

Preporučuje se da pročitate ...

- Borislav Đorđević, Dragan Pleskonjić, Nemanja Maček, *Operativni sistemi: UNIX i Linux*, VETŠ 2004.
5. Rad sa datotekama iz komandne linije

ili:

- uputstva za Bourne Again Shell i komande koje se u vežbi pominju - mogu se lako pronaći na Internetu, na primer na sajtovima:
 - The Linux Documentation Project: <http://www.tldp.org/docs.html>
 - <http://www.linuxmanpages.com/>
 - <http://man.linuxquestions.org/>

Teorijski deo - laboratorijska vežba 5

Komandni interpreter je proces koji obavlja sledeće funkcije u cilju obezbeđivanja interfejsa između korisnika i operativnog sistema: interpretaciju komandne linije, pokretanje programa, redirekciju ulaza i izlaza, povezivanje komandi u pipeline, zamenu imena datoteka, rukovanje promenljivim i kontrolu okoline, shell programiranje.

Tekst unet u shell prompt naziva se komandna linija (command line), čiji je opšti oblik:

```
$ command [opcije] [argumenti]
```

- \$ je odzivnik (shell prompt) i prikazuje se svaki put kad je shell spreman da od korisnika prihvati novu komandu.
- Command može biti interna komanda (ugrađena u shell) ili eksterna (realizovana kao poseban program koji se nalazi u sistemskoj putanji).
- Opcije i argumenti su parametri koje shell prenosi komandi, pri čemu su argumenti najčešće obavezni i predstavljaju ime neke datoteke, direktorijuma, korisnika ili, na primer, identifikator procesa.

Ime komande, opcije i argumenti su osetljivi na velika i mala slova (case-sensitive) i razdvajaju se razmakom.

```
$ free                # samo komanda
$ df /dev/hda1         # komanda (fd) i argument (/dev/hda1)
$ cp 1.txt 2.txt       # komanda (cp) i dva argumenta (1.txt i 2.txt)
$ date -u              # komanda (date) i opcija (-u)
$ ls -l /etc           # komanda (ls), opcija (-l) i argument (/etc)
```

Nakon interpretacije komandne linije shell inicira izvršenje zadate komande. Ukoliko komanda nije interna (ugrađena u shell, poput komande cd) shell traži izvršnu datoteku koja odgovara imenu komande u direktorijumima navedenim u sistemskoj putanji (promenljiva \$PATH). Nakon toga shell pokreće program i prosleđuje mu argumente i opcije navedene u komandnoj liniji.

Ukoliko se izvršna datoteka nalazi u tekućem direktorijumu ili u nekom direktorijumu koji nije u sistemskoj putanji, ime komande se mora zadati sa putanjom:

```
$ ./myscript          # pokretanje programa iz tekućeg direktorijuma
$ /usr/sbin/useradd    # pokretanje programa iz /usr/sbin
```

Više UNIX komandi mogu se navesti u istoj komandnoj liniji ukoliko su razdvojene znakom tačka-zarez.

```
$ clear ; date        # dve komande koje se izvršavaju jedna za drugom
```

Redirekcija ulaza i izlaza

UNIX komande:

- primaju podatke sa standardnog ulaza (stdin) – najčešće tastatura
- rezultate izvršenja šalju na standardni izlaz (stdout) – najčešće ekran
- poruke o greškama na standardni uređaj za greške (stderr) – najčešće ekran

UNIX omogućava da se ulaz i izlaz komande preusmere:

- Ulaz komande preusmerava se pomoću znaka < (manje od). Na primer, moguće je standardni ulaz sa tastature zameniti nekom tekstualnom datotekom:

```
$ wc -l < /tmp/johnsmith.dat
```

- Za redirekciju izlaza se koristi znak > (veće od). Ukoliko se redirekcija vrši u postojeću datoteku datoteka se briše, a zatim se kreira nova u koju se smešta rezultat izvršenja komande. Ukoliko korisnik želi da se rezultat izvršenja doda na postojeću datoteku bez brisanja njenog sadržaja, za redirekciju izlaza koristi se znak >>. Sledeći primer ilustruje redirekciju izlaza na štampač, u novu datoteku i postojeću datoteku i kreiranje prazne datoteke.

```
$ sort kyuss.txt > /dev/lp0
$ ls -l /home/johnsmith > myfile
$ ls -l /tmp/johnsmith >> myfile
$ >emptyfile
```

- Standardni izlaz za greške preusmerava se ukoliko korisnik želi da sačuva rezultat izvršenja komande u nekoj datoteci radi kasnije analize grešaka (na primer, debugovanje programa u fazi razvoja). Izlaz se uglavnom preusmerava u tekstualne datoteke pomoću znaka 2>.

```
$ ./testprogram 2> debugging.txt
```

Povezivanje komandi u pipeline

Pipeline funkcioniše na sledeći način: standardni izlaz komande sa leve strane znaka pipe - cev (|) postaje standardni ulaz komande sa desne strane znaka. Na primer, pretpostavimo da korisnik želi da zna koliko datoteka ima u direktorijumu u /etc. Korisnik može izlistati sadržaj direktorijuma u takozvanom long-listing formatu komandom ls -l /etc i izvršiti redirekciju izlaza u privremenu datoteku /tmp/files_in_etc. Dalje, korisnik može

prebrojati broj linija u privremenoj datoteci pomoću komande `wc -l /tmp/files_in_etc` (word count).

```
$ ls -l /etc > /tmp/files_in_etc
$ wc -l < /tmp/files_in_etc
145
```

Prva komanda koristi datoteku `/tmp/files_in_etc` kao standardni izlaz, a druga kao standardni ulaz, što znači da se prebrojavanje datoteka u direktorijumu `/etc` može realizovati pomoću pipeline sprege komandi `ls` i `wc`:

```
$ ls -l /etc | wc -l
145
```

U pipeline se može povezati veći broj komandi. Sledeći primer ilustruje pipeline koji prebrojava sve korisnike koji su prijavljeni na sistem a ime im počinje slovom `a` i prikazuje ih sortirane po abecednom redu.

```
$ finger | grep a* | wc -l | sort
```

I pipeline i redirekcija kao funkcije komandnog interpretera imaju svoje prednosti. Preusmeravanje se koristi kad se koristi podatak iz datoteke koja će se i dalje koristiti. U suprotnom se koristi pipe, čijom se upotrebom sprečava nepotrebno gomilanje podataka na diskovima.

Džokeri

Imena datoteka mogu se zameniti džoker karakterima (joker) `*`, `?` i `[]`. Argument komande koji sarži džoker karakter zamenjuje se odgovarajućom listom datoteka shodno pravilima zamene. Komandni interpreter izvršava ovu zamenu pre izvršavanja same komande, odnosno pre pokretanja programa.

- karakter `*` menja bilo koji niz znakova proizvoljne dužine

```
# ls -d /var/s*
```

```
/var/spool          /var/stat
```

- karakter `?` menja bilo koji znak (tačno jedan znak)

```
# ls -d /?bin
/sbin
```

- opseg [poc-kraj] menja tačno jedan znak koji pripada tom opsegu. Opseg se ne sme zadati u opadajućem redu.

```
# ls -d /etc/[a-d][a-d]*
/etc/adduser.conf  /etc/bash.bashrc      /etc/bash_completion.d
/etc/adjtime       /etc/bash_completion  /etc/calendar
```

Kontrolni karakteri

Kontrolni karakteri koriste se za izvršavanje raznih specifičnih zadataka, kao je prekidanje izvršenja procesa.

Korisnik zadaje kontrolne karaktere tako što istovremeno pritisne taster <Ctrl> i još jedan karakter. Kontrolni karakteri Bourne-again shella koji se najčešće koriste su:

- <Ctrl-c> prekida izvršenje procesa koji radi u prvom planu;
- <Ctrl-d> označava kraj datoteke. Koristi se za napuštanje programa koji podatke čitaju sa standardnog ulaza (tastatura).
- <Ctrl-s> privremeno zaustavlja izvršenje procesa u prvom planu. Može se koristiti prilikom pregledanja sadržine nekog velikog direktorijuma komandom ls ili ukoliko se neka datoteka prikazuje na ekranu programom cat;
- <Ctrl-g> nastavlja se izvršenje procesa u prvom planu.

Alijasi

Alias je način dodele kraćeg imena pomoću kog se određena komanda, ili niz komandi, može pozvati iz komandnog interpretera. Na primer, može se dodeliti alias ll (long listing) koji izvršava komandu ls -l.

```
# alias myroot="ls -ld /root"
# myroot
drwxr-x--- 7 root root 1024 May 4 17:41 /root
# unalias myroot
# myroot
bash: myroot: command not found
```

History

Komandni interpreter bash upisuje svaku komandnu liniju u history datoteku. Ovo omogućava da se prethodne komande ponove, pri čemu se pre ponovnog izvršavanja mogu i izmeniti. Bash shell history datoteku smešta u home direktorijum korisnika (~/.bash_history), i u njoj podrazumevano čuva 1000 prethodno izvršenih komandi. Broj komandi koje se mogu smestiti u ovu datoteku može se promeniti pomoću promenljive HISTSIZE. Komanda history u bash shellu prikazuje prethodno izvršene komande:

```
$ date
$ whoami
$ finger
$ mail
$ history 5
 329 date
 330 whoami
 331 finger
 332 mail
 333 history 5
```

Najjednostavniji način za ponavljanje komandi je korišćenje gornje strelice (ili kombinacije tastera <Ctrl+p>) i donje strelice (ili kombinacije tastera <Ctrl+n>). Ovim tasterima korisnik bira jednu komandu iz datoteke ~/.bash_history, pri čemu se počinje od poslednje zadate komande.

Kompletiranje imena datoteka

Bourne-again shell sadrži mogućnost za kompletiranje imena datoteka. Korisnik može da unese samo nekoliko prvih karaktera imena datoteke, a zatim da pritisne karakter <Tab> kojim se naznačava shellu da završi ime datoteke. Na primer:

```
$ ls -l /etc/pas<Tab>
$ ls -l /etc/passwd
```

Ukoliko shell u tekućem direktorijumu pronađe više od jedne datoteke čije ime počinje tim karakterima, korisnik će morati da unese još nekoliko karaktera u imenu datoteke, a zatim da ponovo pritisne taster <Tab>.

Prilagođavanje odzivnika (shell prompt)

U podrazumevanom stanju, odziv u bash shellu je znak \$ za regularne korisnike, odnosno # za superusera. Svaki korisnik može promeniti svoj shell prompt pomoću lokalne promenljive PS1 (prompt string). Ove promene važe samo za tekući shell. Kada se korisnik ponovo prijavi na sistem dobiće podrazumevani prompt.

Sledeći primer ilustruje korišćenje komande (uname -n) u odzivu – prompt koji pokazuje ime računara na kome je korisnik prijavljen. Znak backquote (') koristi se radi zamene niza pod navodnicima izlazom komande uname.

```
$ PS1="`uname -n` $ "
d-fens.requiem.org $
```

Korisnik takođe može pomoću specijalnih karaktera formirati odziv koji prikazuje: "\u" korisničko ime, "\d" tekući datum, "\h" ime host računara, "\W" tekući direktorijum:

```
tmp $ PS1="\u@\h, \d, \W $ "
nmacek@tulip, Wed Dec 24, tmp $
```

Dobijanje pomoći

Prilikom rada na UNIX sistemu korisnik može dobiti pomoć od sistema na nekoliko načina. Najjednostavniji način za dobijanje pomoći je navođenje opcije -help samoj komandi. Na primer:

```
$ mkdir -help
```

Jedan od najkompletnijih izvora pomoći (ponekad i jako komplikovan i nejasan) su stranice uputstva za korišćenje komande (manual page, odnosno man page). Određena stranica uputstva prikazuje se pomoću komande man, čija je opšta sintaksa:

```
$ man mount
```

Stranice su troff formatirane, a za navigaciju po stranici se mogu koristiti sledeći tasteri: <Space> sledeća stranica, "b" prethodna stranica, <Enter> (ili <UP>) sledeći red, <DOWN> sledeći ili prethodni red, "q" napuštanje programa man i povratak u shell.

Za interne shell komande postoji još jedan vid dokumentacije - infopage. Slično stranicama uputstva, info

stranice se dobijaju pomoću komande info:

```
$ info groups
```

Lokatori komandi

Lokacija određenog programa u aktivnom UNIX stablu može se odrediti komandama which i whereis. Komanda whereis prikazuje lokaciju izvršnih datoteka, izvornog koda i prateće dokumentacije programa. Primer ilustruje upotrebu komande whereis za pronalaženje lokacije programa insmod (koji se koristi za dodavanje modula u aktivno jezgro):

```
$ whereis insmod
insmod: /sbin/insmod /usr/share/man/man8/insmod.8.gz
```

Za razliku od komande whereis, komanda which prikazuje samo lokaciju izvršnih datoteka. Komanda koristi sličan algoritam kao i Bourne-again shell - traži izvršnu datoteku u direktorijumima navedenim u sistemskoj putanji i ukoliko je nađe, prikazuje putanju i ime prve pronađene komande na standardnom izlazu.

```
# which fdisk
/sbin/fdisk
```

Prikazivanje informacija o sistemu

Komanda uname -a na standardnom izlazu štampa informacije o sistemu, poput imena operativnog sistema, arhitekture hardvera i imena host računara. Bez parametra -a, prikazuje se samo ime operativnog sistema.

```
$ uname
Linux
$ uname -a
Linux tulip 2.4.19-686 #1 SMP Sat Apr 24 10:56:05 CEST 2004 i686 unknown
```

Dodatne informacije o sistemu, kao što su količina slobodne memorije, skup aktiviranih sistema datoteka i skup modula učitanih u kernel dostupne su svim korisnicima pomoću odgovarajućih komandi (u ovom slučaju free, mount i /sbin/lsmmod).

Određivanje tipa datoteke

Za razliku od programa koji rade u grafičkom okruženju (kao što su gimp i Open Office), programi koji rade u UNIX komandnoj liniji ne prepoznaju datoteke na osnovu ekstenzija. Posmatrano sa stanovišta komandne linije može se reći da UNIX nema pojam ekstenzije datoteke. Zato na UNIX sistemima postoji poseban program file pomoću koga korisnici mogu odrediti tip datoteke. Određivanjem tipa datoteke korisnik određuje i program kojim će izvršiti neku akciju nad datotekom (kao što je prevođenje C programa, izvršavanje shell programa ili prikazivanje slike ili video zapisa).

Tipovi datoteka

Na UNIX sistemima postoji nekoliko osnovnih tipova datoteka:

- tekstualne datoteke - ASCII (neformatiran tekst), English text (tekst sa interpunkcijskim karakterima) i izvršni shell programi
- izvršne (binarne) datoteke
- datoteke u koje su smešteni podaci (na primer, Open Office Writer dokument)

Prilikom određivanja tipa datoteke komanda file izvršava sledeće testove: test specijalnih datoteka (filesystem test), test magičnih brojeva (magic number test) i jezički test (language test).

Sledeći primeri ilustruju upotrebu komande file:

```
$ file /etc/hosts
/etc/hosts: ASCII English text
$ file /bin/cp
/bin/cp: ELF 32-bit LSB executable, Intel 80386, version 1 (SYSV),
dynamically linked (uses shared libs), stripped
$ file /dev/hda
/dev/hda: block special (3/0)
$ file /dev/lp0
/dev/lp0: character special (6/0)
$ file /sbin/rmt
/sbin/rmt: symbolic link to /usr/sbin/rmt
$ file /etc
/etc: directory
```

Komanda strings

Komanda strings može se upotrebiti za prikazivanje vidljivih karaktera u izvršnoj ili binarnoj datoteci. Razni aplikativni programi upisuju ime programa u prvih par linija datoteke, tako da se komanda strings može upotrebiti radi određivanja programa kojim je datoteka napravljena, a samim tim i (alternativnog) programa koji je pogodan za otvaranje datoteke. Komanda strings je ovde upotrebljena samo radi demonstracije prikaza štampanih karaktera izvršne datoteke.

```
# strings /bin/cp
/lib/ld-linux.so.2
libc.so.6
strcpy
...
Try `strings --help' for more information.
Usage: strings [OPTION]... SOURCE DEST
  or:  strings [OPTION]... SOURCE... DIRECTORY
  or:  strings [OPTION]... --target-directory=DIRECTORY SOURCE...
...
Written by strings.
This is free software; see the source for copying conditions.  There is NO
warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
memory exhausted
```