

Bash komande

Darko Novaković



U prošlom broju smo počeli da se upoznajemo sa osnovnim komandama bash-a. U ovom nastavku ćemo obraditi još nekoliko osnovnih komandi koje su apsolutno neophodne u radu. Bilo bi dobro i korisno da svaki primer odradite samostalno u shell-u, jer se tako najlakše i najbrže uči. Pored komandi, kao i u prošlom nastavku, nastavićemo da se upoznajemo sa nekim stvarima karakterističnim kako za bash tako i za sam Linux. Ovo je neizbežno u radu sa shell-om, a ujedno ćete steći potpuniji uvid u konstrukciju Linux-a.

mkdir (make directory)

Kao što samo ime nagoveštava, ova komanda kreira novi direktorijum u tekućem direktorijumu. Veoma jednostavno se koristi. Npr., pozicionirajte se u svoj home direktorijum i u bash-u otkucajte: **mkdir** prvi_dir. Komandom **ls** proverite da li je rezultat primene ove komande očekivan. U listi direktorijuma i fajlova treba da se nađe i direktorijum koji ste upravo napravili.

rm (remove)

Ova komanda se koristi za brisanje fajlova ili direktorijuma. Za potrebe ilustracije, u prethodnom direktorijumu *prvi_dir* kreirajmo jedan fajl sa imenom *prvi_fajl.txt* pomoću naredbe **touch**:

```
~/prvi_dir>touch prvi_fajl.txt
```

Pomoću naredbe **ls** možete proveriti da li je fajl zaista kreiran. Na isti način kreirajmo još jedan fajl, *drugi_fajl.txt*.

Ukoliko nam posle nekog vremena više nije potreban fajl *prvi_fajl.txt*, možemo ga izbrisati pomoću sledeće naredbe:

```
~/prvi_dir>rm prvi_fajl.txt
```

Sa **ls** možete proveriti da li je došlo do željenog rezultata.

Ako iz nekog razloga poželite da izbrišete kreirani direktorijum, to možete uraditi takođe pomoću komande **rm**. Pozicionirajte se u roditeljski direktorijum sa **cd ..** i pokušajte na sledeći način:



Dobićete poruku da shell ne može da izvrši zadatu komandu jer je *prvi_dir* direktorijum a ne fajl. Da biste izbrisali željeni direktorijum sa čitavim njegovim sadržajem otkucajte:

```
~>rm -r prvi_dir
```

Opcija **-r** je opcija za rekurzivno uklanjanje sadržaja direktorijuma. Treba naglasiti da postoji i komanda **rmdir**, ali ona služi za brisanje praznih direktorijuma i zato je **rm** mnogo praktičnija komanda od **rmdir**.

mv (move)

Komanda **mv** ima sledeću sintaksu: **mv** [opcije] *izvor odredište* i pomoću nje se premeštaju ili reimenuju fajlovi. Ukoliko je izvor neki postojeći fajl, a odredište neko nepostojeće ime (ime kojim nije imenovan nijedan fajl ili direktorijum), tada je rezultat primene gornje komande reimenovanje izvorišnog fajla imenom koje je navedeno kao odredište. Ako je situacija slična gornjoj, samo je odredište ime postojećeg fajla, postojeći fajl (odredište) se prepisuje izvorišnim fajlom. Ako je izvor direktorijum a odredište nepostojeće ime, reimenuje se direktorijum. Ako je izvor neki direktorijum, a odredište neki drugi postojeći direktorijum, direktorijum se prebacuje da bude poddirektorijum direktorijuma navedenog kao odredište. Ako je izvor jedan ili više fajlova, a odredište postojeći direktorijum, izvorišni fajlovi se premeštaju u odredišni direktorijum. Sve će ovo biti jasnije posle sledećeg primera.

U home direktorijumu kreirajmo direktorijum *primer2* sa dva poddirektorijuma, *pr1* i *pr2*, i neka se u svakom od njih nalaze po dva fajla: u *pr1* fajlovi *list1.txt* i *list2.txt*, a u *pr2* fajlovi *pismo1* i *pismo2*:

```
~>mkdir primer2
```

```
~>cd primer2
```

```
~/primer2>mkdir pr1 pr2
```

```
~/primer2>cd pr1
```

```
~/primer2/pr1>touch list1.txt list2.txt
```

```
~/primer2/pr1>cd ../pr2
```

```
~/primer2/pr2>touch pismo1 pismo2
```

Promenimo ime fajla *pismo1* u *pismo1.txt*:

```
~/primer2/pr2>mv pismo1 pismo1.txt
```

Premestimo ovaj reimenovani fajl u direktorijum *pr1*:

Omega magazin



OMEGA

```
~/primer2/pr2>mv pismo1.txt ../pr1
```

Komandom **ls** može se proveriti da li je sve korektno urađeno. Na kraju premestimo direktorijum *pr2* u direktorijum *pr1*:

```
~/primer2/pr2>cd ..
```

```
~/primer2>mv pr2 pr1
```

Sami se možete igrati sa prepisivanjem fajla sadržajem nekog drugog fajla. Pri ovome mogu vam zatrebati neke od opcija komande **mv** čiji opis možete pronaći npr. na man stranici ove komande.

```
darke@linux:~/primer2 - Shell - Konsole
Session Edit View Bookmarks Settings Help
darke@linux:~> mkdir primer2
darke@linux:~> cd primer2
darke@linux:~/primer2> mkdir pr1 pr2
darke@linux:~/primer2> cd pr1
darke@linux:~/primer2/pr1> touch list1.txt list2.txt
darke@linux:~/primer2/pr1> cd ../pr2
darke@linux:~/primer2/pr2> touch pismo1 pismo2
darke@linux:~/primer2/pr2> mv pismo1 pismo1.txt
darke@linux:~/primer2/pr2> mv pismo1.txt ../pr1
darke@linux:~/primer2/pr2> ls ../pr1
list1.txt list2.txt pismo1.txt
darke@linux:~/primer2/pr2> ls
pismo2
darke@linux:~/primer2/pr2> cd ..
darke@linux:~/primer2> mv pr2 pr1
darke@linux:~/primer2> ls
pr1
darke@linux:~/primer2> ls pr1
list1.txt list2.txt pismo1.txt pr2
darke@linux:~/primer2> ls pr1/pr2
pismo2
darke@linux:~/primer2>
```

cp (copy)

Ovo je još jedna korisna komanda na čiji ste ekvivalent u grafičkom okruženju već sigurno navikli (copy/paste). Sintaksa joj je sledeća:

cp [opcije] *fajl1 fajl2*, ili

cp [opcije] *lista_fajlova direktorijum*

U prvom slučaju, kopira se fajl1 u fajl2, dok se u drugom slučaju lista_fajlova kopira u listu fajlova sa istim imenima u direktorijum datog naziva. Ako je odredište postojeći fajl, on se prepisuje, a ako je odredište postojeći direktorijum, on se ne prepisuje, već se fajlovi kopiraju u njega. Siguran sam da ste u stanju da se samostalno poigrate sa ovom komandom da biste videli kako funkcioniše.

I na kraju – džoker u rukavu!

Džokeri služe za manipulisanje fajlovima koji imaju neke zajedničke nizove karaktera u imenu. U shell-u postoji 4 tipa džokera:

Omega magazin



OMEGA

- ? odgovara tačno jednom karakteru
- * odgovara proizvoljnom broju karaktera
- [skup_vrednosti] odgovara jednom od karaktera iz skupa vrednosti u zagradama
- [!skup_vrednosti] odgovara jednom karakteru koji ne pripada skupu vrednosti u zagradama.

Pogledajmo na gornjem primeru kako funkcionišu džokeri. Pozicionirajte se u direktorijum ~/primer2/pr1. Posle gornjeg primera, u njemu se nalaze fajlovi: *list1.txt*, *list2.txt* i *pismo1.txt*, kao i direktorijum *pr2* u kome se nalazi fajl *pismo2*.

Rezultat sledeće pretrage direktorijuma pr1:

```
~/primer2/pr1>ls list?.txt
```

biće fajlovi *list1.txt* i *list2.txt*, jer se njihova imena razlikuju upravo na mestu karaktera gde je u pretrazi stavljen džoker ' ?' .

Rezultat pretrage:

```
~/primer2/pr1>ls *1.txt
```

su imena fajlova *list1.txt* i *pismo1.txt*, jer džoker ' *'zamenjuje proizvoljan broj karaktera, pa će rezultat biti svi fajlovi koji na odgovarajućim pozicijama imaju niz karaktera ' 1.txt' .

Sami se igrajte sa ostala dva džokera. Džokeri se mogu i kombinovati ukoliko je to potrebno da bi se došlo do odgovarajućeg rezultata.

U ovom broju toliko. U sledećem nastavku ćemo se više baviti tajnama samog Linux-a, ali ćemo nastaviti i sa komandama bash-a.