/file2
```

- da li je kopiranje uspešno? proverite komandom:

```
ls dir2
```

- ukoliko je kopiranje uspešno, obrišite kopiju

```
rm dir2/file2
```

- ako nije uspešno, znači da je to pravo (write) bilo potrebno za kopiranje.
- dodelite ponovo sebi pravo “w”

```
chmod u+w dir1
```

3.3. Na sličan način, proverite da li je za kopiranje potrebno pravo izvršavanja “x” izvorišnog direktorijuma.

- ukinite pravo “x”

```
chmod u-x dir1
```

- probajte da iskopirate:

```
cp dir1/file1 dir2/file2
```

- da li je kopiranje uspešno? proverite komandom:

```
ls dir2
```

- ukoliko je kopiranje uspešno, obrišite kopiju

```
rm dir2/file2
```

- ako nije uspešno, znači da je to pravo (execute) bilo potrebno za kopiranje.
- dodelite ponovo sebi pravo “x”

```
chmod u+w dir1
```

4. Ukidajte redom prava nad direktorijumom `dir2` i utvrdite koja su vam prava neophodna nad odredišnim direktorijumom da bi ste mogli da izvršite kopiranje:

4.1 pravo čitanja “r” odredišnog direktorijuma

- ukinite pravo “r”

```
chmod u-r dir2
```

- probajte da iskopirate:

```
cp dir1/file1 dir2/file2
```

- da li je kopiranje uspešno? proverite komandom:

```
ls dir2
```

- ukoliko je kopiranje uspešno, obrišite kopiju

```
rm dir2/file2
```

- ako nije uspešno, znači da je to pravo (read) bilo potrebno za kopiranje.
- dodelite ponovo sebi pravo “r”

```
chmod u+r dir2
```

4.2. Na sličan način, proverite da li je za kopiranje potrebno pravo upisa “w” u odredišni direktorijum.

- ukinite pravo “w”

```
chmod u-w dir2
```

- probajte da iskopirate:

```
cp dir1/file1 dir2/file2
```

- da li je kopiranje uspešno? proverite komandom:

```
ls dir2
```

- ukoliko je kopiranje uspešno, obrišite kopiju

```
rm dir2/file2
```

- ako nije uspešno, znači da je to pravo (write) bilo potrebno za kopiranje.
- dodelite ponovo sebi pravo “w”

```
chmod u+w dir2
```

4.3. Na sličan način, proverite da li je za kopiranje potrebno pravo izvršavanja “x” odredišnog direktorijuma.

- ukinite pravo “x”

```
chmod u-x dir2
```

- probajte da iskopirate:

```
cp dir1/file1 dir2/file2
```

- da li je kopiranje uspešno? proverite komandom:

```
ls dir2
```

- ukoliko je kopiranje uspešno, obrišite kopiju

```
rm dir2/file2
```

- ako nije uspešno, znači da je to pravo (execute) bilo potrebno za kopiranje.
- dodelite ponovo sebi pravo “x”

```
chmod u+x dir2
```

5. Zapišite sva prava koja su vam neophodna nad izvorišnom datotekom, direktorijumom u kojem se ona nalazi i odredišnim direktorijumom da bi ste mogli da izvršite kopiranje datoteke.
6. Obrišite direktorijume `dir1` i `dir2`:

```
rm -rf dir1 dir2
```

Aktivnost 11: potrebni uslovi za pomeranje datoteke

Odredite potrebne i dovoljne uslove da nešto pomerite.

Kreirajte dva direktorijuma u svom home direktorijumu:

```
mkdir dir1 dir2
```

Kreirajte datoteku u direktorijumu `dir1`:

```
cp /etc/passwd dir1/file1
```

Dodelite svima sva prava nad direktorijumima:

```
chmod 777 dir1 dir2
```

Dodelite svima sva prava nad datotekom:

```
chmod 777 dir1/file1
```

Koristeći sličan postupak kao u zadatku 11. odredite prava koja su potrebna da bi korisnik pomerio datoteku. Potrebno je da ispitate prava koja imate nad izvorišnim i odredišnim direktorijumom i samom datotekom! **SAMOSTALNO ODRADITE!**

Kada završite, obrišite direktorijume `dir1` i `dir2`:

```
rm -rf dir1 dir2
```

Aktivnost 12: potrebni uslovi za brisanje datoteke

Odredite potrebne i dovoljne uslove da nešto obrišete.

Kreirajte direktorijum u svom home direktorijumu

```
mkdir dir1
```

Kreirajte datoteku u tom poddirektorijumu

```
cp /etc/passwd dir1/file1
```

Dodelite svima sva prava nad direktorijumom:

```
chmod 777 dir1
```

Dodelite svima sva prava nad datotekom:

```
chmod 777 dir1/file1
```

Koristeći sličan postupak kao u zadatku 11. odredite prava koja su potrebna da bi korisnik obrisao datoteku. Potrebno je da ispitate prava koja imate nad direktorijumom u kome se datoteka nalazi. **SAMOSTALNO ODRADITE!**

Kada završite, obrišite direktorijum dir1:

```
rm -rf dir1
```

5. Sticky bit i SUID

5.1. Sticky bit (t)

Na javno dostupnim direktorijumima (kao što je `/tmp`) tipična prava pristupa su 777.

- Problem: svako može da briše šta god hoće u tom direktorijumu, pa i fajlove drugog korisnika.
- Rešenje: Postavljanjem sticky bita za direktorijum uvodi se ograničenje da svaki korisnik može obriše samo svoje datoteke u tom direktorijumu (datoteke čiji je on vlasnik).

NAPOMENA: Da bi se postigao ovaj efekat sticky bit se dodaje isključivo direktorijumu. Ako se doda datoteci ima drugu primenu!

Sticky bit se postavlja/ukida na dva načina:

1. dodelom/oduzimanjem prava “t” svim vlasničkim kategorijama u simboličkom režimu:

```
ls -l public_dir
-rwxrwxrw-x   1   nm    nm   4096   dec 23   15:25   public_dir1
```

postavlja se sticky bit:

```
chmod +t public_dir1
ls -l public_dir1
-rwxrwxrw-t   1   nm    nm   4096   dec 23   15:25   public_dir1
```

ukida se sticky bit:

```
chmod -t public_dir1
ls -l public_dir1
-rwxrwxrw-x   1   nm    nm   4096   dec 23   15:25   public_dir1
```

2. navođenjem cifre “1” (ukida se cifrom “0”) pre pristupnih prava u oktalnom režimu

```
ls -l public_dir2
```

```
-rwxrwxrwx   1   nm   nm   4096   dec 23   15:25   public_dir2
```

postavlja se sticky bit:

```
chmod 1777 public_dir2
ls -l public_dir2
-rwxrwxrwt   1   nm   nm   4096   dec 23   15:25   public_dir2
```

ukida se sticky bit:

```
chmod 0777 public_dir2
ls -l public_dir2
-rwxrwxrwx   1   nm   nm   4096   dec 23   15:25   public_dir2
```

Aktivnost 13: sticky bit

NAPOMENA: Kreiranje korisnika I komanda su se detaljno rade na sledećoj vežbi. Međutim, za sada su neophodne za ilustraciju sticky bit-a. Ukratko ako navedete komandu

```
su george
```

sistem će od vas zatražiti da unesete lozinku korisnika george. Nakon toga vi privremeno preuzimate njegov UID. Kada želite da preuzmete svoj stari identitet, zadajete komandu:

```
exit
```

1. Kao root korisnik kreirajte jedan direktorijum:

```
sudo mkdir /sticker
```

2. Dajte svima sva prava nad njim:

```
sudo chmod 777 /sticker
```

3. Postavite sticky bit

```
sudo chmod +t /sticker
```

4. Proverite šta ste napravili:

```
ls -ld /sticker
```

Trebalo bi da dobijete nešto slično ovome:

```
drwxrwxrwt  2  root  staff  4096  dec 23  15:25  sticker
```

Uočite slovo t na kraju pristupnih prava.

5. Kreirajte dva korisnika iz grafičkog okruženje:
   - korisničko ime: kor1
   - lozinka: password1
   - korisničko ime: kor2
   - lozinka: password2
6. Preuzmite akreditive korisnika kor1:

```
su – kor1
```

Ovde navodite lozinku korisnika kor1, dakle “password1”

Kreirajte jednu datoteku na direktorijumu `/sticker`:

```
touch /sticker/myfile
```

Preuzmite stare akreditive

```
exit
```

7. Preuzmite akreditive korisnika `kor2`:

```
su – kor2
```

Probajte da obrišete datoteku `/sticker/myfile`:

```
rm /sticker/myfile
```

Da li je brisanje uspelo ? Da li bi uspelo da nema t flega? Ojasnite.

Preuzmite stare akreditive:

```
exit
```

8. Obrišite sa akreditivnima root korisnika direktorijum `/sticker`:

```
sudo  rm -rf /sticker
```

5.2. SUID (s)

Ovi bitovi služe da drugi korisnici mogu da izvršavaju datoteku sa privilegijama vlasnika, bez promene naloga.

SUID bit se postavlja/uklanja komandom `chmod`, tako što se kategoriji vlasnika dodaje/oduzima pravo “s”:

```
chmod u+s myexploit
ls -l myexploit
-rwsr-xr-x   1   root   staff   2344   dec 23   15:25   myexploit
```

Na mestu prava “x” vlasnika nalazi se oznaka “s” programa.

U numeričkom formatu SUID pravo se dodaje navođenjem cifre 4 pre pristupnih prava a ukida se navođenjem cifre 0. :

```
chmod 4755 fajl
```

NAPOMENA: U retkim situacijama se na mestu SUID prava umesto malog s , koje pokazuje da je sve u redu sa dozvolama, može se pojaviti veliko S. To znači da je SUID pravo podešeno ali da vlasnik iz nekog razloga nema x pravo. To znači da da ni SUID pravo neće imati nikakvog smisla.

VAŽNA NAPOMENA: Jako je glupo davati SUID komandama čiji je vlasnik root a koje se mogu iskoristiti na destruktivan način – bilo namerno, bilo slučajno. S druge strane, zgodno je da npr komandama za arhiviranje ili update sistema zalepite SUID bit.

Aktivnost 14: SUID

1. Otvorite terminal i prijavite se kao korisnik kor1:

```
su - kor1
```

Ovde navodite lozinku korisnika kor1, dakle “password1”

- Pređite na home direktorijum

```
cd
```

- Kreirajte jednu datoteku

```
cp /bin/cp mycopy1
```

- Dajte svima pravo izršavanja

```
chmod a+x mycopy1
```

- Postavite SUID bit

```
chmod u+s mycopy1
```

- Proverite šta ste uradili

```
ls -l mycopy1
```

Objasnite čiji se šta se dešava ako neki korisnik pokrene datoteku mycopy1. Ovo je kopija datoeke cp, dakle to je program za kopiranje datoteka.

2. Iskopirajte kao drugi korisnik neku datoteku u direktorijum /tmp koristeći ovu komandu:

- Preuzmite identitet korisnika kor2

```
su – kor2
```

- Ovde navodite lozinku korisnika kor2, dakle “password1”
- Kopiranje nešto pomoću mycopy1

```
/home/korisnik1/mycopy1 /etc/passwd /tmp/kopija
```

3. Pogledajte kopiju i analizirajte vlasništvo

```
ls -l /tmp/kopija
```

Zaključite sa čijim je akreditivima pokrenuta komanda mycopy1. To možete da odredite tako što ćete proveriti ko je vlasnik datoteke /tmp/kopija.

```
ls -l /tmp/kopija
```

4. Obrišite datoteku /tmp/kopija

```
rm /tmp kopija
```

6. Liste za kontrolu pristupa

Liste za kontrolu pristupa su proširenje tradicionalnog koncepta kontrole pristupa koje omogućava da se pojedinačnim korisnicima ili grupama eksplicitno dodele prava. Osim toga, direktorijumima se mogu dodeliti podrazumevane liste kojima se definišu dozvole koje objekti u direktorijumu nasleđuju. Na taj način moguće je realizovati složeniji scenario kontrole pristupa.

Funkcionisanje liste za kontrolu pristupa ilustrovano je sledećim primerom:

```
# file: myfile
# owner: nmacek
# group: siginf
user::rwx
user:zbanjac:rwx
group::r-x
group:viser:rwx
mask::r-x
other::---
```

Objašnjenje ove liste za kontrolu pristupa je sledeće: vlasnik datoteke myfile je korisnik nmacek, čija su prava RWX. Datoteka je dodeljena grupi siginf, čija su prava R-X. U listi je dodat korisnik zbanjac, kome su dodata prava RWX, ali je maska R-X oduzela pravo upisa, tako da su efektivna prava tog korisnika R-X. Slično, efektivna prava grupe viser su R-X.

Fajl sistemi koji podržavaju ACL koncept moraju se aktivirati sa opcijom acl u datoteci /etc/fstab. Na sledećem primeru se vidi da root FS podrćava ACL koncept dok `/home/data` ne podržava.

```
/dev/sda1 / ext3 acl,relatime 0 1
/dev/sdb2 /home/data auto noacl,defaults 0 0
```

6.1. Rad sa listama za kontrolu pristupa

Liste za kontrolu pristupa se čitaju komandom `getfacl`, a postavljaju, menjaju ili brišu komandom `setfacl`.

Aktivnost 15: Postavljanje liste za kontrolu pristupa

1. Kreirajte dva korisnička naloga

```
sudo useradd korisnik1
sudo useradd korisnik2
```

2. Pozicionirajte se u `/tmp` direktorijum i kreirajte datoteku `lista1`.

```
cd /tmp
touch lista1
```

3. Pogledajte ACL datoteke komandom getfacl

```
getfacl lista1
# file: lista1
# owner: korisnik
# group: users
user::rw-
group::r--
other::r--
```

4. Dodajte korisnika korisnik1 u ACL i dodelite mu prava RWX.

```
setfacl -m u:korisnik1:7 lista1
```

5. Dodajte korisnika korisnik2 u ACL i dodelite mu prava R-X.

```
setfacl -m u:korisnik2:5 lista1
```

6. Pregledajte ACL datoteke komandom getfacl

```
getfacl lista1
# file: lista1
# owner: korisnik
# group: users
user::rw-
user:korisnik1:rwx
user:korisnik2:r-x
group::r--
mask::rwx
other::r--
```

Aktivnost 16: Izmena i brisanje ACL

1. Korisniku korisnik1 oduzmite pravo izmene datoteke lista1.

```
setfacl -m u:korisnik1:5 lista1
```

2. Pregledajte ACL datoteke komandom getfacl

```
getfacl lista1 | grep korisnik1
user:korisnik1:r-x
```

3. Obrišite korisnika korisnik1 iz liste za kontrolu pristupa datoteke lista1.

   setfacl -x korisnik1 lista1

4. Pregledajte ACL datoteke komandom getfacl

```
getfacl lista1
# file: lista1
# owner: korisnik
# group: users
user::rw-
user:korisnik2:r-x
group::r--
mask::rwx
other::r--
```

5. Obrišite celu ACL datoteke

   setfacl -x korisnik1 lista1

6. Pregledajte ACL datoteke komandom getfacl

   getfacl lista1

Da li je lista obrisana?

6.2. ACL maske

ACL maska definiše najviše efektivne dozvole za svakog korisnika ili grupu u ACL. Može se zaobići navođenjem opcije --no-mask.

Aktivnost 17: Rad sa ACL maskama

1. Dodajte korisnika korisnik1 u ACL i dodelite mu prava RWX.

```
setfacl -m u:korisnik1:7 lista1
getfacl lista1
# file: lista1
# owner: korisnik
# group: users
user::rw-
user:korisnik1:rwx
group::r--
```

```
mask::r-x
other::r--
```

Za korisnika korisnik1 efektivna dozvola je R-X zato što maska ukida pravo upisa!

2. Promenite vrednost maske tako da ne ukida pravo upisa.

```
setfacl -m mask:7 lista1
getfacl lista1
# file: lista1
# owner: korisnik
# group: users
user::rw-
user:korisnik1:rwx
group::r--
mask::rwx
other::r--
```

7. Specijalni atributi

Neki od specijalnih atributa na Linux-specifičnim sistemima datoteka koji se tiču kontrole pristupa su:

A Don´t update access time - zabranjuje izmenu vremena poslednjeg pristupa

-a Append only - dozvoljava isključivo dodavanje novih podataka u datoteku, ali ne i izmenu ili brisanje starih, ukoliko je datoteka otvorena u režimu čitanja. Samo superuser može postaviti ili obrisati ovaj atribut;

-i Immutable - datoteka ne može biti modifikovana ili obrisana, ne može joj se promeniti ime niti se može kreirati link koji ukazuje na tu datoteku;

-s Secure deletion - Prilikom brisanja datoteke u sve blokove koji čine datoteku upisuju se nule;

-u Undelete - prilikom brisanja datoteke čuva se njen sadržaj, čime je omogućen povratak obrisane datoteke.

Specijalni atributi mogu se dodeliti datotekama komandom chattr, čija je sintaksa:

```
chattr [-R] mode files
```

Parametar mode se zadaje u simboličkom režimu (slično kao pristupna prava u simboličkom režimu komande chmod). Format parametra mode je:

```
+-=[ASacdistu]
```

Don´t update access time - zabranjuje izmenu vremena poslednjeg pristupaOperator + znači da se postojećoj listi atributa dodaju atributi navedeni u komandnoj liniji,
Atributi su predstavljeni slovima "ASacdijsu".

- Operatorom + dodaju se atributi postojećim
- Operatorom - od postojeće liste atributa oduzimaju atributi navedeni u komandnoj
- liniji.
- Operator = se koristi za dodelu tačno određenog skupa atributa datoteci.
- Opcija -R se koristi za rekurzivnu promenu atributa celokupnog sadržaja direktorijuma
- (datoteka i poddirektorijuma) koji je naveden kao argument komande.

Specijalni atributi datoteka mogu se videti pomoću komande `lsattr`. Na primer ovako izgleda datoteka file1 koja ima flegove c,d i i (Compressed, No Dump i Immutable):

```
lsattr file1
---i-d-c------ file1
```

Aktivnost 16: Upotreba flega immutabile.

1. Iskopirajte datoteku `/etc/passwd` u `/tmp direktorijum` i dodelite svima pravo upisa:

```
cp /etc/passwd /tmp/specpasswd
chmod 666 /tmp/passwd
```

2. Proverite specijalne atribute datoteke

```
lsattr /tmp/specpasswd
```

Dobićete rezultat:

```
-------------e- /tmp/specpasswd
```

što znači da je postavljen specijalni bit e (extent format).

3. Pokušajte editorom nano da izmenite sadržaj datoteke

```
nano /tmp/specpasswd
```

Unesite jednu liniju teksta
Pritinsite Ctrl-X, potvrdite sa Y.

4. Dodajte specijalni bit immutable datoteci:

```
sudo chattr +i /tmp/specpasswd
```

5. Proverite specijalne atribute datoteke

```
lsattr /tmp/specpasswd
```

Dobićete rezultat:

```
----i--------e- /tmp/specpasswd
```

6. Pokušajte editorom nano da izmenite sadržaj datoteke

```
nano /tmp/specpasswd
```

Unesite jednu liniju teksta

Pritinsite Ctrl-X, potvrdite sa Y.

Sistem će prijaviti da NE može da upiše izmene u datoteku, čak iako svi imaju pravo upisa!

7. Pokušajte da obrišete datoteku kao običan korisnik I kao root.

```
rm /tmp/specpasswd
sudo rm /tmp/specpasswd
```

Da li ste u tome uspeli?

8. Ukoliko niste, uklonite immutable bit i obrišite datoteku.

```
sudo chattr -i /tmp/specpasswd
rm /tmp/specpasswd
```