tar - Package multiple files into a single file.

gzip - Compress files with GNU Zip

gunzip - Uncompress GNU Zip files.

bzip2 - Compress files in BZip format.

bzcat - Compress/uncompress Bzip files via standard input/output

compress - Compress files with traditional Unix compression

uncompress - Uncompress files with traditional Unix compression

zcat - Compress/uncompress file via standard input/output (gzip or compress)

zip - Compress files in Windows Zip format.

unzip - Uncompress Windows Zip files.

munpack - Extract MIME data to files.

mpack - Convert a file into MIME format

tar [options] [files]

tar - “pakuje” / komprimuje zadate fajlove i direktorijume u jedan (komprimovani) fajl.

$ tar -czf myarchive.tar.gz mydir (*Create*)

$ ls -Lg myarchive.tar.gz

-rw-r–r– 1 smith 350 Nov 7 14:09 myarchive.tar.gz

$ tar -tf myarchive.tar.gz (*List contents*)

$ tar -xvf myarchive.tar myfile myfile2 myfile3

-c - Create an archive. You’ll have to list the input files and directories on the command line.

-r - Append files to an existing archive.

-u - Append new/changed files to an existing archive.

-A - Append one archive to the end of another: for example, $ tar -A -f first.tar second tar appends the contents of second.tar to first.tar. Does not work for compressed archives.

-t - List (test) the archive.

-x - Extract files from the archive.

-f file - Read the archive from or write the archive to, the given files. This is usually a tar file on disk (such as myarchive.tar) but also be a tape drive (suck as /dev/tape).

-d - Diff (compare) the archive against the filesystem

-z - Use gzip compression

-j - Use bzip2 compression

-Z - Use Unix compression

-v - Verbose mode: print extra information.

-h - Follow symbolic links rather than merely copying them.

-p - When extracting files, restore their original permissions and ownership

gzip [options] [files] gunzip [options] [files]

gzip i gunzip - (de)komprimuju fajlove u GNU Zip format; komprimovani fajlovi (arhive) imaju ekstenziju gz; po difoltu, originalni fajl se briše

gzip file - Compress file to create file.gz. Original file is deleted.

gzip -c file - Produce compressed data on standard output.

cat file | gzip - Produce compressed data from a pipeline.

gunzip file.gz - Uncompress file.gz to create file. Original file.gz is deleted.

gunzip -c file.gz - Uncompress the data on standard output.

cat file.gz | gunzip - Uncompress the data from a pipeline.

bzip2 [options] [files] bunzip2 [options] [files]

bzip2 i bunzip2 - (de)komprimuju fajlove u Burrows-Wheeler format; kod njega je fajlovi više komprimuju, ali procedura duže traje; komprimovani fajlovi imaju ekstenziju bz2

bzip2 file - Compress file to create file.bz2. Original file is deleted.

bzip2 -c file - Produce compressed data on standard output.

cat file | bzip2 - Produce compressed data on standard output.

bunzip2 file.bz2 - Uncompress file.bz2 to create file. Original file.bz2 is deleted

bunzip2 -c file.bz2 - Uncompress the data on standardoutput.

cat file.bz2 | bunzip2 - Compress the data on standard output.

compress [options] [files] uncompress [options] [files]

compress i uncompress - (de)komprimuju fajlove u standardni Unix komprimovani format (prema Lempel Ziv algoritmu, koji se koristi i kod grafičkog GIF formata); komprimovani fajlovi imaju ekstenziju Z

compress file - Compress file to create file.Z. Original file is deleted.

compress -c file - Produce compressed data on standard output.

Cat file | compress - Produce compressed data from a pipeline.

uncompress file.Z - Uncompress file.Z to create file. Original file.Z is deleted.

uncompress -c file.Z - Uncompress the data on standard output.

cat file.Z | uncompress - Uncompress the data from a pipeline.

zip [options] [files]

zip - komprimuje fajlove u Windows Zip format, dok ih komanda unzip dekomprimuje; ekstenzija komprimovanih fajlova je zip; originalni fajl ovom komandom se ne briše.

zip myfile.zip file1 file2 file3 - Pack

zip -r myfile.zip dirname - Pack recursively

unzip -l myfile.zip - List contents.

unzip myfile.zip - Unpack

diff - Line-by-line comparison of two file or directories. (liniju po liniju za text file)

comm - Line-by-line comparison of two sorted files. (liniju po liniju za text file)

cmp - Byte-by-byte comparison of two files. (bajt po bajt, za txt i binarne fileove)

md5sum - compute a checksum of given files (MD5) (poredjenje parametara checksum)

diff [options] file1 file2

diff - poredi dva fajla liniju po liniju; kada poredi tekstualne fajlove, komanda generiše detaljan izveštaj o uočenim razlikama; ako su fajlovi isti, ona ne daje nikakav izlaz; ako se upotrebi za poređenje binarnih fajlova, ona samo izvesti da li su isti ili nisu

Default izlaz ove komande, pri poređenju dva tekstualna fajla (A i B), izgleda ovako:

Broj linije u fajli i vrsta razlike

< Sekcija fajla A, ako postoji

—

> Sekcija fajla B, ako postoji

Ovakav tekst se ispisuje za svaku uočenu razliku; znak < označava odgovarajuću sekciju (jednu ili više uzastopnih linija teksta) u fajlu A, a znak > označava odgovarajuću sekciju u fajlu B; separator između dve sekcije je povorka od tri crte —

Pri uočavanju razlika, fajl A služi kao referenca; ova komanda zapravo uočava promene koje bi trebalo napraviti u fajlu A kako bi se dobio fajl B

fileA izgleda ovako:

Hello, this is a wonderful file.

The quick brown fox jumped over the lazy dogs.

Goodbye for now.

fileB izgleda ovako:  
The quick blue fox jumped over the lazy dogs.

Goodbye for now.

Linux r00lz

$ diff fileA fileB

1,2c1 (*fileA lines 1-2 became fileB line1*)

< Hello, this is wonderful file. (*Lines 1-2 of fileA*)

< The quick brown jumped over

— (diff separator)

> The quick blue fox jumped over (line 1 of fileB)

4a4 (Line 4 was added in fileB)

> Linux r00lz (*The added line*)

Opcije za kontrolu izlaznog formata:

-y - Side-by-side format. Use -W to adjust the width of the output

-e - Create an ed script that would change fileA into fileB if run.

-q - Dont report changes, just say whether the files differ

$ diff dir1 dir2 (poredjenje direktorijuma)

-b - Don't consider whitespace.

-B - Don't consider blank lines

-i - Ignore case.

-r - When comparing directories, recurse into subdirectories.

sdiff - oucene razlike izmedju 2 file-a ispisuje u 3. file, osnovu instrukcija korisnika

comm [options] file1 file2

comm - poredi dva sortirana tekstualna fajla (ona kod kojih su linije sortirane po abecedi), i generiše izlaz u tri kolone, odvojene tabovima:

Commfile1: Comfile2:

apple baker

baker charlie

charlie dark

$ comm commfile1 commfile2

apple

baker

charlie

dark

-1 - Suppress column 1.

-2 - Suppress column 2.

-3 - Suppress column 3.

cmp [options] file1 file2 [offset1 [offset2]]

cmp - poredi dva fajla (tekstualna ili binarna), bajt po bajt (to jest karakter po karakter): ako su fajlovi isti, ne vraća nikakav rezultat; ako nisu, komanda po difoltu daje obaveštenje o poziciji na kojoj se nalazi prva razlika; preciznije, daje redni broj prvog bajta (karaktera) koji se u dva fajla razlikuje

fajlA.txt: hello ab, fajlB.txt: hello cd

$ cmp fajlA.txt fajlB.txt

fajlA.txt fajlB.txt differ: byte 7, line 1

$ cmp -l fajlA.txt fajlB.txt

7 141 143

8 142 144

$ cmp docA.docx docB.docx

docA.docx docB.docx differ: byte 2416, line 5

-l - Long output: print all differences, byte by byte:

$ cmp -l myfile yourfile

225 167 127

This means at offset 255 (in decimal), myfile has a small “w” (octal 167) but yourfile has a capital “W” (octal 127)

-s - Silent output: don't print anything, just exit with an appropriate return code. 0 if the files match, 1 if they don't. (Or other codes if the comparison fails)

md5sum files | –check file

md5sum - izračunava parametar checksum za dati fajl, upotrebom algoritma koji se naziva MD5; checksum je 32-cifreni heksadecimalni broj

$ md5sum myfile

48760f921ec6111e3979efa14e22535d myfile

$ md5sum myfile | cut -c1-32 > sum1

$ md5sum myfile2 | cut -c1-32 > sum2

$ diff -q sum1 sum2

Files sum1 and sum2 differ

shasum - zasnovana na algoritmima SHA-1 i SHA-256; stare komande sum i cksum treba izbegavati kao nepouzdane

lpr - Print a file

lpq - View the print queue

lprm - Remove a print from the queue

lpr [options] [files]

lpr ("line printer") - šalje fajl štampaču na štampanje

$ lpr -P myprinter myfile

-P *printername* - Send the file to printer printername, which you have previously set up.

-# *N* - Print *N* copies of the file.

lpq [options]

lpq ("line printer queue") - ispisuje listu svih poslova štampanja (eng. pring job) koji čekaju na izvršenje

$ lpq -P *printername*

Rank Owner Job Files Total size

Active root 133 pr1 467 bytes

1st chavez 61 l1726.f 74578 bytes

2nd harvey 78 fpppp.F 12394 bytes

-P *printername* - List the queue for printer *printername*

-a - List the queue for all printers

-l - Be verbose: Display informations in a longer format

lprm [options] [job\_IDs]

lprm ("line printer remove") zaustavlja/poništava jedan ili više poslova štampanja: da bi poništio neki posao, korisnik prvo pomoću komande lpq treba da sazna ID tog posla, i da ga zatim zada komandi lprm

$ lprm -P printername 61 78

df - Displa available space on mounted filesystems

mount - Make a disk partition accessible.

umount - Unmount a disk partition (make it inaccessible)

fsck - Check a disk partition for eros

eject - Eject a CD, DVD or other removable disk

df [options] [disk devices | files | directories]

df (“disk free”) - prikazuje za datu particiju: ukupan memorijski prostor, iskorišćeni prostor i slobodan prostor; ako se zada fajl ili direktorijum, ova komanda daje informaciju o particiji na kojoj se on nalazi; bez argumenata, komanda daje informaciju o svim montiranim fajlsistemima (uređajima)

$ df

Filesystem 1k-blocks Used Avail Use% Mounted on

/dev/sda 1011928 225464 735060 24% /

/dev/sda9 521748 249148 246096 51% /var

/dev/sda8 8064272 4088636 3565984 54% /sr

/dev/sda10 8064272 4586576 3068044 60% /home

-k - List sizes in kilobytes (the default)

-m - List sizes in megabytes.

-B N - Display sizes in blocks of N bytes. (Default = 1024)

-h - Print human-readable output and choose the most appropriate unit for each size. For example, if your two disks have 1 gigabyte and 25 kilobytes free, respectively df -h prints 1G and 25K. The -h option uses powers of 1024, whereas -H uses powers of 1000

-l - Display only local filesystems, not networked filesystems

-T - Include the filesystem type (ext3, vfat, etc.) in the output.

-t type - Display only filesystems of the given type.

-x type - Dont display filesystems of the given type.

mount [options] device | directory

mount - montira uređaj (ili particiju) device na direktorijum directory; tako na primer ona montira disk /dev/sda1 (formalno, to je matični fajl diska) na direktorijum /mnt/mydir, čime fajlovi koji se nalaze na disku postaju dosupni u datom direktorijumu

$ sudo mkdir /mnt/mydir

$ ls /mnt/mydir (*Notice its empty*)

$ sudo mount /dev/sda1 /mnt/mydir

$ ls /mnt/mydir

file1 file2 file3 (*Files on the mounted partition*)

$ df /mnt/mydir

Filesystem 1k-blocks Used Avail Use% Mounted on

/dev/sda1 1011928 275744 674780 30% /mnt/mydir

-t *type* - Specify the type of filesystem, such as ext3 or ntfs.

-l - List all mounted filesystems. Works with -t too

-a - Mount all filesystems listed in /etc/fstab. Ignores entries that include the noauto option. Works well with -t too.

umount [options] [device | directory]

umount - radi obrnuti proces od mount: ona obavlja odmontiranje uređaja, tako da više ne bude vidljiv u okviru virtualnog fajlsistema

$ umount /dev/cdrom /cd (*odmontira se CD*)

$ sudo umount -a (*ocitava tabelu montiranja*)

fsck [options] [devices]

fsck ("filesystem check") - proverava zadatu particiju (to jest fajlsistem na njoj) i, ako joj se to zada, ispravlja greške koje detektuje; ova komanda se automatski startuje 135 prilikom butovanja sistema, ali korisnik može da je ručno pokrene kad hoće, ako/kad mu je to potrebno

$ sudo umount /dev/sda10

$ sudo fsck -f /dev/sda10

Pass 1: Checking inodes, blocks and sizes

Pass 2: Checking directory structure

Pass 3: Checking directory connectivity

Pass 4: Checking reference counts

Pass 5: Checking group summary informations

/home/172/1281696 files (11.6% non-contiguous)

-A - Check all disks listed in /etc/fstab in order

-N - Print a description of the checking that would be done, but exit without performing any checking.

-r - Fix errors interactively, prompting before each fix.

-a - Fix errors automatically (use only if you really know what are you doing…)

eject [options] [device\_name]

eject obavlja istu aktivnost kao kada korisnik pritisne open/close dugme na CD-DVD drajvu

$ eject

-n - Don’t eject anything, just say what would be done. Combine with -v for a detailed description.

-v - Produce verbose output.

-d - Print the na,e of the default device to be ejected, such as /dev/cdrom and exit

rsync - Efficiently copy a set of files, even across a network.

dd - Low-level copying of data

growisofs - Burn a DVD or Blue-ray disc

rsync [options] source destination

Rsync ("remote sync", sync=synchronization, to jest sinhronizacija) - kopira skup fajlova (za bekap ili mirorovanje); prilikom pravljenja kopije, korisnik može da odabere da li će se istovremeno iskopirati i dozvole pristupa i ostali atributi (opcija -a, od archive); može se izvršavati na mreži ili na jednom računaru; ima više od 50 opcija (!)

$ rsync -a mydir mydir2

$ rsync -a mydir/ mydir2

$ rsync -a mydir [smith@server.example.com](mailto:smith@server.example.com):D2

-o - Copy the ownership of the file. (You might need superuser privileges on the remote host)

-g - Copy the group ownership of the file. (You might need superuser privileges on the remote host)

-p - Copy the file permissions.

-t - Copy the file timestamps.

-r - Copy directories recursively (i.e., including their contents).

-l - Permit symbolic links to be copied (not the files they point to).

-D - Permit devices to be copied. (Superuser only)

-a - Mirroring: copy all attributes of the original files. This implies all of the options -ogptrlD.

-x - When copying a tree of files, remain within the current filesystem. Do not cross over into other mounted filesystems.

-n - Dry-run mode: don't actually do any copying. Just display what would be done

-v - Verbose mode: print informations about what’s happening during the copy. Add –progress to display a numeric progress meter while files are copied.

dd [options]

dd - kopira podatke na niskom nivou (što se zove i raw copy = sirovo kopiranje), bajt po bajt, umesto preko fajlsistema; ovakav način kopiranja ima nekih svojih prednosti na primer kada je u pitanju spasavanje izgubljenih podataka (neće se ulaziti u detalje)

$ dd if=fileA of=fileC

7+1 records in

7+1 records out

3816 bytes (3.8 kB) copied, 0.000356028 s, 10.7 MB/s

$ dd if=/dev/device1 of=/dev/devicxe2 bs=512 conv=noerror

noerror nalaže da se kopiranje obavi čak i ako se detektuje neka greška pri očitavanju

if=file - Specify an input file or device

of=file - Specify an output file or device.

bs=N - Copy N bytes at a time, known as the “blocksize”. (To set the block size differently for the input and the output use ibs and obs, respectively)

skip=N - Skip past N blocks of input before starting the copy

seek=N - Discard N blocks of output before starting the copy.

conv=spec - Convert the data being copied. Spec can be ucase (covert to uppercase), lcase(convert to lowercase), ascii (convert to ASCII from EBCDIC) and many other listed on the manpage.

growisofs [options] tracks

growisofs - bekapuje podatke na optičkom disku (CD/DVD ili Blueray); operacija bekapa sastoji se iz dva koraka

$ more /proc/sys/dev/cdrom/info

CD-ROM informations, Id: cdrom.c 3.20 2003/12/17

drive name: sr1 sr2

drive speed: 48 12

drive # of slots: 1 1

more ispisuje sadrzaj tektualnog fajla (slično kao less)

$ growisofs -Z /dev/sr1 -R -J /home/smith

-Z nalaže da se podaci upisuju na disk direktno (bez upotrebe takozvanog ISO image fajla – neće se objašnjavati).

pomoću opcija -R i -J se omogućava veća fleksibilnost pri upisivanju (neće se objašnjavati)

ps - List process.

uptime - View the system load.

w - List active processes for all users.

top - Monitor resource-intensive processes interactively

free - Display free memory

ps [options]

ps ("process status") - ispisuje listu svih procesa sâmog korisnika, i opciono procesa drugih korisnika; ima oko 80 opcija (od kojih su mnoge neupotrebljive ili nekonzistentne

$ ps

PID TTY TIME CMD

4706 pts/2 00:00:01 bash

15007 pts/2 00:00:00 emacs

16729 pts/2 00:00:00 ps

$ ps -ef

UID PID PPID C STIME TTY TIME CMD

root 0 0 0 09:36:35 ? 0:00 sched

root 1 0 0 09:36:35 ? 0:02 /etc/init

mari 7997 1 10 09:49:32 ttpy3 0:04 ps-ef

mike 12923 11324 9 10:19:49 ttyp5 56:12 csh

Opcija -e traži prikaz svih procesa (i drugih korisnika); opcija -f traži "full format listing"; da se dobije lista sa celim komandama, kuca se opcija -efww; može se dodati i opcija - less da bi se prikazivalo ekran po ekran

$ ps -U smith (prikaz svih procesa korisnika smith)

$ ps -ef | grep smith (isto, ali sa grep)

$ ps -p1,2,3505 (prikaz procesa čiji su PID 1, 2 i 3505)

$ ps -uax | grep emacs (prikaz svih procesa koje je pokrenula komanda emacs)

uptime

uptime - daje informaciju koliko vremena je proteklo od poslednjeg butovanja sistema; to vreme se zapravo naziva uptime

$ uptime

10:54pm up 8 days, 3:44, 3 users,

load average: 0.89, 1.00, 2.15

htop [options]

htop (od "table of processes") - omogućava interaktivni pregled (monitoring) procesa u realnom vremenu (slično kao Task Manager u Windows OS ali mnogo detaljnije); ona je zamenila zastarelu komandu top; mora se instalirati

$ htop -d 10 (brojna vrednost unosi u desetinkama sec (ekran se osvežava na svaku sec))

$ htop -p 1,2,3,3891 (podaci samo o procesima sa zadatim PID)

$ htop -s PID (prikaz bude sortiran po koloni PID(po brojnim vrednostima PID))

free [options]

free (mora se instalirati) - ispisuje korišćenje memorije u kilobajtima

$ free

total used free shared buffers cached

Mem: 523812 491944 31868 0 67856 199276

-/+ buffers/cache: 224812 299000

Swap: 530104 0 530104

-s N - Run continuously and update the display every N seconds.

-b - Display amounts in bytes.

-m - Display amounts in megabytes.

-t - Add a totals row at the bottom.

-o - Don't display the “buffers/cache” row.

kill - Terminate a process (or send it a signal).

timeout - Kill a command that runs for too long.

nice - Invoke a program at a particular priority.

renice - Change a process’s priority as it runs.

flock - Ensure that only one copy of a process runs at the same time, using locks

kill [options] [process\_ids]

kill - proces se može ubiti (eng. to kill), bez ostavljanja mogućnosti da se nastavi; kao sinonim koristi se pojam obustaviti (eng. to terminate); to je difolt opcija; uz to, proces se može i prekinuti (eng. to interrupt), to jest zaustaviti uz mogućnost da se ponovo nastavi – neće se raditi

$ kill -l (lista svih 30 vrsta signala)

SIGTERM (15) njega proces može ignorisat

SIGKILL (9) (jaci od SIGMTERM) proces ne može ignorisati ( forsirano obustavljanje, i uvek daje rezultat)

$ kill 13243 (Ubija proces čiji je PID jednak 13243)

$ pidof emacs

8374

$ kill `pidof emacs`

$ killall emacs

timeout [options] seconds command

timeout - definiše maksimalno trajanje izvršenja nekog programa, u sekundama; navodi se zatim odgovarajuća komanda (koja pokreće taj program); ako izvršenje traje duže od definisanog limita, komanda timeout će ubiti odgovarajući proces

$ timeout 3 sleep 60 (Killed after 3 seconds)

$ timeout 3600 mplayer \*.mp3

-s signal - Send a signal other than the default (TERM). The choices are the same ones listed by kill -l.

-k seconds - If the program doesn't die after the first signal, wait this many seconds longer and send a deadly KILL signal

nice [-n level] command\_line

nice - Kada korisnik hoće da pokrene neku komandu (odnosno proces) za koju zna da će biti "skupa" (eng. expensive), to jest da će zahtevati mnogo računarskih resursa (posebno CPU vremena), on može da bude ljubazan/uviđavan (eng. nice) prema ostalim procesima (to jest korisnicima) tako što će smanjiti prioritet svog procesa;

$ nice -7 sort hugefile > outfile

$ sudo nice –7 imporant\_script

$ ps -o pid,user,args,nice

renice [-n N] [options] PID

renice - menja (naviše ili naniže) prioritet procesa (to jest, ljubaznost) u toku njegovog izvršenja, za razliku od nice koja pokreće izvršenje programa uz zadati prioritet

$ renice 5 -p 28734 (postavlja se ljubaznost na 5, za proces 28734)

$ nice -4 sleep 120 &

[1] 56961

$ renice 7 -p 56961

-p pid - Affect the given process ID. You can omi the -p and just provide a PID (renice -n 5 1328)

-u username - Affect all processes owned by the given user.

flock [options] lockfile command

flock ("file lock" = fajl za zaključavanje) - sprečava paralelno izvršenje neke dve komande, to jest programa, za koje iz nekih razloga nije dobro da se paralelno izvršavaju

$ flock -n /tmp/mylock rsync

$ flock -n /tmp/mylock sleep 60

-n - Instantly fail if another command is already running.

-w N - Fail after waiting N seconds, if another command is already running.

sleep - Wait a set numbers of seconds, doing nothing.

watch - Run a program at set intervals.

at - Schedule a job for a single,. Future time.

crontab - Schedule jobs for many future times.

sleep time\_specification

sleep - ne radi ništa, to jest "spava" (čeka) definisani vremenski interval; zadata vrednost mora biti ceo broj, a može se posle broja staviti oznaka jedinice, kao što je s (sekunde -- difolt), zatim m (minute), h (sati) i d (dani);

$ sleep 5m (Do nothing for 5 minutes)

$ sleep 10 && echo ‘Ten seconds have passed’

(10 seconds pass)

Ten seconds have passed

watch [options] command

watch - zvršava zadatu komandu periodično, a difolt je na svake 2 sekunde; rezultati se ispisuju u full-screen modu, kako bi korisnik mogao da prati nastanak eventualnih promena;

$ watch -n 60 date (izvršava komandu date na svakih 60 sekundi (^C exit))

-n seconds - Set the time between executions in seconds.

-d - Highlight differences in the output, to emphasize what has changed from one execution to the next.

-g - Exit when the command produces output that is different from the previous execution.

at [options] time\_specification

at (=u koje vreme) - pokreće zadatu komandu jednom, u zadato vreme; drugim rečima, ona zakazuje izvršenje komande u zadatom terminu; preciznije, može se zadati i sekvenca komandi.

$ at 7am next sunday

at> echo Remember to go to shopping | mail smith

at> lpr $HOME/shopping-list

at> ^D

<EOT>

job 559 at 2015-09-14 21:30

$ echo “command\_to\_be\_run” | at 09:00

$ at 09:00 -f /home/linuxize/script.sh

$ atq (("at queue") - lista zakazanih poslova)

559 2015-09-14 07:00 a smith

$ atrm 559 (("at remove")briše zakazani posao)

-f filename - Read commands from the given file instead of standard input

-c job\_number - Print the job commands to standard output

crontab [options] [file]

Pre objašnjenja komande crontab, treba objasniti nekoliko uvodnih pojmova; prvo, daemon (srpski prevod "demon" se ne koristi) je Linux program koji se izvršava (to jest proces) neprekidno i u pozadini, bez uticaja i interakcije sa korisnikom (koji ga samo eventualno podesi)

$ service naziv\_daemona start

$ service naziv\_daemona stop

$ service –status-all (lista svih daemona)

Daemon pod nazivom crond služi za obavljanje periodičnih (repetitivnih) poslova: on se "budi" na svaki minut (ostalo vreme "spava"), proveri listu ranije zakazanih poslova, pokrene izvršenje onih za koje je došlo vreme, i nastavi da "spava"

$ crontab -e (pokreće editovanje crontab fajla u difolt editoru)

$ crontab -l (ispisuje (od list) sadržaj crontab fajla na standardnom izlazu)

$ crontab -r (briše (od remove) crontab fajl)

$ crontab myfile (postavlja myfile kao crontab fajl)

-u (Za superkorisnika) - pristup crontab fajlovima ostalih korisnika

\* \* \* \* \* - Every minute

45 \* \* \* \* - 45 minutes after each hour (1:45, 2:45, etc)

45 9 \* \* \* - Every day at 9:45 am

45 9 8 \* \* - The eighth day of every month at 9:45 am

45 9 8 12 \* - Every December 8 at 9:45 am

45 9 8 dec \* - Every December 8 at 9:45 am

45 9 \* \* 6 - Every sunday at 9:45 am

45 9 \* \* sat - Every saturday at 9:45 am

45 9 \* 12 6 - Every saturday in December, at 9:45 am

Primer linije u crontab fajlu:

45 23 \* \* 6 /home/scripts/export\_dump.sh

su [options] [-] [user]

su (“switch user”) - omogućava korisniku X da se uloguje i otvori novi šel kao neki drugi korisnik Y; pošto X otkuca komandu su za kojom sledi korisničko ime korisnika Y, treba da upiše i njegovu lozinku (koju X treba da zna, izuzev ako je X superkorisnik – od njega se lozinka i ne traži), posle čega mu se otvara šel (isti kakav vidi Y kad se uloguje)

$ su -l

Password: \*\*\*\*\*\*\*\*\* (*root password*)

#

sudo [command]

sudo rm protected\_file

Password: \*\*\*\*\*\*\*\*\* (*Your own password*)

shutdown [options] time [message]

shutdown - dovodi do isključenja ili rebutovanja Linux sistema.

$ sudo shutdown -h +10 “scheduled maintenance” (pc se nakon 10min gasi sa porukom u “”)

-r - Reboot the system.

-h - Halt the system.

-k - Kidding: don't really perform a shutdown, just broadcast warning messages to all users as if the system were going down.

-c - Cancel a showdown in progress (omit the time argument)

-f - On reboot, skip the usual filesystem check performed by the fsck program

-F - On reboot, require the usual filesystem check.

systemctl [options] command [arguments]

systemctl - e višenamenska komanda koja pokreće i zaustavlja razne servise; ona je deo novog softverskog paketa koji se naziva systemd; preciznije, to je daemon koji upravlja radom celog sistema

sudo systemctl poweroff - Shut down the system

sudo systemctl reboot - Reboot the system.

sudo systemctl suspend - Suspend the system.

logname - Print your login name.

whoami - Print your current, effective username

id - Print the user ID and group membership of a user

who - List logged-in users, long output.

users - List logged-in users, short output

finger - Print information about users

last - Determine when someone last logged in.

printenv - Print your environment

logname

logname - ispisuje korisniku na ekranu korisničko ime (username) pod kojim se ulogovao

$ logname

smith

$ echo $LOGNAME (*moze i ovako*)

$ sudo logname

smith

whoami

whoami - ispisuje na ekranu efektivno korisničko ime (eng. effective username)

$ whoami

smith

$ sudo whoami

root

id [options] [username]

id - ispisuje sve navedene informacije (numeričke vrednosti), zajedno sa odgovarajućim imenima (korisničkim imenom i grupnim imenima)

$ id

uid=500(smith) gid=500(smith)

groups=500(smith),6(disk),490(src),501(cdwrite)

-u - Print the effective user ID and exit

-g - Print the effective group ID and exit.

-G - Print the IDs of all other groups to which the user belongs.

-n - Print names (for users and groups) rather than numeric IDs. Must be combined with -u, -g or -G. For example, id -Gn produces the same output as the groups command.

-r - Print login values instead of effective values. Must be combined with -u, -g or -G.

who [options] [filename]

who - ispisuje listu svih ulogovanih korisnika; ako neki korisnik koristi N šelova, njegovo ime biće ispisano N puta

$ who

smith pts/0 Sep 6 17:09 (:0)

barrett pts/1 Sep 6 17:10 (10.24.19.240)

jones pts/2 Sep 8 20:58 (192.168.13.7)

jones pts/4 Sep 3 05:11 (192.168.13.7)

Po default-u, komanda who preuzima podatke iz sistemskog fajla /var/run/utmp, ali pomoću argumenta filename može da joj se zada i neki drugi fajl za tu namenu.

-H - Print a row of headings as the first line.

–lookup - For remotely logged-in users, print the hostnames of origin.

-u - Also print each user’s idle time at his/her terminal

-m - Display information only about yourself (i.e., the user associated with the current terminal)

-q - Quick display of username only and a count of users. Much like the users command, but it adds a count

finger [options] [user[@host]]

finger - prilično slična prethodnoj (na nekim distro treba da se instalira), ispisuje informacije o svim ulogovanim korisnicima.

$ finger

Login Name Tty Idle Login Time

smith Sandy Smith :0 Sep 6 17:09

barrett Daniel Barrett :pts/1 24 Sep 6 17:10

jones Jill Jones :pts/2 Sep 8 20:58

$ finger smith

Login: smith Name: Sandy Smith

Directory: /home/smith Shell: /bin/bash

On since Sat Sep 6 17:09 (EDT) on :0

Last login Mon Sep 8 21:07 on the pts/6 from web1

No mail.

Project:

Enhance world peace

Plan:

Mistrust first impulses; they are always right

-l - Print in long format

-s - Print in short format.

-p - Dont display the project and Plan sections, which are ordinarily read from the user,s ~/.project and ~/.plan files, respectively.

$ last [options] [users] [ttys]

last - ispisuje istoriju logovanja, u obrnutom hronološkom redu (poslednji login prvi na listi); prikazuju se vremena kad se korisnik ulogovao i kad se izlogovao, kao i trajanje sesije

$ last

bob pts/3 localhost Mon Sep 8 21:07 - 21:08 (00:01)

sue pts/6 :0 Mon Sep 8 20:25 - 20:56 (00:31)

bob pts/4 myhost Mon Sep 8 21:07 still logged in

-N - Print only the latest N lines of output, where N is a positive integer.

-i - Display Ip addresses instead of hostnames.

-R - Don't display hostnames.

-x - Also display system shutdowns and changes in system runlevel (e.g., from single-user mode into multiuser mode)

-f filename - Read from some other data file than /var/run/wtmp; See the who command for more details

printenv [environment\_variables]

printenv - spisuje sve varijable okruženja (eng. environment variable) koje su šelu dostupne, i njihove vrednosti

$ printenv

HOME=/home/smith

MAIL=/var/spool/mail/smith

NAME=Sandy Smith

SHELL=/bin/bash

$ printenv HOME SHELL

/home/smith

/bin/bash

useradd - Create an account.

userdel - Delete an account.

usermod - Modify an account.

passwd - Change a password.

chfn - Change a user’s personal information

chsh - Change a user’s shell.

useradd [options] username

useradd - Kreiranje/dodavanje novog (novih) korisnika (naloga)

$ sudo useradd -d /home/smith -s /bin/bash \

-d *dir* - Set the user’s home directory to be *dir*.

-s *shell* - Set the user’s login shell to be *shell*

-u uid - Set the user’s ID to be uid. Unless you know what are you doing, omit this option and accept the default.

-c string - Set the user’s comment field (historically called the GECOS field). THis is usually the user’s full name, but it can be any string. The chfn command can also set this information.

-g *group* - Set user’s initial (default) group to *group*, which can either be a numeric groupd ID or a group name and which must already exist.

-G *group1*,*group2* Make the user a member of the additional, existing groups *group1*,*group2*

userdel -r username

userdel - briše se postojeći korisnički nalog

$ sudo userdel smith

-r - briše fajlove u korisnikovom home direktorijumu

usermod [options] username

usermod - administrator menja razna svojstva datog korisničkog naloga

$ sudo usermod -d /home/another smith

-d *dir* - Change the user’s home directory to be *dir*.

-s *shell* - Change the user’s login shell to be *shell*.

-g group - Change the user initial (default) group to group, which can either be a numeric group ID or a group name and which must already exist

-G *group1*,*group2* - Make the user a member only of the additional, existing groups *group1*,*group2* and so on. If the user previously belonged to other groups, but you don't specify them here, the user will no longer belong to them.

-L - Disable (lock) the account so the user cannot log in.

-U - Unlock the account after a lock (-L) operation.

passwd [options] [username]

passwd - služi za promenu korisničke lozinke.

$ passwd smith

$ passwd

$ sudo passwd smith

chfn [options] [username]

chfn (“change finger”) - dodaje/menja neke podatke o korisniku u njegov nalog: puno ime, brojeve telefona i slično, a koje prikazuje komanda finger

$ chfn

Password: \*\*\*\*\*\*\*\*

Name [Shawn Smith]: Shawn E. Smith

Office [100 Barton Hall]:

Office Phone [212-555-1212]: 212-555-1234

Home Phone []:

-f *name* - Change the full name to *name*.

-h *phone* - Change the home phone number to *phone.*

-p *phone* - Change the office phone number to *phone.*

-o *office* - Change the office location to *office*.

chsh [options] [username]

chsh (“change shell”) - definiše šel program koga će korisnik koristiti; ako korisnik pozove komandu bez korisničkog imena, promena će se ticati njegovog naloga

$ chsh

Changing shell for smith.

Password: \*\*\*\*\*\*\*\*

New shell [/bin/bash]: /bin/tcsh

-s shell - Specify the new shell.

-l - List all permissible shells.

groups - Print the group membership of a user.

groupadd - Create a group.

groupdel - Delete a group.

groupmod - Modify a group.

groups [names]

groups - ispisuje korisniku listu svih grupa kojima on pripada, ili kojima pripada neki drugi korisnik

$ groups

smith users

$ groups jones root

jones : jones root

root : root bin daemon sys adm disk wheel src

groupadd [options] group

groupadd - kreiranje nove grupe

$ sudo groupadd -f friends

-g gid - Specify your own numeric groupd ID instead of letting groupadd choose one.

-f - If the specified group exists already, complain and exit.

groupdel group

groupdel - brisanje postojeće grupe

$ sudo groupdel friends

$ sudo find / -group friends -print

groupmod [options] group

groupmod - menja podatke o datoj grup

$ sudo groupmod -n newname friends

-n *name* - Change the group’s name to *name*(safe).

-g *gid* - Change the group’s ID to *gid*(risky)

echo [options] strings

echo - ispisuje svoj argument na ekranu.

$ echo We are having fun

We are having fun

$ echo -e ‘a\tb’

a b

-n - Dont print a final newline character.

-e - Recognize and interpret escape characters. For example try echo ‘hello \a’ and echo -e ‘hello\a’. The first prints literally and second makes a beep

-E - Don't interpret escape characters: the opposite of -e

\a - Alert (play a beep)

\b - Backspace

\c - Dont print the final newline (same effect as -n)

\f - Form feed

\n - Line feed (newline)

\r Carriage return

\t - Horizontal tab

\v - Vertical tab

\\ - A bacls;ash

\’ - Single quote

\” - Double quote

\nnn - The character whose ASCII value is nnn in octal

printf format\_string [arguments]

printf - ("obogaćeni" echo) - ispisuje formatirani tekst

$ printf “User %s is %d years old.\n” sandy 29

User sandy is 29 years old.

%d - Decimal integer

%x - Hexadecimal integer

%f - Floating point

%s - String

seq [options] specification

seq ("sequence") - ispisuje sekvencu (aritmetički niz) celih brojeva, pogodnu za ulančavanje sa drugim programima

$ seq 3

1

2

3

$ seq 2 5

2

3

4

5

$ seq 5 -1 2

5

4

3

2

clear

clear - briše sadržaj prozora šela

cal - print a calendar

date - Print or set the date and time.

ntpdate - Set the system time using a remote timeserver.

cal [options] [month [year]]

cal - ispisuje kalendar za jedan mesec, po difoltu aktuelni/tekuć

$ cal

November 2015

Su Mon Tu We Th Fr Sa

1 2 3 4 5 6 7

8 9 10 11 12 13 14

15 16 17 18 19 20 21

22 23 24 25 26 27 28

29 30

date [options] [format]

date - ispisuje trenutni datum i vreme (po difoltu, komanda ispisuje sistemski datum i vreme u lokalnoj vremenskoj zoni (srpska zona je CET = Central European Time))

$ date

Fri Mar 18 22:32:04 EDT 2016

$ date ‘+%D’

03/18/16

$ date ‘+The time is %l:%M %p on a lovely %A in %B’

The time is 10:32 PM on a lovely Friday in March

Whole dates and times:

%c - Full date and time, 12-hour clock - Sun 28 Sep 2003, 09:01:25 PM EDT

%D - Numeric date, 2-digit year - 09/28/03

%x - Numeric date, 4-digit year - 09/28/2003

%T - Time, 24-hour clock - 21:01:25

%X - Time, 12-hour clock - 09:01:25 PM

Words:

%a - Day of week (abbreviated) - Sun

%A - Day of week (complete) - Sunday

%b - Month name (abbreviated) - Sep

%B - Month name (complete) - September

%Z - Time zone - EDT

%p - AM or PM - PM

Numbers:

%d - Day of month, leading zero - 02

%e - Day of month, leading blank - 2

%m - Month number, leading zero - 09

%y - Year, 2 digits - 03

%Y - Year, 4 digits - 2003

%M - Minute, leading zero - 09

%S - Seconds, leading zero - 05

%l - Hour, 12-hour clock, leading blank - 9

%k - Hour, 24 hour clock, leading blank - 9

%H - Hour, 24-hour clock, leading blank - 9

%s - Seconds since the beginning of linux time: - 1068583983

Midnight January1, 1970

-d *string* - Display the given date or time *string*, formatted as you wish

-r - filename - Display the last-modified timestamp of the given file, formatted as you wish.

-s *string* - Set the system date and/or time to be *string*. Only the superuser can do this.

ntpdate timeserver

ntpdate - postavlja aktuelno sistemsko vreme i datum, pošto kontaktira i dobije informaciju od nekog vremenskog servera (eng. timeserver) -- ovo može da uradi samo superkorisnik

$ sudo /sr/sbin/ntpdate timeserver .someplace.edu

7 Sep 21:01:25 ntpdate[2399]: step time server

178.99.1.8

offset 0.51 sec

$ cat /etc/os-release - Korisnik može da proveri koju Linux distribuciju (i verziju) koristi

dnf [options] [packages]

dnf - najnovija komanda za RPM instalacione pakete

Search for a package that meets your needs (supports wildcards \* and ?) - dnf search command\_name

Check if a package is installed - dnf list installed package\_name

Download a package but don't install it - dnf download package\_name

Download and install a package - sudo dnf install package\_name

Install a package file - sudo dnf install file.rpm

Learn about a package - dnf info package\_name

List the contents of a package - rpm -ql package\_name

Discover which package an installed file belongs to - dnf provides /path/to/file

Update an install package - sudo dnf upgrade package\_name

Remove an installed package - sudo dnf remove package\_name

List all packages install on the system - dnf list installed | less

Check for updates for all packages on the system - dnf check-update

Update all packages on the system - sudo dnf upgrade

yum [options] [packages]

yum - često korišćena komanda za RPM instalacione pakete

Search for a package that meets your needs (supports wildcards \* and ?) - yum search command\_name

Check if a package is installed - yum list installed package\_name

Download a package but don't install it. (Requires the downloadonly plugin) - sudo yum –downloadonly install package\_name

Download and install a package - sudo yum install package\_name

Install a package file - rpm -vh package.rpm

Learn about a package - yum info package\_name

List the contents of a package - rpm -ql package\_name

Discover which package an installed file belongs to - yum provides /path/to/file

Update an install package - sudo yum update package\_name

Remove an installed package - sudo yum remove package\_name

List all packages install on the system - yum list installed | less

Check for updates for all packages on the system - yum check-update

Update all packages on the system - sudo yum upgrade

rpm [options] [files]

rpm - ne samo što instalira softver, već i proverava da li sistem ispunjava sve zahteve za instalaciju

Check if a package is installed - rpm -q package\_name

Install a package file - sudo rpm -ivh package\_file.rpm

Learn about a package - rpm -qi package\_name

List the contents of a package - rpm -ql package\_name

Discover which package an installed file belongs to - rpm -qf /path/to/file

Update an installed package - sudo rpm -Uvh package\_file.rpm

Remove an installed package - sudo rpm -e package\_name

List all packages installed on the system - rpm -qa | less

apt-get [options] packages

apt-file [options] string

apt-cache [options] packages

dpkg [options] packages

Paket komandi APT ("Advance Packaging Tool", koji se sastoji od 4 komande: apt-get, apt-file, apt-cache i dpkg) služi da instalira, briše i manipuliše DEB instalacionim paketima

Search for a package that meets your needs - apt-file search package\_name

Check if a package is installed - dpkg -s package\_name

Download a package but dont install it - apt-get -d package\_name

Download and install a package - sudo apt-get install package\_name

Install a package file - dpkg - i package\_file.deb

Learn about a package - apt-cache show package\_name

List the contents of a package - dpkg -L package\_name

Discover which package an installed file - dpkg -S /path/to/file

Update an installed package - sudo apt-get upgrade package\_name

Remove an install package - sudo apt-get remove package\_name

List all packages installed on the system - dpkg -l

Check for updates for all packages on teh system - sudo apt-get -u upgrade

Update all packages on the system (to include kernel packages, replace upgrade by dist-upgrade) - sudo apt-get upgrade

aptitude [options] [packages]

aptitude - alternativa paketu komandi APT, koja takođe radi sa DEB instalacionim paketima; korisnik često mora da u svakom slučaju koristi dpkg jer komanda aptitude ne omogućava rad sa lokalnim .deb fajlovima

Search for a package that meets your needs - aptitude search package\_name

Check if a package is installed (examine the output for “State: not installed” or “State:installed”) - aptitude show package\_name

Download a package but don't install it - aptitude download package\_name

Download and install a package - sudo aptitude install package\_name

Install a package file - dpkg - i package\_file.deb

Learn about a package - aptitude show package\_name

List the contents of a package - dpkg -L package\_name

Discover which package an installed file belongs to - dpkg -S /path/to/file

Update an installed package - sudo aptitude safe-upgrade package\_name

Remove an installed package - sudo aptitude remove package\_name

List all packages installed on the system - aptitude search ‘~i’ | less

Check for updates for all packages on the system - aptitude –simulate full-upgrade

Update all packages on the system - sudo aptitude full-upgrade

$ sudo apt install neofetch (Ubuntu)

$ sudo packman -S neofetch (Arch)

$ sudo dnf install neofetch (Fedora)

$ sudo apt-get install neofetch (Debian)

$sudo zypper install neofetch (OpenSUSE)

$ nano myscript (korisnik pokrene editor u kome će kucati skript)

#~/bin/bash

echo hello

(^O ^X) (Save i Exit)

$ cat myscript

$ chmod u+x myscript (skripti se dodaje dozvola za izvršenje)

$ myscript (start script)

$ ./myscript (Nacin 1 za pokretanje)

$ bash myscript (Nacin 2 za pokretanje)

$ MYVAR=6

$ echo $MYVAR

6

$ expr $MYVAR + 1

7

$ expr MYVAR+1

6+1

$ NUMBER=”10”

$ expr $NUMBER + 5

15

$ FILENAME=”My Document” (*Space in the name*)

$ ls $FILENAME (*Try to list it*)

ls: My: No such file or directory (*ls saw 2 arguments*)

ls: Document: No such file or directory

$ ls -l “$FILENAME” (*List it properly*)

My Document (*ls saw only 1 argument*)

$ echo “Hello world”

Hello world

$ printf “I am %d yeard old\n” `expr 20 + 20`

I am 40 years old

$ read NAME

John Smith <ENTER>

$ echo “I read the name $NAME”

I read the name John Smith

Povratne vrednosti

0: izvršenje OK

od 1 do 255: desila se neka greška u izvršenju, pri čemu sâma ta celobrojna vrednost predstavlja kôd greške (eng. error code)

$ cat myfile

My name is Sandy Smith and

I Really like Ubuntu Linux

$ grep Smith myfile

My name is Sandy Smith and (*A match was found…*)

$ echo $?

0 (*...so return code is “success”*)

$ grep aardvark myfile

$ echo $? (*No match was found…*)

1 (*...so return code if “failure”*)

test expression

test - testira/ispituje vrednosti string i numeričkih promenljivih; ona može testirati i neka svojstva fajlova, kao na primer datum kreiranja

$ test 10 -lt 5 (*Is 10 less than 5?*)

$ echo $?

1 (*No, it isn't*)

$ test -n “hello” (*Does “hello” have nonzero length?*)

$ echo $?

0 (*Yes, it does*)

Testiranje fileova:

-d *name* - File *name* is directory

-f *name* - File *name* is a regular file

-L *name* - File *name* is symbolic link

-r *name* - File *name* exists and is readable

-w name - FIle *name* exists and is writable

-x *name* - File *name* exists and is executable

-s *name* - File *name* exists and its size in nonzero

*f1* -nt *f2* - File *f1* is newer than file *f2*

*f1* -ot *f2* - File *f1* is older than file *f2*

Testiranje vrednosti string promenljivih:

*s1* = *s2* - String *s1* equals string *s2*

s1 != *s2* - String s1 does not equal string *s2*

-z *s1* - String *s1* has zero length

-n *s1* - String *s1* has nonzero length

Testiranje vrednosti celobrojnih promenljivih:

*a* -eq *b* - Integers *a* and *b* are equal

*a* -ne *b* - Integers *a* and *b* are not equal

*a* -gt *b* - Integer *a* is greater than integer *b*

*a* -ge *b* - Integer *a* is greater than or equal to integer *b*

*a* -lt *b* - Integer *a* is less than integer *b*

*a* -le *b* - Integer *a* is less than or equal to integer *b*

t1 -a t2 - And: both tests t1 and t2 are true

t1 -o t2 - Or: Either get t1 or t2 is true

!your\_test - Negate the test (i.e., your\_test is false)

\(your\_test\) - Parentheses are used for grouping, as in algebra

$ [ 10 -lt 5 ]

$ echo $?

1

$ [ -n “hello” ]

$ echo $?

0

$ [ 5 -lt 4] (*No space between 4 and ]*)

bash: [: missing ‘]’

(if-then-else-fi) sintaxa

if condition (*if return code is 0*)

then

body

fi

$ cat script-if

#!/bin/bash

FILE = myfile

$ if [ -s $FILE ]

then

echo “File $FILE exists.”

fi

(if-then-else-fi) sintaxa

if condition

then

body1

else

body2

fi

$ cat script-else

#!/bin/bash

if [ `whoami` = “root” ]

then

echo “Your are the superuser”

else

echo “You are a mere mortal”

fi

$ ./script-else

You are a mere mortal

$ sudo ./script-else

Password: \*\*\*\*\*\*\*\*\*

You are the superuser

(if-then-elif-then-...-else-fi) sintaxa

if condition1

then

body1

elif condition2

then

body2

elif …

…

else

bodyN

fi

$ cat script-elif

#!/bin/bash

echo “How old are you?”

read AGE

if [ “$AGE” -le 21 ]

then

echo “You are too young. Leave now!”

elif [ “$AGE” -ge 60 ]

then

echo “You are too old! Go away!”

else

echo “You are just the right age. Welcome!”

fi

while condition (while return code = 0)

do

body

done

$ cat script-while

#!/bin/bash

i=0

while [ $i -lt 3 ]

do

echo “$i”

i=`expr $i + 1`

done

$ ./script-shile

0

1

2

Until petlja - telo se ponavlja sve dok dati uslov ne postane ispunjen

until condition (while return code = 1)

do

body

done

$ cat script-until

#!/bin/bash

i=0

until [ $i -ge 3 ]

do

Echo “$i”

i=`expr $i + 1`

done

$ ./script-shile

0

1

2

for petlja - iterira kroz vrednosti neke liste, i za svaku odrađuje telo

(for-do-done) sintaxa

for variable in list

do

body

done

$ cat script-for

#!/bin/bash

for NAME in Tom Jane Harry

do

echo “$NAME is my friend”

done

$ ./script-for

Tom is my friend

Jane is my friend

Harry is my friend

$ cat script-for2

#!/bin/bash

for FILE in \*.docx

do

echo “$FILE is a Word file”

done

$ ./script-for2

letter.docx is a Word file

$ cat script-seq

#!/bin/bash

for NUM in $(seq 1 3)

do

echo “iteration $NUM”

done

$ ./script-seq

iteration 1

iteration 2

iteration 3

$ cat script-args

#!/bin/bash

echo “My name is $1 and i come from $2”

$ ./script-args Johnson Wisconsin

My name is Johnson and i come from Wisconsin

$ ./script-args Bob

My name is Bob and I come from

$ cat script-args2

#!/bin/bash

if [ $# -lt 2 ]

then

echo “$0 error: you must supply two arguments”

else

echo “My name is $1 and i come from $2”

fi

$ ./script-args2 Barbara

./scripts-args2 error: you must supply two arguments

$ cat script-args3

#!/bin/bash

for ARG in $@

do

echo “I found the argument $ARG”

done

$ ./script-args3 One Two Three

I found the argument One

I found the argument Two

I found the argument Three

exit - trenutno prekida izvršenje skripta, i šelu vraća povratni kôd n koji joj je zadat

(exit n) sintaxa

n=0 ako nema greške, i 1£n£255 ako je ima, a ta sâma vrednost n predstavlja kôd greške; ako ne dođe do izvršenja nijedne exit komande, povratni kôd je jednak 0

$ cat script-exit

#!/bin/bash

if [ $# -lt 2 ]

then

echo “$0 error: you must supply two arguments”

exit 11

else

echo “My name is $1 and I come from $2”

fi

exit 0

$ ./script-exit Bob

./script-exit error: you must supply two arguments

$ echo $?

11