

# VSO, Laboratorijske vaje 2

## 2.1 Zaščita datotek, uporabniki, skupine

1. Ustvarite dva uporabnika, *janka* in *metko*. Navedite samo osnovne podatke (polnega imena ni treba navajati). Uporabnikom določite tudi gesla.
2. Ustvarite novo skupino (grupo) z imenom *studenti*. V to skupino vključite oba uporabnika. Ustvarite še uporabnika *janeza*, ki pa ga ne vključite v to skupino.
3. Prijavite se kot *root*. To najlažje naredite z ukazom *su*. Iz ukazne vrstice ustvarite direktorij */home/zasciten*. Za lastniško skupino direktorija z ukazom *chgrp* določite skupino *studenti*. Lastniku in skupini dodelite vse pravice, ostalim pa nobenih. Uporabite ukaz *chmod*.
4. Prijavite se kot *janko*. Napišite besedilo s pomočjo programa *vi* in ga shranite v direktorij */home/zasciten* kot *besedilo.txt*.
5. Prijavite se kot *metka*. Ali lahko berete *besedilo.txt*?
6. Kot uporabnik *janko* na zgornjo datoteko določite zaščite, da jo lahko bere le lastnik (ki ima vse pravice razen zaganjanja).
7. Prijavite se kot *metka*. Ali lahko *metka* bere datoteko, ki jo je napisal *janko*? Kaj bi moral narediti *janko*, da bi datoteko lahko brala tudi *metka*.
8. Prijavite se kot *janez*. Ali lahko dostopate v direktorij */home/zasciten*?
9. Ostanite prijavljeni kot *janez*. Zaženite ukazno lupino. Z ukazom *su janko* se v lupino prijavite kot *janko*. Ali sedaj lahko dostopate do direktorija */home/zasciten*.
10. Kaj se zgodi, če uporabite ukaz *exit*?
11. Kaj pa, če to storite še enkrat?
12. Kaj naredi ukaz *su* brez parametrov? Prijavite se v lupini kot *root*. Bodite pazljivi, ker imate neomejene pravice.
13. Vključite *janeza* v skupino *studenti*.
14. Naredite še dva uporabnika. To bosta učitelja. Naredite še skupino *ucitelji*. Vanjo dodajte uporabnika, ki ste ju naredili.
15. Za direktorij *zasciten* določite lastniško skupino *ucitelji* tako, da bodo vse datoteke in poddirektorije v njem podedovale to skupino ('rekurzivno'). Pomagajte si z ukazom *man*. Enako dodelite vse pravic uporabniku in skupini, ostalim pa nič (na direktoriju, poddirektorijih, datotekah).
16. Ali lahko učenci počnejo karkoli v direktoriju *zasciten*?
17. Kakšna sta lastnik in lastniška skupina datoteke */home/zasciten/besedilo.txt*? Ali lahko lastnik dostopa do svoje datoteke? Kdo lahko sploh dostopa do datoteke?
18. Kdo lahko spremeni lastništvo zgornje datoteke (ukaz *chown*)? Kdo lahko spremeni lastniško skupino? Ta naj jo postavi na skupino *ucenci*.

## 2.2 Cevovodi, filtri

19. Izpišite seznam uporabikov (datoteka */etc/passwd*), ki uporabljajo lupino Bash (zadnja beseda v vrsticah omenjene datoteke je *bash*). Uporabite *grep*.
20. Izpišite tiste vrstice datoteke */etc/group*, ki se začnejo z besedo "daemon".
21. Koliko datotek README je v direktoriju in poddirektorijih */usr/share/doc*? (find, wc). Kako lahko nadomestimo *wc -l*, z ukazom *grep*? (štejte znake za konec vrstice)
22. Napišite skript, ki preveri, ali določen uporabnik obstaja – uporabite datoteko */etc/passwd*. Ime uporabnika naj skript sprejme kot parameter.
23. Izpišite konfiguracijske datoteke v direktoriju */etc*, ki imajo v imenu številke (find).

### 2.2.1 naloge za sed

24. Spremenite vse letnice 2005 v letnico 2006.
25. Vse številke (znake 0-9) nadomestite z zvezdico.
26. Zbrišite vse vrstice, v katerih je prvi znak lojtra.
27. Zbrišite vse morebitne presledke in tabulatorje na začetku vrstic.
28. Dana je tekstovna datoteka spremenite jo tako, da bo vsaka beseda v svoji vrstici.
29. V datoteki je spisek telefonskih števil, ki vsebujejo okrogle oklepaje, pomišljaje in presledke. Spremenite jo tako, da bodo v njej le številke.

### 2.3 Skripti v bash

30. Napišite skript, ki izpiše pet števil: 5,4,3,2,1. Uporabite zanko while.
31. Napišite skript, ki dano število izpiše od zadaj naprej, npr. če je število 123, naj izpiše 321. (while, /, %)
32. Napišite skript, ki za dano število izpiše vsoto cifer, npr. če je dano število 123, naj izpiše 6 (1+2+3). Rešitev je podobna tisti iz naloge 6.
33. Za računanje z realnimi števili v ukazni vrstici običajno uporabljamo program **bc**. Preberite osnovne informacije o uporabi tega programa: `man bc`. Če želimo izračunati in izpisati rezultat `5.12 + 2.5`, naredimo to tako:

```
$ echo 5.12 + 2.5 | bc
```

Uporabimo torej cev. Popravite program iz prvih lab. Vaj tako, da bo deloval tudi z realnimi števili.

34. Kako rezultat `bc` spravimo v spremenljivko? Uporabite substitucijo ukazov; program iz prejšnje naloge preuredite tako, da rezultat **bc** shranite v spremenljivko in nato spremenljivko izpišete.
35. Napišite skript, ki preveri, ali neka datoteka obstaja ali ne. Ime datoteke podamo kot parameter; v primeru, da program požene s napačnim številom parametrov, naj izpiše obvestilo o napaki.
36. Napišite skript, ki preveri, ali ime skripta vsebuje črko `a`. Navodilo: ime programa shranite v začasno datoteko in uporabite **grep**. Začasno datoteko nato zbrišite. (`$0`, `$$`, `grep`). Program naj izpiše:

```
Ime programa vsebuje a
```

37. Napišite skript, ki izpiše samo določene vrstice dane datoteke; npr. če napišemo:

```
prog 5 5 datoteka
```

bo program izpisal vrstice 5-10 datoteke. (5 vrstic od pete vrstice dalje...)  
(`head`, `tail`, lahko tudi `sed`)

38. Napišite skript, ki v zanki `for` izpiše vse datoteke v trenutnem direktoriju. Uporabite `for .. in *`. V vsaki iteraciji preverite ali gre za datoteko (`if [ -f ]`) in jo v tem primeru izpišite.
39. Uporabite program iz prejšnje naloge. Pred izpisom imena datoteke, ime obdelajte s `sed` tako, da datotekam, katerih imena se končajo s `txt` spremenite končnico v `doc`.